

RSA Authentication Manager 7.1 Administrator's Guide



The Security Division of EMC

Contact Information

Go to the RSA corporate web site for regional Customer Support telephone and fax numbers: www.rsa.com

Trademarks

RSA and the RSA logo are registered trademarks of RSA Security Inc. in the United States and/or other countries. For the most up-to-date listing of RSA trademarks, go to www.rsa.com/legal/trademarks_list.pdf. EMC is a registered trademark of EMC Corporation. All other goods and/or services mentioned are trademarks of their respective companies.

License agreement

This software and the associated documentation are proprietary and confidential to RSA, are furnished under license, and may be used and copied only in accordance with the terms of such license and with the inclusion of the copyright notice below. This software and the documentation, and any copies thereof, may not be provided or otherwise made available to any other person.

No title to or ownership of the software or documentation or any intellectual property rights thereto is hereby transferred. Any unauthorized use or reproduction of this software and the documentation may be subject to civil and/or criminal liability.

This software is subject to change without notice and should not be construed as a commitment by RSA.

Third-party licenses

This product may include software developed by parties other than RSA. The text of the license agreements applicable to third-party software in this product may be viewed in the [thirdpartylicenses.html](#) files.

Note on encryption technologies

This product may contain encryption technology. Many countries prohibit or restrict the use, import, or export of encryption technologies, and current use, import, and export regulations should be followed when using, importing or exporting this product.

Distribution

Limit distribution of this document to trusted personnel.

RSA notice

The RC5™ Block Encryption Algorithm With Data-Dependent Rotations is protected by U.S. Patent #5,724,428 and #5,835,600.

Contents

Preface	13
About This Guide.....	13
RSA Authentication Manager Documentation	13
Related Documentation.....	14
Getting Support and Service	14
Before You Call Customer Support.....	14
Chapter 1: Preparing RSA Authentication Manager for Administration	15
Logging On to the RSA Security Console.....	15
Logging On to the RSA Operations Console.....	16
Creating Your Organizational Hierarchy.....	17
Creating Realms.....	17
Creating Security Domains.....	18
Accessing Users from an LDAP Identity Source.....	21
Setting Up SSL for LDAP	23
Adding Custom Attributes to User Records	23
Re-Indexing your Directory for Improved Searches	24
Adding an Identity Source in RSA Authentication Manager	25
Linking an Identity Source to a Realm	28
Verifying the LDAP Identity Source.....	29
Editing Users in LDAP with the RSA Security Console.....	29
Identifying Orphaned LDAP Users	29
Removing an Identity Source	29
Adding Users to the Internal Database Using the RSA Security Console.....	30
Migrating Users from an Existing RSA ACE/Server or RSA Authentication Manager Deployment.....	31
Adding Administrators.....	32
Predefined Administrative Roles	33
Creating Administrative Roles.....	37
Assigning Administrative Roles	42
Organizing Users for Administration.....	44
Protecting the RSA Security Console	45
Chapter 2: Configuring Authentication Policies	47
Setting Password Requirements.....	47
Requiring Use of System-Generated Passwords	48
Requiring Periodic Password Changes	49
Restricting Reuse of Old Passwords.....	49
Limiting Password Lengths	50
Using an Excluded Words Dictionary	50
Setting Password Character Requirements.....	51



- Setting Token Usage Requirements 51
 - Limiting the Number of Incorrect Passcodes Allowed..... 52
 - Setting Tokencode Ranges for Event-Based Tokens..... 53
 - Requiring Periodic RSA SecurID PIN Changes..... 53
 - Restricting Reuse of Old PINs..... 54
 - Limiting RSA SecurID PIN Length..... 54
 - Setting RSA SecurID PIN Character Requirements..... 55
 - Requiring Periodic Fixed Passcode Changes..... 55
 - Restricting Reuse of Old Fixed Passcodes 56
 - Limiting Fixed Passcode Length 56
 - Setting Fixed Passcode Character Requirements..... 57
 - Setting Emergency Access Code Formats 57
- Locking Users Out of the System 57
 - Locking Out Users After a Specified Number of Logon Attempts 58
- Setting Offline Authentication Requirements..... 59
 - Integrating Your Windows Password with RSA SecurID..... 61
 - Setting Minimum Online Passcode Lengths..... 61
 - Handling Offline Authentication with Devices that Do Not Meet Security Recommendations..... 62
 - Setting Offline Emergency Codes 62
 - Refreshing Users' Supplies of Offline Authentication Data 63
- Setting Self-Service Troubleshooting Requirements..... 63
- Chapter 3: Protecting Network Resources with RSA SecurID 65**
 - Overview of RSA SecurID Authentication..... 65
 - Installing Authentication Agent Software on the Resource You Want to Protect..... 66
 - Creating an RSA Agent Record Using the RSA Security Console 66
 - Allowing Agents to Automatically Add Authentication Agent Records 68
 - Creating and Installing the RSA Authentication Manager Configuration File..... 71
 - Specifying Where Agents Send Authentication Requests 72
 - Using Authentication Agents to Restrict User Access..... 73
 - Granting Access to Restricted Agents Using User Groups 74
 - Setting Restricted Access Times for User Groups..... 75
 - Deploying Tokens to Users..... 75
 - Importing Hardware and Software Token Records 78
 - Transferring Hardware and Software Token Records to Other Security Domains ... 79
 - Assigning and Unassigning Hardware and Software Tokens..... 79
 - Distributing Hardware Tokens to Users 80
 - Distributing Software Tokens to Users..... 81

Delivering Tokencodes Using Text Message or E-mail	85
Configuring RSA Authentication Manager for On-Demand Authentication	87
Changing the SMS Service Provider	89
Enabling Users for On-Demand Authentication.....	89
Setting PINs for On-Demand Tokencodes	90
Preventing and Handling User Authentication Problems	91
Educating Users About Security Responsibilities	91
Chapter 4: Administering Users	93
Enabling and Disabling Users	93
Assisting Users Who Have Been Locked Out of the System	94
Assisting Users Whose Tokens Are Lost, Stolen, Damaged, or Expired	95
Providing Users with Temporary Emergency Access	96
Providing Temporary Emergency Access for Online Authentication	97
Providing Temporary Emergency Access for Offline Authentication	99
Replacing Tokens.....	101
Enabling and Disabling Tokens	102
Resynchronizing Tokens.....	103
Clearing PINs.....	104
Requiring Users to Change Their PINs.....	105
Providing Users with Fixed Passcodes	106
Clearing Incorrect Passcodes	106
Designating a Default Shell for UNIX Users.....	106
Assigning Logon Aliases	107
Updating Phone Numbers and E-mail Addresses for On-Demand Tokencodes	107
Granting Access with User Groups.....	107
Chapter 5: Administering RSA Authentication Manager	109
Modifying Administrator Permissions.....	109
Displaying All of the Administrative Roles with View or None Permission for a Specific Attribute.....	111
Using the Store Utility to Display Administrative Roles with a View or None Permission for a Specific Attribute.....	112
Securing Communications Between the Authentication Agent and RSA Authentication Manager	112
Refreshing the Node Secret Using the Node Secret Load Utility.....	113
Determining Limits on Administrative Sessions	114
Limiting the Number of Concurrent Administrative Sessions	115
Limiting the Length of Administrative Sessions	116
Limiting Periods of Inactivity Allowed for Administrative Sessions.....	116
Viewing and Closing Administrative Sessions.....	117
Configuring the System Cache for Improved Performance.....	118
Updating Identity Source Attributes	119



- Adding and Updating Token Attributes..... 120
- Adding Additional Software Token Device Types to Your Deployment..... 120
- Configuring RSA Security Console Preferences 121
- Licenses..... 121
- Chapter 6: Administering RSA Credential Manager 125**
 - Overview of RSA Credential Manager..... 125
 - Licensing Options 125
 - RSA Self-Service Console..... 126
 - RSA Security Console 127
 - Self-Service..... 128
 - Provisioning 128
 - Configuring RSA Credential Manager 128
 - Configuring the Authentication Method for the RSA Self-Service Console..... 129
 - Selecting Identity Sources for Enrollment..... 129
 - Selecting Security Domains for Enrollment 131
 - Customizing User Profiles for Enrollment 132
 - Configuring Self-Service Troubleshooting for the RSA Self-Service Console 132
 - Configuring Provisioning..... 134
 - Configuring Workflows for Requests..... 134
 - Adding Administrators 135
 - Selecting User Groups 135
 - Configuring E-mail..... 136
 - Selecting RSA SecurID Tokens..... 137
 - Approving Requests..... 140
 - Creating Multiple Requests and Archiving Requests 141
 - Distributing Tokens 141
 - Selecting the On-Demand Tokencode Service 142
 - Assisting Users..... 143
 - Logging On to the RSA Self-Service Console..... 143
 - Customizing Features of RSA Credential Manager..... 144
- Chapter 7: Administering Trusted Realms 145**
 - Overview of Trusted Realm Deployments..... 145
 - Creating Trusted Realm Relationships 148
 - Creating and Configuring a Trust 149
 - Editing a Trust 152
 - Adding and Enabling Authentication Agents for Trusted Realm Authentication 152
 - Enabling an Authentication Agent in the Trusted Realm 153
 - Adding a Duplicate Authentication Agent..... 154
 - Managing Trusted Users and Trusted User Groups..... 155
 - Creating Trusted Users 156
 - Creating Trusted User Groups 156
 - Allowing Trusted Users to Authenticate Using RSA RADIUS 157

Chapter 8: Managing RSA RADIUS	159
Overview of RSA RADIUS	159
RSA RADIUS Supports Secure Network Access	160
How You Manage RSA RADIUS	160
How RSA RADIUS Helps Enforce Access Control	161
Other Attribute Types Provide Flexibility	165
How RSA RADIUS Maintains Secure Communications	166
Managing User Access.....	167
Managing Profiles.....	167
Managing Profile Assignments.....	168
Managing RADIUS User Attributes.....	169
Managing RADIUS Clients	169
Managing RSA RADIUS Servers.....	170
Starting and Stopping RSA RADIUS Servers	170
Adding a New RSA RADIUS Server	170
List or Delete Existing RSA RADIUS Server Entries.....	170
View or Edit Existing RSA RADIUS Server Properties	171
Managing Replication.....	171
Manage EAP-POTP Configuration	172
Monitoring System Usage.....	173
Viewing RSA RADIUS Usage Statistics.....	173
View RADIUS Server Accounting Statistics	174
View RADIUS Server's Client Authentication and Accounting Statistics.....	176
Choosing Accounting Attributes and Administrator Actions to Record	177
Displaying the Authentication Log Files	178
Configuring the Log Retention Period.....	182
Using the Server Log File.....	182
Using the Accounting Log File.....	183
Maintaining RSA RADIUS Servers	188
Removing an RSA RADIUS Server from Service	188
Backing Up a RADIUS Server	189
Restoring a RADIUS Server.....	189
Promoting an RSA RADIUS Replica Server	191
Modify RSA RADIUS Server Configuration and Dictionary Files	192
Change the IP Address or Name of an RSA RADIUS Server.....	193
Chapter 9: Logging and Reporting	195
Configuring RSA Authentication Manager Logging.....	195
Archiving Log Files	197
Generating Reports	198
Creating Custom Reports.....	198
Running Reports	200
Scheduling Recurring Reports	201
Setting Report Ownership.....	202
Viewing Reports	203

Configuring SNMP	203
RSA Authentication Manager Message IDs	205
Using the Activity Monitor	206
Chapter 10: Disaster Recovery	209
Backing Up and Restoring the Internal Database	209
When to Perform a Backup	209
Prerequisites	210
Performing the Backup	210
Automated Backups	211
Restoring the Database from a Backup	212
Restoring Event-Based Token Data	217
Restoring an Installation for a Standalone Primary Instance	217
Restoring the Installation	217
Detecting a Failed Primary Instance or Replica Instance	219
Determining Why an Instance Might Stop Responding	219
What To Do When a Primary Instance Stops Responding	220
What To Do When a Replica Instance Stops Responding	220
Promoting a Replica Instance to a Primary Instance	221
Promoting a Replica Instance to Recover From a Disaster	221
Promoting a Replica Instance to Migrate the Primary Instance	221
What Happens During Replica Instance Promotion	221
Step 1: Identify the Replica Instance to Be Promoted	222
Step 2: Promote the Selected Replica Instance	223
Step 3: Reattach all Replica Instances to the New Primary Instance	228
Reconfiguring CT-KIP After Promoting a Replica	231
Removing a Replica Instance	231
Reattaching a Demoted Primary Instance	232
Resynchronizing a Diverged Replica Instance	232
Restoring a Super Admin	234
When You Need to Restore the Super Admin	234
Recovering from a Lockout	234
Options for restore-admin	236
Appendix A: Integrating Active Directory Forests	237
Overview of Active Directory Forest Identity Sources	237
Adding an Active Directory Forest as an Identity Source	238
Password Policy Considerations	240
Supporting Groups	240
Mapping Attributes to Active Directory	240

Appendix B: Customizing RSA Credential Manager	243
Customizing E-mail Notifications	243
Guidelines for Customizing E-mail	244
Example of Customized E-mail Template	244
Customizing E-mail Notifications for Proxy Servers	245
Using E-mail Template Tags	247
Conditional Statements in E-mail Templates	250
Customizing Help for the RSA Self-Service Console	252
Customizing Token Graphics	252
Customizing Workflow and Non-Workflow Operations	254
Customizing the Self-Service Console Home Page	254
Appendix C: Managing RSA SecurID Tokens with the Microsoft Management Console (MMC)	255
Overview of the Microsoft Management Console (MMC)	255
Assigning and Unassigning Tokens	258
Disabling and Enabling Tokens	258
Editing User Authentication Attributes	259
Editing Token Properties	259
Replacing Tokens	259
Managing PINs	260
Providing Emergency Access	260
Generating a Temporary Tokencode for Online Authentication	261
Assigning a Temporary Tokencode for Offline Authentication	262
Appendix D: Command Line Utilities	263
Overview	263
Archive Requests Utility	265
Using the Archive Requests Utility	266
Options for archive-ucm-request	267
Collect Product Information Utility	268
Using the Collect Product Information Utility	268
Options for collect-product-info	269
Import PIN Unlocking Key Utility	269
Using the Import PIN Unlocking Key Utility	270
Options for import-puk	270
Manage Backups Utility	271
Using the Manage Backups Utility	271
Transferring the Internal Database from One Machine to Another Machine	272
Restoring the Internal Database in a Replicated Environment	273
Options for manage-backups	274
Manage Batchjob Utility	275
Using the Manage Batchjob Utility	275
Options for manage-batchjob	277

Manage Database Utility.....	277
Using the Manage Database Utility	277
Options for manage-database.....	280
Manage Operations Console Administrators Utility.....	282
Using the Manage Operations Console Administrators Utility	282
Options for manage-oc-administrators	283
Manage Secrets Utility.....	285
Using the Manage Secrets Utility	285
Options for manage-secrets	287
Register Custom Extension Utility	288
Using the Register Custom Extension Utility.....	288
Options for register-custom-extension.....	290
Creating a Custom Extension Property File for Workflow Operations.....	290
Creating a Custom Extension Property File for Non-Workflow Operations.....	296
Set Trace Utility	298
Options for set-trace.....	300
Diagnostic Monitors for set-trace	301
User Groups and Token Bulk Requests Utility.....	302
Using the User Groups and Token Bulk Requests Utility	302
Options for import-bulk-request	303
Creating Input Files for Bulk Requests.....	304
CSV Format for Token Requests Input File	305
CSV Format for User Group Membership Requests Input File	306
Log Files for Bulk Requests	306
PIN and Protection of Distribution Files	307
Verify Archive Log Utility	307
Using the Verify Archive Log Utility	307
Options for verify-archive-log	307
Appendix E: Updating Server IP Addresses and Names.....	309
Update Instance Nodes Utility	309
Using the Update Instance Nodes Utility	309
Options for update-instance-node	310
Changing an IP Address or a Fully Qualified Domain Name on a Standalone Deployment	311
Changing the IP Address or a Fully Qualified Domain Name on a Cluster Deployment	312
Changing an IP Address or a Fully Qualified Domain Name in a Replicated Deployment.....	329

Appendix F: RSA Authentication Manager Message IDs	361
Audit Log Messages.....	361
System Log Messages	363
RSA Authentication Manager Administrative Audit Log Messages.....	373
RSA Authentication Manager Runtime Audit Log Messages	380
RSA Authentication Manager System Log Messages	382
Appendix G: Troubleshooting	391
Common Problems and Resolutions	391
General Troubleshooting Tips	400
Using the Activity Monitor and Log Messages for Troubleshooting	400
Making Sure the RSA Authentication Manager Machine Meets	
Minimum System Requirements.....	407
Supported Browsers.....	411
Configuring Browser Settings for the RSA Security Console, RSA Operations	
Console, and RSA Self-Service Console	411
Assessing the Impact of Firewalls on RSA Authentication Manager	412
Configuring the Cache for Improved Performance	415
Test User Access to Restricted Agent.....	415
User and Token-Related Resolutions.....	416
Unlocking a User	416
Assisting Users with Lost, Stolen, Damaged or Expired Tokens	416
Providing Emergency Access	417
Clearing PINs.....	417
Forcing PIN Changes.....	417
Clearing Incorrect Passcodes	417
Resynchronizing a Token	418
System-Related Resolutions	418
RSA Authentication Manager Does Not Start.....	418
RSA Security Console Does Not Start	419
RSA Authentication Manager Microsoft Management Console Snap-in	
Does Not Start.....	419
RSA Security Console Times Out When Searching for Users	419
Name and IP Address Resolution in RSA Authentication Manager	420
Managing the Node Secret.....	421
Resynchronizing RSA Authentication Manager with	
Coordinated Universal Time.....	421
Updating an Agent Configuration File	422
Reconfiguring CT-KIP After Promoting a Replica Instance.....	422
Changing the IP Address or Hostname of a Server	423
Glossary	425
Index	445

Preface

About This Guide

This guide describes how to administer RSA Authentication Manager. It is intended for administrators and other trusted personnel.

RSA Authentication Manager Documentation

For more information about RSA Authentication Manager, see the following documentation:

Release Notes. Provides information about what is new and changed in this release, as well as workarounds for known issues.

Getting Started. Lists what the kit includes (all media, diskettes, licenses, and documentation), specifies the location of documentation on the DVD or download kit, and lists RSA Customer Support web sites.

Planning Guide. Provides a general understanding of RSA Authentication Manager, its high-level architecture, its features, and deployment information and suggestions.

Installation and Configuration Guide. Describes detailed procedures on how to install and configure RSA Authentication Manager.

Administrator's Guide. Provides information about how to administer users and security policy in RSA Authentication Manager.

Migration Guide. Provides information for users moving from RSA Authentication Manager 6.1 to RSA Authentication Manager 7.1, including changes to terminology and architecture, planning information, and installation procedures.

Developer's Guide. Provides information about developing custom programs using the RSA Authentication Manager application programming interfaces (APIs). Includes an overview of the APIs and Javadoc for Java APIs.

Performance and Scalability Guide. Provides information to help you tune your deployment for optimal performance.

RSA Security Console Help. Describes day-to-day administration tasks performed in the RSA Security Console. To view Help, click the **Help** tab in the Security Console.

RSA Operations Console Help. Describes configuration and setup tasks performed in the RSA Operations Console. To log on to the Operations Console, see "Logging On to the RSA Operations Console" in the *Administrator's Guide*.

RSA Self-Service Console Frequently Asked Questions. Provides answers to frequently asked questions about the RSA Self-Service Console, RSA SecurID two-factor authentication, and RSA SecurID tokens. To view the FAQ, on the **Help** tab in the Self-Service Console, click **Frequently Asked Questions**.

Note: To access the *Developer's Guide* or the *Performance and Scalability Guide*, go to <https://knowledge.rsasecurity.com>. You must have a service agreement to use this site.

Related Documentation

RADIUS Reference Guide. Describes the usage and settings for the initialization files, dictionary files, and configuration files used by RSA RADIUS.

Getting Support and Service

RSA SecurCare Online	https://knowledge.rsasecurity.com
Customer Support Information	www.rsa.com/support
RSA Secured Partner Solutions Directory	www.rsa.com/rsasecured

RSA SecurCare Online offers a knowledgebase that contains answers to common questions and solutions to known problems. It also offers information on new releases, important technical news, and software downloads.

The RSA Secured Partner Solutions Directory provides information about third-party hardware and software products that have been certified to work with RSA products. The directory includes Implementation Guides with step-by-step instructions and other information about interoperation of RSA products with these third-party products.

Before You Call Customer Support

Make sure you have access to the computer running the RSA Authentication Manager software.

Please have the following information available when you call:

- Your RSA License ID. You can find this number on your license distribution media, or in the RSA Security Console by clicking **Setup > Licenses > Manage Existing**, and then clicking **View Installed Licenses**.
- The Authentication Manager software version number. You can find this in the RSA Security Console by clicking **Help > About RSA Security Console > See Software Version Information**.
- The names and versions of the third-party software products that support the Authentication Manager feature on which you are requesting support (operating system, data store, web server, and browser).
- The make and model of the machine on which the problem occurs.

1

Preparing RSA Authentication Manager for Administration

- [Logging On to the RSA Security Console](#)
- [Logging On to the RSA Operations Console](#)
- [Creating Your Organizational Hierarchy](#)
- [Accessing Users from an LDAP Identity Source](#)
- [Adding Users to the Internal Database Using the RSA Security Console](#)
- [Adding Administrators](#)
- [Organizing Users for Administration](#)
- [Protecting the RSA Security Console](#)

Logging On to the RSA Security Console

RSA Authentication Manager includes an administrative user interface called the RSA Security Console. You use the Security Console for most of your day-to-day administrative activities. For example, you use the Security Console to:

- Add and manage users and user groups.
- Add and manage administrators.
- Assign and manage RSA SecurID tokens.
- Create security policies.
- Designate which network resources you want to protect.

When you install Authentication Manager, a default account for the Security Console is automatically created. This account is given Super Admin permissions, meaning that the account can perform all tasks within Authentication Manager. You can select the user name and password for this account when you install Authentication Manager.

Note: The user name and password that you specify at installation are also the initial user name and password for the RSA Operations Console.

To log on to the Security Console, go to the following URL:

`https://<fully qualified domain name>:7004/console-ims`

If this is the first logon after installation, use the user name and password selected during installation. If this is not the first logon, enter the credentials required by the Security Console.

Important: If the Security Console is protected with RSA SecurID, the SecurID PIN is case sensitive.

For more information on protecting the Security Console with RSA SecurID, see [“Protecting the RSA Security Console”](#) on page 45, or see the Security Console Help topic “Configuring Authentication Settings.”

Note: Do not use your Internet browser's Back button to return to previously visited Console pages. Instead, use the Security Console's navigation menus and buttons to navigate.

Logging On to the RSA Operations Console

Authentication Manager includes a second administrative user interface called the RSA Operations Console. You use the Operations Console to configure and set up Authentication Manager. For example, you use the Operations Console to:

- Add and manage identity sources.
- Add and manage instances.
- Disaster recovery.

When you install RSA Authentication Manager, a default account for the Operations Console is automatically created. Although this account is different than the default account created for the Security Console, the Operations Console account initially uses the same user name and password that you specified for the Security Console account.

Important: Initially, the Security Console and Operations Console both use the user name and password you specified at installation. If you change the user name or password for either the Security Console or the Operations Console, the user name and password for the other Console remain unchanged.

To log on to the Operations Console, go to the following URL and enter the password:

`http://<fully qualified domain name>:7071/operations-console`

If you are using an SSL connection, use the following URL instead:

`https://<fully qualified domain name>:7072/operations-console`

You must use the Manage RSA Operations Console utility to add additional Operation Console administrators. For more information on adding administrators, see [“Manage Operations Console Administrators Utility”](#) on page 282.

Note: There are certain operations in the Operations Console that require you to enter password-based Super Admin credentials. If the Super Admin account is set to require a new password at next logon, you must create the new password before proceeding in the Operations Console.

Note: Do not use your Internet browser's Back button to return to previously visited Console pages. Instead, use the Operations Console's navigation menus and buttons to navigate.

Creating Your Organizational Hierarchy

The organizational hierarchy of your Authentication Manager deployment is made up of realms and security domains. Create this hierarchy to help you organize and administer your Authentication Manager deployment.

Creating Realms

A realm is an independent organizational unit that contains all objects in your deployment, such as users and user groups, administrative roles, tokens, and password, lockout, and token policies. Realms allow you to divide data and administration within your deployment.

When you install Authentication Manager, a default realm is automatically created. You can set up your entire organizational hierarchy within this one realm, or you can create additional realms.

Note: You can only have multiple realms if your license allows it. For more information on licensing, see [“Licenses”](#) on page 121.

Know the following about realms:

- Each realm has its own set of objects such as users, user groups, tokens, and authentication agents.
- You cannot transfer objects such as users, user groups, or tokens between realms.
- Users in one realm cannot use an authentication agent in another realm to authenticate.
- You cannot administer two different realms at the same time. Administer each realm in its own Security Console session.

For example, assume that you have two realms, Realm A and Realm B. To add users to Realm A, you must be logged on to Realm A. To add users to Realm B, you must log off of Realm A and then log on to Realm B.

Users and user groups managed by a realm are stored in identity sources. An identity source can be:

- An external LDAP directory
- The Authentication Manager internal database

Note: Each realm may be associated with multiple identity sources, but each identity source may only be associated with a single realm.

See [“Accessing Users from an LDAP Identity Source”](#) on page 21.

You can create as many realms as your organization needs. Typically, an organization needs very few realms, and many only use the default realm.

You might choose to add an additional realm if your organization has a separate subsidiary. In such a case, the parent company and subsidiary each have their own realm within the parent company's Authentication Manager deployment. Users, user groups, tokens, and other objects in each realm remain separate.

Administrators can only manage the realm in which their user record is stored. They cannot manage multiple realms. Super Admins are the only exception to this. Super Admins can manage all realms in the deployment.

If your organizational needs require you to move users, user groups, tokens, and other objects between organizational units—departments within your company, for example—create a security domain hierarchy, rather than multiple realms. Multiple security domains allow you more flexibility to reorganize your deployment than multiple realms. See the following section, [“Creating Security Domains.”](#)

Use the Security Console to create and manage realms. In a deployment with multiple realms, the Super Admin is prompted at logon to decide which realm he or she wants to log on to. To switch between realms, the Super Admin must log off, and then log on to the other realm. All other administrators log on to the realm where they are managed.

To add a new realm:

1. Click **Administration > Realms > Add New**.
2. In the **Realm Name** field, enter a name for the new realm.
3. From the list of available identity sources, select the identity sources that you want to link with the realm, and click the right arrow.
4. Click **Save**.

Creating Security Domains

Security domains represent areas of administrative responsibility. All Authentication Manager objects are managed by a security domain. Security domains allow you to:

- Organize and manage your users.
- Enforce system policies.
- Limit the scope of administrators' control by limiting the security domains to which they have access.

When a new realm is created—either automatically when you install Authentication Manager, or by an administrator—a top-level security domain is automatically created in the realm. The top-level security domain is assigned the same name as the realm.

Note: You cannot edit the name of the top-level security domain.

You can create as many security domains as you need. When you create a security domain, it is nested within another security domain called the parent security domain. You can nest the security domains to create an administrative hierarchy.

By default, all users and tokens are managed in the top-level security domain. You can transfer users and SecurID tokens from the top-level security domain to other security domains within the realm.

For example, you can create separate security domains for each department, such as Finance, Research and Development (R&D), and Human Resources (HR), and then move users and user groups from each department into the corresponding security domain. To manage users in a given security domain, an administrator must have permission to manage that security domain.

Know the following about security domains:

- Security domains are organized in a hierarchy within a realm. Create as many security domains as your organization requires.
- Security domains are often created to mirror the departmental structure or the geographic locations of an organization.

You also use security domains to enforce system policies. Policies control various aspects of a user's interaction with Authentication Manager, such as RSA SecurID PIN lifetime and format, fixed passcode lifetime and format, password length, format, and frequency of change.

The following policies are assigned to security domains:

- Password policies
- Token policies
- Lockout policies
- Offline authentication policies
- Self-service troubleshooting policies

You can use the default policy, or create a custom policy, for each policy type, for each security domain. See the following sections:

- [“Setting Password Requirements”](#) on page 47
- [“Setting Token Usage Requirements”](#) on page 51
- [“Locking Users Out of the System”](#) on page 57
- [“Setting Offline Authentication Requirements”](#) on page 59
- [“Setting Self-Service Troubleshooting Requirements”](#) on page 63

When you create a new security domain, default policies are automatically assigned. You can optionally assign a custom policy to the new security domain. If you assign the default policy to a security domain, whatever policy designated as the default is automatically assigned to the security domain. When a new policy is designated as the default, the new default is automatically assigned to the security domain.

For example, suppose you create a custom policy named “Finance Token Policy” and designate it as the default token policy. Finance Token Policy is automatically assigned to all new security domains, and to all security domains configured to use the default token policy. Then, a few months later you create another custom policy named “Miller and Strauss Token Policy,” and designate it as the default token policy. The Miller and Strauss Token Policy is then used by all security domains configured to use the default token policy, and will automatically be applied to all new security domains.

Note that the policy assigned to a lower-level security domain is not inherited from upper-level security domains. New security domains are assigned the default policy regardless of which policy is assigned to security domains above them in the hierarchy. For example, if the top-level security domain is assigned a custom policy, lower-level security domains are still assigned the default policy.

To add a security domain:

1. Click **Administration > Security Domains > Add New**.
2. In the **Security Domain Name** field, enter a name for the new security domain.
3. From the **Parent** drop-down list, select the parent security domain of the new security domain. The parent security domain is the upper-level security domain in which you want the new security domain to exist.
4. From the **Password Policy** drop-down list, assign a password policy to the security domain. Password policies establish characteristics such as the required length of passwords, characters, and words whose use in passwords is restricted.
5. From the **Emergency Authentication Policy** drop-down list, assign an emergency authentication policy to the security domain. Emergency authentication policies allow you to designate which authentication method (security questions, password, or none), that a user can use to access RSA Credential Manager in the event that their primary authentication method fails. This policy applies to Credential Manager users only.
6. From the **Lockout Policy** drop-down list, assign a lockout policy to the security domain. Lockout policies allow you to lock out users who have made too many unsuccessful logon attempts. Users who are locked out cannot authenticate and cannot access protected resources.
7. From the **Default Authentication Grade** drop-down list, assign the default authentication grade to the security domain.

For information about using Authentication Grades with Authentication Manager, see the Security Console Help topic “Use Authentication Grades with Authentication Manager.”

8. From the **SecurID Token Policy** drop-down list, assign a SecurID token policy to the security domain. Token policies determine RSA SecurID PIN lifetime and format, and fixed passcode lifetime and format. The token policy also determines how to handle users or unauthorized people who enter a series of incorrect passcodes.
9. From the **Offline Authentication Policy** drop-down list, select an offline authentication policy for the security domain. Offline authentication policies define how users authenticate when they are not connected to the network.
10. Click **Save**.

Accessing Users from an LDAP Identity Source

Authentication Manager can use your existing Active Directory and Sun Java System Directory Server as sources of user and user group data, rather than requiring you to manually enter user data into the Authentication Manager internal database.

Directories integrated with Authentication Manager are called identity sources. When you integrate the identity source, you choose whether you want Authentication Manager to have read-only access or read/write access. If you choose:

Read-only access. This is the recommended setting. Authentication Manager only reads data from your directory.

Note: Active Directory Global Catalogs are always read-only.

Read/write access. You can use the Security Console to make changes to user and user group data stored in the directory.

In both cases, Authentication Manager data, such as token, realm, and security domain information, is stored in the internal database, and not in your directory.

Note: When you use your Sun Java System Directory Server or Active Directory tools to make changes to the user data in your directory, because of the system cache, the changes are not immediately visible through the Security Console. See [“Configuring the System Cache for Improved Performance”](#) on page 118.

You may add as many identity sources to Authentication Manager as you need. If you add more than one identity source to Authentication Manager, remember:

- You can link multiple identity sources to the same realm.
- Each identity source may only be linked to one realm.

If you link multiple identity sources to the same realm, consider how to handle situations in which users with the same User ID exist in each identity source, for example, jsmith.

Important: If two users with the same user name attempt to access the same protected resource, an authentication failure can result.

When the same User IDs are present in multiple identity sources, you have the following options:

- Map the User ID to another field where there are no duplicate values. For example, if your identity source is Active Directory, you might be able to map to the UPN field.
- Create an alias within Authentication Manager, which allows the user to log on under a different User ID. See the Security Console Help topic “Manage User Authentication Attributes.”
- Change one of the User IDs in your identity source so that both User IDs become unique. This option may not be practical if the User ID is used for other applications.
- Assign tokens to only one of the users with the non-unique User ID. This option is not practical if tokens must be assigned to more than one user with the non-unique User ID.
- To allow administrators with duplicate User IDs to log on to the Security Console, select **Non-Unique User IDs** on the Authentication Methods Configuration page. When administrators with duplicate User IDs log on, they must identify themselves by specifying to which identity source they belong. To set this option, see the Security Console Help topic “Configure Authentication Settings.”

Note: If your Authentication Manager deployment has multiple instances, identity source data, such as users and user groups, might not be immediately visible through the replica instance. This delay is due to the cache refresh interval. Data should replicate within 10 minutes.

The following steps provide a high-level overview of the tasks you must perform to add an identity source:

1. Use the Operations Console to set up the SSL connections. This step is required if your identity source is Active Directory with read-write access. This step is optional for Active Directory with read-only access, and Sun Java System Directory Server.
2. Optional. Use the Security Console to create custom user attributes. Custom user attributes contain information tailored to your organization. Create custom attributes required by other applications within your organization.
For example, if you have applications that require the UPN field in Active Directory, use the Add New Identity Attribute Definition page in the Security Console to manually create this field in Authentication Manager.
3. Optional. If you are using Sun Java System Directory Server, RSA recommends that you reindex the directory and increase the cache size. If you do not do this, searches will take longer and will use a larger amount of disk space.

4. Use the Operations Console to add the identity source. When you add the identity source, you need to configure the following:
 - General identity source information.
 - LDAP connection.
 - Global Catalog (Active Directory only).
 - Map user attributes. This can include standard or custom user attributes.

Important: There are special considerations when mapping attributes for Active Directory. For more information, see [“Mapping Attributes to Active Directory”](#) on page 240.

For more information on adding custom attributes, see [“Adding Custom Attributes to User Records”](#) on page 23.

5. Use the Security Console to link the identity source to a realm so you can assign tokens to the users stored in the identity source.
6. Use the Security Console to verify the identity source.

Important: There are special considerations when using an Active Directory Global Catalog as your authoritative identity source. For example, you must integrate both the Global Catalog and all Active Directories that replicate data to the Global Catalog. Before adding the identity source, see Appendix A, [“Integrating Active Directory Forests”](#) on page 237.

Setting Up SSL for LDAP

If you are using Active Directory with read-write access, you must configure the SSL connection before adding the identity source. This step is optional for Active Directory with read-only access and Sun Java System Directory Server.

Note: Your directory server must already be configured for SSL connections and have ready access to the CA certificate. If your system does not meet these requirements, see your directory server documentation for instructions on setting up SSL.

Use the Operations Console to add the SSL certificate. For instructions, see the Operations Console Help topic “Add Identity Source SSL Certificates.”

Adding Custom Attributes to User Records

Mapping the fields in your identity source to the fields in the Security Console allows you to use the Security Console to view user and user group data stored in your identity source.

In addition to the default fields contained in a user record, you can define custom attributes, called identity attribute definitions, that contain information tailored to your organization.

For example, you might decide to define an attribute named “Department” in which you can enter a user’s department name, such as “HR” or “Finance.”

When you define a custom attribute, the attribute definition is stored in the Authentication Manager internal database. By default, the attribute value is also stored in the internal database.

You can, however, map custom attributes to your external identity sources, which allows Authentication Manager to read attribute values from the directory. You map custom attributes when creating an identity source in the Operations Console.

Use the Security Console to add identity attribute definitions. For instructions, see the Security Console Help topic “Add Identity Attribute Definitions.”

Re-Indexing your Directory for Improved Searches

If you are using Sun Java System Directory Server, RSA recommends that you re-index the directory and increase the cache size. Re-indexing the directory and increasing the cache size increases the efficiency of your searches.

Important: If you do not do this, searches will take longer and will use a larger amount of disk space.

To re-index the directory:

1. In the Sun Java System Directory Server console, select the Directory Server, and click **Open**.
2. In the Sun Java System Directory Server, click the **Configuration** tab.
3. Expand the Data node and select the suffix that you want to index.
4. Click the **Indexes** tab.
5. Under Additional Indexes, select all of the checkboxes for **uid**, including **Approximate**, **Equality**, **Presence** and **Substring**.
6. Click **Save**.
7. Click **Reindex Suffix**.
8. Click **Check All**.
9. Click **OK** to begin re-indexing.
10. Click **Yes** to confirm.
11. When re-indexing completes, click **Close**.

After re-indexing the directory, RSA also recommends that you increase the cache size for more efficient searching.

To increase the cache size:

1. In the Sun Java System Directory Server console, click the **Configuration** tab.
2. Click the **Performance** node.
3. Click the **Caching** tab.
4. Change the Database cache size to 100 MB.

5. Click **Save**.
6. Click **OK**.
7. On the **Tasks** tab, click **Restart Directory Server**.

Adding an Identity Source in RSA Authentication Manager

Use the Operations Console to add an identity source to the system, and to configure identity source options. Adding an identity source allows you to use the Security Console to view and manage users in your existing directory servers.

To add an identity source record with the Operations Console:

Important: You must use the Operations Console on the primary instance to perform this task.

1. Click **Deployment Configuration > Identity Sources > Add New**.
2. Enter your Security Console user name and password. The user name to whom the credential is assigned must be assigned the Super Admin role.
3. In the Identity Source Basics section, do the following:
 - a. In the **Identity Source Name** field, enter the name of the identity source.
 - b. In the **Type** field, select the type of identity source you want to add.
 - c. In the **Notes** field, enter any important information about the identity source.
4. In the Directory Connection section, do the following:
 - a. In the **Directory URL** field, enter the URL of the new identity source.
 - b. If you are using the standard SSL-LDAP port 636, specify the value as `ldaps://hostname/`. For any other port, you must also specify the port number, for example, `ldaps://hostname:port/`.
 - c. For Active Directory identity sources, RSA recommends that you use an SSL connection because it is required for password management.

Note: If you are adding an Active Directory Global Catalog, the default port numbers are 3268 for a non-SSL connection, and 3269 for an SSL connection.

- d. Optional. In the **Directory Failover URL** field, enter the URL of a failover identity source.
- e. The system connects to the failover LDAP if the connection with the primary LDAP directory fails.
- f. In the **Directory User ID** field, enter the LDAP administrator's User ID.

- g. In the **Directory Password** field, enter the LDAP administrator's password.

Important: Do not let the directory password expire. If the directory password expires, the connection to the directory will fail. If the password does expire, reset it in Active Directory using the **Active Directory Users and Computers** tool, and reset it on the Identity Source page in the Operations Console.

- h. Click **Test Local Connection** to make sure that the system can successfully connect to the LDAP directory.

Note: You must configure a connection for the primary instance and each replica instance in your deployment.

5. Click **Next**.
6. In the Directory Settings section, do the following:
 - a. In the **User Base DN** field, enter the base DN for directory user definitions.
 - b. In the **User Group Base DN** field, enter the base DN for directory user group definitions.
 - c. Optional. Select **Read-only** to prevent administrators from using the Security Console to edit the identity source.
 - d. Optional. In the **Search Results Time-out** field, limit the amount of time a search is allowed to continue. If searches for users or groups are timing out on the directory server, either extend this time, or narrow individual search results. For example, instead of Last Name = *, try Last Name = G*.
 - e. In the **User Account Enabled State** drop-down menu, specify whether the system checks an external directory or the internal database to determine whether a user is enabled.
 - f. Optional. Select **Validate Against Schema** if you want the mapping of identity attribute definitions to the LDAP schema to be validated when identity attribute definitions are created or modified.
 - g. Use the **Manage Password** button to decide if you want to require users in the identity source to have passwords.
7. If the identity source is an Active Directory, in the Active Directory Options section, do one of the following:
 - If the identity source you are adding is a Global Catalog, select **Global Catalog**.
 - If the identity source is not a Global Catalog, select whether to authenticate users to this identity source, or select a Global Catalog to which you want users to authenticate.

- In the **Default Group Type**, select a default group type for the identity source. All user groups created in the new Active Directory identity source are assigned the default group type. You can edit the user group type if necessary.

Note: If the identity source uses an Active Directory forest, or if a Global Catalog must authenticate against restricted agents, the group type must be Universal.

8. In the Directory Configuration - Users section, do the following:
 - a. Enter the directory attributes that you want to map to user attributes. For example, First Name might map to givenname, and Last Name might map to sn.
 - b. Use the **User ID** buttons to specify whether you want to map the User ID to a specific attribute, or to the same attribute to which the E-mail field is mapped. If you choose to map to the same attribute as the E-mail field, the User ID and E-mail fields have the same value.

Important: If you map User ID to **samAccountName** or **userPrincipalName**, you must create an additional custom attribute. For more information, see [“Mapping Attributes to Active Directory”](#) on page 240.

- c. Select **Unique Identifier**. This field helps the Security Console find users whose DNs have changed. This checkbox is selected by default. The default unique identifier for Active Directory is ObjectGUID, and for Sun Java Directory Server is nsUniqueId, however you may edit these identifiers to point to other fields.
 - d. In the **Search Filter** field, enter the filter that specifies how user entries are distinguished in your LDAP directory, such as a filter on the user object class. Any valid LDAP filter for user entries is allowed. For example, (objectclass=inetOrgPerson).
 - e. In the **Search Scope** drop-down box, select the scope of user searches in the LDAP tree.
 - f. In the **RDN Attribute** field, enter the user attribute used for the Relative Distinguished Name.
 - g. In the **Object Classes** field, enter the object class of users that are created or updated using the Security Console. For example, inetOrgPerson,organizationalPerson,person.
9. In the Directory Configuration - User Groups section, do the following:
 - a. Enter the directory attribute that maps to the user group name attribute. For example, User Group Name might map to cn.
 - b. In the **Search Filter** field, enter an LDAP filter that returns only group entries, such as a filter on the group object class. For example, (objectclass=groupOfUniqueNames).
 - c. In the **Search Scope** drop-down box, select the scope of user group searches in the LDAP tree.

- d. In the **RDN Attribute** field, enter the user group attribute used for the Relative Distinguished Name.
 - e. In the **Object Classes** field, enter the object class of users that are created or updated using the Security Console. For example, inetOrgPerson,organizationalPerson,person.
 - f. In the **Membership Attribute** field, enter the attribute that contains the DNs of all of the users and user groups that are members of a user group.
 - g. Select **User MemberOf Attribute** to enable the system to use the MemberOf Attribute for resolving membership queries.
 - h. In the **MemberOf Attribute** field, enter the attribute of users and user groups that contains the DNs of the user groups to which they belong.
10. Click **Save**.

Note: If you are logged on to the Security Console, you must log off and log back on to view the new identity source. You do not have to log off if your browser is Internet Explorer 7 or later, or Mozilla Firefox 2.0 or later, and the Operations Console and Security Console are on different tabs within the same browser session.

After adding the identity source, use the Security Console to link the identity source to a realm.

Linking an Identity Source to a Realm

To enable administration of an identity source, you must link it to a realm for administration.

Note: Do not configure multiple identity sources with overlapping scope in the same realm or across realms. For example, make sure that two identity sources do not point to the same base DNs for user and group searches. For Active Directory, a runtime identity source can have overlapping scope with the corresponding administrative source(s), but two runtime identity sources cannot have overlapping scope.

To link the new identity source to a realm:

1. Log on to the Security Console as Super Admin.
2. Click **Setup > Administration > Realms > Manage Existing**.

Note: You can also create a new realm.

3. Use the search fields to find the realm with which you want to work.
4. Click on the realm with which you want to work.
5. From the Context menu, click **Edit**.

6. From the list of available identity sources, select the identity sources that you want to link with the realm, and click the right arrow.
The names displayed are the values that you entered when you added the identity source (see the previous section, "[Adding an Identity Source in RSA Authentication Manager](#)").
7. Click **Save**.

Verifying the LDAP Identity Source

To verify that you have successfully added an identity source, you can view the particular users and groups from the LDAP identity source through the Security Console.

To verify the LDAP identity source:

1. Click **Identity > Users > Manage Existing**.
2. Use the search fields to find the appropriate realm and identity source, and click **Search**.
3. View the list of users from your LDAP identity source.

Editing Users in LDAP with the RSA Security Console

By default, Authentication Manager access to external LDAP identity sources is read-only. You can, however, configure your identity sources for read/write access. When you do this, administrators with the proper permissions can use the Security Console to add and edit users and user groups in the identity source.

You configure an identity source as read/write when you use the Operations Console to define the identity source in Authentication Manager. The read/write configuration can also be edited using the Operations Console.

Identifying Orphaned LDAP Users

After an LDAP user's DN has been changed, Authentication Manager can no longer retrieve that user. You need to periodically run the Orphaned Data Report to view the External Unique Identifiers (EXUID) for users who are no longer associated with their original DNs. EXUID is a directory-specific unique identifier that Authentication Manager can use to find users that have moved in the directory information tree of the directory server.

Removing an Identity Source

CAUTION: Removing an identity source is an irreversible process and can result in the loss of user-token associations.

Deleting an Identity Source

Use the Operations Console to remove an identity source. For instructions, see the Operations Console Help topic "Delete Identity Sources."

Unlinking an Identity Source from a Realm

To disable an identity source, it must be unlinked from a realm. Use the Security Console to unlink an identity source from a realm. For instructions, see the Security Console Help topic “Unlink Realms from Identity Sources.”

Adding Users to the Internal Database Using the RSA Security Console

If your organization does not have an LDAP directory, or is setting up an Authentication Manager pilot with a small subset of users, you can store users in the Authentication Manager internal database.

Use the Security Console to add users to the internal database.

To add users to the internal database:

1. Click **Identity > Users > Add New**.
2. In the Administrative Control section, do the following:
 - a. From the **Identity Source** drop-down list, select the internal database.
 - b. From the **Security Domain** drop-down list, select the security domain where you want the user to be managed. The user is managed by administrators whose administrative scope includes the security domain you select.
3. In the User Basics section, do the following:
 - a. In the **Last Name** field, enter the last name of the user.
 - b. In the **User ID** field, enter the User ID for the user. The user ID cannot contain multibyte characters.

Note: Do not create a user ID that is longer than 48 characters.

- c. Optional. In the **Certificate DN** field, enter the certificate DN of the user. The certificate DN must match the subject line of the certificate issued to the user for authentication.

4. In the Identity Source Password section, do the following:

Note: This password is not used for authenticating through authentication agents.

- a. In the **Password** field, enter a password for the user. Password requirements are determined by the password policy assigned to the security domain where the user is managed. This is the user's identity source password, which may be different from alternate passwords provided by applications.
 - b. In the **Confirm Password** field, enter the same password that you entered in the Password field.
 - c. Optional. Select **Force Password Change** if you want to force the user to change his or her password the next time the user logs on. You might select this checkbox, for example, if you assign a standard password to all new users, which you want them to change when they start using the system.
5. In the Account Information section, do the following:
 - a. From the **Account Starts** drop-down lists, select the date and time you want the user's account to become active. The time zone is determined by local system time.
 - b. From the **Account Expires** drop-down lists, select the date and time you want the user's account to expire, or configure the account with no expiration date. The time zone is determined by local system time.
 - c. Optional. Select **Disabled** if you want to disable the new account.
 6. Click **Save**.

After you add users to Authentication Manager, you can use the Security Console to manage them and assign tokens to them.

You can also add users to the internal database, even if you are using an LDAP directory as your primary identity source. You would likely not want to duplicate all users from your directory in the internal database, but you might choose to create a subset, or you might add a set of users different from those in your directory. For example, you might store a group of temporary contractors or a specific group of administrators in the internal database.

Migrating Users from an Existing RSA ACE/Server or RSA Authentication Manager Deployment

For information on migrating users and tokens from an existing deployment of RSA Authentication Manager 6.1 to an RSA Authentication Manager 7.1 deployment, see the *Migration Guide*.

Adding Administrators

Administrators manage all aspects of your Authentication Manager deployment, such as users, tokens, and security domains. Each administrator is assigned an administrative role that has its own set of administrative privileges and areas of responsibility.

Administrative roles control what an administrator can manage. Once an administrative role is assigned to a user, the user becomes an administrator.

You can choose from two types of administrative roles:

Predefined. Authentication Manager provides you with a set of predefined roles. You can assign predefined roles in their default form, or you can edit the permissions assigned to the roles.

For example, if you do not want administrators with the Help Desk role to be able to view authentication agents, you can use the Security Console to edit that role and remove that permission.

Custom. Authentication Manager allows you to create custom roles with different privileges and areas of administrative responsibility, depending on your organization's needs. You can create as many administrators as your deployment needs.

For example, suppose your organizational hierarchy is divided into three security domains: HR, R&D, and Finance. Because financial data is sensitive, you might create a custom administrator who can run and view finance reports in the Finance security domain.

You can only create and assign roles with privileges equal to or less than those granted by your own administrative role. That is, you cannot grant or assign privileges that you do not have.

For example, if your administrative role only allows you to add, edit and delete users, and create and assign administrative roles, you cannot create or assign a role that also grants permission to assign tokens to users.

Likewise, you cannot edit an existing administrative role that has more permissions than your role.

Once you have decided what roles you need for your deployment, and added any custom roles you require, you can assign the roles to your administrators.

The following sections cover these tasks:

- Predefined administrative roles.
- Creating administrative roles.
- Assigning roles to administrators.

Predefined Administrative Roles

The predefined administrative roles are described in the following sections.

Super Admin

The Super Admin role is a predefined administrative role and is the only role with full administrative permission in all realms and security domains in your deployment. Use it to create other administrators, and to create your realm and security domain hierarchy.

You can assign the Super Admin role to as many administrators as you want, but because of the broad scope and wide array of permissions of the role of Super Admin, RSA recommends that you only assign it to the most trusted administrators.

Note: The Super Admin role can only be managed by other Super Admins.

The number of Super Admins is based on the needs of your organization. You assign the Super Admin role in the same way that you assign all of the other administrative roles. For information, see [“Assigning Administrative Roles”](#) on page 42.

RSA recommends that you have at least two administrators who are Super Admins. This ensures that you have full administrative control in situations where a Super Admin leaves for vacation or some other extended absence.

For example, if your organization has locations in Boston, New York, and San Jose, you might choose to have four Super Admins: two in Boston, and one each in New York and San Jose. This arrangement allows for a vacation or extended-leave backup for the Super Admins in the Boston site, where most of the system management occurs. It also allows management of the deployment to occur from the New York or San Jose site if the Boston headquarters loses connectivity, or is otherwise unable to manage the deployment.

No one, including the administrators assigned the Super Admin role, can modify the Super Admin role.

The following occurrences can leave your deployment with no Super Admin:

- The only user assigned to the Super Admin role is deleted.
- The only user assigned to the Super Admin role is unassigned from the role.
- The only user assigned to the Super Admin role is locked out of the Security Console. This can happen if you change the Security Console logon requirement from password to SecurID, and you change the requirement before assigning a token to the Super Admin.

If any of these occur, use the Super Admin Restoration utility to restore a Super Admin. For instructions, see [“Restoring a Super Admin”](#) on page 234.

Realm Administrator

This role grants complete administrative responsibility for managing all aspects of the realm. This role is limited in scope to the realm in which it is created and it does not include Super Admin permissions. The Realm Administrator can delegate some of the responsibilities of this role.

The default permissions for this role include complete management of the following:

- All realm permissions
- Replica instances
- Disaster recovery
- Agent deployment
- Import tokens
- Lower-level security domains
- Groups
- User assignments
- Reports
- Log maintenance

Security Domain Administrator

This role grants permission to manage all aspects of a branch of the security domain hierarchy. This administrator has all permissions within that branch except the ability to manage top-level objects, such as policies and attribute definitions. By default, this role's scope includes the entire realm.

To limit this role's scope to a lower-level security domain in the realm, edit the scope of the duplicate role. This role has the same permissions as the Realm Administrator and Super Admin, but is limited to the security domain in which it is created. The Security Domain Administrator can delegate some responsibilities of this role.

The default permissions for this role include complete management of the following:

- Security domains
- Administrative roles
- Permissions
- User groups
- Reports
- Tokens
- User accounts
- Agents and server nodes

User Administrator

This role grants administrative responsibility to manage users, assign tokens to users, and access to selected authentication agents. This administrator cannot delegate the responsibilities of this role.

The default permissions for this role include limited management of the following:

- Users—add, delete, edit, view
- User groups—view user group, assign user group membership
- Reports—add, delete, edit, view, run, schedule
- Tokens—view, reset SecurID PINs, enable and disable SecurID tokens, resynchronize tokens, assign tokens to users, replace tokens, issue software tokens, manage online and offline emergency access tokencodes
- User accounts—manage fixed passcode, manage logon aliases, edit default shell, manage incorrect passcode count, clear cached Windows credential, manage offline emergency access passcode
- Agents—view, grant user groups access to restricted authentication agents

Token Administrator

This role grants complete administrative responsibility to import and manage tokens, and to assign tokens to users. This administrator cannot delegate the responsibilities of this role.

The default permissions for this role include limited management of the following:

- Users—view
- Reports—add, delete, edit, view report definition, run, schedule
- Tokens—import, delete, edit, view token, reset SecurID PINs, resynchronize tokens, manage online and offline emergency access tokencodes, assign tokens to users, replace tokens, issue software tokens, export tokens, manage incorrect passcode count

Privileged Help Desk Administrator

This role grants administrative responsibility to resolve user access issues through password reset, and unlocking or enabling accounts. It also grants permission to provide online and offline emergency access help. This administrator cannot delegate the responsibilities of this role.

The default permissions for this role include limited management of the following:

- Users—view, reset passwords, enable and disable accounts, terminate active sessions
- User groups—view user groups
- Reports—run, schedule

- Tokens—view token, reset SecurID PINs, resynchronize tokens, manage online and offline emergency access tokencodes
- User accounts—manage fixed passcode, manage logon aliases, edit default shell, manage incorrect passcode count, clear cached Windows credential, manage offline emergency access passcode
- Agents—view

Help Desk Administrator

This role grants administrative responsibility to resolve user access issues through password reset, and unlocking or enabling accounts. This administrator cannot delegate the responsibilities of this role.

The default permissions for this role include limited management of the following:

- Users—view, reset passwords, enable and disable accounts, terminate active sessions
- User groups—view user groups
- Tokens—view token, reset SecurID PINs, resynchronize tokens, enable and disable SecurID tokens
- User accounts—manage logon aliases, edit default shell, manage incorrect passcode count, clear cached Windows credential
- Agents—view

Agent Administrator

This role grants administrative responsibility to manage authentication agents and grants access to selected authentication agents. This administrator cannot delegate the responsibilities of this role.

The default permissions for this role include limited management of the following:

- Users—view
- User groups—view user groups, assign user group membership
- Reports—add, delete, edit, view report definition, run, schedule
- Agents—add, delete, edit, view, manage node secret, grant user groups access to agents

Request Approver

This role grants administrative responsibility to view and approve requests. This administrator can delegate the responsibilities of this role.

The default permissions for this role include limited management of the following:

- Requests—view, and approve

For a Request Approver, scope works essentially the same as it does for Authentication Manager, however, if a user requests enrollment in a security domain, the approver in that security domain can approve the request for enrollment before the user is in the actual domain.

Token Distributor

This role grants administrative responsibility to view requests and distribute requests. Distributors also determine how to assign and deliver tokens to users. This administrator can delegate the responsibilities of this role.

The default permissions for this role include limited management of the following:

- Requests—view and distribute

For information on assigning predefined roles, see [“Assigning Administrative Roles”](#) on page 42.

Creating Administrative Roles

An administrative role has two components:

- A collection of permissions based on a job function profile
- The scope (security domains and identity sources) in which the permissions can be applied

To create an administrative role, the administrator creating the new administrative role must have the following:

- A role that grants permission to create administrative roles.
- An administrative role that includes the permissions he or she wants to add to the new administrative role.
- An administrative role that allows him or her to delegate the permissions granted to his or her role. This is determined by the Permission Delegation setting for the role assigned to the administrator who is creating the new role.

Note: You can create as many administrators as your deployment needs.

Permissions

The permissions that you assign to an administrative role govern the actions that an administrator can take. Be sure to assign permissions to administrative roles that allow administrators to manage all of the objects—for example, users, user groups, and attributes—necessary to accomplish their assigned tasks, but not so many permissions as to let them manage objects not vital to their responsibilities.

