5. Audit. Customer shall maintain accurate records as necessary to verify compliance with this Agreement. Upon request by Juniper, Customer shall furnish such records to Juniper and certify its compliance with this Agreement.

6. Confidentiality. The Parties agree that aspects of the Software and associated documentation are the confidential property of Juniper. As such, Customer shall exercise all reasonable commercial efforts to maintain the Software and associated documentation in confidence, which at a minimum includes restricting access to the Software to Customer employees and contractors having a need to use the Software for Customer's internal business purposes.

7. Ownership. Juniper and Juniper's licensors, respectively, retain ownership of all right, title, and interest (including copyright) in and to the Software, associated documentation, and all copies of the Software. Nothing in this Agreement constitutes a transfer or conveyance of any right, title, or interest in the Software or associated documentation, or a sale of the Software, associated documentation, or copies of the Software.

8. Warranty, Limitation of Liability, Disclaimer of Warranty. The warranty applicable to the Software shall be as set forth in the warranty statement that accompanies the Software (the "Warranty Statement"). Nothing in this Agreement shall give rise to any obligation to support the Software. Support services may be purchased separately. Any such support shall be governed by a separate, written support services agreement.

9. Termination. Any breach of this Agreement or failure by Customer to pay any applicable fees due shall result in automatic termination of the license granted herein. Upon such termination, Customer shall destroy or return to Juniper all copies of the Software and related documentation in Customer's possession or control.

10. Taxes. All license fees for the Software are exclusive of taxes, withholdings, duties, or levies (collectively "Taxes"). Customer shall be responsible for paying Taxes arising from the purchase of the license, or importation or use of the Software.

11. Export. Customer agrees to comply with all applicable export laws and restrictions and regulations of any United States and any applicable foreign agency or authority, and not to export or re-export the Software or any direct product thereof in violation of any such restrictions, laws or regulations, or without all necessary approvals. Customer shall be liable for any such violations. The version of the Software supplied to Customer may contain encryption or other capabilities restricting Customer's ability to export the Software without an export license.

12. Commercial Computer Software. The Software is "commercial computer software" and is provided with restricted rights. Use, duplication, or disclosure by the United States government is subject to restrictions set forth in this Agreement and as provided in DFARS 227.7201 through 227.7204-4, FAR 12.212, FAR 27.405(b)(2), FAR 52.227-19, or FAR 52.227-14(ALT III) as applicable.

13. Interface Information. To the extent required by applicable law, and at Customer's written request, Juniper shall provide Customer with the interface information needed to achieve interoperability between the Software and another independently created program, on payment of applicable fee, if any. Customer shall observe strict obligations of confidentiality with respect to such information and shall use such information in compliance with any applicable terms and conditions upon which Juniper makes such information available.

14. Third Party Software. Any licensor of Juniper whose software is embedded in the Software and any supplier of Juniper whose products or technology are embedded in (or services are accessed by) the Software shall be a third party beneficiary with respect to this Agreement, and such licensor or vendor shall have the right to enforce this Agreement in its own name as if it were Juniper. In addition, certain third party software may be provided with the Software and is subject to the accompanying license(s), if any, of its respective owner(s). To the extent portions of the Software are distributed under and subject to open source licenses obligating Juniper to make the source code for such portions publicly available (such as the GNU General Public License ("GPL") or the GNU Library General Public License ("LGPL"), Juniper will make such source code portions (including Juniper modifications, as appropriate) available upon request for a period of up to three years from the date of distribution. Such request can be made in writing to Juniper Networks, Inc., 1194 N Mathilda Ave., Sunnyvale, CA 94089, ATTN: General Counsel. You may obtain a copy of the GPL at http://www.gnu.org/licenses/gpl.html, and a copy of the LGPL at http://www.gnu.org/licenses/lgpl.html.

15. Miscellaneous. This Agreement shall be governed by the laws of the State of California without reference to its conflicts of laws principles. The provisions of the U.N. Convention for the International Sale of Goods shall not apply to this Agreement. For any disputes arising under this Agreement, the Parties hereby consent to the personal and exclusive jurisdiction of, and venue in, the state and federal courts within Santa Clara County, California. This Agreement constitutes the entire and sole agreement between Juniper and the Customer with respect to the Software, and supersedes all prior and contemporaneous agreements relating to the Software, whether oral or written (including any inconsistent terms contained in a purchase order), except that the terms of a separate written agreement executed by an authorized Juniper representative and Customer shall govern to the extent such terms are inconsistent or conflict with terms contained herein. No modification to this Agreement nor any waiver of any rights hereunder shall be effective unless expressly assented to in writing by the party to be charged. If any portion of this Agreement is held invalid, the Parties agree that such invalidity shall not affect the validity of the remainder of this Agreement. This Agreement and associated documentation has been written in the English language, and the Parties agree that the English version will govern. (For Canada: Les parties aux présents confirment leur volonté que cette convention de même que tous les documents y compris tout avis qui s'y rattaché, soient rédigés en langue anglaise. (Translation: The parties confirm that this Agreement and all related documentation is and will be in the English language)).
# Table of contents

## About this guide

<table>
<thead>
<tr>
<th>Audience</th>
<th>xi</th>
</tr>
</thead>
<tbody>
<tr>
<td>Where to find additional information</td>
<td>xi</td>
</tr>
<tr>
<td>Administrator and developer documentation</td>
<td>xi</td>
</tr>
<tr>
<td>Odyssey Access Client Documentation</td>
<td>xii</td>
</tr>
<tr>
<td>Hardware documentation</td>
<td>xii</td>
</tr>
<tr>
<td>Product downloads</td>
<td>xii</td>
</tr>
<tr>
<td>Conventions</td>
<td>xiii</td>
</tr>
<tr>
<td>Documentation</td>
<td>xiii</td>
</tr>
<tr>
<td>Release Notes</td>
<td>xiii</td>
</tr>
<tr>
<td>Web Access</td>
<td>xiv</td>
</tr>
<tr>
<td>Requesting Technical Support</td>
<td>xiv</td>
</tr>
<tr>
<td>Self-Help Online Tools and Resources</td>
<td>xiv</td>
</tr>
<tr>
<td>Opening a Case with JTAC</td>
<td>xiv</td>
</tr>
</tbody>
</table>

## Part 1

### Getting Started

#### Chapter 1

**Unified Access Control Solution Overview**

<table>
<thead>
<tr>
<th>Introducing the Unified Access Control Solution</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>UAC Components</td>
<td>3</td>
</tr>
<tr>
<td>Access Control Basics</td>
<td>5</td>
</tr>
<tr>
<td>The UAC Solution in the Network</td>
<td>5</td>
</tr>
<tr>
<td>How the UAC Solution Determines User Access and Protects Resources</td>
<td>7</td>
</tr>
<tr>
<td>How the UAC Components Work Together</td>
<td>8</td>
</tr>
</tbody>
</table>

#### Chapter 2

**Initial Configuration of the Unified Access Control Solution**

<table>
<thead>
<tr>
<th>Overview of Configuring the UAC Solution</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task Summary: Configuring the UAC Solution</td>
<td>11</td>
</tr>
<tr>
<td>How the Infranet Enforcer and Infranet Controller Work Together</td>
<td>14</td>
</tr>
<tr>
<td>Juniper Networks ScreenOS Features by Releases</td>
<td>14</td>
</tr>
<tr>
<td>Configuration Overview</td>
<td>14</td>
</tr>
<tr>
<td>Communication between the Infranet Controller and the Infranet Enforcer</td>
<td>15</td>
</tr>
<tr>
<td>Setting Up Certificates for the Infranet Controller and Infranet Enforcer</td>
<td>16</td>
</tr>
<tr>
<td>Task Summary: Setting Up Certificates for the Infranet Controller and Infranet Enforcer</td>
<td>16</td>
</tr>
<tr>
<td>Using OpenSSL to Create a CA and Sign the Server Certificate</td>
<td>18</td>
</tr>
<tr>
<td>Configuring the Infranet Controller to Connect to the Infranet Enforcer</td>
<td>20</td>
</tr>
<tr>
<td>Task Summary: Configuring the Infranet Controller and the Infranet Enforcer</td>
<td>21</td>
</tr>
</tbody>
</table>
Configuring the Infranet Enforcer ................................................................. 22
Setting Up an Infranet Enforcer in Route Mode ........................................ 23
Setting Up an Infranet Enforcer in Transparent Mode ............................... 24
Binding an interface to a Security Zone ..................................................... 26
Configuring Certificate Authority Server Settings .................................... 27
Creating an Infranet Controller Instance in Route Mode ......................... 29
Creating an Infranet Controller Instance in Transparent Mode .................. 31
Viewing the Configuration of an Infranet Controller Instance ................. 32
IPsec Enforcement on the Infranet Enforcer ........................................... 32
Source IP Enforcement on the Infranet Enforcer .................................... 33
Configuring a Captive Portal on the Infranet Enforcer ......................... 33
CLI Command Syntax for the Infranet Enforcer ...................................... 38
Deploying the UAC Solution to Users ................................................... 41
Initial Deployment User Experience ....................................................... 43
Creating an initial Configuration of Odyssey Access Client ................... 44
Using Odyssey Access Client Licenses with the Infranet Controller ........ 54
Manually installing and Configuring Odyssey Access Client for 802.1X .. 54
Configuring Agentless Access to Protected Resources .......................... 56
Task Summary: Configuring Agentless Access to Protected Resources ... 57
Configuring the Java Agent for Endpoint Access .................................... 58
Task Summary: Configuring the Java Agent ........................................ 59
Using Network and Security Manager with the Infranet Controller ......... 60
How the Infranet Controller and NSM communicate ............................. 60
Task Summary: Configuring DMI Communication for NSM ................. 62

Part 2

Access Management Framework

Chapter 3  General Access Management  67

Realm and Role Restrictions ................................................................. 68
Accessing Authentication Realms ........................................................ 68
Accessing User Roles ........................................................................ 69
Specifying Resources for User Access Control ..................................... 70
Infranet Enforcer Policies ................................................................. 71
Configuring Infranet Enforcer Resource Access Policies ......................... 72
Setting up IPsec Enforcement on the Infranet Enforcer ...................... 75
Configuring IPsec Enforcement on the Infranet Controller ................. 78
Task summary: Configuring IPsec Enforcement .................................. 80
Refreshing IPsec Policies on the Infranet Controller .......................... 82
Configuring IPsec Routing Policies .................................................... 82
Configuring IP Address Pool Policies for IPsec in a NAT Environment ... 85
Task Summary: Configuring an IP Address Pool Policy ....................... 87
Configuring Source Interface Policies ................................................. 89
Upgrading IPsec Enforcement Policies from Version 1.x of the Infranet
Controller ......................................................................................... 90
Setting Up Source IP Enforcement on the Infranet Enforcer .............. 91
Task Summary: Configuring Source IP Enforcement ........................ 92
Dynamic Auth Table Allocation ......................................................... 92
Configuring Auth Table Mapping Policies for Source IP Enforcement ... 93
Changing the Number of Auth Table Entries on the Infranet Enforcer ... 95
Introduction to Network Access Policies ............................................ 96
Host Enforcer Policies ...................................................................... 97
Policies, Rules & Restrictions, and Conditions Evaluation ............................ 101
Dynamic Policy Evaluation........................................................................... 105
Understanding Dynamic Policy Evaluation ........................................... 106
Understanding Standard Policy Evaluation ........................................... 106
Enabling Dynamic Policy Evaluation .................................................... 106
Policy Evaluation.......................................................................................... 107
Configuring Security Requirements.............................................................. 108
Specifying Source IP Access Restrictions .............................................. 108
Specifying Browser Access Restrictions ................................................ 110
Specifying Certificate Access Restrictions ............................................. 112
Specifying Password Access Restrictions .............................................. 113
Specifying Host Checker Access Restrictions ........................................ 114
Specifying Session Limits...................................................................... 114

Chapter 4 User Roles 117
User Role Evaluation.................................................................................... 118
Permissive Merge Guidelines................................................................ 119
Configuring User Roles................................................................................. 120
Configuring General Role Options.......................................................... 121
Configuring Role Restrictions................................................................ 121
Specifying Session Options................................................................... 122
Specifying UI Options........................................................................... 123
Defining Default Options for User Roles................................................ 124
Creating a Role for MAC Address Authentication.................................... 125
Configuring Access Options on a Role.......................................................... 126
Use Case: Using a Host Checker Policy with the Host Enforcer.............. 128

Chapter 5 The UAC RADIUS Server and Layer 2 Access 131
Configuring the Infranet Controller as a RADIUS Server............................... 132
Authentication Protocols on the Infranet Controller.................................... 133
Configuring Authentication Protocol Sets.............................................. 135
Using RADIUS Proxy............................................................................. 138
Time Limits on RADIUS Authentication and Accounting........................... 141
Using the Infranet Controller for 802.1X Network Access ............................ 142
Task summary: Configuring the Infranet Controller as a RADIUS Server for a
802.1X Network Access Device................................................. 144
Configuring Location Groups ................................................................ 145
Configuring RADIUS Clients.................................................................. 147
Configuring RADIUS Attributes Policies ................................................ 151
Using an Infranet Enforcer as a RADIUS Client of the Infranet Controller .... 159
Using a Non-Juniper 802.1X Supplicant ...................................................... 160
Configuring Access to Switches and Access Points from a Browser.............. 162
Task Summary: Configuring the Infranet Controller to Authenticate with
Non-Tunneled Protocols................................................................. 163
Controlling Unmanaged Device Access.................................................... 163
Task Summary: Configuring the Infranet Controller to Allow Unmanaged
Device Access ................................................................................ 164
Use Case: Using an External LDAP Server for MAC Address Authentication. 167
Configuring Network Access Policies for Unmanageable Devices ....... 169
Chapter 6 Authentication and Directory Servers

Task Summary: Configuring Authentication Servers ........................................ 174
Defining an Authentication Server Instance ................................................. 175
  Defining an Authentication Server Instance ........................................ 176
  Modifying an Existing Authentication Server Instance ......................... 176
  Authentication Server Limitations with Authentication Protocols ............ 176
Configuring an Anonymous Server Instance .............................................. 177
  Anonymous Server Restrictions ............................................................. 177
  Defining an Anonymous Server Instance .............................................. 178
Configuring an ACE/Server Instance ......................................................... 179
  Defining an ACE/Server Instance ......................................................... 180
  Generating an ACE/Agent Configuration File ......................................... 181
Configuring an Active Directory or NT Domain Instance ............................ 182
  Defining an Active Directory or Windows NT Domain Server Instance ..... 183
  Using Kerberos Single Sign-On with the Infranet Controller .................. 185
  Multi-Domain User Authentication ....................................................... 187
  Active Directory and NT Group Lookup Support .................................... 189
Configuring a Certificate Server Instance .................................................. 190
Configuring an LDAP Server Instance ....................................................... 191
  Defining an LDAP Server Instance ....................................................... 192
  Monitoring and Deleting Active User Sessions ................................. 195
  Enabling LDAP Password Management ............................................... 195
Configuring a Local Authentication Server Instance .................................. 200
  Defining a Local Authentication Server Instance .................................. 200
  Creating User Accounts on a Local Authentication Server .................... 202
  Managing User Accounts ................................................................. 203
  Delegating User Administration Rights to End-Users ............................. 204
Configuring an NIS Server Instance ......................................................... 205
Configuring a RADIUS Server Instance ................................................... 206
  User Experience for RADIUS Users .................................................... 207
  Configuring the Infranet Controller to Work with a Back-end RADIUS Server 208
  Enabling RADIUS Accounting ........................................................... 211
Configuring an eTrust SiteMinder Server Instance ..................................... 219
  eTrust SiteMinder Overview ............................................................... 220
  Configuring SiteMinder to Work with the Infranet Controller ................. 224
  Configuring the Infranet Controller to Work with SiteMinder ................. 230
  Debugging SiteMinder and Infranet Controller Issues .......................... 241
Configuring a MAC Address Authentication Server for Unmanageable Devices 242

Chapter 7 Authentication Realms

Creating an Authentication Realm .............................................................. 246
Defining Authentication Policies ................................................................. 248
Creating Role Mapping Rules .................................................................... 249
Specifying Role Mapping Rules for an Authentication Realm .................... 250
Customizing User Realm UI Views .............................................................. 254

Chapter 8 Sign-In Policies

Task summary: Configuring Sign-In Policies .............................................. 258
Configuring Sign-In Policies .................................................................... 259
  Defining Administrator Sign-In Policies ............................................... 259
  Authentication Realms and Protocols in User Sign-in Policies ................ 260
Defining User Sign-in Policies ............................................................... 263
Enabling and Disabling Sign-In Policies ................................................ 264
Specifying the Order in Which Sign-In Policies are Evaluated .......... 264
Configuring Sign-In pages ................................................................. 265
Configuring Standard Sign-In Pages ................................................. 266

Part 3

Endpoint Defense

Chapter 9  Host Checker

The TNC Architecture Within Host Checker.............................................. 271
Host Checker Overview............................................................................ 272
Task Summary: Configuring Host Checker............................................... 273
Creating Global Host Checker Policies................................................... 275
Creating and Configuring New Client-side Policies................................. 276
Using Host Checker for Machine Account Logins .................................. 277
Checking for Third-Party Applications Using Pre-defined Rules (Windows Only) 278
  Configuring a Predefined Antivirus Rule with Remediation Options ..... 279
  Configuring a Predefined Firewall Rule with Remediation Options ...... 281
  Configuring a Pre-defined Spyware Rule.............................................. 283
  Configuring Virus Signature Version Monitoring and Patch Assessment Data Monitoring ......................................................... 284
Specifying Customized Requirements Using Custom Rules ..................... 286
  Using a Wildcard or Environment Variable in a Host Checker Rule ...... 293
  Evaluating Multiple Rules in a Single Host Checker Policy .......... 294
  Configuring the PatchLink IMC/IMV (Windows Only)............... 295
  Configuring Patch Assessment Policies ....................................... 296
Using third-party integrity Measurement Verifiers .................................. 299
  Configuring a Remote IMV Server.................................................... 299
  Implementing the Third-Party IMV Policy ....................................... 304
Statement of Health Integration ............................................................ 305
  Enabling EAP on the Windows Machine ........................................ 309
Combining Multiple Integrity Measurement Rules with Custom Expressions 309
  Enabling Customized Server-side Policies ..................................... 310
Implementing Host Checker Policies..................................................... 311
  Executing Host Checker Policies ................................................... 312
  Configuring Host Checker Restrictions ....................................... 313
Remediating Host Checker Policies....................................................... 315
  General Host Checker Remediation User Experience ..................... 316
  Configuring General Host Checker Remediation ......................... 317
Specifying General Host Checker Options .......................................... 320
Specifying Odyssey Access Client and Host Checker Installation Options 322
  Automatically install Host Checker ............................................... 323
  Manually install Host Checker ..................................................... 324
Using Host Checker Logs........................................................................ 325
# Table of Contents

## Part 4  
**System Management**

### Chapter 10  
**General System Management**

- Configuring Network Settings ........................................ 329
  - Configuring General Network Settings ............................. 329
  - Configuring Internal and External Ports ......................... 330
  - Configuring VLANs ..................................................... 332
  - Configuring Virtual Ports ............................................ 335
  - Configuring Static Routes for Network Traffic ................ 336
  - Creating ARP Caches .................................................. 337
  - Specifying Host Names for the Infranet Controller to Resolve Locally... 337
  - Configuring the Infranet Controller to Work with a Load Balancer....... 338
- Using Central Management Features.................................. 339
  - Modifying Central Management Dashboard Graphs ............... 339
- Configuring System Utilities ........................................... 341
  - Reviewing System Data ................................................ 341
  - Upgrading or Downgrading the Infranet Controller ................ 343
  - Setting System Options .............................................. 343
  - Downloading Application Installers ................................ 344
- Configuring Infranet Controller Licensing and Security ............ 345
  - Entering or Upgrading Infranet Controller Licenses ............... 346
  - Activating and Deactivating Emergency Mode ........................ 349
  - Click the **Enable** link in the right side of the license column to activate emergency mode or click **Disable** to deactivate it.Setting Security Options .......................................................... 350
  - Installing a Juniper Software Service Package ..................... 352

### Chapter 11  
**Certificates**

- Using Device Certificates .............................................. 356
  - Importing Certificates Into the Infranet Controller ............... 357
  - Downloading a Device Certificate From the Infranet Controller ...... 359
  - Creating a Certificate Signing Request (CSR) for a New Certificate... 359
  - Using Intermediate Server Ca Certificates .......................... 360
- Using Multiple Infranet Controller Device Certificates ............... 361
- Using Trusted Client CAs .............................................. 363
  - Enabling Trusted Client CAs .......................................... 364
  - Enabling Client CA Hierarchies ...................................... 371
  - Enabling CRLs ................................................................ 372
  - Enabling OCSP ............................................................ 376
- Using Trusted Server CAs ................................................ 377
  - Uploading Trusted Server CA Certificates ............................ 379
  - Renewing a Trusted Server CA Certificate ........................... 379
  - Deleting a trusted server CA Certificate .............................. 380
  - Viewing Trusted Server CA Certificate Details ...................... 380

### Chapter 12  
**System Archiving**

- Archiving Infranet Controller Binary Configuration Files ............ 381
  - Creating Local Backups of Infranet Controller Configuration Files ... 384
  - Importing and Exporting Infranet Controller Configuration Files ...... 385
  - Exporting a System Configuration File ................................ 386
  - Importing a System Configuration File ............................... 387
  - Exporting Local User Accounts or Resource Policies .................. 387
<table>
<thead>
<tr>
<th>Chapter 13</th>
<th>Logging and Monitoring</th>
<th>415</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logging and Monitoring Overview</td>
<td>415</td>
<td></td>
</tr>
<tr>
<td>Log File Severity Levels</td>
<td>416</td>
<td></td>
</tr>
<tr>
<td>Custom Filter Log Files</td>
<td>417</td>
<td></td>
</tr>
<tr>
<td>Dynamic Log Filters</td>
<td>417</td>
<td></td>
</tr>
<tr>
<td>Viewing and Deleting User Sessions</td>
<td>418</td>
<td></td>
</tr>
<tr>
<td>Configuring the Log Monitoring Features</td>
<td>419</td>
<td></td>
</tr>
<tr>
<td>Configuring events, user access, and Admin Access Logs</td>
<td>419</td>
<td></td>
</tr>
<tr>
<td>Creating, Resetting, or Saving a Dynamic Log Query</td>
<td>420</td>
<td></td>
</tr>
<tr>
<td>Specifying Which Events to Save in the Log File</td>
<td>421</td>
<td></td>
</tr>
<tr>
<td>Creating, Editing, or Deleting Log Filters</td>
<td>422</td>
<td></td>
</tr>
<tr>
<td>Creating custom filters and Formats for Your Log Files</td>
<td>422</td>
<td></td>
</tr>
<tr>
<td>Monitoring the Infranet Controller as an SNMP Agent</td>
<td>424</td>
<td></td>
</tr>
<tr>
<td>Viewing System Statistics</td>
<td>430</td>
<td></td>
</tr>
<tr>
<td>Enabling Client-Side Logs</td>
<td>430</td>
<td></td>
</tr>
<tr>
<td>Viewing General Status</td>
<td>431</td>
<td></td>
</tr>
<tr>
<td>Viewing System Capacity Utilization</td>
<td>431</td>
<td></td>
</tr>
<tr>
<td>Specifying Time Range and Data to Display in Graphs</td>
<td>432</td>
<td></td>
</tr>
<tr>
<td>Configuring Graph Appearance</td>
<td>432</td>
<td></td>
</tr>
<tr>
<td>Downloading the Current Service Package</td>
<td>433</td>
<td></td>
</tr>
<tr>
<td>Editing the System Date and Time</td>
<td>433</td>
<td></td>
</tr>
<tr>
<td>Monitoring Active Users</td>
<td>434</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 14</th>
<th>Troubleshooting</th>
<th>437</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tracking Events</td>
<td>437</td>
<td></td>
</tr>
<tr>
<td>Tracking Events Using Policy Tracing</td>
<td>438</td>
<td></td>
</tr>
<tr>
<td>Creating Snapshots of the Infranet Controller System State</td>
<td>439</td>
<td></td>
</tr>
<tr>
<td>Creating TCP Dump Files</td>
<td>440</td>
<td></td>
</tr>
<tr>
<td>Testing Infranet Controller Network Connectivity</td>
<td>442</td>
<td></td>
</tr>
<tr>
<td>Address Resolution Protocol (ARP)</td>
<td>442</td>
<td></td>
</tr>
<tr>
<td>Ping</td>
<td>442</td>
<td></td>
</tr>
<tr>
<td>Traceroute</td>
<td>442</td>
<td></td>
</tr>
<tr>
<td>NSlookup</td>
<td>443</td>
<td></td>
</tr>
<tr>
<td>Running Debugging Tools Remotely</td>
<td>443</td>
<td></td>
</tr>
<tr>
<td>Creating Debugging Logs</td>
<td>444</td>
<td></td>
</tr>
<tr>
<td>Using the RADIUS Diagnostic Log</td>
<td>444</td>
<td></td>
</tr>
<tr>
<td>Monitoring Cluster Nodes</td>
<td>445</td>
<td></td>
</tr>
<tr>
<td>Configuring Group Communication Monitoring on a Cluster</td>
<td>446</td>
<td></td>
</tr>
<tr>
<td>Configuring Network Connectivity Monitoring on a Cluster</td>
<td>448</td>
<td></td>
</tr>
</tbody>
</table>
## Table of contents

### Chapter 15  
**Clustering**

- Task Summary: Deploying a Cluster ............................................................450
- Creating and Configuring a Cluster............................................................451
  - Defining and Initializing a Cluster .........................................................451
  - Joining an Existing Cluster ....................................................................452
- Configuring Cluster Properties ...............................................................454
  - Deploying Two Nodes in an Active/Passive Cluster ...............................455
  - Deploying Two or More Units in an Active/Active Cluster .....................457
  - Synchronizing the Cluster State ............................................................459
- Configuring Cluster Properties ..............................................................461
- Managing and Configuring Clusters ............................................................464
  - Adding Multiple Cluster Nodes ..............................................................464
  - Managing Network Settings for Cluster Nodes ......................................465
  - Upgrading Clustered Nodes .................................................................465
  - Deleting a Cluster .................................................................................465
  - Restarting or Rebooting Clustered Nodes..............................................466
- Admin Console Procedures...................................................................466
  - Troubleshooting Clusters ......................................................................468
- Configuring Cluster Properties ..............................................................461
- Serial Console Procedures........................................................................470
  - Joining an Infranet Controller to a Cluster Through Its Serial Console ...470
  - Disabling a Clustered Infranet Controller by Using Its Serial Console ....471

### Chapter 16  
**Delegating Administrator Roles**

- Creating and Configuring Administrator Roles .............................................474
  - Creating Administrator Roles ................................................................474
  - Modifying Administrator Roles .............................................................475
  - Deleting Administrator Roles ................................................................475
- Specifying Management Tasks to Delegate ................................................476
  - Delegating System Management Tasks.................................................476
  - Delegating User and Role Management .................................................476
  - Delegating User Realm Management ....................................................477
  - Delegating Administrative Management ...............................................478
  - Delegating Resource Policy Management .............................................480
- Defining General System Administrator Role Settings ..................................481
  - Defining Default Options for Administrator Roles .................................481
  - Managing General Role Settings and Options .......................................481
  - Specifying Access Management Options for the Role ...........................482
  - Specifying General Session Options ....................................................482
  - Specifying UI Options ...........................................................................483

### Chapter 17  
**Infranet Controller and IDP Interoperability**

- Licensing: IDP Availability ........................................................................486
- Deployment Scenarios .................................................................................486
- Configuring the Infranet Controller to Interoperate with IDP ....................488
  - Using ISG-IDP .......................................................................................488
  - Configuring Stand-alone IDP Connections ............................................489
  - Interaction Between the Infranet Controller and IDP ............................491
- Defining Automatic Response Sensor Event Policies ..................................491
- Identifying and Managing Quarantined Users Manually ............................493
# Part 5 System Services

## Chapter 18 Infranet Controller Serial Console
- Connecting to an Infranet Controller Appliance’s Serial Console: 497
- Rolling Back to a Previous System State: 498
  - Rolling Back to a Previous System State Through the Admin Console: 498
  - Rolling Back to a Previous System State Through the Serial Console: 499
- Resetting an Infranet Controller Appliance to the Factory Setting: 499
- Performing Common Recovery Tasks: 501

## Chapter 19 Customizable Admin and End-User UIs
- Customizable Admin Console Elements Overview: 503
- Customizable End-User Interface Elements Overview: 504

## Chapter 20 Infranet Controller 4500 and 6500
- Standard Hardware: 505
- Infranet Controller 6500 Field-Replaceable Units: 505
- Device Status LED Behavior: 506
  - Ethernet Port LED Behavior: 507
- Replacing the Cooling Fans: 507
  - Removing and Installing a Cooling Fan: 507
- Replacing a Hard Drive: 508
  - Removing and Installing a Hard Drive: 508
- Replacing IOC Modules: 509
  - Removing a Blank IOM Faceplate: 509
  - Installing an IOM: 509
  - Removing an IOM: 510
- Replacing an AC Power Supply: 510
  - Removing and Installing an AC Power Supply: 510
  - Removing and Installing a DC Power Supply: 510
- IC6000 Standard Hardware: 511
- Infranet Controller 6000 field-replaceable units: 512

## Chapter 21 Compression
- Compression Execution: 513
- Supported Data Types: 514
- Enabling Compression at the System Level: 514

# Part 6 Supplemental Information

## Appendix A Writing Custom Expressions
- Custom Expressions: 517
- Wildcard Matching: 520
- DN Variables and Functions: 521
- System Variables and Examples: 521
About this guide

This guide provides the information you need to understand, configure, and maintain the Juniper Networks Unified Access Control (UAC) solution, including:

- Getting started material to help you complete the initial configuration of the Infranet Controller and Infranet Enforcer appliances
- Overview material to familiarize yourself with the Infranet Controller, Infranet Enforcer, and Odyssey Access Client and the underlying access management system
- Instructions for configuring and managing your Infranet Controller appliance or cluster

Audience

This guide is for the system administrator responsible for configuring the following products for the UAC solution:

- Infranet Controller
- Odyssey Access Client
- Infranet Enforcer

Where to find additional information

You can obtain PDF versions of this administration guide and the other following guides are available on the Unified Access Control Product Documentation page of the Juniper Networks Customer Support Center.

You can either go to the site or you can download the guides in the following sections by clicking the title.

Administrator and developer documentation

- You can download a PDF version of this Unified Access Control Administration Guide.
For the basic tasks of configuring the Infranet Controller and Infranet Enforcer for the Unified Access Control solution, refer to the *Unified Access Control Quick Start Guide*.

For information about the changes that Odyssey Access Client and Infranet Controller clients make on client computers, including installed files and registry changes, and for information about the rights required to install and run the clients, refer to the *Unified Access Control Client-side Changes Guide*.

For information on how to personalize the look-and-feel of the pre-authentication and password management pages that the Infranet Controller displays to end-users and administrators, refer to the *Unified Access Control Custom Sign-In Pages Solution Guide*.

For information and recommendations for deploying five example scenarios of the Unified Access Control solution, refer to the *Unified Access Control Deployment Scenarios Guide*.

**Odyssey Access Client Documentation**

- For the basic tasks of configuring and using Odyssey Access Client, refer to the *Odyssey Access Client Quick Start Guide*.

- For complete information about installing, configuring, and using Odyssey Access Client, refer to the *Odyssey Access Client User Guide*.

- For information about using Odyssey Access Client Administrator, refer to the *Odyssey Access Client Administration Guide*.

You can also obtain Odyssey Access Client software and documentation on the Odyssey Access Client User page of the *Juniper Networks Customer Support Center*.

**Hardware documentation**

- For information about installing the Infranet Controller, refer to the *Installation Guide* that is included with the Infranet Controller.

- For safety information, refer to the *Juniper Networks Security Products Safety Guide*.

**Product downloads**

- To download the latest build of the Infranet Controller OS and release notes, go to the Infranet Controller OS Software page of the *Juniper Networks Customer Support Center*. 
Conventions

Table 1 defines notice icons used in this guide, and Table 2 defines text conventions used throughout the book.

### Table 1: Notice icons

<table>
<thead>
<tr>
<th>Icon</th>
<th>Meaning</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Icon" /></td>
<td>Informational note</td>
<td>Indicates important features or instructions.</td>
</tr>
<tr>
<td><img src="image" alt="Icon" /></td>
<td>Caution</td>
<td>Indicates that you may risk losing data or damaging your hardware.</td>
</tr>
<tr>
<td><img src="image" alt="Icon" /></td>
<td>Warning</td>
<td>Alerts you to the risk of personal injury.</td>
</tr>
</tbody>
</table>

### Table 2: Text conventions (except for command syntax)

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Bold typeface</strong></td>
<td>Indicates buttons, field names, dialog box names, and other user interface elements.</td>
<td>Select Enable Custom Instructions on the Host Checker Policy page.</td>
</tr>
<tr>
<td><strong>Plain sans serif typeface</strong></td>
<td>Represents: Code, commands, and keywords, URLs, file names, and directories.</td>
<td>Examples: Code: <code>certAttr.OU = 'Retail Products Group'</code> URL: Download the JRE application from: <a href="http://java.sun.com/j2se/">http://java.sun.com/j2se/</a></td>
</tr>
<tr>
<td><strong>Italics</strong></td>
<td>Identifies: Terms defined in text, Variable elements, Book names.</td>
<td>Examples: Defined term: An RDP client is a Windows component that enables a connection between a Windows server and a user’s machine. Variable element: Navigate to the <a href="#">Users &gt; User Realms &gt; Select Realm &gt; Authentication Policy &gt; Host Checker</a> page. Book name: See the <a href="#">Installation Guide</a>.</td>
</tr>
</tbody>
</table>

Documentation

**Release Notes**

Release notes are included with the product software and are available on the Web. In the Release Notes, you can find the latest information about features, changes, known problems, and resolved problems. If the information in the Release Notes differs from the information found in the documentation, follow the Release Notes.
**Web Access**

To view the documentation on the Web, go to:

http://www.juniper.net/techpubs/

**Requesting Technical Support**

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active J-Care or JNASC support contract, or are covered under warranty, and need postsales technical support, you can access our tools and resources online or open a case with JTAC.

- JTAC policies—For a complete understanding of our JTAC procedures and policies, review the JTAC User Guide located at http://www.juniper.net/customers/support/downloads/710059.pdf.
- Product warranties—For product warranty information, visit http://www.juniper.net/support/warranty/.
- JTAC hours of operation—The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.

**Self-Help Online Tools and Resources**

For quick and easy problem resolution, Juniper Networks has designed an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings—http://www.juniper.net/customers/support/
- Find product documentation—http://www.juniper.net/techpubs/
- Find solutions and answer questions using our Knowledge Base—http://kb.juniper.net/
- Download the latest versions of software and review your release notes—http://www.juniper.net/customers/csc/software/
- Search technical bulletins for relevant hardware and software notifications—http://www.juniper.net/alerts/
- Join and participate in the Juniper Networks Community Forum—http://www.juniper.net/company/communities/
- Open a case online in the CSC Case Manager—http://www.juniper.net/customers/cm/
- To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool—https://tools.juniper.net/SerialNumberEntitlementSearch/

**Opening a Case with JTAC**

You can open a case with JTAC on the Web or by telephone.
- Use the Case Manager tool in the CSC at [http://www.juniper.net/customers/cm/](http://www.juniper.net/customers/cm/).
- Call 1-888-314-JTAC (1-888-314-5822—toll free in USA, Canada, and Mexico).

For international or direct-dial options in countries without toll-free numbers, visit us at [http://www.juniper.net/customers/support/requesting-support/](http://www.juniper.net/customers/support/requesting-support/).
Part 1
Getting Started

This section provides product overview and initial configuration instructions for the Unified Access Control (UAC) solution.

Contents
- “Unified Access Control Solution Overview” on page 3
- “Initial Configuration of the Unified Access Control Solution” on page 9
Chapter 1

Unified Access Control Solution

Overview

This section provides an introduction to the Unified Access Control (UAC) solution.

Introducing the Unified Access Control Solution

The Juniper Networks UAC solution provides a mechanism for authenticating users and assessing the health of their host machines to control network access.

The UAC solution coordinates network security compliance and provides the control required to support network applications, manage network use, and reduce threats from unauthorized users and compromised host machines attempting to access the network.

You configure rules in Host Checker policies to specify the minimum criteria for the security compliance of host machines that are allowed to enter your network.

The policies that you create control access for end users, the client or agent that users access the network with, and the host machine or endpoint on which the clients run.

Policy enforcement is through Juniper firewalls (the Infranet Enforcer), 802.1X enabled switches, wireless access points, and/or packet filters configured on the endpoints.

The UAC solution can also provide access control for unmanaged devices like printers or IP phones using MAC address authentication.

UAC Components

The UAC solution consists of these Juniper Networks components:

- **Infranet Controller**—You create and save access policies on the Infranet Controller, a central policy management server that validates the user’s identity, determines the endpoint’s security compliance, and manages network policies. The Infranet Controller pushes the policies to the endpoint, and optionally the Infranet Enforcer.
- **UAC agent**—The UAC solution utilizes a UAC agent to connect with endpoints. The UAC agent is client software that runs on the endpoint and determines the endpoint’s compliance to the enterprise security policies you specify on the Infranet Controller. The UAC agent communicates frequently with the Infranet Controller to verify the endpoint’s continued compliance with the policies using the built-in Host Checker. Host Checker allows you to create policies to assess and monitor the health status of endpoints that access your network.

**NOTE:** You can also deploy the solution to endpoints with a subset of features using a non-UAC agent, for example a non-Juniper 802.1X supplicant. This introduction focuses on using a UAC agent.

You can use the following UAC agents:

- **Odyssey Access Client**—You can configure the Infranet Controller to automatically install Odyssey Access Client on supported Windows endpoints. Odyssey Access Client includes built-in components (including Host Checker) to provide maximum protection and functionality.

- **Java agent**—For Linux and Macintosh endpoints, you can install a lightweight Java agent. With the Java agent, Host Checker is downloaded automatically to assess and monitor endpoint security.

- **Host Checker (agentless)**—You can configure the Infranet Controller to automatically install Host Checker for agentless access deployments on Windows, Macintosh, Linux or Solaris endpoint platforms. You use agentless access for endpoints onto which you do not want to download Odyssey Access Client or the Java agent.

**NOTE:** In this guide and other documentation for the UAC solution, the names Odyssey Access Client, Java agent, and Host Checker (for agentless access) are used to refer to the specific type of UAC agent.

- **Enforcement points**—Devices that dynamically enforce access policies for protected resources. You can control user access with Layer 2 or Layer 3 enforcement. The following types of devices can be used as UAC enforcement points:

  - **Infranet Enforcer**—A Juniper Networks security device is an optional component that operates with the Infranet Controller to enforce access policies. The Infranet Enforcer is deployed in front of servers and resources that you want to protect, and serves as a firewall to enforce the security policies that you configure to control access to protected resources. You can use the **Infranet Enforcer** in Layer 2 or Layer 3 deployments.

  - **802.1X devices**—You can use IEEE 802.1X-enabled switches or access points with the UAC solution components to control access to your network using Layer 2 authentication. The 802.1X protocol provides port-based authenticated access to a LAN. This standard applies to wireless as well as wired networks. In a wireless network, the 802.1X authentication occurs after the client has associated to an access point using an 802.11 association method. Wired networks use the 802.1X standard without any 802.11 association.
Chapter 1: Unified Access Control Solution Overview

You can use 802.1X enabled switches or access points with or without the Infranet Enforcer as part of the solution. If you do not deploy the Infranet Enforcer, the 802.1X enabled switch or access point functions as the enforcement point. You can create different security zones by configuring VLANs on your network and assigning different roles to the appropriate VLAN.

Access Control Basics

After you have set up the network components and determined the method of Enforcement (Layer 2 or Layer 3, with or without the Infranet Enforcer) there are a few basic things you need to do to configure access control.

- Use the built-in local authentication server, or external authentication servers to populate the user database.
- Create and assign user roles to control who can access different resources and applications on the network.
- Define user realms to establish authentication domains.
- Associate the roles with appropriate realms to define your access control hierarchy.
- Define security restrictions for endpoints with Host Checker policies.

When users authenticate to a realm, they are assigned a role, based on the role mappings that you create and an endpoint health assessment. Access to specific resources is permitted only for users and devices that provide the proper credentials for the realm, are associated with the appropriate roles, and whose endpoints meet security restrictions.

If a user attempts to connect to the network from an endpoint that does not comply with the security restrictions you have defined, the user is not permitted to access the realm or role.

For added protection, you can integrate the UAC solution with Juniper Networks Intrusion Detection and Prevention (IDP) to detect internal traffic that is malicious in nature or behaving inappropriately.

The UAC Solution in the Network

The UAC solution is extremely flexible. There are numerous options for integration into your existing network.

Figure 1 illustrates an example of a deployment using 802.1X with a switch or access point for Layer 2 connectivity. Figure 2 illustrates an example of a network deployment using Layer 3. These examples take advantage of the Infranet Enforcer to protect network resources.

You can also deploy the solution without the Infranet Enforcer by using VLANs to segregate unauthenticated or unauthorized traffic. Figure 3 illustrates this example.
For initial configuration steps, see “Initial Configuration of the Unified Access Control Solution” on page 9.
How the UAC Solution Determines User Access and Protects Resources

You create policies on the Infranet Controller’s admin interface to control access to resources and services. Access is based on successful authentication, the user’s assigned role, and the security compliance of the endpoint device. For example, you can provide full access to protected resources for an employees role, and limited access for a contractors role.

You can create Host Checker policies that require endpoints to satisfy security requirements that you define on the Infranet Controller. For example, you can require an endpoint to use a minimum version of an antivirus application with up-to-date antivirus definitions. If the endpoint does not meet the security requirements, you can configure the Host Checker policy to display remediation instructions that tell the user how to bring the endpoint into compliance.

After you have populated the Infranet Controller with users, policies and authentication services, you determine how users will gain access to network resources.

The Infranet Controller and Infranet Enforcer appliances can work together to provide granular endpoint security and firewall services to control access to protected resources for qualified users.

If you are using the Infranet Enforcer, the Infranet Controller pushes policies to the Infranet Enforcer when the two devices connect.

Based on the user’s identity and endpoint status, the Infranet Controller assigns the user a set of roles. These roles specify which resources the user can access. The Infranet Controller pushes the set of roles associated with each endpoint’s source IP address (called auth table entries) to the Infranet Enforcer. The Infranet Enforcer allows traffic between the endpoint and the protected resources based on resource access policies that you create.

For 802.1X Layer 2 deployments in which you are not using the Infranet Enforcer, you can set up network VLANs and direct endpoints that do not meet security requirements to a quarantine VLAN.

The user authenticates through a switch or access point to the Infranet Controller. The user’s identity and the endpoint health assessment are used to determine which VLAN to use. The quarantine VLAN can limit access to remediation servers that provide users with instructions and possibly the software they need to bring their endpoint into compliance with security policies.
How the UAC Components Work Together

The following example summarizes how the Infranet Controller, the Odyssey Access Client, and the Infranet Enforcer work together when a user connects and signs in:

1. The endpoint running Odyssey Access Client communicates with the Infranet Controller by means of Extensible Authentication Protocol (EAP) messages. EAP is similar to Point-to-Point Protocol (PPP) in that it provides a transport mechanism for authentication details. The EAP messages contain information about user credentials and the health of the endpoint.

   - In 802.1X deployments, Odyssey Access Client on the endpoint connects to an 802.1X network access device. Odyssey Access Client and the Infranet Controller exchange EAP messages by means of 802.1X and RADIUS through the wired switch or wireless access point (see Figure 1 on page 6).
   - In non-802.1X deployments, the endpoint and the Infranet Controller exchange EAP messages over HTTP (see Figure 2 on page 6).

2. The user signs in to the Infranet Controller. For more information about how end-users sign in, see “Deploying the UAC Solution to Users” on page 41.

3. On Windows, Odyssey Access Client checks the endpoint for security compliance with Host Checker policies. You can implement Host Checker policies to check the endpoint before and/or after the user signs in.

   For Java agent and agentless access configurations, (see “Configuring Agentless Access to Protected Resources” on page 56), the Infranet Controller automatically installs and runs Host Checker on the endpoint to check for security compliance (if configured to do so).

4. If the user is authenticated, the Infranet Controller directs the Infranet Enforcer by means of authentication table entries to allow the user on the validated endpoint access to the specified resources.

5. The user is permitted to access the specified resources.

6. The Host Checker component in Odyssey Access Client periodically checks the status of the endpoint and reports the status to the Infranet Controller. If the endpoint becomes non-compliant, or the user disconnects or signs off, the Infranet Controller informs the Infranet Enforcer by means of authentication table entries to deny the endpoint access to resources.
Chapter 2
Initial Configuration of the Unified Access Control Solution

There are several ways to deploy the Infranet Controller to provide access control for your network assets. You can use Layer 2 or Layer 3 authentication with the Infranet Enforcer, or you can use Layer 2 802.1X without the Infranet Enforcer to direct users to different VLANs.

This chapter provides a high-level overview of the steps required to configure the solution in each of these scenarios. See “The UAC Solution in the Network” on page 5 for deployment examples.

NOTE: The Juniper Networks Unified Access Control Quick Start Guide describes the basic steps for configuring the Infranet Controller and the Infranet Enforcer in an example of a server front-end deployment scenario. You can adapt the information in that guide to apply to your specific deployment.

This chapter contains the following sections:

- “Overview of Configuring the UAC Solution” on page 10
- “How the Infranet Enforcer and Infranet Controller Work Together” on page 14
- “Setting Up Certificates for the Infranet Controller and Infranet Enforcer” on page 16
- “Configuring the Infranet Controller to Connect to the Infranet Enforcer” on page 20
- “Configuring the Infranet Enforcer” on page 22
- “Deploying the UAC Solution to Users” on page 41
- “Using Network and Security Manager with the Infranet Controller” on page 60
Overview of Configuring the UAC Solution

Table 3, “Summary of Actions Required to Configure the UAC Solution” on page 10 outlines the general steps to install and configure the UAC solution. There are many variables to consider depending on your network topology and the nature of your access control needs. Use this table as a general guide.

Your access control needs are complex, and the UAC solution is versatile. Please take the time to thoroughly read the product documentation to discover all of access control options available with the UAC solution.

<table>
<thead>
<tr>
<th>Action</th>
<th>Required or Optional</th>
</tr>
</thead>
<tbody>
<tr>
<td>Install the Infranet Controller</td>
<td>Required</td>
</tr>
<tr>
<td>Upgrade and license the Infranet Controller</td>
<td>Required</td>
</tr>
<tr>
<td>Install the Infranet Enforcer</td>
<td>Or use 802.1X</td>
</tr>
<tr>
<td>Install Certificates</td>
<td>Only with Infranet Enforcer</td>
</tr>
<tr>
<td>Connect the Infranet Controller and the Infranet Enforcer</td>
<td>Only with Infranet Enforcer</td>
</tr>
<tr>
<td>Configure authentication server(s)</td>
<td>Required</td>
</tr>
<tr>
<td>Configure Roles and Realms</td>
<td>Required</td>
</tr>
<tr>
<td>Configure Odyssey Access Client options</td>
<td>Or 3rd party client</td>
</tr>
<tr>
<td>Configure Infranet Enforcer Resource Access policies</td>
<td>Only with Infranet Enforcer</td>
</tr>
<tr>
<td>Configure IPsec and/or Source IP enforcement</td>
<td>Only with Infranet Enforcer</td>
</tr>
<tr>
<td>Configure Sign-in policies, add realms and authentication protocols</td>
<td>Required</td>
</tr>
<tr>
<td>Configure third-party agent</td>
<td>Or Odyssey Access Client</td>
</tr>
<tr>
<td>Configure Host Enforcer policies</td>
<td>Optional</td>
</tr>
<tr>
<td>Configure Host Checker policies</td>
<td>Required</td>
</tr>
<tr>
<td>Configure 802.1X for Layer 2 access</td>
<td>Or use Infranet Enforcer</td>
</tr>
</tbody>
</table>

Shaded cells indicate that the action is optional, depending on your deployment scenario. For example, the Infranet Enforcer and 802.1X are options.

The Task Summary that follows will guide you through a basic configuration of the UAC Solution.
Task Summary: Configuring the UAC Solution

This section summarizes the steps required to completely configure the UAC solution. You will need more information than the steps here provide, so go to the referenced sections for complete details of each step of the process.

Table 4: Before Beginning

<table>
<thead>
<tr>
<th>Topic</th>
<th>Detail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date and Time of the Infranet Enforcer and the Infranet Controller</td>
<td>Be sure to set the date and time of the Infranet Enforcer to match the date set for the Infranet Controller. If possible, use a Network Time Protocol (NTP) server to set the date and time for both appliances.</td>
</tr>
<tr>
<td>Kerberos</td>
<td>If you configure your Infranet Controller to use Active Directory for user authentication, Windows endpoint users can automatically sign in to the Infranet Controller using the same credentials they use to access their Windows desktops. See “Using Kerberos Single Sign-On with the Infranet Controller” on page 185.</td>
</tr>
<tr>
<td>Non-Juniper supplicants</td>
<td>If you are connecting with 802.1X, and you are using a non-Juniper Networks supplicant (a non-UAC agent), the Infranet Enforcer is not supported.</td>
</tr>
</tbody>
</table>

To configure the UAC solution:

1. If you have not already done so, install the Infranet Controller. For information, see the Unified Access Control Installation Guide.

2. If you have not already done so, upgrade and license the Infranet Controller. See “Upgrading or Downgrading the Infranet Controller” on page 343, and “Entering new Infranet Controller License Keys” on page 348.

3. If you are using the Infranet Enforcer, install the device. For cabling, rack mounting, and basic configuration instructions for Infranet Enforcer platforms, see the Juniper Networks User’s Guide that shipped with the Infranet Enforcer.

4. If you are using the Infranet Enforcer, do both of the following steps to set up certificates on the Infranet Controller and Infranet Enforcer:
   a. Import a signed server certificate into the Infranet Controller.
   b. Import the certificate of the certificate authority (CA) that signed the Infranet Controller’s server certificate into the Infranet Enforcer.

   See “Setting Up Certificates for the Infranet Controller and Infranet Enforcer” on page 16.

5. If you are using the Infranet Enforcer, configure the Infranet Controller connection to the Infranet Enforcer. See “Configuring the Infranet Controller to Connect to the Infranet Enforcer” on page 20.
6. Configure user authentication and authorization on the Infranet Controller by setting up roles, authentication and authorization servers, and authentication realms:

a. Define user and administrator roles. Roles define user session parameters and Odyssey Access Client or agent/agentless options. The Infranet Controller is preconfigured with one user role (Users) and two administrator roles (Administrators and Read-Only Administrators). See “User Roles” on page 117 and “Delegating Administrator Roles” on page 473.

b. Define authentication and authorization servers. Authentication and authorization servers authenticate user credentials and determine user privileges within the system. The Infranet Controller is preconfigured with one local authentication server (System Local) to authenticate users and one local authentication server (Administrators) to authenticate administrators. You must add users to either the local authentication server or to external authentication servers. See “Defining an Authentication Server Instance” on page 175.

c. Define authentication realms. Authentication realms contain policies specifying conditions the user or administrator must meet to sign in to the Infranet Controller. For example, you can use an authentication policy to specify that users can only access the protected resources if they are signing in from a particular location. When configuring an authentication realm, you must create rules to map users to roles and specify which server (or servers) the Infranet Controller should use to authenticate and authorize realm members.

The Infranet Controller is preconfigured with one realm (Users) that maps all users authenticated through the System Local server to the “Users” role. The Infranet Controller is also preconfigured with one realm (Admin Users) that maps all users authenticated through the Administrators server to the “Administrators” role. See “Creating an Authentication Realm” on page 246.

7. Select and configure Odyssey Access Client or agent options such as timeout values and restrictions. For configuration instructions, see “Configuring Access Options on a Role” on page 126. Alternately, you can configure endpoints to connect to the Infranet Controller with agentless access, or you can configure the lightweight Java agent for access with Macintosh and Linux endpoints. With an 802.1X deployment scenario, you can also use a non-Juniper supplicant.

8. If you are using the Infranet Enforcer, configure resource access policies to specify which roles are allowed or denied access to resources. See “Configuring Infranet Enforcer Resource Access Policies” on page 72.

9. If you are using the Infranet Enforcer, do one of the following to set up source IP enforcement and/or IPsec enforcement:

- Set up source IP enforcement by configuring an infranet-auth policy on the Infranet Enforcer. Source IP enforcement allows the Infranet Enforcer to control which zones use resource access policies to allow or deny traffic. See “Setting Up Source IP Enforcement on the Infranet Enforcer” on page 91.
Set up IPsec enforcement on Windows endpoints that Odyssey Access Client supports. You can use IPsec enforcement between the endpoint and the Infranet Enforcer instead of source IP enforcement. To use IPsec, you must set up a VPN tunnel for a dialup user with IKE on the Infranet Enforcer. For configuration instructions, see “Setting up IPsec Enforcement on the Infranet Enforcer” on page 75.

10. If you are using the Infranet Controller in a Layer 2 environment without the Infranet Enforcer, configure Odyssey Access Client or a non-Juniper Networks 802.1X supplicant on endpoints. You must also configure policies to allow the Infranet Controller RADIUS server to work with your network access device. See “Using the Infranet Controller for 802.1X Network Access” on page 142.

11. (Optional) Configure Host Enforcer policies to protect Windows endpoints and enforce policies on the endpoint itself by allowing only the traffic you specify in the Host Enforcer policies for the role. While not a substitute for a firewall, Host Enforcer policies can add another layer of access control. See “Host Enforcer Policies” on page 97.

12. Create Host Checker policies and set remediation options. See “Task Summary: Configuring Host Checker” on page 273.

13. Determine at which levels within the Infranet Controller framework you want to enforce the Host Checker policies:

- To enforce Host Checker policies when the user first accesses the Infranet Controller, implement the policies at the realm level.
- To allow or deny users access to roles based on their compliance with Host Checker policies, implement the policies at the role level.
- To map users to roles based on their compliance with Host Checker policies, use custom expressions.

For configuration instructions, see “Implementing Host Checker Policies” on page 311.

14. If necessary, configure agentless access to protected resources from endpoint platforms that Odyssey Access Client does not support, including Macintosh, Linux and Solaris. See “Configuring Agentless Access to Protected Resources” on page 56.

15. If necessary, configure the Java agent for access to protected endpoints for Macintosh or Linux. See “Configuring the Java Agent for Endpoint Access” on page 58.
16. Deploy the UAC solution to end users. See “Deploying the UAC Solution to Users” on page 41.

How the Infranet Enforcer and Infranet Controller Work Together

The Infranet Enforcer and the Infranet Controller work together to establish communications and enforce security policies with Layer 2 or Layer 3 using the Odyssey Access Client. To use Layer 2 enforcement with a non-Juniper supplicant, see “The UAC RADIUS Server and Layer 2 Access” on page 131.

If you are not using an Infranet Enforcer, this material does not apply.

Juniper Networks ScreenOS Features by Releases

Depending on the release version of ScreenOS that you are using on the Infranet Enforcer, not all of the features outlined in this guide are supported. Table 5 indicates what release version of ScreenOS is required for specific features.

<table>
<thead>
<tr>
<th>Feature</th>
<th>ScreenOS Release Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Captive portal</td>
<td>5.4</td>
</tr>
<tr>
<td>Configurable auth table entries</td>
<td>6.0</td>
</tr>
<tr>
<td>Seamless cluster failover</td>
<td>6.0R2</td>
</tr>
<tr>
<td>Dynamic auth table allocation</td>
<td>6.1</td>
</tr>
<tr>
<td>Messages to authenticated but unauthorized users</td>
<td>6.2</td>
</tr>
<tr>
<td>Support for ISG-IDP</td>
<td>6.2</td>
</tr>
</tbody>
</table>

Configuration Overview

The following basic tasks must be accomplished to use the Infranet Controller with the Infranet Enforcer:

- Configure the Infranet Controller and Infranet Enforcer to communicate over an SSL connection.
- Configure user authentication and authorization by setting up user roles, authentication and authorization servers, and authentication realms on the Infranet Controller.
- Configure resource access policies on the Infranet Controller to specify which endpoints are allowed or denied access to protected resources.
- Configure traffic enforcement between each source and destination zone with ScreenOS policies using one of the following methods:
  - Source IP enforcement
  - IPsec enforcement
Communication between the Infranet Controller and the Infranet Enforcer

This section describes the communication between the Infranet Controller and the Infranet Enforcer.

- At startup, the Infranet Enforcer contacts the Infranet Controller over an SSL connection using the NetScreen Address Change Notification (NACN) protocol.

- You can configure an Infranet Enforcer to work with up to eight Infranet Controllers. The Infranet Enforcer communicates with only one Infranet Controller at a time; the other Infranet Controllers are used for fail-over within an Infranet Controller cluster. If the Infranet Enforcer cannot connect to the first Infranet Controller you added as an instance, it attempts to connect to the next one in the configuration list until a connection occurs. If the currently-connected Infranet Controller fails, the Infranet Enforcer attempts to connect to the first Infranet Controller again. The Infranet Controllers configured on an Infranet Enforcer should all be members of the same Infranet Controller cluster.

- After the Infranet Enforcer successfully establishes an NACN connection with the Infranet Controller, the Infranet Controller opens an SSH connection with the Infranet Enforcer. The Infranet Controller uses this SSH connection to push policy and user authentication information to the Infranet Enforcer.

- Internal communication between the Infranet Controller and the Infranet Enforcer differs depending on the ScreenOS release you are using. With ScreenOS releases to 6.0 or earlier, you must configure auth table mapping policies to use source IP enforcement. If you are using release 6.1 or later, you can configure the Infranet Controller to create auth table entries dynamically, and there is no need for auth table mapping policies.

- With ScreenOS release 6.0 or earlier, when the Infranet Controller authenticates a user and verifies that the user’s endpoint is compliant with endpoint security policies, the Infranet Controller pushes user-specific configuration information to the Infranet Enforcer. This user-specific configuration information includes an auth table entry for each authenticated user. An auth table entry consists of the user’s name, a set of roles, and the IP address of the wired adapter, wireless adapter, or virtual adapter in the user’s computer. With ScreenOS release 6.1 or later, you can configure the Infranet Controller to dynamically create auth table entries on the Infranet Enforcer after a specific resource is requested.

- To use source IP enforcement, you create an infranet-auth policy on the Infranet Enforcer that allows traffic from the endpoint to the protected resource. (With ScreenOS release 6.0 and earlier, you must also create auth table mapping policies to use source IP enforcement).

- To use IPsec enforcement, you set up a VPN tunnel for a dialup user with Internet Key Exchange (IKE) on the Infranet Enforcer. The VPN tunnel and infranet-auth policy enable the user’s endpoint to use an IPsec tunnel to the Infranet Enforcer and access protected resources.

- When the Infranet Enforcer detects traffic from an endpoint that matches an infranet-auth policy, it uses the endpoint’s auth table entry to determine the role(s) associated with that user.
The Infranet Enforcer then matches the destination IP address of the protected resource (from the ScreenOS policy) with a resource access policy. The Infranet Enforcer searches to find the matching role and applies the policy action.

As necessary, the Infranet Controller sends commands to the Infranet Enforcer to remove policies or auth table entries and deny access to the protected resources. This can occur, for example, when the user’s computer becomes non-compliant with endpoint security policies or loses its connection with the Infranet Controller, when you change the configuration of a user’s role, or when you disable all user accounts on the Infranet Controller in response to a security problem such as a virus on the network.

The Infranet Controller is not FIPS compatible, but the Infranet Controller can be used with an Infranet Enforcer that is in FIPS mode. If FIPS is enabled on the Infranet Enforcer, the admin console displays that FIPS is enabled on the Infranet Enforcer > Connection page.

Setting Up Certificates for the Infranet Controller and Infranet Enforcer

To allow the Infranet Enforcer to communicate with the Infranet Controller, you must do all of the following steps:

1. If you do not have one already, create a CA certificate for the Infranet Enforcer.
2. Create a certificate signing request (CSR) for an Infranet Controller server certificate, and use the CA certificate to sign the server certificate.
3. Import the server certificate into the Infranet Controller.
4. Import the CA certificate into the Infranet Enforcer.

If the server certificate or the CA certificate is missing or expired, the Infranet Enforcer does not allow communications with the Infranet Controller. Note also that the Infranet Enforcer does not accept the temporary self-signed certificate that the Infranet Controller created during initialization.

**Task Summary: Setting Up Certificates for the Infranet Controller and Infranet Enforcer**

To set up certificates for the Infranet Controller and Infranet Enforcer:

1. If you do not have a certificate authority, install and use OpenSSL to generate a CA certificate. See “Using OpenSSL to Create a CA and Sign the Server Certificate” on page 18.
2. Create a certificate signing request (CSR) for a server certificate, and then sign the certificate signing request (CSR) by using your CA or by using OpenSSL. See “Creating and Signing a CSR” on page 19.
3. Import the signed server certificate created from the CSR into the Infranet Controller by clicking **System > Configuration > Certificates > Device Certificates** from the left navigation bar of the Infranet Controller admin console.

   a. Under Certificate Signing Requests, click the Pending CSR link that corresponds to the signed certificate. The Pending Certificate Signing Request dialog box appears.

   b. Under Step 2: Import signed certificate, browse to the certificate file you received from the CA. For example:

      ```
      c:\openssl\certs\ic.crt
      ```

   c. Click **Import**.

4. By default, the signed server certificate is automatically associated with the internal port on the Infranet Controller. If you want to associate the certificate with an external or virtual port, do the following steps:

   a. Click **System > Configuration > Certificates > Device Certificates** from the left navigation bar, and then

   b. Click the link that corresponds to a certificate that you want to use. The Certificate Details dialog box appears.

   c. Under Present certificate on these ports, specify the port(s) that the Infranet Controller should associate with the certificate—you can choose internal or external ports and primary or virtual ports, but you cannot choose a port that is already associated with another certificate.

   d. Click **Save Changes**.

5. Import the certificate of the CA that signed the Infranet Controller’s server certificate into the Infranet Enforcer. See “Configuring Certificate Authority Server Settings” on page 27.

---

**NOTE:**

- You can prevent your Web browser’s security warning from appearing each time you sign into the Infranet Controller by importing the certificate of the CA that signed the Infranet Controller’s server certificate into your browser’s list of trusted root certification authorities.

- If later you import a different server certificate and CA certificate, you may need to initiate a new connection to use them. To initiate a new connection, click **Maintenance > System > Platform > Restart Services** in the Infranet Controller admin console. The Infranet Enforcer connects to the Infranet Controller and validates the new certificate.
Using OpenSSL to Create a CA and Sign the Server Certificate

If you do not have a certificate authority, use the instructions in this section to use OpenSSL on Windows to create a CA certificate and sign the CSR for the server certificate.

NOTE: You can also use OpenSSL to create a trusted root CA certificate for Odyssey Access Client certificate validation of the Infranet Controller. Use the instructions in this section to create a CA certificate and sign the CSR for the Infranet Controller’s server certificate. Then see “Validating the Infranet Controller Certificate” on page 53 for more specific information about Odyssey Access Client certificate validation.

Setting up OpenSSL

To set up OpenSSL:

1. Download and install OpenSSL from this site:

   http://www.slproweb.com/products/Win32OpenSSL.html

2. At the Windows command prompt, type the following commands:

   C: cd \openssl
   C: md certs
   C: cd certs
   C: md demoCA
   C: md demoCA\newcerts
   C: edit demoCA\index.txt

3. Press ALT+F, S keys to save the file.

4. Press the ALT+F, X keys to exit the editor.

5. At the Windows command prompt, type the following command:

   C: edit demoCA\serial

6. Type 01 in the document window.

7. Press ALT+F, S keys to save the file.

8. Press the ALT+F, X keys to exit the editor.

9. At the Windows command prompt, type the following command:

   C: set path=c:\openssl\bin;%path%

Creating a CA Key

To create a CA key, type the following command at the Windows command prompt in the c:\openssl\certs directory:

   C: openssl genrsa -out ca.key 1024

The following output appears:
Chapter 2: Initial Configuration of the Unified Access Control Solution

Creating a CA Certificate

To create a CA certificate:

1. Type the following command at the Windows command prompt in the `c:\openssl\certs` directory:

   ```
   C: openssl req -new -x509 -days 365 -key ca.key -out demoCA/cacert.pem
   ```

2. Enter the appropriate distinguished name (DN) information for the CA certificate. You can leave some fields blank by entering a period.

   For example:
   
   ```
   Country Name: US
   State or Province Name: CA
   Locality Name: Sunnyvale
   Organization Name: XYZ
   Org. Unit Name: IT
   Common Name: ic.xyz.com
   Email Address: user@xyz.com
   ```

Creating and Signing a CSR

To create and sign a certificate signing request (CSR):

1. Click the **System > Configuration > Certificates > Device Certificates** on the left navigation bar.

2. Click **New CSR**. The new Certificate Signing Request page appears.

3. Enter the required information. For example:

   ```
   Country Name: US
   State or Province Name: CA
   Locality Name: Sunnyvale
   Organization Name: XYZ
   Org. Unit Name: IT
   Common Name: ic.xyz.com
   Email Address: user@xyz.com
   ```

   **NOTE:** The organization name in the CSR must match the CA certificate’s organization name. If the organization names do not match, you cannot sign the CSR.

4. Enter random characters in the Random Data box.

5. Click **Create CSR**. The Pending Certificate Signing Request page appears.
6. Select and copy all of the text in the text box in Step 1 into a text editor, and save the text file in the certs directory as:

c:\openssl\certs\ic.csr

7. To sign the certificate, type the following command at the Windows command prompt in the c:\openssl\certs directory:

    openssl ca -in ic.csr -out ic.crt -keyfile ca.key

8. Enter Y to sign the certificate.

9. Enter Y to commit the certificate.

You are now ready to import the server certificate into the Infranet Controller and the CA certificate into the Infranet Enforcer. (See “Task Summary: Setting Up Certificates for the Infranet Controller and Infranet Enforcer” on page 16).

---

**Configuring the Infranet Controller to Connect to the Infranet Enforcer**

The Infranet Enforcer connects with the Infranet Controller over an SSH connection that uses the NetScreen Address Change Notification (NACN) protocol. To configure a connection between the two appliances, you must specify the following items on the Infranet Controller in an Infranet Enforcer connection policy:

- An NACN password
- An administrator name and password for signing into the Infranet Enforcer using SSH
- The Serial number(s) of the Infranet Enforcer(s)

The Infranet Controller uses the NACN password and serial number for a connection from the Infranet Enforcer. When the Infranet Enforcer first turns on, it sends an NACN message containing the NACN password and serial number to the Infranet Controller. The Infranet Controller uses the serial number to determine which Infranet Enforcer is attempting to connect, and the Infranet Controller uses the NACN password to authenticate the Infranet Enforcer. The Infranet Controller then begins communicating with the Infranet Enforcer using SSH.
## Task Summary: Configuring the Infranet Controller and the Infranet Enforcer

### Table 6: Before Beginning

<table>
<thead>
<tr>
<th>Topic</th>
<th>Detail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Using an Infranet Enforcer with 802.1X</td>
<td>To use the Infranet Enforcer as an 802.1X enforcement point in front of protected resources, you must first create the location group, which is added to the Location Group list. You can select the location group you have created from the list. See “Using an Infranet Enforcer as a RADIUS Client of the Infranet Controller” on page 159.</td>
</tr>
<tr>
<td>Initiating the connection</td>
<td>To initiate a connection immediately after you finish configuring the Infranet Enforcer, restart the Infranet Controller services by clicking Maintenance &gt; System &gt; Platform &gt; Restart Services from the Infranet Controller admin console.</td>
</tr>
<tr>
<td>IPSec and Source IP for endpoints</td>
<td>See “Setting up IPsec Enforcement on the Infranet Enforcer” on page 75 for information about setting up IPsec enforcement on the Infranet Enforcer. See “Setting Up Source IP Enforcement on the Infranet Enforcer” on page 91 for information on configuring source IP enforcement.</td>
</tr>
</tbody>
</table>

To configure the Infranet Controller to connect to the Infranet Enforcer:

1. From the left navigation bar in the Infranet Controller admin console, select **UAC > Infranet Enforcer > Connection**.

2. Click **New Enforcer**. The New Infranet Enforcer dialog box appears.

3. Enter the name of the Infranet Enforcer in the Name box.

4. Enter an NACN password for this Infranet Enforcer in the NACN password box. You must enter this same NACN password when configuring the Infranet Enforcer. See “Creating an Infranet Controller Instance in Route Mode” on page 29.

5. Enter the administrator name and password for signing into the Infranet Enforcer in the appropriate boxes.

6. Enter the serial number(s) of the Infranet Enforcer. You can view the serial number on the Home page of the Infranet Enforcer WebUI, or by entering the following Juniper Networks ScreenOS CLI command: `get system`

7. Select **No 802.1X** from the Location Group list if you are not using an Infranet Enforcer as an 802.1X RADIUS client of the Infranet Controller. This is the typical setting.

8. Click **Save Changes**.

When you finish configuring the Infranet Enforcer, the Infranet Enforcer attempts to connect to the Infranet Controller. If the connection is successful, a green dot appears next to the Infranet Enforcer icon under Enforcer Status in System > Status > Overview in the Infranet Controller admin console. The Infranet Enforcer IP address also appears in UAC > Infranet Enforcer > Connection. See “Configuring the Infranet Enforcer” on page 22.
Configuring the Infranet Enforcer

This section contains instructions for configuring the Infranet Enforcer. For cabling, rack mounting, and basic configuration instructions for platforms, such as the ISG 1000 or ISG 2000, refer to the User’s Guide that shipped with the system or visit the Juniper Networks Web site at www.juniper.net/techpubs/ to download the User’s Guide appropriate for the platform you are using. For overview information, see “How the Infranet Enforcer and Infranet Controller Work Together” on page 14.

This section describes how to configure an Infranet Enforcer using Route mode or Transparent mode, how to create an infranet policy, and how to troubleshoot the configuration.

**NOTE:**

- You can only use one mode to configure a Juniper Networks security device: Route or Transparent. By default, a Juniper Networks security device operates in Route mode.
- We recommend configuring the Infranet Enforcer to use SSHv2.

For more information about how to set up routing, see the “Routing” volume of the Concepts & Examples ScreenOS Reference Guide, which you can download from www.juniper.net/techpubs/software/screenos. For more information about Route or Transparent mode, see the “Fundamentals” volume of the Concepts & Examples ScreenOS Reference Guide.

To perform these tasks the administrator must have root access.

This section contains the following topics:

- “Setting Up an Infranet Enforcer in Route Mode” on page 23
- “Setting Up an Infranet Enforcer in Transparent Mode” on page 24
- “Binding an interface to a Security Zone” on page 26
- “Configuring Certificate Authority Server Settings” on page 27
- “Creating an Infranet Controller Instance in Route Mode” on page 29
- “Creating an Infranet Controller Instance in Transparent Mode” on page 31
- “Viewing the Configuration of an Infranet Controller Instance” on page 32
- “IPsec Enforcement on the Infranet Enforcer” on page 32
- “Source IP Enforcement on the Infranet Enforcer” on page 33
- “Configuring a Captive Portal on the Infranet Enforcer” on page 33
- “CLI Command Syntax for the Infranet Enforcer” on page 38
For additional topics relevant to resource access with the Infranet Enforcer see “Infranet Enforcer Policies” on page 71.

**Setting Up an Infranet Enforcer in Route Mode**

This example explains the setup with the Infranet Controller on the untrust interface side (same side as end users).

**NOTE:** It is possible to have the Infranet Controller reside on either side of the Infranet Enforcer. If the Infranet Controller resides on the trust interface side, and users come in through the untrust interface, the administrator must configure a policy (untrust to trust) on the Infranet Enforcer that allows traffic to pass between the Infranet Controller and Odyssey Access Client. Infranet Enforcer traffic from the untrust interface to the trust interface is denied by default.

To configure an Infranet Enforcer in Route mode:

1. Set up the trust interface. The trust interface connects to the protected resource(s). The untrust interface connects to the Infranet Controller. Set the following interface (that is, ethernet1/1) settings:

   - Set routing.
   - Enable manageability of the following services:
     - SSL
     - SSH
     - IP (optional)

2. Ensure that the DHCP server is disabled or enabled as desired for the deployment.

3. Import the certificate of the CA (Certificate Authority) that signed the Infranet Controller’s server certificate into the Infranet Enforcer. See “Configuring Certificate Authority Server Settings” on page 27.

**NOTE:**

- You cannot load the self-signed Infranet Controller SSL certificate into the Juniper Networks security device. For more information about certificates and certificate options, see the *Concepts & Examples ScreenOS Reference Guide: Volume 5, VPNs*.

- If you set up an NSRP cluster before importing the CA certificate into the Infranet Enforcer, the CA certificate is automatically synchronized to all Infranet Enforcers in the cluster. However, if you set up the NSRP cluster after importing the CA certificate, you must manually synchronize the certificate to the other Infranet Enforcers in the cluster by using the following CLI command:

  `exec nsrp sync pki`
The certificate of the CA that signed the Infranet Controller’s certificate must be imported on the Infranet Enforcer because the Infranet Enforcer must be able to trust the Infranet Controller during an SSL session. When you, as a user, sign into a server by means of SSL, the server displays a dialog box where you can manually choose to accept the certificate that is associated with that server. For the Infranet Enforcer to skip that manual step and automatically accept the Infranet Controller’s certificate, the Infranet Enforcer must have the certificate of the CA that signed the Infranet Controller’s certificate.

4. Create an instance of the Infranet Controller on the Juniper Networks security device.

5. Enable SSH.

6. Verify routing from the Infranet Controller to the untrust interface.

**NOTE:** Ensure that both the Infranet Enforcer and the Infranet Controller have the correct time. If possible, use a Network Time Protocol (NTP) Server to set the date and time of both appliances.

See “Creating an Infranet Controller Instance in Route Mode” on page 29 for example WebUI and CLI input.

**Setting Up an Infranet Enforcer in Transparent Mode**

In Transparent mode, the Juniper Networks security device is usually installed between a core router and an access distribution device. Services are enabled at the zone level and VLAN1 is used for management.

Transparent mode allows the following functionality:

- The device can act as a Layer-2 forwarding device, such as a bridge
- You can control traffic flow between Layer-2 security zones through policies

To configure an Infranet Enforcer in Transparent mode:

1. Set up Transparent mode using the predefined security zones, v1-trust and v1-untrust.
   
   a. Assign interfaces to v1-trust and v1-untrust as appropriate. See “Binding an interface to a Security Zone” on page 26.

   b. Configure the IP address for a source interface to establish connectivity with the Infranet Controller. You can use V1-trust, V1-untrust or V1-dmz.

   c. Configure the broadcast mechanism to flooding (default) or ARP/trace-route.

   **NOTE:** ARP/trace-route is more secure than broadcast.

   d. Enable manageability of the following services for VLAN1:
2. Set up the Juniper Networks security device zones. The protected resources can be in either zone (v1-trust or v1-untrust), as long as the protected resources are in a zone different from the endpoints.

NOTE: The Infranet Controller can also reside in either zone. If the Infranet Controller resides in a zone different from the endpoints, configure a policy that allows traffic to the endpoints through the Infranet Enforcer.

3. Import the certificate of the CA that signed the Infranet Controller’s server certificate into the Infranet Enforcer.

NOTE: Do not import the Infranet Controller SSL certificate into the Juniper Networks security device.

For more information about certificates and certificate options, see the Concepts & Examples ScreenOS Reference Guide: Volume 5, VPNs.

The certificate of the CA that signed the Infranet Controller’s certificate must be imported on the Infranet Enforcer because the Infranet Enforcer must be able to trust the Infranet Controller during an SSL session. When you, as a user, sign into a server by means of SSL, the server displays a dialog box where you can manually choose to accept the certificate that is associated with that server. For the Infranet Enforcer to skip that manual step and automatically accept the Infranet Controller’s certificate, the Infranet Enforcer must have the certificate of the CA that signed the Infranet Controller’s certificate.


5. Enable SSH.

6. Verify routing from the Infranet Controller to the V1-untrust zone.

If you want to use IPsec enforcement with an Infranet Enforcer in Transparent mode, you may need to use the Infranet Controller admin console to configure a source interface policy on the Infranet Controller. See “Configuring Source Interface Policies” on page 89.

NOTE: Ensure that both the Infranet Enforcer and the Infranet Controller have the correct time. If possible, use a Network Time Protocol (NTP) Server to set the date and time of both appliances.
**Binding an Interface to a Security Zone**

You might need to bind the physical interfaces on the Juniper Networks security device to security zones or change a binding to accommodate your deployment. Figure 4 shows a typical setup for an Integrated Security Gateway 2000 (ISG 2000).

![Diagram of a typical setup for an ISG 2000](image)

The endpoints need to reside in a security zone different from the protected resources. The trust and untrust security zones, used in this example and in Figure 4, are default zones for Route mode configurations. For Transparent mode, you need to use the v1-trust and v1-untrust security zones (not shown). The Infranet Controller (not shown) can reside in any security zone. If, however, you place the Infranet Controller in a security zone that is different from the security zone that contains an endpoint, you must set a policy allowing traffic from the endpoints to the Infranet Controller.

To view the zones available on your platform, enter the `get zone` command in the ScreenOS CLI. To configure custom zones, refer to the Concepts & Examples ScreenOS Reference Guide: Volume 2, Fundamentals.

**NOTE:** Slot numbering varies by platform, and interface numbering varies by module type. Refer to the User Guide that accompanied the device for slot and interface numbering information or visit www.juniper.net/techpubs/hardware to obtain a copy of the User Guide specific to your platform.

In Figure 4 the endpoints reside in the Untrust zone, and the protected resources reside in the Trust zone. Port numbering is dependent on chassis configuration (two to eight physical ports on each module faceplate). In this example the port names are as follows:

- Upper left-hand module has two ports: ethernet1/1 and ethernet1/2
- Lower left-hand module has two ports: ethernet2/1 and ethernet2/2
- Upper right-hand module has eight ports: ethernet3/1 through 3/8
- Lower right-hand module has eight ports: ethernet4/1 through 4/8
You can bind an interface to a security zone from either the WebUI or the CLI.

**WebUI**
Select Network > Interfaces > Edit: Select a Zone from the pull-down menu then Click **OK**.

**CLI**
set interface ethernet4/1 zone trust
set interface ethernet3/8 zone untrust
save

### Configuring Certificate Authority Server Settings

The CA server you use can be owned and operated by an independent CA or by your own organization, in which case you become your own CA. If you use an independent CA, you must contact them for the addresses of their CA and CRL servers (for obtaining certificates and certificate revocation lists), and for the information they require when submitting certificate requests. When you are your own CA, you determine this information yourself.

To import and configure a CA certificate on the Infranet Enforcer:

1. On the Infranet Enforcer WebUI, click **Objects > Certificates** from the left navigation bar.
2. Select **CA** from the Show list.
3. Click **Browse**, to find and select the CA certificate (such as c:\OpenSSL\certs\demoCA\cacert.pem), and then click **Load**.
4. Select **CA** from the Show list to display the CA certificate.
5. Click **Server Settings** next to the certificate to configure the CA certificate. You can configure the following options:

   - **X509 Certificate Path Validation Level**: Within X509 is a specification for a certificate that binds an entity's distinguished name to its public key through the use of a digital signature. Select **Full** to validate the certificate path all the way back to the root, or select **Partial** to validate it only part of the way. The CRL distribution point extension (.cdp) in an X509 certificate can be either an HTTP URL or an LDAP URL.

   - **Certificate Revocation Check Settings**:
     - **CRL**: Enables the Juniper Networks security device to use only CRL to check the certificate status.
     - **OCSP**: Enables the Juniper Networks security device to use only OCSP to check the certificate status.
None: Disables CRL certificate checking.

**NOTE:** If you are not using CRL certificate checking, be sure to disable it in the CA Server Settings dialog box.

- **Best Effort**: Enables the Juniper Networks security device use CRL to check the certificate status, but if there is no information that indicates that the certificate is revoked, accept the certificate.

**CRL Settings:**
- **URL Address**: Specifies the internal Web-based URL of the LDAP server managing your CRL.
- **LDAP Server**: Specifies the IP address or domain name of the LDAP Root CA server that manages the CRL.
- **Refresh Frequency**: Applies only to the CRL only. From the drop-down list, select whether you want to update the CRL daily, weekly, monthly, or according to the default setting (which updates the CRL shortly after the next scheduled update).

**OCSP Settings:**
- **URL Address**: Specifies the internal Web-based URL of the OCSP server.
- **Advanced Settings**: Specifies a CA with which the Juniper Networks security device verifies the OCSP response.

**SCEP Settings:**
- **RA CGI** (registration authority certificate generation information): Specifies the RA URL where the Juniper Networks security device will request a CA certificate.
- **CA CGI** (certificate authority certificate generation information): Specifies the CA URL.
- **CA IDENT**: Specifies the name of the certificate authority for purposes of certificate ownership, if necessary.
- **Challenge**: Specifies the challenge word(s) sent to you by the CA that will prove your identity to the CA.
- **Advanced Settings**: Configures Advanced SCEP settings such as polling interval and certificate authentication.

6. Click **OK** to save the settings.
Creating an Infranet Controller Instance in Route Mode

Before you begin this procedure, you must possess the certificate of the CA that signed the Infranet Controller’s server certificate or generate a self-signed CA if you do not already have an authentic CA from a trusted source. See “Setting Up Certificates for the Infranet Controller and Infranet Enforcer” on page 16.

After you achieve connectivity to the Juniper Networks security device and set interface management options, you can create an Infranet Controller instance. You need to name the instance and set these items:

- IP address or host name of the Infranet Controller
- Password that will be used when the Infranet Enforcer uses NACN to contact the Infranet Controller
- Source interface
- CA index number (ca-idx)

You can set these items in the WebUI or through the CLI. Because the Juniper Networks security device can store more than one Infranet Controller instance, you need to include the name of the Infranet Controller with each command.

You can configure an Infranet Enforcer to work with up to eight Infranet Controllers. The Infranet Enforcer only communicates with one Infranet Controller at a time; the other Infranet Controllers are used for failover within an Infranet Controller cluster. If the Infranet Enforcer cannot connect to the first Infranet Controller you added as an instance, it tries connecting to the next one in its configuration list until a connection occurs. If the currently connected Infranet Controller fails, the Infranet Enforcer attempts to connect to the first Infranet Controller again. The Infranet Controllers configured on an Infranet Enforcer should all be members of the same Infranet Controller cluster.

In the following example, you first set interface management options and disable the DCHP server option. Then you enable SSHv2 and configure an Infranet Controller instance named controller1. Next, you set the host IP address, which is the IP address of the Infranet Controller, to 10.64.12.1. The NACN password is 8fJsP37cK9a*_HiEwe. The NACN password must match the NACN password that you entered for the Infranet Controller. The source-interface is the interface that the Infranet Enforcer uses to communicate with the Infranet Controller, and the CA index number is 001. For this example, the source interface is ethernet 1/1. You can retrieve a descriptive list of CA index numbers by entering `get ssl ca-list` in the ScreenOS CLI.

---

**NOTE:** You must use the WebUI to import the CA.

For more information about certificates, see the *Concepts & Examples ScreenOS Reference Guide: Volume 5, VPNs*.

If you need to change SSH versions, you will need to use the CLI to delete SSH settings with the `delete ssh device all` command.
When using the WebUI, you do not need to fill in the Full Subject Name of IC Cert field. But, if you do fill it in, be sure to enter the entire certificate subject. For example:

CN=ic1.juniper.net,CN=14087306185,CN=06990218,OU=Software,O=Juniper,S=CA, C=US

After creating the Infranet Controller instance, you need to set the Infranet Controller with the serial number of the Infranet Enforcer. To view the serial number of the system, use the `get system` CLI command or look at the Device Information pane on the home page of the WebUI. See “Configuring the Infranet Controller to Connect to the Infranet Enforcer” on page 20.

**WebUI**

1. Select Network > Interfaces > Edit > Services from the left navigation bar to set management options.

2. Select Network > DHCP > Edit to disable the DHCP server for both interfaces (Trust and Untrust).

3. Select and load the Certificate Authority if you have not already done so.
   a. Select Objects > Certificates.
   b. Click Browse to find and select the certificate and click Load.
   c. Select CA from the show list.
   d. Click on Server Settings and make sure Check Method is set correctly for the certificate you are using.
   e. Click OK.

4. Create the Infranet Controller instance.
   a. Select Configuration > Infranet Auth > Controllers (List) > New.
   b. Type controller1 in the Infranet Controller instance box.
   c. Type IP/domain name: 10.64.12.1 in the IP/Domain Name box.
   d. For the NACN Parameters select ethernet1/1 from the Source Interface list.
   e. Type 8IJsp37cK9a*_HiEwe in the Password box.
   f. Select the CA from the Selected CA list.

**NOTE:** For information about the Redirect URL option, see “Configuring a Captive Portal on the Infranet Enforcer” on page 33.

5. Enable SSH version 2.
   a. Select Configuration > Admin > Management > Enable SSH (v2).
**CLI**

set interface ethernet1/1 manage ssl
set interface ethernet1/1 manage ssh
set interface ethernet1/1 manage ip

set interface ethernet2/1 manage ping

set interface ethernet2/1 dhcp server disable
set interface ethernet1/1 dhcp server disable

delete ssh device all
set ssh version v2
set ssh enable

set infranet controller name controller1 host-name 10.64.12.1
set infranet controller name controller1 password 8!JsP37cK9a*_HiEwe
set infranet controller name controller1 src-interface ethernet1/1
set infranet controller name controller1 ca-idx 001
save

---

**Creating an Infranet Controller Instance in Transparent Mode**

To create an Infranet Controller instance in transparent mode:

In the following example, you use the CLI to perform the following actions:

- Assign all interfaces to Layer 2 zones.
- Assign an IP address to vlan1 and set the `route` command.
- Set interface management options.
- Configure an Infranet Controller instance named `controller1`.
- Set the host IP address, which is the IP address of the Infranet Controller, to 10.64.12.1.
- Enter the NACN password. The NACN password is `8!JsP37cK9a*_HiEwe`. The NACN password must match the NACN password that you entered for the Infranet Controller.
- The source-interface, vlan1, is the interface that the Infranet Enforcer uses to communicate with the Infranet Controller; and the CA index number is 001.

You can retrieve a descriptive list of CA index numbers by entering `get ssl ca-list`.

---

**NOTE:** You must use the WebUI to import the CA.

---

**CLI**

set interface eth1 zone v1-trust
set interface eth2 zone v1-untrust
set interface vlan1 ip 10.64.12.x
set interface vlan1 route
set interface vlan1 ip manageable
unset interface vlan1 manage ping
unset interface vlan1 manage telnet
unset interface vlan1 manage snmp
unset interface vlan1 manage web
set infranet controller name controller1 host-name 10.64.12.1
set infranet controller name controller1 password 8!JsP37cK9a*_HiEwe
set infranet controller name controller1 src-interface vlan1
set infranet controller name controller1 ca-idx 0001

**Viewing the Configuration of an Infranet Controller Instance**

You can view the configuration of an Infranet Controller instance through the WebUI and the CLI, which includes the following information:

- Name of the Infranet Controller instance
- IP address or domain name of the Infranet Controller
- Port number (should always be 11122)
- Timeout (60 seconds by default)
- Source interface

The WebUI also allows you to view the NACN password and CA parameters.

**WebUI**

Select Configuration > Infranet Auth > Controllers from the left navigation bar.
Select Configuration > Infranet Auth > General Settings from the left navigation bar.

**CLI**

get infranet controller name controller1

**IPsec Enforcement on the Infranet Enforcer**

On supported Windows endpoints that use Odyssey Access Client, you can use IPsec enforcement to encrypt the traffic between an endpoint and the Infranet Enforcer. See “Setting up IPsec Enforcement on the Infranet Enforcer” on page 75.
Source IP Enforcement on the Infranet Enforcer

To use source IP enforcement, you must configure an infranet-auth policy on the Infranet Enforcer to control which zones use Infranet Enforcer resource policies to allow or deny traffic. For example, you can configure an infranet-auth policy to enforce Infranet Enforcer resource access policies on traffic from the Untrust zone to the Trust zone. See “Setting Up Source IP Enforcement on the Infranet Enforcer” on page 91.

Configuring a Captive Portal on the Infranet Enforcer

When you deploy the Infranet Controller and Infranet Enforcer, users may not know that they must first sign into the Infranet Controller for authentication and endpoint security checking before they are allowed to access a protected resource behind the Infranet Enforcer.

To help users sign into the Infranet Controller, you can configure a redirect infranet-auth policy in the Infranet Enforcer to automatically redirect HTTP traffic destined for protected resources to the Infranet Controller. When the sign-in page for the Infranet Controller appears, the user signs in and Odyssey Access Client or Host Checker checks the endpoint for compliance to security policies. This Infranet Enforcer feature is called a captive portal.
You can configure a captive portal for deployments that use either source IP enforcement or IPsec enforcement, or a combination of both enforcement methods. For more information about using a captive portal in deployments, see “Deploying the UAC Solution to Users” on page 41.

### Table 7: Before Beginning

<table>
<thead>
<tr>
<th>Topic</th>
<th>Detail</th>
</tr>
</thead>
<tbody>
<tr>
<td>ScreenOS version</td>
<td>The captive portal feature requires ScreenOS release 5.4 or later running on the Infranet Enforcer.</td>
</tr>
<tr>
<td>External Web Server</td>
<td>You can configure the Infranet Enforcer to redirect HTTP traffic to an external Web server instead of the Infranet Controller. For example, you can redirect HTTP traffic to a Web page that explains to users the requirement to sign into the Infranet Controller before they can access the protected resource. You could also include a link to the Infranet Controller on that Web page to help users sign in.</td>
</tr>
<tr>
<td>HTTP vs. HTTPS</td>
<td>The captive portal feature redirects HTTP traffic only. If the user attempts to access a protected resource by using HTTPS or a non-browser application (such as an e-mail application), the Infranet Enforcer does not redirect the user’s traffic. When using HTTPS or a non-browser application, the user must manually sign into the Infranet Controller first before attempting to access the protected resource.</td>
</tr>
<tr>
<td>HTTP Proxy</td>
<td>If there is an HTTP proxy between the endpoint and the Infranet Enforcer, the Infranet Enforcer might not redirect the HTTP traffic.</td>
</tr>
</tbody>
</table>

### Creating a Redirect Infranet-Auth Policy on the Infranet Enforcer

To configure the captive portal feature, you must create a redirect infranet-auth policy on the Infranet Enforcer.

**WebUI**

To create a redirect infranet-auth policy on the Infranet Enforcer.

1. Click **Policies** from the left navigation bar.
2. Select a source zone from the **From** list.
3. Select a destination zone from the **To** list.
4. Click **New**. The advanced Policy Settings page appears.
5. Enter the policy configuration information such as source and destination addresses. (See the Infranet Enforcer WebUI online Help for more details about the policy configuration options.)
6. Click **Advanced**.
7. Select **Authentication** and then select **Infranet-Auth**.
8. Specify a redirection option for the infranet-auth policy:
   - **No Redirect**—Select this option to disable redirection on this policy. The Infranet Enforcer does not redirect any traffic.
Redirect unauthenticated traffic—Select this option if your deployment uses source IP only or a combination of source IP and IPsec. The Infranet Enforcer redirects clear-text traffic from unauthenticated users to the currently connected Infranet Controller, or to an IP address or domain name that you specify in a redirect URL. (See “Overriding the Default Redirection Destination for Captive Portal” on page 36.)

After a user signs in to the Infranet Controller and the user’s endpoint system meets the requirements of the Infranet Controller’s security policies, the Infranet Enforcer allows the user’s clear-text traffic to pass through in source IP deployments. For IPsec deployments, the Odyssey Access Client creates a VPN tunnel between the user and the Infranet Enforcer. The Infranet Enforcer then applies the VPN policy allowing the encrypted traffic to pass through.

Redirect all traffic—Select this option if your deployment uses IPsec only. The Infranet Enforcer redirects all clear-text traffic to the currently-connected Infranet Controller, or to an IP address or domain name that you specify in a redirect URL. (See “Overriding the Default Redirection Destination for Captive Portal” on page 36.)

After a user signs in to the Infranet Controller, Odyssey Access Client creates a VPN tunnel between the user and the Infranet Enforcer. The Infranet Enforcer then applies the VPN policy allowing the user’s encrypted traffic to pass through. This option does not allow clear text traffic to pass through the Infranet Enforcer, which protects your network from IP spoofing.

9. Click OK.

10. Click OK.

**CLI**

To configure a redirect infranet-auth policy for deployments that use source IP only, or a combination of source IP and IPsec enter:

```
set policy from source-zone to dest-zone src_addr dst_addr any permit infranet-auth redirect-unauthenticated
```

To configure a redirect infranet-auth policy for deployments that use IPsec only enter:

```
set policy from source-zone to dest-zone src_addr dst_addr any permit infranet-auth redirect-all
```
Overriding the Default Redirection Destination for Captive Portal

By default, after you configure a redirect infranet-auth policy, the Infranet Enforcer redirects HTTP traffic to the currently connected Infranet Controller by using HTTPS. To perform the redirection, the Infranet Enforcer uses the IP address or domain name that you specified when you configured the Infranet Controller instance on the Infranet Enforcer. (See “Creating an Infranet Controller Instance in Route Mode” on page 29.) The format of the URL that the Infranet Enforcer uses for default redirection is:

https://<connected Infranet Controller IP or domain name>?target=%dest-url%

NOTE: The default redirect URL is not displayed in the WebUI or the CLI.

If you configured your Infranet Enforcer to work with multiple Infranet Controllers in a cluster, and the current Infranet Controller becomes disconnected, the Infranet Enforcer automatically redirects HTTP traffic to the next active Infranet Controller in its configuration list. The Infranet Enforcer redirects traffic to only one Infranet Controller at a time.

You do not need to override the default redirection destination except in these situations:

- You are using a VIP for a cluster of Infranet Controller appliances and the Infranet Enforcer is configured to connect to the Infranet Controller’s physical IP addresses.

- You want to redirect traffic to a Web server instead of the Infranet Controller.

- If, because of split DNS or IP routing restrictions at your site, the Infranet Enforcer uses a different address for the Infranet Controller than endpoints, you must specify the domain name or IP address that endpoints must use to access the Infranet Controller. For example, if your Infranet Enforcer connects to the internal port of the Infranet Controller, but endpoints connect to the external port of the Infranet Controller, you must override the default redirection destination by specifying the external port of the Infranet Controller.

NOTE: The Infranet Controller domain name that you specify when configuring the Infranet Controller instance in the Infranet Enforcer must match the Infranet Controller domain name in the Web browser certificate that is used when users sign in. Otherwise, the browser displays a certificate warning to users when they sign in.
By default, the Infranet Enforcer also encodes and forwards to the Infranet Controller the protected resource URL that the user entered. The Infranet Controller uses the protected resource URL to help users navigate to the protected resource. The manner in which the Infranet Controller uses the protected resource URL depends on whether or not the user’s endpoint is running Odyssey Access Client:

- If the user’s endpoint is not running Odyssey Access Client (that is, it is in an agentless or Java agent configuration), the Infranet Controller automatically opens a new browser window and uses HTTP to access the protected resource after the user signs in.

- If the endpoint is using Odyssey Access Client, the Infranet Controller inserts a hypertext link in the Web page that automatically opens after the user signs in. The user must then click that hypertext link to access the protected resource by means of HTTP in the same browser window.

If you need to override the default redirection destination, use the following instructions from either the WebUI or the CLI.

**WebUI**

1. Click **Configuration > Infranet Auth > Controllers > Edit** from the left navigation bar.

2. In the Redirect URL box specify the IP address or domain name of the Infranet Controller or external Web server using HTTP or HTTPS in the following format:

   ```plaintext
   https://<IP or domain name>/<url path>?target=%dest-url%
   ```

   For example, to redirect to an Infranet Controller and forward the protected resource URL, enter:

   ```plaintext
   https://abc.company.com/?target=%dest-url%
   ```

   For example, to redirect to a Web server and forward the protected resource URL enter:

   ```plaintext
   https://server1.company.com/cgi-bin/redirect.cgi?target=%dest_url%
   ```

   The Infranet Enforcer replaces the `%dest-url%` parameter with the protected resource URL, and then forwards the protected resource URL in encrypted form to the Infranet Controller.

   **NOTE:** In the Redirect URL string, you can omit the `?target=%dest-url%` string. For example:

   ```plaintext
   http://server1.company.com
   ```

   If you do not include the `?target=%dest-url%` string, the user must manually open a new browser window and enter the protected resource URL again after signing in.
**CLI**

To specify the redirect URL, enter:

```bash
set infranet controller name controller1 url "http://10.64.12.1/?target=%dest-url%"
```

To specify the redirect URL without the `?target=%dest-url%` string, enter:

```bash
set infranet controller name controller1 url http://abc.company.com
```

---

**CLI Command Syntax for the Infranet Enforcer**

This section lists the CLI command syntax and keyword descriptions for the Infranet Enforcer.

**infranet**

Use the `infranet` commands to set up a Juniper Networks security device to work in conjunction with an Infranet Controller or to configure Infranet Enforcer policies.

**Syntax**

```bash
exec
exec infranet controller { connect | disconnect | IP ip_addr keepalive }
```

---

**NOTE:** If you run an `exec infranet controller disconnect` command, the Infranet Enforcer does not attempt to automatically connect with the Infranet Controller. To reconnect, you must run an `exec infranet controller connect` command or restart the Infranet Enforcer.

---

**get**

```bash
get infranet { controller [ name string ] | enforcer }
```

**set**

```bash
set infranet
{ controller
  { contact-interval number |
    name string
  [ ca-idx number |
    cert-subj string |
    host-name string [ port number ] |
    password string |
    src-interface interface |
    timeout number
    url string
  }
  connect-interval number |
  host-name string |
  keepalive |
  name string |
  password string |
  src-interface interface |
  timeout number |
  url string
}```
Configuring the Infranet Enforcer

Chapter 2: Initial Configuration of the Unified Access Control Solution

```
} |
  timeout action { close | no-change | open }
} |
enforcer mode test |
policy command string|
}

unset
unset infranet
{
  controller
  {
    contact-interval |
    name string |
    timeout |
    url |
  } |
enforcer mode test |
policy command string
}

Keyword and Variables

<table>
<thead>
<tr>
<th>keyword</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>interface</strong></td>
<td>Specifies the name of the interface.</td>
</tr>
<tr>
<td><strong>number</strong></td>
<td>Defines the port number or number of seconds for a particular argument.</td>
</tr>
<tr>
<td><strong>string</strong></td>
<td>Specifies the name of the Infranet Enforcer, a policy command, a password, or a cert-subject.</td>
</tr>
</tbody>
</table>

**policy command**

set infranet policy command string

<table>
<thead>
<tr>
<th>policy command</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>string</strong></td>
<td>The policy command pushes the resource access policies from the Infranet Controller to the Infranet Enforcer. Use the dynamic command designator (-n) command to view the resource access policies in the Infranet Enforcer. For example:</td>
</tr>
<tr>
<td></td>
<td>set -n infranet policy command &quot;get all&quot;</td>
</tr>
</tbody>
</table>
**controller**

- `exec infranet controller { connect | disconnect }`
- `get infranet controller name string`
- `set infranet controller contact-interval number`
- `set infranet controller timeout action [ ... ]`
- `set infranet controller name string [ ... ]`
- `unset infranet controller name string [ ... ]`
- `unset infranet controller contact-interval number`
- `unset infranet controller timeout action [ ... ]`

**controller**

Defines the Infranet Controller configuration parameters:

- **connect** re-establishes a connection with the Infranet Controller.
- **disconnect** removes the connection to the currently-connected Infranet Controller.
- **contact-interval** `number` specifies how often the Infranet Enforcer is going to ping the Infranet Controller for connectivity. The default value is 10 seconds and the range is 3-300 seconds.
- **IP** `ip_addr` is the IP address of the Infranet Controller.
- **keepalive** command is issued periodically by the Infranet Controller to the Infranet Enforcer. If the Infranet Enforcer does not receive a keepalive command within a timeout period, the Infranet Enforcer considers the connection down.
- **name** `string` is the name of the Infranet Controller and must be fewer than 32 characters in length.
- **ca-idx** `number` is the number for the certificate authority (CA) certificate index.
- **cert-subj** `string` is the string subject that matches the certificate.
- **host-name** `string [ port number ]` is the host name or IP address of the Infranet Controller. The port number must be 11122.
- **password** `string` is the NACN password of the Infranet Controller.
- **src-interface** `interface` identifies the outgoing interface.
- **timeout** `number` defines the timeout limit for idle Infranet Controller links. The default timeout is 60 seconds and the range is 1-10,000 seconds.
- **url** `string` is the redirect URL (1-512 characters) to the Infranet Controller or external Web server to which you want the Infranet Enforcer to redirect HTTP traffic. If you do not specify the URL, the Infranet Enforcer uses the default redirect URL.

Use the following format for the URL within double quotes:

```
“http://<IP or domain name>/<url path>?target=%dest-url%”
```

For more information, see “Configuring a Captive Portal on the Infranet Enforcer” on page 33.

- **timeout-action** specifies what action to take when the Infranet Controller times out:
  - **open** allows existing and new session traffic as allowed by infranet policies.
  - **no-change** preserves existing connections and dynamic configuration, such as tunnels, but new sessions require authentication.
  - **close** removes existing sessions and dynamic configuration and blocks further traffic.

**enforcer**

- `set infranet enforcer mode test`
- `unset infranet enforcer mode test`
Deploying the UAC Solution to Users

When you deploy the Infranet Controller and Infranet Enforcer, users must connect to the Infranet Controller for authentication and endpoint security checking before they are allowed to access protected resources either behind the Infranet Enforcer, or on a protected VLAN.

**NOTE:** The Infranet Controller runs Host Checker endpoint assessments before users are allowed onto the network. This is accomplished inside a Trusted Network Connect (TNC) handshake. The TNC component is downloaded through a Network Control Protocol (NCP) connection at Layer 3, once an IP connection has been established.

With Layer 2 authentication, the initial authentication always fails, because without an IP address the endpoint cannot download the TNC component. You can create a special remediation role with no access restrictions to allow endpoints to authenticate and download the required components. See “Host Checker” on page 271.

To enable users to connect to the Infranet Controller, do one of the following:

- **Install Odyssey Access Client on Windows endpoints**—Odyssey Access Client provides support for user authentication and endpoint security checking on both wired and wireless networks. It also includes an 802.1X supplicant for authentication on 802.1X networks. You can preconfigure Odyssey Access Client with the appropriate settings for your environment. See “Creating an initial Configuration of Odyssey Access Client” on page 44.

  The easiest way to install Odyssey Access Client on endpoints is by having users browse to the Infranet Controller’s sign-in URL. Odyssey Access Client then automatically installs on the user’s endpoint. See “Initial Deployment User Experience” on page 43.

  You can also manually install Odyssey Access Client on endpoints by downloading the installer package from the Infranet Controller and providing the package to endpoint users.

  If you are using 802.1X network access devices that support a preconfigured VLAN that allows limited unauthenticated network access, users can browse to the Infranet Controller for installation of the 802.1X supplicant included in Odyssey Access Client.
Direct endpoints that will use the Java agent to a Web site to download the agent—If you select the Java agent for access on Macintosh or Linux platforms, the Infranet Controller automatically downloads the Java agent to the client machine after the user has authenticated. See “Configuring the Java Agent for Endpoint Access” on page 58.

Instruct users how to use agentless access—Users can employ a browser on Windows, Macintosh, Linux and Solaris endpoints to connect to the Infranet Controller for user authentication and endpoint security checking. See “Task Summary: Configuring Agentless Access to Protected Resources” on page 57.

Ensure that a non-Juniper Networks supplicant is installed—Endpoints that do not have Odyssey Access Client installed can authenticate with 802.1X through the Infranet Controller with a non-Juniper Networks supplicant. See “Using a Non-Juniper 802.1X Supplicant” on page 160.

NOTE: A non-Juniper Networks 802.1X supplicant is also termed a non-UAC agent. A non-UAC agent is generically any client that connects to the Infranet Controller using non-Juniper Networks protocols (JUAC, for example) for authentication.

You can use the following methods to help direct users in your deployment:

Redirect HTTP traffic destined for protected resources to the Infranet Controller—To help users authenticate to the Infranet Controller, you can configure the Infranet Enforcer to automatically redirect HTTP traffic destined for protected resources to the Infranet Controller. This Infranet Enforcer feature is called a captive portal. See “Configuring a Captive Portal on the Infranet Enforcer” on page 33.

Instruct users to access the Infranet Controller using a browser at the URL you specify—This is necessary if you do not configure a captive portal, or if users access protected resources by using non-HTTP methods. (The captive portal feature redirects HTTP traffic only).

Preinstall Odyssey Access Client—This is necessary if you are using 802.1X network access devices that do not allow users to connect to the Infranet Controller without Odyssey Access Client installed. This method is also necessary for users who do not have the required administrator rights on their endpoints to install Odyssey Access Client. In these cases, you must use some other method to preinstall Odyssey Access Client on these endpoints, such as SMS, remote login, or manual preinstallation. See “Manually installing and Configuring Odyssey Access Client for 802.1X” on page 54.

Install and configure a non-Juniper Networks 802.1X supplicant—If endpoints will connect using a supplicant other than Odyssey Access Client, you must ensure that the authentication protocols that the supplicant uses for authentication are compatible with the protocols on the Infranet Controller. See “Authentication Protocols on the Infranet Controller” on page 133.
Table 8 summarizes the various scenarios and methods of deployment.

<table>
<thead>
<tr>
<th>Scenarios</th>
<th>Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>Odyssey Access Client or non-Juniper Networks 802.1X supplicant</td>
<td>- Captive portal—Redirect HTTP traffic in user’s browser to Infranet Controller user sign-in URL</td>
</tr>
<tr>
<td></td>
<td>- Announcement—Instruct users to use a web browser to manually find the Infranet Controller user sign-in URL</td>
</tr>
<tr>
<td>Unauthenticated wired network access (no 802.1X authentication)</td>
<td></td>
</tr>
<tr>
<td>802.1X switches that allow unauthenticated access to Infranet Controller by using a preconfigured VLAN that allows limited network access</td>
<td>- Captive portal—Redirect HTTP traffic in user’s browser to Infranet Controller user sign-in URL</td>
</tr>
<tr>
<td></td>
<td>- Announcement—Instruct users to use a web browser to manually find the Infranet Controller user sign-in URL</td>
</tr>
<tr>
<td>802.1X switches or wireless access points that do not allow any means to access the Infranet Controller</td>
<td>- Preinstallation of Odyssey Access Client or third-party supplicant by means of SMS or remote login on endpoints</td>
</tr>
<tr>
<td>Users who do not have administrator rights on endpoint, which is required for Odyssey Access Client installation</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Agentless or Java agent (no 802.1X authentication)</td>
<td>- Captive portal—Redirect HTTP traffic in user’s browser to Infranet Controller user sign-in URL</td>
</tr>
<tr>
<td></td>
<td>- Announcement—Instruct users to use a browser to manually find the Infranet Controller user sign-in URL</td>
</tr>
</tbody>
</table>

**Initial Deployment User Experience**

The user experience during initial deployment depends on whether the user is using Odyssey Access Client, the Java agent, an agentless deployment, or a non-UAC client (third-party 802.1X supplicant). Additionally, you can preconfigure the settings for Odyssey Access Client on the Infranet Controller (which is recommended), and you can configure single sign-on for Windows endpoints.

- **Odyssey Access Client on supported Windows endpoint platforms**—The first time the user accesses the Infranet Controller using a browser, the Infranet Controller automatically installs Odyssey Access Client on the user’s computer using the Odyssey Access Client configuration settings you specify. If you enable validation of the Infranet Controller certificate, the user must allow the root CA certificate to be installed. (See “Creating an initial Configuration of Odyssey Access Client” on page 44.)

If you evaluate or enforce a Host Checker policy at the realm level, Odyssey Access Client automatically runs its built-in Host Checker on the endpoint to check it for security compliance. If the endpoint is in compliance, the user signs into the Infranet Controller using a sign-in dialog box. After the initial Odyssey Access Client installation, Odyssey Access Client automatically starts when the user signs into his computer and displays a sign-in dialog box to sign in to the Infranet Controller.
If you configure your Infranet Controller to use single sign-on by using a Windows domain controller, Windows endpoint users automatically sign in to the Infranet Controller using the same credentials they use to access their Windows desktops. The sign-in dialog box for Odyssey Access Client does not appear. For configuration instructions, see “Using Kerberos Single Sign-On with the Infranet Controller” on page 185.

- **Java agent**—If you provision a Macintosh or Linux user for access with the Java agent, a lightweight client is automatically downloaded after the user authenticates through a browser. The agent displays connection status, the IP address, and a logout mechanism. The user is not required to leave the browser window open, but if the session expires the user must reauthenticate through a browser again. See “Configuring the Java Agent for Endpoint Access” on page 58.

- **Agentless access**—If you configure agentless access for users on Windows, Macintosh, Linux, or Solaris endpoint platforms, the user always signs into the Infranet Controller directly using a browser instead of Odyssey Access Client. (See “Configuring Agentless Access to Protected Resources” on page 56.) If you evaluate or enforce a Host Checker policy at the realm level, the Infranet Controller automatically installs and runs Host Checker on the endpoint to check it for security compliance.

**NOTE:** When using agentless access, the user must leave the browser window open on the Infranet Controller sign-in page to stay signed into the Infranet Controller. If the user closes the browser window or opens a different window, the endpoint loses the connection to the Infranet Controller and the Infranet Enforcer denies the user access to protected resources.

- **Non-UAC client software (third-party 802.1X supplicant)**—Users of Non-UAC client software must pre-install a security certificate and configure authentication protocols to connect to the Infranet Controller. These clients can only connect via Layer 2, so if any restrictive Host Checker policies are configured, users will not be able to connect. You can configure a default VLAN with no Host Checker restrictions for the initial login.

**Creating an Initial Configuration of Odyssey Access Client**

You can preconfigure Odyssey Access Client with the settings necessary to connect to the Infranet Controller, and you can configure all of the settings for the client on a per-role basis. When the user first accesses the Infranet Controller using a browser, the Infranet Controller automatically installs Odyssey Access Client on the user’s computer. Each time the user accesses a resource that is protected by the Infranet Controller, the Odyssey Access Client configuration settings you specify will be used.

**NOTE:** Except for the login name in the profile, all of the other configuration settings you specify on the Infranet Controller overwrite any existing settings on the endpoint if Odyssey Access Client is already installed when the user accesses the Infranet Controller.
You can create a unique set of configuration settings for each role. For example, you can create a role for users that use wired adapters, and another role for users that use wireless adapters.

You determine whether or not Odyssey Access Client will be installed at the role level, and you configure Odyssey Access Client options through the Users > User Roles > Agent > Odyssey Access Client section of the Infranet Controller interface.

There are two tabs under Odyssey Settings in the admin console. The first tab, IC Access allows you to configure authentication and connection settings for the Odyssey Access Client. The second tab, Preconfigured Installer provides an interface that allows you to upload a preconfigured version of Odyssey Access Client that you can deploy to users when they access a role. See “Using the Preconfigured Installer for Odyssey Access Client” on page 51.

NOTE: The Odyssey Access Client is automatically configured to use the authentication protocol settings in the default 802.1X authentication protocol set, which includes Juniper Networks JUAC protocol. If you want to use different protocols for authentication, you must configure a new protocol set on the Infranet Controller, and you must configure matching settings on the Odyssey Access Client. If you alter the protocol settings on the Odyssey Access Client, the client will function only as a 802.1X supplicant for basic connectivity, and will not have any of the features of the Odyssey Access Client. This includes Host Checker, role and realm restriction enforcement and connection with an Infranet Enforcer. See “Authentication Protocols on the Infranet Controller” on page 133.

IC Access Configuration Settings

Table 9 lists settings that you can configure on the Infranet Controller and the settings that the Infranet Controller automatically configures in Odyssey Access Client for you—that is, the settings you do not need to specify as part of the configuration.

<table>
<thead>
<tr>
<th>The Infranet Controller adds these items to Odyssey Access Client:</th>
<th>You can preconfigure these settings in the Infranet Controller:</th>
<th>The Infranet Controller automatically configures these settings for you:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Profile</td>
<td>Name of profile instance in Odyssey Access Client.</td>
<td>Enables Odyssey Access Client to validate the server certificate of the Infranet Controller. (see “Validating the Infranet Controller Certificate” on page 53).</td>
</tr>
<tr>
<td></td>
<td>Login name.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Options for using the user’s Windows credentials or prompting user for login name and/or password.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Outer authentication protocol—Tunneled TLS (TTLS) or Protected EAP (PEAP).</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Personal certificate usage.</td>
<td></td>
</tr>
<tr>
<td>Infranet Controller</td>
<td>Name of Infranet Controller instance in Odyssey Access Client.</td>
<td>Server URL is sign-in URL.</td>
</tr>
<tr>
<td></td>
<td>New profile is associated with Infranet Controller connection.</td>
<td></td>
</tr>
</tbody>
</table>
To define the initial configuration of Odyssey Access Client:

1. Select Users > User Roles > New User Role from the left navigation bar of the Infranet Controller admin console.

2. Enter a name for this role in the Name box.

3. Select the check boxes for Odyssey Settings for IC Access and Odyssey Settings for Preconfigured Installer if you want to preconfigure the Odyssey Access Client.

4. Click Save Changes. The roles configuration page appears with the name you entered for this role at the top of the page.

5. Select the Agent tab. The Install Agent for this role check box is selected by default.

6. Select the Odyssey Settings tab. The IC Access configuration page appears.

7. Select an option for naming the profile and Infranet Controller instance in Odyssey Access Client:
Deploying the UAC Solution to Users

Chapter 2: Initial Configuration of the Unified Access Control Solution

- **Use Infranet Controller's host name**—Specifies the name of the profile and the Infranet Controller instance in Odyssey Access Client. If the Infranet Controller does not have a hostname configured (see “Configuring General Network Settings” on page 330), the URL for the Infranet Controller or the redirect URL from a captive portal is used instead for the name.

- **Use this name**—Specifies the name of the profile and the Infranet Controller instance in Odyssey Access Client.

8. Under Infranet Controller, select **Require connection to this Infranet Controller** to require the enforcement of Host Enforcer policies on the endpoint that apply to the user’s role. This option requires Odyssey Access Client to always attempt to connect to this Infranet Controller and prevents the user from disconnecting from this Infranet Controller. The user also cannot delete the properties of this Infranet Controller from the Odyssey Access Client configuration.

   In effect, this option forces the enforcement of any applicable Host Enforcer policies whenever the endpoint is on the network. If the endpoint is not on the network or unable to connect to the required Infranet Controller for any reason, the Host Enforcer policies are not enforced. See “Infranet Enforcer Policies” on page 71.

9. Under Profile, specify the settings you want to configure in the Odyssey Access Client profile:

   a. Under **Login name**, specify how you want to configure the Login name setting in the profile:

      - **Use qualified Windows login name (domain\user)** configures the login name with the user’s Windows domain name and user name in the format `domain name\user name`. Use this option if you are using an Active Directory authentication server that requires a domain name in addition to a user name for authentication.

      - **Use unqualified Windows login name** configures the login name with the user’s Windows user name only. Use this option for authentication servers that require a user name only for authentication.

      - **Prompt for login name using the following prompt** displays a dialog box for the user to enter a name during the initial Odyssey Access Client installation only. The login name is then configured and the user is not prompted again. You can also configure the text string used for the prompt in the dialog box.

   b. Select **Permit login using password** to enable password authentication, and then select an option for how you want Odyssey Access Client to obtain the user’s credentials to sign into the Infranet Controller:
Use Windows password enables Odyssey Access Client to automatically authenticate the user to the Infranet Controller by using the user’s Windows password. During the initial Odyssey Access Client installation, the user must enter a password once, but then the Odyssey Access Client automatically uses the Windows password after that.

Prompt for password enables Odyssey Access Client to prompt the user to enter a password when the user is authenticated the first time after startup. Odyssey Access Client reuses the user’s credentials for the duration of the Windows session. If you choose this option and if you have configured single sign-on, Odyssey Access Client does not prompt the user for the password. For more information, see “Using Kerberos Single Sign-On with the Infranet Controller” on page 185.

c. Specify whether you want to use Tunneled TLS (TTLS) or Protected EAP (PEAP) as the outer authentication protocol for traffic between Odyssey Access Client and the Infranet Controller by selecting either Use EAP-TTLS as outer authentication protocol or Use EAP-PEAP as outer authentication protocol.

If you select Use EAP-TTLS as outer authentication protocol and you want to use a client certificate as part of the EAP-TTLS authentication, select Use the user's certificate and perform inner authentication. This option uses EAP-TTLS certificate-based authentication and tunnels password credentials with inner authentication. Note that the most typical use of EAP-TTLS authentication is without a client certificate.

If you select Use EAP-PEAP as outer authentication protocol and you want to use a client certificate as part of the EAP-PEAP authentication, select Inner authentication is required.

NOTE:

- Only enable the personal client certificate option for either EAP-TTLS or EAP-PEAP to use a client certificate if you also configure a realm or role to require a client certificate on the endpoint. See “Specifying Certificate Access Restrictions” on page 112. If you enable the personal client certificate option and do not configure the realm or role certificate restriction, you will cause unnecessary restrictions on the use of this Odyssey Access Client profile.

- If you enable the personal client certificate option, the Infranet Controller automatically selects Permit login using my Certificate and Use automatic certificate selection in the Odyssey Access Client profile.

d. Enter a name in the Anonymous name box to enable users to appear to log in anonymously while passing the user's login name (called the inner identity) through an encrypted tunnel. As a result, the user’s credentials are secure from eavesdropping and the user’s inner identity is protected.
As a general rule enter anonymous in the Anonymous name box, which is the default value. In some cases, you may need to add additional text. For example, if the outer identity is used to route the user’s authentication to the proper server, you may be required to use a format such as anonymous@acme.com. If you leave the Anonymous name box blank, Odyssey Access Client passes the user’s login name (inner identity) as the outer identity.

10. (Only if you are using 802.1X enforcement) Under Adapters, specify the type of adapter(s) you want to configure in Odyssey Access Client:

- **Configure wired adapter(s)**—Odyssey Access Client configures the wired adapter on the user’s computer that is actively being used to access the Infranet Controller on an 802.1X-enabled network. If the user is accessing the Infranet Controller through a wireless adapter during Odyssey Access Client installation, then Odyssey Access Client will automatically configure a wired adapter to use for wired access to the Infranet Controller at a later time.

- **Configure wireless adapter(s)**—Odyssey Access Client configures the wireless adapter on the user’s computer that is actively being used to access the Infranet Controller on an 802.1X-enabled network. If the user is accessing the Infranet Controller through a wired adapter during Odyssey Access Client installation, then Odyssey Access Client will automatically configure a wireless adapter to use for wireless access to the Infranet Controller at a later time. Select this option only if the endpoint is connecting to the Infranet Controller by using 802.1X. If you select this option, you must also configure the network name (SSID) under Network. You may also need to configure other Network properties depending on your environment.

**NOTE:** On Windows computers, if you select **Configure wireless adapter(s)**, Windows Wireless Zero Configuration (WZC) is disabled for the wireless adapter that Odyssey Access Client configures. If, for some reason, the user removes a wireless adapter from the local Odyssey Access Client configuration, the user must enable the adapter again by opening Control Panel > Network Connections > adapter name > Properties > Wireless Networks and selecting the **Use Windows to configure my network settings** option.

11. (Only if you enabled **Configure wireless adapter**) Under Network, specify the network settings you want to configure in Odyssey Access Client for wireless adapters:

- **Network name (SSID)**—Specify the network name or SSID (service set identifier) of the wireless network to which you want Odyssey Access Client to connect. A network name can be up to 32 alphanumeric characters and is case-sensitive. You must enter the name correctly to connect successfully. For example: <MyCorpNet>
- **Association mode**—Specify the association mode you want Odyssey Access Client to use when associating to the access point hardware on your network:
  - **Open**—Connects to a network through an access point or switch that implements 802.1X authentication. Select this mode if users are not required to use shared mode or Wi-Fi Protected Access (WPA).
  - **WPA**—Connects to a network through an access point that implements WPA.
  - **WPA2**—Connects to a network through an access point that implements WPA2, the second generation of WPA that satisfies 802.11i.

- **Encryption method**—Specify the encryption method you want Odyssey Access Client to use. The available choices depend on the association mode you selected:
  - **None**—Uses 802.1X authentication without WEP keys. This option is available only if you configure access point association in open mode. This is a typical setting to use for wireless hotspots.
  - **WEP**—Uses WEP keys for data encryption. You can select this option if you selected open mode association. Select WEP encryption if the access points in your network require WEP encryption. Odyssey Access Client automatically generates the WEP keys.
  - **TKIP**—Uses the temporal key integrity protocol. Select TKIP if the access points in your network require WPA or WPA2 association and are configured for TKIP data encryption.
  - **AES**—Uses the advanced encryption standard protocol. Select AES if the access points in your network require WPA or WPA2 association and are configured for AES data encryption.

---

**NOTE:** If you select WEP encryption, the Infranet Controller automatically selects the **Keys will be generated automatically for data privacy** option in the Odyssey Access Client Network properties for the wireless adapter on Odyssey Access Client.
12. Click **Save Changes**.

13. From the admin Roles > **user** > General > Overview page in the Infranet Controller admin console, select the check box for **Odyssey Settings for IC Access**. Note that if you also have the check box for Odyssey Settings for Preconfigured Installer selected, the options on the Odyssey Settings for IC Access page will overwrite the options from the preconfigured installer.

---

**NOTE:**

- Because the configuration settings you specify on the Infranet Controller overwrite any existing settings on the endpoint (except for login name), you can use the Odyssey Configuration page in the admin console to change settings for users. For example, if you want to remove the requirement to connect to a particular Infranet Controller, you can clear the **Require connection to this Infranet Controller** check box in the sign-in policy. Then instruct users to access the Infranet Controller again using a browser to download Odyssey Access Client with the new setting.

- The settings you specify on the Odyssey Configuration page do **not** configure the settings for the Odyssey Access Client installer (called OdysseyAccessClient.msi) that you can manually download from the Maintenance > System > Installers page (see “Downloading Application Installers” on page 344.) However, after installation, you can instruct users to access the Infranet Controller using a Web browser to automatically configure Odyssey Access Client using the configuration settings you specified on the Odyssey Configuration page.

- See the *Odyssey Access Client User Guide* available on the Juniper Networks Customer Support Center.

---

**Using the Preconfigured Installer for Odyssey Access Client**

You can create a preconfigured installer for the Odyssey Access Client that is downloaded to endpoints using the Custom Installer in the Odyssey Client Administrator (OCA). The OCA is a utility that allows you to fully configure all of the Odyssey Access Client settings. See the *Odyssey Access Client Administration Guide*.

A preconfigured installer contains the settings you configured using the OCA, as well as certificates. The installer may also contain a license key, and a flag indicating whether or not GINA is installed on the client. You must include the license key for Odyssey Access Client upgrades if you are upgrading to UAC 2.2 from an earlier version.

After you configure all of the settings with the OCA, you export the preconfigured installer as a .zip file to a directory that is accessible to the Infranet Controller. You upload the preconfigured installer file from the Infranet Controller admin console interface.
The settings options available in the OCA allow you to comprehensively control security and accessibility features that are available to users who access the Infranet Controller. For example, you can hide or disable the configuration icons on the sidebar of the Odyssey Access Client, you can control whether or not endpoints can modify adapter settings, and you can configure settings to prevent endpoints from disabling Odyssey Access Client.

The preconfigured installer downloads are role-based. When you create new user roles, you can specify a preconfigured installer that you have created specifically for the role. You can configure a different Odyssey Access Client feature set for each role that you create. For example, you might want to create a client configuration with few restrictions for employee roles and a more restrictive configuration for visitor roles. Alternately, you can choose to select the same configuration for different roles.

When the Odyssey Access Client is installed or upgraded on an endpoint, the preconfigured installer file is downloaded from the Infranet Controller. The file that is downloaded depends on the authenticated user’s role.

If a user is assigned to more than one role, and the roles have different Odyssey Access Client preconfigured installer files, the settings from the first role assigned are retained.

To preconfigure Odyssey Access Client:

1. From the admin Roles > user > General > Overview page in the Infranet Controller admin console, select the check box for Odyssey Settings for Preconfigured Installer. Note that if you also have the check box for Odyssey Settings for Access, the options on that page will overwrite the options from the preconfigured installer.

2. Open Odyssey Access Client Manager.


4. Configure all of the Odyssey Access Client settings that you want to apply for a preconfigured installer, including certificates and license using the tools in the OCA. See the Odyssey Access Client Administration Guide.

5. Double-click Custom Installer.

6. Select the option button for preconfigured file.

7. Enter a name for the preconfigured file.

8. Click Browse to download the file to a selected location.

9. Click OK.

NOTE: The download consists of a .zip file containing a properties.xml file with licenses and GINA settings, a preconfig.xml file, and certificates.
10. Select an existing role from the Users > User Roles page on the Infranet Controller admin console, or create a new role.

11. Select the check box for Odyssey Configuration.

12. Select the Agent tab.

13. Select the Odyssey Access Client tab.

14. Select the Preconfigured Installer link. The preconfigured installer upload page appears.

15. Select the Browse button and locate the file from the selected location. The preconfigured installer downloads the customized Odyssey Access Client version to all users who access the specified role.

**NOTE:** If you do not create and upload an Odyssey Access Client configuration for a role, users who access that role will get the factory default Odyssey Access Client version.

### Validating the Infranet Controller Certificate

Whenever users install Odyssey Access Client by accessing the Infranet Controller by means of a browser, the Validate server certificate option in the Authentication tab of the user’s profile is automatically enabled. When this option is enabled, Odyssey Access Client validates the server certificate of the Infranet Controller. Odyssey Access Client is automatically configured to trust the Infranet Controller if it can verify that the Infranet Controller is passing a valid certificate.

For this verification to occur, the trusted root certificate of the CA that signed the Infranet Controller’s server certificate must be installed on the endpoint. If the CA certificate is not installed, then the user is unable to authenticate.

There are three ways to install the trusted root CA certificate on the endpoint:

- You can use a CA certificate that is chained to a root certificate that is already installed on the endpoint, such as Verisign.

- Users or you can import the CA certificate(s) on the endpoint using Internet Explorer or other Microsoft Windows tools through whatever method your organization uses to distribute Root Certificates.

- You can upload the CA certificate and any intermediate CA certificates to the Infranet Controller. During Odyssey Access Client installation, the Infranet Controller automatically installs the CA certificate(s) on the endpoint. When prompted during installation, the user must allow the installation of the CA certificate(s). See “Uploading Trusted Server CA Certificates” on page 379.
Using Odyssey Access Client Licenses with the Infranet Controller

You can configure the Odyssey Access Client to download by default from the Infranet Controller. The downloaded edition contains all of the functionality in the Enterprise Edition, including an 802.1X wired and wireless supplicant. The currently installed Infranet Controller Endpoint License determines the maximum number of concurrent endpoints that can access the Odyssey Access Client with the Infranet Controller.

The FIPS Edition of Odyssey Access Client can also be used with the Infranet Controller. All editions support the full Odyssey Access Client feature set and Odyssey Access Client Administrator. The licenses for the FIPS edition and Enterprise Edition must be purchased in addition to the Infranet Controller Endpoint License. To prevent user sessions generated by these two editions from increasing the concurrent user count against the Infranet Controller Endpoint License, you can install the OAC-Add-UAC license on the Infranet Controller. See “Entering or Upgrading Infranet Controller Licenses” on page 346.

NOTE: The Infranet Controller is not FIPS compliant, but the FIPS edition of Odyssey Access Client can be used with the Infranet Controller. Data between the Odyssey Access Client and an 802.1X switch is protected by FIPS-validated encryption.

In UAC versions prior to 2.2, administrators had to distribute new license keys to endpoints when the software version was upgraded. In version 2.2 or greater, if you are updating from 2.0 or 2.1 this is no longer required. You can use the preconfigured installer for the Odyssey Access Client that includes an updated license key. See “Using the Preconfigured Installer for Odyssey Access Client” on page 51.

Manually installing and Configuring Odyssey Access Client for 802.1X

If your 802.1X switch does not allow network connectivity from your computer to the Infranet Controller, use the instructions in this section to download the Odyssey Access Client MSI to another computer that does have network connectivity to the Infranet Controller. Then transfer the Odyssey Access Client MSI to your computer to manually install and configure Odyssey Access Client.

Manually Installing Odyssey Access Client

To manually install Odyssey Access Client:

1. Use a computer that has network connectivity to the Infranet Controller admin console and then select Maintenance > System > Installers.

2. Click the Download link to the right of Odyssey Access Client. The File Download dialog box appears.

3. Click the Save button on the File Download dialog box. The Save As dialog box appears.

4. Choose an appropriate location in the Save As dialog box.

5. Click the Save button on the Save As dialog box.
6. Copy the Odyssey Access Client MSI installer program to the endpoint that does not have network connectivity to the Infranet Controller.

7. Double-click the Odyssey Access Client installer and follow the prompts. When the installation is complete, use the following instructions to configure Odyssey Access Client.

8. Install the trusted root CA certificate or intermediate CA on the endpoint that you generated. (See “Validating the Infranet Controller Certificate” on page 53.)

**NOTE:** You must have the same root CA or intermediate CA for the server certificate chain installed in the trusted root or intermediate certificate store of your machine. To install the CA on Windows systems, open Internet Explorer and from the menu bar select **Tools > Internet Options > Content > Certificates** and click **Import**.

---

**Manually Configuring Odyssey Access Client for 802.1X**

To manually configure Odyssey Access Client for 802.1X:

1. Double-click the Odyssey Access Client tray icon to display Odyssey Access Client Manager.

2. Configure a user profile in Odyssey Access Client:
   a. From the side pane of the Odyssey Access Client Manager, click **Configuration > Profiles > Add**. The Add Profile dialog box appears.
   b. Enter a name in the Profile box.
   c. Enter a name in the Login name box.
   d. Click the **Password** tab and select a password option.
   e. Click **OK**.

3. Configure Odyssey Access Client for 802.1X:
   a. From the side pane of the Odyssey Access Client Manager, click **Configuration > Adapters > Add**. The Add Adapter dialog box appears.
   b. Click either the **Wireless** or **Wired 802.1X** tab to choose a wireless or wired adapter.
   c. Select the adapter you want to use for 802.1X.
   d. Click **OK**.

**NOTE:** If you do not see your wireless adapter in the list, select **All Adapters**. Make sure that each of the adapters that you select under the Wireless tab is wireless. You cannot configure Odyssey Access Client for wireless connections unless you have a wireless adapter installed.
4. Connect to a 802.1X wireless network:
   a. From the side pane of the Odyssey Access Client Manager, click
      Configuration > Networks > Add. The Networks dialog box appears.
   b. Specify the following settings for your wireless 802.1X network:
      - Choose the appropriate setting for Association mode.
      - Choose the appropriate setting for Encryption method.
      - Choose Authenticate using profile and then select the profile you created
        earlier from the list box.
      - Choose Keys will be generated automatically for data privacy.

5. Connect to the network:
   a. From the side pane of the Odyssey Access Client Manager, click
      Configuration > Adapters and select the adapter you just added.
   b. If you are using a wired network, select the profile you just created from
      the Profile list on the right. If you are using a wireless network, select the
      wireless network you just added from the Network list on the right.
   c. Click Connect to the network.
   d. When you are prompted for a login name, sign in using the user name you
      configured and click OK.
   e. When you are prompted for a password, enter the password you
      configured and click OK.

Configuring Agentless Access to Protected Resources

Odyssey Access Client, which is one way to allow users to access protected
resources, supports the following endpoint platforms:

- Windows 2000 SP4
- Windows XP SP1, SP2
- Windows Vista, (32-bit)

The Infranet Controller also supports third-party 802.1X supplicants. See “Using the
Infranet Controller for 802.1X Network Access” on page 142 for instructions on
configuring 802.1X on the Infranet Controller.

You can install a basic Java agent with minimal functionality for Macintosh and
Linux platforms. See “Configuring the Java Agent for Endpoint Access” on page 58.

You can also allow users on Windows, Macintosh, Linux, or Solaris platforms to
access protected resources without installing and running Odyssey Access Client on
the endpoint. This type of access is referred to as agentless access.
You specify whether you want the Infranet Controller to install Odyssey Access Client or provision agentless access for endpoints at the role level. Then you can use role mapping to associate users with specific roles. This will allow you, for example, to assign specific users to roles that are configured to provision agentless access.

For example, if you have contract employees with non company machines onto which you do not want to install Odyssey Access Client, you can create two roles: one that allows agentless access, and the other requiring installation of the Odyssey Access Client. Then, create two associated realms: one for agentless access and one for Odyssey Access Client. Add role mapping rules based on user names to assign the contract employees to the agentless role, and employees to the agent role. When a user attempts to log in, they will be assigned to a role which will either provision agentless access or install Odyssey Access Client.

You can associate different realms with different sign-in policies and sign-in pages, so users who log in to a resource can see a sign-in page based on whether or not they are a regular employee or a contractor.

**NOTE:**

- When using agentless access, the user must leave the browser window open on the Infranet Controller sign-in page to stay signed into the Infranet Controller. If the user closes the browser window or opens a different page, the endpoint loses the connection to the Infranet Controller and the Infranet Enforcer denies the user access to protected resources. See “Deploying the UAC Solution to Users” on page 41.

- IPsec enforcement is not supported on agentless or java agent endpoints. You must use source IP enforcement by setting up a source-based policy on the Infranet Enforcer. See “Setting Up Source IP Enforcement on the Infranet Enforcer” on page 91.

- Host Enforcer policies are not supported on agentless endpoints. See “Infranet Enforcer Policies” on page 71.

- Agentless access with the Firefox browser is the only option for the Solaris platform.

- Some older Mozilla browsers may crash over time with agentless access. This is because the Infranet Controller normally uses AJAX to send heartbeat messages to the endpoint. You can disable AJAX when you configure agentless access if you have clients using older browsers.

**Task Summary: Configuring Agentless Access to Protected Resources**

To configure agentless access to protected resources:

1. Configure a source IP-based policy on the Infranet Enforcer. For example:

   set policy from "Trust" to "Untrust" "Any" "Any" "ANY" permit infranet-auth

   See “Source IP Enforcement on the Infranet Enforcer” on page 33.
2. Select **Users > User Roles > New User Role** in the left navigation bar of the admin console.

3. Enter a name for this role.

4. Click **Save Changes**. The Roles configuration page appears.

5. Select the **Agentless** tab.

6. Select the check box **Enable Agentless Access for this role**.

7. If you have endpoints that will be accessing the Infranet Controller from older Mozilla browsers, select the **Disable use of AJAX for heartbeats** check box. You can then specify these agentless roles for clients using version 10 or earlier.

   **NOTE:** You can continue configuring this role, or you can complete the configuration for agentless access. See “Configuring User Roles” on page 120.

8. Select **Users > User Realms > Select Realm > Role Mapping > New Rule** to configure role mapping rules that map agentless access users to the role(s) you configured. See “Creating Role Mapping Rules” on page 249.

9. Select **Authentication > Signing In > Sign-in Policies** to associate a sign-in policy with the realm you configured. See “Configuring Sign-In Policies” on page 259.


**Configuring the Java Agent for Endpoint Access**

You can allow Macintosh and Linux machines to access protected resources by configuring the Java agent option on the Infranet Controller for these platforms.

If you provision a role with Java agent access and direct users to a browser to sign in, the Java agent automatically downloads onto a Macintosh or Linux machine after the endpoint successfully authenticates and any applicable Host Checker policies are enforced.

The Java agent appears on these platforms as an icon in the menu bar. The Java agent displays connected or disconnected status, the assigned IP address and a link to log out from the session.

The Java agent maintains a heartbeat between the endpoint and the Infranet Controller. It is not necessary to leave a browser window open to maintain connectivity.

   **NOTE:** IPSec enforcement is not supported with Java agent access. You must use source IP enforcement by setting up a source-based policy on the Infranet Enforcer. See “Setting Up Source IP Enforcement on the Infranet Enforcer” on page 91.
Task Summary: Configuring the Java Agent

To configure Java agent access for Macintosh or Linux endpoints:

1. Select **Users > User Roles > New User Role** in the left navigation bar of the admin console.
2. Enter a name for this role.
3. Click **Save Changes**. The Roles configuration page appears.
4. Select the **Agent** tab.
5. Select the **Install Java agent for this role** check box. When a user assigned to this role authenticates through a portal that you have configured for access to the Infranet Controller, the agent is automatically downloaded to the user’s machine.
6. Click **Save Changes**

**NOTE:** You can continue configuring this role, or you can complete the configuration for Java agent access. See “Configuring User Roles” on page 120, or you can complete the configuration for Java agent access.

7. Select **Users > User Realms > Select Realm > Role Mapping > New Rule** to configure role mapping rules that map Java agent users to the role(s) you configured. For configuration instructions, see “Creating Role Mapping Rules” on page 249.

8. Select **Authentication > Signing In > Sign-in Policies** to associate a sign-in policy with the realm you configured. For configuration instructions, see “Configuring Sign-In Policies” on page 259.


Installing the Java Agent on SUSE Linux using KDE

If your end users on the Linux platform are using KDE on SUSE, you must first ensure that the gnome panel package is installed, otherwise the Java agent installation will not be successful.

To install the gnome panel package for SUSE users:

1. Open YaST installation and configuration tool for SUSE.
2. Select **Software > Software Management**.
3. Search for gnome panel.
4. In the package lists, check the gnome panel package and select **accept**.
5. Install the Java agent. See “Task Summary: Configuring the Java Agent” on page 59.
Using Network and Security Manager with the Infranet Controller

Network and Security Manager (NSM) is Juniper Networks network management tool that allows distributed administration of network appliances. You can use the NSM application to centralize status monitoring, logging, and reporting, and to administer Infranet Controller configurations.

You can also manage the Infranet Enforcer and IDP using NSM. Managing the complete UAC solution can be accomplished from a single management interface.

With NSM you can manage most of the parameters that you can configure through the Infranet Controller admin console. The configuration screens rendered through NSM are similar to the Infranet Controller’s native interface.

NSM incorporates a broad configuration management framework that allows co-management using other methods. You can import and export XML via the Infranet Controller’s admin console interface, or you can manage from the Infranet Controller’s admin console.

How the Infranet Controller and NSM communicate

The Infranet Controller and the NSM application communicate through the Device Management Interface (DMI). DMI is a collection of schema-driven protocols that run on a common transport (TCP). DMI is designed to work with Juniper Networks platforms to make device management consistent across all administrative realms. The DMI protocols that are supported include NetConf (for inventory management, XML-based configuration, text-based configuration, alarm monitoring, and device-specific commands), structured syslog, and threat flow for network profiling. DMI supports third-party network management systems that incorporate the DMI standard, however only one DMI-based agent per device is supported.

The Infranet Controller’s configuration is represented as a hierarchical tree of configuration items. This structure is expressed in XML that can be manipulated with NetConf. NetConf is a network management protocol that uses XML. DMI uses NetConf’s generic configuration management capability and applies it to allow remote configuration of the device.

To allow NSM to manage the Infranet Controller using the DMI protocol, NSM must import the schema and meta-data files from the Juniper Schema Repository, a publicly-accessible resource that is updated with each device release. In addition to downloading the Infranet Controller’s current schema, NSM may also download upgraded software.

The Schema Repository enables access to XSD and XML files defined for each device, model and software version.

Before attempting to communicate with NSM, you must first complete the initial configuration of the Infranet Controller. Initial configuration includes network interface settings, DNS settings, licensing, and password administration.

If you have several Infranet Controllers that will be configured in a clustering environment, the cluster abstraction must first be created in the NSM Cluster Manager. Then, you can add individual nodes.
After you have completed the initial network configuration, you can configure the Infranet Controller to communicate with NSM using the appropriate network information. Once the Infranet Controller has been configured to communicate with NSM, the Infranet Controller contacts NSM and establishes a DMI session through an initial TCP handshake.

All communications between the Infranet Controller and NSM occur over SSH to ensure data integrity.

After the Infranet Controller initially contacts NSM and a TCP session is established, interaction between the Infranet Controller and NSM is driven from NSM, which issues commands to get hardware, software, and license details of the Infranet Controller. NSM connects to the Schema Repository to download the configuration schema that is particular to the Infranet Controller.

NSM then issues a command to retrieve configuration information from the Infranet Controller. If NSM is contacted by more than one Infranet Controller as a member of a cluster, information from only one of the cluster devices is gathered. NSM attempts to validate the configuration received from the Infranet Controller against the schema from Juniper Networks.

Once the Infranet Controller and NSM are communicating, the Infranet Controller delivers syslog and event information to NSM.

After NSM and the Infranet Controller are connected, you can make any configuration changes directly on the Infranet Controller, bypassing NSM. NSM automatically detects these changes and imports the new configuration data. Changes to Infranet Controller cluster members will similarly be detected by NSM.

When you make changes to the Infranet Controller configuration through NSM you must push the changes to the device by performing an Update Device operation.

When you double-click the Infranet Controller device icon in the Device Manager and select the Config tab, the configuration tree appears in the main display area in the same orientation as items appears on the Infranet Controller interface.

**Available Services and Configuration Options**

The following services and options are provided to NSM by the Infranet Controller:

- **Inventory management service**—inventory management service enables management of the Infranet Controllers software, hardware, and licensing details. Adding or deleting licenses or upgrading/downgrading software are not supported.

- **Status monitoring service**—status monitoring service allows the Infranet Controller’s status to be obtained, including name, domain, OS version, synchronization status, connection details, and current alarms.

- **Logging service**—logging service allow the Infranet Controllers logs to be obtained in a time-generated order. Logging configuration details that are set on the Infranet Controller will apply to NSM.
XML-based configuration management service—configuration management service enables NSM to manage the configuration of the Infranet Controller. NSM uses the same XML Schema as the Infranet Controller, so you can troubleshoot NSM using XML files downloaded from the Infranet Controller. See “Importing and Exporting XML Configuration Files” on page 388.

The following device configuration items are not supported:

- Editing licensing information, (though licenses can be viewed)
- Creating clusters, joining nodes to clusters, or enabling or disabling cluster nodes
- Packaging log files or debug files for remote analysis
- Rebooting the Infranet Controller

**Task Summary: Configuring DMI Communication for NSM**

To configure the Infranet Controller to communicate with NSM you must coordinate actions between the Infranet Controller and NSM administrators. Items such as IP address, password, HMAC key (a one-time password), and the device ID must be shared between administrators of both the Infranet Controller and NSM.

To connect the Infranet Controller and NSM you will need to do the following:

- Install and configure the Infranet Controller.
- Add the Infranet Controller as a device in NSM.
- Configure and activate the DMI agent on the Infranet Controller.
- Confirm connectivity and import the Infranet Controller configuration into NSM.

**Configuring the Infranet Controller for the Initial DMI Connection**

To permit the Infranet Controller and NSM to make an initial connection, you must add an NSM administrative user to the Infranet Controller configuration. This section provides a summary of adding the NSM administrator and configuring the DMI agent to allow the Infranet Controller and NSM to communicate. Complete configuration of the Infranet Controller for authenticating users is outside the scope of this section.

To initiate a DMI session for communication between the Infranet Controller and NSM:

1. Ensure that basic connection information is configured on the Infranet Controller (network address, DNS, password).
2. Ensure that the proper licenses are installed on the Infranet Controller.
3. From the NSM UI client Device Manager, click the Add icon and select Device to open the Add Device wizard, and enter the applicable information required to add an Infranet Controller to NSM. See *Juniper Networks Netscreen Security Manager Administrator’s Guide*.

**NOTE:** You must enter a unique NSM admin username and password on the NSM UI client. This username will be used on the Infranet Controller as the username for the administrator account that will be used to manage the Infranet Controller. NSM must have a unique account login to avoid interrupting the communication with the Infranet Controller. NSM automatically generates a unique ID which is used for the HMAC key.

4. From the Infranet Controller admin console, select Authentication > Auth. servers and enter the username and password of the NSM admin using the credentials you entered on NSM in the applicable authentication server. Use the NSM username and password that you entered in the NSM UI Client. See “Authentication and Directory Servers” on page 173.

**NOTE:** Only password-based authentication servers can be used. One-time-password authentication is not supported.

5. Select Administrators > Admin Roles and create a DMI agent administrator role. See “User Roles” on page 117.

6. Select Administrators > Admin Realms and create a new DMI agent admin realm for the DMI agent on the Infranet Controller and use role mapping to associate the DMI agent role and realm. See “Authentication Realms” on page 245.

7. On the NSM interface, select the Domain menu and choose the domain to which the Infranet Controller will be added.

8. In Device Manager, click the Add icon and select Device to open the Add Device wizard, and enter the applicable information required to add an Infranet Controller to NSM. See *Juniper Networks Netscreen Security Manager Administrator’s Guide*.

**NOTE:**

- In a clustering environment, each cluster-node must have its own unique DMI agent and its own device-id and HMAC key, as each cluster node maintains its own persistent DMI connection to the management application.

- The HMAC key and the device id are hashed to identify individual devices to the application. Juniper recommends that you use a strong password for the HMAC key value to ensure that the key isn’t guessed.

9. After you have added the Infranet Controller to NSM, select System > Configuration > DMI Agent on the Infranet Controller admin console and enter the NSM Primary Server IP address or hostname, Primary Port, Backup Server and Backup Port (if applicable), the Device ID, and the HMAC Key.
10. Select the **Admin realm** that you have configured for the DMI agent.

11. Select the **Enabled** option button.

The Infranet Controller initiates a TCP connection to NSM. After the Infranet Controller is identified to NSM through the HMAC key and device ID hash, the Infranet Controller and NSM negotiate an SSH tunnel, and NSM requests authentication to the Infranet Controller based on the username and password.

If you need to disconnect the device from NSM, you can either disable the DMI agent from the device, or you can delete the device from the NSM interface. If the DMI connection is later reestablished, NSM will automatically retrieve any configuration changes, as well as logs that have accumulated in the interim.

**Using NSM with the Infranet Controller and the Infranet Enforcer**

To manage Infranet Controllers and Infranet Enforcers together with NSM, you must manually perform the initial configuration on the devices, and then import the devices through the NSM interface.

**Adding Infranet Controller Clusters**

To add an Infranet Controller cluster in NSM, you first add the cluster, then you add each member. Adding a member is similar to adding a standalone Infranet Controller. You must have a cluster object and all of the cluster members defined in NSM to allow NSM to access the cluster.
Part 2
Access Management Framework

The Infranet Controller protects resources by using the following access management mechanisms:

- **Authentication realm**—Resource accessibility begins with the authentication realm. An *authentication realm* specifies conditions that users must meet in order to sign into the Infranet Controller. An authentication realm specification includes several components, including an authentication server which verifies that the user is who he claims to be. The user must meet the security requirements you define for a realm's authentication policy or else the Infranet Controller does not forward the user's credentials to the authentication server.

- **User roles**—A role's configuration serves as the second level of resource access control. A *role* is a defined entity that specifies Infranet Controller session properties for users who are mapped to the role. Not only does a role specify the access mechanisms available to a user, but you can also specify restrictions with which users must comply before they are mapped to a role.

- **UAC policies**—A UAC policy serves as the third level of access control. A *resource policy* is a set of resource names (such as IP addresses/netmasks or ranges) to which you grant or deny access to endpoints. A resource policy can also specify a connection from the Infranet Controller to the Infranet Enforcer, and properties of connections from endpoints to the Infranet Enforcer. Policies are role-based. Each policy specifies a destination, a set of roles, and a set of actions.

- **Infranet Enforcer**—The Infranet Enforcer can be deployed in front of servers and resources that you want to protect. The Infranet Enforcer enforces security policies to control access in a number of ways.

This section contains the following information about the Infranet Controller access management framework:

- “General Access Management” on page 67
- “User Roles” on page 117
- “The UAC RADIUS Server and Layer 2 Access” on page 131
- “Authentication and Directory Servers” on page 173
- “Authentication Realms” on page 245
“Sign-In Policies” on page 257
Chapter 3
General Access Management

The enforcement policies that you configure on the Infranet Controller determine how the UAC solution components work together to provide granular end-point security services.

Whether you are using the Infranet Controller, Infranet Enforcer, and Odyssey Access Client together, or deploying the Infranet Controller without the Infranet Enforcer using 802.1X, the policies you configure provide the access management control to ensure that only qualified users can successfully authenticate.

There are several types of policies. Role and realm restrictions allow you to apply parameters to define endpoint, user, and agent profiles. Infranet Enforcer policies are firewall policies that allow you to configure resource access policies, IPsec and source IP enforcement. Host Enforcer policies are packet filters that run on the endpoint. Network access policies are used to control access if you are using 802.1X.

This section contains the following information about enforcement policies and the access management framework:

- “Realm and Role Restrictions” on page 68
- “Specifying Resources for User Access Control” on page 70
- “Infranet Enforcer Policies” on page 71
- “Introduction to Network Access Policies” on page 96
- “Host Enforcer Policies” on page 97
- “Policies, Rules & Restrictions, and Conditions Evaluation” on page 101
- “Dynamic Policy Evaluation” on page 105
- “Policy Evaluation” on page 107
- “Configuring Security Requirements” on page 108
Realm and Role Restrictions

The Infranet Controller enables you to secure your company resources using authentication realms and user roles. This flexibility allows you to control access from a very broad level (controlling who may sign into the Infranet Controller) down to a very granular level (controlling which authenticated users may access a particular URL or file).

You create policies on the Infranet Controller that permit or deny access to resources and services based on a user’s role and the security compliance of the endpoint device. With Odyssey Access Client, you can incorporate the Infranet Enforcer to more effectively control access.

The Infranet Controller manages the user authentication and roles, and stores the policies. Based on the user’s identity and endpoint status, the Infranet Controller assigns the user a set of roles. These roles, in turn, specify what resources the endpoint can access. The Infranet Controller pushes the allow or deny information for the user in the form of firewall policies to the Infranet Enforcer and Odyssey Access Client. Once the Infranet Enforcer and Odyssey Access Client have policies that allow access for the endpoint, the Infranet Enforcer allows traffic between the endpoint and the protected resources.

Realm and role restrictions are not supported for deployments in which users access the Infranet Controller using non-UAC agents, for example non-Juniper 802.1X supplicants.

This section contains the following information about realm and role restrictions:

- “Accessing Authentication Realms” on page 68
- “Accessing User Roles” on page 69

Accessing Authentication Realms

Resource accessibility begins with the authentication realm. An authentication realm is a grouping of authentication resources, including:

- **An authentication server**, which verifies that the user is who they claim to be. The Infranet Controller forwards credentials that a user submits by using Odyssey Access Client, or by using a sign-in page for agentless and Java agent deployments) to an authentication server. For more information, see “Authentication and Directory Servers” on page 173.

- **An authentication policy**, which specifies realm security requirements that need to be met before the Infranet Controller submits a user’s credentials to an authentication server for verification. For more information, see “Defining Authentication Policies” on page 248.

- **A directory server**, which is an LDAP server that provides user and group information to the Infranet Controller that the Infranet Controller uses to map users to one or more user roles. For more information, see “Authentication and Directory Servers” on page 173.
Role mapping rules, which are conditions a user must meet for the Infranet Controller to map the user to one or more user roles. These conditions are based on either user information returned by the realm’s directory server or the user's username. For more information, see “Creating Role Mapping Rules” on page 249.

You can associate one or more authentication realms with an Infranet Controller sign-in page. When more than one realm exists for a sign-in page, a user can specify a realm. When the user submits credentials, the Infranet Controller checks the authentication policy defined for the realm. The user and the endpoint must meet the security requirements you define for a realm’s authentication policy or the Infranet Controller does not forward the user’s credentials to the authentication server.

At the realm level, you can specify security requirements based on various elements such as the user’s source IP address or the possession of a client-side certificate. If the user meets the requirements specified by the realm’s authentication policy, the Infranet Controller forwards the user’s credentials to the appropriate authentication server. If this server successfully authenticates the user, then the Infranet Controller evaluates the role mapping rules defined for the realm to determine which roles to assign to the user. See “Authentication Realms” on page 245.

# Accessing User Roles

A role specifies Infranet Controller session properties for users who are mapped to the role. These session properties include information such as session time-outs, limitations, and restrictions. A role’s configuration serves as the second level of user access control. Not only does a role specify the access mechanisms available to a user, but you can also specify restrictions with which users must comply before they are mapped to a role. The user must meet these security requirements or else the Infranet Controller does not map the user to a role.

At the role level, you can specify security requirements based on elements such as the user’s source IP address and possession of a client-side certificate. If the user meets the requirements specified either by a role mapping rule or a role’s restrictions, then the Infranet Controller maps the user to the role. When a user makes a request to the backend resources available to the role, the Infranet Enforcer and/or Odyssey Access Client evaluate the corresponding resource policies.

Note that you may specify security requirements for a role in two places—in the role mapping rules of an authentication realm (using custom expressions) or by defining restrictions in the role definition. The Infranet Controller evaluates the requirements specified in both areas to make sure the user complies before it maps the user to a role. See “User Roles” on page 117.

Figure 5 shows the order in which the Infranet Controller evaluates realm and role and restrictions after a user submits their credentials on a sign-in page.
Specifying Resources for User Access Control

The Infranet Controller’s engine that evaluates access policies requires that the resources listed in a policy’s Resources list follow a canonical format. This section describes the canonical formats available for specifying resources in a Host Enforcer policy or an Infranet Enforcer Resource Access Policy.

When the Infranet Controller or a user tries to access or use a resource, an Infranet Controller appliance compares the requested resource to the resources specified in the corresponding policies, starting with the first policy in a policy list. When the engine matches a requested resource to a resource specified in a policy’s Resources list, it then evaluates further policy constraints and returns the appropriate action to the appliance (no further policies are evaluated). If no policy applies, then the appliance performs the default action for the policy, which is some cases may be to deny access by taking no action.

When specifying server resources for Infranet Controller resource policies, note the following guidelines.

**Canonical format:** protocol://IP address/net-mask:[ports]

The components are:

- **Protocol (optional)**—Possible case-insensitive values:
  - tcp
  - tcp_in and tcp_out, (Host Enforcer policies only)
  - udp
  - udp_in and udp_out, (Host Enforcer policies only)
  - icmp
**NOTE:** The only allowed ICMP configuration in a Host Enforcer policy is:
```
icmp://*:*  
```
That is, you cannot specify `icmp://ip-addr/net-mask:port` as you can for the other protocols. For more information about the format of resources in Host Enforcer policies, see “Infranet Enforcer Policies” on page 71.

If the protocol is missing, then all protocols are assumed. If a protocol is specified, then the delimiter “://” is required. No special characters are allowed.

- **IP address/net-mask**—The IP address is required, but the netmask is optional. You can use * to specify all IP addresses.

- **Host (required)**—Possible values:

```

```

**Table 10: DNS Hostname Special Characters**

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>Matches ALL characters</td>
</tr>
<tr>
<td>%</td>
<td>Matches any character except dot (.)</td>
</tr>
<tr>
<td>?</td>
<td>Matches exactly one character</td>
</tr>
</tbody>
</table>

- **Ports (optional)**—Possible values:

**Table 11: Port Possible Values**

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>Matches ALL ports; no other special characters are allowed</td>
</tr>
<tr>
<td>port[,port]*</td>
<td>A comma-delimited list of single ports. Valid port numbers are [1-65535].</td>
</tr>
<tr>
<td>[port1]-[port2]</td>
<td>A range of ports, from port1 to port2, inclusive.</td>
</tr>
</tbody>
</table>

**NOTE:** You may mix port lists and port ranges, such as: 80,443,8080-8090

If the port is missing, then the default port 80 is assigned for http, 443 for https. If a port is specified, then the delimiter “:” is required. For example:
```
tcp://10.10.149.149:22,23
```
```
tcp://10.11.0.10:80
```
```
udp://10.11.0.10:*
```

---

**Infranet Enforcer Policies**

The Infranet Enforcer can be deployed in front of servers and resources that you want to protect. The Infranet Enforcer enforces security policies to control access in a number of ways.
- **Infranet Enforcer resource access policy**—An Infranet Enforcer resource access policy specifies which users are allowed or denied access to a set of protected resources. You specify which users you want to allow or deny by choosing the roles for each Infranet Enforcer access policy. For example, an Infranet Enforcer access policy can specify that a user must have an Antivirus role to access a network, and the Antivirus role requires the user’s computer to run a particular antivirus program.

- **Infranet Enforcer IPsec routing policy**—This type of policy specifies which Infranet Enforcer an endpoint must use to access a resource. This policy also specifies whether that resource requires an IPsec tunnel for Windows endpoints to access it. Note that an IPsec tunnel does not automatically give the endpoint access.

- **Infranet Enforcer ScreenOS IPsec policy**—An IPsec policy is a ScreenOS infranet-auth policy that contains a source and destination. The Infranet Controller uses the source and destination to set up a user, user group, IKE gateway, and VPN for each source interface in the source zone of the policy. You can set up a Screen OS IPsec policy on the Infranet Controller and push the policy to the Infranet Enforcer, or you can set up the policy using ScreenOS WebUI or the command line.

- **Infranet Enforcer IP address pools policy**—This type of policy specifies a pool of virtual IP addresses that you want the Infranet Controller to automatically assign to endpoints in NAT environments that use IPsec tunnels to the Infranet Enforcer.

- **Infranet Enforcer source interface policy**—This type of policy specifies the source interface on the Infranet Enforcer that receives traffic from endpoints if the Infranet Enforcer is in Transparent mode. You only need to create this type of policy if you want endpoints to use IPsec to communicate with an Infranet Enforcer that is in Transparent mode, and you are using virtual routers.

- **Infranet Enforcer ScreenOS source IP policy**—A source IP policy is a ScreenOS infranet-auth policy that contains a source and destination that permits the Infranet Enforcer to route clear-text traffic between source and destination zones. You can set up a Screen OS source IP policy on the Infranet Controller and push the policy to the Infranet Enforcer, or you can set up the policy using ScreenOS WebUI or the command line.

- **Infranet Enforcer auth table mapping policy**—This type of policy specifies which Infranet Enforcer device(s) an endpoint must use to access resources when the endpoint is using source IP enforcement. If you are using ScreenOS version 6.1 or greater, it is not necessary to configure auth table mapping policies.

### Configuring Infranet Enforcer Resource Access Policies

An Infranet Enforcer resource access policy specifies which users are allowed or denied access to a set of protected resources.
You identify resources within your protected network, then you specify which users you want to allow or deny by choosing the roles for each Infranet Enforcer resource access policy. Auth table entries on the Infranet Enforcer match user requests for access with resource access policies.

The Infranet Enforcer evaluates traffic and enforces access control based on the policies that you specify. When traffic from a role with a ScreenOS security policy enabled is compared to a policy and there is a matching entry, the Infranet Enforcer applies the appropriate policy action.

For example, an Infranet Enforcer resource access policy can specify that a user must have an *Antivirus* role to access a particular network, and the *Antivirus* role requires the user’s computer to run a particular antivirus program.

You can create an additional layer of security by applying ScreenOS Infranet Enforcer security policy actions to endpoints. Antispam, logging, IDP, web filtering, antivirus, and deep inspection policies can be applied to any role.

The Infranet Controller pushes the policies to the Infranet Enforcer when the Infranet Enforcer first connects to the Infranet Controller, and any time you make a change to an Infranet Enforcer resource access policy.

When any change is made on the resource access policies page, all resource access policies on the Infranet Enforcer are refreshed, and endpoints that are accessing resources through resource access policies will be temporarily interrupted.

### Notifying the User of Denied Traffic

If you are using ScreenOS version 6.2 or greater on the Infranet Enforcer, you can configure customized messages for authenticated Odyssey Access Client users who attempt to access a resource to which you have denied access.

When you specify the deny action in a resource access policy for a role or roles, a text field appears. You can use the following variables to display information for the user:

- `<USER>` the login name of the user
- `<SOURCEIP>` the source IP of the packet
- `<DESTIP>` the destination IP of the packet
- `<PROTOCOL>` the protocol of the packet
- `<DESTPORT>` the destination port of the packet

You can use these variables along with your own text to compose the deny message that you send to users.

When the message is sent to the user, the applicable information about the session or the resource is substituted. The information is displayed on the user’s endpoint in a balloon in the tray icon.

To configure Infranet Enforcer access policies:

1. In the Infranet Controller admin console, choose **UAC > Infranet Enforcer > Resource Access**.
2. Click **New Policy**.
3. On the **New Policy** page:
   
   a. For **Name**, enter a name to label this Infranet Enforcer resource access policy.
   
   b. For **Description**, enter an optional description.

4. For **Resources**, specify the protocol, IP address, network mask, and port of each resource (or range of addresses) for which this Infranet Enforcer resource access policy applies, one per line. Do not insert any spaces in your entries, or the policy may not be applied correctly.

   You cannot specify a host name in an Infranet Enforcer resource access policy. You can only specify an IP address. You can use TCP, UDP, or ICMP. For more information, see “Specifying Resources for User Access Control” on page 70.

5. Specify one of the following in the **Roles** section:

   - **Policy applies to ALL roles**—Choose this option to apply this Infranet Enforcer resource access policy to all users.
   
   - **Policy applies to SELECTED roles**—Choose this option to apply this Infranet Enforcer resource access policy only to users who are mapped to roles in the **Selected roles** list. Be sure to add roles to this list from the **Available roles** list.
   
   - **Policy applies to all roles OTHER THAN those selected below**—Choose this option to apply this Infranet Enforcer resource access policy to all users except those who map to the roles in the **Selected roles** list. Be sure to add roles to this list from the **Available roles** list.

6. In the **Action** section, specify whether you want to use this Infranet Enforcer resource access policy to **allow** or **deny** access to the specified resources.

---

**NOTE:**

- If you choose **deny**, a text box appears that allows you to customize a deny message for users. See “Notifying the User of Denied Traffic” on page 73.

- If you want to record deny actions in the User Access Log, select the check box for **Enforcer Deny Messages** on the Log/monitoring > User Access > Settings page. The log records the user, source IP, destination IP, protocol, and destination port.
7. In the **ScreenOS Options** section, use the option buttons to determine the policy options that you want to apply to selected roles. Use the **Add** and **Remove** buttons to specify antispam, logging, IDP, web filtering, antivirus, and deep inspection.

**NOTE:** By default, all policy options are enabled on the Infranet Controller. To enforce the policies, you must create corresponding policies on the Infranet Enforcer. If the Infranet Controller is upgraded from a previous version, all ScreenOS options are enabled for all of the resource access policies that were available prior to the upgrade.

8. Click **Save Changes**.

**Setting up IPsec Enforcement on the Infranet Enforcer**

On supported Windows endpoints that use Odyssey Access Client, you can use IPsec enforcement to encrypt the traffic between an endpoint and the Infranet Enforcer, adding an additional layer of protection for network assets.

**NOTE:** IPsec is not supported on agentless or Java agent endpoints, or on endpoints that connect with non-UAC software. Instead, you must use source IP enforcement. For more information, see “Setting Up Source IP Enforcement on the Infranet Enforcer” on page 91.

To use IPsec enforcement with Odyssey Access Client, you set up a VPN tunnel with IKE (Internet Key Exchange) on the Infranet Enforcer. You can configure IPsec enforcement by creating ScreenOS policies and IPsec routing policies on the Infranet Controller (see “Configuring IPsec Enforcement on the Infranet Controller” on page 78).

Alternately, you can manually set up the VPN tunnel by using Infranet Enforcer ScreenOS CLI commands or WebUI. Juniper Networks recommends that you use the Infranet Controller to set up VPN policies on the Infranet Enforcer. If necessary, you can modify the VPN tunnel setup afterwards on the Infranet Enforcer by using ScreenOS CLI commands or WebUI.

**NOTE:** If you modify the VPN tunnel setup on the Infranet Enforcer by using the ScreenOS CLI commands or WebUI, you must refresh the policies on the Infranet Controller. For more information, see “Refreshing IPsec Policies on the Infranet Controller” on page 82.

When you use the Infranet Controller to configure IPsec on the Infranet Enforcer, the Infranet Controller sets up a user, user group, IKE gateway, and VPN for each source interface in the source zone of the policy in addition to the policy itself. The Infranet Controller uses the source interface number and the ID of the destination zone to uniquely name each of these objects.
For example, if you use the Infranet Controller to set up IPsec enforcement from Untrust to Trust, and Untrust has one source interface ethernet2. If ethernet2 has an interface number of 5, and Trust has a zone id of 2, then the Infranet Controller runs the following CLI commands on the Infranet Enforcer:

```
set user $infra-u-5-2 ike-id u-fqdn u5-2.juniper.net share-limit 50000
set user $infra-u-5-2 type ike
set user-group $infra-g-5-2 user $infra-u-5-2
set ike gateway $infra-gw-5-2 dialup $infra-g-5-2 Aggr outgoing-interface ethernet2
seed-preshare ARANDOMSTRING proposal pre-g2-3des-sha
set ike gateway $infra-gw-5-2 xauth server $infranet query-config
set ike gateway $infra-gw-5-2 xauth server auth-method chap
set vpn $infra-vpn-5-2 gateway $infra-gw-5-2 no-replay tunnel idletime 0 sec-level compatible
set policy from Untrust to Trust "Dial-Up VPN" any any tunnel vpn $infra-vpn-5-2 infranet-auth
```

**Configuring the IPsec Encryption Settings on the Infranet Enforcer**

When you use the Infranet Controller to configure IPsec on the Infranet Enforcer, the Infranet Controller creates an IPsec policy that uses these default IPsec encryption settings for the phase 2 proposal: NoPFS, ESP, 3DES, and SHA1. You can, however, change the phase 2 proposal by using the Infranet Enforcer CLI or Web UI. For example, you can change the phase 2 proposal using this CLI command:

**CLI**

```
set vpn $infra-vpn-2-11 gateway $infra-gw-2-11 proposal nopfs-esp-aes128-sha
```

The possible values for the phase 2 proposal are:

- nopfs-esp-des-md5
- nopfs-esp-des-sha1
- nopfs-esp-3des-md5
- nopfs-esp-3des-sha1
- nopfs-esp-aes128-md5
- nopfs-esp-aes128-sha1
- nopfs-esp-aes256-sha1
- nopfs-esp-null-sha1
- g2-esp-des-md5
- g2-esp-des-sha1
- g2-esp-3des-md5
- g2-esp-3des-sha1
- g2-esp-aes128-md5
- g2-esp-aes128-sha1
- g2-esp-aes192-sha1
- g2-esp-aes256-sha1
WebUI

VPN > AutoKey IKE > edit > advanced > select user defined custom

NOTE:

- It does not matter what preshared seed you use for the IKE gateway because
  the Infranet Controller will override it when it connects to the Infranet
  Enforcer.

- Be sure to use the following settings for the VPN tunnel setup or else the IPsec
  enforcement will not work correctly between the Infranet Controller and the
  Infranet Enforcer:
    - xauth authentication
    - dialup VPN
    - CHAP auth method

For more information on setting up a VPN tunnel for a dialup user with IKE, see the
Concepts & Examples ScreenOS Reference Guide: Volume 5, Virtual Private Networks
or the ScreenOS CLI Reference Guide, which you can download from
www.juniper.net/techpubs/.

Creating a Bidirectional VPN Policy on the Infranet Enforcer

If users need to connect from a computer inside the protected network back to the
endpoint that is connected by means of a dialup VPN policy, you can create a
bidirectional VPN policy on the Infranet Enforcer. You create a bidirectional VPN
policy on an existing IPsec (dialup VPN) policy that you created previously to allow
traffic from the endpoint to the protected network.

Here are two example use cases for bidirectional VPN policies:

- A user’s endpoint at the office is connected by means of a dialup VPN policy,
  and the user wants to remotely connect to his office computer from another
  computer outside the office.

- To diagnose a problem, a Helpdesk employee needs to use a remote desktop
  system such as Virtual Network Computing (VNC) or Microsoft Remote Desktop
  on another computer to connect to a user’s computer, and the user’s computer
  is connected by means of a dialup VPN policy.
To create a bidirectional VPN policy on the Infranet Enforcer:

**CLI**

```bash
set pol from <protected-resource-zone> to <endpoint-zone> any "Dial-up VPN" any
tunnel vpn <vpn-from-policy-going-the-other-way> pair-policy <id-of-policy-going-the-other-
way>
```

For example, suppose you have an infranet-auth dialup VPN policy from untrust to trust on your Infranet Enforcer. The ID of the policy is 36, and the vpn is $infra-vpn-1-2. The following command sets up the corresponding bidirectional VPN policy to allow traffic from trust to untrust:

```bash
set pol from trust to untrust any "Dial-up VPN" any tunnel vpn $infra-vpn-1-2
pair-policy 36
```

**WebUI**

1. Using the Infranet Controller, create the IPsec (dialup VPN) policy on the UAC > Infranet Enforcer > ScreenOS Policies page.
2. Using the Infranet Enforcer Web UI, edit the dialup VPN policy that you created in the previous step:

   Policies > Edit <existing dialup VPN policy>

3. Select **Modify matching bidirectional VPN policy** and click **OK**.
4. Edit the new bidirectional VPN policy that you just created in the previous step:

   Policies > Edit <new bidirectional policy>
   a. Deselect **Modify matching bidirectional VPN policy**.
   b. Click **Advanced**.
   c. Deselect **Authentication** on the **Advanced Policy Settings** page.
   d. Click **OK**.

**Configuring IPsec Enforcement on the Infranet Controller**

This section details the tasks required to configure IPsec on the Infranet Controller.

Depending on the version of ScreenOS you are using, the steps for configuring IPsec differs. If you are using a ScreenOS version prior to 6.1, you configure ScreenOS IPsec policies and an IPsec routing policy for each resource that you wish to protect. With ScreenOS version 6.1 or later the Infranet Controller can dynamically provision IPsec routing policies for you, eliminating the need to configure a separate policy for each resource.

The Infranet Controller initially queries all connected Infranet Enforcers for version information. If ScreenOS version 6.1 or greater is detected, the Infranet Controller initiates a command to enable dynamic provisioning.
When an authenticated endpoint without an auth table entry attempts to access a resource through the Infranet Enforcer, the Infranet Enforcer sends a message to the Infranet Controller including the source and destination IP, the source interface the dropped packet was received on, and the ID of the ScreenOS policy associated with the resource.

This information allows the Infranet Controller to negotiate an IPsec tunnel between the endpoint and the Infranet Enforcer and provision a dynamic IPsec routing policy.

Odyssey Access Client must be running on the Windows endpoint to use IPsec. IPsec is not supported on agentless endpoints, or on endpoints with the Java agent. On all other platforms, such as Macintosh, Linux and Solaris, or Windows machines that connect using a non-UAC client, you must use source IP enforcement by setting up a source-based policy on the Infranet Enforcer. For more information, see “Setting Up Source IP Enforcement on the Infranet Enforcer” on page 91.

IPsec is optional on Windows endpoints. Instead of IPsec, you can use source IP enforcement on the Infranet Enforcer.

If you upgraded your Infranet Controller from a version 1.x to 2.0 or later, be aware that the Infranet Controller does not automatically upgrade any previously configured IPsec enforcement policies. You must manually reconfigure all IPsec enforcement policies. For more information, see “Upgrading IPsec Enforcement Policies from Version 1.x of the Infranet Controller” on page 90.

**Related Information**

- “Task summary: Configuring IPsec Enforcement” on page 80
- “Refreshing IPsec Policies on the Infranet Controller” on page 82
- “Configuring IPsec Routing Policies” on page 82
- “Configuring IP Address Pool Policies for IPsec in a NAT Environment” on page 85
- “Configuring Source Interface Policies” on page 89
- “Upgrading IPsec Enforcement Policies from Version 1.x of the Infranet Controller” on page 90
Task summary: Configuring IPsec Enforcement

Table 12: Before Beginning

<table>
<thead>
<tr>
<th>Topic</th>
<th>Detail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connection</td>
<td>The Infranet Enforcer and the Infranet Controller must be connected before you use the Infranet Controller to set up IPsec enforcement on the Infranet Enforcer.</td>
</tr>
</tbody>
</table>
| IPSec access server          | You cannot establish an IPsec tunnel if IAS (IPsec access server) is enabled on the Infranet Enforcer. To determine if IAS is enabled, enter the following command on the Infranet Enforcer Command Line Interface:  
  
  `get ipsec access_session status`
  
  To turn off IAS, use the following command:  
  
  `unset ipsec access-session enable`.  
  
  See the Concepts & Examples ScreenOS Reference Guide, which you can download from www.juniper.net/techpubs/ for details of this functionality. |
| Multiple interfaces          | If you configure IPsec enforcement for an Infranet Enforcer that has multiple interfaces in the source zone, the Infranet Controller configures a unique IKE gateway, VPN, and tunnel policy for each interface. To distinguish between the tunnel policies, the Infranet Controller displays the name of the vpn for each tunnel policy in the VPN column on the UAC > Infranet Enforcer > ScreenOS Policies page after you click Save Changes. |
| Dynamic IPSec routing policies | If you are using ScreenOS version 6.1 or higher you can configure the Infranet Enforcer to dynamically provision IPSec routing policies. To implement this feature, you create a ScreenOS source IP policy with a source and destination zone that matches the ScreenOS IPsec policy. Then, you configure captive for the source IP policy. |
| Default encryption settings  | When you use the Infranet Controller to configure IPsec on the Infranet Enforcer, the Infranet Controller creates an IPsec policy that uses these default IPSec encryption settings: NoPFS, ESP, 3DES, and SHA1 You can, however, change the phase 2 proposal on the Infranet Enforcer by using the CLI or Web UI. For more information, see “Configuring the IPsec Encryption Settings on the Infranet Enforcer” on page 76. |
| Bi-directional VPN policy    | If you need to connect from a computer inside the protected network back to the endpoint, you can create a bidirectional VPN policy on the Infranet Enforcer. For configuration instructions, see “Creating a Bidirectional VPN Policy on the Infranet Enforcer” on page 77. |
| CLI commands                 | To include the CLI commands that the Infranet Controller sends to the Infranet Enforcer in the Infranet Controller event logs, select the Enforcer Command Trace option on the System > Log/Monitoring > Events > Settings page. For more information, see “Specifying Which Events to Save in the Log File” on page 421. |
| Troubleshooting              | To troubleshoot your IPsec configuration, you can view the Event logs on the Infranet Controller (see “Configuring events, user access, and Admin Access Logs” on page 419). You can also view IPsec information on the endpoint by choosing Tools > Diagnostics > IPsec Diagnostics in Odyssey Access Client. |
To configure IPsec enforcement:

1. On the **UAC > Infranet Enforcer > Connection** page, click **New Enforcer** if you have not already configured a connection to the Infranet Enforcer on which you want to configure IPsec enforcement. See “Configuring the Infranet Controller to Connect to the Infranet Enforcer” on page 20.

2. Select **UAC > Infranet Enforcer > Connection** and click the name in the **Enforcer** column of the Infranet Enforcer on which you want to configure IPsec enforcement.

3. On the **UAC > Infranet Enforcer > ScreenOS Policies** page:
   a. Choose the source zone for the policy from the **Source Zone** drop-down list. The source zone is the zone where the endpoint is located.
   
   b. Choose the destination zone for the policy from the **Destination Zone** drop-down list. The destination zone is where the protected resources are located.
   
   c. Choose **IPsec** from the **Type** drop-down list.
   
   d. Click **Add**.
   
   e. Click **Save Changes** to save the IPsec policy.

As required, the Infranet Controller sets up a VPN tunnel with IKE on the Infranet Enforcer that consists of a user, user-group, IKE gateway, and VPN for each source interface in the source zone of the policy. The Infranet Controller uses the source interface number and the ID of the destination zone to uniquely name each of these objects.

4. To configure the Infranet Controller to provision dynamic IPsec routing policies, create a new source IP policy on the Infranet Enforcer with the same source and destination zone as the ScreenOS IPsec policy and the captive portal feature configured to **Redirect all traffic**. See “Configuring a Captive Portal on the Infranet Enforcer” on page 33. Configuring an identical source IP policy is also required to use dynamic auth table allocation. See “Dynamic Auth Table Allocation” on page 92.

5. On the **UAC > Infranet Enforcer > Resource Access** page, configure Infranet Enforcer resource access policies to specify which users are allowed or denied access to a set of protected resources. (See “Configuring Infranet Enforcer Resource Access Policies” on page 72.)

6. On the **UAC > Infranet Enforcer > IPsec Routing** page, configure IPsec routing policies to specify which Infranet Enforcer device the endpoints must use to access each set of resources when using IPsec. (See “Configuring IPsec Routing Policies” on page 82.)

7. If you are using IPsec in a NAT environment, configure IP address pool policies on the **UAC > Infranet Enforcer > IP Address Pools** page. (See “Configuring IP Address Pool Policies for IPsec in a NAT Environment” on page 85.)
8. If you want endpoints to use IPsec to communicate with an Infranet Enforcer that is in Transparent mode, in some cases you may need a source interface policy. (See “Configuring Source Interface Policies” on page 89.)

**Refreshing IPsec Policies on the Infranet Controller**

Each time the Infranet Enforcer and Infranet Controller establish a new connection with each other, the Infranet Controller queries the Infranet Enforcer for policy information, which it uses for provisioning IPsec to endpoints. If you modify the IPsec policies on the Infranet Enforcer by using the ScreenOS CLI or Web UI while the Infranet Enforcer is connected to the Infranet Controller, you must refresh the policies on the Infranet Controller because the Infranet Controller does not automatically detect changes to Infranet Enforcer policies that occur while the Infranet Enforcer is connected. If you don’t refresh the policies, the Infranet Controller continues to use the older policies until the next time the Infranet Controller disconnects and reconnects to the Infranet Enforcer.

To refresh the IPsec policies on the Infranet Controller:

1. While the Infranet Enforcer is connected to the Infranet Controller, navigate to the UAC > Infranet Enforcer > Connection page.
2. Click the name in the Enforcer column of the Infranet Enforcer on which you want to configure IPsec enforcement.
3. On the UAC > Infranet Enforcer > ScreenOS Policies page, click the Refresh Policies button.

**Configuring IPsec Routing Policies**

An IPsec routing policy specifies which Infranet Enforcer device endpoints must use to access resources when using IPsec. The IPsec routing policy also specifies that endpoints must use an IPsec tunnel to the Infranet Enforcer to access resources.

To use IPsec with ScreenOS versions earlier than 6.1, you must configure separate IPsec routing policies for different resources that you wish to protect. IPsec routing policies are specific to the resources that you add.

If you are using Screen OS 6.1 or later, you can configure one dynamic IPsec routing policy for each destination zone on each Infranet Enforcer.

In IPsec routing policies with ScreenOS versions earlier than 6.1, you can specify a range of exceptions for traffic to certain resources that you do not want to use IPsec. The exceptions can fall within the ranges of resources that the Infranet Enforcer protects.

In this case, if you create an exception for traffic that flows through the Infranet Enforcer, you must also create another policy on the Infranet Enforcer that allows the exception traffic to flow through.

For example, you might create an IPsec routing policy that uses IPsec for 0.0.0.0/0 (the entire network). In the same policy, you could specify the resources that are exceptions and do not use IPsec, such as 172.24.80.30 (the Infranet Controller), 172.24.80.31 (the Infranet Enforcer), and 172.24.144/21 (a wireless network).
With ScreenOS 6.1 or greater, there is no need for policy exceptions. Dynamic IPsec routing ensures that IPsec tunnels are created based only for destinations that are governed by the ScreenOS IPsec policy.

### Table 13: Before Beginning

<table>
<thead>
<tr>
<th>Topic</th>
<th>Detail</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP address exceptions</td>
<td>Do not use IPsec for the Infranet Controller, the Infranet Enforcer, and networks where your endpoints are located. For example, if you create an IPsec routing policy that uses IPsec on an entire network range (such as 0.0.0.0/0) for your protected resources, be sure to specify exceptions in the same policy for the IP addresses assigned to Infranet Controller, Infranet Enforcer, and the endpoints.</td>
</tr>
<tr>
<td>Using Source IP</td>
<td>You can apply different IPsec routing policies to users on different networks by using source IP restrictions on roles. For configuration instructions, see “Specifying Source IP Access Restrictions” on page 108.</td>
</tr>
<tr>
<td>UDP encapsulation and virtual adapters</td>
<td>For maximum inter-operability with other third-party IPsec clients, select both <strong>Always use UDP encapsulation</strong> and <strong>Always use a virtual adapter</strong>. When both options are selected, NAT is simulated even if a NAT device is not present. Juniper Networks recommends that you select both options or neither option. For example, if an endpoint contains two network interfaces, such as a wired and a wireless interface, and a NAT device is not present between the endpoint and the Infranet Enforcer. If the endpoint doesn’t access a protected resource by using the interface that is connected to the network where the Infranet Enforcer is installed, then the user will not be able to access the protected resource through either interface without a virtual adapter. Because the Infranet Controller does not automatically install a virtual adapter unless a NAT device is detected, enable the <strong>Always use a virtual adapter</strong> option to simulate NAT and force the use of a virtual adapter for this use case.</td>
</tr>
</tbody>
</table>

To configure an IPsec routing policy:

1. In the Infranet Controller admin console, choose **UAC > Infranet Enforcer > IPsec Routing**.
2. Click **New Policy**.
3. On the **New Policy** page:
   a. For **Name**, enter a name to label this IPsec routing policy.
   b. For **Description**, enter an optional description.
4. If you are ScreenOS version 6.1 or later, and you want to configure the Infranet Controller to dynamically provision IPsec routing policies, select the **Dynamic** check box. The Resources and Exceptions text boxes and the Enforcer check boxes disappear.
5. Go to step 10 to continue configuring this policy for ScreenOS 6.1 or greater.
6. For **Resources**, enter the IP address and netmask of each resource that requires endpoints to use IPsec, one per line, in the following format:

\[<\text{ip address}>[/\text{netmask}]\]

You cannot specify a host name in an IPsec routing policy. You must specify an IP address.

7. For **Exceptions**, use the following format, one per line, to specify the IP address and netmask of each resource that has traffic which you do not want to flow through the Infranet Enforcer:

\[<\text{ip address}>[/\text{netmask}]\]

Each exception must be a subset of what you specify for **Resources**.

8. For **Destination Zone**, enter the zone that is configured on the Infranet Enforcer where the protected resources specified in this IPsec routing policy are located. For example: **trust**

9. Select these options to configure IPSec interoperability and tunnel persistence:

- **Always use UDP encapsulation**—This option allows Odyssey Access Client and Infranet Enforcer to create an IPsec tunnel inside a third-party IPsec tunnel by using UDP encapsulation even if a NAT device is not present. For example, for interoperability with third-party IPsec clients running on the endpoint The Infranet Controller uses port 4500 for UDP encapsulation in compliance with RFC 3948.

- **Always use a virtual adapter**—This option forces the use of a virtual adapter on the endpoint. If you select this option, you must also set up IP address pools even if a NAT device is not present. For more information, see "Configuring IP Address Pool Policies for IPsec in a NAT Environment" on page 85.

- **Persistent Tunnel Mode**—This option allows you to determine whether or not a tunnel is established when a user first connects to the Infranet Controller. If the check box is selected, an IPsec tunnel is established, and users can access protected resources behind the Infranet Enforcer. If the check box is not selected, the tunnel is not automatically set up: a tunnel will not be initiated until there is a request for traffic.

10. From the **Enforcer** drop-down list, choose the Infranet Enforcer to which endpoints connect to access the resources specified in this IPsec routing policy.

11. In the **Roles** section, specify:

- **Policy applies to ALL roles**—Choose this option to apply this IPsec routing policy to all users.

- **Policy applies to SELECTED roles**—Choose this option to apply this IPsec routing policy only to users who are mapped to roles in the **Selected roles** list. Be sure to add roles to this list from the **Available roles** list.
Policy applies to all roles OTHER THAN those selected below—Choose this option to apply this IPsec routing policy to all users except for those who map to the roles in the Selected roles list. Be sure to add roles to this list from the Available roles list.

12. Click Save Changes.

Configuring IP Address Pool Policies for IPsec in a NAT Environment

The Infranet Controller and Odyssey Access Client support using IPsec tunnels through Network Address Translation (NAT) devices to enable users to have secure access to protected resources. In a NAT environment, a virtual IP address must be used for the IPsec tunnel’s inner address.

You can configure a pool of virtual IP addresses that you want the Infranet Controller to automatically assign to endpoints by creating IP address pool policies. An IP address pool is a contiguous range of IP addresses which you configure by specifying the starting address and the number of addresses in the pool. You can associate an IP address pool with one or more Infranet Enforcers.

IP address pool policies are required if one of the following applies to your situation:

- You are using IPsec in a NAT environment.
- You selected the Always use a virtual adapter option in an IPsec routing policy to enable inter-operability with other third-party IPsec clients running simultaneously on the endpoint, such as Juniper Network Connect or Microsoft IPsec. For more information, see “Configuring IPsec Routing Policies” on page 82.

If you want to use Network Address Translation (NAT) devices in the Unified Access Control solution, the endpoints must be located on one side of the NAT devices, and the Infranet Controller and Infranet Enforcer must both be located on the other side of the devices.

Also note the following if you are using NAT:

- NAT is not supported between the Infranet Controller and Infranet Enforcer.
- If there is a NAT device between the endpoint and the Infranet Controller, but not between the endpoint and the Infranet Enforcer, source IP enforcement does not work. This is also true if there is a NAT device between the endpoint and the Infranet Enforcer, but not between the endpoint and the Infranet Controller.

If NAT is not detected, Odyssey Access Client uses the endpoint’s physical IP address when creating the IPsec tunnel to the Infranet Enforcer. The Infranet Controller does not allocate an IP address from the pool.

Figure 6 illustrates an example of a NAT environment where endpoints 1 and 2 have the same physical IP address: 192.168.1.1. By using an IP address pool policy, you can configure the Infranet Controller to assign a unique, routable, virtual IP address to each endpoint.
The following sequence of events occur when the Infranet Controller and Odyssey Access Client use an IPsec tunnel through a NAT device:

- When Odyssey Access Client connects to the Infranet Controller and authenticates the user, Odyssey Access Client returns the endpoint’s source IP address that will be used to access the Infranet Enforcer to the Infranet Controller. The Infranet Controller saves the source IP address internally.

- When the user attempts to access a protected resource, an IKE exchange occurs to set up an IPsec tunnel between the endpoint and the Infranet Enforcer.

- During the IKE exchange, the Infranet Enforcer detects the source IP address of the endpoint and sends that IP address to the Infranet Controller.

- The Infranet Controller compares its saved source IP address for the endpoint with the endpoint’s IP address sent from the Infranet Enforcer. If the addresses do not match, the Infranet Controller determines there is a NAT device between the endpoint and the Infranet Enforcer. The Infranet Controller automatically provisions an IP address from an IP address pool policy that you configured (for example, 10.11.0.10-100). The Infranet Controller assigns the IP address to the endpoint based on the IP address pool policy that applies to the user’s role. The Infranet Controller then sends the IP address to the Infranet Enforcer.

- The Infranet Enforcer sends that IP address from the IP address pool (for example, 10.11.0.10) to Odyssey Access Client on the endpoint during the Xauth authentication exchange.

- Odyssey Access Client on the endpoint configures a virtual network adapter using the IP address sent from the Infranet Enforcer.

- The endpoint initiates an IPsec connection to the Infranet Enforcer, and the Infranet Enforcer sets up dynamic routes for each IP address. The user can now use the endpoint to access protected resources.
The Infranet Controller allocates one IP address for the duration of each Odyssey Access Client session, which lasts as long as Odyssey Access Client is connected to the Infranet Controller. After a session ends, the Infranet Controller may reuse the allocated address for a different endpoint.

When the Infranet Controller needs to provide an inner IP address for a new IPsec tunnel, it attempts to reuse an existing inner IP address assigned to the endpoint before allocating a new one. The Infranet Controller checks all of the inner IPs allocated from IP address pools for the endpoint. For each IP address, the Infranet Controller determines whether the policy from which that address was allocated applies to the user and the Infranet Enforcer for the new IPsec tunnel. If a compliant IP address is found, it is used and no new IP address is allocated. If no compliant IP address is found, a new IP address is allocated.

**Task Summary: Configuring an IP Address Pool Policy**

<table>
<thead>
<tr>
<th>Topic</th>
<th>Detail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiple Infranet Enforcers</td>
<td>If you are using multiple Infranet Enforcers across a LAN, make sure that the IP address pool contains addresses that are valid for each Infranet Enforcer.</td>
</tr>
<tr>
<td>Multiple unclustered Infranet Controllers</td>
<td>If you are using multiple unclustered Infranet Controllers across a LAN, make sure that the IP address pool contains addresses that are unique for each Infranet Controller. The endpoint is assigned an virtual IP address for each unclustered Infranet Controller to which Odyssey Access Client connects.</td>
</tr>
<tr>
<td>IP address conflicts</td>
<td>Make sure that the IP pool addresses do not conflict with addresses of hosts that the endpoint might access.</td>
</tr>
<tr>
<td>Deleting IP addresses in an IP pool</td>
<td>If you change or delete the IP addresses in an IP address pool, you must delete all user sessions in order to stop using those addresses. To delete all user sessions, click <strong>Delete All Sessions</strong> on the <strong>System &gt; Status &gt; Active Users</strong> page of the Infranet Controller admin console.</td>
</tr>
<tr>
<td>Number of available IP addresses</td>
<td>Be sure to specify a sufficient number of addresses in the IP address pool for all of the endpoints in your deployment. When all of the addresses in the pool have been assigned to endpoints, additional endpoints are unable to obtain a virtual IP address and are blocked from accessing protected resources. The Infranet Controller logs a message in the Event log when an IP address cannot be assigned to an endpoint.</td>
</tr>
</tbody>
</table>

To configure an IP address pool policy:

1. In the Infranet Controller admin console, choose **UAC > Infranet Enforcer > IP Address Pools**.
2. Click **New Policy**.
3. On the **New Policy page**:
   a. For **Name**, enter a name to label this IP address pool policy.
b. For **Description**, enter an optional description.

4. For **IP address pool**, specify IP addresses or a range of IP addresses for the Infranet Controller to assign to endpoints. The IP address range can be specified as shown in Table 15 where the last component of the IP address is a range delimited by a hyphen (-). No special characters are allowed.

<table>
<thead>
<tr>
<th>IP address range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>a.b.c.d</td>
<td>Specifies a single IP address</td>
</tr>
<tr>
<td>a.b.c.d-e.f.g.h</td>
<td>Specifies all IP addresses from the first address to the last address, inclusive</td>
</tr>
<tr>
<td>a.b.c.d-f.g.h</td>
<td>An abbreviated form that specifies the range a.b.c.d through a.f.g.h</td>
</tr>
<tr>
<td>a.b.c.d-g.h</td>
<td>An abbreviated form that specifies the range a.b.c.d through a.b.g.h</td>
</tr>
<tr>
<td>a.b.c.d-h</td>
<td>An abbreviated form that specifies the range a.b.c.d through a.b.c.h</td>
</tr>
<tr>
<td>a.b.c.d[mask]</td>
<td>Specifies all addresses in a network</td>
</tr>
</tbody>
</table>

For example, to allocate all addresses in the range 172.20.0.0 through 172.20.3.255, specify 172.20.0.0-3.255. Or, to allocate all addresses in a class C network, specify 10.20.30.0/24.

5. Under **Infranet Enforcer**, select the Infranet Enforcer(s) to which you want to apply this IP address pool policy and click **Add**. To apply the policy to all Infranet Enforcers, do not add any Infranet Enforcers and leave the default setting (all) listed in the **Selected Enforcers** list.

6. In the **Roles** section, specify:

- **Policy applies to ALL roles**—Choose this option to apply this IP address pool policy to all users.

- **Policy applies to SELECTED roles**—Choose this option to apply this IP address pool policy only to users who are mapped to roles in the **Selected roles** list. Be sure to add roles to this list from the **Available roles** list.

- **Policy applies to all roles OTHER THAN those selected below**—Choose this option to apply this IP address pool policy to all users except for those who map to the roles in the **Selected roles** list. Be sure to add roles to this list from the **Available roles** list.

7. Click **Save Changes**.

**Enabling NAT-src on the Infranet Enforcer**

If the IP addresses you specify in the IP address pool policies (that is, the virtual IP addresses) are not routable from the network where your protected resources are located, make sure you enable Source Network Address Translation (NAT-src) on the infranet-auth tunnel policies that configure IPsec on the Infranet Enforcer(s).

To enable NAT-src using the Infranet Enforcer WebUI:

1. Click **Policies**.
2. Click Edit on the infranet-auth tunnel policy.

3. Click Advanced.

4. Select Source Translation and click OK.

For information on enabling NAT-src on the infranet-auth tunnel policy, see the Concepts & Examples ScreenOS Reference Guide: Volume 8, Address Translation or the ScreenOS CLI Reference Guide, which you can download from www.juniper.net/techpubs/.

**Configuring Source Interface Policies**

If you want endpoints to use IPsec to communicate with an Infranet Enforcer that is in Transparent mode, in some cases you may need to configure a source interface policy on the Infranet Controller. A source interface policy specifies the source interface on the Infranet Enforcer that receives traffic from endpoints. For information on Transparent mode, see “Setting Up an Infranet Enforcer in Transparent Mode” on page 24.

The use cases for configuring source interface policies are limited. You will need to use a source interface policy if you have multiple virtual routers, AND you have an IPsec routing policy with destination zone DEST, and one of the following is true:

- There are multiple IPsec policies on the enforcer with a destination zone DEST and different source zones.
- There is an IPsec policy on the enforcer with a destination zone DEST whose source zone has multiple interfaces.

For more information on how to set up a virtual router in the Infranet Enforcer, see the “Routing” volume of the Concepts & Examples ScreenOS Reference Guide, which you can download from www.juniper.net/techpubs/.

**NOTE:** You can apply different source interface policies to users on different networks by using source IP restrictions on roles. For configuration instructions, see “Specifying Source IP Access Restrictions” on page 108.

To configure a source interface policy:

1. In the Infranet Controller admin console, choose UAC > Infranet Enforcer > IPsec Routing.

2. Select the check box for **Always show source interface policies**. The Save Changes button begins to blink.

3. Click **Save Changes**. The Source Interface tab appears at the top of the page.

4. Select the **Source Interface** tab.

5. Click **New Policy**.
6. On the **New Policy page**:
   a. For **Name**, enter a name to label this source interface policy.
   b. For **Description**, enter an optional description.

7. From the **Enforcer** drop-down list, choose the Infranet Enforcer to which endpoints connect.

8. For **Source Interface**, specify the interface on the Infranet Enforcer to which traffic from endpoints connects. This is the default interface for the zone you want to build a gateway for, not the interface name. To view the zone table on the Infranet Enforcer, enter the following command:

   `get zone`

9. In the **Roles** section, specify:

   - **Policy applies to ALL roles**—Choose this option to apply this source interface policy to all users.
   - **Policy applies to SELECTED roles**—Choose this option to apply this source interface policy only to users who are mapped to roles in the **Selected roles** list. Be sure to add roles to this list from the **Available roles** list.
   - **Policy applies to all roles OTHER THAN those selected below**—Choose this option to apply this source interface policy to all users except for those who map to the roles in the **Selected roles** list. Be sure to add roles to this list from the **Available roles** list.

10. Click **Save Changes**.

### Upgrading IPsec Enforcement Policies from Version 1.x of the Infranet Controller

If you upgraded your Infranet Controller from a version 1.x to 2.0 or later, be aware that the Infranet Controller does not automatically upgrade any previously configured IPsec enforcement policies. You must manually reconfigure all IPsec enforcement policies by using the following instructions.

To manually reconfigure IPsec enforcement policies from version 1.x of the Infranet Controller:

1. In the Infranet Controller admin console, choose **UAC > Infranet Enforcer > IPsec Routing**. Note the Infranet Enforcer and destination zone specified in each previously configured IPsec routing policy.

2. While the Infranet Enforcer is connected to the Infranet Controller, navigate to the **UAC > Infranet Enforcer > Connection** page.

3. Click the name in the **Enforcer** column of the Infranet Enforcer on which you want to configure IPsec enforcement.

4. Click the **ScreenOS Policies** tab.

5. On the **ScreenOS Policies** page, create a new IPsec enforcement policy for each IPsec routing policy and specify the:
Source zone (the zone on which end-points are connected)

Destination zone (as specified in each IPsec routing policy)

6. If necessary, use the ScreenOS Web UI or CLI to customize the new IPsec enforcement policy.

7. Configure IP address pool policies if you are using IPsec in a NAT environment, or if the endpoints are simultaneously running other third-party IPsec clients such as Juniper Network Connect or Microsoft IPsec. For more information, see “Configuring IP Address Pool Policies for IPsec in a NAT Environment” on page 85.

Setting Up Source IP Enforcement on the Infranet Enforcer

You can use source IP enforcement (clear text) on the Infranet Enforcer to protect resources alone, or with IPsec enforcement.

To use source IP enforcement, you configure a ScreenOS infranet-auth policy on the Infranet Enforcer. Infranet-auth policies control which zones use Infranet Enforcer resource access policies to allow or deny traffic (see “Configuring Infranet Enforcer Resource Access Policies” on page 72). For example, you can configure an infranet-auth policy to enforce resource access policies on traffic from the Untrust zone to the Trust zone.

You can configure source IP enforcement by using the Infranet Controller to run the appropriate CLI commands to configure basic infranet-auth policies on the Infranet Enforcer, or, you can manually configure infranet-auth policies on the Infranet Enforcer using ScreenOS CLI commands or WebUI. Juniper Networks recommends that you use the Infranet Controller to set up source IP enforcement on the Infranet Enforcer. If necessary, you can modify the infranet-auth policies afterwards on the Infranet Enforcer by using the ScreenOS CLI commands described in this section. The Infranet Controller allows you to set basic infranet-auth policies from a source to destination zone. If you want to set more complex infranet-auth policies, such as between IP address ranges, set the policies on the Infranet Enforcer.

Before setting a policy using the CLI, you must create address book entries for the destination and source addresses unless you use address book entries that already exist, such as Any.

In the following example, you set an infranet-auth policy and add it to the top of the list of policies using Any. The policy allows all traffic of any type from any host to another host. The following policy allows traffic according to the Infranet Enforcer resource access policies that you configure on the Infranet Controller (see “Configuring Infranet Enforcer Resource Access Policies” on page 72).

```
set policy top from untrust to trust any any any permit infranet-auth
```

In the following example, you create two address book entries and a policy between them for anyone in the 10.64.0.0/16 range to be able to reach the 10.65.0.0/16 range.

```
set address Trust “10.64 Range” 10.64.0.0 255.255.0.0
set address Untrust “10.65 Range” 10.65.0.0 255.255.0.0
set policy from trust to untrust “10.64 Range” “10.65 Range” any permit infranet-auth
```
Task Summary: Configuring Source IP Enforcement

To configure source IP enforcement:

1. The Infranet Enforcer must be connected to the Infranet Controller before you can use the Infranet Controller to set up source IP enforcement on the Infranet Enforcer. On the UAC > Infranet Enforcer > Connection page, click New Enforcer to configure a connection to the Infranet Enforcer on which you want to configure source IP enforcement. For more information, see “Configuring the Infranet Controller to Connect to the Infranet Enforcer” on page 20.

2. On the UAC > Infranet Enforcer > Connection page, click the name in the Enforcer column of the Infranet Enforcer on which you want to configure source IP enforcement. A new page with the ScreenOS Policies tab appears.

3. On the UAC > Infranet Enforcer > ScreenOS Policies page:
   a. Choose the source zone for the policy from the Source Zone drop-down list. The source zone is the zone where the endpoint is located.
   b. Choose the destination zone for the policy from the Destination Zone drop-down list. The destination zone is where the protected resources are located.
   c. Choose Source IP from the Type drop-down list.
   d. Click Add.

   The Infranet Controller creates an infranet-auth policy on the Infranet Enforcer that specifies a source and destination zone that allows all traffic of any type between the specified source and destination zones from any host to another host.

4. Click Save Changes to save the policy.

5. On the UAC > Infranet Enforcer > Resource Access page, configure Infranet Enforcer resource access policies to specify which users are allowed or denied access to a set of protected resources.

   The policy allows traffic to the protected resources specified in the Infranet Enforcer resource access policies that you configure on the Infranet Controller (see “Configuring Infranet Enforcer Resource Access Policies” on page 72).

Dynamic Auth Table Allocation

Prior to ScreenOS release 6.1, you manually created auth table mapping policies to use Source IP enforcement. Each authenticated user had an auth table entry on the Infranet Enforcer, whether they were accessing resources or not. With ScreenOS 6.1 the Infranet Enforcer can dynamically create auth table entries when a user attempts to access a protected resource, eliminating the need to manually create Auth Table Mapping Policies.
If you are using ScreenOS release 6.1 or higher, you can configure the Infranet Enforcer to dynamically provision auth table entries for users. Auth table entries are required to match network requests from users to resource access policies through the Infranet Controller.

Dynamic auth table allocation reduces auth table entries to only those that are needed, enabling you to deploy smaller firewalls with a larger user population.

After the user has disconnected, the Infranet Enforcer automatically expires the auth table entry.

Dynamic auth table allocation does not work with http traffic if the Infranet Enforcer’s captive portal feature is configured to redirect user traffic to an external web server other than the Infranet Controller.

If you permit dynamic auth table provisioning on the Infranet Controller, and the DNS server for your network is behind the Infranet Enforcer, endpoints may occasionally experience DNS time-out issues before resources are provisioned.

To permit dynamic auth table allocation:

1. From the UAC > Infranet Enforcer > Auth Table Mapping page on the admin console, delete the Default Policy, or specify an Infranet Enforcer for which you do not want to configure this feature.

2. Click **Save Changes**.

3. You must configure a source IP policy for all traffic, whether source IP or IPsec. See “Task summary: Configuring IPsec Enforcement” on page 80.

### Configuring Auth Table Mapping Policies for Source IP Enforcement

**NOTE:** If you are using ScreenOS version 6.1, and have permitted dynamic auth table allocation, you do not need to configure auth table mapping policies. See “Dynamic Auth Table Allocation” on page 92.

The Infranet Controller configuration includes one default auth table mapping policy. When the default auth table mapping policy is enabled, the Infranet Controller pushes one auth table entry for each authenticated user to all Infranet Enforcer devices connected to the Infranet Controller. An auth table entry consists of the user’s name, a set of roles, and the IP address of the wired adapter, wireless adapter, or virtual adapter in the user’s computer. If your deployment includes a mixture of low and high capacity Infranet Enforcer devices, the lower capacity Infranet Enforcer devices may reach the limit of concurrent auth table entries and prevent additional users from accessing protected resources behind the lower capacity Infranet Enforcer devices.

Figure 7 on page 94 illustrates an example of a deployment that includes a higher capacity Infranet Enforcer ISG 2000 and a lower capacity 5GT in two branch offices. If the default auth table mapping policy is enabled and the number of users that attempt to access protected resources behind the ISG 2000 in Branch Office 1 exceeds the limit of concurrent auth table entries on the 5GT, then additional users are unable to access protected resources behind the 5GT in Branch Office 2.
You can prevent overloading the lower capacity Infranet Enforcer devices in mixed deployments by deleting the default auth table mapping policy and creating new policies. An **auth table mapping** policy specifies which Infranet Enforcer device(s) each user role is allowed to use when the endpoint is using source IP enforcement. These policies prevent the Infranet Controller from creating unnecessary auth table entries on all connected Infranet Enforcer devices. In the example in Figure 7, auth table entries for users assigned the Branch1-Role are mapped to the ISG 2000 in Branch Office 1 only, which avoids overloading the ISG 5GT in Branch Office 2. Likewise, the auth table entries for users assigned the HQ-Role are mapped to the ISG 2000 in HQ Office only, and auth table entries for users assigned the Branch2-Role are mapped to the ISG 5GT in Branch Office 2 only.

![Figure 7: Using Auth Table Mapping Policies](image)

You can also use auth table mapping policies to restrict users from accessing resources behind an Infranet Enforcer based on the user’s assigned role.

If your deployment does not use source IP enforcement, you do not need to create auth table mapping policies. For IPsec enforcement, the Infranet Controller pushes auth table entries only to the Infranet Enforcer devices you specify in IPsec routing policies. If you are using a combination of source IP enforcement and IPsec enforcement, you only need to create auth table mapping policies for the endpoints that use source IP enforcement.

Auth table mapping policies apply to Odyssey Access Client and agentless deployments.

To configure auth table mapping policies:

1. In the Infranet Controller admin console, choose **UAC > Infranet Enforcer > Auth Table Mapping**.
2. Select the default auth table mapping policy called Default Policy and click **Delete**. The Infranet Controller includes this default auth table mapping policy that allows all source IP endpoints to use all Infranet Enforcer devices. Make sure you delete the Default Policy if you configure any of your own auth table mapping policies. Click **New Policy**.

3. On the **New Policy** page:
   a. For **Name**, enter a name to label this auth table mapping policy.
   b. For **Description**, enter an optional description.
   c. In the **Infranet Enforcer** section, specify the Infranet Enforcer device(s) to which you want to apply this auth table mapping policy.
   d. In the **Roles** section, specify:
      - **Policy applies to ALL roles**—Choose this option to apply this auth table mapping policy to all users.
      - **Policy applies to SELECTED roles**—Choose this option to apply this auth table mapping policy only to users who are mapped to roles in the **Selected roles** list. Be sure to add roles to this list from the **Available roles** list.
      - **Policy applies to all roles OTHER THAN those selected below**—Choose this option to apply this auth table mapping policy to all users except for those who map to the roles in the **Selected roles** list. Be sure to add roles to this list from the **Available roles** list.
   e. In the **Action** section, specify auth table mapping rules for the specified Infranet Enforcer device(s):
      - **Always Provision Auth Table**—Choose this option to automatically provision auth table entries for chosen roles on the specified Infranet Enforcer.
      - **Provision Auth Table as Needed**—Choose this option to provision auth table entries only when a user with a chosen role attempts to access a resource behind the specified Infranet Enforcer.
      - **Never Provision Auth Table**—Select this option to prevent chosen roles from accessing resources behind the specified Infranet Enforcer.

4. Click **Save Changes**.

**Changing the Number of Auth Table Entries on the Infranet Enforcer**

The number of authenticated user sessions on the Infranet Controller corresponds to the number of auth table mapping policies. Each auth table entry holds the authentication information for a single user. You can change the platform-specific preconfigured auth table entries to allocate additional auth table entries with the appropriate CLI command.
This command is only supported on Infranet Enforcers that are using ScreenOS version 6.0 or higher. This command is hidden in the Infranet Enforcer CLI.

The maximum number of authentication table entries per platform is 50,000 for ISG-1000, NS2000 and NS5XXX firewalls, and 10,000 for all other firewalls.

The default number of authentication tables for various platforms is as follows:
NS5XXX = 12,228
NS2000 = 12,228
ISG1000 = 8,192
All other platforms = 4,096

In the following example, the number of authentication table entries is increased to 50,000 on the Infranet Enforcer.

```
set auth table count 50000
```

The value of the entry must be an integer between the default number allowed and the platform-specific maximum.

Use the following command to reset the count to the default setting. This means that the firewall will revert to using the platform-specific maximum preconfigured authentication table entries.

```
unset auth table count
```

You can display the currently configured auth table count.

```
get auth table count
```

This command does not return any output if the authentication table count is set to the default value.

Changes in the auth table count take effect after the Infranet Enforcer is rebooted.

---

Introduction to Network Access Policies

Network access policies are RADIUS attributes policies associated with Layer 2 802.1X deployments at the edge of the network. Layer 2 network access policies allow the Infranet Controller to work with external network access devices to provide access management and control.

- **RADIUS attributes policy**—This type of policy specifies the VLAN that endpoints must use to access the network, and the return list attributes to send to an 802.1X network access device. See “Configuring RADIUS Attributes Policies” on page 151.

In addition to network access policies, you can configure location groups and RADIUS clients for 802.1X.

- **Location groups**—Location groups organize network access devices by associating the devices with specific sign-in policies. See “Configuring Location Groups” on page 145 for more information.
Host Enforcer Policies

Host Enforcer is a stateful packet filter that is built into the Odyssey Access Client. You configure Host Enforcer policies on the Infranet Controller.

NOTE:

- Odyssey Access Client is required for Host Enforcer. Host Enforcer is not supported on Java agent, agentless endpoints, or non-Juniper supplicants.

- If you delete the Default Global Rules policy and do not create any additional policies, Host Enforcer allows the types of traffic specified by the internal rules.

- Juniper Networks recommends that you configure Host Enforcer policies to allow the specific traffic on endpoints before you enable the Host Enforcer option on a role to avoid access problems on the endpoints.

- Avoid creating Host Enforcer policies and Infranet Enforcer resource access policies that conflict with one another. For example, don’t create a Host Enforcer policy that denies access to resources and an Infranet Enforcer policy that allows access. For more information, see “Configuring Infranet Enforcer Resource Access Policies” on page 72.

- If a user is assigned multiple roles and each role has different Host Enforcer policies, the Infranet Controller pushes all of the Host Enforcer policies that apply to the user to the endpoint. However, the policies are evaluated in the order specified in the list in the Host Enforcer Policies page, and the allow or deny action is enforced for the first policy in the list that matches the resource and user’s role(s).

You can use the Host Enforcer feature to protect Windows endpoints and enforce access policies on the endpoint itself. The Host Enforcer is especially useful in situations where an Infranet Enforcer is not deployed in front of resources that you want to protect.

To use the Host Enforcer feature, you enable the Host Enforcer option on a role. Odyssey Access Client protects endpoints and resources by only allowing the incoming and outgoing traffic on the endpoints that you specify in the Host Enforcer policies for that role, and the traffic allowed by default. For example, you can create a Host Enforcer policy that denies all incoming TCP connections, and denies all outgoing TCP traffic with a few exceptions. The Host Enforcer policy also only allows users access to the resources you specify.
The Host Enforcer provides these security functions:

- Protects a Windows endpoint from attacks from other endpoints and systems by allowing only the incoming and outgoing traffic you specify in the Host Enforcer policies for a role, and the traffic allowed by default.

- Allows access to the resources you specify in the Host Enforcer policies for a role. This feature is useful if you want to protect resources that are not already protected by the Infranet Enforcer.

If you enable the Host Enforcer option on a role (see “Configuring Access Options on a Role” on page 126) *and* you do not specify any Host Enforcer policies for the role, Odyssey Access Client *denies* all traffic on the endpoint *except* for the following types of traffic that are either specified internally or explicitly allowed in the *Default Global Rules* policy.

The following types of traffic are allowed by default Host Enforcer internal rules. You cannot override these rules:

- TCP traffic on port 443 to and from each connected Infranet Controller

- UDP (IKE) traffic on port 500 to each Infranet Enforcer configured by the Infranet Controller

- UDP traffic on port 4500 to each configured Infranet Enforcer. This traffic is used for NAT traversal

- WINS, DNS, and DHCP traffic

- ICMP Echo Request (Send) and ICMP Echo Reply (Receive)—You can send ping requests from endpoints to other hosts, and receive ping replies from other hosts back to endpoints for troubleshooting. That is, the endpoint can ping other hosts. Incoming echo requests and outgoing echo replies are blocked on the endpoint; other hosts cannot ping the endpoint

- ICMP Destination Unreachable (Receive only), ICMP Source Quench (Receive only), ICMP Time Exceeded (Receive only), ICMP Traceroute (Send only), ICMP Mobile IP Reg. Request (Send only), and ICMP Mobile IP Reg. Reply (Receive only).

In addition to the traffic allowed by the internal rules, the following types of traffic are allowed by the Host Enforcer *Default Global Rules* policy for all roles. You can change, delete, or override this policy.

- All ICMP and ESP traffic

- Outgoing TCP traffic on all ports

- Outgoing UDP traffic on ports 137 and 138

Since only these types of traffic are allowed by default, it’s important that you specify the additional traffic you want to *allow* on each endpoint for a particular role or for all roles. For example, it’s important that you configure Host Enforcer policies to specify the incoming TCP traffic that you want to allow.
You can specify the additional allowed traffic by changing the Default Global Rules policy or creating new Host Enforcer policies for particular roles. If you create a new policy, the Infranet Controller positions it at the top of the policy list in the Host Enforcer Policies page to allow your Host Enforcer policy settings to override the Default Global Rules policy or other policies below it. You can use the arrow buttons on the Host Enforcer Policies page to rearrange the Host Enforcer policies in the order of priority you want them enforced.

You can also specify the traffic you want to deny on an endpoint. For example, you can specify a policy that denies outgoing TCP traffic for a particular role, and then use the Default Global Rules policy placed below that policy to allow outgoing TCP traffic on all other roles. All of the Host Enforcer policies that apply to the current user’s role(s) are enforced on the endpoint.

**NOTE:**
- The Infranet Controller downloads the Host Enforcer policies to the endpoint after the user signs in and is authenticated for the first time. If you change the Host Enforcer policies for the role while the user is signed in, the Infranet Controller pushes the updated Host Enforcer policies to the endpoint. Updates to the Host Enforcer policies only occur while the user is signed in.
- Be sure to avoid creating rules that block traffic to the protected resources behind the Infranet Enforcer(s).
- If the endpoint becomes disconnected from the Infranet Controller and Odyssey Access Client is still running, the Host Enforcer policies are no longer enforced on the endpoint.
- You can enable the Require connection to this Infranet Controller option in the Odyssey Configuration page to require an endpoint running Odyssey Access Client to connect to a particular Infranet Controller so that you can require enforcement of the Host Enforcer policies configured on that Infranet Controller. For more information, see “Creating an initial Configuration of Odyssey Access Client” on page 44.

To configure a Host Enforcer policy:

1. In the Infranet Controller admin console, choose UAC > Host Enforcer.
2. Click New Policy.
3. On the New Policy page:
   a. For Name, enter a name to label this Host Enforcer policy.
   b. For Description, enter an optional description.
4. For **Resources**, specify the traffic you want to allow or deny on the endpoints where this Host Enforcer policy applies to roles, one rule per line using the following syntax:

\[
<\text{protocol}>://<\text{host}>[://<\text{net-mask}>]:<\text{DestinationPorts}>[://<\text{SourcePorts}>]
\]

where:

- **protocol** is either `<ProtocolNumber>` or `<ProtocolText>[''_'<Direction>]`

  where:

  - **ProtocolNumber** is the IP protocol number. For example, the protocol number for UDP is 17.
  - **ProtocolText** can be any of the following protocols specified as a text string: TCP, ICMP, UDP, or ESP

  If you do not specify a protocol, the rule applies to all of the allowed protocols. Table 16 contains some examples of specifying a protocol.

- **Direction** is the direction of the traffic and specified as one of these two text strings: in or out. You can specify the traffic direction for TCP or UDP only. For example: tcp_out. If you do not specify a direction, for example, tcp, the rule applies to both inbound and outbound traffic.

- **host** is the IP address of the remote host. If you do not specify an IP address, the rule applies to all IP addresses. You cannot specify a host name in a Host Enforcer policy. You can only specify an IP address.

- **net-mask** is the net-mask number or address of the remote host. If you do not specify a net-mask, the rule applies to a single IP address.

- **DestinationPorts** is a range or comma-delimited list of the destination ports for the outgoing traffic.

- **SourcePorts** is a range or comma-delimited list of the source ports of the incoming traffic. If you do not specify the source ports, the rule applies to all ports.

**Table 16: Examples of Specifying Resources in a Host Enforcer Policy**

<table>
<thead>
<tr>
<th>Specify this protocol</th>
<th>To allow</th>
</tr>
</thead>
<tbody>
<tr>
<td>tcp_out://*:21,80,443</td>
<td>Outgoing TCP traffic on ports 21, 80, and 443 only</td>
</tr>
<tr>
<td>tcp_in://10.11.0.0/255.255.0.0:*:20</td>
<td>Incoming FTP traffic from 10.11.0.0/255.255.0.0 on FTP server port 20 to all ports on the endpoint</td>
</tr>
<tr>
<td>udp_in://*:</td>
<td>Incoming UDP traffic from all IP addresses to all ports on the endpoint</td>
</tr>
<tr>
<td>icmp://*:</td>
<td>Incoming and outgoing ICMP traffic from all IP addresses to all ports on the endpoint</td>
</tr>
</tbody>
</table>
5. In the Roles section, specify:

- **Policy applies to ALL roles**—Choose this option to apply this Host Enforcer policy to all users.

- **Policy applies to SELECTED roles**—Choose this option to apply this Host Enforcer policy only to users who are mapped to roles in the Selected roles list. Be sure to add roles to this list from the Available roles list.

- **Policy applies to all roles OTHER THAN those selected below**—Choose this option to apply this Host Enforcer policy to all users except for those who map to the roles in the Selected roles list. Be sure to add roles to this list from the Available roles list.

6. Under Action, choose whether you want this policy to allow or deny the traffic you specified for Resources. For example, you can create a policy that denies outgoing TCP traffic for a particular role.

7. Click **Save Changes**.

---

**NOTE:**

- If you haven’t already done so, enable the Host Enforcer option on a role (see “Configuring Access Options on a Role” on page 126).

- To copy a Host Enforcer policy as a starting point for a new policy that you want to create, select the policy on the Host Enforcer Policies page and then click Duplicate.

---

**Policies, Rules & Restrictions, and Conditions Evaluation**

The flowcharts in this section show the transactions that take place between a user and the Infranet Controller and the Infranet Controller and an authentication server or Infranet Enforcer. The flowcharts begin with a user signing in, and end with the user ending the user session by exiting Odyssey Access Client (Windows) or signing out (agentless access deployments).
Figure 8: Infranet Controller Authenticates User Against Realm and Primary Server

**USER**
- User signs into Infranet Agent (Windows) or enters URL in browser (non-supported platforms)

**Infranet Controller (IC)**
- IC evaluates pre-auth. restrictions
  - Determines if user’s system passes host checks and other pre-authentication requirements.

**Auth Server**
- Pre-authentication realm checks passed?
  - YES
    - IC evaluates post-auth. restrictions
      - Determines whether user’s password conforms to specified limits and whether their system passes other post-authentication realm requirements.

**USER**
- User submits credentials
  - IC allows user to submit username and password for realm whose pre-authentication checks succeeded, if >1 realm exists, enters realm or chooses realm from list. If required by realm, enters two sets of credentials.

**Infranet Controller (IC)**
- Post-authentication realm checks passed?
  - NO
    - User denied access to IC
  - YES
    - IC forwards user’s primary credentials to realm’s primary authentication server

**Auth Server**
- Authentication server returns “success” or “failure”
  - A RADIUS or TACACS+ authentication server also returns attributes for the ID to use in role mapping.

**USER**
- User denied access to IC
  - NO
    - Authentication success?
  - YES
Figure 9: Infranet Controller Authorizes User

USER → Infranet Controller (IC) → Auth Server

Does realm have separate LDAP authorization server?

IC stores user’s primary credentials

IC queries LDAP

IC stores user’s group information

IC evaluates realm role mapping rules

User information from LDAP or RADIUS server or the user’s certificate to determine eligible user roles.

IC pushes policies

IC creates “session user role”

IC enforces role restrictions

Determined valid user roles for users who meet requirements or role restrictions, which may include:
- Source IP
- Browser type
- Client-side Certificate
- Host Certificate

IC pushes authentication table entries and policies to Infranet Enforcer and Odyssey Access Client

- The Infranet Controller pushes authentication table entries specifying the user and applicable roles to the Infranet Enforcer.
- The Infranet Controller pushes time-routing policies for the roles to the Odyssey Access Client to specify firewall connections.
- If the Host Enforcer option is activated on the role, the Infranet Controller pushes Host Enforcer access policies for the roles to the Odyssey Access Client. Note: The Infranet Controller pushes Host Enforcer access policies to the Infranet Enforcer during the initial connection between the two devices.

Figure 10: Infranet Controller Maps User to One or More User Roles and Pushes Policies

USER → Infranet Controller (IC) → Infranet Enforcer or Odyssey Access Client

IC enforces role restrictions

Determined valid user roles for users who meet requirements or role restrictions, which may include:
- Source IP
- Browser type
- Client-side Certificate
- Host Certificate

IC creates “session user role”

Merges valid roles to determine session permissions; allowed resources from each valid role are granted to the user. If the IC is configured not to merge roles, then the IC assigns the user to the first role to which the user is mapped.

IC pushes authentication table entries and policies to Infranet Enforcer and Odyssey Access Client

- The Infranet Controller pushes authentication table entries specifying the user and applicable roles to the Infranet Enforcer.
- The Infranet Controller pushes time-routing policies for the roles to the Odyssey Access Client to specify firewall connections.
- If the Host Enforcer option is activated on the role, the Infranet Controller pushes Host Enforcer access policies for the roles to the Odyssey Access Client. Note: The Infranet Controller pushes Host Enforcer access policies to the Infranet Enforcer during the initial connection between the two devices.
Figure 11: Odyssey Access Client and Infranet Enforcer Evaluate Policies Based on User Roles (Windows Only)

1. User requests a resource

2. Is Host Enforcer enabled on the user’s role for the selected resource?
   - NO
   - YES
     - Is a Host Enforcer policy defined on the user’s role for the selected resource?
       - NO
       - YES
         - Odyssey Access Client evaluates Host Enforcer policies
           - The Odyssey Access Client on the endpoint evaluates Host Enforcer policies related to the user’s request, sequentially processing each policy until it finds the policy whose resource list and designated roles match the user’s request.
           - The Odyssey Access Client then sends a request to the resource via the protocol specified in the Host Enforcer policy.

3. Is PAC enabled on the user’s role for the selected resource?
   - NO
   - YES
     - LAN
     - Odyssey Access Client uses PAC to connect to Infranet Enforcer

4. Infranet Enforcer evaluates Infranet Enforcer policies
   - The Infranet Enforcer evaluates Infranet Enforcer policies related to the user’s request, sequentially processing each policy until it finds the policy whose resource list and designated roles match the user’s request.
   - If a policy is found, the Infranet Enforcer then performs the action specified—allow or deny access to the resource.

5. User denied access to resource

6. User accesses resource or application server

7. *At any point during a user session, the IC may force the user out if the user session reaches the maximum session length, or if dynamic policy evaluation is enabled and the user fails a policy.*

*Signs out of IC by exiting the Odyssey Access Client*
Dynamic policy evaluation allows you to automatically or manually refresh the assigned roles of users by evaluating a realm’s authentication policy, role mappings, role restrictions, and resource policies. When the Infranet Controller performs a dynamic evaluation, it checks whether the client’s status has changed. (For instance, the client’s Host Checker status may have changed. Or, if the user is roaming, the computer’s IP address may have changed.) If the status has changed, the Infranet Controller enables or denies the user access to the dependent realms, roles, or resource policies accordingly.

**NOTE:** The Infranet Controller does not check for changes in user attributes from a RADIUS, LDAP, or SiteMinder server when performing dynamic policy evaluation. Instead, the Infranet Controller re-evaluates rules and policies based on the original user attributes that it obtained when the user signed into the Infranet Controller.

This section contains the following information about dynamic policy evaluation:

- “Understanding Dynamic Policy Evaluation” on page 106
- “Understanding Standard Policy Evaluation” on page 106
- “Enabling Dynamic Policy Evaluation” on page 106
Understanding Dynamic Policy Evaluation

If the Infranet Controller determines after a dynamic policy evaluation that a user no longer meets the security requirements of a role, the Infranet Controller terminates the connection immediately with the user. The user must take the necessary steps to meet the security requirements of the role, and then sign into the Infranet Controller again.

The Infranet Controller logs information about policy evaluation and changes in roles or access in the Event log.

Understanding Standard Policy Evaluation

If you do not use dynamic policy evaluation, the Infranet Controller evaluates policies and roles only when the following events occur:

- When the user first tries to access the Infranet Controller sign-in page, the Infranet Controller evaluates the Host Checker policies (if any) for a realm.
- Immediately after the user’s initial authentication, the Infranet Controller evaluates the user’s realm restrictions in the authentication policy, role mapping rules, and role restrictions.
- Whenever the user makes a request for a resource, the Infranet Enforcer evaluates resource access policies to determine if the associated role is allowed to access the resource.
- Whenever the Host Checker status of the user’s machine changes, the Infranet Controller evaluates the Host Checker policies (if any) for a role.

If you do not use dynamic policy evaluation and you make changes to an authentication policy, role mapping rules, role restrictions, or resource policies, the Infranet Controller enforces those changes only when the events described above occur. (For more information, see “Policies, Rules & Restrictions, and Conditions Evaluation” on page 101.)

If you do use dynamic policy evaluation, the Infranet Controller enforces changes when the events described above occur and it also enforces changes at the times you specify. For more information, see “Enabling Dynamic Policy Evaluation” on page 106.

Enabling Dynamic Policy Evaluation

You can use dynamic policy evaluation in the following ways:

- Evaluate all signed-in users in a realm—You can automatically or manually refresh the roles of all currently signed-in users of a realm by using the General tab of the Administrators > Admin Realms > Select Realm or Users > User Realms > Select Realm page. You can trigger the Infranet Controller to perform a dynamic policy evaluation at the realm level based on:
Chapter 3: General Access Management

An automatic timer—You can specify a refresh interval that determines how often the Infranet Controller performs an automatic policy evaluation of all currently signed-in realm users, such as every 30 minutes. When using the refresh interval, you can also fine-tune Infranet Controller performance by specifying whether or not you want to refresh roles and resource policies as well as the authentication policy, role mapping rules, and role restrictions.

On-demand—At any time, you can manually evaluate the authentication policy, role mapping rules, role restrictions, and resource policies of all currently signed-in realm users. This technique is especially useful if you make changes to an authentication policy, role mapping rules, role restrictions, or resource policies and you want to immediately refresh the roles of a realm’s users.

Evaluate all signed-in users in all realms—At any time, you can manually refresh the roles of all currently signed-in users in all realms by using settings in the System > Status > Active Users page. For information, see “Monitoring Active Users” on page 434.

Policy Evaluation

User roles determine which resources can be accessed and how the access is provided. The Infranet Enforcer applies access policies based on the user’s role(s). The Infranet Controller sends the access policies to an Infranet Enforcer when connecting, and whenever the access policies change. When a user first signs in or if the user’s role changes (for example, if the security compliance of the endpoint changes), the Infranet Controller sends the current role(s) to each Infranet Enforcer. This enables the Infranet Enforcer to enforce the access policies.

An Infranet Enforcer evaluates a set of access policies from the top down, meaning that it starts with the policy numbered one and then continues down the policy list until it finds a matching policy.

Details regarding each evaluation step:

1. The Infranet Controller assigns the user’s role(s) during the authentication process, and reevaluates the roles periodically if dynamic role mapping is enabled. You can also manually reevaluate a user’s role(s). For more information, see “Dynamic Policy Evaluation” on page 105.

2. The Infranet Enforcer stops processing policies as soon as the requested resource is found in a policy’s Resource list.

NOTE: If you use automatic (time-based) dynamic policy evaluation or you perform a manual policy evaluation, the Infranet Enforcer repeats the resource evaluation process described in this section. For more information, see “Dynamic Policy Evaluation” on page 105.
Configuring Security Requirements

The Infranet Controller makes it easy to specify security requirements for administrators and users through the options and features described in the following sections:

- “Specifying Source IP Access Restrictions” on page 108
- “Specifying Browser Access Restrictions” on page 110
- “Specifying Certificate Access Restrictions” on page 112
- “Specifying Password Access Restrictions” on page 113
- “Specifying Host Checker Access Restrictions” on page 114
- “Specifying Session Limits” on page 114

Specifying Source IP Access Restrictions

Use a source IP restriction to control from which IP addresses users can access an Infranet Controller sign-in page, be mapped to a role, or access a resource.

You can restrict resource access by source IP:

- **When administrators or users try to sign in to the Infranet Controller** — The user must sign in from a machine whose IP address/netmask combination meets the specified source IP requirements for the selected authentication realm. If the user’s machine does not have the IP address/netmask combination required by the realm, the Infranet Controller does not forward the user’s credentials to the authentication server and the user is denied access to the Infranet Controller. You can allow or deny access to any specific IP address/netmask combination. For example, you can deny access to all users on a wireless network (10.64.4.100), and allow access to all other network users (0.0.0.0).

- **When administrators or users are mapped to a role** — The authenticated user must be signed in from a machine whose IP address/netmask combination meets the specified Source IP requirements for each role to which the Infranet Controller may map the user. If the user’s machine does not have the IP address/netmask combination required by a role, then the Infranet Controller does not map the user to that role.

You can also use a source IP restriction in the following ways:

- **Source IP restriction on realms** — Suppose an Infranet Enforcer is installed between a particular access network and the rest of the network, and the Infranet Controller routes all traffic through this Infranet Enforcer. You can use a source IP restriction to only allow users to login from the access network because logging in from any other network results in denial of network access. For example, you can use this configuration to prevent users from logging into the Infranet Controller from networks other than a wireless network.
**Source IP restriction on roles**—You can use source IP restrictions to set up different roles for different access networks. Only endpoints authenticating from a particular access network are assigned the role corresponding to that network. You can then create Infranet Enforcer IPsec routing policies (see “Configuring IPsec Routing Policies” on page 82) and Infranet Enforcer source interface policies (see “Configuring Source Interface Policies” on page 89) specifically for the role that corresponds to an access network so that endpoints route network traffic through the appropriate Infranet Enforcer.

To specify source IP restrictions:

1. Select the level at which you want to implement IP restrictions:

   - **Realm level**—navigate to:
     - Administrators > Admin Realms > SelectRealm > Authentication Policy > Source IP
     - Users > User Realms > SelectRealm > Authentication Policy > Source IP

   - **Role level**—Navigate to:
     - Administrators > Admin Roles > Select Role > General > Restrictions > Source IP
     - Users > User Roles > Select Role > General > Restrictions > Source IP

2. Choose one of the following options:

   - **Allow users to sign in from any IP address** — Enables users to sign into the Infranet Controller from any IP address in order to satisfy the access management requirement.

   - **Allow or deny users from the following IP addresses** — Specifies whether to allow or deny users access to the Infranet Controller from all of the listed IP addresses, based on their settings. To specify access from an IP address:
     i. Enter the IP address and netmask.
     ii. Select either:
        - **Allow** to allow users to sign in from the specified IP address.
- **Deny** to prevent users from signing in from the specified IP address.

iii. Click **Add**.

iv. If you add multiple IP addresses, move the highest priority restrictions to the top of the list by selecting the check box next to the IP address, and then clicking the up arrow button. For example, to deny access to all users on a wireless network (10.64.4.100) and allow access to all other network users (0.0.0.0), move the wireless network address (10.64.4.100) to the top of the list and move the (0.0.0.0) network address below the wireless network.

5. Click **Save Changes** to save your settings.

### Specifying Browser Access Restrictions

Use a browser restriction to control from which Web browsers users can access an Infranet Controller sign-in page or be mapped to a role. If a user tries to sign in to the Infranet Controller using an unsupported browser, the sign-in attempt fails and a message displays stating that an unsupported browser is being used. This feature also enables you to ensure that users sign in to the Infranet Controller from browsers that are compatible with corporate applications or are approved by corporate security policies.

You can restrict Infranet Controller and resource access by browser-type:

- **When administrators or users try to sign in to the Infranet Controller**—The user must sign in from a browser whose user-agent string meets the specified user-agent string pattern requirements for the selected authentication realm. If the realm “allows” the browser's user-agent string, then the Infranet Controller submits the user’s credentials to the authentication server. If the realm “denies” the browser's user-agent string, then the Infranet Controller does not submit the user’s credentials to the authentication server.

- **When administrators or users are mapped to a role**—The authenticated user must be signed in from a browser whose user-agent string meets the specified user-agent string pattern requirements for each role to which the Infranet Controller may map the user. If the user-agent string does not meet the “allowed” or “denied” requirements for a role, then the Infranet Controller does not map the user to that role.

---

**NOTE:** The browser restrictions feature is not intended as a strict access control since browser user-agent strings can be changed by a technical user. It serves as an advisory access control for normal usage scenarios.

---

### Specifying Browser Restrictions

To specify browser restrictions:

1. Select the level at which you want to implement browser restrictions:

   - **Realm level**—Navigate to:
2. Choose one of the following options:

- **Allow all users matching any user-agent string sent by the browser**—Allows users to access the Infranet Controller or resources using any of the supported Web browsers.

- **Only allow users matching the following User-agent policy**—Allows you to define browser access control rules. To create a rule:
  
  i. For the **User-agent string pattern**, enter a string in the format

  ```
  *
  <browser_string>
  ```

  where start (*) is an optional character used to match any character and `<browser_string>` is a case-sensitive pattern that must match a substring in the user-agent header sent by the browser. Note that you cannot include escape characters (\) in browser restrictions.

  ii. Select either:

  - **Allow** to allow users to use a browser that has a user-agent header containing the `<browser_string>` substring

  - **Deny** to prevent users from using a browser that has a user-agent header containing the `<browser_string>` substring.

  iii. Click **Add**.
3. Click **Save Changes** to save your settings.

---

**NOTE:**

- Rules are applied in order, so the first matched rule applies.
- Literal characters in rules are case sensitive, and spaces are allowed as literal characters.

For example, the string **Netscape** matches any user-agent string that contains the substring **Netscape**.

The following rule set grants resource access only when users are signed in using Internet Explorer 5.5x or Internet Explorer 6.x. This example takes into account some major non-IE browsers that send the 'MSIE' substring in their user-agent headers:

- *Opera* Deny
- *AOL* Deny
- *MSIE 5.5* Allow
- *MSIE 6.* Allow
- * Deny

---

**Specifying Certificate Access Restrictions**

When you install a client-side certificate on the Infranet Controller through the **System > Configuration > Certificates > Trusted Client CAs** page of the admin console, you can restrict Infranet Controller and resource access by requiring client-side certificates:

- **When administrators or users try to sign in to the Infranet Controller**—The user must sign in from a machine that possesses the specified client-side certificate (from the proper certificate authority (CA) and possessing any optionally specified field/value pair requirements). If the user's machine does not possess the certificate information required by the realm, the user can access the sign-in page, but once the Infranet Controller determines that the user's browser does not possess the certificate, the Infranet Controller does not submit the user's credentials to the authentication server and the user cannot access features on the Infranet Controller.

To implement certificate restrictions at the realm level, navigate to:

- **Administrators > Admin Realms > SelectRealm > Authentication Policy > Certificate**
- **Users > User Realms > SelectRealm > Authentication Policy > Certificate**

---

Juniper Networks Unified Access Control Administration Guide

112 - Configuring Security Requirements
When administrators or users are mapped to a role—The authenticated user must be signed in from a machine that meets the specified client-side certificate requirements (proper certificate authority (CA) and optionally specified field/value pair requirements) for each role to which the Infranet Controller may map the user. If the user's machine does not possess the certificate information required by a role, then the Infranet Controller does not map the user to that role.

To implement certificate restrictions at the role level, navigate to:

- Administrators > Admin Roles > SelectRole > General > Restrictions > Certificate
- Users > User Realms > SelectRealm > Role Mapping > Select|CreateRule > CustomExpression
- Users > User Roles > SelectRole > General > Restrictions > Certificate

Specifying Password Access Restrictions

You can restrict Infranet Controller and resource access by password-length when administrators or users try to sign in to an Infranet Controller. The user must enter a password whose length meets the minimum password-length requirement specified for the realm. Note that local user and administrator records are stored in the Infranet Controller authentication server. This server requires that passwords are a minimum length of 6 characters, regardless of the value you specify for the realm’s authentication policy.

To specify password restrictions:

1. Select an administrator or user realm for which you want to implement password restrictions.

   Navigate to:

   - Administrators > Admin Realms > Select Realm > Authentication Policy > Password
   - Users > User Realms > Select Realm > Authentication Policy > Password

2. Choose one of the following options:

   - Allow all users (passwords of any length) — Does not apply password length restrictions to users signing in to the Infranet Controller.
   - Only allow users that have passwords of a minimum length — Requires the user to enter a password with a minimum length of the number specified.
3. Select **Enable Password Management** if you want to enable password management. You must also configure password management on the Infranet Controller authentication server configuration page (local authentication server) or through an LDAP server. For more information about password management, see “Enabling LDAP Password Management” on page 195.

4. Click **Save Changes** to save your settings.

---

**NOTE:** By default, the Infranet Controller requires that user passwords entered on the sign-in page be a minimum of four characters. The authentication server used to validate a user’s credentials may require a different minimum length. The Infranet Controller local authentication database, for example, requires user passwords to be a minimum length of six characters.

**Specifying Host Checker Access Restrictions**

For information about restricting a user’s access to the Infranet Controller, a role, or a resource based on his Host Checker status, see “Implementing Host Checker Policies” on page 311.

**Specifying Session Limits**

A session is a single authenticated connection between an endpoint and the Infranet Controller. You can limit the number of session for a given realm.

Setting the minimum or maximum limit amount allows you to configure realms that are more likely to be available when the Infranet Controller is nearing the amount of licensed users.

Valid numbers for the minimum amount of sessions are between 0 and the license limit. A default of 0 means there are no limits. All of the realms minimum limits can add up to the license limit but cannot exceed it. You cannot modify an existing realm’s minimum limit or add a new realm’s minimum limit that exceeds the license limit. The maximum limit can be equal to or greater than the minimum limit for a particular realm. Value 0 for maximum limit means no user can log in to the realm.