For example, if an administrator's only task is assigning tokens to users, you would probably assign the following permissions to the role:

- View users
- View tokens
- Assign tokens to users
- Issue assigned software tokens
- Replace assigned tokens
- Import tokens (optional)
- Enable and disable tokens (optional)

The optional permissions above give the administrative role slightly expanded capabilities that complement the stated task of assigning tokens to users. You would not, however, assign the permissions to add and delete users, resynchronize tokens, or manage emergency offline authentication, as they are not related to the stated task of assigning tokens to users.

When you assign permissions to a role, remember that to associate two objects in the deployment, an administrator must have the appropriate permissions for both objects. For example:

- To assign tokens to users, an administrator must be able to view and assign tokens, and view users.
- To move users between security domains, an administrator must be able to view security domains, and view and edit users.
- To assign administrative roles to users, an administrator must be able to view and assign roles, and view users.

Scope

The scope of an administrative role dictates where an administrator may perform administrative tasks. An administrative role's scope consists of two parts:

- The security domains that the administrative role can manage
- The identity sources that the administrative role can manage

Be sure to assign a scope broad enough so that the administrator can access the necessary security domains and identity sources. However, for security reasons, avoid assigning a scope that grants access to security domains and identity sources where the administrator has no responsibilities.

For example, suppose an administrator's only responsibility is to assign users to user groups in the security domain "Boston." In such a case, assign the administrator an administrative role that has a scope that includes only the security domain "Boston." Avoid including security domains in the scope that are not vital to the administrator's responsibilities.

Also, avoid creating situations in which an administrator can view and manage a certain user group, but cannot at least view all the users in that user group. This happens when a user group from a security domain within the administrator's scope contains users from a security domain outside the administrator's scope. When the administrator views the user group members, he or she only sees the members from the security domain within his or her scope, and therefore is unaware of the subset of users from other security domains.

For example, suppose an administrator can manage the Boston security domain. Included in the Boston security domain is a user group with members that belong to the Boston security domain and members that belong to the New York security domain. When the administrator views the members of the user group, he or she only sees the users from the Boston security domain because the users from the New York security domain are outside of his or her scope.

This creates a situation in which administrators may take action on a group, for example, granting a group access to a restricted agent, without being aware of all the users affected. This can result in users being granted privileges they should not have.

To avoid this situation, RSA recommends that you do one or more of the following when you specify the scope of an administrative role:

- Allow all administrators to at least have view permission on all users in all security domains. This ensures that there are no cases where administrators are unaware of any members of a group they are administering.
- Make sure that a user group and all members of the user group are in the same security domain. This ensures that administrators who have permissions to view user groups and to view users are able to see all member users.

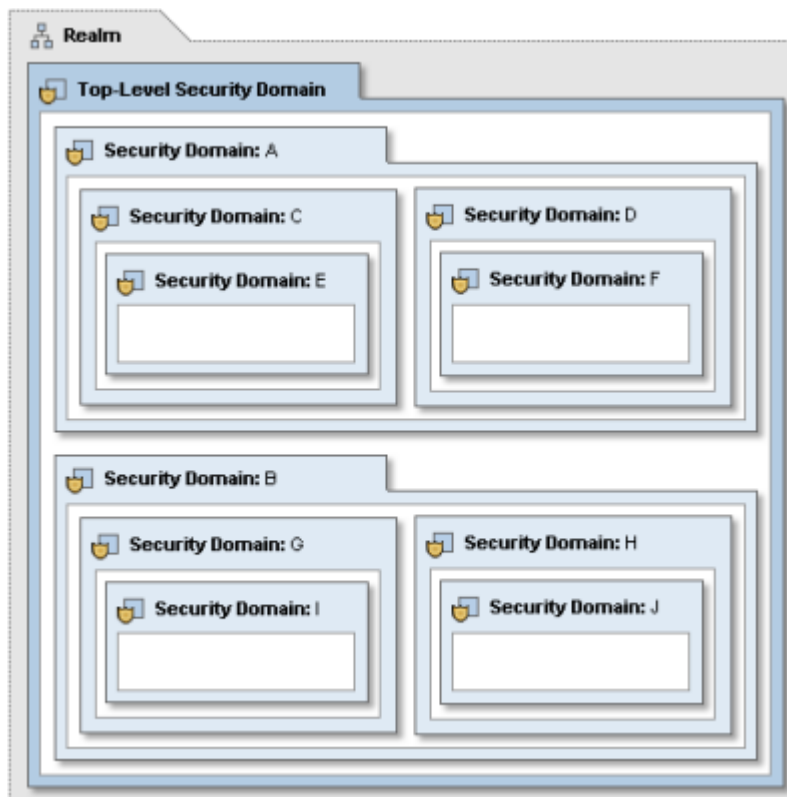
Important: After you save an administrative role, you cannot edit the scope.

Know the following about scoping administrative roles:

- When an administrative role's scope is defined most broadly, the role can manage the security domain where the role definition was saved and all of the lower-level security domains beneath it.
- An administrative role that manages an upper-level security domain always manages the lower-level security domains beneath it.
- You can limit the scope of an administrative role to specific security domains, as long as those security domains are at or below the security domain that owns the role. An administrative role can only manage down the security domain hierarchy, never up.
- The security domain where you save the administrative role impacts the scope of the role. For example, suppose the top-level security domain is Boston, and the lower-level security domains are named New York and San Jose. If you save an administrative role to the New York security domain, administrators with that role can only manage objects in the New York security domain and in lower-level security domains within the New York security domain. Administrators with that role cannot manage objects in the Boston or San Jose security domains.

RSA recommends that you save the Super Admin role in the top-level security domain, and then save all other administrative roles in a lower-level security domain. This prevents lower-level administrators—Help Desk Administrators, for example—from editing the Super Admin's password and then using the Super Admin's password to access the Security Console.

For example, consider the following hierarchy.



- An administrative role saved in the top-level security domain can be scoped to manage any security domain in the realm. For example, it can manage only security domain F, or every security domain in the realm.
- An administrative role saved in security domain A can be defined to manage security domain A and all the lower-level security domains below it.
- An administrative role saved in security domain C can be defined to manage E, or both C and E.
- An administrative role saved in security domain E can be defined to manage only security domain E.

An administrative role can be defined so that it manages only users or user groups within a particular administrative scope who match certain criteria. These are called attribute-based roles. For example, if you have defined an identity attribute definition for location, an administrative role can manage all users in security domain A and down the hierarchy (C, D, E, and F) who live in Toronto.

To use an identity attribute definition in an attribute-based role, you must enable the identity attribute definition for use in administrative scope restrictions. You do this when you define the identity attribute definition for the realm. For instructions, see the Security Console Help topic “Add New Identity Attribute Definitions.”

You can also create a role that only allows an administrator to edit specified custom user identity attribute definitions. You configure permissions to a specific identity attribute definition as part of the role's permissions.

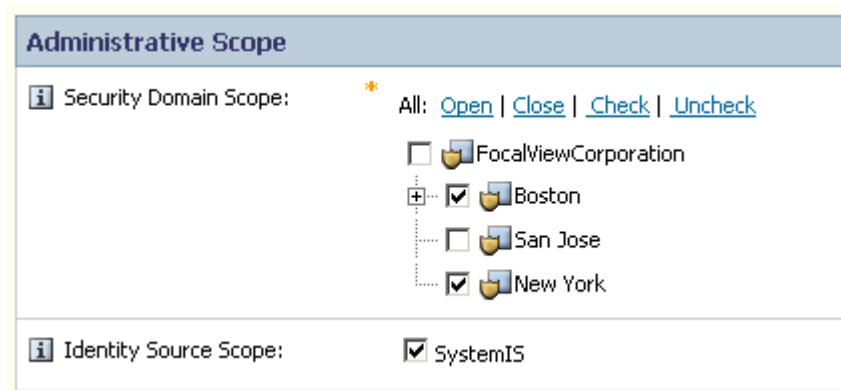
For a given attribute, the role can specify one of the following access permissions:

- None
- Read-Only
- Modify

To create an administrative role:

1. Click **Administration > Administrative Roles > Add New**.
2. In the **Administrative Role Name** field, enter a name for the new administrative role. A role name must be unique in the security domain where it is defined, but does not have to be unique in the realm.
Administrative role names typically reflect administrators' functions within an organization, such as Finance Admin or HR Admin. RSA recommends that you enter a brief explanation of the role in the **Notes** field.
3. Decide if you want to allow administrators to delegate permissions granted to them by this administrative role to other administrators. If you want to allow this, select **Permission Delegation**. This selection only applies to administrators who also have the ability to create, edit, and assign administrative roles.
4. In the **Security Domain Scope** tree, select the security domains to include in the scope of the role. The scope determines where an administrator assigned this role has administrative permissions. When an administrative role grants permissions in a security domain, permissions are also granted to the lower-level security domains in the hierarchy.
5. In the **Identity Source Scope** field, select the identity sources where you want this administrative role to grant permissions.

The following figure shows the **Security Domain Scope** and **Identity Source Scope** fields of the Add New Administrative Role page. Note that the security domains are nested to show the hierarchy.



6. Click **Next**.

7. Assign permissions to the administrative role.
Permissions assigned to the administrative role determine what actions an administrator assigned the role can take on objects such as users, user groups, and security domains. The following permissions are available for all objects in your Authentication Manager deployment:
 - All.** Grants an administrator permission to perform any administrative action on the object.
 - Delete.** Grants an administrator permission to delete an object.
 - Add.** Grants an administrator permission to add an object.
 - Edit.** Grants an administrator permission to view and edit an object, but not the ability to add or delete.
 - View.** Grants an administrator permission to view an object, but not the ability to add, edit, or delete.
8. Click **Next**, and enter additional permissions.
9. Use the **Security Domain** drop-down menu to select the security domain to which you want to assign the administrative role. The new administrative role can only be managed by administrators whose scope includes the security domain to which you assign this role.
10. Click **Save**.

After you save the new administrative role, you can assign it to users.

Assigning Administrative Roles

When you assign an administrative role to users, the users become administrators and can use the Security Console to administer the system.

You can assign administrative roles to any user in your identity source. When you assign a role to a user, the user becomes an Authentication Manager administrator. You will probably only assign administrative roles to members of your information technology (IT) organization, and possibly a few other trusted individuals in your organization.

When you assign an administrative role to an administrator, the administrator is then able to perform the administrative actions specified by the role in the security domains specified by the scope of the role.

You may also assign more than one administrative role to an administrator. When you do this, the administrator can only perform administrative actions in a security domain that is included in the scope of the role that grants the permission.

For example, suppose an administrator has one role that grants him permission to manage users in the San Jose security domain, and another role that grants him permission to manage tokens in the New York security domain. When the administrator logs on, he is allowed to manage users in the San Jose security domain, but not the New York security domain, and to manage tokens in the New York security domain, but not in the San Jose security domain. If the administrator searches for users in the New York security domain, no results are returned because he does not have permission to manage them.

When assigning roles to administrators, be sure to assign roles that grant only enough permissions and include a scope just broad enough to accomplish their tasks. Avoid granting administrative roles to administrators who do not need them.

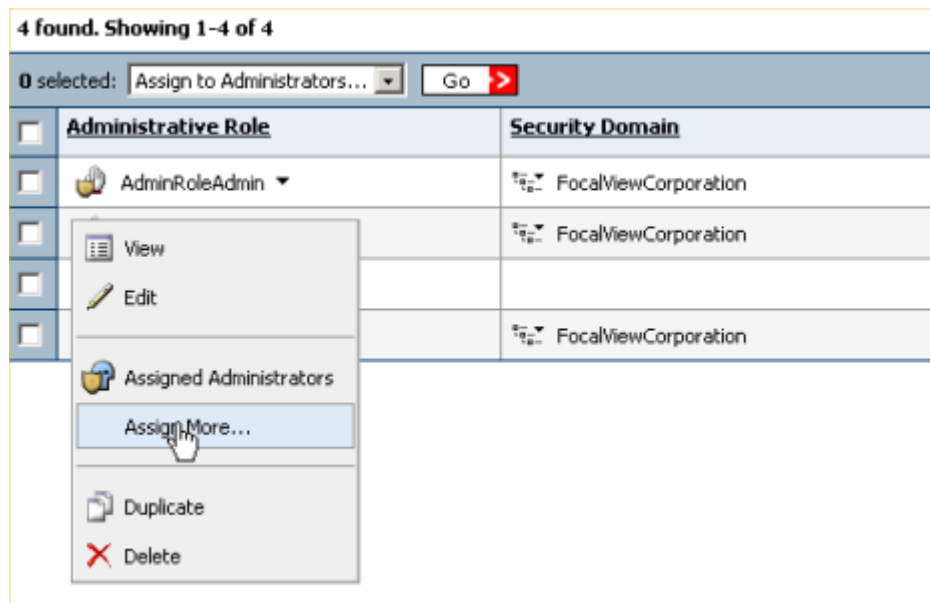
For example, if an administrator's job only requires him to administer users in the Boston security domain, avoid including the San Jose and New York security domains in the scope of his role.

You can only assign administrative roles that have equal or fewer permissions to your own role. For example, if your administrative role only allows you to add, edit and delete users, and create and assign administrative roles, you cannot assign a role that also grants permission to assign tokens to users. Remember that you cannot assign roles with permissions that you do not have.

To assign an administrative role to a single user:

1. Click **Administration > Administrative Roles > Manage Existing**.
2. Use the search fields to find the administrative role that you want to assign.
3. Click on the administrative role that you want to assign.
4. From the Context menu, click **Assign More**.

The following figure shows the Context menu on the Administrative Roles list page.



5. Use the search fields to find the user that you want to assign to the administrative role.

Note: User searches are case sensitive.

6. Click on the user that you want to assign to the administrative role, and from the Context menu, click **Assign to Role**.

Note: If a password is required to access the Security Console, administrators use the password assigned to them in their user record.

Organizing Users for Administration

Once you create your security domain hierarchy and link your identity source with your realm, all users appear by default in the top-level security domain. To help you organize and manage your system, and to help you limit administrative scope, it is likely that you will want to transfer users to one of the other security domains in your hierarchy.

Just as you have likely created security domains to match either your organization's structure or geographic locations, you can use the Security Console to transfer users from each department or location to their respective security domains.

For example, if you have a top-level security domain named FocalView, and lower-level security domains named Boston, New York, and San Jose, you would likely move users located in each of those locations to their respective security domains.

Note: Each user can exist in only one security domain.

Organizing users in security domains helps you find users and assign them tokens or add them to user groups. For example, you can search for all users in the Boston security domain and then assign a token to each member.

Note: User searches are case sensitive.

In addition to making management of your users easier, security domains allow you to limit the scope of an administrator's permissions. For example, suppose you have security domains named Boston, New York, and San Jose. When you set the administrative scope for your administrative roles, you might choose to limit the role to only the Boston security domain. This means that the administrator assigned that role only has permission to manage the Boston security domain, and not the New York and San Jose security domains.

An arrangement such as this allows you greater control over administrators. You can limit administrators by department, geographic location, or in any other way that you choose.

Protecting the RSA Security Console

The default method for protecting the Security Console is the RSA password. For additional security, you can configure Authentication Manager to require administrators to present a credential more secure than a password, such as an RSA SecurID passcode, before they can access the Security Console.

Presenting a SecurID passcode before being allowed access ensures that your Authentication Manager deployment is protected by the same two-factor authentication that protects your network resources.

Important: If the Security Console is protected with a SecurID passcode, the SecurID PIN is case sensitive.

If you use LDAP as your Authentication Manager identity source, you may also want to enable LDAP passwords as an authentication method. This allows administrators whose user records are saved in the LDAP identity source to access the Security Console.

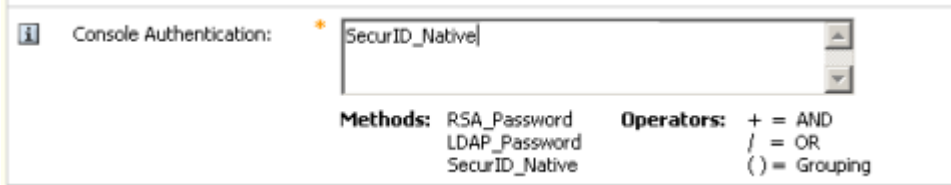
When you first enable a new Security Console authentication method, RSA recommends that you also continue to allow administrators to authenticate with the previous method for a period of time. This gives you time to ensure that all administrators can authenticate with the new method before you discontinue the previous method.

To configure RSA Security Console protection:

1. Click **Setup > Authentication Methods Configuration**.
2. In the Security Console **Authentication** field, enter the authentication method that administrators must use to log on to the Security Console.

For example, to require a SecurID passcode, enter **SecurID_Native**.

To require both a SecurID passcode and LDAP password, enter **SecurID_Native+LDAP_Password**.



Console Authentication: * SecurID_Native

Methods: RSA_Password
LDAP_Password
SecurID_Native

Operators: + = AND
/ = OR
() = Grouping

Important: If you change the Security Console authentication method, all administrators who are logged on to the Security Console are immediately logged off and must authenticate with the new credential. Make sure that the administrators have the new credential before requiring them to log on with it.

3. Click **Save**.

2

Configuring Authentication Policies

- [Setting Password Requirements](#)
- [Setting Token Usage Requirements](#)
- [Locking Users Out of the System](#)
- [Setting Offline Authentication Requirements](#)
- [Setting Self-Service Troubleshooting Requirements](#)

Setting Password Requirements

All RSA Authentication Manager users are required to have a password as part of their user record. If you use the Authentication Manager internal database as your identity source, the password is stored in the internal database. It may only be used by administrators, if policy permits, to log on to the RSA Security Console.

If you use an external LDAP directory as your identity source, the password field in the Authentication Manager user record may be mapped to the LDAP password, which is stored in the LDAP directory. This password may be used to log on to other applications or resources within your organization. If policy permits, administrators may also use a password to log on to the Security Console. A password is required by Authentication Manager because the external LDAP directory requires all users to have passwords.

Password characteristics are controlled by password policies. Password policies define users' password length, format, and frequency of change. You assign password policies to security domains. The password policy assigned to a security domain dictates the password-related requirements for all of the users that are assigned to that security domain.

Note: LDAP password characteristics are only controlled by Authentication Manager password policies when you edit the LDAP password with the Security Console.

When you install Authentication Manager, a default password policy is automatically created. You can edit this policy, or create a custom password policy and designate it as the default.

One password policy is always designated as the default policy. When you create new security domains, Authentication Manager automatically assigns the default password policy to the new security domains. You can use the default password policy or assign a custom policy to each security domain.

To use the default policy, make sure that **Use the default policy** is selected from the **Password Policy** drop-down list on the Add and Edit Security Domains pages. The policy designated as the default is automatically assigned to the security domain.

To use a password policy other than the default, specify a policy name from the drop-down list. The policies listed in the drop-down list are custom policies. You can create custom policies on the Password Policies page in the Security Console. For more information, see the Security Console Help topic “Add Password Policies.”

Note: Password policies assigned to upper-level security domains are not inherited by lower-level security domains. For example, if you assign a custom policy to the top-level security domain, all new security domains that you create below it in the hierarchy are still assigned the default password policy.

Requiring Use of System-Generated Passwords

Enabling this option requires users to use passwords generated by Authentication Manager according to the password policy applied to the users' security domain.

Enabling this option ensures that users' passwords are random and therefore less likely to be guessed by an unauthorized person attempting to access your network.

When users are initially assigned their password, or when their passwords expire, they are prompted to choose from a list of system-generated passwords when they attempt to use their password.

Note: Administrators typically only use their passwords to access the Security Console. Most other users do not use their passwords.

This option is part of a password policy, and can be enabled on the Add New Password Policy and Edit Password Policy pages.

Using System-Generated PINs with RSA RADIUS

RSA RADIUS does not allow system-generated PINs by default. If you choose to allow system-generated PINs, authentications will fail unless you change the RADIUS configuration file, **securid.ini**, to allow system-generated PINs.

In the **securid.ini** file, system-generated PINs are set in the following line:

```
AllowSystemPins = 0
```

The 0 indicates that system-generated PINs are not allowed. To allow system-generated PINs, change the setting to the following:

```
AllowSystemPins = 1
```

The 1 indicates that system-generated PINs are allowed.

For instructions on editing RADIUS configuration files, see the Operations Console Help topic “List and Edit RADIUS Configuration Files.”

Requiring Periodic Password Changes

Enabling this option allows you to set minimum and maximum password lifetimes.

The minimum password lifetime is the minimum amount of time that a password can exist before the user can change it. For example, suppose the minimum password lifetime is set to 14 days. If users change their passwords on June 15, they cannot change it again until June 29.

Setting a minimum password lifetime prevents users from circumventing restrictions on the reuse of old passwords that you may have set. For example, suppose you restrict users from reusing their five most recent passwords. The minimum password lifetime prevents them from immediately changing their password six times so they can reuse a particular password.

The maximum password lifetime is the maximum amount of time a user can keep a password before being required to change it. For example, suppose the maximum password lifetime is set to 90 days. If users change their password on June 1, they are required to change it again on August 30.

Setting a maximum password lifetime prevents users from indefinitely keeping the same password, which increases the likelihood that it might be guessed by an unauthorized person trying to access your network.

Be sure to temper the need for security with consideration of what is reasonable for the members of your organization. An overly strict maximum lifetime, such as one that requires a password change every seven days, may irritate users. It also may be counter-productive in that remembering a new password every seven days is difficult and may increase the number of employees who write down their passwords, which negates the effectiveness of the strict policy.

This option is part of a password policy, and can be enabled on the Add New Password Policy and Edit Password Policy pages.

Restricting Reuse of Old Passwords

Setting a restriction on the reuse of old passwords prevents users from reusing the same two or three passwords over and over. For example, suppose you set the option to restrict the last three passwords, and the last three passwords are “password1,” “password2,” and “password3.” Users cannot enter those three passwords, and must choose another.

Reusing the same passwords over and over increases the likelihood that an unauthorized person may guess a password, especially if the passwords are all similar, for example, “favoritepet1” or “favoritepet11.”

The goal of the restriction is to force users to move beyond those couple of passwords that they are most comfortable with—for example, a pet’s name or child’s birthday—and choose more secure passwords.

This option is part of a password policy, and can be enabled on the Add New Password Policy and Edit Password Policy pages.

Limiting Password Lengths

Setting minimum and maximum password lengths prevents users from creating passwords that are too short and easily guessed by an unauthorized person attempting to access your network, or that are too long and difficult to be remembered by authorized users.

For example, suppose you set the minimum length to six characters, and the maximum length to eight characters. Short, easily-guessed words are not allowed, and longer words or combinations of letters, which users have a hard time remembering, are also not allowed.

Be sure to temper the need for security with consideration of what is reasonable for the members of your organization. Required password lengths that are too long may irritate users. It also may be counter-productive in that remembering a long password is difficult and may increase the number of employees who write down their passwords, which negates the effectiveness of the strict policy. Long passwords that users cannot remember can also lead to more users locked out of your network, and more calls to the Help Desk for assistance.

This option is part of a password policy, and can be enabled on the Add New Password Policy and Edit Password Policy pages.

Using an Excluded Words Dictionary

The excluded words dictionary is a record of words that users cannot use as passwords. The excluded words dictionary includes several thousand commonly used words that are likely to be included as part of any dictionary attacks on the system, for example, “password.”

The excluded words dictionary prevents users from using common, and therefore, easily guessed words as passwords.

You use the Security Console to manage the excluded words dictionary. For instructions, see the Security Console Help topics “Add a Password Dictionary,” “Delete a Password Dictionary,” and “Export a Password Dictionary.”

Note: Your deployment can only have one password dictionary. If a password dictionary is already installed, you must delete it before adding a new one.

Once you add a dictionary, you can select it on the Add New Password Policy or Edit Password Policy page. For instructions, see the Security Console Help topics “Add New Password Policies” and “Edit Password Policies.”

Setting Password Character Requirements

Requiring specific characters in passwords can make guessing the password more difficult, particularly when the required characters are not alphanumeric.

Dictionary attacks on your system, in which unauthorized users use software to systematically enter all words in a dictionary in an attempt to guess valid passwords, are rendered less effective when passwords contain special characters.

For example, the password “maryland” is more likely to be included in a dictionary attack than “mary%land” or “mary**land.”

You can require the following types of characters:

- Alphabetic characters
- Uppercase characters
- Lowercase characters
- Numeric characters
- Non-alphanumeric characters

This option is part of a password policy, and can be enabled on the Add New Password Policy and Edit Password Policy pages.

Setting Token Usage Requirements

Token policies define users' RSA SecurID PIN lifetime and format, and fixed passcode lifetime and format, as well as how your deployment handles users or unauthorized people who enter a series of incorrect passcodes. Passcodes are your SecurID PIN + tokencode. The tokencode is the number displayed on the front of your SecurID token.

Important: Fixed passcodes are essentially passwords, and are not recommended because they eliminate the advantages of two-factor authentication. Use fixed passcodes only in test environments and in situations where a user is authenticating to an authentication agent inside the corporate firewall.

You assign token policies to security domains. The token policy assigned to a security domain dictates the token-related requirements for all of the users assigned to that security domain.

When a user authenticates with a token, requirements such as PIN requirements and fixed passcode requirements are dictated by the token policy of the users' security domain, rather than by the policy of the token's security domain. For example, if a user assigned to the New York security domain authenticates with a token assigned to the Boston security domain, the token policy of the New York security domain dictates policy requirements.

When you install Authentication Manager, a default token policy is automatically created. You can edit this policy, or create a custom token policy and designate it as the default.

Note: When you edit an existing token policy, existing PINs and fixed passcodes are not validated against the excluded words dictionary and history requirements. They are, however, validated against all other policy requirements.

One token policy is always designated as the default policy. Authentication Manager assigns the default policy to each new security domain. You can use the default token policy or assign a custom policy to each security domain.

To use the default policy, select **Use the default policy** from the **SecurID Token Policy** drop-down list. The policy designated as the default is automatically applied to the security domain.

To use a token policy other than the default, specify a policy name from the drop-down list. The policies listed in the drop-down list are custom policies. You can create custom policies on the Token Policies page in the Security Console. For more information, see the Security Console Help topic “Add Token Policies.”

Note: Token policies assigned to upper-level security domains are not inherited by lower-level security domains. For example, if you assign a custom policy to the top-level security domain, all new security domains that you create below it in the hierarchy are still assigned the default password policy.

Limiting the Number of Incorrect Passcodes Allowed

This option lets you specify the number of incorrect passcodes allowed before users are prompted to enter the next tokencode from their tokens.

For example, suppose the number of incorrect passcodes is set at three in the token policy, and the number of allowed incorrect authentication attempts is set to four in the lockout policy. When users enter three incorrect passcodes and then enter a correct passcode, they are prompted to enter the next tokencode that displays on their token. If they enter the next tokencode correctly, they are successfully authenticated. If they enter it incorrectly, they are locked out.

You also have the option of allowing unlimited incorrect passcodes. Do this if you never want users to be prompted to enter their next tokencode, regardless of how many incorrect passcodes they enter before finally entering one correctly. Be aware, however, that if the lockout policy assigned to the security domain is set to lock users out after a specified number of failed authentications, when the user exceeds the allowed failed authentications, the user is locked out. See [“Locking Users Out of the System”](#) on page 57.

You select the number of allowable incorrect passcodes as part of a token policy. This option can be enabled on the Add New Token Policy and Edit Token Policy pages.

Setting Tokencode Ranges for Event-Based Tokens

If you plan on assigning event-based tokens to your users, you need to configure the accepted tokencode ranges. The tokencode range dictates the degree to which the event-based tokencode count can differ from the Authentication Manager tokencode count and still allow successful authentication. Successful authentications occur only when the token's tokencode count and the Authentication Manager tokencode count are within the specified range.

The tokencode range specified in the token policy is divided into two sub-ranges:

Normal authentication. This is the allowable tokencode range for successful authentication with one tokencode.

Next-tokencode Mode. This is the allowable range for authentication with two consecutive tokencodes.

For example, assume you just assigned an event-based token and set the tokencode range at 10 for normal authentication and 15 for next-tokencode mode. The first time the user authenticates with the token, the tokencode count advances to one for both the token and Authentication Manager. Now assume that the user throws the token into the bottom of his briefcase. There, the button that advances the tokencode is hit repeatedly by some papers in the briefcase. Suddenly, the tokencode count is at 20, but the Authentication Manager count remains at one. When the user attempts to authenticate with the twentieth tokencode, the authentication attempt fails because Authentication Manager is expecting the second tokencode. In this situation, the user must contact the administrator so that the token can be resynchronized.

In the preceding example, the authentication failed because the twentieth tokencode was out of the range (15) set forth by the token policy. If the tokencode had been advanced to 14 instead of 20, the user could have successfully authenticated by entering two consecutive tokencodes. If the tokencode had been advanced to 6, the user could have successfully authenticated with one tokencode.

When creating a token policy, you can use a default tokencode range, or you can specify the two ranges. If you specify your own range, make sure that the range is not too big, as a large range provides an opportunity for an unauthorized user to guess the tokencode.

For more information on event-based tokens, see [“Deploying Tokens to Users”](#) on page 75.

Requiring Periodic RSA SecurID PIN Changes

Enabling this option allows you to set minimum and maximum SecurID PIN lifetimes.

The minimum SecurID PIN lifetime is the minimum amount of time that a PIN can exist before the user can change it. For example, suppose the minimum PIN lifetime is set to 14 days. If users change their PIN on June 15, they cannot change it again until June 29.

Setting a minimum PIN lifetime prevents users from circumventing restrictions on the reuse of old PINs that you may have set. For example, suppose you restrict users from reusing their five most recent PINs. The minimum PIN lifetime prevents them from immediately changing their PIN six times so they can reuse a particular PIN.

The maximum PIN lifetime is the maximum amount of time a user can keep a PIN before being required to change it. For example, suppose the maximum PIN lifetime is set to 90 days. If users change their PIN on June 1, they are required to change it again at their next logon on or after August 30.

Setting a maximum PIN lifetime prevents users from indefinitely keeping the same PIN, which increases the likelihood that it might be guessed by an unauthorized person trying to access your network.

Be sure to temper the need for security with consideration of what is reasonable for the members of your organization. An overly strict maximum lifetime, such as one that requires a PIN change every seven days, may irritate users. It also may be counter productive in that remembering a new password every seven days is difficult and may increase the number of employees who write down their PINs, which negates the effectiveness of the strict policy.

This option is part of a token policy, and can be enabled on the Add New Token Policy and Edit Token Policy pages.

Restricting Reuse of Old PINs

Setting a restriction on the reuse of old SecurID PINs prevents users from reusing the same two or three PINs over and over. For example, suppose that you set the option to restrict the last three PINs, users cannot enter those three PINs, and must choose another.

Reusing the same PINs over and over increases the likelihood that an unauthorized person may guess a PIN, especially if the PINs are all similar.

The goal of the restriction is to force users to move beyond those couple of PINs that they are most comfortable with and choose more secure PINs.

This option is part of a token policy, and can be enabled on the Add New Token Policy and Edit Token Policy pages.

Limiting RSA SecurID PIN Length

Setting minimum and maximum SecurID PIN lengths prevents users from creating PINs that are too short and easily guessed by an unauthorized person attempting to access your network, or that are too long and difficult to be remembered by authorized users.

For example, suppose you set the minimum length to six characters, and the maximum length to eight characters. Short, easily guessed PINs are not allowed, and longer PINs or combinations of letters, which users have a hard time remembering, are also not allowed.

Be sure to temper the need for security with consideration of what is reasonable for the members of your organization. Required PIN lengths that are too long may irritate users. It also may be counter productive in that remembering a long PIN is difficult and may increase the number of employees who write them down, which negates the effectiveness of the strict policy. Long PINs that users cannot remember can also lead to more users locked out of your network, and more calls to the Help Desk for assistance.

This option is part of a token policy, and can be enabled on the Add New Token Policy and Edit Token Policy pages.

Setting RSA SecurID PIN Character Requirements

Requiring specific characters in SecurID PINs can make guessing the PIN more difficult.

You can choose from the following types of PINs:

- Alphabetic
- Numeric
- Alphanumeric

For alphanumeric PINs, you can choose the number of alphabetic and numeric characters that are required within the PIN.

This option is part of a token policy, and can be enabled on the Add New Token Policy and Edit Token Policy pages.

Requiring Periodic Fixed Passcode Changes

Enabling this option allows you to set minimum and maximum fixed passcode lifetimes.

Important: Fixed passcodes are essentially passwords, and are not recommended because they eliminate the advantages of two-factor authentication. Use fixed passcodes only in test environments and in situations where a user is authenticating to an authentication agent inside the corporate firewall.

The minimum fixed passcode lifetime is the minimum amount of time that a fixed passcode can exist before the user can change it. For example, suppose the minimum lifetime is set to 14 days. If users change their fixed passcode on June 15, they cannot change it again until June 29.

Setting a minimum fixed passcode lifetime prevents users from circumventing restrictions on the reuse of old fixed passcodes that you may have set. For example, suppose you restrict users from reusing their five most recent fixed passcodes. The minimum fixed passcode lifetime prevents them from immediately changing their fixed passcode six times so they can reuse a particular fixed passcode.

The maximum fixed passcode lifetime is the maximum amount of time a user can keep a fixed passcode before being required to change it. For example, suppose the maximum fixed passcode lifetime is set to 90 days. If users change their fixed passcode on June 1, they are required to change it again on August 30.

Setting a maximum fixed passcode lifetime prevents users from indefinitely keeping the same fixed passcode, which increases the likelihood that it might be guessed by an unauthorized person trying to access your network.

Be sure to temper the need for security with consideration of what is reasonable for the members of your organization. An overly strict maximum lifetime, such as one that requires a fixed passcode change every seven days, may irritate users. It also may be counter productive in that remembering a new fixed passcode every seven days is difficult and may increase the number of employees who write down their fixed passcodes, which negates the effectiveness of the strict policy.

This option is part of a token policy, and can be enabled on the Add New Token Policy and Edit Token Policy pages.

Restricting Reuse of Old Fixed Passcodes

Setting a restriction on the reuse of old fixed passcodes prevents users from reusing the same two or three fixed passcodes over and over. For example, suppose you set the option to restrict the last three fixed passcodes, users cannot enter those three fixed passcodes, and must choose another.

Reusing the same fixed passcodes over and over increases the likelihood that an unauthorized person may guess a fixed passcode, especially if the fixed passcodes are all similar.

The goal of the restriction is to force users to move beyond those couple of fixed passcodes that they are most comfortable with and choose more secure fixed passcodes.

This option is part of a token policy, and can be enabled on the Add New Token Policy and Edit Token Policy pages.

Limiting Fixed Passcode Length

Setting minimum and maximum fixed passcode lengths prevents users from creating fixed passcodes that are too short and easily guessed by an unauthorized person attempting to access your network, or that are too long and difficult to be remembered by authorized users.

Be sure to temper the need for security with consideration of what is reasonable for the members of your organization. Required fixed passcode lengths that are too long may irritate users. It also may be counter productive in that remembering a long fixed passcode is difficult and may increase the number of employees who write them down, which negates the effectiveness of the strict policy. Long fixed passcodes that users cannot remember can also lead to more users locked out of your network, and more calls to the Help Desk for assistance.

This option is part of a token policy, and can be enabled on the Add New Token Policy and Edit Token Policy pages.

Setting Fixed Passcode Character Requirements

Requiring specific characters in fixed passcodes can make guessing the fixed passcode more difficult.

You can require the following types of characters:

- Alphabetic characters
- Numeric characters

This option is part of a token policy, and can be enabled on the Add New Token Policy and Edit Token Policy pages.

Setting Emergency Access Code Formats

Emergency access tokencodes and passcodes are used to temporarily replace an assigned SecurID token or SecurID PIN if the user does not have access to their token or has forgotten their PIN.

Use the Emergency Access Code Format section of the Add and Edit Token Policy pages in the Security Console to select required characters for emergency access tokencodes and passcodes. Required characters are included in each passcode and tokencode that is generated.

You may choose to require the following types of characters:

- Numeric characters
- Alphabetic characters
- Special characters

Locking Users Out of the System

Lockout policies define how many failed logon attempts users can make before the Authentication Manager locks their account. You assign lockout policies to security domains. The lockout policy assigned to a security domain dictates the lockout requirements for all the users assigned to that security domain.

Important: Lockout policies determine the lockout criteria for Authentication Manager only. To set lockout requirements for RSA Credential Manager, see [“Setting Self-Service Troubleshooting Requirements”](#) on page 63.

Lockout policies apply to all logon attempts regardless of how many different tokens a user uses to authenticate. For example, if a user has two failures with their software token and one failure with their hardware token, that adds up to three failed attempts.

When you install Authentication Manager, a default lockout policy is automatically created. You can edit this policy, or create a custom lockout policy and designate it as the default.

One lockout policy is always designated as the default policy. Authentication Manager assigns the default policy to each new security domain. You can use the default lockout policy or assign a custom policy to each security domain.

To use the default policy, select **Use the default policy** from the **Lockout Policy** drop-down list. The policy that has been designated as the default is automatically applied to the security domain.

To use a lockout policy other than the default, specify a policy name from the drop-down list. The policies listed in the drop-down list are custom policies. You can create custom policies on the Lockout Policies page in the Security Console. For more information, see the Security Console Help topic “Add Lockout Policies.”

Note: Lockout policies assigned to upper-level security domains are not inherited by lower-level security domains. For example, if you assign a custom policy to the top-level security domain, all new security domains that you create below it in the hierarchy are still assigned the default lockout policy.

Locking Out Users After a Specified Number of Logon Attempts

Enabling this option allows you to lock users out of the system after a specified number of failed authentication attempts set by the lockout policy assigned to their security domain. That is, they can no longer access the system until they are re-enabled either by an administrator or automatically by the system, depending on the settings of the lockout policy.

The number of failed authentication attempts allowed by the lockout policy should be higher than the number of incorrect passcodes allowed in the token policy. This allows the system to request the users next tokencode after they enter a specified number of incorrect passcodes.

If the number of failed authentication attempts allowed is lower than the number of incorrect passcodes allowed, the user is locked out and never prompted to enter their next tokencode. This means that more users are locked out because they accidentally entered incorrect passcodes, and an increased burden on administrators who are required to unlock user accounts.

Once locked out of the system, the lockout policy governs how a user is re-enabled. If the policy is configured to automatically re-enable the user, the locked out user is re-enabled after a specified amount of time elapses. The time is specified in the lockout policy. If configured so that locked out users must be re-enabled by an administrator, the user remains locked out until an administrator explicitly re-enables the user.

Setting Offline Authentication Requirements

Offline authentication extends RSA SecurID authentication to users who use SecurID for Windows. SecurID for Windows requires users to authenticate when logging on to their computer. Since the user must authenticate to access his or her own computer, the user must authenticate even when working offline (for example, when the user is working from home). You enable, disable, and configure offline authentication through the Authentication Manager by specifying an offline authentication policy and applying that policy to Authentication Manager security domains.

Important: Offline policies apply to security domains. Collisions between two policies can occur if the user is in one security domain and the agent (their computer) is in a different security domain, and both security domains have different offline authentication policies. For more information on collisions and merging offline authentication policies, see the following section, "[Merging Offline Authentication Policies.](#)"

When offline authentication is enabled, Authentication Manager downloads a configurable number of "offline days" of tokencode data to users' machines. This data is used when users attempt to authenticate offline.

You enable local authentication and Windows password integration through the Security Console, as part of an offline authentication policy. Only users assigned to security domains with an offline authentication policy that allows offline authentication and Windows password integration can use these features.

When you install Authentication Manager, a default offline authentication policy is automatically created. You can edit this policy, or create a custom offline authentication policy and designate it as the default.

One offline authentication policy is always designated as the default policy. Authentication Manager assigns the default policy to each new security domain. You can use the default offline authentication policy or assign a custom policy to each security domain.

To use the default policy, select **Use the default policy** from the **Offline Authentication Policy** drop-down list. The policy designated as the default is automatically assigned to the security domain.

To use an offline authentication policy other than the default, specify a policy name from the drop-down list. The policies listed in the drop-down list are custom policies. You can create custom policies on the Token Policies page in the Security Console. For more information, see the Security Console Help topic "Add Offline Authentication Policies."

Note: Offline authentication policies assigned to upper-level security domains are not inherited by lower-level security domains. For example, if you assign a custom policy to the top-level security domain, all new security domains that you create below it in the hierarchy are still assigned the default offline authentication policy.

Merging Offline Authentication Policies

Offline authentication policies apply to security domains. When a user in one security domain tries to access an agent in a different security domain, a collision can occur if both security domains have different offline authentication policies.

For example, assume that the user is in security domain A, which allows the user to download five days of offline data. Assume that the agent is in security domain B, which allows only three days of offline data. When the user authenticates and accesses the agent, the stricter policy applies, so the user would only have three days of offline data.

Authentication Manager merges the two policies in different ways depending on the offline authentication parameter. For example, sometimes the stricter policy is enforced, as in the previous example. Other times, Authentication Manager combines the values from the two policies.

For example, assume that the offline authentication policy for the user's security domain allows emergency authentication with a passcode but not a tokencode. Assume that the policy for the agent's security domain allows emergency authentication with a tokencode, but not a passcode. In this case, the two parameters combine, and the user cannot have emergency access with either a tokencode or a passcode.

The following table indicates how merging works for each offline authentication parameter.

Field Name	How the Policies Merge
Offline Authentication Enabled	Both policies must enable this feature for the user to gain offline access.
Windows Password Integration	Both policies must enable this feature for the user to gain access using the Windows password.
Allow Offline Authentication Using PINPad or Software Token	Both policies must enable this feature for the user to use PINPad or software tokens.
Allow Offline Authentication Using PIN-less Token	Both policies must enable this feature for the user to use tokens that do not require PINs.
Allow Offline Authentication Using Fixed Passcode	Both policies must enable this feature for the user to use a fixed passcode.
Allow Offline Emergency Access Tokencodes	Both policies must enable this feature for the user to use an emergency access tokencode for offline access.
Allow Offline Emergency Access Passcodes	Both policies must enable this feature for the user to use an emergency access passcode for offline access.

Field Name	How the Policies Merge
Allow Numbers, Characters, or Symbols in Offline Emergency Access Codes	If either policy enables this feature, the user can use an emergency access code containing numbers, characters, or symbols.
Days Offline Emergency Access Code Valid	The system uses the smaller of the two values.
Maximum Days of Offline Data	The system uses the smaller of the two values.
Days of Offline Data Warning	The system uses the larger of the two values.
Number of Authentication Failures	The system uses the smaller of the two values.
Upload Offline Authentication Log Entries	If either policy enables this feature, log entries are uploaded when the user reconnects.
Minimum PIN Length	The system uses the larger of the two values.

Integrating Your Windows Password with RSA SecurID

Windows password integration integrates RSA SecurID into the Windows password logon process. Users provide their Windows logon passwords only during their initial online authentication. Passwords are then stored with the users' authentication data in the internal database and, for offline authentication, in the offline data.

During subsequent authentications, users enter only their user names and SecurID passcodes. The Authentication Agent gets the Windows password from the Authentication Manager and passes it to the Windows logon system.

You enable Windows password integration on the Add Offline Authentication Policy and Edit Offline Authentication Policy pages in the Security Console.

To clear the cached copy of a user's Windows credential, clear the **Windows Password Integration** checkbox on the Assigned SecurID Tokens & Authentication Attributes page in the Security Console. When you clear this checkbox, the user must enter his or her Windows password the next time he or she logs on.

Setting Minimum Online Passcode Lengths

This setting adjusts the cryptographic strength of your offline authentication policy. RSA recommends that the minimum online passcode length setting be at least twelve characters. If your RSA SecurID tokens display six characters, for example, require your users to specify PINs that are at least six characters.

To download offline data, the user's passcode (PIN + tokencode) length must be 8 to 16 characters.

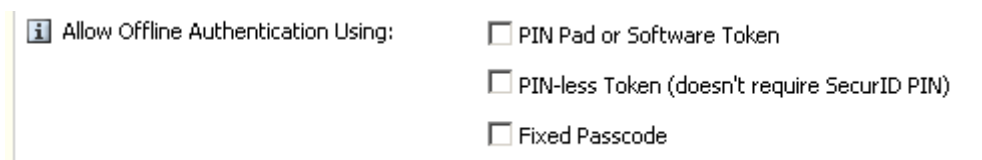
Handling Offline Authentication with Devices that Do Not Meet Security Recommendations

RSA does not recommend offline authentication for the following authenticators:

- PINPad or software tokens
- Tokens that do not require PINs
- Fixed passcodes

These authenticators are likely to contain fewer characters than required by the minimum offline passcode length setting. You can override this setting by using the Security Console to explicitly allow offline authentication using these authenticators.

The following figure shows the settings where you can explicitly allow devices that are not recommended.



Allow Offline Authentication Using:
 PIN Pad or Software Token
 PIN-less Token (doesn't require SecurID PIN)
 Fixed Passcode

This option is configured on the Add Offline Authentication Policy or Edit Offline Authentication Policy page.

Setting Offline Emergency Codes

Users can use offline emergency codes to authenticate when their computers are disconnected from the network. There are two types of offline emergency codes:

Offline emergency tokencodes. For users who have misplaced or do not have immediate access to their token.

Offline emergency passcodes. For users who have forgotten their PIN and need a full passcode.

Important: Because emergency passcodes enable authentication without a PIN, RSA recommends that you use emergency tokencodes instead. Users still must enter their PIN followed by the emergency tokencode to gain entry to their computers. Provide emergency passcodes only in situations where users have forgotten their PINs. In such cases, make sure that you properly identify the users before providing them with emergency passcodes.

The **Set Offline Emergency Code Lifetimes** field allows you to set offline emergency codes to expire after a specified amount of time.

You enable offline emergency codes through the Security Console as part of a token policy. Only users assigned to security domains with an offline authentication policy that allows offline emergency codes can use this feature.

Refreshing Users' Supplies of Offline Authentication Data

When a user's supply of offline days is low, they are automatically recharged when:

- The user authenticates to the network directly.
- After the user authenticates offline, the user connects to the network directly during the same authentication session (for example, if the Authentication Agent on the user's computer is configured to automatically reconnect after offline authentication).
- After authenticating offline, the user authenticates to the network remotely.

Even if the user's supply of offline days is full, offline days are automatically updated when:

- While authenticated online, users change their Windows password.
- While the user is connected online, an administrator issues a new policy to the Authentication Agent on the user's computer allowing emergency codes.

Offline days are not automatically recharged if the authentication session has expired (the user remains online for 24 hours or more). In this situation, the user has to recharge offline days manually. Additionally, unlocking a workstation with only a SecurID PIN does not initiate an automatic recharge.

You can also configure how many days of offline data a user is allowed to download. This is the number of days worth of token codes that are downloaded to the user's machine. This is configured in the Security Console as part of an offline authentication policy.

Setting Self-Service Troubleshooting Requirements

Note: Self-service troubleshooting policies apply to RSA Credential Manager users only.

Self-service troubleshooting policies allow you to define secondary authentication methods. A secondary authentication method allows a user to access Credential Manager even if their primary authentication method has failed. You can also use the self-service troubleshooting policy to specify lockout and unlock settings.

Important: Self-service troubleshooting policies determine the lockout criteria for Credential Manager only. To set lockout requirements for Authentication Manager, see [“Locking Users Out of the System”](#) on page 57.

Similar to the other policies, you assign self-service troubleshooting policies to security domains. The self-service troubleshooting policy assigned to a security domain dictates the Credential Manager self-service troubleshooting requirements for all of the users assigned to that security domain.

Note: Self-service troubleshooting policies assigned to upper-level security domains are not inherited by lower-level security domains. For example, if you assign a custom policy to the top-level security domain, all new security domains that you create below it in the hierarchy are still assigned the default self-service troubleshooting policy.

Self-service troubleshooting policies apply to all logon attempts regardless of how many different tokens a user uses to authenticate. For example, if a user has two failures with their software token and one failure with their hardware token, that adds up to three failed attempts.

For more detailed information on self-service troubleshooting policies, see [“Configuring Self-Service Troubleshooting for the RSA Self-Service Console”](#) on page 132.

3

Protecting Network Resources with RSA SecurID

- [Overview of RSA SecurID Authentication](#)
- [Installing Authentication Agent Software on the Resource You Want to Protect](#)
- [Creating an RSA Agent Record Using the RSA Security Console](#)
- [Creating and Installing the RSA Authentication Manager Configuration File](#)
- [Specifying Where Agents Send Authentication Requests](#)
- [Using Authentication Agents to Restrict User Access](#)
- [Deploying Tokens to Users](#)
- [Delivering Tokencodes Using Text Message or E-mail](#)
- [Preventing and Handling User Authentication Problems](#)

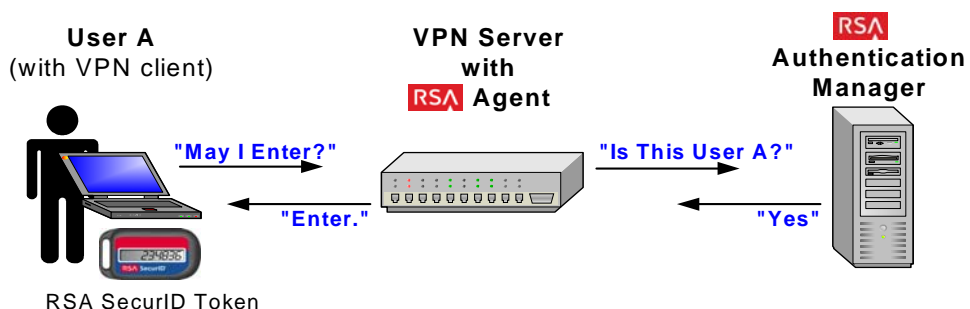
Overview of RSA SecurID Authentication

When a user successfully authenticates through RSA Authentication Manager, he or she is able to access a resource, a VPN server for example, that is protected by Authentication Manager. Authentication Manager uses authentication agents to protect network resources.

Authentication agents must be installed on each machine that you want to protect with Authentication Manager and RSA SecurID. You can either install an agent manually or use hardware that comes with preinstalled authentication agents. Authentication agents are software applications that securely pass authentication requests to and from Authentication Manager.

When a user attempts to gain access to a network resource, the agent receives the authentication request and submits it to Authentication Manager. The Authentication Manager then approves or denies the request, prompting the agent to allow or deny access to the user.

The following figure shows the flow of SecurID authentication:



Installing Authentication Agent Software on the Resource You Want to Protect

There are different types of authentication agents. The agent that you need depends on what type of resource you want to protect. For example, to protect an Apache Web server, download and install RSA Authentication Agent 5.3 for Web for Apache.

RSA provides the latest RSA Authentication Agent software for your platform at <http://www.rsa.com/node.asp?id=1174>. Included with the agent download package is an *Installation and Administration Guide* and a *Readme*. RSA recommends that you read these documents before installing the agent.

Important: For information about installing agent software, see your agent documentation.

You may also purchase products that contain embedded RSA Authentication Agent software. The software is embedded in a number of products, such as remote access servers, VPNs, firewalls, and web servers. For more information about products with embedded RSA Authentication Agents, go to <http://www.rsasecured.com>.

Creating an RSA Agent Record Using the RSA Security Console

After you install and configure authentication agent software on the machines that you want to protect, use the RSA Security Console to create an agent record in the Authentication Manager for each agent. This process is called registering the agent.

The agent record identifies the agent to the Authentication Manager and contains the following configuration information:

Hostname. The name of the machine where you installed the agent software. In most cases, the hostname must be a fully qualified domain name (machine name + domain). For example: 'mymachine.example.net'. However, if the machine is a member of a Windows workgroup, the hostname is the machine name only. For example, 'mymachine.'

If you add an authentication agent to a server node that is also running Authentication Manager, select the hostname from the list of existing server nodes.

IP Address. The IP address of the machine where you installed the agent software.

Protect IP Address. Select this option to prevent the agent auto-registration utility from reassigning the agent's IP address. For more information on agent auto-registration, see "[Allowing Agents to Automatically Add Authentication Agent Records](#)" on page 68.

Alternate IP Address. A secondary IP address for the machine where you installed the agent software. You can specify as many as necessary.

Agent Type. This can be Standard Agent or Web Agent. The default agent type is **Standard Agent**. Select **Web Agent** if you are adding an agent to a web server. Select **Standard Agent** for all other agents. This field is for informational use only, and is used primarily to simplify the task of searching for agents.

RADIUS Profile. Select a RADIUS profile for the agent.

Disabled. Select to disable the agent.

Agent May Be Accessed By. You can choose whether to allow all users to authenticate to a specific agent, or allow only users who are members of a user group that has been explicitly given permission to authenticate to the agent.

Agents that allow all users to authenticate are called unrestricted agents. Agents that require users to be members of user groups that are explicitly given permission to authenticate to the agent are called restricted agents. See [“Using Authentication Agents to Restrict User Access”](#) on page 73.

Authentication Manager Contact List. By default, authentication agents send authentication requests to the server node that responds first. That server node sends the agent an automatically maintained contact list informing the agent of other server nodes to communicate with if the original server node is offline.

You can override this default by manually assigning the agent a contact list. You should only choose this option if you have specific requirements for managing your authentication request traffic. See [“Specifying Where Agents Send Authentication Requests”](#) on page 72.

Trusted Realm Authentication. Enable the agent for trusted realm authentication. For more information on trusted realm authentication, see [“Administering Trusted Realms”](#) on page 145.

For instructions, see the Security Console Help topic “Add New Authentication Agents.”

Another way to add agents is to duplicate an existing agent. You might do this when you are adding agents with settings similar to an existing agent. For instructions, see the Security Console Help topic “Duplicate Authentication Agents.”

You may also configure the system so that agent records are added to the internal database automatically. See the following section, [“Allowing Agents to Automatically Add Authentication Agent Records.”](#)

Note: To edit an agent record after you add it to the Authentication Manager internal database, see the Security Console Help topic “Edit Authentication Agents.”

Allowing Agents to Automatically Add Authentication Agent Records

The Automated Agent Registration and Update utility (**sdadmreg.exe**), included with your RSA Authentication Agent software, enables new authentication agents to automatically add their agent record to the Authentication Manager internal database. This process is called registering the agent. Allowing authentication agents to automatically register themselves saves time and money by eliminating the need for an administrator to perform these tasks.

By default, the Automated Agent Registration and Update utility automatically runs whenever the agent host is started to allow any IP address changes to be registered in the internal database before the agent is started. This is useful for systems that use the Dynamic Host Configuration Protocol (DHCP) to assign IP addresses. If you use DHCP and do not enable this utility, you must manually update the IP addresses each time the agent host changes its IP address.

You can also run the Automated Agent Registration and Update utility manually whenever the IP address of an agent host changes to update the IP address in the internal database.

Note: The RSA Authentication Agent 6.1.2 for Microsoft Windows automatically updates the internal database with any IP address changes. If you are using this agent, you do not need to manually run the utility.

To allow agents to automatically register themselves in Authentication Manager, do the following:

- Install the Automated Agent Registration and Update utility on the agent hosts. Do this during agent installation.
For utility installation instructions or for information about manually running the utility, see your RSA Authentication Agent documentation.
- Enable agent auto-registration on the Authentication Manager Settings page in the Security Console. This enables agent auto-registration on the Authentication Manager server.
For configuration instructions, see the Security Console Help topic “Allowing Agents to Register Themselves with RSA Authentication Manager.”

Default Agent Settings

When the Automated Agent Registration and Update utility is run on a newly installed, unregistered agent, a record for the agent is created in the internal database. By default, the agent has the following characteristics:

Disabled. The agent is unable to process authentication requests.

Unrestricted. All users are allowed to authenticate with this agent.

Has not been passed the node secret. The node secret is a shared secret known only to the authentication agent and the Authentication Manager.

IP address is unprotected. The agent IP address is not protected by default, so the agent auto-registration utility can reassign the IP address if the agent is inactive. Select this option if you want to prevent the agent auto-registration utility from reassigning the IP address to another agent.

If these default settings are not appropriate for the new agent, edit the agent record to change the settings. For instructions, see the Security Console Help topic “Edit Authentication Agents.”

Agent Auto-Registration and Denial of Service (DOS)

It is important that you protect your critical IT infrastructure from potential Denial of Service (DOS) attacks. To reduce the vulnerability of your system:

- Disable agent auto-registration on critical machines such as e-mail and VPN servers.
- In your IT infrastructure, give critical agents static IP addresses.
- Protect IP addresses within Authentication Manager. To do this, select Protect IP Address on the Authentication Agent page in the Security Console. For more information, see [“Creating an RSA Agent Record Using the RSA Security Console”](#) on page 66.