You can also limit the number of concurrent users per session. A user can have multiple sessions. For example, if a user logs in from two machines in the same realm, an additional session is created.

Users who enter through a realm with this feature enabled must have no more than the specified number of sessions open. If the user attempts to open a new session that exceeds the limit, the Odyssey Access Client, or a browser dialog on agentless connections, displays a message giving the user the option to continue or cancel. The current user sessions are displayed in a table, and the user can delete individual sessions to reach compliance. If the user’s session limit comes into compliance, the user is provided access. If the user chooses to cancel, the Infranet Controller does not create the session.
If a user connected with agentless access attempts to log in from the same source IP, the dialog displays the IP address with an asterisk (*) and gives the agentless user the option to delete the existing session(s).

**NOTE:** If a user with agentless access or Java agent access attempts to log in from a source IP from which a session has already been established, the Infranet Controller automatically replaces the old session with a new session.

Users can authenticate through different realms. If an endpoint accesses the Infranet Controller through multiple realms, multiple sessions are possible. These sessions do not count against individual realm session limits.

The Infranet Controller performs the session limit check after authentication, but before a session is created.

The Infranet Controller performs the session limit check during authentication. If administrators choose to reduce the session limits, it would not impact existing sessions unless the Dynamic policy evaluation option is enabled. With Dynamic policy evaluation enforced, the oldest session(s) of a non-compliant user are silently dropped.

These limits will not be enforced if the realm is configured to proxy outer authentication.

To limit the number of simultaneous sessions:

1. Navigate to **Users > User Realms > SelectRealm > Authentication Policy > Limits**.
2. To limit the number of concurrent sessions, select the check box for **Limit number of concurrent sessions** and type a **Guaranteed minimum** and/or **Guaranteed maximum**.
3. To limit the number of sessions for users, select **Limit the number of concurrent sessions for users**.
4. Specify the number of sessions permitted for users in the **Session Limit** text box. By default, the number is 1 if the realm maximum is greater than 0; otherwise, the default is 0. The maximum number must be no greater than the maximum number of concurrent users for the realm.
5. Click **Save Changes**.
A *user role* defines user session parameters (session settings and options) and personalization settings (user interface customization). At the role level you specify whether associated endpoints will download the Odyssey Access Client, the Java agent or whether agentless access will be permitted.

A user role does not specify resource access control or other resource-based options for an individual request. For example, a user role might enable the Host Enforcer for Windows endpoint platforms. Another user role might enable agentless access (see “Configuring Agentless Access to Protected Resources” on page 56), and still another may be used to provision access via the Java agent (see “Configuring the Java Agent for Endpoint Access” on page 58). However, the individual resources that a user may access are defined by the Infranet Enforcer resource access policies or Host Enforcer policies that you configure separately.

The Infranet Controller supports two types of user roles:

- **Administrators**—An administrator role specifies Infranet Controller management functions and session properties for administrators who map to the role. You can customize an administrator role by selecting the Infranet Controller feature sets and user roles that members of the administrator role are allowed to view and manage. You can create and configure administrator roles through the Delegated Admin Roles page. Click **Administrators > Admin Roles** in the admin console.

- **Users**—A user role defines user session parameters and personalization settings. You can customize a user role by specifying access restrictions, enabling Host Enforcer (Windows) or agentless or Java agent access, and configuring session settings. You can create and configure user roles through the **Users > User Roles** page of the admin console.

This topic includes the following information about roles:

- “User Role Evaluation” on page 118
- “Configuring User Roles” on page 120
- “Configuring Access Options on a Role” on page 126
User Role Evaluation

The Infranet Controller’s role mapping engine determines a user’s session role, or combined permissions valid for a user session, as illustrated in the following figure. A detailed description of each step follows the diagram.

The Infranet Controller’s role mapping engine determines a user’s session role, or combined permissions valid for a user session, as illustrated in the following diagram. A detailed description of each step follows after the diagram.

**NOTE:** If you assign a role to a RADIUS proxy realm, role restrictions cannot be enforced. Host Checker policies, Source IP restrictions, and any other limits that you have been assigned will be bypassed. RADIUS proxy should only be used if no restrictions have been applied. Additionally, outer proxy cannot be used if a role mapping rule based on user names is being used, as the Infranet Controller cannot see the user name, and a session cannot be created.

**Figure 13: Security Checks Performed by the Infranet Controller to Create a Session Role**

START

1. Infranet Controller evaluates first role-mapping rule

2. Infranet Controller evaluates subsequent rules

3. Infranet Controller compiles list of valid roles

4. Infranet Controller merges settings or requires user to pick role

END

The Infranet Controller performs the following security checks to create a session role:

1. The Infranet Controller begins rule evaluation with the first rule on the Role Mapping tab of the authentication realm to which the user successfully signs in. During the evaluation, the Infranet Controller determines if the user meets the rule conditions. If so, then:
Chapter 4: User Roles

a. The Infranet Controller adds the corresponding roles to a list of "eligible roles" available to the user.

b. The Infranet Controller considers whether or not the "stop on match" feature is configured. If so, then the engine jumps to step 5.

2. The Infranet Controller evaluates the next rule on the authentication realm’s Role Mapping tab according to the process in Step 1 and repeats this process for each subsequent rule. When the Infranet Controller evaluates all role mapping rules, it compiles a comprehensive list of eligible roles.

3. The Infranet Controller evaluates the definition for each role in the eligibility list to determine if the user complies with any role restrictions. The Infranet Controller then uses this information to compile a list of valid roles, whose requirements the user also meets.

   If the list of valid roles contains only one role, then the Infranet Controller assigns the user to that role. Otherwise, the Infranet Controller continues the evaluation process.

4. The Infranet Controller evaluates the setting specified on the Role Mapping tab for users who are assigned to more than one role:

   - **Merge settings for all assigned roles**—If you choose this option, then the Infranet Controller performs a permissive merge of all the valid user roles to determine the overall (net) session role for a user session.

   - **User must select from among assigned roles**—If you choose this option, then the Infranet Controller presents a list of eligible roles to an authenticated user. The user must select a role from the list, and the Infranet Controller assigns the user to that role for the duration of the user session.

   - **User must select the sets of merged roles assigned by each rule**—If you choose this option, the Infranet Controller presents a list of eligible rules to an authenticated user (that is, rules whose conditions the user has met). The user must select a rule from the list, and the Infranet Controller performs a permissive merge of all the roles that map to that rule.

**NOTE:** If you use automatic (time-based) dynamic policy evaluation or you perform a manual policy evaluation, the Infranet Controller repeats the role evaluation process described in this section. See “Dynamic Policy Evaluation” on page 105.

---

**Permissive Merge Guidelines**

A *permissive merge* is a merge of two or more roles that combines enabled features and settings following these guidelines:

- Any enabled access feature in one role takes precedence over the same feature set to disabled in another role. For example, if a user maps to two roles, one of which disables the Host Enforcer while the other role enables the Host Enforcer, the Infranet Controller enables the Host Enforcer for that session.
In the case of user interface options, the Infranet Controller applies the settings that correspond to the user's first role.

In the case of maximum session lengths, the Infranet Controller applies the greatest value from all of the roles to the user's session.

If more than one role enables the Roaming Session feature, then the Infranet Controller merges the netmasks to formulate a greater netmask for the session.

## Configuring User Roles

To create a user role:

1. In the admin console, choose Users > User Roles.

2. Click New Role and then enter a name and optionally a description. This name appears in the list of Roles on the Roles page.

Once you have created a role, you can click the role's name to begin configuring it using the instructions in the following sections:

- “Configuring General Role Options” on page 121
- “Configuring Role Restrictions” on page 121
- “Specifying Session Options” on page 122
- “Specifying UI Options” on page 123
- “Defining Default Options for User Roles” on page 124

### NOTE:

- To create individual user accounts, you must add the users through the appropriate authentication server (not the role). For instructions, see “Creating User Accounts on a Local Authentication Server” on page 202 for local authentication servers. Or for instructions on how to create users on third-party servers, see the documentation that comes with that product.

- To display the role ID, place the mouse cursor over the role's name on the Roles pages. The role ID appears at the end of the link text shown on the status bar at the bottom of the Web browser window. To show information on the Infranet Enforcer about the role ID of a specific Infranet Controller authentication table entry, use this CLI command:

  ```
  get auth table infranet auth-id <x>
  ```
Configuring General Role Options

Use the Overview tab to edit a role’s name and description, and toggle session and user interface options on and off. When you enable an access feature, make sure to create corresponding resource policies.

To manage general role settings and options:

1. In the admin console, click **Users > User Roles > Role Name > General > Overview**.
2. Revise the name and description and then click **Save Changes** (optional).
3. Under Options, check the role-specific options that you want to enable for the role. If you do not select role-specific options, the Infranet Controller uses default settings instead, as described in “Defining Default Options for User Roles” on page 124. Role-specific options include:
   - **Session Options**—Select this option to apply the role settings in the General > Session Options page to the role. See “Specifying Session Options” on page 122.
   - **UI Options**—Select this option to apply the role settings in the General > UI Options page to the role. See “Specifying UI Options” on page 123.
   - **Odyssey Settings for IC Access**—Select this option to specify Odyssey Access Client connection and authentication options. By Default, this option is not selected. See “IC Access Configuration Settings” on page 45.
   - **Odyssey Settings for Preconfigured Installer**—Select this option to upload a preconfigured installer for Odyssey Access Client. By Default, this option is not selected. See “Using the Preconfigured Installer for Odyssey Access Client” on page 51.
4. Click **Save Changes** to apply the settings to the role.

Configuring Role Restrictions

Click **Restrictions** at the top of the General tab to specify access management options for the role. The Infranet Controller considers these restrictions when determining whether or not to map a user to the role. The Infranet Controller does not map users to this role unless they meet the specified restrictions. See “General Access Management” on page 67.

You may configure any number of access management options for the role. If a user does not conform to all of the restrictions, then the Infranet Controller does not map the user to the role.

If you configure a role that is assigned to a RADIUS proxy realm, role restrictions cannot be enforced. The proxy target will authenticate users without regard to any restrictions that have been configured.
To specify access management options for the role:

1. In the admin console, click **Users > User Roles > Role Name > General > Restrictions**.

2. Click the tab corresponding to the option you want to configure for the role, and then configure it using instructions in the following sections:
   - “Specifying Source IP Access Restrictions” on page 108
   - “Specifying Browser Access Restrictions” on page 110
   - “Specifying Certificate Access Restrictions” on page 112
   - “Specifying Password Access Restrictions” on page 113
   - “Specifying Host Checker Access Restrictions” on page 114

### Specifying Session Options

Use the **Session** tab to specify the maximum session length, roaming capabilities, and session persistence. Check the **Session Options** check box on the **Overview** tab to enable these settings for the role.

By default, this option is selected.

To specify general session options:

1. In the admin console, click **Users > User Roles > RoleName > General > Session Options**.

2. Under **Session lifetime**:
   a. For **Max. Session Length**, specify the number of minutes an active non-administrative user session may remain open before ending. The minimum is six minutes. During an end user session, prior to the expiration of the maximum session length, the Infranet Controller prompts the user to re-enter authentication credentials, which avoids the problem of terminating the user session without warning.
   b. For **Heartbeat Interval**, set the frequency at which the endpoint should send out a heartbeat to the Infranet Controller to keep the session alive. For agentless access, the browser refreshes the page with every heartbeat. Users should not navigate away from the browser, as this will interrupt the heartbeat and the session will be ended. The Odyssey Access Client and the Java agent respectively provide the heartbeat. You should ensure that the heartbeat interval of the agent is greater than the Host Checker interval, otherwise performance could be affected.
   c. For **Heartbeat Timeout**, specify the amount of time that the Infranet Controller should “wait” before terminating a session when the endpoint does not send a heartbeat response.
3. Under Roaming session, specify:

- **Enabled**—Select this option to enable roaming user sessions for users mapped to this role. A roaming user session works across source IP addresses, which allows mobile users (laptop users) with dynamic IP addresses to sign in to the Infranet Controller from one location and continue working from another. Disable this feature to prevent users from accessing a previously established session from a new source IP address. This helps protect against an attack spoofing a user’s session, provided the hacker was able to obtain a valid user’s session cookie.

- **Limit to subnet**—Select this option to limit the roaming session to the local subnet specified in the Netmask box. Users may sign in from one IP address and continue using their sessions with another IP address as long as the new IP address is within the same subnet.

- **Disabled**—Select this option to disable roaming user sessions for users mapped to this role. Users who sign in from one IP address may not continue an active Infranet Controller session from another IP address; user sessions are tied to the initial source IP address.

4. Click **Save Changes** to apply the settings to the role.

**Specifying UI Options**

Use the **UI Options** tab to specify customized settings for the Infranet Controller welcome page for Odyssey Access Client users mapped to this role. The Infranet Controller welcome page (or home page) is the Web interface presented to authenticated Infranet Controller users in agentless access deployments (see “Configuring Agentless Access to Protected Resources” on page 56). Select the **UI Options** option on the **Overview** tab to enable custom settings for the role; otherwise, the Infranet Controller uses the default settings.

If a user maps to more than one role, then the Infranet Controller displays the user interface settings corresponding to the first role to which the user is mapped.

By default, this option is selected.

To customize the Infranet Controller welcome page for role users:

1. Click **Users > User Roles > Role Name > General > UI Options**.

2. Under Header, specify a custom logo and alternate background color for the header area of the Infranet Controller welcome page (optional):

   - Click the **Browse** button and locate your custom image file. The new logo appears in the Current appearance box only after you save your changes.

   - Type the hexadecimal number for the background color or click the **Color Palette** icon and pick the desired color. The Current appearance box updates immediately.
Select **Show notification message** and enter a message in the associated text box (optional). The message appears at the top of the Infranet Controller welcome page after you save changes and the user refreshes that page. You may format text and add links using the following HTML tags: `<i>`, `<b>`, `<br>`, `<font>`, and `<a href>`. However, the Infranet Controller does not rewrite links on the sign-in page (since the user has not yet authenticated), so you should only point to external sites. Links to sites behind a firewall will fail. You may also use Infranet Controller system variables and attributes in this field, as explained in “System Variables and Examples” on page 521.

---

**NOTE:**

- The length of the personalized greeting cannot exceed 12K, or 12288 characters.
- If you use unsupported HTML tags in your custom message, the Infranet Controller may display the end user’s Infranet Controller home page incorrectly.

3. Under Other, specify whether or not you want the copyright notice and label shown in the footer (optional). This setting applies only to those users whose license permits disabling the copyright notice. For more information about this feature, call Juniper Networks Support.

4. Click **Save Changes**. The changes take effect immediately, but current user browser sessions may need to be refreshed to see the changes.

5. Click **Restore Factory Defaults** to reset all user-interface options back to factory defaults (optional).

**Defining Default Options for User Roles**

You can define default options for all user roles, just as you can for delegated administrator roles. The default options include, but are not limited to:

- Session Options
- UI Options
- Odyssey Settings
  - IC Access
  - Preconfigured Installer

**Defining Default Options for User Roles**

To define the default options for all user roles:

1. Select **Users > User Roles**.
2. Click Default Options.

3. Modify settings in the Session Options, UI Options, and Odyssey Settings tabs using instructions in “Configuring General Role Options” on page 121.

4. If you want to use the Odyssey Preconfigured Installer option, de-select the check box for Enable Odyssey Settings for IC Access on the Users > User Roles > Default Options page. If you leave this option enabled, the settings on the page will overwrite the options on the preconfigured installer.

5. Click Save Changes. These become the new defaults for all new user roles.

**NOTE:** If you do not want user roles to see the copyright notice, you can also clear the Show copyright notice and “Secured by Juniper Networks” label in footers check box for user roles, in general. That way, all subsequent roles you create do not allow the notice to appear on the end user UI.

### Creating a Role for MAC Address Authentication

To create a role for MAC address authentication, you only need to specify a name, and a session timeout parameter.

1. In the admin console, choose Users > User Roles > New User Role...

2. Add a name and an optional description, then click Save Changes.

**NOTE:** Do not specify any role restrictions for MAC address authentication.

3. Select Users > User Roles > RoleName > General > Session Options.

4. For Max. Session Length, specify the number of minutes the active session may remain open before ending. The minimum is six minutes.

You must also set up a reauthentication interval on the switch. A device’s change in status does not propagate immediately to the switch. Instead, you have to wait for the switch to reauthenticate and fail the authentication. If you configure the device to reauthenticate frequently, changes can be propagated quickly. If you configure the device to reauthenticate less frequently, there will be less load on the Infranet Controller.

Set the switch reauthentication interval to somewhat shorter than the session length that you set for the role. Consult the applicable manufacturer’s documentation for information about configuring the switch.

5. Click Save Changes.
### Configuring Access Options on a Role

You can specify the following role options for access:

- **Connection timeout values**—You can configure timeout values for the connection between Odyssey Access Client and the Infranet Controller.

- **Install Odyssey Access Client**—On Windows platforms, the Odyssey Access Client downloads by default.

- **Enable agentless access**—On Windows, Macintosh, Linux and Solaris platforms, you can also allow users to access protected resources without installing and running Odyssey Access Client on the endpoint. This type of access is referred to as *agentless* access. For more information, see “Configuring Agentless Access to Protected Resources” on page 56.

- **Install Java agent**—On Macintosh and Linux platforms, you can install a lightweight Java agent to provide status and session control. See “Configuring the Java Agent for Endpoint Access” on page 58.

- **Enable Host Enforcer**—By enabling Host Enforcer for a role and specifying endpoint traffic in a Host Enforcer policy, you can control endpoint access to resources and protect endpoints from attacks from other computers. For more information, see “Infranet Enforcer Policies” on page 71.

- **Run session start and end scripts**—You can specify scripts to run on Windows endpoints for users assigned to a role after Odyssey Access Client connects or disconnects with the Infranet Controller. For example, you can specify a script that maps network drives on an endpoint to shares on protected resources as a session start script, and you can specify another script that disconnects the mapped network drives as session end script.

To configure access options on a role:

1. In the admin console, choose **Users > User Roles > RoleName > Agent**.

2. If you want to allow users to use agentless access to access protected resources, select the **Agentless** tab, then select **Enable Agentless Access for this role**. You can also select this to allow access to endpoints in addition to using Odyssey Access Client on Windows machines. If you don’t select the agentless option, the Infranet Controller allows access to protected resources by means of Odyssey Access Client only.

**NOTE:** To configure agentless access, you must also configure a **permit infranet-auth** policy on the Infranet Enforcer to allow access for agentless endpoint platforms. For configuration instructions, see “Source IP Enforcement on the Infranet Enforcer” on page 33.
3. If you want to allow users to download and install the lightweight Java agent for Macintosh or Linux platforms, select the Agent tab, then select Install Java Agent for this role. See “Configuring the Java Agent for Endpoint Access” on page 58.

4. For Odyssey Access Client configurations, select Enable Host Enforcer to enable Host Enforcer on the endpoint and send Host Enforcer policies to Odyssey Access Client for this role (Windows only).

**NOTE:**

- By default, after you enable the Host Enforcer option on a role, Odyssey Access Client denies all traffic on the endpoint except for the following allowed types: traffic to and from the Infranet Controller and Infranet Enforcer, WINS, DNS, IPSec, DHCP, ESP, IKE, outgoing TCP traffic, and some ICMP messages (for example, PING from the endpoint to other devices is allowed). Therefore, it’s important that you configure Host Enforcer policies to specify the additional types of traffic you want to allow on each endpoint. For example, you must configure Host Enforcer policies to allow any incoming TCP traffic. For more information, see “Infranet Enforcer Policies” on page 71.

- To avoid blocking all traffic on endpoints and preventing users from accessing all network and internet resources, we recommend that you configure Host Enforcer policies to allow the specific types of traffic on endpoints before you enable the Host Enforcer option on a role.

5. Under Session Scripts, specify the location of the session start and end scripts you want to run on Windows endpoints after Odyssey Access Client connects or disconnects with the Infranet Controller. You can specify a fully qualified path. Scripts can be accessed locally or remotely by means of file share or other permanently-available local network resource. You can also use environment variables, such as %USERNAME% in the script path name. For example:

```
\abc\users\%USERNAME%\myscript.bat
```
6. When Odyssey Access Client connects to the Infranet Controller, the Infranet Controller copies the session start and end scripts to a temporary directory on the endpoint (defined by the %TEMP% environment variable). When Odyssey Access Client disconnects, the Infranet Controller deletes the copied scripts from the temporary directory. Click **Save Changes**.

---

**NOTE:**

- Windows only supports scripts with the .bat, .cmd, or .exe extension. To run a .vbs script, the user must have a batch file to call the .vbs script.

- Any files referenced in a script are not copied to the endpoint; only the script itself is copied. Any references to files in scripts must take the temporary directory on the endpoint location into account.

- After connecting to the Infranet Controller, Odyssey Access Client copies the session end script from a network drive to a temporary directory on the endpoint so that the end script can run if the network connection fails.

- The session scripts are run in the user’s context.

- If a user qualifies for multiple roles, all scripts for all roles are run. You cannot configure the order in which to run the scripts when multiple roles are assigned to a user.

---

**Use Case: Using a Host Checker Policy with the Host Enforcer**

This use case illustrates how to use a Host Checker policy to verify whether a particular third-party firewall is running on the endpoint. If the third-party firewall is not running, you can map the user to a role that enables Host Enforcer on the endpoint. If the third-party firewall is running, you can map the user to a role that does not enable Host Enforcer.

To use a Host Checker policy with Host Enforcer:

1. Through the **Authentication > Endpoint Security > Host Checker** page of the admin console, create a Host Checker policy that uses a Process Name rule to verify that a particular third-party firewall process is running on the endpoint. Select the **Deny** option for this policy. For configuration instructions, see “Task Summary: Configuring Host Checker” on page 273.

2. Create a role that enables the Host Enforcer for users who do not have the third-party firewall running:
   a. Use the **Users > User Roles** page to create a new role (such as “Role-1”).
   b. On the **Users > User Roles > RoleName > General > Restrictions > Host Checker** page of the admin console, add the Host Checker policy under **Selected Policies**, and select **Allow users whose workstations meet the requirements specified by these Host Checker policies**.
   c. On the **Users > User Roles > RoleName > Agent** page, select **Enable Host Enforcer**.

---

**NOTE:**

- Windows only supports scripts with the .bat, .cmd, or .exe extension. To run a .vbs script, the user must have a batch file to call the .vbs script.

- Any files referenced in a script are not copied to the endpoint; only the script itself is copied. Any references to files in scripts must take the temporary directory on the endpoint location into account.

- After connecting to the Infranet Controller, Odyssey Access Client copies the session end script from a network drive to a temporary directory on the endpoint so that the end script can run if the network connection fails.

- The session scripts are run in the user’s context.

- If a user qualifies for multiple roles, all scripts for all roles are run. You cannot configure the order in which to run the scripts when multiple roles are assigned to a user.
3. Through the Users > User Roles page, create a second role (such as “Role-2”) that does not enable the Host Enforcer for users who do have the third-party firewall running. (For this role, do not select the Host Checker policy and do not select Enable Host Enforcer.)

4. On the Users > User Realms > RealmName > Role Mapping > Role Mapping Rule page, add both roles under Selected Policies.

If the third-party firewall process is not running, the Host Checker policy is successful. The user is mapped to Role-1 which enables Host Enforcer. In this case, the end point is using Host Enforcer.

If the third-party firewall process is running, the Host Checker policy fails because it is set to Deny. The user is not qualified for Role-1 because of the Host Checker restriction. The user is instead mapped to Role-2 which does not enable Host Enforcer. In this case, the end point is using the third-party firewall instead of Host Enforcer.
Configuring Access Options on a Role
Chapter 5

The UAC RADIUS Server and Layer 2 Access

Using the Infranet Controller’s internal RADIUS server, you can provision 802.1X authentication for endpoints. Layer 2 Authentication and enforcement is used to control network access policies at the edge of the network using an 802.1X enabled switch or access point such as a Juniper Networks EX-series switch.

The user authenticates through the switch to the Infranet Controller. The user’s identity and the endpoint health assessment are used to determine which VLAN to use for the switch port that the endpoint is connected to. Typically, if the endpoint does not meet minimum criteria for health assessment as defined by the administrator, the endpoint will be placed on a restricted VLAN which allows access to servers which can aid in remediating the endpoint.

You define VLAN policies for endpoints that authenticate through switches via 802.1X. After an authenticated endpoint has been mapped to a set of roles, the VLAN policies are evaluated and the VLAN information is communicated to the switch through RADIUS attributes. RADIUS attributes vary by make and model of switch. You specify the make and model when configuring a RADIUS client on the Infranet Controller.

In addition to authenticating endpoints with 802.1X the Infranet Controller’s RADIUS server can be used to authenticate 802.1X IP phones, switches, and the Infranet Controller can perform non-802.1X MAC Address based authentication for unmanageable devices.

This chapter contains the following topics:

- “Configuring the Infranet Controller as a RADIUS Server” on page 132
- “Using the Infranet Controller for 802.1X Network Access” on page 142
- “Configuring Access to Switches and Access Points from a Browser” on page 162
- “Controlling Unmanaged Device Access” on page 163
Configuring the Infranet Controller as a RADIUS Server

The Infranet Controller contains an internal RADIUS server that can be configured to perform Extensible Authentication Protocol (EAP) inner and outer authentication, non-tunneled web authentication without EAP, and MAC address authentication. EAP provides for extensibility and is a standard for communication between network access devices and servers.

EAP allows specialized knowledge about authentication protocols to be taken out of the network access device so that it acts solely as a conduit between the authentication server and the client. With EAP, new types of authentication can be supported by adding the appropriate functionality to the server and client without any changes to the network access device or the protocol. The use of EAP can facilitate 802.1X access as well as traditional RADIUS authentication for non-802.1X.

The Infranet Controller supports a variety of authentication protocols. In addition to EAP-TTLS and EAP-PEAP, which the Infranet Controller uses for Odyssey Access Client 802.1X connectivity, the Infranet Controller RADIUS server supports non-tunneled protocols that permit different methods of authentication. For example, MAC address authentication, 802.1X connectivity with non-Juniper supplicants and CHAP authentication (to allow web access to switches) can be configured on the Infranet Controller.

Using the Infranet Controller RADIUS server and the supported EAP protocols, you can configure a network access device to support any combination of the following uses:

- Unmanageable device authentication
- Switch authentication using traditional RADIUS
- Non-Juniper 802.1X supplicant authentication
- Odyssey Access Client 802.1X authentication
- 802.1X IP phone authentication

The network access device’s location group and sign-in policy govern which of the uses is allowed. See “Authentication Realms and Protocols in User Sign-in Policies” on page 260. A broader understanding of the configurable parameters on the Infranet Controller is presented in this section.

This section provides the following information:

- “Authentication Protocols on the Infranet Controller” on page 133
- “Configuring Authentication Protocol Sets” on page 135
- “Using RADIUS Proxy” on page 138
- “Time Limits on RADIUS Authentication and Accounting” on page 141
Chapter 5: The UAC RADIUS Server and Layer 2 Access

**Authentication Protocols on the Infranet Controller**

The Infranet Controller supports a variety of EAP and non-EAP authentication methods to allow you to determine how endpoints will authenticate. Authentication methods can have different purposes. For example, you can use the default EAP methods with the Odyssey Access Client, or you can use different methods to permit authentication with different endpoints such as non-Juniper 802.1X supplicants and IP phones.

For UAC agents (Odyssey Access Client, the Java agent and Host Checker agentless access), authentication is supported via EAP-TTLS (Tunneled Transport Layer Security) and EAP-PEAP (Protected EAP) as the outer protocols and EAP-JUAC (a proprietary Juniper protocol) by default.

EAP-TTLS first authenticates the server and sets up an encrypted TLS tunnel for secure transport of authentication information. Within the TLS tunnel, a second authentication protocol is used to authenticate the user. EAP-TTLS is the “outer” authentication, while the second protocol is the “inner” authentication.

EAP-TTLS consists of two phases. In the first phase, an X.509 digital certificate of the authentication server is used by the supplicant to verify its identity, and to validate the network’s authenticity.

The Authentication Server is required to present a digital certificate. This digital certificate is used in the outer authentication to establish the TLS tunnel from the supplicant to a AAA Server. If there are certificate restrictions, or the inner protocol is EAP-TLS, a user certificate will also be used.

EAP-PEAP is similar to EAP-TTLS, with a difference being that the inner authentication must be another EAP exchange. PEAP can only use EAP-compatible authentication methods. PEAP starts the TLS tunnel, then uses EAP again, encapsulated inside the tunnel to perform the authentication.

EAP-TTLS and EAP-PEAP provide authentication of the user and the network, and produce dynamic keys that can be used to encrypt communications between the endpoint and access point. With mutual authentication, not only does the network authenticate the user credentials, but Odyssey Access Client or the third-party supplicant also authenticates the authentication server.

Requiring mutual authentication is an important security precaution when using wireless networking. Verifying the identity of the authentication server assures that you connect to your intended network, and not to an access point that is pretending to be your network.

You can authenticate with Odyssey Access Client or a third-party 802.1X supplicant when you configure the endpoint to validate the certificate of the authentication server. If the certificate identifies a server that you trust, and if the authentication server can prove that it is the owner of that certificate, then you can safely connect to the network.

**Summary of EAP-TLS, EAP-TTLS and EAP-PEAP:**

EAP-TLS, EAP-TTLS, and EAP-PEAP all employ TLS, the successor of SSL. TLS is the protocol used to secure communications between Web browsers and secure Web servers.
The three protocols do not all use TLS for the same purposes. EAP-TLS uses TLS with server and client certificates to provide mutual authentication. EAP-TTLS uses TLS with a server certificate to provide server authentication, an optional client certificate for client authentication, and an inner, protected EAP or RADIUS exchange for client authentication. EAP-PEAP uses TLS with a server certificate to provide server authentication and an inner, protected EAP exchange for client authentication.

The EAP-JUAC inner protocol allows the Odyssey Access Client to take advantage of the full set of Infranet Controller features, including Host Checker, firewall provisioning and IP address restrictions.

In general, the outer protocol ensures that the client or agent is communicating with a valid, trusted server, and the inner protocol proves your identity to the Infranet Controller.

In addition to EAP-TTLS and EAP-PEAP, the following standard protocols are supported for interactivity with RADIUS clients other than Odyssey Access Client:

- PAP (Password Authentication Protocol) with plain-text passwords
- EAP-GTC (EAP Generic Token Card)
- CHAP (Challenge Handshake Authentication Protocol) and the CHAP family including MS-CHAP, MS-CHAP-V2, EAP-MD5-Challenge, and EAP-MS-CHAP-V2
- EAP-TLS (EAP Transport Layer Security) - the Infranet Controller supports EAP-TLS to allow non-Juniper 802.1X supplicants to authenticate via a certificate authentication server.
- EAP-SOH (State of Health)

The Infranet Controller supports these authentication protocols as non-tunneled authentication methods as well as inner authentication methods, subject to the policies that you configure. You can configure protocol sets with or without EAP, with the exception of MD5, EAP-GTC, EAP-TLS and EAP-SOH which are only supported for EAP. See “Configuring Authentication Protocol Sets” on page 135 for more information.

EAP-SOH is a special protocol used only with Windows Vista and Windows XP Service Pack 3 802.1X supplicants in a Statement of Health Host Checker policy. The EAP-SOH protocol allows the endpoint to exchange state of health messages with the Infranet Controller to assess endpoint qualification for passing Statement of Health rules in a Host Checker policy. To use EAP-SOH you must use EAP-PEAP as an outer authentication protocol. See “Statement of Health Integration” on page 305.

If you use a protocol set with inner and outer authentication, both the outer authentication protocol, and the inner authentication protocol must match the inner and outer protocol that is configured for the endpoint.
There are several ways to access the Infranet Controller. The method used for access and the protocols selected determine the realm(s) through which endpoints can authenticate. Any authentication methods that are incompatible with the authentication server being used are not even attempted. You associate realms with authentication protocols when you configure a sign-in policy. See “Authentication Realms and Protocols in User Sign-in Policies” on page 260.

You can configure any combination of authentication protocols on the Infranet Controller for use with non-Juniper 802.1X supplicants, compatible IP phones, or for non-tunneled access (for example, web access to a switch. See “Configuring Access to Switches and Access Points from a Browser” on page 162).

There are two default pre-configured protocol sets on the Infranet Controller. **802.1X** is used by default with UAC agents. **802.1X-Phones** is used for authenticating 802.1X IP phones. When you configure a new sign-in policy, you must associate realms that you have configured with authentication protocol sets. You can select a protocol set you have created, or you can use one of the default protocol sets, depending on the endpoint. Endpoints can only authenticate through realms that are configured with compatible authentication protocol sets.

NOTE: The **802.1X** protocol set is used by default for endpoints connecting with the Odyssey Access Client. If you disable the JUAC protocol, (a proprietary Juniper protocol) on the Odyssey Access Client or the Infranet Controller, the Odyssey Access Client behaves to the Infranet Controller as a standard non-Juniper supplicant.

You can select several authentication protocols for each protocol set. If you select more than one protocol for inner and outer authentication, the order in which you list the protocols is important. The EAP protocols are considered in order by the Infranet Controller, with selections at the top of the list considered first for each connection attempt.

NOTE: The protocols that support password changing on the Infranet Controller include JUAC, MS-CHAP-V2 (only within a TTLS tunnel), EAP-MS-CHAP-V2 (only within a PEAP or TTLS tunnel), and EAP-GTC. If you use CHAP, PAP or MS-CHAP for a layer 2 connection (for example, with an Active Directory Server) password changing is not supported through the Infranet Controller.

You can direct users with expired passwords to a web interface to access a default VLAN to allow users to login with a cleartext password and change their password. See “Configuring Access to Switches and Access Points from a Browser” on page 162.

Password restrictions (for example, password length) cannot be enforced if you use the CHAP family protocols for authentication.
If you choose EAP-TTLS or EAP-PEAP as primary authentication protocols, you must select separate inner authentication protocols.

You can duplicate an existing protocol set and make changes, and you can delete protocol sets you have created. You cannot delete the default 802.1X protocol set, but you can delete the 802.1X-Phone protocol set.

When an endpoint attempts to authenticate, realm selection is based on which authentication protocols match. For example, if a client and the Infranet Controller do not agree on using a selected protocol set, the realm will not be considered.

Clients that connect to the Infranet Controller include Odyssey Access Client, non-Juniper 802.1X supplicants, 802.1X IP phones, and switches. The Infranet Controller can accept authentication requests from all of these endpoints from a single Network Access Server and route the traffic depending on authentication protocols that are configured for individual realms. See “Authentication Realms and Protocols in User Sign-in Policies” on page 260.

Table 17 lists the available authentication protocol combinations and provides usage recommendations for various combinations.

<table>
<thead>
<tr>
<th>Outer Protocol</th>
<th>Inner</th>
<th>Basis</th>
<th>Usage recommendation</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHAP [1]</td>
<td>n/a</td>
<td>Password</td>
<td>Captive portal or authentication of switch administrators for HP ProCurve switch</td>
</tr>
<tr>
<td>EAP-MD5-Challenge [1]</td>
<td>n/a</td>
<td>Password</td>
<td>Captive portal or authentication of switch administrators, some IP phones</td>
</tr>
<tr>
<td>MS-CHAP [1]</td>
<td>n/a</td>
<td>Password</td>
<td></td>
</tr>
<tr>
<td>MS-CHAP-V2 [1]</td>
<td>n/a</td>
<td>Password</td>
<td></td>
</tr>
<tr>
<td>EAP-MS-CHAP-V2</td>
<td>n/a</td>
<td>Password</td>
<td></td>
</tr>
<tr>
<td>EAP-GTC [1]</td>
<td>n/a</td>
<td>Token</td>
<td>802.1X supplicant, some IP phones</td>
</tr>
<tr>
<td>EAP-TLS</td>
<td>n/a</td>
<td>User Certificate</td>
<td>Non-Juniper 802.1X supplicant</td>
</tr>
<tr>
<td>EAP-PEAP</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EAP-MS-CHAP-V2</td>
<td>Password</td>
<td>Local or Active Directory server</td>
<td></td>
</tr>
<tr>
<td>EAP-GTC</td>
<td>Token</td>
<td>802.1X supplicant</td>
<td></td>
</tr>
<tr>
<td>EAP-TLS</td>
<td>User Certificate</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EAP-JUAC</td>
<td>Various</td>
<td>Odyssey Access Client</td>
<td></td>
</tr>
<tr>
<td>EAP-SOH</td>
<td>Password</td>
<td>Windows supplicant with Statement of Health Host Checker policy</td>
<td></td>
</tr>
</tbody>
</table>
Chapter 5: The UAC RADIUS Server and Layer 2 Access

Table 17: Authentication Protocols

<table>
<thead>
<tr>
<th>Outer</th>
<th>Inner</th>
<th>Basis</th>
<th>Usage recommendation</th>
</tr>
</thead>
<tbody>
<tr>
<td>EAP-TTLS</td>
<td></td>
<td>Odyssey Access Client, other supplicant</td>
<td></td>
</tr>
<tr>
<td>PAP</td>
<td></td>
<td>LDAP auth server</td>
<td></td>
</tr>
<tr>
<td>CHAP</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EAP-MD5-Challenge</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MS-CHAP</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MS-CHAP-V2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EAP-MS-CHAP-V2</td>
<td></td>
<td>Local or Active Directory server</td>
<td></td>
</tr>
<tr>
<td>EAP-GTC</td>
<td></td>
<td>802.1X supplicant</td>
<td></td>
</tr>
<tr>
<td>EAP-JUAC</td>
<td></td>
<td>Odyssey Access Client</td>
<td></td>
</tr>
</tbody>
</table>

[1] If the supplicant or client supports EAP-TTLS or EAP-PEAP, Juniper recommends putting this protocol into one of those tunnels for added security.

[2] With LDAP, there are 3 protocol possibilities:

a. If the LDAP server is also an Active Directory server, configure the server on the Infranet Controller as an Active Directory server, not as an LDAP server. On the Infranet Controller, PEAP-MS-CHAP-V2 is enabled by default. You can also enable MS-CHAP and MS-CHAP-V2 if necessary.

b. If passwords in the LDAP server are stored irreversibly hashed, CHAP family protocols will not work, only PAP and TTLS-PAP will work. On the Infranet Controller TTLS-PAP is enabled by default. You can enable PAP if required, but this is the least secure protocol.

c. Some LDAP servers allow you to store the passwords in cleartext or reversibly encrypted. In this situation, all of the CHAP family protocols will work.

To configure an authentication protocol set:

1. In the Infranet Controller admin console, choose Authentication > Signing In > Authentication Protocols.

   NOTE: The default 802.1X protocol set is configured to work with EAP-TTLS or EAP-PEAP as primary (outer) authentication protocols, and EAP-JUAC or EAP-MS-CHAP-V2 for inner authentication (if EAP-PEAP is used) and EAP-JUAC, PAP, MS-CHAP-V2, EAP-MS-CHAP-V2, or EAP-GenericTokenCard (if EAP-TTLS is used).

2. To create a new protocol set, click New Authentication Protocol, or select the check box beside the existing 802.1X protocol set and click Duplicate.

3. Enter a Name and optional Description for the new authentication protocol set. You select the protocol set by name when you create a sign-in policy.

4. Under Authentication Protocol, select authentication protocol(s) from the Available Protocol list and click Add.
5. If you choose EAP-PEAP as the main authentication protocol, under **PEAP** select an inner authentication protocol from the **Available Protocol** list and click **Add**.

**NOTE:** If you are configuring a protocol set to work with the Windows client and a Host Checker Statement of Health policy, you must choose the EAP-SOH protocol as the inner authentication method within a PEAP tunnel.

6. If you choose EAP-TTLS as the main authentication protocol, under **TTLS** select an inner authentication protocol from the **Available Protocol** list and click **Add**.

7. If you are using inner RADIUS proxy, do not select an inner protocol with EAP-PEAP or EAP-TTLS. See “Using RADIUS Proxy” on page 138.


### Using an 802.1X IP Phone with the Infranet Controller

IP telephones that support 802.1X support EAP, either as EAP-MD-5-Challenge or EAP-TLS, depending on the manufacturer. You can associate a realm with the default 802.1X-Phones protocol, and then use role mapping to assign phones to a role within the realm. The Infranet Controller automatically directs phones that attempt to authenticate using the 802.1X-Phones protocol to the associated realm. See “Authentication Realms and Protocols in User Sign-in Policies” on page 260.

If you are planning to use 802.1X IP phones on a network segment that will also accommodate switches using Web-based authentication, you will need to assign role-mapping rules to ensure that phones can be recognized, since a switch using MD-5 Challenge would automatically be authenticated through the same realm. For example, Avaya phones can be recognized by the expression \[0-9afA-F]*. You can create a role mapping rule that specifies if user = \[0-9afA-F]*, then assign to a role specific to IP phones.

### Using RADIUS Proxy

You can configure the Infranet Controller to proxy RADIUS inner or outer authentication to an external RADIUS server.

In environments with many distributed users, it may be difficult or impossible to maintain a centralized database of users. Proxying inner or outer authentication gives you the flexibility to direct requests for authentication through whatever realm is most appropriate for each particular user. Whether you proxy inner or outer RADIUS authentication depends on where you want the authentication tunnel to terminate.
With RADIUS proxy, the Infranet Controller RADIUS server can forward authentication requests from a network access device to an external RADIUS server. The proxy target receives the request, performs the authentication and returns the results. The Infranet Controller RADIUS server then passes the results to the network access device.

**NOTE:** When RADIUS proxy is used, realm or role restrictions cannot be enforced. Host Checker policies, Source IP restrictions, and any other limits that have been assigned are bypassed. RADIUS proxy should be used only if no restrictions have been applied. The exception is that session limitations can be enforced for inner proxy. With outer proxy, no session is established.

You configure RADIUS proxy at the realm level. If the authentication server for the Realm is a RADIUS server, option buttons on the page allow you to select inner proxy, outer proxy or do not proxy. Do not proxy is selected by default. If the authentication server is not a RADIUS server, the proxy option buttons are hidden. See “Creating an Authentication Realm” on page 246.

If an incoming RADIUS authentication or accounting request is assigned to a realm that uses RADIUS proxy, the Infranet Controller proxies the request to the external RADIUS server.

With outer proxy, all RADIUS attributes are passed from the Infranet Controller RADIUS server to the network access device.

With inner proxy, the network access device sends tunneled authentication requests and the Infranet Controller decrypts the TLS traffic and forwards the inner traffic to another RADIUS server, the proxy target. The Infranet Controller receives the responses from the second RADIUS server, encrypts the responses using TLS, and sends the response back to the network access device inside the tunnel. If you use inner proxy, traffic between the Infranet Controller and the external RADIUS server should be well-protected with physical security or some other means.

With a tunneled request, inner proxy allows the Infranet Controller to inspect the inner traffic to obtain the username and RADIUS return attributes.

With outer proxy, the network access device sends tunneled or bare authentication requests, and the Infranet Controller forwards the requests without TLS processing. With outer proxy, the Infranet Controller acts as a conduit between the network access device and the proxy target.

Outer proxy cannot be used if a role mapping rule based on user names is being used, as the Infranet Controller cannot see the user name, and a session cannot be created.

If the authentication server selected for a realm is a RADIUS server, the **Proxy Outer Authentication** option button controls whether outer authentication is proxied, and the **Proxy Inner Authentication** option button controls whether inner authentication is proxied.

You can also choose the **Do not proxy** option button if you do not want inner or outer authentication to be proxied. In this case, the Infranet Controller handles both inner and outer authentication. You must enable the JUAC protocol for this option. See “Configuring Authentication Protocol Sets” on page 135.
There are special considerations for RADIUS proxy with respect to realm selection. See “Authentication Realms and Protocols in User Sign-in Policies” on page 260.

**Use Cases for RADIUS Proxy**

RADIUS proxy can permit greater flexibility in network design and accommodate existing topologies. In many networks, authentication data for different workgroups is grouped in different ways. For example, authentication groups could be configured by department, subsidiary or by acquired company. You can configure the local network access device to use the Infranet Controller for authentication of local endpoints, and you could use second-tier RADIUS servers (proxy targets) to handle the different groups.

One advantage of this setup is in the simplification of configuration. The network access devices and each RADIUS server must share a secret passcode. The Infranet Controller allows network access devices to not have to communicate directly with each RADIUS server, every second-tier RADIUS server does not have to share a secret with every network access device in the company. The shared secrets are handled by the Infranet Controller.

If the different network components (Infranet Controller, authentication server, network access device, and RADIUS server) are managed by different individuals, the local administrators can configure authentication servers to communicate with local RADIUS servers without the overhead of connecting each authentication server to Infranet Controllers or Infranet Controller clusters throughout the company.

With RADIUS proxy you can easily transition using a RADIUS-based AAA service, avoiding the need to enter users on the Infranet Controller. Using your existing RADIUS server gives you access to powerful RADIUS features that are not supported on the Infranet Controller RADIUS server.

With inner proxy, the proxy target specializes in authentication, and the Infranet Controller specializes in access control.

The Infranet Controller has local knowledge that is critical to controlling user access to the network. The Infranet Controller can be configured to determine what VLAN numbers and ACL identifiers are relevant at each site. This data could be different between remote sites.

With outer proxy, you can use outer protocols that are not supported on the Infranet Controller (for example, EAP-PEAPv1 or EAP POTP).

If the proxy target can do things that the Infranet Controller cannot do, for example communicate with SQL, the Infranet Controller can offload to a proxy server that can communicate with SQL.
Time Limits on RADIUS Authentication and Accounting

All requests for authentication have a time limit. Depending on the endpoint, the authentication protocols used, the network access device (NAD) settings, and the Host Checker policies configured at the role and realm level, RADIUS time limits could affect the success or failure of authentication and the performance and memory allocation of the RADIUS server. Table 18, “RADIUS Event Time Limits” on page 141 displays network events and the device or endpoint response when the timeout is exceeded.

You can use this information along with the RADIUS Diagnostic Log and User Log as a guide for troubleshooting the Infranet Controller. See “Using the RADIUS Diagnostic Log” on page 444.

Table 18: RADIUS Event Time Limits

<table>
<thead>
<tr>
<th>Interval Starts:</th>
<th>Interval Ends:</th>
<th>Limited by:</th>
<th>Effect of Timeout</th>
</tr>
</thead>
<tbody>
<tr>
<td>when the NAD sends a single RADIUS request to the Infranet Controller</td>
<td>when the NAD receives the RADIUS response</td>
<td>NAD: sometimes five seconds, usually configurable</td>
<td>NAD resends an exact copy of the RADIUS request (if it is configured to do so) RADIUS diagnostic log indicates that a duplicate was received</td>
</tr>
<tr>
<td>when the NAD sends the first copy of a RADIUS request to the Infranet Controller</td>
<td>when the NAD receives the RADIUS response</td>
<td>NAD: (the timeout interval above) x (the maximum number of retries + 1) The maximum number of retries is typically 2 or 3 and is usually configurable</td>
<td>The NAD assumes a communication failure with the RADIUS server. It may record the event in the log and report it to the endpoint. The Infranet Controller RADIUS diagnostic log shows turnaround times longer than the NAD’s limit.</td>
</tr>
<tr>
<td>when NAD forwards an EAP request from the Infranet Controller to an endpoint</td>
<td>when the NAD receives an EAP response from the endpoint</td>
<td>NAD: (this may be limited by a configuration setting on the NAD, or the NAD may honor the Session Timeout attribute that the Infranet Controller included in the Access-Challenge packet - see next row)</td>
<td>The Infranet Controller user log reports timeout while waiting for a RADIUS continuation request.</td>
</tr>
<tr>
<td>when the Infranet Controller sends the first EAP message of an EAP exchange to the NAD for forwarding to the endpoint</td>
<td>when the Infranet Controller receives the last EAP response</td>
<td>Infranet Controller: This limit was two minutes and has been increased to four minutes</td>
<td>The Infranet Controller user log reports timeout while waiting for a RADIUS continuation request.</td>
</tr>
</tbody>
</table>
Using the Infranet Controller for 802.1X Network Access

The IEEE 802.1X protocol provides authenticated access to a LAN. This standard applies to wireless as well as wired networks. In a wireless network, the 802.1X authentication occurs after the client has associated to an access point using an 802.11 association method. Wired networks use the 802.1X standard without any 802.11 association by connecting to a port on an 802.1X enabled switch. See the Juniper Networks Application note 3rd Party Switch Configurations.

This section contains the following topics:

- “Task summary: Configuring the Infranet Controller as a RADIUS Server for an 802.1X Network Access Device” on page 144
- “Configuring Location Groups” on page 145
- “Configuring RADIUS Clients” on page 147
- “Configuring RADIUS Attributes Policies” on page 151
- “Using an Infranet Enforcer as a RADIUS Client of the Infranet Controller” on page 159
- “Using a Non-Juniper 802.1X Supplicant” on page 160

Table 18: RADIUS Event Time Limits

<table>
<thead>
<tr>
<th>Interval Starts:</th>
<th>Interval Ends:</th>
<th>Limited by:</th>
<th>Effect of Timeout</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>NAD: Some NADs limit this. The limit is not always configurable</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>The NAD takes the endpoint off the network unless it has been reauthenticated.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>NAD: This may be fixed in the NADs configuration or controlled by the Session Timeout attributes that the sends as part of the Access-Accept packet. The Session-Timeout attribute is set by the roles assigned to the user, or by the RADIUS attributes policy.</td>
<td>Endpoint loses network connectivity. NAD sends a RADIUS Accounting-Stop packet (if configured to do so). The Infranet Controller records in the user log.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Odyssey Access Client automatically initiates reauthentication.</td>
<td>Odyssey Access Client automatically initiates reauthentication. User intervention is typically needed for a SecureID card only. If reauthentication succeeds, the endpoint retains network access.</td>
</tr>
</tbody>
</table>

Using the Infranet Controller for 802.1X Network Access

The IEEE 802.1X protocol provides authenticated access to a LAN. This standard applies to wireless as well as wired networks. In a wireless network, the 802.1X authentication occurs after the client has associated to an access point using an 802.11 association method. Wired networks use the 802.1X standard without any 802.11 association by connecting to a port on an 802.1X enabled switch. See the Juniper Networks Application note 3rd Party Switch Configurations.

This section contains the following topics:

- “Task summary: Configuring the Infranet Controller as a RADIUS Server for an 802.1X Network Access Device” on page 144
- “Configuring Location Groups” on page 145
- “Configuring RADIUS Clients” on page 147
- “Configuring RADIUS Attributes Policies” on page 151
- “Using an Infranet Enforcer as a RADIUS Client of the Infranet Controller” on page 159
- “Using a Non-Juniper 802.1X Supplicant” on page 160

Table 18: RADIUS Event Time Limits

<table>
<thead>
<tr>
<th>Interval Starts:</th>
<th>Interval Ends:</th>
<th>Limited by:</th>
<th>Effect of Timeout</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>NAD: Some NADs limit this. The limit is not always configurable</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>The NAD takes the endpoint off the network unless it has been reauthenticated.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>NAD: This may be fixed in the NADs configuration or controlled by the Session Timeout attributes that the sends as part of the Access-Accept packet. The Session-Timeout attribute is set by the roles assigned to the user, or by the RADIUS attributes policy.</td>
<td>Endpoint loses network connectivity. NAD sends a RADIUS Accounting-Stop packet (if configured to do so). The Infranet Controller records in the user log.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Odyssey Access Client automatically initiates reauthentication.</td>
<td>Odyssey Access Client automatically initiates reauthentication. User intervention is typically needed for a SecureID card only. If reauthentication succeeds, the endpoint retains network access.</td>
</tr>
</tbody>
</table>
With 802.1X, the user is authenticated to the network by means of user credentials, such as a password, certificate, or a token card. The keys used for data encryption are generated dynamically. The authentication is not performed by the network access device, but rather by the Infranet Controller as the RADIUS server.

802.1X uses EAP messages to perform authentication. Newer EAP protocols can dynamically generate the WEP, TKIP, or AES keys that are used to encrypt data between the client and the wireless access point. Dynamically created keys are more difficult to break than preconfigured keys because their lifetimes are much shorter. Known cryptographic attacks against WEP can be thwarted by reducing the length of time that an encryption key remains in use. Furthermore, encryption keys generated using EAP protocols are generated on a per-user and per-session basis. The keys are not shared among users, as they must be with preconfigured keys or preshared passphrases.

**NOTE:** 802.1X authentication is supported on Odyssey Access Client endpoints and on endpoints running non-Juniper 802.1X supplicants. With non-Juniper supplicants, you cannot use an Infranet Enforcer in the configuration. See “Task Summary: Configuring a Non-Juniper 802.1X Suppliant to Authenticate to the Infranet Controller” on page 161.

The Infranet Controller RADIUS server can fulfill RADIUS authentication requests from RADIUS clients that support 802.1X. (If you are using an external RADIUS server for authentication, you can use the Infranet Controller RADIUS proxy feature).

A RADIUS client, the network access device, accepts EAPOL (EAP over LAN) connection requests from 802.1X supplicants.

The network access device, which can be a wired switch or wireless access point, uses the RADIUS protocol to communicate with the Infranet Controller to authenticate and authorize endpoints before allowing them access to the network.

The Infranet Controller RADIUS server receives requests for authentication from the network access device and authenticates the endpoint. The Infranet Controller then sends the response back to the network access device.

The network access device and the Infranet Controller exchange messages in a series of request/response transactions. The network access device sends a request and expects a response from the Infranet Controller. If the response does not arrive, the network access device can retry the request periodically.

Figure 14 illustrates how the Infranet Controller functions as a RADIUS server for an 802.1X network access device within the UAC solution with Odyssey Access Client.
The endpoint connects to an 802.1X network access device. The endpoint and the Infranet Controller exchange EAP messages by means of 802.1X and RADIUS through the network access device. The EAP messages contain information about user credentials and the health of the endpoint.

The Infranet Controller uses its local or an external authentication server to verify the user’s identity.

If the Infranet Controller successfully authenticates the user, the Infranet Controller sends a message to the network access device to allow the endpoint access to the network. The type of access granted depends on the user’s identity and the health of the endpoint. For example, if the endpoint meets the requirements of all Host Checker policies, the user can have full network access. If the endpoint does not meet some security requirements, the user can be granted access to a remediation server. If the endpoint is using Odyssey Access Client as its 802.1X supplicant, the Infranet Controller and the endpoint exchange messages as necessary throughout a session (for example, to monitor the endpoint’s security compliance). If the endpoint is using a non-Juniper supplicant, Host Checker is not supported.

If the endpoint is using Odyssey Access Client, and the endpoint meets the requirements of all Host Checker policies when the user attempts to access a protected resource, the Infranet Controller sends auth table entries to the Infranet Enforcer to allow the user access to the protected resources. If the endpoint is using a non-Juniper supplicant, the Infranet Controller opens the network port.

**Task summary: Configuring the Infranet Controller as a RADIUS Server for an 802.1X Network Access Device**

To configure the Infranet Controller as a RADIUS server for an 802.1X network access device, perform these tasks:

1. Create a location group by using the **UAC > Network Access > Location Group** page of the admin console. A location group associates a sign-in policy with a group of network access devices. For more information, see “Configuring Location Groups” on page 145.
2. Create a RADIUS client using the **UAC > Network Access > RADIUS Client** page of the admin console. A RADIUS client specifies network access device parameters such as the IP address that enables the Infranet Controller to respond to the device. For more information, see “Configuring RADIUS Clients” on page 147.

**NOTE:** If you want to use an Infranet Enforcer as a RADIUS client of the Infranet Controller, you do **not** configure a RADIUS client for the Infranet Enforcer. For more information, see “Using an Infranet Enforcer as a RADIUS Client of the Infranet Controller” on page 159.

3. Optionally, create a RADIUS attribute policy using the **UAC > Network Access > RADIUS Attributes** page of the admin console. A RADIUS attribute policy associates RADIUS return attributes such as VLAN tunnel assignment with user roles. RADIUS return attributes determine how the endpoint is allowed to access the network. For more information, see “Configuring RADIUS Attributes Policies” on page 151.

**Configuring Location Groups**

You can use location groups to organize or logically group network access devices by associating the devices with specific sign-in policies. Sign-in policies provide a way to define and direct independent access control policies with the network. Location groups associate sign-in policies with network access devices.

A sign-in policy defines the realm(s) that the users of network access devices can use to access the Infranet Controller. When creating a sign-in policy, you associate it with the appropriate realm(s). When creating a realm, you associate it with an authentication server. Thus, by associating a location group with a sign-in policy, you can associate a group of network access devices with an authentication server along with the other realm settings such as an authentication policy and role mapping (see “Authentication Realms” on page 245).

For example, you might create location group policies to logically group the network access devices in each building at a corporate campus. You can also use location group policies to specify a special realm for MAC address authentication. See “Configuring Network Access Policies for Unmanageable Devices” on page 169.

As shown in the example in Figure 15, you can create two location group policies called **Wired** and **Wireless** to require different levels of authentication credentials from wired versus wireless endpoints. You might do this because you require the strictest authentication modes for your wireless access points, while your wired networks have an acceptable level of physical security.

In this example, each location group is associated with a different sign-in policy, each sign-in policy uses a different realm, and each realm uses a different authentication server.

- The “Wired” location group for wired switches is associated with a sign-in policy that uses an Active Directory authentication server. Users who connect to the network by using wired switches must sign in using Active Directory credentials.
For stricter authentication, the “Wireless” location group for wireless access points is associated with a sign-in policy that uses an ACE authentication server. Users who connect to the network by using wireless access points must sign in using their ACE server credentials. These credentials are a username and password that consists of the concatenation of a PIN and an RSA SecurID hardware token’s current value.

**Figure 15: Example of Using Location Groups to Logically Group Network Access Devices**

To configure a location group on the Infranet Controller:

1. Create a sign-in policy that you want to associate with the location group. For more information, see “Sign-In Policies” on page 257.

2. In the Infranet Controller admin console, choose **UAC > Network Access > Location Group**.

3. On the **New Location Group** page, enter a name to label this location group and an optional **Description**.

4. For **Sign-in Policy**, select the sign-in policy you want to associate with the location group.

5. If this location group is for controlling an unmanageable device using MAC address authentication, select a **MAC Authentication Realm** that you have created from the list. See “Controlling Unmanaged Device Access” on page 163.

6. Click **Save Changes**.
**Configuring RADIUS Clients**

To enable the Infranet Controller to respond to a network access device, you must configure a RADIUS client in the Infranet Controller with the following information about the device:

- The IP address of the network access device

  In large-scale deployments, if several network access devices use the same RADIUS attributes and have contiguous IP addresses, you can also specify a group of network access devices by using a contiguous range of IP addresses instead of an IP address for each device. When the Infranet Controller receives a RADIUS request that includes a source IP address in this range, it uses the RADIUS client policy for the range to determine the appropriate shared secret, make/model, and location group.

- The shared secret used by both the Infranet Controller and the network access device

- The make and model of the network access device, which you select from a list of devices in the Infranet Controller admin console

  The Infranet Controller supports a large number of specific network access devices by using its built-in standard RADIUS and vendor-specific, proprietary dictionary files, and you can upload new dictionaries to add new RADIUS clients.

  The Infranet Controller uses the dictionary files to store lists of RADIUS attributes, parse authentication requests, and generate responses. See “Uploading a New RADIUS Client Dictionary” on page 148.

  When you select the device’s make and model in a RADIUS client policy, you are selecting a dictionary file that contains the vendor-specific attributes (VSAs) for that device. Whenever the Infranet Controller receives a RADIUS packet from that device, it consults the dictionary file for any non-standard attributes that it encounters in the packet. If you do not know the make and model of a device, you can use the standard RADIUS attributes by choosing the **Standard RADIUS** setting in a RADIUS client policy.

---

**NOTE:**

- You must also configure each network access device with the following information:
  - The IP address of the Infranet Controller
  - The shared secret you specified in the RADIUS client policy for the device

  See the documentation provided with the network access device for configuration instructions.

---

Also in this section:

- “Uploading a New RADIUS Client Dictionary” on page 148
“Creating a RADIUS Dictionary Based on an Existing Model” on page 149

“Creating a New RADIUS Client” on page 150

**Uploading a New RADIUS Client Dictionary**

The Infranet Controller uses dictionary files to store lists of RADIUS attributes. The Infranet Controller uses these dictionaries to parse authentication/accounting requests and generate responses.

The main dictionary file (*radius.dct*) lists attributes defined by the RADIUS standard.

In addition to the standard attributes, many network access devices use Vendor-Specific Attributes (VSAs) to complete a connection. The Infranet Controller supports a large number of specific network access devices by providing vendor-specific, proprietary dictionary files.

During Infranet Controller configuration, when you make a selection in the RADIUS Client Make/Model field, you are telling the server which dictionary file contains the VSAs for this client device. Thereafter, whenever the server receives a RADIUS packet from this client device, it can consult this dictionary file for any nonstandard attributes that it encounters in the packet. Standard RADIUS attributes are always defined by the *radius.dct* file.

**UAC > Network Access > RADIUS Dictionary** displays all of the built-in RADIUS dictionaries on the Infranet Controller. You can upload new dictionaries to define makes and models that are not pre-configured on the Infranet Controller, and you can copy and modify existing dictionaries.

To upload a new RADIUS client dictionary to the Infranet Controller:

1. In the Infranet Controller admin console, choose **UAC > Network Access > RADIUS Dictionary**. This page displays the pre-configured dictionaries and the associated vendors.

2. Click **New RADIUS Dictionary**.

3. Enter a **Name** and optional **Description** for the new Dictionary.

4. Use the **Browse** button to search for the dictionary file (.dct) on a local or connected drive, then **Save Changes**. The uploaded dictionary appears on the main **RADIUS Dictionary** page, and in the **Make/Model** pull-down list on the **RADIUS Client** page.

---

**NOTE:**

- You can only remove dictionaries that are not associated with a vendor.

- You can download any dictionary from the list, including pre-installed dictionaries. You can modify the downloaded dictionary, then upload it as a new make/model.

5. Click **Save Changes**. The new dictionary appears in the **RADIUS Dictionary** list.
Creating a RADIUS Dictionary Based on an Existing Model

To create a new RADIUS dictionary based on an existing manufacturer's model:

1. In the Infranet Controller admin console, choose UAC > Network Access > RADIUS Dictionary. This page displays the listing of pre-configured dictionaries on the Infranet Controller and their associated vendors.

2. Click on the existing dictionary that you would like to copy.

3. Click the .dct file to download the existing dictionary.

4. Make the required modifications to the downloaded .dct file and rename the file.

5. From the UAC > Network Access > RADIUS Dictionary page, click New RADIUS Dictionary.

6. Browse for the file you have modified, and enter a new Name and optional Description for the new Dictionary.

7. Click Save Changes to upload the modified .dct file. The modified file appears on the RADIUS Dictionary page. Note that there is no vendor associated with the new dictionary.

8. From the UAC > Network Access RADIUS Vendor page click New RADIUS Vendor.

9. Enter a new Name and optional Description for the new RADIUS vendor.

10. Select the new dictionary you have created from the pull-down menu.

11. Click Save Changes. The new vendor and the associated dictionary will appear on the RADIUS Vendor page.
Creating a New RADIUS Client

Table 19: Before Beginning

<table>
<thead>
<tr>
<th>Topic</th>
<th>Detail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overlapping IP address ranges</td>
<td>The address range assigned to one group of network access devices in a RADIUS client cannot overlap the address ranges assigned in another RADIUS client.</td>
</tr>
<tr>
<td>Starting IP address range restrictions</td>
<td>The starting address of the address range assigned to a group of network access devices cannot be the same as the IP address of an individual network access device. The starting address of the address range assigned to a group of network access devices cannot be the same as the IP address of an individual network access device.</td>
</tr>
<tr>
<td>IP address range restrictions</td>
<td>If an individual network access device has an IP address that falls within an address range assigned to a group of network access devices, the Infranet Controller uses the RADIUS client for the individual network access device. For example, suppose an individual network access device is configured in the NAD1 RADIUS client policy with IP address 192.168.21.55, and a group of network access devices is configured in the BLDG1 RADIUS client policy with an IP address range of 192.168.21.50–192.168.21.60. If the Infranet Controller receives a RADIUS request from 192.168.21.55, it uses the NAD1 RADIUS client information. If the Infranet Controller receives a RADIUS request from 192.168.21.56, it uses the BLDG1 RADIUS client information.</td>
</tr>
<tr>
<td>IP address limitations</td>
<td>A RADIUS client for a group of network access devices cannot use a Class D, E, or F IP address (that is, an address greater than 223.255.255.0).</td>
</tr>
<tr>
<td>Shared secret</td>
<td>You must configure the network access device with the same shared secret that you enter in the Infranet Controller. If you change a shared secret, your connection will be disrupted. Choose a complex password initially in accordance with your security policies.</td>
</tr>
<tr>
<td>Radius Dictionary</td>
<td>If you are not sure which make and model switch you are using or if your device is not in the list, select - Standard RADIUS - for Make/Model. Alternately, you can upload additional dictionaries to add a new network access device. See “Uploading a New RADIUS Client Dictionary” on page 148.</td>
</tr>
<tr>
<td>RFC3680</td>
<td>If the network access device is not fully RFC compliant and does not accept RFC3680 Tunnel Attributes with tags, select - Standard RADIUS: No VLAN tags - for Make/Model.</td>
</tr>
</tbody>
</table>

To create a RADIUS client on the Infranet Controller:

1. If you have not already done so, configure a location group. At least one location group is required before you can configure a RADIUS client. For more information, see “Configuring Location Groups” on page 145.

2. In the Infranet Controller admin console, choose UAC > Network Access > RADIUS Client.

3. Click New RADIUS Client.
4. On the RADIUS Client page, enter a name to label this RADIUS client. Although you can assign any name to a RADIUS client entry, you should use the device’s SSID or IPv4 address to avoid confusion.

5. For Description, enter an optional description.

6. For IP Address, enter the IP address of the network access device.

7. (Optional) For IP Address Range, enter the number of IP addresses in the IP address range for the network access devices, starting with the address you specified for IP Address. You can specify a range up to a maximum of 32,768 addresses.

8. For Shared Secret, enter the RADIUS shared secret. A RADIUS shared secret is a case-sensitive password used to validate communications between the Infranet Controller and network access device. The Infranet Controller supports shared secrets of up to 127 alphanumeric characters, including spaces and the following special characters:

```
~!@#$%^&*()_+|\=-'{}[]:";'<>?/.,
```

9. For Make/Model, select the make and model of the network access device. The make/model selection tells the Infranet Controller which dictionary of RADIUS attributes to use when communicating with this client.

10. For Location Group, select the location group you want to use with this network access device. (See “Configuring Location Groups” on page 145.)

11. Click Save Changes.

**Configuring RADIUS Attributes Policies**

You can configure RADIUS attributes policies on the Infranet Controller to send return list attributes to an 802.1X network access device. For example, you can specify which VLAN endpoints must use to access the network. (See “Using a RADIUS Attributes Policy to Specify VLANs for Endpoints” on page 152). You can also configure other functions on a network access device’s port based on the role assigned to the user who is currently using that port. For example, a particular switch might let you use return list attributes to configure Quality-of-Service (QoS) functions (Bandwidth and/or Priority) on the device’s port based on the current user’s role.

A return list is a set of attributes that the Infranet Controller returns to the network access device after authentication succeeds. The return list usually provides additional parameters that the network access device needs to complete the connection. Return list attributes are authorization configuration parameters.

The specific attributes in each RADIUS packet depend upon the network access device or RADIUS server that sent the packet. Different kinds of network access devices may require different attributes to control their behavior.

In the RADIUS attributes policy, you can select RADIUS attributes by name from a predefined list. For each attribute, you specify values using strings or numbers.
By default, the Infranet Controller sends a session timeout value on all RADIUS accepts that is equal to the timeout value of the configured session length. You can bypass the default timeout.

If you do not want to assign endpoints to a VLAN, or return any RADIUS attributes, you can select the Open Port option. With this check box selected, the Infranet Controller will not return any RADIUS attributes.

See “Use Cases for Using RADIUS Attributes Policies” on page 156.

Also in this section:
- “Using a RADIUS Attributes Policy to Specify VLANs for Endpoints” on page 152
- “Enabling Endpoints to Connect to the Infranet Controller” on page 154
- “Creating a RADIUS Attributes Policy” on page 154
- “Use Cases for Using RADIUS Attributes Policies” on page 156
- “Use Case: Using RADIUS Attributes with Odyssey Access Client to Avoid Disconnecting Concurrent Network Connections” on page 158

---

**NOTE:**

- Be sure to select the correct make and model of the network access device (see “Configuring RADIUS Clients” on page 147.) During authentication, the Infranet Controller filters the return list based on the dictionary for the network access device that sent the authentication request. The Infranet Controller omits any return list attribute that is not valid for the device.

- You can return RADIUS attributes that are in the installed dictionaries or in dictionaries you have uploaded to the Infranet Controller. See “Uploading a New RADIUS Client Dictionary” on page 148.

- The RADIUS return attributes are based on the first RADIUS attributes policy that matches both the location group of the network access device and the roles assigned to the user.

---

**Using a RADIUS Attributes Policy to Specify VLANs for Endpoints**

You can use a RADIUS attributes policy to specify a VLAN that endpoints must use to access the network. You can also specify how endpoints to which the RADIUS attributes policy is applied must communicate with the Infranet Controller once they are on the network.
Figure 16 illustrates an example of using a RADIUS attributes policy to specify VLANs for endpoints.

**Figure 16: Using a RADIUS Attributes Policy to Specify VLANs for Endpoints**

- To enable the Infranet Controller to receive traffic on all VLANs, you can connect the Infranet Controller internal interface to the trunk port on a VLAN-enabled switch that sees all VLAN traffic.

- Because User 1 is authenticated and her endpoint complies with Host Checker security policies, she is assigned a role on the Full Access VLAN that allows her full network access and access to protected resources.

- Although User 2 is authenticated, his endpoint does not comply with Host Checker security policies; therefore, he is assigned a role on the Quarantine VLAN that only allows him access to a remediation server.
Enabling Endpoints to Connect to the Infranet Controller

After an endpoint successfully authenticates to the Infranet Controller and gains access to the network, the endpoint connects to the Infranet Controller. The Infranet Controller can continuously monitor the health status of the endpoint and apply any policy changes.

To enable endpoints to connect to the Infranet Controller, use one of the following configurations:

- If you are using more than two VLANs, connect the Infranet Controller internal interface to the trunk port on a VLAN-enabled switch that sees all of the VLAN traffic. You must also configure a RADIUS attributes policy with the Automatic setting, which enables the Infranet Controller to take advantage of VLAN tagging. When connected to a trunk port on a VLAN-enabled switch, the Infranet Controller detects traffic from all VLANs. This is useful if you want to configure separate VLANs for separate classes of users or endpoints, and you want to make the Infranet Controller accessible from all VLANs.

  In this configuration, you must also create VLAN ports on the Infranet Controller and specify an existing VLAN ID on the network infrastructure. See “Configuring VLANs” on page 333.

- You can also configure routing on your network to enable endpoints to access the Infranet Controller over the network. In this case, you must configure RADIUS attributes policies with the VLAN IDs you are using for endpoints, but you do not need to configure any VLAN ports on the Infranet Controller.

Creating a RADIUS Attributes Policy

Before configuring a RADIUS attributes policy, verify the following configuration on the network access devices you want to use with the Infranet Controller:

- The network access device must support RADIUS-based, dynamic VLAN assignment if the VLAN check box is selected.

- The ports must be 802.1X enabled.

- The VLAN IDs you want to use in the Infranet Controller RADIUS VLAN policies must be configured on the network access devices if the VLAN check box is selected.

- The endpoints must be able to obtain an IP address from a DHCP server that is in the VLAN you are using.

Any modifications to the RADIUS attributes page will cause endpoints with sessions associated with the attributes policy to re-connect. It is recommended that you schedule any changes at a time when endpoints will not be affected.

To configure a RADIUS attributes policy:

1. In the Infranet Controller admin console, choose **UAC > Network Access > RADIUS Attributes**.

2. Click **New Policy**.
3. On the **New Policy** page:
   a. For **Name**, enter a name to label this policy.
   b. For **Description**, enter an optional description.

4. Under **Location Group**, select the location groups to which you want to apply this policy and click **Add**. To apply the policy to all location groups, do not add any location groups and leave the default setting (all) listed in the **Selected Location Groups** list.

5. Under **RADIUS Attributes**, select the following options. For more information on using these options, see “Use Cases for Using RADIUS Attributes Policies” on page 156.

   - **Open Port**—Select this option if you do not want to assign endpoints to a VLAN or return any RADIUS attributes. Selecting this check box disables all other RADIUS Attributes options.

   - **VLAN**—Select VLAN to configure VLAN assignment according to RFC 3580 by returning the RADIUS tunnel attributes to the network access device. Specify the existing VLAN ID on the network infrastructure that you want to use for the role(s) to which this policy applies. Selecting this option is equivalent to manually specifying the three RFC 3580 RADIUS tunnel attributes in the **Return Attribute** section.

   - **Return Attribute**—To specify the return attributes you want sent to the network access device, select **Return Attribute** and then do the following:
     1. From the **Attribute** drop-down list, select the return attribute you want to send. For **Value**, enter the value for the selected attribute, and then click **Add**.
     2. You can specify multiple return attributes and values for this policy.
     3. To add an attribute, select a new attribute from the list and enter the appropriate value. To change an attribute value, click the value, enter the appropriate value, and then click the check mark icon next to the value.
     4. To rearrange the order in which you want to send the return attributes, select the check box next to the attribute name and then click the up or down arrow icons.
     5. To delete an attribute, select the check box next to the attribute name and then click **Delete**.

   - **Add Session-Timeout attribute with value equal to the session lifetime**—Clear this checkbox to prevent the Infranet Controller from sending a session timeout value equal to the timeout value of the configured session length on all RADIUS accepts. This allows you to set the re-authentication timer statically on the switch port, if required.
6. For **Interface**, specify the Infranet Controller network interface to which you want endpoints affected by this policy to use to connect to the Infranet Controller:

- **Automatic (use configured VLANs)**—Choose this option to use VLAN tagging (see configuration #2 in “Enabling Endpoints to Connect to the Infranet Controller” on page 154). You must also connect the Infranet Controller internal interface to the trunk port on a VLAN-enabled switch that sees all of the VLAN traffic.

- **Internal**—Choose this option if the endpoints using this RADIUS attributes policy should use the IP address of the Infranet Controller’s internal interface to communicate with the Infranet Controller.

- **External**—Choose this option if the endpoints on the configured VLAN should use the IP address of the Infranet Controller’s external interface to communicate with the Infranet Controller.

7. In the **Roles** section, specify:

- **Policy applies to ALL roles**—Choose this option to apply this policy to all users.

- **Policy applies to SELECTED roles**—Choose this option to apply this policy only to users who are mapped to roles in the **Selected roles** list. Be sure to add roles to this list from the **Available roles** list.

- **Policy applies to all roles OTHER THAN those selected below**—Choose this option to apply this policy to all users except for those who map to the roles in the **Selected roles** list. Be sure to add roles to this list from the **Available roles** list.

8. Click **Save Changes**.

**Use Cases for Using RADIUS Attributes Policies**

This section describes how to use the **RADIUS Attributes** options on RADIUS attributes policies to configure network access devices in the following use cases.

**Use Case 1: Configuring VLAN Assignment by Returning RADIUS Tunnel Attributes**

This use case describes how to configure VLAN assignment on network access devices by returning RADIUS tunnel attributes according to RFC 3580.

1. On the **UAC > Network Access > RADIUS Attributes** page, select **VLAN**.

2. Specify a VLAN ID.

**Use Case 2: Configuring VLAN Assignment or Policies by using the Filter-ID Return Attribute**

This use case describes how to configure VLAN assignment or other policies on network access devices by using the Filter-ID return attribute.

1. On the **UAC > Network Access > RADIUS Attributes** page, select **Return Attribute**.
2. Select **Filter-ID** from the **Attribute** drop-down list.

3. Specify the policy name for **Value**.

4. Configure the filter on the network access device.

**Use Case 3: Configuring VLAN Assignment Along with Other Attributes**

This use case describes how to configure VLAN assignment and other features on network access devices by returning RADIUS tunnel attributes in addition to returning other attributes.

1. On the **UAC > Network Access > RADIUS Attributes** page, select **VLAN**.

2. Specify a VLAN ID.

3. Select **Return Attribute**.

4. Select the attribute you want to return from the **Attribute** drop-down list.

5. Specify an attribute value for **Value**.

**Use Case 4: Configuring VLAN Assignment in a Heterogeneous Environment**

This use case describes how to configure VLAN assignment or other policies on network access devices by using the Filter-ID return attribute.

This use case assumes you have a heterogeneous network environment that includes network access devices from a variety of vendors. For example, you might have one type of switch that supports RADIUS tunnel attributes only, a second type of switch that supports the Filter-ID return attribute only, and third type of switch that supports both.

1. Using the **UAC > Network Access > Location Group** page, create a location group policy for each type of network access device. (See “Configuring Location Groups” on page 145.) For example:
   - a. Create a location group policy for switches that support RADIUS tunnel attributes only.
   - b. Create a second location group policy for switches that support the Filter-ID return attribute only.
   - c. Create a third location group policy for switches that support both RADIUS tunnel attributes and the Filter-ID return attribute.

2. Using the **UAC > Network Access > RADIUS Client** page, create a RADIUS client policy for each type of network access device and associate each RADIUS client policy with the appropriate location group. (See “Configuring RADIUS Clients” on page 147.) For example:
   - a. Create a RADIUS client policy and specify a make/model for **Make/Model** that supports the RADIUS tunnel attributes. Associate this policy with the location group policy for switches that support RADIUS tunnel attributes only.
b. Create a second RADIUS client policy and specify a make/model that supports the Filter-ID return attribute. Associate this policy with the location group policy for switches that support the Filter-ID return attribute only.

c. Create a third RADIUS client policy and specify a make/model that supports the both RADIUS tunnel attributes and the Filter-ID return attribute. Associate this policy with the location group policy for switches that support both RADIUS tunnel attributes and the Filter-ID return attribute.

3. Using the UAC > Network Access > RADIUS Attributes page:

   a. Create a RADIUS Attributes policy that specifies only the VLAN option and a value for VLAN ID. Associate this policy with the location group policy for switches that support RADIUS tunnel attributes only.

   b. Create a second RADIUS Attributes policy that specifies only the Filter-ID option from the Attribute drop-down list and a policy name for Value. Associate this policy with the location group policy for switches that support the Filter-ID return attribute only.

   c. Create a third RADIUS Attributes policy that specifies both the VLAN option and a value for VLAN ID, and the Filter-ID option with a policy name for Value. Associate this policy with the location group policy for switches that support both RADIUS tunnel attributes and the Filter-ID return attribute.

   NOTE: If all of the dictionaries are correct, it is not necessary to create three separate RADIUS attributes policies. The Infranet Controller will strip out attributes that do not conform to the RADIUS client’s dictionaries.

Use Case 5: Configuring any Attribute

This use case describes how to configure a feature on network access devices by returning attributes.

1. On the UAC > Network Access > RADIUS Attributes page, select Return Attribute.

2. Select the attribute you want to return from the Attribute drop-down list.

3. Specify an attribute value for Value.

Use Case: Using RADIUS Attributes with Odyssey Access Client to Avoid Disconnecting Concurrent Network Connections

You can configure RADIUS attributes to work with a connected switch to prevent expired sessions from disconnecting concurrent network connections.

When an Infranet Controller session reaches its maximum lifetime, as controlled by the Session Options tab on the Role settings configuration page, all access to the network through UAC is terminated. If Odyssey Access Client is used for access, the Odyssey Access Client logs off from the network (via EAPoL-LogOff). Any access provisioned through the Infranet Enforcer is removed.
The Odyssey Access Client then attempts to start a new session. If a new session can be established, network connection is reprovisioned. However, in most cases any tcp connections that were established prior to the end of the Infranet Controller session expire and must be reestablished. For example, any remote desktop or telnet sessions will end and the user must restart them.

You can configure a timeout on a switch or wireless access point that is shorter than the Infranet Controller session lifetime so that the Infranet Controller can periodically verify that Odyssey Access Client is still operating correctly. A shorter session timeout on a switch or wireless access point may be configured in a number of ways, including:

- You can configure a shorter Session-Timeout RADIUS return attribute in RADIUS Attributes policies. Depending on the switch or wireless access point. It may also be necessary to configure a Timeout-Action RADIUS return attribute, and it may be necessary to configure the switch or wireless access point so that it will respond to these attributes.

- You can configure the switch or wireless access point with a shorter session timeout. The switch or wireless access point will also need to be configured to ignore Session-Timeout RADIUS return attributes from the Infranet Controller.

When the switch or wireless access point times out a session, the Odyssey Access Client can resume the Infranet Controller session by interacting with the Infranet Controller without interrupting network access. There are two ways that this can be done on the Odyssey Access Client:

- TTLS session resumption—Odyssey Access Client reauthenticates to the Infranet Controller based on TLS keying material from the previous session.

- DSID session resumption—this happens when TTLS session resumption fails but the Infranet Controller session is still valid. TTLS session resumption can fail if Odyssey Access Client is configured for a shorter TTLS session resumption maximum than the length of the IC session. In DSID session resumption, Odyssey Access Client authenticates to the Infranet Controller using new TLS keying material, but without creating a new IC session. You configure Session Resumption on the Odyssey Access Client Tools > Options panel.

### Using an Infranet Enforcer as a RADIUS Client of the Infranet Controller

You can use an Infranet Enforcer as an 802.1X RADIUS client of the Infranet Controller. Unlike all other network access devices, you do not configure a RADIUS client for the Infranet Enforcer. When you use the following instructions, the Infranet Controller automatically creates an internal RADIUS client for the Infranet Enforcer that you cannot change. This RADIUS client for the Infranet Enforcer is not displayed in the Infranet Controller admin console.

To use an Infranet Enforcer as an 802.1X RADIUS client of the Infranet Controller, perform these tasks:

1. Configure a location group policy for the Infranet Enforcer:
   
   a. Create a sign-in policy that you want to associate with the location group. For more information, see “Sign-In Policies” on page 257.
b. Choose UAC > Network Access > Location Group.

c. Click New Location Group.

d. On the New Location Group page, enter a name to label this location group policy.

e. For Description, enter an optional description.

f. For Sign-in Policy, select the sign-in policy you want to associate with the location group.

g. Click Save Changes.

For more information on location groups, see “Configuring Location Groups” on page 145.

2. Associate the location group with the Infranet Enforcer:

a. On the UAC > Infranet Enforcer > Connection page, click the name in the Enforcer column of the Infranet Enforcer that you want to use as an 802.1X RADIUS client of the Infranet Controller.

b. On the UAC > Infranet Enforcer > Connection page, select the location group from the Location Group list.

c. Click Save Changes.

3. To configure how authenticated endpoints connect to the Infranet Controller, specify VLANs in RADIUS attributes policies by using the UAC > Network Access > RADIUS Attributes page of the admin console. See “Configuring RADIUS Attributes Policies” on page 151.

Using a Non-Juniper 802.1X Supplicant

You can configure users to authenticate to the Infranet Controller using Odyssey Access Client, or you can use a non-Juniper 802.1X supplicant. Odyssey Access Client is preconfigured with standard protocols to work with the Infranet Controller. To use a non-Juniper supplicant you must configure the authentication protocols manually. A non-Juniper supplicant is any client that is configured without the JUAC protocol.

For example, the Microsoft Vista built-in supplicant allows you to choose from a selection of authentication protocols for inner and outer authentication. To permit the client to authenticate to the Infranet Controller, you choose the protocols on the endpoint, then select corresponding protocol sets on the Infranet Controller, depending on the authentication server type you are using. See “Authentication Server Limitations with Authentication Protocols” on page 176.

You must also install a certificate on the client machine and select the certificate as a trusted root CA. The certificate should be generated from the same Certificate Authority (CA) that the Infranet Controller is using for trusted client CAs.
Using the Infranet Controller for 802.1X Network Access

Chapter 5: The UAC RADIUS Server and Layer 2 Access

IF you configure endpoints to connect through layer 2 with non-Juniper supplicants, layer 3 functionality of the Infranet Controller is not supported, and the user cannot choose a realm or a role interactively. Configuration options like Host Checker, session limits, and other restrictions are not applied.

For non-Juniper supplicants, a user name suffix can be used to select a realm in the form user@realm. If a suffix is not used, there are additional options for specifying a realm. See “Authentication Realms and Protocols in User Sign-in Policies” on page 260.

Windows Vista and XP Service Pack 3 supplicants are supported. If you use these clients, you can take advantage of Statement of Health (SOH) policies in a Host Checker policy. See “Statement of Health Integration” on page 305.

Task Summary: Configuring a Non-Juniper 802.1X Supplicant to Authenticate to the Infranet Controller

Table 20: Before Beginning

<table>
<thead>
<tr>
<th>Topic</th>
<th>Detail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wildcard characters in hostname</td>
<td>With Odyssey Access Client when you connect with an Infranet Controller that you have not connected with before, certificate information is presented for the user to accept and trust dynamically. With non-Juniper 802.1X supplicants, you must install the certificate before attempting to connect to the Infranet Controller.</td>
</tr>
<tr>
<td>Odyssey Access Client sign-in</td>
<td>When a non-Juniper supplicant attempts to connect to the Infranet Controller and more than one realm is available, the user can select a realm by adding a suffix to the outer user name with @realm-name. If no suffix is present, and you have configured a sign-in policy with more than one realm, the Infranet Controller searches for a realm whose authentication server supports the authentication protocol that the endpoint requests. For example, if CHAP is requested, the Infranet Controller skips any realms that use an Active Directory server. See “Authentication Realms and Protocols in User Sign-in Policies” on page 260.</td>
</tr>
<tr>
<td>Outer proxy realms</td>
<td>Host Checker is not downloaded to endpoints that connect with non-Juniper supplicants. If a realm or a role includes Host Checker restrictions, only endpoints with Odyssey Access Client can pass the restrictions. Non-Juniper clients can not sign in to the role or realm.</td>
</tr>
<tr>
<td>Anonymous authentication servers</td>
<td>You must configure the access point to send accounting stops to enable the Infranet Controller to log when a session ends and update the session tables.</td>
</tr>
<tr>
<td>Username suffixes</td>
<td>By default, the User may specify the realm name as a username suffix check box is not selected. If you choose this option, non-UAC endpoints access the Infranet Controller by entering their credentials in the format user@realm.</td>
</tr>
<tr>
<td>Proxy realm sign-in</td>
<td>If you configure a sign-in policy with multiple realms, and one of the realms is a proxy realm, the user must append a suffix to the username to access the proxy realm.</td>
</tr>
</tbody>
</table>

To configure a Non-Juniper Supplicant:

1. Configure authentication protocols on the third-party supplicant per the instructions in the vendor’s documentation.

3. Install the certificate from the CA that the Infranet Controller is using for trusted Client CAs.


5. Create a role for the user(s) that will access the Infranet Controller using a non-Juniper supplicant. See “Configuring User Roles” on page 120.

6. Create a realm for the endpoint using the Users > User Realms page. Use role mapping to associate the role you created for non-Juniper Networks supplicants with the realm. For authentication server, select the Certificate Server you created. See “Creating an Authentication Realm” on page 246.

7. Create a new sign-in policy from the Authentication > Signing In > Sign-In Policies page of the admin console. Associate the authentication protocol set you created with the realm you created for this connection. See “Configuring Sign-In Policies” on page 259.

8. Configure a new location group from the UAC > Network Access > Location Group page and select the sign-in policy that you previously created from the Sign-in Policy list. See “Configuring Location Groups” on page 145.

9. Create a new RADIUS client from the UAC > Network Access > RADIUS Client page and select the location group that you previously created from the Location Group list. See “Configuring RADIUS Clients” on page 147.

10. Create a role for endpoints using the third-party supplicant from the User > User Roles page. See “Configuring User Roles” on page 120.

11. Configure a RADIUS attributes policy from the UAC > Network Access > RADIUS Attributes page and select the location group created for this connection from the Location Group section, and select the role(s) configured for this access from the Roles section. See “Configuring RADIUS Attributes Policies” on page 151.

12. Complete the remaining steps in “Using the Infranet Controller for 802.1X Network Access” on page 142.

Configuring Access to Switches and Access Points from a Browser

Some switches support web-based port authentication with CHAP, PAP, or EAP-MDS Challenge (non-tunneled) authentication. You can configure the Infranet Controller RADIUS server to support this functionality.

When a PC is connected to a port via captive portal, the PC receives an IP address from the local DHCP server resident on the switch.
Chapter 5: The UAC RADIUS Server and Layer 2 Access

If a user browses to a properly configured switch, an authentication page is presented by the switch. After the user submits the proper credentials, the switch consults the Infranet Controller RADIUS server.

On successful authentication, the temporary IP address expires, and the port is opened to the user. The PC then gets an IP address from the network DHCP server and access to the network is allowed.

Additionally, some switches can authenticate the administrator by consulting a RADIUS server using these protocols.

Task Summary: Configuring the Infranet Controller to Authenticate with Non-Tunneled Protocols

Following are the basic instructions for configuring the Infranet Controller to authenticate users through a switch using non-tunneled protocols:

1. Configure an external server or the local auth server to include authentication credentials for the device.


3. Create a new role. It is not necessary to specify detailed role options. See “Configuring User Roles” on page 120.

4. Create a new realm that references the authentication server from the Users > User Realms page. See “Creating an Authentication Realm” on page 246.

5. Create a new protocol set to include CHAP, PAP or EAP-MD5 Challenge at Authentication > Signing In > Authentication Protocols. See “Configuring Authentication Protocol Sets” on page 135.

6. Create a sign-in policy at Authentication > Signing In > Sign-In Policy and specify the default sign-in page, the protocol set you have created and the new realm. See “Task summary: Configuring Sign-In Policies” on page 258.

7. Create a location group from the UAC > Network Access > Location Group page and set the sign-in policy to the sign-in policy created for CHAP authentication. See “Configuring Location Groups” on page 145.

8. Configure a RADIUS client at UAC > Network Access > Radius Client to specify the new location group. See “Configuring RADIUS Clients” on page 147.

9. Configure the switch in accordance with the manufacturer’s instructions.

Controlling Unmanaged Device Access

Unmanaged devices are devices that cannot run Odyssey Access Client, supplicants or web browsers. Examples of unmanaged devices include IP phones, printers and NAS appliances. You can configure the Infranet Controller to authenticate these unmanaged devices using MAC address authentication.
Unmanageable devices each have a unique MAC address. With MAC-based authentication the MAC address serves as both the username and the password.

MAC address authentication is deployed at the edge of the network to provide port-based security. MAC address authentication uses RADIUS as the method for information exchange.

When a device connects to a switch, the switch forwards the MAC address to the Infranet Controller as the login credential. The Infranet Controller RADIUS server consults the authentication server, either a local database or an external LDAP server, and allows or denies access to the device based on whether there is a matching entry.

MAC addresses are not generally guarded as secrets, so an attacker could obtain a MAC address and pose as the device, gaining network access. For added security, access should be limited by creating a special VLAN for each device type.

After you direct unmanaged devices to a default VLAN, other resources in the VLAN can access the device. For example, if a printer that is plugged into a UAC-integrated switch is registered as a print server on the default VLAN, hosts that can access that VLAN on the network can access the printer.

You can add MAC addresses manually, you can provision MAC address authentication server from an external LDAP server, or you can utilize a third-party device that can profile endpoints and detect MAC addresses on the network.

**NOTE:** MAC-based authentication is not as secure as agent access or agentless access authentication. A MAC address can be spoofed, so use appropriate caution in granting MAC-authenticated devices access to sensitive areas.

**Task Summary: Configuring the Infranet Controller to Allow Unmanaged Device Access**

To allow access for unmanaged devices, you must perform these basic tasks:

1. Configure the necessary VLANs on your internal network to accommodate the different devices that you want to allow. On the Infranet Controller, you assign devices to VLANs through the location groups that are added to RADIUS attributes policies. See “Configuring a Location Group for MAC Address Authentication” on page 170.

In Figure 17 an example network is configured with different phones and printers, an external LDAP server and separate VLANs for different devices. MAC address authentication on the Infranet Controller is extremely flexible, and you can configure your network using any or all of these components.
2. Create MAC address authentication server(s), and populate the server(s) with MAC addresses and wildcards from the Authentication > Auth. Servers page. Use the MAC address for both the user name and the password. See “Configuring a MAC Address Authentication Server for Unmanageable Devices” on page 242 for more information.

**NOTE:**


- Optionally, you can configure an external LDAP server or a third-party appliance to monitor and classify devices on the network. See “MAC Address Authentication with Third-Party Appliance Integration” on page 166 and “Use Case: Using an External LDAP Server for MAC Address Authentication” on page 167.

3. Create MAC address realms that reference the authentication server(s) or LDAP server(s) from the UAC > MAC Address Realms page. See “Creating a MAC Address Realm” on page 169 for more information.

4. Create a sign-in policy and an authentication protocol set to associate with the MAC address realm from the Signing In > Sign-in Policies page. See “Defining User Sign-in Policies” on page 263.
5. Create location groups that reference the realms from the **UAC > Network Access > Location Groups** page. See “Configuring a Location Group for MAC Address Authentication” on page 170 for information.

6. Create RADIUS client policies for the switches that reference the applicable location groups from the **UAC > Network Access > RADIUS Client** page. See “Configuring a RADIUS Client for MAC Address Authentication” on page 171.

   - Create roles from the **Users > Roles** page, and give the authentication server role mappings through the realm as required. You must configure a session length for the role that is appropriate for the re-authentication interval of the switch. See “Creating a Role for MAC Address Authentication” on page 125 for more information.

   ![NOTE:](image) Do not configure any role restrictions, otherwise roles cannot get assigned to devices, and do not apply any Host Checker policies at the role or realm level.

7. Configure RADIUS attributes, to include the applicable VLAN assignments from the **UAC > Network Access > RADIUS Attributes** page. See “Configuring RADIUS Attributes for MAC Address Authentication” on page 171 for more information.

8. Configure the switch to communicate with the Infranet Controller for MAC address authentication. The Infranet Controller supports HP Procurve, Cisco Catalyst, and Nortel Secure Network Access switches. You will need to configure the following options on the switch:

   - Configure the desired ports to use the appropriate VLAN for unauthenticated traffic
   - Configure the ports to perform MAC-based RADIUS authentication
   - Specify the Infranet Controller as the RADIUS server, with the appropriate shared secret and IP address.

The HP and Cisco switches can use CHAP and EAP-MD5-Challenge protocols for MAC address authentication with the user name (the MAC address) as the clear text password. By default, the Nortel switch uses PAP, with a password in the format `<MAC address>..` Juniper Networks recommends using PAP with the Nortel switch.

See the Supported Platforms Guide on the *Juniper Networks Customer Support Center* for information about devices that are supported with this feature.

**MAC Address Authentication with Third-Party Appliance Integration**

The Infranet Controller can utilize a third-party solution to supplement MAC address identification and authentication.

Some third-party appliances have the functionality to detect and categorize network objects based on MAC addresses. These appliances allow you to arrange devices into types or profiles that serve a common functionality. You can map specific types or profiles to one or more roles on the Infranet Controller. The Infranet Controller uses LDAP to query the appliance for MAC addresses of interest.
You configure the third-party device to monitor your network and observe traffic to recognize and classify the types of devices that are on the network. The third-party device can then serve as the LDAP interface for the Infranet Controller to properly assign devices to the appropriate VLAN.

When you integrate the third party appliance into a heterogeneous network consisting of IP phones, printers, computer work stations, or any type of device that has a MAC address, devices in the network will automatically be enrolled in a profile type, for example “IP Phone”. You can then configure the appliance to interoperate with the Infranet Controller. See the Juniper Solutions Guide Using a Third-party Device for MAC Address Authentication.

**Use Case: Using an External LDAP Server for MAC Address Authentication**

If you are using an external LDAP server, you can configure the LDAP server to interface with the Infranet Controller instead of manually entering MAC addresses to the MAC address authentication type server.

This configuration represents one example of an LDAP implementation with the Infranet Controller. Refer to your vendor’s LDAP instructions for specific details.

1. Populate your external LDAP server with MAC address entries for devices on your network that you would like to provision through the Infranet Controller. The MAC address should be both the user name and the password.

2. On the Infranet Controller, create an LDAP server instance.
   - **Name:** MyLDAPAuthServer
   - **Authentication Required**
     - Authentication Required: Yes
     - **Admin DN:** cn = root, o = appliance
     - **Password:** ********
   - **Finding User Entries**
     - **Base DN:** o = appliance
     - **Filter:** (& (objectClass = ieee802Device) (macAddress = <USER>))
   - **Determining Group Membership**
     - **Base DN:** o = appliance
     - **Filter:** (& (objectClass = groupOfUniqueNames) (cn = <GROUPNAME>))
     - **Member Attribute:** UniqueMember
     - **Nested Group Level:** 0

3. Save the configuration by clicking **Save Changes**, then click the **Server Catalog** link.
a. Click Search....

b. Check the entries corresponding to the profiles you want to use (example: cn = IP Phone).

c. Click Add Selected.

4. Create a new MAC address authentication server, specifying your LDAP server (MyLDAPAuthServer in this example) under Optional LDAP Servers on the New MAC Address Authentication page.

   - **Name**: MACAuthServer
   - Under Optional LDAP Servers, add MyLDAPAuthServer.

5. Create a new MAC address realm. In the Servers section, select the following:

   - **Name**: MACAuthRealm
   - **Authentication**: MACAuthServer
   - **Directory/Attribute**: MyLDAPAuthServer.

6. Create a new location group with the following details:

   - **Name**: MACAuthLocationGroup.
   - For **MAC Authentication Realm**, select MACAuthRealm.

7. Create a RADIUS client for the switch as follows:

   - **Name**: MACAuthRADIUSClient.
   - For **Make/Model**, select HP Procurve Family or Cisco Systems, depending on the switch you are using.
   - For **Location Group**, select MACAuthLocationGroup.

8. Create a new role for the network devices.

   - **Name**: MyPhoneRole

**NOTE:** Do not configure any role restrictions, otherwise roles cannot get assigned to devices, and do not apply any Host Checker policies at the role or realm level.

9. In the MACAuthRealm configuration page, create a role mapping as follows:

   a. Click the New Rule button on the Role Mapping tab.
   b. Select Group membership after Rule Based on:
   c. Enter the Name IPPhoneRule (optional).
   d. Click Update.
e. Under **Rule: If user has any of these custom expressions...** select the group created in step 3.

f. Under **...then assign these roles**, add MyPhoneRole to Selected Roles:

g. Click **Save Changes**.

10. Create a RADIUS attributes policy.

   - **Name**: MyPhonePolicy.
   - **Location Group**: MACAuthLocationGroup.
   - **RADIUS Attributes**: 
     - **VLAN**: add the VLAN number that you have allocated for IP phones from your network.

11. Configure the switches to use MAC address LDAP authentication with the Infranet Controller as a RADIUS server. See the Juniper Solutions Guide *3rd Party Switch Configurations*.

### Configuring Network Access Policies for Unmanageable Devices

Unmanageable devices each have a unique MAC address. With MAC-based authentication the MAC address serves as both the username and the password.

MAC addresses are not generally guarded as secrets, so an attacker could obtain a MAC address and pose as the device, gaining network access. MAC-based authentication is typically used for devices like IP phones and printers. For security, access should be limited by creating a special VLAN for each device type.

#### Creating a MAC Address Realm

A realm consists of a grouping of authentication resources, including the authentication server, directory server, and accounting server. A MAC address realm is a special type of realm used only for MAC address authentication.

To configure a MAC address realm:

1. Create a MAC address authentication server. Populate the server with each device’s MAC address, and/or specify the LDAP server that stores MAC addresses.

2. In the admin console, choose **UAC > MAC Address Realms**.

3. Enter a name to label this realm and (optionally) a description.

4. Select **When editing, start on the Role Mapping page** if you want the Role Mapping tab to be selected when you open the realm for editing.

5. Under **Servers**, specify:
   - The MAC Address Authentication server to use for authenticating users who sign in to this realm.
- A directory/attribute server to use for retrieving device attributes.

6. Click **Save Changes** to save this MAC address realm.

You can limit the number of concurrent devices allowed on the Infranet Controller.

7. To limit the number of concurrent users on the realm, select the **Authentication Policy** tab, then **Limit the number of concurrent users** and then specify limit values for these options:

   a. **Guaranteed minimum**—You can specify any number of users between zero (0) and the maximum number of concurrent users defined for the realm, or you can set the number up to the maximum allowed by your license if there is no realm maximum.

   b. **Maximum** (optional)—You can specify any number of concurrent users from the minimum number you specified up to the maximum number of licensed users. If you enter a zero (0) into the **Maximum** field, no users are allowed to login to the realm.

8. Click **Save Changes**.

9. Create role mapping rules as necessary for this realm from the **Role Mapping** tab. Attributes of various device types can be used to assign roles, which can be referenced in RADIUS attributes policies. This configuration allows you to assign devices to the correct VLAN. See “Creating Role Mapping Rules” on page 249.

10. Click **Save Changes**.

**Configuring a Location Group for MAC Address Authentication**

To configure a location group policy for MAC address authentication on the Infranet Controller:

1. Create a sign-in policy to associate with the location group and choose the default sign-in page. See “Task summary: Configuring Sign-In Policies” on page 258.

2. Create a new location group from **UAC > Network Access > Location Group**. See “Configuring Location Groups” on page 145.

3. On the **New Location Group** page, enter a name and an optional description.

4. For **Sign-in Policy**, select the sign-in policy you want to associate with the location group.

5. Select a **MAC Authentication Realm** that you have created. See “Creating an Authentication Realm” on page 246.

6. Click **Save Changes**.

   After you create the MAC address authentication location group, you must create a RADIUS client.
**Configuring a RADIUS Client for MAC Address Authentication**

To configure a RADIUS client policy for unmanageable devices:

1. Create a new RADIUS client in accordance with the instructions in “Configuring RADIUS Clients” on page 147.
2. For **IP Address** and **IP Address Range**, enter the IP address of the switch.
3. For **Shared Secret**, enter the shared secret of the switch.
4. For **Make/Model**, choose **Cisco** or **HP Procurve Family** (these are the platforms supported for MAC address authentication).
5. Select the **Location Group** you created for MAC address authentication.
6. Click **Save Changes**.

**Configuring RADIUS Attributes for MAC Address Authentication**

To configure a RADIUS attributes policy for unmanageable devices:

1. Create a new RADIUS attributes policy for unmanageable devices in accordance with the instructions in “Configuring RADIUS Attributes Policies” on page 151.
2. Select the location group that you created for unmanageable devices.
3. Specify the VLAN that devices from this location group should be directed to. For example, IP phones should be directed to a VLAN that contains the VoIP infrastructure.
4. Specify the interface on which the network device(s) are connected to the Infranet Controller.
5. Select the role(s) that you have created for MAC address authentication. See “Creating a Role for MAC Address Authentication” on page 125.
6. Click **Save Changes** to save this RADIUS attributes policy.
Chapter 6

Authentication and Directory Servers

An authentication server is a database that stores user credentials—username and password—and typically group information. When a user signs in to the Infranet Controller, the user specifies an authentication realm, which is associated with an authentication server. If the user meets the realm’s authentication policy, the Infranet Controller forwards the user’s credentials to the associated authentication server. The authentication server’s job is to verify that the user exists and is who she claims to be. After verifying the user, the authentication server sends approval to the Infranet Controller and, if the realm also uses the server as a directory/attribute server, the user’s group information or other user attribute information. The Infranet Controller then evaluates the realm’s role mapping rules to determine to which user roles the user may be mapped.

The Infranet Controller supports the most common authentication servers, including Windows NT Domain, Active Directory, RADIUS, LDAP, NIS, RSA ACE/Server, and eTrust SiteMinder, and enables you to create one or more local databases of users who the Infranet Controller itself authenticates. For server overview and configuration information, see “Authentication and Directory Servers” on page 173.

A directory server is a database that stores user and group information. You can configure an authentication realm to use a directory server to retrieve user or group information for use in role mapping rules. Currently, the Infranet Controller supports LDAP servers for this purpose, which means you can use an LDAP server for both authentication and authorization. You define one server instance, and then the LDAP server’s instance name appears in both the Authentication and Directory/Attribute drop-down lists on a realm’s General tab. You can use the same server for any number of realms.

In addition to LDAP, you can use a RADIUS or SiteMinder server for retrieving user attributes that can be used in role mapping rules. Unlike an LDAP server instance, however, a RADIUS or SiteMinder server instance name does not appear in a realm’s Directory/Attribute drop-down list. To use a RADIUS or SiteMinder server to retrieve user information, you simply choose its instance name in the Authentication list and then choose Same as Above in the Directory/Attribute list. Then, you configure role mapping rules to use attributes from the RADIUS or SiteMinder server, which the Infranet Controller provides in an attribute list on the Role Mapping Rule page after you select Rule based on User attribute.

The Infranet Controller also supports MAC address authentication. To use this feature, you configure a MAC address authentication server instance. See “Configuring a MAC Address Authentication Server for Unmanageable Devices” on page 242.
This section contains the following information about authentication and directory servers:

- “Task Summary: Configuring Authentication Servers” on page 174
- “Defining an Authentication Server Instance” on page 175
- “Configuring an Anonymous Server Instance” on page 177
- “Configuring an ACE/Server Instance” on page 179
- “Configuring an Active Directory or NT Domain Instance” on page 182
- “Configuring a Certificate Server Instance” on page 190
- “Configuring an LDAP Server Instance” on page 191
- “Configuring a Local Authentication Server Instance” on page 200
- “Configuring an NIS Server Instance” on page 205
- “Configuring a RADIUS Server Instance” on page 206
- “Configuring an eTrust SiteMinder Server Instance” on page 219
- “Configuring a MAC Address Authentication Server for Unmanageable Devices” on page 242

Task Summary: Configuring Authentication Servers

To specify an authentication server that a realm may use, you must first configure a server instance on the Authentication > Auth. Servers page. When you save the server’s settings, the server name (the name assigned to the instance) appears on the realm’s General tab in the Authentication drop-down list. If the server is a(n):

- **LDAP or Active Directory server**—The instance name also appears in the Directory/Attribute drop-down list on the realm’s General tab. You may use the same LDAP or Active Directory server for both authentication and authorization for a realm, as well as use these servers for authorization for any number of realms that use different authentication servers.

- **RADIUS server**—The instance name also appears in the Accounting drop-down list on the realm’s General tab. You may use the same RADIUS server for both authentication and accounting for a realm, as well as use these servers for accounting for any number of realms that use different authentication servers.

To configure authentication servers:

1. Set up your authentication/authorization server using instructions from the provider.

2. Create an instance of the server starting at the Authentication > Authentication > Auth. Servers page in the admin console.
3. Create an authentication realm using settings in the Users > User Realms or Administrators > Admin Realms page of the admin console. For instructions, see “Creating an Authentication Realm” on page 246.


5. Password management only: set up password management options using instructions in “Enabling LDAP Password Management” on page 195.

**NOTE:** An authentication server must be able to contact the Infranet Controller. If an authentication server such as RSA ACE/Server does not use IP addresses for the agent hosts, the authentication server must be able to resolve the Infranet Controller host name, either through a DNS entry or an entry in the authentication server’s host file.

**NOTE:** When determining which server type to select:

- You can only create one eTrust Siteminder server instance per Infranet Controller.

- If you authenticate your Active Directory server with:
  
  - NTLM protocol—Choose Active Directory/Windows NT Domain. For more information, see “Configuring an ACE/Server Instance” on page 179.
  
  - LDAP protocol—Choose LDAP Server. For more information, see “Configuring an LDAP Server Instance” on page 191.

- If you are creating a local authentication server instance to authenticate user administrators, you must select Local Authentication. For more information, see “Configuring a Local Authentication Server Instance” on page 200.

**Defining an Authentication Server Instance**

Use the Auth. Servers page to define authentication server instances. Authentication servers authenticate user credentials and authorization servers provide user information that the Infranet Controller uses to determine user privileges within the system. For example, you can specify a certificate server instance to authenticate users based on their client-side certificate attributes and then create an LDAP server instance to authorize the users based on values contained within a CRL (certificate revocation list). For more information about authentication servers, see “Authentication and Directory Servers” on page 173.
This section contains the following information about authentication servers:

- “Defining an Authentication Server Instance” on page 176
- “Modifying an Existing Authentication Server Instance” on page 176

**Defining an Authentication Server Instance**

To define an authentication server instance:

1. In the admin console, choose **Authentication > Auth. Servers**.
2. Choose a server type from the **New** drop down menu.
3. Click **New Server**.
4. Depending on which server you selected, specify settings for the individual server instance.
5. Specify which realms should use the server to authenticate and authorize administrators and users. For more information, see “Defining Authentication Policies” on page 248.
6. If you are configuring the local authentication server, define local user accounts. For instructions, see “Configuring a Local Authentication Server Instance” on page 200.

**Modifying an Existing Authentication Server Instance**

To modify an authentication server instance:

1. In the admin console, choose **Authentication > Auth. Servers**.
2. Click the link to the server you want to modify.
3. Make your modifications on the appropriate server page.
4. Click **Save Changes**.

**Authentication Server Limitations with Authentication Protocols**

The Infranet Controller supports a variety of authentication methods. Not all of these protocols are compatible with all of the authentication server types. See “Authentication Protocols on the Infranet Controller” on page 133.

- **Local authentication servers**—For local auth servers with passwords stored hashed, the protocols available are PAP, MS-CHAP-V1 with or without EAP. Local auth servers with passwords stored as cleartext CHAP and MD5-Challenge are also available.

- **Active directory**—The protocols available for inner authentication are PAP, MS-CHAP, MS-CHAP-V2, with or without EAP.
Chapter 6: Authentication and Directory Servers

LDAP—CHAP, EAP-MD5-Challenge. MS-CHAP-V1, and MS-CHAP-V2 can be used with an LDAP auth server only if the Infranet Controller, using the configured administration password can retrieve the password, and the value is in cleartext. By default, challenge-response protocols are disabled for LDAP servers. These protocols should be used only with non-interactive devices (for example, phones), as password management is not possible if these protocols are used for authentication.

Anonymous authentication server—The anonymous authentication server is not supported with the open protocols.

Configuring an Anonymous Server Instance

The anonymous server feature allows users to access the Infranet Controller without providing a username or password. Instead, the following occurs:

- **On endpoints that use Odyssey Access Client**—When the user signs into his computer, Odyssey Access Client automatically starts, bypasses the Infranet Controller sign-in page, signs in the user anonymously, and immediately connects to the Infranet Controller.

- **On endpoints that use agentless access**—When a user enters the URL of an Infranet Controller that is configured to authenticate against an anonymous server, the Infranet Controller bypasses the Infranet Controller sign-in page, and immediately displays the Infranet Controller welcome page to the user.

You may choose to use anonymous authentication if you think that some resources do not require extreme security, or if you think that other security measures provided through the Infranet Enforcer are sufficient. For example, you may create a guest user role with limited access to certain resources. You create a guest user realm that uses an anonymous server to authenticate users, and map users that authenticate to the guest user realm to the guest user role. This enables guest users who do not have accounts on your authentication servers limited access to the network. Note that both the guest user realm and guest user role can still have host checker restrictions which restrict access from endpoints which do not have adequate endpoint security.

This section contains the following information about anonymous servers:

- “Anonymous Server Restrictions” on page 177
- “Defining an Anonymous Server Instance” on page 178

**Anonymous Server Restrictions**

When defining and monitoring an anonymous server instance, note that:

- You can only add one anonymous server configuration.
- You cannot authenticate administrators using an anonymous server.
During configuration, you must choose the anonymous server as both the authentication server and the directory/attribute server in the Users > User Realms > General tab. For more information, see “Creating an Authentication Realm” on page 246.

When creating role mapping rules through the Users > User Realms > Role Mapping tab (as explained in “Creating Role Mapping Rules” on page 249), the Infranet Controller does not allow you to create mapping rules that apply to specific users (such as “Joe”), since the anonymous server does not collect username information. You can only create role mapping rules based on a default username (*), certificate attributes, or custom expressions.

For security reasons, you may want to limit the number of users who sign in through an anonymous server at any given time. To do this, use the option on the Users > User Realms > [Realm] > Authentication Policy > Limits tab (where [Realm] is the realm that is configured to use the anonymous server to authenticate users). For more information, see “Specifying Session Limits” on page 114.

You cannot view and delete the sessions of anonymous users through a Users tab (as you can with other authentication servers), because the Infranet Controller cannot display individual session data without collecting usernames.

You cannot use an anonymous server if you are using the open protocols. See “Authentication Protocols on the Infranet Controller” on page 133.

**Defining an Anonymous Server Instance**

To define an anonymous server:

1.  In the admin console, choose Authentication > Auth. Servers.

2.  Do one of the following:

   - To create a new server instance on the Infranet Controller, select Anonymous Server from the New list, and then click New Server.

   - To update an existing server instance, click the appropriate link in the Authentication/Authorization Servers list.

3.  Specify a name to identify the server instance.

4.  Click Save Changes.

5.  Specify which realms should use the server to authorize users. For more information, see “Defining Authentication Policies” on page 248.
Configuring an ACE/Server Instance

When authenticating users with an RSA ACE/Server, users may sign in using two methods:

- **Using a hardware token and the standard Infranet Controller sign-in page**—The user browses to the standard Infranet Controller sign-in page, then enters her username and password (consisting of the concatenation of her PIN and her RSA SecurID hardware token’s current value). The Infranet Controller then forwards the user’s credentials to ACE/Server.

- **Using a software token and the custom SoftID Infranet Controller sign-in page**—The user browses to the SoftID custom sign-in page. Then, using the SoftID plug-in, she enters her username and PIN. The SoftID plug-in generates a pass phrase by concatenating the user’s PIN and token and passes the pass phrase to the Infranet Controller. For information about enabling the SoftID custom sign-in pages, see the Custom Sign-In Pages Solution Guide.

If ACE/Server positively authenticates the user, she gains access to the Infranet Controller. Otherwise, the ACE/Server:

- Denies the user access to the system if the user’s credentials were not recognized.

- Prompts the user to generate a new PIN (New PIN mode) if the user is signing in to the Infranet Controller for the first time. (The user sees different prompts depending on the method she uses to sign in. If the user signs in using the SoftID plug-in, she sees the RSA prompts for creating a new pin; otherwise the user sees the Infranet Controller prompts.)

- Prompts the user to enter her next token (Next Token mode) if the token entered by the user is out of sync with the token expected by ACE/Server. (Next Token mode is transparent to users signing in using a SoftID token. The RSA SecurID software passes the token through the Infranet Controller to ACE/Server without user interaction.)

- Redirects the user to the standard Infranet Controller sign-in page (SoftID only) if the user tries to sign-in to the RSA SecurID Authentication page on a computer that does not have the SecurID software installed.

When a user enters the New PIN or Next Token mode, she has three minutes to enter the required information before the Infranet Controller cancels the transaction and notifies the user to re-enter her credentials.

The Infranet Controller can handle a maximum of 200 ACE/Server transactions at any given time. A transaction only lasts as long as is required to authenticate against the ACE/Server. For example, when a user signs into the Infranet Controller, the ACE/Server transaction is initiated when the user submits her request for authentication and ends once the ACE/Server has finished processing the request. The user may then keep her Infranet Controller session open, even though her ACE/Server transaction is closed.
The Infranet Controller supports the following ACE/Server features: New PIN mode, Next Token mode, DES/SDI encryption, AES encryption, slave ACE/Server support, name locking, and clustering. The Infranet Controller also supports the New PIN and Next Token modes of RSA SecurID through the RADIUS protocol.

**NOTE:** Due to UNIX limitations of the ACE/Server library, you may define only one ACE/Server configuration. For information on generating an ACE/Agent configuration file for the Infranet Controller on the ACE server, see “Generating an ACE/Agent Configuration File” on page 181.

The Infranet Controller does not support load balancing between multiple ACE servers.

This section contains the following information about ACE/Servers:

- “Defining an ACE/Server Instance” on page 180
- “Generating an ACE/Agent Configuration File” on page 181

### Defining an ACE/Server Instance

**NOTE:** You can add only one ACE/Server instance.

To define an ACE/Server:

1. Generate an ACE/Agent configuration file (sdconf.rec) for the Infranet Controller on the ACE server. For more information, see “Generating an ACE/Agent Configuration File” on page 181.

2. In the admin console, choose **Authentication** > **Auth. Servers**.

3. Do one of the following:
   - To create a new server instance on the Infranet Controller, select **ACE Server** from the **New** list, and then click **New Server**.
   - To update an existing server instance, click the appropriate link in the **Authentication/Authorization Servers** list.

4. Specify a name to identify the server instance.

5. Specify a default port the **ACE Port** field. Note that the Infranet Controller only uses this setting if no port is specified in the sdconf.rec file.

6. Select the **Users authenticate using tokens or one-time passwords** option if you want the Infranet Controller to prompt the user for a token instead of a password. You should generally select this option if users submit tokens or one-time use passwords to the Infranet Controller.
For example, you can use this option to dynamically prompt for a password or token based on sign-in policies by configuring two instances of the same authentication server. You can use one instance for wireless users that has this option enabled and prompts the user for a token, and another instance for wired users that has this option disabled and prompts the user for a password.

7. Import the RSA ACE/Agent configuration file. Make sure to update this file on the Infranet Controller anytime you make changes to the source file. Likewise, if you delete the instance file from the Infranet Controller, go to the ACE Server Configuration Management application, as described in “Generating an ACE/Agent Configuration File” on page 181, and remove the check from the Sent Node Secret check box.

8. Click Save Changes. If you are creating the server instance for the first time, the Settings and Users tabs appear.

9. Specify which realms should use the server to authenticate and authorize administrators and users. For more information, see “Defining Authentication Policies” on page 248.

**NOTE:** For information about monitoring and deleting the sessions of users who are currently signed in through the server, see “Monitoring Active Users” on page 434.

**Generating an ACE/Agent Configuration File**

If you use ACE/Server for authentication, you must generate an ACE/Agent configuration file (**sdconf.rec**) for the Infranet Controller on the ACE Server.

To generate an ACE/Agent configuration file:

1. Start the ACE/Server Configuration Management application and click Agent Host.

2. Click Add Agent Host.

3. For Name, enter a name for the Infranet Controller agent.

4. For Network Address, enter the IP address of the Infranet Controller.

5. Enter a Site configured on your ACE server.

6. For Agent Type, select Communication Server.

7. For Encryption Type, select DES.

8. Verify that Sent Node Secret is not selected (when creating a new agent).

The first time that the ACE server successfully authenticates a request sent by the Infranet Controller, the ACE server selects Sent Node Secret. If you later want the ACE server to send a new Node Secret to the Infranet Controller on the next authentication request, do the following:

a. Click the Sent Node Secret check box to unchecked.
When authenticating users with an NT Primary Domain Controller (PDC) or Active Directory, users sign in to the Infranet Controller using the same username and password they use to access their Windows desktops. The Infranet Controller supports Windows NT authentication and Active Directory using NTLM or Kerberos authentication.

If you configure a native Active Directory server, you may retrieve group information from the server for use in a realm’s role mapping rules. In this case, you specify the Active Directory server as the realm’s authentication server, and then you create a role mapping rule based on group membership. The Infranet Controller displays all groups from the configured domain controller and its trusted domains.

The Infranet Controller provides separate check boxes for each of the primary authentication protocols: Kerberos, NTLMv2, and NTLMv1, allowing you to select or ignore each of these protocols independent of one another. This more granular control of the authentication process avoids unnecessarily raising the failed login count policy in Active Directory and lets you fine-tune the protocols based on your system requirements.
See “Creating Role Mapping Rules” on page 249 for more information.

NOTE:

- The Infranet Controller honors trust relationships in Active Directory and Windows NT environments.

- When sending user credentials to an Active Directory authentication server, the Infranet Controller uses whichever authentication protocol(s) you specify on the New Active Directory/Windows NT page. The Infranet Controller defaults to the authentication protocols in order. In other words, if you have selected the check boxes for Kerberos and NTLMv2, the Infranet Controller sends the credentials to Kerberos. If Kerberos succeeds, the Infranet Controller does not send the credentials to NTLMv2. If Kerberos is not supported or fails, the Infranet Controller uses NTLMv2 as the next protocol in order. The configuration sets up a cascading effect if you choose to use it by setting multiple check boxes.

- The Infranet Controller supports Domain Local Groups, Domain Global Groups, and Universal Groups defined in the Active Directory forest.

- The Infranet Controller allows only Active Directory security groups, not distribution groups. Security groups allow you to use one type of group for not only assigning rights and permissions, but also as a distribution list for email.

- If multiple Active Directory servers are configured on an Infranet Controller, each of the servers must be associated with a different and unique machine account name. The same machine account name should not be used for all servers.

This section contains the following information about Active Directory and NT Domain servers:

- “Defining an Active Directory or Windows NT Domain Server Instance” on page 183
- “Multi-Domain User Authentication” on page 187
- “Active Directory and NT Group Lookup Support” on page 189

**Defining an Active Directory or Windows NT Domain Server Instance**

To define an Active Directory or Windows NT Domain server:

1. In the admin console, choose Authentication > Auth. Servers.

2. Do one of the following:

   - To create a new server instance on the Infranet Controller, select Active Directory/ Windows NT from the New list and then click New Server.

   - To update an existing server instance, click the appropriate link in the Authentication/Authorization Servers list.
3. Specify a name to identify the server instance.

4. Specify the name or IP address for the primary domain controller or Active Directory server.

5. Specify the IP address of your back-up domain controller or Active Directory server. (optional)

6. Enter the domain name of the Active Directory or Windows NT domain. For example, if the Active Directory domain name is `us.amr.asgqa.net` and you want to authenticate users who belong to the `US` domain, enter `US` in the domain field.

7. If you want to specify a computer name, enter it into the **Computer Name** field. The computer name field is where you specify the name that the Infranet Controller uses to join the specified Active Directory domain as a computer. Otherwise, leave the default identifier which uniquely identifies your system.

**NOTE:** You may note that the computer name is pre-filled with an entry in the format of `vc0000H H H H H H`, where `H H H H H H` is a hex representation of the IP address of the Infranet Controller. A unique name, either the one provided by default or one of your own choosing, you can more easily identify your systems in the Active Directory. For example, the name could be something like `vc0000a1018dF2`.

In a clustered environment with the same AD authentication server, this name is also unique among all cluster nodes, and the Infranet Controller displays all of the identifiers for all attached cluster nodes.

8. Select the **Allow domain to be specified as part of username** check box to allow users to sign in by entering a domain name in the **Username** field in the format: `domain\username`

9. Select the **Allow trusted domains** check box to get group information from all trusted domains within a forest.

10. For **Admin Username** and **Admin Password**, enter an administrator username and password for the AD or NT server.

**NOTE:**

- Make sure the administrator you specify is a domain administrator in the same domain as the AD or NT server.
- Do not include a domain name with the server administrator username in the **Admin Username** field.
- After you save changes, the Infranet Controller masks the administrator password using five asterisk characters, regardless of the password length.
11. Under **Authentication Protocol**, specify which protocol the Infranet Controller should use during authentication.

12. Under **Kerberos Realm Name**:
   - Select **Use LDAP to get Kerberos realm name** if you want the Infranet Controller to retrieve the Kerberos realm name from the Active Directory server using the specified administrator credentials.
   - Enter the Kerberos realm name in the **Specify Kerberos realm name** field if you know the realm name.
     

13. Click **Test Configuration** to verify the Active Directory server configuration settings, such as do the specified domain exists, are the specified controllers Active Directory domain controllers, does the selected authentication protocol work, and so forth. (optional)

14. Click **Save Changes**. If you are creating the server instance for the first time, the **Settings** and **Users** tabs appear.

15. Specify which realms should use the server to authenticate and authorize administrators and users. For more information, see “Creating an Authentication Realm” on page 246.

---

**NOTE:**

- For information about monitoring and deleting the sessions of users who are currently signed in through the server, see “Monitoring Active Users” on page 434.

- The admin console provides last access statistics for each user account on various **Users** tabs throughout the console, under a set of columns titled **Last Sign-in Statistic**. The statistics reported include the last successful sign-in date and time for each user, the user’s IP address, and the agent or browser type and version.

---

**Using Kerberos Single Sign-On with the Infranet Controller**

The Kerberos single sign-on feature uses Kerberos authentication to automatically sign users into the Infranet Controller with the same credentials they entered to access their Windows desktops. After you configure Kerberos single sign-on to the Infranet Controller, the sign-in dialog box for the Infranet Controller does not appear to users.

Odyssey Access Client users can override 802.1X single sign-on and display the sign-in dialog box by selecting the **Prompt for login name and password** option in the Odyssey Access Client profile. This does not affect Infranet Controller single sign-on.
The Kerberos single sign-on feature requires a Windows NT Primary Domain Controller (PDC) or Active Directory for user authentication.

**NOTE:**

- The Kerberos single sign-on feature requires a Windows 2000 Server or later and is not supported on Windows NT Server 4.0 or earlier.
- The Kerberos single sign-on feature does not provide credentials for 802.1X authentication.
- When configuring Kerberos single sign-on, follow these guidelines:
  - The clocks on the Infranet Controller and the Windows Active Directory authentication server must be synchronized to within two minutes of each other.
  - The Windows endpoint computers must be joined to the same domain that the Infranet Controller uses for authentication. Alternatively, make sure the Windows endpoint computers are joined to a domain that has a trust relationship with the domain that the Infranet Controller uses for authentication.
  - Users must sign in to their endpoint computers in the domain of the Windows Active Directory authentication server, or in a trusted domain.

To configure Kerberos single sign-on to the Infranet Controller:

1. Select **Authentication > Auth. Servers** from the left navigation bar in the admin console.

2. Select **New Active Directory / Windows NT** and click **New**. The server you specify must be an Active Directory or Windows 2000 server or later for user authentication with the Infranet Controller. See “Configuring an Active Directory or NT Domain Instance” on page 182.

3. Select the check box to enable the **Kerberos** authentication protocol option on the authentication server configuration page.

4. On the **Administrators > Admin Realms** or **Users > User Realms** page, specify which realm(s) should use the Active Directory server to authenticate and authorize administrators and users. See “Creating an Authentication Realm” on page 246.
5. Make sure the Enable SSO option is enabled (the default setting) on the Administrators > Admin Realms > Select Realm > Authentication Policy > SSO page, or on the Users > User Realms > Select Realm > Authentication Policy > SSO page.

**NOTE:**

- The Enable SSO option is visible in the Infranet Controller admin console only if the Windows Active Directory authentication server is used for authenticating users of the selected realm.

- The Active Directory controller must be in front of the Infranet Controller to use Kerberos single sign-on.

### Multi-Domain User Authentication

The Infranet Controller allows for multi-domain Active Directory and Windows NT authentication. The Infranet Controller authenticates users in the domain you configure on the Authentication > Auth. Servers > New Active Directory / Windows NT page, users in child domains, and users in all domains trusted by the configured domain.

After you specify the address of a domain controller and a default domain in the Infranet Controller Active Directory server configuration, users in the default domain authenticate to the Infranet Controller using either just their username, or using the default domain plus username in the format `defaultdomain\username`.

When you enable trusted domain authentication, users in trusted or child domains authenticate to the Infranet Controller using the name of the trusted or child domain plus the username in the format `trusteddomain\username`. Note that enabling trusted domain authentication adds to the server's response time.

### Windows 2000 and Windows 2003 Multi-Domain Authentication

The Infranet Controller supports Kerberos-based Active Directory authentication with Windows 2000 and Windows 2003 domain controllers. When a user logs in to the Infranet Controller, the Infranet Controller performs Kerberos authentication and attempts to fetch the Kerberos realm name for the domain controller, as well as all child and trusted realms, using LDAP calls.

You can alternately specify the Kerberos realm name when configuring an Active Directory authentication server, but we do not recommend this method for two reasons:

- You cannot specify more than one realm name. The Infranet Controller cannot then authenticate against child or trusted realms of the realm you specify.

- If you misspell the realm name, the Infranet Controller cannot authenticate users against the proper realm.
Windows NT4 Multi-Domain Authentication

The Infranet Controller does not support Kerberos-based authentication in Windows NT4 domain controllers. Instead of Kerberos authentication, the Infranet Controller uses NTLM authentication.

NOTE:

- For user authentication, the Infranet Controller joins the default domain controller server using the machine name in the format of vc0000HHHHHHHH, where HHHHHHHHHH is a hex representation of the IP address of the Infranet Controller.

- If the DNS configuration on the Windows NT4 domain controller changes, make sure that the Infranet Controller can still resolve names (child and trusted domains) using either WINS, DNS, or the Hosts file, that were able to resolve the names prior to the configuration change.

NT User Normalization

In order to support multi-domain authentication, the Infranet Controller uses “normalized” NT credentials when contacting an Active Directory or NT4 domain controller for authentication. Normalized NT credentials include both the domain name and the username: domain\username. Regardless of how the user signs in to the Infranet Controller, either using just a username or using the domain\username format, the Infranet Controller always treats the username in the domain\username format.

When a user attempts to authenticate using only their username, the Infranet Controller always normalizes their NT credentials as defaultdomain\username. Authentication succeeds only if the user is a member of the default domain.

For a user who signs to the Infranet Controller using the domain\username format, the Infranet Controller always attempts to authenticate the user as members of the domain the user specifies. Authentication succeeds only if the user-specified domain is a trusted or child domain of the default domain. If the user specifies an invalid or untrusted domain, authentication fails.

Two variables, <NTUser> and <NTDomain>, allow you to individually refer to domain and NT username values. The Infranet Controller populates these two variables with the domain and NT username information.

NOTE: When using pre-existing role mapping rules or writing a new role mapping rule for Active Directory authentication where USER = someusername, the Infranet Controller treats this rule semantically as NTUser = someusername AND NTDomain = defaultdomain. This allows the Infranet Controller to work seamlessly with pre-existing role mapping rules.
Active Directory and NT Group Lookup Support

The Infranet Controller supports user group lookup in Domain Local, Domain Global, and Universal groups in the Active Directory forest, and Domain Local, and Domain Global groups for NT4 servers.

NOTE: For the NT/AD group lookup to work, the Infranet Controller first tries to join the domain using the default computer name. For this operation to succeed, you must specify valid domain administrator credentials in the Active Directory server configuration on the Infranet Controller.

Active Directory Lookup Requirements

The Infranet Controller supports user group lookup in Domain Local, Domain Global, and Universal groups in the default domain, child domains, and all trusted domains. The Infranet Controller obtains group membership using one of three methods that have different capabilities:

- **Group information in User’s Security Context**—Returns information about a user’s Domain Global groups.
- **Group information obtained using LDAP search calls**—Returns information about the user’s Domain Global groups, and information about the user’s Universal groups if the Infranet Controller queries the Global Catalog Server.
- **Group information using native RPC calls**—Returns information about the user’s Domain Local Group.

With respect to role mapping rules, The Infranet Controller attempts group lookup in the following order:

- The Infranet Controller checks for all Domain Global groups using the user’s security context.
- If the Infranet Controller has not found that the user is a member of some of the groups referenced in the role mapping rules, the Infranet Controller performs an LDAP query to determine the user’s group membership.
- If the Infranet Controller has not found that the user is a member of some of the groups referenced in the role mapping rules, the Infranet Controller performs an RPC lookup to determine the user’s Domain Local group membership.

NT4 Group Lookup Requirements

The Infranet Controller supports group lookup in the Domain Local and Domain Global groups created in the default domain, as well as all child, and other trusted domains. The Infranet Controller obtains Domain Global group information from the user’s security context, and Domain Local information using RPC calls. The Infranet Controller uses no LDAP-based search calls in the NT4 environment.
Configuring a Certificate Server Instance

The certificate server feature allows users to authenticate based on attributes contained in client-side certificates. You may use certificate server by itself or in conjunction with another server to authenticate users and map them to roles.

For example, you may choose to authenticate users solely based on their certificate attributes. If the Infranet Controller determines that the user’s certificate is valid, it signs the user in based on the certificate attributes you specify and does not prompt the user to enter a username or password.

Or, you may choose to authenticate users by passing their client-side certificate attributes to a second authentication server (such as LDAP). In this scenario, the certificate server first determines if the user’s certificate is valid. Then, the Infranet Controller can use realm-level role-mapping rules to compare the certificate attributes with the user’s LDAP attributes. If it cannot find the proper match, the Infranet Controller can deny or limit the user’s access based on your specifications.

NOTE: When using client-side certificates, we strongly recommend that you train your end-users to close their Web browsers after signing out of the Infranet Controller. If they do not, other users may be able to use their open browser sessions to access certificate-protected resources on the Infranet Controller without re-authenticating. (After loading a client-side certificate, both Internet Explorer and Netscape cache the certificate’s credentials and private key. The browsers keep this information cached until the user closes the browser (or in some cases, until the user reboots the workstation). For details, see: http://support.microsoft.com/?kbid=290345.) To remind users to close their browsers, you may modify the sign out message in the Authentication > Signing In Pages tab.

When defining a certificate server on the Infranet Controller, you must perform the following steps:

1. Use settings in the System > Configuration > Certificates > CA Certificates tab to import the CA certificate used to sign the client-side certificates.

2. Create a certificate server instance:


   b. Select Certificate Server from the New list, and then click New Server.

   c. Specify a name to identify the server instance.

   d. In the User Name Template field, specify how the Infranet Controller should construct a username. You may use any combination of certificate variables contained in angle brackets and plain text. For a list of certificate variables, see “System Variables and Examples” on page 521.
Configuring an LDAP Server Instance

The Infranet Controller supports two LDAP-specific authentication options:

- **Unencrypted**, in which the Infranet Controller sends the username and password to the LDAP Directory Service in clear, simple text.

- **LDAPS**, in which the Infranet Controller encrypts the data in the LDAP authentication session using Secure Socket Layer (SSL) protocol before sending it to the LDAP Directory Service.

The Infranet Controller performs substantial input validation for the following items:

- **LDAP Server**—The Infranet Controller provides a warning if the server is not reachable.

**NOTE:** If you choose a certificate attribute with more than one value, the Infranet Controller uses the first matched value. For example, if you enter `<certDN.OU>` and the user has two values for the attribute (ou=management, ou=sales), the Infranet Controller uses the “management” value. To use all values, add the SEP attribute to the variable. For example, if you enter `<certDN.OU SEP=":">` the Infranet Controller uses “management:sales”.

- Click **Save Changes**. If you are creating the server instance for the first time, the **Settings** and **Users** tabs appear.

**NOTE:** For information about monitoring and deleting the sessions of users who are currently signed in through the server, see “Monitoring Active Users” on page 434.

3. If you want to verify certificate attributes against an LDAP server, use settings in the **Authentication > Auth. Servers** page to create an LDAP server instance. Note that you must use the **Finding user entries** section in the LDAP configuration page to retrieve the user-specific attributes that you want verify through the certificate.

4. Use settings in the **Users > User Realms > RealmName > General** tab or **Administrators > Admin Realms > RealmName > General** tab to specify which realms should use the certificate server to authenticate users. (You may also use settings in these tabs to specify realms that should use an LDAP server to verify certificate attributes.)

5. If you want to restrict users’ access to realms or roles based on individual certificate attributes, use the settings described in “Specifying Certificate Access Restrictions” on page 112.
LDAP Port—The Infranet Controller provides a warning if the LDAP server is not reachable.

Administrator credentials—The Infranet Controller generates an error if the verification of admin credentials fails.

Base DN for users—The Infranet Controller generates an error if the base-level search on the Base DN value fails.

Base DN for groups—The Infranet Controller generates an error if the base-level search on the Base DN value fails.

This section contains the following information about LDAP servers:

- “Defining an LDAP Server Instance” on page 192
- “Monitoring and Deleting Active User Sessions” on page 195
- “Enabling LDAP Password Management” on page 195

Defining an LDAP Server Instance

To define an LDAP server instance:

1. In the admin console, choose Authentication > Auth. Servers.

2. Do one of the following:
   - To create a new server instance on the Infranet Controller, select LDAP Server from the New list and then click New Server.
   - To update an existing server instance, click the appropriate link in the Authentication/Authorization Servers list.

3. Specify a name to identify the server instance.

4. Specify the name or IP address of the LDAP server that the Infranet Controller uses to validate your users.

5. Specify the port on which the LDAP server listens. This port is typically 389 when using an unencrypted connection and 636 when using SSL.

6. Specify parameters for backup LDAP servers (optional). The Infranet Controller uses the specified servers for failover processing; each authentication request is first routed to the primary LDAP server and then to the specified backup server(s) if the primary server is unreachable.

**NOTE:** Backup LDAP servers must be the same version as the primary LDAP server. Also, we recommend that you specify the IP address of a backup LDAP server instead of its host name, which may accelerate failover processing by eliminating the need to resolve the host name to an IP address.

7. Specify the type of LDAP server that you want to authenticate users against.
8. Specify whether or not the connection between the Infranet Controller and LDAP Directory Service should be unencrypted, use SSL (LDAPs), or should use TLS.

9. Specify how long you want the Infranet Controller to wait for a connection to the primary LDAP server first, and then each backup LDAP server in turn.

10. Specify how long you want the Infranet Controller to wait for search results from a connected LDAP server.

11. Click Test Connection to verify the connection between the Infranet Controller appliance and the specified LDAP server(s). (optional)

12. Select the Authentication required? check box if the Infranet Controller needs to authenticate against the LDAP directory to perform a search or to change passwords using the password management feature. Then, enter an administrator DN and password. For more about password management, see “Enabling LDAP Password Management” on page 195. For example:

   CN=Administrator,CN=Users,DC=eng,DC=Juniper,DC=com

13. Under Finding user entries, specify a:

   - Base DN at which to begin searching for user entries. For example:
     
     DC=eng,DC=Juniper,DC=com

   - Filter if you want to fine-tune the search. For example:
     
     samAccountname=<username> or cn=<username>
     
     - Include <username> in the filter to use the username entered on the sign-in page for the search.
     
     - Specify a filter that returns 0 or 1 user DNs per user; the Infranet Controller uses the first DN returned if more than 1 DN is returned.

14. If endpoints send user names and domain names, and the LDAP server expects usernames without domain names, select the check box Strip domain from Windows user names under Remove Domain from Windows user names? to remove the domain from a domain\user name pair. This feature allows the Infranet Controller to pass the user name without the domain to the LDAP server.

   If you are configuring this LDAP server instance for non-interactive endpoints (for example, IP telephones) and you want to use a challenge-response protocol for authentication, select the check box for Enable Challenge-Response open protocols. By default, these protocols are disabled for LDAP servers, because account management is not possible.

   **NOTE:** If the LDAP server is configured to limit the rate of password-guessing attacks, and you select the Enable Challenge-Response open protocols check box, the LDAP server’s rate-limiting feature is bypassed.
15. The Infranet Controller supports both static and dynamic groups. (Note that the Infranet Controller only supports dynamic groups with LDAP servers.) To enable group lookup, you need to specify how the Infranet Controller searches the LDAP server for a group. Under **Determining group membership**, specify a:

- **Base DN** at which to begin searching for user groups.
- **Filter** if you want to fine-tune the search for a user group.
- **Member Attribute** to identify all the members of a static group. For example:
  
  member
  uniquemember (iPlanet-specific)

- **Reverse group search** to start the search from the member instead of the group. This option is available only for Active Directory server types.

- **Query Attribute** to specify an LDAP query that returns the members of a dynamic group. For example:
  
  memberURL

- **Nested Group Level** to specify how many levels within a group to search for the user. Note that the higher the number, the longer the query time, so we recommend that you specify to perform the search no more than 2 levels deep.

- **Nested Group Search** to search by:
  
  - **Nested groups in the LDAP Server Catalog**. This option is faster because it can search within the implicit boundaries of the nested group.
  
  - **Search all nested groups**. With this option, the Infranet Controller searches the Server Catalog first. If the Infranet Controller finds no match in the catalog, then it queries LDAP to determine if a group member is a sub-group.

**NOTE:** Because the Infranet Controller looks in the Server Catalog to determine if a member of a parent group is a user object or group object, you must add both the parent and all child (nested) groups to the Server Catalog.

16. Under **Bind Options**, select:

- **Simple bind** to send a user’s credentials in the clear (no encryption) to the LDAP Directory Service.

- **StartTLS bind** to encrypt a user’s credentials using the Transport Layer Security (TLS) protocol before the Infranet Controller sends the data to the LDAP Directory Service.
17. Click **Save Changes**. If you are creating the server instance for the first time, the **Settings** and **Users** tabs appear.

18. Specify which realms should use the server to authenticate and authorize administrators and users. For more information, see “Defining Authentication Policies” on page 248.

---

**NOTE:** The Infranet Controller supports referral chasing if enabled on your LDAP server.

---

### Monitoring and Deleting Active User Sessions

For information about monitoring and deleting the sessions of users who are currently signed in through the server, see “Monitoring Active Users” on page 434.

---

**NOTE:** The admin console provides last access statistics for each user account on various **Users** tabs throughout the console, under a set of columns titled **Last Sign-in Statistic**. The statistics reported include the last successful sign-in date and time for each user, the user’s IP address, and the agent or browser type and version.

---

### Enabling LDAP Password Management

The Infranet Controller password management feature enables users who authenticate through an LDAP server to manage their passwords through the Infranet Controller using the policies defined on the LDAP server. For example, if a user tries to sign in to the Infranet Controller with an LDAP password that is about to expire, the Infranet Controller catches the expired password notification, presents it to the user through the Infranet Controller interface, and then passes the user’s response back to the LDAP server without requiring the user to sign in to the LDAP server separately.

Users, administrators, and help desk administrators who work in environments where passwords have set expiration times may find the password management feature very helpful. When users are not properly informed that their passwords are about to expire, they can change them themselves through the Infranet Controller rather than calling the Help Desk.

Once enabled, the Infranet Controller performs a series of queries to determine user account information, such as when the user’s password was last set, if his account is expired, and so forth. The Infranet Controller does this by using its internal LDAP or Samba client. Many servers, such as Microsoft Active Directory or Sun iPlanet, offer an Administrative Console to configure account and password options.

LDAP-based Password Management works with only three types of LDAP servers:

- Microsoft Active Directory
- Sun Microsystems i-Planet
- Novell e-Directory.
LDAP-based Password Management does not work on generic LDAP servers like OpenLDAP.

The Infranet Controller enforces password policies by reading password attributes from the LDAP server. Therefore, for password management to work correctly, password policy attributes on backend server need to be configured properly.

- For Active Directory, password policy attributes can be configured in the user entry container level or any organization level above the user container. If these attributes are configured at multiple levels, the level closest to the user node takes precedence.

- The Infranet Controller does not support customized password policies.

- The password management feature is not supported on the Active Directory Global Catalog because password policy attributes are not fully populated on the Active Directory Global Catalog.

The Infranet Controller relies on the backend server to pinpoint the cause of error when a password change operation fails. However, while LDAP servers may report errors accurately to human operators, they do not always do so when communicating programmatically to systems like the Infranet Controller. Therefore, reported errors may at times be generic or cryptic.

This section includes the following topics with information about the LDAP password management feature:

- “Task Summary: Enabling LDAP Password Management” on page 196
- “Supported LDAP Directories and Servers” on page 196
- “Supported LDAP Password Management Functions” on page 198

**Task Summary: Enabling LDAP Password Management**

To enable password management through the Infranet Controller, you must:

1. Create an instance of the LDAP server through the Authentication > Auth. Servers page of the admin console.

2. Associate the LDAP server with a realm through the Administrators/Users > User Realms > [Realm] > General page of the admin console.

3. Enable password management for the realm in the Administrators/Users > User Realms > [Realm] > Authentication Policy > Password page of the admin console. Note that the Enable Password Management option only appears if the realm’s authentication server is an LDAP or NT/AD server.

**Supported LDAP Directories and Servers**

The Infranet Controller supports password management with the following LDAP directories:

- Microsoft Active Directory/Windows NT
- Sun iPlanet
Novell eDirectory

Generic LDAP directories, such as IBM Secure Directory and OpenLDAP

Additionally, the Infranet Controller supports password management with the following Windows servers:

- Microsoft Active Directory
- Microsoft Active Directory 2003
- Windows NT 4.0

The following sections list specific issues related to individual server types.

**Microsoft Active Directory**

- Changes on the Active Directory domain security policy may take 5 minutes or more to propagate among Active Directory domain controllers. Additionally, this information does not propagate to the domain controller on which it was originally configured for the same time period. This is a limitation of Active Directory.

- When changing passwords in Active Directory using LDAP, the Infranet Controller automatically switches to LDAPS, even if LDAPS is not the configured LDAP method. To support LDAPS on the Active Directory server, you must install a valid SSL certificate into the server’s personal certificate store. Note that the certificate must be signed by a trusted CA and the CN in the certificate’s Subject field must contain the exact host name of the Active Directory server, for example: adsrv1.company.com. To install the certificate, select the Certificates Snap-In in the Microsoft Management Console (MMC).

- The Account Expires option in the User Account Properties tab only changes when the account expires, not when the password expires. As explained in “Supported LDAP Password Management Functions” on page 198, Microsoft Active Directory calculates the password expiration using the Maximum Password Age and Password Last Set values retrieved from the User Policy and Domain Security Policy LDAP objects.

**Sun iPlanet**

When you select the User must change password after reset option on the iPlanet server, you must also reset the user’s password before this function takes effect. This is a limitation of iPlanet.

**General**

The Infranet Controller only displays a warning about password expiry if the password is scheduled to expire in 14 days or less. The Infranet Controller displays the message during each Infranet Controller sign in attempt. The warning message contains the remaining number of days, hours, and minutes that the user has to change his password before it expires on the server. The default value is 14 days; however, you may change it through the Administrators|Users > Admin Realms|User Realms > Authorization > Password configuration page of the admin console.
Supported LDAP Password Management Functions

The following matrix describes the password management functions supported by Juniper Networks, their corresponding function names in the individual LDAP directories, and any additional relevant details. These functions must be set through the LDAP server itself before the Infranet Controller can pass the corresponding messages, functions, and restrictions to end-users. When authenticating against a generic LDAP server, such as IBM Secure Directory, the Infranet Controller only supports authentication and allowing users to change their passwords.

Password Management functions are not supported when the CHAP family protocols are used for authentication. All functions are available when the JUAC protocol is used for authentication.

Table 21: Supported Password Management Functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Active Directory</th>
<th>iPlanet</th>
<th>Novell eDirectory</th>
<th>Generic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authenticate user</td>
<td>unicodePwds</td>
<td>userPassword</td>
<td>userPassword</td>
<td>userPassword</td>
</tr>
<tr>
<td>Allow user to change password if enabled</td>
<td>Server tells us in bind response (uses ntSecurityDescriptor)</td>
<td>If passwordChange = = ON</td>
<td>If passwordAllowChange = = TRUE</td>
<td>Yes</td>
</tr>
<tr>
<td>Log out user after password change</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Force password change at next login</td>
<td>If pwdLastSet = = 0</td>
<td>If passwordMustChange = = ON</td>
<td>If pwdMustChange = = TRUE</td>
<td></td>
</tr>
<tr>
<td>Password expired notification</td>
<td>userAccountControl = = 0x80000</td>
<td>If Bind Response includes control OID 2.16.840.1.113730.3.4.4 == 0</td>
<td>Check date/time value in passwordExpirationTime</td>
<td></td>
</tr>
<tr>
<td>Password expiration notification (in X days/hours)</td>
<td>if pwdLastSet - now() &lt; maxPwdAge - 14 days (maxPwdAge is read from domain attributes) (Infranet Controller displays warning if less than 14 days)</td>
<td>If Bind Response includes control OID 2.16.840.1.113730.3.4.5 (contains date/time) (Infranet Controller displays warning if less than 14 days)</td>
<td>If now() - passwordExpirationTime &lt; 14 days (Infranet Controller displays warning if less than 14 days)</td>
<td></td>
</tr>
<tr>
<td>Disallow authentication if *account disabled/locked</td>
<td>userAccountControl = = 0x2 (Disabled)</td>
<td>Bind ErrorCode: 53 &quot;Account Inactivated&quot;</td>
<td>Bind ErrorCode: 53 &quot;Account Expired&quot;</td>
<td></td>
</tr>
<tr>
<td>Enforce &quot;minimum password length&quot;</td>
<td>If set, Infranet Controller displays message telling user minPwdLength</td>
<td>userAccountControl = = 0x10 (Locked)</td>
<td>If set, Infranet Controller displays message telling user passwordMinLength</td>
<td></td>
</tr>
<tr>
<td>Disallow user from changing password too soon</td>
<td>If pwdLastSet - now() &lt; minPwdAge, then we disallow</td>
<td>If passwordMinAge &gt; 0, then if now() is earlier than passwordAllowChangeTime, then we disallow</td>
<td>Server tells us in bind response</td>
<td></td>
</tr>
</tbody>
</table>
Chapter 6: Authentication and Directory Servers

**Table 21: Supported Password Management Functions (Continued)**

<table>
<thead>
<tr>
<th>Function</th>
<th>Active Directory</th>
<th>iPlanet</th>
<th>Novell eDirectory</th>
<th>Generic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Honor &quot;password complexity&quot;</td>
<td>If pwdProperties = = 0x1, then enabled. Complexity means the new password does not contain username, first or last name, and must contain characters from 3 of the following 4 categories: English uppercase, English lowercase, Digits, and Non-alphabetic characters (ex. 1, $, %)</td>
<td>Server tells us in bind response</td>
<td>Server tells us in bind response</td>
<td></td>
</tr>
</tbody>
</table>

**AD/NT Password Management Matrix**

The following matrix describes the Password Management functions supported by Juniper Networks.

These functions are not supported for a layer 2 connection when CHAP, MS-CHAP, or PAP are used as authentication protocols.

**Table 22: AD/NT Password Management Matrix**

<table>
<thead>
<tr>
<th>Function</th>
<th>Active Directory</th>
<th>Active Directory 2003</th>
<th>Windows NT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authenticate user</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Allow user to change password if licensed and if enabled</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Log out user after password change</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Force password change at next login</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Password expired notification</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Account disabled</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Account expired</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**Troubleshooting LDAP Password Management on the Infranet Controller**

When troubleshooting, please provide any pertinent Infranet Controller logs, server logs, configuration information, and a TCP trace from the Infranet Controller. If you are using LDAPS, please switch to the "Unencrypted" LDAP option in the Infranet Controller LDAP server configuration while taking the LDAP TCP traces.
Configuring a Local Authentication Server Instance

The Infranet Controller enables you to create one or more local databases of users who are authenticated by the Infranet Controller. You might want to create local user records for users who are normally verified by an external authentication server that you plan to disable or if you want to create a group of temporary users. Note that all administrator accounts are stored as local records, but you can choose to authenticate administrators using an external server using instructions in “Defining Authentication Policies” on page 248.

This section contains the following information about local authentication servers:

- “Defining a Local Authentication Server Instance” on page 200
- “Creating User Accounts on a Local Authentication Server” on page 202
- “Managing User Accounts” on page 203

Defining a Local Authentication Server Instance

When defining a new local authentication server instance, you need to give the server a unique name and configure password options and password management. These password options enable you to control the password length, character composition, and uniqueness. If desired, you can enable users to change their passwords and to force users to change passwords after a specified number of days. You can also prompt the user to change the password within a certain number of days of its expiration date.

To define a local authentication server instance:

1. In the admin console, choose Authentication > Auth. Servers.

2. Do one of the following:

   - To create a new server instance on the Infranet Controller, select Local Authentication from the New list, and then click New Server.

   - To update an existing server instance, click the appropriate link in the Authentication/Authorization Servers list.

3. Specify a name to identify the new server instance or edit the current name for an existing server.

4. Specify password options:

   a. Under Password options, set the minimum character length for passwords.
b. Set the maximum character length for passwords (optional). The maximum length cannot be less than the minimum length. There is no maximum limit to the length.

NOTE:
- If the maximum length set on the authentication server is shorter than the maximum length specified on the Infranet Controller, you may receive an error if you enter a password that is longer than that specified on the authentication server. The admin console allows you to enter passwords of any length, but your authentication server maximum determines the validity of the password length.
- If you want all passwords to be the same character length, set both the minimum and maximum lengths to the same value.

c. Enable the **Password must have at least digits** check box and specify the number of digits required in a password (optional). Do not require more digits than the value of the **Maximum length** option.

d. Enable the **Password must have at least letters** check box and specify the number of letters required in a password (optional). Do not require more letters than the value of the **Maximum length** option. If you enable the previous option, the combined total of the two options cannot exceed that of the value specified in the **Maximum length** option.

e. Enable the **Password must have mix of UPPERCASE and lowercase letters** check box if you want all passwords to contain a mixture of upper- and lowercase letters (optional).

NOTE: Require passwords to contain at least two letters if you also require a mix of upper- and lowercase letters.

f. Enable the **Password must be different from username** check box if the password cannot equal the username (optional).

g. Enable the **New passwords must be different from previous password** check box if a new password cannot equal the previous password (optional).

h. If you have configured open protocol sets for authentication, select the **Password stored as clear text** check box. CHAP and EAP-MD5-Challenge work with local auth servers only if this option is selected. See “Authentication Protocols on the Infranet Controller” on page 133.

NOTE: Be aware of the security implications of storing passwords as clear text.
5. Specify password management options:
   a. Under **Password management**, enable the **Allow users to change their passwords** check box if you want users to be able to change their passwords (optional).
   b. Enable the **Force password change after _ days** check box and specify the number of days after which a password expires (optional).

   **NOTE:** The default is 64 days, but you can set this value to any number you desire.

   c. Enable the **Prompt users to change their password _ days before current password expires** check box and provide the number of days before password expiration to prompt the user (optional).

   **NOTE:** The default value is 14 days, but you can set the value to any number up to the number placed in the previous option.

6. Click **Save Changes**. If you are creating the server instance for the first time, the **Users** tab appears.

   **NOTE:** After you set password options and password management options, you also need to specify which realms should use the server to authenticate and authorize administrators and users. Use the **Enable Password Management option on the Administrators/Users > Admin Realms/User Realms > Realm > Authentication Policy > Password** page to specify whether or not the realm inherits password management settings from the local authentication server instance. See “Specifying Password Access Restrictions” on page 113 for information about enabling password management.

---

**Creating User Accounts on a Local Authentication Server**

When you create a local authentication server instance, you need to define local user records for that database. A local user record consists of a username, the user’s full name, and the user’s password. You may want to create local user records for users who are normally verified by an external authentication server that you plan to disable or if you want to quickly create a group of temporary users.

To create local user records for a local authentication server:

1. In the admin console, choose **Authentication > Auth. Servers**.
2. Click the Infranet Controller database to which you want to add a user account.
3. Select the **Users** tab and click **New**.
4. Enter a username and user’s full name. Note:
   - Do not include “~” in a username.
   - If you want to change a user’s username after creating the account, you must create an entirely new account.

5. Enter and confirm the password. Make sure that the password you enter conforms to the password options specified for the associated local authentication server instance.

6. Select One-time use (disable account after the next successful sign-in) if you want to limit the user to one login. After one successful login, the user’s login state is set to Disabled and the user receives an error message when attempting subsequent sign-ins. However, you can manually reset this option in the admin console to allow the same user to login again. If you leave this option unchecked, it means that you are creating a permanent user.

7. Select Enabled if not already selected. This option is used by the administrator to selectively enable or disable any user (one time or permanent). Selected by default. If the One-time use option is checked, this option changes to Disabled after the user logs in successfully. If a permanent or one-time user is logged in and you disable this option, the user is immediately logged out of the system and receives an error message.

8. Select Require user to change password at next sign in if you want to force the user to change his or her password at the next login.

9. Click Save Changes. The user record is added to the Infranet Controller database.

**NOTE:** If you force the user to change passwords, you must also enable the Allow users to change their passwords option. Use options on the Administrators/Users > Admin Realms/User Realms > [Realm] > Authentication Policy > Password page to specify which realms should inherit the server’s password management capabilities.

**NOTE:** The admin console provides last access statistics for each user account on various Users tabs throughout the console, under a set of columns titled Last Sign-in Statistic. The statistics reported include the last successful sign-in date and time for each user, the user’s IP address, and the agent or browser type and version.

**Managing User Accounts**

To manage a local user account:

1. In the admin console, choose Authentication > Auth. Servers.

2. Click the appropriate server link in the Authentication/Authorization Servers list.
3. Select the Users tab.

4. Perform any of the following tasks:
   - Enter a username in the Show users named field and click Update to search for a specific user.
     Alternatively, you can use an asterisk (*) as a wildcard, where * represents any number of zero or more characters. For example, if you want to search for all usernames that contain the letters jo, enter *jo* in the Show users named field. The search is case-sensitive. To display the entire list of accounts again, either enter * or delete the field’s contents and click Update.
   - Enter a number in the Show N users field and click Update to control the number of users displayed on the page.
   - Click the check box next to individual users and click Delete to terminate their Infranet Controller sessions.

**Delegating User Administration Rights to End-Users**

User administrators can be created to give individuals some administrative capabilities on the Infranet Controller. User administrators cannot manage realms or role mappings. Therefore, we recommend enabling the User Admin feature only if the authentication realm’s role mapping rules permit “unmatched” users (*) to sign in to the Infranet Controller so the user administrator can successfully add new users without administrator interference. (When the role mappings are automatic, the user administrator does not need the administrator to manually map the new users to a role.)

A user admin user can add new users, change passwords and delete existing users. You can delegate these abilities to user admins per local authentication server. For example, you can give administrative personnel the capability to add guest users for one-time or temporary use.

To delegate user administration rights to an end-user:

1. In the admin console, choose Authentication > Auth. Servers.

2. Select the local authentication server instance that you want the user administrator to manage, and then click the Admin Users tab.

   **NOTE:** User administrators can only administer local authentication servers.

3. Enter the Username of the user who you want to manage accounts for the selected authentication server. This user does not need to be added as a local user on the server

   **NOTE:** Be careful when entering the user administrator’s username—it must match exactly.
4. Select the **Authentication Realm** that the user administrator maps to when they sign in to the Infranet Controller.

5. Click **Add**. The Infranet Controller adds the new user administrator to the **User Admins** list using the format: `username@servername`.

6. If the specified user administrator maps to multiple realms, optionally repeat steps 3-5 for each realm so that the admin user may manage the server regardless of which account they use to sign in to the Infranet Controller.

7. To revoke a user’s administration rights, select the name from the **User Admins** list and click **Remove**.

---

### Configuring an NIS Server Instance

When authenticating users with a UNIX/NIS server, the Infranet Controller verifies that the username and password entered through the sign-in page correspond to a valid user ID and password pair in the NIS server. Note that the username submitted to the Infranet Controller cannot contain two consecutive tilde symbols (`~~`).

**NOTE:** You can only use NIS authentication with the Infranet Controller if your passwords are stored on the NIS server using Crypt or MD5 formats. Also note that you can only add one NIS server configuration to the Infranet Controller, but you can use that configuration to authenticate any number of realms.

To define an NIS server instance:

1. In the admin console, choose **Authentication > Auth. Servers**.

2. Do one of the following:
   - To create a new server instance on the Infranet Controller, select **NIS Server** from the **New** list, and then click **New Server**.
   - To update an existing server instance, click the appropriate link in the **Authentication/Authorization Servers** list.

3. Specify a name to identify the server instance.

4. Specify the name or IP address of the NIS server.

5. Specify the domain name for the NIS server.

6. Click **Save Changes**. If you are creating the server instance for the first time, the **Settings** and **Users** tabs appear.
7. Specify which realms should use the server to authenticate and authorize administrators and users. For more information, see “Defining Authentication Policies” on page 248.

**NOTE:** For information about monitoring and deleting the sessions of users who are currently signed in through the server, see “Monitoring Active Users” on page 434.

---

## Configuring a RADIUS Server Instance

A Remote Authentication Dial-In User Service (RADIUS) server is a type of server that allows you to centralize authentication and accounting for users. When using an external RADIUS server to authenticate Infranet Controller users, you need to configure it to recognize the Infranet Controller as a client and specify a shared secret for the RADIUS server to use to authenticate the client request.

The Infranet Controller also supports RADIUS proxy. You can configure your external RADIUS server as an inner or outer proxy target. When you specify RADIUS proxy, some fields in the RADIUS server configuration page are not applicable. See “Using RADIUS Proxy” on page 138.

The Infranet Controller supports the standard RADIUS authentication schemes, including:

- **Access-Request**
- **Access-Accept**
- **Access-Reject**
- **Access-Challenge**

The Infranet Controller also supports the RSA ACE/Server using the RADIUS protocol and a SecurID token (available from Security Dynamics). If you use SecurID to authenticate users, users must supply their user ID and the concatenation of a PIN and the token value.

When defining a RADIUS server, the Infranet Controller gives administrators the ability to use either hard-coded (default) challenge expressions that support Defender 4.0 and some RADIUS server implementations (such as Steel-Belted RADIUS and RSA RADIUS) or to enter custom challenge expressions that allow the Infranet Controller to work with many different RADIUS implementations and new versions of the RADIUS server, such as Defender 5.0. The Infranet Controller looks for the response in the Access-Challenge packet from the server and issues an appropriate Next Token, New Pin, or Generic Passcode challenge to the user.

This topic contains the following information about RADIUS servers:

- “User Experience for RADIUS Users” on page 207
- “Configuring the Infranet Controller to Work with a Back-end RADIUS Server” on page 208
User Experience for RADIUS Users

The user experience varies depending on whether you are using a RADIUS server like Steel-Belted RADIUS, PassGo Defender RADIUS server or CASQUE authentication.

The Infranet Controller itself can also be used as a RADIUS server.

Using a PassGo Defender RADIUS Server
If you are using a PassGo Defender RADIUS Server, the user sign-in process is:

1. The user signs in to the Infranet Controller with a username and password. The Infranet Controller forwards these credentials to Defender.
2. Defender sends a unique challenge string to the Infranet Controller and the Infranet Controller displays this challenge string to the user.
3. The user enters the challenge string in a Defender token and the token generates a response string.
4. The user enters the response string on the Infranet Controller and clicks Sign In.

Using CASQUE Authentication
CASQUE authentication uses a token-based challenge/response authentication mechanism employing a CASQUE player installed on the client system. Once configured with CASQUE authentication, the RADIUS server issues a challenge with a response matching the custom challenge expression (\([0-9a-zA-Z/+=]+\:(-?\[[0-9a-zA-Z/+=]+\])\):). The Infranet Controller then generates an intermediate page that automatically launches the CASQUE player installed on the user’s system.

NOTE: If the CASQUE player does not launch automatically, click the Launch CASQUE Player link.

Users must then use their CASQUE Optical Responder tokens to generate the corresponding passcode, enter the passcode in the Response field, and click Sign In.
Defining an Infranet Controller RADIUS Server Instance

To configure a connection to the RADIUS server on the Infranet Controller:

1. In the admin console, choose Authentication > Auth. Servers.
2. Do one of the following:
   - To create a new server instance on the Infranet Controller, select Radius Server from the New list, and then click New Server.
   - To update an existing server instance, click the appropriate link in the Authentication/Authorization Servers list.
3. At the top of the Radius Server page, specify a name to identify the server instance.
4. In the NAS-Identifier field, enter the name that identifies the Infranet Controller Network Access Server (NAS) client that communicates with the RADIUS server. If you leave this field empty, the Infranet Controller uses the value specified in the Hostname field of the System > Network > Overview page of the admin console. If no value is specified in Hostname field, the Infranet Controller uses the value “Juniper Infranet Controller.”
   
   If you are using the RADIUS proxy feature, this field is not used. Proxy passes on the entire RADIUS packet including the NAS-ID from the client.
5. Specify the name or IP address in the RADIUS server text box.
6. Enter the authentication port value for the RADIUS server. Typically this port is 1812, but some legacy servers might use 1645.

7. Enter a string for the shared secret. You also need to enter this string when configuring the RADIUS server to recognize the Infranet Controller machine as a client.

8. Enter the accounting port value for the RADIUS server. Typically this port is 1813, but some legacy servers might use 1646.

9. Enter the NAS IP Address. This allows you to control the NAS IP address value passed to RADIUS requests. If you leave this field empty, then the Infranet Controller’s internal IP address will be passed to RADIUS requests. If you configure the NAS IP address, then the value will be passed, regardless of which cluster node sends the requests.

   If you are using the RADIUS proxy feature, this field is not used. Proxy passes on the entire RADIUS packet including the NAS IP address from the client.

10. Enter the interval of time for the Infranet Controller to wait for a response from the RADIUS server before timing out the connection.

11. Enter the number of times for the Infranet Controller to try to make a connection after the first attempt fails.

12. Select the Users authenticate using tokens or one-time passwords option if you want the Infranet Controller to prompt the user for a token instead of a password. You should generally select this option if users submit tokens or one-time use passwords to the Infranet Controller.

   For example, you can use this option to dynamically prompt for a password or token based on sign-in policies by configuring two instances of the same authentication server. You can use one instance for wireless users that has this option enabled and prompts the user for a token, and another instance for wired users that has this option disabled and prompts the user for a password.

   If you are using the RADIUS proxy feature, this field is not used.

13. In the Backup Server section, enter a secondary RADIUS server for the Infranet Controller to use if the primary server—the one defined in this instance—is unreachable. For the secondary server, enter the server:

   a. Name or IP address

   b. Authentication port

   c. Shared secret

   d. Accounting port

   If you are using the RADIUS proxy feature, the fields in this section are not used.
14. If you want to track Infranet Controller user activity using this instance of the RADIUS server, enter the following information in the Radius Accounting section:

a. In the User-Name field, specify the user information that the Infranet Controller should send to the RADIUS accounting server. You may enter any of the applicable session variables described in “System Variables and Examples” on page 521. Applicable variables include those that are set at the time after the user signs in and maps to a role. The default variables for this field are:

- `<username>` logs the user’s Infranet Controller username to the accounting server.
- `<REALM>` logs the user’s Infranet Controller realm to the accounting server.
- `<ROLE>` logs the user’s Infranet Controller role to the accounting server. If the user is assigned to more than one role, the Infranet Controller comma-separates them.

b. Add an Interim Update Level (in minutes). The interim update level enables you to accomplish more precise billing for long-lived session clients and in case of a network failure. For more information, see “Understanding the Interim Update Feature” on page 219.

If you are using the RADIUS proxy feature, the fields in this section are not used.

15. Select the Use NC assigned IP Address for FRAMED-IP-ADDRESS attribute value in Radius Accounting checkbox to use the IP address returned from the Infranet Controller for the Framed-IP-Address attribute.

Two IP addresses are recorded: one prior to authenticating with the Infranet Controller, and one returned by Network Connect after authentication. Select this option to use the Network Connect IP address for the Framed-IP-Address attribute instead of the pre-authenticated (original) IP address.

16. Add a custom challenge expression (optional). Three types of challenge expressions exist with each automatically set to its pre-populated default. The custom option allows the administrator to configure the actual string pattern to match for any of the three modes. To add a custom expression, select the checkbox next to the appropriate challenge expression type, and add a custom expression in the associated text box.

If you are using the RADIUS proxy feature, the fields in this section are not used.

17. Click Save Changes. If you are creating the server instance for the first time, the Settings and Users tabs appear.
18. Specify which realms should use the server to authenticate, authorize, or account for administrators and users. For more information, see “Defining Authentication Policies” on page 248.

**NOTE:** For information about monitoring and deleting the sessions of users from this server who are currently signed, see “Monitoring Active Users” on page 434.

### Configuring the RADIUS Server to Recognize the Infranet Controller

You need to configure the RADIUS server to recognize the Infranet Controller by specifying:

- The host name given to the Infranet Controller.
- The network IP address of the Infranet Controller.
- The Infranet Controller client type—if applicable. If this option is available, select Single Transaction Server or its equivalent.
- The type of encryption to use for authenticating client communication. This choice should correspond to the client type.
- The shared secret you entered in the admin console for the RADIUS server on the Authentication > Auth. Servers > Radius Server page.

### Enabling RADIUS Accounting

You can configure the Infranet Controller to send session start and stop messages to a RADIUS accounting server. The Infranet Controller sends a user-session start message after the user successfully signs in and the Infranet Controller maps him to a role.

Whenever a user session terminates, the Infranet Controller sends a user-session stop message to the accounting server. A user session terminates whenever the user:

- Manually signs out of the Infranet Controller
- Times out of the Infranet Controller either due to inactivity or because of exceeding the maximum session length
- Is denied access due to Host Checker role-level restrictions
- Is manually forced out of the Infranet Controller by an administrator or due to dynamic policy evaluation.

**NOTE:** If users are signed into an Infranet Controller cluster, the RADIUS accounting messages may show the users signing in to one node and signing out of another.
The following three tables describe the attributes that are common to start and stop messages, attributes that are unique to start messages, and attributes that are unique to stop messages.

Table 23: Attributes Common to both Start and Stop Messages

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>User-Name (1)</td>
<td>String that the Infranet Controller administrator specifies during RADIUS server configuration</td>
</tr>
<tr>
<td>NAS-IP-Address (4)</td>
<td>Infranet Controller’s IP address</td>
</tr>
<tr>
<td>NAS-Port (5)</td>
<td>The Infranet Controller sets this attribute to 0 if the user signed in using an internal port, or 1 if an external port.</td>
</tr>
<tr>
<td>Framed-IP-Address (8)</td>
<td>User’s source IP address</td>
</tr>
<tr>
<td>NAS-Identifier (32)</td>
<td>Configured name for the Infranet Controller client under the RADIUS server configuration</td>
</tr>
<tr>
<td>Acct-Status-Type (40)</td>
<td>The Infranet Controller sets this attribute to 1 for a start message, or 2 for a stop message in a user-session or a sub-session</td>
</tr>
<tr>
<td>Acct-Session-Id (44)</td>
<td>Unique accounting ID that matches start and stop messages corresponding to a user-session or to a sub-session.</td>
</tr>
<tr>
<td>Acct-Multi-Session-Id (50)</td>
<td>Unique accounting ID that you can use to link together multiple related sessions. Each linked session must have a unique Acct-Session-Id and the same Acct-Multi-Session-Id.</td>
</tr>
<tr>
<td>Acct-Link-Count (51)</td>
<td>The count of links in a multi-link session at the time the Infranet Controller generates the accounting record</td>
</tr>
</tbody>
</table>

Table 24: Start Attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acct-Authentic (45)</td>
<td>The Infranet Controller sets this attribute to:</td>
</tr>
<tr>
<td></td>
<td>- RADIUS—if the user authenticated to a RADIUS server</td>
</tr>
<tr>
<td></td>
<td>- Local—if the user authenticated to an Local Authentication Server</td>
</tr>
<tr>
<td></td>
<td>- Remote—for anything else</td>
</tr>
</tbody>
</table>

Table 25: Stop Attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acct-Session-Time (46)</td>
<td>Duration of the user-session</td>
</tr>
<tr>
<td>Acct-Terminate-Cause (49)</td>
<td>The Infranet Controller uses one of the following values to specify the event that caused the termination of a user session:</td>
</tr>
<tr>
<td></td>
<td>- User Request (1) – User manually signs out</td>
</tr>
<tr>
<td></td>
<td>- Idle Timeout (4) – User Idle time out</td>
</tr>
<tr>
<td></td>
<td>- Session Timeout (5) – User Max Session Timeout</td>
</tr>
<tr>
<td></td>
<td>- Admin Reset (6) – User Forced Out from Active Users page</td>
</tr>
</tbody>
</table>
**Supported RADIUS Attributes**

The following RADIUS attributes are supported in RADIUS role mapping. For more information, see the full descriptions (from which these descriptions were derived) at the FreeRADIUS website located at [http://www.freeradius.org/rfc/attributes.html](http://www.freeradius.org/rfc/attributes.html).

Table 26: RADIUS Role Mapping Attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARAP-Features</td>
<td>Sent in an Access-Accept packet with Framed-Protocol of ARAP. Includes password information that the NAS must send to the user in an ARAP feature flags packet.</td>
</tr>
<tr>
<td>ARAP-Password</td>
<td>Present in an Access-Request packet containing a Framed-Protocol of ARAP. Only one of User-Password, CHAP-Password, or ARAP-Password must be included in an Access-Request, or one or more EAP-Messages.</td>
</tr>
<tr>
<td>ARAP-Security</td>
<td>Identifies the ARAP Security Module to be used in an Access-Challenge packet.</td>
</tr>
<tr>
<td>ARAP-Security-Data</td>
<td>Contains a security module challenge or response, and is in Access-Challenge and Access-Request packets.</td>
</tr>
<tr>
<td>ARAP-Zone-Access</td>
<td>Indicates how to use the ARAP zone list for the user.</td>
</tr>
<tr>
<td>Access-Accept</td>
<td>Provides specific configuration information necessary to begin delivery of service to the user.</td>
</tr>
<tr>
<td>Access-Challenge</td>
<td>To send the user a challenge requiring a response, the RADIUS server must respond to the Access-Request by transmitting a packet with the Code field set to 11 (Access-Challenge).</td>
</tr>
<tr>
<td>Access-Reject</td>
<td>If any value of the received Attributes is not acceptable, then the RADIUS server must transmit a packet with the Code field set to 3 (Access-Reject).</td>
</tr>
<tr>
<td>Access-Request</td>
<td>Conveys information specifying user access to a specific NAS, and any special services requested for that user.</td>
</tr>
<tr>
<td>Accounting-Request</td>
<td>Conveys information used to provide accounting for a service provided to a user.</td>
</tr>
<tr>
<td>Accounting-Response</td>
<td>Acknowledges that the Accounting-Request has been received and recorded successfully.</td>
</tr>
<tr>
<td>Acct-Authentic</td>
<td>Indicates how the user was authenticated, whether by RADIUS, the NAS itself, or another remote authentication protocol.</td>
</tr>
<tr>
<td>Acct-Delay-Time</td>
<td>Indicates how many seconds the client has been trying to send this record.</td>
</tr>
<tr>
<td>Acct-Input-Gigawords</td>
<td>Indicates how many times the Acct-Input-Octets counter has wrapped around $2^{32}$ over the course of this service being provided.</td>
</tr>
<tr>
<td>Acct-Input-Octets</td>
<td>Indicates how many octets have been received from the port during the current session.</td>
</tr>
<tr>
<td>Acct-Input-Packets</td>
<td>Indicates how many packets have been received from the port during the session provided to a Framed User</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Acct-Interim-Interval</td>
<td>Indicates the number of seconds between each interim update in seconds for this specific session.</td>
</tr>
<tr>
<td>Acct-Link-Count</td>
<td>The count of links known to have been in a given multilink session at the time the accounting record is generated.</td>
</tr>
<tr>
<td>Acct-Multi-Session-Id</td>
<td>A unique Accounting ID to make it easy to link together multiple related sessions in a log file.</td>
</tr>
<tr>
<td>Acct-Output-Gigawords</td>
<td>Indicates how many times the Acct-Output-Octets counter has wrapped around $2^{32}$ during the current session.</td>
</tr>
<tr>
<td>Acct-Output-Octets</td>
<td>Indicates how many octets have been sent to the port during this session.</td>
</tr>
<tr>
<td>Acct-Output-Packets</td>
<td>Indicates how many packets have been sent to the port during this session to a Framed User.</td>
</tr>
<tr>
<td>Acct-Session-Id</td>
<td>A unique Accounting ID to make it easy to match start and stop records in a log file.</td>
</tr>
<tr>
<td>Acct-Session-Time</td>
<td>Indicates how many seconds the user has received service.</td>
</tr>
<tr>
<td>Acct-Status-Type</td>
<td>Indicates whether this Accounting-Request marks the beginning of the user service (Start) or the end (Stop).</td>
</tr>
<tr>
<td>Acct-Terminate-Cause</td>
<td>Indicates how the session was terminated.</td>
</tr>
<tr>
<td>Acct-Tunnel-Connection</td>
<td>Indicates the identifier assigned to the tunnel session.</td>
</tr>
<tr>
<td>Acct-Tunnel-Packets-Lost</td>
<td>Indicates the number of packets lost on a given link.</td>
</tr>
<tr>
<td>CHAP-Challenge</td>
<td>Contains the CHAP Challenge sent by the NAS to a PPP Challenge-Handshake Authentication Protocol (CHAP) user.</td>
</tr>
<tr>
<td>CHAP-Password</td>
<td>The response value provided by a PPP Challenge-Handshake Authentication Protocol (CHAP) user in response to the challenge.</td>
</tr>
<tr>
<td>Callback-Id</td>
<td>The name of a location to be called, to be interpreted by the NAS.</td>
</tr>
<tr>
<td>Callback-Number</td>
<td>The dialing string to be used for callback.</td>
</tr>
<tr>
<td>Called-Station-Id</td>
<td>Allows the NAS to send the phone number that the user called, using Dialed Number Identification (DNIS) or similar technology.</td>
</tr>
<tr>
<td>Calling-Station-Id</td>
<td>Allows the NAS to send the phone number that the call came from, using Automatic Number Identification (ANI) or similar technology.</td>
</tr>
<tr>
<td>Class</td>
<td>Sent by the server to the client in an Access-Accept and then sent unmodified by the client to the accounting server as part of the Accounting-Request packet, if accounting is supported.</td>
</tr>
<tr>
<td>Configuration-Token</td>
<td>For use in large distributed authentication networks based on proxy.</td>
</tr>
<tr>
<td>Connect-Info</td>
<td>Sent from the NAS to indicate the nature of the user’s connection.</td>
</tr>
</tbody>
</table>
Table 26: RADIUS Role Mapping Attributes (Continued)

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EAP-Message</td>
<td>Encapsulates Extended Access Protocol [3] packets to allow the NAS to authenticate dial-in users by means of EAP without having to understand the EAP protocol.</td>
</tr>
<tr>
<td>Filter-Id</td>
<td>The name of the filter list for this user.</td>
</tr>
<tr>
<td>Framed-AppleTalk-Link</td>
<td>The AppleTalk network number used for the serial link to the user, which is another AppleTalk router.</td>
</tr>
<tr>
<td>Framed-AppleTalk-Network</td>
<td>The AppleTalk Network number which the NAS can probe to allocate an AppleTalk node for the user.</td>
</tr>
<tr>
<td>Framed-AppleTalk-Zone</td>
<td>The AppleTalk Default Zone to be used for this user.</td>
</tr>
<tr>
<td>Framed-Compression</td>
<td>A compression protocol to be used for the link.</td>
</tr>
<tr>
<td>Framed-IP-Address</td>
<td>The address to be configured for the user.</td>
</tr>
<tr>
<td>Framed-IP-Netmask</td>
<td>The IP netmask to be configured for the user when the user is a router to a network.</td>
</tr>
<tr>
<td>Framed-IPv6-Pool</td>
<td>Contains the name of an assigned pool used to assign an IPv6 prefix for the user.</td>
</tr>
<tr>
<td>Framed-IPv6-Route</td>
<td>Routing information to be configured for the user on the NAS.</td>
</tr>
<tr>
<td>Framed-IPX-Network</td>
<td>The IPX Network number to be configured for the user.</td>
</tr>
<tr>
<td>Framed-MTU</td>
<td>The Maximum Transmission Unit to be configured for the user, when it is not negotiated by some other means (such as PPP).</td>
</tr>
<tr>
<td>Framed-Pool</td>
<td>The name of an assigned address pool used to assign an address for the user.</td>
</tr>
<tr>
<td>Framed-Protocol</td>
<td>The framing to be used for framed access.</td>
</tr>
<tr>
<td>Framed-Route</td>
<td>Routing information to be configured for the user on the NAS.</td>
</tr>
<tr>
<td>Framed-Routing</td>
<td>The routing method for the user, when the user is a router to a network.</td>
</tr>
<tr>
<td>Idle-Timeout</td>
<td>Sets the maximum number of consecutive seconds of idle connection allowed to the user before termination of the session or prompt.</td>
</tr>
<tr>
<td>Keep-Alives</td>
<td>Use SNMP instead of keep-alives.</td>
</tr>
<tr>
<td>Login-IP-Host</td>
<td>Indicates the system with which to connect the user, when the Login-Service Attribute is included.</td>
</tr>
<tr>
<td>Login-LAT-Group</td>
<td>Contains a string identifying the LAT group codes that this user is authorized to use.</td>
</tr>
<tr>
<td>Login-LAT-Node</td>
<td>Indicates the Node with which the user is to be automatically connected by LAT.</td>
</tr>
<tr>
<td>Login-LAT-Port</td>
<td>Indicates the Port with which the user is to be connected by LAT.</td>
</tr>
<tr>
<td>Login-LAT-Service</td>
<td>Indicates the system with which the user is to be connected by LAT.</td>
</tr>
<tr>
<td>Login-Service</td>
<td>Indicates the service to use to connect the user to the login host.</td>
</tr>
</tbody>
</table>
Table 26: RADIUS Role Mapping Attributes (Continued)

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Login-TCP-Port</td>
<td>Indicates the TCP port with which the user is to be connected, when the Login-Service Attribute is also present.</td>
</tr>
<tr>
<td>MS-ARAP-Challenge</td>
<td>Only present in an Access-Request packet containing a Framed-Protocol Attribute with the value 3 (ARAP).</td>
</tr>
<tr>
<td>MS-ARAP-Password-Change-Reason</td>
<td>Indicates the reason for a server-initiated password change.</td>
</tr>
<tr>
<td>MS-Acct-Auth-Type</td>
<td>Represents the method used to authenticate the dial-up user.</td>
</tr>
<tr>
<td>MS-Acct-EAP-Type</td>
<td>Represents the Extensible Authentication Protocol (EAP) [15] type used to authenticate the dial-up user.</td>
</tr>
<tr>
<td>MS-BAP-Usage</td>
<td>Describes whether the use of BAP is allowed, disallowed or required on new multilink calls.</td>
</tr>
<tr>
<td>MS-CHAP-CPW-1</td>
<td>Allows the user to change password if it has expired.</td>
</tr>
<tr>
<td>MS-CHAP-CPW-2</td>
<td>Allows the user to change password if it has expired.</td>
</tr>
<tr>
<td>MS-CHAP-Challenge</td>
<td>Contains the challenge sent by a NAS to a Microsoft Challenge-Handshake Authentication Protocol (MS-CHAP) user.</td>
</tr>
<tr>
<td>MS-CHAP-Domain</td>
<td>Indicates the Windows NT domain in which the user was authenticated.</td>
</tr>
<tr>
<td>MS-CHAP-Error</td>
<td>Contains error data related to the preceding MS-CHAP exchange.</td>
</tr>
<tr>
<td>MS-CHAP-LM-Enc-PW</td>
<td>Contains the new Windows NT password encrypted with the old LAN Manager password hash.</td>
</tr>
<tr>
<td>MS-CHAP-MPPE-Keys</td>
<td>Contains two session keys for use by the Microsoft Point-to-Point Encryption Protocol (MPPE).</td>
</tr>
<tr>
<td>MS-CHAP-NT-Enc-PW</td>
<td>Contains the new Windows NT password encrypted with the old Windows NT password hash.</td>
</tr>
<tr>
<td>MS-CHAP-Response</td>
<td>Contains the response value provided by a PPP Microsoft Challenge-Handshake Authentication Protocol (MS-CHAP) user in response to the challenge.</td>
</tr>
<tr>
<td>MS-CHAP2-CPW</td>
<td>Allows the user to change password if it has expired.</td>
</tr>
<tr>
<td>MS-CHAP2-Response</td>
<td>Contains the response value provided by an MS-CHAP-V2 peer in response to the challenge.</td>
</tr>
<tr>
<td>MS-CHAP2-Success</td>
<td>Contains a 42-octet authenticator response string.</td>
</tr>
<tr>
<td>MS-Filter</td>
<td>Used to transmit traffic filters.</td>
</tr>
<tr>
<td>MS-Link-Drop-Time-Limit</td>
<td>Indicates the length of time (in seconds) that a link must be underutilized before it is dropped.</td>
</tr>
<tr>
<td>MS-Link-Utilization-Threshold</td>
<td>Represents the percentage of available bandwidth utilization below which the link must fall before the link is eligible for termination.</td>
</tr>
<tr>
<td>MS-MPPE-Encryption-Policy</td>
<td>Signifies whether the use of encryption is allowed or required.</td>
</tr>
<tr>
<td>MS-MPPE-Encryption-Types</td>
<td>Signifies the types of encryption available for use with MPPE.</td>
</tr>
</tbody>
</table>
## Table 26: RADIUS Role Mapping Attributes (Continued)

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS-MPPE-Recv-Key</td>
<td>Contains a session key for use by the Microsoft Point-to-Point Encryption Protocol (MPPE).</td>
</tr>
<tr>
<td>MS-MPPE-Send-Key</td>
<td>Contains a session key for use by the Microsoft Point-to-Point Encryption Protocol (MPPE).</td>
</tr>
<tr>
<td>MS-New-ARAP-Password</td>
<td>Transmits the new ARAP password during an ARAP password change operation.</td>
</tr>
<tr>
<td>MS-Old-ARAP-Password</td>
<td>Transmits the old ARAP password during an ARAP password change operation.</td>
</tr>
<tr>
<td>MS-Primary-DNS-Server</td>
<td>Indicates the address of the primary Domain Name Server (DNS) [16, 17] server to be used by the PPP peer.</td>
</tr>
<tr>
<td>MS-Primary-NBNS-Server</td>
<td>Indicates the address of the primary NetBIOS Name Server (NBNS) [18] server to be used by the PPP peer.</td>
</tr>
<tr>
<td>MS-RAS-Vendor</td>
<td>Indicates the manufacturer of the RADIUS client machine.</td>
</tr>
<tr>
<td>MS-RAS-Version</td>
<td>Indicates the version of the RADIUS client software.</td>
</tr>
<tr>
<td>MS-Secondary-DNS-Server</td>
<td>Indicates the address of the secondary DNS server to be used by the PPP peer.</td>
</tr>
<tr>
<td>MS-Secondary-NBNS-Server</td>
<td>Indicates the address of the secondary DNS server to be used by the PPP peer.</td>
</tr>
<tr>
<td>NAS-IP-Address</td>
<td>Indicates the identifying IP Address of the NAS that is requesting authentication of the user, and must be unique to the NAS within the scope of the RADIUS server.</td>
</tr>
<tr>
<td>NAS-Identifier</td>
<td>Contains a string identifying the NAS originating the Access-Request.</td>
</tr>
<tr>
<td>NAS-Port</td>
<td>Indicates the physical port number of the NAS that is authenticating the user.</td>
</tr>
<tr>
<td>NAS-Port-Id</td>
<td>Contains a text string that identifies the port of the NAS that is authenticating the user.</td>
</tr>
<tr>
<td>NAS-Port-Type</td>
<td>Indicates the type of the physical port of the NAS that is authenticating the user.</td>
</tr>
<tr>
<td>Password-Retry</td>
<td>Indicates how many authentication attempts a user is allowed to attempt before being disconnected.</td>
</tr>
<tr>
<td>Port-Limit</td>
<td>Sets the maximum number of ports to be provided to the user by the NAS.</td>
</tr>
<tr>
<td>Prompt</td>
<td>Indicates to the NAS whether it should echo the user’s response as it is entered, or not echo it.</td>
</tr>
<tr>
<td>Proxy-State</td>
<td>A proxy server can send this attribute to another server when forwarding an Access-Request. The attribute must be returned unmodified in the Access-Accept, Access-Reject or Access-Challenge.</td>
</tr>
<tr>
<td>Reply-Message</td>
<td>Text that can be displayed to the user.</td>
</tr>
<tr>
<td>Service-Type</td>
<td>The type of service the user has requested, or the type of service to be provided.</td>
</tr>
</tbody>
</table>
### Table 26: RADIUS Role Mapping Attributes (Continued)

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Session-Timeout</td>
<td>Sets the maximum number of seconds of service to be provided to the user before termination of the session or prompt.</td>
</tr>
<tr>
<td>State</td>
<td>A packet must have only zero or one State Attribute. Usage of the State Attribute is implementation dependent.</td>
</tr>
<tr>
<td>Telephone-number</td>
<td>Using the Calling-Station-Id and Called-Station-Id RADIUS attributes, authorization and subsequent tunnel attributes can be based on the phone number originating the call, or the number being called.</td>
</tr>
<tr>
<td>Termination-Action</td>
<td>The action the NAS should take when the specified service is completed.</td>
</tr>
<tr>
<td>Tunnel-Assignment-ID</td>
<td>Indicates to the tunnel initiator the particular tunnel to which a session is to be assigned.</td>
</tr>
<tr>
<td>Tunnel-Client-Auth-ID</td>
<td>Specifies the name used by the tunnel initiator during the authentication phase of tunnel establishment.</td>
</tr>
<tr>
<td>Tunnel-Client-Endpoint</td>
<td>Contains the address of the initiator end of the tunnel.</td>
</tr>
<tr>
<td>Tunnel-Link-Reject</td>
<td>Marks the rejection of the establishment of a new link in an existing tunnel.</td>
</tr>
<tr>
<td>Tunnel-Link-Start</td>
<td>Marks the creation of a tunnel link.</td>
</tr>
<tr>
<td>Tunnel-Link-Stop</td>
<td>Marks the destruction of a tunnel link.</td>
</tr>
<tr>
<td>Tunnel-Medium-Type</td>
<td>The transport medium to use when creating a tunnel for those protocols (such as L2TP) that can operate over multiple transports.</td>
</tr>
<tr>
<td>Tunnel-Medium-Type</td>
<td>The transport medium to use when creating a tunnel for those protocols (such as L2TP) that can operate over multiple transports.</td>
</tr>
<tr>
<td>Tunnel-Password</td>
<td>A password to be used to authenticate to a remote server.</td>
</tr>
<tr>
<td>Tunnel-Preference</td>
<td>If the RADIUS server returns more than one set of tunneling attributes to the tunnel initiator, you should include this attribute in each set to indicate the relative preference assigned to each tunnel.</td>
</tr>
<tr>
<td>Tunnel-Private-Group-ID</td>
<td>The group ID for a particular tunneled session.</td>
</tr>
<tr>
<td>Tunnel-Reject</td>
<td>Marks the rejection of the establishment of a tunnel with another node.</td>
</tr>
<tr>
<td>Tunnel-Server-Auth-ID</td>
<td>Specifies the name used by the tunnel terminator during the authentication phase of tunnel establishment.</td>
</tr>
<tr>
<td>Tunnel-Server-Endpoint</td>
<td>The address of the server end of the tunnel.</td>
</tr>
<tr>
<td>Tunnel-Start</td>
<td>Marks the establishment of a tunnel with another node.</td>
</tr>
<tr>
<td>Tunnel-Stop</td>
<td>Marks the destruction of a tunnel to or from another node.</td>
</tr>
<tr>
<td>Tunnel-Type</td>
<td>The tunneling protocol(s) to be used (in the case of a tunnel initiator) or the tunneling protocol in use (in the case of a tunnel terminator).</td>
</tr>
</tbody>
</table>
Understanding Clustering Issues
Accounting messages are sent to the RADIUS server by each cluster node without consolidation. RADIUS accounting on the Infranet Controller follows these assumptions:

- If the cluster is active/passive, all users are connected to one node at a time.
- If the cluster is active/active and does not use a balancer, users are connected to different nodes but are static.
- If the cluster is active/active and uses a balancer, the balancer usually enforces a persistent source IP. In this case, users are always connected to the same node.

The Infranet Controller does not support load balancing for RADIUS.

Understanding the Interim Update Feature
If you want a server to receive interim accounting messages, you can statically configure an interim value on the client, in which case, the locally-configured value overrides any value that might be included in the RADIUS Access-Accept message.

The octet count reported in the accounting messages is the cumulative total since the beginning of the user session.

The interim update byte count is only supported based on a user session, not on SAM or NC sessions.

The minimum interim update interval is 15 minutes. The data statistics (bytes in and bytes out) for RADIUS Accounting may not be sent for a J-SAM/W-SAM/NC session if the session is less than five minutes long and the applications keep the connections open all the time.

Configuring an eTrust SiteMinder Server Instance

When you configure the Infranet Controller to authenticate users with an eTrust SiteMinder policy server, the Infranet Controller passes the user’s credentials to SiteMinder during authentication. Once SiteMinder receives the credentials, it may use standard username and password authentication, ACE SecurID tokens, or client-side certificates to authenticate the credentials (as explained in “Authentication Using Various Authentication Schemes” on page 223).
The Infranet Controller also passes a protected resource URL to SiteMinder during authentication in order to determine which SiteMinder realm it should use to authenticate the user. When the Infranet Controller passes the protected resource URL, SiteMinder authorizes the user’s URL against the realm that is associated with the resource and allows the user to seamlessly access any resources whose protection levels are equal to or less than the resource the Infranet Controller passed.

This topic includes the following information about eTrust SiteMinder servers:

- “eTrust SiteMinder Overview” on page 220
- “Configuring SiteMinder to Work with the Infranet Controller” on page 224
- “Configuring the Infranet Controller to Work with SiteMinder” on page 230

**eTrust SiteMinder Overview**

The Infranet Controller enables single sign-on (SSO) from the Infranet Controller to SiteMinder-protected resources using SMSESSION cookies. A SMSESSION cookie is a security token that encapsulates SiteMinder session information. Depending on your configuration, either the SiteMinder Web agent or the Infranet Controller creates a SMSESSION cookie and then posts the cookie to the following locations so the user does not have to re-authenticate if he wants to access additional resources:

- **The Infranet Controller:** If the user tries to access a SiteMinder resource from within his Infranet Controller session (for example, from the Infranet Controller file browsing page), the Infranet Controller passes its cached SMSESSION cookie to the Web agent for authentication.

- **The user’s Web browser:** If the user tries to access a SiteMinder resource from outside of his Infranet Controller session (for example, when using a protected resource on a standard agent), SiteMinder uses the cached SMSESSION cookie stored in the user’s Web browser to authenticate/authorize the user.

If you enable the Automatic Sign-In option (as explained in “Automatic Sign-In” on page 235), the Infranet Controller can use an SMSESSION cookie generated by another agent to enable single sign-on from a SiteMinder resource to the Infranet Controller. When a user accesses the Infranet Controller sign-in page with an SMSESSION cookie, the Infranet Controller verifies the SMSESSION cookie. Upon successful verification, the Infranet Controller establishes an Infranet Controller session for the user. You can use the following authentication mechanisms when you enable automatic sign-in through the Infranet Controller:

- **Custom agent:** The Infranet Controller authenticates the user against the policy server and generates a SMSESSION cookie. When you select this option, you can enable SSO on other SiteMinder agents that use the same policy server. To enable SSO on these agents, update each of them to accept third party cookies (as explained in “Authenticate using custom agent” on page 236). If you select this option and the user enters his Infranet Controller session with an SMSESSION cookie, the Infranet Controller attempts automatic sign-in when the user enters the Infranet Controller session.
- **HTML form post**: The Infranet Controller posts credentials to a standard Web agent that you have already configured. The Web agent then creates `SMSESSION` cookies. If you select this option, you cannot use SecurID New Pin and Next Token modes or client-side certificate authentication (as explained in “Authenticate using HTML form post” on page 236). If you select this option and the user enters his Infranet Controller session with an `SMSESSION` cookie, the Infranet Controller attempts automatic sign-in when the user enters the Infranet Controller session.

- **Delegated authentication**: The Infranet Controller delegates authentication to a standard agent. If this option is enabled, the Infranet Controller tries to determine the FCC URL associated with the protected resource. The Infranet Controller then redirects the user to the FCC URL with the Infranet Controller sign-in URL as the TARGET. Upon successful authentication, the user is redirected back to the Infranet Controller with an `SMSESSION` cookie and the Infranet Controller does an automatic sign-in for the user (as explained in “Delegate authentication to a standard agent” on page 238).
NOTE:

- At the time of this printing, Juniper Networks supports eTrust SiteMinder server version 6.0 and version 5.5 with standard agent versions 6 and 5QMR5. If you run older agents than the supported agents, you may experience cookie validation problems, including crossed log entries and intermittent user timeouts.

- You can choose which eTrust SiteMinder server version you want to support when you create a server instance. You can choose version 5.5, which supports both versions 5.5 and 6.0, or you can choose version 6.0, which supports only version 6.0. There is no difference in the SiteMinder authentication server functionality based on which version you select. This option only controls the version of the Netegrity SDK to use. We recommend you match the compatibility mode with the version of the Policy Server.

- When you use SiteMinder to authenticate, the primary and backup policy servers must run the same SiteMinder server software version. A mixed deployment (where the primary server runs a different server software version than the backup) is not supported.

- SiteMinder does not store the IP address in the \texttt{SMSESSION} cookie, and therefore cannot pass it to the Infranet Controller appliance.

- SiteMinder sends the \texttt{SMSESSION} cookie to the Infranet Controller as a persistent cookie. To maximize security, the Infranet Controller resets the persistent cookie as a session cookie once authentication is complete.

- When you use SiteMinder to authenticate, the Infranet Controller disregards any Infranet Controller session and idle timeouts and uses session and idle timeouts set through the SiteMinder realm instead.

- When you use SiteMinder to authenticate, users must access the Infranet Controller using a fully-qualified domain name. This is because the SiteMinder \texttt{SMSESSION} cookie is only sent for the domain for which it is configured. If users access the Infranet Controller using an IP address, they may receive an authentication failure and will be prompted to authenticate again.

- The Infranet Controller logs any SiteMinder error codes on the System > Log/Monitoring > User Access page. For information on the SiteMinder error codes, see the SiteMinder documentation.
Authentication Using Various Authentication Schemes

Within SiteMinder, an authentication scheme is a way to collect user credentials and determine the identity of a user. You may create different authentication schemes and associate different protection levels with each. For example, you may create two schemes—one that authenticates users based solely on the users’ client-side certificates and provides them a low protection level, and a second that uses ACE SecurID token authentication and provides users a higher protection level. The Infranet Controller works with the following types of SiteMinder authentication schemes:

- **Basic username and password authentication**—The user’s name and password are passed to the SiteMinder policy server. The policy server may then authenticate them itself or pass it to another server for authentication.

- **ACE SecurID token authentication**—The SiteMinder policy server authenticates users based on a username and password generated by an ACE SecurID token.

- **Client-side certificate authentication**—The SiteMinder policy server authenticates users based on their client-side certificate credentials. If you choose this authentication method, the Web browser displays a list of client certificates from which users can select.

**NOTE:**

- If you choose to authenticate users with this method, you must import the client certificate into the Infranet Controller through the **System > Certificates > Trusted Client CAs** tab. For more information, see “Using Trusted Client CAs” on page 363.

- If you do not want to display the standard Infranet Controller sign in page to users, you may change it using the customizable sign-in pages feature. For more information, see the **Custom Sign-In Pages Solution Guide**.

- SiteMinder client-side certificate authentication is separate from Infranet Controller client-side certificate authentication. If you choose both, the Infranet Controller first authenticates using the Infranet Controller configuration parameters. If this succeeds, it then passes certificate values to SiteMinder for authentication.

For configuration information, see:

- “Creating a SiteMinder Authentication Scheme for the Infranet Controller” on page 226

- “Configuring the Infranet Controller to Work with Multiple Authentication Schemes” on page 230
Determining the User’s Username

With the availability of different authentication schemes and sign-in points, the Infranet Controller may obtain a username from various sources, such as a policy server header, certificate attribute, or from the Infranet Controller sign-in page. Listed below are the various methods a user may employ to access the Infranet Controller and how the Infranet Controller determines the username for each. When a user:

- **Signs in through the standard Infranet Controller sign-in page**—The Infranet Controller first checks the username that the policy server returned in its `OnAuthAccept` response header. If SiteMinder does not define a username, the Infranet Controller uses the name that the user entered during sign-in. Otherwise, if neither SiteMinder nor the user provide a username because the user authenticates using a client certificate, the Infranet Controller uses the `UserDN` value set by the policy server.

- **Automatically signs in to the Infranet Controller using SiteMinder credentials**—The Infranet Controller first checks the username that the policy server returned in its `OnAuthAccept` response header. If SiteMinder does not define a username, the Infranet Controller checks the `SMSESSION` cookie. Otherwise, if SiteMinder does not populate the response header or `SMSESSION` cookie with a username, the Infranet Controller checks the `UserDN` value in the `SMSESSION` cookie.

Once the Infranet Controller determines which username to use, it saves it in its session cache and references it when a user wants to access additional resources (as explained in “eTrust SiteMinder Overview” on page 220).

In order to consistently return the correct username to the Infranet Controller, you should configure the `OnAuthAccept` response on the SiteMinder policy server, as explained in “Creating a Rule/Response Pair to Pass Usernames to the Infranet Controller” on page 228.

**Configuring SiteMinder to Work with the Infranet Controller**

The following procedures outline how to configure a SiteMinder policy server to work with the Infranet Controller. These are not complete SiteMinder configuration instructions—they are only intended to help you make SiteMinder work with the Infranet Controller. For in-depth SiteMinder configuration information, refer to the documentation provided with your SiteMinder policy server.

---

**NOTE:** The instructions shown here are for SiteMinder policy server version 5.5. Instructions may vary slightly if you are using a different product version.

To configure SiteMinder to work with the Infranet Controller, you must:

1. “Configuring the SiteMinder Agent” on page 225
2. “Creating a SiteMinder Authentication Scheme for the Infranet Controller” on page 226
3. “Creating a SiteMinder Domain for the Infranet Controller” on page 227
4. “Creating a SiteMinder Realm for the Infranet Controller” on page 227

5. “Creating a Rule/Response Pair to Pass Usernames to the Infranet Controller” on page 228

6. “Creating a SiteMinder Policy Under the Domain” on page 230

**Configuring the SiteMinder Agent**

A SiteMinder *agent* filters user requests to enforce access controls. For instance, when a user requests a protected resource, the agent prompts the user for credentials based on an authentication scheme, and sends the credentials to a SiteMinder policy server. A *Web agent* is simply an agent that works with a Web server. When configuring SiteMinder to work with the Infranet Controller, you must configure the Infranet Controller as a Web agent in most cases.

**NOTE:** If you select the Delegate authentication to a standard agent option, you must set the following options in the agent configuration object of the standard Web agent host the FCC URL:

- `EncryptAgentName=no`
- `FCCCompatMode=no`

To configure the Infranet Controller as a Web agent on the SiteMinder policy server:

1. In the SiteMinder Administration interface, choose the *System* tab.

2. Right-click on *Agents* and choose *Create Agent*.

3. Enter a name for the Web agent and (optionally) a description. Note that you need to enter this name when creating a SiteMinder realm, (as explained in “Creating a SiteMinder Realm for the Infranet Controller” on page 227) and when configuring the Infranet Controller (as explained in “Agent Name, Secret” on page 233).

4. You must select the **Support 5.x agents** option for compatibility with the Infranet Controller.

5. Under *Agent Type*, select *SiteMinder* and then select *Web Agent* from the drop-down list. You must select this setting for compatibility with the Infranet Controller.

6. Under *IP Address or Host Name*, enter the name or IP address of the Infranet Controller.

7. In the *Shared Secret* fields, enter and confirm a secret for the Web agent. Note that you need to enter this secret when configuring the Infranet Controller (as explained in “Agent Name, Secret” on page 233).

8. Click **OK**.
**Creating a SiteMinder Authentication Scheme for the Infranet Controller**

Within SiteMinder, an *authentication scheme* provides a way to collect credentials and determine the identity of a user.

To configure a SiteMinder authentication scheme for the Infranet Controller:

1. In the SiteMinder Administration interface, choose the **System** tab.
2. Right-click on **Authentication Schemes** and choose **Create Authentication Scheme**.
3. Enter a name for the scheme and (optionally) a description. Note that you need to enter this name when configuring the SiteMinder realm (as explained in “Creating a SiteMinder Realm for the Infranet Controller” on page 227).
4. Under **Authentication Scheme Type**, select one of the following options:
   - **Basic Template**
   - **HTML Form Template**
   - **SecurID HTML Form Template**
   - **X509 Client Cert Template**
   - **X509 Client Cert and Basic Authentication**

**NOTE:**
- The Infranet Controller only supports the authentication scheme types listed here.
- If you select **X509 Client Cert Template** or **X509 Client Cert and Basic Authentication**, you must import the certificate into the Infranet Controller through the **System > Certificates > Trusted Client CAs** tab. For more information, see “Using Trusted Client CAs” on page 363.

5. Enter a protection level for the scheme. Note that this protection level carries over to the SiteMinder realm that you associate with this scheme. For more information, see “Creating a SiteMinder Realm for the Infranet Controller” on page 227.

6. Select the **Password Policies Enabled for this Authentication Scheme** if you want to reauthenticate users who request resources with a higher protection level than they are authorized to access.

7. In the **Scheme Setup** tab, enter the options required by your authentication scheme type.

---

1. If you are using SecurID authentication, you must choose SecurID HTML Form Template (instead of SecurID Template). Choosing this option enables the Policy Server to send ACE sign-in failure codes to the Infranet Controller.
1. In the SiteMinder Administration interface, choose the **Domains** tab.

2. Expand the domain that you created for the Infranet Controller. For more information, see “Creating a SiteMinder Domain for the Infranet Controller” on page 227.

3. Right-click on **Realms** and choose **Create Realm**.

4. Enter a name and (optionally) description for the realm.

5. In the **Agent** field, select the Web agent that you created for the Infranet Controller. For more information, see “Configuring the SiteMinder Agent” on page 225.
6. In the **Resource Filter** field, enter a protected resource. This resource inherits the protection level specified in the corresponding authentication scheme. For the default protection level, enter `/ive-authentication`. Note that you need to enter this resource when configuring the Infranet Controller (as explained in “Protected Resource” on page 233). Or, if you use sign-in policies with non-default URLs such as `*/nete` or `*/cert`, you must have corresponding resource filters in the SiteMinder configuration.

7. From the **Authentication Schemes** list, select the scheme that you created for the Infranet Controller (as explained in “Creating a SiteMinder Authentication Scheme for the Infranet Controller” on page 226).

8. Click **OK**.

**Creating a Rule/Response Pair to Pass Usernames to the Infranet Controller**

Within SiteMinder, you can use rules to trigger responses when authentication or authorization events take place. A response passes DN attributes, static text, or customized active responses from the SiteMinder policy server to a SiteMinder agent. When you configure SiteMinder to work with the Infranet Controller, you must create a rule that fires when a user successfully authenticates. Then, you must create a corresponding response that passes the user’s username to the Infranet Controller Web agent.

To create a new rule:

1. In the SiteMinder Administration interface, choose the **Domains** tab.

2. Expand the domain that you created for the Infranet Controller (as explained in “Creating a SiteMinder Domain for the Infranet Controller” on page 227) and then expand **Realms**.

3. Right-click on the realm that you created for the Infranet Controller (as explained in “Creating a SiteMinder Realm for the Infranet Controller” on page 227) and choose **Create Rule under Realm**.

4. Enter a name and (optionally) description for the rule.

5. Under **Action**, choose **Authentication Events** and then select **OnAuthAccept** from the drop-down list.

6. Select **Enabled**.

7. Click **OK**.

To create a new response:

1. In the SiteMinder Administration interface, choose the **Domains** tab.

2. Expand the domain that you created for the Infranet Controller (as explained in “Creating a SiteMinder Domain for the Infranet Controller” on page 227).

3. Right-click on **Responses** and select **Create Response**.
4. Enter a name and (optionally) a description for the response.

5. Select SiteMinder and then select the Infranet Controller Web agent (as explained in “Configuring the SiteMinder Agent” on page 225).

6. Click Create.

7. From the Attribute list, select WebAgent-HTTP-Header-Variable.

8. Under Attribute Kind, select Static.

9. Under Variable Name, enter IVEUSERNAME.

10. Under Variable Value, enter a user name.

11. Click OK.

Creating SiteMinder User Attributes for Infranet Controller Role Mapping

If you create SiteMinder user attributes on a SiteMinder policy server, you can use those user attributes in Infranet Controller role mapping rules to map users to roles. For example, you might want to map users to various Infranet Controller roles based on their department. To use a SiteMinder user attribute in a role mapping rule, you reference the cookie name contained in the SiteMinder user attribute cookie.

The following procedure is required only if you want to use SiteMinder user attributes in Infranet Controller role mapping rules.

To create user attributes on a SiteMinder server:

1. In the SiteMinder Administration interface, choose the Domains tab.

2. Expand the domain that you created for the Infranet Controller (as explained in “Creating a SiteMinder Domain for the Infranet Controller” on page 227).

3. Right-click on Responses and select Create Response.

4. Enter a name and (optionally) a description for the response.

5. Select SiteMinder and then select the Infranet Controller Web agent (as explained in “Configuring the SiteMinder Agent” on page 225).

6. Click Create.

7. From the Attribute list, select WebAgent-HTTP-Cookie-Variable.

8. Under Attribute Kind, select User Attribute.

9. For Cookie Name, enter a name for the cookie, such as department. You can reference this cookie name in an Infranet Controller role mapping rule.

10. For Attribute Name, enter the name of the attribute in the SiteMinder user directory. (This refers to the attribute in the LDAP server that SiteMinder uses.)

11. Click OK.
12. Assign the User Attribute response to an OnAuthAccept type rule. (See “Creating a Rule/Response Pair to Pass Usernames to the Infranet Controller” on page 228.)

13. Reference the cookie name in a role mapping rule for an Infranet Controller realm that uses the SiteMinder policy server. For instructions, see “Using SiteMinder User Attributes for Infranet Controller Role Mapping” on page 240.

Creating a SiteMinder Policy Under the Domain

Within SiteMinder, a policy associates users with rules. To configure a SiteMinder policy under a domain, in the SiteMinder Administration interface, choose the Domains tab, select the domain to which you want to add a policy, right-click on Policies, and choose Create Policy.

Configuring the Infranet Controller to Work with SiteMinder

This section includes the following instructions for configuring the Infranet Controller to work with a SiteMinder policy server:

- “Configuring the Infranet Controller to Work with Multiple Authentication Schemes” on page 230
- “Configuring the Infranet Controller to Grant Users Different Protected Resources” on page 231
- “Defining an eTrust SiteMinder Server Instance” on page 232
- “Defining a SiteMinder Realm for Automatic Sign-In” on page 241

Configuring the Infranet Controller to Work with Multiple Authentication Schemes

To configure the Infranet Controller to work with multiple SiteMinder authentication schemes, you must:

1. Configure the authentication schemes on the SiteMinder policy server. For instructions, see “Creating a SiteMinder Authentication Scheme for the Infranet Controller” on page 226.

2. Create one Infranet Controller instance of the SiteMinder policy server for all SiteMinder authentication schemes you want to use. For instructions, see “Defining an eTrust SiteMinder Server Instance” on page 232.

3. Specify which Infranet Controller realm should use the Infranet Controller instance of the SiteMinder policy server to authenticate and authorize administrators and users. For instructions, see “Creating an Authentication Realm” on page 246.
4. For each protected resource on the SiteMinder policy server, create an Infranet Controller sign-in policy. In the Authentication > Authentication > Signing In Policies > New Sign-In Policy page:

- Specify an Infranet Controller sign-in URL that matches the SiteMinder protected resource URL on the policy server. Make the path portion of the URL match the SiteMinder resource filter in the SiteMinder realm configuration. For example, you can specify */ACE/ as an Infranet Controller sign-in URL to match a SiteMinder URL of XYZ/ACE, where XYZ is the name of a realm.

- Select the Infranet Controller realm that you specified should use the SiteMinder policy server.

For instructions, see “Configuring Sign-In Policies” on page 259.

The user signs into the Infranet Controller using one of the Infranet Controller sign-in URLs. The Infranet Controller sends the protected resource URL to SiteMinder, and based on the resource, SiteMinder determines which type of scheme to use to authenticate the user. The Infranet Controller collects the credentials that the authentication scheme requires, and then passes them to SiteMinder for authentication.

**Configuring the Infranet Controller to Grant Users Different Protected Resources**

To configure the Infranet Controller to grant users access to various SiteMinder protected resources (and by association, different protection levels), you must:

1. Define which resources the SiteMinder server should protect. Each of these resources inherits a protection level from a corresponding SiteMinder authentication scheme. For instructions, see “Creating a SiteMinder Realm for the Infranet Controller” on page 227.

2. Create one Infranet Controller instance of the SiteMinder policy server for all protected resources and corresponding protection levels that you want to allow. For instructions, see “Defining an eTrust SiteMinder Server Instance” on page 232.

3. Specify which Infranet Controller realm should use the Infranet Controller instance of the SiteMinder policy server. For instructions, see “Creating an Authentication Realm” on page 246.

4. For each resource on the SiteMinder policy server, create an Infranet Controller sign-in policy for each realm-level resource filter. In the configuration page for the sign-in policy, specify:

- An Infranet Controller sign-in URL that matches the protected resource URL on the policy server. Make the path portion of the URL match the SiteMinder resource filter. For example, you may define the following URLs:
  
  - https://employees.yourcompany.com/sales
  - https://employees.yourcompany.com/engineering
When users sign into the first URL, they can access the “sales” protected resource, and when they sign into the second URL, they can access the “engineering” protected resource.

To define a default resource (ive-authentication), enter * in the path portion of the URL.

Select the Infranet Controller realm that you specified should use the SiteMinder policy server.

For instructions, see “Configuring Sign-In Policies” on page 259.

During production, the user signs into the Infranet Controller using one of the URLs. The Infranet Controller extracts the protected resource from the URL and authenticates the user against the appropriate realm.

**Defining an eTrust SiteMinder Server Instance**

Within the Infranet Controller, a SiteMinder instance is a set of configuration settings that defines how the Infranet Controller interacts with the SiteMinder policy server. After defining the SiteMinder server instance, specify which Infranet Controller realm(s) should use the Infranet Controller instance of the SiteMinder policy server to authenticate and authorize administrators and users. For instructions, see “Creating an Authentication Realm” on page 246.

To define an eTrust SiteMinder server instance:

1. In the admin console, choose Authentication > Auth. Servers.

2. Do one of the following:

   - To create a new server instance on the Infranet Controller, select SiteMinder Server from the New list, and then click New Server.

   - To update an existing server instance, click the appropriate link in the Authentication/Authorization Servers list.

3. Configure the server using the settings described in Table 27.

4. To add SiteMinder user attributes to the SiteMinder server instance:

   a. Click Server Catalog to display the server catalog.

   b. Enter the SiteMinder user attribute cookie name in the Attribute field in the server catalog and then click Add Attribute. (For information on SiteMinder user attribute cookies, see “Creating SiteMinder User Attributes for Infranet Controller Role Mapping” on page 229.)

   c. When you are finished adding cookie names, click OK. The Infranet Controller displays the names of the SiteMinder user attribute cookies in the Attribute list on the Role Mapping Rule page. For configuration instructions, see “Using SiteMinder User Attributes for Infranet Controller Role Mapping” on page 240.

5. Click Save Changes.
6. Set advanced SiteMinder configuration options (optional) using the settings described in Table 28.

**NOTE:** For information on monitoring and deleting the sessions of users who are currently signed in through the server, see "Monitoring Active Users" on page 434.

**Table 27: eTrust SiteMinder Configuration Options**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name to identify the server instance.</td>
</tr>
<tr>
<td>Policy Server</td>
<td>Enter the name or IP address of the SiteMinder policy server that you want to authenticate users.</td>
</tr>
<tr>
<td>Backup Server(s), Failover Mode</td>
<td>Enter a comma-delimited list of backup policy servers (optional). Then, choose a failover mode:</td>
</tr>
<tr>
<td></td>
<td>Select <strong>Yes</strong> to have the Infranet Controller appliance use the main policy server unless it fails.</td>
</tr>
<tr>
<td></td>
<td>Select <strong>No</strong> to have the Infranet Controller appliance load balance among all the specified policy servers.</td>
</tr>
<tr>
<td>Agent Name, Secret</td>
<td>Enter the shared secret and agent name specified in &quot;Configuring the SiteMinder Agent&quot; on page 225. Note that these are case-sensitive.</td>
</tr>
<tr>
<td>Compatible with</td>
<td>Choose a SiteMinder server version. Version 5.5 supports versions 5.5 and 6.0. Version 6.0 supports only version 6.0 of the SiteMinder server API. The default value is 5.5 policy servers.</td>
</tr>
<tr>
<td>On logout, redirect to</td>
<td>Specify a URL to which users are redirected when they sign out of the Infranet Controller (optional). You leave this field empty, users see the default Infranet Controller sign-in page. Note: The On logout, redirect to field is included in the product release for backwards-compatibility, but is scheduled for discontinuance. If you want to redirect users to a different sign-in page when they sign out, we strongly recommend that you use the customizable sign-in pages feature instead. For more information, see the Custom Sign-In Pages Solution Guide.</td>
</tr>
<tr>
<td>Protected Resource</td>
<td>Specify a default protected resource specified in &quot;Creating a SiteMinder Realm for the Infranet Controller&quot; on page 227. If you do not create sign-in policies for SiteMinder, the Infranet Controller uses this default URL to set the user’s protection level for the session. The Infranet Controller also uses this default URL if you select the Automatic Sign-In option. If your users are signing in to the &quot;*&quot; URL (default Infranet Controller sign-in page), enter any URL (&quot;/Infranet Controller-authentication&quot; is the default) to set the protection level to the default Infranet Controller value. If you do create sign-in policies for SiteMinder, the Infranet Controller uses those sign-in policies instead of this default URL. Note: You must enter a forward slash (/) at the beginning of the resource (for example, &quot;/five-authentication&quot;).</td>
</tr>
<tr>
<td>Resource Action</td>
<td>(Read-only) For new SiteMinder server instances, the Infranet Controller sets the resource action to GET. If your SiteMinder instance is upgraded from a 3.x instance, the Infranet Controller uses the resource action (for example, GET, POST, or PUT) that you previously chose. Note that to change an existing resource action to GET, you must delete the old SiteMinder server instance and then create a new instance that uses GET.</td>
</tr>
</tbody>
</table>
Users authenticate using tokens or one-time passwords

Select this option if you want the Infranet Controller to prompt the user for a token instead of a password, that is, if users submit tokens or one-time use passwords to the Infranet Controller. For example, you can use this option to dynamically prompt for a password or token based on sign-in policies by configuring two instances of the same authentication server. You can use one instance for wireless users that has this option enabled and prompts the user for a token, and another instance for wired users that has this option disabled and prompts the user for a password.

SMSESSION cookie settings

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| Cookie Domain              | Enter the cookie domain of the Infranet Controller. (A cookie domain is a domain in which the user’s cookies are active—The Infranet Controller sends cookies to the user’s browser in this domain.) Note:  
  - Multiple domains should use a leading period and be comma-separated. For example: .sales.myorg.com, .marketing.myorg.com  
  - Domain names are case-sensitive.  
  - You cannot use wildcard characters.  
  For example, if you define “.juniper.net”, the user must access the Infranet Controller as “http://ive.juniper.net” in order to ensure that his SMSESSION cookie is sent back to the Infranet Controller. |
| Protocol                   | (Read-only) Indicates that the Infranet Controller uses HTTPS protocol to send cookies to the user’s Web browser.                                                                                                                                                                                                                                                                                                                                                                           |
| IC Cookie Domain           | Enter the Internet domain(s) to which the Infranet Controller sends the SMSESSION cookie using the same guidelines outlined for the Cookie Domain field. (An Infranet Controller cookie domain enables single sign-on across multiple cookie domains. It allows a user’s information to carry with him when he navigates from one domain to another.) If you have configured a cookie provider to enable single sign-on across multiple cookie domains, enter the domain of the cookie provider. Otherwise, enter the domain(s) of the Web agents for which single sign-on is desired. For example: .juniper.net |
| Protocol                   | Choose HTTPS to send cookies securely if other Web agents are set up to accept secure cookies, or HTTP to send cookies non-securely.                                                                                                                                                                                                                                                                                                                                                         |
SiteMinder authentication settings

**Automatic Sign-In**
Select the **Automatic Sign-In** option to automatically sign in users who have a valid SMSESSION cookie into the Infranet Controller. Then, select the authentication realm to which the users are mapped. If you select this option, note that:

- If the protection level associated with a user’s SMSESSION cookie is different from the protection level of the Infranet Controller realm, the Infranet Controller uses the protection level associated with the cookie.
- In order to enable single sign-on from another Web agent to the Infranet Controller, the Infranet Controller needs to validate an existing SMSESSION cookie created by a standard Web agent.
- The Infranet Controller supports the following realm and role limitations with the **Automatic Sign-in** feature: Host Checker, IP address, browser, and concurrent user limit checks. Certificate and password restrictions are not supported since they are not applicable to automatically signed-in users.
- The Infranet Controller does not support the **Automatic Sign in** feature for administrator roles. This feature is only available for end-users.

When you select the **Automatic Sign-In** option, you must also configure the following sub-options:

- **To assign user roles, use this authentication realm**
  Select an authentication realm for automatically signed-in users. The Infranet Controller maps the user to a role based on the role mapping rules defined in the selected realm.

  **Note:** If you map users to roles based on username, see “Determining the User’s Username” on page 224 for information about which username the Infranet Controller uses.

- **If Automatic Sign In fails, redirect to**
  Enter an alternative URL for users who sign into the Infranet Controller through the Automatic Sign-In mechanism explained in “Automatic Sign-In” on page 235. The Infranet Controller redirects users to the specified URL if the Infranet Controller fails to authenticate and no redirect response is received from the SiteMinder policy server. If you leave this field empty, users are prompted to sign back in to the Infranet Controller.

  **Note:**
  - Users who sign in through the Infranet Controller sign-in page are always redirected back to the Infranet Controller sign-in page if authentication fails.
  - If you are using the customizable UI (Custom Pages) option explained in the *Custom Sign-In Pages Solution Guide*.
    You must account for both of these special cases in your custom page:
    - Session and idle timeouts: /dana-na/auth/welcome.cgi?p=timed-out
    - Failed cookie validation: /dana-na/auth/welcome.cgi?p=failed
Authenticate using custom agent

Choose this option if you want to authenticate using the Infranet Controller custom Web agent. Note that if you select this option, you must also:

- Update all of your standard Web agents to the appropriate SiteMinder Agent Quarterly Maintenance Release (QMR) in order to accept the cookies created by the Infranet Controller. If you are running SiteMinder version 5 Web agents, use the QMR5 hot fix. The Infranet Controller is compatible with version 5.x and later SiteMinder agents. Older versions of SiteMinder agents are susceptible to cookie validation failures.

- Set the Accept Third Party Cookie attribute (AcceptTPCookie) to yes in the Web agent’s configuration file (webagent.conf) or to 1 in the Windows Registry for the IIS Web server. The location of the attribute depends on the SiteMinder version and Web server you are using. For more information, please refer to the documentation provided with your SiteMinder server.

Authenticate using HTML form post

Choose this option if you want to post user credentials to a standard Web agent that you have already configured rather than contacting the SiteMinder policy server directly. If you select this option, the Web agent contacts the policy server to determine the appropriate sign-in page to display to the user. In order to configure the Infranet Controller to “act like a browser” that posts credentials to the standard Web agent, you must enter the information defined below. The easiest way to find this information is to:

1. Open a Web browser and enter the URL of the standard web agent that you want to use. For example, http://webagent.juniper.net
2. Note the URL of the SiteMinder sign-in page that appears. For example:
   http://webagent.juniper.net/siteminderagent/forms/login.fcc?TYPE=3&REALMOID=06-2525fa65-5a7f-11d5-9ee0-0003471b786c&GUID=&SMAUTHREASON=0&TARGET=$SM$http%3a%2f%2fwebagent%2ejuniper%2enet%2fportal%2findex%2ejs
3. Extract information from the URL to enter in the fields that follow.

Note:

- You cannot use SecurID New Pin and Next Token modes, client-side certificate authentication, or SNMP traps with the Authenticate using HTML form post option.
- The Authorize While Authenticating option is not applicable with the HTML form post option.
- You can authenticate users using this option, but if you want to authorize them as well, you must select Authenticate using custom agent.

When you select the Authenticate using HTML form post option, you must also configure the following sub-options:
Configuring an eTrust SiteMinder Server Instance

Chapter 6: Authentication and Directory Servers

Table 27: eTrust SiteMinder Configuration Options (Continued)

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Target</strong></td>
<td>URL on the external, eTrust-enabled Web server. In the Web agent sign-in page URL, the target appears after <code>&amp;TARGET=$SM$</code>. For example, in the URL shown in “Authenticate using HTML form post” on page 236, the target is: <code>http%3a%2f%2fwebagent%2ejuniper%2enet%2fportal%2findex%2ejs</code> After converting special characters (%3a = colon, %2f = backslash, %2e = period), the final target is: <code>http://webagent.juniper.net/portal/index.jsp</code></td>
</tr>
<tr>
<td><strong>Protocol</strong></td>
<td>Protocol for communication between Infranet Controller and the specified Web agent. Use HTTP for non-secure communication or HTTPS for secure communication. In the Web agent sign-in page URL, the protocol appears first. For example, in the URL shown in “Authenticate using HTML form post” on page 236, the protocol is HTTP.</td>
</tr>
<tr>
<td><strong>Web Agent</strong></td>
<td>Name of the Web agent from which the Infranet Controller is to obtain SMSESSION cookies. An IP address is not allowed for this field. (Specifying the IP address as the Web agent prevents some browsers from accepting cookies.) In the Web agent sign-in page URL, the Web agent appears after the protocol. For example, in the URL shown above in “Authenticate using HTML form post” on page 236, the Web agent is: <code>webagent.juniper.net</code></td>
</tr>
<tr>
<td><strong>Port</strong></td>
<td>Port 80 for HTTP or port 443 for HTTPS.</td>
</tr>
<tr>
<td><strong>Path</strong></td>
<td>Path of the Web agent’s sign-in page. Note that the path must start with a backslash (/) character. In the Web agent sign-in page URL, the path appears after the Web agent. For example, in the URL shown in “Authenticate using HTML form post” on page 236, the path is: <code>/siteminderagent/forms/login.fcc</code></td>
</tr>
<tr>
<td><strong>Parameters</strong></td>
<td>Post parameters to be sent when a user signs in. Common SiteMinder variables that you can use include <code>_USER_</code>, <code>_PASS_</code>, and <code>_TARGET_</code>. These variables are replaced by the username and password entered by the user on the Web agent’s sign-in page and by the value specified in the Target field. These are the default parameters for login.fcc—if you have made customizations, you may need to change these parameters.</td>
</tr>
</tbody>
</table>
Delegate authentication to a standard agent

Choose this option if you want to delegate authentication to a standard agent. When the user accesses the Infranet Controller sign-in page, the Infranet Controller determines the FCC URL associated with the protected resource’s authentication scheme. The Infranet Controller redirects the user to that URL, setting the Infranet Controller sign-in URL as the target. After successfully authenticating with the standard agent, an SMSESSION cookie is set in the user’s browser and he is redirected back to the Infranet Controller. The Infranet Controller then automatically signs in the user and establishes an Infranet Controller session. For information about configuring the authentication scheme, see “Creating a SiteMinder Authentication Scheme for the Infranet Controller” on page 226.

NOTE:

- You must enable the Automatic Sign-In option in order to use this feature.
- If you enable this option and a user already has a valid SMSESSION cookie when he tries to access a resource, the Infranet Controller tries to automatically sign in using the existing SMSESSION cookie. If the cookie is invalid, the Infranet Controller clears the SMSESSION cookie and corresponding Infranet Controller cookies and presents the user with a “timeout” page. The Infranet Controller successfully delegates authentication when the user clicks the “sign back in” option.
- If you select this option, your authentication scheme must have an associated FCC URL.

Table 28: eTrust SiteMinder Advanced Configuration Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Poll Interval</td>
<td>Enter the interval at which Infranet Controller polls the SiteMinder policy server to check for a new key.</td>
</tr>
<tr>
<td>Max. Connections</td>
<td>Controls the maximum number of simultaneous connections that the Infranet Controller is allowed to make to the policy server. The default setting is 20.</td>
</tr>
<tr>
<td>Max. Requests/Connection</td>
<td>Controls the maximum number of requests that the policy server connection handles before the Infranet Controller ends the connection. If necessary, tune to increase performance. The default setting is 1000.</td>
</tr>
<tr>
<td>Idle Timeout</td>
<td>Controls the maximum number of minutes a connection to the policy server may remain idle (the connection is not handling requests) before the Infranet Controller ends the connection. The default setting of “none” indicates no time limit.</td>
</tr>
</tbody>
</table>
Authorize while Authenticating

Specifies that the Infranet Controller should look up user attributes on the policy server immediately after authentication to determine if the user is truly authenticated. For example, if your eTrust server authenticates users based on an LDAP server setting, you can select this option to indicate that the Infranet Controller should authenticate users through the eTrust server and then authorize them through the LDAP server before granting them access. If the user fails authentication or authorization, he is redirected to the page configured on the policy server.

Note:

- If you do not select this option and you have authorization options setup through the Policy Users > Exclude tab of the policy server configuration utility, a user whom you have denied access may successfully authenticate into the Infranet Controller. Not until the user tries to access a protected resource does the Infranet Controller check his authorization rights and deny him access.
- The Infranet Controller sends the same resource to the policy server for authorization as for authentication.
- This option is not supported with the Authenticate using HTML form post option described in “Authenticate using HTML form post” on page 236 or the Automatic sign-in option described in “Automatic Sign-In” on page 235.

Enable Session Grace Period, Validate cookie every N seconds

You can eliminate the overhead of verifying a user’s SMSESSION cookie each time the user requests the same resource by indicating that the Infranet Controller should consider the cookie valid for a certain period of time. During that period, the Infranet Controller assumes that its cached cookie is valid rather than re-validating it against the policy server. If you do not select this option, the Infranet Controller checks the user’s SMSESSION cookie on each request. Note that the value entered here does not affect session or idle timeout checking.

Ignore Query Data

By default, when a user requests a resource, the Infranet Controller sends the entire URL for that resource to the policy server (including the query parameter, if present). For example, the Infranet Controller may send the following URL to the policy server: http://foo/bar?param=value. (Query data appears after the ? character in the URL. Within this URL, param=value represents the query parameter.)

The Infranet Controller then caches the result of the authorization request for 10 minutes, including the query parameter. If the user then requests the same resource that is specified in the cached URL, the request fails since the query portion of the cached URL does not match the new request. The Infranet Controller then has to re-contact the policy server to make a request that includes the new query parameter.

If you select the Ignore Query Data option, the Infranet Controller does not cache the query parameter in its URLs. Therefore, if a user requests the same resource as is specified in the cached URL, the request should not fail. For example, if you enable the Ignore Query Data option, both of the following URLs are considered the same resource:

http://foo/bar?param=value1
http://foo/bar?param=value2

Enabling this option may improve performance.

Accounting Port

The value entered in this field must match the accounting port value entered through the Policy Server Management Console. By default, this field matches the policy server’s default setting of 44441.
Using SiteMinder User Attributes for Infranet Controller Role Mapping

After you create user attributes on a SiteMinder policy server (see “Creating SiteMinder User Attributes for Infranet Controller Role Mapping” on page 229), you can use them in role mapping rules for a realm that uses the SiteMinder policy server.

To use SiteMinder user attributes for Infranet Controller role mapping:

1. In the admin console, choose **Administrators > Admin Realms** or **Users > User Realms**.

2. On the **General** tab of the **Authentication Realms** page for the Infranet Controller realm that uses the SiteMinder policy server, choose **Same as Above** from the **Directory/Attribute** list. (For instructions, see “Creating an Authentication Realm” on page 246.)

3. On the Infranet Controller **Role Mapping** tab, create a rule based on Infranet Controller user attributes that references a SiteMinder user attribute cookie.

   For example, to reference a SiteMinder user attribute cookie named `department`, add `department` to the list of Infranet Controller user attributes on the Infranet Controller **Role Mapping** tab. Then specify a value for the SiteMinder user attribute cookie, such as `sales`. For instructions, see “Creating Role Mapping Rules” on page 249.

   You can also use the following syntax to reference a SiteMinder user attribute cookie in a custom expression for a role mapping rule:

   ```plaintext
   userAttr.<cookie-name>
   ```

   For example:

   ```plaintext
   userAttr.department = ("sales" and "eng")
   ```
Defining a SiteMinder Realm for Automatic Sign-In

SiteMinder Automatic Sign In requires a realm whose authentication server is the SiteMinder server. If you perform an upgrade and you have already defined the Automatic Sign In realm that does not specify the SiteMinder server for authentication, and you have configured the SiteMinder server:

- The realms do not appear in the SiteMinder realm list under SiteMinder authentication settings in the admin console.
- The upgrade process creates a new realm called eTrust-Auto-Login-Realm which is based on your existing realm, but which configures the SiteMinder server as its authentication server.

To configure the SiteMinder realm on a new installation:

2. Choose SiteMinder from the New list and click New Server.
3. Specify the settings you want, as described in “Defining an eTrust SiteMinder Server Instance” on page 232.
4. Click Save Changes.
5. Configure the realm, as described in “Creating an Authentication Realm” on page 246, and select the SiteMinder server as the authentication server.
7. Choose the SiteMinder server you defined previously.
8. Under SiteMinder authentication settings, select the Automatic Sign In check box.
9. Choose the realm you just configured from the user authentication realm list.
10. Click Save Changes.

NOTE: The user authentication realm list on the SiteMinder server page only displays realms that are configured for SiteMinder. If you have not configured any SiteMinder realms, the drop down menu is empty.

Debugging SiteMinder and Infranet Controller Issues

At some point, you may encounter problems configuring the eTrust SiteMinder server interactions with the Infranet Controller. You can use a number of debugging tools to identify and resolve problems:

- Review the Infranet Controller log file. The Infranet Controller tracks failures of cookie validation and key rollovers.
- Review the Policy Server Authentication log files.
■ Review the Standard Web Agent log file if you have selected the Authentication using HTLM Form POST option.

■ Confirm that the Infranet Controller contains the proper suffix that you defined in the Cookie Domain field. If the Infranet Controller is not properly addressed, the browser may not forward the correct SMSESSION cookie to the Infranet Controller and you may not be able to sign in. You must enter the Infranet Controller’s FQDN on the browser, not the Infranet Controller IP address, otherwise, your login fails.

■ Confirm that the Infranet Controller system time is synchronized with the SiteMinder server’s system time. If the two system times are too divergent, the timeout settings may not function correctly, rejecting your attempts to sign in.

■ In the SiteMinder server, confirm that you have defined the proper Session Timeout options max timeout and idle in the Siteminder Realm dialog.

---

### Configuring a MAC Address Authentication Server for Unmanageable Devices

1. In the admin console, choose Authentication > Auth. Servers.

2. To create a new MAC address authentication server instance on the Infranet Controller, select MAC Address Authentication from the New list, and then click New Server.

3. Specify a name to identify the server instance.

4. Do one of the following:
   - Enter MAC address entries in the MAC Address field in the format 00:11:85:bb:8c:66, select Allow or Deny and click Add. To enter wildcards, use the format 00:11:22*:.*:* (a single asterisk represents two characters).

     In the Attributes field, enter a value pair to associate the MAC address with a particular group or organization. For example: dept=eng represents that this MAC address belongs to engineering. When you create the MAC Address Realm you can create a custom expression to assign the MAC address to a specific role.


5. In the Attributes field, enter attributes to associate a particular

6. Click Save Changes.

7. After you have configured the MAC address authentication server, you must add the server instance to a MAC address realm. For more information see “Creating a MAC Address Realm” on page 169.
You create authentication realms to permit clients to request authentication from the Infranet Controller. The Infranet Controller supports different types of agent access: UAC clients, the Odyssey Access Client, the Java agent or endpoints using agentless access, third-party 802.1X supplicants, and 802.1X phones.

Depending on the client, and the authentication server you are using, different authentication protocols can be paired with different realms.

You pair realms with the appropriate authentication protocol sets when you configure sign-in policies. See “Authentication Realms and Protocols in User Sign-in Policies” on page 260.

An authentication realm specifies the conditions that users must meet in order to sign into the Infranet Controller. A realm consists of a grouping of authentication resources, including:

- **An authentication server**, which verifies that the user is who he claims to be. The Infranet Controller forwards credentials that a user submits to an authentication server. For more information, see “Authentication and Directory Servers” on page 173.

- **A directory server**, which is an LDAP server that provides user and group information to the Infranet Controller that the Infranet Controller uses to map users to one or more user roles. For more information, see “Authentication and Directory Servers” on page 173.

- **An authentication policy**, which specifies realm security requirements that need to be met before the Infranet Controller submits a user’s credentials to an authentication server for verification. For more information, see “Defining Authentication Policies” on page 248.

- **Role mapping rules**, which are conditions a user must meet in order for the Infranet Controller to map the user to one or more user roles. These conditions are based on either user information returned by the realm’s directory server or the user’s username. For more information, see “Creating Role Mapping Rules” on page 249.

This section contains the following information about authentication realms:

- “Creating an Authentication Realm” on page 246
- “Defining Authentication Policies” on page 248
Creating an Authentication Realm

To create an authentication realm:

1. In the admin console, choose Administrators > Admin Realms or Users > User Realms.

2. On the respective Authentication Realms page, click New. Or, select a realm and click Duplicate to base your realm on an existing realm.

3. Enter a name to label this realm and (optionally) a description.

4. If you are copying an existing realm, click Duplicate. Then, if you want to modify any of its settings, click the realm’s name to enter into edit mode.

5. Select When editing, start on the Role Mapping page if you want the Role Mapping tab to be selected when you open the realm for editing.

6. Under Servers, specify:
   - An authentication server to use for authenticating users who sign in to this realm.
   - A directory/attribute server to use for retrieving user attribute and group information for role mapping rules and resource policies. (optional)
   - A RADIUS accounting server to use to track when a user signs in and out of the Infranet Controller (optional).