Agent Auto-Registration and Multi-Realm Deployments

When an Authentication Manager deployment has more than one realm, only one of the realms can have full agent auto-registration support. Full agent auto-registration support means that the auto-registration service adds new authentication agents and updates the IP addresses of existing authentication agents. The remaining realms have only partial agent auto-registration support. With partial support, the auto-registration service only updates the existing agent record. It does not add new agent records to the database.

To designate the default realm for full auto-registration support, do the following:

1. Enable agent auto-registration on the Authentication Manager Settings page in the Security Console. On that page, select the default realm from the list of available realms.
2. For each realm other than the default realm, go to the Authentication Manager Realm Level Configuration page in the Security Console and enable that realm for agent auto-registration.

For complete instructions, see the Security Console Help topics “Configure your RSA Authentication Manager Deployment” and “Allow Agents to Register Themselves with RSA Authentication Manager.”

Changing Agent Auto-Registration to Legacy Mode

Use the Store utility, `store`, to switch between enhanced mode, the way agent auto-registration works in the current version of Authentication Manager, and legacy mode, the way auto-registration worked in previous versions. When you install the current version, enhanced mode is set by default. Switching from enhanced mode to legacy mode allows you to run agent auto-registration with its previous functionality after upgrading Authentication Manager.

Auto-registration running in enhanced mode is more flexible when updating agent IP addresses. This allows Authentication Manager to make agent IP address changes efficiently in a Dynamic Host Configuration Protocol (DHCP) environment. Because this new efficiency changes the way that auto-registration is accomplished, it may not meet your security requirements. By switching between these two modes, you can evaluate how enhanced mode is different from legacy mode. Based on the needs of your system, you can choose whether or not to take advantage of enhanced mode.

To switch from enhanced mode to legacy mode:

1. On the primary instance, open a new command shell and change directories to **`RSA_AM_HOME/utls`**.
2. Type:


```
rsutil store
--action config
auth_manager.agent_protocol.flexible_agent_auto_reg FALSE
global
```
3. When prompted, enter the master password of the encrypted properties file.

Important: Although it is possible to enter the master password on the command line along with the other options, this creates a potential security vulnerability. RSA recommends that you enter the master password only when the utility presents a prompt.

To switch from legacy mode to enhanced mode:

1. On the primary instance, open a new command shell and change directories to **`RSA_AM_HOME/utls`**.
2. Type:


```
rsutil store
--action config
auth_manager.agent_protocol.flexible_agent_auto_reg TRUE
global
```
3. When prompted, enter the master password of the encrypted properties file.

Creating and Installing the RSA Authentication Manager Configuration File

The Authentication Manager configuration file contains the IP addresses of Authentication Manager server nodes with which an agent can communicate.

You must perform the following tasks for each agent in your deployment:

- Use the Security Console to generate a server configuration file. For instructions, see the Security Console Help topic “Generate the RSA Authentication Manager Configuration File.”
- Install the server configuration file (**sdconf.rec**) on the machine where an authentication agent is installed, called the agent host. For instructions on installing the configuration file, see your agent documentation.

Authentication agents use the server node IP addresses in the configuration file to establish initial contact with the Authentication Manager. One of the IP addresses listed in the configuration file must be available for the first authentication.

After an agent makes initial contact with the Authentication Manager, the Authentication Manager provides the agent with a new list of server nodes, called the contact list, where the agent can direct authentication requests. See the following section, “[Specifying Where Agents Send Authentication Requests.](#)”

If an agent cannot contact any of the server nodes in the contact list, the agent reverts to the Authentication Manager configuration file and uses one of the IP addresses in the configuration file to reconnect with the Authentication Manager.

The Authentication Manager automatically populates the Authentication Manager configuration file with a list of IP addresses, up to the maximum of 11, as follows:

- If you have only one instance in your deployment, an IP address for each server node is included until the list reaches the limit of 11.
- If you have multiple instances in your deployment, an IP address is included for one server node in each instance. IP addresses from each instance are added until the list reaches the limit of 11.

The configuration file also contains port numbers for the Authentication Service and the Agent Auto-Registration Service. You can edit these port numbers on the Authentication Manager Settings page in the Security Console. For instructions, see the Security Console Help topic “Configure RSA Authentication Manager.”

Specifying Where Agents Send Authentication Requests

Depending on your license type, your Authentication Manager deployment can have a primary instance, as well as multiple replica instances, each of which may have multiple server nodes that process authentication requests. To increase the efficiency of your deployment, use contact lists to route authentication requests from agents to the server nodes that can respond the quickest.

Contact lists are ordered lists of server nodes available to accept authentication requests, and are created either automatically by the Authentication Manager, or manually by an administrator.

Automatic contact lists. An automatic contact list is assigned to each instance in your deployment. The list contains the IP addresses of each server node in the instance the contact list is assigned to, and the IP address of one server node from each other instance in your deployment, up to a limit of 11. Agents are sent automatic contact lists by default.

These lists are automatically maintained by the Authentication Manager, and are automatically updated each time a new server node is added to the deployment. When the list is updated, a time stamp associated with the list is also updated. Agents use this time stamp to determine when to request an updated list.

The Super Admin can edit an automatic contact list on the Edit Authentication Manager Contact List page in the Security Console. Any edits that you make to an automatic contact list may be overwritten when a new server node is added to the deployment.

Manual contact lists. The Super Admin maintains manual contact lists. They must be updated manually to reflect the most recent list of server nodes. Manual lists can contain the IP address of any server node in the deployment, up to a limit of 11.

You create manual server lists on the Add New Authentication Manager Contact List page in the Security Console. You can edit a manual contact list on the Edit Authentication Manager Contact List page in the Security Console. For instructions, see the Security Console Help topics “Add a Manual Contact List” and “Edit Manual Contact Lists.”

Authentication Manager uses contact lists to determine to which server node authentication requests are sent. Contact lists are sent to each agent by Authentication Manager after the initial contact between the agent and Authentication Manager.

Agents request new contact lists as a part of subsequent authentications. Periodically, the agent reviews all the server nodes listed in the contact list to determine where to send authentication requests. The agent uses metrics, such as the amount of time it takes the server node to respond to authentication requests, to determine where to send requests.

If none of the servers on the contact list respond to authentication requests, the agent reverts to the Authentication Manager configuration file and uses one of the IP addresses in the configuration file to reconnect with the Authentication Manager.

For many organizations, automatic contact lists are sufficient. However, you may choose to create a manual contact list if you have a specific way that you want to route authentication requests.

For example, suppose that you are an administrator at a company that has Boston, New York, and San Jose locations. The New York and San Jose locations are small and all authentications are routed to Authentication Manager replica instances at each site. The Boston location, however, is largest, and the primary instance at that location handles all of your Boston location users, as well as all VPN requests from external users. You may choose to create a manual contact list that routes authentication requests to all of your server nodes, except the database sever. This leaves the database server free to replicate data to your replica instances in New York and San Jose.

For instructions, see the Security Console Help topics, “Manage the RSA Authentication Manager Contact List,” “Assign a Contact List to an Authentication Agent,” and “Edit Manual Contact Lists.”

Using Authentication Agents to Restrict User Access

Authentication Manager allows you to configure authentication agents in two ways:

Unrestricted agents. Unrestricted agents process all authentication requests from all users in the same realm as the agent. They eliminate the need to grant access to user groups on the agent.

Restricted agents. Restricted agents only process authentication requests from users who are members of user groups that have been granted access to the agent. Users who are not members of a permitted user group cannot use the restricted agent to authenticate.

For example, when an authentication request comes from a restricted agent, Authentication Manager checks to see if the request comes from a user that is a member of a user group that is granted access to the agent. If a user is a member, he or she is authenticated and access is granted. If a user is not a member, he or she is not authenticated and access is denied.

You can grant access to existing user groups, or you can create new user groups specifically for use with restricted agents.

Important: Active Directory supports multiple types of groups. When configured to use Active Directory groups, Authentication Manager only supports Universal groups. When you view the Active Directory groups from the Security Console, the Security Console displays all groups, regardless of type. If you select a group from this list to activate users on restricted agents, make sure that you select a Universal group. Use the Active Directory Users and Computers MMC Console to examine the type of group. If you use any other type of Active Directory group, the user cannot authenticate.

Resources protected by restricted agents are considered to be more secure because, rather than allowing access to any user in the identity source, only a subset of users are allowed access. If you want to limit access to certain network resources, protect those resources with restricted agents.

New agents are unrestricted by default. The use of restricted agents increases administrative overhead because administrators need to associate user groups with the agents. In contrast, any user in your identity source can be authenticated on an unrestricted agent without explicitly being activated.

For more information, see the following section, [“Granting Access to Restricted Agents Using User Groups.”](#)

Granting Access to Restricted Agents Using User Groups

To provide access to restricted agents, you must first create the appropriate user group. A user group is a group of one or more users, all belonging to the same identity source. User groups have the following characteristics:

- They can be made up of one or more user groups.
- They can occur across security domains. This means that users in security domain A and users in security domain B can both be members of the same user group and thus access the same protected resources.
- A user can be a member of more than one user group.

You can create user groups in one of two ways:

- Use the **User Groups** option in the **Identity** menu in the Security Console. For instructions on how to create user groups using the Security Console, see the Security Console Help topic “Add New User Groups.”
- For external data sources such as Active Directory, create the groups using the directory’s user interface.

After you create the user groups for your restricted agents, use the Security Console to nest user groups within other user groups (if applicable to your deployment), or add users to the various user groups. For instructions, see the Security Console Help topic “Add Users to User Groups.”

Note: The Security Console cannot display a user’s primary Active Directory user group, such as Domain Users. The group appears empty even though it has members.

After creating the user groups and adding users, give the user groups access to the restricted agents.

For example, assume that you are a system administrator. Your company VPN is a restricted agent so you need to create a user group whose members can access this agent. The members of the VPN user group do not need to be in the same security domain, but they do have to belong to the same identity source. After creating the user group, grant the group access to the agent on the VPN machine. Only these users can access the restricted agent.

For instructions, see the Security Console Help topics “Add Users to User Groups” and “Grant Access to Restricted Authentication Agents.”

To view the list of user groups that have access to a restricted agent, see the Security Console Help topic “View User Groups with Access to a Restricted Authentication Agent.”

Note: You can also configure your user groups so that the users in the group can only access the restricted agents at certain times of day. For more information, see the following section, “[Setting Restricted Access Times for User Groups.](#)”

Setting Restricted Access Times for User Groups

You can assign restricted access times to your user groups. Restricted access times allow you to control the days and hours in which a user group can access a restricted agent.

Note: Restricted access times apply to restricted agents only.

For example, assume that you have a user group with Outlook Web Access (OWA). Because OWA is for work purposes only, you decide that members of the user group should only be able to access the OWA agent during regular business hours. To enforce these time constraints, you create a Time Restricted Access policy for the user group. The policy specifies the days and times that the user can access OWA.

To use the Security Console to configure a Time Restricted Access policy for a user group, use the **Restricted Access Times** option in the user group Context menu. Once on the Time Restricted Access page, you can select the days and times of allowable access. You can also use one of the Access Time templates. To use the example above, you may choose to use the “8am - 5pm Weekdays” template instead of configuring the policy manually.

Note: Fractional time zones are not available from the Access Times drop-down menu. You must select an available time zone closest to the desired fractional time zone.

Deploying Tokens to Users

Deploy tokens to users to allow them to authenticate using Authentication Manager.

A token is a device used to deliver a tokencode to the user. A tokencode is a pseudorandom number, usually six digits in length. A tokencode, combined with the user's PIN, is one way in which a user can authenticate through Authentication Manager.

Note: You can also deliver tokencodes using text message or e-mail, instead of assigning the user a token. For more information, see “[Delivering Tokencodes Using Text Message or E-mail](#)” on page 85.

Token Types

There are two kinds of SecurID tokens, hardware tokens and software tokens:

- Hardware tokens are usually key fobs or USB keys that display the tokencode.
- Software tokens and their accompanying application are installed on devices such as Palm Pilots and BlackBerries. Once installed in a device, the application can be used to display the tokencode.

While the two types of tokens perform the same function, the situations in which you use them can be very different.

For example, suppose your organization has internal users who must authenticate with a SecurID token when they log on to their desktop computer, as well as a remote sales force whose members must authenticate with a SecurID token when they log on to their laptop computers.

You might choose to distribute hardware tokens to your internal users. Because they generally log on at their desktop machine each day, the internal users are less likely to lose their tokens than someone who travels frequently. Many users choose to attach the key fob to their keychain, so that as long as they have their car keys, they have their token.

You might choose to distribute software tokens to your remote sales force. Your sales force is on the go constantly, and with a software token installed directly on a PDA or cell phone, they will be less likely to leave it at home, or lose it in an airport. As long as they have their PDA, they have their token.

Tokencode Delivery Methods

When a user authenticates with a token, Authentication Manager matches the tokencode entered by the user to the tokencode maintained within Authentication Manager. When the two tokencodes match, authentication is successful.

Hardware and software tokens deliver their tokencodes in one of two ways: time-based or event-based. The tokencode delivery dictates how Authentication Manager verifies the tokencode and authenticates the user:

Time-based. A time-based token displays a tokencode that automatically changes at a set interval, typically every 60 seconds.

For time-based tokens, the tokencodes are kept synchronized with Authentication Manager based on their internal “clocks” or time. So when the tokencode advances every 60 seconds, the corresponding tokencode in Authentication Manager advances as well. When a user authenticates, Authentication Manager matches the tokencodes based on time.

Event-based. An event-based token displays a tokencode only when initiated by the user. For example, an RSA SecurID Display Card only displays a tokencode when the user presses the appropriate button.

For event-based tokens, the tokencodes are kept synchronized with Authentication Manager based on tokencode count. For example, assume that you just assigned an event-based token to a user. To authenticate, the user presses the button on the token and receives the first tokencode. This advances the token's count to one. When the user attempts to authenticate with tokencode one, Authentication Manager matches the entered tokencode to its own tokencode one. The authentication is successful and the user gains access. At this point, Authentication Manager advances its tokencode count to two. The next time the user needs to authenticate, he or she presses the button on the token to get tokencode two, and the cycle continues.

Important: Because successful authentication attempts are based on count, it is very important that the user only advances the tokencode when they need to authenticate. Needless advancing can cause the token to become out-of-sync with Authentication Manager. For information on resynchronizing tokens, see [“Resynchronizing Tokens”](#) on page 103.

So a hardware token can be time-based or event-based, and a software token can be time-based or event-based. Regardless of the tokencode delivery method, each tokencode can only be used once.

Note: Event-based tokens cannot be used for offline authentication.

Deployment Steps

To successfully deploy hardware and software tokens to users, you must perform the following steps:

1. Import the hardware or software tokens to Authentication Manager using the Security Console.
2. Optional. Transfer token records to other security domains. You may want to do this for administrative reasons.
3. Assign the hardware or software tokens to users.
4. Distribute the tokens to users.
 - Hardware tokens - distribute the tokens to the users.
 - Software tokens - electronically deliver the software tokens to the assigned users in a token file or using remote token-key generation (CT-KIP).

Note: The deployment steps are the same for time-based and event-based tokens.

All of these steps are described in detail in the sections that follow.

Importing Hardware and Software Token Records

Before you can assign tokens to users, use the Security Console to import the token records into the internal database.

Hardware tokens are shipped with associated token records stored as XML files. Software token records are shipped as XML files. You import token records on the Import SecurID Tokens Job page in the Security Console.

When you import token records, you must select the security domain where you want to import the token records. You can import token records into any security domain that is included in the scope of your administrative role. To administer the token records, administrators must have an administrative role that includes this security domain, and grants permission to administer tokens.

Token record XML files may be password protected when you receive them. Be sure to get the password from RSA before you try to import the token records.

Note: When importing tokens, you can choose to ignore or overwrite duplicate tokens. If you choose to overwrite duplicate tokens, there are certain cases when a duplicate will not get overwritten. For the complete list of exceptions, see the Security Console Help topic “Import Tokens.”

For instructions, see the Security Console Help topic “Import Tokens.”

After you import the token records, you can view them in the Security Console, and assign them to users.

Re-Importing Token Records

There are times when you might need to re-import a token record. For example, you might need to re-import a token record that was deleted. If you are re-importing token records for event-based tokens, you must resynchronize the tokens before reassigning them.

For more information on resynchronizing event-based tokens, see [“Resynchronizing Tokens”](#) on page 103.

Transferring Hardware and Software Token Records to Other Security Domains

Depending upon how you have your organizational hierarchy configured, you may want to move tokens from one security domain to another. You can do this through the the Security Console.

Transferring token records allows you to move them in or out of an administrator's scope, or to move them to a security domain associated with the location where the tokens will be used.

By default, token records are imported into the top-level security domain. If you have more than one security domain in your deployment, you can transfer token records from one security domain to another.

For example, assume an organization has created security domains for each of its geographic locations—Boston, New York, and San Jose. Hardware tokens are shipped to each of the locations so that they can be assigned and distributed to users in each location by an on-site administrator. Because the scope of the administrators that assign tokens at each location is limited to their respective security domain, a Super Admin transfers token records to each of the security domains so that they can assign the tokens. After the token records are transferred, the on-site administrators can view the token records and assign the tokens to users.

Assigning and Unassigning Hardware and Software Tokens

Use the Security Console to assign hardware and software tokens to users. A token assigned to a user can be used by that user to authenticate.

Before you can assign tokens to users, you must:

- Import the token records from the XML file to the internal database.
- Make sure a user record exists in Authentication Manager for each user to whom you want to assign a token.

Note: A maximum of three tokens can be assigned to each user. If you attempt to assign more than three tokens at the same time, no tokens are assigned. For example, if a user has no assigned tokens, and you attempt to assign four tokens, no tokens are assigned to the user.

There are two ways to assign a token through the Security Console:

- Select **Assign More** or **Assign Next Available SecurID Token** in the user Context menu on the Users page.
- Select **Assign to User** in the token Context menu. This option only appears for unassigned tokens.

For complete instructions, see the Security Console Help topics “Assign Hardware Tokens” and “Assign Software Tokens.”

After you assign a token to a user, distribute the token to the user. For hardware tokens, see “[Distributing Hardware Tokens to Users](#)” on page 80. For software tokens, see “[Distributing Software Tokens to Users](#)” on page 81.

Tokens Configured to Not Require PINs

Authentication Manager supports authentication with tokens that are configured so that they do not require a PIN. To authenticate, instead of entering the PIN followed by the tokencode, the user enters only the tokencode displayed on the token.

Note: Tokens that do not require PINs are not as secure as tokens that require PINs. RSA recommends that you configure all tokens to require a PIN.

Authenticating with just a tokencode is useful in situations such as:

- When a token is stored on a smart card and must be unlocked by the user with a PIN
- When a software token is on a desktop and must be unlocked with a password

In these situations, the resource is protected by two-factor authentication without the user having to enter two different PINs.

When assigning a token, you can configure both hardware and software tokens so that they do not require PINs. For instructions, see the Security Console Help topic “Authenticate without an RSA SecurID PIN.”

Distributing Hardware Tokens to Users

Because hardware tokens are physical devices, you must deliver them to users before they can be used to authenticate.

If your organization has a single location, the fastest and most secure method is to have users pick up tokens at a central location.

If your organization has multiple locations, consider having administrative personnel at each site distribute the tokens. Alternatively, have your administrative staff travel to different locations at pre-announced times. The advantages of this method are the assurance that the hardware tokens are delivered to the right users and that they work when users receive them.

Another distribution method is to mail tokens to users. Mailing hardware tokens through interoffice mail, post, or overnight express, for example, might be more practical for your organization. However, this usually involves more up-front work, such as developing a process for generating mailing labels, and verifying that users receive their tokens, to ensure success.

RSA recommends that you only mail disabled tokens, which can be enabled after receipt by the correct user. Send information about how to enable tokens separately from the actual tokens or make it accessible only from a secure location. You may also want to consider grouping users so mailing can be accomplished in a controlled manner.

Ultimately, you may decide to use a combination of these delivery methods. For example, if you must distribute enabled tokens to assigned users, be sure to use secure channels, such as having them delivered in person by trusted staff members.

Distributing Software Tokens to Users

Distributing a software token is a different process than distributing a hardware token. Because a software token is installed on a device and cannot be mailed, distribution is electronic, and involves generating a token file and delivering the token file to the user.

Note: Before distributing the software token, make sure that you have imported the token records and assigned the token to a user.

There are four steps in the distribution process:

1. Make sure that the user has the token application. The token application is installed on the device and displays the tokencodes on the device screen. To get a token application, go to <http://www.rsa.com/node.asp?id=1313>. Installation instructions are included in the token application download kit.
2. Distribute the software token file. Software token files (.sdtid) are generated using the Security Console, and must be distributed to users and installed on desktops and handheld devices. These files can be distributed in two ways:
 - Token file (XML) - Save the software token to an XML file, and deliver it through secure e-mail or other electronic medium.
 - CT-KIP (Remote Token-Key Generation) - Use the Cryptographic Token-Key Initialization Protocol (CT-KIP). This option can only be used with CT-KIP-capable SecurID software tokens. A CT-KIP-capable SecurID software token is a 128 bit token.
3. Deliver the token file to the user through secure e-mail or other secure means. If using CT-KIP, provide the appropriate URL.
4. Instruct the user to install the software token on his or her device.

Instructions for distributing software tokens by file and CT-KIP are in the sections that follow.

Distributing Software Tokens By Token File (XML)

When you distribute software tokens by token file, you can e-mail the token file to the user who can then download the file to install the token. Token files are in XML.

You need the following token information when distributing software tokens by token file:

Note: Different token types require different sets of token information. Depending on the type of token you are distributing, you might not need all of the information described below.

Software Token Device Type. The type of device on which the token is being installed. An RSA SecurID Toolbar Token is an example of a software token device type. You have to select the device type and enter information for the device specific attributes.

You can add additional software token types to Authentication Manager. For more information, see [“Adding Additional Software Token Device Types to Your Deployment”](#) on page 120.

Device Nickname. The **Device Nickname** field allows a user to assign a user-friendly name to the software token. For example, a user might name software tokens “Office Token” or “Home Token” to differentiate between the tokens he or she uses at home and the office.

Binding a Software Token to a Device. RSA software tokens include a predefined field named **Device Serial Number**. When you issue the software token to a user, you can enter the serial number of the device in this field, which binds the issued token to the specific device with the corresponding serial number. A token that is bound to a specific device cannot be installed on any other device.

Software Token Selection Criteria. Know which tokens you want to distribute. You can search by security domain, token file format, serial number, and other token data.

Method for Issuing Software Tokens. You can select from the following methods for issuing software tokens:

- Multiple tokens per file. Authentication Manager packs up all token records into a single .sdtid file, and adds the .sdtid file to a .zip archive when it is downloaded.
- One token per file. One software token record is written to an .sdtid file.

Enabling Copy Protection. The Enable Copy Protection option ensures that the software token cannot be copied or moved from the directory in which it is installed on a user’s computer or other device. By default, the Copy Protection option is enabled. RSA strongly recommends that you use copy protection.

Note: Copy protection creates a system fingerprint of the user’s device and associates this information with the software token. When a device is repaired or upgraded, this information changes. Software tokens must be reissued if a user’s computer hardware or device is repaired or upgraded.

Password Protection. When you issue software tokens, you can select from the following protection methods:

- **Password.** Enter a single password of your choice that applies to all software tokens that you issue.
- **User ID.** The user's default logon ID is used as the password.
- **Combination.** The user's default logon is appended to the password that you enter.

When users install the software token on their device, they are prompted for the User ID, password, or both. Passwords prevent unauthorized people from intercepting and using the software tokens. This password is only used when installing the software token.

RSA strongly recommends that you protect the software token files with passwords. You can assign passwords to the software token files as part of the issuing process. Software Token 3.0 passwords can be up to 24 characters. Software Token 2.0 passwords can be up to 8 characters.

Important: If you protect software tokens with a password, be sure to communicate the password to the user in a secure manner. For example, tell the user verbally, and do not write down the password.

Regenerating Software Tokens. Regenerating a software token changes the sequence of numbers generated by the token file.

When you regenerate the token, devices with the token already installed can no longer use it to authenticate.

Regenerating a token allows you to reuse the software token without fear that an old installation of the token will be used by an unauthorized person to authenticate.

Do this when you reissue a software token, move a software token from one device to another, or if a device containing a software token is lost.

You regenerate tokens as part of the issuing process on the Issue Software Tokens page in the Security Console.

When you distribute software token files, you can complete the operation for one or more software tokens at a time. Choose one of the following distribution methods:

- To distribute one or more software tokens at a time, use the **Distribute Software Tokens Job** option in the **Authentication > SecurID Tokens** menu. From there, select **Add New > Issue Software Token Files**.
- You can distribute software tokens individually from the Edit Tokens page. Click **Save and Distribute** to follow the process.

After choosing individual or multiple distribution, do the following:

- E-mail the token file to the user.
- Instruct the user to download the token file to his or her device.
- Instruct the user to download the token application. The token application is installed on the device, and displays the tokencodes on the device screen. Token applications are available from the following URL:
<http://www.rsa.com/node.asp?id=1313>.

Installation instructions are included in the token application download kit.

Distributing Software Tokens Using Remote Token-Key Generation (CT-KIP)

Note: Before distributing the software token, make sure that you have imported the token records and assigned a CT-KIP-capable token to the user.

When you assign a CT-KIP-capable software token to a user, you can optionally select to use remote token-key generation (CT-KIP) to deploy a token on user devices.

CT-KIP is more secure than other delivery methods because it enables Authentication Manager and the device that hosts the software token, such as a web browser, to simultaneously and securely generate the same token file on a device and the Authentication Manager.

This allows you to put a token file on a user's device without actually sending the token file through e-mail or putting it on external electronic media. This greatly decreases the chances that the token file will be intercepted by an unauthorized person.

You need the following information when distributing software tokens using CT-KIP:

Software Token Device Type. The type of device on which the token is being installed. An RSA SecurID Toolbar Token is an example of a software token device type. You have to select the device type and enter information for the device specific attributes.

You can add additional software token types to Authentication Manager. For more information, see "[Adding Additional Software Token Device Types to Your Deployment](#)" on page 120.

Device Nickname. The **Device Nickname** field allows a user to assign a user-friendly name to the software token. For example, a user might name software tokens "Office Token" or "Home Token" to differentiate between the tokens he or she uses at home and the office.

Binding a Software Token to a Device. RSA software tokens include a predefined field named **Device Serial Number**. When you issue the software token to a user, you can enter the serial number of the device in this field, which binds the issued token to the specific device with the corresponding serial number. A token that is bound to a specific device cannot be installed on any other device.

CT-KIP Activation Code. Choose the format of the CT-KIP activation code. The code can be system generated, or you can choose to use a device-specific attribute as the activation code.

Software Token Selection Criteria. Know which tokens you want to distribute. You can search by security domain, token file format, serial number, and other token data.

When you distribute software token files, you can complete the operation for one or more software tokens at a time. Choose one of the following distribution methods:

- To distribute one or more software tokens at a time, use the **Distribute Software Tokens Job** option in the **Authentication > SecurID Tokens** menu. From there, select the **Add New > Generate Token CT-KIP Credentials** option.
- You can distribute software tokens individually from the Edit Tokens page. Click **Save and Distribute** to follow the process.

Important: When you select the RSA SecurID Toolbar Token from the **Software Token Type** menu, be sure to enter the correct serial number in the **Device Serial Number** field. If you enter the serial number incorrectly, the token does not load properly. If you are unsure of the serial number, leave this field blank.

After choosing individual or multiple distribution, do the following:

- Distribute the token-key generation URL to the assigned user through secure e-mail or other secure means.
- Instruct the user to click the URL or to paste it into a browser window running on the user's device. This step generates a token file and loads it on the device.
- Instruct the user to download the token application. The token application is installed on the device, and displays the tokencodes on the device screen. Token applications are available from the following URL:
<http://www.rsa.com/node.asp?id=1313>.
Installation instructions are included in the token application download kit.

Delivering Tokencodes Using Text Message or E-mail

In addition to receiving tokencodes on hardware and software tokens, users can receive tokencodes using cell phones or personal e-mail. You can deliver tokencodes to a cell phone using Short Message Service (SMS) or to an e-mail address using Simple Mail Transfer Protocol (SMTP). Tokencodes delivered by SMS or SMTP are called on-demand tokencodes.

Similar to the tokencode generated by a hardware or software token, you use on-demand tokencodes with a PIN to achieve two-factor authentication. The difference is that on-demand tokencodes are user-initiated, as Authentication Manager only sends a tokencode to the user when it receives a user request.

Important: RSA SecurID hardware tokens offer the highest level of security. Other methods of tokencode delivery, such as software tokens and on-demand tokencodes, may be more convenient for some users, but do not provide the same level of security as a hardware token. RSA recommends using hardware tokens.

Users must be enabled to receive on-demand tokencodes before they can request them. You can use the Security Console to enable users for on-demand tokencodes. Users who are enabled for Credential Manager can also use Credential Manager to request the on-demand tokencode service.

Once enabled for the on-demand tokencode service, the user can request a tokencode in two ways:

- If the user is enabled for Credential Manager, the user can log on to Credential Manager and request a tokencode.
- If the user is not enabled for Credential Manager, the user can go to the Credential Manager home page and click **On-Demand Tokencode Service**. The link directs the user to the screen where they can request a tokencode.

In both cases, the user enters his or her on-demand tokencode PIN to request the tokencode. The on-demand tokencode is sent to the user's cell phone or e-mail address. The on-demand tokencode can only be used once, and it expires after the lifetime you specify when configuring Authentication Manager for on-demand tokencodes.

Note: The delivery time for the tokencode depends on the mail server or SMS service.

If you plan on enabling users to request and receive on-demand tokencodes, you need to first set up and configure the service in Authentication Manager. You need to perform the following steps:

1. If you plan on delivering on-demand tokencodes to user cell phones, establish a relationship with Clickatell, the Authentication Manager SMS service provider. For more information, go to www.clickatell.com/rsa/secuid.php.
2. Configure Authentication Manager for on-demand authentication. You must configure Authentication Manager for SMS and SMTP integration. See [“Configuring RSA Authentication Manager for On-Demand Authentication”](#) on page 87.
If you want to choose a different service provider, see [“Changing the SMS Service Provider”](#) on page 89.
3. Use the Security Console to enable users to receive on-demand tokencodes. See [“Enabling Users for On-Demand Authentication”](#) on page 89.
4. Set PINs for SMS and SMTP authentication. See [“Setting PINs for On-Demand Tokencodes”](#) on page 90.

Important: On-demand tokencodes cannot be used for trusted realm authentication, and users cannot use an alias with an on-demand tokencode.

Configuring RSA Authentication Manager for On-Demand Authentication

You must configure Authentication Manager to send on-demand tokencodes. To use the Security Console to configure Authentication Manager for on-demand authentication, go to the On-Demand Tokencodes page in the Authentication Manager Component Configuration menu. There you can configure the Clickatell plug-in, configure tokencode delivery by cell phone or e-mail address, and specify the text of the tokencode message.

Note: When enabling a user to receive on-demand tokencodes, you can only select one delivery method, cell phone or e-mail.

More information is provided in the sections that follow.

Configuring Tokencode Delivery by Text Message

Configure Authentication Manager to deliver on-demand tokencodes to a user's cell phone.

Delivery by SMS. Select to enable Authentication Manager for SMS authentication.

User Attribute to Provide SMS Destination. Select the user attribute that will maintain SMS data.

If you use the internal database for user information, you can map to an attribute there, such as phone number, or create a custom attribute. If you use an external identity source, you can choose an attribute that is mapped to an attribute in the external identity source (phone number, for example).

Default Country Code. Select the country code for the phone number. This field is optional.

SMS Provider Integration. Enter service provider information such as API ID, account user name and password, and proxy server information.

Note: This information is specific to Clickatell, the default SMS service provider.

For more information, see the Security Console Help topic "Configure the On-Demand Tokencode Service."

Configuring Tokencode Delivery by E-mail

Configure Authentication Manager to send on-demand tokencodes to a user's e-mail address.

Important: Before configuring tokencode delivery by e-mail, you must configure the e-mail server connection for each instance. To configure the instance server connection, select **Mail Server (SMTP)** from the Instances Context menu.

Delivery by E-mail. Select to enable Authentication Manager for SMTP authentication.

User Attribute to Provide E-mail Destination. Select the user attribute for SMTP data.

If you use the internal database for user information, you can map to an attribute there, such as e-mail address, or create a custom attribute. If you use an external identity source, you can choose an attribute that is mapped to an attribute in the external identity source (e-mail address, for example).

E-mail Server Connection. Provides instructions on configuring the e-mail server.

For more information, see the Security Console Help topic "Configure the On-Demand Tokencode Service."

Configuring the Text for Tokencode Messages

You can customize the tokencode message that is sent to the user's cell phone or e-mail address. Configure the message in the General Settings section of the On-Demand Tokencodes Configuration page:

On-Demand Tokencode Message. Enter the text message for on-demand tokencode delivery.

For more information, see the Security Console Help topic "Configure the On-Demand Tokencode Service."

Configuring Tokencode Lifetime

You can configure the lifetime for the on-demand tokencode. Once the tokencode expires, the user can no longer use it to authenticate. Configure the tokencode lifetime in the General Settings section of the On-demand Tokencodes Configuration page:

On-Demand Tokencode Lifetime. Enter the desired lifetime in minutes.

For more information, see the Security Console Help topic "Configure the On-Demand Tokencode Service."

Testing the Provider Configuration

You can send test SMS messages to a cell phone to make sure that you have the plug-in and provider information configured correctly.

To test your plug-in configuration, click the **Test SMS Provider Integration** button on the On-Demand Tokencodes Configuration page in the Security Console. There you can enter a cell phone number and test the connection.

For more information, see the Security Console Help topic “Test Your SMS Plug-In.”

Changing the SMS Service Provider

Authentication Manager uses plug-ins to integrate with Clickatell, the service provider. If you want to use a different service provider, you need to install a new plug-in.

To implement a custom plug-in for another provider, contact RSA Professional Services Organization (PSO) for more information.

Enabling Users for On-Demand Authentication

You must enable your users for on-demand authentication. Unlike a hardware or software token, you do not assign an on-demand authenticator. Instead, you enable the user to request and receive on-demand tokencodes.

Note: Users who are enabled on Credential Manager can use Credential Manager to request the on-demand tokencode service.

To use the Security Console to enable a user for on-demand tokencodes, select a user, go to the SecurID Tokens page, and select **Enable user to request and receive on-demand tokencodes**. You can configure the following information:

Send On-Demand Tokens To. Decide whether to send tokencodes to the user's e-mail address or cell phone.

Phone number. The destination phone number information (for cell phone delivery only).

Note: The label for this field is the name of the attribute to which you mapped.

E-mail Address. The destination e-mail address information (for e-mail delivery only).

Note: The label for this field is the name of the attribute to which you mapped.

Associated PIN. On-demand tokencodes use a different PIN than SecurID tokens. Clear the existing PIN or specify a new, temporary PIN. You must communicate the PIN to the user.

If you choose not to set a PIN, the user can request or set up the PIN through Credential Manager.

Note: Similar to RSA SecurID PINs, PINs for on-demand tokencodes are governed by the user's token policy.

On-Demand Service Lifetime. The on-demand tokencode service lifetime. Select no expiration or set an expiration date for the service. When the service expires, the user can no longer request on-demand tokencodes.

Last Used to Authenticate. Read-only field that displays the date of the last authentication with an on-demand tokencode.

Note: When enabling a user to receive on-demand tokencodes, you can only select one delivery method, cell phone or e-mail.

You can also enable or disable multiple users to request and receive on-demand tokencodes. To do this, go to **Authentication > On-Demand Tokencodes > Enable Users**, select the appropriate users, and select **Enable for On-Demand Tokencodes** from the Action menu.

Note: To enable or disable multiple users at once, the users must already have their destination addresses (phone number or e-mail address) set up.

For complete instructions, see the Security Console Help topic "Enable On-Demand Tokencodes for a User."

Setting PINs for On-Demand Tokencodes

On-demand tokencodes use a different PIN than SecurID tokens. Users need a specific PIN to use with their on-demand tokencodes. You can specify the PIN and communicate it to the user, or the user can request and configure the PIN using Credential Manager.

Note: Similar to RSA SecurID PINs, PINs for on-demand tokencodes are governed by the user's token policy.

To use the Security Console to specify a PIN, select **Clear existing PIN and set a temporary PIN for the user** on the Assigned SecurID Tokens page. Then, enter a temporary PIN. You must communicate this PIN to the user.

For more information, see the Security Console Help topic "Generate a PIN for the Initial On-Demand Tokencode Authentication."

Note: Users who are enabled for Credential Manager can use Credential Manager to request PINs and configure existing PINs.

Preventing and Handling User Authentication Problems

This section describes educational measures you can take to facilitate administration of your Authentication Manager deployment.

Educating Users About Security Responsibilities

A critical part of implementing a secure system is educating users about their security responsibilities. No security product can fully protect your system if users do not take their security responsibilities seriously.

Authentication Manager can offer no protection against an intruder who has obtained both a user's PIN and SecurID token. Therefore, it is essential to make sure that users are aware of the following obligations. Users must:

- Notify an administrator immediately if a PIN is compromised.
- Notify an administrator immediately if a token is missing.
- Protect tokens from physical abuse.
- Advance event-based tokens only when they need a tokencode for authentication.
- Lock unattended workstations.
- Log off of secure applications and sites when finished, and close open web browsers.

You use the Security Console to disable tokens and clear PINs. For instructions, see the Security Console Help topics "Disable Tokens" and "Clear an RSA SecurID PIN."

4

Administering Users

- [Enabling and Disabling Users](#)
- [Assisting Users Who Have Been Locked Out of the System](#)
- [Assisting Users Whose Tokens Are Lost, Stolen, Damaged, or Expired](#)
- [Providing Users with Temporary Emergency Access](#)
- [Replacing Tokens](#)
- [Enabling and Disabling Tokens](#)
- [Resynchronizing Tokens](#)
- [Clearing PINs](#)
- [Requiring Users to Change Their PINs](#)
- [Providing Users with Fixed Passcodes](#)
- [Clearing Incorrect Passcodes](#)
- [Designating a Default Shell for UNIX Users](#)
- [Assigning Logon Aliases](#)
- [Updating Phone Numbers and E-mail Addresses for On-Demand Tokencodes](#)
- [Granting Access with User Groups](#)

Note: See Appendix C, “[Managing RSA SecurID Tokens with the Microsoft Management Console \(MMC\)](#),” if you are using the Microsoft Management Console for token-related tasks.

Enabling and Disabling Users

As an administrator, one of your tasks is enabling and disabling users for authentication. Enabled and disabled are terms used to describe the user's authentication status. An enabled user can authenticate using RSA Authentication Manager, but a disabled user cannot.

Users who are added to Authentication Manager, whether added manually or by linking to an identity source, are automatically enabled. You can assign RSA SecurID tokens to enabled users so that they can gain access to the resources protected by Authentication Manager.

You may choose to disable a user if you know that the user does not need to authenticate for an extended period of time, such as during a short-term or long-term leave.

Note: When a user is disabled, any tokens belonging to that user remain enabled. Disabling tokens is a separate function. See [“Enabling and Disabling Tokens”](#) on page 102.

For example, assume that one of your users is taking a one-time leave of absence. Although the user will be out of the office for one month, the user will need the ability to authenticate upon returning to work. Since the user's account is going to be inactive for one month, you disable the user's account during that time period. When the user returns to work, you enable the user's account so that the user can authenticate and access the resources protected by Authentication Manager.

Important: A disabled user is different than a user who has been locked out of the system. Disabling is done manually, by the administrator, and means that the user's account has been turned off. Lockout occurs when the system locks the user's account for violating the lockout policy. For more information on assisting users with locked accounts, see the following section, [“Assisting Users Who Have Been Locked Out of the System.”](#)

Before enabling and disabling users, note the following:

- Enable and disable users on the Edit User page in the RSA Security Console.
- Administrators can only enable and disable users within their scope. For example, the administrator of the Greenley security domain can enable and disable users in Greenley and all of Greenley's lower-level security domains.
- Disabling a user does not remove the user from the identity source.
- Authentication Manager verifies the identity source enable/disable setting at each authentication. Authentication Manager accounts for external identity source enable/disable settings. For example, if you use Active Directory, and the user is disabled in Active Directory, then that user cannot authenticate.

For instructions, see the Security Console Help topics “Enable Users” and “Disable Users.”

Assisting Users Who Have Been Locked Out of the System

Each user is governed by the lockout policy of the security domain to which the user is assigned. The lockout policy specifies the number of failed authentication attempts allowed before the system locks a user's account.

Note: A user's account gets locked, not the user's assigned token.

Lockout policies are designed to protect your company's resources from unauthorized individuals who attempt to authenticate by posing as authorized users and guessing passcodes until they find the correct one. It is not uncommon, however, for authorized users to be locked out of the system for exceeding the number of failed authentication attempts. This usually happens when the user incorrectly enters the PIN or tokencode.

When users violate the lockout policy, their accounts are locked and they can no longer authenticate. You can manually unlock the users so that they can authenticate.

Note: Lockout policies can be created so that user accounts are automatically unlocked after a specified period of time. These accounts can also be unlocked manually.

For example, one of your users calls the Help Desk because he has made four authentication attempts and cannot gain access to the system. Because the default lockout policy only allows three failed authentication attempts, you realize that the user's account has been locked. Since the lockout policy also specifies that the account must be unlocked by an administrator, you must unlock the account.

You can manually unlock the user on the Edit User page in the Security Console.

For instructions, see the Security Console Help topic "Enable Users."

Note: Users can also use RSA Credential Manager to unlock their accounts.

Assisting Users Whose Tokens Are Lost, Stolen, Damaged, or Expired

You may occasionally encounter users who are unable to use their tokens because the tokens are either damaged, lost, temporarily misplaced, stolen, or expired.

Important: Encourage your users to report lost or stolen tokens as soon as possible.

When a token is unavailable or expired, the user may need a new token, or require temporary access to Authentication Manager. To assist the user, you can:

- Provide temporary access.

A user might need to authenticate despite the lost or destroyed token, or while waiting for the arrival of the replacement token. Even with a missing token, two-factor authentication is still possible with the use of an Emergency Access Tokencode, a temporary tokencode generated by Authentication Manager and used for access to the protected resources. For more information, see "[Providing Users with Temporary Emergency Access](#)" on page 96.

Important: If the user has an expired token, replace the token and then provide temporary access. An Emergency Access Tokencode cannot be assigned to an expired token.

- Replace the token.
Permanently lost, stolen, damaged, or expired tokens must be replaced. For more information on replacing tokens, see [“Replacing Tokens”](#) on page 101.

Note: Users whose tokens are temporarily unavailable (the token was left at home, for example), but known to be in a safe place, do not require replacement tokens. However, these users may require temporary access. See [“Providing Users with Temporary Emergency Access”](#) on page 96.

Note: Users can also use Credential Manager to request replacement tokens or to request temporary access to Authentication Manager.

Providing Users with Temporary Emergency Access

Users may occasionally require temporary emergency access to Authentication Manager if their token is temporarily unavailable. For example, users may require temporary access if they leave their token at the office while traveling for business, or if the token has been temporarily misplaced. Users with lost, stolen, damaged, or expired tokens may also require temporary emergency access while waiting for their replacement tokens.

You can provide temporary emergency access to Authentication Manager for the following two scenarios:

- **Online authentication.**
Provide emergency access for users with misplaced, lost, stolen, or damaged tokens. Temporary emergency access is available using an Online Emergency Access Tokencode. There are two types of Online Emergency Access Tokencodes:
 - **Temporary Fixed Tokencode.** A temporary tokencode used in conjunction with the user's PIN. You can configure the expiration date.
 - **One-Time Tokencode set.** A set of tokencodes. Each tokencode can be used only once, and is used with the user's PIN.
- **Offline authentication.**
Provide emergency access for RSA SecurID for Windows users who require emergency access while authenticating offline. These are users with lost or stolen tokens, or users who have forgotten their PIN. Temporary emergency access can be provided in one of two ways:
 - **Offline Emergency Access Tokencode.** Use this option if the user has a temporarily misplaced, lost, or stolen token. The Offline Emergency Access Tokencode is used with the user's PIN.
 - **Offline Emergency Passcode.** Use this option if the user has forgotten his or her PIN. The Offline Emergency Passcode is used in place of the user's PIN and tokencode.

Important: If the user has an expired token, replace the token, and then provide temporary access. An Emergency Access Tokencode cannot be assigned to an expired token. See [“Replacing Tokens”](#) on page 101.

These scenarios are described in detail in the following sections.

Note: Users can also use Credential Manager to request temporary access to Authentication Manager.

Providing Temporary Emergency Access for Online Authentication

Even with a missing token, two-factor authentication is still possible with the use of an Online Emergency Access Tokencode. The Online Emergency Access Tokencode is an 8-character alphanumeric code generated by Authentication Manager and used for online access to the protected resources. Similar to the tokencode, the Online Emergency Access Tokencode is combined with the user's PIN to create a passcode.

There are two types of Online Emergency Access Tokencodes: Temporary Fixed Tokencodes and One-Time Tokencode sets. A Temporary Fixed Tokencode is a tokencode that can be used more than once. You can configure the expiration date and other Temporary Fixed Tokencode attributes. A One-Time Tokencode set is a set of tokencodes, each of which can be used only once. You can specify how many tokencodes are in the set. Both tokencode types work in the same way, using the user's PIN, and are described below.

Note: The format of the Online Emergency Access Tokencode (Temporary Fixed Tokencodes and One-Time Tokencode sets) is determined by the token policy of the security domain to which it belongs. For example, if the token policy is set to allow special characters, the Online Emergency Access Tokencode can include special characters.

For example, assume that a user has lost his or her token. Despite having lost the token, the user needs to authenticate immediately. You assign a set of One-Time Tokencodes for the user to use with his or her PIN until you can replace the lost token.

Important: If the user has an expired token, replace the token, and then provide temporary access. An Online Emergency Access Tokencode cannot be assigned to an expired token. See [“Replacing Tokens”](#) on page 101.

To use the Security Console to generate an Online Emergency Access Tokencode for online authentication, select the **Manage Emergency Access Tokencodes** option in the token Context menu (use the Context menu belonging to the missing/lost token). On this page, you can control the use and security of the Online Emergency Access Tokencode.

In the Online Emergency Access section of the Manage Emergency Access Tokencodes page, you can configure the following attributes:

- Select the type of Online Emergency Access Tokencode:
 - Temporary Fixed Tokencode
 - One Time-Tokencode set
- Select the number of tokencodes in the set (One-Time Tokencode sets only).
- Set the Online Emergency Access Tokencode lifetime.
For security reasons, you may want to limit the length of time the Online Emergency Access Tokencode can be used. Because the Online Emergency Access Tokencode is a fixed code, it is not as secure as the pseudorandom number generated by the token.
- Specify what happens if the missing token is recovered (if the user finds the lost token, for example). You have the following options:
 - Deny authentication with token
Use this option if you do not want the token to be used for authentication if recovered.

Important: If the token is permanently lost or stolen, use this option. This safeguards the protected resources in the event the token is found by an unauthorized individual who attempts to authenticate.

- Allow authentication with token at any time and disable online emergency tokencode
Use this option if the token is temporarily misplaced (the user left the token at home, for example). When the user recovers the token, he or she can immediately resume using the token for authentication. The Online Emergency Access Tokencode is disabled as soon as the recovered token is used.
- Allow authentication with token only after the emergency code lifetime has expired and disable online emergency tokencode
You can also use this option for temporarily misplaced tokens, however when the missing token is recovered, it cannot be used for authentication until the Online Emergency Access Tokencode expires.

Note: You cannot assign an Online Emergency Access Tokencode (Temporary Fixed Tokencode or One-Time Tokencode set) to a disabled token.

For example, a user calls because he or she left his or her SecurID token at the office. The user is currently at home and needs to authenticate immediately. Although the token is not lost, the user still requires temporary access. In this situation, you can generate an Temporary Fixed Tokencode for the user.

Because you know that the user will have the token the following day, you can set a lifetime for the Temporary Fixed Tokencode. You may also choose to specify that the Temporary Fixed Tokencode is automatically disabled when the user attempts to authenticate with his or her token.

Important: You may also encounter a situation where the token is permanently lost. For example, assume one of your users calls to tell you that his or her SecurID token has been stolen. In this situation, you can grant temporary access by generating an Online Emergency Access Tokencode for the user. When granting temporary access, it is extremely important that you choose to deny authentication with the token if it is recovered. This protects your resources in the event an unauthorized individual attempts to authenticate.

For instructions, see the Security Console Help topic “Generate An Online Emergency Access Tokencode.”

Note: If the user is certain that the token is permanently lost, destroyed, or expired, you must replace the token. See [“Replacing Tokens”](#) on page 101.

Providing Temporary Emergency Access for Offline Authentication

RSA SecurID for Windows users may need temporary emergency access so that they can authenticate while working offline. Temporary emergency access is necessary for users with misplaced, lost, or stolen tokens, or users who have forgotten their PIN.

Note: Temporary emergency access for offline authentication cannot be provided for event-based tokens. Users cannot use event-based tokens for offline authentication.

Users who are authenticating offline can gain temporary emergency access using one of the following options:

- **Offline Emergency Access Tokencode.** Use this option if the user has a misplaced, lost, or stolen token. The Offline Emergency Access Tokencode is used with the user's PIN and allows two-factor authentication.
- **Offline Emergency Passcode.** Use this option if the user has forgotten his or her PIN. The Offline Emergency Passcode is used in place of the user's PIN and tokencode.

Important: If the user has an expired token, replace the token, and provide temporary access. An Offline Emergency Access Tokencode cannot be assigned to an expired token. See [“Replacing Tokens”](#) on page 101.

Both types of temporary emergency access are described in the sections that follow.

Assigning a Temporary Tokencode for Offline Authentication

To use the Security Console to view and configure an Offline Emergency Access Tokencode for offline authentication, select the **Manage Emergency Access Tokencodes** option in the token Context menu (use the Context menu belonging to the missing/lost token).

In the Offline Emergency Access section of the Manage Emergency Access Tokencodes page, you can:

- View the Offline Emergency Access Tokencode.
- View the Offline Emergency Tokencode expiration date.
- Reset the Offline Emergency Tokencode.
- Allow the user to use the Offline Emergency Access Tokencode for emergency online access.

Note: Offline Emergency Access Tokencodes can only be issued if the user has used the token to authenticate, at least once, to an agent that can provide offline data for SecurID for Windows users.

For instructions, see the Security Console Help topic “Assign an Offline Emergency Access Tokencode.”

Assigning a Temporary Passcode for Offline Authentication

To use the Security Console to view and configure an Offline Emergency Passcode for offline authentication, select the **Manage Emergency Offline Access** option in the user Context menu.

On the Manage Offline Emergency Access page, you can:

- View the Offline Emergency Access Passcode.
- View the Offline Emergency Passcode expiration date.
- Reset the Offline Emergency Passcode.

For instructions, see the Security Console Help topic “Assign an Offline Emergency Passcode.”

Replacing Tokens

Sometimes you must assign a new token to a user. For example, you must assign a new token if a user's token has been permanently lost or destroyed, or if the current token has expired.

Note: Use the View option in the token Context menu to check the token expiration date.

In the Security Console, there are two ways to assign a replacement token:

- Select the **Replace with Next Available SecurID Token** option on the token Context menu if you want the system to automatically assign the next available token to the user.
- Select the **Replace SecurID Tokens** option on the token action menu if you want to choose the replacement token. If your company has different locations, select a replacement token from the user's office location. This makes distribution more efficient.

For example, assume that you are the administrator for the New York security domain and one of your users has permanently destroyed his token. You select the **Replace SecurID Tokens** option to assign a new token to the user. Since you have multiple office locations, you select a token that also belongs to the New York security domain.

In the above example, if all of the available tokens belong to a different security domain, San Jose, for example, you can assign the token as long as both security domains are included in your administrative scope. When you assign the token, you can leave it in the San Jose security domain, or you can transfer it to the New York security domain so that it belongs to the same security domain as the user to which it is assigned. When you assign the token, disable it to keep it secure during the transit to New York. Tell the user to notify you as soon as the token arrives so that you can enable it for authentication.

If you must mail a replacement token, disable the token after you assign it. Once you have confirmation that the user has received the replacement token, re-enable it so that it can be used for authentication. This safeguards the system in the event the token is lost in the mail.

For instructions, see the Security Console Help topic "Replace a Token."

Note: Users can also use Credential Manager to request replacement tokens.

The user may need access to Authentication Manager while waiting for the replacement token. In this case, you can give the user temporary access by generating an Emergency Access Tokencode for the existing token. The user can use the Emergency Access Tokencode to authenticate until the replacement token arrives. For more information on providing temporary access to Authentication Manager, see "[Providing Users with Temporary Emergency Access](#)" on page 96.

Enabling and Disabling Tokens

As an administrator, one of your tasks is enabling and disabling tokens so that they can be assigned to users and used for authentication. Enabled and disabled are terms that describe the token's authentication status. An enabled token can be used for authentication, but a disabled token cannot.

After Authentication Manager is installed, tokens must be imported into the system. All imported tokens are automatically disabled. This is a security feature that protects the system in the event that the tokens are lost or stolen.

Note: A disabled token does not refer to a token belonging to a user who has been locked out of the system. Disabling a token is done manually, by the administrator, and means that the token cannot be used for authentication. Lockout applies to a user's account, not a user's token.

You can manually enable and disable tokens on the Edit Token page in the Security Console. You must enable a token before it can be used for authentication.

Important: Tokens are automatically enabled when first assigned to a user.

In these situations, you should disable a token after it has been assigned to a user:

- When it is going to be mailed or delivered to a user. Re-enable the token when you know that it has been successfully delivered to the user to whom it has been assigned.
- If you know that the user to whom the token is assigned does not need to authenticate for some period of time. For example, you may want to disable a token belonging to a user who is going away on short-term leave or extended vacation. Once you disable the token, that user cannot authenticate with the token until the token is re-enabled.

Note: Disabling a token does not remove it from the system. Disabled tokens can be viewed using the Security Console.

For example, assume that one of your users is taking a one-time leave of absence. Although the user will be out of the office for one month, the user will need the ability to authenticate upon returning to work. Since the user's account is going to be inactive for one month, you disable the user's token and the user's account during that time period. When the user returns to work, you enable the user's account and the user's token so that the user can authenticate and access the resources protected by Authentication Manager.

Note: You can only enable and disable tokens in security domains that are included in your administrative scope.

For instructions, see the Security Console Help topics "Enable Tokens" and "Disable Tokens."

Resynchronizing Tokens

You can use the Security Console to resynchronize tokens that have become unsynchronized with Authentication Manager. A token needs to be resynchronized when the following occurs:

- For time-based tokens, resynchronization is necessary when the token clock and the Authentication Manager system clock do not match. When the clocks do not match, the tokencodes are not the same. If the tokencodes are different, authentication attempts fail.
- For event-based tokens, resynchronization is necessary when the token's tokencode count and the Authentication Manager tokencode count are not the same. When the tokencode counts are different, authentication attempts fail.

Important: You must resynchronize event-based tokens if you have re-imported them.

When a token becomes unsynchronized with the system, when the user attempts to authenticate, the system prompts the user to enter the tokencode. If the tokencode is correct, the system prompts the user to enter the next tokencode. This behavior can be confusing to users, as they are used to entering only one tokencode to authenticate. In this case, resynchronize the user's token so that he or she is not prompted for a second tokencode when authenticating.

To use the Security Console to resynchronize the token, select the **Resynchronize Token** option in the token Context menu to launch the Resynchronize Token page.

If you want to resynchronize multiple event-based tokens at the same time, you can enable Database Recovery Mode. When Authentication Manager is in database recovery mode, the system resynchronizes the event-based tokens at the first post-disaster logon.

For instructions, see the Security Console Help topic "Resynchronize a Token."

For more information on database recovery mode for event-based tokens, see the Security Console Help topic "Enable Database Recovery Mode."

Note: Users can also use Credential Manager to resynchronize their tokens.

Clearing PINs

You need to clear a user's PIN if the user has forgotten it. When you clear a PIN, the current PIN is deleted so that the user can create a new one.

When a PIN has been cleared, the user is prompted to create a new PIN on the next authentication attempt. Similar to what happens to users who are authenticating for the first time, the user initially enters their current tokencode only. Upon successfully entering the tokencode, the user is prompted to create and then confirm a new PIN. The new PIN is then associated with the token.

Note: Encourage users to create PINs containing both letters and numbers, as they are more secure. You can also set PIN requirements in the token policy. See "[Setting Token Usage Requirements](#)" on page 51.