**NOTE:**

- The Infranet Controller supports RADIUS proxy for both inner and outer authentication. RADIUS proxy allows you to use an external RADIUS server for authentication. If the authentication server for a realm is a RADIUS server, three option buttons: Proxy RADIUS Inner Authentication, Proxy RADIUS Outer Authentication, and Do not proxy are visible. If the authentication server is not a RADIUS server, the proxy check boxes are hidden. See “Using RADIUS Proxy” on page 138.

- When RADIUS proxy is used, realm or role restrictions cannot be enforced. Host Checker policies, Source IP restrictions, and any other limits that have been assigned will be bypassed. RADIUS proxy should be used only if no restrictions have been applied.
Chapter 7: Authentication Realms

7. If you have previously selected a RADIUS server for Authentication, the RADIUS Proxy option buttons appear. Select Proxy Outer Authentication or Proxy Inner Authentication to allow the Infranet Controller to proxy EAP authentication methods. Select Do not proxy if you do not want to use RADIUS proxy.

8. If you want to use dynamic policy evaluation for this realm (as explained in “Dynamic Policy Evaluation” on page 105), select Dynamic policy evaluation to enable an automatic timer for dynamic policy evaluation of this realm’s authentication policy, role mapping rules, and role restrictions. Then:
   a. Use the Refresh interval option to specify how often you want the Infranet Controller to perform an automatic policy evaluation of all currently signed-in realm users. Specify the number of minutes (5 to 1440).
   b. Select Refresh roles to also refresh the roles of all users in this realm. (This option does not control the scope of the Refresh Now button.)
   c. Select Refresh resource policies to also refresh the resource policies (not including Meeting and Email Client) for all users in this realm. (This option does not control the scope of the Refresh Now button.)
   d. Click Refresh Now to manually evaluate the realm’s authentication policy, role mapping rules, role restrictions, user roles, and resource policies of all currently signed-in realm users. Use this button if you make changes to an authentication policy, role mapping rules, role restrictions, or resource policies and you want to immediately refresh the roles of this realm’s users.

NOTE: If you select Dynamic policy evaluation and you do not select Refresh roles and Refresh resource policies, the Infranet Controller evaluates the realm’s authentication policy, role mapping rules, and role restrictions only.

NOTE: Since dynamic policy evaluation can potentially impact system performance, keep these guidelines in mind:

- Since automatic (timer-based) refreshing of user roles and resource policies can affect system performance, you can improve performance by disabling either or both of the Refresh roles and Refresh resource policies options to reduce the scope of the refresh.
- To improve performance, set the Refresh interval option to a longer time period.
- Use the Refresh Now button at times when users may not be affected.
9. Click **Save Changes** to create the realm on the Infranet Controller. The **General**, **Authentication Policy**, and **Role Mapping** tabs for the authentication realm appear.

10. Perform the next configuration steps:
   a. Configure one or more role mapping rules as described in “Creating Role Mapping Rules” on page 249.
   b. Configure an authentication policy for the realm as described in “Defining Authentication Policies” on page 248.

11. After you have configured the authentication realm, go to **Authentication > Signing In > Sign-in Policies** and add the realm to a sign-in policy, and associate the realm with an authentication protocol set. See “Authentication Realms and Protocols in User Sign-in Policies” on page 260.

---

**Defining Authentication Policies**

An *authentication policy* is a set of rules that controls one aspect of access management—whether or not to present a realm’s sign-in page to a user. An authentication policy is part of an authentication realm’s configuration, specifying rules for the Infranet Controller to consider before presenting a sign-in page to a user. If a user meets the requirements specified by the realm’s authentication policy, then the Infranet Controller presents the corresponding sign-in page to the user and then forwards the user's credentials to the appropriate authentication server. If this server successfully authenticates the user, then the Infranet Controller moves on to the role evaluation process.

To specify an authentication realm policy:

1. In the admin console, choose **Administrators > Admin Realms** or **Users > User Realms**.
2. On the respective **Authentication Realms** page, click a realm and then click the **Authentication Policy** tab.
3. On the **Authentication Policy** page, configure one or more of the access management options described in the following sections:
   - “Specifying Source IP Access Restrictions” on page 108
   - “Specifying Browser Access Restrictions” on page 110
   - “Specifying Certificate Access Restrictions” on page 112
   - “Specifying Password Access Restrictions” on page 113
   - “Specifying Host Checker Access Restrictions” on page 114
   - “Specifying Session Limits” on page 114
   - “Using Kerberos Single Sign-On with the Infranet Controller” on page 185
Creating Role Mapping Rules

Role mapping rules are conditions a user must meet in order for the Infranet Controller to map the user to one or more user roles. These conditions are based on either user information returned by the realm's directory server or the user's username. You must specify role mapping directives in the following format:

If the specified condition is | is not true, then map the user to the selected roles.

You create a role mapping rule on the Role Mapping tab of an authentication realm. (For administrators, you create role mapping rules on the Administrators > Admin Realms > [Realm] > Role Mapping tab. For users, you create role mapping rules on the Users > User Realms > [Realm] > Role Mapping tab.) When you click New Rule on this tab, the Role Mapping Rule page appears with an inline editor for defining the rule. This editor leads you through the three steps of creating a rule:

1. Specify the type of condition on which to base the rule. Options include:
   - Username
   - Certificate or certificate attribute
   - Custom expressions

2. Specify the condition to evaluate, which consists of:
   a. One or more usernames, certificate attributes, or expressions depending on the type of condition selected in step 1. Outer proxy cannot be used for the realm if a role mapping rule based on user names is created, as the Infranet Controller cannot see the user name, and a session cannot be created.
   b. To what the value(s) should equate, which may include a list of usernames, client-side certificate values (static or compared to LDAP attributes), or pre-defined custom expressions. If you are using proxy RADIUS for outer authentication, you cannot create a role mapping rule based on username.

3. Specify the roles to assign to the authenticated user.

The Infranet Controller compiles a list of eligible roles to which a user may be mapped, which are roles specified by the role mapping rules to which the user conforms. Next, the Infranet Controller evaluates the definition for each role to determine if the user complies with any role restrictions. The Infranet Controller uses this information to compile a list of valid roles, which are roles for which the user meets any additional requirements. Finally, the Infranet Controller either performs a permissive merge of the valid roles or presents a list of valid roles to the user, depending on the configuration specified on the realm’s Role Mapping tab.

For more information about roles, see “User Roles” on page 117. For more information about specifying role mapping rules, see “Specifying Role Mapping Rules for an Authentication Realm” on page 250.
**Specifying Role Mapping Rules for an Authentication Realm**

When creating a new rule that uses LDAP or SiteMinder user attributes, LDAP group information, or custom expressions, you must use the server catalog. For information about this catalog, see “Using the LDAP Server Catalog” on page 252.

To specify role mapping rules for an authentication realm:

1. In the admin console, choose **Administrators > Admin Realms** or **Users > User Realms**.

2. On the respective **Authentication Realms** page, select a realm and then click the **Role Mapping** tab.

3. Click **New Rule** to access the **Role Mapping Rule** page. This page provides an inline editor for defining the rule.

4. In the **Rule based on** list, choose one of the following:

   - **Username**—*Username* is the Infranet Controller username entered on the sign-in page. Choose this option if you want to map users to roles based on their Infranet Controller usernames. If this is a RADIUS realm, and you are using RADIUS proxy for outer authentication, you cannot configure a role mapping rule with a username.

   - **User attribute**—*User attribute* is a user attribute from a RADIUS, LDAP, or SiteMinder server. Choose this option if you want to map users to roles based on an attribute from the corresponding server. This type of rule is available only for realms that use a RADIUS server for the authentication server, or that use an LDAP or SiteMinder server for either the authentication server or directory server. After choosing the **User attribute** option, click **Update** to display the **Attribute** list and the **Attributes** button. Click the **Attributes** button to display the server catalog.

   - To add SiteMinder user attributes, enter the SiteMinder user attribute cookie name in the **Attribute** field in the server catalog, and then click **Add Attribute**. When you are finished adding cookie names, click **OK**. The Infranet Controller displays the names of the SiteMinder user attribute cookies in the **Attribute** list on the **Role Mapping Rule** page.

   - For information on how to use the server catalog to add LDAP user attributes, see “Using the LDAP Server Catalog” on page 252.

   - **Certificate or Certificate attribute**—*Certificate or Certificate attribute* is an attribute supported by the users’ client-side certificate. Choose this option if you want to map users to roles based on certificate attributes. The **Certificate** option is available for all realms; the **Certificate attribute** option is available only for realms that use LDAP for the authentication or directory server. After choosing this option, click **Update** to display the **Attribute** text box.
Creating Role Mapping Rules

Chapter 7: Authentication Realms

- **Group membership**—*Group membership* is group information from an LDAP or native Active Directory server that you add to the server catalog **Groups** tab. Choose this option if you want to map users to roles based on either LDAP or Active Directory group information. This type of rule is available only for realms that use an LDAP server for either the authentication server or directory server or that use an Active Directory server for authentication. (Note that you cannot specify an Active Directory server as an authorization server for a realm.)

- **Custom Expressions**—*Custom Expressions* is one or more custom expressions that you define in the server catalog. Choose this option if you want to map users to roles based on custom expressions. This type of rule is available for all realms. After choosing this option, click **Update** to display the **Expressions** lists. Click the **Expressions** button to display the **Expressions** tab of the server catalog.

5. Under **Rule**, specify the condition to evaluate, which corresponds to the type of rule you select and consists of:

   a. Specifying one or more usernames, SiteMinder user attribute cookie names, RADIUS or LDAP user attributes, certificate attributes, LDAP groups, or custom expressions.

   b. Specifying to what the value(s) should equate, which may include a list of Infranet Controller usernames, user attribute values from a RADIUS, SiteMinder, or LDAP server, client-side certificate values (static or LDAP attribute values), LDAP groups, or custom expressions.

   For example, you can choose a SiteMinder user attribute cookie named **department** from the **Attribute** list, choose **is** from the operator list, and then enter "sales" and "eng" in the text box.

   Or, you can enter a custom expression rule that references the SiteMinder user attribute cookie named **department**:

   `userAttr.department = ("sales" and "eng")`

6. Under **...then assign these roles**:

   a. Specify the roles to assign to the authenticated user by adding roles to the **Selected Roles** list.

   b. Check **Stop processing rules when this rule matches** if you want the Infranet Controller to stop evaluating role mapping rules if the user meets the conditions specified for this rule.

7. Click **Save Changes** to create the rule on the **Role Mapping** tab. When you are finished creating rules:

   - Make sure to order them in the order in which you want the Infranet Controller to evaluate them. This task is particularly important when you want to stop processing role mapping rules upon a match.

   - Specify whether or not you want to merge settings for all assigned roles. See “Permissive Merge Guidelines” on page 119.
Using the LDAP Server Catalog

The LDAP server catalog is a secondary window through which you specify additional LDAP information for the Infranet Controller to use when mapping users to roles, including:

- **Attributes**—The Server Catalog Attributes tab shows a list of common LDAP attributes, such as `cn`, `uid`, `uniqueMember`, and `memberof`. This tab is accessible only when accessing the Server Catalog of an LDAP server. You can use this tab to manage an LDAP server's attributes by adding custom values to and deleting values from its Infranet Controller server catalog. Note that the Infranet Controller maintains a local copy of the LDAP server's values; attributes are not added to or deleted from your LDAP server's dictionary.

- **Groups**—The Server Catalog Groups tab provides a mechanism to easily retrieve group information from an LDAP server and add it to the server's Infranet Controller server catalog. You specify the BaseDN of your groups and optionally a filter to begin the search. If you do not know the exact container of your groups, you can specify the domain root as the BaseDN, such as `dc=juniper, dc=com`. The search page returns a list of groups from your server, from which you can choose groups to enter into the Groups list.

  You can also use the Groups tab to specify groups. You must specify the Fully Qualified Distinguished Name (FQDN) of a group, such as `cn=GoodManagers, ou=HQ, ou=Juniper, o=com, c=US`, but you can assign a label for this group that appears in the Groups list. Note that this tab is accessible only when accessing the Server Catalog of an LDAP server.

- **Expressions**—The Server Catalog Expressions tab provides a mechanism to write custom expressions for the role mapping rule. For more information about custom expressions, see “Writing Custom Expressions” on page 517.

To display the LDAP server catalog:

1. After choosing the **User attribute** option on the Role Mapping Rule page (see “Specifying Role Mapping Rules for an Authentication Realm” on page 250), click **Update** to display the Attribute list and the Attributes button.

2. Click the Attributes button to display the LDAP server catalog. (You can also click Groups after choosing the Group membership option, or click Expressions after choosing the Custom Expressions option.)
Creating Role Mapping Rules
Customizing User Realm UI Views

You can use customization options on the User Authentication Realms page to quickly view the settings that are associated with a specific realm or set of realms. For instance, you can view the role-mapping rules that you have associated with all your user realms. Additionally, you can use these customized views to easily link to the authentication policies, servers, role-mapping rules, and roles associated with a user realms.
To view a sub-set of data on the **User Authentication Realms** page:

1. Navigate to **Users > User Realms**.

2. Select one of the following options from the **View** menu:
   - **Overview**—Displays the authentication servers and dynamic policy evaluation settings that you have set for the specified user realms. You may also use this setting to link to the specified server configuration pages.
   - **Authentication Policy**—Displays Host Checker restrictions that you have enabled for the specified user realms. You may also use this setting to link to the specified Host Checker configuration pages.
   - **Role Mapping**—Displays rule conditions and corresponding role assignments that you have enabled for the specified user realms. You may also use this setting to link to the specified rule conditions and role assignments configuration pages.
   - **Servers**—Displays authentication server names and corresponding types that you have enabled for the specified user realms. You may also use this setting to link to the specified server configuration pages.
   - **Roles**—Displays role assignments and corresponding permissive merge settings that you have enabled for the specified user realms.

3. Select one of the following options from the **for** list:
   - **All realms**—Displays the selected settings for all user realms.
   - **Selected realms**—Displays the selected settings for the user realms you choose. If you select this option, select one or more of the check boxes in the **Authentication Realm** list.

4. Click **Update**.
Chapter 8
Sign-In Policies

Sign-in policies define the URLs that users and administrators use to access the Infranet Controller and the sign-in pages that they see. The Infranet Controller has two types of sign-in policies—one for users and one for administrators. When configuring sign-in policies, you associate realms, sign-in pages, and URLs.

To allow users to sign in to the Infranet Controller you add user authentication realms to sign-in policies. You can associate realms with a variety of authentication protocols to accommodate different types of endpoints. For example, Odyssey Access Client, IP phones, and non-Juniper networks 802.1X supplicants can access the Infranet Controller, but each of these endpoints requires different authentication protocols.

NOTE: If you are upgrading to UAC from a version earlier than 2.1, some settings may be affected by the upgrade. Specifically, formerly the option to specify that Odyssey Access Client should download to endpoints was configured in the sign-in policy. This option has been moved to the role configuration page. If you have sign-in policies that specify that Odyssey Access Client should be downloaded to endpoints, the action will no longer be in affect.

After upgrading, go to the roles page and select the checkbox for Install Agent for this role. Depending on your configuration, it may then be necessary to re-assign role-mapping rules to ensure that the Odyssey Access Client is downloaded to the endpoints that you choose. See “Creating an initial Configuration of Odyssey Access Client” on page 44.

You can create multiple user sign-in policies to enable different users to sign into different URLs and pages. When configuring a sign-in policy, you associate it with a sign-in URL, a sign-in page, one or more realms, and an authentication protocol set. Only members of the specified authentication realm(s) may sign in using the URL defined in the policy.

When defining sign-in policies, you may use different host names (such as users1.yourcompany.com and users2.yourcompany.com) or different paths (such as yourcompany.com/users1 and yourcompany.com/users2) to differentiate between URLs.

For Windows machines, you can display different sign-in pages for users based on whether or not you want the endpoint to download Odyssey Access Client.
You specify whether you want the Infranet Controller to install Odyssey Access Client on endpoints at the role level. If you use role mapping to associate roles with specific realms, you can specify which users get Odyssey Access Client installed and which users do not, and the associated message that each group views on the sign-in page.

For example, if you have contract employees with non-company machines onto which you do not want to install Odyssey Access Client, you can create two roles: one that allows agentless access, and the other requiring installation of the Odyssey Access Client. Then, create two associated realms: one for agentless access and one for Odyssey Access Client. Add role mapping rules based on user names to assign the contract employees to the agentless role, and employees to the agent role. When a user attempts to log in, they will be assigned to a role which will either provision agentless access or install Odyssey Access Client.

You can associate the different realms with different sign-in policies and sign-in pages, so users who login to a resource can see a sign-in page based on whether or not they are a regular employee or a contractor.

This section contains the following information about sign-in policies:

- “Task summary: Configuring Sign-In Policies” on page 258
- “Configuring Sign-In Policies” on page 259
- “Configuring Sign-In pages” on page 265

---

**Task summary: Configuring Sign-In Policies**

To configure sign-in policies, you must:

1. Create an authentication realm through one of the **Administrators > Admin Realms** or **Users > User Realms** page of the admin console.

2. (Optional) Modify an existing sign-in page or create a new one using options in the **Authentication > Signing In > Sign-in Pages** page of the admin console.

3. (Optional) Create a new authentication protocol set to associate with the realm. This is only necessary if the realm is required to provide access for a non-UAC supplicant (for example, Microsoft Vista with a Statement of Health Host Checker policy). For users authenticating with Odyssey Access Client, the Java agent or for Macintosh or Linux clients with agentless access use the default 802.1X protocol set.

4. Specify a sign-in policy that associates a realm, sign-in URL, and sign-in page using settings in the **Authentication > Signing In > Sign-in Policies** page of the admin console.

5. If you differentiate between URLs using host names, you must associate each host name with its own certificate or upload a wildcard certificate into the Infranet Controller using options in the **System > Configuration > Certificates > Device Certificates** page.
### Configuring Sign-In Policies

*Sign-in policies* define the URLs that users and administrators can use to access the Infranet Controller, as explained in “Sign-In Policies” on page 257.

User sign-in policies also determine the realm(s) that users and administrators can access.

Depending on whether a sign-in policy is for endpoints (users) or administrators, the configuration options are different. For users, different authentication protocol sets can be configured, and realm selection is based on the authentication method that is associated with the realm.

This section contains the following information about sign-in policies:

- “Defining Administrator Sign-In Policies” on page 259
- “Authentication Realms and Protocols in User Sign-in Policies” on page 260
- “Defining User Sign-in Policies” on page 263
- “Specifying the Order in Which Sign-In Policies are Evaluated” on page 264
- “Enabling and Disabling Sign-In Policies” on page 264

### Defining Administrator Sign-In Policies

To create or configure administrator sign-in policies:

1. In the admin console, choose **Authentication > Signing In > Sign-in Policies**.

2. To create a new sign-in policy, click **New URL**. Or, to edit an existing policy, click a URL in the **Administrator URLs** or **User URLs** column.

3. To create an administrator sign-in policy, select the **Administrators** option button at the top of the page. By default, the **Users** option button is selected.

4. In the **Sign-in URL** field, enter the URL that you want to associate with the policy. Use the format `<host>/<path>` where `<host>` is the host name of the Infranet Controller, and `<path>` is any string you want users to enter. For example: `users1.yourcompany.com/ic`. To specify multiple hosts, use the * wildcard character. For instance:

   - To specify that all administrator URLs should use the sign-in page, enter `*/admin`.

   **NOTE:** You may only use wildcard characters (*) in the beginning of the host name portion of the URL. The Infranet Controller does not recognize wildcards in the URL path.

5. Enter a **Description** for the policy (optional).
6. From the Sign-in Page list, select the page that you want to associate with the policy. You may select the default page that comes with the Infranet Controller, a variation of the standard sign-in page, or a custom page that you create using the customizable UI feature. For more information, see “Configuring Standard Sign-In Pages” on page 266.

7. For administrator sign-in policies, under Authentication realm, specify which realm(s) map to the policy, and how users and administrators should pick from among realms. If you select:

- **User types the realm name**—The Infranet Controller maps the sign-in policy to all authentication realms, but does not provide a list of realms from which the administrator can choose. Instead, the administrator must manually enter the realm name into the sign-in page.

- **User picks from a list of authentication realms**—The Infranet Controller only maps the sign-in policy to the authentication realms that you choose. The Infranet Controller presents this list of realms when the administrator signs-in to the Infranet Controller and allows a realm to be chosen from the list. (Note that the Infranet Controller does not display a drop-down list of authentication realms if the URL is only mapped to one realm. Instead, only the realm you specify is displayed).

8. Use the Add button to add Available realms to the Selected realms box.

9. Click Save Changes.

**Authentication Realms and Protocols in User Sign-in Policies**

There are many types of endpoints that can request authentication through the Infranet Controller including UAC clients, third-party 802.1X supplicants (including 802.1X IP phones), switches, and endpoints that request authentication with agentless access.

A UAC client is software that can use Juniper Networks JUAC protocol. UAC clients include Odyssey Access Client and the Java agent. By default, the Infranet Controller can communicate with UAC clients and endpoints with agentless access. To accommodate other types of endpoints, it may be necessary to create authentication protocol sets within sign-in policies.

When you add a realm in a sign-in policy, you select an authentication protocol set to be used with that realm. There are two default authentication protocol sets. For UAC clients, you can use the default 802.1X authentication protocol set. For 802.1X IP phones, you can use the default 802.1X-Phones protocol set.

Third-party 802.1X supplicants cannot use the preconfigured 802.1X protocol set that is used by default with UAC clients. For example, some switches can request authentication using CHAP or EAP-MD5-Challenge. You must define an authentication protocol set for these authentication requests.
To define an endpoint’s authentication method, you add authentication realms to sign-in policies. You configure authentication protocol sets as required, based on authentication methods that are compatible with the authentication server that you are using. The Infranet Controller maps the sign-in policy to the authentication realms that you choose. Users who sign in using the URL that you provide have access only to those realms that you have specified.

For non-UAC clients, you must select the protocols that the client and the authentication server are compatible with. See Table 29, “RADIUS Sub-Protocols and Compatible Authentication Servers” on page 261 for details of what authentication protocols are compatible with different authentication servers.

<table>
<thead>
<tr>
<th>Authentication Servers</th>
<th>Protocols</th>
<th>Certificate</th>
<th>Local</th>
<th>LDAP</th>
<th>Active Directory</th>
<th>ACE</th>
<th>Mac Auth</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>EAP-GTC</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PAP</td>
<td>Y Y Y Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CHAP, EAP-MD5-Challenge</td>
<td>Y Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MS-CHAP</td>
<td>Y Y Y Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MS-CHAP-V2, EAP-MS-CHAP-V2</td>
<td>Y Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EAP-TLS</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Mac-based auth</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EAP-JUAC</td>
<td>Y Y Y Y Y Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The decision of what realm(s) are available to the user within a sign-in policy containing multiple realms is based on two factors. First, the order of realms in the list is considered. Realms at the top of the list are attempted first. Second, the authentication protocol set that you choose must be compatible with the client or supplicant.

To determine a compatible realm, the Infranet Controller looks for a RADIUS sub-protocol that is compatible with the client or supplicant’s available protocols, and the Infranet Controller automatically selects compatible realms. When the endpoint is using a UAC client, the Infranet Controller presents a list of realms. Any realm that has both outer and inner protocols matching the outer and inner protocols on the client is considered compatible.
Protocol compatibility does not guarantee successful authentication. For example, CHAP and EAP-MD-5 challenge sign-in succeeds only if the stored password is retrievable as cleartext. In addition, if the client or supplicant is configured with a non-JUAC protocol (for example, the Windows Vista supplicant), the Infranet Controller searches for a realm without TNC Host Checker restrictions, browser restrictions, or certificate restrictions.

**NOTE:** If you are configuring a realm for a Windows client, with a Statement of Health Host Checker policy, you must use an authentication protocol set with the EAP-SOH protocol. When you select EAP-SOH in an authentication protocol set, EAP-SOH is always offered first, regardless of protocol ordering.

If an endpoint is using UAC client software, the Infranet Controller presents the list of realms to the user or administrator when the user signs in to the Infranet Controller and allows the user to choose a realm from the list. The Infranet Controller does not display a list of authentication realms if the URL is only mapped to one realm. Instead, it automatically uses the realm you specify.

For endpoints using a non-UAC client, you can select the **User may specify the realm name as a username suffix** check box. When the user provides a user name with a suffix in the format user@realm, the suffix determines the realm assignment. If you do not select this option, the endpoint is assigned to the first realm in the list whose authentication server is a match with the endpoint’s software. For example, if the endpoint’s software is configured for tokens (EAP-Generic Token Card), and if the sign in policy permits EAP-GTC, the endpoint will be assigned the first realm in the list whose authentication server supports tokens.

When an 802.1X IP phone connects to the Infranet Controller through a realm with the 802.1X-Phone protocol set selected, the device is automatically directed to the proper realm for authentication based on the compatible protocol.

If you are using inner or outer RADIUS proxy with a selected realm, routing with respect to authentication protocols is different. The Infranet Controller forwards all traffic to a proxy target, which rejects protocols it does not support. With an outer proxy realm, the Infranet Controller ignores the authentication protocol set. For an inner proxy realm, the authentication protocol set directs the Infranet Controller as it negotiates the outer protocol (EAP-PEAP or EAP-TTLS) but does not affect the inner protocol. See “Authentication Protocols on the Infranet Controller” on page 133.
Defining User Sign-In Policies

To create or configure user sign-in policies:

1. In the admin console, choose Authentication > Signing In > Sign-in Policies.

2. To create a new sign-in policy, click New URL. Or, to edit an existing policy, click a URL in the Administrator URLs or User URLs column.

3. In the Sign-in URL field, enter the URL that you want to associate with the policy. Use the format `<host>/path` where `<host>` is the host name of the Infranet Controller, and `<path>` is any string you want users to enter. For example: users1.yourcompany.com/ic. To specify multiple hosts, use the * wildcard character. For example, to specify that all end-user URLs should use the sign-in page, enter */.

4. Enter a Description for the policy (optional).

5. From the Sign-in Page list, select the sign-in page that you want to associate with the policy. You may select the default page that comes with the Infranet Controller, a variation of the standard sign-in page, or a custom page that you create using the customizable UI feature. For more information, see “Configuring Standard Sign-In Pages” on page 266.

Table 30: Before Beginning

<table>
<thead>
<tr>
<th>Topic</th>
<th>Detail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wildcard characters in host name</td>
<td>You may only use wildcard characters (*) in the beginning of the host name portion of the URL. The Infranet Controller does not recognize Wildcard characters in the URL path.</td>
</tr>
<tr>
<td>Odyssey Access Client sign-in</td>
<td>Endpoints that will use Odyssey Access Client with Layer 3 access the sign-in page for the initial login. After clients are assigned a role, Odyssey Access Client automatically downloads and subsequent authentication requests are performed through the client.</td>
</tr>
<tr>
<td>Outer proxy realms</td>
<td>If you are configuring an outer proxy realm, you do not have to specify an authentication protocol set, and Not Applicable should be used as the authentication protocol.</td>
</tr>
<tr>
<td>Anonymous authentication servers</td>
<td>If you allow a user with Odyssey Access Client to select from multiple realms, and one of those realms uses an anonymous authentication server, the Infranet Controller does not display that realm in the drop-down list. To effectively map your sign-in policy to an anonymous realm, you must add only that realm to the Authentication realm list.</td>
</tr>
<tr>
<td>Username suffixes</td>
<td>By default, the <strong>Username suffix</strong> check box is not selected. If you choose this option, non-UAC endpoints access the Infranet Controller by entering their credentials in the format user@realm.</td>
</tr>
<tr>
<td>Proxy realm sign-in</td>
<td>If you configure a sign-in policy with multiple realms, and one of the realms is a proxy realm, the user must append a suffix to the username to access the proxy realm.</td>
</tr>
</tbody>
</table>

To create or configure user sign-in policies:
6. Under **Authentication realm**, specify the realm(s) that should be mapped to the sign-in policy. Under **Available realms**, select from the pull-down menu. The Infranet Controller only maps the sign-in policy to the authentication realms that you add.

7. Under **Authentication protocol set**, select one of the authentication protocol sets that you have previously configured, or if endpoints will connect with a UAC agent, select the default **802.1X** protocol set. The protocol set used with a realm must be compatible with the authentication server that is associated with the realm. Table 29, “RADIUS Sub-Protocols and Compatible Authentication Servers” on page 261 displays a matrix of RADIUS sub-protocols that are compatible with different authentication servers.

8. Click the **Add** button to add the new realm and authentication protocol pair.

9. Select the **User may specify the realm name as a username suffix** check box to allow non-UAC endpoints to access the Infranet Controller by entering their credentials in the format `user@realm`.

10. Select the **Remove realm suffix before passing to authentication server** check box if users will enter their credentials with a suffix to send the user name without the suffix. Most authentication servers are not compatible with a realm suffix or decorated user name.

11. Click **Save Changes**.

**Enabling and Disabling Sign-In Policies**

To enable and disable sign-in policies:

1. In the admin console, choose **Authentication > Signing In > Sign-in Policies**.

2. To enable or disable:

   - **An individual policy**—Select the check box next to the policy that you want to change, and then click **Enable** or **Disable**.

   - **All user policies**—Select or deselect the **Restrict access to administrators only** check box at the top of the page.

3. Click **Save Changes**.

**Specifying the Order in Which Sign-In Policies are Evaluated**

The Infranet Controller evaluates sign-in policies in the same order that you list them on the **Sign-in Policies** page. When it finds a URL that matches exactly, it stops evaluating and presents the appropriate sign-in page to the administrator or user. For example, you may define two administrator sign-in policies with two different URLs:

- The first policy uses the URL `*/admin` and maps to the default administrator sign-in page.
The second policy uses the URL `yourcompany.com/admin` and maps to a custom administrator sign-in page.

If you list the policies in this order on the Sign-in Policies page, the Infranet Controller never evaluates or uses the second policy because the first URL encompasses the second. Even if an administrator signs in using the `yourcompany.com/admin` URL, the Infranet Controller displays the default administrator sign-in page. If you list the policies in the opposite order, however, the Infranet Controller displays the custom administrator sign-in page to those administrators who access the Infranet Controller using the `yourcompany.com/admin` URL.

Note that the Infranet Controller only accepts wildcard characters in the host name section of the URL and matches URLs based on the exact path. For example, you may define two administrator sign-in policies with two different URL paths:

- The first policy uses the URL `*/marketing` and maps to a custom sign-in page for the entire Marketing Department.
- The second policy uses the URL `*/marketing/joe` and maps to a custom sign-in page designed exclusively for Joe in the Marketing Department.

If you list the policies in this order on the Sign-in Policies page, the Infranet Controller displays Joe's custom sign-in page to him when he uses the `yourcompany.com/marketing/joe` URL to access the Infranet Controller. He does not see the Marketing sign-in page, even though it is listed and evaluated first, because the path portion of his URL does not exactly match the URL defined in the first policy.

To change the order in which administrator sign-in policies are evaluated:

1. In the admin console, choose Authentication > Signing In > Sign-in Policies.
2. Select a sign-in policy in the Administrator URLs or User URLs list.
3. Click the up and down arrows to change the selected policy’s placement in the list.
4. Click Save Changes.

---

### Configuring Sign-In pages

A sign-in page defines the customized properties in the end-user’s welcome page such as the welcome text, help text, logo, header, and footer. The Infranet Controller allows you to create two types of sign-in pages to present to users and administrators:

- **Standard sign-in pages**—Standard sign-in pages are produced by Juniper and are included with all versions of the Infranet Controller. You can modify standard sign-in pages through the Authentication > Signing In > Sign-in Pages tab of the admin console. For more information, see “Configuring Standard Sign-In Pages” on page 266.
**Customized sign-in pages**—Customized sign-in pages are THTML pages that you produce using the Template Toolkit and upload to the Infranet Controller in the form of an archived ZIP file. The customized sign-in pages feature enables you to use your own pages rather than having to modify the sign-in page included with the Infranet Controller.

For more information on customized sign-in pages, see the *Custom Sign-In Pages Solution Guide*.

**Configuring Standard Sign-In Pages**

You can modify the default sign-in page that the Infranet Controller displays to users when they sign into the Infranet Controller. You can also create new standard sign-in pages that contain custom text, logo, colors, and error message text using settings in the Authentication > Signing In > Sign-in Pages tab of the admin console.

To create or modify a standard sign-in page:

1. In the admin console, choose **Authentication > Signing In > Sign-in Pages**.
2. If you are:
   - Creating a new page—Click **New Page**.
   - Modifying an existing page—Select the link corresponding to the page you want to modify.
3. Enter a name to identify the page.
4. In the **Custom text** section, revise the default text used for the various screen labels as desired. When adding text to the **Instructions** field, note that you may format text and add links using the following HTML tags: `<i>`, `<b>`, `<br>`, `<font>`, and `<a href>`. However, the Infranet Controller does not rewrite links on the sign-in page (since the user has not yet authenticated), so you should only point to external sites. Links to sites behind a firewall will fail.

   **NOTE:** If you use unsupported HTML tags in your custom message, the Infranet Controller may display the end-user’s Infranet Controller home page incorrectly.

5. In the **Header appearance** section, specify a custom logo image file for the header and a different header color.
6. In the **Custom error messages** section, revise the default text that is displayed to users if they encounter certificate errors.

   You can include `<host>`-, `<port>`-, `<protocol>`-, and `<request>`- variables and user attribute variables, such as `<userAttr.cn>`, in the custom error messages. Note that these variables must follow the format `<variable>` to distinguish them from HTML tags which have the format `<tag>`.
7. To provide custom help or additional instructions for your users, select **Show Help button**, enter a label to display on the button, and specify an HTML file to upload to the Infranet Controller. Note that the Infranet Controller does not display images and other content referenced in this HTML page.

8. Click **Save Changes**. The changes take effect immediately, but users with active sessions might need to refresh their Web browsers.

---

**NOTE:** Click **Restore Factory Defaults** to reset the sign-in page, Infranet Controller user home page, and admin console appearance.
Part 3

Endpoint Defense

The Trusted Computing Group (TCG) is a not-for-profit organization formed in 2003 to develop, define, and promote open standards for hardware-enabled trusted computing and security technologies across multiple platforms, peripherals, and devices. The TCG has over 100 members that include component vendors, software developers, systems vendors, and network companies.

Trusted Network Connect (TNC) is a subgroup of the TCG that created an architecture and set of standards for verifying endpoint integrity and policy compliance during or after a network access request. Many of the TCG members participated in the definition and specification of the TNC’s architecture. The TNC defined several standard interfaces that enable components from different vendors to securely operate together. The TNC architecture is designed to build on established standards and technologies, such as 802 1X, RADIUS, IPsec, EAP, and TLS/SSL. For more information about TNC, see www.trustedcomputinggroup.org.

Using technology based on the TNC architecture and standards, the Host Checker component of the Unified Access Control solution provides a comprehensive approach to assess the trustworthiness of endpoints. Host Checker is a client-side agent that runs on agentless endpoints, and Odyssey Access Client includes Host Checker as a built-in component.

You can use Host Checker to perform endpoint checks on hosts before allowing them to connect to the Infranet Controller and access protected resources. Host Checker can check for third party applications, files, process, ports, registry keys, and custom DLLs on hosts. Based on the results of the checks, it can then deny or allow access to protected resources. For example, you may choose to check for virus detection software before allowing a user access to any of the Infranet Controller realms, launch the software on the user’s system if necessary, map the user to roles based on individual policies defined in your own DLL, and then further restrict access to individual resources based on the existence of spyware detection software. When a user’s computer does not meet the requirements you specify, you can display remediation instructions to users so they can bring their computers into compliance.

Host Checker also supports TNC-based integrity measurement collectors (IMCs) and integrity measurement verifiers (IMVs). IMCs are software modules that run on the host and collect information such as antivirus, antispyware, patch management, firewall, and other configuration and security information about the host. IMVs are software modules that run on the Infranet Controller and verify a particular aspect of an host’s integrity. Each IMV on the Infranet Controller works with the corresponding IMC on the host to verify that the host meets the requirements of the integrity measurement custom rule(s) that you configure. IMCs frequently scan the...
client machine for changes in security status. Some IMCs can detect a change in status (for example, if the user turns off virus checking) and then trigger a new check to make sure the modified system complies with the requirements of the Host Checker policy. You can configure Host Checker to monitor third-party IMCs installed on client computers by using third-party IMVs that are installed on a remote IMV server.

For more information on Host Checker, see “Host Checker” on page 271.
Chapter 9

Host Checker

Host checker is a client-side agent that performs endpoint health and security checks for hosts that attempt to connect to the Infranet Controller. You can invoke Host Checker at the role level, or the realm level to specify access requirements for endpoints attempting to authenticate.

The Host Checker component is built into Odyssey Access Client and the Java agent, and a client-side Host Checker application is downloaded for agentless access. Non-UAC agent clients (the Windows native supplicant) can take advantage of Host Checker’s Statement of Health (SOH) integration feature.

The TNC Architecture Within Host Checker

All Host Checker rules are implemented through IMCs (integrity measurement collectors) and IMVs (integrity measurement verifiers) based on the TNC (Trusted Network Computing) open architecture.

IMCs are software modules that Host Checker runs on the client machine. IMCs are responsible for collecting information, such as antivirus, antispyware, patch management, firewall, and other configuration and security information for a client machine.

IMVs are software modules running on the Infranet Controller that are responsible for verifying a particular aspect of an endpoint’s integrity.

The Infranet Controller and Host Checker manage the flow of information between the corresponding pairs of IMVs and IMCs. Each IMV on the Infranet Controller works with the corresponding IMC on the client machine to verify that the client meets the Host Checker rules.
You can also configure Host Checker to monitor third-party IMCs installed on client computers by using third-party IMVs that are installed on a remote IMV server. For more information, see “Using third-party integrity Measurement Verifiers” on page 299.

**NOTE:**

- The TNC integrity measurement rules apply to Windows client machines that are running either Odyssey Access Client or Host Checker (agentless), and to Macintosh with the Safari browser and Linux and Solaris platforms using Firefox browser with Java support. Windows Java Host Checker is not supported.

- The Infranet Controller and Host Checker comply with the standards produced by the Trusted Network Connect (TNC) subgroup of the Trusted Computing Group. For more information about IMVs and IMCs, see [www.trustedcomputinggroup.org](http://www.trustedcomputinggroup.org).

Host Checker runs on agentless and Java agent endpoints, and Odyssey Access Client includes a built-in Host Checker component. In this chapter, the name Host Checker refers to both the software that runs on agentless and Java agent endpoints and the built-in Host Checker component that runs as part of Odyssey Access Client.

You can also configure Statement of Health Host Checker policies to protect Odyssey Access Client and Windows clients (Vista and XP with Service Pack 3) that authenticate with 802.1X.

## Host Checker Overview

The Infranet Controller can check hosts for endpoint properties using a variety of rule types, including:

- Predefined rules that check for antivirus software and up-to-date virus signatures, firewalls, malware, spyware, and specific operating systems from a wide variety of industry leaders

- Custom rules that use integrity measurement collectors (IMCs) and integrity measurement verifiers (IMVs) to perform customized client-side checks.

- Custom rules that check for third party DLLs that perform customized client-side checks.

- Custom rules that check for ports, processes, files, registry key settings, and the NetBIOS name, MAC addresses or certificate of the client machine.

If the user’s computer does not meet any of the Host Checker policy requirements, you can display a custom-made HTML remediation page to the user. This page can contain your specific instructions as well as links to resources to help the user bring his computer into compliance with each Host Checker policy.
If you have upgraded to version 2.2 from a previous version, you should ensure that existing endpoints are upgraded to the current version of Odyssey Access Client before you configure Host Checker policies that implement new features. The agent is required to connect and establish a session with the server to perform an agent-based upgrade, therefore the Odyssey Access Client must pass any Host Checker restrictions required for a realm and role(s) to gain access.

If an endpoint with an older version of Odyssey Access Client attempts to log into the Infranet Controller with a new Host Checker feature implemented at the realm level, the user’s authentication will fail, and the endpoint cannot upgrade to the latest version of Odyssey Access Client.

If you implement new features before all of your endpoints have been upgraded, you must create a remediation role that does not require any of the new features to allow users to authenticate and download the current version of Odyssey Access Client.

This section contains the following information about Host Checker:

- “Task Summary: Configuring Host Checker” on page 273
- “Creating Global Host Checker Policies” on page 275
- “Implementing Host Checker Policies” on page 311
- “Remediating Host Checker Policies” on page 315
- “Specifying General Host Checker Options” on page 320
- “Specifying Odyssey Access Client and Host Checker Installation Options” on page 322
- “Using Host Checker Logs” on page 325

---

## Task Summary: Configuring Host Checker

To configure Host Checker, you must perform these tasks:

1. Create and enable Host Checker policies through the **Authentication > Endpoint Security > Host Checker** page of the admin console, as explained in “Creating Global Host Checker Policies” on page 275.

2. Configure additional system-level options through the **Authentication > Endpoint Security > Host Checker** page of the admin console as necessary:
   - If you want to display remediation information to users when they fail to meet the requirements of a Host Checker policy, configure remediation options through the **Authentication > Endpoint Security > Host Checker** page of the admin console, as explained in “Remediating Host Checker Policies” on page 315.
If you want to change the default Host Checker settings, configure them through the **Authentication > Endpoint Security > Host Checker** page of the admin console, as explained in “Specifying General Host Checker Options” on page 320.

3. Determine at which levels within the Infranet Controller access management framework you want to enforce the policies:

- To enforce Host Checker policies when the user first accesses the Infranet Controller, implement the policies at the realm level by using the **Administrators > Admin Realms > Select Realm > Authentication Policy > Host Checker** or the **Users > User Realms > Select Realm > Authentication Policy > Host Checker** pages of the admin console.

- To allow or deny users access to roles based on their compliance with Host Checker policies, implement the policies at the role level by using the **Administrators > Admin Roles > Select Role > General > Restrictions > Host Checker** or the **Users > User Roles > Select Role > General > Restrictions > Host Checker** pages of the admin console.

- To map users to roles based on their compliance with Host Checker policies, use custom expressions in the **Administrators > Admin Realms > Select Realm > Role Mapping** or the **Users > User Realms > Select Realm > Role Mapping** pages of the admin console.

For more information, see “Configuring Host Checker Restrictions” on page 313.

4. Specify how users can access the Host Checker client-side agent that enforces the policies you define:

- (Windows only) If you enable Odyssey Access Client installation (see “Creating an initial Configuration of Odyssey Access Client” on page 44), the Infranet Controller automatically downloads Odyssey Access Client with its built-in Host Checker when the user first accesses the Infranet Controller by means of a web browser. If you evaluate or enforce a Host Checker policy at the realm level, Odyssey Access Client automatically runs its built-in Host Checker on the endpoint to check it for security compliance.

- (Java agent or agentless deployments only) To enable automatic installation of the Host Checker component on agentless computers, you must evaluate or enforce a Host Checker policy on the **Administrators > Admin Realms > Select Realm > Authentication Policy > Host Checker** page or the **Users > User Realms > Select Realm > Authentication Policy > Host Checker** page of the admin console.

- To download Odyssey Access Client or Host Checker installer and manually install it on your Windows users’ systems, use the **Maintenance > System > Installers** page of the admin console.
Creating Global Host Checker Policies

To use Host Checker as a policy enforcement tool for managing endpoints, you must create global Host Checker policies through the Authentication > Endpoint Security > Host Checker page of the admin console, and then implement the policies at the realm or role levels.

The Infranet Controller provides several mechanisms that you can use to enable, create, and configure Host Checker policies:

- **Pre-defined rules (check for third party applications)**—Host Checker contains a wide vast array of pre-defined rules that check for antivirus software, firewalls, malware, spyware, and specific operating systems from a wide variety of industry leaders. You can enable one or more of these rules within a Host Checker client-side policy to ensure that the integrated third party applications that you specify are running on your users’ computers in accordance with your specifications. For more information, see “Checking for Third-Party Applications Using Pre-defined Rules (Windows Only)” on page 278.

- **Custom rules (check for additional requirements)**—In addition to the Pre-defined rules, you can create custom rules within a Host Checker policy to define requirements that your users’ computers must meet. Using custom rules, you can:
  - Configure Host Checker to check for custom third party DLLs that perform customized client-side checks.

**NOTE:** If multiple administrators or end-users to a single Infranet Controller are signed in from the same client system and at least one of them deploys Host Checker, unexpected results may occur. For example, Host Checker may shut down, role privileges may be lost and forced disconnections may occur.
Creating and Configuring New Client-side Policies

You can create a variety of policies through the Host Checker client that check for antivirus software, firewalls, malware, spyware, and specific operating systems from a wide variety of industry leaders. You can also create Host Checker policies that use third-party integrity measurement verifiers (IMVs) and third-party DLLs, or check for ports, processes, files, registry keys, and the NetBIOS name, MAC addresses or certificate of the client machine.

When creating the policies, you must define the policy name, and either enable pre-defined rules, or create custom rules that run the specified checks. Optionally, you can specify how Host Checker should evaluate multiple rules within a single policy.
To create a standard client-side policy:

1. In the admin console, choose **Authentication > Endpoint Security > Host Checker**.

2. Under **Policies**, click **New**.

3. Enter a name in the **Policy Name** field and then click **Continue**. (Users see this name on the Host Checker remediation page if you enable custom instructions for this policy.)

4. Create one or more rules to associate with the policy using instructions in the following sections:
   - “Checking for Third-Party Applications Using Pre-defined Rules (Windows Only)” on page 278
   - “Specifying Customized Requirements Using Custom Rules” on page 286

5. Specify how Host Checker should evaluate multiple rules within the policy using instructions in “Evaluating Multiple Rules in a Single Host Checker Policy” on page 294.

6. (Recommended) Specify remediation options for users whose computers do not meet the requirements specified in the policy. For instructions, see “Configuring General Host Checker Remediation” on page 317.

7. Implement the policy at the realm or role levels using the options described in “Configuring Host Checker Restrictions” on page 313.

### Using Host Checker for Machine Account Logins

With Odyssey Access Client you can use a machine account configuration to authenticate a physical machine to the network, rather than a user. This type of configuration uses either a statically defined user account or the machine credentials that were created when the machine account was set up in Active Directory. See the **Odyssey Access Client Administration Guide**.

If you use the machine account feature, do not configure any restrictive Host Checker policies that require human interaction for remediation, as these policies will fail. Instead, you should configure a separate default role for machine accounts that fail Host Checker policies.

To allow the machine account to login, you configure a special role for machines. You can create a role mapping policy that maps all machine roles to a realm that automatically accepts your domain with a wildcard as a username. For example, “QA\*$”. 

Checking for Third-Party Applications Using Pre-defined Rules (Windows Only)

Host Checker comes pre-equipped with a vast array of pre-defined rules that check for antivirus software, firewalls, malware, spyware, and specific operating systems from a wide variety of industry leaders. You can enable one or more of these rules within a Host Checker client-side policy to ensure that the integrated third party applications that you specify are running on your users’ computers in accordance with your specifications. For firewall and antivirus rules, you can specify remediation actions to automatically bring the endpoint into compliance.

- **Predefined: AntiVirus**—Select this option to create a rule that checks for the antivirus software that you specify, and to specify remediation options. See “Configuring a Predefined Antivirus Rule with Remediation Options” on page 279.

- **Predefined: Firewall**—Select this option to create a rule that checks for the firewall software that you specify, and to specify remediation options. See “Configuring a Predefined Firewall Rule with Remediation Options” on page 281.

- **Predefined: Malware**—Select this option to create a rule that checks for the malware protection software that you specify.

- **Predefined: Spyware**—Select this option to create a rule that checks for the spyware protection software that you specify. See “Configuring a Predefined Spyware Rule” on page 283.

- **Predefined: OS Checks**—Select this option to create a rule that checks for the Windows operating systems and minimum service pack versions that you specify. (Any service pack whose version is greater than or equal to the version you specify satisfies the policy.) See

To create a Host Checker rule using Predefined Malware or Predefined OS Checks rules:

1. In the admin console, choose **Authentication > Endpoint Security > Host Checker**.

2. Create a new policy using instructions in “Creating and Configuring New Client-side Policies” on page 276 or click on an existing policy in the **Policies** section of the page.

3. Under **Rule Settings**, choose one of the following options and click **Add**:

   - Predefined Malware
   - Predefined OS Checks

4. In the **Add Predefined Rule** page:

5. In the **Rule Name** field, enter an identifier for the rule.
6. Under Criteria, select the specific malware or operating systems that you want to check for and click Add. (When checking for an operating system, you may also specify a service pack version.)

**NOTE:** When you select more than one type of software within a pre-defined rule, Host Checker considers the rule satisfied if any of the selected software applications are present on the user’s machine.

7. Under Optional, select Monitor this rule for change in result to continuously monitor the policy compliance of endpoints. If this check box is selected, and a change in compliance status on an endpoint that has successfully logged in occurs, the Infranet Controller initiates a new handshake to re-evaluate realm or role assignments.

**NOTE:** If the underlying TNCC service is killed or stopped, the endpoint can remain on the network, potentially out of compliance, until the next Host Checker policy refresh. For more information about TNCC see “The TNC Architecture Within Host Checker” on page 271.

8. Click Save Changes.

9. Optionally add additional rules to the policy, specify how Host Checker should evaluate multiple rules within the policy, and define remediation options using instructions in “Creating and Configuring New Client-side Policies” on page 276.

**NOTE:** To view the currently supported applications, go to Authentication > Endpoint Security > Host Checker and create a new policy. You can choose pre-defined rule types from the Select Rule Type drop down list box to see a list of the supported applications within that category. The lists of applications can be quite extensive and are updated at each support release, so it is useful to check the list periodically.

### Configuring a Predefined Antivirus Rule with Remediation Options

You can configure antivirus remediation actions with Host Checker. You can specify a requirement for the age (in days) of the last successful virus scan, and you can specify that virus signatures installed on client machines should not be older than a specified number of updates.

You can also monitor policies to ensure that logged-in endpoints maintain compliance status, and remediate the endpoint to another role or realm depending on the current status.

If a client attempts to log in, and the client machine does not meet the requirements you specify, Host Checker can attempt to correct the deficiencies to allow the client to successfully log in. With Host Checker antivirus remediation, you can prompt the endpoint to download the latest virus signature files, turn on antivirus protection, and initiate an antivirus scan.
All of the remediation options are not supported for all antivirus software vendors’ products. All available vendors and products that are supported are displayed when you select the **Require any supported product** option button.

Alternately, you can select the **Require specific products/vendors** option button and select either the **Require any supported product from a specific vendor** or **Require specific products** check boxes, then add an available type to **Selected Types**. The remediation options appear, and you can determine which remediation options are available for specific products or vendors.

To configure a Predefined Antivirus rule:

1. In the admin console, choose **Authentication > Endpoint Security > Host Checker**.
2. Create a new policy using instructions in “Creating and Configuring New Client-side Policies” on page 276 or click on an existing policy in the **Policies** section of the page.
3. Under **Rule Settings**, choose **Predefined: Antivirus** and click **Add**.
4. Enter a name for this antivirus rule.
5. To determine if your software vendor’s product is supported for the System Scan check, click **these Antivirus products**. A new window will open with a list of all of the products that support the feature.
6. Select or clear the check box next to **Successful System Scan must have been performed in the last _ days**, and enter the number of days in the field.
   
   If you select this check box, a new option appears. If the remediation action to start an antivirus scan has been successfully begun, you can override the previous check.
7. Select or clear the check box next to **Consider this rule as passed if ‘Full System Scan’ was started successfully as remediation**.
8. Select or clear the check box next to **Virus definition files should not be older than _ updates**. Enter a number between 1 and 10. If you enter 1, the client must have the latest update. You must import the virus signature list for the supported vendor. See “Configuring Virus Signature Version Monitoring and Patch Assessment Data Monitoring” on page 284.
9. Select your antivirus vendor(s) and product(s) by using either the **Require any supported product** or **Require specific products/vendors** option buttons.

   Require any supported product allows you to check for any product (rather than requiring you to select every product separately). This option button reveals a list of products in the remediation section to allow you to enable remediation options which are product specific.

   Require specific products/vendors allows you to define compliance by allowing any product by a specific vendor (for example, any Symantec product).
Require specific products provides functionality that allows you to select individual products to define compliance.

After you select your vendor(s) and product(s), remediation options will appear on the page.

For each of the following remediation actions:

- Download latest virus definition files—(obtains the latest available file for the specified vendor)
- Turn on Real Time Protection—(launches the virus scanning mechanism for the specified vendor)
- Start Antivirus Scan—(performs a real-time virus scan for the specified vendor)

the check box is active (clickable) if the action is supported for your product.

If your antivirus product is not supported, you can click the remediation column headers to determine what vendors and products are supported.

10. If your product is supported, select the check box for any or all of the remediation actions that you want to apply.

11. Under Optional, select Monitor this rule for change in result to continuously monitor the policy compliance of endpoints. If this check box is selected, and a change in compliance status on an endpoint that has successfully logged in occurs, the Infranet Controller initiates a new handshake to re-evaluate realm or role assignments.

**NOTE:** If the underlying TNCC service is killed or stopped, the endpoint can remain on the network, potentially out of compliance, until the next Host Checker policy refresh. For more information about TNCC see “The TNC Architecture Within Host Checker” on page 271.

12. Click Save Changes to save the antivirus rule and enforce antivirus remediation.

13. Optionally add additional rules to the policy, specify how Host Checker should evaluate multiple rules within the policy, and define remediation options using instructions in “Creating and Configuring New Client-side Policies” on page 276.

**Configuring a Predefined Firewall Rule with Remediation Options**

You can configure firewall remediation actions with Host Checker after you create a Host Checker firewall rule that requires the endpoint to have a specific firewall installed and running prior to connecting to the network.

After you enforce the Host Checker rule with firewall remediation actions, if an endpoint attempts to log in without the required firewall running, Host Checker can attempt to enable the firewall on the client machine.
The remediation option is not supported for all firewall products. All available products are displayed by using the Require any supported product or Require specific products/vendors option buttons.

To configure a Host Checker Predefined Firewall rule:

1. In the admin console, choose Authentication > Endpoint Security > Host Checker.

2. Create a new policy using instructions in “Creating and Configuring New Client-side Policies” on page 276 or click an existing policy in the Policies section of the page.


4. Enter a name for the firewall rule.

5. Select your firewall vendor(s) and product(s) by using either the Require any supported product or Require specific products/vendors option buttons.

   Require any supported product allows you to check for any product (rather than requiring you to select every product separately). This option button reveals a list of products in the remediation section to allow you to enable remediation options which are product specific.

   When you add an available product to Selected Products, the remediation option appears, and you can determine if the remediation option is available for your selected firewall.

   Require specific products/vendors allows you to define compliance by allowing any product by a specific vendor (for example, any Symantec product).

   Require specific products provides functionality that allows you to select individual products to define compliance.

   After you select your vendor(s) and product(s), the remediation options on will appear on the page. The Turn on Firewall check box is active (clickable) if the action is supported for your product.

6. If your firewall is supported, select the check box to Turn on Firewall.

7. Under Optional, select Monitor this rule for change in result to continuously monitor the policy compliance of endpoints. If this check box is selected, and a change in compliance status on an endpoint that has successfully logged in occurs, the Infranet Controller initiates a new handshake to re-evaluate realm or role assignments.

   **NOTE:** If the underlying TNCC service is killed or stopped, the endpoint can remain on the network, potentially out of compliance, until the next Host Checker policy refresh. For more information about TNCC see “The TNC Architecture Within Host Checker” on page 271.

8. Click Save Changes to save the firewall rule and enforce firewall remediation.
9. Optionally add additional rules to the policy, specify how Host Checker should evaluate multiple rules within the policy, and define remediation options using instructions in “Creating and Configuring New Client-side Policies” on page 276.

**Configuring a Pre-defined Spyware Rule**

You can configure Host Checker to check for installed spyware on endpoints.

After you enforce the Host Checker rule, if an endpoint attempts to log in without the required spyware, the Host Checker rule will fail.

The option is not supported for all spyware products. All available products are displayed by using the Require any supported product or Require specific products/vendors option buttons.

To configure a Host Checker Predefined Spyware rule:

1. In the admin console, choose **Authentication > Endpoint Security > Host Checker**.

2. Create a new policy using instructions in “Creating and Configuring New Client-side Policies” on page 276 or click an existing policy in the **Policies** section of the page.

3. Under **Rule Settings**, choose **Predefined: Spyware** and click **Add**.

4. Enter a name for the firewall rule.

5. Select one of the following options:

6. Select the **Require any supported product** option button to check for any product (rather than requiring you to select every product separately).

7. Select the **Require specific products/vendors** option button to specify the spyware that you want to check for.

8. Choose either the Require any supported product from a specific vendor or Require specific products to select your spyware.

   Add your available spyware from Available Products to Selected Products.

9. Under **Optional**, select **Monitor this rule for change in result** to continuously monitor the policy compliance of endpoints. If this check box is selected, and a change in compliance status on an endpoint that has successfully logged in occurs, the Infranet Controller initiates a new handshake to re-evaluate realm or role assignments.

10. Click **Save Changes** to save the spyware rule.

11. Optionally add additional rules to the policy, specify how Host Checker should evaluate multiple rules within the policy, and define remediation options using instructions in “Creating and Configuring New Client-side Policies” on page 276.
Configuring Virus Signature Version Monitoring and Patch Assessment Data Monitoring

You can configure Host Checker to monitor and verify that the virus signatures, operating systems, software versions, and patches installed on client computers are up to date, and remediate those endpoints that do not meet the specified criteria. Host Checker uses the current virus signatures and patch assessment versions from the vendor(s) you specify for pre-defined rules in a Host Checker policy.

You can automatically import the current Virus signature version monitoring or Patch Management Info Monitoring lists from the Juniper Networks staging site at a specified interval, or you can download the files from Juniper and use your own staging server.

You can also configure a proxy server as a staging site between the Infranet Controller and the Juniper site. To use a proxy server, you enter the servers network address, port and authentication credentials, if applicable.

To access the Juniper Networks staging site for updates, you must enter the credentials for your Juniper Networks Support account.

To configure the Infranet Controller to automatically import the current virus signature version monitoring and patch management version monitoring list(s) from the Juniper staging site:

2. Click Virus signature version monitoring, or Patch Management Info Monitoring.
3. Select Auto-update virus signatures list or Auto-update Patch Management data.
4. For Download path, leave the existing URL(s) of the staging site(s) where the current list(s) are stored. The default URLs are the paths to the Juniper Networks staging site:
   - https://download.juniper.net/software/av/uac/epupdate_hist.xml (for auto-update virus signatures list)
   - https://download.juniper.net/software/hc/patchdata/patchupdate.dat (for auto-update patch management)
5. For Download interval, specify how often you want the Infranet Controller to automatically import the current list(s).
6. For Username and Password, enter your Juniper Networks Support credentials.
7. Click Save Changes.

To download the files to use your own staging server:

2. Click **Virus signature version monitoring**, or **Patch Management Info Monitoring**.

3. Download the list(s) from the Juniper staging site to a network server or local drive on your computer by entering the Juniper URLs in a browser window.

   https://download.juniper.net/software/av/uac/epupdate_hist.xml
   (for auto-update virus signatures list)

   https://download.juniper.net/software/hc/patchdata/patchupdate.dat
   (for auto-update patch management)

4. Under **Manually import virus signatures list**, click **Browse**, select the list, and then click **OK**.

5. Click **Save Changes**.

**NOTE:** If you use your own staging site for storing the current list(s), you must upload the trusted root certificate of the CA that signed the staging’s server certificate to the Infranet Controller. For more information, see “Uploading Trusted Server CA Certificates” on page 379.

---

To use a proxy server as the auto-update server:

1. Choose **Authentication > Endpoint Security > Host Checker**.

2. Click **Virus signature version monitoring**, or **Patch Management Info Monitoring**.

3. Select **Auto-update virus signatures list** or **Auto-update Patch Management data**.

4. For **Download path**, leave the existing URL(s) of the staging site(s) where the current list(s) are stored. The default URLs are the paths to the Juniper Networks staging site:

   https://download.juniper.net/software/av/uac/epupdate_hist.xml
   https://download.juniper.net/software/hc/patchdata/patchupdate.dat

5. For **Download interval**, specify how often you want the Infranet Controller to automatically import the current list(s).

6. For **Username** and **Password**, enter your Juniper Networks Support credentials.

7. Select the check box for **Use Proxy Server**.

8. Enter the IP **Address** of your proxy server.

9. Enter the **Port** that the Juniper Networks Support site will use to communicate with your proxy server.
10. If your proxy server is password protected, type the Username and Password of the proxy server.

11. Click Save Changes.

---

**Specifying Customized Requirements Using Custom Rules**

If the pre-defined client-side policies and rules that come with the Infranet Controller do not meet your needs, you can create custom rules within a Host Checker policy to define requirements that your users’ computers must meet. Using custom rules, you can:

- Configure remote integrity measurement verifiers (IMVs) to perform customized client-side checks.
- Configure Host Checker to check for custom DLLs that perform customized client-side checks.
- Verify that certain ports are open or closed on the user’s computer.
- Confirm that certain processes are or are not running on the user’s computer.
- Check that certain files are or are not present on the client machine.
- Evaluate the age and content of required files through MD5 checksums.
- Configure patch assessment rules to determine that users have the latest software versions installed.
- Configure PatchLink rules.
- Confirm that registry keys are set on the client machine, and specify remediation actions.
- Confirm the NETBIOS name of the client machine.
- Confirm the MAC addresses of the client machine.
- Check the validity of the machine certificate that is installed on the user’s computer.

---

**NOTE:** You can only check for registry keys, third-party DLLs, NETBIOS names, MAC addresses, and machine certificates on Windows computers.

---

To create a client-side Host Checker policy:

1. In the admin console, choose **Authentication > Endpoint Security > Host Checker**.

2. Create a new policy using instructions in “Creating and Configuring New Client-side Policies” on page 276 or click on an existing policy in the Policies section of the page.
3. Click the tab that corresponds to the operating system for which you want to specify Host Checker options—Windows, Mac, Linux or Solaris. In the same policy, you can specify different Host Checker requirements on each operating system. For example, you can create one policy that checks for different files or processes on each operating system.

**NOTE:** You must explicitly create policies for each operating system you want to allow. For example, if you create a Windows Host Checker policy, but don’t create one for Mac or Linux, users who sign into the Infranet Controller from a Mac or Linux machine will not comply with the Host Checker policy and therefore will not be able to access the realm, role, or resource on which you enforce Host Checker.

4. Under Rule Settings, choose the options in the following sections and click Add. The Add Custom Rule page for the rule type appears.

- **Custom: Remote IMV**—Use this rule type to configure integrity measurement software that a client must run to verify a particular aspect of the client’s integrity, such as the client’s operating system, patch level, or virus protection.

- **3rd Party NHC Check** (Windows only)—Use this rule type to specify the location of a custom DLL. Host Checker calls the DLL to perform customized client-side checks. If the DLL returns a success value to Host Checker, then the Infranet Controller considers the rule met. In the 3rd Party NHC Check configuration page:
  i. Enter a name and vendor for the 3rd Party NHC Check rule
  ii. Enter the location of the DLL on client machines (path and file name).
  iii. Click Save Changes.

**NOTE:** The 3rd Party NHC Check feature is primarily provided for backwards compatibility. We recommend that you use IMCs and IMVs instead.
- **Ports**—Use this rule type to control the network connections that a client can generate during a session. This rule type ensures that certain ports are open or closed on the client machine before the user can access the Infranet Controller. In the **Ports** configuration page:
  
  i. Enter a name for the port rule.
  
  ii. Enter a comma delimited list (without spaces) of ports or port ranges, such as: **1234,11000-11999,1235**.
  
  iii. Select **Required** to require that these ports are open on the client machine or **Deny** to require that they are closed.
  
  iv. Under **Optional**, select **Monitor this rule for change in result** to continuously monitor the policy compliance of endpoints. If this check box is selected, and a change in compliance status on an endpoint that has successfully logged in occurs, the Infranet Controller initiates a new handshake to re-evaluate realm or role assignments.
  
  v. Click **Save Changes**.

- **Process**—Use this rule type to control the software that a client may run during a session. This rule type ensures that certain processes are running or not running on the client machine before the user can access resources protected by the Infranet Controller. In the **Processes** configuration page:
  
  i. Enter a name for the process rule.
  
  ii. Enter the name of a process (executable file), such as: **good-app.exe**.
  
  You can use a wildcard character to specify the process name. For example:
  
  **good*.exe**
  
  For more information, see “Using a Wildcard or Environment Variable in a Host Checker Rule” on page 293.
  
  iii. Select **Required** to require that this process is running or **Deny** to require that this process is not running.
  
  iv. Specify the MD5 checksum value of each executable file to which you want the policy to apply (optional). For example, an executable may have different MD5 checksum values on a desktop, laptop, or different operating systems. On a system with OpenSSL installed—many Macintosh, Linux and Solaris systems have OpenSSL installed by default—you can determine the MD5 checksum by using this command:
  
  ```bash
  openssl md5 <processFilePath>
  ```

  **NOTE:** For Linux, Macintosh and Solaris systems, the process that is being detected must be started using an absolute path.
v. Click **Save Changes**.

- **File**—Use this rule type to ensure that certain files are present or not present on the client machine before the user can access the Infranet Controller. You may also use file checks to evaluate the age and content (through MD5 checksums) of required files and allow or deny access accordingly. In the **Files** configuration page:

  i. Enter a name for the file rule.

  ii. Enter the name of a file (any file type), such as: `c:\temp\bad-file.txt` or `/temp/bad-file.txt`.

     You can use a wildcard character to specify the file name. For example:

     `*.txt`

     You can also use an environment variable to specify the directory path to the file. (You cannot use a wildcard character in the directory path.) Enclose the variable between the `<%` and `%>` characters. For example:

     `<%windir%>\bad-file.txt`

     For more information, see “Using a Wildcard or Environment Variable in a Host Checker Rule” on page 293.

  iii. Select **Required** to require that this file is present on the client machine or **Deny** to require that this file is not present.

  iv. Specify the minimum version of the file (optional). For example, if you require notepad.exe to be present on the client, you can enter `5.0` in the field. Host Checker accepts version 5.0 and above, of notepad.exe.

  v. Specify the maximum age (**File modified less than n days**) (in days) for a file (optional). If the file is older than the specified number of days, then the client does not meet the attribute check requirement.

---

**NOTE:** You can use the maximum age option to check the age of virus signatures. Make sure you specify the path to a file in the **File Name** field whose timestamp indicates when virus signatures were last updated, such as a virus signature database or log file that updates each time the database updates. For example, if you use TrendMicro, you may specify:

`C:\Program Files\Trend Micro\OfficeScan Client\TmUpdate.ini`. 
vi. Specify the MD5 checksum value of each file to which you want the policy to apply (optional). On Macintosh, Linux and Solaris, you can determine the MD5 checksum by using this command:

\[ \text{openssl md5 <filePath>} \]

vii. Select **Monitor this rule for change in result** to continuously monitor the policy compliance of endpoints. If this check box is selected, and a change in compliance status on an endpoint that has successfully logged in occurs, the Infranet Controller initiates a new handshake to re-evaluate realm or role assignments.

viii. Click **Save Changes**.

- **Registry Setting** (Windows only)—Use this rule type to control the corporate PC images, system configurations, and software settings that a client must have to access the Infranet Controller. This rule type ensures that certain registry keys are set on the client machine before the user can access the Infranet Controller. You may also use registry checks to evaluate the age of required files and allow or deny access accordingly. In the **Registry Settings** configuration page:

  i. Enter a name for the registry setting rule.

  ii. Select a root key from the drop-down list.

  iii. Enter the path to the application folder for the registry subkey.

  iv. Enter the name of the key’s value that you want to require (optional). This name appears in the **Name** column of the Registry Editor.

  v. Select the key value’s type (**String**, **Binary**, or **DWORD**) from the drop-down list (optional). This type appears in the **Type** column of the Registry Editor.

  vi. Specify the required registry key value (optional). This information appears in the **Data** column of the Registry Editor.

  If the key value represents an application version, select **Minimum version** to allow the specified version or newer versions of the application. For example, you can use this option to specify version information for an antivirus application to make sure that the client antivirus software is current. The Infranet Controller uses lexical sorting to determine if the client contains the specified version or higher. For example:

  - 3.3.3 is newer than 3.3
  - 4.0 is newer than 3.3
  - 4.0a is newer than 4.0b
4.1 is newer than 3.3.1

NOTE: If you specify only the key and subkey, Host Checker simply verifies the existence of the subkey folder in the registry.

vii. Under Optional, select Monitor this rule for change in result to continuously monitor the policy compliance of endpoints. If this check box is selected, and a change in compliance status on an endpoint that has successfully logged in occurs, the Infranet Controller initiates a new handshake to re-evaluate realm or role assignments.

You can configure registry setting remediation actions with Host Checker. If a client attempts to login, and the client machine does not meet the requirements you specify, Host Checker can attempt to correct the discrepancies to allow the client to login.

viii. Select the check box for Set Registry value specified in criteria.

ix. Click Save Changes.

- **NetBIOS** (Windows only, does not include Windows Mobile)—Use this rule type to check the NetBIOS name of the client machine before the user can access the Infranet Controller. In the NetBIOS configuration page:

  i. Enter a name for the NetBIOS rule.

  ii. Enter a comma-delimited list (without spaces) of NetBIOS names. The name can be up to 15 characters in length. You can use wildcard characters in the name and it is not case-sensitive. For example, md*, m*xp and *xp all match MXP.

  iii. Select Required to require that NETBIOS name of the client machine match one of the names you specify, or Deny to require that the name does not match any name.

  iv. Click Save Changes.

- **MAC Address** (Windows only)—Use this rule type to check the MAC addresses of the client machine before the user can access the Infranet Controller. In the MAC Address configuration page:

  i. Enter a name for the MAC address rule.

  ii. Enter a comma-delimited list (without spaces) of MAC addresses in the form XX:XX:XX:XX:XX:XX where the X’s are hexadecimal numbers. For example:

  00:0e:1b:04:40:29

  You can use a * wildcard character to represent a two-character section of the address. For example, you can use a * to represent the “04”, “40”, and “29” sections of the previous example address:

  00:0e:1b:*:*:*
But you cannot use a * to represent a single character. For example, the * in the following address is not allowed:

00:0e:1b:04:40:*9

iii. Select **Required** to require that a MAC address of the client machine matches any of the addresses you specify, or **Deny** to require that the all addresses do not match. A client machine will have at least one MAC address for each network connection, such as Ethernet, wireless, and VPN. This rule’s requirement is met if there is a match between any of the addresses you specify and any MAC address on the client machine.

iv. Click **Save Changes**.

**NOTE:** Since the MAC address is changeable on some network cards, this check may not guarantee that a client machine meets the requirements of your Host Checker policy.

- **Machine Certificate** (Windows only)—Use this rule type to check that the client machine is permitted access by validating the machine certificate stored on the client machine, as explained in “Using Trusted Client CAs” on page 363. In the **Machine Certificate** configuration page:

  i. Enter a name for the machine certificate rule.

  ii. From the **Select Issuer Certificate** list, select the certificate that you want to retrieve from the user’s machine and validate. Or, select **Any Certificate** to skip the issuer check and only validate the machine certificate based on the optional criteria that you specify below.

  iii. From the **Optional** fields (**Certificate field** and **Expected value**), specify any additional criteria that Host Checker should use when verifying the machine certificate.

  iv. Click **Save Changes**.

**NOTE:**

- If more than one certificate is installed on the client machine that matches the specified criteria, The Host Checker client passes the first certificate it finds to the Infranet Controller for validation.

- Do not apply remediation options for endpoints that authenticate using Machine Certificates.

5. Optionally add additional rules to the policy, specify how Host Checker should evaluate multiple rules within the policy, and define remediation options using instructions in “Creating and Configuring New Client-side Policies” on page 276.
Using a Wildcard or Environment Variable in a Host Checker Rule

You can use the following wildcards to specify a file name in a Custom File rule or a process name in a Custom Process rule:

**Table 31: Wildcard Characters for Specifying a File Name or Process Name**

<table>
<thead>
<tr>
<th>Wildcard Character</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>Matches any character</td>
<td>*.txt</td>
</tr>
<tr>
<td>?</td>
<td>Matches exactly one character</td>
<td>app?.exe</td>
</tr>
</tbody>
</table>

In a Custom File rule for Windows, you can use the following environment variables to specify the directory path to a file:

**Table 32: Environment Variables for Specifying a Directory Path on Windows**

<table>
<thead>
<tr>
<th>Environment variable</th>
<th>Example Windows Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;%-APPDATA%&gt;</td>
<td>C:\Documents and Settings\jdoe\Application Data</td>
</tr>
<tr>
<td>&lt;%-windir%&gt;</td>
<td>C:\WINDOWS</td>
</tr>
<tr>
<td>&lt;%-ProgramFiles%&gt;</td>
<td>C:\Program Files</td>
</tr>
<tr>
<td>&lt;%-CommonProgramFiles%&gt;</td>
<td>C:\Program Files\Common Files</td>
</tr>
<tr>
<td>&lt;%-USERPROFILE%&gt;</td>
<td>C:\Documents and Settings\jdoe</td>
</tr>
<tr>
<td>&lt;%-HOMEDRIVE%&gt;</td>
<td>C:</td>
</tr>
<tr>
<td>&lt;%-Temp%&gt;</td>
<td>C:\Documents and Settings &lt;user name&gt;\Local Settings\Temp</td>
</tr>
</tbody>
</table>

In a Custom File rule for Macintosh, Linux and Solaris, you can use the following environment variables to specify the directory path to a file:

**Table 33: Environment Variables for Specifying a Directory Path on Macintosh, Linux and Solaris**

<table>
<thead>
<tr>
<th>Environment variable</th>
<th>Example Macintosh Value</th>
<th>Example Linux and Solaris Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;%-java.home%&gt;</td>
<td>/System/Library/Frameworks/JavaVM .framework/ Versions/1.4.2/Home</td>
<td>/local/local/java/j2sdk1.4.1_02/ jre</td>
</tr>
<tr>
<td>&lt;%-java.io.tmpdir%&gt;</td>
<td>/tmp</td>
<td>/tmp</td>
</tr>
<tr>
<td>&lt;%-user.dir%&gt;</td>
<td>/Users/admin</td>
<td>/home-shared/cknouse</td>
</tr>
<tr>
<td>&lt;%-user.home%&gt;</td>
<td>/Users/admin</td>
<td>/home/cknouse</td>
</tr>
</tbody>
</table>

**NOTE:** Although environment variables are formatted in the same way as Toolkit Template directives, they are not interchangeable and you should not confuse them.
Evaluating Multiple Rules in a Single Host Checker Policy

If you choose to include multiple rules within a single client-side policy, you must specify how Host Checker should evaluate those rules.

To specify requirements for multiple rules within a Host Checker policy:

1. In the admin console, choose Authentication > Endpoint Security > Host Checker.
2. In the Policies section of the page, click on an existing policy that includes multiple rules.
3. In the Require section, select one of the following options:
   - All of the above rules—Select this option to specify that the user’s computer must return a success value for all of the policy’s rules to gain access.
   - Any of the above rules—Select this option to specify that the user’s computer must return a success value for any of the policy’s rules to gain access.
   - Custom—Select this option to customize the rules that the user’s computer must meet to gain access. Then, create the custom rule using instructions in the following step.
4. (Custom expressions only) If you want to use alternative sets of rules in the policy, combine rules with Boolean operators (AND, OR, NOT) using the following guidelines:
   - Enter the name of the rules in the Rules expression text box.
   - Use the AND or && operator to require two rules or sets of rules to return a true value.
   - Use the OR or || operator to require either of two rules or sets to return a true value.
   - Use the NOT or ! operator to exclude a rule.
   - Use parenthesis to combine sets of rules.
   - When combining integrity measurement custom rules, you can use the key words Allow, Deny, Isolate, and NoRecommendation. For example, you can use the following expressions to require a personal firewall to run, and require either of two possible antivirus products to run:
     
     ZoneLabsFirewall AND (McAfeeAntivirus OR NortonAntivirus)
5. Click Save Changes.
Configuring the PatchLink IMC/IMV (Windows Only)

To configure the PatchLink integrity measurement custom rule in a client-side Host Checker policy:

1. In the admin console, choose Authentication > Endpoint Security > Host Checker.

2. Create a new policy using instructions in “Creating and Configuring New Client-side Policies” on page 276 or click on an existing policy in the Policies section of the page.

3. Click the Windows tab.


5. To display remediation information to users, select the Send Reason Strings option under Remediation. For more information about this option, see “Configuring General Host Checker Remediation” on page 317.


7. Enter a name for the integrity measurement rule.

8. For Server, enter the IP address or DNS name of the PatchLink server you want to use for compliance checking.

9. Under Criteria, specify the integrity measurement requirements for the vendor you selected:

   - If you select Required and the criteria in this rule matches, the rule results in an Allow recommendation; otherwise the result is Deny.

   - If you select Deny and the criteria in this Integrity Measurement rule matches, the rule results in a Deny recommendation; otherwise the result is Allow.

   
   If the IMV cannot determine the outcome of the rule (for example, the related IMC is not running on the client machine), the result of the rule is No Recommendation.

   - For Patch Status, select Equal or Not Equal from the first drop-down list and choose one of the following values from the second drop-down list:

     - **Compliant**: An endpoint is compliant if PatchLink updates are implemented according to the endpoint baseline definition.

     - **Non-Compliant**: An endpoint is noncompliant if PatchLink updates are not implemented according to the endpoint baseline definition.

     - **No Baseline Defined**: If a PatchLink baseline definition does not exist at the endpoint, you can define a policy for endpoints in this state.

---

**NOTE:** PatchLink software must be installed on endpoints to run the Patchlink IMC.
Not Found: If the PatchLink server has not registered the endpoint, it is considered to be not found. You can define a policy for endpoints in this state.

For Age, select Less, Less or Equal, Greater, or Greater or Equal from the drop-down menu, and enter the number of seconds that determine the maximum time in seconds since the client last communicated with the Patchlink server.

See the PatchLink documentation for more information on these values.

10. Click Save Changes.

Configuring Patch Assessment Policies

You can configure Host Checker policies that check for Windows endpoint’s operating system service pack, software version, or desktop application patch version compliance.

Host Checker uses a list of the most current patch versions from the vendor for predefined rules in the Host Checker policy.

You obtain the most current patch version information from a staging site at Juniper Networks. You can manually download and import the current list into the Infranet Controller, or you can automatically import the current list from the Juniper Networks staging site or your own staging site at the specified interval.

Checks can be based on one or more specified products or on specific patches, though not in the same policy. For example, you could check for Internet Explorer version 7 with one policy, and Patch MS00-039: SSL Certificate Validation Vulnerabilities with a second policy. Then, apply both policies to endpoints at the role or realm level to ensure that the user has the latest browser version with a specific patch. Additionally, you can specify the severity level of patches that you wish to ignore for Microsoft products; for example, you could choose to ignore low or moderate threats.

The Infranet Controller can send remediation instructions (e.g. a message describing what patches or software are non-compliant, and a link to where the endpoint can obtain the patch). The Infranet Controller does not auto-remediate in the event of a non-compliant endpoint. However, you can choose to send the items to the client for manual remediation of managed machines.

When an endpoint first connects to the Infranet Controller, the latest versions of the data files and libraries of the IMC are downloaded to the host computer. The initial check takes 10-20 seconds to run, depending on the link speed. If outdated, these files are automatically updated at subsequent checks. If this is the first time the endpoint has connected to an Infranet Controller with the patch assessment policy, and the connection is a Layer 2 connection, the IMC required to run the Patch Assessment check cannot download. In this case, you should configure a remediation role which displays instructions to direct the user to retry with a Layer 3 connection or contact the administrator.
To configure a patch assessment custom rule:

1. In the admin console, choose **Authentication > Endpoint Security > Host Checker**.

2. Create a new policy using instructions in “Creating and Configuring New Client-side Policies” on page 276 or click on an existing policy in the **Policies** section of the page.

3. Click the **Windows** tab.

4. Under **Rule Settings**, choose **Custom: Patch assessment**.

5. Click **Add** under **Rule Settings**. The **Add Custom Rule: Patch assessment** page appears.

6. Enter a name for the integrity measurement rule.

**NOTE:** If a selection that is not applicable is included in a policy, i.e. the endpoint does not have the targeted software, the rule will be ignored and the check for that particular selection will pass.

7. Select either **Scan for specific products** or **Scan for specific patches**.

   If you select **Scan for specific products** you must further select either **All Products** or **Specific Products**.

   If you select **All Products**, Host Checker checks for all of the exposed patches on the endpoint.

   To configure a policy based on specific products:

   a. Choose the **All Products** option button.

   b. Optionally, select specific patches that you wish to ignore for all products by clicking the **Add** button under **Ignore following patches**.

   c. Click **Save Changes**.

   d. Optionally, for Microsoft products, clear the check boxes to determine the severity level of the patches that you wish to ignore. For example, if you wanted to check for only critical patches for the selections, clear the check boxes for **Important**, **Moderate**, **Low**, and **Unspecified**.

   e. Click **Save Changes**.

   If you select **Specific Products**, two new dialogs open. You can select from an extensive listing of products and versions, and you can choose to ignore specific patches.

   For example, if you add Internet Explorer 6 to the Selected Products list, you can choose to ignore any patches that you do not consider critical for the product. You can further fine-tune the severity level of specific patches to be ignored by clearing the severity check boxes for Microsoft products.
To configure a policy based on specific products:

a. Choose the **Specific Products** option button.

b. Select software from the **Available products** window and add to the **Selected products** window.

c. Click **Save Changes**.

d. Optionally, select specific patches that you wish to ignore for the chosen products by clicking the **Add** button under **Ignore following patches**.

   When you click the **Add** button, a new dialog opens, displaying all of the available patches for the software you have selected.

e. Select specific patches that you wish to ignore from the **Available patches** window and add to the **Selected patches** window.

f. Click the **Add** button under **Add**.

   When you click the **Add** button, the Ignore following patches window is populated with the patches you have chosen.

g. Optionally, clear the check boxes to determine the severity level of the patches for Microsoft products that you wish to ignore. For example, if you wanted to check only for critical patches for the selections, clear the check boxes for **Important**, **Moderate**, **Low**, and **Unspecified**.

h. Click **Save Changes**.

   The **Scan for specific patches** option allows you to choose from a list of all available patches.

To configure a policy based on patches:

a. Choose the **Scan for specific patches** option button.

   When you select the **Scan for specific patches** option a new dialog opens, allowing you to add specific patches.

b. Click the **Add** button.

c. Select specific patches that you wish to check for from the **Available patches** window and add to **Selected patches**.

d. Click the **Add** button.

e. Click **Save Changes**.

8. Click **Save Changes**.
You can display remediation information for users based on which patch/version needs to be updated. For example, you can configure a reason string to display information about a patch that is missing and specify a link to take the user to the web page to get the patch.

9. To display remediation information to users, select the Send Reason Strings option under Remediation. For more information about this option, see “Configuring General Host Checker Remediation” on page 317.

Using third-party integrity Measurement Verifiers

The Trusted Network Connect (TNC) standard enables the enforcement of security requirements for endpoints connecting to networks. The client-side components of the TNC are the IMCs and the TNC-client (TNCC). The TNCC compiles the IMC measurements and sends them to the server. At the server, there is a corresponding set of components: the TNC-server (TNCS) and the IMVs. The TNCS manages the messages between the IMVs and the IMCs and sends the recommendations, based on the IMVs, to the policy engine.

The Infranet Controller and Host Checker comply with the standards produced by the TNC. For more information about the TNC, IMVs and IMCs, see www.trustedcomputinggroup.org.

You can configure Host Checker to monitor third-party TNC-compliant IMCs installed on client computers. To do so, you must:

1. Run the Third-party Integrity Measurement Verifier (IMV) Server installer on the system designated as the remote IMV server. Install the third-party IMVs and create the server certificates. See “Configuring a Remote IMV Server” on page 299.

2. Specify the remote IMV server so that the Infranet Controller can communicate with it. See “Specifying the Remote IMV Server” on page 303.

3. Implement the Host Checker policy. See “Implementing the Third-Party IMV Policy” on page 304.

Configuring a Remote IMV Server

During this step, you install third-party IMVs. Third-party IMVs are installed on the remote IMV server, not on the Infranet Controller.

During this step, you also obtain a server certificate for the remote IMV server. You import the trusted root CA certificate of the CA that generated the server certificate onto the Infranet Controller. The Infranet Controller then authenticates with the remote IMV server through the certificate. If you do not have a certificate authority, install and use OpenSSL to generate a CA certificate.

To install and configure the server software:

1. In the admin console of the Infranet Controller, choose Maintenance > System > Installers and download the Third-party Integrity Measurement Verifier (IMV) Server installer.
2. Run the installer on the system designated as the remote IMV server.

3. Install the third-party IMVs on the remote IMV server and the corresponding IMCs on the client systems.

4. Generate a server certificate from a certificate authority for the remote IMV server. The server’s certificate **Subject CN** value must contain the actual host name or IP address of the remote IMV server. For more information on creating certificates, see “Using Trusted Server CAs” on page 377.

   The server certificate and the private key must be combined into a single PKCS#12 file and encrypted with a password.

   If you do not have a certificate authority, you can use the following steps to create a CA and then create a server certificate for the remote IMV server.

   **NOTE:** Install the full version of OpenSSL. The "light" version of OpenSSL will not work.

Follow the steps below to set up OpenSSL:

   i. Download and install OpenSSL from this site:

      http://www.slproweb.com/products/Win32OpenSSL.html

   ii. At the Windows command prompt, type the following commands:

      ```
      cd \openssl
      md certs
      cd certs
      md demoCA
      md demoCA\newcerts
      edit demoCA\index.txt
      ```

   iii. Press the ALT-F keys and then the S key to save the file.

   iv. Press the ALT-F keys and then the X key to exit the editor.

   v. At the Windows command prompt, type the following commands:

      ```
      edit demoCA\serial
      ```

   vi. Type the following in the document window: 01

   vii. Press the ALT-F keys and then the S key to save the file.

   viii. Press the ALT-F keys and then the X key to exit the editor.

   ix. At the Windows command prompt, type the following command:

      ```
      set path=c:\openssl\bin;%path%
      ```
Follow the steps below to create a CA key:

ix. To create a CA key, type the following command at the Windows command prompt in the \c:\openssl\certs directory:

```bash
openssl genrsa -out ca.key 1024
```

The following output should appear:

```
Loading 'screen' into random state - done
Generating RSA private key, 1024 bit long modulus
........+++++
.++++++
e is 65537 (0x10001)
```

Follow the steps below to create a CA Certificate:

i. Type the following command at the Windows command prompt in the \c:\openssl\certs directory:

```bash
openssl req -new -x509 -days 365 -key ca.key -out demoCA/cacert.pem
```

ii. Enter the appropriate Distinguished Name (DN) information for the CA certificate. You can leave some fields blank by entering a period.

For example:

```
Country Name: US
State or Province Name: CA
Locality Name: Sunnyvale
Organization Name: XYZ
Org. Unit Name: IT
Common Name: ic.xyz.com
Email Address: user@xyz.com
```

c. To set up the CA, type the following command at the Windows command prompt in the directory \c:\openssl\certs:

```bash
copy ca.key demoCA
notepad demoCA.cnf
```

d. When prompted to create a new file, press the yes button.

e. Type the following lines in the document, pressing the Enter key at the end of each line.

```
[ca]
default_ca = demoCA

[demoCA]
dir = ./demoCA
database = $dir/index.txt
new_certs_dir = $dir/newcerts
certificate = $dir/cacert.pem
serial = $dir/serial
private_key = $dir/ca.key
```
default_days = 365
default_md = md5
policy = policy_any
email_in_dn = no
name_opt = ca_default
cert_opt = ca_default
copy_extensions = none

[policy_any]
countryName = supplied
stateOrProvinceName = optional
organizationName = optional
organizationalUnitName = optional
commonName = supplied
emailAddress = optional

f. Save the file and close notepad.

g. Type the following command to generate an RSA private key for the remote IMV server:

openssl genrsa -out rimvs_key.pem 1024

h. Type the following command to generate a CSR for the remote IMV server:

openssl req -new -key rimvs_key.pem -out rimvs_csr.pem

i. Type the following lines:

Country Name:
State or Province Name:
Locality Name:
Organization Name:
Organizational Unit Name:
Common Name: [IPAddress]
Email Address:
A challenge password:
An optional company name:

You may enter any value you like for most fields, but the **Common Name** field must contain the IP address of the machine running the remote IMV server. This machine should have a static IP address.

j. Type the following command to generate a certificate for the remote IMV server:

openssl ca -config demoCA.cnf -in rimvs_csr.pem -out rimvs_cert.pem

k. Type ‘y’ twice when prompted to generate the certificate. This certificate is valid for 365 days by default. If you want a different certificate lifetime, change the default_days parameter in the demoCA.cnf file, or use the –days parameter to the openssl ca command to specify a different lifetime.

l. Type the following command to place the remote IMV server key and certificate in a PKCS#12 file (substitute your password):

5. On the remote IMV server, choose Programs > Juniper Networks > Remote IMV Server > Remote IMV Server Configurator from the Start menu.

6. Under Client Info, click Add.

7. Configure the port to service SOAP requests from the Infranet Controller.

8. Enter the client’s IP address, the number of addresses to use, and the shared secret used by both the Infranet Controller and the remote IMV server.

9. Change logging settings if you choose (log is generated in the install directory).

10. Browse and find the PKCS#12 file you generated in the filesystem.

11. Specify the password associated with the certificate.

12. In the admin console of the Infranet Controller, use the System > Configuration > Certificates > Trusted Server CAs tab to import the trusted root CA certificate of the CA that issued the certificate for the remote IMV server.

   If you used OpenSSL to generate the Remote IMV Server’s server certificate is: demoCA\cacert.pem.

   If you did not use OpenSSL to generate this certificate, ensure that the file you import has the CA certificate (not the root certificate).

13. Click Import Trusted Server CA and browse for the server certificate used on the remote IMV server.

14. Add the new remote IMV server on the “Specifying the Remote IMV Server” on page 303.

**Specifying the Remote IMV Server**

To specify the remote IMV server so that the Infranet Controller can communicate with it:

1. In the admin console, choose Authentication > Endpoint Security > Host Checker.


3. In the New Server page:

   a. Create a label for the server using the Name and (optional) Description fields.

   b. In the Hostname field, enter either the IP address or host name as defined in the server certificate.
c. In the **Port** field, enter the unique port number the Infranet Controller uses to communicate with the remote IMV server. Ensure that no other service is using this port number.

The default port number is the same as the default https port number. If you are running a web server on the same system as the Remote IMV Server, enter a new port number in the **Port** field.

d. In the **Shared Secret** field, enter the same shared secret used in the client information entry on the remote IMV server.

e. Click **Save Changes**

4. Under **Remote IMV**, click **New IMV** to specify the third-party IMV.

5. In the **New IMV** page:

a. Create a label for the IMV using the **Name** and (optional) **Description** fields.

b. In the **IMV Name** field, enter the name of the IMV. This name must match the “human readable name” in the IMV’s well-known registry key on the remote IMV server. For more information about human readable names and the well-known registry key, see www.trustedcomputinggroup.org.

c. From the **Primary Server** pop-up menu, select the remote IMV server where this IMV is installed.

d. (Optional) From the **Secondary Server** pop-up menu, select the secondary remote IMV server where this IMV is installed. The secondary server acts as a failover in case the primary server becomes unavailable.

The Infranet Controller continues to try to re-establish connection to the primary remote IMV Server, and uses the primary Remote IMV Server on subsequent handshakes once it becomes available.

e. Click **Save Changes**.

6. Click **Save Changes**.

**Implementing the Third-Party IMV Policy**

To use Host Checker as a policy enforcement tool for managing endpoints, you must create global Host Checker policies at the system level through the **Authentication > Endpoint Security > Host Checker** page of the admin console, and then implement the policies at the realm and role levels.

To implement the third-party IMV policy:

1. In the admin console, choose **Authentication > Endpoint Security > Host Checker**.

2. Under **Policies**, click **New**.
3. Enter a name in the **Policy Name** field and then click **Continue**. (Users see this name on the Host Checker remediation page if you enable custom instructions for this policy.)

4. Under **Rule Settings**, choose **Custom: Remote IMV** and click **Add**.

5. In the **Add Custom Rule: Remote IMV** page:
   a. In the **Rule Name** field, enter an identifier for the rule.
   b. Under **Criteria**, select the third-party IMV to be associated with this rule.
   c. Click **Save Changes**.


7. (Recommended) Specify remediation options for users whose computers do not meet the requirements specified in the policy. For instructions, see “Configuring General Host Checker Remediation” on page 317.

8. Click **Save Changes**.

9. Implement the policy at the realm or role level using the options described in “Configuring Host Checker Restrictions” on page 313.

---

**Statement of Health Integration**

You can use Host Checker policies on the Infranet Controller with the open standard Statement of Health (SOH) protocol by leveraging the existing protocols built into the Infranet Controller. To use this functionality, you must obtain SOH licensing from Juniper Networks.

SOH support is implemented either through local Host Checker criteria that you configure on the Infranet Controller, or you can deploy an external Network Policy Server (NPS) to use third-party system health agents (SHAs) and system health validators (SHVs) as shown in Figure 22 on page 306.

The SOH architecture provides a set of components that evaluates an endpoint’s state of health and makes policy decisions for network access based on the result of the health check. The Infranet Controller supports SOH interoperability for Windows Vista, Windows XP Service Pack 3, and Odyssey Access Client. Windows Security Center (WSC) functionality is built into the Host Checker component that implements this feature for local Host Checking.
You can configure Odyssey Access Client SOH Host Checker policies if you are using Layer 2 and 802.1X or Layer 3 with the Infranet Enforcer. The Windows supplicant can connect only with 802.1X. There is no heartbeat between the Infranet Controller and the Windows client, and no other Host Checker policies are supported with the Windows client. If you are using Odyssey Access Client, you must use the default EAP-JUAC protocol. If you configure a different protocol set for Odyssey Access Client and do not include EAP-JUAC, the SOH cannot be transmitted. SOH Host Checker policies are not supported for agentless access.

The SOH health state validation can be used to determine which roles or realms can be accessed by endpoints. If an endpoint fails the SOH check, or if the SOH cannot be successfully negotiated, the Host Checker policy fails.

Local Host Checker SOH enforcement can be used with either Odyssey Access Client or the Windows supplicant. With local enforcement, you can specify remediation actions if the endpoint does not pass the Host Checker policy. Local enforcement supports the Windows Security Center SHA which includes support for checking the following:

- Antivirus enabled
- Antivirus up to date
- Antispyware enabled
- Antispyware up to date
- Firewall enabled
- Automatic updating Enabled

If you wish to implement additional third-party SHAs, you must use an NPS.

Multiple local evaluations, or multiple NPS evaluations are supported, but clients are limited to one state of health response (SOHR). The client receives the SOHR from the first failed policy. If the endpoint fails multiple policies, the agent receives SOHRs from the failed policies in the order in which they are performed. If the endpoint passes multiple policies, the agent receives the SOHR from the first passed policy.

**Figure 22: SOH Integration with a Network Policy Server**

![Diagram of SOH Integration with a Network Policy Server]

Infranet Controller as Policy Management Server

Network Policy Server

Endpoint with Odyssey Access Client, Microsoft Vista, or Microsoft XP SP3

Policy Enforcement Point
The external NPS and the Infranet Controller communicate using a RADIUS-based protocol. The endpoint requests authentication from the Infranet Controller using the EAP-PEAPv0 protocol. The Infranet Controller communicates SOH information between the NPS and the endpoint as request/response messages. The server connection information that allows the NPS to communicate with the Infranet Controller is configured on the Host Checker SOH interface.

When you use an external NPS, as in all other Host Checker policies, you can restrict users whose endpoints fail to meet the requirements of the Health Registration Authority (HRA) on the NPS.

You can use the NPS as both the AAA server and the health policy server. Windows Server 2008 is the only server supported as an NPS. Please consult the applicable Microsoft documentation for details of setting up the NPS.

If you use an external NPS for evaluation of the SOH, you cannot use local enforcement. Additionally, you should avoid putting any other Host Checker rules into a policy that is designed for SOH enforcement with the Windows client.

An SOH license is required to use this feature. See “Configuring Infranet Controller Licensing and Security” on page 345.

To configure a SOH policy in Host Checker:

1. In the admin console, choose Authentication > Endpoint Security > Host Checker.

2. Create a new policy using instructions in “Creating and Configuring New Client-side Policies” on page 276 or click on an existing policy in the Policies section of the page.

3. Click the Windows tab.


5. Click Add under Rule Settings. The Statement of Health page appears.

6. Type a Rule Name for this rule.

To configure a local SOH Host Checker policy:

1. Under Criteria, enter a Label for a SOH parameter.

2. Select a SOH policy option from the Parameter pull-down menu, then click Add for the following types:
   - Antivirus Enabled
   - Antivirus up to date
   - Antispyware enabled
   - Antispyware up to date
   - Firewall Enabled
3. Select additional options from the Parameter pull-down menu to add additional SOH parameters.

4. Select the check box to the left of each item to activate that item for this Host Checker SOH policy.

**NOTE:** Each SOH rule receives one SOHR. If a single Host Checker rule has multiple policy configurations, all of these are remediated by one SOHR.

5. For each rule you wish to enforce, optionally select the Enable automatic remediation check box. If you select this option for a rule, the user will receive a message from the SOH agent, and appropriate remediation is performed, if possible. If the box is not checked the user receives a message.

**NOTE:** For remediation to succeed, you must configure at least one remediation role for the user. If the user cannot negotiate a realm with at least one role, the user is unable to authenticate and the SOHR is ignored by the SOH agent.

6. Click Save Changes.

To configure a SOH Host Checker policy using a remote NPS:

1. If you have not already done so, configure the Infranet Controller for 802.1X enforcement with the Windows client. See “Using a Non-Juniper 802.1X Supplicant” on page 160.

2. Ensure that your NPS server is configured in accordance with the applicable server documentation.

3. Ensure that the non-Juniper supplicants are configured to authenticate using the appropriate authentication protocols for SOH communication.

4. Select the check box for Use NPS Server under Settings. The Criteria section on the page disappears. You cannot combine a remote NPS rule with a local rule.

5. Enter the following remote server settings:
   - Server IP address
   - Shared Secret of the server
   - Maximum Retries

6. Click Save Changes.
Enabling EAP on the Windows Machine

To use SOH Host Checking on a Windows host machine, you must configure the client to use EAP.

For Windows Vista instructions see the following URL:

For Windows XP SP3 instructions see the following URL:

Use the EAP enforcement ID 79623 for both the Vista and XP SP3 supplicants.

Combining Multiple Integrity Measurement Rules with Custom Expressions

You can combine multiple integrity measurement rules in a single Host Checker policy, and each rule can return one of two results: Allow or Deny.

A Required rule will return Allow if it matches and Deny if it fails. A Deny rule will return Deny if it matches and Allow if it fails. If you choose the Any of the above rules option on the Host Checker policy, the rule will be satisfied if at least one of the rules returns Allow. If you choose All of the above rules, then the rule will be satisfied if all the rules return Allow. In either case, if the result of a compliance rule cannot be completed for some reason, that rule will be considered to have failed.

Some third-party IMVs can also return an Isolate result. When using either the Any of the above rules or All of the above rules option, an Isolate result is treated as Deny. If you wish to treat Isolate results differently, you must choose the Custom option to use the exact results of the rules in determining the final outcome. For example, if you have two rules Sym1 and Mac1, you could use this custom combination of rules:

(Sym1 == Isolate AND Mac1 == NoRecommendation) OR
(Sym1 == NoRecommendation AND Mac1 == Allow)

You can omit the Allow keyword in the custom expression. For example:

rule1 OR rule2

is equivalent to:

rule1 == Allow OR rule2 == Allow
**Enabling Customized Server-side Policies**

For Windows clients, you can create global Host Checker policies which take a third-party J.E.D.I. DLL that you upload to the Infranet Controller and run on client machines.

NOTE: This feature is primarily provided for backwards compatibility. We recommend that you use IMCs and IMVs instead.

**Uploading a Host Checker Policy Package to the Infranet Controller**

For the Infranet Controller to recognize a package definition file, you must:

1. Name the package definition file MANIFEST.HCIF and include it a folder named META-INF.

2. Create a Host Checker policy package by creating a zip archive. The archive should include the META-INF folder that contains the MANIFEST.HCIF file along with the interface DLL and any initialization files. For example, Host Checker policy package might contain:

```
META-INF/MANIFEST.HCIF
hcif-myPestPatrol.dll
hcif-myPestPatrol.ini
```

3. Upload the Host Checker package (or packages) to the Infranet Controller using the instructions in “Enabling Customized Server-side Policies” on page 310. You can upload multiple policy packages to the Infranet Controller, each containing a different MANIFEST.HCIF file.

NOTE: After you upload a Host Checker policy package to the Infranet Controller, you cannot modify the package contents on the server. Instead, you must modify the package on your local system and then upload the modified version to the Infranet Controller.

4. Implement the policy at the realm or role levels using the options described in “Configuring Host Checker Restrictions” on page 313. If you want to verify that the package itself is installed and running on the client computer (as opposed to a specific policy in the package passing or failing) you can use the name you specified when you uploaded the policy package (for example, myPestPatrol). To enforce a particular policy in the package, use the syntax `<PackageName>.<PolicyName>`. For example, to enforce the FileCheck policy in the myPestPatrol package, use myPestPatrol.FileCheck. For instructions, see “Configuring Host Checker Restrictions” on page 313.

To enable a customized server-side Host Checker policy:

1. In the admin console, choose Authentication > Endpoint Security > Host Checker.

3. Enter a name to identify your zip file on the Infranet Controller.

4. Browse to the local directory where your zip file is located.

5. (Optional) Specify remediation instructions and actions for users whose computers do not meet the requirements specified in the policy. For instructions, see “Configuring General Host Checker Remediation” on page 317.

6. Click **Save Changes**. The Infranet Controller adds the policies defined in your zip file to the list of policies on the Host Checker page.

7. Implement the policies at the realm or role levels using the options described in “Configuring Host Checker Restrictions” on page 313.

---

**Implementing Host Checker Policies**

After you create global policies through the **Authentication > Endpoint Security > Host Checker** page of the admin console, you can restrict Infranet Controller and resource access by requiring Host Checker in a:

- **Realm authentication policy**—When administrators or users try to sign in to the Infranet Controller, the Infranet Controller evaluates the specified realm’s authentication policy to determine if the pre-authentication requirements include Host Checker. You can configure a realm authentication policy to download Host Checker, launch Host Checker and enforce Host Checker policies specified for the realm, or not require Host Checker. The user must sign in using a computer that adheres to the Host Checker requirements specified for the realm. If the user’s computer does not meet the requirements, then the Infranet Controller denies access to the user unless you configure remediation actions to help the user bring his computer into compliance. You can configure realm-level restrictions through the **Administrators > Admin Realms > SelectRealm > Authentication Policy > Host Checker** page or the **Users > User Realms > SelectRealm > Authentication Policy > Host Checker** page of the admin console.

- **Role**—When the Infranet Controller determines the list of eligible roles to which it can map an administrator or user, it evaluates each role’s restrictions to determine if the role requires that the user’s computer adheres to certain Host Checker policies. If it does and the user’s computer does not follow the specified Host Checker policies, then the Infranet Controller does not map the user to that role unless you configure remediation actions to help the user bring his computer into compliance. You can configure role-mapping using settings in the **Users > User Realms > SelectRealm > Role Mapping** page. You can configure role-level restrictions through the **Administrators > Admin Roles > SelectRole > General > Restrictions > Host Checker** page of the admin console or the **Users > User Roles > SelectRole > General > Restrictions > Host Checker** page.
**Executing Host Checker Policies**

When the user tries to access the Infranet Controller, Host Checker evaluates its policies in the following order:

1. **Initial evaluation**—When a user first tries to access the Infranet Controller sign-in page, Host Checker performs an initial evaluation. Using the rules you specify in your policies, Host Checker verifies that the client meets your endpoint requirements and returns its results to the Infranet Controller. Host Checker performs an initial evaluation regardless of whether you have implemented Host Checker policies at the realm, role, or resource policy level.

For agentless access deployments, if the user navigates away from the Infranet Controller sign-in page after Host Checker starts running but before signing in to the Infranet Controller, Host Checker continues to run on the user’s machine until the Host Checker process times out. If the Infranet Controller does not receive a result from Host Checker for any reason (including because the user manually terminated Odyssey Access Client or Host Checker), the Infranet Controller displays the remediation instructions if they are enabled, or else displays an error and directs the user back to the sign-in page.

Otherwise, if the Host Checker process returns a result, the Infranet Controller goes on to evaluate the realm level policies.

2. **Realm-level policies**—The Infranet Controller uses the results from Host Checker’s initial evaluation to determine which realms the user may access. Then, the Infranet Controller displays or hides realms from the user, only allowing him to sign into those realms that you enable for the sign-in page, and if he meets the Host Checker requirements for each realm. If the user cannot meet the Host Checker conditions required by any of the available realms, the Infranet Controller does not display the sign-in page. Instead, it displays an error stating the user has no access unless you configure remediation actions to help the user bring his computer into compliance.

Note that Host Checker performs realm-level checks when the user first signs into the Infranet Controller and during the user’s session.

3. **Role-level policies**—After the user signs into a realm, the Infranet Controller evaluates role-level policies and maps the user to the role or roles if he meets the Host Checker requirements for those role(s). Then, the Infranet Controller pushes the role and policy information to the Infranet Enforcer and Odyssey Access Client.

If Host Checker returns a different status during a periodic evaluation, the Infranet Controller dynamically remaps the user to roles based on the new results. If the user loses rights to all available roles during one of the periodic evaluations, the Infranet Controller disconnects the user’s session unless you configure remediation actions to help the user bring his computer into compliance.
4. **Infranet Enforcer resource access policies and Host Enforcer policies**—After the Infranet Controller pushes the role and policy information to the Infranet Enforcer and Odyssey Access Client, the user may try to access a protected resource that is controlled by an Infranet Enforcer resource access policy or Host Enforcer policy. When he does, the Infranet Enforcer or Odyssey Access Client determines whether or not to allow or deny the user access to the protected resource based on the user’s assigned role.

If Host Checker returns a different status during a periodic evaluation, the new status can change the assigned roles. The Infranet Controller then pushes the role and policy information to the Infranet Enforcer and Odyssey Access Client, which could prevent the user from accessing the protected resource.

With either a success or fail, Odyssey Access Client or Host Checker remains on the client. Windows users can manually uninstall Odyssey Access Client by choosing **Odyssey Access Client** in the **Add or Remove Programs** control panel.

If you enable client-side logging through the **System > Log/Monitoring > Client Logs** page, the directory where Odyssey Access Client is installed contains a log file, which the Infranet Controller appends each time Odyssey Access Client or Host Checker runs.

You may specify that the Infranet Controller evaluate your Host Checker policies only when the user first tries to access the realm or role that references the Host Checker policy. Or, you may specify that the Infranet Controller periodically re-evaluate the policies throughout the user’s session. If you choose to periodically evaluate Host Checker policies, the Infranet Controller dynamically maps users to roles and instructs the Infranet Enforcer or Odyssey Access Client to allow users access to new resources based on the most recent evaluation. For more information, see “Executing Host Checker Policies” on page 312.

Use a Host Checker restriction to require client machines to meet the specified Host Checker policies to access an Infranet Controller sign-in page or be mapped to a role.

### Configuring Host Checker Restrictions

To specify Host Checker restrictions:

1. Navigate to: **Authentication > Endpoint Security > Host Checker** and specify global options for Host Checker to apply to any user for whom Host Checker is required in an authentication policy or a role mapping rule.

2. If you want to implement Host Checker at the **realm level**:

   a. Navigate to:
      
      - **Administrators > Admin Realms > Select Realm > Authentication Policy > Host Checker**
      
      - **Users > User Realms > Select Realm > Authentication Policy > Host Checker**
b. Choose one of the following options for either all available policies or for individual policies listed in the **Available Policies** column:

- **Evaluate Policies**—Evaluates without enforcing the policy on the client and allows user-access. This option does not require Host Checker to be installed during the evaluation process; however, Host Checker is installed once the user signs in to the Infranet Controller.

- **Require and Enforce**—Requires and enforces the policy on the client in order for the user to log in to the specified realm. Requires that Host Checker is running the specified Host Checker policies in order for the user to meet the access requirement. Requires the Infranet Controller to download Host Checker to client machines that do not support Odyssey Access Client. (Odyssey Access Client includes Host Checker.) If you choose this option for a realm’s authentication policy, then the Infranet Controller downloads Host Checker to the client machine after the user is authenticated and before the user is mapped to any roles in the system. Selecting this option automatically enables the **Evaluate Policies** option.

c. Select the **Allow access to realm if any ONE of the selected “Require and Enforce” policies is passed** check box if you do not want to require users to meet all of the requirements in all of the selected policies. Instead, the user can access the realm if he meets the requirements of any one of the selected Host Checker policies.

3. If you want to implement Host Checker at the **role level**:

a. Navigate to:

- **Administrators > Admin Roles > Select Role > General > Restrictions > Host Checker**

- **Users > User Roles > Select Role > General > Restrictions > Host Checker**

b. Choose one of the following options:

- **Allow all users** — Does not require Host Checker to be installed in order for the user to meet the access requirement.

- **Allow only users whose workstations meet the requirements specified by these Host Checker policies** — Requires that Host Checker is running the specified Host Checker policies in order for the user to meet the access requirement.

c. Select the **Allow access to role if any ONE of the selected “Require and Enforce” policies is passed** check box if you do not want to require users to meet all of the requirements in all of the selected policies. Instead, the user can access the role if he meets the requirements of any one of the selected Host Checker policies.

4. If you want to create **role-mapping rules** based on a user’s Host Checker status:

a. Navigate to: **Users > User Realms > Select Realm > Role Mapping**.
b. Click New Rule, select Custom Expressions from the Rule based on list, and click Update. Or, to update an existing rule, select it from the When users meet these conditions list.

c. Click Expressions.

d. Write a custom expression for the role mapping rule to evaluate Host Checker’s status using the hostCheckerPolicy variable. For help writing the custom expressions, use tips in the Expressions Dictionary. Or, see “Custom Expressions” on page 517.

ej. In the ...then assign these roles section, select the roles that the Infranet Controller should map users to when they meet the requirements specified in the custom expression and click Add.

f. Select the Stop processing rules when this rule matches if you want the Infranet Controller to stop evaluating role mapping rules if the user successfully meets the requirements defined in this rule.

These options allow you to control which version of an application or service runs on client machines.

---

**Remediating Host Checker Policies**

You can specify general remediation actions that you want Host Checker to take if an endpoint does not meet the requirements of a policy. For example, you can display a remediation page to the user that contains specific instructions and links to resources to help the user bring their endpoint into compliance with Host Checker policy requirements.

You can also choose to include a message to users (called a reason string) that is returned by Host Checker or an integrity measurement verifier (IMV) and explains why the client machine does not meet the Host Checker policy requirements.

For example, the user may see a remediation page that contains the following custom instructions, a link to resources, and reason strings:

**Your computer's security is unsatisfactory.**

Your computer does not meet the following security requirements. Please follow the instructions below to fix these problems. When you are done click Try Again. If you choose to Continue without fixing these problems, you may not have access to all of your intranet servers.

1. **Symantec**

   Instructions: You do not have the latest signature files. [Click here to download the latest signature files.](#)

   Reasons: The AntiVirus Product Version is too low. The age of the Virus Definitions is not acceptable.

For each Host Checker policy, you can configure two types of remediation actions:
- **User-driven**—Using custom instructions and reason strings, you can inform the user about the failed policy and how to make his computer conform. The user must take action to successfully re-evaluate the failed policy unless you configure an IMV to automatically remediate his computer. For instance, you can create a custom page that is linked to a policy server or Web page and enables the user to bring his computer into compliance.

- **Automatic (system-driven)**—You can configure Host Checker to automatically remediate the user’s computer. For example, when the initial policy fails, you can kill processes, delete files, or allow automatic remediation by an antivirus rule (see “Configuring a Predefined Antivirus Rule with Remediation Options” on page 279), a firewall rule (see “Configuring a Predefined Firewall Rule with Remediation Options” on page 281), or a registry setting rule (see “Specifying Customized Requirements Using Custom Rules” on page 286). On Windows, you can also call the `HCIF_Module.Remediate()` API function as part of a third-party J.E.D.I. DLL. Host Checker does not inform users when performing automatic actions. (You could, however, include information in your custom instructions about the automatic actions.)

You can enable these remediation actions in both client-side and server-side policies. For configuration instructions, see “Creating and Configuring New Client-side Policies” on page 276 or “Enabling Customized Server-side Policies” on page 310.

### General Host Checker Remediation User Experience

Users may see the remediation page in the following situations:

- **Before the user signs in:**
  - If you enable custom instructions or reason strings for a policy that fails, the Infranet Controller displays the remediation page to the user. The user has two choices:
    - Take the appropriate actions to make his computer conform to the policy and then click the **Try Again** button on the remediation page. Host Checker checks the user’s computer again for compliance with the policy.
    - Leave his computer in its current state and click the **Continue** button to sign in to the Infranet Controller. He cannot access the realm, role, or resource that requires compliance with the failed policy.

**NOTE:** If you do not configure the Infranet Controller with at least one realm that allows access without enforcing a Host Checker policy, the user must bring his computer into compliance before signing into the Infranet Controller.
If you do not enable custom instructions or reason strings for a policy that fails, Host Checker does not display the remediation page to the user. Instead, a message appears telling the user that no additional information has been provided and to contact the system administrator. The Infranet Controller does not assign the user a role that allows access to protected resources.

After the user signs in:

- **(Odyssey Access Client only)** During a session, if a user’s computer becomes non-compliant with the requirements of a Host Checker policy, a pop-up message appears briefly in the system tray that informs the user of the non-compliance. The user can display the remediation page by right-clicking the Odyssey Access Client icon in the system tray, choosing Odyssey Access Client Manager from the context menu, and then clicking the How do I resolve this problem link in the status section of Odyssey Access Client window.

- **(Agentless—Windows, Macintosh, Linux and Solaris)** During a session, if a user’s agentless computer becomes non-compliant with the requirements of a Host Checker policy, the Infranet Controller displays the remediation page to inform the user of the non-compliance. On Windows agentless computers, Host Checker displays a bubble and tray icon if the endpoint becomes non-compliant. The user must click the bubble or tray icon to open a browser window that contains the remediation instructions. On Macintosh, Linux or Solaris agentless computers, Host Checker automatically opens a browser window that contains the remediation instructions as soon as the endpoint is non-compliant.

**Configuring General Host Checker Remediation**

To specify remediation actions for a Host Checker policy:

1. In the admin console, choose **Authentication > Endpoint Security > Host Checker**.

2. Create or enable Host Checker policies using instructions in either of the following sections:
   - “Creating and Configuring New Client-side Policies” on page 276
   - “Enabling Customized Server-side Policies” on page 310

3. Specify the remediation actions that you want Host Checker to perform if a user’s computer does not meet the requirements of the current policy:

   **Enable Custom Instructions**—Enter the instructions you want to display to the user on the Host Checker remediation page. You can use the following HTML tags to format text and add links to resources such as policy servers or web sites: `<i>`, `<b>`, `<br>`, `<font>`, and `<a href>`. For example:

   You do not have the latest signature files.  
   `<a href="www.company.com">Click here to download the latest signature files.</a>`
- **Kill Processes**—On each line, enter the name of one or more processes you want to kill if the user’s computer does not meet the policy requirements. You can include an optional MD5 checksum for the process. (You cannot use wildcards in the process name.) For example:

  keylogger.exe
  MD5: 6A7DFAF12C3183B56C44E89B12DBEF56

- **Delete Files**—Enter the names of files you want to delete if the user’s computer does not meet the policy requirements. (You cannot use wildcards in the file name.) Enter one file name per line. For example:

  c:\temp\bad-file.txt
  /temp/bad-file.txt

- **Send reason strings**—Select this option to display a message to users (called a reason string) that is returned by Host Checker or integrity measurement verifier (IMV) and explains why the client machine does not meet the Host Checker policy requirements. This option applies to predefined rules, custom rules, and to third-party IMVs that use extensions in the Juniper Networks TNC SDK. For example, an antivirus IMV might display the following reason string:

  The AntiVirus Product Version is too low. The age of the Virus Definitions is not acceptable.

  **NOTE:** By sending reason strings, you are disclosing to users what the IMV is checking on the client machine.

4. Click **Save Changes**.

**Upgrading the Endpoint Security Assessment Plug-In**

The Endpoint Security Assessment Plug-in (ESAP) on the Infranet Controller checks third-party applications on endpoints for compliance with the pre-defined rules you configure in a Host Checker policy. (See “Checking for Third-Party Applications Using Pre-defined Rules (Windows Only)” on page 278.) This plug-in is included in the Infranet Controller system software package.

Juniper Networks frequently adds enhancements, bug fixes, and support for new third-party applications to the plug-in. New plug-in releases are available independently and more frequently than new releases of the Infranet Controller system software package. If necessary, you can upgrade the plug-in on the Infranet Controller independently of upgrading the Infranet Controller system software package.

You can upload up to four versions of the plug-in to the Infranet Controller, but the Infranet Controller uses only one version at a time (called the active version). If necessary, you can rollback to a previously active version of the plug-in.
To upgrade the Endpoint Security Assessment Plug-in:

1. Download the Endpoint Security Assessment Plug-in from the Juniper Networks Customer Support Center to your computer:
   a. Open the following page:
      https://www.juniper.net/customers/csc/software/ive/releases/enterprise_infranet/index.jsp
   b. To access the Customer Support Center, enter a user name and password for a Juniper Networks Support account.
   c. Click the ESAP Download Page link.
   d. Navigate to the ESAP release you want.
   e. Download the plug-in zip file to your computer.

2. Choose Authentication > Endpoint Security > Host Checker.

3. At the bottom of the Host Checker page under Manage Endpoint Security Assessment Plug-In Versions:
   a. If you have previously uploaded four versions of the component software, you must delete one of the versions before you can upload another one. Select the version you want to delete and click Delete.
   b. If you want the Infranet Controller to actively begin using the new component software immediately after you upload it, select the Set as active after upload option.
   c. Click Browse, select the plug-in file you want to upload to the Infranet Controller, and click OK.
   d. Click Upload. While the Infranet Controller uploads and decrypts the plug-in .zip file, the message “Loading...” appears in the plug-in list under Manage Endpoint Security Assessment Plug-In Versions. If the Infranet Controller is a member of a cluster, the Infranet Controller displays the message “Loading...” while the plug-in is transferred to the other cluster nodes. After the plug-in is installed, the date and time of the plug-in installation appears in the plug-in list.

**NOTE:** If the endpoints in your deployment connect to multiple Infranet Controllers simultaneously, all of those connected Infranet Controllers must use the same version of the ESAP plug-in.
e. If you did not select the **Set as active after upload** option, activate the plug-in you want to use by selecting the version in the plug-in list and clicking **Activate**.

---

**NOTE:**

- If you attempt to activate a version of the plug-in that does not support *all* of the pre-defined rules already configured in all Host Checker policies, the Infranet Controller does not allow activation of that plug-in version. For example, if a Host Checker policy is configured to use a pre-defined rule to check for a version of antivirus software, and you attempt to activate a plug-in version that does not support that particular version of the antivirus software, the Infranet Controller does not allow you to activate that plug-in version. To view the list of supported products for a particular plug-in version, click the plug-in’s version number under **Manage Endpoint Security Assessment Plug-In Versions**.

- You can rollback to an older plug-in version after upgrading to a later version by selecting the older version as the active version. But, if you modified any Host Checker policies after upgrading to the later version, the rollback may not succeed. Rollback is guaranteed to succeed only if the policies did not change.

- If you upgrade the Infranet Controller system software to a newer version, or you import a user configuration file, the currently active plug-in version does *not* change. If you want to use a different plug-in version after upgrading or importing a user configuration file, you must manually activate that plug-in version.

- If the Infranet Controller already has four versions of the plug-in installed when you upgrade the Infranet Controller system software to a newer version, the Infranet Controller automatically deletes the oldest plug-in version and installs, but does not activate, the plug-in included with the new Infranet Controller system software.

---

**Specifying General Host Checker Options**

You can specify global options for Host Checker that apply to any user for whom Host Checker is required in an authentication policy or a role mapping rule.

To specify general Host Checker options:

1. In the admin console, choose **Authentication** > **Endpoint Security** > **Host Checker**.
2. Under **Options**:

- In the **Perform check every X minutes** field, specify the interval at which you want Host Checker to perform policy evaluation on a client machine. If the client machine fails to meet the requirements of the Host Checker policies required by a role, then the Infranet Enforcer or Odyssey Access Client denies the associated user requests.

For example, you may require that a user runs a specific third-party antivirus application to map to Role A, which enables the user access to protected resources. If the user’s client machine is running the required antivirus application when the user signs in to the Infranet Controller, then the user maps to Role A and is granted access to the protected resources specified for Role A in an Infranet Enforcer resource access policy. If the antivirus application stops running during the user session, however, the next time Host Checker runs, the user fails to meet the security requirements for Role A and therefore loses all access privileges for Role A.

When an end-user logs into a Realm, Host Checker performs an initial policy check, regardless of whether or not the policy is enforced at the Realm, Role, and/or Resource level. The initial policy check establishes a start time. Host Checker evaluates policies at the frequency set by the **Perform check every X minutes** option starting the clock at the initial policy check. Although the frequency setting is set globally for all Host Checker policy checking, it is not synchronized for all end-user clients connected to the Infranet Controller. Each client performs its own initial policy check and starts its own X minute countdown.

If you configure the authentication policy within a realm where Host Checker enforces policies (versus installing), the enforcement occurs only during the pre-authentication phase. After an end-user signs in and for the duration of the user’s session, any subsequent Host Checker policy checks have no impact on realm access, meaning that there is no concept of removing an end-user session from a realm once an end-user successfully authenticates into that realm.

If you configure a role restriction where Host Checker enforces policies, the enforcement occurs just after authentication during role mapping. Role restrictions are enforced periodically during the end-user session at an interval specified using the Host Checker frequency setting. If the end-user successfully passes the Host Checker evaluation during role mapping but later fails X minutes after login, that specific user loses rights to that role. If the end-user loses rights to all available roles due to Host Checker policy evaluation, the end-user session is disconnected.

---

**NOTE:** If you enter a value of zero, Host Checker only runs on the client machine when the user first signs into the Infranet Controller.

- (Applies to agentless access deployments only) For the **Client-side process, login inactivity timeout** option, specify an interval to control timing out in the following situations.
If the user navigates away from the Infranet Controller sign-in page after Host Checker starts running but before signing in to the Infranet Controller, Host Checker continues to run on the user’s machine for the interval you specify.

If the user is downloading Host Checker over a slow connection, increase the interval to allow enough time for the download to complete.

3. Click Save Changes.

4. Implement the policy at the realm or role levels using the options described in “Configuring Host Checker Restrictions” on page 313. If you want to verify that the package itself is installed and running on the client computer (as opposed to a specific policy in the package passing or failing) you can use the name you specified when you uploaded the policy package (for example, myPestPatrol). To enforce a particular policy in the package, use the syntax <PackageName>.<PolicyName>. For example, to enforce the FileCheck policy in the myPestPatrol package, use myPestPatrol.FileCheck. For instructions, see “Configuring Host Checker Restrictions” on page 313.

---

**Specifying Odyssey Access Client and Host Checker Installation Options**

You can use the following methods to install Odyssey Access Client or Host Checker on a user’s system:

- **Automatic installation:**

  - **Odyssey Access Client**—If you enable Odyssey Access Client installation (see “Creating an initial Configuration of Odyssey Access Client” on page 44), the first time the user accesses the Infranet Controller using a Web browser, the Infranet Controller automatically downloads agentless Host Checker to evaluate any Host Checker restrictions for role mapping. After the initial check is successfully completed, the Infranet Controller downloads Odyssey Access Client with its built-in Host Checker component to the user’s computer. If you evaluate or enforce a Host Checker policy at the realm level, Odyssey Access Client automatically runs its built-in Host Checker on the endpoint to check it for security compliance.

  - **Agentless and Java agent**—For agentless or Java agent deployments (see “Configuring Agentless Access to Protected Resources” on page 56 or “Configuring the Java Agent for Endpoint Access” on page 58), the user signs into the Infranet Controller directly using a Web browser instead of Odyssey Access Client. If you evaluate or enforce a Host Checker policy at the realm level (see “Configuring Host Checker Restrictions” on page 313), the Infranet Controller automatically installs and runs Host Checker on the endpoint to check it for security compliance.
If you evaluate or enforce a Host Checker policy for either Odyssey Access Client, the Java agent or agentless deployments, the Infranet Controller evaluates the realm-level option when the user connects to the Infranet Controller or accesses the sign-in page and then determines if the current version of Odyssey Access Client or Host Checker is installed on the user’s machine. If Odyssey Access Client or Host Checker is not installed, the Infranet Controller attempts to install it using either an ActiveX or a Java delivery method:

- When a Windows user first signs in to the Infranet Controller, the Infranet Controller attempts to install an ActiveX control on the user’s system. If the Infranet Controller successfully installs the ActiveX control, the control manages the installation of Odyssey Access Client or Host Checker in agentless access deployments. If the Infranet Controller cannot install the ActiveX control because ActiveX is turned off on the user’s system, the Infranet Controller attempts to install Odyssey Access Client or Host Checker using Java.

- For Macintosh, Linux and Solaris hosts, the Infranet Controller always uses the Java delivery method. The Java delivery method requires only user privileges, but Java must be enabled on the user’s system. For the Firefox browser on Linux, the Java runtime and plug-in must be installed.

- If the Infranet Controller cannot use the Java delivery method because Java is disabled on the user’s system, the Host Checker policy will fail and may cause restrictions on realms or roles where the policy is evaluated. If no other realms or roles are available to the user, the Infranet Controller displays a no-access error message.

- The user or administrator manually installs Odyssey Access Client or Host Checker (Windows only)—Download Odyssey Access Client installer or the Host Checker installer from the Maintenance > System > Installers page of the admin console and use it to manually install Odyssey Access Client or Host Checker on the user’s system. (For configuration instructions, see “Downloading Application Installers” on page 344.)

**NOTE:** To install Odyssey Access Client or Host Checker, users must have appropriate privileges, as described in the *Client-side Changes Guide* on the Juniper Networks Support site.

### Automatically install Host Checker

For agentless access deployments only, you can configure the Infranet Controller to automatically install Host Checker on client computers.

To automatically install Host Checker on client computers:

1. In the admin console, choose **Authentication > Endpoint Security > Host Checker**.
2. Under **Options**, select **Auto-upgrade Host Checker** if you want the Infranet Controller to automatically download the Host Checker application to a client computer when the version of Host Checker on the Infranet Controller is newer than the version installed on the client. Here is a summary of what happens when the **Auto-upgrade Host Checker** option is selected or not selected:

- If Host Checker is not installed on the client computer, Host Checker is installed automatically regardless of whether the **Auto-upgrade Host Checker** option is selected or not selected.

- If the **Auto-upgrade Host Checker** option is selected and a previous version of Host Checker is installed, Host Checker is upgraded on the client automatically.

- If the **Auto-upgrade Host Checker** option is not selected and a previous version of Host Checker is installed, Host Checker is not upgraded on the client automatically.

If you select the **Auto-upgrade Host Checker** option, note the following:

- On Windows, the user must have administrator privileges in order for the Infranet Controller to automatically install the Host Checker application on the client. For more information, see the *Client-side Changes Guide* on the Juniper Networks Customer Support Center.

- If a user uninstalls Host Checker and then signs in to an Infranet Controller for which the **Auto-upgrade Host Checker** option is not enabled, the user no longer has access to Host Checker.

3. Click **Save Changes**.

**Manually install Host Checker**

The **Maintenance > System > Installers** page of the admin console provides Odyssey Access Client and Host Checker client applications. You can download the applications or service as a Windows executable file, which enables you to:

- Distribute the file to client machines using software distribution tools. This option enables you to install an application or service on client machines whose users do not have Administrator privileges, which are required to install the application or service.

- Post the executable in a secure repository so that users with the proper administrator right may download and install the appropriate version.

- Download and execute a script that automatically retrieves the proper version of the installer from an FTP server.
Using Host Checker Logs

Use the System > Log/Monitoring > Client Logs > Settings tab to enable client-side logging for the Host Checker and Odyssey Access Client. When you enable this option, the Infranet Controller writes a client-side log to endpoints. The Infranet Controller appends to the log file during user session. This feature is useful when working with the support team to debug problems with the respective feature.

**NOTE:** Since these settings are global, the Infranet Controller writes a log file to all clients that use the feature for which you enable client-side logging. Also, the Infranet Controller does not remove client-side logs. Users need to manually delete log files from their clients. For information about where the Infranet Controller installs log files, see the Client-side Changes Guide on the Juniper Networks Customer Support Center.

To specify global client-side logging settings:

1. In the admin console, choose System > Log/Monitoring > Client Log > Settings.

2. Select the desired features for which the Infranet Controller writes client-side logs.

3. Click Save Changes to save these settings globally.

4. 
Part 4
System Management

This section contains the following information about managing your Infranet Controller system:

- “General System Management” on page 329
- “Certificates” on page 355
- “System Archiving” on page 381
- “Logging and Monitoring” on page 415
- “Troubleshooting” on page 437
- “Clustering” on page 449
- “Delegating Administrator Roles” on page 473
- “Infranet Controller and IDP Interoperability” on page 485
Chapter 10
General System Management

The Infranet Controller provides a number of features that allow you to maintain your system easily. You can define, modify, and monitor system management capabilities, such as:

- “Configuring Network Settings” on page 329
- “Using Central Management Features” on page 339
- “Configuring System Utilities” on page 341
- “Configuring Infranet Controller Licensing and Security” on page 345

Configuring Network Settings

The Infranet Controller enables you to modify the network settings that you enter through the serial console during your initial Infranet Controller configuration as well as configure additional network settings such IP filters in order to enable other Infranet Controller features. This section provides the following overviews of network settings that you can set through the admin console:

- “Configuring General Network Settings” on page 330
- “Configuring Internal and External Ports” on page 331
- “Configuring VLANs” on page 333
- “Configuring Virtual Ports” on page 335
- “Configuring Static Routes for Network Traffic” on page 336
- “Creating ARP Caches” on page 337
- “Specifying Host Names for the Infranet Controller to Resolve Locally” on page 337
- “Configuring the Infranet Controller to Work with a Load Balancer” on page 338
**Configuring General Network Settings**

The Infranet Controller enables you to view the status of the internal and external ports, specify a host name for the Infranet Controller, and configure DNS name resolution and a Windows Internet Naming Service (WINS) server for the Infranet Controller through settings in the System > Network > Overview page in the admin console. You may also use settings in this page to view the DNS and WINS settings that you entered through the serial console during initial configuration.

Use settings in this tab to configure general network settings. The Status area displays read-only status of the following items:

- **Node Name**—Name of the current node, if running a cluster.
- **Internal Port**—The status and speed of the internal port.
- **External Port**—The status and speed of the external port, if used.

To configure general network settings:

1. In the admin console, choose System > Network > Overview.
2. Under **Network Identity**, enter the fully-qualified host name of the Infranet Controller.

**NOTE:**

- The host name that you enter in this field cannot exceed 30 characters.
- If your Infranet Controller appliances are clustered, the network identity host name that you specify is synchronized across the cluster. In multi-site clusters, however, we recommend that you override this setting and specify different host names for the individual nodes in the cluster using options in the System > Clustering page.

3. Under **DNS Name Resolution**, update the primary DNS address, secondary DNS address, and default DNS domain name for the individual Infranet Controller appliance.

   You may enter a comma delimited list of DNS domains in these fields; the Infranet Controller searches for them in the order that they are listed.

4. Under **Windows Networking**, enter the name or IP address of a local or remote the Windows Internet Naming Service (WINS) server that you use to associate workstation names and locations with IP addresses (if applicable).

5. Click **Save Changes**.
Configuring Internal and External Ports

The internal port, also known as the internal interface, handles all LAN requests and listens for authentication requests. Use the internal port for connections to the Infranet Enforcer and authentication servers. You configure the internal port by providing IP address, gateway, DNS server and domain, and MTU settings during the initial setup of the Infranet Controller. You can also change them on the System > Network > Internal Port > Settings tab. For more information, see “Configuring Internal and External Ports” on page 331.

You can use the external port to connect the Infranet Controller to a different VLAN (for example a quarantine VLAN) for Odyssey Access Client endpoints. The external port should not be used for the Infranet Enforcer or Layer 2 switch connections.

The routes that you specify for each interface apply after the Infranet Controller determines whether to use the internal or external interface. The Infranet Controller does not initiate requests from the external interface, and this interface does not accept any other connections (except ping, traceroute, and NACN connections). The Infranet Controller initiates all requests from the internal interface. (For more information, see “Configuring General Network Settings” on page 330.)

To modify network settings for the internal port (LAN interface):

1. In the admin console, choose System > Network > Internal Port > Settings.

   **NOTE:** Most fields on this page are pre-populated with the settings specified during Infranet Controller installation.

2. In the Port Information section, update the IP address, netmask, and default gateway settings for the individual Infranet Controller appliance. By default, these fields are populated with the settings entered during initial Infranet Controller setup.

3. In the Link Speed field, specify the speed and duplex combination you want to use for the internal port.
4. In the **ARP Ping Timeout** field, specify how long the Infranet Controller should wait for responses to Address Resolution Protocol (ARP) requests before timing out. Clustered Infranet Controller appliances send ARP requests\(^1\) to the gateways of other Infranet Controller appliances in order to determine if they are properly communicating with one another.

**NOTE:** If you are not running the Infranet Controller in a clustered environment, the Infranet Controller does not use this setting. If your Infranet Controller systems are clustered, the timeout interval that you specify is synchronized across the cluster. In multi-site clusters, you can override this setting for the individual nodes in the cluster using options in the **System > Clustering** page. Use caution when changing this setting in active/passive clusters, however, because the Infranet Controller also uses the **ARP Ping Timeout** setting on the **Internal** tab as a failover timer for the VIP.

5. In the **MTU** field, specify a maximum transmission unit for the Infranet Controller’s internal interface.

**NOTE:** Use the default MTU setting (1500) unless you must change the setting for troubleshooting purposes.

6. Click **Save Changes**.

To enable the external port:

1. In the admin console, choose **System > Network > Port 1 > Settings**.

2. Under **Use Port**, select **Enable**.

3. In the **Port Information** section, enter the IP address, netmask, and default gateway information for the external port of the Infranet Controller. Typically, you should use the settings from the **Internal Port > Settings** page and then change the internal port information to a local IP address, netmask, and gateway.

4. In the **Link Speed** field, specify the speed and duplex combination you want to use for the external port.

---

\(^1\) The Infranet Controller makes two ARP requests—one to the gateway on the internal port and another to the gateway on the external port—when trying to establish communications in the cluster.
5. In the **ARP Ping Timeout** field, specify how long the Infranet Controller should wait for responses to Address Resolution Protocol (ARP) requests before timing out. Clustered Infranet Controller appliances send ARP requests\(^1\) to the gateways of other Infranet Controller appliances in order to determine if they are properly communicating with one another.

---

### NOTE:
If your Infranet Controllers are clustered, the timeout interval that you specify is synchronized across the cluster. In multi-site clusters, however, you can override this setting for the individual nodes in the cluster using options in the **System > Clustering** page. If you are not running the Infranet Controller in a clustered environment, the Infranet Controller does not use the **ARP Ping Timeout** setting.

6. In the **MTU** field, specify a maximum transmission unit for the Infranet Controller’s external interface.

---

### NOTE:
Use the default MTU setting (1500) unless you must change the setting for troubleshooting purposes.

7. Click **Save Changes**.

8. Choose

**Configuring VLANs**

You can configure the Infranet Controller to take advantage of VLAN tagging. When connected to a trunk port on a VLAN-enabled switch, the Infranet Controller sees traffic from all VLANs. This is useful if you want to configure separate VLANs for separate classes of users or endpoints, and you want to make the Infranet Controller accessible from all VLANs. For more information on using VLANs in a UAC deployment, see “Configuring RADIUS Attributes Policies” on page 151.

You must define a VLAN port for each VLAN. You assign the specific VLAN ID when defining the VLAN port.

For each VLAN you configure, the virtualized Infranet Controller provisions a unique, logical VLAN interface, or port, on the internal interface. There is no relationship between the internal port IP address and any VLAN port IP address. Each VLAN port has its own route table.

The Internal Port must be assigned to the root system and marked as the default VLAN. Additionally, VLAN interfaces can be assigned to the root system. All authentication servers configured for the root system, however, must have routes in the Internal Port’s route table, even if the servers are reachable via VLAN interfaces. Routes to servers reachable via VLAN interfaces must have the next-hop gateway set to the configured gateway for the VLAN interface, and the output port defined as the VLAN port.

---

\(^1\) The Infranet Controller makes two ARP requests—one to the gateway on the internal port and another to the gateway on the external port—when trying to establish communications in the cluster.
For an Active/Passive clustered deployment, the root admin of an MSP network should configure all VLAN ports with at least one virtual port (*System > Network > VLANs > Virtual Ports*). The router administrator must configure routes for the IVS Network Connect IP ranges that point to the VLAN virtual port’s IP address as the next-hop gateway. This is required for Network Connect session failover from an IVS in the Active Node to the corresponding IVS in the Passive Node.

Each VLAN port definition consists of:

- **Port Name.** Must be unique across all VLAN ports that you define on the system or cluster.
- **VLAN ID.** An integer in the range from 1 to 4094 that uniquely identifies the VLAN.
- **IP Address/Netmask.** Must be an IP address or netmask from the same network as the VLAN. VLAN IP addresses must be unique. You cannot configure a VLAN to have the same network as the internal port. For example, if the internal port is 10.64.4.30/16 and you configure a VLAN as 10.64.3.30/16, you may get unpredictable results and errors.

**NOTE:** If you are using 802.1X enforcement, be sure to read the VLAN configuration information in “Using a RADIUS Attributes Policy to Specify VLANs for Endpoints” on page 152. The information above for **IP Address/Netmask** applies to non 802.1X deployments.

- **Default gateway.** The IP address of the default router for the VLAN.
- **Other network settings.** Inherited from the internal port.

When you create a new VLAN port, the system creates two static routes, by default:

- The default route for the VLAN, pointing to the default gateway.
- The interface route to the directly connected network.

To create a VLAN port, perform the following steps:

1. Select **System > Network > VLANs** to open the **VLAN Network Settings** tab.
2. Click **New Port.** If you are running a cluster, you must create the VLAN for the entire cluster, not just for a single node.
3. Under **VLAN settings,** enter a name for the VLAN port.
4. Enter a VLAN ID.

**NOTE:** The VLAN ID must be between 1 and 4094 and must be unique on the system.

5. Enter the IP address for the VLAN.
6. Enter a netmask for the VLAN.
7. Enter a default gateway for the VLAN.

8. Click **Save Changes**.

**Configuring Virtual Ports**

The Infranet Controller enables you to create virtual ports and associate different device certificates with each virtual port. A **virtual port** activates an IP alias on a physical port and shares all of the network settings (except IP address) with the internal or external port that hosts the virtual port. An **IP alias** is an IP address that is bound to a virtual port. (Note that an IP alias is different from the Infranet Controller’s primary IP address, which is a required Infranet Controller setting that you configure during the Infranet Controller initialization process.)

You can use virtual ports to provide different groups of users access to the same Infranet Controller using different IP aliases, domains, and certificates. For more information, see “Associating a Certificate With a Virtual Port” on page 362.

To create a virtual port for a stand-alone Infranet Controller:

1. In the admin console, choose **System > Network > Port Tab > Virtual Ports**.

   The **Port Tab** could be for the internal or external ports.

2. Click **New Port**.

3. Enter a unique name for the virtual port.

4. Enter a unique IP alias to associate with the virtual port—do not use an IP address that is already associated with another virtual port.

   **NOTE:** If you do not enter an IP address, the Infranet Controller does not activate the virtual port.

5. Click **Save Changes**.

   **NOTE:** If you need to associate the virtual port with a device certificate, use settings in the **System > Configuration > Certificates > Device Certificates** tab. For more information, see “Associating a Certificate With a Virtual Port” on page 362.

To create a virtual port on a cluster node:

1. In the admin console, choose **System > Network > Port Tab > Virtual Ports**.

   The **Port Tab** could be for the internal or external ports.

2. From the **Settings for** drop-down list, select **Entire cluster** and then click **Update**.

3. Click **New Port**.
4. Enter a unique name for the virtual port and then click **Save Changes**. The Infranet Controller adds the virtual port name to the **Virtual Ports** list and provides access to each node in the cluster.

5. Click the link to a node to access the IP address configuration page. Enter a unique IP alias to associate with the virtual port—do not use an IP address that is already associated with another virtual port.

---

**NOTE:** If you do not enter an IP address, the Infranet Controller does not activate the virtual port.

6. Click **Save Changes**. The **Virtual Ports** page returns to the virtual port tab. If necessary, re-select **Entire cluster** from the **Settings for** drop-down list and then repeat the last 2 steps of this procedure.

For more information about using virtual ports in clusters, see “Deploying Two Nodes in an Active/Passive Cluster” on page 455.

**Task summary: Associating Certificates With Virtual Ports**

To associate certificates with virtual ports:

1. Use settings in the **System > Network > Internal Port** tab to create virtual ports. For instructions, see “Configuring Virtual Ports” on page 335.

2. Use settings in the **System > Configuration > Certificates > Device Certificates** page of the admin console to import the server certificates that you want to use to validate user certificates. Also use this tab to specify which ports the Infranet Controller should associate with the certificates. For instructions, see “Associating a Certificate With a Virtual Port” on page 362.

---

**Configuring Static Routes for Network Traffic**

The Infranet Controller enables you to add routing table entries using settings in the **System > Network > Routes** tab. All connection requests come from the Infranet Controller internal port regardless of route settings. The route settings on the external port are used only to route packets associated with connections on the network connected to the external port.

You can add static routes, if you want to indicate a specific route that the Infranet Controller should use when routing requests. You need to specify a valid IP address, gateway, and DNS address. The metric is a way of comparing multiple routes to establish precedence. Generally, the lower the number, from 1 to 15, the higher the precedence. So, a route with a metric of 2 would be chosen over a route with a metric of 14. The metric value of zero (0) identifies the route as one that should not be used.

To specify static routes for network traffic:

1. In the admin console, choose **System > Network > Routes**.

2. Select a destination route table from the **View route table for:** drop down menu.
3. Click **New Route**.

4. Enter the required information.

5. Click **Add to [destination] route table**.

Destination route tables may be available for the Internal port or External port.

**Creating ARP Caches**

You can use ARP caching to determine the physical (MAC) address of a network device such as a router that connects to the Infranet Controller. Use the **System > Network > Internal Port > ARP Cache** tab to manage the following types of ARP (address resolution protocol) entries:

- **Static entries**—You can add a static ARP entry to the cache associated with the IP and MAC address. The Infranet Controller stores static entries during reboots and re-activates them after re-booting. Static entries are always present on the Infranet Controller.

- **Dynamic entries**—The network “learns” dynamic ARP entries during normal use and interaction with other network devices. The Infranet Controller caches dynamic entries for up to 20 minutes and deletes them during a reboot or when you manually delete them.

You can view and delete static and dynamic entries from the ARP cache as well as add static entries. If you have a cluster of Infranet Controllers, note that ARP cache information is node-specific. The Infranet Controller only synchronizes ARP cache information across non-multi-site clusters.

To add a static entry to the ARP Cache:

1. In the admin console, choose **System > Network > Port Tab > ARP Cache**.

   The **Port Tab** could be for the internal or external ports.

2. Enter the **IP Address** and the **Physical Address** in their respective fields at the top of the table.

   **NOTE:** If you add an entry containing an existing IP address, the Infranet Controller overwrites the existing entry with your new entry. Also note that the Infranet Controller does not verify the validity of MAC addresses.

3. Click **Add**.

**Specifying Host Names for the Infranet Controller to Resolve Locally**

Specify host names that the Infranet Controller can resolve to IP addresses locally by using the **Hosts** tab. This feature is useful when:

- Your DNS server is not accessible to the Infranet Controller.
- You use WINS to access servers within the LAN.
Your corporate security policy does not allow internal servers to be listed on an external DNS or requires that internal host names are masked.

To specify host names for the Infranet Controller to resolve locally:

1. In the admin console, choose the **System > Network > Hosts** tab.
2. Enter an IP address, a comma delimited list of host names that resolve to the IP address, a comment of 200 words or less (optional), and then click **Add**.

### Configuring the Infranet Controller to Work with a Load Balancer

You can deploy a load balancer between several Infranet Controllers with or without clustering. You can use Juniper's DX appliance as the load balancer.

As part of a layer 2 authentication, the Infranet Controller sends Odyssey Access Client the IP address to connect to for the NCP control channel. If you are using an Infranet Enforcer, the IC sends a keepalive message to the Infranet Enforcer. The Infranet Enforcer uses that IP address to identify the Infranet Controller config.

If you deploy a load balancer, you must provide the load balancer IP address to which agents connect. The IP address of the load balancer is substituted for the IP addresses of the Infranet Controller.

Additionally, you must specify whether the load balancer is used between endpoints and the IC, or between endpoints and the Infranet Controller.

To configure the Infranet Controller to work with a load balancer:

1. In the admin console, choose the **System > Network > Load Balancer** tab.
2. If a load balancer is connected to the internal interface of the Infranet Controller, enter the IP address of the load balancer in the **Internal Address** field.
3. If a load balancer is connected to the external interface of the Infranet Controller, enter the IP address of the load balancer in the **External Address** field.
4. If you are deploying a load balancer between endpoints and the Infranet Controller, select the **Between endpoints and the** Infranet Controller check box.
5. If you are deploying a load balancer between an Infranet Enforcer and the Infranet Controller, select the **Between Infranet Enforcer and** Infranet Controller check box.
6. Click **Save Changes**.
Using Central Management Features

The following Central management features consist of a two-tier (client/server) system that enables you to manage multiple Infranet Controllers, regardless of whether or not they are clustered:

- **System dashboard**—The system dashboard feature displays system capacity graphs and alarms that allow you to easily monitor the system. You can access the system dashboard from the **System > Status** page of the admin console.

- **Improved logging and monitoring**—The logging feature enables you to create custom filters so that you may view and save only those log messages that you choose in the format of your choice. You can access the logging feature from the **System > Log/Monitoring** page of the admin console.

- **Minimal downtime upgrades**—The minimal downtime upgrade feature enables you to expedite upgrades across a cluster, ensuring that one cluster member is always functional during the upgrade process. You can access the upgrade feature from the **Maintenance > System > Upgrade/Downgrade** page of the admin console.

- **Deterministic cluster recovery**—The deterministic cluster recovery feature enables you to assign ranks to various nodes in a cluster such that when a cluster recovers from “split-brain” situation, the node with the highest rank propagates the correct cluster state.

- **Improved SNMP MIB**—The enhanced SNMP MIB feature provides you with capacity utilization metrics. You can download the improved MIB from the **System > Log/Monitoring > SNMP** page of the admin console.

- **Saving local backups**—The local backups feature enables you to save up to 10 local backups of system and user configuration files on the Infranet Controller. You can access this feature through the **Maintenance > Archiving > Local Backups** page of the admin console.

### Modifying Central Management Dashboard Graphs

The system dashboard appears when you open the administrator console. The dashboard displays system capacity graphs that allow you to easily monitor your system.

If you want to analyze or display the information from these graphs using your own tools, you may use the graph download feature. From the system dashboard, you may download the data from each of the graphs into an XML file. Then, you may use your own tools to reformat or analyze the data in the XML files.

### Central Management Dashboard Graphs XML Schemas

The XML files for all of the system dashboard graphs contain the same basic XML elements:

- `<xport>`—Top level element.
- `<meta>`—Second level element.
- `<start>`—Time that the system collected the first data point for the graph, in UTC format.
- `<step>`—Interval at which the system collected data points for the graph, in seconds. For example, the following XML entry indicates that the system collects data every minute: `<step>60</step>`
- `<end>`—Time that the system collected the last data point for the graph, in UTC format.
- `<rows>`—Number of data points collected for the graph.
- `<columns>`—Number of metrics collected for the graph. (Corresponds to the number of lines displayed in the graph in the administrator console.)
- `<legend>`—Contains a list of `<entry>` sub-elements that define the names of each of the metrics collected for the graphs. For example, sub-elements for the Concurrent Users graph may include:
  ```xml
  <legend>
    <entry>Local users</entry>
    <entry>Concurrent users</entry>
  </legend>
  ```
- `<data>`—Contains a list of `<row>` sub-elements that include the periodic data collected for each entry. Each `<row>` element contains a `<t>` sub-element that includes the time that the system collected the data and `<v>` sub-elements for each piece of data. For example, a row within the Concurrent Users graph may include:
  ```xml
  <data>
    <row>
      <t>1089748980</t><v>2.1000000000e+01</v><v>2.1000000000e+01</v>
    </row>
  </data>
  ```

**Sample XML Schema**

The following sample shows the XML output for a Concurrent Users graph. (Note that for the purposes of brevity, some of the original `<row>` elements have been deleted from the sample.)

```xml
<xport>
  <meta>
    <start>1089748980</start>
    <step>60</step>
    <end>1089763440</end>
    <rows>242</rows>
    <columns>2</columns>
    <legend>
      <entry>Local users</entry>
      <entry>Concurrent users</entry>
    </legend>
  </meta>
  <data>
    <row>
      <t>1089748980</t><v>2.1000000000e+01</v><v>2.1000000000e+01</v>
    </row>
  </data>
</xport>
```
 Configuring System Utilities

The Infranet Controller system utilities enable you to manage your server, upgrade or downgrade system software, enable version monitoring, and to download client applications and services. For more information refer to the following topics:

- “Reviewing System Data” on page 341
- “Upgrading or Downgrading the Infranet Controller” on page 343
- “Setting System Options” on page 343
- “Downloading Application Installers” on page 344

Reviewing System Data

The Maintenance > System > Platform page in the admin console lists Infranet Controller system data and contains controls for restarting, rebooting, or shutting down an Infranet Controller. It also contains a control to test server connectivity. When the Infranet Controller is a member of a cluster, this page lists additional, cluster-specific, system data and the controls operate on a cluster-wide basis.

Restarting, Rebooting, Shutting Down, or Testing Server Connectivity

The Maintenance > System > Platform page lists the following system data for an Infranet Controller:

- **Model**—Displays the model of the Infranet Controller
- **Version**—Displays the Infranet Controller’s software version
- **Rollback**—Displays the software version to which the Infranet Controller reverts when you roll back the installed image.

- **Last Reboot**—Displays amount of time since the Infranet Controller’s last reboot.

When the Infranet Controller is a member of a cluster, the **Platform** page lists the following additional system data:

- **Cluster**—Displays the name of the cluster to which the Infranet Controller belongs.

- **Member**—Displays the Infranet Controller’s cluster member name.

The **Platform** page contains the following controls:

- **Restart Services/Cluster**—Kills all processes and restarts the Infranet Controller. When the Infranet Controller is a member of a cluster, this control kills all processes and restarts all members of a cluster.

- **Reboot**—Power cycles and reboots the Infranet Controller. When the Infranet Controller is a member of a cluster, this control power cycles and reboots all members of the cluster.

- **Shut down**—Shuts down the Infranet Controller, requiring you to press the reset button on the appliance to restart the server. When the Infranet Controller is a member of a cluster, this control shuts down all members of a cluster, requiring you to press the reset button on all clustered appliances to restart the cluster. Note that the machine power remains on after a server shutdown.

**NOTE:** If you want to restart, reboot, or shut down, or upgrade one Infranet Controller in a cluster, you first disable the Infranet Controller using the controls on the **System > Clustering > Status** page, and then return to the **Platform** page to employ the control of your choice.

- **Rollback**—Rolls back the software image and reboots the Infranet Controller. After the Infranet Controller reboots, the image on the Infranet Controller is automatically rolled back to the image displayed in the **Rollback** field, above.

- **Test Connectivity**—Sends an ICMP ping from the Infranet Controller to all the servers the Infranet Controller is configured to use and report their connectivity. Each server’s status is reported under **Server Connectivity Results**.

**NOTE:** If you are looking for information on performing a factory reset or rolling the system back to the previous state, please see “Infranet Controller Serial Console” on page 497.
Upgrading or Downgrading the Infranet Controller

You can install a different service package by first obtaining the software from the Juniper Support Web site and then uploading it through the admin console. Package files are encrypted and signed so that the Infranet Controller server accepts only valid packages issued by Juniper Networks. This measure prevents the Infranet Controller server from accepting Trojan horse programs.

NOTE: If you are upgrading to UAC version 2.1 from any earlier version, note that location groups and RADIUS client configuration have been moved. To preserve all system settings, you must first import the system configuration (system.cfg) followed by the user configuration (user.cfg).

This feature is typically used to upgrade to newer versions of the system software, but you can also use this process to downgrade to a previous version or to delete all your current configuration settings and start from a "clean slate." You may also roll back to a previous system state through the serial console, as described in "Rolling Back to a Previous System State" on page 498.

You can upgrade or downgrade the Infranet Controller from the Maintenance > System > Upgrade/Downgrade page of the admin console.

NOTE: Installing a service package can take several minutes and requires the Infranet Controller to reboot. Because existing system data is backed up during this process, you can decrease installation time by clearing your system log before trying to install a service package.

Setting System Options

You can set a number of system options, such as:

- **Version monitoring**—Keep your system current and secure by having the Infranet Controller notify you about critical software patches and updates. To do this, it reports to Juniper Networks the following data: your company name, an MD5 hash of your license settings, and information describing the current software version.

- **gzip compression**—Reduce download speeds when using HTTP compression-enabled browsers.

To set system options:

1. In the admin console, choose Maintenance > System > Options.

2. Select the Automatic Version Monitoring check box to automatically receive notifications of critical software patches and updates.

NOTE: For your protection, we strongly recommend that you enable this automatic service, but if necessary, you can disable it.

3. Select the Enable gzip compression check box to reduce the amount of data sent to browsers that support HTTP compression.
4. Click **Save Changes**.

**Downloading Application Installers**

You can download an application or service as a Windows executable file, which enables you to:

- Distribute the file to client machines using software distribution tools. This option enables you to install an application or service on client machines whose users do not have Administrator privileges, which are required to install the application or service.

- Post the executable in a secure repository so that users with the proper administrator right may download and install the appropriate version.

- Download and execute a script that automatically retrieves the proper version of the installer from an FTP server.

These options allow you to control which version of an application or service runs on client machines.

The **Installers** page contains the following controls:

- **Host Checker**—This installer (**HCInst.exe**) installs Host Checker on users’ systems. Host Checker is a client-side agent that performs endpoint security checks on hosts that connect to the Infranet Controller.

  **NOTE:** If you decide to distribute Host Checker, make sure to uncheck the **Auto-upgrade Host Checker** option on the **Authentication > Endpoint Security > Host Checker** page (see “Specifying General Host Checker Options” on page 320). Otherwise the Infranet Controller downloads the Host Checker application to a user’s machine, which may not be the same version as the distributed version.

- **Third-party Integrity Measurement Verifier (IMV) Server**—This installer (**RemoteIMVServerInstall.exe**) installs IMVs on users’ systems. IMVs are software modules running on the Infranet Controller that are responsible for verifying a particular aspect of an endpoint’s integrity.

- **UAC Agent**—This installer (**OdysseyAccessClient.msi**) installs Odyssey Access Client (for Windows endpoints only). Odyssey Access Client is a client-side application that performs endpoint security checks on hosts that connect to the Infranet Controller and can protect endpoints from attacks. (See “How the UAC Components Work Together” on page 8.)

This installer also includes the Juniper Unified Network Service. The Juniper Unified Network Service allows users to download, install, upgrade, and run client-side applications without administrator privileges. In order to perform these tasks (which require administrator privileges), the Juniper Unified Network Service runs under the client’s Local System account (a powerful account with full access to the system) and registers itself with Windows’ Service Control Manager (SCM). An Active-X control or a Java applet running inside the user’s Web browser communicates the details of the installation processes to be performed through a secure channel between the Infranet Controller and the client system.
To download an application or service:

1. In the admin console, choose Maintenance > System > Installers.

2. Click on the Download link to the right of the application or service you want to download. The File Download dialog box appears.

3. Click the Save button on the File Download dialog box. The Save As dialog box appears.

4. Choose an appropriate location in the Save As dialog box.

5. Click the Save button on the Save As dialog box.

**NOTE:** When installing Odyssey Access Client on client systems, note that:

- To be installed, the Odyssey Access Client requires the user to either have Administrator privileges, or the Juniper Unified Network Service must be pre-installed on the endpoint. For additional information, see the *Client-side Changes Guide* on the Juniper Networks Customer Support Center.

- The service appears in the Windows Services (Local) list as Juniper Unified Network Service.

- The service starts automatically on install and during client system start up.

- Once Odyssey Access Client is installed, users with restricted privileges can upgrade seamlessly.

---

**Configuring Infranet Controller Licensing and Security**

Use the System > Configuration pages to apply your initial license or upgrade your license, to set default security options, to configure NCP or JCP communication protocols, as described in the following topics:

- “Entering or Upgrading Infranet Controller Licenses” on page 346—Use this feature to enter a new license, to add options to your existing license, or to upgrade a license to a new release.

- “Click the Enable link in the right side of the license column to activate emergency mode or click Disable to deactivate it.Setting Security Options” on page 350—Use this feature to set all security options, including system-wide options, cookies, intermediation, and more.

- “Installing a Juniper Software Service Package” on page 352—Use this feature to install a new service package to the Infranet Controller.
**Entering or Upgrading Infranet Controller Licenses**

The Infranet Controller appliance ships with a license that allows you basic access to the Infranet Controller. To take full advantage of your appliance, however, you must access the Juniper Networks License Management System, provide your Licensing Hardware ID and Authorization Code(s) to obtain your license keys, and sign in to the administrator admin console to enter the license keys you receive from Juniper Networks.

Separate licenses that allows you to take advantage of Infranet Controller and Intrusion Detection and Prevention (IDP) interoperability and Statement of Health (SOH) functionality are available from Juniper Networks. See “Licensing: IDP Availability” on page 486 and “Statement of Health Integration” on page 305.

A **Licensing Hardware ID** is a unique 16-character code Juniper Networks uses to identify your particular Infranet Controller when generating license keys. You can find the Infranet Controller’s Licensing Hardware ID above the menu options in the serial console and at the bottom of the admin console.

An **Authorization Code** is a pass key required to generate and activate license keys you or your company have purchased for your Infranet Controller. You receive your **Authorization Code(s) separate from the Infranet Controller after you purchase your Infranet Controller and associated product and feature licenses.**

**Figure 23: License Key Generation and Activation**

The package you download from the Juniper Networks License Management System or the email message you receive from Juniper Networks may contain different types of licenses:

- **Infranet Controller endpoint keys**—The Infranet Controller endpoint key enables you to host as many endpoints as are specified in the license key code. Infranet Controller endpoint license keys are **additive**, meaning that you can expand the number of endpoints that can access the Infranet Controller by simply acquiring an additional endpoint license key and adding it to your configuration. For example, if you initially purchase an IC4500-ADD-100E license and then purchase another IC4500-ADD-100E license in the future, your Infranet Controller could accommodate up to 200 endpoints.
Infranet Controller cluster license keys—Infranet Controller cluster license keys enable clustering behavior among Infranet Controllers. You can purchase Infranet Controller cluster license keys in conjunction with Infranet Controller endpoint license keys, but the number of endpoints that can access the Infranet Controllers in the cluster is restricted to the maximum number of endpoints allowed by the Infranet Controller cluster license key. Like Infranet Controller endpoint license keys, Infranet Controller cluster license keys are additive in that you can increase the number of endpoints who are able to access the cluster by purchasing additional license keys in the future. For example, if you initially purchase an IC4500-CL-100E license and then purchase another IC4500-CL-100E license in the future, your Infranet Controller could accommodate up to 200 endpoints. If you purchase an additional IC4500-ADD-100E license key, however, despite being able to accommodate up to 200 endpoints by means of your endpoint license keys, you can still only accommodate 100 endpoints in the cluster of Infranet Controllers. For more information on clustering Infranet Controllers, refer to “Clustering” on page 449.

NOTE: All nodes in a cluster must feature the same license key as on the primary cluster Infranet Controller to enable cluster operation. You cannot add an ADD and a CL license to the same machine at the same time. For a node to be able to join a cluster, you must add a CL license to the node.

Infranet Controller OAC Add UAC license key—This license prevents user sessions generated by the Enterprise Edition or FIPS Edition of Odyssey Access Client from increasing the concurrent user count against the Infranet Controller endpoint license. For more information, see “Using Odyssey Access Client Licenses with the Infranet Controller” on page 54.

Infranet Controller lab license keys—Lab license keys allow you to deploy new Infranet Controller functionality in a “test” or “laboratory” environment before deciding whether or not to purchase and roll out the up-to-date functionality in your live network. Lab license keys are valid for 52 weeks and grant access to a limited number of endpoints. Although you can purchase multiple lab license keys, it does not increase the number of endpoints to whom you can provide access. Instead, you can increase the duration of the license by multiples of 52 weeks (104 weeks, 156 weeks, and so forth). For example, if you purchase two IC4500-LAB licenses, you can grant 5 users access for 104 weeks, rather than just 52.

Infranet Controller Disaster Recovery (DR) license keys—The DR license has an eight week expiration period, and provides full support for all regular features on the Infranet Controller. The DR license has no rolling timer, so you can enable the DR feature and use it for a period of time, for example to synchronize policies. If the device is subsequently turned off, the expiration period will now be eight weeks minus the time already elapsed.

Infranet Controller evaluation license keys—Evaluation license keys allow you to enable and roll out the latest Infranet Controller functionality for a limited time before deciding whether or not to purchase license keys and enable the new Infranet Controller functionality on a permanent basis. Evaluation license keys are valid for 4, 8, or 16 weeks.
Use the **System > Configuration > Licensing** tab to enter the license keys for your site, view their expiration dates, and delete them (if required).

---

**NOTE:** Make sure that you read the license agreement, which is accessible from the **Licensing** tab, before submitting your license key. The license agreement available from the **Licensing** tab is the same text displayed in the serial console during the initial setup.

---

**Entering new Infranet Controller License Keys**

To create and enter new Infranet Controller license keys or transfer license keys to a replacement Infranet Controller:

1. Ensure that you have your Licensing Hardware ID and Authorization Code(s) readily available.

   You can find the Infranet Controller’s **Licensing Hardware ID** above the menu options in the serial console and at the bottom of the admin console.

   You receive your **Authorization Code(s)** separate from the Infranet Controller after you purchase your Infranet Controller and associated product and feature licenses.

2. Navigate to the Juniper Networks License Management System at [https://www.juniper.net/generate_license](https://www.juniper.net/generate_license).

   **NOTE:** Make sure that you read the license agreement, which is accessible from the **Licensing** tab, before submitting your license key. The license agreement available from the **Licensing** tab is the same text displayed in the serial console during the initial setup.

---

**NOTE:** The Juniper Networks License Management System offers you an access point where you can obtain detailed information about Juniper Networks licenses, including all licenses registered to you and your company, as well as licenses currently associated with specific Licensing Hardware IDs.

You must have a valid Juniper Networks Customer Support Center user ID and password to access the information at this location. To obtain a Juniper Networks Customer Support Center user ID and password, access the **Customer Support Center**.

---

3. Choose **Infranet Controller Series** from the menu and then click **Go** to generate new Infranet Controller license keys or click **Generate Replacement License for RMA Device** to create a license key based on an existing license for an Infranet Controller that you are replacing.

   **NOTE:** The **Generate Replacement License for RMA Device** option is designed to accommodate RMA hardware-replacement scenarios only. It cannot be used to replace a license key that was created in error (for example, using an Authorization Code to create a license key for an Infranet Controller other than the Infranet Controller for which the license was originally purchased).
4. On the **Generate Licenses** page:
   - If you are creating a license key for only one Infranet Controller, enter the Licensing Hardware ID and one or more Authorization Codes in the appropriate fields.
   - If you want to create license keys for multiple Infranet Controllers at the same time, click **Generate License Keys for Multiple Infranet Controller Series Devices** and follow the on-screen procedure to create the Excel file necessary to generate your license keys.

5. Click **Generate**.

   The **Confirm License Information** page appears, displaying a summary of the information you submitted to the License Management System.

6. Review the information to ensure everything is correct and then click **Generate License**.

   The **Generate Infranet Controller License** page appears, displaying a summary of your license keys, including a link that displays the details of your new license keys.

7. Click **Download/Email** and specify the file format and delivery method you want to use to obtain your new license keys.

After you download or receive your license keys by using email:

1. In the admin console, choose **System > Configuration > Licensing**.
2. Click on the license agreement link. Read the license agreement and, if you agree to the terms, continue to the next step.
3. Enter your license key(s) and click **Add**.

---

**Activating and Deactivating Emergency Mode**

The Infranet Controller emergency mode feature allows you to temporarily enable the Infranet Controller for a large number of users, as explained in “Entering or Upgrading Infranet Controller Licenses” on page 346.

In order to activate the Infranet Controller in emergency mode, you must first install a Disaster Recovery (DR) license using the standard Infranet Controller license installation procedure. Then, when the emergency occurs, you can easily activate emergency mode through the Infranet Controller Web console. When your emergency has passed, you should then deactivate the emergency mode.

**NOTE:** The DR license is permanent until you activate emergency mode. Activating emergency mode switches the DR license to a temporary license and only enables you to operate in emergency mode for 8 weeks. Once the DR license expires, all features disappear and your users can no longer access the Infranet Controller using the emergency mode.

To activate or deactivate emergency mode:
1. In the Web console, choose **System > Configuration > Licensing**.

2. Find the **In Case of Emergency License** entry in the license list. Sample DR license names include:
   - IC4500-DR
   - IC6500-DR

Click the **Enable** link in the right side of the license column to activate emergency mode or click **Disable** to deactivate it.

**Setting Security Options**

**NOTE:** When you enable and disable emergency mode, the Infranet Controller decrements the corresponding license in 5 minute intervals.

Use the **System > Configuration > Security** page to change the default security settings for your Infranet Controller. We recommend that you use the default security settings, which provide maximum security, but you may need to modify these settings if your users cannot use certain browsers or access certain Web pages. You can also configure lockout options for protecting the Infranet Controller and back-end systems from DOS/DDOS/Password Guessing attacks from the same IP address.

**Setting System-Wide Security Options**

If any of your users experience browser problems when accessing certain Web pages, consider adjusting the following settings:

- **Allowed SSL and TLS Version**—Specify encryption requirements for Infranet Controller users. The Infranet Controller honors this setting for all Web server traffic including oNCP, and all types of clients. (The Infranet Controller requires SSL version 3 and TLS by default.) You can require users who have older browsers that use SSL version 2 to update their browsers or change the Infranet Controller setting to allow SSL version 2, SSL version 3, and TLS.

- **Allowed Encryption Strength**—The Infranet Controller requires 128-bit encryption by default, or you can specify that the Infranet Controller requires 168-bit encryption. Older browsers, which pre-date the 2000 change in the U.S. export law that required 40-bit cipher encryption for international export, may still use 40-bit encryption. You can either require users to update to a browser with 128-bit cipher encryption or change the required encryption strength to also allow 40-bit ciphers.

   If you select the **Accept only 168-bit and greater** option, the Infranet Controller gives preference to 256-bit AES over 3DES. If you select the **Accept only 128-bit and greater** option or the **Accept 40-bit and greater** option, the Infranet Controller gives preference to RC4 ciphers.
To specify a combination of cipher suites for the incoming connection from the user’s browser, choose the **Custom SSL Cipher Selection** option under **Allowed Encryption Strength**. If you select the AES/3DES option, the Infranet Controller gives preference to 256-bit AES over 3DES. If you select any of the custom cipher suite options, the Infranet Controller uses 168-bit or higher ciphers for backend rewriter connections. The Infranet Controller gives preference to 256-bit AES encryption for backend mail proxy SSL connections.

---

**NOTE:** When using 168-bit encryption on the Infranet Controller, some Web browsers may still show 128-bit encryption (the gold lock on the browser status bar) even though the connection is 168-bit. This may be a limitation of the browser’s capability.

- **Encryption Strength option**—Normally, the allowed encryption strength is enforced after an SSL session is established, so that a user connecting with a disallowed encryption strength receives a Web page describing the problem. This option prevents a browser with a weak cipher from establishing a connection.

- **SSL Handshake Timeout option**—Determines how many seconds before the SSL handshake times out. The default is 60 seconds.

- **Delete all cookies at session termination**—For convenience, the Infranet Controller sets persistent cookies on the user’s machine to support functions such as multiple sign-in, last associated realm, and the last sign-in URL. If you desire additional security or privacy, you may choose to not set them.

- **Last Login options**—Display the day and time the user last logged in to the system. For users, this information appears on their bookmark page. For administrators, this information appears on the System Status Overview page.

- **SAML version**—By default, the Infranet Controller uses SAML 1.1 protocol and schema. If you are using SAML 1.0 in your environment, select the **SAML 1.0** option.

### Configuring Lockout Options

You can configure the following **Lockout options** to protect the Infranet Controller and other systems from Denial of Service (DoS), Distributed Denial of Service (DDoS), and password-guessing attacks from the same IP address:

- **Rate**—Specify the number of failed sign-in attempts to allow per minute.

- **Attempts**—Specify the maximum number of failed sign-in attempts to allow before triggering the initial lockout. The Infranet Controller determines the maximum initial period of time (in minutes) to allow the failed sign-in attempts to occur by dividing the specified number of attempts by the rate. For example, 180 attempts divided by a rate of 3 results in a initial period of 60 minutes. If 180 or more failed sign-in attempts occur within 60 minutes or less, the Infranet Controller locks out the IP address being used for the failed sign-in attempt.
Lockout period—Specify the number of minutes you want the Infranet Controller to lock out the IP address.

The Infranet Controller reacts quickly to an attack that persists, and then gradually becomes less restrictive when the attack subsides. After a lockout occurs, the Infranet Controller gradually recovers by maintaining the Rate. If the current failure rate since the last lockout exceeds the specified Rate, the Infranet Controller locks out the IP address again. If the failure rate is less than the specified Rate for the period of Attempts/Rate, the Infranet Controller returns to the initial monitoring state.

For example, if you use the following settings for the Lockout options, the Infranet Controller locks out the IP address for the time periods in the following scenario.

- Rate = 3 failed sign-in attempts/minute
- Attempts = 180 maximum allowed in initial period of 60 minutes (180/3)
- Lockout period = 2 minutes

1. During a period of three minutes, 180 failed sign-in attempts occur from the same IP address. Because the specified value for Attempts occurs in less than the allowed initial period of 60 minutes (180/3), the Infranet Controller locks out the IP address for 2 minutes (4th and 5th minutes).

2. In the 6th minute, the Infranet Controller removes the lock on the IP address and begins maintaining the rate of 3 failed sign-in attempts/minute. In the 6th and 7th minutes, the number of failed sign-in attempts is 2 per minute, so the Infranet Controller does not lock the IP address. However, when the number of failed sign-in attempts increases to 5 in the 8th minute, which is a total of 9 failed sign-in attempts within 3 minutes, the Infranet Controller locks out the IP address for 2 minutes again (9th and 10th minutes).

3. In the 11th minute, the Infranet Controller removes the lock on the IP address and begins maintaining the rate of 3 failed sign-in attempts/minute again. When the rate remains below an average of 3/minute for 60 minutes, the Infranet Controller returns to its initial monitoring state.

NOTE: In environments where two or more users share the same IP address (as seen by the Infranet Controller), the lockout feature prevents all users from logging in from the shared IP address even when only one of them is the offending user. Sharing of the IP address as seen by the Infranet Controller can happen when, for example, users are logging in from behind a NAT box.

Installing a Juniper Software Service Package

Before installing a new service package, please export your current system configuration, local user accounts, customized user settings, and role and policy information using instructions in “Importing and Exporting Infranet Controller Configuration Files” on page 385.
To install a service package:

1. Browse to the Juniper Networks Customer Support Center and obtain the desired service package.

2. In the admin console, choose Maintenance > System > Upgrade/Downgrade.

3. Click Browse to find the service package on your hard drive that you obtained from the Juniper Networks Customer Support Center. If you want to delete your current configuration settings but continue to use the same Infranet Controller version, choose the service package that is currently installed on your appliance.

4. If you are rolling back to an older service package or deleting your configuration settings, select Delete all system and user data.

**NOTE:** If you choose to revert to delete all system and user data from the appliance using this option, you will have to reestablish network connectivity before re-configuring the system.

5. Select the service package file and click Install Now.
An Infranet Controller uses PKI to secure the data that it sends to clients over the Internet. PKI (public key infrastructure) is a security method that uses public and private keys to encrypt and decrypt information. These keys are enabled and stored through digital certificates. A digital certificate is an encrypted electronic file issued that establishes a Web server’s or user’s credentials for client-server transactions.

An Infranet Controller uses the following types of digital certificates to establish credentials and secure Infranet Controller session transactions:

- **Device certificates**—A device certificate helps to secure network traffic to and from an Infranet Controller using elements such as your company name, a copy of your company’s public key, the digital signature of the certificate authority (CA) who issued the certificate, a serial number, and expiration date. For more information, see “Using Device Certificates” on page 356. The Infranet Controller also uses a device certificate for NetScreen Address Change Notification (NACN) communications with the Infranet Enforcer. (See “Configuring the Infranet Controller to Connect to the Infranet Enforcer” on page 20.)

- **Trusted client CAs**—A trusted client CA is a client-side certificate issued by a certificate authority (CA) that allows you to control access to realms, roles, and resource policies based on certificates or certificate attributes. For example, you may specify that users must present a valid client-side certificate with the OU attribute set to “yourcompany.com” in order to sign into the “Users” authentication realm. For more information, see “Using Trusted Client CAs” on page 363.

- **Trusted server CAs**—A trusted server CA is the certificate of a Web server that you trust. If you have a Web browsing license, you may install a trusted server CA on the Infranet Controller in order to validate the credentials of the Web sites that users access through the Infranet Controller appliance. For more information, see “Using Trusted Server CAs” on page 377.

**NOTE:** The Infranet Controller can verify certificates that use SHA2 as the message digest.

**NOTE:** DSA certificates are currently not supported.
This section contains the following information about certificates:

- “Using Device Certificates” on page 356
- “Using Trusted Client CAs” on page 363
- “Using Trusted Server CAs” on page 377

## Using Device Certificates

A *device certificate* helps to secure network traffic to and from an Infranet Controller using elements such as your company name, a copy of your company’s public key, the digital signature of the certificate authority (CA) who issued the certificate, a serial number, and expiration date. The Infranet Controller also uses a device certificate for NetScreen Address Change Notification (NACN) communications with the Infranet Enforcer. (See “Configuring the Infranet Controller to Connect to the Infranet Enforcer” on page 20.)

When receiving encrypted data from an Infranet Controller, the client’s browser first checks whether the Infranet Controller’s certificate is valid and whether the user trusts the CA that issued the Infranet Controller’s certificate. If the user has not already indicated that he trusts the Infranet Controller’s certificate issuer, the Web browser prompts the user to accept or install the Infranet Controller’s certificate.

When you initialize an Infranet Controller, it creates a temporary self-signed digital certificate locally that enables you to immediately begin setting up your Infranet Controller. Users are prompted with a security alert each time they sign in to an Infranet Controller because the certificate is not issued by a trusted certificate authority (CA).

**NOTE:** If you use a self-signed certificate, the Infranet Enforcer does not allow a connection with the Infranet Controller. You must first import a signed device certificate on the Infranet Controller. Then, you must import the certificate of the CA (Certificate Authority) that signed the Infranet Controller’s device certificate into the Infranet Enforcer (see “Setting Up Certificates for the Infranet Controller and Infranet Enforcer” on page 16).

The Infranet Controller supports X.509 device certificates in DER and PEM encode formats (file extensions include `.cer`, `.crt`, `.der`, and `.pem`) as well as PKCS #12 (file extensions include `.pfx` and `.p12`). The Infranet Controller also supports using the following additional features with device certificates:

- **Intermediate device CA certificates**—Within a certificate hierarchy, one or more intermediate certificates are branched off of a single root certificate.

- **Multiple device certificates**—When using multiple device certificates, each certificate handles validation for a separate host name or fully-qualified domain name (FQDN) and may be issued by a different CA.

This section contains the following instructions for working with device certificates:

- “Importing Certificates Into the Infranet Controller” on page 357
Using Device Certificates

“Downloading a Device Certificate From the Infranet Controller” on page 359

“Creating a Certificate Signing Request (CSR) for a New Certificate” on page 359

“Using Intermediate Server Ca Certificates” on page 360

“Using Multiple Infranet Controller Device Certificates” on page 361

**Importing Certificates Into the Infranet Controller**

This section contains the following import instructions:

- “Importing an Existing Root Certificate and Private Key” on page 357
- “Importing a Renewed Certificate That Uses the Existing Private Key” on page 358

**Importing an Existing Root Certificate and Private Key**

You can create Web server certificates from servers such as Apache, IIS, Sun ONE (formerly iPlanet), or Netscape, and then import the certificate into the Infranet Controller. To export a digital server certificate and key, please follow your Web server’s instructions for exporting certificates. Then, use the Device Certificates tab to import these files.

To import an existing root server certificate and private key:

1. In the admin console, choose **System > Configuration > Certificates > Device Certificates**.
2. Click **Import Certificate & Key**.
3. Choose the appropriate form to import the certificate:
   - If the certificate and key are contained in one file, use the **If certificate file includes private key** form.
   - If the certificate and key are separate files, use the **If certificate and private key are separate files** form.
   - If the certificate and key are contained in a system configuration file, use the **Import via System Configuration file** form. When you choose this option, the Infranet Controller imports all of the certificates specified in the configuration file into the Device Certificates page (including private keys and pending CSRs, but not the corresponding port mappings).
4. In the appropriate form, browse to the certificate and key file. If the file is encrypted, enter the password key.
5. Click **Import**.
Importing a Renewed Certificate That Uses the Existing Private Key

You can renew a device certificate in two ways:

- **Submit a new CSR to a CA**—This process of renewing a certificate is more secure, because the CA generates a new certificate and private key, retiring the older private key. To use this renewal method, you must first create a CSR through the admin console. For more information, see “Creating a Certificate Signing Request (CSR) for a New Certificate” on page 359.

- **Request renewal based on the CSR previously submitted to the CA**—This process of renewing a certificate is less secure, because the CA generates a certificate that uses the existing private key.

**NOTE:** When ordering a renewed certificate, you must resubmit your original CSR or ensure that the CA has a record of the CSR that you submitted for your current certificate.

To import a renewed device certificate that uses the existing private key:

1. Follow your CA's instructions for renewing a certificate that you previously purchased through them.

**NOTE:** Ensure you specify the same information you used in the original CSR. Your CA uses this information to create a new certificate that corresponds to the existing key.

Even though you specify the same information used in the original CSR, your rootCA may have different serial numbers and keys from the original. You may need to support both new client and old client certificates during the transition period, which means that you will need to maintain two rootCA certificates (your existing cert and the renewed cert), at least temporarily.

2. In the admin console, choose **System > Configuration > Certificates > Device Certificates**.

3. If you want to renew an intermediate certificate, click the **Intermediate Device CAs** link at the top of the page.

4. Click the link that corresponds to the certificate that you want to renew.

5. Click **Renew Certificate**.

6. In the **Renew the Certificate** form, browse to the renewed certificate file, enter the password for the certificate key, and click **Import**.
**Downloading a Device Certificate From the Infranet Controller**

The **Device Certificates** page enables you to easily download the Infranet Controller’s device certificate.

To download a device certificate from the Infranet Controller:

1. In the admin console, choose **System > Configuration > Certificates > Device Certificates**.
2. Click the link that corresponds to the certificate that you want to save.
3. Click **Download**.
4. Browse to the location where you want to save the certificate and click **Save**.

**Creating a Certificate Signing Request (CSR) for a New Certificate**

If your company does not own a digital certificate for its Web servers, you can create a CSR (certificate signing request) through the admin console and then send the request to a CA for processing. When you create a CSR through the admin console, a private key is created locally that corresponds to the CSR. If you delete the CSR at any point, this file is deleted, too, prohibiting you from installing a signed certificate generated from the CSR.

To create a certificate signing request:

1. In the admin console, choose **System > Configuration > Certificates > Device Certificates**.
2. Click **New CSR**.
3. Enter the required information and click **Create CSR**.
4. Follow the instructions on-screen, which explain what information to send to the CA and how to send it.
5. When you receive the signed certificate from the CA, import the certificate file using the instructions that follow.

**NOTE:** Do not send more than one CSR to a CA at one time. Doing so may result in duplicate charges. You may view details of any pending requests that you previously submitted by clicking the **Certificate Signing Request Details** link in the **Device Certificates** tab.

To create a certificate signing request:

1. In the admin console, choose **System > Configuration > Certificates > Device Certificates**.
2. Click **New CSR**.
3. Enter the required information and click **Create CSR**.
4. Follow the instructions on-screen, which explain what information to send to the CA and how to send it.
5. When you receive the signed certificate from the CA, import the certificate file using the instructions that follow.

**NOTE:** When submitting a CSR to a CA authority, you may be asked to specify either the type of Web server on which the certificate was created or the type of Web server the certificate is for. Select **apache** (if more than one option with apache is available, choose any). Also, if prompted for the certificate format to download, select the standard format.
**Importing a Signed Certificate Created From a CSR**

If you create a CSR through the admin console, the Infranet Controller displays a Pending CSR link for the CSR in the Device Certificates tab until you import the signed device certificate distributed by the certificate authority (CA).

To import a signed device certificate created from a CSR:

1. In the admin console, choose System > Configuration > Certificates > Device Certificates.
2. Under Certificate Signing Requests, click the Pending CSR link that corresponds to the signed certificate.
3. Under Import signed certificate, browse to the certificate file you received from the CA and then click Import.

---

**NOTE:** Import the CA certificate of the CA that signed the Infranet Controller’s device certificate into the Infranet Enforcer. The Infranet Enforcer uses the CA certificate to verify the Infranet Controller’s device certificate. For more information, see “Creating an Infranet Controller Instance in Route Mode” on page 29.

---

**Using Intermediate Server Ca Certificates**

Within a certificate hierarchy, one or more intermediate certificates are branched off of a single root certificate. The root certificate is issued by a root certificate authority (CA) and is self-signed. Each intermediate certificates is issued by the certificate above it in the chain.

If you are securing traffic using chained certificates, you must ensure that the Infranet Controller and Web browser together contain the entire certificate chain. For example, you may choose to secure traffic using a chain that stems from a Verisign root certificate. Assuming your users’ browsers come pre-loaded with Verisign root certificates, you only need to install the lower-level certificates in the chain on the Infranet Controller. Then, when your users browse to the Infranet Controller, the Infranet Controller presents any required certificates within the chain to the browser in order to secure the transaction. (The Infranet Controller creates the proper links in the chain using the root certificate’s IssuerDN.) If the Infranet Controller and browser together do not contain the entire chain, the user’s browser will not recognize or trust the Infranet Controller’s device certificate since it is issued by another certificate instead of a trusted CA.

For information about chained client certificates, see “Enabling Client CA Hierarchies” on page 371.

When installing certificates through the Infranet Controller, you may install certificates in any order. The Infranet Controller supports uploading one or more intermediate CAs in a PEM file.
To import an intermediate device certificate and private key:

1. In the admin console, choose System > Configuration > Certificates > Device Certificates.

2. Click the Intermediate Device CAs link at the top of the page.

3. Click Import CA certificate.

4. Browse to the CA certificate that you want to upload to the Infranet Controller and click Import Certificate.

Using Multiple Infranet Controller Device Certificates

When you use multiple Infranet Controller device certificates, each certificate handles validation for a separate host name or fully qualified domain name (FQDN) and may be issued by a different CA. You can use different certificates for users who sign in using different virtual ports.

Task summary: Enabling Multiple Device Certificates

To enable multiple device certificates, you must:

1. Specify the IP addresses from which users may access the Infranet Controller and then create a virtual port for each. A virtual port activates an IP alias on a physical port. To create virtual ports for:

   - **Internal physical interface**—Use settings in the System > Network > Internal Port > Virtual Ports tab to create virtual ports for the first set of certificates. For instructions, see “Configuring Virtual Ports” on page 335.

   - **External physical interface**—Use settings in the System > Network > Port 1 > Virtual Ports tab to create virtual ports for a different set of certificates. For instructions, see “Configuring Virtual Ports” on page 335.

2. Upload your device certificates to the Infranet Controller. You can import certificates from the System > Configuration > Certificates > Device Certificates page of the admin console or the Maintenance > Import/Export > System Configuration page of the admin console. Upload one device certificate for each domain (host name) that you want to host on the Infranet Controller. For instructions, see “Importing an Existing Root Certificate and Private Key” on page 357.

3. Specify which virtual ports the Infranet Controller should associate with the certificates using settings in the System > Configuration > Certificates > Device Certificates tab. When a user tries to sign into the Infranet Controller using the IP address defined in a virtual port, the Infranet Controller uses the certificate associated with the virtual port to initiate the SSL transaction, and for NACN communications with the Infranet Enforcer. For instructions, see “Associating a Certificate With a Virtual Port” on page 362.
Associating a Certificate With a Virtual Port

On the Infranet Controller, you must associate the signed certificate with the port that is connected to the Infranet Enforcer. You can use the same port and certificate for Odyssey Access Client. Or, you can choose to import other signed certificates and associate them with ports connected to Odyssey Access Client.

- **Associate all host names with a single wildcard certificate**—With this method, you use a single wildcard certificate to validate the identity of all Infranet Controllers to the user or the Infranet Enforcer, regardless of which host name is used to sign into the Infranet Controller. A *wildcard certificate* includes a variable element in the domain name, making it possible for users signing in from multiple hosts to map to the “same” domain. For example, if you create a wildcard certificate for *.yourcompany.com, the Infranet Controller uses the same certificate to validate its identity to users who sign into employees.yourcompany.com as it does to users who sign into partners.yourcompany.com.

- **Associate each host name with its own certificate**—With this method, you associate different host names with different certificates. Since the Infranet Controller does not know the host name that the end-user uses to sign into the Infranet Controller, however, you must create a virtual port for each host name and then associate your certificates with the virtual ports. A *virtual port* activates an IP alias on a physical port. For example, you may choose to create two virtual ports on a single appliance, mapping the first virtual port to the IP address 10.10.10.1 (sales.yourcompany.com) and the second virtual port to the IP address 10.10.10.2 (partners.yourcompany.com). Then, you can associate each of these virtual ports with its own certificate, ensuring that the Infranet Controller authenticates different users through different certificates.

To associate different certificates with different virtual ports:

1. In the admin console, navigate to the **System > Network > Internal Port** tab or **Port 1** tab. Then, create your virtual ports using settings in the **Virtual Ports** page.

2. Import the device certificates that you want to use to validate user certificates. You can import certificates from the **System > Configuration > Certificates > Device Certificates** page of the admin console or the **Maintenance > Import/Export > System Configuration** page of the admin console.

3. On the **System > Configuration > Certificates > Device Certificates** page, click the link that corresponds to a certificate that you want to use.

4. Under **Present certificate on these ports**, specify the port(s) that the Infranet Controller should associate with the certificate—you can choose internal or external ports and primary or virtual ports, but you cannot choose a port that is already associated with another certificate.

5. Click **Save Changes**.

6. Repeat steps 3-6 for each of the certificates that you want to use to authenticate users.
Using Trusted Client CAs

A trusted client CA is a certificate authority (CA) trusted by the Infranet Controller. The Infranet Controller trusts any certificate issued by that CA. To use client CA certificates, you must install and enable the proper certificates on the Infranet Controller. Additionally, you must install the corresponding client-side certificates in the Web browsers of your end-users or use MMC Certificates snap in your users’ computer accounts (machine certificate). When validating a client-side CA certificate, the Infranet Controller checks that the certificate is not expired or corrupt and that the certificate is signed by a CA that the Infranet Controller recognizes. If the CA certificate is chained (described below) the Infranet Controller also follows the chain of issuers until it reaches the root CA, checking the validity of each issuer as it goes. The Infranet Controller supports X.509 CA certificates in DER and PEM encode formats.

When installing a client-side certificate, you must determine whether you want to use the certificate to identify individual users or individual machines. To use the certificate to identify individual users, you must install the certificate in each user’s individual certificate store. Then you must enable authentication through the Infranet Controller administration console using a certificate server (as explained in “Configuring a Certificate Server Instance” on page 190) or enable authorization using realm, role, and/or resource policy settings (as explained in “Specifying Client-side Certificate Restrictions” on page 369). To use the certificate to identify individual machines, you must install the certificate in each computer’s certificate store. Then, you must configure a Host Checker policy that checks for the machine certificate and authorizes access to realms, roles, and/or resource policies based on the certificate’s validity (as explained in “Specifying Customized Requirements Using Custom Rules” on page 286).

The Infranet Controller supports using the following additional features with CA certificates:

- Certificate servers—A certificate server is a type of local authentication server that allows you to authenticate Infranet Controller users based solely on their certificate attributes rather than authenticating them against a standard authentication server (such as LDAP or RADIUS), as well as requiring specific certificates or certificate attributes. For more information, see “Configuring a Certificate Server Instance” on page 190.
Certificate hierarchies—Within a certificate hierarchy, one or more subordinate certificates (called intermediate certificates) are branched off of a root certificate creating a certificate chain. Each intermediate certificate (also called a chained certificate) handles requests for a part of the root CA's domain. For example, you may create a root certificate that handles all requests to the yourcompany.com domain and then branch off intermediate certificates that handle requests to partners.yourcompany.com and employees.yourcompany.com. When you install a chained certificate on the Infranet Controller, the appliance confirms that the chain is valid and allows users to authenticate using the leaf certificate (that is, the lowest certificate in the chain). For more information, see “Enabling Client CA Hierarchies” on page 371.

Certificate revocation lists—Certificate revocation is a mechanism by which a CA invalidates a certificate before its expiration date. A certificate revocation list (CRL) is a list of revoked certificates published by a CA. Within CRLs, each entry contains the serial number of the revoked certificate, the date that the certificate was revoked, and the reason that the certificate was revoked. The CA may invalidate a certificate for various reasons such as the employee to whom the certificate is issued has left the company, the certificate's private key is compromised, or the client-side certificate is lost or stolen. Once the CA revokes a certificate, the Infranet Controller can appropriately deny access to users who present a revoked certificate. For more information, see “Enabling CRLs” on page 372.

Enabling Trusted Client CAs

If you require users to provide a client-side certificate to sign in to the Infranet Controller, you must upload the corresponding CA certificate into the Infranet Controller. You can upload CA certificates manually or configure the Infranet Controller to upload CA certificates automatically. The Infranet Controller uses the uploaded certificate to verify that the browser-submitted certificate is valid. In addition, you can specify whether or not to automatically import CA certificates for validation and the CRL/OCSP retrieval method the Infranet Controller uses when automatically importing the CA certificates.
This section contains the following CA certificate instructions:

- “Automatically Import a CA Certificate” on page 365
- “Manually Upload CA Certificates” on page 367
- “Specifying Attributes for the Trusted Client CA Certificate” on page 368
- “Specifying Client-side Certificate Restrictions” on page 369

**Automatically Import a CA Certificate**

To automatically import and specify options for a trusted client CA certificate on the Infranet Controller:

1. In the admin console, choose System > Configuration > Certificates > Trusted Client CAs.

2. Click Auto-import options. The Auto-import options page appears.
3. Click Auto-import Trusted CAs.

4. Under Client certificate status checking, specify the method the Infranet Controller uses to verify client certificate status:

- **None**—Specifies that the Infranet Controller should not validate this trusted client certificate.
- **Use OCSP**—Specifies that the Infranet Controller should use the OCSP method, validating the client certificate in real-time, as needed. After you select this option, you can specify options for OCSP as described in “Specifying Client-side Certificate Restrictions” on page 369.
- **Use CRLs**—Specifies that the Infranet Controller should use CRLs to validate the client certificate. After you select this option, you can specify options for OCSP as described in “Specifying Client-side Certificate Restrictions” on page 369.
- **Use OCSP with CRL fallback**—Specifies that the Infranet Controller should use the OCSP validation method when possible, but attempt to validate client certificates using CRLs should the OCSP method fail (for example, if the link to the OCSP Responder were to fail). After you select this option, you can specify options for OCSP as described in “Specifying Client-side Certificate Restrictions” on page 369 and for CDP as described in “Specifying CDP Options” on page 374.

5. Under CDP(s)/OCSP responder, specify the CRL/OCSP retrieval method from the associated drop-down list:

- **None**—Specifies that the Infranet Controller does not use a CRL/OCSP retrieval method.
- **From client certificate**—Specifies that the Infranet Controller use a CRL/OCSP retrieval method found in the client certificate.
- **From trusted CA certificates**—Specifies that the Infranet Controller use a CRL/OCSP retrieval method found in the trusted client CA certificate.

6. Enable the Verify imported CA certificates option if you want the Infranet Controller to validate the CRL from which the certificate is issued.

7. Click Save.

8. Use one of the following methods to specify how the Infranet Controller should use the certificate to authenticate users and/or authorize access to resources:

- Use a certificate server to authenticate individual users as explained in “Configuring a Certificate Server Instance” on page 190,
- Use realm, role, and resource policy settings to authorize individual users access to resources as explained in “Specifying Client-side Certificate Restrictions” on page 369,
- Use a Host Checker policy to authorize individual machines to access resources as explained in “Specifying Customized Requirements Using Custom Rules” on page 286.
Manually Upload CA Certificates

To manually upload CA certificates to the Infranet Controller:

1. Install a client-side user certificate or machine certificate through the user’s Web browser. For help, see the instructions provided with the browser.

2. In the admin console, choose System > Configuration > Certificates > Trusted Client CAs.

3. Click Import CA Certificate. The Import Trusted Client CA page appears.

4. Browse to the CA certificate that you want to upload to the Infranet Controller and click Import Certificate.

5. Under Client certificate status checking, specify the method the Infranet Controller uses to verify client certificate status:
   - None—Specifies that the Infranet Controller should not validate this trusted client certificate.
   - Use OCSP—Specifies that the Infranet Controller should use the OCSP method, validating the client certificate in real-time, as needed. After you select this option, you can specify options for OCSP as described in “Specifying Client-side Certificate Restrictions” on page 369.
   - Use CRLs—Specifies that the Infranet Controller should use CRLs to validate the client certificate. After you select this option, you can specify options for OCSP as described in “Specifying Client-side Certificate Restrictions” on page 369 and for CDP as described in “Specifying CDP Options” on page 374.
   - Use OCSP with CRL fallback— Specifies that the Infranet Controller should use the OCSP validation method when possible, but attempt to validate client certificates using CRLs should the OCSP method fail (for example, if the link to the OCSP Responder were to fail). After you select this option, you can specify options for OCSP as described in “Specifying Client-side Certificate Restrictions” on page 369 and for CDP as described in “Specifying CDP Options” on page 374.

6. Enable the Verify Trusted Client CA option if you want the Infranet Controller to validate the CRL from which the certificate is issued.

7. Enable the Verify for Client Authentication? option if you want the Infranet Controller to trust this certificate when authenticating client certificates. If you added this certificate for non-authentication purposes (such as for SAML signature verification or machine certificate validation), disable this option, indicating that the Infranet Controller should not trust any client certificate issued by this CA.

8. Click Save.

After you have manually imported the CA certificate, you can specify CA certificate attributes as described in “Specifying Attributes for the Trusted Client CA Certificate” on page 368.

9. Use one of the following methods to specify how the Infranet Controller should use the certificate to authenticate users and/or authorize access to resources:
Use a certificate server to authenticate individual users as explained in “Configuring a Certificate Server Instance” on page 190,

Use realm, role, and resource policy settings to authorize individual users access to resources as explained in “Specifying Client-side Certificate Restrictions” on page 369,

Use a Host Checker policy to authorize individual machines to access resources as explained in “Specifying Customized Requirements Using Custom Rules” on page 286.

Specifying Attributes for the Trusted Client CA Certificate
To specify attributes for the trusted client CA certificate:

1. In the admin console, choose System > Configuration > Certificates > Trusted Client CAs.

2. Click the certificate that you want to view. The Trusted Client CA page appears displaying all of the information about the certificate you selected.

3. Under Certificate, use the arrow next to the following field names to view certificate details:
   - **Issued To**—Name and attributes of the entity to whom the certificate is issued.
   - **Issued By**—Name and attributes of the entity that issued the certificate. Note that the value of this field should either match the Issued To field (for root certificates) or the Issued To field of the next certificate up in the chain (for intermediate certificates).
   - **Valid Dates**—Time range that the certificate is valid. If your certificate is expired, see the instructions in “Importing a Renewed Certificate That Uses the Existing Private Key” on page 358.
   - **Details**—Includes various certificate details, including its version, serial number, signature algorithm, CRL distribution points, public key algorithm type, and the public key. Note that although the Infranet Controller may display a CRL distribution point in the Details field, it does not check the CDP unless you enable it. For more information, see “Enabling CRLs” on page 372.

4. If you want to renew the certificate:
   a. Click Renew Certificate.
   b. Browse to the renewed CA certificate that you want to upload to the Infranet Controller and click Import Certificate.

5. In the Client Certificate Status Checking section, specify the method the Infranet Controller uses to validate the client certificate:
   - **None**—Specifies that the Infranet Controller should not validate this trusted client certificate.
- **Use OCSP**—Specifies that the Infranet Controller should use the OCSP method, validating the client certificate in real-time, as needed. After you select this option, you can specify options for OCSP as described in “Specifying Client-side Certificate Restrictions” on page 369.

- **Use CRLs**—Specifies that the Infranet Controller should use CRLs to validate the client certificate. After you select this option, you can specify options for OCSP as described in “Specifying CDP Options” on page 374.

- **Use OCSP with CRL fallback**—Specifies that the Infranet Controller should use the OCSP validation method when possible, but attempt to validate client certificates using CRLs should the OCSP method fail (for example, if the link to the OCSP Responder were to fail). After you select this option, you can specify options for OCSP as described in “Specifying Client-side Certificate Restrictions” on page 369 and for CDP as described in “Specifying CDP Options” on page 374.

6. Enable the **Verify Trusted Client CA** option to instruct the Infranet Controller to validate the trusted client CA.

7. Click **Save Changes**.

### Specifying Client-side Certificate Restrictions

Use a certificate restriction to require client machines to possess a valid client-side certificate in order to access an Infranet Controller sign-in page or be mapped to a role. If you use this feature, make sure that you import a CA certificate to verify the client-side certificate. To maximize the security of this feature, make sure that a user’s client settings are set to require the user to enter a password each time the user signs in. The default setting for some browser versions is to remember the certificate password, which means the user won’t be prompted for this additional sign-in information after installing the certificate.

To specify certificate restrictions:

1. Navigate to: **System > Configuration > Certificates > Trusted Client CAs** and specify the root certificate authority that you want to use to validate the client-side certificate restrictions that you enable at the realm, role, and resource policy levels.

2. Select the level at which you want to implement certificate restrictions:

   - **Realm level**—Navigate to:
     - **Administrators > Admin Realms > Select Realm > Authentication Policy > Certificate**
     - **Users > User Realms > Select Realm > Authentication Policy > Certificate**

   - **Role level**—Navigate to:
     - **Administrators > Admin Roles > Select Role > General > Restrictions > Certificate**
3. Choose one of the following options:

- **Allow all users (no client-side certificate required)** — Does not require a user’s client to have a client-side certificate.

- **Allow all users and remember certificate information while user is signed in** — Does not require a user’s client to have a client-side certificate, but if the client does have a certificate, the Infranet Controller remembers the certificate information during the entire user session.

- **Only allow users with a client-side certificate signed by Trusted Client CAs to sign in** — Requires a user’s client to have a client-side certificate in order to satisfy the access management requirement. To restrict access even further, you can define unique certificate attribute-value pairs. Note that the user’s certificate must have all the attributes you define.

4. Add a certificate field name and an expected value to optionally require specific values in the client certificate. You can specify variables in the **Expected Value** field. For example, you can add the value `uid` to the **Certificate** field and `<userAttr.uid>` to the **Expected Value** field.

**NOTE:** The user attribute can come from any authentication server that supports attributes. Any attribute name specified in a certificate restriction must be included in the server catalog so the values are collected during authentication and added to the session context data.
5. Click **Save Changes** to save your settings.

### NOTE:

- The Infranet Controller supports all **X.509 Distinguished Name (DN)** attributes (such as C, CN, L, O, OU).
- The attribute and value fields are not case-sensitive.
- Define only one value for each attribute. If you specify multiple values, the client-side certificate may not authenticate correctly against the CA certificate.
- The Infranet Controller currently recognizes an e-mail address in the **subjectAltName** attribute in a certificate.
- The Infranet Controller can extract the **User Principal Name (UPN)** from the **subjectAltName** attribute. The Infranet Controller locates a specific **UPN Object Identifier (OID)** in the certificate and decodes the value. To represent the **UPN** in the **subjectAltName** attribute, use the token `<certAttr.altName.UPN>`.

### Enabling Client CA Hierarchies

Within a certificate hierarchy, one or more intermediate certificates are branched off of a single root certificate. The root certificate is issued by a root certificate authority (CA) and is self-signed. Each intermediate certificate is issued by the certificate above it in the chain.

To enable authentication in a chained certificate environment, you must install the appropriate client-side certificates in each user’s Web browser and then upload the corresponding CA certificates to the Infranet Controller.

For information about chained device certificates, see “Using Intermediate Server Ca Certificates” on page 360.

You can install client CAs through the **System > Configuration > Certificates > Trusted Client CAs** page of the admin console. When uploading the certificate chain to the Infranet Controller, you must use one of the following methods:

- **Import the entire certificate chain at once**—When installing a chain of certificates contained in a single file, the Infranet Controller imports the root certificate and any sub-certificates whose parents are in the file or on the Infranet Controller. You can include certificates in any order in the import file.

- **Import the certificates one at a time in descending order**—When installing a chain of certificates contained in multiple files, the Infranet Controller requires that you install the root certificate first, and then install the remaining chained certificates in descending order.
When you install chained certificates using one of these methods, the Infranet Controller automatically chains the certificates together in the correct order and displays them hierarchically in the admin console.

NOTE: If you install multiple certificates in a user’s Web browser, the browser prompts the user to choose which certificate to use whenever he signs into the Infranet Controller.

**Enabling CRLs**

A *certificate revocation list (CRL)* is a mechanism for cancelling a client-side certificate. As the name implies, a CRL is a list of revoked certificates published by a CA or delegated CRL issuer. The Infranet Controller supports *base CRLs*, which include all of the company’s revoked certificates in a single, unified list.

The Infranet Controller knows which CRL to use by checking the client’s certificate. (When issuing a certificate, the CA includes CRL information for the certificate in the certificate itself.) To ensure that it receives the most up-to-date CRL information, the Infranet Controller periodically contacts a CRL distribution point to get an updated list of revoked certificates. A *CRL distribution point (CDP)* is a location on an LDAP directory server or Web server where a CA publishes CRLs. The Infranet Controller downloads CRL information from the CDP at the interval specified in the CRL, at the interval that you specify during CRL configuration, and when you choose to manually download the CRL. The Infranet Controller also supports CRL partitioning. CRL partitioning enables you to verify portions of very large CRLs without having to spend the time and bandwidth necessary to access and validate a very large CRL or collection of large CRLs. CRL partitioning is only enabled on the Infranet Controller when you employ the **Specify the CDP(s) in the client certificates** method (described below). In this case, the Infranet Controller validates the user by verifying only the CRL specified in the client certificate.
Although CAs include CRL information in client-side certificates, they do not always include CDP information, as well. A CA may use any of the following methods to notify the Infranet Controller of a certificate’s CDP location:

- **Specify the CDP(s) in the CA certificate**—When the CA issues a CA certificate, it may include an attribute specifying the location of the CDP(s) that the Infranet Controller should contact. If more than one CDP is specified, the Infranet Controller chooses the first one listed in the certificate and then fails over to subsequent CDPs, if necessary.

- **Specify the CDP(s) in the client certificates**—When the CA issues client-side certificates, it may include an attribute specifying the location of the CDP(s) that the Infranet Controller should contact. If more than one CDP is specified, the Infranet Controller chooses the first one listed in the certificate and then fails over to subsequent CDPs, if necessary. When the Infranet Controller employs CRL partitioning and the client certificate specifies only one CRL, the Infranet Controller performs verification using only that CRL.

**NOTE:** If you choose this method, the user receives an error the first time he tries to sign into the Infranet Controller because no CRL information is available. Once the Infranet Controller recognizes the client’s certificate and extracts the CRL location, it can start downloading the CRL and subsequently validate the user’s certificate. In order to successfully sign into the Infranet Controller, the user must try to reconnect after a few seconds.

- **Require the administrator to manually enter the CDP location**—If the CA does not include the CDP location in the client or CA certificates, you must manually specify how to download the entire CRL object when configuring the Infranet Controller. You may specify a primary and backup CDP. (Manually entering the CDP location provides the greatest flexibility because you do not need to reissue certificates if you change your CDP location.)

The Infranet Controller checks the user’s certificate against the appropriate CRL during authentication. If it determines that the user’s certificate is valid, the Infranet Controller caches the certificate attributes and applies them if necessary during role and resource policy checks. If it determines that the user’s certificate is invalid, if it cannot contact the appropriate CRL, or if the CRL is expired, the Infranet Controller denies the user access.

You can configure CRL checking through the **System > Configuration > Certificates > Trusted Client CAs** page of the admin console.

**NOTE:**

- The Infranet Controller only supports CRLs that are in a PEM or DER format and that are signed by the CA for which the revocations apply.

- The Infranet Controller only saves the first CRL in a PEM file.

- The Infranet Controller does not support the Issuing Distribution Point (IDP) CRL extension.
Specifying CDP Options
If you selected either Use CRLs or Use OCSP with CRL fallback in the procedures outlined in “Specifying Attributes for the Trusted Client CA Certificate” on page 368, you can enable and periodically download certificate revocation lists (CRL) from CRL distribution points (CDPs) in order to verify the ongoing validity of client-side certificates.

1. In the admin console, choose System > Configuration > Certificates > Trusted Client CAs.
2. Click the link that corresponds to the certificate for which you want to enable CRL checking.

   ![NOTE: Since the Infranet Controller supports CRL partitioning, you may see multiple CRLs displayed under CRL distribution points. This is because the partitioned portions of a revocation list are not identified individually, but referred to as the CDP from which they are derived.]

3. Click CRL Checking Options. The CRL Checking Options page appears.
4. Under CRL Distribution Points, specify where the Infranet Controller should find access information for the CDP. Options include:
   - No CDP (no CRL Checking)—When you select this option, the Infranet Controller does not check CRLs issued by the CA, so you do not need to enter any parameters to access the CDP that issued the CRL.
   - CDP(s) specified in the Trusted Client CA—When you select this option, the Infranet Controller checks the CRL distribution point attribute in the certificate and displays the URIs of the CDPs that it finds in the CRL Checking Options page. If the CA certificate does not include all of the information required to access the CDP, specify the additional required information:
     - CDP Server: (LDAP only)—Enter the location of the CDP server. When using LDAP protocol, enter the IP address or host name (for example, ldap.domain.com).
     - CRL Attribute: (LDAP only)—Enter the LDAP attribute on the object that contains the CRL (for example, CertificateRevocationList).
     - Admin DN, Password: (LDAP only)—If the CDP server does not allow anonymous searches of the CRL, enter the admin DN and password that are required to authenticate into the CDP server.
   - CDP(s) specified in client certificates—If the client certificate does not include all of the information required to access the CDP, specify the additional required information:
     - CDP Server (LDAP only)—Enter the location of the CDP server. When using LDAP protocol, enter the IP address or host name (for example, ldap.domain.com).
Using Trusted Client CAs

CRL Attribute (LDAP only)—Enter the LDAP attribute on the object that contains the CRL (for example, CertificateRevocationList).

Admin DN, Password (LDAP only)—If the CDP server does not allow anonymous searches of the CRL, enter the admin DN and password that are required to authenticate into the CDP server.

Manually configured CDP—When you select this option, the Infranet Controller accesses the CDP that you specify. Enter the URL of the primary CDP and optionally of a backup CDP. For an LDAP server, use the syntax: ldap://Server/BaseDN?attribute?Scope?Filter. For a Web server, enter the complete path to the CRL object. For example: http://domain.com/CertEnroll/CompanyName%20CA%20Server.crl

Additionally, if the CDP server does not allow anonymous searches of the CRL, enter the admin DN and password that are required to authenticate into the CDP server. (LDAP only)

NOTE: If you choose to download CDPs using one method and then select a different method, the Infranet Controller deletes any CDPs from disk that were downloaded using the previous method.

5. In the CRL Download Frequency field, specify how often the Infranet Controller should download the CRL from the CDP. The allowable range is from 1 to 9999 hours.

6. Click Save Changes.

7. If you want to check the validity of your CA certificate (in addition to client-side certificates) against the CRL specified in the previous steps, select Verify Trusted Client CA on the Trusted Client CA page.

NOTE:

When you choose to verify an intermediate certificate, make sure that CRLs are available for all of the CA certificates that are above the intermediate certificate in the chain—when verifying a CA certificate, the Infranet Controller also verifies all issuing CAs above the certificate in the chain.

If you select this option but do not enable CRL checking, the Infranet Controller checks the CA certificate against the CDP for the CA’s issuer. If no CRL is enabled for the issuer, user authentication fails.

8. Click Save Changes. The Infranet Controller downloads the CRL using the method you specified (if applicable) and displays CRL checking details (described in the following section).

9. Click Update Now in the Trusted Client CA page to manually download the CRL from the CDP (optional).
Enabling OCSP

The Online Certification Status Protocol (OCSP) offers you the ability to verify client certificates in real-time. Using OCSP, the Infranet Controller becomes a client of an OCSP responder and forwards validation requests for users, based on client certificates. The OCSP responder maintains a store of CA-published CRLs and maintains an up-to-date list of valid and invalid client certificates. Once the OCSP responder receives a validation request from the Infranet Controller (which is commonly an HTTP or HTTPS transmission), the OCSP responder either validates the status of the certificate using its own authentication database or calls upon the OCSP responder that originally issued the certificate to validate the request. After formulating a response, the OCSP responder returns the signed response to the Infranet Controller and the original certificate is either approved or rejected, based on whether or not the OCSP responder validates the certificate.

This section contains the following OCSP instructions:

- “Specifying OCSP Options” on page 376
- “Specifying OCSP Responder Options” on page 377

Specifying OCSP Options

If you selected either Use OCSP or Use OCSP with CRL fallback in the procedures specified in “Specifying Attributes for the Trusted Client CA Certificate” on page 368, the Infranet Controller displays a list of known OCSP responders and enables you to configure OCSP responder options:

1. Delete, enable, or disable OCSP Responder configuration using the Delete, Enable, or Disable buttons, respectively.

2. If you want to configure OCSP options, click OCSP Options. The OCSP Options page appears.

3. Specify the type of OCSP responder the Infranet Controller uses to validate trusted client CAs in the Use drop-down list:

- None—The Infranet Controller does not use OCSP to verify the status of certificates issued by this CA.

- Responder(s) specified in the CA certificate—The Infranet Controller uses OCSP responders specified in the imported client CA to perform verification. When you select this option, the Infranet Controller displays a list of OCSP responders specified in the imported CA (if any) and the last time they were used.

- Responder(s) specified in the client certificates—The Infranet Controller uses responders specified during client authentication to perform verification. When you select this option, the Infranet Controller displays a list of known OCSP responders (if any) and the last time they were used.

- Manually configured responders—The Infranet Controller uses primary and secondary OCSP responders at the addresses you specify.
4. Under the **Options** section, specify whether or not the Infranet Controller signs the certificate validation request with an identifier and whether or not the Infranet Controller uses Nonce during verification.

**NOTE:** A *nonce* is random data the Infranet Controller includes in an OCSP request and the OCSP Responder returns in the OCSP response. The Infranet Controller compares the nonce in the request and response to ensure that the response is generated by the OCSP responder. If the two do not match, the Infranet Controller disregards the response and sends a new request. Nonces are a common way of prevent replay attacks.

5. Click **Save Changes**.

**Specifying OCSPResponder Options**

To specify OCSP Responder Signer Certificate options for one or more OCSP responders:

1. Click the name of the OCSP responder you want to configure in the OCSP responders list. The option specification page for the OCSP responder appears.

2. Browse to the network path or local directory location of a Responder Signer Certificate. This is the certificate the OCSP responder uses to sign the response. You must specify the Responder Signer Certificate if the signer certificate is not included in the response.

3. If you want to allow an OCSP responder certificate that matches the responder signer certificate, activate the **Trust Responder Certificate** checkbox.

4. Enable the **Revocation Checking** option to ensure that the certificate the Infranet Controller and OCSP responder are using has not recently been revoked. This option only has any implications if you specified the **Use OCSP with CRL fallback** option in the procedures outlined in “Specifying Attributes for the Trusted Client CA Certificate” on page 368.

5. Specify a clock discrepancy value in the **Allow clock discrepancy** field to account for possible mismatches in timestamps between the Infranet Controller and the OCSP responder. If the mismatch is significant enough, the Infranet Controller simply disregards the response from the OCSP responder as out-of-date or expired.

6. Click **Save Changes**.

**Using Trusted Server CAs**

In the following situations, you may need to install the CA certificate of the servers that you trust on the Infranet Controller:
Whenever users install Odyssey Access Client by accessing the Infranet Controller by means of a Web browser, the **Validate server certificate** option in the **Authentication** tab of the user’s profile is automatically enabled. (For more information, see “Creating an initial Configuration of Odyssey Access Client” on page 44.) When this option is enabled, Odyssey Access Client validates the server certificate of the Infranet Controller. Odyssey Access Client is automatically configured to trust the Infranet Controller if it can verify that the Infranet Controller is passing a valid certificate. In order for this verification to occur, the trusted root certificate of the CA that signed the Infranet Controller’s server certificate must be installed on the endpoint. If the CA certificate is not installed, then the user is unable to authenticate.

There are three ways to install the trusted root CA certificate on the endpoint:

- You can use a CA certificate that is chained to a root certificate that is already installed on the endpoint, such as Verisign.

- You can upload the CA certificate and any intermediate CA certificates to the Infranet Controller. During Odyssey Access Client installation, the Infranet Controller automatically installs the trusted root device CA certificate(s) on the endpoint. When prompted during installation, the user must allow the installation of the CA certificate(s). For more information on uploading the trusted root CA certificate(s) to the Infranet Controller, see “Uploading Trusted Server CA Certificates” on page 379.

- Users or you can import the CA certificate(s) on the endpoint using Internet Explorer or other Microsoft Windows tools through whatever method your organization uses to distribute Root Certificates.

- If you are using third-party IMVs that are installed on a remote IMV server, you must upload to the Infranet Controller the trusted root certificate of the CA that signed the Remote IMV server’s server certificate. For more information, see “Using third-party integrity Measurement Verifiers” on page 299.

- If you are using virus signature version monitoring with your own staging site for storing the current virus signatures list, you must upload the trusted root certificate of the CA that signed the staging’s server certificate to the Infranet Controller. For more information, see “Configuring Virus Signature Version Monitoring and Patch Assessment Data Monitoring” on page 284.

---

**NOTE:** All of the trusted root CAs for the Web certificates installed in Internet Explorer 6.0 and Windows XP service pack 2 are pre-installed on the Infranet Controller appliance.

This section contains the following trusted server CA instructions:

- “Uploading Trusted Server CA Certificates” on page 379
- “Renewing a Trusted Server CA Certificate” on page 379
- “Deleting a trusted server CA Certificate” on page 380
“Viewing Trusted Server CA Certificate Details” on page 380

**Uploading Trusted Server CA Certificates**

Use the **System > Configuration > Certificates > Trusted Server CAs** tab to upload the trusted root certificate of the CA that signed the Infranet Controller’s server certificate into the Infranet Controller. The Infranet Controller supports X.509 CA certificates in PEM (Base 64) and DER (binary) encode formats.

---

**NOTE:**

- When uploading a certificate chain to the Infranet Controller, you must either install the certificates one at a time in descending order starting with the root certificate (DER or PEM files), or you must upload a single file to the Infranet Controller that contains the entire certificate chain (PEM files only). By using one of these methods, you ensure that the Infranet Controller can link the certificates together in the correct order.

- The Infranet Controller does not support CRL revocation checks for trusted server CA certificates.

---

To upload CA certificates to the Infranet Controller:

1. In the admin console, choose **System > Configuration > Certificates > Trusted Server CAs**.
2. Click **Import Trusted Server CA**.
3. Browse to the CA certificate that you want to upload to the Infranet Controller and click **Import Certificate**.

**Renewing a Trusted Server CA Certificate**

If a trusted CA renews its certificate, you must upload the renewed CA certificate to the Infranet Controller.

To import a renewed CA certificate into the Infranet Controller:

1. In the admin console, choose **System > Configuration > Certificates > Trusted Server CAs**.
2. Click the link that corresponds to the certificate that you want to renew.
3. Click **Renew Certificate**.
4. Browse to the renewed CA certificate that you want to upload to the Infranet Controller and click **Import Certificate**.
Deleting a trusted server CA Certificate

You can delete any trusted server CA certificate that is installed on the Infranet Controller, including the pre-installed certificates for Internet Explorer 6 and Windows XP service pack 2.

To delete a trusted server CA certificate from the Infranet Controller:

1. In the admin console, choose **System > Configuration > Certificates > Trusted Server CAs**.

2. Select the checkbox next to the certificate you want to delete.

3. Click **Delete** and then confirm that you want to delete the certificate.

Viewing Trusted Server CA Certificate Details

You can view a variety of details about each of the CA certificates installed on the Infranet Controller.

To view trusted server CA certificate details:

1. In the admin console, choose **System > Configuration > Certificates > Trusted Server CAs**.

2. Click the certificate that you want to view.

3. Under **Certificate**, use the arrow next to the following field names to view certificate details:
   - **Issued To**—Name and attributes of the entity to whom the certificate is issued.
   - **Issued By**—Name and attributes of the entity that issued the certificate. Note that the value of this field should either match the **Issued To** field (for root certificates) or the **Issued To** field of the next certificate up in the chain (for intermediate certificates).
   - **Valid Dates**—Time range that the certificate is valid. If your certificate is expired, see the instructions in “Importing a Renewed Certificate That Uses the Existing Private Key” on page 358.
   - **Details**—Includes various certificate details, including its version, serial number, signature algorithm, CRL distribution points, public key algorithm type, and the public key. (Note that the Infranet Controller does not support CRL checking for trusted server CA certificates.)
Chapter 12
System Archiving

The Infranet Controller provides different ways to backup and restore configuration files containing user and system data. The Infranet Controller utilities you can use to backup and restore data preserve the configuration data in two different formats: binary and XML. The method you choose to use depends on your requirements.

- “Archiving Infranet Controller Binary Configuration Files” on page 381
- “Creating Local Backups of Infranet Controller Configuration Files” on page 384
- “Importing and Exporting Infranet Controller Configuration Files” on page 385
- “Importing and Exporting XML Configuration Files” on page 388
- “Strategies for Working With XML Instances” on page 396
- “Pushing Configurations from one Infranet Controller to Another” on page 409

Archiving Infranet Controller Binary Configuration Files

The Infranet Controller enables you to use SCP (Secure Copy) or FTP to automatically archive a binary copy of your system logs, configuration files, and user accounts on a daily or weekly basis. The Infranet Controller encrypts the configuration files and user accounts to ensure secure transmission and secure storage on other servers, and then archives the files to the server and directory you specify on the chosen day(s) and time.

If the archive process fails, archiving restarts at the next scheduled time. The Infranet Controller does not continue to retry the process if it fails. Log files are not deleted if the archive process fails.

Automatic archiving occurs only at the scheduled time. No “unscheduled” archiving is done automatically. For example, if a log file exceeds the maximum file size, the archiving process does not automatically backup the file prior to the scheduled time to prevent data loss.

SCP is a file transfer utility similar to FTP. SCP encrypts all data during transfer. When the data reaches its destination, it is rendered in its original format. SCP is included in most SSH distributions, and is available on all major operating system platforms.
The name of archive files have the following format:

- **System events**: JuniperAccessLog-[clustername] | standalone|- [nodename | hostname | IVSname | root]-[date]-[time]

- **User events**: JuniperEventsLog-[clustername] | standalone|- [nodename | hostname | IVSname | root]-[date]-[time]

- **Administrator events**: JuniperAdminLog-[clustername] | standalone|- [nodename | hostname | IVSname | root]-[date]-[time]

- **System configuration files**: JuniperConf-[clustername] | standalone|- [nodename | hostname | IVSname | root]-[date]-[time]

- **User accounts**: JuniperUserAccounts-[clustername] | standalone|- [nodename | hostname | IVSname | root]-[date]-[time]

Following are some examples of file names. In these examples, “gen” is the cluster name.

- JuniperAccessLog-gen-node1-Root-20090109-1545.gz
- JuniperEventsLog-gen-node1-Root-20090109-1545.gz
- JuniperAdminLog-gen-node1-Root-20090109-1545.gz
- JuniperConf-gen-node1-Root-20090109-1545
- JuniperUserAccounts-gen-node2-Root-20090109-1542

To specify archive parameters:

1. In the admin console, choose **Maintenance > Archiving > Archiving Servers.**

2. Under **Archive Settings**, specify the destination server, a directory, and your credentials for that server. Do not include a drive specification for the destination directory, such as: juniper/log.

   - For UNIX computers, although you can specify an absolute or relative path, depending on the user’s home directory, we recommend using a full path instead.
   - For Windows computers, specify a path that is relative to the ftproot directory. We recommend using a full path to the directory.

3. For **Method**, specify SCP or FTP. SCP is the default method.

4. Under **Archive Schedule**, specify one or more of the following components to archive by enabling its associated checkbox:

   - **Archive events log** (For more information, see “Logging and Monitoring Overview” on page 415.)

   - **Archive user access log** (For more information, see “Logging and Monitoring Overview” on page 415.)
5. Specify an archive schedule for each selected component. Through the options for each component, schedule archives on any combination of weekdays including weekends.

NOTE: If you schedule an archival operation to occur during the hour that your system switches to Daylight Savings Time (DST) the operation may not occur as scheduled. For example, if your system is set to change to DST at 1:00 a.m. and you have scheduled an archival operation to occur at anytime between 1:01 a.m. and 1:59 a.m., the operation is not accomplished, because at 1:00 a.m. the system clock is moved forward to 2:00 a.m. and the system never reaches your archival time for that date.

6. Define a specific time when you want the Infranet Controller to archive data or elect to archive data every hour, which produces twenty-four files with unique timestamps.

NOTE: We recommend you schedule an archival operation during hours when traffic is light in order to minimize its impact to your users. The automatic archiving process compresses files and, if the system is busy, can degrade performance for users. Also, a cluster node may appear unresponsive if the system is busy with traffic and performing archiving simultaneously.

7. Select a log filter from the drop-down list. See “Custom Filter Log Files” on page 417 for information about the filter types.

8. Specify to clear system events, access, and administrator log files after archiving (optional).

NOTE: If an archive process fails, log files are not deleted.

9. Provide a password if you want to encrypt system configuration or user account archives with a password (optional).

10. Click Save Changes.
Creating Local Backups of Infranet Controller Configuration Files

You can save backups of your current system configuration and user accounts directly to the Infranet Controller in binary format. You may then use these configurations to restore the Infranet Controller or a cluster of Infranet Controllers to the state contained in the encrypted file. Note that these files only contain configuration information—they do not include logs.

NOTE: During an import operation to a cluster node, the sync rank of the node may change temporarily to allow the propagation of the imported data to all nodes. The sync rank will be returned to its original value after the import operation is complete.

You may save up to 5 system configuration backups and 5 user account backups on the Infranet Controller. If you try to exceed this limit, the Infranet Controller overwrites the oldest backup with the new backup. If you do not want to overwrite the oldest backup, choose another backup to delete instead, before saving the most current one.

To save your current system configuration:

1. In the admin console, choose Maintenance > Archiving > Local Backups.

2. Click Save Configuration or Save User Accounts. The Infranet Controller adds a new backup to the list, naming it with the current date and time.

You may use system and user backups to update a single Infranet Controller or a cluster. If you choose to restore an Infranet Controller that is enabled as part of a cluster, that Infranet Controller automatically pushes the configuration to all other cluster members. The cluster is disabled until all cluster members have updated their settings using the backup configuration. Then, they restart and re-enable the cluster.

You can save a backup of your current configuration or to restore your system or user account state from a backup, as explained in “Creating Local Backups of Infranet Controller Configuration Files” on page 384.

To override your configuration with settings from a backup file:

1. In the admin console, choose Maintenance > Archiving > Local Backups.

2. Select the checkbox next to the system configuration or user account backup file that you want to use to restore your system.
3. If you are restoring from a system configuration, indicate whether or not you want to use the certificate, IP address, and network settings contained in the configuration file.

**NOTE:** If you are upgrading an entire cluster, you should use caution when including network settings. Since IP addresses and other settings may not apply to all members of the cluster, cluster members may not be able to communicate with one another if the settings are pushed out to all members.

4. Click Restore. The Infranet Controller must restart before changes can take effect. After the Infranet Controller restarts, you must sign back in to the Infranet Controller in order to access the admin console.

**Importing and Exporting Infranet Controller Configuration Files**

The Infranet Controller enables you to import and export Infranet Controller system and network settings using binary Infranet Controller configuration files. When importing a system configuration file, you can exclude the device certificate and the Infranet Controller server’s IP address or network settings from the imported information. For example, to set up multiple Infranet Controllers behind a load balancer, import everything except for the IP address. To set up an Infranet Controller as a backup server, import everything except for the digital certificate and the network settings.

**NOTE:**

- When importing a configuration file that contains licenses, the Infranet Controller gives precedence to any existing licenses that are currently installed on the Infranet Controller. Archived licenses are only imported if no licenses currently exist on the Infranet Controller.

- When importing certificates, note that the Infranet Controller only imports device certificates—not the chains corresponding to the device certificates or trusted client CAs.
The Infranet Controller also enables you to import and export all local user accounts you have defined for any local authentication servers.

**NOTE:**

- If you want to export resource policies including all Infranet Enforcer, Odyssey Access Client, or Host Enforcer configuration information, you must export user accounts, not the system settings. You can export resource policies on the **Maintenance > Import/Export > User Accounts** tab. For more information, see “Exporting Local User Accounts or Resource Policies” on page 387.

- To export or import client-side logs, export or import both the system and user configuration files.

- Sensor configurations are included in the system configuration file while sensor event policies are included in the user configuration file. To export or import sensor-related configuration to an Infranet Controller, export or import both the system and user configuration files.

The user configuration file, not the system configuration file, includes resource profiles, resource policies, and the local user database. To perform a complete backup, export both the system and user configuration files.

---

**Exporting a System Configuration File**

Typically, if a UI menu item falls under Authentication, Administration, Users, or UAC, the item is included in the user.cfg file. The exception is Sensors event policies which are under system, but appear in the user.cfg file.

Export the system configuration file to export:

- Network settings
- Cluster configuration
- Licenses
- SNMP settings
- Sensor configuration

To export a system configuration file:

1. In the admin console, choose **Maintenance > Import/Export > Configuration**.

2. Under **Export**, enter a password if you’d like to password-protect the configuration file.

3. Click **Save Config As** to save the file.
Importing a System Configuration File

**NOTE:** Existing node-specific settings are erased when an Infranet Controller node joins a cluster. These settings including network interface addresses, route tables, virtual ports, ARP caches, VLAN interface, SNMP settings, and so forth. The administrator must manually re-configure these settings for the newly-joined node. You cannot use the Import system configuration feature to import these configurations and settings onto an Infranet Controller node that has been joined to the cluster.

To import a configuration file:

1. Choose Maintenance > Import/Export > Import/Export Configuration in the admin console.

2. Specify whether you want to import the Infranet Controller device certificate. The certificate is not imported unless you check the Import Device Certificate(s)? checkbox.

3. Choose one of the following import options.

   - **Import everything (except Device Certificate(s))**—This option imports all configuration setting except Infranet Controller device certificates.

   - **Import everything but the IP address**—This option excludes only the IP address from the imported configuration file. If you exclude the IP address, the server’s IP address does not change when you import the file. When you select this option, the Infranet Controller also imports any SNMP settings that you have defined. In other words, choosing this option preserves the IP address, netmask, default gateway, VIPs, ARPs, and routes of the network interfaces on the target device.

   - **Import everything except network settings and licenses**—This option imports all configuration settings except the network settings. If you exclude the network settings, the information on the System > Network page (internal port, external port, and static route settings) does not change when you import the file. When you select this option, network configurations, licenses, cluster configurations, certificates, defined SNMP settings and syslog configurations are not imported.

   - **Import only Device Certificate(s)**—This option imports only the Infranet Controller server certificates. Be sure to enable the Import Device Certificate(s)? checkbox when using this option.

4. Browse to the configuration file, which is named system.cfg by default.

5. Enter the password you specified for the file. If you did not specify a password before exporting the file, then leave this field blank.

6. Click Import Config.

Exporting Local User Accounts or Resource Policies

Export the user accounts if you want to export:
- Sign in settings (includes sign in policies, sign in pages, all authentication servers, authentication protocol sets, and Odyssey Access Client settings)
- Authentication realms (including admin realms, user realms and MAC authentication realms)
- Roles
- Infranet Enforcers
- Network Access
- Host Enforcer
- Resource policies
- Sensor event policies
- User accounts

To export local user accounts or resource policies:

1. In the admin console, choose Maintenance > Import/Export > Import/Export User Accounts.
2. Under Export, enter a password if you’d like to password-protect the configuration file.
3. Click Save Config As to save the file.

**Importing Local User Accounts or Resource Policies**

To import local user accounts or resource policies:

1. In the admin console, choose Maintenance > Import/Export > Import/Export Users.
2. Browse to the configuration file, which is named user.cfg by default.
3. Enter the password you specified for the file. If you did not specify a password before exporting the file, then leave this field blank.
4. Click Import Config.

**Importing and Exporting XML Configuration Files**

The XML Import/Export feature enables you to make significant changes to your system configuration and provides a number of benefits, particularly when it comes to making a large number of repetitive changes, or when you want to add, update, and delete configuration data all at once.

Some of the tasks you might want to perform using exported XML configuration files include:
- Adding a large number of users.
- Deleting all or many of your auth servers, users, or other Infranet Controller objects.
- Tracking configuration changes by performing a diff on weekly exports.
- Modifying multiple instances of a single setting, for example, an auth server name.
- Creating a configuration template to use when setting up new Infranet Controller appliances.

**NOTE:** You can only export and import XML instance files between Infranet Controllers that have the same version of the Infranet Controller system software. You cannot use the XML Import/Export feature to upgrade an older product release from configuration files exported from a new product release. You also cannot downgrade a newer product release using configuration files exported from an older release of the product.

The Infranet Controller enables you to export several types of configuration data, including some network settings, sign-in settings, auth servers, realms, roles, resource policies, and users. You can then import those settings into the same or another Infranet Controller.

**NOTE:** When importing AD authentication server configuration with an XML file or through Push Config, the Computer Object name needs to be changed manually after the import. Unexpected problems might arise if two systems join an AD domain using the same Computer Object name.

You can export XML configuration files containing settings in the following list. Additional settings may also be available.

- **System Settings**—Licenses, certificates, nodes, node identifiers, DNS servers, DNS Domains, hosts, NICs, NIC identifiers, ARP cache, ARP ping timeout, default gateway, IP address, MTU, NIC name, net mask, static routes, link speed, NIC type, host name, and WINS address SNMP settings, including trap settings and limits.

**NOTE:** You must never modify the two NIC identifiers in the XML instance file. The Infranet Controller relies on knowing that each appliance has two interface cards, known as NIC0 and NIC1.

The identifiers appear in the NIC elements `<NICIdentifier>0</NICIdentifier>` and `<NICIdentifier>1</NICIdentifier>`. 
- **Sign-in Settings**—Authentication servers, password options, password management options, standard sign-in pages, custom text, header options, custom error messages, help options, page name, sign-in URLs, and page type.

- **Endpoint Security**—Host Checker policies, ESAP and Remote IMV servers and IMVs.

- **Authentication Realms**—User and admin realms, realm names, realm types, server settings, dynamic policy evaluation settings, authentication policies, limits, password policies, role mapping settings, and role processing option.

- **Roles**—User roles, admin roles, role names, enabled features, restrictions, session options, UI options, Admin system options, and resource policy settings.

- **Infranet Enforcer**—Resource Access, IPsec routing, auth table mapping.

- **Network Access**—RADIUS Dictionary, RADIUS Vendor, Location Group, RADIUS Client, RADIUS Attributes.

- **Local User Accounts**—Users, auth server name, email address, full name, login name, password, change password option, user status, and user type.

- **Maintenance Settings**—System options, push configuration targets, archiving and snapshot.

---

**NOTE:** These lists may not be a complete listing of available settings. For a complete list of supported settings, consult the XML Import/Export page and the Push Config page on the admin console.

The basic process for exporting and importing an XML configuration file is as follows:

1. Choose the configuration settings you want to modify.
2. Export the file from the Infranet Controller.
3. Open the file and edit configuration data in a text editor.
4. Save and close the file.
5. Import the file to the Infranet Controller.

You can learn more about XML configuration files and how to use them in the sections that follow:

- “Creating and Modifying XML Instances” on page 391
- “Strategies for Working With XML Instances” on page 396
- “Strategies for Working With XML Instances” on page 396
Creating and Modifying XML Instances

When you export your configuration file, the Infranet Controller saves the file as an XML instance. The instance is the file you will modify.

The XML file uses the same schema as push config and NSM. You can use XML files to troubleshoot instances of these files.

The XML Instance

Upon export, the instance file shows you the current state of the Infranet Controller configuration. The XML instance is based on an XML schema. The schema is a separate file that defines the metadata, and which serves as a model or a template for the instance file. You will only use the schema file for reference purposes, if at all.

The data in the instance file is based on the selections you make on the XML Import/Export tab in the admin console when you perform the export operation.

Instance files usually end with the .xml file extension.

Creating an Instance File

You can create an instance file by exporting an XML configuration file from the Infranet Controller. Even if you want to replace all of your existing configuration settings for a given object, you should start with an exported instance file. The exported instance file contains all of the required XML processing instructions and namespace declarations, which must be included exactly as defined.

To export an XML configuration file, see “Strategies for Working With XML Instances” on page 396.

Editing the Instance File

All of the Infranet Controller’s XML instance files share a similar structure. Once you become familiar with the basic structure, you should be able to navigate the files easily. The files can become large, so you might find it more efficient to use a commercial or open source XML editor. XML editors often separate the editable data from the structural display. This separation reduces or eliminates the chance of accidentally modifying an XML element rather than its data, which is possible when editing in a simple text editor.

Despite the potential advantages of using an XML editor, you can do an adequate job of editing your configuration data using a simple text editor.

Instance Elements

An element is a discrete XML unit that defines an Infranet Controller object or part of an object. The element typically consists of a pair of tags that may or may not surround string data. Tags are delimited by angle brackets (< >). You can find several examples of tags in the following discussion.

Every tag fits into one of the following tag types:
- **Start tag**—Defines the beginning of an element. The start tag consists of an open angle bracket ( `< ` ), a name, zero or more attributes, and a close angle bracket ( `>` ). Every start tag must be followed by an end tag at some point in the document.

- **End tag**—Defines the end of an element. The end tag consists of an open angle bracket and a forward slash ( `< /` ), followed by the same name defined in its corresponding start tag, and ends with a close angle bracket ( `>` ).

- **Empty tag**—The empty tag is denoted in two forms. If a tag pair has no data between them, the tag pair is considered an empty tag. Officially, according to the XML specification, an empty tag looks something like this:

  ```xml
  <empty tag example/>
  ```

  In this form, the empty tag consists of an open angle bracket ( `< ` ), followed by an element name, a slash and a close angle bracket ( `/>` ). When you see an empty tag in your configuration files, it signifies an element that the schema requires to be included in the instance file, but whose data is optional.

Start tags can contain attributes, and tag pairs (elements) can contain additional elements. The following example shows an XML instance file for the Users object. In this example, you see only the Administrator configuration settings. Italicized items signify user data.

```xml
<configuration xmlns="http://xml.juniper.net/ive-sa/6.2R1"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <authentication>
    <auth-servers>
      <auth-server>
        <local>
          <users>
            <user>
              <username>admin</username>
              <fullname>Platform Administrator</fullname>
              <password-encrypted>3u+U</password-encrypted>
              <one-time-use>false</one-time-use>
              <enabled>true</enabled>
              <change-password-at-signin>false</change-password-at-signin>
            </user>
          </users>
        </local>
        <name>Administrators</name>
      </auth-server>
    </auth-servers>
  </authentication>
</configuration>
```

You make your changes to the string data that appears between start and end tags. For example, from the preceding example, you can add to or change the following elements:

- `<name>Administrators</name>`
- `<fullname>Platform Administrator</fullname>`
- `<username>admin</username>`
The preceding sample displays the Password element’s data as encrypted data, indicating that you cannot change the password value. By default, the XML export operation provides encrypted passwords with a password-encrypted. You can modify the password, if you change the element to password-cleartext. If you modify the password in the instance file, the password value is visible until it is imported back into the Infranet Controller. Once imported, the Infranet Controller encrypts the password.