For example, assume that you are a system administrator and one of your users calls. It has been months since the user has made an authentication attempt, and she has since forgotten her PIN. The user asks you to clear her PIN so that she can create a new one. After verifying the user's identity, you clear the PIN. Tell the user to enter her tokencode when prompted for her passcode on the next authentication attempt. After entering the tokencode, the user is prompted to create a new PIN.

To use the Security Console to clear a PIN, select the **Clear SecurID PIN** option on the token Context menu.

For instructions, see the Security Console Help topic "Clearing an RSA SecurID PIN."

Note: Users can also use Credential Manager to reset their PIN.

RSA SecurID SID800 Authenticators

Users with SID800 Smart Cards need a PIN Unlocking Key to access their token if they have forgotten their PIN. You can view the PIN Unlocking Key on the Token Properties page in the Security Console.

For more information, see the Security Console Help topic "Obtain the PIN-Unlocking Key for a SID800 Smart Card."

Note: You must load the SID800 Smart Card data into Authentication Manager before you can view it. To load the data, use the "[Import PIN Unlocking Key Utility](#)" on page 269.

Requiring Users to Change Their PINs

You can force users to change their PINs if there is concern that the PIN has been compromised. A compromised PIN puts the resources protected by Authentication Manager at risk.

Important: Instruct users to report compromised PINs as soon as possible, as they pose a significant security risk.

Forcing a user to change a PIN assumes that the user knows the current PIN. When you force a PIN change, on the next authentication attempt the user authenticates as he or she normally would, using the existing PIN and tokencode. After successfully authenticating, the user is immediately prompted to create a new PIN. The user creates a new PIN, confirms the new PIN, and then the PIN is associated with the token.

For example, assume that you are a system administrator and one of your users calls, concerned that her PIN has been compromised. She was using her computer at a local coffee shop and she is worried that someone may have seen her type her PIN. Because she knows the PIN, it is not necessary to clear the PIN. Instead, require her to create a new PIN on her next authentication attempt.

Depending on the token policy, the user may be required to use a system-generated PIN instead of creating one. In this case, on the next authentication attempt, the system provides the user with the new, system-generated PIN. The user then authenticates again using the new, system-generated PIN.

Note: Encourage users to create PINs containing both letters and numbers, as they are more secure. You can also enforce PIN requirements using a token policy. See [“Setting Token Usage Requirements”](#) on page 51.

In the Security Console, select the **Require SecurID PIN Change on Next Logon** option in the token Context menu to force users to change their PINs on the next authentication attempt.

For instructions, see the Security Console Help topic “Force RSA SecurID PIN Changes.”

Note: Because this feature requires knowledge of the current PIN, you cannot use it for users who have forgotten their PIN. For more information on how to help users who have forgotten their PINs, see the preceding section [“Clearing PINs.”](#)

Note: Users can also use Credential Manager to reset their PIN.

Providing Users with Fixed Passcodes

You can assign a fixed passcode to users, which allows them to authenticate without an RSA SecurID PIN and tokencode. Instead, users enter their fixed passcode to gain access to the resources protected by Authentication Manager.

Important: Fixed passcodes are essentially passwords, and are not recommended as they eliminate all of the benefits of two-factor authentication. Use fixed passcodes only in test environments and in situations when users are authenticating to authentication agents within the corporate firewall.

To use the Security Console to set a fixed passcode, use the **Fixed Passcode** field on the Authentication Settings page. The Authentication Settings page is accessed through the user Context menu.

For instructions, see the Security Console Help topic “Managing Fixed Passcodes.”

Clearing Incorrect Passcodes

The system counts each time the assigned user enters an incorrect passcode, clearing this count automatically with each correct passcode. If a user enters more incorrect passcodes than are allowed by the SecurID Token policy and then enters a correct passcode, the user is prompted for his or her next tokencode. If you do not want a user to be prompted for the next tokencode, you can use the Security Console to clear incorrect passcodes. Select **Clear Incorrect Passcodes** on the Authentication Settings page (this page is accessed through the user Context menu).

When you select this checkbox, the user is not prompted for the next tokencode on his or her next authentication attempt. Keep in mind, however, that if the user exceeds the number of failed logon attempts allowed by the lockout policy, the user is locked out of the system.

This operation only clears the existing count. To clear future counts, you must perform the procedure again.

For instructions, see the Security Console Help topic “Manage User Authentication Attributes.”

Designating a Default Shell for UNIX Users

The default shell is the shell the user logs on to when accessing a UNIX machine.

To use the Security Console to assign a default shell, use the **Default Shell** field on the Authentication Settings page. The Authentication Settings page is accessed through the user Context menu.

For instructions, see the Security Console Help topic “Manage User Authentication Attributes.”

Assigning Logon Aliases

Logon aliases allow for situations where users are able to log on with their own user ID and a user group ID. The user group ID is associated with a user group that has access to a restricted agent.

For example, users may be able to use an account name based on their first initial and last name as well as an administrative account with a specific name, such as “root.” If a logon alias has been set up, Authentication Manager verifies the authentication using the user’s passcode, regardless of the account name the user used to log on to the operating system. For backward compatibility, a shell value is also maintained by the system.

You can assign logon aliases on the Authentication Settings page in the Security Console. The Authentication Settings page is accessed through the user Context menu.

For instructions, see the Security Console Help topic “Manage User Authentication Attributes.”

Updating Phone Numbers and E-mail Addresses for On-Demand Tokencodes

You can enable users to receive tokencodes by cell phone or e-mail address. Tokencodes sent by cell phone or e-mail address are called on-demand tokencodes.

You can use the Security Console to enable a user to receive on-demand tokencodes. You can also configure the destination address for the tokencode, either a cell phone number or e-mail address. Configure this information on the Assigned SecurID Tokens page in the Security Console.

For instructions, see the Security Console Help topic “Update SMS Phone Number and E-mail Address.”

For more information on on-demand tokencodes, see [“Delivering Tokencodes Using Text Message or E-mail”](#) on page 85.

Granting Access with User Groups

There are two types of authentication agents in Authentication Manager, unrestricted and restricted. Unrestricted agents can be accessed by all registered users in the same realm as the agent. Restricted agents can be accessed only by users belonging to the user group associated with the restricted agent.

For more information on restricted agents and user groups, see [“Granting Access to Restricted Agents Using User Groups”](#) on page 74.

Note: You can also configure your user groups so that the users in the group can only access the restricted agents at certain times of day. For more information, see [“Setting Restricted Access Times for User Groups”](#) on page 75.

5

Administering RSA Authentication Manager

- [Modifying Administrator Permissions](#)
- [Displaying All of the Administrative Roles with View or None Permission for a Specific Attribute](#)
- [Securing Communications Between the Authentication Agent and RSA Authentication Manager](#)
- [Determining Limits on Administrative Sessions](#)
- [Viewing and Closing Administrative Sessions](#)
- [Configuring the System Cache for Improved Performance](#)
- [Updating Identity Source Attributes](#)
- [Adding and Updating Token Attributes](#)
- [Adding Additional Software Token Device Types to Your Deployment](#)
- [Configuring RSA Security Console Preferences](#)
- [Licenses](#)

Modifying Administrator Permissions

All RSA Authentication Manager administrators are assigned an administrative role that defines the administrator's permissions. You can use the RSA Security Console to edit both predefined roles (Privileged Help Desk, for example) and roles that you create based on your deployment model.

Note: The Super Admin role cannot be modified.

Before editing administrative roles, note the following:

- You can only modify roles, including your own role, that fall within your administrative scope.
- When you modify a role, all administrators with that role are affected by the change.

Use the Edit Administrative Role page in the Security Console to modify the following administrator permissions:

- Administrative Role Basics
- Realm Administrator
- Manage Delegated Administration
- Manage Users
- Manage Use Groups

- Manage Reports
- Manage RSA SecurID tokens
- Manage User Authentication Attributes
- Manage Authentication Agents
- Trusted Realm Management
- Manage RADIUS
- On-Demand Tokencodes
- Provision Requests

For example, assume that you are the Super Admin for FocalView Software Company. You have an administrator in your Boston office whose role limits him to assigning and managing tokens. You want to expand the administrator's responsibility so that he can also manage agents. You decide to modify the administrator's current role instead of creating a new one. When assigning permissions, select the appropriate actions in the Manage Authentication Agents section. You can choose from the following actions: Add, Edit, View, Delete, and All.

These actions give the administrator permission within the Boston security domain and any of Boston's lower-level security domains, if applicable. Note that an administrator whose administrative scope only includes the Boston security domain can only manage the objects (users, tokens, and agents, for example) belonging to that domain.

Using the previous example, instead assume that multiple administrators have the role that grants permission to manage tokens. If you modify the role so that one of the administrators can also manage agents, all of the other administrators with that role can also manage agents. In this case, you may want to create a new role for the one administrator who manages both tokens and agents.

Another option is to create a second role that allows agent management and then assign the role to the administrator. In this case, the administrator would have two assigned roles.

For instructions, see the Security Console Help topic "Edit Administrative Roles."

For more detailed information on administrative roles, permissions, and administrative scope, see "[Adding Administrators](#)" on page 32.

Displaying All of the Administrative Roles with View or None Permission for a Specific Attribute

Use the Store utility, `store`, to display all of the administrative roles with a view or none permission for a specific identity attribute. This allows you to determine if an administrative role is affected when the entry type for that attribute changes from optional or read-only to required. Changing the entry type of an attribute to required can have a negative impact on administrative roles with view or none permissions for that attribute.

When you define an identity attribute, you define whether the attribute is optional, read-only, or required. When you define an administrative role, you define the attributes an administrator can view and modify. In order for administrators to edit or add users, they must have a modify permission for each required attribute.

For example, if the Cell Phone Number attribute entry type is optional, an administrator without a modify permission for that attribute can add and edit users because the Cell Phone Number attribute is not required.

If the Cell Phone Number attribute entry type changes from optional to required and a user does not have a value for that attribute, administrators without a modify permission for that attribute cannot add or edit the user because they cannot supply all of the required attributes. If an administrator does not supply all of the required attributes, the Security Console does not add or update a user.

To avoid unintended results for administrators, follow these steps when you change an attribute from optional or read-only to required:

1. Use the Store utility to identify the administrative roles with a view or none permission for the attribute you want to change.
2. For each administrative role identified in [step 1](#), do one of the following:
 - a. If you want an administrative role to continue to manage users, use the Security Console to change that administrative role to have a modify permission for the attribute you want to change.
 - b. If you do not want an administrative role to continue to manage users, use the Security Console to remove the permission to manage users from that administrative role.
For more information, see the Security Console Help topic, “Edit Administrative Roles.”
3. Use the Security Console to change the attribute from optional or read-only to required. For more information, see the Security Console Help topic, “Edit Identity Attribute Definitions.”

Using the Store Utility to Display Administrative Roles with a View or None Permission for a Specific Attribute

To display administrative roles with a view or none permission:

1. On the primary instance, open a new command shell, and change directories to `RSA_AM_HOME/utills`.

2. Type:

```
rsautil store
--action admin_roles attribute
```

where *attribute* is the name of the attribute you want to change from optional or read-only to required. For example, Cell Phone Number.

3. When prompted, enter the master password of the encrypted properties file.

Important: Although it is possible to enter the master password on the command line along with the other options, this creates a potential security vulnerability. RSA recommends that you enter the master password only when the utility presents a prompt.

Securing Communications Between the Authentication Agent and RSA Authentication Manager

The node secret is a shared secret known only to the authentication agent and the Authentication Manager. Authentication agents use the node secret to encrypt authentication requests that they send to Authentication Manager.

Authentication Manager automatically creates and sends the node secret to the agent in response to the first successful authentication on the agent.

The agent and the Authentication Manager server must agree on the state of the node secret. For example, if the server thinks the agent has a node secret but the agent does not have one, the agent is unable to authenticate. If the agent thinks it has a node secret and the server does not think the agent has one, the agent is unable to authenticate.

In most deployments, automatically delivering the node secret is sufficient. However, you can choose to manually deliver the node secret. When you manually deliver the node secret, you must:

- Use the Security Console to create the node secret. For instructions, see the Security Console Help topic “Manage the Node Secret.”
- Deliver the node secret to the agent (on a disk, for example), and use the agent’s Node Secret Load utility to load the node secret on to the agent.

When you run the Node Secret Load utility, the utility:

- Decrypts the node secret file.
- Renames the file after the authentication service name, usually **securid**.
- Stores the renamed file in the `%SYSTEMROOT%\system32` directory on Windows machines and the **ACEDATA** directory on UNIX machines.

When you manually deliver the node secret, take the following security precautions:

- Use the longest possible, alphanumeric password.
- If possible, deliver the node secret on external electronic media to the agent administrator, and verbally deliver the password. Do not write down the password. If you deliver the node secret through e-mail, deliver the password separately.
- Make sure that all personnel involved in the node secret delivery are trusted personnel.

For additional information about creating and sending the node secret file, see the Security Console Help topic “Manage the Node Secret.”

Refreshing the Node Secret Using the Node Secret Load Utility

The Node Secret Load utility, `agent_nsload`, is stored in the `/utils/bin/ace_nsload` directory on the Authentication Manager host.

To refresh the node secret using the Node Secret Load utility:

1. Create a node secret using the Security Console.
2. On the Authentication Manager instance, copy `agent_nsload` from the `\utils\bin\ace_nsload` directory for the agent's platform to the agent host. RSA provides the following platform-specific versions of the utility:
 - Windows
 - LINUX
 - Solaris
 - HP-UX
 - IBM AIX
3. From a command line on the agent host, run the Node Secret Load utility. Type:

```
agent_nsload -f path -p password
```

where:
 - *path* is the directory location and name of the node secret file.
 - *password* is the password used to protect the node secret file.

Determining Limits on Administrative Sessions

Every time an administrator logs on to the Security Console, a new administrative user session is created. The system maintains data about each administrative session, storing information about the administrator's use of the Security Console. Session data includes the following information:

- User ID
- User's administrative role
- Authentication methods the administrator used to log on to the Security Console
- Session creation time
- Last session access time
- Maximum allowed time for inactivity
- Session lifetime

You can view individual session information on the Active User Session page in the Security Console.

Although you cannot control the information stored for each administrative user session, you can configure how Authentication Manager handles individual administrator sessions by:

- Limiting the number of concurrent administrator sessions. You can designate the maximum number of active sessions an administrative user can have open at any given time.
- Limiting the length of an administrator session. You can set a maximum session lifetime. When the session reaches its session lifetime, the Security Console ends the session, regardless of periods of inactivity.
- Limiting periods of inactivity for a session. You can set the maximum amount of time a session can be inactive before the Security Console automatically ends the session.

Note: Only the Super Admin can change session attributes.

Session attributes are set for each instance of Authentication Manager. For example, you can create one set of session attributes for the primary instance, and you can create a different set of session attributes for each replica instance.

To use the Security Console to configure session attributes, go to the RSA Server Instance Configuration page. Select the **Session Handling** option on the instance Context menu to go to the Session Handling Configuration page.

Each of the session attributes is described in detail in the following sections.

Limiting the Number of Concurrent Administrative Sessions

To help the system maintain optimum performance, you can limit the number of concurrent sessions in the Security Console. You can limit the number of concurrent sessions for the system as well as the number of allowable concurrent sessions for an individual administrator.

When the system reaches the maximum number of concurrent sessions, the next logon attempt automatically triggers the system to terminate the existing session with the longest period of inactivity.

You can also limit the number of sessions an administrator can have open at any given time. Limiting the number of individual sessions keeps the total number of system sessions in check, and it also is an important security feature. An administrator who can have an unlimited number of concurrent sessions is more likely to open a session on a computer and forget about it, leaving the session subject to unauthorized access.

You can configure how the system handles administrators who reach their session limit. You have the following options:

- Deny access to a new session.
- Allow the administrator to open a new session, but ask the administrator to confirm the closing of one of the administrator's old sessions.
- Allow the administrator to open a new session while automatically closing one of the administrator's old sessions (no confirmation).

For example, assume that you are an administrator whose instance settings allow two concurrent sessions on the Security Console. You initiate your first session. You open a second session. You attempt to initiate a third session. Since you are only configured for two sessions, the Security Console asks you if you want to continue opening the third session. If you choose to continue opening the third session, the system closes either your first session or your second session, depending on which one has been inactive longer. After closing the older session, the Security Console allows you to proceed with the new session.

Important: Closing the web browser does not end the session. A session is closed when the administrator logs off of the Security Console. A session can also be closed by another administrator.

Designate the number of concurrent system and administrator sessions on the RSA Server Instance Configuration page in the Security Console. The default for total system sessions is 10,000. Set the total number of allowable sessions high enough to account for administrators who like to have more than one active session.

Note: Only the Super Admin can designate concurrent session limits.

For instructions, see the Security Console Help topic "Configure Session Handling."

Limiting the Length of Administrative Sessions

Each Security Console session has a session lifetime. Session lifetime refers to the amount of time that an administrative user session can be active before Authentication Manager terminates it. Session lifetime is an important security feature, as it prevents administrators from creating sessions and leaving them open indefinitely, leaving them subject to unauthorized users.

Session lifetime is independent of session inactivity. For example, if a session lifetime is eight hours, an administrator is automatically logged off of the system after eight hours, even if there have been no periods of inactivity during the session.

Specify the session lifetime on the RSA Server Instance Configuration page in the Security Console. You can use the system default policy (sets the session lifetime to eight hours), or you can create a custom session lifetime policy for the instance. Custom session lifetime policies are created on the Session Lifetimes page in the Security Console.

Note: Only the Super Admin can add, edit, and delete session lifetimes.

For instructions, see the Security Console Help topic “Configure Session Handling.”

Limiting Periods of Inactivity Allowed for Administrative Sessions

Most sessions have some period of inactivity. Administrators may step away from their desks or work in another software program. Short periods of inactivity are expected, but longer periods of inactivity can be a security risk. If administrators leave their desks for an extended period of time, all open sessions become subject to unauthorized use.

To protect your system from unauthorized activity, you can specify the maximum period of inactivity allowed in a session before the administrator is automatically logged off of the system.

For example, assume that you are an administrator and the maximum period of inactivity allowed for one of your sessions is twenty minutes. Assume that you step away from your desk for thirty minutes. When you return, your session will have been terminated and you will have been logged off of the Security Console.

Maximum allowable period of inactivity (time-out) is a component of a session lifetime policy, and is specified on the Session Lifetimes page in the Security Console. You can use the system default policy (sets the maximum period of inactivity to thirty minutes), or you can create a custom period of inactivity policy. Use the RSA Server Instance Configuration page in the Security Console to apply period of inactivity policies to an instance.

Note: Only the Super Admin can change session attributes.

For instructions, see the Security Console Help topic “Configure Session Handling.”

Viewing and Closing Administrative Sessions

You can use the Security Console to view and close active administrative sessions. You can see all of the administrators who are currently logged on to Authentication Manager.

Note: Only administrators with the appropriate permissions can view and close administrator sessions.

You can view the following session information on the Active User Session page in the Security Console:

User ID. The administrator who initiated the session.

Client IP. The IP address of the Security Console in which the session is active.

Session Created. The date and time that the session was created.

Last Accessed. The date and time of the administrator's last successful logon.

The Active User Sessions page shows all of the administrative users who are currently logged on to Authentication Manager. You can also view a specific user session by searching by User ID, client IP address, or authentication status.

Note: User searches are case sensitive.

You can also use the Active User Session page to close an active administrator session. You may want to close an active session for security reasons. For example, if you know that one of your Help Desk Administrators forgot to log off of Authentication Manager before leaving the office for the day, you can use the Security Console to close the session.

When you close a session, the administrator is logged off of the system and directed to the Security Console logon page. If the administrator is in the middle of a task when the session is closed, he or she sees an error message.

For instructions, see the Security Console Help topics "View Active User Sessions" and "Close Active User Sessions."

Configuring the System Cache for Improved Performance

You can fine-tune system performance by managing the caching of system objects such as users, user groups, policies, and roles. The cache contains an object's data so that it can be retrieved more quickly and more efficiently for use in Authentication Manager tasks.

In the Security Console, the Caching Configuration page is accessed using the **Caching** option in the instance Context menu. On the Caching Configuration page, there is a list of Authentication Manager objects. Each object has the following attributes:

Cached Objects. Designates the size of the cache limit for the object. For example, if you set a limit of 5,000, the cache stores up to 5,000 records for that object.

Default values are provided. The default values are recommendations for a medium-sized company. You can change these values to any number between 3 and 10 million.

Account for system memory when designating the numbers for Cached Objects. If these numbers are set too high, and all caches reach their limits, system performance can be compromised. Set the number high enough to be effective, but low enough to avoid using all of the system memory.

Current Utilization. A read-only field that tells you how many records are currently occupying the object cache.

Use the Current Utilization value as a guide when setting the Cached Objects limit. For example, if AdminRoleCache is set to 5,000 but your current utilization is only 3, you might want to lower the AdminRoleCache setting and conserve storage space.

Refresh Interval (Secs.). Specifies the number of seconds between refreshes. A refresh checks to see if the object database record has changed since the cache was last updated. If the object has changed in the database, the cached version of the object is updated.

Important: Do not set the refresh interval to less than 60 seconds. This can have a negative impact on Authentication Manager performance.

When the Current Utilization exceeds the number designated in the **Cached Objects** field, the cached object that has been unused for the longest length of time is deleted to make space for the new object. This keeps the cache current.

Note: Only the Super Admin can manage the cache.

For instructions, see the Security Console Help topic "Configure the Cache."

Updating Identity Source Attributes

Occasionally it is necessary to update identity source properties after the identity source has been linked to Authentication Manager. You can use the RSA Operations Console to edit user attribute mapping and some configuration data for the identity source.

You can edit the following:

- **Identity Sources Basics** - Edit any Notes for the identity source.
- **Directory Settings** - Edit the **Search Results Time-out**, **User Account Enabled State**, and **Validate Map Against Schema** fields.
- **Active Directory Options** - Edit the Global Catalog options, including whether you want users to authenticate to the identity source or the Global Catalog.
- **Directory Configuration - Users** - Edit user identification information such as name, ID, and password. Edit the directory attribute mapping. You can also designate a search filter and the search scope.
- **Directory Configuration - User Groups** - Edit directory attribute mappings. You can also designate a search filter and the search scope.

Note that the following fields are read-only once the identity source has been linked to Authentication Manager:

- Identity Source Name
- Type
- Secure Connections
- Failover

Important: Only the Super Admin can edit the identity source attributes.

For instructions, see the Operations Console Help topic “Edit Identity Sources.”

Adding and Updating Token Attributes

You can add custom token attribute definitions to Authentication Manager. Token attribute definitions allow you to store additional token information with the token data. This information is in addition to the standard set of token attributes.

For example, assume that you want to store the user's phone number with their assigned token data. You can add a token attribute definition for phone number. When you add a token attribute definition, the new attribute is available in the Token Attributes section of the SecurID token page in the Security Console. There, you can enter a value for the token attribute and the data is stored with the other token data.

Note: Token attribute definitions are applied to each token in your deployment.

To use the Security Console to add a token attribute definition, select the **Token Attribute Definition** option in the Authentication menu.

For more information, see the Security Console Help topic "Add Token Attribute Definitions."

Adding Additional Software Token Device Types to Your Deployment

Periodically, new software token types are released by RSA. Before you can assign the new token types to users, it is necessary for you to add information about them to your Authentication Manager deployment.

Use the Security Console to add the new software token type to Authentication Manager. You can add new token device types on the Add New Import Software Token Device Type page.

For complete instructions, see the Security Console Help topic "Add New Software Token Type."

Configuring RSA Security Console Preferences

You can customize the Security Console by configuring the Security Console display and behavior attributes. Attributes are set on an administrator basis, so each administrator can configure his or her own Security Console sessions.

You can configure the following Security Console attributes:

- **Items Per List Page** - you can set the number of items returned on each search results page. You can choose from 25, 50, 100, 250, or All. The default value is 25.
- **Language** - you can set the language used in the Security Console. The default is English.
- **Keyboard Shortcuts** - you can set keyboard shortcuts within the Security Console. For example, you may configure the H key so that you can use Alt-H to return to the Security Console home page. Default values are provided for each of the shortcuts.

Configure your Security Console from the Personal Preferences page, located under My Console on the Security Console home page.

Note: When you update your personal preferences, the new settings are not visible until the next time you log on to the Security Console.

For instructions, see the Security Console Help topic “Set Personal Preferences.”

Licenses

Each Authentication Manager installation has one or more software licenses associated with it. The license represents permission to use the Authentication Manager software.

These are the license types:

Base Server. A permanent license allowing up to 2 instances of Authentication Manager.

Enterprise Server. A permanent license allowing up to 15 instances of Authentication Manager.

Each license type has a limit on the number of instances of Authentication Manager that can be installed and whether or not multiple realms are allowed. User limits are determined on an individual basis, based on the customer's usage requirements.

For example, a customer with 10,000 employees may purchase a license for 11,000 users in order to accommodate current employees and to allow for any hiring that may happen in the future.

The following table shows the attributes for each license type.

License Feature	Base Server	Enterprise Server
Number of users	Specified by customer at time of purchase	Specified by customer at time of purchase
Number of instances	2 ¹	15
Allows multiple realms?	No	Yes
Allows clusters?	No	Yes
RSA Credential Manager self-service	Yes	Yes
RSA Credential Manager provisioning	No	Yes
On-demand tokencode service	Optional	Optional
RADIUS	Yes	Yes
Business Continuity	Optional	Optional
Allows offline authentication?	Yes	Yes

¹Licenses with a two instance limit allow a third instance for disaster recovery situations.

The business continuity option allows you to temporarily enable more users to use RSA SecurID authentication than your license normally allows. RSA recommends that for users created with the temporary license, you enable them to receive on-demand tokencodes so that you do not have to assign and deliver tokens to them. However, if you want, you can assign them RSA SecurID tokens.

You can view the following license information on the License Status page in the Security Console:

Status. Indicates the status of the license with respect to user limits. The possible values are OK, Approaching Limit, or Limit Exceeded.

Limitation Type. Indicates how the license is restricted, either by User Count or Expiration Date. Only Evaluation licenses are restricted by expiration date.

Limit. Indicates the number of active users allowed for the license. An active user is a user who has been assigned a static passcode or at least one token.

Actual. Indicates the actual number of active users using the license. An active user is a user who has been assigned a static passcode or at least one token.

Note: The active user count is updated by an hourly batch job. If you want to view the current count without waiting for the batch job, select the **Manage Existing** option in the Licenses menu to view the License Status page. The License Status page displays the current active user count.

On the License Status page, click **View Installed Licenses** to go to the Installed Licenses page and see all of the licenses for Authentication Manager. Licenses are listed by License ID. The following information is displayed for each license:

License ID. The license's identification number.

The license ID Context menu has two options: you can select View to view more detailed information or you can select Uninstall to uninstall the license.

Category. The license category.

Issued. Indicates the date and time that the license was issued.

Installed. Indicates the date and time that the license was installed.

Note: Only the Super Admin can perform license-related tasks.

The Security Console shows warning messages as you approach the user limit indicated by your license type. A message appears when you come within 5% of the user limit.

If you find that you need to upgrade your license to allow for additional users or instances, or to add additional features, you can obtain a new license from RSA. You can add the new license on the Install New License page in the Security Console.

Note: When you add a new license that contains features with new menu items, you must log off of Authentication Manager and then log back on to see the new system menus.

For instructions, see the Security Console Help topics "Install New Licenses" and "Delete License."

6

Administering RSA Credential Manager

- [Overview of RSA Credential Manager](#)
- [Configuring RSA Credential Manager](#)
- [Configuring Provisioning](#)
- [Assisting Users](#)
- [Logging On to the RSA Self-Service Console](#)
- [Customizing Features of RSA Credential Manager](#)

Overview of RSA Credential Manager

RSA Credential Manager is a web-based workflow system that automates the token deployment process and provides user self-service options.

Provisioning streamlines the token deployment process if you are rolling out a large-scale token deployment. It also reduces administrative services and the time typically associated with deploying tokens.

Self-service allows you to reduce the time that the Help Desk spends servicing deployed tokens—when users forget their PINs, misplace their tokens, and require emergency access, or resynchronization. Users perform token maintenance tasks and troubleshoot tokens using the RSA Self-Service Console without involving administrators.

Licensing Options

The Base Server license includes self-service. The Enterprise Server license includes self-service and provisioning.

Note: If you want provisioning, and have a Base Server license, you must upgrade to the Enterprise Server license.

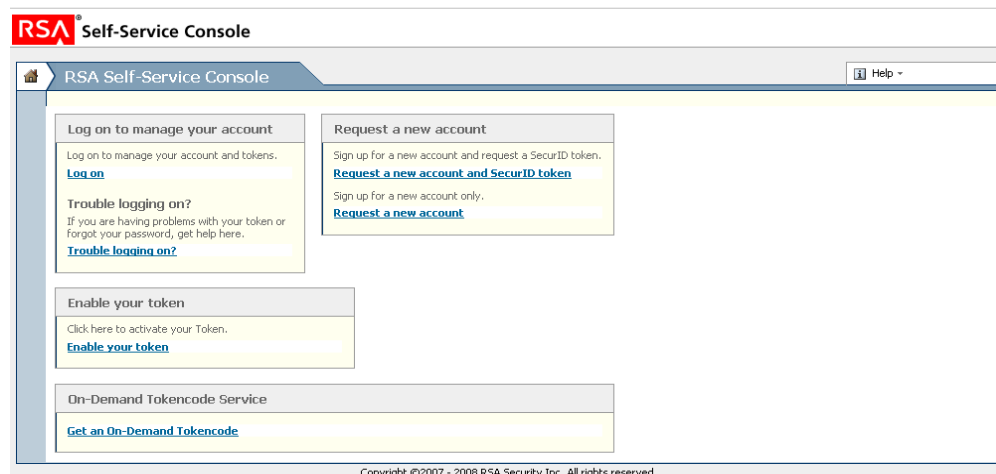
RSA Self-Service Console

The Self-Service Console is a browser-based interface where users can request tokens, troubleshoot tokens, and perform token maintenance tasks. You can customize the header text of the landing page of the Self-Service Console using the RSA Security Console. For more information, see the Security Console Help topic “Customize the RSA Self-Service Console Landing Page.”

You can customize the Self-Service Console Help (RSA Self-Service Console Frequently Asked Questions) to reflect how your company uses self-service and provisioning. For more information, see [“Customizing Help for the RSA Self-Service Console”](#) on page 252.

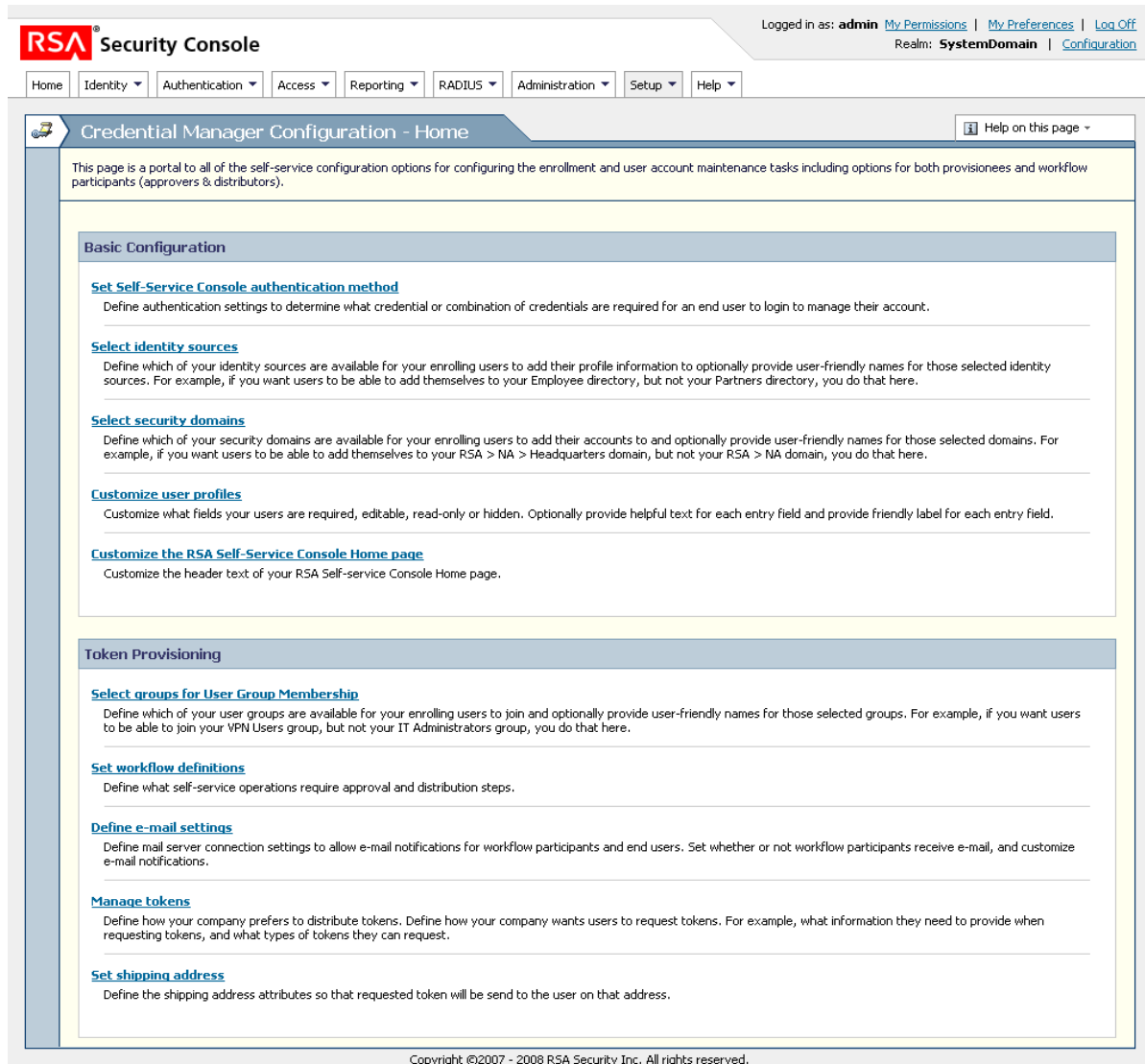
Note: The tasks that users can perform from the Self-Service Console depend on the type of access to identity sources and the license installed. For more information, see [“Using Read-Only and Read/Write Identity Sources”](#) on page 129.

The following figure shows the landing page of the Self-Service Console.



RSA Security Console

Super Admins use the Security Console to configure Credential Manager. The following figure shows the Credential Manager Configuration - Home page. For more information, see the Security Console Help topic “Configure Credential Manager.”



RSA Security Console | Logged in as: **admin** | [My Permissions](#) | [My Preferences](#) | [Log Off](#)
 Realm: **SystemDomain** | [Configuration](#)

Home | Identity | Authentication | Access | Reporting | RADIUS | Administration | Setup | Help

Credential Manager Configuration - Home

This page is a portal to all of the self-service configuration options for configuring the enrollment and user account maintenance tasks including options for both provisionees and workflow participants (approvers & distributors).

Basic Configuration

- [Set Self-Service Console authentication method](#)
Define authentication settings to determine what credential or combination of credentials are required for an end user to login to manage their account.
- [Select identity sources](#)
Define which of your identity sources are available for your enrolling users to add their profile information to optionally provide user-friendly names for those selected identity sources. For example, if you want users to be able to add themselves to your Employee directory, but not your Partners directory, you do that here.
- [Select security domains](#)
Define which of your security domains are available for your enrolling users to add their accounts to and optionally provide user-friendly names for those selected domains. For example, if you want users to be able to add themselves to your RSA > NA > Headquarters domain, but not your RSA > NA domain, you do that here.
- [Customize user profiles](#)
Customize what fields your users are required, editable, read-only or hidden. Optionally provide helpful text for each entry field and provide friendly label for each entry field.
- [Customize the RSA Self-Service Console Home page](#)
Customize the header text of your RSA Self-service Console Home page.

Token Provisioning

- [Select groups for User Group Membership](#)
Define which of your user groups are available for your enrolling users to join and optionally provide user-friendly names for those selected groups. For example, if you want users to be able to join your VPN Users group, but not your IT Administrators group, you do that here.
- [Set workflow definitions](#)
Define what self-service operations require approval and distribution steps.
- [Define e-mail settings](#)
Define mail server connection settings to allow e-mail notifications for workflow participants and end users. Set whether or not workflow participants receive e-mail, and customize e-mail notifications.
- [Manage tokens](#)
Define how your company prefers to distribute tokens. Define how your company wants users to request tokens. For example, what information they need to provide when requesting tokens, and what types of tokens they can request.
- [Set shipping address](#)
Define the shipping address attributes so that requested token will be send to the user on that address.

Copyright ©2007 - 2008 RSA Security Inc. All rights reserved.

Self-Service

With self-service, users can use the Self-Service Console to:

- Enroll. When users enroll, they become users without administrative privileges.
- Test tokens, resynchronize tokens, change token PINs, and report problems with tokens themselves, instead of calling the Help Desk.
- Update user profiles.
- Change passwords for the Self-Service Console if the identity source is read/write. They may want to do this if they forget their passwords or want to change their passwords.
- Troubleshoot tokens and request emergency access for lost, broken, or temporarily unavailable tokens.
- Request replacement tokens if tokens are lost, broken, temporarily unavailable, or about to expire.

Provisioning

With provisioning, users use the Self-Service Console to:

- Request enrollment. Users need approval to enroll in provisioning. When users get approval and enroll, they become users without administrative privileges.
- Request new or additional tokens.
- Enable a token.
- Request the on-demand tokencode service.
- Request on-demand tokencodes.
- Request user group membership for access to protected resources.

Configuring RSA Credential Manager

Use the Security Console to configure Credential Manager. You must configure the following:

- Authentication method for the Self-Service Console.
- Identity sources for user enrollment.
- Security domains for enrollment.
- User profiles for enrollment.
- Self-service troubleshooting policies.

Configuring the Authentication Method for the RSA Self-Service Console

The following table lists the possible authentication methods for accessing the Self-Service Console.

Note: If users are not required to have a password, make sure to configure an authentication method other than password for the Self-Service Console.

Authentication Method	Description
RSA password	The RSA password is the default method for protecting the Self-Service Console.
LDAP password	If you use a directory server as your identity source, you may also want to enable an LDAP password as an authentication method. This allows users whose user records are saved in the identity source to access the Self-Service Console.
SecurID token	For additional security, you can configure the Self-Service Console to require users to present an RSA SecurID passcode.

For instructions, see the Security Console Help topic “Configure Authentication Methods for the RSA Self-Service Console.”

Selecting Identity Sources for Enrollment

Use the Security Console to select the identity sources that you want to make available for user enrollment. For instructions, see the Security Console Help topic “Select Identity Sources for RSA Credential Manager.”

Note: If you have a multirealm configuration, the display name that you set for the identity source or security name is global and not per realm. The display name appears on the Self-Service Console for all realms. If you change this display name when you log on to one realm, this display name is changed for all realms.

Using Read-Only and Read/Write Identity Sources

All Credential Manager user and user group data is stored in Authentication Manager identity sources.

If you configure Authentication Manager to use an external directory, such as Sun Java System Directory Server or Microsoft Active Directory, for user and user group data, instead of the Authentication Manager internal database, you need to consider how read/write or read-only access affects the tasks that users can do.

The following table shows the tasks that users can perform with the Base Server and Enterprise Server licenses, and whether these tasks are available if the identity source is set to read/write or read-only access.

Note: If a directory server is read-only, user information must exist in the directory server or in the Authentication Manager internal database for users to perform tasks using the Self-Service Console.

User Task	Identity Source (Base Server License)		Identity Source (Enterprise Server License)	
	Read/Write	Read-Only	Read/Write	Read-Only
Enrollment Tasks				
Request an account	✓	✓	✓	✓
Request an account, a token, or the on-demand tokencode service			✓	✓
Select identity source	✓	✓	✓	✓
Select security domain	✓	✓	✓	✓
Create user profile	✓		✓	
Create password	✓		✓	
Answer security questions	✓	✓	✓	✓
Select user group membership			✓	
Troubleshoot problems using the self-service troubleshooting authentication method	✓	✓	✓	✓
Log On to the Self-Service Console				
Log on to the Self-Service Console	✓	✓	✓	✓
Token Management Tasks				
Request a token or the on-demand tokencode service			✓	✓

User Task	Identity Source (Base Server License)		Identity Source (Enterprise Server License)	
	Read/Write	Read-Only	Read/Write	Read-Only
Enable a token			✓	✓
Change token PIN	✓	✓	✓	✓
Test a token	✓	✓	✓	✓
Report a problem with a token	✓	✓	✓	✓
Request a replacement token			✓	✓
Management Tasks				
Update profile	✓		✓	
Change password	✓		✓	
Request additional user group membership			✓	

Selecting Security Domains for Enrollment

The security domains that you make available for user enrollment determine which users administrators can manage, and limit the scope of the administrators' control by limiting the security domains to which they have access.

Use the Security Console to select the security domains that you want to make available for user enrollment. For instructions, see the Security Console Help topic "Select Security Domains for RSA Credential Manager."

Note: If you have a multirealm configuration, the display name that you set for a security domain or identity source is global and not per realm. The display name appears on the Self-Service Console for all realms. If you change this display name when you log on to one realm, this display name is changed for all realms.

Customizing User Profiles for Enrollment

When users enroll in Credential Manager, they must enter information about themselves in a user profile. Credential Manager uses the information in user profiles to allow users to log on to the Self-Service Console, to send e-mail notifications to users about requests, and for the delivery of on-demand token codes.

You can customize a user profile for each identity source that you make available for enrollment. Users who request enrollment in a particular identity source use the profile you created for that identity source.

If an identity source has read/write access, you can make specific fields read-only. If an identity source is read-only, all user profile fields are read-only.

Also, if you create custom attributes for an identity source, you can add custom attributes to the user profile. For instructions, see the Security Console Help topic “Customize User Profiles.”

Configuring Self-Service Troubleshooting for the RSA Self-Service Console

If users forget the password for the Self-Service Console, they can use another method to authenticate and perform troubleshooting tasks by clicking **Trouble logging on?** on the Self-Service Console Home page. For instructions about configuring self-service troubleshooting, see the Security Console Help topic “Add Self-Service Troubleshooting Policies.”

Users can perform the following troubleshooting tasks after authenticating:

- Reset their password.
- Reset their PIN.
- Resynchronize their token.
- Request a replacement token if they lost the old one or if their token has expired.
- Request emergency access.

Self-Service Troubleshooting Authentication Method

Use the Security Console to configure one of the following authentication methods to allow users to authenticate to perform troubleshooting:

- Security Questions—Users must answer security questions when they enroll. With this authentication method, users can troubleshoot tokens and passwords.

Note: Security questions are only for troubleshooting. They cannot be used as a primary authentication method to the Self-Service Console.

If security questions are enabled, users may be prompted to create answers to security questions during enrollment or when they log on to the Self-Service Console. If users are unable to log on to the Self-Service Console, they may be prompted to answer these questions again to verify their identity. Their answers must match the answers that they originally provided. When users correctly answer the security questions, the Self-Service Console troubleshooting screens help them resolve their problems.

When you configure security questions, you determine how many questions the users must answer during enrollment and during troubleshooting. For instructions, see the Security Console Help topic “Configure Security Questions.”

If a user forgets the answers to security questions, you can clear the answers for a particular user. For example, you might need to do this if the user forgot the answers, or the answers were compromised in some way. After you clear the answers, the user must provide new answers at the next logon. For instructions, see the Security Console Help topic “Clear Security Question Answers.”

- Passwords—Users must enter the password that is associated with their identity source (either directory server or the internal database) to troubleshoot problems. With passwords as the authentication method, users can only troubleshoot tokens, not passwords. You can decide whether to require users to use passwords.

Note: Passwords are less secure than two-factor authentication.

If users forget their passwords, they must call the Help Desk for assistance.

- None—Users cannot perform any troubleshooting tasks. This may result in additional calls to the Help Desk for assistance.

Number of Incorrect Self-Service Troubleshooting Authentication Attempts

You can configure an unlimited number of incorrect self-service troubleshooting authentication attempts, or you can allow a specified number of failed attempts within a specified number of days, hours, or minutes.

Note: The number of attempts applies only to self-service troubleshooting authentication attempts. All other authentication attempts are governed by the lockout policies associated with the security domain that manages the authenticating user.

For more information, see the Security Console Help topic “Self-Service Troubleshooting Policies.”

You can require that administrators must unlock accounts after users have exceeded the limit of incorrect attempts, or you can allow the system to automatically unlock accounts after a specified number of days, hours, or minutes.

For instructions, see the Security Console Help topic “Add Self-Service Troubleshooting Policies.”

You can also edit, duplicate, and delete self-service troubleshooting policies. For more information, see the Security Console Help topics “Edit Self-Service Troubleshooting Policies,” “Duplicate Self-Service Troubleshooting Policies,” and “Delete Self-Service Troubleshooting Policies.”

Configuring Provisioning

To enable provisioning, you must perform the following configuration tasks:

Note: Provisioning comes with the Enterprise Server license and is optional with the Base Server license. If you want to add provisioning to a Base Server license, you need to purchase the Credential Manager Provisioning license.

- Configure workflows for requests.
- Add administrators (or other trusted users) to the “Request Approver” and “Token Distributor” roles, or add approver and distributor permissions to other existing administrative roles.
- Select user groups to make available to users.
- Configure e-mail.
- Select SecurID tokens to make available to users.
- Optional. Make the on-demand tokencode service available to users.

Configuring Workflows for Requests

Provisioning uses workflows to automate token deployment. A workflow defines the number of steps or work items for user requests. People who perform tasks in a workflow are called workflow participants.

The following table lists the types of workflow participants and the tasks they can perform.

Workflow Participant	Tasks
Approver	From the Security Console, the approver: <ul style="list-style-type: none"> • Views all requests. • Approves, rejects, cancels, or defers action on requests. • Comments on requests, if necessary. • Changes the token type, if necessary.
Distributor	From the Security Console, the distributor: <ul style="list-style-type: none"> • Views user requests that require distribution. • Determines how to assign and deliver tokens to users. • Records how tokens are delivered to users.

Each type of user request has a default workflow definition associated with it. The default workflow definitions consist of a combination of the following steps or work items for each type of request:

- One or two approval steps
- One distribution step for token requests

The following is an example of a workflow for a hardware token request, in which the company requires new employees to get the approval of their manager and an administrator for all requests:

1. A new employee requests a hardware token.
2. The manager of the new employee approves the request.
3. The Help Desk administrator approves the request.
4. The new employee picks up the hardware token at the Help Desk administrator's office.
5. The user activates the token and uses it to successfully log on to protected resources.

Using the Security Console, you can configure the workflow definitions for each type of request. For instructions, see the Security Console Help topic “Configure Workflow Definitions.”

Adding Administrators

There are two types of predefined administrative roles:

Request Approvers. Can view and approve requests using the Security Console.

Token Distributors. Can view requests using the Security Console and distribute hardware tokens.

You can use the predefined roles for Request Approvers and Token Distributors. For more information, see “[Predefined Administrative Roles](#)” on page 33.

Selecting User Groups

After you set up user groups and grant access to restricted agents using Authentication Manager, you need to make the user groups available to Credential Manager users so that users can access restricted agents or protected resources on a network.

Note: You must create at least one user group before you can make user groups available to users.

The user groups that you select are displayed on the Self-Service Console. Users can select user group membership when they enroll in Credential Manager or they can request additional user group membership after they enroll.

You can add and remove user groups, create default user groups, create a display name for each user group, and add notes to describe the user group or add information to help users select the correct user group. For instructions, see the Security Console Help topic “Select User Groups for Provisioning.”

Configuring E-mail

Credential Manager sends e-mail automatically when users submit requests for enrollment, tokens, the on-demand tokencode service, or user group membership. The recipients of the e-mail notifications can be users, approvers, distributors, Super Admins, and workflow participants in the parent security domain. When you nest security domains to create an administrative hierarchy, the top security domain is called the parent security domain.

When E-mail Notifications Get Sent

Credential Manager automatically sends e-mail to workflow participants when:

- Users submit requests that require approval or that require distributors to take action.
- Requests fail or cannot be processed.
- Approvers approve, reject, or cancel requests.

You need to perform the following steps to configure e-mail:

- Configure the e-mail server. For more information, see the Security Console Help topic “Configuring the SMTP Mail Service.”
- Optionally enable or disable e-mail notifications. For more information, see the following section “[Enabling or Disabling E-mail Notifications.](#)”
- Customize e-mail notifications for each type of request. For more information, see Appendix B, “[Customizing RSA Credential Manager](#)” on page 243.

Enabling or Disabling E-mail Notifications

The default setting for e-mail notifications is to send e-mail notifications to workflow participants (approvers and distributors). If workflow participants do not want to receive e-mail notifications about requests, you can disable e-mail notifications sent to them. Workflow participants who decide not to get e-mail notifications can view all requests on the **Pending Request** tab of the Provisioning Requests page in the Security Console.

Note: Credential Manager sends e-mail notifications automatically to users about their requests for enrollment, tokens, the on-demand tokencode service, and user group membership. You cannot disable e-mail notifications to users.

You can also enable e-mail notifications to Super Admins and workflow participants in the parent security domain. If you enable e-mail to workflow participants in the parent security domain, all approvers and distributors in security domains above the security domain where a request originates, receive e-mail notifications.

Selecting RSA SecurID Tokens

To access network resources protected by Authentication Manager, users can request SecurID tokens using the Self-Service Console. You must configure the types of tokens that you want to make available to users. You can also set a default token type for all token requests.

The following table lists the types of tokens available.

Tokens	Description
Hardware tokens	Handheld devices, such as a key fob, that display pseudorandom codes, that change at regular intervals.
Software tokens	Software-based tokens that reside on a user's computer, PDA, or cell phone. Once installed, the software token generates tokencodes, that are displayed on the device screen.

For instructions, see the Security Console Help topic “Select Tokens for Provisioning.”

Note: You can also allow users to request the on-demand tokencode service to receive tokencodes by text message or e-mail. For more information, see [“Selecting the On-Demand Tokencode Service”](#) on page 142.

Selecting a File Format For Software Tokens

You can configure the file format for software tokens to be one of the following:

- ZIP file format. Credential Manager packs up the token record into a single .sdtid file, and adds the .sdtid file to a .zip archive to e-mail to the user.
- SDTID file format. The software token record is written to an .sdtid file and Credential Manager e-mails it to the user.

For instructions, see the Security Console Help topic “Select Tokens for Provisioning.”

Protecting Software Tokens with Passwords

Credential Manager automatically e-mails software token files and a download link to the token application to users after token requests are approved.

Note: Users cannot download software token files from a server.

You can configure whether you want Credential Manager to protect token files with passwords. The default setting is to password protect token files. For tokens with PINs, passwords are optional. For tokens without PINs, that have only one-factor authentication, users must provide passwords. For instructions, see the Security Console Help topic “Select Tokens for Provisioning.”

Users create passwords for token files when they request software tokens using the Self-Service Console. Credential Manager assigns the password to the token file and then sends the token file to the user in e-mail. When users install the software token on their device, they are prompted for the password. If users forget their password, they must call the Help Desk for assistance. For instructions, see the Security Console Help topic “Select Tokens for Provisioning.”

Emergency Access Tokencodes

Users whose tokens are temporarily or permanently unavailable may require emergency access to the resources protected by Authentication Manager. Credential Manager allows users to troubleshoot their token problems and obtain emergency access themselves rather than calling the Help Desk. You can configure the type of emergency access tokencodes to make available to users.

The following table lists the types of emergency access tokencodes.

Type of Tokencode	Description	Characteristics
Temporary fixed tokencode	Users who are online in the network can access their protected resources (for example, when they have lost their tokens).	<ul style="list-style-type: none"> • Must be combined with the user's RSA SecurID PIN. • Is displayed on the Self-Service Console.
One-time tokencode	Users who are online in the network can access their protected resources. One-time tokencodes are issued in sets. You can determine the number of tokencodes in a set.	<ul style="list-style-type: none"> • Must be combined with the user's RSA SecurID PIN. • Is displayed on the Self-Service Console. • Users can download the set of one-time tokencodes in a file. • Each tokencode in the set is valid one time.
On-demand tokencode service	Users who are online in the network can access their protected computers with a tokencode that allows one access. Users request on-demand tokencodes. You can set the number of days that users can get on-demand tokencodes.	<ul style="list-style-type: none"> • Must be combined with a specific PIN. The PIN is not the same as the user's RSA SecurID PIN. • User must request each tokencode. • Tokencodes are delivered either as text messages by cell phone or by e-mail. • Each tokencode expires after one use, or within the amount of time you specify when configuring Authentication Manager for on-demand tokencodes.

You can also configure the following attributes for emergency access:

- Allow users to put a token into emergency access mode using the Self-Service Console.
- Set the lifetime of emergency access for permanently lost or broken token in days, hours, minutes, or seconds. The default is 7 days.
- Set the lifetime of emergency access for a temporarily unavailable token in days, hours, minutes, or seconds. The default is 7 days.
- Specify the behavior of the emergency access tokencode if the temporarily unavailable token becomes available. You can set one of the following:
 - Deny authentication with the recovered token
 - Allow authentication with the recovered token and disable emergency access
 - Allow authentication with the recovered token after the emergency access lifetime expires, and then disable emergency access

For instructions, see the Security Console Help topic “Select Tokens for Provisioning.”

Replacement Tokens for Expired Tokens

Users can get a replacement token through the Self-Service Console if a token is about to expire. You can configure at what point users can request replacement tokens for expiring tokens. The default is 30 days.

For instructions, see the Security Console Help topic “Select Tokens for Provisioning.”

Changing Token Types or PIN Policy

Approvers and distributors can change the type of token requested by users, if necessary. For example, they can change a request for a keyfob token to a request for a USB token.

Note: Approver and distributors cannot change hardware token requests to software token requests, or the reverse. Also, approvers and distributors cannot change requests for the on-demand tokencode service to hardware or software token requests, or the reverse. RSA recommends that approvers and distributors do not change a request for a keyfob to a PINPad-style token because the PIN for a keyfob is not compatible with the PINPad-style token and the PIN fails when users try to use it.

To make tokens secure, tokens must either have a password or a PIN. Approvers and distributors cannot lower the level of security when they change token types. For CT-KIP-capable software tokens, you must require PINs.

Note: If you change the PIN policy, you must verify that it does not affect any of the pending token requests. RSA recommends that you take appropriate action on pending requests before you change the PIN policy for any tokens.

Approving Requests

Credential Manager automatically sends e-mail to approvers to notify them about user requests that they need to review.

Note: Approvers can only approve requests in their security domain.

Approvers use the Security Console to perform the following tasks:

- View the status of a particular request.
- View requests that require their approval in their security domain.
- Move requests to any security domain. If a request is not in the correct security domain, approvers can move requests into the correct security domain even if they do not have scope for that security domain.
- Approve, reject, defer, or cancel requests.

For instructions, see the Security Console Help topic “Approve Requests.”

Scope for Approvers

Scope for approvers is the same as scope for an Authentication Manager administrator. For example, when approving tokens, approvers can approve requests for tokens only if there are unassigned tokens available in their scope.

The exceptions are:

- Approvers can only approve requests for group membership if both the user and the group is in their scope.
 For example, suppose an approver's only responsibility is to assign users in the identity source “Developers” to groups in the security domain “Boston.” Users can request membership in any group that is in the same identity source to which they belong. If a user requests membership in a group in the security domain “Newton,” which is in their identity source, but the approver does not have scope over the “Newton” security domain, then the approver cannot approve that request.
- If you set default groups for Credential Manager enrollment, any approver with scope for a user, can approve the user request for enrollment in the default group, regardless of the approver's scope over the group.
 For example, if a user requests membership in a default group in the “Newton” security domain, and the approver does not have scope for “Newton,” the approver can approve that request because the request is for a default group.
- If you set up default groups from multiple identity sources and there is more than one identity source, users can only belong to default groups from the identity source in which they are registered.

Creating Multiple Requests and Archiving Requests

If you need to create multiple requests, you can use the User Groups and Token Bulk Requests utility to create batch multiple requests. You must have Super Admin privileges to run the User Groups and Token Bulk Requests utility. For example, if you want every salesman in your organization to receive a SecurID token, you can, on their behalf, create a bulk request. The system acts as if each salesman made a separate request using the Self-Service Console.

If you want to archive requests to free up disk space, you can use the Archive Requests utility to archive requests with their associated attributes and process data. If you need to retrieve archived requests, you can import the archived requests. You must have Super Admin privileges to run the Archive Request utility.

For more information about using these utilities, see Appendix D, [“Command Line Utilities.”](#)

Distributing Tokens

Credential Manager automatically sends e-mail to distributors to notify them about token requests that need distribution.