If you enter passwords for new users in cleartext format, the passwords are visible in the instance file, therefore, you might consider setting the Change Password at Next Login option to true.

NOTE:
- Because passwords are encrypted by default, they are fully portable from one system to another.
- You should never attempt to encrypt a password manually in the XML file. The Infranet Controller rejects any attempt to do so. Use the password-cleartext and enter a text password when changing passwords through the XML file.

Namespaces
Namespaces allow you to use the same words or labels in your code from different contexts or XML vocabularies. Prefixing elements with namespace qualifiers allows an instance file to include references to different objects that originate in different XML vocabularies and that share the same name. If you do not prefix elements with namespace qualifiers, the namespace is the default XML namespace and you refer to element type names in that namespace without a prefix.

A namespace declaration looks like:

```xml
<configuration xmlns="http://xml.juniper.net/ive-sa/6.2R1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
```

When you see namespace identifiers in your instance files, you do not need to be concerned about them, as long as you do not delete or modify the identifiers.
**Element Sequence**
You should avoid changing the sequence of elements in the instance file, whenever possible. Although the schema does not enforce sequence in all cases, you gain no benefit from changing the order of elements from the order in which they appear in the exported instance file, and, in some cases, you might invalidate an instance document by changing element sequence.

**Referential Integrity Constraints**
Infranet Controller configuration objects are part of a data model that is enforced through the use of referential integrity constraints. You cannot change these constraints, but you should understand them before you attempt to delete objects that maintain dependencies to other objects.

If you violate the Infranet Controller referential integrity constraints, your import operation fails. The following diagram illustrates the relationships among several Infranet Controller objects.

**Figure 24: Infranet Controller Object Referential Integrity Constraints**

In Figure 24 the boxes represent Infranet Controller object types and the arrows represent dependent relationships between the object types. Arrows point from dependent objects to objects.

The system does not allow you to delete an object upon which another object depends. Conversely, when you add an object, you must add any other objects upon which that object depends.

In Figure 24, sign-in URLs depend upon realms and sign-in pages. Realms depend upon both auth servers and roles. Policies depend upon roles. Users depend upon auth servers.

Consider the following scenarios based on Figure 24:

- If you add sign-in URLs, you must add realms, sign-in pages, roles, and auth servers. You need to add an auth server and at least one role to support the realm, and you need to add the realm and the sign-in page to support the new sign-in URL.

- If you add a user, you must be able to assign it to an auth server. If there is no auth server on the target Infranet Controller yet, you must add one in the instance file.
If you add a policy, you must be able to assign it to a role. If there is no role on the target Infranet Controller yet, you must add one in the instance file.

If you delete an auth server, you might strand realms and users, therefore, you need to make sure no realms or users depend on the auth server before you attempt to delete it.

If you delete a role, you might strand policies and realms. To delete a role, you must first delete any policy that depends upon the role, or reassign associated policies to another role. Also, to delete a role, you must first delete or reassign any realm that depends upon that role.

If you delete a sign-in page, you might strand one or more sign-in URLs. To delete a sign-in page, you must first delete any associated sign-in URLs or reassign them to other sign-in pages.

**NOTE:** Referential integrity checks are performed only during XML Import.

### Mapping the XML Instance to UI Components

The elements in the XML instance are closely related to the objects and their options as you see them in the admin console. The element names in the XML instance file correlate closely with the displayed object and option names.

For example, go to **Users > User Roles > [Role] > General > Session Options** in the admin console. The admin console renders the possible values for a roaming session as a radio button group, consisting of the values:

- Enabled
- Limit to subnet
- Disabled

The following snippet, from the exported configuration file, shows the session options for the **Users** role. On the bolded line, the roaming session option is set to disabled:

```xml
(SessionOptions>
    <MaxTimeout>60</MaxTimeout>
    <RoamingNetmask />
    <Roaming>Enable</Roaming>
    <PersistentSession>false</PersistentSession>
</SessionOptions>
```

In the schema file, you can locate the allowable values for the roaming session option:

```xml
<!-Attribute roaming:START->
    <xsd:element name="roaming" minOccurs="0">
        ...
        <xsd:enumeration value="enabled">
            ...
    </xsd:element>
</-Attribute roaming:END>
```
If you want to change the value for the roaming session from disabled to limit to subnet, replace `disabled` with `limit-to-subnet`.

This example shows you that the admin console often provides all of the allowable values, displayed either in a radio button group, as checkboxes, as drop down listboxes, or as other types of user interface components. The instance file displays only the current state of your Infranet Controller configuration. The schema file displays all of the actual values for the configuration options that are supported in the XML Import/Export feature.

For more information about specific elements, review the schema files directly.

**Downloading the Schema File**

You can download the schema (.xsd) file for the Infranet Controller objects, if you want to review the structure and rules that apply to the objects.

You can download the schema file in two ways:

- From the XML Import/Export pages, by clicking a hyperlink.
- Directly, by accessing the URL where the files are stored on the system.

To access the .xsd file, access the following URL, either directly or by way of a script:

```
https://<IP-or-hostname>/dana-na/xml/config.xsd
```

where `<IP-or-hostname>` is the Infranet Controller’s IP address or hostname. Using this method, you do not need to sign in to the Infranet Controller.

---

**NOTE:** This feature might change in the future. Be aware of this if you use scripts to access the zip file by way of the URL. The items that might change are:

- The URL.
- The filename.
- The file extension.

---

**Strategies for Working With XML Instances**

The following strategies might be useful to you when exporting and importing XML instance files:

- Define your goal for a given XML Import/Export operation.
- What Infranet Controller object or objects do you need to add, update, or delete?

- Do you need to complete all modifications in one operation, or can you modify the configuration in separate operations?

- Will your process be a one-time operation, or will you need to perform the same operation multiple times?

- Are you updating an existing Infranet Controller, or are you using one Infranet Controller configuration as a template for configuring other Infranet Controllers?

- Document the changes to the Infranet Controller objects you intend to modify.

- Make a list of objects to be added, updated, or deleted.

- For objects to add or update, list specific attribute data.

- List pages or tabs from the admin console that correspond to the objects and attributes you intend to change.

- Make a binary system snapshot or a binary configuration backup immediately before performing the import.

- Make a plan to verify that the completed configuration meets your goals.

- Check the Admin Access log to make sure the export and import operations succeeded.

- Perform a random check of the modified items. Make sure items were added, updated, or deleted as you expected.

You will almost always need to use the XML instance file and the admin console in combination, particularly when you first begin modifying the XML instance files. You may also need to view the XML schema files.

Use the XML instance file to:

- Identify the configuration objects, expressed as XML elements.

- Locate and modify the configuration data.

Use the admin console to:

- Correlate visual components to XML schema and instance elements.

- Confirm the accuracy of modifications to specific objects.

Use the XML schema file to:

- Identify the structure and sequence of configuration objects.
Identify optional and required elements, allowable values, default values, and other attributes of the configuration objects.

**NOTE:** Importing and exporting XML configuration files can take several minutes to complete. Do not perform any operations that might modify or remove data currently being imported or exported.

**Importing XML Configuration Data**

To import XML configuration data:

1. Choose Maintenance > Import/Export > Import XML in the admin console.

2. Under Schema Files, click the link to download the XML Schema (.xsd) files that describe the Infranet Controller objects (Optional). For more information about schema files, see “Downloading the Schema File” on page 396.

3. Browse to, and select, the XML data file that you want to import. You can import a valid XML fragment file if you want to import only a partial configuration.

4. Click Import. The Import XML Results page displays containing information about the imported network settings, roles, resource policies, and other settings.

   If there are errors in the XML, the import operation stops and rolls back the configuration to the previous state. Error messages are displayed on the Import XML Results page.

5. Click OK to return to the Import page.

Please note the following when importing XML configuration data:

- An import resource policy is associated with a role that does not exist on the target Infranet Controller and the role does not exist in the XML file. The XML import process fails.

- An ESAP package does not contain support for a product that is configured in any of the Host Checker rules. Importing the ESAP package fails.

- The following Host Checker conditions must be met, otherwise an error occurs:
  - Custom expressions defined in a Host Checker policy must only reference rules configured within that Host Checker policy.
  - Host Checker policies must contain only rule types valid for that platform. For example, you cannot have Predefined AV rules for Macintosh or Linux platforms.
  - Antivirus, firewall, and spyware policies must contain only the types that are available in the currently activated ESAP packages.
  - You can import a license only on to the same system. You cannot import a license from a different Infranet Controller (an error is logged).
Importing clustering is not supported.

**Exporting XML Configuration Data**

Note the following when exporting XML configuration data:

- Role associations are exported but individual role data is not.
- Exporting CDPs is not supported.
- Third-party policies are exported as the main policy plus the Jedi package (as a zip file) and a set of sub-policies. You can not export any of these items separately.
- ESAP packages are exported as encrypted BLOBs.
- The AV signature data file associated with Virus Signature Version Monitoring is exported as an encrypted BLOB.

To export XML configuration data:

1. Choose **Maintenance > Import/Export > Export XML** in the admin console.
2. (optional) Under **Schema Files**, click the link to download the XML Schema (.xsd) file that describe the Infranet Controller objects (Optional). For more information about schema files, see “Downloading the Schema File” on page 396.
3. Click **Expand All** to view all settings; click **Select All** to select all settings identified on the page. Otherwise, select the specific information you want to export. Within each section, you can click **Select All** to select all settings within that particular section.
4. Select the **System Settings** checkbox to export network settings, including internal port settings, external port settings, and licensing information. Options include:
   - System date and time—exports the server time zone and Network Time Protocol (NTP) settings.
   - Cockpit page—exports settings on the System Status Overview page, such as the graphs to display and the refresh rate.
   - Licenses—exports the licenses in an encrypted format.
NOTE: The following rules apply to exported and imported licenses:

- You cannot edit the license data that is exported, as it is encrypted.
- An XML import of licenses is only valid if the machine importing the license does not currently have a license installed. If there is a license installed already, any imported licenses are dropped. If you still intend to import a license, you must perform a factory reset on the Infranet Controller, then perform the import operation.
- If you import a license after deleting a temporary license from the Infranet Controller, the imported license will be dropped, because you might still be able to reactivate the deleted license and the import operation attempts to preserve any licensing data currently on the Infranet Controller.

- Sensors—exports sensor events and sensor policies.
- Security—exports security configuration settings, including SSL and TLS versions, encryption strength, SSL handshake timeout value, and cookie options.
- Overview—exports settings on the Network Settings Overview tab, such as DNS, WINS and bandwidth management.
- Internet Port—exports settings on the Network Settings Internal Port tab, such as IP address, netmask, default gateway, link speed, ARP ping timeout, and MTU.
- External Port—exports settings on the Network Settings External Port tab, such as IP address, default gateway, link speed, ARP ping timeout and MTU.
- VLANs—exports settings on the Network Settings VLAN tab, such as name, ID, IP address, netmask and gateway.
- Routes—exports settings on the Network Settings Routes tab, such as IP address, netmask, gateway and interface.
- Hosts—exports settings on the Network Settings Hosts tab, such as IP address and name.
- Clustering—exports clustering properties, such as the cluster name, configuration settings, synchronization settings, and network healthcheck settings. This option is visible only when the device is part of a cluster.
- Events—exports event log settings, such as max log size, syslog server, and which event to log.
- User Access—exports user access log settings, such as max log size, syslog server, and which event to log.
- Admin Access—exports admin access log settings, such as max log size, syslog server, and which event to log.
- Sensors—exports sensor log settings, such as max log size and syslog server.

- Client Logs—exports client log settings, such as which client-side feature to log and disk space size.

- SNMP—export SNMP log settings, including node name, system name and location, trap settings and limits.

5. Select the **Sign-in Settings** checkbox to export Authentication servers, password options, password management options, standard sign-in pages, custom text, header options, custom error messages, help options, page name, sign-in URLs, and page type.

   - From **Sign-in URLs**, choose **ALL sign-in URLs** to export all sign-in URLs or choose **SELECTED sign-in URLs** to specify which sign-in URLs to export.

   - From **Sign-in Pages**, select **ALL Pages** to export all sign-in pages, **SELECTED pages** to specify which sign-in pages to export, or **ONLY pages used by URLs selected above** to export only those pages that are valid for the sign-in URLs selected above.

   - From **Authentication servers**, select **ALL auth servers** to export all authentication servers or **SELECTED auth servers** to export which authentication servers to export.

6. Select the **Endpoint Security** checkbox to export Host Checker and Cache Cleaner settings.

   - From **Host Checker**, choose **Host Checker options** to export settings in the Endpoint Security Host Checker tab, including live update settings and interval and timeout values. The exported antivirus signature data file associated with the virus signature version monitoring option is encrypted.

   - Select **ALL policies** to export all Host Checker policies or choose **SELECTED policies** to specify which Host Checker policies to export. JEDI package files are encrypted. Third-party policies are exported as the main policy plus the JEDI package plus any sub-policies. You cannot separate out these packages for export.

   - Select **Remote IMV** to export all remote IMV servers settings and remote IMV rules.

   - Select **ESAP** to export all ESAP settings. ESAP packages are encrypted when exported.

7. Select the **Authentication Realms** checkbox to export administrator and user authentication realms.

   Within each group,

   - Choose **ALL resource realms** to export all realms within that group.

   - Choose **SELECTED realms**, select realms from the **Available Reams** list, and click **Add** to export only those selected authentication realms.
8. Select the **Roles** checkbox to export admin and user roles.

   - Choose **ALL roles** to export all roles within that group.
   
   - Choose **SELECTED roles**, select roles from the **Available Roles** list, and click **Add** to export only those selected roles.

9. Select the **Resource Profiles** checkbox to export resource profile settings, including the list of associated resource policies, bookmarks and roles.

   Select **Hosted Java Applets** to export all applets that have been uploaded to the system. You can not select individual Java applets to export.

   Within each group,

   - Choose **ALL resource profiles** to export all resource profiles within that group.
   
   - Choose **SELECTED resource profiles**, select profiles from the **Available Profiles** list, and click **Add** to export only those selected profiles.

10. Select the **Resource Policies** checkbox to export resource policies. Next, select the checkboxes that correspond to the types of resource policies that you want to export.

11. Select the **Local User Accounts** checkbox to export local user accounts.

    - Choose **From ALL local auth servers** to export all local user accounts from all of the local authentication servers.
    
    - Choose **From SELECTED local auth servers**, select authentication servers from the **Available Servers** list, and click **Add** to export local users from only those authentication servers.

12. Select the **Infranet Enforcers** checkbox to export the Infranet Enforcer connection policy.

    - Choose **From ALL infranet enforcers** to export all Infranet Enforcer connection policies.
    
    - Choose **From SELECTED infranet enforcers**, select the enforcer from the **Available Enforcers** list, and click **Add** to export only from those Infranet Enforcers.

13. Select the **Radius Policies** checkbox to export the Radius location and client connection policies.

    From **Location Group**, select one of the following options:

    - Choose **ALL location groups** to export all Radius location group policies.
    
    - Choose **SELECTED location groups**, select the location from the **Available Location** groups list, and click **Add** to export only those policies.

    From **Radius Clients**, select one of the following options:
Choose **All Radius clients** to export all Radius client connection policies.

Choose **ONLY Clients group by Location Groups selected above** to export only the Radius client policies associated with the selected location groups.

Choose **SELECTED Radius clients**, select the client from the **Available clients** list, and click **Add** to export only those policies.

14. Select the **Maintenance Settings** checkbox to export

- System Options—exports the settings on the System Maintainance Options tab.
- Push Config Targets—exports selected targets and whether this device can accept push configurations.
- Archiving—exports archive settings, such as the archive server, destination directory, username and password, and the components to archive.
- Snapshot—exports system snapshot options, including automatic snapshot settings and whether to include system configuration and debug logs.

15. Click **Export** to save the information in an XML file.

**System Restarts**

While every attempt has been made to reduce the number of restarts, some changes still require restarting the server. The following table lists the system behavior when certain options are changed. Restarts occur after you change and save these settings and when you import an XML configuration that contains different values.

**Table 34: System Behavior When Editing Options**

<table>
<thead>
<tr>
<th>Window</th>
<th>System Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>System &gt; Status &gt; Overview</td>
<td>All processes update their time zone settings when updating the System Date and Time time zone.</td>
</tr>
<tr>
<td>Authentication &gt; Auth Servers &gt; NIS Server</td>
<td>Restarts Linux YP services</td>
</tr>
<tr>
<td>System &gt; Log/Monitoring &gt; SNMP</td>
<td>SNMP Server restarts</td>
</tr>
<tr>
<td>Authentication &gt; Signing In &gt; Sign-in Policies</td>
<td>Web server restarts</td>
</tr>
<tr>
<td>Maintenance &gt; System &gt; Options</td>
<td>Web server restarts</td>
</tr>
<tr>
<td>System &gt; Configuration &gt; Security</td>
<td>Changing the following options restarts the web server with new SSL settings:</td>
</tr>
<tr>
<td></td>
<td>■ Allows SSL and TLS version</td>
</tr>
<tr>
<td></td>
<td>■ Allowed Encryption Strength</td>
</tr>
<tr>
<td></td>
<td>■ Encryption Strength Option</td>
</tr>
<tr>
<td></td>
<td>■ SSL Handshake Timeout</td>
</tr>
<tr>
<td>Auth Servers</td>
<td>Creating a nete auth server or changing any nete option restarts the nete server</td>
</tr>
</tbody>
</table>
XML Import/Export Use Cases

The following use cases illustrate common examples of how you can use the XML Import/Export feature. Each use case consists of a brief description and a procedure for accomplishing the use case. These use cases are abbreviated and do not cover all of the intricacies and details of performing a full set of procedures. The use cases are included solely as illustrations of the potential uses for the XML Import/Export feature.

Use Case: Adding Multiple New Users to an Infranet Controller

You have just added a new Infranet Controller appliance to your network, and you want to add your two thousand users to the system. You do not want to add them one at a time in the admin console, but would like to perform a mass import and force the users to change their passwords the first time they log in to the system.

You can export the user accounts, extract the relevant XML that defines users, replicate each element as needed, then import them to the Infranet Controller.

In this procedure, you only see examples for User 1, User 2, and User 2000. All other users are assumed as being included in your import file. You set the passwords to numbered instances of the word password, such as password1, password2, and so on. All users in this example are assigned to the same auth server, although you can specify any combination of auth servers that are valid on your system.

To add multiple new users to an Infranet Controller:

1. In the admin console, go to Maintenance > Import/Export > Export XML.

2. Follow the instructions to export local user accounts as described in “Strategies for Working With XML Instances” on page 396.

3. Save the exported file as users.xml.
4. Open the *users.xml* file.

5. Copy and paste the *User* container element until you have added the necessary number of users. Although the example shows only three new users, you might add hundreds of new users to the file.

6. Update the appropriate data in each *User* container element, as shown in the following example:

```xml
<configuration xmlns="http://xml.juniper.net/ive-sa/6.2R1"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
 <authentication>
  <auth-servers>
   <auth-server>
    <local>
     <users>
      <user>
       <username>user1</username>
       <fullname>User1</fullname>
       <password-cleartext>password1</password-cleartext>
       <one-time-use>false</one-time-use>
       <enabled>true</enabled>
       <change-password-at-signin>true</change-password-at-signin>
      </user>
      <user>
       <username>user2</username>
       <fullname>User2</fullname>
       <password-cleartext>password2</password-cleartext>
       <one-time-use>false</one-time-use>
       <enabled>true</enabled>
       <change-password-at-signin>true</change-password-at-signin>
      </user>
     </users>
    </local>
   </auth-server>
  </auth-servers>
 </authentication>
</configuration>
```

7. Save the *users.xml* file.

8. In the admin console, go to Maintenance > Import/Export > XML Import/Export > Import.
9. Click **Browse** and locate your *users.xml* file.

10. Click **Import**.

**Use Case: Using XML Import/Export in a Clustered Environment**

You can use the XML Import/Export feature in a clustered environment. You must, however, adhere to certain rules and you must follow a particular procedure to complete the operation successfully.

- The XML instance you want to import must contain the same set of nodes as the original cluster. The signature used to synchronize the cluster when the nodes are re-enabled is derived from the IP addresses of the cluster nodes, so the remaining nodes cannot rejoin the cluster if the imported configuration yields a different signature.
- Do not modify node name, IP address, or IP netmask in the instance file.
- Do not change any network settings in the instance file that render the primary node unreachable. For example, do not change the default gateway configuration for a multi-site cluster.
- On import, the instance file overwrites the node-specific cluster configuration network settings of the remaining nodes. If you change these node-specific network settings, make sure you do not make the remaining nodes unreachable.
- Do not modify existing virtual port settings or add new virtual port settings in the instance file.
- When performing an import operation on a cluster, all of the cluster nodes should be enabled and running. If you attempt to import a configuration into a cluster in which a node is not running, the import operation may hang or your import results may be unpredictable.

**Using Operation Attributes**

Data editing operations are determined by the operation attribute of the element in imported XML data. Operation attributes define the positioning or action of XML data within the schema. If you do not specify an operation attribute, the modified data is merged by default.

The operation attribute is applied to all children objects unless a new operation attribute is defined in children objects.

XML data with an operation attribute has the following format:

```xml
<object1 xc:operation="operator for object1 and its children unless new operator is defined">
    ...
<object2>
    ...
<object3 xc:operation="operator for object3">
    ...
```
Following are the supported operation attributes:

- **Merge**—The configuration data identified by the element containing this attribute is merged with the configuration at the corresponding level in the configuration datastore identified by the target parameter. This is the default behavior.

- **Replace**—The configuration data identified by the element containing this attribute replaces any related configuration in the configuration datastore identified by the target parameter. Only the configuration actually present in the config parameter is affected.

- **Create**—The configuration data identified by the element containing this attribute is added to the configuration if and only if the configuration data does not already exist on the device.

- **Delete**—The configuration data identified by the element containing this attribute is deleted in the configuration datastore identified by the target parameter.

- **Insert before**—Change the position of a configuration element in an ordered set.

- **Insert after**—Change the position of a configuration element in an ordered set.

- **Rename**—Change the name of one or more of a configuration object’s identifiers.

If you are merging a listing of objects to an existing list of objects in the configuration store, the results of the merged list of objects could be unexpected. During a merge operation the order of the objects in the new list is not maintained. If you are importing a listing of objects and would like to preserve the order of the new list, you should use the replace operation attribute. You can also use insert before or insert after to ensure that you produce the hierarchy that you intended.

Operation attributes are applied to elements recursively unless new operators are also defined within lower level elements. There are limitations on the legal operator that can be used in child elements without conflict with the parent operator. Table 35 displays the legal operator relationships between parent and child elements.

**Table 35: Legal Operation Attribute Relationships**

<table>
<thead>
<tr>
<th>Child &gt; V-Parent</th>
<th>Create</th>
<th>Merge</th>
<th>Replace</th>
<th>Delete</th>
<th>Insert before</th>
<th>Insert after</th>
<th>Rename</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
</tr>
<tr>
<td>Create</td>
<td>OK</td>
<td>OK</td>
<td>Error</td>
<td>Error</td>
<td>OK</td>
<td>OK</td>
<td>Error</td>
</tr>
<tr>
<td>Merge</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
<td>OK</td>
</tr>
</tbody>
</table>
Following are two examples of the import operation:

Example 1: Set the MTU to 1500 on an interface named “Ethernet0/0” in the running configuration.

```xml
<interface>
  <name>Ethernet0/0</name>
  <mtu>1500</mtu>
</interface>
```

Example 2: Add an interface named “Ethernet0/0” to the running configuration, replacing any previous interface with that name.

```xml
<interface xc:operation="replace">
  <name>Ethernet0/0</name>
  <mtu>1500</mtu>
  <address>
    <name>192.0.2.4</name>
    <prefix-length>24</prefix-length>
  </address>
</interface>
```

**General Import Rules**

The default import modes have equivalent attributes on the root object of the configuration tree:

- Standard Import is always a merge operation
- Quick Import is a create operation
- Full Import is a replace operation.
Infranet Controller appliances enable you to copy all configuration settings or selected configuration settings from one Infranet Controller to another using the Push Configuration feature. This feature provides simple configuration management across an enterprise without requiring you to cluster Infranet Controller appliances. With the Push Configuration feature, you can decide exactly which settings you do and do not want to copy across the enterprise. The interface for selecting the settings is similar to the XML Import/Export feature.

You can push to a single Infranet Controller or to multiple Infranet Controllers. For example, if you install several new Infranet Controllers, you can push to set their initial configuration. You can also push to an Infranet Controller that is a member of a cluster as long as the target Infranet Controller is not a member of the same cluster as the source. Target Infranet Controllers have the option of not accepting pushed configuration settings. For instructions, see “Defining the Target Infranet Controllers” on page 410. If a push to a target Infranet Controller fails, Push Configuration continues to the next target until all identified targets are updated. The results page displays the status and any problems encountered during the process.

Note the following when pushing configurations:

- After the Infranet Controller updates the configuration on a target Infranet Controller, the target Infranet Controller appliance restarts its services. Brief interrupts may occur while the service restarts. We recommend you push to target Infranet Controllers when they are idle or when you can accommodate brief interruptions.

- Target Infranet Controllers display no warning message when receiving pushed configurations. Push Configuration updates the Administrator Access log file with the push results.

- The target Infranet Controller automatically logs out administrators during the push process.

- The source and target Infranet Controllers must have the same build version and number.

**NOTE:** When importing AD authentication server configuration with an XML file or through Push Config, the Computer Object name needs to be changed manually after the import. Unexpected problems might arise if two systems join an AD domain using the same Computer Object name.
The source Infranet Controller pushes data only over the internal port. The target Infranet Controller can receive data over the internal or external port.

You can push up to 8 targets per push operation; up to 25 push operations can be run simultaneously. The maximum number of targets the Infranet Controller pushes to at any time is 200.

The source Infranet Controller saves and displays up to 25 push configuration results in the Results tab. If 25 results are currently displayed, the Infranet Controller removes the oldest result data when push configuration runs again.

You can not push the following configurations: licensing, clustering, networking and timezone.

For Push Configuration to work, the administrator account on the source Infranet Controller must sign in to the target Infranet Controller without any human interaction. For example, you cannot have dynamic credentials or multiple roles that are not merged as these both require manual interaction.

Prior to using Push Configuration, you must configure your system following specific conditions:

- You must map to the Administrators role, thereby creating a “super administrator” with full administration privileges. Use settings in the Authentication > Auth Servers > [Administrator Server] > Users tab to add yourself to the Administrators role.

- The target Infranet Controller administrator account must use static password authentication or two-factor tokens that do not use challenge-response type authentication. For example, certificates, Soft ID, and Defender Authentication are not supported. Use settings in the Administrators > Admin Realms > [Administrator Realm] > General tab to select the proper authentication server for the administrator realm.

- You must not configure the administrator account in a way that requires the administrator to select a role to sign in to the target Infranet Controller. For example, you must not map a single user to multiple roles, including the push configuration administrator role, and then fail to permissively merge those roles. We recommend creating an account exclusively for push configuration administrators to guarantee that the administrator does not need to choose a role during the sign-in process and to clearly distinguish the actions of push configuration administrators in your log files. Use settings in the Administrators > Admin Realms > [Administrator Realm] > Role Mapping tab to set the appropriate role-mapping rules.

**Defining the Target Infranet Controllers**

If the target Infranet Controller is part of a cluster, you can push to any member of the cluster as long as the target is not a member of the source cluster. You must enable the Allow this IC to be a target setting on all cluster members. This setting is important when specifying the virtual IP (VIP) in the sign-in URL of a destination as it ensures that the push succeeds regardless of which node is hosting the VIP.

Note the following about target Infranet Controllers.
Target names and target sign-in URLs cannot be edited once they are created.

You cannot edit or delete a target Infranet Controller while the push operation is pushing configuration data to that target Infranet Controller.

When deleting a target Infranet Controller, all push configuration results associated with that target Infranet Controller are also deleted.

To define target Infranet Controllers:

1. Create administrator accounts on both Infranet Controllers. For instructions, see “Creating Administrator Roles” on page 474. (See restrictions in “Pushing Configurations from one Infranet Controller to Another” on page 409.)

2. In the admin console, choose Maintenance > Push Config > Targets.

3. If you do not want this Infranet Controller to accept pushed configuration settings, uncheck the Allow this IC to be a target checkbox.

4. To create a new target Infranet Controller, click New Target. On the New Target page:
   a. In the Name field, enter a name for the target Infranet Controller.
   b. In the Sign-in URL field, enter the sign-in URL defined in the Authentication > Signing In > Sign-In Policies page.
   c. Enter the username, password, and authentication realm of an administrator account on the target Infranet Controller that provides full administration privileges.
   d. Click Save Changes.

5. To delete a target Infranet Controller:
   a. Select the checkbox next to the target Infranet Controller you want to delete.
   b. Click Delete and then confirm that you want to delete the Infranet Controller.

6. Click Save Changes.

**Pushing the Configuration Settings**

To push selected roles, resources, sign-in settings, auth servers, and local users from one Infranet Controller to another:

1. In the admin console, choose Maintenance > Push Config.

2. If you have not set up your target Infranet Controllers, click the Targets tab and follow the instructions in “Defining the Target Infranet Controllers” on page 410.

3. Select one of the following options from the What to push list:
• **Entire configuration** to push all configuration settings, except for the following:
  - Network configurations
  - Licenses
  - Cluster configurations
  - Certificates
  - SNMP settings
  - Syslog server settings
  - Push configuration targets configured on the source Infranet Controller

---

**NOTE:** User bookmarks and preferences from the source Infranet Controller are pushed to all target Infranet Controllers with this option. Any bookmarks and preferences already set on the target Infranet Controllers are overwritten.

---

• **Selected configuration** to choose specific settings to push.

---

**NOTE:** You cannot copy network settings to another Infranet Controller using the Push Configuration feature. You can use the XML Import/Export feature to export selected network settings and then import those settings to another Infranet Controller. For more information, see “Importing and Exporting XML Configuration Files” on page 388.

---

See “Exporting XML Configuration Data” on page 399 for information on the options and settings you can push.

4. Select the target Infranet Controllers from the **Available Targets** list and click **Add** to move them to the **Selected Targets** list.

5. Select the **Overwrite duplicate settings** checkbox if you want to overwrite settings on the target Infranet Controller that have the same name as settings on the source Infranet Controller.

---

**NOTE:**

- If **Overwrite duplicate settings** is **off**, and if the name of any setting in the imported file matches the name of a corresponding setting on the target Infranet Controller, then Push Configuration does not copy the values for that setting to the target Infranet Controller. Push Configuration only copies new objects to the target Infranet Controller.

- If **Overwrite duplicate settings** is **on**, Push Configuration copies all new and updated objects to the target Infranet Controller.
6. Click **Push Configuration** to copy the selected settings to the target Infranet Controllers. The Infranet Controller displays the push status in the **Results** tab.

---

**NOTE:** Once you click **Push Configuration**, you cannot halt the process or change the target Infranet Controllers until the entire push configuration process completes.

---

If there are errors during the push process, the operation stops and rolls back the configuration to the previous state. Error messages are displayed on the **Results** page.

7. Correct the problems described by the error messages and push to the failed target Infranet Controller again.
Chapter 13
Logging and Monitoring

The Infranet Controller provides logging and monitoring capabilities to help you track events and user activities. This chapter describes the various logging and monitoring features included with the Infranet Controller.

This section contains the following information about logging and monitoring features:

- “Logging and Monitoring Overview” on page 415
- “Configuring the Log Monitoring Features” on page 419
- “Configuring events, user access, and Admin Access Logs” on page 419
- “Monitoring the Infranet Controller as an SNMP Agent” on page 424
- “Viewing System Statistics” on page 430
- “Enabling Client-Side Logs” on page 430
- “Viewing General Status” on page 431
- “Monitoring Active Users” on page 434

Logging and Monitoring Overview

Infranet Controller log files are text files stored on an Infranet Controller appliance that track system events. An Infranet Controller appliance produces the following types of log files:

- **Events log**—This log file contains a variety of system events, such as session timeouts (including idle and maximum length session timeouts), system errors and warnings, requests to check server connectivity, and Infranet Controller service restart notifications. (The Infranet Controller Watchdog process periodically checks the Infranet Controller server and restarts it if the Infranet Controller does not respond.)

- **User Access log**—This log file contains information about when users access the appliance, including the number of simultaneous users at each one hour interval (logged on the hour) and user sign-ins and sign-outs.
Administrator Access log—This log file contains administration information, including administrator changes to user, system, and network settings, such as changes to session timeouts and machine and server information. It also creates a log entry whenever an administrator signs in, signs out, or changes licenses on the appliance.

The System > Log/Monitoring pages lets you specify which events are logged, the maximum file size for the system log, and whether to log events to the syslog server in addition to logging them locally. The System > Log/Monitoring pages also let you view the specified number of events, save the log files to a network, and clear the logs.

When one of the logs reaches the configured maximum log file size (200MB by default), the current data is rolled over to a backup log file. A new, empty file is then created for all subsequent (new) log messages. Using the log viewer, the administrator can see the most recent 5000 log messages (the viewer's display limit). If the current log file contains less than 5000 log messages, older log messages from the backup log file are displayed, up to a total of 5000 log messages. This makes the log files appear as one, even though they are stored separately, according to the configured maximum log file size.

Log File Severity Levels

The events, user access, and administrator access log files rank events according to these guidelines:

- **Critical (severity level 10)**—When the Infranet Controller cannot serve user and administrator requests or loses functionality to a majority of subsystems, it writes a critical event to the log.

- **Major (severity levels 8-9)**—When the Infranet Controller loses functionality in one or more subsystems, but users can still access the appliance for other access mechanisms, the Infranet Controller writes a major event to the log.
**Minor (severity levels 5-7)**—When the Infranet Controller encounters an error that does not correspond to a major failure in a subsystem, it writes a minor event to the log. Minor events generally correspond to individual request failures.

**Info (severity levels 1-4)**—When the Infranet Controller displays a notification message, when an end-user makes a request, or when an administrator makes a modification, the Infranet Controller writes an informational event to the log.

### Custom Filter Log Files

The Infranet Controller allows you to filter and format the data in your events, user access, and administrator access log files.

When you filter log files, the Infranet Controller appliance only saves those messages specified within the filter query. For example, you may create a query that only logs entries for a particular range of IP addresses or for users who are signed into a specific realm. To create a query, use the Infranet Controller custom expression language.

When you format log files, the Infranet Controller appliance simply changes the “look” of the log messages based on your specifications. Log formats do not affect which data the appliance saves; formats only affect how the appliance displays the data. An Infranet Controller appliance includes standard, WELF, and W3C log formats, but you may also choose to create your own custom format. To create a custom format, use log fields.

For configuration instructions, see “Configuring the Log Monitoring Features” on page 419.

### Dynamic Log Filters

You can quickly change the log view by clicking on any data log variable link in the currently viewed log. For instance, if you want to temporarily view the User Access Log based on a particular IP address, create a “quick filter” by clicking on any occurrence of that IP address in the current log and the Infranet Controller immediately redraws the log to show all entries containing the specified IP address. Furthermore, clicking on additional data log variable links expands the quick filter and updates the current view of the log.

**NOTE:** As with custom log filters, dynamic log filters change only the current view of the log — not the data that the Infranet Controller saves.

Although quick filters act as temporary filter agents, the Infranet Controller gives you the option of saving the temporary query strings as new custom filters.

For configuration instructions, see “Dynamic Log Filters” on page 417. For more information on Central Manager, see “Using Central Management Features” on page 339.
Viewing and Deleting User Sessions

The configuration page for most Infranet Controller authentication servers contain a Users tab that you can use to view and delete active Infranet Controller user sessions. Authentication server types that do not display this tab include:

- **Anonymous server**—The Infranet Controller cannot display individual session data about users who sign in through an anonymous server, because it does not collect usernames or other credentials for users signing in through an anonymous server.

- **Local authentication server**—The Infranet Controller displays a Local Users tab instead of a Users tab for local authentication servers, allowing you to add and delete user accounts instead of user sessions.

For all other types of authentication servers, you may view and delete active user sessions using the instructions below.

To view or delete an active user session:

1. In the admin console, choose Authentication > Auth. Servers.
2. Click the appropriate link in the Authentication/Authorization Servers list.
3. Select the Users tab.
4. Perform any of the following tasks:
   - Enter a username in the Show users named field and click Update to search for a specific user.
     
     Or, you can use an * character as a wildcard, where an * represents any number of zero or more characters. For example, if you want to search for all usernames that contain the letters jo, enter *jo* in the Show users named field. The search is case-sensitive. To display the entire list of accounts again, either enter an * character, or delete the field’s contents and click Update.

   - Enter a number in the Show N users field and click Update to control the number of users displayed on the page.

   - Click the checkbox next to individual users and click Delete to terminate their Infranet Controller sessions.

**NOTE:** You can find several access statistics for any user account on the Users tab in the Last Access Statistics columns. These columns appear on any of the Users tabs anywhere they appear in the admin console. The statistics include the last sign-in date and time a user successfully signed in and the browser type and version.
Configuring the Log Monitoring Features

Log Monitoring features on the Infranet Controller enable you to monitor events, user access, and administrator access logs which you can filter and save for later review. Additionally, the Infranet Controller allows you to use SNMP to monitor its activities, and provides statistics and client-side logs for applications such as Host Checker and Odyssey Access Client.

For more information, refer to the following topics:

- “Configuring events, user access, and Admin Access Logs” on page 419
- “Monitoring the Infranet Controller as an SNMP Agent” on page 424
- “Viewing System Statistics” on page 430
- “Enabling Client-Side Logs” on page 430
- “Viewing General Status” on page 431
- “Monitoring Active Users” on page 434

For information about Infranet Controller logs and monitoring capabilities, see “Logging and Monitoring” on page 415.

Configuring events, user access, and Admin Access Logs

Use the System > Log/Monitoring > Events, User Access, and Admin Access pages to save log files, create dynamic log queries, specify which events to save in the log files, and create custom filters and formats.

**NOTE:** The events, user access, and admin access logs are three distinct files. Although the basic configuration instructions for each is the same, modifying the settings for one does not affect settings for another. For more information about the contents of each file, see “Logging and Monitoring” on page 415.

To save, view, or clear the events log file:

1. In the admin console, choose System > Log/Monitoring.
2. Select either the Events, User Access, Admin Access, or Sensors, and then choose Log.
3. From the View by filter list, choose the custom filter that the Infranet Controller should use to filter data.
4. Enter a number in the Show field and click Update if you want to change the number of log entries that the Infranet Controller displays at one time.
5. Click **Save Log As**, navigate to the desired network location, enter a file name, and then click **Save** to manually save the log file.

**NOTE:** To save all log files—**Events Log**, **User Access**, **Admin Access**, and **Sensors**—click **Save All Logs** and the Infranet Controller prompts you for a location where it saves the log files in one compressed file. You can access the **Save All Logs** button from any one of the three log tabs.

6. Click **Clear Log** to clear the local log and **log.old** file.

**NOTE:** When you clear the local log, events recorded by the syslog server are not affected. Subsequent events are recorded in a new local log file.

### Creating, Resetting, or Saving a Dynamic Log Query

To create, reset, or save a dynamic log filter query:

1. Choose **System > Log/Monitoring** in the admin console.

2. Select the **Events**, **User Access**, **Admin Access**, or **Sensors** tab, and then choose **Log**.

3. Click on any data log variable link in the current log. The log immediately redraws based on the chosen variable.

4. Continue adding variables in the same manner (optional). Each data log variable link you select adds an additional variable to the **Edit Query** text field and the log updates with each added variable.

5. Click the **Reset Query** button to clear the **Edit Query** text field and reset the log to the view determined by the filter specified in the **View by filter** field (optional).

6. Click the **Save Query** button to save the dynamic log query as a custom filter (optional). The **Filters** tab displays with the **Query** field pre-populated with the variables you selected from the log. Next:

   a. Enter a name for the filter.

   b. Make the new filter the default filter by selecting **Make default** (optional).

   c. Set the start and end dates for the filter:

      - In the **Start Date** section, click **Earliest Date** to write all logs from the first available date stored in the log file. Or, manually enter a start date.

      - In the **End Date** section, click **Latest Date** to write all logs up to the last available date stored in the log file. Or, manually enter an end date.
7. Choose a format in the **Export Format** section. For more information about the available formats, see “Custom Filter Log Files” on page 417.

8. Select the **Save** button to save the new filter.

**Specifying Which Events to Save in the Log File**

Use options in the **Settings** tab to specify what the Infranet Controller writes to the log file, which syslog servers it uses to store the log files, and the maximum file size.

---

**NOTE:** You may also use the **Archiving** page to automatically save the logs to an FTP accessible location. For more information, see “Archiving Infranet Controller Binary Configuration Files” on page 381.

---

To specify events log settings:

1. In the admin console, choose **System > Log/Monitoring**.

2. Select **Events**, **User Access**, **Admin Access**, or **Sensors** tab, and then choose **Settings**.

3. In the **Maximum Log Size** field, specify the maximum file size for the local log file. (The limit is 500 MB.) The system log displays data up to the amount specified.

**NOTE:** **Maximum Log Size** is an internal setting that most closely corresponds with the size of logs formatted with the **Standard** format. If you choose to use a more verbose format such as **WELF**, your log files may exceed the limit that you specify here.

4. Under **Select Events to Log**, select the checkbox for each type of event that you want to capture in the local log file.

**NOTE:** If you disable the **Statistics** checkbox in the **Events Log** tab, the Infranet Controller does not write statistics to the log file, but continues to display them in the **System > Log/Monitoring > Statistics** tab. For more information, see “Viewing System Statistics” on page 430.

5. Under **Syslog Servers**, enter information about the syslog servers where you want to store your log files (optional):
   
   a. Enter the name or IP address of the syslog server.
   
   b. Enter a facility for the server. The Infranet Controller provides 8 facilities (LOCAL0-LOCAL7) which you can map to facilities on your syslog server.
   
   c. Choose which filter you want to apply to the log file.
   
   d. Click **Add**.
e. Repeat for multiple servers if desired, using different formats and filters for different servers and facilities.

NOTE: Make sure your syslog server accepts messages with the following settings:
facility = LOG_USER and level = LOG_INFO.

6. Click Save Changes.

Creating, Editing, or Deleting Log Filters

Use the controls on the Filters tab to create custom log filters, or to edit or delete the following set of pre-defined log filters:

- **Standard** (default)—This log filter format logs the date, time, node, source IP address, user, realm, and the Infranet Controller event ID and message.

- **WELF**—This customized WebTrends Enhanced Log Format (WELF) filter combines the standard WELF format with information about the Infranet Controller appliance’s realms, roles, and messages.

- **WELF-SRC-2.0-Access Report**—This filter adds access queries to our customized WELF filter. You can easily use this filter with NetIQ’s SRC to generate reports on user access methods.

- **W3C**—The World Wide Web Consortium’s extended log file format is a customizable ASCII format with a variety of different fields. Visit [http://www.w3.org](http://www.w3.org) for more information about this format. Only the User Access log offers this filter as an option.

Creating custom filters and Formats for Your Log Files

Use options in the Filters tab to specify which data is written to your log files as well as its format.

1. In the admin console, choose System > Log/Monitoring.

2. Select the Events, User Access, Admin Access, or Sensors tab, and then choose Filters.

3. Do one of the following:
   - To modify an existing filter, click its name.
   - To create a new filter, click New Filter.

4. Enter a name for the filter.

NOTE: If you select a format and then create a new name for it in the Filter Name field, the Infranet Controller does not create a new custom filter format that is based on the existing format. Instead, it overwrites the existing format with the changes you make.
5. Click **Make Default** to define the selected filter as the default for the log file type. You may set different default filters for the events, user access, and administrator access logs.

6. Use options in the **Query** section to control which subset of data the Infranet Controller writes to the log:
   a. In the **Start Date** section, click **Earliest Date** to write all logs from the first available date stored in the log file. Or, manually enter a start date.
   b. In the **End Date** section, click **Latest Date** to write all logs up to the last available date stored in the log file. Or, manually enter an end date.
   c. In the **Query** section, use the Infranet Controller custom expression language to control which subset of data the Infranet Controller writes to the log. For instructions, see “Writing Custom Expressions” on page 517.

   **NOTE:** Any string (including a * wildcard character) you manually enter in a query, must be enclosed in double-quotes. For example, the query `protocol="UDP" AND sourceip=172.27.0.0/16 AND port="*"` must be presented as `protocol="UDP" AND sourceip=172.27.0.0/16 AND port="*"` or the logging component returns an error.

7. Use one of the options the **Export Format** section to control the format of the data in the log:
   - Select the **Standard**, **WELF**, or **W3C** option to format the log entries using one of these standardized formats. For more information, see “Dynamic Log Filters” on page 417.
   - Select the **Custom** option and enter the format you want to use in the **Format** field. When entering a format, surround variables with percentage symbols (for example `%user%`). All other characters in the field are treated as literals.

8. Click **Save**.
Monitoring the Infranet Controller as an SNMP Agent

You can use a network management tool such as HP OpenView to monitor the Infranet Controller as an SNMP agent. The Infranet Controller supports SNMP (Simple Network Management Protocol) v2, implements a private MIB (management information base), and defines its own traps. To enable your network management station to process these traps, you need to download the Juniper Networks MIB file and specify the appropriate information to receive the traps.

To specify SNMP settings:

1. In the admin console, choose System > Log/Monitoring > SNMP.

2. Click the Juniper Networks MIB file link to access the MIB file, and then save the file from your browser to a network location. For descriptions of the Get and Trap objects in the MIB file, see “Monitoring the Infranet Controller as an SNMP Agent” on page 424.

3. Under Agent Properties enter information in the following fields, and then click Save Changes:

   - Enter information in the System Name, System Location, and System Contact fields that describes the Infranet Controller agent (optional).
   - Enter a string in the Community field (required).

4. Under Trap Thresholds, set the values for the following traps (optional):

   - Check Frequency
   - Log Capacity

NOTE:

- To monitor Infranet Controller vital system statistics, such as CPU utilization, load the UC-Davis MIB file into your SNMP manager application. You can obtain the MIB file from: http://net-snmp.sourceforge.net/docs/mibs/UCD-SNMP-MIB.txt.

- The Infranet Controller supports standard MIB objects, including the system uptime (sysUpTime) object.

- The system uptime (sysUpTime) object returns the time elapsed (in hundredths of a second) since the SNMP agent was started.

NOTE:

- In order to query the Infranet Controller, your network management station must send the Community string to the Infranet Controller.

- To stop the SNMP system, clear the Community field.
- Users
- Memory
- Swap Memory
- Disk
- Meeting Users
- CPU

For information about trap thresholds, see “Monitoring the Infranet Controller as an SNMP Agent” on page 424.

5. Under **Optional traps**, select one or both of the following (optional):

   - **Critical Log Events**
   - **Major Log Events**

   For more information about these event types, see “Logging and Monitoring Overview” on page 415.

6. Under **SNMP Servers**, specify servers to which you want the Infranet Controller to send the traps that it generates by entering information in the following fields, and then clicking **Add**:

   - The server’s host name or IP address
   - The port on which the server listens (typically port 162)
   - The community string required by the network management station (if applicable)

7. Click **Save Changes**.

8. At your network management station:
   a. Download the Juniper Networks MIB file.
   b. Specify the community string required when querying the Infranet Controller (see step 3).
   c. Configure the network management software to receive Infranet Controller traps.

<table>
<thead>
<tr>
<th>Table 36: Configuration Objects</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Object</strong></td>
</tr>
<tr>
<td>logFullPercent</td>
</tr>
<tr>
<td>signedInWebUsers</td>
</tr>
</tbody>
</table>
Table 36: Configuration Objects (Continued)

<table>
<thead>
<tr>
<th>Object</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>signedInMailUsers</td>
<td>Returns the number of users signed in to the Email client.</td>
</tr>
<tr>
<td>blockedIP</td>
<td>Returns the IP address—blocked due to consecutive failed login attempts—sent by the <code>iveTooManyFailedLoginAttempts</code> trap. The system adds the blocked IP address to the <code>blockedIPList</code> table.</td>
</tr>
<tr>
<td>authServerName</td>
<td>Returns the name of an external authentication server sent by the <code>externalAuthServerUnreachable</code> trap.</td>
</tr>
<tr>
<td>productName</td>
<td>Returns the Infranet Controller licensed product name.</td>
</tr>
<tr>
<td>productVersion</td>
<td>Returns the Infranet Controller system software version.</td>
</tr>
<tr>
<td>fileName</td>
<td>Returns the file name sent by the <code>archiveFileTransferFailed</code> trap.</td>
</tr>
<tr>
<td>meetingUserCount</td>
<td>Returns the number of concurrent meeting users sent by the <code>meetingUserLimit</code> trap.</td>
</tr>
<tr>
<td>iveCpuUtil</td>
<td>Returns the percentage of CPU used during the interval between two SNMP polls. This value is calculated by dividing the amount of CPU used by the amount of CPU available during the current and previous SNMP polls. If no previous poll is available, the calculation is based on the interval between the current poll and system boot.</td>
</tr>
<tr>
<td>iveMemoryUtil</td>
<td>Returns the percentage of memory utilized by the Infranet Controller at the time of an SNMP poll. The system calculates this value by dividing the number of used memory pages by the number of available memory pages.</td>
</tr>
<tr>
<td>iveConcurrentUsers</td>
<td>Returns the total number of users logged in for the Infranet Controller node.</td>
</tr>
<tr>
<td>clusterConcurrentUsers</td>
<td>Returns the total number of users logged in for the cluster.</td>
</tr>
<tr>
<td>iveTotalHits</td>
<td>Returns the total number of hits to the Infranet Controller since last reboot. Includes total values from <code>iveFileHits</code>, <code>iveAppletHits</code>, <code>meetingHits</code>, and <code>iveWebHits</code>.</td>
</tr>
<tr>
<td>iveFileHits</td>
<td>Returns the total number of file hits to the Infranet Controller since last reboot Incremented by the web server with each GET/POST corresponding to a file browser request.</td>
</tr>
<tr>
<td>iveWebHits</td>
<td>Returns the total number of hits by means of the Web interface since last reboot. Incremented by the web server for each http request received by the Infranet Controller, excluding file hits, applet hits, and meeting hits.</td>
</tr>
<tr>
<td>iveAppletHits</td>
<td>Returns the total number of applet hits to the Infranet Controller since last reboot Incremented by the web server for each GET request for a Java applet.</td>
</tr>
<tr>
<td>ivetermHits</td>
<td>Returns the total number of terminal hits to the Infranet Controller since last reboot.</td>
</tr>
<tr>
<td>iveSAMHits</td>
<td>Returns the total number of Secure Application Manager hits to the Infranet Controller since last reboot.</td>
</tr>
<tr>
<td>iveNCHits</td>
<td>Returns the total number of Network Connect hits to the Infranet Controller since last reboot.</td>
</tr>
</tbody>
</table>
Table 36: Configuration Objects (Continued)

<table>
<thead>
<tr>
<th>Object</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>meetingHits</td>
<td>Returns the total number of meeting hits to the Infranet Controller since last reboot. Incremented by the web server for each http request received by the Infranet Controller for a meeting-related URL.</td>
</tr>
<tr>
<td>meetingCount</td>
<td>Returns the number of concurrent meetings sent by the meetingLimit trap.</td>
</tr>
<tr>
<td>logName</td>
<td>Returns the name of the log (admin/user/event) for the logNearlyFull and iveLogFull traps.</td>
</tr>
<tr>
<td>iveSwapUtil</td>
<td>Returns the percentage of swap memory pages used by the Infranet Controller at the time of an SNMP poll. The system calculates this value by dividing the number of swap memory pages used, by the number of available swap memory pages.</td>
</tr>
<tr>
<td>diskFullPercent</td>
<td>Returns the percentage of disk space used in the Infranet Controller for the iveDiskNearlyFull trap. The system calculates this value by dividing the number of used disk space blocks by the number of total disk space blocks.</td>
</tr>
<tr>
<td>blockedIPList</td>
<td>Returns a table with the 10 most recently blocked IP addresses. The blockedIP MIB adds blocked IP addresses to this table.</td>
</tr>
<tr>
<td>ipEntry</td>
<td>An entry in the blockedListIP table containing a blocked IP address and its index (see IPEntry).</td>
</tr>
<tr>
<td>IPEntry</td>
<td>The index (ipIndex) and IP address (ipValue) for an entry in the blockedIPList table.</td>
</tr>
<tr>
<td>ipIndex</td>
<td>Returns the index for the blockedIPList table.</td>
</tr>
<tr>
<td>ipValue</td>
<td>A blocked IP address entry in the blockedIPList table.</td>
</tr>
<tr>
<td>logID</td>
<td>Returns the unique ID of the log message sent by the logMessageTrap trap.</td>
</tr>
<tr>
<td>logType</td>
<td>Returns a string sent by the logMessageTrap trap stating whether a log message is major or critical.</td>
</tr>
<tr>
<td>logDescription</td>
<td>Returns a string sent by the logMessageTrap trap stating whether a log message is major or critical.</td>
</tr>
<tr>
<td>ivsName</td>
<td>Returns the name of a virtual system.</td>
</tr>
<tr>
<td>ocspResponderURL</td>
<td>Returns the name of an OCSP responder.</td>
</tr>
<tr>
<td>fanDescription</td>
<td>Returns the status of the Infranet Controller fans.</td>
</tr>
<tr>
<td>psDescription</td>
<td>Returns the status of the Infranet Controller power supplies.</td>
</tr>
<tr>
<td>raidDescription</td>
<td>Returns the status of the Infranet Controller RAID device.</td>
</tr>
</tbody>
</table>

**NOTE:** The options for sending SNMP traps for critical and major events are set to OFF by default, for security purposes.
### Table 37: Status/Error Objects

<table>
<thead>
<tr>
<th>Object</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>iveLogNearlyFull</td>
<td>The log file (system, user access, or administrator access) specified by the <code>logName</code> parameter is nearly full. When this trap is sent, the <code>logFullPercent</code> parameter is also sent. You can configure this trap to be sent at any percentage. To disable the trap, set <code>iveLogNearlyFull</code> to 0%. The trap’s default value is 90%.</td>
</tr>
<tr>
<td>iveLogFull</td>
<td>The log file (system, user access, or administrator access) specified by the <code>logName</code> parameter is completely full.</td>
</tr>
<tr>
<td>iveMaxConcurrentUsersSignedIn</td>
<td>Maximum number or allowed concurrent users are currently signed in. You can configure this trap to be sent at any percentage. To disable the trap, set <code>iveMaxConcurrentUsersSignedIn</code> to 0%. The trap’s default value is 100%.</td>
</tr>
<tr>
<td>iveTooManyFailedLoginAttempts</td>
<td>A user with a specific IP address has too many failed sign-in attempts. Triggered when a user fails to authenticate according to the settings for the Lockout options on the Security Options tab. (See “Configuring Lockout Options” on page 351.) When the system triggers this trap, the system also triggers the <code>blockedIP</code> (source IP of login attempts) parameter.</td>
</tr>
<tr>
<td>externalAuthServerUnreachable</td>
<td>An external authentication server is not responding to authentication requests. When the system sends this trap, it also sends the <code>authServerName</code> parameter.</td>
</tr>
<tr>
<td>iveStart</td>
<td>Infranet Controller has just been turned on.</td>
</tr>
<tr>
<td>iveShutdown</td>
<td>Infranet Controller has just been shut down.</td>
</tr>
<tr>
<td>iveReboot</td>
<td>Infranet Controller has just been rebooted.</td>
</tr>
<tr>
<td>archiveServerUnreachable</td>
<td>Infranet Controller is unable to reach configured FTP or SCP Archive server.</td>
</tr>
<tr>
<td>archiveServerLoginFailed</td>
<td>Infranet Controller is unable to log into configured FTP or SCP Archive server.</td>
</tr>
<tr>
<td>archiveFileTransferFailed</td>
<td>Infranet Controller is unable to successfully transfer archive to configured FTP or SCP Archive server. When the system sends this trap, it also sends the <code>fileName</code> parameter.</td>
</tr>
<tr>
<td>meetingUserLimit</td>
<td>Supplies notification that the user count is over the license limit. When the system sends this trap, it also sends the <code>meetingUserCount</code> parameter.</td>
</tr>
<tr>
<td>iveRestart</td>
<td>Supplies notification that the Infranet Controller has restarted according to the administrator’s instruction.</td>
</tr>
<tr>
<td>meetingLimit</td>
<td>Supplies notification that the concurrent meeting count is over the licensed limit. When the system sends this trap, it also sends the <code>meetingCount</code> parameter. You can configure this trap to be sent at any percentage. To disable the trap, set <code>meetingLimit</code> to 0%. The trap’s default value is 100%.</td>
</tr>
</tbody>
</table>
Table 37: Status/Error Objects (Continued)

<table>
<thead>
<tr>
<th>Object</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>iveDiskNearlyFull</td>
<td>Supplies notification that the Infranet Controller’s disk drive is nearly full. When the system sends this trap, it also sends the diskFullPercent parameter. You can configure this trap to be sent at any percentage. To disable the trap, set iveDiskNearlyFull to 0%. This trap’s default value is 80%.</td>
</tr>
<tr>
<td>iveDiskFull</td>
<td>Supplies notification that the Infranet Controller’s disk drive is full.</td>
</tr>
<tr>
<td>logMessageTrap</td>
<td>The trap generated from a log message. When the system sends this trap, it also sends the logID, logType, and logDescription parameters.</td>
</tr>
<tr>
<td>memUtilNotify</td>
<td>Supplies notification that the system has met the configured threshold for memory utilization. To disable the trap, set memUtilNotify to 0. The threshold is 0%, by default.</td>
</tr>
<tr>
<td>cpuUtilNotify</td>
<td>Supplies notification that the system has met the configured threshold for CPU utilization. To disable the trap, set cpuUtilNotify to 0. The threshold is 0%, by default.</td>
</tr>
<tr>
<td>swapUtilNotify</td>
<td>Supplies notification that the system has met the configured threshold for swap file memory utilization. To disable the trap, set swapUtilNotify to 0. The threshold is 0%, by default.</td>
</tr>
<tr>
<td>iveMaxConcurrentUsersVirtualSystem</td>
<td>Supplies notification that the maximum number of concurrent virtual system users have signed in to the IVS.</td>
</tr>
<tr>
<td>ocspResponderUnreachable</td>
<td>Supplies notification that the OCSP Responder is not responding.</td>
</tr>
<tr>
<td>iveFanNotify</td>
<td>Supplied notification that the status of the fans has changed.</td>
</tr>
<tr>
<td>ivePowerSupplyNotify</td>
<td>Supplies notification that the status of the power supplies has changed.</td>
</tr>
<tr>
<td>iveRaidNotify</td>
<td>Supplies notification that the status of the RAID device has changed.</td>
</tr>
<tr>
<td>iveNetExternalInterfaceDownTrap</td>
<td>Supplies the type of event that brought down the external interface. The nicEvent parameter can contain values of “external” for an external event and “admin” for an administrative action.</td>
</tr>
<tr>
<td>iveNetInternalInterfaceDownTrap</td>
<td>Supplies the type of event that brought down the internal interface. The nicEvent parameter can contain values of “external” for an external event and “admin” for an administrative action.</td>
</tr>
<tr>
<td>iveClusterDisableNodeTrap</td>
<td>Supplies the name of the cluster that contains disabled nodes, as well as a string containing the names of all disabled nodes. Node names are separated by white space in the string.</td>
</tr>
<tr>
<td>iveClusterChangedVIPTrap(vipType, currentVIP, newVIP)</td>
<td>Supplies the status of a virtual IP for the cluster. The vipType indicates whether the changed VIP was external or internal. The currentVIP contains the VIP prior to the change, and newVIP contains the VIP after the change.</td>
</tr>
</tbody>
</table>
Viewing System Statistics

Every hour, the Infranet Controller logs the peak load of Web users.

The Statistics page displays that information for the past seven days. The Infranet Controller writes that information to the system log once a week. Note that upgrading the Infranet Controller clears all statistics. If you configure the system to log statistics hourly, however, old statistics are still available in the log file after an upgrade.

To view system statistics:

1. In the admin console, choose System > Log/Monitoring > Statistics.
2. Scroll the page to view all four categories of data.

Enabling Client-Side Logs

You can enable client-side logging for the Host Checker and Odyssey Access Client. When you enable this option, the Infranet Controller writes a client-side log to endpoints that use the feature. The Infranet Controller appends to the log file each time the feature is invoked during subsequent user sessions. This feature is useful when working with the support team to debug problems with the respective feature.

NOTE: Because the client-side logging settings are global, the Infranet Controller writes a log file to all clients that use Host Checker or Odyssey Access Client. Also, the Infranet Controller does not remove client-side logs. Users need to manually delete log files from their clients. For information about where the Infranet Controller installs log files, see the Client-side Changes Guide on the Juniper Networks Support Site.

To specify global client-side logging settings:

1. In the admin console, choose System > Log/Monitoring > Client Logs > Settings.
2. Select the desired features for which the Infranet Controller writes client-side logs.

3. Click **Save Changes** to save these settings globally.

### Viewing General Status

When you sign in to the admin console, the Infranet Controller displays the System > Status page, showing the Overview tab. This tab summarizes details about the Infranet Controller server and system users. When you make changes on other admin console pages, the Infranet Controller updates corresponding information on the Overview tab.

![NOTE: This tab is the home page for all administrators, including delegated administrators without read or write access to the System > Status tabs.]

### Viewing System Capacity Utilization

The Central Management dashboard provides system capacity utilization graphs that allow you to easily view and understand how much of your system capacity you are using on a regular basis.

To use this information for data reporting elsewhere, export it as an XML file using options on the Maintenance > Import/Export > Configuration page.

These graphs are displayed in the System > Status > Overview tab when you open the admin console, and allow you to easily view:

- **Concurrent Users**—This graph shows the number of users signed into the Infranet Controller. In clustered environments, the graph includes two lines. The first line displays the number of local users signed into the node selected from the drop-down list and the second line displays the number concurrent users signed into the entire cluster.

- **Hits Per Second**—This graph shows the number of hits currently being processed by the Infranet Controller. In a clustered environment, you may choose an Infranet Controller from the drop-down list to determine which node’s data is displayed in the graph. The graph includes four lines: number of hits, number of Web hits, number of file hits, and number of client/server hits.

- **CPU and Virtual (Swap) Memory Utilization**—This graph shows the percentage of the CPU and available memory currently being used. In a clustered environment, you may choose an Infranet Controller from the drop-down list to determine which node’s data is displayed in the graph.

- **Throughput**—This graph shows the amount of data (in KB) currently being processed. In a clustered environment, you may choose an Infranet Controller from the drop-down list to determine which node’s data is displayed in the graph. The graph includes four lines: external in, external out, internal in, and internal out.
You may also use the **Page Settings** window to configure which graphs the Infranet Controller displays in the dashboard and the period of time that the Infranet Controller tracks.

To download the graph data to an XML file:

1. In the admin console, choose **System > Status > Overview**.
2. Click the **Download** link that corresponds to the graph that you want to download.
3. Click **Save**, specify the directory where you want to save the XML file, and click **Save**.

**Specifying Time Range and Data to Display in Graphs**

You can also specify the time range and other data to display in the graphs.

To specify the time range and data displayed in the graphs:

1. In the admin console, choose **System > Status > Overview**.
2. Click **Page Settings**.
3. Select which utilization graphs to display.
4. Select the range of time that you want to plot in the graphs. Graphing intervals range from 1 hour to 1 year.
5. Indicate how often you want to refresh the graphs.
6. Click **Save Changes**.

**Configuring Graph Appearance**

You can also specify colors and line weights, to change the appearance of the graphs on the Status page.

To specify the colors and line weights displayed in the graphs:

1. In the admin console, choose **System > Status > Overview**.
2. Click the **Edit** link that corresponds to the graph that you want to modify.
3. Use settings in the **Graph Settings** dialog box to edit the background color, graph line colors, text color, line color, and line width displayed in the graph.
4. Click **Save Changes**.

The dashboard for the Infranet Controller allows you to easily view the last 10 critical system events. Using the **Event Monitor** window, you can quickly access and address any critical system problems. Once you have opened the **Event Monitor** window, you may keep it open and continually monitor system events while navigating through the admin console to perform standard maintenance and configuration tasks.
To quickly review critical system events:

1. In the admin console, choose **System > Status > Overview**.

2. Click **Critical Events**. The Event Monitor window displays the severity and message of any critical events recorded in the system’s log file.

3. Click **Refresh** to view the most up-to-date events (optional).

4. Click **See All** to navigate to the **System > Log/Monitoring > Events > Log** tab, where all events—ranging from informational to critical—are displayed (optional). For more information, see “Configuring the Log Monitoring Features” on page 419.

**Downloading the Current Service Package**

You can download the service package currently installed on the Infranet Controller for backup and to install it onto another Infranet Controller.

To download your current service package:

1. In the admin console, choose **System > Status > Overview**.

2. Click **Download Package**.

3. Click **Save**.

4. Specify a name and location for the service package.

5. Click **Save**.

**Editing the System Date and Time**

You need to set the server time in order to accurately record system events and user file transfers. You may use a Network Time Protocol (NTP) server to sync the Infranet Controller with a series of computers, or you may set the Infranet Controller time manually.

To edit the system date and time:

1. In the admin console, choose **System > Status > Overview**.

2. In the **System Date & Time** section, click **Edit**.

3. Select a time zone from the **Time Zone** menu. The Infranet Controller automatically adjusts the time for Daylight Saving Time.

4. Set the system time using one of these methods:

   - **Use NTP server**—Select the **Use NTP Server** option, enter the server’s IP address or name, and specify an update interval.

   - **Set Time Manually**—Select the **Set Time Manually** option and enter values for the date and time. You can also click **Get from Browser** to populate the **Date** and **Time** fields.
5. Click **Save Changes**.

### Monitoring Active Users

You can monitor users signed in to the Infranet Controller. Each user’s name, authentication realm, role, and sign-in time are listed on the **Active Users** page.

---

**NOTE:**

- If a user signs into the Infranet Controller and then the user’s computer is placed on a VLAN without an IP address, the Infranet Controller does not display an IP address under **Signed in IP** for the user’s status on the **Active Users** page.

- If there is a NAT device between the user’s computer and the Infranet Enforcer, the Infranet Controller displays both the NAT device’s IP address and the endpoint’s virtual source IP address under **Signed in IP** for the user’s status on the **Active Users** page. For example, if the NAT device’s IP address is 10.64.9.26, and the endpoint’s virtual source IP address is 192.168.80.128, the following information appears under **Signed in IP**:

  10.64.9.26 (192.168.80.128 behind NAT)

---

To monitor users signed in to the Infranet Controller:

1. In the admin console, choose **System > Status > Active Users**.

2. Perform these tasks (optional):

   - **Sign users out of their Infranet Controller sessions**:
     - To forcibly sign out one or more end-users or administrators, select the checkbox next to the appropriate names and then click **Delete Session**.
     - To forcibly sign out all end-users who are currently signed-in, click **Delete All Sessions**.
     - To forcibly sign out all end-users who are currently signed-in and also prevent any other users from signing in, click **Disable All Users**. To allow users to sign in again after you disable all users, click **Enable All Users**.

---

**NOTE:** If you want to sign out administrators, you must choose them individually and use the **Delete Session** button.
Perform a dynamic policy evaluation of all signed-in users:

- To manually evaluate all authentication policies, role mapping rules, role restrictions, user roles, and resource policies for all currently signed-in users, click Refresh Roles. Use this button if you make changes to an authentication policy, role mapping rules, role restrictions, or resource policies and you want to immediately refresh the roles of all users. For more information, see “Dynamic Policy Evaluation” on page 105.

Configure which data is shown and its order:

- To display a specific user, enter his username in the Show Users Named field and click Update. If you do not know the user’s exact username, use the * wildcard character. For example, if you have a user named “Joseph Jones,” but you do not remember if his username is “Joe” or “Joseph,” enter Jo* in the Show Users Named field. The Infranet Controller returns a list of all users whose usernames start with the letters jo.

- To control how many users and administrators are displayed in the Active Users page, enter a number in the Show N users field and click Update.

- To sort the table of currently signed-in users and administrators, click a column header.

- To refresh the page’s content, click Update.

Link to related tabs:

- To edit a user’s authentication realm, click the Realm link next to his name and follow the instructions in “Creating an Authentication Realm” on page 246.

- To edit a user’s role, click the Role link next to his name and follow the instructions in “Creating Administrator Roles” on page 474 (if he is an administrator) or “Configuring User Roles” on page 120 (if he is an end-user).
Chapter 14
Troubleshooting

The Infranet Controller provides several troubleshooting utilities that enable you to monitor the state of your system, including clusters, if you use them. This section provides an overview of the various troubleshooting tasks that are available by using the Infranet Controller:

- “Tracking Events” on page 437
- “Creating Snapshots of the Infranet Controller System State” on page 439
- “Creating TCP Dump Files” on page 440
- “Testing Infranet Controller Network Connectivity” on page 442
- “Running Debugging Tools Remotely” on page 443
- “Creating Debugging Logs” on page 444
- “Using the RADIUS Diagnostic Log” on page 444
- “Monitoring Cluster Nodes” on page 445
- “Configuring Group Communication Monitoring on a Cluster” on page 446
- “Configuring Network Connectivity Monitoring on a Cluster” on page 448

Tracking Events

You can determine why your Infranet Controller does not allow you to accomplish a task that you desire by tracking problematic Infranet Controller events using settings in the Maintenance > Troubleshooting > User Sessions > Policy Tracing page of the admin console. This page guides you through all the realms, roles, and policies that are currently configured in the Infranet Controller and print log messages at various steps of the authentication, authorization, and access process.

The events in question are related to authentication, authorization, and access for a particular user. They are entirely driven by what happens during a user session.
Tracking Events Using Policy Tracing

The Infranet Controller allows you to troubleshoot problems by tracking events when a user signs into a realm. The Maintenance > Troubleshooting > User Sessions > Policy Tracing page allows you to record a policy trace file for an individual user. The Infranet Controller displays log entries that list the user’s actions and indicates why he is allowed or denied access to various functions.

To create a policy trace file:

1. In the admin console, choose Maintenance > Troubleshooting > User Sessions > Policy Tracing.

2. In the User field, enter the Infranet Controller username of the user you want to trace. Note that you may use a wildcard character (*) in place of a username. For example, if your users are signing into an anonymous server, you may want to use the wildcard character (*) since you cannot know the internal username that the Infranet Controller will assign to the user.

3. In the Realm field, select the user’s realm. Note that the Infranet Controller does not allow you to select a realm that maps to an anonymous authentication server.

4. Under Events to log, select the types of events you want to write to the policy tracing log file.

5. Click Start Recording. Ask the user to sign into the Infranet Controller after you have started recording.

6. Click View Log to see the log entries.

7. Click Stop Recording when you have obtained enough information.

8. Review messages in the log file to determine what is causing the unexpected behavior. If you cannot determine and fix the problem, click Save Log As to save a copy of the log file to a location on your network. Then, send the file to Juniper Networks Support for review.
Creating Snapshots of the Infranet Controller System State

The **Maintenance > Troubleshooting > System Snapshot** tab allows you to create a snapshot of the Infranet Controller system state. When you use this option, the Infranet Controller runs various utilities to gather details on the Infranet Controller system state, such as the amount of memory in use, paging performance, the number of processes running, system uptime, the number of open file descriptors, and the ports in use.

You can choose to include or exclude system configuration and debug logs. However, debug logs are particularly important in the event of a problem. You will need to set the debug log at a certain level and add the events list as directed by your Support representative. Recreate the problem or event and then take a snapshot and send it to Support. The debug log is encrypted; you cannot view it.

**NOTE:**

- The Infranet Controller stores up to ten snapshots, which are packaged into an encrypted “dump” file that you can download to a network machine and then email to Juniper Networks Support. If you take more than ten snapshots, the Infranet Controller overwrites the oldest snapshot file with the new snapshot. If the Infranet Controller runs out of disk space, the Infranet Controller does not store the newest snapshot and logs a message in the Event log. We recommend that you download the snapshots to a network machine in a timely manner to avoid losing them.

- In a cluster, the snapshot occurs on an individual node basis only. That is, the snapshot settings you specify are not synchronized in all nodes of the cluster.

To take a snapshot of the Infranet Controller system state:

1. In the admin console, choose **Maintenance > Troubleshooting > System Snapshot**.

2. Select the **Include system config** checkbox to include system configuration information in your snapshot (optional).

3. Select the **Include debug log** checkbox to include log file created through the **Debug Log** tab in your system snapshot. For more information, see “Creating Debugging Logs” on page 444.

4. Click **Take Snapshot** to manually take a snapshot immediately.

5. To automatically take a snapshot at regular intervals:

   a. Select **Schedule automatic snapshots**. Additional configuration items appear.
b. Specify how often you want to take a snapshot (in hours).

c. Specify the maximum file size of each snapshot (in MB).

NOTE: If the size of the snapshot exceeds the maximum file size you specify, the snapshot will fail and the Infranet Controller will log a message in the Event log.

d. (Optional) If you want to stop taking snapshots at a particular time, specify a date and time. Otherwise, the periodic snapshots continue until you manually stop them.

e. If you want to disable debug logs at the stop time you specified, select Disable debug logs at stop time.

6. Click Save Changes.

7. When the Infranet Controller finishes taking the snapshot, click the link for the snapshot listed under Snapshot, click Save, navigate to the folder where you want to store the snapshot file, and then click Save.

8. Email the file to Juniper Networks Support for review.

9. When you are finished, select the snapshot listed under Snapshot and then click Delete to delete the snapshot.

NOTE: You can also take a system snapshot from the serial console. This method is useful if you cannot get to the admin console and need to save the system configuration. For more information, see “Performing Common Recovery Tasks” on page 501.

Creating TCP Dump Files

The Maintenance > Troubleshooting > Tools > TCP Dump tab allows you to sniff network packet headers and save the results in an encrypted “dump” file that you can download to a network machine and then email to Juniper Networks Support.

This feature uses the TCP/IP network stack to capture packets at the TCP layer. It captures all communication that passes through the Infranet Controller. However, certain encrypted higher level protocols cannot be decrypted. This feature is useful for troubleshooting common customer problems. A TCP dump file helps the Juniper Networks Support team observe the communication protocols used between Infranet Controller and any other intranet server and how the intranet server responded to requests from the Infranet Controller.

On the admin console, you can select which interface you want to capture packets from, whether internal or external, you can select promiscuous mode, which increases the level of detail in the dump file, and you can specify a filter.
To sniff network packet headers:

1. In the admin console, choose **Maintenance > Troubleshooting > Tools > TCP Dump**.
2. Select the Infranet Controller port on which you want to sniff network packet headers.
3. Turn off **Promiscuous mode** to sniff only for packets intended for the Infranet Controller.
4. Create a custom filter using TCPDump Filter Expressions (optional). This option provides the ability to filter the sniffed network packets so that the resulting dump file contains only the information you require. See Table 38 below for examples.
5. Click **Start Sniffing**.
6. Click **Stop Sniffing** to stop the sniffing process and create an encrypted file.
7. Click **Download** to download the file to a network machine.
8. Email the file to Juniper Networks Support for review.

### Table 38: Examples of TCPDump Filter Expressions

<table>
<thead>
<tr>
<th>Example</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>tcp port 80</td>
<td>Sniffs packets on TCP port 80.</td>
</tr>
<tr>
<td>port 80</td>
<td>Sniffs packets on TCP or UDP port 80.</td>
</tr>
<tr>
<td>ip</td>
<td>Sniffs the IP protocol.</td>
</tr>
<tr>
<td>tcp</td>
<td>Sniffs the TCP protocol.</td>
</tr>
<tr>
<td>dst #.#.#.#</td>
<td>Sniffs the destination IP address specified, where #.#.#.# is a valid IP address.</td>
</tr>
<tr>
<td>src #.#.#.#</td>
<td>Sniffs the source IP address specified, where #.#.#.# is a valid IP address.</td>
</tr>
<tr>
<td>port 80 or port 443</td>
<td>Sniffs on port 80 or port 443.</td>
</tr>
<tr>
<td>src #.#.#.# and dst #.#.#.#</td>
<td>Sniffs the source and destination IP addresses or hosts specified, where each #.#.#.# represents a valid IP address.</td>
</tr>
<tr>
<td>tcp port 80 or port 443 and src #.#.#.#</td>
<td>This example shows how to specify multiple parameters to create a filter that sniffs on TCP port 80, or on TCP or UDP port 443, and on the destination and source ports, where each #.#.#.# represents a valid IP address.</td>
</tr>
</tbody>
</table>

For more information about TCPDump Filter Expressions, visit the following Web site: [http://www.tcpdump.org/tcpdump_man.html](http://www.tcpdump.org/tcpdump_man.html)

For more information on using TCP Dump, see “Creating TCP Dump Files” on page 440.
The Maintenance > Troubleshooting > Tools > Commands tab allows you to run UNIX commands such as arp, ping, traceroute, and NSlookup to test Infranet Controller network connectivity. You can use these connectivity tools to see the network path from the Infranet Controller to a specified server.

**Address Resolution Protocol (ARP)**

Use the arp command to map IP network addresses to the hardware addresses. The Address Resolution Protocol (ARP) allows you to resolve hardware addresses.

To resolve the address of a server in your network, a client process on the Infranet Controller sends information about its unique identify to a server process executed on a server in the intranet. The server process then returns the required address to the client process.

**Ping**

Use the ping command to verify that the Infranet Controller can connect to other systems on the network. In the event of a network failure between the local and remote nodes, you will not receive a reply from a pinged device. In that case, contact your LAN administrator for help.

The ping command sends packets to a server and returns the server response, typically a set of statistics including the target server’s IP address, the time spent sending packets and receiving the response, and other data. You can ping unicast or multicast addresses, and you must include the target server name in the request.

**Traceroute**

Use the traceroute command to discover the path that a packet takes from the Infranet Controller to another host. Traceroute sends a packet to a destination server and receives an ICMP TIME_EXCEEDED response from each gateway along its path. The TIME_EXCEEDED responses and other data are recorded and displayed in the output, showing the path of the packet round-trip.

To run a UNIX command to test Infranet Controller network connectivity:

1. In the admin console, choose Maintenance > Troubleshooting > Tools > Commands.
2. From the Command list, select the command to run.
3. In the Target Server field, enter the IP address of the target server.
4. Enter other arguments or options.
5. Click OK to run the command.
NSlookup

Use NSlookup to get detailed information about a name server on the network. You can query on several different types of information, including a server's IP address, alias IP address, start-of-authority record, mail exchange record, user information, well-known services information, and other types of information.

To run NSlookup to test name server connectivity:

1. In the admin console, choose Maintenance > Troubleshooting > Tools > Commands.
2. From the Command list, select NSLookup.
3. Select the Query Type from the drop down menu.
4. Enter the query, which is a host name, an IP address, or other information, depending on your selection of query type.
5. Enter the DNS server name or IP address.
6. Enter other options.
7. Click OK to run the command.

Running Debugging Tools Remotely

The Juniper Networks Support team can run debugging tools on your production Infranet Controller if you configure it to do so through the Maintenance > Troubleshooting > Remote Debugging page. To enable this option, you must work with Juniper Networks Support to obtain a debugging code and host to which your Infranet Controller connects.

To enable remote debugging:

1. Contact Juniper Networks Support to set up the terms of a remote debugging session.
2. In the admin console, choose Maintenance > Troubleshooting > Remote Debugging.
3. Enter the debugging code provided by Juniper Networks Support.
4. Enter the host name provided by Juniper Networks Support.
5. Click Enable Debugging to allow the Juniper Networks Support team to access the Infranet Controller.
6. Notify Juniper Networks Support that your Infranet Controller is accessible.
7. Click Disable Debugging when Juniper Networks Support notifies you that the remote debugging session is over.
Creating Debugging Logs

If you have a problem, a Juniper Networks Support representative may ask you to create debugging logs to assist with debugging Infranet Controller internal issues. When you enable logging, the Infranet Controller records certain events and messages based on event codes you enter into admin console on the Maintenance > Troubleshooting > Monitoring > Debug Log tab. Using the debug log that results, the support team can identify the code flow for any discrepancies. Your support representative gives you all of the information you need to create the log file, including the debug detail log level and the event codes.

**NOTE:** Running debug logging can impact your system performance and stability. You should only generate debug logs when directed by your Juniper Networks Support representative.

To enable the debug log:

1. In the admin console, choose Maintenance > Troubleshooting > Monitoring > Debug Log.

2. Select the Debug Logging On checkbox.

3. Enter the log size, detail level, and event code specified by Juniper Networks Support.

**NOTE:** Setting the detail level to 0 displays only Critical messages, it does not disable logging completely.

4. Click Save Changes.

5. Choose the Maintenance > Troubleshooting > System Snapshot tab.

6. Check the Include debug log checkbox.

7. Click Take snapshot to create a file that contains the debug log.

8. Click Download.

9. Attach the snapshot file an email message and sent it to Juniper Networks Support.

Using the RADIUS Diagnostic Log

The RADIUS Troubleshooting Log allows you to view the full suite of RADIUS logging features, including traffic trace and debug-level messages. In releases prior to 2.2 these logs were only available in an encrypted format that required decryption by Juniper Networks.
The RADIUS Troubleshooting Log monitors all requests that the Infranet Controller receives from RADIUS clients. RADIUS requests that the Infranet Controller initiates do not appear in the log.

Raw traffic is not available in the log. To view raw traffic, use the tcpdump feature. See “Creating TCP Dump Files” on page 440.

You can configure the maximum size of the log. When the log fills up, logging stops. You can clear the log to restart logging.

All events that appear in the log have an ID code, and all messages in a thread are tagged with the same ID. This allows you to track individual logins or login attempts.

The RADIUS Troubleshooting Log is secure, as passwords are suppressed and do not appear in the logs.

Performance of the Infranet Controller is affected with RADIUS logging turned on.

Source IP addresses are represented in the RADIUS Troubleshooting Log as 127.0.0.1 (the loopback address). To determine the real IP address, refer to the Funk-Source-IP-Address for Layer 3 connections. For Layer 2 connections, the calling station ID is the MAC address of the endpoint.

Please refer to http://www.iana.org/assignments/radius-types and http://www.iana.org/assignments/eap-numbers for information to help you understand RADIUS logs. When referring to external information, note that EAP-JUAC is a Juniper Networks protocol. External references do not mention EAP-JUAC.

**NOTE:** In an active-active cluster, logs are not synced.

To configure the RADIUS Troubleshooting log:

1. Select **Troubleshooting > Monitoring > RADIUS** from the left navigation bar of the admin console.
2. Select the **RADIUS Diagnostic Logging On** check box.
3. Enter the maximum log size (up to 1,000 MB) in the **Max Diagnostic Log Size** box.
4. Click **Save Changes**.

### Monitoring Cluster Nodes

If you have a problem with a cluster, a Juniper Networks Support representative may ask you to create a snapshot that includes node monitoring statistics to assist with debugging the cluster problem. When you enable the node monitor on the **Maintenance > Troubleshooting > Monitoring > Node Monitor** tab, the Infranet Controller captures certain statistics specific to the cluster nodes on your system. Using the snapshot that results, the support team can identify important data, such as network statistics and CPU usage statistics.
To enable node monitoring:

1. Enter the maximum size for the node monitor log.
2. Enter the interval, in seconds, at which node statistics are to be captured.
3. Select the **Node monitoring enabled** checkbox to start monitoring cluster nodes.
4. For **Maximum node monitor log size**, enter the maximum size (in MB) of the log file. Valid values are 1-30.
5. Specify the interval (in seconds) that defines how often nodes are to be monitored.
6. Select the commands to use to monitor the node.
   
   If you select **dsstatdump**, enter its parameters as well.
7. Click **Save Changes**.
8. If you want to include the node monitoring results in the system snapshot, choose **Maintenance > Troubleshooting > System Snapshot**, and select the **Include debug log** checkbox.
9. Take a system snapshot to retrieve the results. For more information, see “Creating Snapshots of the Infranet Controller System State” on page 439.

---

### Configuring Group Communication Monitoring on a Cluster

If you have a problem with a cluster, a Juniper Networks Support representative may ask you to create a snapshot that includes group communication statistics to assist with debugging the cluster problem. When you enable the group communication monitor on the **Maintenance > Troubleshooting > Monitoring > Cluster > Group Communication** tab, the Infranet Controller records statistics related to all of the cluster nodes on your system. As the local node communicates with other nodes in the cluster, the Infranet Controller captures statistics related to intra-cluster communication. The **Maintenance > Troubleshooting > Monitoring > Cluster > Group Communication** tab appears only when you enable clustering on your system. On a standalone Infranet Controller, you do not have access to the **Maintenance > Troubleshooting > Monitoring > Cluster > Group Communication** tab.
To enable group communication monitoring:

1. Enter the maximum size for the statistics log.
2. Enter the interval, in seconds, at which events are to be logged.
3. If you want to monitor all cluster nodes from the current local node, select the **Monitor all cluster nodes from this node** checkbox. If you do not check this option, the group communication monitor gathers statistics only for the local node.
4. Select the **Enable group communication monitoring** checkbox to start the monitoring tool.
5. Click **Save Changes**.
6. If you want to include the node monitoring results in the system snapshot, choose **Maintenance > Troubleshooting > System Snapshot**, and select the **Include debug log** checkbox.
7. Take a system snapshot to retrieve the results. For more information, see “Creating Snapshots of the Infranet Controller System State” on page 439.
Configuring Network Connectivity Monitoring on a Cluster

If you have a problem with a cluster, a Juniper Networks Support representative may ask you to enable the cluster node troubleshooting server. When you enable the server on the **Maintenance > Troubleshooting > Cluster > Network Connectivity** tab, the Infranet Controller attempts to establish connectivity between the node on which the server resides and another node you specify. As the nodes communicate, the Infranet Controller displays network connectivity statistics on the page. The **Maintenance > Troubleshooting > Cluster > Network Connectivity** tab appears only when you enable clustering on your system. On a standalone Infranet Controller, you do not have access to the **Maintenance > Troubleshooting > Cluster > Network Connectivity** tab.

Use the Network Connectivity page to enable the cluster node troubleshooting server and to select a node on which to perform troubleshooting tasks. The troubleshooting tool allows you to determine the network connectivity between cluster nodes.

The server component of this tool runs on the node to which connectivity is being tested. The client component runs on the node from which connectivity is being tested. The basic scenario for testing connectivity is this:

- The administrator starts the server component on the passive node.
- The administrator then tests the connectivity to the server node from the Active node, by starting the client component on the Active node and contacting the Passive node running the server component.

**NOTE:** The server component must be run on nodes that are configured as either standalone, or in a cluster but disabled. Cluster services cannot be running on the same node as the server component.

1. Select the **Enable cluster network troubleshooting server** checkbox to enable the server component.
2. Click **Save Changes**.
3. On another machine, select **Maintenance > Troubleshooting > Cluster > Network Connectivity**.
4. Perform one of the following steps:
   - Select a node from the drop down menu.
   - Enter the IP address of the server node.
5. Click **Go** to begin troubleshooting the machine on which the server component is running.
6. Click the **Details** link that appears on the page below the fields, to view the results.
You can purchase a clustering license to deploy two or more Infranet Controller appliances as a cluster. These appliances support active/passive or active/active configurations across a LAN to provide high availability, increased scalability, and load balancing capabilities.

The Infranet Controller also supports clustering of Infranet Enforcers in either active/passive or active/active configurations (NSRP clustering). For active/active support, dynamic IPsec enforcement must be configured with VSD interfaces for IPsec. ScreenOS version 6.1 or later is required. See “Concepts & Examples ScreenOS Reference Guide: Vol 11, High Availability.”

You define a cluster on one Infranet Controller by specifying three pieces of data:

- A name for the cluster
- A password for the cluster members to share
- A name to identify the machine in the cluster

Entering this information enables you to initiate the first member of your cluster. You then need to specify which Infranet Controllers you want to add to the cluster. After an Infranet Controller is identified as an intended member, you may add it to the cluster through its:

- **Admin console**—If a configured Infranet Controller is running as a stand alone machine, you can add it to a cluster through its admin console.

- **Serial console**—If an Infranet Controller is in its factory-state, you can add it to a cluster through its serial console by entering minimal information during initial setup.

When an Infranet Controller joins a cluster, it initializes its state from the existing member that you specify. The new member sends a message to the existing member requesting synchronization. The existing member sends the system state to the new member, overwriting all system data on that machine. After that point, the cluster members synchronize data when there is a state change on any member. Cluster member communication is encrypted to prevent attacks from inside the corporate firewall. Each Infranet Controller uses the shared password to decrypt communication from another cluster member. For security reasons, the cluster password is not synchronized across Infranet Controllers.
During synchronization, the new node receives the current service package, which upgrades the node if it is running an older service package.

This chapter contains the following sections:

- “Task Summary: Deploying a Cluster” on page 450
- “Creating and Configuring a Cluster” on page 451
- “Configuring Cluster Properties” on page 454
- “Managing and Configuring Clusters” on page 464

**NOTE:** All Infranet Controllers in a cluster must feature the same cluster license. You cannot add an ADD and a CL license to the same machine at the same time. For a node to be able to join a cluster, it must either have no licenses or only CL licenses.

---

**Task Summary: Deploying a Cluster**

To create an Infranet Controller cluster:

1. Ensure that all intended Infranet Controller nodes use the same hardware platform (for example, all are Infranet Controller 6500 machines).

2. Ensure that all intended Infranet Controller nodes have been initially configured (for example, Infranet Controller host name is specified and the internal and external IP addresses are assigned), and they are running the same service package version.

3. From the admin console, click the **System > Configuration > Licensing** page, enable the clustering feature on the primary server by entering a stand alone license and any feature licenses.

4. From the **System > Clustering > Create Cluster** page, initialize the Infranet Controller cluster by defining the cluster name and adding the first/primary Infranet Controller to the cluster. For instructions, see “Defining and Initializing a Cluster” on page 451.

5. From the **System > Clustering > Status** page, add the names and IP addresses of future cluster Infranet Controllers to the primary Infranet Controller. For instructions, see “Joining an Existing Cluster” on page 452.

6. From the **System > Clustering > Join Cluster** page, populate the cluster with additional Infranet Controllers as necessary. For instructions, see “Adding an Infranet Controller to a Cluster Through Its Admin Console” on page 453 or, in the case of a pre-configured/factory-set Infranet Controller, “Joining an Infranet Controller to a Cluster Through Its Serial Console” on page 470.
Creating and Configuring a Cluster

If an Infranet Controller is not part of a cluster, the Clustering page displays the Create tab. The Create tab allows you create the configurations for cluster nodes, even if you have no physical devices available to join the cluster.

After you add a clustering license to the Infranet Controller, the Clustering page displays the Join tab. The Join tab enables you to join an initialized Infranet Controller to an existing cluster, as explained in “Joining an Existing Cluster” on page 452.

After creating the cluster, the Clustering page shows Status and Properties tabs, which replace the original Join and Create tabs. Use the Status tab to specify an Infranet Controller to add to a cluster, manage network settings for cluster nodes, and upgrade the cluster service package. The Properties tab allows you to specify active/passive, active/active, and other cluster settings, and to delete a cluster.

NOTE:

- The Create tab only appears on an Infranet Controller that does not possess the cluster license key. You cannot create a cluster unless you have entered a cluster license key.
- The clustering license key or CL license is necessary for the secondary node’s ability to join the cluster.
- All nodes in a cluster must feature the same license key as on the primary cluster Infranet Controller to enable cluster operation. You cannot add an ADD and a CL license to the same machine at the same time. For a node to be able to join a cluster, you must add a CL license to the node.

This section contains the following information about clustering:

- “Defining and Initializing a Cluster” on page 451
- “Joining an Existing Cluster” on page 452

Defining and Initializing a Cluster

If you are currently running stand alone Infranet Controllers that you want to cluster, we recommend that before you create a cluster, you first configure system and user settings on one machine. After doing so, use the same machine to create the cluster. This machine joins the cluster as part of the creation process. When other Infranet Controllers join the cluster, this machine propagates its configuration to the new cluster member.
To define and initialize a cluster:

1. Configure one Infranet Controller with the appropriate license and system, user, resource, and application data, as explained in “Task Summary: Deploying a Cluster” on page 450.

2. From the admin console select **System > Clustering > Create** and enter a name for the cluster, a cluster password, and a name for this machine, such as Server-1.

3. Click **Create Cluster** When prompted to confirm the cluster creation, click **Create**. After the Infranet Controller initializes the cluster, the Clustering page displays the Status and Properties tabs. Use the Status tab to specify additional cluster members before trying to add another Infranet Controller to the new cluster. For more information, see “Specifying an Infranet Controller to Join to a Cluster” on page 452.

### Joining an Existing Cluster

The method you use to add an Infranet Controller to a cluster depends on whether or not the Infranet Controller is configured or uninitialized (still in its factory state). For an Infranet Controller in its factory state, we recommend that you use the serial console procedure because it requires you to enter minimal information for the machine to join a cluster. See “Joining an Infranet Controller to a Cluster Through Its Serial Console” on page 470.

NOTE: If you add an Infranet Controller running a previous version service package to a cluster, the Infranet Controller automatically detects the mismatch, gets the newer package from the cluster, and joins the cluster. If the new node has no license, it is added with cluster status set to Enabled, Unqualified until you apply a valid CL license using the new node’s machine ID. See “Entering or Upgrading Infranet Controller Licenses” on page 346.

Existing node-specific settings are erased when an Infranet Controller node joins a cluster. These settings include network interface addresses, route tables, virtual ports, ARP caches, VLAN interface, SNMP settings, and so forth. The administrator must manually reconfigure these settings for the newly joined node. You cannot use the Import system configuration feature to import these configurations and settings onto an Infranet Controller node that has been joined to the cluster. See “Importing a System Configuration File” on page 387.

### Specifying an Infranet Controller to Join to a Cluster

Before an Infranet Controller can join a cluster, you must specify its network identity on an active cluster member.
To specify an Infranet Controller that you intend to join to an existing cluster:

1. From the admin console of an active cluster member, select the System > Clustering > Cluster Status tab.

2. Click Add Members to specify an Infranet Controller that will join the cluster:
   a. Enter a name for the member.
   b. Enter the machine’s internal IP address.
   c. Enter the machine’s external IP address if necessary. Note that the External IP address field does not appear if you have not enabled the external port on the System > Network > External Port tab.
   d. Change the netmask and gateway settings for the node if necessary.
   e. Click Add Node. When prompted to confirm adding the new member, click Add.
   f. Repeat this procedure for each Infranet Controller you intend to add to a cluster.

Adding an Infranet Controller to a Cluster Through Its Admin Console

Before you can add an Infranet Controller to a cluster (either through the Web or serial console), you need to make its identity known to the cluster. To specify an Infranet Controller that you intend to add to a cluster, see “Specifying an Infranet Controller to Join to a Cluster” on page 452. Note that if an Infranet Controller has a cluster license key, it has only a Clustering > Join tab.

To add an Infranet Controller to a cluster through its admin console:

1. From an existing cluster member, select the System > Clustering > Cluster Status tab and specify the Infranet Controller you want to add to the cluster. See “Specifying an Infranet Controller to Join to a Cluster” on page 452.

2. From the admin console of the Infranet Controller you want to add to a cluster:
   a. Select System > Configuration > Licensing and enter the correct license key (containing the machine type, the initials CL to indicate a cluster, and the number of endpoints—for example, IC6500-CL-1000E) to enable the clustering feature.
   b. Select the System > Clustering > Join tab and enter:
      - The name of the cluster to join
      - The cluster password you specified when defining the cluster
      - The IP address of an active cluster member
   c. Click Join Cluster. When prompted to confirm joining the cluster, click Join. After the Infranet Controller joins the cluster, you may need to sign in again.
While the new node synchronizes its state with the existing cluster member, each node’s status on the System > Clustering > Cluster Status page indicates Enabled, Enabled, Transitioning, or Enabled, Unreachable.

**Re-adding a Node to a Cluster**

With some maintenance operations, it may be necessary to remove a node from a cluster, then re-add and re-join it to the cluster.

When an Infranet Controller node joins a cluster, all of its node-specific settings (including network interface addresses, route tables, virtual ports, ARP caches, VLAN interface, SNMP settings) are overwritten by the corresponding configuration setting it receives from the cluster.

To populate the newly joined node with the correct node-specific settings:

1. Add the node to the cluster.

2. From any of the existing nodes in the cluster, manually configure the desired node-specific settings for the newly added node.

3. Join the node to the cluster.

When the node joins the cluster, it receives its newly configured node-specific settings from the cluster.

**NOTE:** You configure the node-specific settings for the newly added node manually because binary import options are not useful. The only recommended binary import option into a cluster is “Import everything except network settings and licenses” from the Maintenance > Import/Export > Configuration page which restores cluster-wide configuration (sign-in, realms, roles, resource policies etc.) from a backup binary file. Because this option skips node-specific settings, you must perform step 2 as a manual step in order to populate the newly-joined node with the right set of node-specific settings.

**Configuring Cluster Properties**

This section contains the following information about managing clustering properties:

- “Deploying Two Nodes in an Active/Passive Cluster” on page 455
- “Deploying Two or More Units in an Active/Active Cluster” on page 457
- “Synchronizing the Cluster State” on page 459
- “Configuring Cluster Properties” on page 461
Deploying Two Nodes in an Active/Passive Cluster

You can deploy Infranet Controllers as a cluster pair in active/passive mode. In this mode, one Infranet Controller actively serves user requests while the other Infranet Controller runs passively in the background to synchronize state data, including system state, user profile, and log messages. User requests to the cluster VIP (virtual IP address) are passed to the active Infranet Controller. If the active Infranet Controller goes off line, the standby Infranet Controller automatically starts servicing user requests. Users do not need to sign in again.

You might need to fail-over the cluster VIP to the other node, manually. You can perform a manual failover by using the Fail-Over VIP button on the Clustering Status page. See “Failing Over the VIP to Another Node” on page 456.

Figure 25 on page 455 illustrates an active/passive Infranet Controller cluster configuration using two Infranet Controllers that have enabled external ports. Note that this mode does not increase throughput or user capacity, but provides redundancy to handle unexpected system failure.

User requests are directed to the cluster VIP, which then routes them to the currently active machine.

Figure 25: Active/Passive Cluster Pair
NOTE: In Active/Passive mode:

- A cluster VIP is optional if all of the endpoints that connect to the Infranet Controller support Odyssey Access Client. Odyssey Access Client on the endpoint can connect directly to a cluster node and download information about the other cluster nodes. If the cluster node to which Odyssey Access Client is connected goes offline, Odyssey Access Client automatically connects to the other nodes in the cluster.

However, a cluster VIP is required if you are using agentless access for endpoint platforms such as Macintosh or Linux that connect to the Infranet Controller. See “Configuring Agentless Access to Protected Resources” on page 56.

- A cluster VIP is required if you want the Infranet Enforcer to connect to the active Infranet Controller node. If you do not use a cluster VIP, you must create an Infranet Controller instance on the Infranet Enforcer that specifies the IP address of each node. You can configure an Infranet Enforcer to operate with up to eight Infranet Controllers. (However, the Infranet Enforcer can only operate with one Infranet Controller at a time.) The group of Infranet Controllers with which an Infranet Enforcer operates must all be members of the same cluster. When an Infranet Controller goes down, the Infranet Enforcer tries to connect to each Infranet Controller in its list until it succeeds.

- If you use a VIP, you must use a device certificate for the Infranet Controller that refers to that VIP.

### Failing Over the VIP to Another Node

In an active/passive cluster, you might need to fail-over the VIP to the other node, regardless of which node you are currently using.

To failover the VIP:

1. Select System > Clustering > Cluster Status from the admin console.

2. Click the Fail-Over VIP button to move to the other node. The Fail-Over VIP button is a toggle button, so you can move from one node to the other, regardless of which is the leader.

The failover occurs immediately.
Deploying Two or More Units in an Active/Active Cluster

**NOTE:** When choosing and configuring a load balancer for your cluster, we recommend that you ensure the load balancer:

- Supports IPsec
- Listens for traffic on multiple ports
- Can be configured to manage traffic using assigned source and destination IP addresses (not destination port)

In active/active mode, you can optionally use an external load balancer with an Infranet Controller cluster, though it is not required. If you do use a load balancer, all the nodes actively handle user requests sent by the load balancer or round-robin DNS. The load balancer hosts the cluster VIP and routes user requests to an Infranet Controller defined in its cluster group based on source-IP routing. If an Infranet Controller goes offline, the load balancer adjusts the load on the active Infranet Controllers. Users do not need to sign in again.

The Infranet Controller hosts an HTML page that provides service status for each Infranet Controller in a cluster. External load balancers can check this resource to determine how to effectively distribute the load among all the cluster nodes.

To perform the Layer 7 health check for a node:

- **From a browser**—Enter the URL:
  
  https://<Infranet Controller-Hostname>/dana-na/healthcheck/healthcheck.cgi

- **From an external load balancer**—Configure a health check policy that sends the following request to cluster nodes:

  GET /dana-na/healthcheck/healthcheck.cgi HTTP/1.1
  Host: localhost

  The node returns one of two values:

  - **“Security gateway is accessible” string**—This value means the node is active.
  - **500**—This value denotes an error and cluster Infranet Controllers stop forwarding user requests to the node.
Figure 26 on page 458 illustrates an active/active Infranet Controller cluster configuration in which the Infranet Controllers have enabled external ports.

This active/active cluster configuration is deployed behind an external load balancer. You can deploy a cluster pair or multi-unit cluster in active/active mode. Infranet Controller user requests are directed to the cluster VIP defined on the load balancer, which routes them to the appropriate machine.

**Figure 26: active/active Configuration**
Synchronizing the Cluster State

Infranet Controller state synchronization occurs only by means of the internal network interface cards (NICs), and each cluster member is required to possess the cluster password to communicate with other members. Cluster members synchronize data when there is a state change on any member. Infranet Controller cluster state data is either persistent—permanently stored on the Infranet Controller—or transient—stored on the Infranet Controller only for the user’s session. Infranet Controller state data is divided into the following major categories:

- **System state**—This state is persistent and does not change often.
  - Network settings
  - Authentication server configurations
  - Authorization group configurations, such as access control list, messaging, and application data
  - NACN password, administrator name and password for signing into the Infranet Enforcer using SSH, and serial number(s) of the Infranet Enforcer(s). See “Configuring the Infranet Controller to Connect to the Infranet Enforcer” on page 20.

- **Infranet Enforcer state**—This state is transient.
- The cluster member to which each Infranet Enforcer connects. This enables other cluster members to use the appropriate cluster member for communications with each Infranet Enforcer.

**NOTE:** When the Infranet Controller connects to the Infranet Enforcer the Infranet Controller examines the setting for Cleanup Infranet State Delay on the Infranet Enforcer. If this setting has a default value of 120 seconds, the Infranet Controller changes it to 180 seconds. This is to give a cluster of Infranet Controllers additional time to reboot without disrupting connections from endpoints to resources protected by the Infranet Enforcer.

- **User session**—This state is transient and dynamic. The user session consists of the following data:
  - Host Checker status of the endpoint
  - The user’s set of roles

- **Monitoring state**—This persistent information consists of log messages.

If you notice too much latency occurring on one or more nodes, you might need to change the Clustering Timeouts Settings. See “Configuring Cluster Properties” on page 461.

Whether you deploy a cluster in active/passive or active/active mode, the Infranet Controller is responsible for synchronizing data between cluster members. The Infranet Controller synchronizes all system data and user session data immediately, so if one cluster member goes offline, users do not need to sign in to the Infranet Controller again.

When you add an Infranet Controller to a cluster, the cluster leader does not send log messages to the new member. Log messages are also not synchronized between cluster members when one member restarts its services or when an offline machine comes back online. Once all machines are online, however, log messages are synchronized.

**NOTE:** If you are running an active/active cluster, you must not allow the cluster to switch to active/passive mode unless the active/active and active/passive clusters share compatible spread timeout settings.

You may also configure synchronization settings to improve performance:

- **Specify the synchronization protocol**—When running three or more Infranet Controllers in a multi-unit or multi-site cluster, you can choose to use the synchronization protocol (Unicast, Multicast, or Broadcast) that best suits your network topology.

**NOTE:** See “Specifying Active/Passive, Active/Active, and Other Cluster Settings” on page 462 for a description of the synchronization settings.
- **Synchronize log messages**—Log messages may create a huge payload on the network and affect cluster performance. This option is disabled by default.

- **Synchronize user sessions**—This option synchronizes all user session information (instances of access to intranet services, for example) among all Infranet Controllers in the cluster.

- **Synchronize last access time for user sessions**—This option allows you to propagate user access information in the cluster. If this option is the sole synchronization item among the cluster nodes, you can significantly reduce CPU impact among the cluster Infranet Controllers.

### NOTE:

- If you configure your cluster as active/passive, the **Synchronize user sessions** and **Synchronize last access time for user sessions** options are automatically checked.

- If you select both the **Synchronize log messages** and **Synchronize user sessions** check boxes, everything is replicated on the cluster nodes, including networking information. Even though networking information, including syslog and SNMP settings, can be configured per node or per cluster, all of the networking information is synchronized between nodes when these two options are set.

- If your cluster node configurations have diverged due to changes made to one node while another is disabled or unavailable, the Infranet Controller manages the remerging of the configurations automatically, for up to 16 updates. Beyond the maximum number of allowable updates, you may need to intervene and remerge the configurations manually. In some instances, the Infranet Controller may be unable to remerge the configurations if there is not enough overlapping configuration information between two nodes to manage the internode communication.

For example, given a two-node cluster in which the two nodes are partitioned from each other because of a network outage, if the internal network IP address of one of the nodes gets changed in one of the partitions, the two partitions are unable to rejoin, even when the network is repaired. In such a case, you must manually remerge the configurations.

### Configuring Cluster Properties

Select the **System > Clustering > Cluster Properties** page to specify active/passive, active/active, and other cluster settings. You can also use this page to delete a cluster.

You can deploy a load balancer between several Infranet Controllers with or without clustering. You can use the Juniper Networks DX appliance as the load balancer.
If you configure the Infranet Controller in an active/active cluster, the Infranet Controller normally sends the IP addresses of all of the nodes in the cluster to the Odyssey Access Client so that Odyssey Access Client can connect with other members of the cluster in the event the one it is connected to goes down. If a load balancer is present, the Infranet Controller does not send the IP addresses of all the nodes. The Infranet Controller relies on the load balancer to direct traffic from Odyssey Access Client to an Infranet Controller that is running. See “Configuring the Infranet Controller to Work with a Load Balancer” on page 338.

If you deploy a load balancer, you must provide the load balancer IP address to which agents can connect.

**Specifying Active/Passive, Active/Active, and Other Cluster Settings**

Use the Properties page to change the name of a cluster, specify in which configuration to run a cluster (active/passive or active/active), specify synchronization and network healthcheck settings, or delete a cluster.

To modify cluster properties:

1. From the admin console of an active cluster member, select the **System > Clustering > Cluster Properties** page.

2. Edit the name of the cluster in the **Cluster Name** field to change the cluster’s name (optional).

3. Under **Configuration Settings**, select one of the following options:

   - **Active/Passive** to run a cluster pair in active/passive mode. Then, specify an internal VIP (virtual IP address) and an external VIP if the external port is enabled.

   **NOTE:** To run a two-unit cluster in active/passive mode, the Infranet Controllers must reside on the same subnet.

   - **Active/Active** runs a cluster of two or more nodes in active/active mode using an external load balancer.

   **NOTE:** To change a two-unit active/passive cluster to an active/active cluster with more than two nodes, first change the configuration of the two-unit cluster to active/active and then add the additional nodes.

4. Under **Synchronization Settings**, specify one or more types of data to synchronize using the following options:

   - **Synchronize log messages**—Propagates all log messages among all of the Infranet Controllers in the cluster.

   - **Synchronize user sessions**—Synchronizes all user session information (instances of access to intranet services, for example) among all Infranet Controllers in the cluster.
5. Under Network Healthcheck Settings, specify the number of ARP ping failures allowed before the Infranet Controller’s internal interface is disabled and whether or not to disable the Infranet Controller’s external interface if the internal interface fails.

6. Select the Advanced Settings check box to specify the timeouts for the underlying cluster system. Do not change any values under this setting unless instructed to do so by Juniper Networks Technical Support.

7. Click Save Changes.

8. If you are using an external load balancer, select System > Network > Load Balancer.

9. Depending on the port to which endpoints connect, enter the IP address of the load balancer in the Internal Address box or the External Address box.

10. Select the Between endpoints and the Infranet Controller check box.

11. Click Save Changes.
Managing and Configuring Clusters

This section contains the following instructions for managing and configuring clusters:

- “Managing Network Settings for Cluster Nodes” on page 465
- “Upgrading Clustered Nodes” on page 465
- “Upgrading the Cluster Service Package” on page 465
- “Deleting a Cluster” on page 465
- “Restarting or Rebooting Clustered Nodes” on page 466
- “Admin Console Procedures” on page 466
- “Monitoring Clusters” on page 467
- “Troubleshooting Clusters” on page 468
- “Serial Console Procedures” on page 470

Adding Multiple Cluster Nodes

You can add multiple cluster nodes at one time. You can configure all of the nodes before saving and enabling the multiple node configuration.

To add multiple nodes to a cluster:

1. Select **System > Clustering > Cluster Status**.
2. Click **Add Members**.
3. Enter the node name and internal IP address.
4. Modify or add the default internal netmask and internal gateway addresses, if necessary.
5. Click **Add**.
6. Repeat the process until you have added all of the nodes.
7. Click **Save Changes** to save the node configurations.

The Infranet Controller automatically enables the added clusters, even if they are unreachable.
**Managing Network Settings for Cluster Nodes**

To modify the network settings for a cluster or each individual node in a cluster, click `System > Network`. You can make your changes on the Network Settings pages. After you create a cluster, these pages provide a drop-down list from which you can select the entire cluster or a specific node to modify. When you save changes on a Network page, the settings are saved for the specified cluster or cluster node. If you change network settings for an entire cluster, they propagate to every node in the cluster.

**NOTE:** You can access a node-specific Network page by clicking `System > Clustering > Cluster Status` on the node’s name in the Member Name column.

**Upgrading Clustered Nodes**

You can easily upgrade every node in a cluster using the Infranet Controller. You simply install a newer service package on one node and, once the installation completes and the node reboots, the node pushes the service package to all nodes in the cluster.

For more information about disabling nodes to upgrade the service package, see “Upgrading the Cluster Service Package” on page 465.

**NOTE:** If you import an XML configuration file into a cluster, all members of a cluster are disabled and all end-user sessions are terminated during the import process. After the import process completes, the cluster members are automatically enabled but users must sign in again. See “Importing XML Configuration Data” on page 398.

**Upgrading the Cluster Service Package**

Install a newer service package on one cluster node only. When the installation process completes and the cluster node reboots, it instructs the other nodes to upgrade. See “Upgrading or Downgrading the Infranet Controller” on page 343 for more information about installing a service package.

**Deleting a Cluster**

If you delete a cluster, all of the nodes begin running as stand alone Infranet Controller systems.

To delete a cluster:

1. From the admin console of an active cluster member, select the `System > Clustering > Cluster Status` page.
2. Select the checkbox next to each cluster node you want to delete.
3. Click the `Remove Cluster` button.
4. When prompted, click `Remove`. 
When the operation completes, all cluster nodes begin running as stand-alone Infranet Controller systems.

### Restarting or Rebooting Clustered Nodes

When you create a cluster of two or more Infranet Controllers, the clustered Infranet Controllers act as a logical entity. As such, when you restart or reboot one of the clustered Infranet Controllers using either the serial console or the admin console, all Infranet Controllers in the cluster restart or reboot.

To restart or reboot only one Infranet Controller in a cluster:

1. Select the **System > Clustering > Status** page to disable the Infranet Controller you want to restart or reboot within the cluster.

2. Use the controls on the **Maintenance > System > Platform** page, or the serial console’s **Reboot this IC, Shutdown this IC, or Restart Services in this IC** menu items under **System Operations**.

3. To restart or reboot the Infranet Controller. After the Infranet Controller restarts or reboots, enable the Infranet Controller within the cluster again.

The Infranet Controller reconciles session state with the Infranet Enforcer upon restart or cluster failover. If the Infranet Enforcer is running ScreenOS 6.0r2 or later, an Infranet Controller restart or failover does not interrupt network traffic of existing sessions, as long as the Infranet Controller restart or cluster failover happens within two minutes.

### Admin Console Procedures

Table 39 describes the information displayed on the **Status** tab and the various management tasks you can perform, including disabling, enabling, and removing an Infranet Controller node from a cluster.

<table>
<thead>
<tr>
<th>User Interface Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status Information labels</td>
<td>Screen displays the cluster name, type, configuration, internal VIP, and external VIP for an active/passive cluster.</td>
</tr>
<tr>
<td>Add Members button</td>
<td>Click this button to specify an Infranet Controller that will join the cluster. You must perform this step for Infranet Controller systems you intend to add to the cluster. By clicking this button, you can add multiple nodes at the same time.</td>
</tr>
<tr>
<td>Enable button</td>
<td>Click this button to add a node that was previously disabled. When you add a node, all state information is synchronized on the node.</td>
</tr>
<tr>
<td>Disable button</td>
<td>Click this button to disable a node within the cluster. The node retains awareness of the cluster, but does not participate in state synchronizations or receive user requests unless members sign in to the node, directly.</td>
</tr>
<tr>
<td>Remove button</td>
<td>Click this button to remove the selected node or nodes from the cluster. Once removed, the node runs in stand-alone mode.</td>
</tr>
<tr>
<td>Member Name column</td>
<td>Lists all nodes belonging to the cluster. You can click on a node’s name to modify its name and network settings.</td>
</tr>
</tbody>
</table>
Table 39: Cluster Status Page Information (Continued)

<table>
<thead>
<tr>
<th>User Interface Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internal Address column</td>
<td>Shows the internal IP address of the cluster member using Classless Inter Domain Routing (CIDR) notation.</td>
</tr>
<tr>
<td>External Address column</td>
<td>Shows the external IP address of the cluster member using CIDR notation. Note that this column only shows the external IP address of the cluster leader unless you specify a different address for the node on its individual network settings page, which is accessible by clicking on its name in the Member Name column. If you change the external IP address on the Network &gt; Network Settings page, the change affects all cluster nodes.</td>
</tr>
</tbody>
</table>
| Status column | Shows the current state of the node:  
  - **Green light/enabled** — The node is handling user requests and participating in cluster synchronization.  
  - **Yellow light/transitioning** — The node is joining cluster.  
  - **Red light/disabled** — The node is not handling user requests or participating in cluster synchronization.  
  - **Red light/enabled, unreachable** — The node is enabled, but due to a network issue, it cannot be reached.  
  
  **Note:** A node’s state is considered “stand-alone” when it is deployed outside of a cluster or after being removed from a cluster. |
| Notes column | Shows the status of the node’s connection to the cluster:  
  - **OK** — The node is actively participating in the cluster.  
  - **Transitioning** — The node is switching from the stand-alone state to the enabled state.  
  - **Unreachable** — The node is not aware of the cluster. A cluster member may be “unreachable” even when it’s online and can be pinged. Possible reasons include: its password is incorrect, it doesn’t know about all cluster nodes, it’s configured with a different group communication mode, it’s running a different service package version, or the machine is turned off. |
| Sync Rank column | Specifies the synchronization order for nodes when rejoining a cluster. Accepts sync ranks from 0 (lowest rank) to 255 (highest rank). The highest rank takes precedence. Where two nodes have identical sync ranks, the alpha-numeric rank of the member name is used to determine precedence. |

**Monitoring Clusters**

You can monitor clusters using the standard logging tools provided by the Infranet Controller. In particular, you can use several cluster-specific SNMP traps to monitor events that occur on your cluster nodes, such as:

- External interface down
- Internal interface down
- Disabled node
- Changed virtual IP (VIP)
- Deleted cluster node (cluster stop)
NOTE: Generally, it is desirable to configure your SNMP traps on a cluster-wide basis, so that any given cluster node can send its generated traps to the right target. Setting up cluster-wide configuration for the traps is particularly important when you also use a load balancer, because you may not know which node is responsible for a specific operation. In that case, the load balancer may independently determine which cluster node can manage an administrative session.

You can use SNMP traps that are included in the Juniper Networks Standard MIB to monitor these events. These traps include:

- **iveNetExternalInterfaceDownTrap**—Supplies type of event that brought down the external interface.
- **iveNetInternalInterfaceDownTrap**—Supplies type of event that brought down the internal interface.
- **iveClusterDisableNodeTrap**—Supplies the cluster name on which nodes have been disabled, along with a space separated list of disabled node names.
- **iveClusterChangedVIPTrap**—Supplies the type of the VIP, whether external or internal, and its value before and after the change.
- **iveClusterDelete**—Supplies the name of the cluster node on which the cluster delete event was initiated.

These traps are always enabled and available in the MIB. You cannot disable the traps. For more information about specific traps, see “Status/Error Objects” on page 428.

**Troubleshooting Clusters**

When you have problems with cluster communication, you may be directed by your Juniper Networks Support representative to use the cluster node troubleshooting tools.

To use the cluster node troubleshooting tools:

From the admin console, select **Maintenance > Troubleshooting > Monitoring > Node Monitor**, in **Maintenance > Troubleshooting > Clustering Network Connectivity**, and in **Maintenance > Troubleshooting > Clustering Group Communication**. See “Monitoring Cluster Nodes” on page 445.

You can use a built-in feature on the clustering Status page to identify the status of each cluster node. Pause the mouse pointer over the Status light icon and the system displays a tool tip containing a hexadecimal number. The hexadecimal number is a snapshot of the status of the Infranet Controller. It is a bit mask indicating a number of states as shown in Table 40.