Note: Distributors can only review and close requests in their security domain.

Distributors use the Security Console to do the following tasks:

- View approved requests
- Decide how to distribute SecurID tokens
- Physically deliver tokens to the requestor
- Record how they deliver tokens and close requests

For instructions, see the Security Console Help topic “Close Requests.”

Configuring Shipping Addresses

If you plan to ship tokens to users or to use a third-party company to distribute hardware tokens, you need the shipping addresses of users.

To configure a way to collect shipping addresses from users for token distribution, you can do one of the following:

- If you store data in the internal database, you can prompt users to enter a shipping address each time that they create a token request that requires a distribution step. The shipping address appears on the user profile at enrollment.

- If you use a directory server as the identity source, you can map the shipping address to existing identity attributes. The identity source automatically fills in the address information on the enrollment page for the user. The user can validate the shipping address and, if necessary, modify it. Any changes made to the shipping address are only for the current token request. Changes to the shipping address are not reflected in the identity source.

Note: If you create an extended attribute for more than one identity source and want to use that attribute in a shipping address, you must map the extended attribute to the shipping address for each identity source. If you do not map the shipping address to the extended attribute in each identity source, the shipping address only displays the attribute for one identity source.

Creating Distribution Reports

You can use Authentication Manager to create and run customized reports describing system events and objects. For example, distributors or approvers can create a report that shows all requests, the status of requests, user information, and shipping addresses.

An administrator, distributor, or approver needs the following permissions to create distribution reports:

- Requests—view
- Users—view
- Reports—add, delete, edit, view report definition, run, schedule

For more information about creating reports, see [“Generating Reports”](#) on page 198.

Selecting the On-Demand Tokencode Service

The on-demand tokencode service allows users to request on-demand tokencodes delivered by text message or e-mail, instead of tokens. You can deliver tokencodes to a cell phone using Short Message Service (SMS) or to an e-mail address using Simple Mail Transfer Protocol (SMTP). Tokencodes delivered by SMS or SMTP are called on-demand tokencodes.

Important: RSA SecurID hardware tokens offer the highest level of security. Other methods of tokencode delivery, such as software tokens and on-demand tokencodes, may be more convenient for some users, but do not provide the same level of security as a hardware token. RSA recommends using hardware tokens.

You configure the on-demand tokencode service for requests using the Security Console. For more information about on-demand tokencodes, see [“Delivering Tokencodes Using Text Message or E-mail”](#) on page 85. For instructions about how to allow users to request this service, see the Security Console Help topic “Select Tokens for Provisioning.”

Assisting Users

Administrators get calls from users when users cannot resolve problems themselves using the Self-Service Console.

You can expect to get Help Desk calls from users if:

- You do not set up a self-service troubleshooting policy.
- Users forget their token file password.
- You set up security questions for the self-service troubleshooting policy, and users forget the answers to the security questions. For instructions, see the Security Console Help topic “Clear Security Questions.” After you clear the answers, the user must provide new answers to use security questions for self-service troubleshooting.
- Users forget their Self-Service Console password.
- Tokens are disabled.
- Tokens are not enabled for emergency access.
- The identity source is read-only and users need to make account changes.
- Users cannot read the tokencode on the token.

Note: When you use the Activity Monitor to view Credential Manager activity when helping users, Self-Service Console activity appears on the Activity Monitor under the Administrator User ID column as SYSTEM.

Logging On to the RSA Self-Service Console

The Self-Service Console is on the same machine where Authentication Manager is installed.

To log on to the Self-Service Console, do one of the following:

- Go to the following URL and enter the user name and password selected during installation:
`https://<fully qualified domain name>:7004/console-selfservice.`
- Click **Start > Programs > RSA Self-Service Console.**

Important: To protect the Self-Service Console with RSA SecurID, see [“Configuring the Authentication Method for the RSA Self-Service Console”](#) on page 129.

Note: Do not use your Internet browser’s Back button to return to previously visited Self-Service Console pages. Instead, use the Console’s navigation menus and buttons to navigate.

Customizing Features of RSA Credential Manager

You can customize the following features of Credential Manager:

- E-mail— automatically sends e-mail to users and workflow participants. You can customize the content of e-mail notifications. You can also use conditional statements in your e-mail templates to include customized information if certain conditions are met.
- Help—You can customize the Self-Service Console Help, the *RSA Self-Service Console Frequently Asked Questions*, to reflect how your company uses self-service and provisioning. For example, if your company does not use provisioning, you can delete the provisioning information.
- Token Graphics—Users view token graphics when they request new or additional tokens from the Self-Service Console. You can replace the default token graphics that ship with Credential Manager with your company's custom tokens.
- Workflow and non-workflow operations—Workflow operations require one or two approval steps and possibly a distribution step, such as token requests. Non-workflow operations do not require any approval or distribution steps, such as changing PINs. You can write custom extensions to Credential Manager for workflow and non-workflow operations by writing an API.
- Self-Service Console Home page—You can customize the header text at the top of the Self-Service Console Home page using the Security Console.

For more information, see Appendix B, [“Customizing RSA Credential Manager.”](#)

7

Administering Trusted Realms

- [Overview of Trusted Realm Deployments](#)
- [Creating Trusted Realm Relationships](#)
- [Adding and Enabling Authentication Agents for Trusted Realm Authentication](#)
- [Managing Trusted Users and Trusted User Groups](#)

Note: This chapter describes trusted realm relationships between two RSA Authentication Manager 7.1 realms. You can also create trusted realm relationships between an RSA Authentication Manager 7.1 realm and an RSA Authentication Manager 5.2 or 6.x realm. You must do this from the version 5.2 or 6.1 realm, using the Database Administration application. For more information, see the version 5.2 or 6.1 Help topic “Setting Up Cross-Realm Authentication.”

Overview of Trusted Realm Deployments

It is possible that you have more than one deployment of Authentication Manager. For example, you may have two deployments if your company acquired another company with its own Authentication Manager deployment.

Authentication Manager deployments function independently of one another. This means that two deployments have separate and different components such as identity sources, realms, and security domains. This also means that users in one deployment can only authenticate to resources within their deployment. In some business cases, you might want to allow users to authenticate across deployments and access resources in another realm protected by Authentication Manager. To do this, you can create a trusted realm relationship between the realms in your deployments.

Trusted realm relationships are created between two realms in separate deployments. When you create a trusted realm relationship, users in one realm can authenticate through a second, trusted realm, and access the resources protected by that realm's deployment of Authentication Manager. When the user attempts to authenticate through the trusted realm, the trusted realm forwards the authentication request to the user's realm. The user's realm processes the authentication request and the user can access the trusted realm's protected resources.

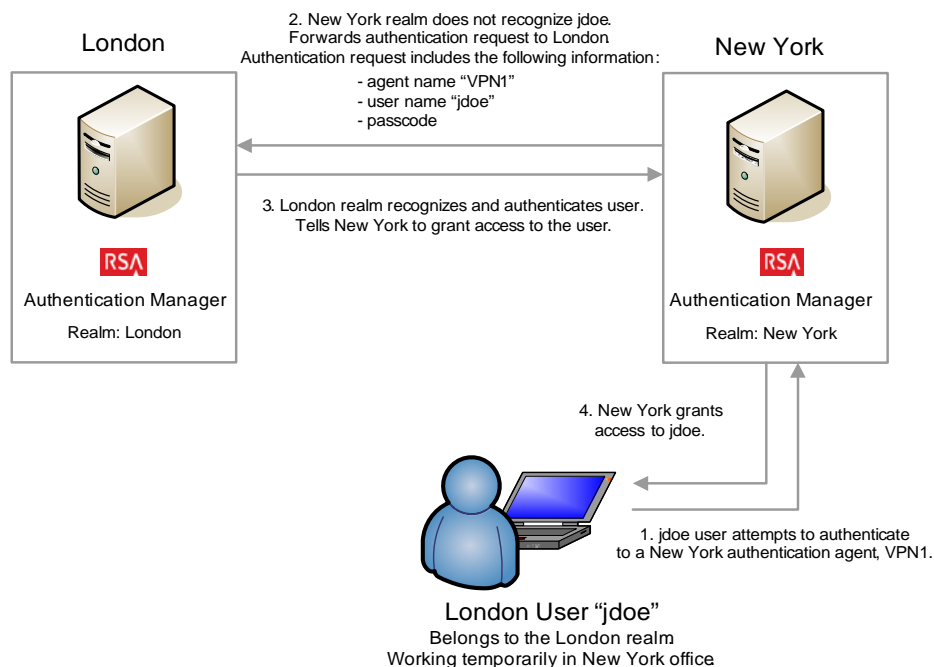
For example, assume that you have an Authentication Manager deployment in London, and a second deployment in New York. You can create a trusted realm relationship between these two realms. When that relationship is established, users in the London realm can travel to the New York office and access the New York authentication agents. New York is the trusted realm. The New York realm forwards the authentication request, with the appropriate user and agent information, to the London realm, which then authenticates the user. The user can then access New York's protected resources.

Using the above example, assume that a user “jdoe” travels to New York for business. The user needs to access the company network, so he attempts to access the VPN. In New York, the VPN server is protected by an authentication agent. When jdoe attempts to access the agent using his credentials (user name and passcode), the agent does not immediately recognize the user. Because the agent has been enabled for trusted realm authentication, it looks for the user in other realms. Finding jdoe in the London realm, the New York realm forwards the user credentials and the agent information to the London realm. The London realm verifies the user credentials, authenticates the user, and tells New York to grant access to the user. The authentication is successful.

Users who can authenticate through realms other than their own are called trusted users. Only trusted users can authenticate through a trusted realm, so trusted users must exist in the trusted realm. In the example above, the London user jdoe is a trusted user in the New York realm. The New York realm maintains a trusted user record for jdoe.

You can group trusted users into trusted user groups. Similar to user groups, you can use trusted user groups to restrict access to an authentication agent. Only members of the trusted user group can access the agent. In addition, trusted users and trusted user groups can only access authentication agents that have been enabled and configured for trusted realm authentication.

The following figure shows this process.



Note: The figure shows a one-way trust because the London user can travel to New York and authenticate, but a New York user cannot travel to London and authenticate. There are also two-way trusts that allow users from both realms to authenticate through the trusted realm. For more information on one-way and two-way trusts, see [“Creating Trusted Realm Relationships”](#) on page 148.

To create a trusted realm relationship:

To create a trusted realm relationship and configure Authentication Manager for trusted realm authentication, you need to do the following:

1. Create and configure the trust. You and the realm administrator with which you want to establish the trust must create the trusted realm relationship in each of your deployments. This involves creating a trust.

For more information, see [“Creating Trusted Realm Relationships”](#) on page 148.

2. Enable authentication agents. When a user tries to authenticate through another realm, the user is actually trying to gain access to one of the trusted realm’s authentication agents. In order for this to happen, the trusted realm administrator must enable their agents for trusted realm authentications.

For more information, see [“Adding and Enabling Authentication Agents for Trusted Realm Authentication”](#) on page 152.

3. Create the trusted users and trusted user groups. Only trusted users can access the agents enabled for trusted realm authentication. The trusted realm can specify the trusted users, or the agent can be configured to automatically create trusted users as authentications occur. Trusted users can also be grouped in trusted user groups. Trusted user groups are associated with the trusted realm’s agents. The trusted realm administrator creates the trusted users and trusted user groups.

For more information on creating trusted users and trusted user groups, see [“Managing Trusted Users and Trusted User Groups”](#) on page 155.

Of the steps listed above, some are performed by you, the realm administrator, and some are performed by the administrator of the trusted realm. The following table lists all of the steps necessary to set up a trusted realm relationship between two realms, and indicates which administrator completes the task.

Note: Only administrators with the appropriate permissions can create and administer trusted realm relationships.

The table assumes a one-way trusted realm relationship, with New York as the trusted realm (London users can travel to New York and authenticate). If you want a setup that allows users from both realms to travel and authenticate remotely, which is called a two-way trust, both administrators must perform these tasks. For more information on one-way and two-way trusts, see [“Creating Trusted Realm Relationships”](#) on page 148.

	Realm Administrator (London)	Trusted Realm Administrator (New York)
Required Tasks:		
Creating Trusted Realm Relationships	X	X
Enabling an Authentication Agent in the Trusted Realm		X ¹

	Realm Administrator (London)	Trusted Realm Administrator (New York)
Managing Trusted Users and Trusted User Groups		X ¹
Optional Tasks:		
Adding a Duplicate Authentication Agent (for access control and aliasing)	X ¹	

¹If the trusted realm relationship is two way, the realm administrator needs to complete this task as well.

The following sections provide more detailed information on creating and configuring trusted realm relationships.

Creating Trusted Realm Relationships

The first step in creating a trusted realm relationship is to create the trusts. In Authentication Manager, there are two types of trusts, one way and two way:

One-way Trust. A relationship in which only one realm is trusted. In this type of trust relationship, authentication requests can be sent from the trusted realm to the home realm, but the trusted realm cannot receive authentication requests from the home realm.

Using the example described on page 145, if London is the home realm and New York is the trusted realm, London users can authenticate through the New York realm. New York, the trusted realm in this case, can forward the authentication request back to London. New York users, however, cannot authenticate through London since the trust is one way.

Two-way Trust. A relationship in which both realms are trusted. In this type of trust relationship, authentication requests can be sent and received by both realms.

Using the example described on page 145, trusted users from London can authenticate through New York. In addition, the relationship also works in reverse: trusted users from New York can authenticate through London.

Once you have decided which type of trust you need to have, you can create the trust. To create the trust, you need to be in contact with the administrator of the trusted realm. You also need to do the following before you start:

- Discuss the configuration with the trusted realm administrator. You should discuss basic information such as which agents to enable for trusted realm authentication, which users to enable, trusted user groups, and security and logistics.
- Generate a realm certificate. A realm certificate is an SSL certificate that allows you to establish secure communication with the other realm.
- Create a trust package. A trust package is an XML file containing configuration information about your realm. You will exchange trust packages with the trusted realm. Discuss with the trusted realm administrator which method to use to send the file.
- You will need a confirmation code. A confirmation code is a code that is derived from the trust package.

The steps involved in creating a new trust are described in detail in the sections that follow.

Creating and Configuring a Trust

The first step in creating a trusted realm relationship is creating the trust in the Authentication Manager. Creating a trust involves exchanging important configuration information with the administrator of the trusted realm. The two realms must have specific information about one another in order to forward authentication requests and grant access to users from other realms.

Important: Both you and the trusted realm administrator must each create a trust, even if the trust is one way.

Creating a trust involves the following steps:

1. Generate a realm certificate to allow secure communication between the two realms. Both you and the trusted realm administrator need to perform this step.
2. Generate and upload the trust package. Both you and the trusted realm administrator need to perform this step.
3. Confirm the configuration code. Exchange confirmation codes with the trusted realm administrator. This can be done over the telephone.
4. Configure the trusted realm. You must name the trusted realm, enable or disable the trusted realm, and add a trusted user name identifier. This is also where you can configure the trust to be one way or two way.

Note: Trusted realm relationships are disabled by default, and are two way by default.

You generate the realm certificate using the RSA Operations Console. The remaining steps are done using on the Add New Trusted Realm page in the RSA Security Console. All four steps are described in detail in the following sections.

Generating the Realm Certificate

The first step is generating the realm certificate. The realm certificate allows secure communication between the two realms. You generate the realm certificate using the Operations Console.

To generate the realm certificate, go to the Realm Certificates page in the Operations Console. Select the realm for which you want to generate the realm certificate and click the **Generate Realm Certificate** button.

For instructions, see the Operations Console Help topic “Generate an SSL Realm Certificate.”

Generating and Uploading the Trust Package

The second step is creating a trust package. A trust package is an XML file containing configuration information about the trusted realm. You generate the trust package using the Security Console.

To generate the trust package, go to the Add New Trusted Realm page in the Security Console. Generate a trust package by clicking the **Generate & Download** button.

Important: After generating a trust package, you and the trusted realm administrator must exchange trust packages.

After exchanging trust packages with the trusted realm administrator, you must upload the trust package. Navigate to the location of the XML file for the other realm, and upload the trust package.

For more information on creating and uploading a trusted package, see the Security Console Help topic “Adding a Trusted Realm.”

Confirming and Exchanging the Confirmation Code

The third step is confirming and exchanging confirmation codes. A confirmation code is a code that is derived from the trust package. When you are creating the trust, the confirmation code displays on the Add New Trusted Realm page in the Security Console. There are two codes: one for your realm and one for the trusted realm.

You and the trusted realm administrator need to exchange codes. Discuss with the trusted realm administrator how and when you can verify the confirmation codes. You and the trusted realm administrator must verify each other's codes at the same time. A telephone call is a good way to confirm this code. Make sure that you and the trusted realm administrator agree on the specific time and place that the call will take place.

Configuring the Trusted Realm

The fourth step is configuring the trust. You can configure the following:

Trusted Realm Name. Name the trusted realm.

Authentication Status. Trusts are two way by default. To create a one-way trust, clear the **Authenticate Trusted Users** checkbox on the Add New Trust screen. You should clear this box for the trusted realm.

Using the figure on page 145 as an example, the London administrator would clear the Authenticate Trusted Users checkbox to indicate that London will not authenticate trusted users from New York. Since the figure shows a one-way trust with London users authenticating through New York, the New York administrator selects the Authenticate Trusted Users checkbox to indicate that London users can authenticate through the New York realm.

Trusted Realm Status. Enable or disable the trusted realm. The trusted realm must be enabled to perform trusted realm authentications. Trusted realms are disabled by default.

Note: If the realm is disabled, the associated trusted users and trusted user groups remain.

Create Trusted Users in Security Domain. You can configure the realm to automatically create trusted users as users attempt to authenticate (this option is selected when configuring the agent). Here, you can specify the security domain in which those trusted users are created.

Trusted User Name Identifier. When creating a trust in Authentication Manager, you can associate a trusted user name identifier with the trust. A trusted user name identifier prevents user name collisions between the two trusted realms.

For example, assume that you have an Authentication Manager deployment in London, and a second deployment in New York. Both realms have a user, "jdoe." Suppose the London user "jdoe" were to travel to New York and attempt to authenticate. The New York realm would look up the user name and find one of its own users. However, upon attempting to authenticate the tokencode and PIN, the authentication would fail because the New York realm sees the credentials for New York "jdoe." The New York realm does not realize that it is a trusted user who is trying to authenticate, and the user is denied access.

To avoid this situation, you can assign a trusted user name identifier to the trusted realm. A trusted user name identifier is a name suffix that the user appends to his or her normal user name. For example, if you created a trusted realm with "abc.com" as the trusted user name identifier, then all users from that realm can authenticate with "username@abc.com."

To continue with the example above, assume that you did create the trusted realm with "abc.com" as the trusted user name identifier. When the London "jdoe" travels to New York, he or she will log on as "jdoe@abc.com." The New York realm will see that this is not a New York user and forward the authentication request back to London. London recognizes the user, approves the authentication, and London "jdoe" has a successful authentication in New York.

Important: If your deployment already uses name suffixes as part of the standard user name, do not configure a trusted user name identifier, as it is not necessary.

For more information on creating and configuring a trusted realm, see the Security Console Help topic “Add Trusted Realms.”

After creating the trust, you can specify which authentication agents are available for trusted realm authentication. For more information, see [“Adding and Enabling Authentication Agents for Trusted Realm Authentication”](#) on page 152.

Editing a Trust

You can also go back and edit a trust after it's been created. You can do the following:

- Update the trust configuration. For more information, see the Security Console Help topic “Edit Trusted Realms.”
- Enable or disable the trusted realm. For more information, see the Security Console Help topic “Enable or Disable a Trusted Realm.”
- Delete a trusted realm.

Important: Deleting a trust also deletes the associated trust users. Trusted user groups associated with the trust are not deleted.

For more information, see the Security Console Help topic “Delete Trusted Realms.”

Adding and Enabling Authentication Agents for Trusted Realm Authentication

The trusted realm administrator must specify which authentication agents are available for trusted realm authentication. This is something that you and the trusted realm administrator should discuss when planning the trust.

The trusted realm administrator can enable the appropriate agent records in the trusted realm. When the administrator enables the appropriate agent, trusted users can access that agent when they authenticate through the trusted realm.

For more information, see the following section, [“Enabling an Authentication Agent in the Trusted Realm.”](#)

There are certain situations when you might want to add an agent record in your realm that corresponds to an agent record in the trusted realm. This is an optional step. For more information, see [“Adding a Duplicate Authentication Agent”](#) on page 154.

Enabling an Authentication Agent in the Trusted Realm

Note: Enable the authentication agents in the trusted realm. If your trust is one way, the trusted realm administrator should enable the appropriate agents for trusted realm authentication. If the trust is two way, both you and the trusted realm administrator should enable the appropriate agents in your realms.

The trusted realm administrator can use the Security Console to enable the appropriate agents for trusted realm authentication. On the Add or Edit Authentication Agent page, you can edit the following in the Trusted Realm Settings section:

Trusted Realm Authentication. Select this checkbox to enable the authentication agent for trusted realm authentication.

Trusted User Authentication. Decide which trusted users can access the agent. You have the following options:

- Open to all trusted users. Any user can access this agent. When the user authenticates through the trusted realm, a trusted user is automatically added in the trusted realm.

Important: RSA recommends that you configure the agent to wait at least 25 seconds for a response from the server. This allows enough time for the initial search for the trusted user. Subsequent authentication requests will not take as long. The response time is specified when you create the agent configuration file, `sdconf.rec`. For more information on creating a new agent configuration file, see [“Creating and Installing the RSA Authentication Manager Configuration File”](#) on page 71.

- Only allow trusted users in trusted groups to access the agent. Here, only trusted users belonging to the trusted user group associated with the agent can access the agent.

For more information on trusted users and trusted user groups, see [“Managing Trusted Users and Trusted User Groups”](#) on page 155.

Enable all of the agents that will allow trusted realm authentication, keeping in mind that not all of your agents will have the same settings. For example, you might leave some agents open to everyone, but you might want to leave some agents restricted to trusted user group members.

For more information, see the Security Console Help topic “Add Authentication Agents.”

Adding a Duplicate Authentication Agent

Important: This step is optional, and only recommended for specific cases.

There are certain situations when you might want to add an agent record in your realm that corresponds to an agent record in the trusted realm. The agent record is a duplicate of the agent record in the trusted realm, with the same name, IP address, and other information. There are two reasons why you might want to do this:

- **Access Control.** By creating a duplicate agent, you can restrict the agent and create associated user groups. This allows you to determine which of your users can access the agent in the trusted realm, instead of going by the trusted group membership specified in the trusted realm.
- **Resolving aliases.** If you use user group aliases, your users may attempt to log on to the trusted realm using their aliases. To ensure successful authentications, you need to create a duplicate agent with associated user groups and aliases.

Each of these scenarios is described in the sections that follow.

Access Control

You can decide which of your users can access the authentication agents in the trusted realm. To do this, create a duplicate authentication record in your realm.

For example, assume the trusted realm, New York, has an unrestricted VPN agent called “VPN1.” As the London realm administrator, you want to restrict which of your users can access VPN1, so you create an agent record in your realm, also called “VPN1.” Your agent record contains the same information as the agent record in the trusted realm, including IP address. You make VPN1 a restricted agent, and create an associated user group “VPN users,” to which you assign a user “jsmith.”

Assume jsmith travels to New York for business, and attempts to access the trusted realm's agent, VPN1. The New York realm does not recognize the user jsmith, and forwards the following information to the user's realm: user name, agent name, and passcode. The user's realm recognizes the user name and agent name, and immediately checks to see if the agent is restricted. The agent is restricted, and jsmith is a member of the associated user group, so the authentication request is approved and New York allows jsmith to authenticate.

If you create a duplicate agent record for access control, it is important to know if the corresponding agent in the trusted realm has associated trusted user groups. If the two user groups do not contain the same users, users might not be able to authenticate. Note the following:

- In the above example, assume that VPN1 in the trusted realm has a trusted user group associated with it. If jsmith is not a member of the trusted user group, then he cannot authenticate, even though he is a member of his realm's corresponding user group.
- In the above example, assume that a second London user, “sbrown,” who is a member of the trusted user group, but is not a member of his realm's corresponding user group. Despite being a member of the trusted user group, sbrown cannot authenticate because his realm has not given him access to the agent.

Resolving Aliases

Another situation that might require adding a duplicate agent record is if you use aliases for your user groups.

For example, assume that you have an agent for VPN. The VPN agent, “VPN1,” is a restricted agent. Only members of the VPN1 user group can access VPN1. In that user group, all users have aliases. For example, the user “jdoe” has the alias “jdoeVPN” for the VPN1 user group, so the user can access the VPN1 agent by logging on as “jdoeVPN.”

The user travels to a trusted realm and attempts to authenticate to the trusted realm’s VPN agent, “VPN2,” using his alias, “jdoeVPN.” The trusted realm does not recognize the user and sends the user name and agent information back to the user’s realm. The user’s realm does not recognize the agent or the user name “jdoeVPN.” Therefore, access is denied and the user cannot authenticate.

Now assume that the user’s realm does have a duplicate agent record, also called “VPN2.” VPN2 also has an associated user group, and jdoe is a member, with “jdoeVPN” as an alias. Now, “jdoeVPN” tries to access VPN2 in the trusted realm, and the trusted realm forwards the user name and agent information back to jdoe’s realm. The realm recognizes the agent name, then finds the alias listed with one of the agent’s associated user groups. The user can authenticate successfully.

Managing Trusted Users and Trusted User Groups

Note: Create trusted users and trusted user groups in the trusted realm. If your trust is one way, the trusted realm administrator should create the trusted users and trusted user groups. If the trust is two way, both you and the trusted realm administrator must create trusted users and trusted user groups.

Only trusted users can access authentication agents and authenticate through a trusted realm. You can enable your authentication agents to automatically create trusted users, or can create them manually. To restrict access using trusted user groups, create your trusted users and trusted user groups manually.

The sections that follow provide more detailed information.

Creating Trusted Users

Important: Create trusted users in the trusted realm.

You can configure the authentication agents to automatically add trusted users, or you can add them manually.

In the Security Console, add new trusted users on the Add New Trusted User page. You can specify the following on the **Trusted User Properties** tab:

Trusted User ID. The trusted user ID.

Trusted Realm Name. Select the trusted realm to which the user belongs.

Security Domain. Select the security domain in which you want the trusted user created.

Default Shell. The default shell.

On the **Trusted User Group Memberships** tab, select trusted user groups for the new trusted user.

For more information, see the Security Console Help topic “Add Trusted Users.”

Creating Trusted User Groups

Important: Create trusted user groups in the trusted realm.

Similar to user groups, trusted user groups control access to an agent. When you create a trusted user group and add associated agents, only members of the trusted user group can access the authentication agent when authenticating through the trusted realm.

In the Security Console, add new trusted user groups on the Trusted User Group page. You can specify the following on the **Trusted User Group Properties** tab:

Trusted User Group Name. The trusted user group name.

Security Domain. Select the security domain in which you want the trusted user group created.

On the **Trusted User Group Members** tab, select which trusted users you want to add to the trusted user group.

On the **Accessible Agents** tab, select which authentication agents you want the trusted user group to have permission to access.

Note: Trusted user groups are associated with the authentication agent, so if a trust is deleted, the trusted user groups are not deleted.

For more information, see the Security Console Help topic “Add Trusted User Groups.”

You can also add restricted access times to trusted user groups. For more information on restricted access, see [“Setting Restricted Access Times for User Groups”](#) on page 75.

Allowing Trusted Users to Authenticate Using RSA RADIUS

To allow trusted users to authenticate using RSA RADIUS, you may define these trusted users in Authentication Manager in the trusted realm before they attempt authenticating in that realm. You may also set up a special RADIUS profile for trusted users.

For more details about configuring RADIUS to handle trusted users, see [“How RSA RADIUS Helps Enforce Access Control”](#) on page 161.

For more information on associating trusted users with RADIUS attributes, see [“RADIUS User Attributes Apply to Individual Users and Can Use Existing Identity Source Information”](#) on page 163.

8

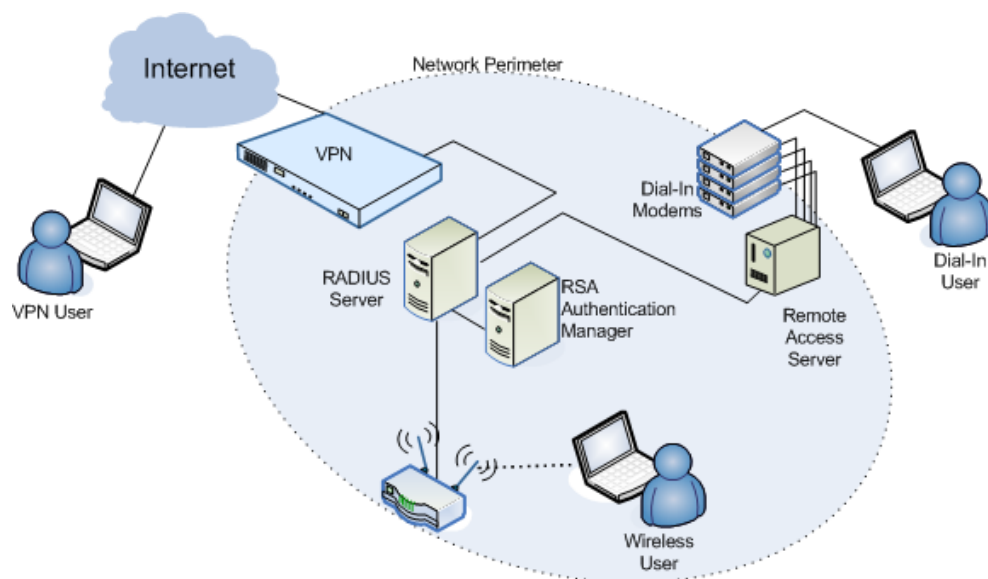
Managing RSA RADIUS

- [Overview of RSA RADIUS](#)
- [Managing User Access](#)
- [Managing RSA RADIUS Servers](#)
- [Monitoring System Usage](#)
- [Maintaining RSA RADIUS Servers](#)

Overview of RSA RADIUS

The following figure shows how an RSA RADIUS server is positioned between users and network access devices (RADIUS clients) at the network perimeter, and RSA Authentication Manager.

RADIUS users who want to access network resources send an access request to the RADIUS server. The RADIUS server establishes a secure connection with the user and requests the user's ID and authentication data, such as an RSA SecurID passcode. The RADIUS server forwards the authentication data to the RSA Authentication Manager for validation. Depending on the result, the RADIUS server returns an Access-Accept or Access-Reject message to the RADIUS client that grants or denies access. The RADIUS server may provide additional attributes to the RADIUS client that can limit a user's session length, allowed IP addresses, or other parameters.



The configuration is actually a bit more complicated as firewalls typically separate the VPN server from the RADIUS server and the Internet. Also, a single RADIUS server is shown but RADIUS replica servers are likely configured along with a primary server for load balancing and failover. Additional RADIUS client devices may also be configured, such as a number of wireless access points in strategic locations throughout a site. These details are somewhat hidden from administrators because most routine administration is applied to a primary server that, in turn, replicates some of those changes to replica servers.

Note that some less routine administration operations require administrators to know about replica servers for failover, disaster recovery, and other system maintenance purposes. These topics are also discussed in this chapter.

RSA RADIUS Supports Secure Network Access

As shown in the previous figure, remote users with direct Internet connections can access network resources using a RADIUS-enabled VPN server. Remote users that do not have direct Internet connections can connect using telephone lines and dial-in modems connected to a RADIUS-enabled network access server. Wireless users can access the network over RADIUS-enabled wireless access points.

For all of these access methods, RADIUS provides the following capabilities:

- Fine-grained access controls that allow administrators to tightly manage individual user access, restricting users to a specific network access device, session length, IP address or range, or other restriction.
- Comprehensive and flexible accounting that allows administrators to tailor event log data to meet specific needs, whether in support of Sarbanes-Oxley or any other auditing requirement. You can save your log data to a flat file.

How You Manage RSA RADIUS

RSA RADIUS administration is integrated within the RSA Security Console. Routine RADIUS administrative procedures made using the Security Console modify configuration information that resides on the RADIUS primary server. The RADIUS replica servers can access the latest changes to configuration data such as profile assignments, user attribute definitions, and realm level settings. Some changes, such as changes to RADIUS clients, RADIUS profiles, and the replication setting for RADIUS servers, require you to manually propagate the changes to the replicas by forcing the replication process. For instructions, see the Security Console Help topic "Force Replication to a Single RADIUS Replica Server."

Operations performed only once or infrequently are carried out using the web-based RSA Operations Console that runs on each RADIUS server. You can invoke an Operations Console for any server. Changes made on an Operations Console of one server do not automatically propagate to other servers. Changes that must be synchronized across all RADIUS servers, such as changes to dictionary files that help integrate RADIUS within your environment, must be manually copied to other RADIUS servers. All of the Operations Consoles are accessed from a central location to ease these manual operations.

How RSA RADIUS Helps Enforce Access Control

User access is enforced by RADIUS clients at the network perimeter. RADIUS clients (VPN servers, wireless access points, or Network Access Servers connected to dial-in modems) interact with RSA RADIUS for user authentication and to establish appropriate access control parameters. When authentication is successful, RADIUS servers return a set of attributes to RADIUS clients for use as session control parameters.

Profiles Establish Session Requirements

A profile is a named collection of attributes that specify session requirements for users authentication using RADIUS. Attributes are contained in a checklist or return list. A checklist is the set of attributes that must be sent from a RADIUS client to a RADIUS server as part of an authentication request. If a required checklist attribute is not present, the RADIUS server returns an Access-Reject message to the RADIUS client. RADIUS clients have their own administration interfaces for entering checklist attributes.

An example of a checklist attribute is “NAS-IP-Address” that specifies the IP address of a RADIUS client that the user is allowed to use. If the NAS-IP-Address attribute is not included in the access request, the request is rejected.

A return list is the set of attributes that a RADIUS server returns to a RADIUS client in an Access-Accept message when a user is authenticated. Return list attributes provide additional parameters, such as VLAN assignment or IP address assignment, that the RADIUS client needs to connect the user.

Profiles are synchronized across all RSA RADIUS servers in the realm. The profile names reside on the Authentication Manager so they can be centrally managed from the Security Console. RSA RADIUS, when shipped, contains no profiles. You create profiles using the Security Console.

For more information on creating RADIUS profiles, see the Security Console Help topic, “Add RADIUS Profiles.”

Important: A default RADIUS profile is not specified at installation. If you want a default profile, you can specify one on the Realm Configuration page in the Security Console. For more information, see the Security Console Help topic “Configure Your Realm.”

Profile Attributes Apply to All Associated Users, Trusted Users, User Aliases, and Agents

Users are defined within Authentication Manager and their association with the correct RSA SecurID token record is maintained there. On the RADIUS pages in the Security Console, an administrator can associate a user ID with a profile. This action applies the attributes of that profile to that user.

Trusted users are also defined within Authentication Manager and are associated with an agent that is enabled for trusted realm authentication. On the RADIUS pages in the Security Console, an administrator can associate a trusted user with a profile. This action applies the attributes of that profile to that trusted user.

User aliases are also defined within Authentication Manager. This capability allows a user to have multiple roles if necessary. For example, user Alice has a user identity Alice_User and a user alias identity Alice_Admin. When Alice logs on as Alice_User, she gets all of the attributes associated with users because her user identity is associated with a profile set up for users. When Alice_Admin logs on, she gets attributes more appropriate for administrators because her user alias identity is associated with a profile set up for administrators. For example, logged on as Alice_Admin, she can access IP addresses needed to manage routers and VPN servers.

Within Authentication Manager, RADIUS clients such as VPN servers, wireless access points, and network access servers supporting dial-in modems have associated authentication agent records. Administrators can associate these agents with profiles that are set up for agents. This causes the attributes in an agent profile to be assigned to all users authenticating using that specific RADIUS client device.

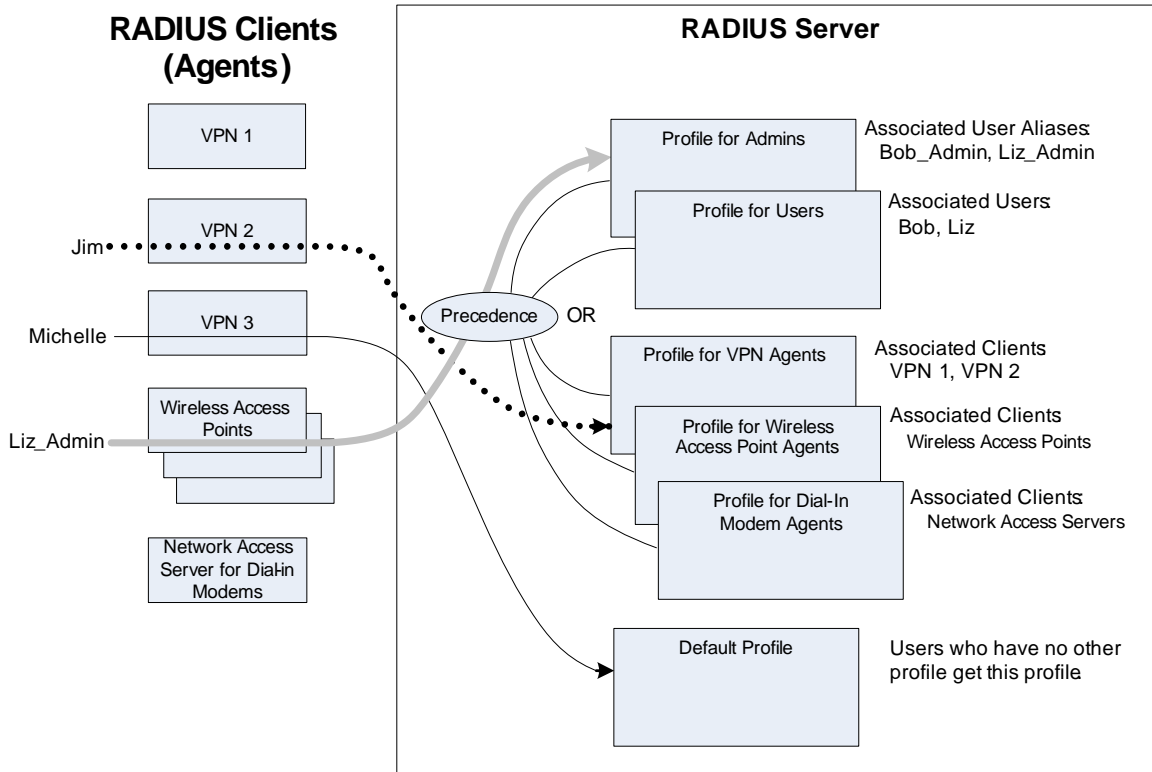
An example of this usage is a remote user authenticating through a VPN server could have access to one set of resources while that same user authenticating over a wireless access point could access a reduced set of services. RSA RADIUS allows administrators to choose whether an agent profile takes precedence over a user profile to avoid conflicts in the case where a profile for users and a profile for an agent could both be applied to a user. You can select this setting on the Realm Configuration page in the Security Console. For instructions, see the Security Console Help topic "Configure Your Realm."

In a case where a user does not have an explicitly associated profile and a profile for agents is not applied to a user (for example, because an agent profile is not set up for the RADIUS client being used), RSA RADIUS applies the default profile to that user, if you have specified one. To support this behavior, you must create at least one profile and you must specify the default profile using the Security Console realm configuration settings. For more information on setting a default RADIUS profile, see the Security Console Help topic "Configure Your Realm."

The following figure illustrates how RADIUS applies profiles to users based on associations of profiles to users and agents (RADIUS clients).

- Jim is not explicitly associated with a profile, but authenticates through VPN2 that does have an explicitly associated profile, so he gets the profile for VPN agents.
- Liz_Admin is explicitly associated with the profile for Administrators and she is authenticating through a wireless access point that also has an explicitly associated profile. As both profiles could be applied to Liz Admin, the precedence mechanism determines which profile is applied.

- Michelle does not have an associated profile and she is authenticating through VPN3 that also does not have an explicitly associated profile for agents. RADIUS applies the default profile because no other profile applies to Michelle.



RADIUS User Attributes Apply to Individual Users and Can Use Existing Identity Source Information

A RADIUS profile is a powerful mechanism because it allows you to assign many attributes to a user with a single operation. Many users may be assigned to a RADIUS profile, and changing an attribute in the profile changes it for all users assigned to that profile. Some attributes, like callback number that are user-specific and apply only to individuals, are inappropriate for use within a profile.

RADIUS user attributes can be defined outside of RADIUS profiles (they reside within RSA Authentication Manager). RADIUS user attributes can be associated with individual users and trusted users (they cannot be associated with user aliases or agents). The function of a RADIUS user attribute depends on how it is defined.

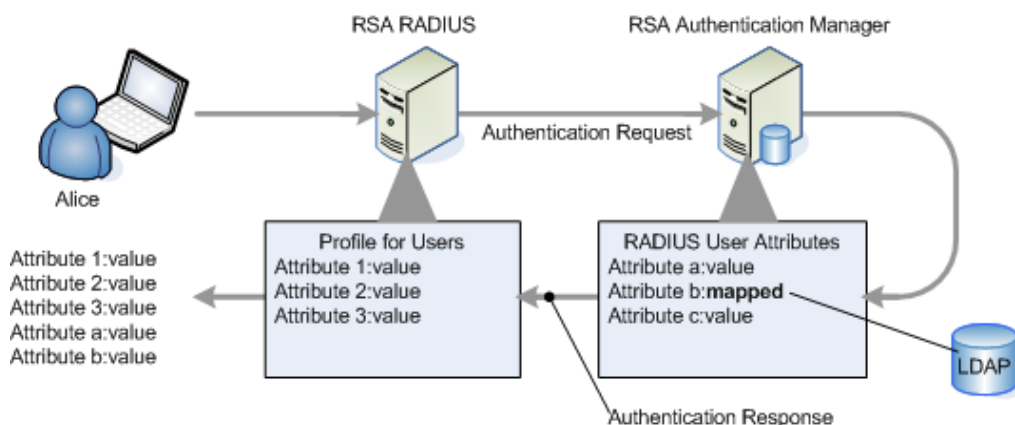
A RADIUS user attribute can be defined without an actual value. When this attribute is assigned to a user, the administrator can enter the value appropriate for that user. For example, say the RADIUS user attribute is Callback Number. On applying to a user, the administrator enters the user's callback phone number. (RADIUS can use a callback number to ensure that a user is calling from a specific phone number). RADIUS user attributes override attributes of the same name included in a profile.

Alternatively, an attribute may be defined as a mapped attribute. In this case, the attribute returns information that is already stored in an identity source, such as an LDAP corporate identity database, avoiding the need to maintain attribute values in multiple places.

In summary, profile attributes and RADIUS user attributes combine to give administrators tight control over user access at the group level and as individual users.

Note: RADIUS user attributes take precedence over RADIUS profile attributes.

The following figure illustrates the relationship of profile return list attributes and RADIUS user attributes when user Alice authenticates using RADIUS. Alice is assigned RADIUS user attributes a and b (attribute c is assigned to someone else). When Alice authenticates successfully, she gets all of the profile attributes and RADIUS user attributes a and b. The value for RADIUS user attribute b is taken from an LDAP identity store.



RADIUS Profiles and Attributes Apply to Trusted Users in Trusted Realms

Within a trusted realm, users from another realm may attempt to access resources using RSA RADIUS. How RADIUS profiles are associated with these users depends on how these remote “trusted users” are defined in the local realm.

A “trusted user” identity may be defined in Authentication Manager in advance, and a RADIUS profile associated with that trusted user identity. That profile may be set up especially for trusted users giving them attributes appropriate for trusted users. When that trusted user’s access request succeeds, RADIUS returns the associated profile attributes to the RADIUS client device along with any assigned RADIUS user attributes.

If the trusted user identity is not set up in advance, the authentication agent on the RADIUS server creates a trusted user account dynamically in Authentication Manager. In turn, Authentication Manager forwards the authentication request to other trusted realms until a success or failure is returned. On success, Authentication Manager adds the trusted user's home realm to the trusted user identity (to speed up future authentication requests) and returns a passcode accepted message to RADIUS.

If an agent profile is associated with the RADIUS client (the VPN server, wireless access point, or network access servers supporting dial-in modems) used by the trusted user, the trusted user gets the attributes of that profile. Otherwise, the user gets the default profile, if one is specified.

Other Attribute Types Provide Flexibility

RSA RADIUS provides more flexibility in controlling system behavior during authentication through the use of single-value and multiple-value attributes and orderable multiple-value attributes.

Single-Value and Multiple-Value Attributes

Attributes can be single value or multiple value. Single-value attributes appear only once in the checklist or return list; multiple-value attributes may appear several times. If an attribute appears more than once in the checklist, this means that any one of the values is valid. For example, you can set up a checklist to include multiple telephone numbers for attribute Calling-Station-ID. All of the telephone numbers are valid, so a user trying to dial into your network can call from any of the designated telephone numbers and still authenticate successfully.

If an attribute appears more than once in the return list, each value of the attribute is sent as part of the response packet. For example, to enable both IP and IPX header compression for a user, the Framed-Compression attribute should appear twice in the return list: once with the value VJ-TCP-IP-header-compression and once with the value IPX-header-compression.

Orderable Multiple-Value Attributes

Certain multiple-value return list attributes are also orderable, which means that the attribute can appear more than once in a RADIUS response, and the order in which the attributes appear is important. For example, the Reply-Message attribute allows text messages to be sent back to the user for display. A multiline message is sent by including this attribute multiple times in the return list, with each line of the message in its proper sequence.

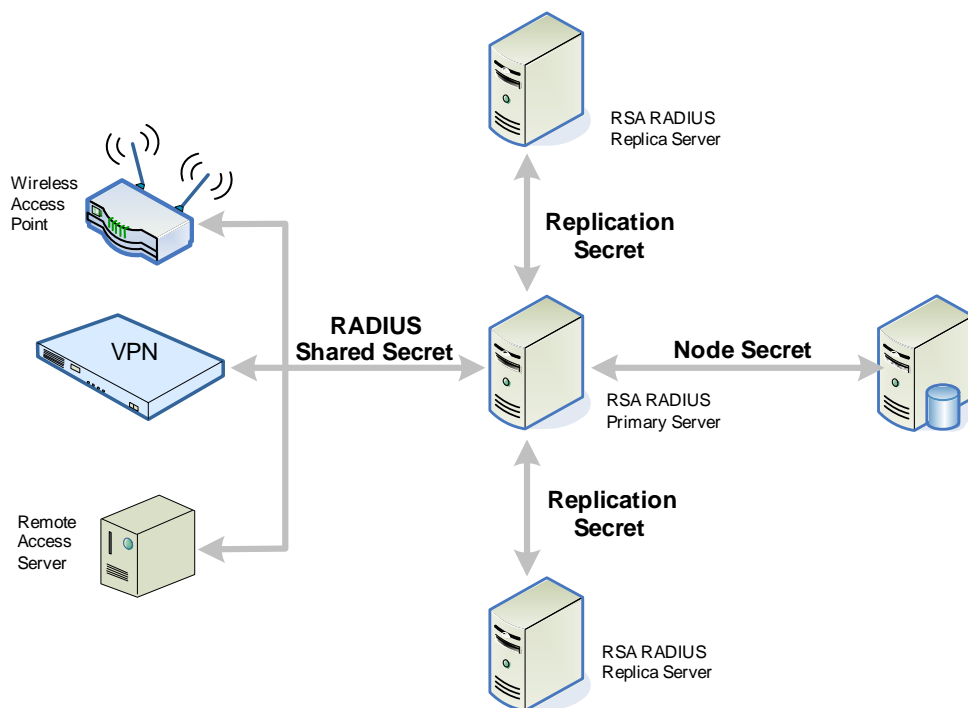
How RSA RADIUS Maintains Secure Communications

RSA RADIUS servers can secure communications with RADIUS clients and Authentication Manager using HTTPS for privacy, and various “shared secrets” that servers use to authenticate management, accounting, and authentication traffic.

A shared secret is a text string that serves as a password between hosts. RADIUS servers use three types of shared secrets:

- RADIUS shared secret – Used to secure communication between a RADIUS server and a RADIUS client.
- Replication secret – Used to secure communication between a RADIUS primary server and a RADIUS replica server.
- Node secret – Used to secure communication between a RADIUS server and an Authentication Manager server. The RADIUS primary and all replicas use the node secret.
- Accounting secret (not shown in the following figure) – Used to secure accounting traffic passed from replicas to the RADIUS primary server. The RADIUS server uses the accounting secret if it has one. Otherwise, the RADIUS server uses the RADIUS shared secret.

Shared secrets are established during installation and configuration and may be managed for RADIUS using the Security Console management interface. To set a RADIUS secret in a RADIUS client you must use that client administration interface to add the secret.



Managing User Access

RSA RADIUS administrators manage RADIUS user access by adding and modifying profiles and attributes, and assigning profiles to users, user aliases, trusted users, and agents.

Managing Profiles

Profiles support groups of users. An administrator creates a profile with attributes suitable for a specific group of users and then assigns the profile to relevant user identities defined within Authentication Manager.

For more information on RADIUS profiles, see the Security Console Help topic “RADIUS Profiles.”

Using the Default Profile

RSA RADIUS supports a default profile. When a RADIUS user authenticates and has no assigned profile, he or she receives the attributes and values that are defined in the default profile, if one is specified.

Administrators must create the default profile and make it the default. Once you have added one or more RADIUS profiles to Authentication Manager, you can specify the default profile on the Realm Configuration page in the Security Console. For more information, see the Security Console Help topic “Configure Your Realm.”

Administrators can add, remove, or modify attributes and their values within checklists and return lists for the default profile in the same manner as for regular profiles.

Note: Although RSA recommends using the Security Console to specify the default profile, you can also specify a default profile in the **securid.ini** RADIUS configuration file. The default profile specified in Authentication Manager always overrides any default profile specified in the **securid.ini** file.

Managing Checklist and Return List Attributes in a Profile

During profile creation or update, administrators can add, remove, or modify attributes and their values within checklists and return lists.

The RADIUS server examines authentication requests from RADIUS clients to confirm that attributes and values defined in a profile as checklist attributes are contained in the authentication request.

Note: By default, a RADIUS client sends all available attributes and values with authentication requests. If a RADIUS client is configured to not send some attribute and that attribute is defined in the profile as a checklist attribute, the authentication request fails. See the RADIUS client device documentation for procedures on how to configure a RADIUS client.

When authentication succeeds, RADIUS sends return list attributes to the RADIUS client along with the Access-Accept message for use in setting session parameters for that user.

Available checklist and return list attributes appear in the Security Console in the drop-down list on the management pages for creating and updating profiles. For step-by-step instructions to add, remove, or modify attributes and their values within checklists and return lists, see the Security Console Help topic “Add RADIUS Profiles”.

Note: The purpose and use of specific attributes is described in the documentation for particular RADIUS client devices and is outside the scope of this guide.

Using Dictionary Files to Customize Attributes

RSA RADIUS provides standard RADIUS attributes defined in dictionary files provided with the server. These standard attributes are sufficient to support most major brands of RADIUS client devices. If you purchase a new or specialized RADIUS client device, that device may also have its own dictionary file that contains client-specific attributes. You can install that dictionary file on each of the RADIUS servers so that new or changed RADIUS client attributes are available for inclusion in profiles.

In some rare cases, attribute names in RADIUS may differ from attribute names used by a particular RADIUS client. The Operations Console allows administrators to modify attributes defined in dictionary files. For more information, see the Security Console Help topic “RSA RADIUS Configuration File Reference.”

Managing Profile Assignments

When a profile exists, an administrator can assign or unassign users, user aliases, trusted users, or agents to the profile. Assigning a user, user alias, or trusted user to a profile applies the attributes of that profile to the user.

Assigning an agent to a profile applies the attributes of that profile to all users who use that agent. Recall that an agent refers to a RADIUS client at the network perimeter such as a VPN server, wireless access point, or network access server with dial-in modems. For more information, see [“Profile Attributes Apply to All Associated Users, Trusted Users, User Aliases, and Agents”](#) on page 161.

To avoid conflicts in cases where a profile for users and a profile for an agent could both be applied to a user, administrators must choose whether an agent profile takes precedence over a user profile. Use the Realm Configuration page in the Security Console to specify whether an agent profile or user profile takes precedence. For instructions, see the Security Console Help topic “Choose the Priority of Agent or User RADIUS Profiles.”

Manage profile assignments for users and agents using the Security Console administration pages for RADIUS profiles. After choosing a profile, use the appropriate tab (associated users, associated user aliases, associated trusted users, or associated agents) to view and manage assignments for that profile. For instructions, see the Security Console Help topics “Assign Users to RADIUS Profiles,” “Assign User Aliases to RADIUS Profiles,” “Assign Trusted Users to RADIUS Profiles,” and “Assign Agents to RADIUS Profiles.”

Managing RADIUS User Attributes

RSA RADIUS supports RADIUS user attributes that exist outside of a profile and can be assigned to individual users. RADIUS user attributes add to attributes contained in profile return lists to provide the scalability of profiles and the fine-grained control offered by individually assigned attributes. For more information, see [“RADIUS User Attributes Apply to Individual Users and Can Use Existing Identity Source Information”](#) on page 163.

There are two types of RADIUS attribute: standard and custom. Standard attributes have fixed attribute IDs (numbered from 1-63) and names defined by the RADIUS specification. In addition to the standard attributes, you can create custom attributes. Custom attribute names must not duplicate standard attribute names and the ID must be from 64 to 255.

Standard and custom RADIUS user attributes may be assigned values directly when you are assigning the attributes to individual users or they may be “mapped” to use values from an existing identity source to eliminate entering duplicate data. For example, a callback number may be stored in the organization LDAP identity store and that number can be used rather than entering it separately when assigning that attribute to a user.

The Security Console pages for RADIUS Attribute Definitions includes interfaces for managing standard and custom RADIUS user attributes. For instructions, see the Security Console Help topics “Add Custom RADIUS Attributes,” “Assign RADIUS Attributes to Users,” and “Map RADIUS User Attributes to Identity Source Attributes.”

Managing RADIUS Clients

RADIUS clients are the RADIUS-enabled VPN servers, wireless access points, network access servers, and dial-in modems at the network perimeter that enforce access control for users attempting to access network resources. These devices rely on RSA RADIUS to handle access requests for them and to provide session control attributes for those access requests that are authenticated and accepted.

To enable RADIUS servers to interact with RADIUS clients you must define them in the Security Console. You assign a name for management purposes and enter other parameters such as the IP address, type of device, and RADIUS secret. From the RADIUS menu in the Security Console, select **RADIUS Clients** and click **Manage Existing** or **Add New**. For instructions, see the Security Console Help topic “Add RADIUS Clients.”

When saving your changes, the default choice is **Save and Create Associated RSA Agent**. This creates an agent for the client that allows you to assign a profile to this client. The assigned profile may contain attributes appropriate for the client type or the users who may authenticate using that client. This choice also supports proxied authentication so the Authentication Manager can determine which RADIUS client is used for authentication and this information is saved in log files.

If you have disabled proxied authentication (by setting the **securid.ini** file parameter **CheckUserAllowed ByClient** to 0), select **Save Without RSA Agent**. In this case, you cannot assign a profile to this client and all authentications appear to Authentication Manager as though they are coming from the RADIUS server.

When a RADIUS client is defined within the Security Console, you may list, edit, and delete those entries.

Managing RSA RADIUS Servers

This section explains the administrative tasks needed to manage servers throughout their life-cycle after the initial deployment has been installed and configured.

Starting and Stopping RSA RADIUS Servers

Normally, RSA RADIUS servers start automatically when the host system starts up. In some cases, such as when you have stopped a running server, you may need to manually restart that server. Use the Operations Console to start or stop RADIUS servers.

For instructions, see the Operations Console Help topic “View RADIUS Servers.”

Adding a New RSA RADIUS Server

RSA RADIUS primary servers register with Authentication Manager automatically during installation. RADIUS replica servers register with the RADIUS primary server and Authentication Manager automatically during installation.

Registration information enables the Authentication Manager server to communicate with the new RADIUS server for management purposes. RADIUS replica servers also pass registration information to the RADIUS primary server so it knows where to send replication updates.

When you add a server, a corresponding RSA Agent is automatically created. The agent enables communication with Authentication Manager. For information about manually adding an RSA Agent, see the following section, “[List or Delete Existing RSA RADIUS Server Entries](#).”

For instructions, see the Security Console Help topic “Add RADIUS Server Agents.”

List or Delete Existing RSA RADIUS Server Entries

When RSA RADIUS servers are registered, you may want to list these servers for the purpose of viewing individual server registration information (called “properties”). You can also use the list to force replication to a selected replica or to delete a server entry in a case where a RADIUS server has been uninstalled. The following functions to list or delete RADIUS servers are in the Security Console:

- List RADIUS server information by clicking **RADIUS Servers** from the **Radius** tab in the Security Console.
- Delete RADIUS server information by selecting a server name from the list of servers, and clicking **Delete**.

For instructions, see the Security Console Help topic “View RADIUS Servers”.

View or Edit Existing RSA RADIUS Server Properties

For RSA RADIUS servers that are registered, you may want to view, change, or remove properties for an individual server. For example, if your organization's security policy requires you to change the replication secret on some periodic basis, you can edit this information directly using the Security Console.

From the list of RADIUS servers, you can view or edit properties for a single server by selecting a server name in the list and selecting **View** or **Edit** from the drop-down list. For instructions, see the Security Console Help topic "Edit RADIUS Servers."

Important: The administrator user name and password specified on the RADIUS Server page in the Security Console must be the same as the user name and password stored on the RADIUS server.

Edit an RSA Agent

Each RADIUS server has an associated RSA Agent. The RSA Agent enables server communication with Authentication Manager. From the list of RADIUS servers, select a server and select **RSA Agent** from the drop-down list to add or modify RSA Agent properties.

Agent-related management functions allow you to:

- Set alternate IP addresses in the event the system has multiple network interface cards.
- Choose whether the Authentication Manager contact list is manually assigned or assigned automatically from the Authentication Manager instance that responds first. A contact list determines the order for contacting Authentication Manager instances in case a server is offline. Automatic contact lists are automatically maintained by the system to account for any new server nodes. Manually maintained lists are customized by system administrators that have specific requirements for managing the authentication request traffic to particular server nodes.

For instructions, see the Security Console Help topic "View or Edit RADIUS Client Agents."

Managing Replication

RSA RADIUS configuration changes that you make using the Security Console are made only on the RADIUS primary server. The RADIUS replica servers can access the latest changes to configuration data such as profile assignments, user attribute definitions, and realm level settings. Some changes, such as changes to RADIUS clients, RADIUS profiles, and the replication setting for RADIUS servers, require you to manually propagate the changes to the replicas by forcing the replication process. For instructions, see the Security Console Help topic "Force Replication to a Single RADIUS Replica Server."

To manage replication, use the Security Console RADIUS Servers pages to synchronize the replication secret, configuration, and data changes across all RADIUS servers (primary and replicas).

When forcing replication, there are two options:

Force Replication to One RADIUS Server. This option is useful if a RADIUS replica server has been offline for a period of time, and is not up to date with the latest configuration changes. When the server comes back online, you can force replication to that server.

Force Replication to All RADIUS Servers. This option is useful when you must propagate changes that are not automatically propagated during the replication process. You might also use this option if multiple RADIUS replica servers have been offline for a period of time, and are not up to date with the latest configuration changes. In this case, it may be convenient to force replication to all servers rather than choose individual servers for forced replication.

Important: Make sure that you are using the correct Administrator user name and password for logging on to the RADIUS servers.

For instructions, see the Security Console Help topics “Force Replication to All RADIUS Replica Servers,” and “Force Replication to a Single RADIUS Replica Server.”

Manage EAP-POTP Configuration

Extensible Authentication Protocol (EAP), is an authentication framework that supports multiple authentication methods. This allows RADIUS to support new authentication methods without requiring any changes to the RADIUS protocol or RADIUS servers. EAP-POTP defines the method for one-time password (RSA SecurID) authentication and provides the following capabilities within RSA RADIUS:

- End-to-end protection of one-time password
- Mutual authentication
- Session key derivation for 802.1x (wireless)
- Support for token exception cases (New PIN, Next Token, others)
- Fast session resumption if a wireless connection is lost

EAP-POTP configuration is a system-wide configuration that sets basic parameters for keying material (and keys) protecting one-time passwords used for authentication.

The default values balance security (cryptographic strength) with system responsiveness and are considered satisfactory for most environments. You may increase or decrease EAP-POTP default values, however even slight changes to values used for key generation may cause a large change in response time during authentication.

EAP-POTP parameters are contained on the Security Console Authentication Manager Settings page. For instructions, see the Security Console Help topic “Configure the EAP-POTP Policy.”

Monitoring System Usage

This section explains how to monitor RSA RADIUS system usage for the purposes of detecting attacks, auditing, and troubleshooting.

Viewing RSA RADIUS Usage Statistics

Authentication statistics summarize the number of authentication acceptances and rejections, with summary totals for each type of rejection or retry.

View RADIUS server authentication statistics on the RADIUS Statistics pages in the Security Console. For instructions, see the Security Console Help topic “View RADIUS Server Statistics.”

The following table explains the authentication statistics fields and describes possible causes for some authentication rejections.

Authentication Statistic	Meaning
Transactions	
Accepts	The current, average, and peak number of RADIUS transactions that resulted in an accept response.
Rejects	The current, average, and peak number of RADIUS transactions that resulted in a reject response. These are detailed in the Reject Details section.
Overall	
Total Transactions	The sum of the accept, reject, and silent discard totals.
Silent Discards	The number of requests in which the client could not be identified. This might occur if a RADIUS client entry cannot be found for a device with the name and/or IP address of a device requesting authentication services.
Challenges	The number of challenges received from RADIUS clients.
Reject Details	
Dropped Packet	The number of RADIUS authentication packets dropped by RADIUS because the server was flooded with more packets than it could handle.
Invalid Request	The number of invalid RADIUS requests made. Possible Cause: A device is sending incorrectly formed packets to RADIUS. Either there is a configuration error or the device does not conform to the RADIUS standard.

Authentication Statistic	Meaning
Failed Authentication	The number of failed authentication requests, where the failure is due to invalid user name or password. Possible Cause: If all transactions are failing authentication, the problem might be that the shared secret entered into RADIUS does not match the shared secret entered on the client device.
Failed on Checklist	The number of requests that were authenticated but failed to meet the checklist requirements.
Insufficient Resources	The number of rejects due to a server resource problem.
Rejects Sent	
Transactions Retried	The number of requests for which one or more duplicates was received.
Total Retry Packets	The number of duplicate packets received.

View RADIUS Server Accounting Statistics

Accounting statistics provide information such as the number of transaction starts and stops and the reasons for rejecting attempted transactions. The start and stop numbers rarely match, as many transactions can be in progress at any given time.

You can view RADIUS server accounting statistics on the RADIUS Statistics pages in the Security Console. For instructions, see the Security Console Help topic “View RADIUS Server Statistics.”

The following table explains the accounting statistics fields and describes possible causes for some accounting errors.

Accounting Statistic	Meaning
Transactions	
Starts	The current, average, and peak number of transactions in which a dial-in connection was started following a successful authentication.
Stops	The current, average, and peak number of transactions in which a dial-in connection was terminated.
Overall	
Ons	The number of Accounting-On messages received, indicating that a RADIUS client has restarted.
Offs	The number of Accounting-Off messages received, indicating that a RADIUS client has shut down.

Accounting Statistic	Meaning
Total Transactions	The sum of the Starts, Stops, Ons, and Offs totals.
Interim Requests	The number of interim accounting packets sent by RADIUS clients to the RADIUS server.
Failure Details	
Dropped Packet	The number of RADIUS accounting packets dropped by RADIUS because the server was flooded with more packets than it could handle.
Invalid Request	The number of invalid RADIUS requests made. Possible Cause: A device is sending incorrectly formed packets to RADIUS. Either there is a configuration error or the device does not conform to the RADIUS standard.
Failed Accounting	The number of failed accounting requests, where the failure is due to invalid user name or password. If all accounting transactions are failing, the problem might be that the accounting shared secret entered into the RADIUS client in the Security Console does not match the shared secret entered on the client device.
Insufficient Resources	The number of rejects due to a server resource problem.
Retries Sent	
Transactions Retried	The number of requests for which one or more duplicates was received.
Total Retry Packets	The number of duplicate packets received.

View RADIUS Server's Client Authentication and Accounting Statistics

Statistic	Meaning
Transactions	
Starts	The current, average, and peak number of transactions in which a dial-in connection was started following a successful authentication.
Stops	The current, average, and peak number of transactions in which a dial-in connection was terminated.
Overall	
Ons	The number of Accounting-On messages received, indicating that a RADIUS client has restarted.
Offs	The number of Accounting-Off messages received, indicating that a RADIUS client has shut down.
Total Transactions	The sum of the Starts, Stops, Ons, and Offs totals.
Interim Requests	The number of interim accounting packets sent by RADIUS clients to the RADIUS server.
Failure Details	
Dropped Packet	The number of RADIUS accounting packets dropped by RADIUS because the server was flooded with more packets than it could handle.
Invalid Request	The number of invalid RADIUS requests made. Possible Cause: A device is sending incorrectly formed packets to RADIUS. Either there is a configuration error or the device does not conform to the RADIUS standard.
Failed Accounting	The number of failed accounting requests, where the failure is due to invalid user name or password. If all accounting transactions are failing, the problem might be that the accounting shared secret entered into the RADIUS client in the Security Console does not match the shared secret entered on the client device.
Insufficient Resources	The number of rejects due to a server resource problem.
Retries Sent	
Transactions Retried	The number of requests for which one or more duplicates was received.
Total Retry Packets	The number of duplicate packets received.

RADIUS client statistics provide information about the number of authentication and accounting requests by client.

You can view an RADIUS server's client authentication and accounting statistics on the RADIUS Statistics pages in the Security Console. You can view the following:

Summary. Displays the number of authentication requests, Access-Accepts, and Accept-Reject messages, and the total number of accounting requests, starts, and stops for each RADIUS client.

Authentication Request Details. Displays the number of duplicate messages, challenges, messages containing invalid authentication information, bad authentication requests, bad types, and dropped requests for each RADIUS client.

Accounting Request Types. Displays the number of accounting start messages, accounting stop messages, interim messages, Accounting-On messages, Accounting-Off messages, and acknowledgement messages sent for each RADIUS client.

Accounting Request Diagnostics. Displays the number of duplicate messages, messages with invalid secrets, malformed messages, messages with incorrect types, and dropped requests for each RADIUS client.

For instructions on viewing client statistics, see the Security Console Help topic "View RADIUS Client Statistics."

Choosing Accounting Attributes and Administrator Actions to Record

When users authenticate, RADIUS clients send attributes for each user authentication attempt. RADIUS servers collect this information and can save as much or as little as needed for billing or monitoring purposes. The server writes the information to a comma-delimited file suitable for inclusion in a spreadsheet or other application.

RADIUS log files record administrator actions including authentications and any changes made using the Security Console or Operations Console.

The following files establish settings for accounting and logging. For more information on modifying configuration files, see "[Modify RSA RADIUS Server Configuration and Dictionary Files](#)" on page 192. Also, see the *RADIUS Reference Guide*.

File Name	Function
account.ini	Controls how RADIUS accounting attributes are logged.
radius.ini	Controls (among other things) the types of messages that RADIUS records in the server log file and the location of the log directory.

Displaying the Authentication Log Files

Security Groups and File Permissions for Log Files (Solaris and Linux)

When you run RADIUS on a Solaris or Linux server, you can specify which users are authorized to read or edit important files, such as authentication and accounting log files. For example, you can specify that system administrators who install and configure RADIUS have read/write access for system log files and that network operators who monitor RADIUS have read-only (or no) access for system log files.

Each file and directory on a Solaris or Linux server has three security groups associated with it:

- The Owner—identifies the person who created or owns the file.
- The Group—identifies the set of users who are members of the group or groups to which the file Owner belongs. Group members can exercise special privileges with respect to that file. A user can belong to more than one group.
- The Other—consists of the set of all users who do not belong to Owner or Group.

Each security group has three flags that control what privileges that group can exercise with respect to the file or directory:

- Read (r)—determines whether the file can be read. The Read flag has an octal value of 4.
- Write (w)—determines whether the security group can create, modify, or delete the file. The Write flag has an octal value of 2.
- Execute (x)—determines whether the security group can run a script or executable file. The Execute flag has an octal value of 1.

For example, a file owner might have rwx permission for a file, which indicates that the file owner has read/write/execute access to the file. Similarly, Other might have r-- permission (where - indicates no permission), which means that the user can read but not edit or execute the file.

You can add the octal values for permission flags to generate a numeric representation of the file permissions for Owner, Group, and Other:

- 1 = execute only
- 2 = write only
- 3 = write and execute (1+2)
- 4 = read only
- 5 = read and execute (4+1)
- 6 = read and write (4+2)
- 7 = read and write and execute (4+2+1)

The security permissions exercised by Owner/Group/Other are typically expressed as string or a three-digit number. The following table provides examples of different file permissions.

Permission	Octal value	What It Means
-rwxrwxrwx	777	Read, write, and executable for Owner/Group/Other
-rw-rw-r--	664	Read and write for Owner/Group; read access for Other
-rw-rw----	660	Read and write for Owner/Group; no access for Other
-rwx-----	700	Read, write, and executable for Owner only
-rw-rw-rw	666	Read and write for Owner, Group, and all others

The UNIX **chown** command allows you to change the Owner or Group (or both) associated with a file or directory. The UNIX **chmod** command allows you to change the permissions of files and directories.

Using the User File Creation Mode Mask

The user file mode creation mode mask (umask) determines the default file system mode for newly created files of the current process. Solaris and Linux hosts typically have a hierarchy of umask values: a server-level umask value, that can be overridden by a user-, shell-, or application-level umask value. The result is an ambient umask value, which determines what file permissions are used when files are created by any given process.

The umask value is a three-digit octal number. The first digit sets the mask for Owner, the second for Group, and the third for Other. The umask value identifies the permissions that are withheld when a file is created: the umask value is subtracted from the full access mode value (777) to determine the access permissions for a new file. For example, if the umask value for a process is set to 022, the write permission for Group and Other are withheld from the full access mode value (777), resulting in a file permission of 755 (rwxr-xr-x). Similarly, if the umask value of 177 is configured for a process (explicitly or by virtue of the ambient umask), files created by the process have a file permission of 600 (rw-----). The following table summarizes the result of using different octal numbers in a umask value.

Octal Number	Access	Permission Resulting From umask Value
0	rwx	Read, Write, Execute
1	rw-	Read, Write
2	r-x	Read, Execute only
3	r--	Read only
4	-wx	Write, Execute only

Octal Number	Access	Permission Resulting From umask Value
5	-w-	Write only
6	--x	Execute only
7	---	No permissions

The umask value affects a file's access permissions only when the file is created. If you change the umask value, access permissions for existing files are not affected. Similarly, you can use the `chown` and `chmod` commands to change a file's access permissions after the file has been created.

Implementing Default File Permissions in RSA RADIUS

The `RADIUSMASK` parameter in the `sbrd.conf` file specifies the application-level umask value used to establish access permissions for all files created by RSA RADIUS. For information on configuring the `sbrd.conf` file, see the *RADIUS Reference Guide*.

If you do not specify a value for the `RADIUSMASK` parameter, RADIUS uses the ambient umask value established by the server-, user- or shell-level umask value to determine the access permissions for files it creates.

Some log files have explicit controls that allow you to override the umask value established by the `RADIUSMASK` parameter or the ambient umask value. For more information, see the following section, "[Implementing Override File Permissions in RSA RADIUS.](#)"

As previously noted, the umask value affects a file's access permissions only when the file is created. If you change the `RADIUSMASK` setting, new files created by RADIUS are assigned the access permission specified by the new setting. This includes files that roll over periodically. The existing file would retain the access file permission it received when it was created, and the new file would be assigned the access permission specified by the new `RADIUSMASK` value.

Note: The execute file permission value for files created by RADIUS is always set to None for Owner, Group, and Other. Thus, a umask value of 0 (no restrictions) is equivalent to a umask value of 1 (read/write permission) for files created by RADIUS.

Implementing Override File Permissions in RSA RADIUS

To override file permissions established by the RSA RADIUS RADIUSMASK or the ambient umask for specific log files, you must modify the LogFilePermissions parameter in the applicable initialization (.ini) file.

The following table identifies the configuration files you must modify to configure non-default file permissions for RADIUS log files.

Controlled Files	Configuration File
Server Diagnostics log (RADIUS log)	radius.ini
Accounting Library logs and header check-point logs	account.ini

The syntax for the LogFilePermissions parameter is:

LogfilePermissions = *owner:group mode*

- Specify the owner and group settings by entering character strings or decimal integers, as used for arguments to the UNIX chown(1) command. For example, ralphw:proj, ralphw:120, or 1007:120.
- Specify the mode setting as a character string or an octal integer. When permissions are specified as a character string, they follow the format that is used by the UNIX ls(1) command; for example, rw-rw-rw-. When permissions are specified as an octal integer, they follow the format used for arguments to the UNIX chmod(1) command; for example, 666.

Note: You can specify only read/write permissions for a RADIUS file. You cannot specify execute permissions for RADIUS files.

The value of each LogFilePermissions parameter is read when the RSA RADIUS server is started or restarted.

- If you enter a valid value for a LogfilePermissions parameter, the ownership and permissions of the controlled log file are set as specified whenever the file is opened or created.
- If you do not enter a value for a LogfilePermissions parameter, the ownership and permissions of the controlled file are not changed. The controlled file is created using the ownership of the account that is executing the server and the permissions that are derived from the default RADIUSMASK value or from the ambient umask setting. If the file already exists, new information is appended without changing the existing ownership and permissions of the controlled file.

- If you enter an invalid value for a LogfilePermissions setting, then the ownership of the controlled log file defaults to the effective user or group ID of the server process (normally root:other on Solaris and root:root on Linux), and the permissions for the controlled file default to 0600 (-rw-----). This ensures that the affected log file can always be opened without any escalation of file access privileges. Messages similar to the following are logged whenever an explicit file access control is misconfigured:
 - Invalid LogfilePermissions specified in radius.ini [Configuration]: -rwx-----
Server log file permissions defaulted to 0:0 0600

Configuring the Log Retention Period

Each day at midnight, the previous day's log files are completed, and new log files are created for the new day's transactions. To prevent the log files from filling up available disk space, you can configure RADIUS to discard the log files after a specified number of days.

For more information on configuring the log retention period, see the *RADIUS Reference Guide*.

Using the Server Log File

The server log file records RADIUS events, such as server startup or shutdown or user authentication or rejection, as a series of messages in an ASCII text file. Each line of the server log file identifies the date and time of the RADIUS event, followed by event details. You can open the current log file while RADIUS is running.

Server log files are kept for the number of days specified, and then deleted to conserve disk space. For more information, see the previous section, "[Configuring the Log Retention Period](#)."

Optionally, you can specify a maximum size for a server log file by entering a non-zero value for the LogfileMaxMBytes setting in the [Configuration] section of the radius.ini file.

- If a maximum file size is set, the server log filename identifies the date and time it was opened (YYYYMMDD_HHMM.log). When the current server log file approaches the specified number of megabytes (1024 x 1024 bytes), the current log file is closed and a new one is opened. The closed file will be slightly smaller than the specified maximum file size.
- If the maximum file size is set to 0 (or if the LogfileMaxMBytes setting is absent), the server log file size is ignored and log filenames are datestamped to identify when they were opened (YYYYMMDD.log).

Note: The size of the log file is checked once per minute, and the log file cannot roll over more than once a minute. The log file may exceed the specified maximum file size temporarily (for less than a minute) after it passes the LogfileMaxMBytes threshold between size checks.

By default, server log files are located in the RADIUS database directory. You can specify an alternate destination directory in the [Configuration] section of the **radius.ini** file.

Level of Logging Detail

You can control the level of detail recorded in server log files by use of the LogLevel, LogAccept, LogReject, and TraceLevel settings.

The LogLevel setting determines the level of detail given in the server log file. The LogLevel can be the number 0, 1, or 2, where 0 is the least amount of information, 1 is intermediate, and 2 is the most verbose. The LogLevel setting is specified in the [Configuration] section of **radius.ini** and in the [Settings] sections of .aut files.

The LogAccept and LogReject flags allow you to turn on or off the logging of Access-Accept and Access-Reject messages in the server log file. These flags are set in the [Configuration] section of **radius.ini**: a value of 1 causes these messages to be logged, and a value of 0 causes the messages to be omitted. An Accept or Reject is logged only if LogAccept or LogReject, respectively, is enabled and the LogLevel is “verbose” enough for the message to be recorded.

The TraceLevel setting specifies whether packets should be logged when they are received and being processed, and what level of detail should be recorded in the log.

If you alter the LogLevel or TraceLevel settings, you can have them take effect without restarting the server by issuing the following command:

- **Linux:** Enter the **kill -HUP *pid*** command.
- **Solaris:** Enter the **kill -HUP *pid*** command.
- **Windows:** Enter the **radhup** command.

Using the Accounting Log File

RADIUS accounting events are recorded in the accounting log file. Accounting events include START messages, which indicate the beginning of a connection; STOP messages, which indicate the termination of a connection, and INTERIM messages, which indicate that a connection is ongoing.

Accounting log files use comma-delimited, ASCII format, and are intended for import into a spreadsheet or database program. Accounting log files are located in the RADIUS database directory by default, although you can specify an alternate destination directory in the [Configuration] section of the **account.ini** file. Accounting log files are named **yyyymmdd.ACT**, where **yyyy** is the 4-digit year, **mm** is the month, and **dd** is the day on which the log file was created.

Accounting log files are kept for the number of days specified, and then deleted to conserve disk space. For more information, see [“Configuring the Log Retention Period”](#) on page 182.

The current log file can be opened while RADIUS is running.

Accounting Log File Format

The first six fields in every accounting log entry are provided by RADIUS for your convenience in reading and sorting the file:

- Date—The date when the event occurred
- Time—The time when the event occurred
- RAS-Client—The name or IP address of the RADIUS client sending the accounting record
- Record-Type—START, STOP, INTERIM, ON, or OFF, the standard RADIUS accounting packet types
- Full-Name—The fully distinguished name of the user, based on the authentication performed by the RADIUS server
- Auth-Type—A number that indicates the class of authentication performed:
 - 0—Native
 - 10—SecurID User
 - 11—SecurID Prefix
 - 12—SecurID Suffix
 - 13—Solaris User
 - 14—Solaris Group
 - 15—TACACS+ User
 - 16—TACACS+ Prefix
 - 17—TACACS+ Suffix
 - 100—Tunnel User
 - 200—External Database
 - (other)—Proxy

By default, the standard RADIUS attributes follow the Auth-Type identifier. See [“Standard RADIUS Accounting Attributes”](#) on page 186.

You can edit the **account.ini** initialization file to add, remove or reorder the standard RADIUS or vendor-specific attributes that are logged. For more information, see the *RADIUS Reference Guide*.

First Line Headings

The first line of the accounting log file is a file header that lists the attributes that have been enabled for logging in the order in which they are logged. The following example of a first line shows required headings in bold italic, standard RADIUS headings in bold, and vendor-specific headings in regular text:

```
"Date", "Time", "RAS-Client", "Record-Type", "Full-Name",
"Auth-Type", "User-Name", "NAS-Port", "Acct-Status-Type",
"Acct-Delay-Time", "Acct-Input-Octets", "Acct-Output-Octets",
"Acct-Session-Id", "Acct-Authentic", "Acct-Session-Time",
"Acct-Input-Packets", "Acct-Output-Packets",
"Acct-Termination-Cause", "Acct-Multi-Session-Id",
"Acct-Link-Count", "Acc-Err-Message",
"Nautica-Acct-SessionId", "Nautica-Acct-Direction",
"Nautica-Acct-CauseProtocol", "Nautica-Acct-CauseSource",
"Telebit-Accounting-Info", "Last-Number-Dialed-Out",
"Last-Number-Dialed-In-DNIS", "Last-Callers-Number-ANI",
"Channel", "Event-Id", "Event-Date-Time",
"Call-Start-Date-Time", "Call-End-Date-Time",
"Default-DTE-Data-Rate", "Initial-Rx-Link-Data-Rate",
"Final-Rx-Link-Data-Rate", "Initial-Tx-Link-Data-Rate",
"Final-Tx-Link-Data-Rate", "Sync-Async-Mode",
"Originate-Answer-Mode", "Modulation-Type",
"Equalization-Type", "Fallback-Enabled", "Characters-Sent",
"Characters-Received", "Blocks-Sent", "Blocks-Received",
"Blocks-Resent", "Retrains-Requested", "Retrains-Granted",
"Line-Reversals", "Number-Of-Characters-Lost",
"Number-of-Blers", "Number-of-Link-Timeouts",
"Number-of-Fallbacks", "Number-of-Upshifts",
"Number-of-Link-NAKs", "Back-Channel-Data-Rate",
"Simplified-MNP-Levels", "Simplified-V42bis-Usage", "PW_VPN_ID"
```

Comma Placeholders

RSA RADIUS writes accounting events to the accounting log file. If an event recorded in the accounting log file does not have data for every attribute, a comma “placeholder” marks the empty entry, so that all entries remain correctly aligned with their headings. For example, based on the “first line” of headings described above, the following is a valid accounting log entry, in which the value of the Acct-Status-Type attribute is 7:

```
"12/23/1997", "12:11:55", "RRAS", "Accounting-On",
,,,,,7,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
```

Standard RADIUS Accounting Attributes

The following table lists the standard RADIUS accounting attributes defined in RFC 2866, "RADIUS Accounting."

RADIUS Accounting Attributes	Description
User-Name	The name of the user as received by the client.
NAS-Port	The port number on the client device.
Acct-Status-Type	A number that indicates the beginning or ending of the user service: 1—Start 2—Stop 3—Interim-Acct 7—Accounting-On 8—Accounting-Off
Acct-Delay-Time	Indicates how many seconds the client has been trying to send this record, which can be subtracted from the time of arrival on the server to find the approximate time of the event generating this request.
Acct-Input-Octets	Number of octets (bytes) received by the port over the connection; present only in STOP records.
Acct-Output-Octets	Number of octets (bytes) sent by the port over the connection; present only in STOP records.
Acct-Session-Id	Identifier used to match START and STOP records in a log file.
Acct-Authentic	Indicates how the user was authenticated by RADIUS, the network access device (local), or another remote authentication protocol: 1—RADIUS 2—Local 3—Remote
Acct-Session-Time	Elapsed time of connection in seconds; present only in STOP records.
Acct-Input-Packets	Number of packets received by the port over the connection; present only in STOP records.
Acct-Output-Packets	Number of packets sent by the port over the connection; present only in STOP records.

RADIUS Accounting Attributes	Description
Acct-Termination-Cause	Number that indicates how the session was terminated; present only in STOP records: 1—User Request 2—Lost Carrier 3—Lost Service 4—Idle Timeout 5—Session Timeout 6—Admin Reset 7—Admin Reboot 8—Port Error 9—NAS Error 10—NAS Request 11—NAS Reboot 12—Port Unneeded 13—Port Preempted 14—Port Suspended 15—Service Unavailable 16—Callback 17—User Error 18—Host Request
Acct-Multi-Session-Id	Unique accounting identifier to make it easy to link together multiple related sessions in a log file.
Acct-Link-Count	The count of links that are known to have been in a given multilink session at the time the accounting record is generated.

Maintaining RSA RADIUS Servers

This section explains how to maintain and modify your RSA RADIUS servers for such purposes as servicing a host system, improving authentication response time, or recovering if one of your RADIUS servers becomes unavailable. This section describes the following maintenance tasks:

- Remove a RADIUS server from service.
- Back up a RADIUS server.
- Restore a RADIUS server.
- Promote a RADIUS server.
- Modify RADIUS configuration and dictionary files.
- Change the IP address or name of a RADIUS server.

Consider a scenario where a deployment of Authentication Manager and RADIUS encompasses multiple geographic locations. Having RADIUS servers and Authentication Manager servers in each location ensures that authentications can be handled locally without traversing a Wide Area Network (WAN) that might slow down response time.

Removing an RSA RADIUS Server from Service

If you need to remove a RADIUS server from service, perform the following steps in advance to minimize disruptions:

- Configure a replacement server before you remove the server in question.
- If the new server has a different IP address than the old server, reconfigure RADIUS clients so that they can send requests to the new server.
- If the server being removed is a primary server, you must promote a replica server to become the primary.

Now you can stop the RADIUS server, removing it from service. Use the Operations Console to stop the server. For instructions, see the Operations Console Help topic “Start or Stop a RADIUS Server.”

If you are permanently removing this server from service, you can delete the server entry using the Operations Console. For instructions, see the Operations Console Help topic “Delete a RADIUS Server.”

Backing Up a RADIUS Server

If you want to back up the RADIUS primary server, you must back up the entire installation of RADIUS. This includes copying all of the associated files and directories located in the RADIUS subdirectory of your Authentication Manager installation. The procedure is the same for both local and remote RADIUS servers.

Important: Make sure to store the backup data in a safe location on a separate machine, in the event that your primary server becomes unavailable

Backing Up RADIUS on Windows

For example, if you have Authentication Manager installed in the default directory:

```
C:\Program Files\RSA Security\RSA Authentication Manager
```

then back up the RADIUS subdirectory:

```
C:\Program Files\RSA Security\RSA Authentication  
Manager\radius
```

Backing Up RADIUS on Linux

For example, if you have Authentication Manager installed in the default directory:

```
/usr/local/RSASecurity/RSAAuthenticationManager
```

then back up the RADIUS subdirectory:

```
/usr/local/RSASecurity/RSAAuthenticationManager/radius
```

Restoring a RADIUS Server

There are times when you need to restore the backed up RADIUS directory, either on the same machine or a new machine.

Restoring RADIUS on the Same Machine

Restoring RADIUS on the same machine involves copying the backup data back on to the machine in the same directory location. For example, if you have RADIUS in the following directory:

```
C:\Program Files\RSA Security\RSA Authentication  
Manager\radius
```

then restore the backup data back to the same RADIUS directory.

To restore the data on the same machine:

1. Copy the backup data back to the appropriate directory location.
2. On the RADIUS server, open a new command shell and change directories to ***RSA_AM_HOME/config***. Do one of the following:

- On Windows, type:

```
configUtil.cmd configure radius
finalize-radius-restore
```

- On UNIX, type: (as root)

```
configUtil.sh configure radius finalize-radius-restore
```

For more information on backing up RADIUS and the directory locations, see [“Backing Up a RADIUS Server”](#) on page 189.

Restoring RADIUS on a New Machine

If you want to restore the data on a different machine, you must copy the backup data on to the new machine in the appropriate directory location.

To restore the data on a new machine:

1. Copy the backup data to the new machine.
2. Install Authentication Manager with RADIUS on the new machine.
3. Stop the RADIUS server.
4. Overwrite the existing RADIUS directories with the backup data.

For more information on the directories and their locations, see [“Backing Up a RADIUS Server”](#) on page 189.

5. On the RADIUS server, open a new command shell and change directories to ***RSA_AM_HOME/config***. Do one of the following:

- On Windows, type:

```
configUtil.cmd configure radius
finalize-radius-restore
```

- On UNIX, type: (as root)

```
configUtil.sh configure radius finalize-radius-restore
```

Promoting an RSA RADIUS Replica Server

You promote a RADIUS replica server to be the new RADIUS primary server in either of these situations:

Planned Promotion. The primary server is online and functioning, but you want to promote an existing replica. You might choose to do this when migrating the primary instance to another machine.

Disaster Recovery. The primary server is unavailable and you want to promote an existing replica, or the replica server is unavailable and you want to add a new replica.

Important: Before promoting a replica, make sure that it is synchronized with the primary. You can do this by forcing replication to the replica. This updates the replica server with the latest configuration changes.

Planned Promotion

To promote a replica server when the existing primary server is available:

1. Force replication to the RADIUS replica that you are going to promote so that the replica has all of the latest configuration changes.
2. Use the Operations Console on the Authentication Manager primary server to promote the existing RADIUS replica server to be the RADIUS primary server. For instructions, see the Operations Console Help topic “Promoting a RADIUS Replica Server.”

Disaster Recovery

If your RADIUS primary server is down, you need to promote one of the RADIUS replicas to be the new RADIUS primary. You might also run into a situation when one of your RADIUS replicas is down, and you want to add a new replica to your deployment.

Important: If your RADIUS primary server is down and you do not have RADIUS replicas in your deployment, you must recover the RADIUS data on to a new machine. For more information, see “[Restoring RADIUS on a New Machine](#)” on page 190

To promote a replica server when the existing primary server is unavailable:

1. Use the Operations Console on the Authentication Manager primary server to promote the existing RADIUS replica server to be the RADIUS primary server. For instructions, see the Operations Console Help topic “Promoting a RADIUS Replica Server.”

Note: When you promote a RADIUS replica, all of the other RADIUS replicas start pointing to the new RADIUS primary.

2. Repeat any administrative changes made to the original RADIUS primary server that were not replicated to RADIUS replica servers. Check the time of the most recent replication provided on the RADIUS Server Properties page in the Security Console. Examine Security Console log files to identify changes that have been made since the most recent replication occurred.

You also might have a situation when one of your RADIUS replicas is down and you want to add a new replica.

To add a new replica when the existing replica is unavailable:

1. Use the installer to install a new RADIUS replica.
2. In the Security Console, force replication to the new replica. For instructions, see the Security Console Help topic “Force Replication to a Single RADIUS Replica Server.”

After forcing replication to the new RADIUS replica, the server is synchronized with the existing RADIUS servers.

Modify RSA RADIUS Server Configuration and Dictionary Files

Each RSA RADIUS server has a set of configuration, initialization, and dictionary files for customizing the server within its working environment. For many situations, the default files provide sufficient functions without any modification.

However some situations may require you to modify one or more files. For example, you may need to modify these files if any of the following situations apply to your deployment:

- You are modifying logging options as explained in “[Choosing Accounting Attributes and Administrator Actions to Record](#)” on page 177.
- You are disabling proxied authentication as explained in “[Managing RADIUS Clients](#)” on page 169.
- You are configuring a RADIUS client that has special attribute names that need to be included in RADIUS profiles. Dictionary files provided with the RADIUS client must be added to the each RADIUS server that will handle requests from that client. You may need to edit the **mapping.ini** file to properly associate attribute names across different RADIUS clients.

Use the Operations Console on each RADIUS server to modify configuration files on that server. RADIUS does not automatically replicate these changes to other servers because parameters and settings may be platform specific. Any changes you make may have to be duplicated on other RADIUS servers as needed.

Other customizations may also be needed for your specific environment. For more information and for details on modifying parameters in configuration files, see the *RADIUS Reference Guide*. For instructions on editing configuration files, see the Security Console Help topic “Manage RADIUS Server.”

Change the IP Address or Name of an RSA RADIUS Server

There may be times when you need to change the IP address or name of an RSA RADIUS server. For example, if you have a test machine in a lab environment that you want to promote to production, you need to update the IP address in the Security Console so that Authentication Manager can communicate with the new server.

For information on changing the IP address or name of a RADIUS server, see Appendix E, [“Updating Server IP Addresses and Names.”](#)

9

Logging and Reporting

- [Configuring RSA Authentication Manager Logging](#)
- [Generating Reports](#)
- [Configuring SNMP](#)
- [Using the Activity Monitor](#)

Configuring RSA Authentication Manager Logging

RSA Authentication Manager maintains logs of all system events. You can use these logs to monitor the system and maintain an audit trail of all logon requests and operations performed using the RSA Security Console.

Authentication Manager maintains the following logs:

Trace. Captures log messages that you can use to debug your system.

Administrative Audit. Captures log messages that record any administrative action, such as adding and editing users.

Runtime Audit. Captures log messages that record any runtime activity, such as authentication and authorization of users.

System. Captures log messages that record system level messages, such as “Authentication Manager Server started,” and “Connection Manager lost db connection.”

For each of the logs, you can configure the level of detail written to the log files. The following logging levels are available for the Trace log:

Fatal. Captures only log messages that imply the imminent crash of the application or the relevant subcomponent. Messages logged at this level require immediate attention.

Error. Captures all fatal messages, as well as error conditions that must be addressed, but may not necessarily cause the application to crash.

Warning. Captures all fatal and error messages, as well as log messages for minor problems while the application is running.

Information. Captures all fatal, error, and warning messages, as well as log messages for significant events in the normal life cycle of the application.

Verbose. Captures all fatal, error, warning, and information messages, as well as log messages associated with minor and frequently occurring but otherwise normal events.

Important: Do not set the trace logging level to “verbose” for extended periods of time unless instructed to do so by RSA Customer Support. This takes up large amounts of disk space and can impact system performance.

The following log levels are available for the Administrative Audit, Runtime Audit, and System logs:

None. No messages are logged.

Error. Captures all error messages.

Warning. Captures all error and warning messages.

Success. Captures all error, warning and successful action messages.

You can select different levels of log detail for each of the four log files. For example, you might choose to record only fatal errors in the Administrative Audit log, but you may want to record all messages in the System log.

If you change the logging levels and want to return to the Authentication Manager default values, use the following:

- Trace Log: Fatal
- Administrative Audit Log: Success
- Runtime Audit Log: Success
- System Log: Warning

Note: Some routine system activity is only written to the System log when the logging level is set to “information” or “verbose.” Use either of those logging levels if you want to see this information written to the internal database.

In addition to selecting the level of detail of recorded messages, you can specify the destination of system log messages. On the Instance Configuration page in the Security Console, select **Send system messages to OS System Log** on the **Logging** tab if you want all system log messages written to the NTEventLog or UnixSysLog instead of the System log.

Important: Log settings are instance based. Any changes you make to one instance are not changed in the other instances.

Note: Data from the Administrative Audit, Runtime Audit, and System logs is stored in the internal database. Data from the Trace log is written to a file in **RSA_AM_HOME/server/logs**. The Administrative Audit, Runtime Audit, and System logs are also written to that location.

To use the Security Console to configure event logging, go to the RSA Server Instance Configuration page and select the **Logging** option in the instance Context menu.

Note: Only the Super Admin can manage log settings.

For instructions, see the Security Console Help topic “Configure Logging.”

For information on viewing log messages in real time, see “[Using the Activity Monitor](#)” on page 206.

Archiving Log Files

You can archive log files so that you have an audit trail of all logon attempts and Security Console operations. By archiving log files, you can maintain a history of all tasks performed using Authentication Manager. During archiving, the log files are recorded from their respective database tables and written to a comma-delimited flat file. You can then copy the flat files to external media and move them to long-term storage.

There are two ways to archive the logs in Authentication Manager. Both are done through the Security Console:

- Schedule a one-time log archive job.
- Schedule a recurring log archive job. Archive jobs can run automatically on specified days, weeks, or months.

Important: When all of the disk space is consumed, Authentication Manager may stop responding and be difficult to restore. You must devise policies and procedures to ensure that logs are archived and moved to external media on a regular basis. The log archiving frequency depends on many factors. For more information, see the chapter “Logging and Reporting” in the *Planning Guide*.

For instructions, see the Security Console Help topics “Schedule a One-Time Log Archive Job” and “Schedule a Recurring Log Archive Job.”

You can also use the Security Console to do the following:

- Cancel an archive log job. For more information, see the Security Console Help topic “Cancel a Log.”
- Export logs from the internal database. For more information, see the Security Console Help topic “Export Logs From the Internal Database.”
- Purge logs from the internal database. For more information, see the Security Console Help topic “Purge Logs From the Internal Database.”

Generating Reports

You can use Authentication Manager to create and run customized reports describing system events and objects (users and tokens, for example). These reports can provide you with more detailed information on the events that occur within the system.

For example, you might want to create a report that shows you all of the user accounts that are disabled. You can design the report so that it includes relevant information such as User ID, name, identity source, and security domain.

You might also want to create a report that shows administrator activities. You can use the report to view activities for all administrators or you can customize the report to display detailed information on one administrator. There are many reporting options available to you.

In addition to creating and running customized reports, you can configure how your reports are run, and you can view completed reports. Use Authentication Manager to perform the following reporting tasks:

View and download reports. View and download completed reports. You can also view reports that are currently running or reports that are waiting in the report queue.

Schedule recurring report jobs. Specify the frequency and run time of your reports.

Transfer reports to other security domains. When running a report, change the report ownership so that the administrative scope is narrowed or broadened.

You can access all of these reporting capabilities using the Reporting menu in the Security Console. All of the reporting functionality is described in detail in the sections that follow.

For more information on the reporting functionality, see the Security Console Help topic "Reports."

Creating Custom Reports

You can create custom reports using the predefined set of report templates provided with Authentication Manager. Each template includes a predefined set of variables, column headings, and other report information. The following templates are available:

- All Users
- All User Groups
- Distributed Token Requests
- Administrators with a Specified Role
- Users with Disabled Accounts
- Administrators of a Security Domain
- Users and User Groups Missing From Identity Source
- Expired User Accounts
- Administrator Activity

- User and User Group Life Cycle Activity
- Object Life Cycle Activity (Non User/User Group)
- System Log Report
- Authentication Activity
- Principals with Tokens Set Online Emergency Access Tokencode
- Agents with Unassigned IP Address
- Agent Not Updated By Auto-registration More than Given Number of Days
- Token Expiration Report
- List All Principals with Assigned RADIUS Profile for a Given Realm
- List All RSA Agents with Assigned RADIUS Profile for a Given Realm
- List All RSA Agents with Assigned RADIUS Client/RADIUS Server for a Given Realm
- Software Token Deployed on Device Report
- Administrators with Fixed Passcode
- Users with Days Since Last Login Using Specific Token
- Principals with Tokens Using Wildcards
- Principal Never Logged In with Token
- General On-Demand Tokencode Service
- Event Token Expiration by Event
- Credential Manager Distribution Reports
- Users Enabled for On-Demand Tokencode Service Who Have Never Logged In

For each template, click **View** in the template Context menu to see the template name and category, the input parameters, and the output columns.

The template that you select is the foundation for your custom report. The template offers a starting point for sets of data, including input parameters and output columns. You can customize the report further by adding and removing columns, applying data filters, and modifying run options. You can customize these report attributes:

Security Domain. Select the security domain. The security domain designates which administrators can manage the report, name the report, manage output columns, and so on.

Report Name. Name the report.

Run As. Choose the administrative scope for the report. You can run the report using your administrative scope, or as the report owner. If you run the report as the report owner, the data falls within the administrative scope of the administrator who created the report.

For example, assume that an administrator runs a report as the report owner, if the report owner has a broader scope than the administrator who runs the report, the report will include data from the broader scope, rather than from the more narrow scope of the administrator. For more information on administrative scope and changing report ownership, see [“Setting Report Ownership”](#) on page 202.

Columns. Select the columns that you want to display in the report. You can choose from all of the fields associated with the template.

Input Parameter Values. Filter the report results. The input parameters are different for each report template, as different information is relevant to each type of report. For example, if you want to create a report that shows disabled user accounts, you can configure the input parameters so that the report only shows disabled accounts belonging to users with a certain last name.

For instructions, see the Security Console Help topic “Add New Reports.”

Running Reports

After creating a report, you can run the report to submit it for processing. To use the RSA Security Console to run a report, select the **Run Report Job Now** option from the report Context menu on the Reports page.

When you run a report, you can specify these parameters:

Report Job Name. Specify the generated report name. The system provides a default value based on the report name and the current date and time.

Test Run. Limit the number of rows of data returned in the report.

Input Parameter Values. Filter the report results. The input parameters are different for each report template, as different information is relevant to each type of report. Note that you cannot configure a report's parameter if that parameter was configured when the report was created.

For example, assume you create a report based on the Administrators template. The Administrators template only includes two input parameters, Role Name and Admin Role Domain ID. If you create the report so that it only displays data for Role Name = Super Admin, you cannot configure that parameter when running the report. You would be able to configure the Admin Role Domain ID, but the Role Name would be read-only. This functionality applies to all of the report templates.

It is important to know that your administrative permissions dictate which reports you can run. For example, most reports gather security domain information, so you must have permission to view security domains in order to run the report. Administrators who do not have permission to view security domains can only run the following reports:

- Distributed Token Requests
- Administrators with a Specified Role
- Users with User Groups Missing from Identity Source
- Administrator Activity
- Object Life Cycle Activity (Non User/User Group)
- System Log Report
- Authentication Activity

When you are done configuring the report parameters, click **Run Report** to submit the report. To view your report output, go to the Report Output page in the Security Console. For more information on viewing and downloading reports, see [“Viewing Reports”](#) on page 203.

For instructions, see the Security Console Help topic “Run Report Jobs.”

Scheduling Recurring Reports

You can use the Security Console to create recurring report jobs. By scheduling recurring report jobs, you can gather administrative information at specific time intervals.

For example, assume you create a custom report that includes information on assigned tokens. You might want to schedule a report job that runs monthly so that you can track the number of tokens that you assign each month. You can create report jobs that run daily, weekly, or monthly.

When creating a recurring report job, you can configure these parameters:

Job Starts. Specify the date on which you want to schedule the first instance of the recurring report job

Frequency. Specify the frequency of the recurring report:

- Daily or Weekly - You can choose to have the report run each day, or you can specify a specific day or multiple days.
- Monthly - You can choose to run the report every month, or you can specify certain months. You can also specify the day on which the report is run.

Run Time. Specify the time of day that the report is run.

Job Expires. Specify the date on which the recurring report job expires. You can choose a job that does not expire or you can enter a specific date.

Input Parameters. Filter the report results. The available values vary depending on the report. For example, if you have a report that shows disabled user accounts, you can schedule the report to only show disabled accounts belonging to users with a certain first name.

To use the Security Console to create a recurring report job, select the **Schedule Report Jobs** option in the Reporting menu.

For instructions, see the Security Console Help topic “Scheduling Recurring Report Jobs.”

Setting Report Ownership

Each custom report has an owner. The owner is the administrator who created the report. When a report is run, the report is limited by the administrative scope of its owner.

For example, assume you work for the FocalView Software Company and you are the administrator of the Boston security domain. You create a custom report that shows all disabled users. By default, the report shows only those disabled users belonging to your security domain.

Authentication Manager gives you the ability to change report ownership when creating or editing a report. When you change report ownership, you are changing the administrative scope of the report.

For example, assume again that you are the administrator for the Boston security domain mentioned in the previous example. When you run your custom report on disabled users, you see only disabled users belonging to the Boston security domain.

Assume another administrator, a Super Admin, chooses to run your report. The Super Admin can edit your report and change the ownership so that the report is based on his administrative scope. In this case, the report would include disabled users in all security domains, not just the Boston security domain.

Report ownership is set using the **Run As** field on the Add New Report and Edit Report pages in the Security Console. Choose from the following options:

- Run the report as the administrator running the report job.
- Run the report as the report owner.

For instructions, see the Security Console Help topic “Edit Reports.”

Viewing Reports

You can view and download completed reports. You can also view a list of the reports that are in progress or waiting in the report queue.

Note: You can only see report output for completed reports.

From the Security Console, view and download report data on the Report Output page. The report jobs are grouped into two categories, Completed and In Progress. You can view the report jobs based on their status, or you can use the search criteria to search for a specific reporting job.

Use the report job Context menu to view or download the report output for a completed report. The Context menu gives you these view and download options:

- View in Browser - View the selected report in a separate browser window. You can save the report after you view it.
- Download CSV file - Download and save the data to a CSV file.
- Download XML file - Download and save the data to an XML file.
- Download HTML file - Download and save the data to an HTML file.

When you download a file, a dialog box prompts you to decide whether you want to open or save the file. You can view the file or save it to your computer.

To use the Security Console to view and download report output, select the **Report Output** option in the Reporting menu.

For instructions, see the Security Console Help topic “View Completed Report Jobs.”

Configuring SNMP

If you use a Network Management System (NMS) and have Simple Network Management Protocol (SNMP) running on your network, you can configure Authentication Manager to send traps to the NMS. You can also configure Authentication Manager so that the NMS can request information from Authentication Manager.

Note: Only the Super Admin can configure network management settings.

You can configure Authentication Manager for both gets and traps:

Gets. A “get” occurs when the NMS requests specific information from Authentication Manager. You configure the NMS to request this data. For example, a get might request the number of successful authentications. In this case, Authentication Manager would send the total number of successful authentications since the server was started.

Traps. SNMP trapping allows you to use the NMS to monitor error events occurring within Authentication Manager. When an error event occurs, Authentication Manager sends a notification to the NMS. You can configure the NMS to receive these notifications. Notifications can be intercepted and filtered based on the data sent in the trap message (message ID, for example).

Gets and traps differ in two ways:

- A get is information that is requested, whereas a trap is information that is sent automatically.
- A get is composed of aggregate data, but a trap is an individual piece of data. Using the example above, assume that you have Authentication Manager configured to send notifications whenever a successful authentication occurs. If there are 100 successful authentications, 100 trap messages are sent. If you were to do a get for successful authentications, you would receive one message showing a value of 100.

Note: SNMP and network settings are instance based. Changes that you make to one instance are not changed in the other instances.

To configure Authentication Manager for SNMP trapping, you must provide the following information:

Network Management. Select the checkbox to enable SNMP.

SNMP Adaptor Port. The port number of the SNMP adaptor.

Important: Ensure that this port is not already in use on any of the machines in the cluster.

SNMP Community String. The password to access the SNMP adaptor.

Access Control List. The IP addresses, hostnames, or subnet (in CIDR notation) of the machines that are able to access the SNMP agent and request information from Authentication Manager.

Include in Trace Log. Decide whether you want SNMP adaptor log messages written to the trace log.

SNMP Trap Receivers. The IP address or hostname of the machines to receive the Authentication Manager SNMP trap notifications. The default port number is 162.

Administrative Audit Log Trap Level. The level of administrative audit log messages to be trapped.

Runtime Audit Log Trap Level. The level of runtime audit log messages to be trapped.

System Log Trap Level. The level of system log messages to be trapped.

Important: Trapping levels are cumulative. For example, if you select Success, the system traps Success, Warning, and Error events. If you select Error, the system traps only Error events.

Interpreting SNMP Values

Your Authentication Manager deployment configuration affects the values returned for SNMP traps and gets. Note the following:

- Although SNMP settings are instance based, trap and get values are node based. This means that if you receive a trap from a particular node, then that is the node where the event occurred. For example, if you are trapping the number of successful authentication requests, you will receive the number of successful authentications for that particular node.

Note: Because all of the nodes share the same database, database-specific data is the same for all nodes. The only time database-specific data might differ among nodes is if there is a delay in replication.

- SNMP gets from the database component obtain the values only from the internal database, not from external identity sources. For example, assume that you have 2000 users in your external identity source but only 1000 users in the Authentication Manager internal database. If you have a get for the total number of users, the value returned is 1000.
- SNMP gets from the database component are aggregated across the entire deployment, not per realm. Therefore, the count may be larger than the number of objects visible in the Security Console, as the Console only displays values for one realm.
- SNMP gets for policies are aggregated across the entire deployment, as mentioned above. The gets count the system default policy upon which all other policies are based. The system default policy exists in the database only, and is not displayed in the Console. Thus the returned value for a policy always includes an extra value. For example, assume the Security Console displays three total password policies. If you do a get to obtain the total number of password policies, the returned value is four.

To use the Security Console to configure for SNMP trapping, go to the RSA Server Instance Configuration page and select the **Network Management (SNMP)** option in the instance Context menu.

For instructions, see the Security Console Help topic “Configure for SNMP Trapping.”

RSA Authentication Manager Message IDs

To view the Authentication Manager message IDs and causes, see Appendix F, [“RSA Authentication Manager Message IDs.”](#)

Using the Activity Monitor

An Activity Monitor allows you to view log messages in real time. You can use these real-time messages to see what is happening in the system, or you can use them for troubleshooting. For example, you might need to assist a user who is having trouble authenticating. Looking at the log messages for the user's activities can help you figure out why the user is having trouble.

There are three different Activity Monitors in Authentication Manager, each of which contains different types of log messages:

Authentication Activity Monitor. Displays authentication-specific events such as authentication requests and restricted agent access checks.

System Activity Monitor. Displays system events such as the replication of data.

Administrator Activity Monitor. Displays administrator activities such as creating and updating system administrators.

Note: The Activity Monitor opens in a new browser window and there is no limit to the number of windows that you can open. For example, you can simultaneously monitor a specific administrator, an entire user group, and an entire security domain.

Once the Activity Monitor is launched, you can filter the log events using the available filter criteria. For example, you can use the criteria to filter the data so that you can view the activity of a single administrator, a single authentication agent, or an entire security domain.

You can also configure the display attributes of the log events. For example, you can set the number of messages that the Activity Monitor displays at any given time, up to 1,500 messages. You can also configure the type of messages that the monitor displays. For example, you can view:

- Successful events
- Warning events
- Failure events
- A combination of any of the above event types

New messages are added at the top of the Activity Monitor display. As you reach the message limit that you specified, older messages are removed from the display. For example, if you configured the monitor to display only 50 messages, each message after 50 is added to the top of the display and the oldest message is removed from the display. Click **Clear Monitor** to clear all of the messages from the display.

You can also pause the Activity Monitor display. This allows you to take as much time as you want to view specific log messages. When you resume monitoring, all of log messages that were generated while the monitor was paused are added at the top of the Activity Monitor. If the number of new messages exceeds the number of messages you chose to display, only the most recent messages are displayed.

The Activity Monitor adds a message to the display whenever you pause or resume the monitor. This allows you to keep track of where the pauses occurred within the set of log messages.

Note: You cannot filter log messages while the Activity Monitor is paused.

You can view more details of an event by clicking on it. This launches a pop-up window that displays detailed information on the event. For example, a system log event includes information on the instance, client and node IP addresses, component, and other important system-related data.

To use the Security Console to launch the Activity Monitors, select the **Real-time Activity Monitor** option in the Reporting menu.

For instructions, see the Security Console Help topic “Manage the Activity Monitor.”

10 Disaster Recovery

- [Backing Up and Restoring the Internal Database](#)
- [Restoring an Installation for a Standalone Primary Instance](#)
- [Detecting a Failed Primary Instance or Replica Instance](#)
- [Promoting a Replica Instance to a Primary Instance](#)
- [Reconfiguring CT-KIP After Promoting a Replica](#)
- [Removing a Replica Instance](#)
- [Reattaching a Demoted Primary Instance](#)
- [Resynchronizing a Diverged Replica Instance](#)
- [Restoring a Super Admin](#)

Important: For RSA RADIUS disaster recovery information, see Chapter 8, [“Managing RSA RADIUS.”](#)

Backing Up and Restoring the Internal Database

Back up your system on a regular basis to avoid losing data if you inadvertently delete data from the primary instance and that mistake is propagated to the replica instances. You can use the RSA Operations Console to back up and restore all data in the primary instance database server, including configuration, policy, users, and groups, in one operation. The operation also backs up the log files that are stored in the database:

- Administrative Audit
- Runtime Audit
- System

For information on backing up an RSA RADIUS server, see [“Backing Up a RADIUS Server”](#) on page 189.

When to Perform a Backup

You need to decide how often to back up the database. Consider these issues:

- How much data can you afford to lose in the event of a crash? Although configuration and policy data might change infrequently, consider that new log data is added regularly. To minimize data loss for data that changes frequently, perform frequent backups.
- Because the backup operation can affect general system performance, consider running backups during off-peak periods. The database can still service requests while you perform the backup.

Prerequisites

If you are backing up or restoring a database on a Network File System (NFS) partition, you must mount the partition with values in the `rsize` and `wsize` fields. These values force connection attempts to retry until the NFS file server responds.

Field	Purpose	Recommended Value for NFS v. 2	Recommended Value for NFS v. 3
<code>rsize</code>	NFS read chunk size	8192	32768
<code>wsize</code>	NFS write chunk size	8192	32768

See your NFS documentation for instructions on configuring these fields.

Performing the Backup

There are two ways to configure your Authentication Manager database:

- The database is on the same machine as the Authentication Manager installation. This is the more common configuration.
- The database is on a different machine than the Authentication Manager installation.

When backing up and restoring Authentication Manager data, the location of the database determines the methods you can use to do the backup and restore. If the database is on the same machine as Authentication Manager, you can use the Operations Console for these tasks. If the database is on a different machine, you must use the Manage Backups utility for these tasks.

If you use RSA Credential Manager, you must manually back up and restore the following Workpoint files located in **`RSA_AM_HOME/workpoint`**:

- **`workpoint-client.properties`**
- **`workpoint-server.properties`**
- **`License.xml`**

If you want to back up your Credential Manager workflow and CLU log files, back up and restore the files in the following folders:

- **`RSA_AM_HOME/server/logs/Workpoint`**
- **`RSA_AM_HOME/server/logs/CLU`**

Note that the backup process backs up the database and the secrets, whether you use the Operations Console or the Manage Backups utility.

Important: After a backup, store the backup files in a safe location so that they are readily available if needed.

Backing Up a Database

If your database is on the same machine as Authentication Manager, use the Operations Console to back up the database. For information and instructions, see the Operations Console Help topic “Back Up Your Database.”

Note: When you use the Operations Console to back up RSA Authentication Manager, you can choose to back up the database and the log files, or the log files only.