**Table 40: Cluster Status**

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x000001</td>
<td>Infranet Controller is in standalone mode.</td>
</tr>
</tbody>
</table>
Each code, as you see it in the Infranet Controller, may relate specifically to one state. However, each code may represent a combination of states, and so the actual code does not appear in Table 40. Instead, the code you see in the Infranet Controller is the sum of several of the hexadecimal numbers shown in Table 40. You will need to factor out the codes, as in the following example:

- **0x38004** —The right-most digit (4) in this hexadecimal number corresponds to:
  - 0x000004 The Infranet Controller is in cluster enabled state.
  - 0x38004 The digit in the fourth position from the right (8) corresponds to:

### Table 40: Cluster Status

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x000002</td>
<td>Infranet Controller is in cluster disabled state.</td>
</tr>
<tr>
<td>0x000004</td>
<td>Infranet Controller is in cluster enabled state.</td>
</tr>
<tr>
<td>0x000008</td>
<td>Infranet Controller is unreachable (because it is offline, has wrong password, has different cluster definition, different version, or a related problem).</td>
</tr>
<tr>
<td>0x00002000</td>
<td>The node owns the VIPs (on) or not (off).</td>
</tr>
<tr>
<td>0x000100</td>
<td>Infranet Controller is syncing state from another Infranet Controller (initial syncing phase).</td>
</tr>
<tr>
<td>0x000200</td>
<td>Infranet Controller is transitioning from one state to another.</td>
</tr>
<tr>
<td>0x00020000</td>
<td>The group communication subsystems at the local and remote nodes are disconnected from each other.</td>
</tr>
<tr>
<td>0x00040000</td>
<td>Management interface (mg0) appears disconnected.</td>
</tr>
<tr>
<td>0x00080000</td>
<td>Management gateway is unreachable for ARP ping.</td>
</tr>
<tr>
<td>0x00080800</td>
<td>Infranet Controller int0 appears disconnected (no carrier).</td>
</tr>
<tr>
<td>0x001000</td>
<td>Infranet Controller int1 appears disconnected (no carrier).</td>
</tr>
<tr>
<td>0x002000</td>
<td>Infranet Controller is syncing its state to another Infranet Controller that is joining.</td>
</tr>
<tr>
<td>0x004000</td>
<td>Initial Synchronization as master or slave is taking place.</td>
</tr>
<tr>
<td>0x008000</td>
<td>This Infranet Controller is the leader of the cluster.</td>
</tr>
<tr>
<td>0x010000</td>
<td>The spread daemon is running and the cache server is connected to it.</td>
</tr>
<tr>
<td>0x020000</td>
<td>The gateway on int0 is unreachable for ARP pings (see log file).</td>
</tr>
<tr>
<td>0x040000</td>
<td>The gateway on int1 is unreachable for ARP pings (see log file).</td>
</tr>
<tr>
<td>0x080000</td>
<td>Leader election is taking place.</td>
</tr>
<tr>
<td>0x100000</td>
<td>Server life cycle process (dsmon) is busy.</td>
</tr>
<tr>
<td>0x200000</td>
<td>System performs post state synchronization activities.</td>
</tr>
<tr>
<td>0x30004</td>
<td>The spread daemon is running and the cache server is connected to it.</td>
</tr>
<tr>
<td></td>
<td>The gateway on int0 is unreachable for ARP pings (see log file).</td>
</tr>
<tr>
<td></td>
<td>Infranet Controller is in cluster enabled state.</td>
</tr>
<tr>
<td>0x38004</td>
<td>The spread daemon is running and the cache server is connected to it.</td>
</tr>
<tr>
<td></td>
<td>The gateway on int0 is unreachable for ARP pings (see log file).</td>
</tr>
<tr>
<td></td>
<td>This Infranet Controller is the leader of the cluster.</td>
</tr>
<tr>
<td></td>
<td>Infranet Controller is in cluster enabled state.</td>
</tr>
</tbody>
</table>
0x008000 This Infranet Controller is the leader of the cluster.

0x38004—The left-most digit (3) in this hexadecimal number does not exist in the table, which indicates that it corresponds to the sum of two other digits, in this case, 1 and 2, as shown in the following codes:

0x020000—The gateway on int0 is unreachable for ARP pings (see log file).

0x010000—The spread daemon is running and the cache server is connected to it.

Serial Console Procedures

If you are adding a factory-set Infranet Controller to a cluster, we recommend that you use the serial console, which enables you to join an existing cluster during the initialization process by entering minimal information. When an Infranet Controller joins a cluster, it receives the cluster state settings, which overwrites all settings on a machine with an existing configuration and provides new machines with the required preliminary information.

You can also use an Infranet Controller’s serial console to disable an Infranet Controller within a cluster. If an Infranet Controller is in synchronization state, you cannot access its admin console. Therefore, if you need to upgrade or reboot the Infranet Controller, for example, you need to first disable the Infranet Controller from a cluster through its serial console.

Joining an Infranet Controller to a Cluster Through Its Serial Console

Before a configured or factory-set Infranet Controller can join a cluster, you need to make its identity known to the cluster. For instructions, see “Specifying an Infranet Controller to Join to a Cluster” on page 452.

NOTE:

- If you want to add an Infranet Controller currently running as a stand-alone machine to a cluster through its admin console, it must be running the same or a more recent version service package on the same hardware platform as the other members.

- If you add an Infranet Controller running a previous version service package to a cluster, the Infranet Controller automatically detects the mismatch, gets the newer package from the cluster, and joins the cluster.

To add an Infranet Controller to a cluster through its serial console:

1. From the admin console of an existing cluster member, select the System > Clustering > Cluster Status tab and specify the Infranet Controller you want to add to the cluster. See “Specifying an Infranet Controller to Join to a Cluster” on page 452.

2. Connect to the serial console of the machine you want to add to the cluster. See “Infranet Controller Serial Console” on page 497.
3. Reboot the machine and watch its serial console. After the system software starts, a message displays stating that the machine is about to boot as a stand-alone Infranet Controller and to press the Tab key for clustering options. Press the Tab key as soon as you see this option.

**NOTE:** The interval to press the Tab key is five seconds. If the machine begins to boot in stand alone mode, wait for it to finish and then reboot again.

4. Enter the number instructing the Infranet Controller to join an existing cluster.

5. Enter the requested information, including:
   - The internal IP address of an active member in the cluster
   - The cluster password, which is the password you entered when defining the cluster
   - The name of the machine you wish to add
   - The internal IP address of the machine you wish to add
   - The netmask of the machine you wish to add
   - The gateway of the machine you wish to add

The active cluster member verifies the cluster password and that the new machine’s name and IP address match what you specified in the admin console by clicking System > Clustering > Cluster Status > Add Cluster Member. If the credentials are valid, the active member copies all of its state data to the new cluster member, including license key, certificate, user, and system data.

6. Enter the number instructing the Infranet Controller to continue the join cluster operation. When you see the message confirming that the machine has joined the cluster, click System > Clustering > Cluster Status tab in the admin console of any active cluster member to confirm that the new member’s Status is green, indicating that the Infranet Controller is an enabled node of the cluster.

**Disabling a Clustered Infranet Controller by Using Its Serial Console**

To disable an Infranet Controller within a cluster using its serial console:

1. Connect to the serial console of the machine you want to disable within the cluster. For more information, see “Infranet Controller Serial Console” on page 497.

2. Enter the number corresponding to the Infranet Controller’s System Operations option.

3. Enter the number corresponding to the Disable Node option.

4. Enter y when the serial console prompts if you are sure you want to disable the node.
5. Verify that the Infranet Controller has been disabled within the cluster by selecting **System > Clustering > Status** in the admin console of any active cluster member to confirm that the disabled member’s Status is red.

**Changing the IP Address of a Cluster Node**

To change the IP address of a cluster node:

1. Select **System > Clustering > Cluster status**.

2. Select the check box next to the name of the node whose IP address you want to change.

3. Click **Remove**.

4. After the node is removed, sign in to the node and change its IP address.

5. Click **Save Changes**.

6. Rejoin the node to the cluster.

---

**NOTE:** If you attempt to change the IP address of a node while it belongs to a cluster, you may experience unpredictable results.
Chapter 16
Delegating Administrator Roles

The Infranet Controller access management system enables you to delegate various Infranet Controller management tasks to different administrators through system administrator roles and security administrator roles. System and security administrator roles are defined entities that specify Infranet Controller management functions and session properties for administrators who are mapped to those roles. You can customize an administrator role by selecting the Infranet Controller feature sets, user roles, authentication realms, and resource policies that members of the administrator role are allowed to view and manage. Note that system administrators may only manage user roles, realms, and resource policies; only security administrators can manage administrator components.

For example, you may choose to create an administrator role called “Help Desk Administrators” and assign users to this role who are responsible for fielding tier 1 support calls, such as helping users understand why they cannot sign in or access protected resources. In order to help with troubleshooting, you may configure settings for the “Help Desk Administrators” role as follows:

- Allow the help desk administrators Write access to the System > Log/Monitoring page so they can view and filter the Infranet Controller logs, tracking down critical events in individual users’ session histories, as well as the Maintenance > Troubleshooting page so they can trace problems on individual users’ systems.

- Allow the help desk administrators Read access to the Users > User Roles pages so they can view the restrictions on individual users’ roles, as well as the Resource Policy pages so they can view the policies that may be denying individual users access to protected resources.

- Deny the help desk administrators any access to the remaining System pages and Maintenance pages, which are primarily used for configuring system-wide settings—such as installing licenses and service packages—not for troubleshooting individual users’ problems.

**NOTE:** In addition to any delegated administrator roles that you may create, the Infranet Controller also includes two basic types of administrators: super administrators (.Administrators role), who can perform any administration task through the admin console and read-only administrators (.Read-only Administrators role), who can view—but not change—the entire Infranet Controller configuration through the admin console.
You can also create a security administrator role called “Help Desk Manager” and assign users to this role who are responsible for managing the Help Desk Administrators. You might configure settings for the “Help Desk Manager” role to allow the Help Desk Manager to create and delete administrator roles on his own. The Help Desk Manager might create administrator roles that segment responsibilities by functional areas of the Infranet Controller. For example, one administrator role might be responsible for all log monitoring issues. Another might be responsible for all problems related to accessing protected resources.

This section contains the following information about delegated administration:

- “Creating and Configuring Administrator Roles” on page 474
- “Specifying Management Tasks to Delegate” on page 476
- “Defining General System Administrator Role Settings” on page 481

Creating and Configuring Administrator Roles

When you navigate to **Administrators > Admin Roles**, you can find the **Administrators** page. From this page, you can set default session and user interface options for delegated administrator roles.

This section contains the following information about creating and configuring delegated administrator roles:

- “Creating Administrator Roles” on page 474
- “Modifying Administrator Roles” on page 475
- “Deleting Administrator Roles” on page 475

**NOTE:** To create individual administrator accounts, you must add the users through the appropriate authentication server (not the role). For example, to create an individual administrator account, you may use settings in the **Authentication > Auth. Servers > Administrators > Users** page of the admin console. For detailed instructions on how to create users on the Administrators server and other local authentication servers, see “Creating User Accounts on a Local Authentication Server” on page 202. For instructions on how to create users on third-party servers, see the documentation that comes with that product.

Creating Administrator Roles

To create an administrator role:

1. In the admin console, choose **Administrators > Admin Roles**.
2. Do one of the following:
   - Click **New Role** to create a new administrator role with the default settings.
Select the checkbox next to an existing administrator role and click **Duplicate** to copy the role and its custom permissions. Note that you cannot duplicate the system default roles (.Administrators and .Read-Only Administrators).

3. Enter a **Name** (required) and **Description** (optional) for the new role and click **Save Changes**.

4. Modify settings for the role using instructions in:

   - “Managing General Role Settings and Options” on page 481
   - “Delegating User and Role Management” on page 476
   - “Delegating User Realm Management” on page 477
   - “Delegating Administrative Management” on page 478
   - “Delegating Resource Policy Management” on page 480

**Modifying Administrator Roles**

To modify an existing administrative role:

1. In the admin console, choose **Administrators > Admin Roles**.
2. Click the name of the administrator role that you want to modify.
3. Modify settings for the role using instructions in:

   - “Managing General Role Settings and Options” on page 481
   - “Delegating User and Role Management” on page 476
   - “Delegating User Realm Management” on page 477
   - “Delegating Administrative Management” on page 478
   - “Delegating Resource Policy Management” on page 480

**NOTE:** If you select one of the Infranet Controller’s default administrator roles (.Administrators or .Read-Only Administrators), you can only modify settings in the **General** tab (since the default Infranet Controller administrators roles always have access to the functions defined through the **System**, **Users**, **Administrators**, and **Resource Policies** tabs).

**Deleting Administrator Roles**

To delete an existing administrative role:

1. In the admin console, choose **Administrators > Admin Roles**.
2. Click the checkbox next to the administrator role that you want to delete and click **Delete**.

3. Click **Delete** to confirm that you want to remove the selected role.

---

**NOTE:** You cannot delete the **Administrators** and **Read Only Administrators** roles since they are default roles defined on the Infranet Controller.

---

### Specifying Management Tasks to Delegate

This section contains the following information about delegating management tasks to various delegated administrator roles:

- “Delegating System Management Tasks” on page 476
- “Delegating User and Role Management” on page 476
- “Delegating User Realm Management” on page 477
- “Delegating Administrative Management” on page 478
- “Delegating Resource Policy Management” on page 480

#### Delegating System Management Tasks

Use the **Administrators > Admin Roles > Select Role > System** tab to delegate various Infranet Controller system management tasks to different administrator roles. When delegating privileges, note that:

- The Infranet Controller allows all administrators read-access (at minimum) to the admin console home page (**System > Status > Overview**), regardless of the privilege level you choose.

- The Infranet Controller does not allow delegated administrators write-access to pages where they can change their own privileges. Only those administrator roles that come with the system (**Administrators** and **Read-Only Administrators**) may access these pages:

  - Maintenance > Import/Export (Within this page, **Read-Only Administrators** can export settings, but cannot import them.)

  - Maintenance > Archiving > Local Backups

#### Delegating User and Role Management

Use the **Administrators > Admin Roles > Select Role > Users > Roles** sub-tab to specify which user roles the administrator role can manage. When delegating role management privileges, note that:

- Delegated administrators can only manage user roles.
Delegated administrators cannot create new user roles, copy existing roles, or delete existing roles.

If you allow the delegated administrator to read or write to any feature within a user role, the Infranet Controller also grants the delegated administrator read-access to the Users > User Roles > Select Role > General > Overview page for that role.

If you grant a delegated administrator write access to a resource policy through the Administrators > Admin Roles > Select Administrator Role > Resource Policies page, he may create a resource policy that applies to any user role, even if you do not grant him read access to the role.

To define role management privileges for an administrative role:

1. In the admin console, choose Administrators > Admin Roles.
2. Select the administrator role that you want to modify.
3. Select the Users > Roles tab.
4. Under Delegate user roles, specify whether the administrator can manage all roles or only selected roles. If you only want to allow the administrator role to manage selected user roles, select those roles in the Available roles list and click Add.
5. Specify which user role pages the delegated administrator can manage by selecting one of the following options:
   - Write All—Specifies that members of the administrator role can modify all user role pages.
   - Custom Settings—Allows you to pick and choose administrator privileges (Deny, Read, or Write) for the individual user role pages.
6. Under Delegate as read-only roles, select the user roles that you want to allow the administrator to view, but not manage.
7. Click Save Changes.

**NOTE:** If you specify both write access and read-only access for a feature, the Infranet Controller grants the most permissive access. For example, if you choose Administrators can manage ALL roles under Delegated user roles, and then select the “Users” role in the Delegate as read-only roles section, the Infranet Controller allows the delegated administrator role full management privileges to the “Users” role.

**Delegating User Realm Management**

Use the Administrators > Admin Roles > Select Role > Users > Authentication Realms tab to specify which user authentication realms the administrator role can manage. When delegating realm management privileges, note that:

- System administrators can only manage user realms.
- System administrators cannot create new user realms, copy existing realms, or delete existing realms.

- If you allow the system administrator to read or write to any user realm page, the Infranet Controller also grants the system administrator read-access to the Users > User Realms > Select Realm > General page for that role.

To define realm management privileges for an administrative role:

1. In the admin console, choose Administrators > Admin Roles.

2. Select the administrator role that you want to modify.

3. Select the Users > Authentication Realms tab.

4. Under Delegate user realms, specify whether the administrator can manage all user authentication realms or only selected user authentication realms. If you only want to allow the administrator role to manage selected realms, select those realms in the Available realms list and click Add.

5. Specify which user authentication realms pages the delegated administrator can manage by selecting one of the following options:
   - Write All—Specifies that members of the administrator role can modify all user authentication realm pages.
   - Custom Settings—Allows you to pick and choose administrator privileges (Deny, Read, or Write) for the individual user authentication realm pages.

6. Under Delegate as read-only realms, select the user authentication realms that you want to allow the administrator to view, but not modify.

7. Click Save Changes.

**Delegating Administrative Management**

Use the Administrators > Admin Roles > Select Roles > Administrators tab to specify which system administrator roles and realms the security administrator role can manage. When delegating security administrative privileges, note that:

- The security administrator role provides control over all administrative roles and realms.
- You can give a security administrator control exclusively over administrator roles, over administrator realms, or over both.
You can restrict or grant the security administrator the permission to add and delete administrator roles and administrator realms.

To define security administrator privileges:

1. In the admin console, choose Administrators > Admin Roles > Select Role > Administrators.

2. Select the Manage ALL admin roles checkbox.

3. If you want to allow the security administrator to add and delete admin roles, check the Allow Add/Delete admin roles checkbox. This allows the security administrator the ability to create administrator roles, even if the security administrator is not part of the .Administrators role.

4. Indicate the level of access that you want to allow the security administrator role to set for system administrators for each major set of admin console pages (General, System tasks, Users, Administrators, and Resource Policies) by choosing one of the following options:
   - **Deny All**—Specifies that members of the security administrator role cannot see or modify any settings in the category.
   - **Read All**—Specifies that members of the security administrator role can view, but not modify, all settings in the category.
   - **Write All**—Specifies that members of the security administrator role can modify all settings in the category.
   - **Custom Settings**—Allows you to pick and choose security administrator privileges (Deny, Read, or Write) for the individual features within the category.

5. Select the Manage ALL admin realms checkbox.

6. If you want to allow the security administrator to add and delete admin realms, check the Allow Add/Delete admin realms checkbox. This allows the security administrator the ability to create and delete administrator realms, even if the security administrator is not part of the .Administrators role.

7. Indicate the level of realm access that you want to allow the security administrator role to set for system administrators for each major set of admin console pages (General, Authentication Policy, and Role Mapping) by choosing one of the following options:
   - **Deny All**—Specifies that members of the security administrator role cannot see or modify any settings in the category.
   - **Read All**—Specifies that members of the security administrator role can view, but not modify, all settings in the category.
   - **Write All**—Specifies that members of the security administrator role can modify all settings in the category.
Custom Settings—Allows you to pick and choose security administrator privileges (Deny, Read, or Write) for the individual features within the category.

NOTE: All administrators that can manage admin roles and realms have at least read-only access to the admin role’s Name and Description and to the realm’s Name and Description, as displayed on the General page.

8. Click Save Changes.

Delegating Resource Policy Management

Use the Administrators > Admin Roles > Resource Policies tab to specify which user resource policies the administrator role can manage. When delegating resource policy management privileges, note that delegated system administrators cannot modify the following characteristics of resource policies:

- The resource itself (that is, the IP address/netmask)
- The order in which the Infranet Enforcer evaluates the resource policies.

To delegate administrator privileges for resource policies:

1. In the admin console, choose Administrators > Admin Roles.
2. Select the administrator role that you want to modify.
4. Indicate the level of access that you want to allow the administrator role for each Resource Policies sub-menu by choosing one of the following options:
   - Deny All—Specifies that members of the administrator role cannot see or modify any resource policies.
   - Read All—Specifies that members of the administrator role can view, but not modify, all resource policies.
   - Write All—Specifies that members of the administrator role can modify all resource policies.
   - Custom Settings—Allows you to pick and choose administrator privileges (Deny, Read, or Write) for each type of resource policy or for individual resource policies.

5. If you want to set custom access levels for an individual policy:
   a. Select Custom Settings (above).
   b. Click the Additional Access Policies link next to the appropriate category.
   c. Choose the access level for the policy (Deny, Read, or Write).
d. Under **Access Policies**, select the resource policy for which you want to provide a custom access level and click **Add**.

6. Click **Save Changes**.

---

**Defining General System Administrator Role Settings**

This section contains the following information about configuring general options for delegated system administrator roles:

- “Defining Default Options for Administrator Roles” on page 481
- “Managing General Role Settings and Options” on page 481
- “Specifying Access Management Options for the Role” on page 482
- “Specifying General Session Options” on page 482
- “Specifying UI Options” on page 483

---

**Defining Default Options for Administrator Roles**

To define the default options for all delegated administrator roles:

1. In the admin console, choose **Administrators > Admin Roles**.
2. Click **Default Options**.
3. Modify settings in the **Session Options** and **UI Options** tabs using instructions in “Managing General Role Settings and Options” on page 481 and click **Save Changes**. These become the new defaults for all new delegated administrator roles.

---

**Managing General Role Settings and Options**

To manage general role settings and options:

1. In the admin console, choose **Administrators > Admin Roles > Select Role > General > Overview**.
2. Create a label for the delegated administrator role using the **Name** and **Description** fields (optional).
3. Under **Options**, check:
   - **Session Options** to apply the settings configured in the **General > Session Options** tab to the role.
   - **UI Options** to apply the settings configured in the **General > UI Options** tab to the role.
4. Click **Save Changes** to apply the settings to the role.
Specifying Access Management Options for the Role

Use the Administrators > Admin Roles > General > Restrictions tab to specify access management options for the role. The Infranet Controller does not map administrators to this role unless they meet the specified restrictions. For more information, see “Access Management Framework” on page 65.

To specify access management options for the role:

1. In the admin console, choose Administrators > Admin Roles > Select Role > General > Restrictions.

2. Click the tab corresponding to the option you want to configure for the role, and then configure it using the instructions in the following sections:

   - “Specifying Source IP Access Restrictions” on page 108
   - “Specifying Browser Access Restrictions” on page 110
   - “Specifying Certificate Access Restrictions” on page 112
   - “Specifying Host Checker Access Restrictions” on page 114

You may configure any number of access management options for the role. If an administrator does not conform to all of the restrictions, then the Infranet Controller does not map the delegated administrator to the role.

3. Click Save Changes to apply the settings to the role.

Specifying General Session Options

To specify general session options:

1. In the admin console, choose Administrators > Admin Roles > Select Role > General > Session Options.

2. Under Session Lifetime, specify values for:

   - **Idle Timeout**—Specify the number of minutes an administrator session may remain idle before ending. The minimum is 5 minutes. The default idle session limit is ten minutes, which means that if an administrator’s session is inactive for ten minutes, the Infranet Controller ends the session and logs the event in the system log (unless you enable session timeout warnings described below).

   - **Max. Session Length**—Specify the number of minutes an active administrator session may remain open before ending. The minimum is 6 minutes. The default time limit for an administrator session is sixty minutes, after which the Infranet Controller ends the session and logs the event in the system log.
3. Under **Roaming session**, specify:

- **Enabled**—To enable roaming user sessions for users mapped to this group. A roaming user session works across source IP addresses, which allows mobile administrators (laptop users) with dynamic IP addresses to sign in to the Infranet Controller from one location and continue working from another. Disable this feature to prevent users from accessing a previously-established session from a new source IP address. This helps protect against an attack spoofing a user’s session, provided the hacker was able to obtain a valid user’s session cookie.

- **Limit to subnet**—To limit the roaming session to the local subnet specified in the **Netmask** field. Administrators may sign in from one IP address and continue using their sessions with another IP address as long as the new IP address is within the same subnet.

- **Disabled**—To disable roaming sessions for administrators mapped to this role. Administrators who sign in from one IP address may not continue an active Infranet Controller session from another IP address; administrator sessions are tied to the initial source IP address.

4. Click **Save Changes** to apply the settings to the role.

**Specifying UI Options**

Use the **Administrators > Admin Roles > Select Role > General > UI Options** tab to customize admin console settings for the administrators mapped to this role, including console colors, logos, and hierarchical navigation menus. (For information about customizing the logo and colors in the admin console sign-in page, see “Configuring Standard Sign-In Pages” on page 266.)

**Hierarchical navigation menus** are dynamic menus that appear when you mouse over one of the menus in the left panel of the admin console. For example, if you enable hierarchical navigation menus and then hover over the **Authentication > Signing In** menu in the admin console, the **Sign-In Policies**, and **Sign-In Pages** sub-menus appear. You can use these menus to quickly navigate through the system without having to click through the whole menu hierarchy.

**NOTE:**

- For information about the environments in which hierarchical menus are supported, see the Supported Platforms Guide on the Juniper Networks Customer Support Center.

- If you have defined over 10 authentication realms or roles under **Administrators** or **Users**, the admin console only displays the 10 most recently accessed roles or realms in the hierarchical navigation menus. Note that the Infranet Controller does not display the 10 roles and realms most recently accessed by the current administrator—instead it displays the 10 roles and realms accessed by all administrators who have signed in to this Infranet Controller.
To customize the Infranet Controller welcome page for role users:

1. In the admin console, choose **Administrators > Admin Roles > Select Role**.

2. Select the **UI Options** checkbox on the **General > Overview** tab to enable settings for the role.

3. Choose **General > UI Options** to customize settings for the role.

4. In the **Header** section, specify a custom logo image file for the header and a different header color.

5. In the **Navigation Menus** section, choose whether you wish to display hierarchical navigation menus. Options include:
   - **Auto-enabled**—The Infranet Controller determines whether the administrator is signed in from a supported platform and enables or disables the hierarchical menus accordingly.
   - **Enabled**—The Infranet Controller enables hierarchical menus, regardless of your platform. If the administrator is signed in from an unsupported platform, they may not be able to use the hierarchical menus, even though they are enabled.
   - **Disabled**—The Infranet Controller disables hierarchical menus for all members of the role.

6. In the **Other** section, select the **Show copyright notice and “Secured by Juniper Networks” label in footers** checkbox to display the Juniper logo.

**NOTE:** If you do not want user roles to see the copyright notice, you can also deselect the option in the **Default Settings** for user roles, in general. That way, all subsequent roles you create do not allow the notice to appear on the end-user UI.

7. Click **Save Changes**. The changes take effect immediately, but current user browser sessions may need to be refreshed to see the changes. Or click **Restore Factory Defaults** to reset the admin console appearance to the default settings.
Chapter 17
Infranet Controller and IDP Interoperability

Securing intranet work application and resource traffic is vital to protecting your network. You can add levels of application security to detect internal threats coming from users who are authenticated through the Infranet Controller by integrating an Infranet Controller with a Juniper Networks Intrusion Detection and Prevention (IDP) Sensor.

The Infranet Controller supports stand alone IDP and IDP through the ISG-1000 and ISG-2000 Infranet Enforcer with the IDP Security Module (supported in ScreenOS release 6.2 or greater).

The IDP sensor monitors the network on which the IDP system is installed. The sensor’s primary task is to detect suspicious and anomalous network traffic based on specific rules defined in IDP rulebases.

The IDP device provides the following types of protection (some forms of protection depend upon the specific configuration):

- Protects against attacks from user to application
- Detects and blocks most network worms based on software vulnerabilities
- Detects and blocks non-file-based Trojan Horses
- Detects and blocks effects of spyware, adware, and key loggers
- Detects and blocks many types of malware
- Detects and blocks zero day attacks through the use of anomaly detection

NOTE: An IDP Sensor can send logs to one Infranet Controller appliance only. However, an Infranet Controller appliance can receive logs from more than one IDP Sensor.

Using the Infranet Controller’s Admin console, you can configure and manage interaction attributes between the Infranet Controller and an IDP, including the following:
(With stand-alone IDP) Global configuration parameters such as the IDP hostname or IP address, the TCP port over which the sensor communicates with the Infranet Controller, and the one-time password the Infranet Controller and IDP use to authenticate with one another

Various levels of attack severity warnings and the action that the Infranet Controller takes

The IDP sits within your network and monitors traffic from endpoints that are connected through the Infranet Controller. You can position the IDP in-line, or you can configure the IDP in sniffer mode.

Once the Infranet Controller has connected with the IDP sensor, the Infranet Controller registers all of the IP addresses that should be monitored for potential threats. With stand alone IDP, you enter the IP addresses that you wish to monitor.

Any abnormal events detected by the IDP Sensor are reported to the Infranet Controller, which you configure to take appropriate action based on the severity level of the reported events. The IDP Sensor performs reporting functions to allow you to determine what IP address within the network has launched the attacks in addition to any normal logging the IDP has been configured to undertake.

With IDP deployments using the Infranet Enforcer and the IDP Security Module, the Infranet Enforcer can send messages to the Odyssey Access Client debug log.

Licensing: IDP Availability

You must obtain a separate license to use both stand alone IDP and ISG-IDP. With the proper licensing, a Sensors tab with two additional page options, Sensors and Sensor Event Policies are visible in the Configuration section of the admin console.

If you do not have a license for IDP installed, the UI elements are not visible.

Deployment Scenarios

Two possible deployment scenarios are shown in Figure 27 and Figure 28. In Figure 27 the stand-alone IDP is located within the internal network. All network traffic originating from endpoints that are registered with the IDP is monitored. You can deploy IDP in sniffer mode, or in-line mode. You can use transparent mode or route mode with an in-line mode configuration.

In the first deployment example, the IDP does not monitor IPSec traffic from the user to protected resources. If you want to monitor all IPSec traffic from users to protected resources, deploy the IDP behind the Infranet Enforcer as shown in Figure 28.
Figure 27: Infranet Controller and stand-alone IDP topology

Figure 28: Infranet Controller and ISG-IDP topology
Configuring the Infranet Controller to Interoperate with IDP

The IDP Sensor is a powerful tool to counter users who initiate attacks. Integration with the Infranet Controller allows you to configure automatic responses as well as manually monitor and manage users.

To configure the Infranet Controller to interoperate with an associated stand-alone IDP Sensor, you must first ensure the IDP has been configured according to the instructions described in the “Infranet Controller Signaling Setup” appendix of the Intrusion Detection and Prevention Concepts & Examples Guide.

Once the IDP Sensor has been set up, you can specify the events you want the IDP to watch for and the actions that the Infranet Controller takes once a particular event has been noted and reported.

There are two locations on the Infranet Controller where you can specify actions to be taken in response to users that perform attacks:

- **Sensor Event policies page**—Define the policy on this page to generate an automatic response to users who perform attacks.

- **Users page**—Manually identify and quarantine or disable users on the System > Status > Active Users page, which lists users who have performed attacks. For more details, see “Identifying and Managing Quarantined Users Manually” on page 493.

**Using ISG-IDP**

If you are using integrated IDP with the ISG-1000 or ISG-2000, see Concepts and Examples ScreenOS Reference Guide: Vol 4. Attack Detection and Defense Mechanisms.

When ISG-IDP is activated, ScreenOS notifies the Infranet Controller when an attack event is detected from any endpoint. To avoid overwhelming the SSH connection between the Infranet Controller and the Infranet Enforcer, the number of attack notifications is limited to 10 per second. If additional attacks are detected, the Infranet Enforcer holds an additional ten notifications in a queue.

ISG-IDP devices attached to any node in a cluster may send messages regarding sessions attached to any node in the cluster.

If you have installed the IDP license, there is a Use IDP module as Sensor check box on the Infranet Enforcer admin console page. If you select the check box and there is no IDP module or if the Infranet Enforcer is not running ScreenOS version 6.2 or greater the Infranet Controller logs an appropriate message.

To activate ISG-IDP on the Infranet Controller:

1. Select UAC > Infranet Enforcer in the Infranet Controller admin console.
2. Select the name of the Enforcer on which you want to activate IDP.
3. Select the **Use IDP Module as Sensor** check box.
4. Select 1 - INFO through 5 - Critical from the Severity drop-down menu. The severity filter allows you to specify the level of attacks that the Infranet Enforcer reports to the Infranet Controller. For example, if you select 3, only level 3 attacks or greater will be reported.

**NOTE:** With ISG-IDP you do not need to specify which IP addresses to monitor. The Infranet Enforcer monitors all IP address for which there are auth tables.

**Configuring Stand-alone IDP Connections**

The Sensors tab allows you to specify the system settings the Infranet Controller uses to establish a connection to a Juniper Network’s Intrusion Detection and Prevention (IDP) device.

Use the System > Configuration > Sensors > Sensors tab to perform a number of tasks related to configuring and managing interaction between the Infranet Controller and an IDP Sensor.

**Creating a New IDP Sensor Entry**

To enable or disable existing IDP Sensor entries on the Infranet Controller:

1. In the admin console, choose System > Configuration > Sensors.

2. Click New Sensor. The admin console displays the New Sensor page.

3. Under Sensor Properties, specify the following information:

   - **Name**—A name the Infranet Controller uses to identify the new connection entry
   - **Hostname**—The hostname or IP address of the IDP Sensor to which the Infranet Controller connects in order to receive application and resource attack alert messages
   - **Port**—The TCP port on the IDP Sensor to which the Infranet Controller listens when receiving application and resource attack alert messages
   - **One-time password**—The encrypted password the Infranet Controller uses when conducting the initial Transport Layer Security (TLS) handshake with the IDP Sensor. You must enter the encrypted Infranet Controller OTP password as displayed on the IDP ACM configuration summary screen.

**NOTE:** The hostname, TCP port, and one-time password must already be configured on the IDP Sensor before this configuration can be successful.

4. Under Monitoring Options, specify IP addresses to monitor and the minimum alert severity level the IDP Sensor will record and submit to the Infranet Controller:
a. In the **Addresses to monitor** field, specify individual IP addresses and address ranges the IDP Sensor monitors for potential attacks, one entry per line. IDP reports attack information only for the IP addresses that you specify.

b. Select one of the options available in the **Severity filter** drop down list. The severity level is a number on a scale from 1 to 5, where 1 is informational and 5 is critical.

5. Click **Save Changes**.

**Enabling or Disabling the Connection to an Existing IDP Sensor**

To enable or disable existing IDP Sensor entries on the Infranet Controller:

1. In the admin console, choose **System > Configuration > Sensors**.
2. Select the checkbox next to one or more IDP Sensor entries you want to enable or disable.
3. Click **Enable** or **Disable** to enable or disable the specified IDP Sensor entries, respectively.

**Deleting an IDP Sensor Entry**

You can delete existing IDP Sensor entries that define a connection between the Infranet Controller and an IDP Sensor.

To delete one or more existing IDP Sensor entries from the Infranet Controller:

1. In the admin console, choose **System > Configuration > Sensors**.
2. Select the checkbox next to the IDP Sensor entry or entries you want to delete.
3. Click **Delete** and then confirm that you want to delete the sensor entry or entries.

**Reconnecting to an IDP and Refreshing IDP Connection Status**

When the connection to an IDP Sensor is down, you can use the admin console on the Infranet Controller to try and re-establish the connection. You can also use the admin console to refresh the status of existing connections between the Infranet Controller and the IDP Sensor.

If you need to re-establish communication with an IDP Sensor, you must generate a new One-time Password, as described in “Creating a New IDP Sensor Entry” on page 489.

To reconnect to an associated IDP Sensor:

1. In the admin console, choose **System > Configuration > Sensors**.
2. Select the checkbox next to the IDP Sensor to which you want to reconnect.
3. Click **Reconnect**.
The admin console displays a message informing you that the Infranet Controller is currently attempting to re-establish connection to the specified IDP Sensor. This page automatically refreshes each second during the reconnection process. Otherwise, the connection status page automatically refreshes once every 30 seconds.

To refresh and display the connection status for the specified IDP Sensor:

1. In the admin console, choose **System > Configuration > Sensors**.
2. Select the checkbox next to one or more IDP Sensor entries for which you want to display current connection status.
3. Click **Refresh**.

**Interaction Between the Infranet Controller and IDP**

The Infranet Controller reads attack information as it is being sent by the IDP sensor. After receiving a block of attack information from the IDP sensor, the Infranet Controller sends a simple acknowledgement to the sensor indicating receipt of the information.

The Infranet Controller receives the source and destination IP addresses and port numbers of the attacking host and the resource against which the attack was launched, along with the attack identifier, severity of the attack, and the time at which the attack was launched.

The Infranet Controller incorporates and displays the attack information received from the IDP sensor on the **System > Status > Active Users** page. Based on the attackers IP address and port number, the Infranet Controller will uniquely identify the user’s session.

You can choose automatic or manual actions for attacks detected by the IDP sensor. For manual action, you look up the information available on the Active Users page and decide on an action. For automatic action, you configure the action in advance when you define your IDP policies.

When you learn that an attack has been launched by an active user, you can disable the user’s account, end the user’s session, or remediate to a different role.

The Infranet Controller displays an error message to the user whose account has been disabled indicating the reason.

**Defining Automatic Response Sensor Event Policies**

Use the **System > Configuration > Sensors > Sensor Event Policies** tab to specify one or more rules that specify the action(s) the Infranet Controller takes when it receives attack alert messages from an IDP Sensor.

To create a new IDP rule:

1. In the admin console, choose **System > Configuration > Sensors > Sensor Event Policies**.
2. On the **Sensor Event Policies** page, click **New Rules**.
3. On the Juniper IDP Rule page, in the Rule: On Receiving... section:
   - Select an existing event from the Event drop-down list.
   - Click Events to edit an existing event or create a new type of event and add it to the options in the Events drop-down list:
     i. Specify a name for the event.
     ii. Populate the Expressions field by manually entering expressions or by selecting one or more clauses from the Expressions Dictionary and clicking Insert Expression.

     For example, to check for all critical/highest severity level attacks, enter the following expression:
     ```
     idp.severity >= 4
     ```

     To check for all critical/highest severity level attacks for HTTP traffic, enter the following expression:
     ```
     idp.severity >= 4 AND idp.attackStr = "*HTTP*"
     ```

     For more information on building IDP policies, refer to:
     

     iii. When you have finished entering the expressions you want to apply to this event, click Add Expression.

     iv. Click Close.

4. In the Count this many times section, specify a number between 1 and 256 to determine the number of times an event must occur before action is taken.

5. In the ...then perform this action section, specify one of the following actions:
   - **Ignore** (just log the event)—Specifies that the Infranet Controller should log the event, but take no further action against the user profile to which this rule applies. This option is best used to deal with very minor “informational” attack alert messages that come from the IDP Sensor.
   - **Terminate User Session**—Specifies that the Infranet Controller should immediately terminate the user session and require the user to sign in to the Infranet Controller again.
   - **Disable user account**—Specifies that the Infranet Controller should disable the user profile associated with this attack alert message, thus rendering the client unable to sign in to the Infranet Controller until the administrator re-enables the user account. (This option is only applicable for users who have a local Infranet Controller user account.)
• **Replace user’s role with this one**—Specifies that the role applied to this user's profile should change to the role you select from the associated drop-down list. This new role remains assigned to the user profile until the session terminates. This feature allows you to assign a user to a specific controlled role of your choice, based on specific IDP events. For example, if the user performs attacks, you might assign the user to a restricted role that limits the user’s access and activities.

• Choose to **make this role assignment**:
  - **Permanent**—User remains in the quarantined state across subsequent logins until the administrator releases the user from the quarantined state.
  - **For this session only**—Default. User can log in to another session.

6. In the **Roles** section, specify:

- **Policy applies to ALL roles**—To apply this policy to all users.

- **Policy applies to SELECTED roles**—To apply this policy only to users who are mapped to roles in the **Selected roles** list. Make sure to add roles to this list from the **Available roles** list.

- **Policy applies to all roles OTHER THAN those selected below**—To apply this policy to all users except for those who are mapped to the roles in the **Selected roles** list. Make sure to add roles to this list from the **Available roles** list.

7. Click **Save Changes**.

**Identifying and Managing Quarantined Users Manually**

When the Infranet Controller quarantines a user based on an attack, you can display and manage the states by locating the user link in the **System > Status > Active Users** page.

You can identify quarantined users based on several elements:

- A small warning icon displayed in front of the user name.
- The hyperlinked user name.
- An enabled Quarantined radio button on the specific user’s page. If the user is not quarantined, the radio button is disabled.

To manage quarantined users:

1. Identify quarantined users at **System > Status > Active Users**.

2. Locate the quarantined user and click on the username link. The user page opens, showing a number of options.

3. Click **Disabled** to disallow a user from authenticating.
4. Click **Quarantined** to leave a user in a quarantined state. The Quarantined option is only enabled if the user is already quarantined.

**NOTE:** The Infranet Controller assigns quarantined users to the quarantined role, regardless of their login realm.

5. Click **Save Changes**.

6. To re-enable previously quarantined or disabled users, select **Authentication** > **Auth. Servers** > **Select Server** > **Users** and click the link for the given user.

**NOTE:** You can also disable users from this location.

7. Click **Enabled** to release the user from quarantine.

8. Click **Save Changes**.

If you want to isolate quarantined users automatically, follow the steps as described in “Defining Automatic Response Sensor Event Policies” on page 491.

**NOTE:** All Sensor events are logged at **System** > **Log/Monitoring** > **Sensors** > **Log**. For more information, see “Logging and Monitoring” on page 415.
This section contains the following information about Infranet Controller system services:

- “Infranet Controller Serial Console” on page 497
- “Customizable Admin and End-User UIs” on page 503
- “Infranet Controller 4500 and 6500” on page 505
- “Compression” on page 513
Connecting to an Infranet Controller Appliance’s Serial Console

Before performing any tasks through an Infranet Controller appliance’s serial console, you need to connect to the console using a terminal console or laptop.

To connect to an Infranet Controller appliance’s serial console:

1. Plug a null modem crossover cable from a console terminal or laptop into the Infranet Controller appliance. This cable is provided in the product box. Do not use a straight serial cable.

2. Configure a terminal emulation utility, such as HyperTerminal, to use these serial connection parameters:
   - 9600 bits per second
   - 8-bit No Parity (8N1)
   - 1 Stop Bit
   - No flow control
3. Press **Enter** until the Infranet Controller serial console appears.

**Figure 29: Infranet Controller Serial Console**

---

**Rolling Back to a Previous System State**

An Infranet Controller appliance stores current system configuration information and that of the previous state.

---

**NOTE:** You may also roll back to a previous system state through the admin console, as described in “Installing a Juniper Software Service Package” on page 352.

---

**Rolling Back to a Previous System State Through the Admin Console**

If you upgrade your server package and decide you would like to revert to the previous state of your machine, we recommend that you perform the following steps from within the admin console:

1. Locate previously exported system and user configuration files that store the desired state data. (This step presumes that you backed up your system and user data by exporting files through the admin console’s **Maintenance > Import/Export** menu.)

2. Download the desired Infranet Controller OS service package from the Juniper Networks Support Customer Support Center.

3. Import the chosen Infranet Controller OS service package through the admin console’s **Maintenance > System > Upgrade/Downgrade** menu.

4. Import the system and user configuration files you locate in the beginning of this section.
**Rolling Back to a Previous System State Through the Serial Console**

If you cannot access the admin console, connect to the serial console to perform a system rollback to the previous system state.

NOTE: If you have not yet performed an Infranet Controller OS service package upgrade, there is no previous state to roll back to and this option is not available. If you have performed an Infranet Controller OS service package upgrade, any system and user configuration data created after the upgrade is lost unless you export the most current configuration files before rolling back the system and then import them afterwards.

To roll back to the previous Infranet Controller OS service package:

1. Connect to your Infranet Controller appliance’s serial console. For instructions, see “Connecting to an Infranet Controller Appliance’s Serial Console” on page 497.

2. In a browser window, sign in to the admin console.


4. Click Reboot Now and then go back to the console utility window. The window displays a message that the system is restarting.

5. After several moments, you are prompted to hit the Tab key for options. Press the Tab key, and when prompted for the configuration to load, type rollback and then press the Enter key.

After clicking Reboot Now on the Maintenance > System > Platform page, the server’s rollback status is output to the screen, and when complete, you are prompted to hit the Return key (Enter) to modify system settings, which returns you to the initial setup options. When you are finished entering data, simply close the utility window.

NOTE: If you wait more than 5 seconds to enter your choice, the current system configuration is automatically loaded and you’ll need to go back to the admin console and click Reboot Now to start the process again. If you have already performed a system rollback, the rollback option is not available again until you upgrade the Infranet Controller OS service package again.

**Resetting an Infranet Controller Appliance to the Factory Setting**

In rare cases, you may need to reset your Infranet Controller appliance to its original factory settings. Before performing this advanced system recovery option, please contact Juniper (http://www.juniper.net/support/). If possible, export the most current system and user configuration data before performing a factory reset.

To perform a factory-reset:

1. Connect to the serial console. For instructions, see “Connecting to an Infranet Controller Appliance’s Serial Console” on page 497.
2. In a browser window, sign in to the admin console.

3. Choose **Maintenance > System > Platform**.

4. Click **Reboot** and then go back to the console utility window. The window displays a message that the system is restarting.

5. After several moments, you are prompted to hit the **Tab** key for options. Press the **Tab** key, and when prompted for the configuration to load, type **factory-reset** and then press the **Enter** key.

```
NOTE: If you wait more than 5 seconds to enter your choice, the current system configuration is automatically loaded and you’ll need to go back to the admin console and click **Reboot Now** to start the process again.
```

6. When you are prompted to confirm performing a factory-reset, type **proceed** and then press **Enter**.

   The system begins the process of resetting the machine to its original settings and outputs several screens of data. After several minutes, you are prompted to hit the **Tab** key to choose configuration choices.

7. When prompted to hit the **Tab** key, either:

   - Wait for the default selection (current) to automatically start, or
   - Press **Tab**, type **current**, and then press **Enter**.

   You are then prompted to enter the initial machine configuration settings. For details on how to proceed, see the **Installation Guide** provided in the product packaging or on the Juniper Networks Support site.
Performing Common Recovery Tasks

If you forget your Infranet Controller administrator username and/or password, lock yourself out of your machine due to configuration errors, or change the Infranet Controller appliance IP address and can no longer reach the machine, you can modify the machine settings through the serial console. Follow the instructions under “Connecting to an Infranet Controller Appliance’s Serial Console” on page 497 and then choose the appropriate configuration task.

- **Network Settings and Tools**—Enables you to change standard network settings; print a routing table; print or clear an ARP cache; ping another server, trace a route to a server, remove static routes, and add an ARP entry.

- **Create admin username and password**—Enables you to create a new super-administrator account.

- **Display log**—Enables you to display system configuration, user logs, or administrator access logs through the serial console. Note that must enter “q” to return to serial console options after viewing the logs.

- **System Operations**—Enables you to reboot, shutdown, restart, rollback, or factory reset the Infranet Controller appliance without using the admin console.

- **Toggle password protection for the console**—Enables you to password protect the serial console. When you toggle this option to “on,” only super-administrators are allowed access.

**NOTE:** You might receive errors from the Infranet Controller during the initial setup or on a factory reset. Before the Infranet Controller starts services it monitors the network port for a maximum of 120 seconds. The Infranet Controller checks the link status and performs an ARPing on the default gateway. If there is a problem, after 5 seconds, the Infranet Controller displays a message on the serial console that starts with **NIC:**. If the link recovers within 120 seconds, the startup process continues. If the link does not recover, the following message appears:

```
Internal NIC: ...............[Down code=0x1]
```

Two codes can appear:

- **0x1** means that the interface link status reported by the NIC remains off (for example, a disconnected cable or a cable in the wrong port).

- **0x2** means that the gateway is unreachable. The Infranet Controller boots but is not reachable from IP addresses bound to that network port.
Create a Super Admin session—Enables you to create a recovery session to the admin console, even if you have configured the Infranet Controller appliance to block access to all administrators. When you select this option, the appliance generates a temporary token that is valid for 3 minutes. Enter the following URL into a browser window:

https://<Infranet Controller-host>/dana-na/auth/recover.cgi

Then, enter the temporary token when prompted in order to sign into the admin console.

**NOTE:** When you choose this option, the Infranet Controller appliance blocks any additional administrators from signing in to the admin console until you sign in to the specified URL and initiate a session using your token. The appliance blocks additional sign-in attempts so that you can fix any configuration problems that the Infranet Controller may have encountered without conflicting with another session.

System Snapshot—Enables you to take a system snapshot without using the admin console. When you choose this option, the Infranet Controller takes the snapshot immediately. You can then send the snapshot file, by way of SCP, to a remote system. The system prompts you for the destination server port, user ID, password, and the destination path to the remote directory.

**NOTE:** If you choose not to send the snapshot file to a remote system, the Infranet Controller saves the file locally. The next time you log in to the admin console, the System Snapshot tab contains a link to the snapshot file. For more information about taking a snapshot from the admin console, see “Creating Snapshots of the Infranet Controller System State” on page 439.
Chapter 19

Customizable Admin and End-User UIs

The Infranet Controller enables you to customize a variety of elements in both the admin console and the end-user interface. This section contains the following information about which elements you can customize and where you can find the appropriate configuration options:

- “Customizable Admin Console Elements Overview” on page 503
- “Customizable End-User Interface Elements Overview” on page 504

Customizable Admin Console Elements Overview

The Infranet Controller enables you to customize the look and feel of the following user interface elements in the admin console:

- **Sign-in pages (default and custom)**—You can customize the page that administrators see when they sign into the admin console using settings in the Authentication > Signing In > Sign-in Pages page. Using settings in this page, you can create welcome messages, sign out messages and other instructions; control page headers; customize select error messages; and create a link to a custom help page within the default Infranet Controller sign-in page. For instructions, see “Configuring Standard Sign-In Pages” on page 266. Or, you can upload your own custom sign-in page to the Infranet Controller. For more information, see the Custom Sign-In Pages Solution Guide.

- **UI look and feel**—You can customize the header, background color, and logo displayed in the admin console using settings in the Administrators > Admin Roles > Select Role > General > UI Options page. You can also use settings in this page to enable or disable the “fly out” hierarchical menus that appear when you mouse over one of the menus in the left panel of the admin console. For instructions, see “Specifying UI Options” on page 483.

- **System utilization graphs**—You can choose which system utilization graphs the Infranet Controller displays on the opening page of the admin console using settings in the System > Status > Overview page. You can also use settings in this page to fine-tune the look and data within each of the graphs. For instructions, see “Viewing General Status” on page 431.
- **User realm views**—You can use customization options on the Users > User Realms page to quickly view the settings that are associated with a specific user realm or set of user realms. For instructions, see “Customizing User Realm UI Views” on page 254.

- **Administrator roles**—You can delegate select responsibilities to other administrators using settings in the Administrators > Admin Roles section of the admin console. In doing so, you can restrict the visibility of certain options and capabilities to those other administrators. For instructions, see “Creating and Configuring Administrator Roles” on page 474.

---

**Customizable End-User Interface Elements Overview**

You can customize the sign-in page that users see when they sign into the Infranet Controller using settings in the Authentication > Signing In > Sign-in Pages page. Using settings in this page, you can create welcome messages, sign out messages and other instructions; control page headers; customize select error messages; and create a link to a custom help page within the default Infranet Controller sign-in page. For instructions, see “Configuring Standard Sign-In Pages” on page 266. Or, you can upload your own custom sign-in page to the Infranet Controller. For instructions, see the Custom Sign-In Pages Solution Guide.
Chapter 20

Infranet Controller 4500 and 6500

The Infranet Controller 4500 and 6500 (IC 4500/6500) are next-generation appliances featuring a number of notable hardware features.

NOTE: The IC400 and IC6000 are still available for purchase and stocking. See “IC6000 Standard Hardware” on page 511.

Standard Hardware

The IC 4500/6500 chassis features the following hardware components:

- **Console port**—You can use the console port to initially set up the IC 4500/6500 before you fully integrate it as the secure gateway to your internal network. You can also use the console port to perform certain configuration and clustering tasks after the Infranet Controller begins operating.

- **Internal and external Ethernet ports**—The IC 4500/6500 primary connections to the corporate network and the outside world are the internal and external Ethernet ports. You can configure the internal and external interfaces by using the System > Network page of the administrator admin console.

The IC4500 supports a two-node active/active or active/passive cluster. The IC6500 supports up to four-node clusters.

Infranet Controller 6500 Field-Replaceable Units

The IC6500 chassis features three types of field-replaceable units (FRUs) that you can add or replace. The FRUs are “hot-swappable,” meaning you do not have to first shut down the IC6500 before adding or replacing any of the FRUs. The IC4500 has a “cold-swappable” power supply.

For safety information, refer to the Juniper Networks Products Safety Guide available on the Juniper Networks Support site.
- **Hard disks**—The IC6500 ships with one hard disk, however, you can add an optional second hard disk to the IC6500 chassis to offer component redundancy and help minimize Infranet Controller down time. When a second (redundant) hard disk is installed, it maintains an exact copy of the software image and configuration information on the working hard disk. Therefore, if the working hard disk fails, the redundant hard disk immediately assumes responsibility for all Infranet Controller operations. This function is referred to as the Redundant Array of Independent Disks (RAID) mirroring process.

---

**NOTE:** The IC6500 hard disk modules are hot-swappable. You must make sure that the Infranet Controller finishes booting and is operating correctly before removing, replacing, or upgrading a hard disk module. After you insert a new hard disk module, you must wait until the RAID mirroring process is completely finished—which takes approximately 40 minutes—before rebooting or turning off the Infranet Controller.

- **Power supplies**—The IC6500 ships with one AC power supply installed in the back of the chassis. You can add an optional second power supply to support redundancy and load-sharing features. In addition, if you need to replace one of the power supplies, you can “swap” the faulty power supply for a replacement while the optional second power supply assumes responsibility for the entire power load, thus avoiding a situation where you have to power off the Infranet Controller before replacing the removable unit.

- **Cooling fans**—The IC6500 ships with two cooling fans installed in the back of the chassis. If you need to replace one of the cooling fans, you can “swap” the faulty fan for a replacement during operation in a matter of moments. You can purchase additional cooling fans from your vendor when you order your IC6500, or you can purchase them in the future to replace faulty or failed cooling fans, as necessary, in the future.

---

**Device Status LED Behavior**

Startup takes approximately one minute to complete. If you want to turn the device off and on again, we recommend you wait a few seconds between shutting it down and powering it back up.

There are three device status LEDs located on the left-side of the front panel:

- **Power**
- **Hard disk access**
- **Fault**

Table 41 on page 507 lists the name, color, status, and description of each device status LED.
Replacing the Cooling Fans

The IC6500 ships with two cooling fans installed in the back of the chassis. If you need to replace one of the cooling fans, you can “hot-swap” the faulty fan for a replacement during operation in a matter of moments. You can purchase additional cooling fans from your authorized Juniper reseller, or you can purchase them in the future to replace faulty or failed cooling fans, as necessary.

Removing and Installing a Cooling Fan

To remove a cooling fan module:

1. To release the cooling fan module, do one of the following:
   - Press and slide the release trigger toward the center of the cooling fan module
   - Loosen the thumbscrews

Table 41: Device Status LEDs

<table>
<thead>
<tr>
<th>Name</th>
<th>Color</th>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POWER</td>
<td>Green</td>
<td>Off</td>
<td>Device is not receiving power</td>
</tr>
<tr>
<td></td>
<td></td>
<td>On Steady</td>
<td>Device is receiving power</td>
</tr>
<tr>
<td>HARD DISK ACCESS</td>
<td>Yellow</td>
<td>Off</td>
<td>Hard disk is idle</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Blinking</td>
<td>Hard disk is being accessed</td>
</tr>
<tr>
<td>FAULT</td>
<td>Red</td>
<td>Off</td>
<td>Device is operating normally</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Slow blinking</td>
<td>Power supply fault</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fast blinking</td>
<td>Fan failure</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Solid</td>
<td>Thermal failure</td>
</tr>
</tbody>
</table>

Table 42: 4-Port Copper Gigabit Ethernet LEDs (available on IC4500 and IC6500)

<table>
<thead>
<tr>
<th>LED</th>
<th>Color and State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Link/Activity</td>
<td>Green</td>
<td>Link</td>
</tr>
<tr>
<td></td>
<td>Blinking green</td>
<td>Activity</td>
</tr>
<tr>
<td>Link Speed</td>
<td>Off</td>
<td>10 Mbps</td>
</tr>
<tr>
<td></td>
<td>Green</td>
<td>100 Mbps</td>
</tr>
<tr>
<td></td>
<td>Yellow</td>
<td>1 Gbps</td>
</tr>
</tbody>
</table>
2. Grasp the cooling fan module and carefully pull it out.

**CAUTION:** Once you remove the cooling fan module, it is important that you replace it with a replacement cooling fan. The second fan is required for proper air flow across the chassis’s internal components; it is not a redundant fan.

To install a cooling fan module:

1. Line the cooling fan module up with an empty cooling fan port on the back of the chassis.
2. Slowly slide the module into the chassis until it clicks into place.
3. If your cooling fan is equipped with thumb screws, tighten the screws.

---

### Replacing a Hard Drive

The IC6500 ships with two standard hard drives to offer component redundancy and help minimize down time. The second (redundant) hard disk maintains an exact copy of the software image and configuration information on the working hard disk. Therefore, if the working hard disk fails, the redundant hard disk immediately assumes responsibility for all operations. This function is referred to as the Redundant Array of Independent Disks (RAID) mirroring process.

**NOTE:** The hard disk modules are hot-swappable. Once a new hard disk module is inserted, you should wait until the RAID mirroring process has completed before rebooting or turning off the appliance.

---

### Removing and Installing a Hard Drive

To remove a hard drive:

1. On the hard drive module, press the blue handle release trigger in and to the right to release the insertion and removal handle.
2. Grasp the handle and pull the hard drive module straight out of the chassis.

Once you have removed the hard drive module, be sure to replace it with a replacement hard drive.

To install a hard drive:

1. With the insertion and removal handle on the hard drive module in the released/out position, line the hard drive module up with an empty hard drive port on the front of the chassis.
2. Carefully slide the hard drive module into the chassis until it is clicks into place.

Retract the handle by swinging it back across the face of the hard drive until it is completely flush with the face of the hard drive module.
Replacing IOC Modules

Removing a Blank IOM Faceplate

To maintain proper airflow through the device, leave blank faceplates in place over slots that do not contain IOMs. Do not remove a blank faceplate unless you are installing an IOM in the empty slot.

To remove a blank faceplate:

1. Unplug the power cord.
2. Loosen the thumbscrews on each side of the faceplate.
3. Grasp the thumbscrews and pull to remove the faceplate.

Installing an IOM

To install an IOM:

1. Unplug the power cord.
2. Line the IOM up with an empty port on the front of the chassis.
3. Carefully slide the IOM in until it seats firmly in the device.
4. Tighten the screws on each side of the IOM faceplate.
5. Insert the appropriate cables into the cable connectors on the IOM.
6. If necessary, arrange the cables to prevent them from dislodging or developing stress points:
   - Secure the cable so that it is not supporting its own weight as it hangs to the floor.
   - Place excess cable out of the way in a neatly coiled loop.
   - Use fasteners to maintain the shape of cable loops.
7. Insert the power cord into the AC power receptacle.

CAUTION: Power off the device before removing or installing IOMs. IOMs are not hot-swappable.
Removing an IOM

To remove an IOM:

1. Unplug the power cord.
2. Disconnect the cables from the IOM.
3. If necessary, arrange the cables to prevent them from dislodging or developing stress points.
4. Loosen the thumb screws on each side of the IOM faceplate.
5. Grasp the thumbscrews and pull to remove the IOM.

If you are not reinstalling an IOM into the empty slot, install a blank IOM faceplate over the empty slot to maintain proper airflow.

Replacing an AC Power Supply

Removing and Installing an AC Power Supply

The Juniper Networks appliance ships with one AC power supply installed in the back of the chassis. You can add an optional second power supply to support redundancy and load-sharing features. In addition, if you need to replace one of the power supplies, you can “hot-swap” the faulty power supply for a replacement while the optional second power supply assumes responsibility for the entire power load, thus avoiding a situation where you have to power off the Infranet Controller before replacing the removable unit.

To remove an AC power supply module:

1. Press the release trigger in and to the right to release the module.
2. Grasp the insertion and removal handle and pull the power supply module straight out of the chassis.

Once you have removed the supply module, be sure to replace it with a replacement power supply or the “dummy” power supply port cover installed in your chassis at the time of shipping.

To install an AC power supply module:

1. Line the power supply module up with an empty power supply port on the back of the chassis.
2. Slowly slide the power supply module into the chassis until it clicks into place.

Removing and Installing a DC Power Supply

To remove a DC power supply module:

1. Unplug the power cord.
2. Disconnect the DC supply wires from the lugs on the DC power supply.
3. Press the release trigger in and to the right to release the module.
4. Grasp the power supply module and pull it straight out of the chassis.

To install a power supply module:
1. Slowly slide the module into the chassis until it clicks into place.
2. Connect the DC supply wires to the module using the lugs. Be sure to attach the ground wire.
3. Attach the power cord.

IC6000 Standard Hardware

The IC 6000 chassis features the following hardware components:

- **Console port**—You can use the console port to initially set up the IC 6000 before you fully integrate it as the secure gateway to your internal network. You can also use the console port to perform certain configuration and clustering tasks after the Infranet Controller begins operating.

- **Internal and external Ethernet ports**—The IC 6000's primary connections to the corporate network and the outside world are the internal and external Ethernet ports. You can configure the internal and external interfaces by using the System > Network page of the administrator admin console. The rightmost LED next to the port indicates network activity. The left-most LED next to the port indicates three possible states:
  - **Off**—no network connection
  - **Green**—100 MHz connection
  - **Orange**—1 GHz connection) to indicate the speed of the connection

- **Status LEDs**—The front of the IC 6000 chassis features the following LEDs:
  - **PWR** (green)—Indicates that the appliance has power and is turned on
  - **HD** (amber)—Indicates that the hard disk is in use (writing or reading data)
  - **TEMP** (red)—A blinking LED indicates that one of the fans has failed or is not seated properly in its port, or that a fan has failed and needs to be replaced. A solid LED indicates a high internal temperature reading that may result in system failure if not addressed.
  - **PS FAIL** (red)—Indicates that one of the power supplies is faulty, has been unplugged, or has experienced an outright failure
For information about installing or replacing any of the hardware mentioned here, see the Infranet Controller 6000 Field Replaceable Units Removal and Installation Guide on the Juniper Networks Support Site.

Infranet Controller 6000 field-replaceable units

The IC 6000 chassis features three types of field-replaceable units (FRUs) that you can add or replace. The FRUs are “hot-swappable,” meaning you do not have to first shut down the IC 6000 before adding or replacing any of the FRUs.

- **Hard disks**—The IC 6000 ships with one hard disk, however, you can add an optional second hard disk to the IC 6000 chassis to offer component redundancy and help minimize Infranet Controller down time. When a second (redundant) hard disk is installed, it maintains an exact copy of the software image and configuration information on the working hard disk. Therefore, if the working hard disk fails, the redundant hard disk immediately assumes responsibility for all Infranet Controller operations. This function is referred to as the Redundant Array of Independent Disks (RAID) mirroring process.

  **NOTE:** The IC 6000 hard disk modules are hot-swappable. You must make sure that the Infranet Controller finishes booting and is operating correctly before removing, replacing, or upgrading a hard disk module. After you insert a new hard disk module, you must wait until the RAID mirroring process is completely finished—which takes approximately 40 minutes—before rebooting or turning off the Infranet Controller.

- **Power supplies**—The IC 6000 ships with one AC power supply installed in the back of the chassis. You can add an optional second power supply to support redundancy and load-sharing features. In addition, if you need to replace one of the power supplies, you can “swap” the faulty power supply for a replacement while the optional second power supply assumes responsibility for the entire power load, thus avoiding a situation where you have to power off the Infranet Controller before replacing the removable unit.

- **Cooling fans**—The IC 6000 ships with two cooling fans installed in the back of the chassis. If you need to replace one of the cooling fans, you can “swap” the faulty fan for a replacement during operation in a matter of moments. You can purchase additional cooling fans from your vendor when you order your IC 6000, or you can purchase them in the future to replace faulty or failed cooling fans, as necessary, in the future.
Chapter 21

Compression

The Infranet Controller improves performance by using HTTP compression when you log into the admin console and when users log in using agentless access. This section contains the following information about compression:

- “Compression Execution” on page 513
- “Supported Data Types” on page 514
- “Enabling Compression at the System Level” on page 514

Compression Execution

The Infranet Controller determines whether it should compress the data accessed by users by using the following process:

1. The Infranet Controller verifies that the accessed data is a compressible type. The Infranet Controller supports compressing many common data types such as such as HTML files. For a complete list, see “Supported Data Types” on page 514.

2. The Infranet Controller verifies that the user’s browser supports compression of the data type.

The Infranet Controller determines compression supportability based on the browser’s user-agent and the accept-encoding header. The Infranet Controller supports the compression of all of the standard Web data types if it determines that the user-agent is compatible with Mozilla 5, Internet Explorer 5, or Internet Explorer 6. The Infranet Controller only supports compressing HTML data, however, if it determines that the browser’s user-agent is only compatible with Mozilla 4.

3. The Infranet Controller verifies that compression is enabled at the system level. You can enable system-level compression through the Maintenance > System > Options page of the admin console, as explained in “Enabling Compression at the System Level” on page 514.
Supported Data Types

The Infranet Controller supports compressing the following types of files:

- text/html (.html, .htm)
- application/x-javascript (.js)
- text/javascript (.js)
- text/css (.css)
- application/perl (.cgi)

Enabling Compression at the System Level

To enable system-level compression:

1. In the admin console, choose **Maintenance > System > Options**.
2. Select the **Enable gzip compression** checkbox to reduce the amount of data sent to browsers that support HTTP compression.
3. Click **Save Changes**.
Part 6
Supplemental Information

This section contains the following supplemental topics:

- “Writing Custom Expressions” on page 517

You can also find supplemental information about the Infranet Controller on the Juniper Networks Customer Support Center. Supplemental documents on the support center include:

- Unified Access Control Administration Guide—You can download a PDF version of this guide.

- Unified Access Control Quick Start Guide—Describes the basic tasks of configuring the Infranet Controller and Infranet Enforcer for the Unified Access Control solution.

- Unified Access Control Client-side Changes Guide—Describes the changes that Odyssey Access Client and Infranet Controller clients make on client computers, including installed files and registry changes, and information about the rights required to install and run the clients.

- Unified Access Control Custom Sign-In Pages Solution Guide—Describes how to personalize the look-and-feel of the pre-authentication and password management pages that the Infranet Controller displays to end-users and administrators.

- Unified Access Control Deployment Scenarios Guide—Describes recommendations for deploying five example scenarios of the Unified Access Control solution.

- Infranet Controller Installation Guide—Describes how to install the Infranet Controller.

- Infranet Controller 6000 Field Replaceable Units Guide—Describes how to install hard disks, power supplies, and cooling fans on the Infranet Controller.

- Odyssey Access Client Quick Start Guide—Describes the basic tasks of configuring and using Odyssey Access Client.


You can also obtain Odyssey Access Client software and documentation on the Odyssey Access Client User page of the *Juniper Networks Customer Support Center*. 
Appendix A
Writing Custom Expressions

This section contains the following topics:

- “Custom Expressions” on page 517
- “System Variables and Examples” on page 521

Custom Expressions

You can write custom expressions that are evaluated in role mapping rules and log filter queries. A custom expression is a combination of variables that the Infranet Controller evaluates as a boolean object to true, false, or error. Custom expressions enable you to better manage resource access control by providing a means to specify complex statements for policy evaluation and log queries.

You can write custom expressions in the following formats. Note that elements of these formats are described in greater detail in the table that follows:

- variable comparisonOperator variable
- variable comparisonOperator simpleValue
- variable comparisonOperator (simpleValue)
- variable comparisonOperator (OR Values)
- variable comparisonOperator (AND Values)
- variable comparisonOperator (time TO time)
- variable comparisonOperator (day TO day)
- isEmpty (variable)
- isUnknown (variable)
- (customExpr)
- NOT customExpr
- ! customExpr
- customExpr OR customExpr
- `customExpr || customExpr`
- `customExpr AND customExpr`
- `customExpr && customExpr`

The elements used in these custom expression formats are described in the following table:

**Table 43: Custom Expression Elements**

| variable | Represents a system variable. A variable name is a dot-separated string, and each component can contain characters from the set [a-z A-Z 0-9_] but cannot start with a digit [0-9]. Variable names are case-insensitive. For system variables that you may use in role mapping rules and resource policies, see “System Variables and Examples” on page 521. When writing a custom expression in a log query field, you need to use system log variables. These variables are described in the Filter Variables Dictionary on the Filter page (System > Log/Monitoring > Events | User Access | Admin Access > Filters > Select Filter tab). |

Quoting syntax for variables:
The Infranet Controller supports a quoting syntax for custom expression variables that allows you to use any character except ‘.’ (period) in a user attribute name. To escape characters in an attribute name, quote some or all of the variable name using `{ }` (curly-braces). For example, these expressions are equivalent:

- `userAttr.(Login-Name) = ‘xyz’`
- `userAttr>Login{-}Name = ‘xyz’`
- `{userAttr>Login-Name} = ‘xyz’`
- `userA{ttr.L}{ogin-}Name = ‘xyz’`

Escape characters supported within quotes:

| \ \ | represents a \ (backslash) |
| \{} | represents a \{} (left curly-brace) |
| \} | represents a \} (right curly-brace) |
| \hh | represents a hexadecimal value where hh is two characters from [0-9A-Fa-f] |

Examples:
- `userAttr.(Tree Frog) = ‘kermit’`
- `userAttr.Tree\20Frog = ‘kermit’`

Notes:
- There is no limit to the number of quotes you can use in a variable name.
- You can use the quoting syntax with any variable—not just `userAttr.*` variables.
- You need to use curly-brace quotes only when writing custom expressions.
### Table 43: Custom Expression Elements (Continued)

<table>
<thead>
<tr>
<th>comparisonOperator</th>
<th>is one of the following:</th>
</tr>
</thead>
<tbody>
<tr>
<td>=</td>
<td>equal to — Use with strings, numbers, and DNs. See “DN Variables and Functions” on page 521 for more information.</td>
</tr>
<tr>
<td>!=</td>
<td>not equal to — Use with strings, numbers, and DNs. See “DN Variables and Functions” on page 521 for more information.</td>
</tr>
<tr>
<td>&lt;</td>
<td>less than — Use with numbers</td>
</tr>
<tr>
<td>&lt;=</td>
<td>less than or equal to — Use with numbers</td>
</tr>
<tr>
<td>&gt;</td>
<td>greater than — Use with numbers</td>
</tr>
<tr>
<td>&gt;=</td>
<td>greater than or equal to — Use with numbers</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>simpleValue</th>
<th>is one of the following:</th>
</tr>
</thead>
<tbody>
<tr>
<td>string</td>
<td>quoted string that may contain wildcards. See “Wildcard Matching” on page 520 for more information.</td>
</tr>
<tr>
<td>IP Address</td>
<td>a.b.c.d</td>
</tr>
<tr>
<td>subnet</td>
<td>a.b.c.d/subnetBitCount or a.b.c.d/netmask</td>
</tr>
<tr>
<td>number</td>
<td>positive or negative integer</td>
</tr>
<tr>
<td>day</td>
<td>SUN MON TUE WED THU FRI SAT</td>
</tr>
</tbody>
</table>

**Notes about strings:**
- A string may contain all characters except <nl> (newline) and <cr> (carriage return).
- Strings can be any length.
- String comparisons are case-insensitive.
- Strings can be quoted with single- or double-quotes. A quoted string may contain wildcards, including start(*), question mark (?), and square brackets ([ ]). See “Wildcard Matching” on page 520 for more information.
- `variable comparisonOperator variable` comparisons are evaluated without wildcard matching.
- Use a backslash to escape these characters:
  - single-quote (‘) — \`
  - double-quote (”) — \
  - backslash (\) — \\n  - hexadecimal — \hh \[0-9a-fA-F]

**Note about day:**
Day and time comparisons are evaluated in the Infranet Controller’s time zone. Day range (`day TO day`) calculations start with the first day and step forward until the second day is reached. In time range (`time TO time`) calculations, the first value must be earlier than the second value. Only time variables can be compared to day and time values. The time variables are: `time.*` and `loginTime.*`. 
Table 43: Custom Expression Elements (Continued)

<table>
<thead>
<tr>
<th>time</th>
<th>is the time of day in one of the following formats:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• HH:MM — 24-hour</td>
</tr>
<tr>
<td></td>
<td>• HH:MMam — 12-hour</td>
</tr>
<tr>
<td></td>
<td>• HH:MMpm — 12-hour</td>
</tr>
<tr>
<td></td>
<td>• H:MM — 24-hour</td>
</tr>
<tr>
<td></td>
<td>• H:MMam — 12-hour</td>
</tr>
<tr>
<td></td>
<td>• H:MMpm — 12-hour</td>
</tr>
</tbody>
</table>

Day and time comparisons are evaluated in the Infranet Controller’s time zone. Day range (day TO day) calculations start with the first day and step forward until the second day is reached. In time range (time TO time) calculations, the first value must be earlier than the second value. Only time variables can be compared to day and time values. The time variables are: time.* and loginTime.*.

OR Value is a string containing one or more OR comparisons:
- variable comparisonOperator (number OR number ...)
- variable comparisonOperator (string OR string ...)

AND Value is a string containing one or more AND comparisons
- variable comparisonOperator (number AND number ...)
- variable comparisonOperator (string AND string ...)

isEmpty is a function that takes a single variable name (variable) argument and returns a boolean value. isEmpty() is true if the variable is unknown or has a zero-length value, zero-length strings, and empty lists.

Example: isEmpty(userAttr.terminationDate)

isUnknown is a function that takes a single variable name (variable) argument and returns a boolean value. isUnknown() is true if the variable is not defined. User attributes (userAttr.* variables) are unknown if the attribute is not defined in LDAP or if the attribute lookup failed (such as if the LDAP server is down).

Example: isUnknown(userAttr.bonusProgram)

NOT, ! is the logical negation comparisonOperator. The negated expression evaluates to true if the customExpr is false and evaluates to false if the customExpr is true. The operators NOT, AND, and OR are evaluated from highest to lowest precedence in this order: NOT (from right), AND (from left), OR (from left).

OR, || is the logical operator OR or ||, which are equivalent. The operators NOT, AND, and OR are evaluated from highest to lowest precedence in this order: NOT (from right), AND (from left), OR (from left).

AND, && is the logical AND or &&, which are equivalent. The operators NOT, AND, and OR are evaluated from highest to lowest precedence in this order: NOT (from right), AND (from left), OR (from left).

customExpr is an expression written in the Custom Expression Syntax (see above).

**Wildcard Matching**

You may use wildcards within a quoted string. Supported wildcards include:

- **star (*)**—A star matches any sequence of zero or more characters.
- **question mark (?)**—A question mark matches any single character.
square brackets ([ ])—Square brackets match one character from a range of possible characters specified between the brackets. Two characters separated by a dash (-) match the two characters in the specified range and the lexically intervening characters. For example, ‘dept[0-9]’ matches strings "dept0", "dept1", and up to "dept9".

To escape wildcard characters, place them inside square brackets. For example, the expression 'userAttr.x = "value[*]"' evaluates to true if attribute x is exactly "value*".

**DN Variables and Functions**

You can compare a distinguished name (DN) to another DN or to a string, but the Infranet Controller ignores wildcards, white space, and case. Note, however, that the Infranet Controller takes the order of DN keys into consideration.

When the Infranet Controller compares an expression to a DN to a string, it converts the string to a distinguished name before evaluating the expression. If the Infranet Controller cannot convert the string due to bad syntax, the comparison fails. The DN variables are:

- userDN
- certDN
- certIssuerDN

The Infranet Controller also supports DN suffix comparisons using the `matchDNSuffix` function. For example:

```
matchDNSuffix( certDn, "dc=danastreet,dc=net")
```

Within the parenthesis, the first parameter is the “full” DN and the second is the suffix DN. You can use a variable or string for each parameter. Note that this first parameter should have more keys than the second (suffix parameter). Otherwise, if they are equal, it is the same as `<firstparam> = <secondparam>` . If the second parameter has more keys, `matchDNSuffix` returns false.

---

**System Variables and Examples**

The following table lists and defines system variables, gives an example for each system variable, and provides a guide as to where you may use system variables.

---

**NOTE:** This list does not include variables used in a filter query or an export format for a system log. These variables are described in the Filter Variables Dictionary on the Filter page (System > Log/Monitoring > Events | User Access | Admin Access > Filters > Select Filter tab).
### Table 44: System Variables and Examples

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>authMethod</strong></td>
<td>Type of authentication method used to authenticates a user.</td>
<td>authMethod = ‘ACE Server’</td>
</tr>
<tr>
<td>Available in:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>role mapping rules</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>certAttr.&lt;cert-attr&gt;</strong></td>
<td>Attributes from a client-side certificate. Examples of certAttr attributes include:</td>
<td>certAttr.OU = 'Retail Products Group'</td>
</tr>
<tr>
<td>Available in:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>role mapping rules</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LDAP configuration</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>certAttr.altName.&lt;Alt-attr&gt;</strong></td>
<td>Subject alternative name value from a client-side certificate where &lt;Alt-attr&gt; may be:</td>
<td></td>
</tr>
<tr>
<td>Available in:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>role mapping rules</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LDAP configuration</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>certAttr.serialNumber</strong></td>
<td>Client certificate serial number.</td>
<td></td>
</tr>
<tr>
<td>Available in:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>role mapping rules</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LDAP configuration</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>certDN</strong></td>
<td>Client certificate subject DN. Wildcards are not permitted.</td>
<td></td>
</tr>
<tr>
<td>Available in:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>role mapping rules</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Note: All characters other than [0-9 a-f A-F] are stripped out of a string before comparison with certAttr.SN. Wildcards are not supported.*
Table 44: System Variables and Examples (Continued)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
<th>Examples</th>
</tr>
</thead>
</table>
| certDN.<subject-attr> | Any variable from the client certificate subject DN, where `<subject-attr>` is the name of the RDN key. Use to test the various subject DN attributes in a standard x.509 certificate. | certDN.E = 'joe@company.com'
certDN.OU = 'company'
certDN.ST = 'CA' |
| certDNText | Client certificate user DN stored as a string. Only string comparisons to this value are allowed. | certDNText = 'cn=John Harding,ou=eng,c=Company' |
| certIssuerDN | Client certificate-issuer subject DN. This variable works like a standard DN attribute such as CertDN. Wildcards are not permitted. | certIssuerDN = 'cn=John Harding,ou=eng,c=Company'
certIssuerDN = userAttr.x509Issuer
certIssuerDN = ('ou=eng,c=Company' or 'ou=operations,c=Company') |
| certIssuerDN.<issuer-attr> | Any variable from the client certificate-issuer subject DN, where `<issuer-attr>` is the name of the RDN key. | certIssuerDN.OU = 'company'
certIssuerDN.ST = 'CA' |
| certIssuerDNText | Client certificate-issuer subject DN stored as a string. Only string comparisons to this value are allowed. | certIssuerDNText = 'cn=John Harding,ou=eng,c=Company' |
| defaultNTDomain | Contains the Domain value set in the Infranet Controller authentication server configuration when you use AD/NT authentication. | defaultNTDomain = "CORP" |
| group.<group-name> | User’s group membership as provided by the realm authentication or directory server. | group.preferredPartner
group.goldPartner or group.silverPartner
group.employees and time.month = 9 |
| groups | List of groups as provided by the realm authentication or directory server. | groups = ('sales managers') |
| hostCheckerPolicy | Host Checker polices that the client has met. | hostCheckerPolicy = ('Norton' and 'Sygate') |
### Table 44: System Variables and Examples (Continued)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>loginHost</td>
<td>Host name or IP address that the browser uses to contact the Infranet Controller.</td>
<td>loginHost = 10.10.10.10</td>
</tr>
<tr>
<td>loginTime</td>
<td>The time of day at which the user submits his credentials to the Infranet Controller. The time is based on the Infranet Controller time. <strong>NOTE:</strong> When using this variable in an SSO parameter field, the variable returns the UNIX string time.</td>
<td>loginTime = (8:00am)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>loginTime= (Mon to Fri)</td>
</tr>
<tr>
<td>loginTime.day</td>
<td>The day of month on which the user submits his credentials to the Infranet Controller, where day is 1-31. The time is based on the Infranet Controller time. <strong>Note:</strong> You cannot use the TO operator with this variable.</td>
<td>loginTime.day = 3</td>
</tr>
<tr>
<td>loginTime.dayOfWeek</td>
<td>The day of the week on which the user submits his credentials to the Infranet Controller, where dayOfWeek is in the range [0-6] where 0 = Sunday. <strong>Note:</strong> The Infranet Controller do not support the TO operator with time.dayOfWeek expressions if you use numbers instead of strings. In other words, &quot;loginTime.dayOfWeek = (2 TO 6)&quot; does not work, but &quot;loginTime.dayOfWeek = (Mon to Fri)&quot; does work.</td>
<td>loginTime.dayOfWeek = (0 OR 6)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>loginTime.dayOfWeek = (mon TO fri)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>loginTime.dayOfWeek = (1)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>loginTime.dayOfWeek = 5</td>
</tr>
<tr>
<td>loginTime.dayOfYear</td>
<td>The numeric day of the year on which the user submits his credentials to the Infranet Controller, where dayOfYear can be set to [0-365]. <strong>Note:</strong> You cannot use the TO operator with this variable.</td>
<td>loginTime.dayOfYear = 100</td>
</tr>
<tr>
<td>loginTime.month</td>
<td>The month in which the user submits his credentials to the Infranet Controller, where month can be set to [1-12] where 1 = January. <strong>Note:</strong> You cannot use the TO operator with this variable.</td>
<td>loginTime.month &gt;= 4 AND loginTime.month &lt;=9</td>
</tr>
<tr>
<td>loginTime.year</td>
<td>The year in which the user submits his credentials to the Infranet Controller, where year can be set to [1900-2999]. <strong>Note:</strong> You cannot use the TO operator with this variable.</td>
<td>loginTime.year = 2005</td>
</tr>
<tr>
<td>loginURL</td>
<td>URL of the page that the user accessed to sign in to the Infranet Controller. The Infranet Controller gets this value from the Administrator URLs</td>
<td>User URLs column on the Authentication &gt; Signing In &gt; Sign-in Policies page of the admin console.</td>
</tr>
<tr>
<td>Variable</td>
<td>Description</td>
<td>Examples</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------------------------------</td>
</tr>
<tr>
<td>ntdomain</td>
<td>The NetBIOS NT domain used in NT4 and Active Directory authentication.</td>
<td>ntdomain = jnpr</td>
</tr>
<tr>
<td>ntuser</td>
<td>The NT username used in Active Directory authentication</td>
<td>ntuser = jdoe</td>
</tr>
<tr>
<td>password</td>
<td>The password entered by the user for the primary authentication server</td>
<td>password = A1defo2z</td>
</tr>
<tr>
<td>password[1]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>realm</td>
<td>The name of the authentication realm to which the user is signed in.</td>
<td>Realm = ('GoldPartners' or 'SilverPartners')</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Note: AND condition will always fail as a user is only allowed to sign in to a single realm in a session.</td>
</tr>
<tr>
<td>role</td>
<td>List of all the user roles for the session.</td>
<td>Role = ('sales' or 'engineering')</td>
</tr>
<tr>
<td>sourceIP</td>
<td>The IP address of the machine on which the user authenticates. You can specify the netmask using the bit number or in the netmask format: '255.255.0.0'. Note that you can evaluate the sourceIP expression against a string variable such as an LDAP attribute.</td>
<td>sourceIP = 192.168.10.20</td>
</tr>
<tr>
<td>time</td>
<td>The time of day at which the role mapping rule or resource policy rule is evaluated. The time of the day can be in 12-hour or 24-hour format.</td>
<td>time = (9:00am to 5:00pm)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>time = (09:00 to 17:00)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>time = (Mon to Fri)</td>
</tr>
</tbody>
</table>

Combination examples:

Allow executive managers and their assistants access from Monday to Friday:

userAttr.employeeType = ('*manager*' or '*assistant*') and group.executiveStaff and time = (Mon to Fri)
Table 44: System Variables and Examples (Continued)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>time.day</strong></td>
<td>The day of month on which the user submits his credentials to the Infranet Controller, where day is 1-31. The time is based on the Infranet Controller time.</td>
<td><code>loginTime.day = 3</code></td>
</tr>
</tbody>
</table>
| **time.dayOfWeek**| The day of the week on which the role mapping rule or resource policy rule is evaluated, where dayOfWeek is in the range [0-6] where 0 = Sunday. | `loginTime.dayOfWeek = (0 OR 6)`  
`loginTime.dayOfWeek = (1 to 5)`  
`loginTime.dayOfWeek = 5`    |
| **time.dayOfYear**| The day of the year on which the role mapping rule or resource policy rule is evaluated. Possible values include: 1-365. | `time.dayOfYear = 100` |
| **time.month**    | The month in which the role mapping rule or resource policy rule is evaluated. Possible values include: 1-12 | `time.month >= 9 and time.month <= 12`  
`time.year = 2004`  
`group.employees and time.month = 9` |
| **time.year**     | The year in which the role mapping rule or resource policy rule is evaluated, where year can be set to [1900-2999]. | `time.year = 2005` |
| **user**          | Infranet Controller username for the user’s primary authentication server (user and user[1]). Use when authenticating against an Active Directory server, domain and username. | `user = 'steve'`  
`user = 'domain\steve'` |
| **username**      | Infranet Controller username for the user’s primary authentication server (username and username[1]). If the user is signing in to a certificate authentication server, then the user’s Infranet Controller username is the same as CertDN.cn. | `username = 'steve' and time = mon`  
`username = 'steve'`  
`username = 'steve*'`  
`username = ('steve' or '*jankowski')` |
| **userAgent**     | The browser’s user agent string.                                            | The browser’s user agent string. |
### Table 44: System Variables and Examples (Continued)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>userAttr.&lt;auth-atr&gt;</td>
<td>User attributes retrieved from an LDAP, RADIUS, or SiteMinder authentication or directory server.</td>
<td>userAttr.building = ('HQ*' or 'MtView[1-3]')&lt;br&gt;userAttr.dept = ('sales' and 'eng')&lt;br&gt;userAttr.dept = ('eng' or 'it' or 'custsupport')&lt;br&gt;userAttr.division = 'sales'&lt;br&gt;userAttr.employeeType != 'contractor'&lt;br&gt;userAttr.salaryGrade &gt; 10&lt;br&gt;userAttr.salesConfirmed &gt;= userAttr.salesQuota</td>
</tr>
<tr>
<td>userDN</td>
<td>The user DN from an LDAP server. If the user is authenticated by the LDAP server, then this DN is from the authentication server; otherwise, the DN comes from the realm's Directory/Attribute server. Wildcards are not permitted.</td>
<td>userDN = 'cn=John Harding,ou=eng,c=Company'&lt;br&gt;userDN = certDN</td>
</tr>
<tr>
<td>userDN.&lt;user-atr&gt;</td>
<td>Any variable from the user DN, where user-atr is the name of the RDN key.</td>
<td>userDNText = 'cn=John Harding,ou=eng,c=Company'</td>
</tr>
<tr>
<td>userDNText</td>
<td>User DN stored as a string. Only string comparisons to this value are allowed.</td>
<td>userDNText = 'cn=John Harding,ou=eng,c=Company'</td>
</tr>
</tbody>
</table>

**Negative examples:**
- userAttr.company != "Acme Inc" or not group.contractors
- not (user = 'guest' or group.demo)

**Combination examples:**
- Allow executive managers and their assistants access from Monday to Friday:

  userAttr.employeeType = ('*manager*' or '*assistant*') and<br>  group.executiveStaff and<br>  time = (Mon to Fri)

- Allow all partners with active status from Monday to Friday but preferred partners Monday through Saturday:

  ((group.partners and time = (Mon to Fri)) or<br>  (group.preferredPartners and time = (Mon to Sat))) and<br>  userAttr.partnerStatus = 'active'
Symbols
 Administrators role476
 Read-Only Administrators role476

Numerics
 128-bit encryption350
 168-bit encryption350
 802.1X IP phone, role mapping138
 802.1X network access142

A
 accelerator card, gzip514
 Access96
 access management, restrictions, specifying122
 access policy for Infranet Enforcer, configuring72
 Access-Accept RADIUS attribute213
 Access-Accept RADIUS authentication206
 Access-Challenge RADIUS attribute213
 Access-Challenge RADIUS authentication206
 Access-Reject RADIUS attribute213
 Access-Reject RADIUS authentication206
 Access-Request RADIUS attribute213
 Access-Request RADIUS authentication206
 accounting server, See authentication server, RADIUS
 Accounting-Request RADIUS attribute213
 Accounting-Response RADIUS attribute213
 Acct-Authentic RADIUS attribute212, 213
 Acct-Delay-Time RADIUS attribute213
 Acct-Input-Gigawords RADIUS attribute213
 Acct-Input-Octets RADIUS attribute213
 Acct-Input-Packets RADIUS attribute213
 Acct-Interim-Interval RADIUS attribute214
 Acct-Link-Count RADIUS attribute212, 214
 Acct-Multi-Session-Id RADIUS attribute212, 214
 Acct-Output-Gigawords RADIUS attribute214
 Acct-Output-Octets RADIUS attribute214
 Acct-Output-Packets RADIUS attribute214
 Acct-Session-Id RADIUS attribute212, 214
 Acct-Session-Time RADIUS attribute212, 214
 Acct-Status-Type RADIUS attribute212, 214
 Acct-Terminate-Cause RADIUS attribute212, 214
 Acct-Tunnel-Connection RADIUS attribute214
 Acct-Tunnel-Packets-Lost RADIUS attribute214
 ACE/Server, See authentication server, ACE/Server
 Active Directory, See authentication server, Active Directory
 adapter, Odyssey Access Client, initial configuration of46
 admin console, homepage339
 administrator
   realms478

See also realms
   roles478
     defined117
 See also roles
  super administrator account, creating502
  user account, creating474
  user administrator
    See user, user administrators
  administrator role, defined473
  Advanced Settings463
  AES encryption, support180
  Agent page, configuring roles126
  agentless access
    configuring56
    enabling on role126
  AND custom expression comparison operator517, 518, 520
  anonymous server, See authentication server, anonymous
  server
  ARAP-Challenge-Response RADIUS attribute213
  ARAP-Features RADIUS attribute213
  ARAP-Password RADIUS attribute213
  ARAP-Security RADIUS attribute213
  ARAP-Security-Data RADIUS attribute213
  ARAP-Zone-Access RADIUS attribute213
  archive, defining416
  archiveFileTransferFailed MIB object428
  archiveServerLoginFailed MIB object428
  archiveServerUnreachable MIB object428
  ARP
    command501
      Ping Timeout, configuring332, 333
    ARP cache, configuring337
    attribute server, See authentication server
    attributes
      configuring250
    auth table
      auth table entry, described15, 93
      mapping policies, configuring93
    auth table entries, number of95
    auth tables, dynamic allocation92
    authentication policies
      configuring248
      defined12, 68, 245
    authentication protocol, Host Checker Statement of Health
    policy138
    authentication protocols132
    authentication protocols, Odyssey Access Client default45
    authentication protocols, overview133
authentication realms, See realms
authentication server
ACE/Server
  Agent configuration file
  configuring180
  overview
  RADIUS protocol support
  resolving host names
  restrictions
  SecurID authentication
  supported modes and features
Active Directory
  configuring
  group lookup support
  multi-domain configuration
  overview
  password management
  restrictions
  supported server features
anonymous server
  configuring
  overview
  restrictions
certificate server
  configuring
  defined
  LDAP, authenticating against
  overview
  restrictions
custom expressions variable
  defined
  directory server, defined
  general
    configuration
    overview
  LDAP
    configuring
    configuring attributes
    group lookup
    overview
    password management
    referral chasing
    restrictions
    retrieving attributes
local authentication
  accounting
  configuring
  creating users
  managing user accounts
  overview
  password management
  restrictions
  user administrators
  mapping to realm
NIS server
  preconfigured servers
  RADIUS
    accounting
    ACE/Server protocol
    attributes
CASQUE authentication
  cluster considerations
  configuring
  configuring attributes
  multiple sessions
  NAS
  overview
  PassGo Defender
  password management
  retrieving attributes
  role-mapping attributes
  SSO
  start attributes
  stop attributes
  supported server features
  user experience
SiteMinder
  ACE SecurID authentication
  agents
  authentication schemes
  automatic sign-in
  certificate authentication
  configuration (general)
  configuring the SiteMinder policy server
  cookie domain
  cookie provider domain
  custom sign-in pages
  debugging
  failover mode
  overview
  password management
  policies
  policy domains
  protected resources
  protection levels
  realms
  response
  restrictions
  retrieving attributes
  role-mapping
  rules
  SMSESSION cookie
  SSO
  supported versions
  user attributes
  usernames, determining
authentication server limitations, authentication protocols
  authentication settings, for users
  authMethod custom expression variable
  authorization server, See authentication server
  authServerName MIB object
base CRL
  defined
  See also CRL
bidirectional VPN policy
  configuring
blockedIP MIB object
Index

blockedIPList MIB object427
browser
  restrictions, configuring110
sign-in restrictions, user110
SiteMinder security settings235

C
CA certificate, See certificate, CA certificate
Callback-Id RADIUS attribute214
Callback-Number RADIUS attribute214
Called-Station-Id RADIUS attribute214
Calling-Station-Id RADIUS attribute214
canonical format, overview70
captive portal
  configuring redirection of user requests33
  overriding default redirection destination36
  redirect infranet-auth policy, configuring34
CASQUE authentication support, See authentication server, RADIUS
CDP, See CRL distribution point
Central Management features overview339
certAttr.altName.Alt-attr custom expression variable522
certAttr.cert-attr custom expression variable522
certAttr.serialNumber custom expression variable522
certDN custom expression variable522
certDN.subject-attr custom expression variable523
certDNText custom expression variable523
certificate
  attributes, configuring250
CA certificate
  enabling CRL checking374
  renewing368
  uploading to the Infranet Controller364
  verifying375
  viewing details380
client-side certificate
  defined355
  SiteMinder223
configuring in Infranet Controller and Infranet Enforcer16, 23, 25
CRLs, enabling374
custom expression variables522
device certificate
  creating CSR359
  defined355
  exporting existing357
  importing CSR360
  importing existing357, 360
  importing renewed358
  multiple certificates, enabling356
  hierarchy
    defined364
    discussed360, 371
intermediate certificate, defined364
key
  exporting existing357
  import existing357
  importing existing360
machine certificate
  checking using Host Checker292
  configuring367
  overview363
  See also
certificate, CA certificate, uploading to the IVE
Host Checker, machine certificate
  restrictions, configuring69
revocation list
    defined364
    discussed372
revocation, defined364
security requirements, defining112
self-signed356
server certificate, defined355
server See authentication server, certificate server
signing request
  creating359
  importing360
  importing certificate from360
SiteMinder security settings235
supported formats356, 363, 379
wildcard certificate, defined362
certificate revocation list, See certificate, revocation list
certissuerDN custom expression variable523
certissuerDN.issuer-attr custom expression variable523
certissuerDNText custom expression variable523
chained certificate, See certificate, intermediate certificate
challenge-response protocols disabled, LDAP193
CHAP-Challenge RADIUS attribute214
CHAP-Password RADIUS attribute214
Class RADIUS attribute214
CLI commands for Infranet Enforcer controller40
  enforcer40
  exec infranet controller38
  get infranet controller38
  infranet38
  policy command39
  set infranet38
  unset infranet39
client policy for 802.1X, configuring147
client-side logs, See logging, client logs
cluster
  ACE/Server support180
  active/active457
  deployment overview457
  figure458
  active/passive455
  deploying overview455
  figure455
changing node IP472
configuring449, 451
deleting465
hostnames330
initializing449, 450
joining452
logging460, 461
managing452, 465
modifying properties462, 465
password459
RADIUS accounting211
restart, reboot, shut down342
state synchronization459
status defined466
synchronization449, 459
system data342
upgrading465
clusterConcurrentUsers MIB object426
Compliant, PatchLink checks295
concurrent sessions, limiting the number of114
concurrent users, SiteMinder security settings235
configuration file
local user accounts, exporting387
local user accounts, importing388
system, exporting386
system, importing387
configuration overview, UAC solution11
Configuration-Token RADIUS attribute214
Connect-Info RADIUS attribute214
connection policy for Infranet Enforcer, configuring20
connectivity, testing341
core, serial, See serial console
coreler CLI command for Infranet Enforcer40
conventions defined
textxiii
cookies, deleting at session termination351
CPU usage, viewing431
cpuUtilNotify MIB object429
credentials
verifying12
verifying See authentication server
critical log messages, defined416
CRL distribution point
discussed372
downloading CRLs from374
CRL, See certificate, revocation list
CSR
creating359
importing360
importing certificate from360
custom expressions
customizable UI
uploading266
using with SiteMinder223
D
Dashboard
configuring431
overview339
XML output339
database, authentication, See authentication server
date and time, setting433
in log filters518
in role mapping rules522
values, defined519
variables
authMethod522
certAttr altName Alt-attr522
certAttr cert-attr522
certAttr serialNumber522
certDN521, 522
certDN subject-attr523
certDNText523
certIssuerDN521, 523
certIssuerDN issuer-attr523
certIssuerDNText523
defaultNTDomain523
group, group-name523
groups523
hostCheckerPolicy523
loginHost524
loginTime524
loginTime.day524
loginTime.dayOfWeek524
loginTime.dayOfYear524
loginTime.month524
loginTime.year524
loginURL524
ntDomain525
ntuser525
overview518
password525
quoting syntax518
realm525
role525
sourceIP525
time520, 525
time.day526
time.dayOfWeek526
time.dayOfYear526
time.month526
time.year526
user526
userAgent526
userAttr auth-attr527
userDN521, 527
userDN user-attr527
userDNText527
username526
wildcard matching520
in LDAP configurations522
defaultNTDomain custom expression variables523
Defender support, See authentication server, RADIUS
deleagated administration
overview473
denied, resource access73
deploying Unified Access Control solution to users41
diagnostic log, RADIUS support180
digital certificate
defined355
See also certificate
directory server
defined173
See also authentication server
disable all users434
diskFullPercent MIB object427
DLL requirements, Host Checker
See Host Checker, client interface
See Host Checker, server integration interface
DMZ, interface331
DNS
for external port331
domain name, configuring330
dynamic auth table allocation92
dynamic policy evaluation105, 247
E
EAP-Message RADIUS attribute215
EAP-PEAP133
EAP-SOH authentication protocol134
EAP-TTLS133
802.1X, configuring on Infranet Controller132
elements, instance391
eligible roles, defined249
emergency mode
activating349
deactivating349
empty tag, XML392
encryption
custom SSL ciphers351
passwords393
strength350
Encryption Strength option351
end tag, XML392
endpoint defense
overview269
See also Host Checker
Endpoint Security Assessment Plug-In, upgrading318
enforcement points, description of4
enforcer CLI command for Infranet Enforcer40
error messages, modifying266
eTrust, See authentication server, SiteMinder
event monitor, viewing432
exec infranet controller CLI command38
export network settings399
external ports331
externalAuthServerUnreachable MIB object427
F
Fail-over456
fanDescription MIB object427
file
access statistics430
check, configuring289
fileName MIB object429
Filter-Id RADIUS attribute215
FIPS compatibility54
Framed-AppleTalk-Link RADIUS attribute215
Framed-AppleTalk-Network RADIUS attribute215
Framed-AppleTalk-Zone RADIUS attribute215
Framed-Compression RADIUS attribute215
Framed-IP-Address RADIUS attribute212, 215
Framed-IP-Netmask RADIUS attribute215
Framed-IPv6-Pool RADIUS attribute215
Framed-IPv6-Route RADIUS attribute215
Framed-IPX-Network RADIUS attribute215
Framed-MTU RADIUS attribute215
Framed-Pool RADIUS attribute215
Framed-Protocol RADIUS attribute215
Framed-Route RADIUS attribute215
Framed-Routing RADIUS attribute215
FTP archive, defining416
G
gateway
configuring331, 332
for external port331
get infranet controller CLI command38
graphs
configuring431
XML output339
group membership
custom expression variables523
LDAP251
retrieving173
group.groupname custom expression variable523
groups custom expression variable523
gzip accelerator514
H
hardware token, using to sign in179
health check URL457
help, modifying267
home page, customize123
Host Checker
auto-upgrade524
custom expression variables523
eexecution312
frequency check321
installer
directory313
enabling314
integration measurement requirements287, 295
logging, disabling325, 430
machine certificate
configuring292
overview363
See also certificate, machine certificate
overview271
policies275
remediation
configuring317
overview315
server integration interface
enabling310
SiteMinder security settings235
specifying restrictions
overview275
realm level311, 314
role level311, 314
uninstalling313
Host Enforcer
enabling127
overview97, 98
policies, configuring97
protocols for policies100
traffic allowed by default98, 127
host, defining in resource policies71
hostCheckerPolicy custom expression variable523
hostname
configuring330
resolving337
HP OpenView support416
icons defined, notice xiii
Idle-Timeout RADIUS attribute215
IDP configuration486
IDP interaction, enabling489
IDP licensing486
IMCs and IMVs, overview269, 271
info log messages, defined417
infranet CLI command for Infranet Enforcer38
Infranet Controller
description of3
overview7
Infranet Enforcer
access policy, configuring72
auth table mapping policy, configuring94
binding interface to security zone26
CLI commands for configuring38
configuring22
configuring Certificate Authority server settings27
connection policy, configuring20
creating instance of29, 31
description of4
IP pool policy, configuring85
IPsec, configuring78
overview7
Route mode, setting up23
source interface policy, configuring89
source IP-based policy33, 91
Transparent mode
setting up24
source interface policy82, 89
using for 802.1X enforcement159
viewing configuration of32
virtual routers, configuring a source interface policy for89
initial configuration of Odyssey Access Client44
installers, downloading344
integrity measurement collectors (IMCs) and verifiers (IMVs)269, 271
integrity measurement PatchLink, Host Checker295
integrity measurement requirements, Host Checker287
intermediate certificate. See certificate, intermediate certificate
internal port, configuring331
IP address
configuring331, 332
for external port331
node472
resolving337
restrictions109, 110, 113, 313, 369
SiteMinder security settings235
specifying user requirements108
IP alias
activating. See virtual port
defined335
IP pool policy for Infranet Enforcer, configuring85
ipEntry MIB object427
ipIndex MIB object427
IPsec
bidirectional VPN policy, configuring77
CLI commands on Infranet Enforcer32, 75
configuring IPsec task summary80
IP pool policies, configuring85
IPsec encryption on Infranet Enforcer, configuring76
IPsec routing policy, configuring82
NAT, configuring IP pool policies for85
overview of78
refreshing policies on Infranet Controller82
upgrading from version 1 x90
ipValue MIB object427
isEmpty custom expression function517, 520
isUnknown custom expression function517, 520
iveAppletHits MIB object426
iveClusterChangedVIPTrap429
iveClusterDelete430
iveConcurrentUsers MIB object428
iveCPUUtil MIB object426
iveDiskFull MIB object429
iveDiskNearlyFull MIB object429
iveFanNotify SNMP trap429
iveFileHits MIB object426
iveFlashFree MIB object429
iveFlashUsage MIB object429
iveFlashUsage MIB object429
iveFileHits MIB object426
iveMemoryUtil MIB object426
iveNetworkUtil MIB object426
ivePacketCount MIB object429
ivePacketErrors MIB object429
ivePacketErrors MIB object429
iveMaxConcurrentUsersSignedIn MIB object428
iveMaxConcurrentUsersSignedIn MIB object428
iveMaxConcurrentUsersSignedIn MIB object429
iveMaxConcurrentUsersSignedIn MIB object429
iveMaxConcurrentUsersSignedIn MIB object429
iveMemoryUtil MIB object426
iveNCHits MIB object426
iveNetExternalInterfaceDownTrap429
iveNetInternalInterfaceDownTrap429
ivePowerSupplyNotify SNMP trap429
iveRaidNotify SNMP trap429
iveReboot MIB object428
iveRestart MIB object428
iveSAMHits MIB object426
iveShutdown MIB object428
iveStart MIB object428
iveSwapUtil MIB object427
ivetermHits MIB object426
iveTooManyFailedLoginAttempts MIB object428
iveTotalHits MIB object426
iveWebHits MIB object426
ivsName MIB object427

J
Java agent, configuring access58
Juniper Installer Service, described344

K
Keep-Alives RADIUS attribute215
Kerberos single sign-on, configuring185
key357
  private, See private key

L
L7 Health Check URL457
LAN, modifying network settings331
LDAP server, See authentication server, LDAP
licenses
  entering Infranet Controller license348
  overview of Infranet Controller346
  upgrading Odyssey Access Client licenses54
licenses for Odyssey Access Client, using with Infranet Controller54
limiting user sessions115
link speed, configuring331, 332
Linux, SUSE, installing Java agent59
load balancer, using in a cluster457
local authentication server, See authentication server, local
authentication
local server, See authentication server, local authentication
location group policy, configuring145
logDescription MIB object427
logFullPercent MIB object425
logging
  client logs
    disabling325, 430
    clusters460, 461
    critical events432
  filters
    configuring422
    overview417
  formats
    configuring422
    overview417
  policy tracing438
  saving log files419
  severity levels416
  specifying events to log421
logID MIB object427
loginHost custom expression variable524
Login-IP-Host RADIUS attribute215
Login-LAT-Group RADIUS attribute215
Login-LAT-Node RADIUS attribute215
Login-LAT-Port RADIUS attribute215
Login-LAT-Service RADIUS attribute215
Login-Service RADIUS attribute215
Login-TCP-Port RADIUS attribute216
loginTime custom expression variable524
loginTime day custom expression variable524
loginTime dayOfWeek custom expression variable524
loginTime dayOfYear custom expression variable524
loginTime month custom expression variable524
loginTime year custom expression variable524
loginURL custom expression variable524
logMessageTrap MIB object429
logName MIB object427
logType MIB object427

M
MAC address authentication163
MAC address authentication server242
MAC address, configuring requirement in Host Checker
policy291
machine account host checker policy277
maintenance tasks, delegating476
major log messages, defined416
management information base overview416
maximum transmission unit, configuring332, 333
memory usage, viewing431
memUtilNotify MIB object429
MIB overview416
minor log messages, defined417
MS-Acct-Auth-Type RADIUS attribute216
MS-Acct-EAP-Type RADIUS attribute216
MS-ARAP-Challenge RADIUS attribute216
MS-ARAP-Password-Change-Reason RADIUS attribute216
MS-BAP-Usage RADIUS attribute216
MS-CHAP2-CPW RADIUS attribute216
MS-CHAP2-Response RADIUS attribute216
MS-CHAP2-Success RADIUS attribute216
MS-CHAP-Challenge RADIUS attribute216
MS-CHAP-CPW-1 RADIUS attribute216
MS-CHAP-CPW-2 RADIUS attribute216
MS-CHAP-Domain RADIUS attribute216
MS-CHAP-Error RADIUS attribute216
MS-CHAP-LM-Enc-PW RADIUS attribute216
MS-CHAP-MPPE-Keys RADIUS attribute216
MS-CHAP-NTEnc-PW RADIUS attribute216
MS-CHAP-Response RADIUS attribute216
MS-Filter RADIUS attribute216
MS-Link-Drop-Time-Limit RADIUS attribute216
MS-Link-Utilization-Threshold RADIUS attribute216
MS-MPPE-Encryption-Policy RADIUS attribute216
MS-MPPE-Encryption-Types RADIUS attribute216
MS-MPPE-Recv-Key RADIUS attribute217
MS-MPPE-Send-Key RADIUS attribute217
MS-New-ARAP-Password RADIUS attribute217
MS-Old-ARAP-Password RADIUS attribute217
MS-Primary-DNS-Server RADIUS attribute217
N

NACN password for Infranet Enforcer connection policy20
name locking support180
namespaces, XML393
NAS, See authentication server, RADIUS, NAS
NAS-Identifier RADIUS attribute212, 217
NAS-IP-Address RADIUS attribute212, 217
NAS-Port RADIUS attribute212, 217
NAS-Port-Id RADIUS attribute217
NAS-Port-Type RADIUS attribute217
NAT
Always use a virtual adapter option, configuring84, 85
Always use UDP encapsulation option, configuring84
configuring IP pool policies for85
enabling NAT-src on Infranet Enforcer88
NetBIOS, configuring requirement in Host Checker policy
netmask
configuring331, 332
defining user requirements108
for external port331
Netscape, web server357
Network Access Server, See authentication server, RADIUS, NAS
Network Healthcheck Settings463
network settings
configuring331, 501
initial529
Network Time Protocol, using433
network, Odyssey Access Client, initial configuration of46
New PIN mode support179
Next Token mode support179
NIS authentication server, See authentication server, NIS
No Baseline Defined, PatchLink check295
nodes, cluster452, 465
Non-Compliant, PatchLink check295
non-tunneled protocols, authenticating with163
non-UAC client, definition260
NOT custom expression comparison operator517, 520
Not Found, PatchLink check295
notice icons definedxiii
NSM, using with the Infranet Controller60
NT Domain, See authentication server, Active Directory
ntdomain custom expression variable525
NTLM configuration, Active Directory server175
NTP, using433
ntuser custom expression variable525
null server, See authentication server, anonymous server
O
ocspResponderUnreachable SNMP trap429
ocspResponderURL MIB object427
Odyssey Access Client
certificates, configuring53
description of4
documentation forxii
Host Enforcer policy, configuring97
initial configuration of44
installation methods41
installing322, 344
licenses, using with Infranet Controller54
wireless settings, initial configuration of49
Odyssey Access Client license, install, UAC version 2.251
open port, RADIUS attributes option155
OpenSSL, using to create CA18
Optical Responder tokens, See authentication server, RADIUS, CASQUE
OR custom expression comparison operator517, 520
P
PassGo Defender support, See authentication server, RADIUS
password
custom expression variable525
encrypted393
for new users393
in XML file393
management
LDAP195
local authentication server200, 203
RADIUS180, 209
SiteMinder226
plaintext393
security requirements, defining113
SiteMinder security settings235
verifying See authentication server
Password-Retry RADIUS attribute217
PatchLink IMVs
compliant check295
no baseline found295
noncompliant295
not found295
permissive merge, overview119
persistent data, defined459
persistent tunnel mode84
phase 2 proposal, configuring on Infranet Enforcer76
PIMs
installing509
ping command501
PKI, defined355
platform, upgrading343
policies
802.1X132
evaluating107
Host Enforcer97
Network Access96
See also authentication policies
See also sign-in policies
policy command CLI command for Infranet Enforcer39
Policy Tracing subtab438
port
Index

defining in resource policies71
external331
internal331
modifying external331
requirements, configuring288
See also virtual port
Port-Limit RADIUS attribute217
preconfigured OAC installer51
private key, importing357
process check, configuring288
productName MIB object426
productVersion MIB object426
profile, Odyssey Access Client, initial configuration of45
Prompt RADIUS attribute217
protocol
defining in resource policies70
protocol, defining in resource policies70
proxy RADIUS138
Proxy-State RADIUS attribute217
psDescription MIB object427
public key infrastructure, defined355

R
RADIUS attributes policy, creating154
RADIUS attributes, disconnecting concurrent network connections158
RADIUS diagnostic log444
RADIUS proxy138
RADIUS server, Infranet Controller132
RADIUS, See authentication server, RADIUS
RADIUS, time limits141
RADIUS-802.1X enforcement
client policy for 802.1X, configuring147
location group policy, configuring145
overview of using 802.1X with UAC, 132
using Infranet Enforcer for159
VLAN policy, configuring151
rapid response, disabling all users434
realm
administrator478
custom expression variable525
defined12, 65, 68
managing477, 478
mapping to sign-in policy264
preconfigured realms12
security requirements69
reason strings, for IMV/IMC remediation295, 299, 315
rebooting the Infranet Controller341
recording user sessions438
redirect infranet-auth policy, configuring34
redirecting users to Infranet Controller33
redundancy, Active/Passive mode455
referential integrity394
referral chasing support195
registry setting checks, configuring290
remediation
configuring317
overview315
reason strings295, 299
Send Reason Strings option, selecting295, 299
Remote Authentication Dial-In User Service, See authentication server, RADIUS
remote IMV server, configuring299
Reply-Message RADIUS attribute217
resource policies
802.1X132
auth table mapping93
exporting399, 402
Host Enforcer97
importing398
IP pool85
IPsec routing82
IPsec, overview of78
managing480
server70
source interface89
restarting the Infranet Controller341
role
administrator478
configuring121
custom expression variable525
defined65, 69, 117
evaluating118
exporting399
importing398
mapping69, 119, 245, 249, 250
merging119
options, managing121
restrictions122
settings, managing121
sign-in restrictions
by Host Checker policy313
user session options, specifying122
role evaluation118
role restrictions, with RADIUS proxy121
Route mode, setting up Infranet Enforcer in23
Routes tab336
routing policy for Infranet Enforcer82
RSA
ACE/Server, See authentication server, ACE/Server
RADIUS support, See authentication server, RADIUS
rule, configuring for role mapping251

S
SCP, system snapshot502
ScreenOS compatibility14
scripts, configuring start and end126, 127
sdconf rec, generating181
SecurID tokens, See authentication server, ACE/Server, SecurID
security administrator478
security options, configuring350
Send Reason Strings option, remediation295, 299
sequence elements394
serial console, using for system tasks497
serial numbers in Infranet Enforcer connection policy20
server
authentication, See authentication server
catalog, configuring252
resource policies70
See also authentication server
service package
installing343, 352
installing in a cluster465
Service-Type RADIUS attribute217
session
ending434
options, specifying122
roaming, configuring123
role, defined118
timeout, maximum length122
session scripts, configuring126, 127
session timeout, RADIUS accept152
session time-outs, configuring69
Session-Timeout attribute155
Session-Timeout RADIUS attribute218
set infranet CLI command for Infranet Enforcer38
shutting down the Infranet Controller341
signedInMailUsers MIB object426
signedInWebUsers MIB object425
sign-in
management tasks, delegating476
options, user restrictions110, 313
pages
custom265
defining265
exporting401
mapping to sign-in policies260, 263
standard265
policies
changing order265
configuring263
defined257, 259
disabling264
enabling264
evaluating264
single sign-on using Kerberos, configuring185
SiteMinder, See authentication server, SiteMinder
slave ACE/Server support180
SMSESSION cookie, See authentication server, SiteMinder
snapshots, creating384, 502
SNMP
iveNetManagementInterfaceDownTrap430
monitoring Infranet Controller as agent424
settings, specifying424
support416
traps
iveClusterChangedVIPTrap429
iveClusterDelete430
iveClusterDisableNodeTrap429
iveNetExternalInterfaceDownTrap429
iveNetInternalInterfaceDownTrap429
SoftID tokens, See authentication server, ACE/Server
software
installing352
installing in a cluster465
token, using to sign in179
SOH Host Checker policy305
source interface policy, configuring89
source IP policies
auth table mapping policies93
configuring with CLI in Infranet Enforcer33, 91
summary of configuring92
source IP restrictions on Infranet Controller69
sourceIP custom expression variable525
spyware, specifying requirements269
SSL
encryption strength allowed350
version allowed350
SSL ciphers, selecting351
SSO, See Kerberos single sign-on
Standard log file format423
start tag, XML392
State RADIUS attribute218
state synchronization459
statement of health, authentication protocol for134
static routes336
statistics, viewing430
Steelbelted-RADIUS support, See authentication server, RADIUS
super administrator account, creating502
supplicant for 802.1X, non-Juniper160
swapUtilNotify MIB object429
Synchronization Settings462
Synchronize last access time for user sessions461, 463
Synchronize log messages461, 462
Synchronize user sessions461, 462
syslog servers, configuring421
system
administrator478
capacity, viewing431
configuration341
configuration, exporting386
configuration, importing387
data341
management tasks, delegating476
snapshot446, 447
software, installing352
state data described459
statistics, viewing430
T
tabs, Routes336
tcp_in and tcp_out, for Host Enforcer policies70
Telephone-number RADIUS attribute218
templates, customizable UI266
Termination-Action RADIUS attribute218
text conventions definedxiii
third-party IMVs, configuring299
THMTL, using to create custom pages266
throughput usage, viewing431
time and date, setting433
time custom expression variable525
time.day custom expression variable526
time.dayOfWeek custom expression variable526
time.dayOfYear custom expression variable526
time.month custom expression variable526

time.year custom expression variable526
timeout, maximum session122
time-out, See also session time-outs

TLS, version allowed350

TO custom expression comparison operators517
token, using to sign in179

traceroute command501

transient data, defined459

Transparent mode

setting up Infranet Enforcer in24

source interface policy, configuring82, 89

traps

configuring425

defining416

troubleshooting, RADIUS diagnostic log444

Trusted Computing Group (TCG)269

Trusted Network Connect (TNC)269

trusted server for Odyssey Access Client, initial configuration46

Tunnel-Assignment-ID RADIUS attribute218

Tunnel-Client-Auth-ID RADIUS attribute218

Tunnel-Client-Endpoint RADIUS attribute218

Tunnel-Link-Reject RADIUS attribute218

Tunnel-Link-Start RADIUS attribute218

Tunnel-Link-Stop RADIUS attribute218

Tunnel-Medium-Type RADIUS attribute218

Tunnel-Password RADIUS attribute218

Tunnel-Preference RADIUS attribute218

Tunnel-Private-Group-ID RADIUS attribute218

Tunnel-Reject RADIUS attribute218

Tunnel-Server-Auth-ID RADIUS attribute218

Tunnel-Server-Endpoint RADIUS attribute218

Tunnel-Start RADIUS attribute218

Tunnel-Stop RADIUS attribute218

Tunnel-Type RADIUS attribute218

U

UAC access policies

65

UAC agent, description of4

UDP port 6543447

Unified Access Control solution

agentless access

configuring56

enabling on role126

components working together8

deploying to users41

Kerberos single sign-on, configuring185

overview of3

setting up certificates for16, 23, 25

summary of steps for configuring10

UNIX, authentication server, See authentication server, NIS

unset infranet CLI command for Infranet Enforcer39

upgrade, to UAC 2.1 from earlier version343

upgrading

Endpoint Security Assessment Plug-In318

IPsec policies from version 1.x90

service package343

URLs sign-in URLs, defining263

URLs, access statistics430

user

accounts

creating120, 202

exporting387, 402

importing388

local402

attributes, configuring250

creating202

custom expression variable526

forcing to exit434

managing accounts203

role

defined117

See also role

session data460

sign-in restrictions by browser110

user administrators

authenticating175

defining204

overview204

viewing activity431

user access control

specifying resources for70

userAgent custom expression variable526

userAttr.auth-attr custom expression variable527

userDN custom expression variable527

userDN user-attr custom expression variable527

userDNText custom expression variable527

username

custom expression variable526

verifying, See authentication server

User-Name RADIUS attribute212, 219

User-Password RADIUS attribute219

V

valid roles, defined119, 249

version monitoring virus signatures and firewall version on client284

virtual

hostname

configuring330

port

associating with a certificate362

defined361

enabling355

virtual adapter, configuring with IP pool policies85

virtual routers in Infranet Enforcer, configuring source interface policies for89

virus signatures

checking age290

VLAN

policy, configuring for 802.1X enforcement151

port definition334

VPN policy, configuring on Infranet Enforcer32, 75, 77
W
W3C log file format423
Web, peak usage statistics430
web-based port authentication162
WELF log file format423
wildcard certificate, defined362
Windows Internet Naming Service server, configuring330
Windows NT authentication, See authentication server,
Active Directory
WINS
   for external port331
   server, configuring330
wireless settings, Odyssey Access Client, initial
   configuration of49

X
XML
code sample392
   Import/Export
      local user accounts402
      network settings399
      resource policies402
      sign-in pages401
   import/export
      element sequence394
      empty tag392
      end tag392
      instance elements391
      instance file391
      mapping XML instance to UI395
      referential integrity394
      start tag392
      namespaces393
      passwords393

Z
zero downtime upgrade overview339