Backing Up a Database on a Separate Machine

If your database is on a separate machine, you cannot use the Operations Console to back up the database. In this case, you can back up the database by running the Manage Backups utility on the machine hosting the database.

The following procedure uses the Manage Backups utility. For more information on the utility and its options, see “[Manage Backups Utility](#)” on page 271.

To back up a database hosted on a separate machine:

1. On the machine hosting the database, open a new command shell and change directories to ***RSA_AM_HOME/utlils***.
2. Type:

```
rsautil manage-backups --action export
-f absolute path
```

where *absolute path* is the absolute path and filename of the backup file, including the file extension. For example: C:\backup\filename.dmp.

In this example, the system generates two files based on the name you provide, and puts them in the C:\backup folder:

filename.dmp The database backup file.

filename.secrets. The user credentials backup file.

Note: To create a script for automated backups, see the following section, “[Automated Backups](#).”

Automated Backups

You can also automate database backups by writing a script that calls the Manage Backups utility. Make sure that the script sends the backed-up data either to a disk that is separate from the actual database, or to tape. These measures prevent a single point of failure for the database and backup.

Note: If your database is on a separate machine, you must run the Manage Backups utility on the machine that hosts the database.

To use the Manage Backups utility to run automated backups, see “[Manage Backups Utility](#)” on page 271.

Restoring the Database from a Backup

You need to restore the primary instance database server if you have inadvertently deleted data from the primary instance and that mistake has been propagated to the replicas. During a restore operation:

- All of the backup data is copied to the primary database server.
- No other processes can connect to the database because the Authentication Manager Service is stopped.
- The backup data overwrites all existing data in the tables.
- The secrets are restored.

How you restore your database depends on your deployment configuration. For example, there are different procedures depending on where the database is located (same or separate machine than Authentication Manager). The procedures are also different in a replicated environment. Based on this, there are four possible scenarios for restoring a database:

- [Restoring a Database in an Environment without Replicas](#)
- [Restoring a Database on a Separate Machine in an Environment without Replicas](#)
- [Restoring a Database in an Environment with Replicas](#)
- [Restoring a Database on a Separate Machine in an Environment with Replicas](#)

For information on restoring an RSA RADIUS server, see “[Backing Up a RADIUS Server](#)” on page 189.

Note: The following procedures use the Manage Backups utility. For more information on the utility and its options, see “[Manage Backups Utility](#)” on page 271.

Restoring a Database in an Environment without Replicas

If you have a remote database, you must restore the data using the Manage Backups utility. Run the utility on the machine hosting the database.

To restore the database in an environment without replicas:

1. Stop all Authentication Manager Services on the primary instance, except for the internal database and the database listener.
2. Stop Authentication Manager on all server nodes, if applicable.
3. Open a new command shell, and change directories to *RSA_AM_HOME/utills*.
4. Remove the primary metadata. Type:

```
rsautil setup-replication -a remove-primary
```

Copy the two backup files, **filename.dmp** and **filename.secrets**, to the primary machine. Note their location.

5. Import the files into the database. Type:

```
rsautil manage-backups -a import -D -f absolute path
```

where *absolute path* is the absolute path and filename of the backup file, including the file extension. For example: C:\backup\filename.dmp.

Important: You must include the -D flag in order for the restore operation to work properly.

6. Reset the primary metadata. Type:

```
rsautil setup-replication -a set-primary
```

7. Start the Authentication Manager server.

Restoring a Database on a Separate Machine in an Environment without Replicas

If you have a remote database, you must restore the data using the Manage Backups utility. Run the utility on the machine hosting the database.

To restore a remote database in an environment without replicas:

1. Stop all Authentication Manager Services on the primary instance, except for the internal database and the database listener.
2. Stop Authentication Manager on all server nodes, if applicable.
3. On the machine hosting the database, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
4. Remove the primary metadata. Change directories to ***RSA_AM_HOME/utills***. Type:

```
rsautil setup-replication -a remove-primary
```

Copy the two backup files, **filename.dmp** and **filename.secrets**, to the machine that hosts the database server. Note their location.

5. Access the machine that hosts the database server so that you can import the files into the database. Change directories to ***RSA_AM_HOME/utills***. Type:

```
rsautil manage-backups -a import -D -f absolute path
```

where *absolute path* is the absolute path and filename of the backup file, including the file extension. For example: C:\backup\filename.dmp.

Important: You must include the -D flag in order for the restore operation to work properly.

6. Reset the primary metadata. Change directories to ***RSA_AM_HOME/utills***. Type:

```
rsautil setup-replication -a set-primary
```

7. Copy the ***RSA_AM_HOME/utils/etc/systemfields.properties*** file from the database server to a temporary location on the primary server.
8. On the primary server, import ***systemfields.properties***. On the primary server, change directories to ***RSA_AM_HOME/utils***. Type:


```
rsautil manage-secrets -a import -f absolute path
```

 where *absolute path* is the absolute path and filename to the temporary copy of ***systemfields.properties***.
9. When prompted for the import file password, enter the master password.

Restoring a Database in an Environment with Replicas

If you have a replicated environment and you experience a problem with the database on the primary instance, you can restore the internal database to a replica instance and promote that replica instance to a primary instance.

For example, if you accidentally delete a large amount of data from the database on the primary instance, you can restore the database to recover the lost data.

To restore the internal database in an environment with replicas, you must:

- Stop the replication process to prevent corrupting the database on each replica instance.
- Restore the database from the database backup files to a replica instance.
- Promote that replica instance to replace the stopped primary instance.
- Attach the remaining replica instances to the new primary instance.

Note: These instructions assume that the Authentication Manager internal database has been backed up and that the backup files are available.

To restore the database in an environment with replicas:

1. Stop Authentication Manager on all server nodes belonging to the primary instance, if applicable.
2. Stop all Authentication Manager Services on the primary instance.
3. Select the replica instance that will be the new primary instance.

Note: You do not need to choose the most recently updated replica instance, because all of the databases are synchronized to the backup.

4. On the replica that you are going to promote, stop all Authentication Manager services, except for the database and the database listener.
5. Copy the two backup files, ***filename.dmp*** and ***filename.secrets***, to the replica that is to be promoted.

6. On the replica that you are promoting to be the new primary, restore the database from the backup file on that instance:
 - a. Change directories to ***RSA_AM_HOME/utills***.
 - b. Type:

```
rsautil manage-backups -a import -f absolute path
```

where *absolute path* is the absolute path and filename of the backup file, including the file extension. For example: C:\backup\filename.dmp.
7. Start the Operations Console service on the replica that you are going to promote.
8. Log on to the Operations Console on the replica instance that you are going to promote to be the new primary instance.
9. Use the Operations Console to promote the replica. Promote the replica in disaster recovery mode.

For information and instructions, see the Operations Console Help topic “Promote Replica Instances.”
10. Use the Operations Console to attach the other replicas to the new primary instance. Use automatic synchronization.

For information and instructions, see the Operations Console Help topic “Attach Replica Instances.”
11. Restart the new primary instance.
12. Start Authentication Manager on all server nodes, if applicable.

Note: If you are restoring the database as part of a planned promotion, and you wish to reattach the newly demoted primary, see [“Reattaching a Demoted Primary Instance”](#) on page 232.

Restoring a Database on a Separate Machine in an Environment with Replicas

If the primary instance in a replicated environment stops working, you can restore the system by restoring the internal database to a replica instance and promoting that replica instance to a primary instance. To restore the internal database, you must:

- Stop the replication process to avoid corrupting the database at each replica instance.
- Restore the replica instance from the database backup.
- Promote a replica instance to replace the stopped primary instance.
- Attach the remaining replica instances to the new primary.

Note: These instructions assume that the Authentication Manager internal database has been backed up and that the backup files are available.

To restore a database on a separate machine in an environment with replicas:

1. Stop Authentication Manager on all server nodes belonging to the primary instance, if applicable.
2. Stop all Authentication Manager Services on the primary instance.
3. Select the replica instance that will become the new primary instance.

Note: You do not need to choose the most recently updated replica instance, because all of the databases are synchronized to the backup.

4. On the replica instance that will become the new primary instance, stop all Authentication Manager services, except for the database and database listener.
5. Copy the two backup files, **filename.dmp** and **filename.secrets**, to the machine that hosts the database server that is used by the replica that is to be promoted.
6. On the replica that you are promoting to be the new primary, restore the database from the backup file on that instance:
 - a. Change directories to **RSA_AM_HOME/utills**.
 - b. Type:


```
rsautil manage-backups -a import -f absolute path
where absolute path is the absolute path and filename of the backup file,
including the file extension. For example: C:\backup\filename.dmp.
```
7. Start the Operations Console on the replica that you are going to promote.
8. Log on to the Operation Console on the replica you are going to promote.
9. Use the Operations Console to promote the replica instance.

There are two ways to promote a replica, Recovery and Planned. Use the Recovery method in this scenario.

For information and instructions, see the Operations Console Help topic “Promote Replica Instances.”
10. Use the Operations Console to reattach replica instances. Select the replica that you want to attach, and select **Automatic** synchronization.

For information and instructions, see the Operations Console Help topic “Attach Replica Instances.”
11. At the end of the attach process, click **Done** to return to the Manage Existing Instances page.
12. Repeat this step for each replica instance. When you are finished, check the status of the attached replicas on the Manage Existing Instances page.
13. Restart the new primary instance.
14. Start Authentication Manager on all server nodes, if applicable.

Note: If you are restoring the database as part of a planned promotion, and you wish to reattach the newly demoted primary, see [“Reattaching a Demoted Primary Instance”](#) on page 232.

Restoring Event-Based Token Data

In the event of a disaster that involves the loss of event-based tokencode data, enable database recovery mode to allow users to resynchronize their event-based tokens with Authentication Manager at the first post-disaster authentication.

For more information, see the Security Console Help topic “Enable Database Recovery Mode.”

Restoring an Installation for a Standalone Primary Instance

This section describes how to restore an installation if the hardware fails for a primary instance without any replicas (standalone primary instance). In this configuration, the database can be on the same machine or on a different machine, and the instance can have server nodes.

If your deployment has a standalone primary instance (no replica instances), you must back up the database immediately after installing Authentication Manager. If the machine hosting the primary instance fails, use this backup to restore the database. Perform this backup periodically to ensure that a current version of the database is always available for disaster recovery. Store the backup in a safe location.

Important: If the database is on a separate machine and either the machine hosting Authentication Manager or the machine hosting the database fails, you must reinstall Authentication Manager on both machines.

Restoring the Installation

This procedure assumes the following:

- You have done a database backup.
- You have access to the backed up data and secret files.

The following procedure uses the Manage Backups utility. For more information on the utility and its options, see “[Manage Backups Utility](#)” on page 271.

To restore the installation from a backup after a hardware failure:

1. Install Authentication Manager on another machine.
2. On the machine that hosts the application server, stop all Authentication Manager Services, except for the internal database and the database listener.
3. Stop Authentication Manager on all server nodes, if applicable.

4. Remove the primary metadata.

Important: If your deployment has a remote database, perform this step on the machine hosting the database.

Change directories to ***RSA_AM_HOME/utills***. Type:

```
rsautil setup-replication -a remove-primary
```

Copy the two backup files, **filename.dmp** and **filename.secrets**, to the machine that hosts the database server. Note their location.

5. Access the machine that hosts the database server so that you can import the files into the database.

Important: If your deployment has a remote database, perform this step on the machine hosting the database.

Change directories to ***RSA_AM_HOME/utills***. Type:

```
rsautil manage-backups -a import -D -f absolute path
```

where *absolute path* is the absolute path and filename of the backup file, including the file extension. For example: C:\backup\filename.dmp.

Important: You must include the -D flag in order for the restore operation to work properly.

6. Reset the primary metadata.

Important: If your deployment has a remote database, perform this step on the machine hosting the database.

Change directories to ***RSA_AM_HOME/utills***. Type:

```
rsautil setup-replication -a set-primary
```

Note: If your deployment includes server nodes, you must uninstall and then reinstall Authentication Manager on the nodes. You can do this before or after you restore the primary instance.

Detecting a Failed Primary Instance or Replica Instance

Use the Operations Console to view the replication status for each instance in the deployment, and to detect failures.

When an instance fails, you might also notice:

- The instance is not generating any logs.
- A Help Desk resolution does not work. For example, you might clear a user's PIN but the user still cannot authenticate.

Note: The log messages indicate if the database connection between the primary instance and the replica instances is disconnected, and if the replication system is functioning as it should. You can request a status report for additional information.

If you shut down the replica instance, you must wait three minutes before restarting. If you restart immediately and the process takes less than three minutes, you might not see any messages in the System Activity Monitor. However, you can run a report and see the events.

Use the Operations Console to generate the replication status report. For information and instructions, see the Operations Console Help topic "Check Replication Status."

If an instance has failed, you need to find out why. See the following section, "[Determining Why an Instance Might Stop Responding](#)."

Determining Why an Instance Might Stop Responding

Determine why the instance failed so that you can take appropriate action. A primary instance or replica instance might fail for any of the following reasons:

Power failure. In this case, you know that the instance can be restarted when power is restored. You probably do not need to take further action. When the primary instance is restarted, Authentication Manager synchronizes the databases.

Hardware failure. Estimate the level of effort required to replace the hardware component. Take into account the fact that you might have to reinstall Authentication Manager for all server nodes in the instance.

Database corruption. If runtime or console error messages indicate that the database is corrupted, you can assume that all databases in the deployment are also corrupted. You must restore the primary instance database from a previous backup using the Manage Backups utility.

Inoperable database. A database is inoperable if it runs out of disk space, or if its configuration prevents Authentication Manager from accessing it. If the surviving replica instance appears to function normally, you need to promote the replica instance to a primary instance. For instructions, see [“Promoting a Replica Instance to a Primary Instance”](#) on page 221.

Network failure. When Wide Area Network (WAN) connectivity is lost between a replica instance and a primary instance, the disconnected replica instance can neither send nor receive database updates. Transactions sent to the replica instance quickly queue up, consuming valuable disk space. Consider removing the replica instance from the system. After removal, agents update their contact lists and stop routing transactions to the disconnected server node. You can add another replica instance to improve performance.

What To Do When a Primary Instance Stops Responding

When deciding whether to recover the failed primary instance, consider these issues:

- How long do you estimate the primary instance will be unavailable?
- Can the problem be fixed?
- Does the surviving replica instance have enough disk space to handle transactions that will queue up while the primary instance is down?

At the site where the primary instance is located, trained and authorized personnel must decide whether it is worthwhile to fix the problem that caused the instance failure, or to promote a replica instance to take over for the failed primary instance.

If you cannot recover the primary instance, see [“Promoting a Replica Instance to a Primary Instance”](#) on page 221.

What To Do When a Replica Instance Stops Responding

If a failed replica instance cannot be recovered, remove the failed replica instance from the system and install a new replica instance as soon as possible to ensure failover and scalability. For instructions, see the chapter “Installing a Replica” in the *Installation and Configuration Guide*.

Important: Agents route authentication requests to specific replica instances based on information defined in an automatic contact list, manual contact list, or an **sdconf.rec** file. When you configure the agents, make sure they point to multiple replica instances. If an agent is configured to use just one replica instance and that instance fails, transactions intended for the failed replica instance queue up and consume valuable disk space.

Promoting a Replica Instance to a Primary Instance

You promote a replica instance to a primary instance in one of these situations:

- To recover from a disaster (crash)
- To migrate the primary instance to another machine

For information on promoting a RADIUS server, see [“Promoting an RSA RADIUS Replica Server”](#) on page 191.

Promoting a Replica Instance to Recover From a Disaster

In a disaster recovery scenario, the primary instance has crashed and cannot be recovered within a reasonable amount of time, affecting the ability to perform administrative operations. A replica instance must be promoted to a primary instance. Because the original primary instance is not accessible, the logs and any changes that have not yet been propagated to the replica instances cannot be recovered.

For procedures for this scenario, see [“Step 1: Identify the Replica Instance to Be Promoted”](#) on page 222.

Promoting a Replica Instance to Migrate the Primary Instance

In a migration scenario, the primary instance is healthy and the system is operating normally. However, due to performance or other considerations, you choose to elect a new primary instance. Because the existing primary instance is accessible, you must obtain a full backup of the primary instance, and restore it on the target replica instance.

For procedures for this scenario, see [“Step 1: Identify the Replica Instance to Be Promoted”](#) on page 222.

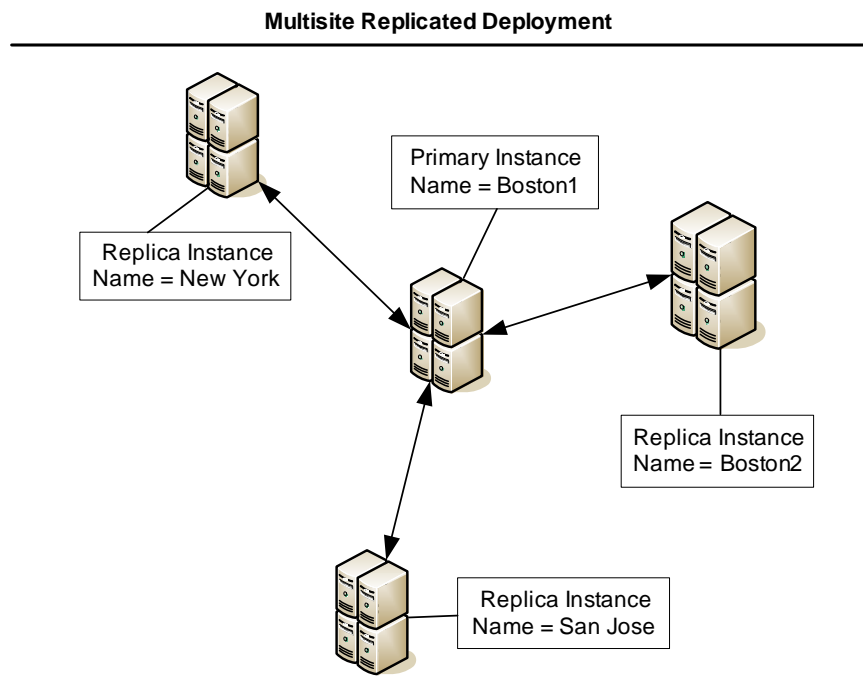
What Happens During Replica Instance Promotion

Promoting a replica instance changes the whole replication topology. After you promote a replica instance, the following occurs:

- The replica instance becomes the new primary instance.
- The original primary instance is automatically detached from all replica instances in the deployment. You must manually reattach each replica, one by one, to the new primary instance. For instructions, see [“Step 3: Reattach all Replica Instances to the New Primary Instance”](#) on page 228.
- All configuration data referring to the original primary instance is removed from all replica instances in the deployment.

Note: During the promotion process, you cannot perform any administrative functions, such as adding or deleting users.

This section describes promotion using a sample multisite replicated deployment containing one primary instance and three replica instances, as shown in the following figure.



Step 1: Identify the Replica Instance to Be Promoted

Identify which replica instance you are promoting to be the new primary instance. Consider these factors:

- How much hardware do you need for satisfactory load and performance?
- Which replica instance was updated most recently?

Evaluating Hardware for Load and Performance

After promotion, the new primary instance performs all administrative functions, such as accumulating all logs from the replica instances. Consider this issue when you review the memory and disk capacity of the machine that hosts the replica instance being promoted. Also think about whether you want to add server nodes to reduce the workload on the new primary instance.

Selecting the Most Current Replica Instance

If you cannot recover the primary instance, consider promoting the replica instance that was most recently updated when the primary instance was available.

To display the status of each replica instance:

Use the Operations Console to the replication status report. For information and instructions, see the Operations Console Help topic "Check Replication Status."

Step 2: Promote the Selected Replica Instance

When you promote a replica instance to a primary instance, the following occurs:

- The original primary instance is detached from all replica instances.
- All references to the original primary instance, including the original configuration data of the primary instance, are removed at each replica instance.
- The selected replica instance is configured as the new primary instance.

After you complete this step, all replica instances are fully functional and capture all updates made locally at each instance. However, any updates made at either the replica instances or the new primary instance are not propagated because the instances are not yet connected.

After promotion, you can add server nodes to the new primary to reduce the workload on the primary server. For information on adding server nodes to a primary or replica, see the chapter “Installing a Server Node” in the *Installation and Configuration Guide*.

Note: After promotion, the new primary instance is configured to use the same identity sources that it used as a replica instance. Consider whether you need to reconfigure the new primary instance to use different identity sources. The identity sources that the replica instance used might offer lower capacity or performance than the new primary instance needs, because replicas do not send a large number of changes to an identity source. Consider using the same identity source that the original primary instance used, which may have better capacity to handle changes.

To promote a replica instance to a primary instance in the disaster recovery scenario:

1. On the replica being promoted, stop all Authentication Manager Services, except for the internal database, the database listener, and the Operations Console.
2. Stop Authentication Manager on all server nodes, if applicable.
3. Log on to the Operation Console of the replica that you are going to promote.

4. Use the Operations Console to promote the replica instance.
There are two ways to promote a replica, Recovery and Planned. Use the Recovery method in this scenario.
For information and instructions, see the Operations Console Help topic “Promote Replica Instances.”

Important: During promotion, the original primary is not accessible, but it remains configured to connect to the other instances in the deployment. If the original primary instance is restarted after a new primary instance has been elected, the original primary instance attempts to connect to the other instances. Although this does not cause failures in the deployment, RSA strongly recommends one of the following options: stop or uninstall the database server on the original primary (see the *Installation and Configuration Guide* for instructions), or attach the original primary as a replica (see [“Reattaching a Demoted Primary Instance”](#) on page 232).

5. Optional. If your deployment includes RADIUS, access the RADIUS primary server and change directories to `RSA_AM_HOME/radiusoc/utills`: Type:


```
rsautil manage-secrets -a set com.rsa.radius.oc.cert.cn.1
<cn>
```

where `cn` is the fully qualified hostname of the server that you just promoted.
6. Optional. If your deployment includes RADIUS, repeat [step 5](#) on each RADIUS server in your deployment.
7. Restart the application server (Security Console).

If the original primary database server is accessible, perform the following procedure to restore the audit log entries from the original primary database server to the new primary database server.

Note: The following procedures use the Manage Backups utility. For more information on the utility and its options, see [“Manage Backups Utility”](#) on page 271.

To restore the audit log entries from the original primary database server to the new primary database server:

1. On the original primary database server, stop the replication service.
2. Change directories to `RSA_AM_HOME/utills` and use the Setup Replication utility to clean up the site. Type:

```
rsautil setup-replication --action cleanup-site
```


3. Back up the log files only.

Important: This step is done differently depending on where your database is hosted, on the same machine as Authentication Manager or on a separate machine.

- If your database is on the same machine as Authentication Manager, log on to the Operation Console for the “old” primary and back up the log files only. For information and instructions, see the Operations Console Help topic “Back Up Your Database.”
- If the database is hosted on a different machine, use the Manage Backups utility to back up the log files.

- On the machine hosting the database, open a new command shell and change directories to ***RSA_AM_HOME/utills***.

- Type:

```
rsautil manage-backups --action export -g  
-f absolute path
```

where *absolute path* is the absolute path and filename of the backup file, including the file extension. For example: C:\backup\filename.dmp.

4. Copy the backup file generated in [step 3](#) to the new primary database server.
5. Restore the log files only.

Important: This step is done differently depending on where your database is hosted, on the same machine as Authentication Manager or on a separate machine.

- If your database is on the same machine as Authentication Manager, log on to the Operation Console for the “new” primary and restore the log files only. For information and instructions, see the Operations Console Help topic “Restore Your Database from a Backup File.”
- If the database is hosted on a different machine, use the Manage Backups utility to restore the log files.

- On the machine hosting the database, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.

- Type:

```
rsautil manage-backups --action import -g  
-f absolute path
```

where *absolute path* is the absolute path and filename of the backup file, including the file extension. For example: C:\backup\filename.dmp.

6. Restart the Authentication Manager server at the machine that hosts the new primary instance.
7. Update the contact lists for the new primary instance. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.
8. Click **Rebalance**.

To promote a replica instance to a primary instance in the planned migration scenario:

1. On the original primary instance, stop all Authentication Manager Services, except for the internal database, the database listener, and the Operations Console.
2. On the replica instance being promoted, stop all Authentication Manager Services, except for the internal database, the database listener, and the Operations Console.
3. Stop Authentication Manager on all server nodes, if applicable.
4. Log on to the Operations Console for the instance you want to promote.
5. Use the Operations Console to promote the replica instance. There are two ways to promote a replica, Recovery and Planned. Use the Planned method in this scenario.

For information and instructions, see the Operations Console Help topic “Promote Replica Instances.”

6. Back up the database.

Important: Do not skip this step. You must create a new backup file before proceeding. Do not use an old backup file.

Important: This step is done differently depending on where your database is hosted, on the same machine as Authentication Manager or on a separate machine.

- If your database is on the same machine as Authentication Manager, log on to the Operation Console on the “old” primary and perform the back up. For information and instructions, see the Operations Console Help topic “Back Up Your Database.”
- If the database is hosted on a different machine, use the Manage Backups utility to do the back up.
 - On the machine hosting the database, open a new command shell and change directories to ***RSA_AM_HOME*/utils**.
 - Type:


```
rsautil manage-backups --action export
-f absolute path
```

where *absolute path* is the absolute path and filename of the backup file, including the file extension. For example: C:\backup\filename.dmp.

7. Stop the Operations Console on the former primary instance.
8. Copy the generated backup file from the directory location you specified to the new primary instance.
9. Restore the database.

Important: Do not skip this step. You must restore the database before reattaching your replicas in the following section, "[Step 3: Reattach all Replica Instances to the New Primary Instance.](#)"

Important: This step is done differently depending on where your database is hosted, on the same machine as Authentication Manager or on a separate machine.

- If your database is on the same machine as Authentication Manager, log on to the Operation Console on the "new" primary and restore the database. For information and instructions, see the Operations Console Help topic "Restore Your Database from a Backup File."
- If the database is hosted on a different machine, use the Manage Backups utility to restore the database.
 - On the machine hosting the database, open a new command shell and change directories to ***RSA_AM_HOME/utills***.

- Type:

```
rsutil manage-backups --action import
-f absolute path
```

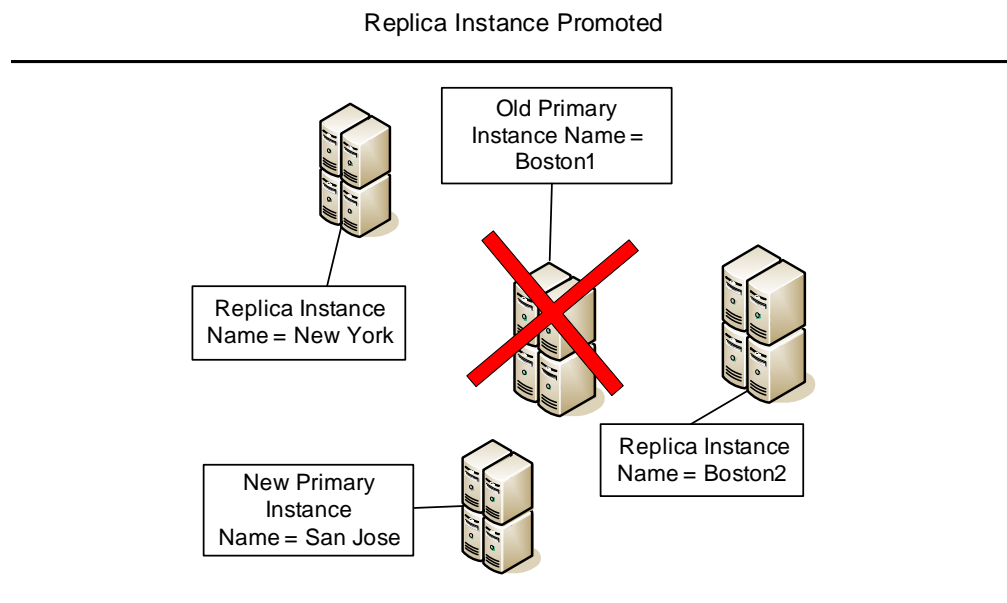
where *absolute path* is the absolute path and filename of the backup file, including the file extension. For example: C:\backup\filename.dmp.

10. On the new primary instance, start the application server (Security Console).
11. Optional. If your deployment includes RADIUS, access the RADIUS primary server and change directories to ***RSA_AM_HOME/radiusoc/utills***: Type:

```
rsutil manage-secrets -a set com.rsa.radius.oc.cert.cn.1
<cn>
```

where *cn* is the fully qualified hostname of the server that you just promoted.
12. Optional. If your deployment includes RADIUS, repeat [step 11](#) on each RADIUS server in your deployment.

The following figure shows the sample deployment after the replica instance is promoted.



Step 3: Reattach all Replica Instances to the New Primary Instance

In this step, you need to:

- Reattach all remaining replica instances to the new primary instance.

Note: A replica instance cannot perform any functions while you are reattaching it. The remaining replica instances continue to function normally.

- Resynchronize data for all instances to ensure identical data at each instance.

This step accomplishes the following tasks:

- The new primary instance is configured to receive data from the replica instance.
- Any changes that occurred on the replica instance since it was detached are propagated to the new primary.
- The replica instance is reinitialized with data from the new primary instance, including the changes that the primary instance just received from the replica instance. You can perform this data synchronization step either automatically or manually.
- The replica instance is configured to receive data from the primary instance.

Note: If you want to add the original primary instance as a replica instance after the promotion, see [“Reattaching a Demoted Primary Instance”](#) on page 232.

Synchronizing Data Between the Replica Instance and the Primary Instance

When you reattach a replica instance, the replica database server must be synchronized with the primary database server. Synchronization ensures that both instances contain the same data. You can perform data synchronization automatically or manually.

Automatic synchronization. This method accomplishes data synchronization by establishing the network connection between the two instances and transferring the data over the network. You can perform this task in one step using the Manage Replication utility. This method automatically generates a new schema data file containing a snapshot of data in the primary instance database server.

Manual synchronization. This method requires you to use the Manage Replication utility to manually generate a schema data file containing a snapshot of data in the primary instance database server. You copy this file to the tablespace directory of the replica instance. The utility initializes the replica instance using the schema data file located in the tablespace directory of the replica instance.

Important: Each time you synchronize manually, you must generate a replication data file. See [“Reattach a Replica Instance Using Manual Synchronization”](#) on page 230. The replication data file holds the schema object for administrative and runtime data such as tables. Do not use the same replication data file for multiple replica instances. Each new replica instance copies instance-specific data to the primary instance at the time of attachment. Using a previously used schema data file causes errors in replication.

Reattach a Replica Instance Using Automatic Synchronization

To reattach a replica instance using automatic synchronization:

1. On the primary instance, stop all Authentication Manager services except for the database, the database listener, and the Operations Console.
2. On the replica instance that you are planning to reattach, stop all Authentication Manager services except for the database, the database listener, and the Operations Console.
3. Log on to the Operations Console on the primary instance.
4. Use the Operations Console to reattach replica instances. Select the replica that you want to attach, and select **Automatic** synchronization.
For information and instructions, see the Operations Console Help topic “Attach Replica Instances.”
5. At the end of the attach process, click **Done** to return to the Manage Existing Instances page.
6. Repeat this step for each replica instance. When you are finished, check the status of the attached replicas on the Manage Existing Instances page.

Reattach a Replica Instance Using Manual Synchronization

To reattach a replica instance using manual synchronization:

1. On the primary instance, stop the application server (Security Console).
2. On the replica instance that you are planning to reattach, stop the application server (Security Console).
3. Log on to the Operations Console on the primary instance.
4. Use the Operations Console to reattach replica instances. Select the replica that you want to attach, and select **Manual** synchronization to generate the replication data file.

For information and instructions, see the Operations Console Help topic “Attach Replica Instances.”

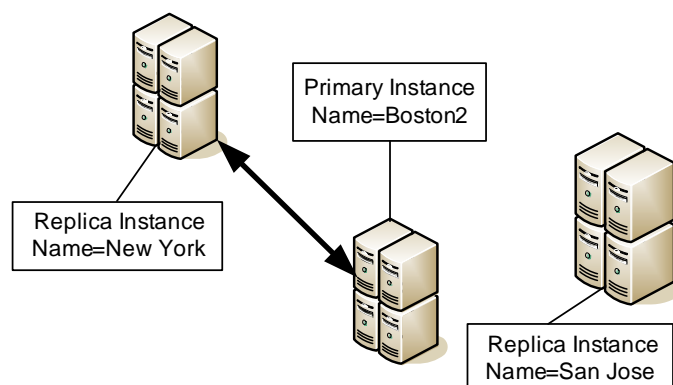
5. At the end of the attach process, click **Done** to return to the Manage Existing Instances page.
6. Copy the replication data file (.dmp) to the replica instance tablespace directory in `..\RSAAM\db\oradata\`.

Important: For Linux systems, make sure that the schema data file is copied as belonging to the user who installed Authentication Manager.

7. Click **Complete the Manual Attach Process**. Select the replica to finish the attach process.
8. At the end of the manual attach process, click **Done** to return to the Manage Existing Instances page.
9. Repeat steps 2-6 for each replica instance.

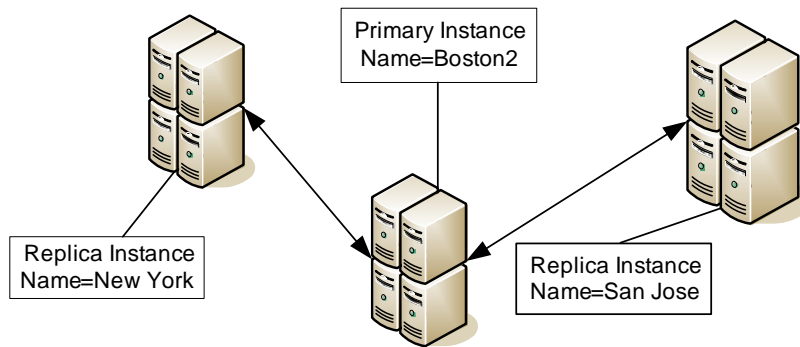
The following figure shows the topology of the sample multisite deployment.

Multisite Replicated Deployment
 Replica New York Attached to New Primary Boston2



The following figure shows the sample deployment topology after all replica instances have been reattached.

Multisite Replicated Deployment
All Replicas Are Reattached



Reconfiguring CT-KIP After Promoting a Replica

When you configured Authentication Manager, you likely configured the following two URLs necessary for Remote Token-Key Generation (CT-KIP):

Token-Key Generation. This is the URL to invoke the CT-KIP application that interacts with the Authentication Manager CT-KIP server for remote key generation.

Service Address. This is the server-side address of the CT-KIP service.

After you promote a replica instance to a primary instance, you must use the Security Console to modify these URLs to point to the new primary instance. For instructions, see the Security Console Help topic “Configuring RSA Authentication Manager.”

Removing a Replica Instance

Use the Operations Console to remove a replica instance from the deployment. When you remove a replica instance, the utility disconnects the instance from the deployment and removes its configuration data from all remaining instances in the deployment.

To remove a replica instance:

Use the Operations Console to remove the replica instances. For information and instructions, see the Operations Console Help topic “Delete Replica Instances.”

Reattaching a Demoted Primary Instance

After you demote a primary instance because of network connectivity or other problems, you can reattach the demoted instance to the new primary instance once the problems are corrected, making the demoted instance a replica instance. To attach a demoted primary instance, you must perform the following high-level steps:

1. Initiate the attachment process. Use the Operations Console to start this process.
2. Clean the instance. This is done automatically as part of the attachment process, and is necessary only if the demoted primary was demoted as part of a disaster recovery process, and configuration information still exists on the instance.
3. Manually copy the file ***RSA_AM_HOME/utils/etc/primary.properties*** from the current primary instance to the instance you want to attach.
4. After copying the file, finish the attach process.

To reattach a demoted primary instance:

1. On the demoted primary, stop all Authentication Manager services except for the database, the database listener, and the Operations Console.
2. Log on to the Operation Console on the demoted primary instance.
3. Use the Operations Console to reattach the demoted primary instance.
For information and instructions, see the Operations Console Help topic “Attach Demoted Primary Instances.”
4. Start Authentication Manager.

Resynchronizing a Diverged Replica Instance

Use the Operations Console to resynchronize a replica instance with a primary instance. Normally, the data between the primary instance and the replica instance is synchronized automatically. However, under certain circumstances, you must resynchronize the replica instance manually. For example, you need to resynchronize if the replica instance is disconnected from the deployment due to network loss or hardware issues. You should resynchronize the replica whenever it is disconnected from the deployment for more than seven days.

Important: Automatic synchronization can be done from the Operations Console on any instance. Manual synchronization can only be done when logged on to the primary instance's Operations Console.

To resynchronize a replica instance using automatic synchronization:

1. On the replica instance, stop all Authentication Manager services except for the database, the database listener, and the Operations Console.
2. Log on to the Operations Console on the primary instance.
3. Use the Operations Console to resynchronize the replica instance. Select the replica to resynchronize, and select **Automatic** synchronization.
For information and instructions, see the Operations Console Help topic "Resynchronize Replica Instances."
4. At the end of the resynchronize process, click **Done** to return to the Manage Existing Instances page.

To resynchronize a replica instance using manual synchronization:

1. On the replica instance, stop all Authentication Manager services except for the database, the database listener, and the Operations Console.
2. Log on to the Operations Console on the primary instance.
3. Use the Operations Console to resynchronize the replica instance. Select the replica to resynchronize, and select **Manual** synchronization.
For information and instructions, see the Operations Console Help topic "Attach Replica Instances."
4. At the end of the resynchronization process, click **Done** to return to the Manage Existing Instances page.
5. Copy the replication data file (.dmp) to the replica instance tablespace directory in `..\RSAAM\db\oradata\`.

Important: For Linux systems, make sure the schema data file is copied as belonging to the user who installed Authentication Manager.

6. Click **Complete the Manual Resynchronization Process**. Select the replica to finish the resynchronization process.
7. At the end of the manual resynchronization process, click **Done** to return to the Manage Existing Instances page.
8. Repeat steps 2-6 for each replica instance.

Important: If you use manual data synchronization, make sure that you generate a new schema data file for each replica instance.

Restoring a Super Admin

The Super Admin is created during installation. If the Super Admin (that is, a user assigned the Super Admin role) is deleted, use the Super Admin Restoration utility, `restore-admin`, to create a new Super Admin. The Super Admin role is a predefined administrative role and the only role with full administrative permission in all realms and security domains. A Super Admin can:

- Delegate roles to other administrators.
- Create the realm and security domain hierarchy.

Because certain tasks can be performed only by a Super Admin, your deployment must have at least one Super Admin. If no one is assigned to this role, use this utility to assign the role to a registered user.

Important: Because of the Super Admin role's broad scope and wide range of permissions, RSA recommends that you only assign it to the most trusted administrators.

Important: Do not run this utility on a replica instance.

When You Need to Restore the Super Admin

You need to restore the Super Admin if any of the following conditions exist:

- The sole Super Admin has been deleted from the system.
- No registered users have been assigned the Super Admin role.
- The sole Super Admin has been locked out.

Recovering from a Lockout

If the Super Admin has been locked out, recovery can occur in any of the following ways:

- Another Super Admin can manually unlock the Super Admin.
- If the lockout policy that applies to the Super Admin is configured for auto-unlock, you can wait for lockout to expire.
- If the previous methods fail, use the Super Admin Restoration utility, described in the following section, "[Using the Super Admin Restoration Utility](#)."

Using the Super Admin Restoration Utility

Prerequisite

Before you run the Super Admin Restoration utility, make sure that you know the master password of the encrypted properties file (**systemfields.properties**).

To use `restore-admin`:

1. On the primary instance, open a new command shell, and change directories to `RSA_AM_HOME/utils`.

2. Type:

```
rsautl restore-admin options
```

For relevant options, see the following section, "[Options for restore-admin.](#)"

Important: Although it is possible to enter the master password on the command line along with the other options, this creates a potential security vulnerability. RSA recommends that you enter the master password only when the utility presents a prompt.

For example:

```
rsautl restore-admin --admin newadmin  
--password adminpassword
```

where:

- `newadmin` is the user ID of the Super Admin you are creating.
- `adminpassword` is the password for the new Super Admin you are creating.

Note: The Super Admin Restoration utility is intended to restore access to the system in an emergency. By default, the time to live for the Super Admin you create with this utility is 24 hours. The password you provide when creating this Super Admin is not validated by the default password policy. RSA recommends that you create a password that conforms to the default password policy when you use this utility.

Note: The Super Admin Restoration utility also resets the Operations Console password policy to LDAP_Password/RSA Password. In order for this change to take effect, use the Operations Console to flush the cache. On the Operations Console, click **Maintenance > Flush Cache**, and click **Flush** to flush all cache objects.

Options for restore-admin

The following table describes the options for this utility.

Flag	Alternate Flag	Description
-h	--help	Displays help for this utility.
-m	--master-password	Master password of the encrypted properties file.
-p	--password	Password for the Super Admin you are creating.
-u	--admin	User ID of the Super Admin you are creating.
-V	--verbose	Displays more output messages.
-v	--version	Displays the version and copyright information.

A

Integrating Active Directory Forests

- [Overview of Active Directory Forest Identity Sources](#)
- [Adding an Active Directory Forest as an Identity Source](#)

Overview of Active Directory Forest Identity Sources

Configuring RSA Authentication Manager to access user and group data from an Active Directory forest entails some additional considerations and procedures.

Runtime and Administrative Identity Sources

To account for the architecture of an Active Directory forest, this section refers to two distinct types of identity sources:

Runtime identity source. An identity source configured for runtime operations only, to find and authenticate users, and to resolve group membership within the forest. If you use Active Directory Global Catalogs, you can configure the Global Catalogs to be runtime, but not administrative, identity sources.

Note: Global Catalogs are not a requirement for Authentication Manager, however if you plan to use Global Catalogs, see the following section, [“Requirements for Using Global Catalogs.”](#)

Administrative identity source. An identity source used for administrative operations such as adding users and groups. This identity source maps to a domain controller.

In a multidomain Active Directory forest setup, the Global Catalog is added as a runtime identity source and the domain controller servers are added as administrative identity sources. The Global Catalog is used at runtime as another directory to find and authenticate users, and to resolve group membership within the forest.

The Global Catalog is used only for runtime operations, such as authentication. Authentication Manager does not use the Global Catalog for administrative operations. Administrative actions (for example, adding users) are performed against the administrative identity sources (domain) only. Changes to the domain are replicated by Active Directory to the Global Catalog.

For example, suppose GC1 is the Global Catalog that you want to use as your identity source, and AD1, AD2, and AD3 replicate a subset of their data to GC1. You must perform the procedure on each of the identity sources.

After you perform the integration, Authentication Manager accesses GC1 for authentication requests only. Authentication Manager accesses AD1, AD2, and AD3 for all other administrative operations. If you grant Authentication Manager read/write access to your identity sources, Authentication Manager makes all administrative changes in AD1, AD2, and AD3, which replicate the changes to GC1.

Note: Active Directory supports multiple types of groups. When configured to use Active Directory groups, Authentication Manager only supports Universal groups. When you view the Active Directory groups from the RSA Security Console, the Security Console displays all groups, regardless of type. If you select a group from this list to activate users on restricted agents, make sure you select a Universal group. Use the Active Directory Users and Computers MMC Console to examine the type of group. If you use any other type of Active Directory group, the user cannot authenticate.

Requirements for Using Global Catalogs

Important: The following requirements only apply to deployments that use restricted authentication agents and user groups.

To use Global Catalog as a runtime identity source, you must meet all of the following requirements:

- All users must be in Windows universal groups.
- To use universal groups, the entire forest must be composed of Windows 2003 domain controllers, running in Windows 2003 native mode.
- If the Active Directory is read-only in Authentication Manager, only the Windows administrator can change the group type in Active Directory.
If the Active Directory is read-write in Authentication Manager, you can use the Security Console to change the group type.

Adding an Active Directory Forest as an Identity Source

When you use the Active Directory Global Catalog as your authoritative identity source, you must integrate the following with Authentication Manager:

- The Global Catalog.

Note: If a forest has more than one Global Catalog, you can use one for failover. In this case, you do not need to deploy the Global Catalog, but you must specify it as a failover URL when you deploy the first Global Catalog.

- All Active Directories that replicate data to the Global Catalog. Each additional Active Directory must be added as an administrative identity source.

For example, for a hypothetical forest composed of three domains and a single Global Catalog, you must enable four identity sources:

- 3 administrative identity sources
- 1 runtime source (Global Catalog)

The following steps provide a high-level overview of the tasks that you must perform to add an identity source:

1. Use the RSA Operations Console to set up the SSL connections. This step is required if your identity source is Active Directory with read-write access. This step is optional for Active Directory with read-only access, and Sun Java System Directory Server.
See [“Setting Up SSL for LDAP”](#) on page 23.
2. Consider the following before integrating Active Directory forests:
 - See [“Password Policy Considerations”](#) on page 240.
 - Verify your domain functional level to support group to group membership and verify your group container as described in [“Supporting Groups”](#) on page 240.
 - The Active Directory server name must be a valid DNS name. Make sure the name is resolvable for both forwards and reverse lookups, and that the Active Directory server can be reached from the Authentication Manager server.
3. Optional. Use the Security Console to create custom user attributes. Custom user attributes contain information tailored to your organization. Create custom attributes required by other applications within your organization.
For example, if you have applications that require the UPN field in Active Directory, use the Add New Identity Attribute Definition page in the Security Console to manually create this field in Authentication Manager.
See [“Adding Custom Attributes to User Records”](#) on page 23.
4. Use the Operations Console to add the identity source. When you add the identity source, configure the following:
 - General identity source information.
 - LDAP connection.
 - Global Catalog (Active Directory only).
 - Map user attributes. This can include standard or custom user attributes. There are special considerations when mapping attributes for Active Directory. See [“Mapping Attributes to Active Directory”](#) on page 240 for more information.
For more information on adding custom attributes, see [“Adding Custom Attributes to User Records”](#) on page 23.
5. Use the Security Console to link the identity source to a realm so that you can assign tokens to the users stored in the identity source.
See [“Linking an Identity Source to a Realm”](#) on page 28.
6. Use the Security Console to verify the identity source.
See [“Verifying the LDAP Identity Source”](#) on page 29.

Password Policy Considerations

Active Directory has a default password policy that is more strict than the default Authentication Manager password policy. This can lead to errors such as “Will Not Perform” when adding and updating users.

To manage password policies with Active Directory identity sources, do one of the following:

- Make your Authentication Manager password policy password requirements more strict. See the chapter “Preparing RSA Authentication Manager for Administration,” in the *Administrator's Guide*.
- Relax the complexity requirements in the Windows 2003 Group Policy Editor. See your Windows documentation.

Supporting Groups

Setting the Domain Level for Group-to-Group Membership

To support group-to-group membership in Active Directory, you must set the domain functional level to Windows 2003. For more information about how to raise the domain functional level, go to

<http://www.microsoft.com/technet/prodtechnol/windowsserver2003/library/ServerHelp/5084a49d-20bd-43f0-815d-88052c9e2d46.mspx>.

Specifying a Group Container in the RSA Security Console

The default organizational unit “Groups” does not exist in the default Active Directory installation. Make sure that a valid container is specified for the Group Base DN when adding the identity source.

Mapping Attributes to Active Directory

When you use the Operations Console to add your identity source, you map identity attribute definitions. Mapping the fields in your identity source to the fields in the Security Console allows you to use the Security Console to view user and user group data stored in your identity source.

You must follow specific guidelines when you use the Security Console to map identity attributes to physical attribute names in an Active Directory identity source schema. You use the Add Identity Source page to map attributes.

If your Active Directory identity source is read-only (the default), make sure that all user fields map to non-null fields. The User ID is mapped to the samAccountName by default, but it can be mapped to any unique attribute for a user.

If your Active Directory identity source is read/write, make sure that you map all of the fields that you need when you add the identity source using the Add Identity Source page in the Security Console. If you do not map a field, the field will remain blank when you add users. Active Directory does not provide any values for user's records for unmapped fields. There are some cases where it is necessary to create custom attributes in order to ensure that these fields are populated correctly.

For example, when adding the identity source, if you mapped User ID to either the samAccountName or userPrincipalName attribute, you must perform one of the following procedures to make sure that both fields contain the correct data.

If you mapped User ID to samAccountName:

1. In the Security Console, click **Identity > Identity Attribute Definitions > Add New**.
2. Create a custom identity attribute to map to **userPrincipalName**.

Important: This attribute must be required. In the **Entry Type** field, select **Required**.

For complete instructions, see the Security Console Help topic "Add Identity Attribute Definitions."

3. Map the new custom attribute to **userPrincipalName**.
For complete instructions, see the Security Console Help topic "Configure Identity Source Attribute Mappings."

If you mapped User ID to userPrincipalName:

1. In the Security Console, click **Identity > Identity Attribute Definitions > Add New**.
2. Create a custom identity attribute to map to **samAccountName**.

Important: This attribute must be required. In the **Entry Type** field, select **Required**.

For complete instructions, see the Security Console Help topic "Add Identity Attribute Definitions."

3. Map the new custom attribute to **samAccountName**.
For complete instructions, see the Security Console Help topic "Configure Identity Source Attribute Mappings."

B

Customizing RSA Credential Manager

- [Customizing E-mail Notifications](#)
- [Customizing Help for the RSA Self-Service Console](#)
- [Customizing Token Graphics](#)
- [Customizing Workflow and Non-Workflow Operations](#)
- [Customizing the Self-Service Console Home Page](#)

Customizing E-mail Notifications

If your RSA Authentication Manager installation includes RSA Credential Manager provisioning, you can customize e-mail notifications using the e-mail templates and e-mail template tags. For a complete list of the e-mail template tags, see “[Using E-mail Template Tags](#)” on page 247.

To customize e-mail templates:

1. Click **Setup > Component Configuration > Credential Manager**.
2. Click **Define e-mail settings**.
3. Optional. In the **E-Mail Notification Templates** section, for each type of template, select **View & manage template details** to edit the e-mail template. You can change the e-mail template tags to customize information in the e-mail. You cannot use the characters (<) and (>) in e-mail templates.

To use conditional statements in your e-mail templates, you must follow the Velocity syntax rules. Velocity is a Java-based template engine that is part of the Apache Velocity Project. For information about Velocity syntax for conditional statements, go to <http://velocity.apache.org>.

The following examples show how to use template tags in conditional statements:

String Example

```
#if( ${Principal.FirstName} = "John" )  
do some thing  
#else  
do nothing  
#end
```

Boolean Example

```
#if( ${Principal.isEnabled} )  
do some thing  
#end
```

Integer Example

```
#if( ${ UCMRequest.Status } < 5 )
do some thing
#else
do nothing
#end
```

4. Click **Save**.

Guidelines for Customizing E-mail

Use the following guidelines when customizing e-mails:

- Include important e-mail recommendations before the first link in e-mails.
- Use consistent and correct terminology in e-mails, and on virtual public networks (VPN).

Example of Customized E-mail Template

The following example shows the default e-mail template for an approved request for a hardware token with customized changes indented.

Customized E-mail Template

Request Approval E-mail Template – Hardware Token

Subject: \${MailComposer.RequestType} is approved.

Your \${MailComposer.RequestType} is approved.

Administrator Comments: \${MailComposer.ApprovalDetails}

Request Details

Requested by: \${Principal.FirstName} \${Principal.LastName} [\${Principal.UserID}]

Confirmation #: \${UCMRequest.RequestID}

Approval Date: \${MailComposer.ApprovalDate}

Token Details

Type: \${MailComposer.TokenType}

Token Enablement Details

Link: \${MailComposer.EnablementURL}

Code: \${MailComposer.EnablementCode}

Serial Number: \${MailComposer.SerialNumber}

(If the Serial Number field is empty, the Serial Number field will not be in the e-mail.)

Self-Service Console Link: \${MailComposer.SelfserviceConsoleURL}

Additional Comments: \${MailComposer.AdditionalEmailComments}

(If the Additional Comments field is empty, the Additional Comments field will not be in the e-mail.)

```
#if( ${Principal.LastName} == "Paulson" )
    Your hardware token will be shipped to your home.
#end
```

If you did not initiate this request, please contact your administrator with the information in this e-mail.

Customizing E-mail Notifications for Proxy Servers

If you set up a proxy server in your network DMZ to protect Authentication Manager, you must customize the e-mail notifications by replacing the URL for the authentication server with the information for the proxy server.

Note: When you set up secure web publishing, you create a server certificate on the application machine that contains a server certificate common name. Be sure to use the server certificate common name for the *<website alias name>* in the proxy server replacement tags.

Replace the default e-mail tags for the authentication server with the proxy server replacement tags listed in the following table.

Default E-mail Tag	Proxy Server Replacement Tag
Request approval for hardware token e-mail template:	
Replace the CT-KIP URL and service address. (You do not need to delete the <code>\${MailComposer.CtkipURL}</code> tag in the e-mail template.)	To replace the CT-KIP URL and service address: <ol style="list-style-type: none"> Click Setup > Component Configuration > Authentication Manager > Basic Settings. Under CT-KIP Generation, replace the content of the Token-Key Generation URL field with: <code>https://<website alias name>/ctkip/trigger.jsp?dest url=http://www.rsa.com</code> where <i><website alias name></i> is the server certificate common name Under CT-KIP Generation, replace the content of the Service Address field with: <code>https://<website alias name>/ctkip/services/CtkipService</code> where <i><website alias name></i> is the server certificate common name
<code>\${MailComposer.EnablementURL}</code>	<code>https://<website alias name>/console-selfservice/EnableToken.do?action=nvEnableToken</code> where <i><website alias name></i> is the server certificate common name



Default E-mail Tag	Proxy Server Replacement Tag
<code>\${MailComposer.SelfserviceConsoleURL}</code>	<code>https://<website alias name>/console-selfservice</code> where <website alias name> is the server certificate common name
Request approval for software token e-mail template:	
Replace the CT-KIP URL and service address. (You do not need to delete the <code>\${MailComposer.CtkipURL}</code> tag in the e-mail template.)	To replace the CT-KIP URL and service address: <ol style="list-style-type: none">Click Setup > Component Configuration > Authentication Manager > Basic Settings.Under CT-KIP Generation, replace the content of the Token-Key Generation URL field with: <code>https://<website alias name>/ctkip/trigger.jsp?dest url=http://www.rsa.com</code> where <website alias name> is the server certificate common nameUnder CT-KIP Generation, replace the content of the Service Address field with: <code>https://<website alias name>/ctkip/services/CtkipService</code> where <website alias name> is the server certificate common name
<code>\${MailComposer.EnablementURL}</code>	<code>https://<website alias name>/console-selfservice/EnableToken.do?action=nv EnableToken</code> where <website alias name> is the server certificate common name
<code>\${MailComposer.DownloadURL}</code>	Specify a link to the location where users can download the software token application.
<code>\${MailComposer.HelpLink}</code>	Add a link to the Help that describes how to use the software token application.
<code>\${MailComposer.SelfserviceConsoleURL}</code>	<code>https://<website alias name>/console-selfservice</code> where <website alias name> is the server certificate common name
Request approval for the on-demand tokencode service e-mail template:	
<code>\${MailComposer.SMSOttURL}</code>	<code>https://<website alias name>/console-self-service/OnDemandOTTLogin.do?action-nvPreEdit</code> where <website alias name> is the server certificate common name

Default E-mail Tag	Proxy Server Replacement Tag
Request available e-mail template:	
<code>\${MailComposer.WorkflowParticipantConsoleURL}</code>	<code>https://<website alias name>/console-ucm</code> where <website alias name> is the server certificate common name
Request approval for non-token e-mail template:	
<code>\${MailComposer.SelfserviceConsoleURL}</code>	<code>https://<website alias name>/console-selfservice</code> where <website alias name> is the server certificate common name

Using E-mail Template Tags

The default e-mail templates use tags that you can modify.

The syntax for e-mail template tags is:

``${objectname.propertyname}`

where:

- *objectname* is the name of a Credential Manager object.
- *propertyname* is the property name of the object.

For example:

``${Principal.FirstName}`

is the first name of an approver, distributor, or user.

The following table lists the default e-mail template tags and indicates whether the tags support conditional statements.

Object Name	Property Name	Type	Description	Conditional Statement Support
Principal	UserID	String	User ID (approver, distributor, or user)	Yes
Principal	FirstName	String	User first name (approver, distributor, or user)	Yes
Principal	LastName	String	User last name (approver, distributor, or user)	Yes
Principal	MiddleName	String	User middle name (approver, distributor, or user)	Yes
Principal	E-mail	String	User e-mail ID (approver, distributor, or user)	Yes

Object Name	Property Name	Type	Description	Conditional Statement Support
Principal	LastLogin	Date	Date of user's last logon	No
Principal	isEnabled	boolean	User account is enabled or disabled	Yes
UCMRequest	CreatedBy	String	User who created the request	No. Use Principal.UserId instead.
UCMRequest	CreatedOn	Date	Date the request was created	No
UCMRequest	ModifiedBy	String	User who made the last change to the request	No
UCMRequest	ModifiedOn	Date	Date when the request was modified	No
UCMRequest	Status	int	Status of the request. The request statuses and their values are: <ul style="list-style-type: none"> • Open - 1 • Complete - 2 • Cancelled - 3 • Error - 5 • Rejected - 6 • Approved - 7 • Distributed - 8 	Yes
UCMRequest	RequestID	String	Confirmation number of request	Yes
SecurityDomain	ParentName	String	Name of the parent security domain of this security domain instance Note: If you create a display name for a security domain, the display name does not appear in e-mail. The system name of the security domain appears in e-mail. Do not use if the parent security domain is not available.	Do not use if there is no parent domain.
SecurityDomain	Name	String	Name of the security domain Note: If you create display names for security domains, the display names do not appear in e-mails. The system names of the security domains appear in e-mails.	Yes

Object Name	Property Name	Type	Description	Conditional Statement Support
IdentitySource	Name	String	Name of the identity source Note: If you create display names for identity sources, the display names do not appear in e-mails. The system names of the identity sources appear in e-mails.	Yes
MailComposer	RequestType	String	Type of request	Yes
MailComposer	WorkflowParticipantConsoleUrl	String	URL for the RSA Security Console	No
MailComposer	RejectionDate	String	Date request was rejected	No
MailComposer	ApprovalDate	String	Date request was approved	No
MailComposer	GroupName	String	Group name or requested group name Note: If you create display names for groups, the display names do not appear in e-mails. The system names, separated by commas appear in e-mails. For example, Eng,QA appears in the e-mail of a user who is a member of the engineering and QA groups.	Yes
MailComposer	SelfserviceConsoleUrl	String	URL for the RSA Self-Service Console	No
MailComposer	TokenType	String	Display name or friendly name of token	Yes
MailComposer	CtkipCode	String	CT-KIP token activation code	Yes
MailComposer	CtkipUrl	String	URL to download CT-KIP token	No
MailComposer	EnablementUrl	String	URL to activate token	No
MailComposer	EnablementCode	String	Code to activate token	Yes
MailComposer	SerialNumber	String	Serial number of token	Yes
MailComposer	DownloadUrl	String	URL to download the software token client application	No
MailComposer	HelpLink	String	Help link for the software token device	No
MailComposer	SmsOttUrl	String	URL to download the on-demand tokencode	No

Object Name	Property Name	Type	Description	Conditional Statement Support
MailComposer	ReasonEnteredByWorkflowParticipant	String	Reason that Approver rejected request	Yes
MailComposer	AdditionalEmailComments	String	Comments added by workflow participant when resending e-mails to users	Yes
MailComposer	ReasonForFailure	String	Reason request failed	Yes
MailComposer	NL	String	Adds a new line. Do not use in conditional statements.	No
MailComposer	TokenTypeCTKIP	String	Checks to see if token type uses CT-KIP	Yes
MailComposer	ParticipantMail ID	String	List of workflow participants' e-mail IDs. For example: dsmith@acme.com, gjones@acme.com.	Yes
MailComposer	ApprovalDetails	String	Comments entered by users	No
MailComposer	RequestHistory	String	Steps in a request. The steps used to complete a request and the status. For example: <ActivityName> - <Activity Status> on <date&time> by <userID>. Comments to user: <comments>	No

Conditional Statements in E-mail Templates

You can use conditional statements in your e-mail templates to include customized information if certain conditions are met.

Guidelines for Conditional Statements in E-mail Templates

Credential Manager uses Velocity, which is part of the Apache Velocity Project, to process e-mail templates. Velocity is a Java-based template engine.

If you use conditional statements in your customized e-mail templates:

- You must follow the Velocity syntax rules. For information about the Velocity syntax for conditional statements, go to <http://velocity.apache.org>.
- Do not add a new line (NL) in a conditional statement.

Note: If a variable has a null value, or if it is a boolean false, the statement evaluates as false, and there is no output.

The following examples show how to use e-mail template tags in conditional statements.

Examples:

- String Example

```
#if( ${Principal.FirstName} == "John" )
    do some thing
#else
    do nothing
#end
```
- Boolean Example

```
#if( ${Principal.isEnabled} )
    do some thing
#end
```
- Integer Example

```
#if( ${UCMRequest.Status} < 5 )
    do some thing
#else
    do nothing
#end
```
- Date Example

```
#if( ${ UCMRequest.ModifiedOn } == "Aug 8, 2007 12:49:14 PM GMT+05:30"
)
    do some thing
#end
```

Customizing Help for the RSA Self-Service Console

You can customize the Self-Service Console Help (RSA Self-Service Console Frequently Asked Questions) to reflect how your company uses self-service and provisioning.

By editing the FAQ HTM files, you can customize the Help to:

- Reflect any changes that you make to the Self-Service Console when editing the properties file.
- Add information for your company using company-specific terminology.
- Remove non-applicable information from the FAQ. For example, if your company does not use the provisioning feature of Credential Manager, you can delete the appropriate HTM files.

The location of the HTM files is:

```
...\\RSA Security\\RSA Authentication Manager\\server\\servers\\rsa-help\\eclipse\\plugins\\com.rsa.ucmuser.console.help\\console-help\\
```

Customizing Token Graphics

Users view token graphics when they request new or additional tokens from the Self-Service Console. You can replace the default token graphics that ship with Credential Manager with your company's custom token graphics.

The token graphics are stored in a Web ARchive (.war) file. A .war file could be a .jar file used to distribute a collection of JavaServer Pages, servlets, Java classes, XML files, tag libraries, and static Web pages (HTML and related files) that together constitute a web application.

To customize a token graphic:

1. Stop the Authentication Manager service.
 - a. Click **Start > Settings > Control Panel**.
 - b. Click **Administrative Tools**.
 - c. Click **Services**.
 - d. Right-click **RSA Authentication Manager Cluster Administration Server** from the list of Services, and then click **Stop**.
2. On the RSA Authentication Manager server, locate the default token images in **...\\RSA Security\\RSA Authentication Manager\\...\\components\\ucm**
3. Back up the three .war files: **console-ucm.war**, **console-selfservice.war**, **console-troubleshoot.war**.

4. Unzip the three .war files: **console-ucm.war**, **console-selfservice.war**, **console-troubleshoot.war** to a temporary folder.
After you unzip the three .war files, the default token images are available in the following directories:
.../console-selfservice/images/default/tokens
.../console-troubleshoot/images/default/tokens
.../console-ucm/images/default/tokens
5. Replace the default token images in each directory with the custom token images for your company.
6. Zip the directories that you changed and save them as .war files. Be sure to rename each .war file with the same names as the original files:
console-ucm.war
console-selfservice.war
console-troubleshoot.war
7. Replace the default .war files in **.../RSA Security/RSA Authentication Manager/.../components/ucm** with the new .war files.
8. Copy the new .war files to their respective directory paths:
RSA_AM_HOME/server/servers/machinename_server/stage/console-ucm
RSA_AM_HOME/server/servers/machinename_server/stage/console-selfservice
RSA_AM_HOME/server/servers/machinename_server/stage/console-troubleshoot
where:
 - **RSA_AM_HOME** is the directory where you installed RSA Authentication Manager.
 - **machinename_server** is the name of the machine where you installed RSA Authentication Manager.
9. Delete the following temporary folder:
RSA_AM_HOME/server/servers/tmp
10. Start the Authentication Manager service.
 - a. Click **Start > Settings > Control Panel**.
 - b. Click **Administrative Tools**.
 - c. Click **Services**.
 - d. Right-click **RSA Authentication Manager Cluster Administration Server** from the list of Services, and then click **Start**.

11. Update the name of each token with the new name for each of the new images. From the Security Console, do the following:
 - a. Click **Setup > Component Configuration > Credential Manager**.
 - b. Click **Manage token self-service**.
 - c. In the **Display Name** field, replace the name of each token that is displayed with the new name for each of the new images in the three edited .war files to help users select the appropriate token.
The display name appears on the RSA Self-Service Console pages that users see when they select tokens.
 - d. Click **Save**.

Customizing Workflow and Non-Workflow Operations

Workflow operations require one or two approval steps and possibly a distribution step. Non-workflow operations do not require any approval or distribution steps.

You can write custom extensions for Credential Manager for workflow and non-workflow operations by writing an API.

Some examples of workflow and non-workflow customizations are:

- For a workflow operation, you can validate that you have the number of tokens needed for token requests.
- For a non-workflow operation, such as when users modify user profiles, you can add a custom extension to update other internal employee databases to reflect these changes.

For more information about creating a custom extension, see the *Developer's Guide*. After you create a custom extension, you must register it using the Register Custom Extension utility. For more information, see Appendix D, "[Command Line Utilities](#)."

Customizing the Self-Service Console Home Page

You can customize the header text of the Self-Service Console Home page using the Security Console. For more information, see the Security Console Help topic "Customize the RSA Self-Service Console Home Page."

C

Managing RSA SecurID Tokens with the Microsoft Management Console (MMC)

- [Overview of the Microsoft Management Console \(MMC\)](#)
- [Assigning and Unassigning Tokens](#)
- [Disabling and Enabling Tokens](#)
- [Editing User Authentication Attributes](#)
- [Editing Token Properties](#)
- [Replacing Tokens](#)
- [Managing PINs](#)
- [Providing Emergency Access](#)

Overview of the Microsoft Management Console (MMC)

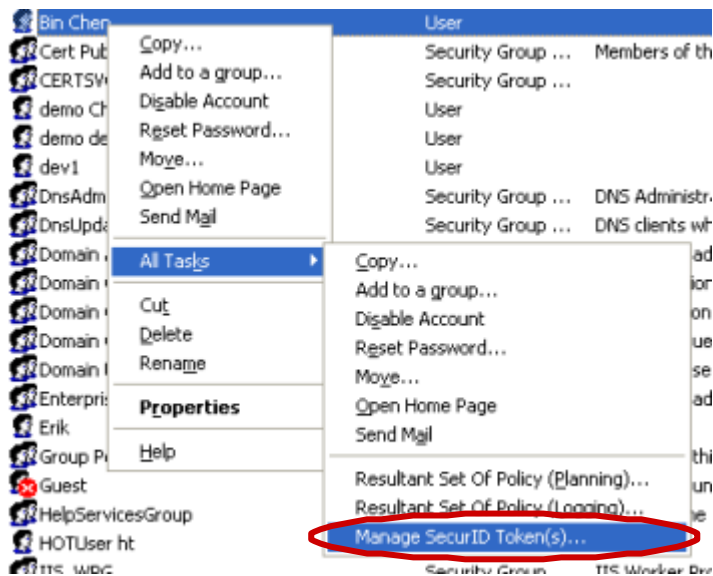
RSA Authentication Manager offers a snap-in that you can use if your system uses Active Directory as its identity source. The Authentication Manager MMC snap-in allows you to use the Microsoft Management Console (MMC) for Active Directory to perform many of your token-related tasks. This eliminates the need to log on to the RSA Security Console to enable or disable a token, assign a token, or perform other token-related tasks.

Note: The pages in the Authentication Manager snap-in are designed to match their corresponding pages in the Security Console. This provides consistency for administrators who want to use both tools for token-related tasks.

The Authentication Manager snap-in adds a menu option, Manage SecurID Token(s), to the user menu in the MMC. To access the Authentication Manager snap-in:

1. Open the **Active Directory Users and Computers** tool.
2. Open the **Users** folder so that you can view all of the users and user groups in the Active Directory.
3. Right-click on a user name to launch the user menu.
4. From the user menu, select **All Tasks**.

- From the **All Tasks** menu, click **Manage SecurID Token(s)**.



The **RSA Token Management** window opens.

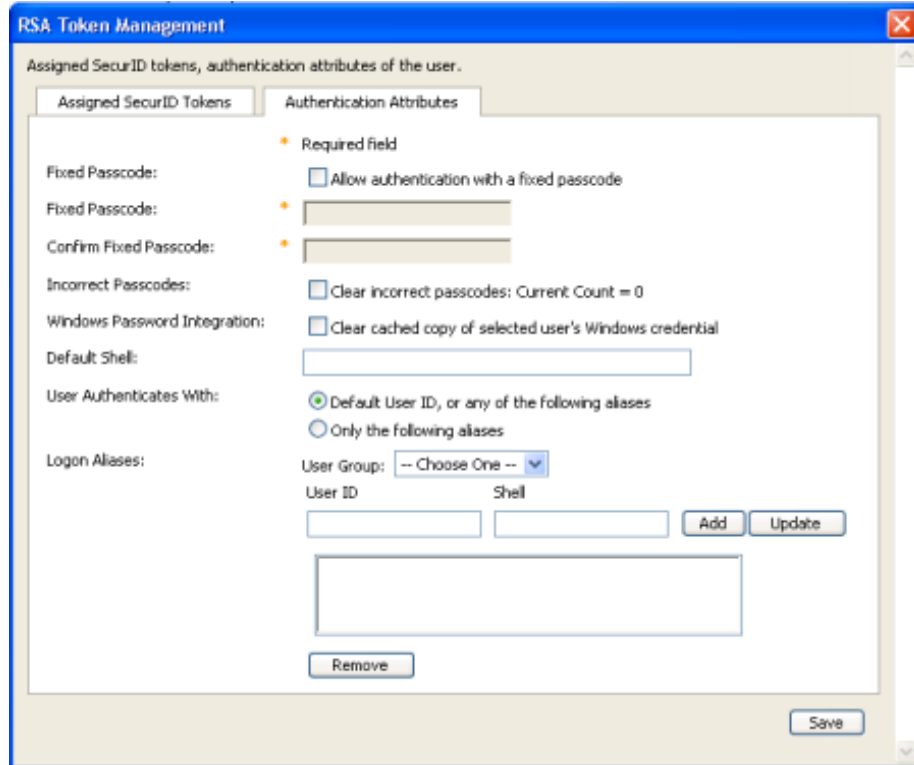
From RSA Token Management, click the **Assigned SecurID Tokens** tab to view assigned tokens and perform other token-related functions.

The Assigned SecurID Tokens page: is displayed in the following figure.



Click the **Authentication Attributes** tab to set authentication attributes such as passcode requirements.

The Authentication Attributes page is displayed in the following figure.



Perform most of your token-related tasks such as assigning and unassigning tokens, managing user PINs, and providing emergency access, on the Assigned SecurID Tokens page.

Some token-related tasks open a new browser window. When you assign a token, for example, a new browser window opens, allowing you to search for SecurID tokens. After you select and assign a token, the browser window closes, and you return to the Assigned SecurID Tokens page.

The Assigned SecurID Tokens page and all of the pages that are launched from it are part of the Authentication Manager snap-in.

Note: When using the Authentication Manager snap-in, press the F1 key to access MMC Help topics.

For more information, see the MMC Help topic “Manage Tokens Using the Microsoft Management Console (MMC).”

Each token-related task is described in detail in the following sections.

Assigning and Unassigning Tokens

You can use the Authentication Manager snap-in to assign or unassign a token to a user. Use the Assign Token to User page to:

- Search for a token by security domain, token status, serial number, expiration date, modified date, or imported date.
- View token information such as serial number, token type, expiration date, security date, and whether the token is enabled or disabled.
- Assign a token.

After assigning a token, the Assign Token to User page closes, and you return to the Assigned SecurID Tokens page. The token now displays in the table of assigned tokens.

To unassign a token, select the token on the Assigned SecurID Tokens page, and click **Unassign Token**.

Note: A token is automatically disabled when it is unassigned.

For more information on using the MMC to assign hardware and software tokens, see the MMC Help topics “Assign Hardware Tokens Using the Microsoft Management Console (MMC)” and “Assign Software Tokens Using the Microsoft Management Console (MMC).”

For more information on unassigning hardware and software tokens, see the MMC Help topic “Unassign Tokens Using the Microsoft Management Console (MMC).”

Disabling and Enabling Tokens

You can use the Authentication Manager snap-in to enable and disable SecurID tokens. An enabled token can be used for authentication, but a disabled token cannot.

There are two ways to enable and disable tokens:

- Assigning and unassigning a token automatically enables or disables it. When you assign a token to a user, it is automatically enabled. When you unassign that same token, it becomes disabled.
- You can manually enable or disable a token on the Assigned SecurID Tokens page.

For instructions, see the MMC Help topics “Disable Tokens Using the Microsoft Management Console (MMC)” and “Enable Tokens Using the Microsoft Management Console (MMC).”

Editing User Authentication Attributes

You can use the Authentication Manager snap-in to view and edit a user's authentication attributes, such as:

- Assign a fixed passcode
- Clear incorrect passcodes
- Windows password integration
- Logon aliases

To edit the user authentication attributes, click the **Authentication Attributes** tab.

For instructions, see the MMC Help topic “View and Edit Authentication Attributes Using the Microsoft Management Console (MMC).”

Editing Token Properties

You can use the Authentication Manager snap-in to view and edit token properties, such as:

- Security domain.
- Token basics such as serial number, token type, imported and exported dates, assigned information, token lifetime, and last modified date.
- Authentication attributes such as PIN status and whether or not a PIN is required for authentication. You can also force a user to change the PIN.
- Disable the token.

View and edit token properties on the Token Properties page.

For instructions, see the MMC Help topic “View and Edit Token Properties Using the Microsoft Management Console (MMC).”

Replacing Tokens

You can use the Authentication Manager snap-in to replace lost, stolen, or damaged tokens.

Replace tokens on the Replace Token page of the snap-in. You can:

- Search for a token by security domain, token status, serial number, expiration date, modified date, or imported date.
- View token information such as serial number, token type, expiration date, security date, and whether the token is enabled or disabled.
- Replace a token by selecting the token, and clicking **Replace Token**.

For instructions, see the MMC Help topic “Replace Tokens Using the Microsoft Management Console (MMC).”

Managing PINs

Occasionally, you must manage user PINs. Users may forget their PIN or fear that their PIN has been compromised. In these situations, it is necessary to clear the existing PIN or force the user to create a new one. You can also allow users to authenticate without having to enter a PIN with the tokencode.

On the Assigned SecurID Tokens page of the snap-in, use the SecurID PIN menu to:

Clear PIN. Clear a user's SecurID PIN.

Require SecurID PIN. Make the PIN a requirement for authentication.

Don't Require SecurID PIN. Allow the user to authenticate without a PIN.

Requires SecurID PIN Change. Force a user to change the PIN on the next successful authentication attempt.

For instructions, see the MMC Help topics "Clear an RSA SecurID PIN Using the Microsoft Management Console (MMC)" and "Force RSA SecurID PIN Changes Using the Microsoft Management Console (MMC)."

Providing Emergency Access

Occasionally, you must provide emergency access for your users. Users may require emergency access if their token is temporarily or permanently unavailable. For example, users need emergency access if they have a misplaced, lost, stolen, or damaged token, or if they are waiting for a replacement token to arrive.

You can provide emergency access to Authentication Manager for the following two scenarios:

- **Online authentication.**
Provide emergency access for users with misplaced, lost, stolen, damaged, or expired tokens. Temporary emergency access is available using an Online Emergency Access Tokencode.
- **Offline authentication.**
Provide emergency access for RSA SecurID for Windows users who require emergency access while authenticating offline. Temporary emergency access can be provided using an Offline Emergency Access Tokencode.

Important: If the user has an expired token, replace the token and provide temporary access. An Emergency Access Tokencode cannot be assigned to an expired token. See "[Replacing Tokens](#)" on page 259.

These scenarios are described in detail in the following sections.

Generating a Temporary Tokencode for Online Authentication

You can provide emergency online access by generating an Online Emergency Access Tokencode. The Online Emergency Access Tokencode is an 8-character alphanumeric code generated by Authentication Manager and used for online access to protected resources.

Similar to the SecurID tokencode, the Online Emergency Access Tokencode is combined with the user's PIN to create a passcode. By using a PIN with the Online Emergency Access Tokencode, the user can still achieve two-factor authentication.

Note: The format of the Online Emergency Access Tokencodes is determined by the token policy of the security domain to which it belongs. For example, if the token policy is set to allow special characters, the Online Emergency Access Tokencode can include special characters.

You can use the Authentication Manager snap-in to create an Online Emergency Access Tokencode and set emergency access attributes. On the Assigned SecurID Tokens page, select the token, click **Emergency Access**, and then click the **Online Emergency Access** tab.

On the Online Emergency Access page, you can:

- Set the Online Emergency Access Tokencode lifetime.
For security reasons, you may want to limit the length of time the Online Emergency Access Tokencode can be used. Because the Online Emergency Access Tokencode is a fixed code, it is not as secure as the pseudorandom number generated by the token.
- Specify what happens if the missing token is recovered (if the user finds the lost token, for example). You have the following options:

- Deny authentication with token

Use this option if you do not want the token to be used for authentication if recovered.

Important: If the token is permanently lost or stolen, use this option. This safeguards the protected resources in the event the token is found by an unauthorized individual who attempts to authenticate.

- Allow authentication with token at any time and disable online emergency tokencode

Use this option if the token is temporarily misplaced (the user left the token at home, for example). When the user recovers the token, he or she can immediately resume using the token for authentication. The Online Emergency Access Tokencode is disabled.

- Allow authentication with token only after the emergency code lifetime has expired and disable online emergency tokencode

You can also use this option for temporarily misplaced tokens, however when the missing token is recovered, it cannot be used for authentication until the Online Emergency Access Tokencode expires.

Note: You cannot assign an Online Emergency Access Tokencode to a disabled token.

For instructions, see the MMC Help topic “Generate an Online Emergency Access Tokencode Using the Microsoft Management Console (MMC).”

Assigning a Temporary Tokencode for Offline Authentication

RSA SecurID for Windows users may need temporary emergency access so that they can authenticate while working offline. Users who are authenticating offline can gain temporary emergency access using an Offline Emergency Access Tokencode.

Similar to the SecurID tokencode, the Offline Emergency Access Tokencode is combined with the user's PIN to create a passcode. By using a PIN with the Offline Emergency Access Tokencode, the user can still achieve two-factor authentication.

You can use the Authentication Manager snap-in to view and configure the Offline Emergency Access Tokencode. On the Assigned SecurID Tokens page, select the token, click **Emergency Access**, and then click the **Offline Emergency Access** tab.

On the Offline Emergency Access page, you can:

- View the Offline Emergency Access Tokencode.
- View the Offline Emergency Access Tokencode expiration date.
- Reset the Offline Emergency Access Tokencode.
- Allow the user to use the Offline Emergency Access Tokencode for emergency online access.

Note: Offline Emergency Access Tokencodes can only be issued if the user has used the token to authenticate, at least once, to an agent that can provide offline data for SecurID for Windows users.

For instructions, see the MMC Help topic “Assign an Offline Emergency Access Tokencode Using the Microsoft Management Console (MMC).”

D

Command Line Utilities

This chapter contains the following sections:

- [Overview](#)
- [Archive Requests Utility](#)
- [Collect Product Information Utility](#)
- [Import PIN Unlocking Key Utility](#)
- [Manage Backups Utility](#)
- [Manage Batchjob Utility](#)
- [Manage Database Utility](#)
- [Manage Operations Console Administrators Utility](#)
- [Manage Secrets Utility](#)
- [Register Custom Extension Utility](#)
- [Set Trace Utility](#)
- [User Groups and Token Bulk Requests Utility](#)
- [Verify Archive Log Utility](#)

Overview

The following information is helpful to know before running a command line utility (CLU).

Using rsautil

The rsautil script provides the execution environment configuration necessary to run the RSA Authentication Manager CLUs that RSA provides. The rsautil script also runs custom Java or Jython applications using the RSA application programming interface (API) and software development kit (SDK). The rsautil script is located in the *RSA_AM_HOME/utills* directory.

Terminating Batch Jobs in Windows

If you are using a Windows environment, you can terminate a command line utility by pressing CTRL-C. When you press CTRL-C, Windows asks you if you would like to terminate the batch job, and allows you to select “Y” or “N.” The command line utility that you are running terminates regardless of whether you choose “Y” or “N.”

The CLU terminates because pressing CTRL-C interrupts the CLU, and Windows is unable to keep the CLU from terminating, even if you choose “N.”

Note: This same behavior occurs when you run WebLogic inside a command window.

Credentials Required to Run Command Line Utilities

The following table lists the credentials that are required to run the command line utilities described in the Authentication Manager documentation set. Each command line utility requires one (or more) of the following:

Master password. The utility requires the password specified at installation. This password is required for most of the utilities.

Super Admin. The utility must be run by an administrator with Super Admin credentials.

Administrator password. The utility must be run by an administrator who has the appropriate permissions for running the utility. Administrators with the appropriate permissions can enter their own passwords on the command line.

Utility	Required Credentials
Archive Requests	master password & Super Admin
Collect Product Information	master password
Data Migration	master password
Generate Database Package	master password
Generate RADIUS Package	master password
Import PIN Unlocking Key	administrator password ¹
Manage Backups	master password
Manage Batchjob	Super Admin
Manage Database	master password
Manage Nodes	master password
Manage RSA Operations Console Administrators	Super Admin
Manage Replication	master password

Utility	Required Credentials
Manage Secrets	master password
Manage SSL Certificate	master password
Multicast Network Test	master password
Register Custom Extension	master password & Super Admin
Restore Super Admin	master password
Set Trace	Super Admin
Setup Replication	master password
Store	master password
Update Instance Nodes	master password
User Groups and Token Bulk Requests	master password & Super Admin
Verify Archive Log	master password

¹The administrator must be assigned a role that has PIN Unlock Key management (import and view) permissions.

Archive Requests Utility

Use the Archive Requests utility, `archive-ucm-request`, to export and import RSA Credential Manager closed requests with their associated attributes and workflow data. Credential Manager, a feature of Authentication Manager, runs this task as a batch job.

You can export closed requests if you want to free up disk space and to increase the speed of search operations in the system. You can import archived requests to resend an approval e-mail to a user who has lost the approval e-mail that contained a token file.

This utility exports the associated attributes and process instance data into a .zip file, **Request_YYYY_MM_DD_HH_MIN_SEC.ZIP** which contains the following files:

- Request Data - **Request_YYYY_MM_DD_HH_MIN_SEC.CSV**
- Request Attribute Data - **Request_Att_YYYY_MM_DD_HH_MIN_SEC.CSV**
- Process instance data - **Request_WF_YYYY_MM_DD_HH_MIN_SEC.ZIP**

Note: When you submit a batch job, make sure that you write down the job ID if you want to view, cancel, or delete an existing batch job. The batch job ID appears on the screen when the `archive_ucm_utility` batch job successfully runs. For more information, see [“Using the Manage Batchjob Utility”](#) on page 275.

Guidelines for Running the Archive Requests Utility

The following guidelines cover the export and import version of this utility:

- You can only export closed requests.
- When you import archived requests, you must use the same .zip file that you use to export requests. Do not change the name of the .zip file.

Using the Archive Requests Utility

To use archive-ucm-request:

1. Open a new command shell, and change directories to *RSA_AM_HOME/utls*.
2. Type:

```
rsautil archive-ucm-request -u username -p password
-m master-password -d directory -a EXPORT -S fromDate
-E toDate -D
```

For relevant options, see the following section [“Options for archive-ucm-request.”](#)

Important: Although it is possible to enter the master password on the command line along with the other options, this creates a potential security vulnerability. RSA recommends that you enter the master password only when the utility presents a prompt.

Examples:

- To export requests from 03/03/2007 to 03/07/2007 and delete the requests after exporting them, type:

```
rsautil archive-ucm-request -u jsmith -p steverest
-m portersq -d C:\archivedRequest -a EXPORT
-S 03/03/2007 -E 03/07/2007 -D
```

The archived requests are saved in a .zip file.

- To import archived requests and the associated process instance data, type:

```
rsautil archive-ucm-request -u jsmith -p st5everest
-m porter95sq -d C:\archivedRequest -a IMPORT -j
Request_2007_03_06_15_48_26.zip
```

Options for archive-ucm-request

The following table describes the options for this utility.

Flag	Alternate Flag	Description
-a	--archiveOption	Archive option: IMPORT or EXPORT.
-d	--directory	Directory path for archived requests.
-h	--help	Optional. Displays help for this utility.
-I	--interactive	Optional. Runs the utility in interactive mode.
-m	--master-password	Master password.
-p	--password	Password of the administrator running the utility.
-u	--userID	User name of the administrator running the utility.
-v	--version	Optional. Displays the version and copyright information.
Export Options		
-D	--delete	Optional. Deletes records after exporting.
-E	--toDate	Requests created on or before this date are exported. [mm/dd/yyyy].
-S	--fromDate	Requests created on or after this date are exported. [mm/dd/yyyy].
Import Options		
-j	--file	.zip file for import.

Collect Product Information Utility

Use the Collect Product Information utility, `collect-product-info`, to collect system information, such as system log files and version information. This information is used to diagnose problems.

This utility collects the information, packages it into a file named **product_info.jar**, and encrypts the file. The encrypted file is transferred to a recipient who analyzes its contents. The recipient uses the Collect Product Information utility to decrypt the file.

Important: An improper DNS configuration can cause errors when this utility collects patch information from nodes in a cluster. If the patch information for a server node is not in the support package file, ensure that the DNS is configured correctly, and restart any of the servers in the cluster.

Important: The user who runs this utility to decrypt the **product_info.jar** file must know the package password you specify when you run this utility to create the **product_info.jar** file.

Using the Collect Product Information Utility

To use `collect-product-info`:

1. Open a new command shell, and change directories to *RSA_AM_HOME/utils*.
2. Type:

```
rsautil collect-product-info options
```

For relevant options, see the following section, [“Options for collect-product-info.”](#)

To collect system information and export it to an encrypted file, you use the following options:

```
rsautil collect-product-info --export
--archive-time "2006-07-17 22:32:10.000"
--package-password package_password
```

where:

- “2006-07-17 22:32:10.000” is the archive time.
- *package_password* is the password to encrypt the **product_info.jar** file.

Important: Although it is possible to enter the master password on the command line along with the other options, this creates a potential security vulnerability. RSA recommends that you enter the master password only when the utility presents a prompt.

Important: The support package file can contain sensitive data. RSA recommends that you move this file to a directory with appropriate access control after it is generated.

Options for collect-product-info

The following table describes the options for this utility.

Flag	Alternate Flag	Description
-t	--archive-time	This is the archive time. The log files are retrieved from the data store after this local time stamp and until the current time. The format is “yyyy-mm-dd hh:mm:ss.SSS”.
		Note: You must have double quotation marks around the archive time, and specify it in local 24-hour military time. If the archive time is not provided, log records from the previous hour are exported.
	--export	Collects system information and exports it to the encrypted product_info.jar file.
-h	--help	Displays help for this utility.
	--import	Decrypts the product_info.jar file.
		Note: The file must be located in the current working directory.
-m	--master-password	Master password of the encrypted properties file.
-p	--package-password	Password to encrypt or decrypt the product_info.jar file.
-v	--version	Displays version and copyright information.

Import PIN Unlocking Key Utility

The PIN Unlocking Keys for the RSA SecurID SID800 Smart Card are stored on a readable XML file. If you want to import this data into Authentication Manager so that it can be viewed in the Security Console, use the Import PIN Unlocking Key utility, `import-puk`.

Once you have imported the XML data, you can view the PIN Unlocking Key on the Token Properties page in the Security Console.

Note: Administrators must have the appropriate permissions to import and view the data.

Using the Import PIN Unlocking Key Utility

To use import-puk:

1. Open a new command shell, and change directories to *RSA_AM_HOME/utlils*.
2. Type:

```
rsautil import-puk -u username -p password -f filename
```

For relevant options, see the following section, [“Options for import-puk.”](#)

Important: Although it is possible to enter the master password on the command line along with the other options, this creates a potential security vulnerability. RSA recommends that you enter the master password only when the utility presents a prompt.

Example:

- To import multiple PIN unlocking key records, type:

```
rsautil import-puk -f PUK.xml -u username -p password
```

Result:

```
Status: IMPORTED 10 PIN UNLOCK KEY (PUK) RECORDS  
SUCCESSFULLY.
```

Options for import-puk

The following table describes the options for this utility.

Flag	Alternate Flag	Description
-u	--user-id	User name of the administrator to access the back-end server.
-p	--password	Password of the administrator running the utility.
-f	--filename	The name of the file containing the data.
-b	--batch	Runs the import as a batch job.
-i	--interactive	Optional. Runs the import in interactive mode.
-v	--version	Optional. Displays the version and copyright information.
-h	--help	Displays help for this utility.

Manage Backups Utility

Use the Manage Backups utility, `manage-backups`, to export the data from the internal database, and restore the data to the database. This utility only supports an internal Oracle database. It does not support Microsoft SQL.

Important: Due to the customization required for setting up the internal database, RSA recommends that you use only the Manage Backups utility for backing up the internal database. Using a third-party application for backups is not supported and can cause problems.

Note: If you are importing to the database or exporting from the database on an NFS partition, the partition must be mounted with the `rsize=value` and `wsize=value` parameters defined. These hard parameters force connection attempts to retry until the NFS file server responds. The NFS read chunk size is `rsize`, and the NFS write chunk size is `wsize`. For these two parameters, a value of 8,192 is recommended for NFS version 2, and 32,768 is recommended for NFS version 3.

Using the Manage Backups Utility

Important: When you run the Manage Backups utility, the database process must be running. The backup operation does not encrypt the backup file. To encrypt the backup file, use a third-party encryption application.

To use `manage-backups`:

1. Open a new command shell, and change directories to `RSA_AM_HOME/utils`.
2. Type:

```
rsautil manage-backups options
```

For relevant options, see the following section, [“Options for manage-backups.”](#)

Important: Although it is possible to enter the master password on the command line along with the other options, this creates a potential security vulnerability. RSA recommends that you enter the master password only when the utility presents a prompt.

Use the indicated options to perform the following tasks:

- To export the database to a backup file, type:

```
rsautil manage-backups --action export
--filename absolute_path_filename
```

where *absolute_path_filename* is the absolute path and name of the file where the database data is stored.

- To import the database from a backup file, type:

```
rsautil manage-backups --action import
--filename absolute_path_filenames
```

where *absolute_path_filenames* is the absolute path and name of the files created using the export action that contain the database data.

Transferring the Internal Database from One Machine to Another Machine

To transfer the internal database from one machine to another machine:

1. On the machine that hosts the source database, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.

2. Type:

```
rsautil manage-backups --action export --transfer
--filename absolute_path_filename
```

where *absolute_path_filename* is the absolute path and name of the backup file that is used to restore the database.

Using the transfer option with the export action generates two files based on the name that you provide:

filename. The database backup file.

filename.secrets. The user credentials backup file.

3. On the machine that hosts the destination Authentication Manager installation, stop the application server.
4. Copy the two files generated in [step 2](#) to the machine that hosts the destination internal database.
5. On the machine that hosts the destination internal database, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
6. Type:

```
rsautil manage-backups --action import --transfer
--filename absolute_path_filenames
```

where *absolute_path_filenames* is the absolute path and names of the two backup files created in [step 2](#).

7. Start the application server on the machine that hosts the destination database server.
8. Open the RSA Operations Console on the destination machine, and verify that the database content on the source machine is present in the destination machine by viewing the identity source.

Restoring the Internal Database in a Replicated Environment

If the primary instance in a replicated environment stops working, you can restore the system by restoring the internal database to a replica instance and promoting that replica instance to a primary instance. To restore the internal database to a replica instance, you must:

- Stop the replication process to prevent corrupting the database on each replica instance.
- Restore the database from the database backup files to a replica instance.
- Promote that replica instance to replace the stopped primary instance.
- Attach the remaining replica instances to the new primary instance.

Note: Before you perform the following procedure, you must use the Manage Backups utility to export the database backup files from the internal database on the primary instance. These backup files must be moved to the replica instance that will become the new primary instance.

Use this procedure for an installation where the internal database is on the same machine as the primary instance.

To restore the internal database from backup files in a replicated environment:

1. If the primary instance is not working, go to [step 2](#). Stop all Authentication Manager services on the primary instance by doing one of the following:
 - On Windows systems, go to the Windows Control Panel. Click **Administrative Tools > Services**. In the Services (Local) list, right-click each Authentication Manager service that is running, and select **Stop** from the context menu. It can take a few minutes for a service to stop.
 - On Linux systems, open a new command shell, and change directories to ***RSA_AM_HOME/server***, and type:

```
./product_name stop all
```

where *product_name* is the name of your RSA product. For example, rsaims.
2. Log on to the replica instance that will be the new primary instance. This is the instance where you have stored the database backup files.

Note: You do not need to use the most recently updated replica instance for the new primary instance. All of the databases are synchronized to the backup.

3. On the machine that will be the new primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
4. Type:

```
rsautil manage-backups --action import  
--filename absolute_path_filenames
```

where *absolute_path_filenames* is the absolute path and names of the backup files that are used to restore the database.

5. In the *RSA_AM_HOME/utils* directory on the machine that will be the new primary instance, type:

```
rsautl manage-replication --action promote
--recovery-mode
```

This action promotes the replica instance to the primary instance.

6. In the *RSA_AM_HOME/utils* directory on the new primary instance, type:

```
rsautl manage-replication --action attach-online
--name replica_instance_name
```

where *replica_instance_name* is the name of the replica instance being attached to the new primary instance.

7. In the *RSA_AM_HOME/utils* directory on the new primary instance, type:

```
rsautl manage-replication --action attach-status
```

Examine the returned information to verify that the replica instance is attached.

Options for manage-backups

The following table describes the options for this utility.

Flag	Alternate Flag	Description
-a	--action	Specifies the action to perform. Select one of the following: import. Imports the database from a backup file. export. Exports the database to a backup file.
-t	--addTimestamp	Optional. Indicates that the time stamp is added as part of the exported filename. Filename is exported in this format: <i>filename_timestamp_extension</i> . If this option is not included, the time stamp is not included as part of the filename.
-f	--filename	Name of the file to import from or export to.
-h	--help	Displays help for this utility.
-L	--includelog	Optional. Indicates that the log data is transferred with the database. If this option is not included, the log data is not transferred.
-g	--logonly	Optional. Indicates that the import or export command is only used for the log data. Cannot be used along with the -L or -D option.
-m	--master-password	Master password of the encrypted properties file.
-q	--quiet	Optional. Indicates that the utility will present prompts to answer. If this option is not included, prompts are not presented.

Flag	Alternate Flag	Description
-D	--transfer	Optional. Indicates that the export or import command is used to transfer a database. When the transfer option is used with the export action, two files are generated: filename and filename.secrets .
-V	--verbose	Optional. Displays more output messages.
-v	--version	Displays the version and copyright information.

Manage Batchjob Utility

Use the Manage Batchjob utility, `manage-batchjob`, to view, cancel, or delete an existing batch job. You can use the utility to manage batch jobs submitted using other command line utilities (for example, the Migration utility).

Note: You need the job ID to run the Manage Batchjob utility. When submitting a batch job, make sure you write down the resulting job ID.

Using the Manage Batchjob Utility

Note: You must run the command from the same system on which the application server is installed.

To use `manage-batchjob`:

1. Open a new command shell, and change directories to **`RSA_AM_HOME/utlils`**.
2. Type:

```
rsautil manage-batchjob -u username -p password -j job ID
-a action
```

For relevant options, see the following section, "[Options for `manage-batchjob`.](#)"

Important: Although it is possible to enter the master password on the command line along with the other options, this creates a potential security vulnerability. RSA recommends that you enter the master password only when the utility presents a prompt.

Examples:

- To view the status of a batch job, type:

```
rsutil manage-batchjob -u username -p password -j job
ID -a view
```

Result:

```
Job ID: ims.cb0f0b12d123456a00aca12b1a12f123
Status: COMPLETED
```

Note: The status can be any of the predefined Authentication Manager statuses.

- To cancel a running batch job, type:

```
rsutil manage-batchjob -u username -p password -j job
ID -a cancel
```

Result:

```
Batch job with ID ims.cb0f0b12d123456a00aca12b1a12f123
has been canceled.
```

- To delete a completed batch job from the database, type:

```
rsutil manage-batchjob -u username -p password -j job
ID -a delete
```

Result:

```
Batch job with ID ims.cb0f0b12d123456a00aca12b1a12f123
has been deleted.
```

Note: Deleting a batch job removes the completed job record from the database. It does not cancel or end the job.

- To display command line debug messages (use this option for view, cancel, or delete), type:

```
rsutil manage-batchjob -u username -p password -j job
ID -a view -X
```

Sample Result:

```
[DEBUG] Attempting to login with the given
administrative credentials...
[DEBUG] Administrative credential login was
successful.
[DEBUG] Obtaining batch current job status...
[DEBUG] Current batch job status obtained.
Job ID: ims.cb0f0b12d123456a00aca12b1a12f123
Status: CANCELED
```

Note: The debug option debugs the operation at the command line level. It does not debug the actual batch job.

Options for manage-batchjob

The following table describes the options for this utility.

Flag	Alternate Flag	Description
-u	--user-id	User name of the administrator to access the back-end server.
-p	--password	Password of the administrator running the utility.
-j	--job-id	Identification number of the batch job.
-a	--action	Set this to one of the following: view. View the state of the batch job. cancel. Cancel a running batch job. delete. Delete a completed batch job from the database.
-X	--debug	Displays debug messages.

Manage Database Utility

Use the Database Storage Management utility, `manage-database`, to examine and adjust the Authentication Manager internal database.

While installing Authentication Manager, you can choose to install the internal database on a different machine to improve performance. If you do this, use this utility to stop, start, or check the status of a database that is installed on another machine.

Note: This utility is intended for an Oracle database. It does not work with MS SQL.

Using the Manage Database Utility

To use manage-database:

1. Open a new command shell, and change directories to `RSA_AM_HOME/utlils`.
2. Type:

```
rsautl manage-database options
```

For relevant options, see the following section, "[Options for manage-database.](#)"

Important: Although it is possible to enter the master password on the command line along with the other options, this creates a potential security vulnerability. RSA recommends that you enter the master password only when the utility presents a prompt.

Use the indicated options to perform the following tasks:

- To view a database file status, type:


```
rsautil manage-database -a list
```

This action returns information for data files and transaction files. The following table shows the information displayed for data files.

Label	Description	Example
Name	filename	data_file_01
Location	file path	C:\oracle\rsa_data.dat
Used/Occupied Space(M)	Current file size in MB/Total size allocated for this file in MB	12/59
MaxSpace(G)	Maximum space available for this file in GB.	30.000
Warning Thres%	System sends warning alert notice when file reaches this percent of allocated size	70
Critical Thres%	System sends critical alert notice when file reaches this percent of allocated size.	90
Need Optimize	Indicates when to run the optimize command.	Yes/No

The following table shows the information returned for transaction files.

Label	Description	Example
Data Name	filename	archived_trans_files
Location	file path	C:\oracle\ordata
Occupied Space (M)	Current file size in MB	1305
Max Space (G)	Maximum space available for this file in GB	8

- To move a database file to another location, type:

Note: Before you run the move-file action, stop the application server and ensure that there are no client connections to the internal database.

```
rsautl manage-database -a move-file
-f name_of_database_file -n path_to_new_file_location
```

After you run the move-file action, run the list action to verify that the file is in the new location.

- To change the size of a database file, type:

Note: Before you run the change-size action, stop the application server and ensure that there are no client connections to the internal database.

```
rsautl manage-database -a change-size
-f name_of_database_file -s new_database_file_size
```

The new size value must be greater than the current size but smaller than the maximum size. After you run the change-size action, run the list action to verify the new file size.

- To change the maximum size of a database file, type:

Note: Before you run the change-max-size action, stop the application server and ensure that there are no client connections to the internal database.

```
rsautl manage-database -a change-max-size
-f name_of_database_file -s new_database_file_size
```

After you run the change-max-size action, run the list action to verify the new maximum file size.

- To change the critical or warning threshold size of a database file for e-mail alert notices, type:

Note: Before you run the change-threshold action, stop the application server and ensure that there are no client connections to the internal database.

```
rsautl manage-database -a change-threshold
-f name_of_database_file -c new_critical_threshold
-w new_warning_threshold
```

After you run the change-threshold action, run the list action to verify the new file threshold size.

- To optimize a database file, type:

Note: Optimize a database file when the action list result for Need Optimize = Yes.

Note: Before you run the optimize action, shut down the application server and ensure that there are no client connections to the internal database.

```
rsautil manage-database -a optimize
```

After you run the optimize action, run the list action to verify that the file is optimized.

- To see the status of the database, type:

```
rsautil manage-database -a db-status
```

After you run the db-status action, a message appears stating whether the database is available or not.

- To start the database, type:

```
rsautil manage-database -a start-db
```

- To stop the database, type:

```
rsautil manage-database -a stop-db
```

Options for manage-database

The following table describes the options for this utility

Flag	Meaning	Description
-a	action	<p>Specifies the action to perform. Select one of the following:</p> <ul style="list-style-type: none"> list. Displays the status of all database files. move-file. Moves a database file to a new location. change-size. Changes the size of a database file. change-max-size. Changes the maximum size of a database file. change-threshold. Changes the critical or warning threshold size of a database file. optimize. Optimizes the database files for better performance. start-db. Starts the database. stop-db. Stops the database. db-status. Returns a message stating whether the database is available or unavailable.

Flag	Meaning	Description
-c	critical threshold	New critical threshold of a database file. The number you provide indicates the percent of the maximum size of the file. The critical threshold must be higher than the warning threshold. Example: 85.
-f	filename	Filename for a database file. Example: DATA_FILE_01
-h	help	Displays help for this utility.
-m	master password	Specifies the master password for the encrypted properties file.
-n	new location	Full path to new location for a database file. Example: D:/oracle/product/10.2.0
-q	quiet	Execute the action without prompting for confirmation.
-s	new size	New maximum size for a database file. Use only M or G to indicate size. Example: 500M
-v	version	Displays the version and copyright information.
-w	warning threshold	New warning threshold of a database file. The number you provide indicates the percent of the maximum size of the file. The warning threshold must be lower than the critical threshold. Example: 60.

Manage Operations Console Administrators Utility

Use the Manage Operations Console Administrators utility, `manage-oc-administrators`, to add, update, or delete Operations Console administrators.

Note: You must be a Super Admin to use this utility.

Using the Manage Operations Console Administrators Utility

To use `manage-oc-administrators`:

1. Open a new command shell, and change directories to `RSA_AM_HOME/utlils`.
2. Type:

```
rsautl manage-oc-administrators options
```

For relevant options, see the following section, [“Options for `manage-oc-administrators`.”](#)

Important: Although it is possible to enter the Super Admin's password on the command line along with the other options, this creates a potential security vulnerability. RSA recommends that you enter the Super Admin's password only when the utility presents a prompt.

Use the indicated options to perform the following tasks:

- To create a new administrator and associated password, type:

```
rsautl manage-oc-administrators --action create
username password
```

where:

- `username` is the user name for the new administrator.
- `password` is the password for the new administrator.

- To provide a new password for an administrator, type:

```
rsautl manage-oc-administrators --action update
username new_password
```

where:

- `username` is the user name of the administrator.
- `new_password` is the new password for the administrator.

- To delete an existing administrator, type:

```
rsautl manage-oc-administrators
--action delete username
```

where `username` is the user name of the administrator you delete.

- To display a list of all administrators, type:


```
rsautil manage-oc-administrators --action list
```
- To assign an administrator to one or more groups, type:


```
rsautil manage-oc-administrators --action create
username --groups group1, group2
```

 where:
 - *username* is user name of the administrator.
 - *group1, group2* are the names of the groups to which you are assigning the administrator.
- To remove an administrator from one or more groups, type:


```
rsautil manage-oc-administrators --action update
user_name --remove-groups group1, group2
```

 where:
 - *user_name* is user name of the administrator.
 - *group1, group2* are the names of the groups that you are removing the administrator from.

Options for manage-oc-administrators

The following table describes the options for this utility.

Flag	Alternate Flag	Description
-a	--action	Specifies an action to perform. Select one of the following: <ul style="list-style-type: none"> create. Creates a new administrator in the encrypted file. update. Provide an existing administrator with a new password. delete. Deletes an existing administrator from the encrypted file. The last administrator cannot be deleted. list. Displays all administrators in the encrypted file. reload. Reloads all users from the database.
-d	--default-none	Optional. Prevents the user from having a default group association.
-D	--disable-password	Optional. Disables the user's password.

Flag	Alternate Flag	Description
-g	--groups	Optional. Specifies the group name that you assign all Operations Console administrators to. To assign an administrator to more than one group, separate each group name with a comma. Note: If you do not specify a group name, new administrators are added to a default group called OperationsConsole-Administrators. All Operations Console administrators must belong to the same group.
-h	--help	Displays help for this utility.
-n	--not-empty	Optional. Prevents the specified list of groups from having no members.
-p	--password	Administrator's password.
-r	--remove-groups	Optional. Specifies the group name that you want to remove an administrator from. To remove an administrator from more than one group, separate each group name with a comma. The administrator must remain a member of the group that all Operations Console administrators belong to.
-S	--script-mode	Optional. Utility does not prompt for missing arguments. If required arguments are missing, the program fails.
-u	--user	Administrator's user name.
-v	--version	Displays the version and copyright information.
-X	--debug	Displays debug messages.

Manage Secrets Utility

The Manage Secrets utility, `manage-secrets`, exports or imports the encrypted **properties** file that contains the system fingerprint to or from a password-protected file. The exporting feature backs up a secured copy of the **properties** file encrypted by a password provided by the administrator. Using the importing feature, the administrator can unlock the **properties** file for disaster recovery.

Note: The Manage Secrets utility is a password storage tool. This utility does not change the passwords for the services, it simply stores the passwords. It is the responsibility of the user to make sure that the passwords and user names in the **properties** file are kept in synchronization with the passwords set through the services. The encrypted passwords are stored in `RSA_AM_HOME/etc/systemfields.properties`.

Using the Manage Secrets Utility

To use `manage-secrets`:

1. Open a new command shell, and change directories to `RSA_AM_HOME/utlils`.
2. Type:

```
rsautl manage-secrets options
```

For relevant options, see the following section, "[Options for manage-secrets.](#)"

Important: Although it is possible to enter the master password on the command line along with the other options, this creates a potential security vulnerability. RSA recommends that you enter the master password only when the utility presents a prompt.

Use the indicated options to perform the following tasks:

- To export a system fingerprint-encrypted file into a password-protected file, type:

```
rsautl manage-secrets --action export  
--file myfile.exp --file-password file_password
```

where:

- *myfile.exp* is the name of the system fingerprint-encrypted file being exported.
- *file_password* is the password to unlock the file.

- To import a password-protected file that was created by the export command on either the same system or a different system, type:

```
rsautl manage-secrets --action import  
--file myfile.exp --file-password file_password
```

where:

- *myfile.exp* is the name of the password-protected file being imported.
- *file_password* is the password to unlock the file.

- To change a system fingerprint-encrypted file master password to a new value, type:

```
rsautil manage-secrets --action change
--new-password new-master-password
```

- To recover the system fingerprint-encrypted file after the host machine is reconfigured, type:

```
rsautil manage-secrets --action recover
```

- To load a number of keys (in bulk) from a plain text file into an encrypted file, type:

```
rsautil manage-secrets --action load
--file mysecrets.properties
```

where *mysecrets.properties* is the name of the plain text file.

- To display a subset of the stored secrets in the file, type:

```
rsautil manage-secrets --action list
```

By default, this displays only the Command API Client User ID and Password.

- To display a subset of the raw key names (not localized names) to use when setting the values, type:

```
rsautil manage-secrets --action listkeys
```

By default, this displays only the raw key names or the Command API Client User ID and Password.

Note: You can use this option to find the raw key name before changing a value using the set or get commands. The set and get commands accept the raw key name, not the localized name.

- To set a previously stored secret to a specified value, type:

```
rsautil manage-secrets --action set com.rsa.appserver.
admin.password administrator_password
```

where *administrator_password* is the name of the password being set for *com.rsa.appserver.admin.password*.

- To list the current value of a single stored secret by name, type:

```
rsautil manage-secrets
--action get secret.raw.key.name
```

Options for manage-secrets

The following table describes the options for this utility.

Flag	Alternate Flag	Description
-a	--action	<p>Specifies an action to perform. Select one of the following:</p> <p>import. Imports a password-protected file to be system fingerprint encrypted. A file can be imported to the same system or a different system.</p> <p>export. Exports a system fingerprint-encrypted file to a password-protected file. This is used for backup purposes or to transport the managed secrets to a new server node that is being bootstrapped.</p> <p>change. Changes a system fingerprint-encrypted file master password. This option only changes the password that is used by the command line utilities to open the fingerprint-encrypted file. It does not affect the machine fingerprint.</p> <p>recover. Recovers a system fingerprint-encrypted file using the master password. This may be necessary if the host machine is reconfigured with more memory, new IP addresses, or new disks.</p> <p>load. Loads a plain text properties file into an encrypted file.</p> <p>list. Displays a subset of the secrets in the file. By default, this action only displays the CmdClient user name and password.</p> <p>listkeys. Displays a subset of the key names used for setting values. By default, this action only displays the CmdClient user name and password key names.</p> <p>set. Sets a property to a specified value. You must specify the name and value of the property to set. This can also be used to add a new secret in the secure storage.</p> <p>get. Lists the current value for a specified property. You must specify the name of the property to get. This option can be useful for scripting applications.</p>
-f	--file	Name of the password-protected file to import, export, or load.
-h	--help	Displays help for this utility.
-k	--file-password	Password to lock or unlock the file.

Flag	Alternate Flag	Description
-m	--master-password	Master password for the encrypted properties file.
-n	--new-password	New master password for the change action.
-v	--version	Displays the version and copyright information.
-X	--debug	Displays debug messages.

Register Custom Extension Utility

Use the Register Custom Extension utility, `register-custom-extension`, to register custom extensions for existing operations in Credential Manager.

Note: When you submit a batch job, make sure that you write down the job ID to run the Manage Batchjob utility to view, cancel, delete, or view the status of an existing batch job. For more information, see [“Using the Manage Batchjob Utility”](#) on page 275.

Using the Register Custom Extension Utility

To use `register-custom-extension`:

1. Open a new command shell, and change directories to `RSA_AM_HOME/utlils`.
2. Type:

```
rsautil register-custom-extension
```

For relevant options, see the following section, [“Options for `register-custom-extension`.”](#)

3. Enter the master password.

Important: Although it is possible to enter the master password on the command line along with the other options, this creates a potential security vulnerability. RSA recommends that you enter the master password only when the utility presents a prompt.

4. Enter the administrator user name and password.
5. Enter the path and filename of the custom extension property file. See [“Creating a Custom Extension Property File for Workflow Operations”](#) and [“Creating a Custom Extension Property File for Non-Workflow Operations”](#) on page 296. If the custom extension registers successfully, the utility displays a confirmation message.
6. To deploy the routines in the application server, package the routine classes and the spring configuration file in a jar file.

Note: You must name the jar file with the bean definitions of the routines the following: `rsa-components.xml`.

7. To restart the Authentication Manager service:
 - a. Stop the Authentication Manager service.
 - b. Copy the jar file to the following locations:
`RSA_AM_HOME/servers/upload/am-app/APP-INF/lib`
`RSA_AM_HOME/servers/fkhan-pc_server/stage/am-app/m-app/APP-INF/lib`
 - c. Delete the temporary directory located in:
`RSA_AM_HOME/servers/fkhan-pc_server/tmp` directory
 - d. Start the Authentication Manager service.

To use `register-custom-extension` in non-interactive mode:

1. Change directories to **`RSA_AM_HOME/utlils`**.
2. Type:

```
rsautil register-custom-extension -u userID -p password
-m masterpassword -f custom_extension_filename
```

Important: Although it is possible to enter the master password on the command line along with the other options, this creates a potential security vulnerability. RSA recommends that you enter the master password only when the utility presents a prompt.

where:

- *userID* is the user name of the administrator running the utility.
- *password* is the password of the administrator running the utility.
- *masterpassword* is the master password.
- *custom_extension_filename* is the custom extension property file and pathname.

For information about creating prerequisite, preprocessing, or postprocessing routines and deploying routines on the application server, see the *Developer's Guide*.

Options for register-custom-extension

The following table describes the options for this utility.

Flag	Alternate Flag	Description
-f	-file	The custom extension property file and pathname.
-h	--help	Optional. Displays help for this utility.
-I	--interactive	Optional. Runs the utility in interactive mode.
-m	--masterPassword	Master password.
-p	--password	Password of the administrator running the utility.
-u	--userID	User name of the administrator running the utility.
-v	--version	Optional. Displays the version and copyright information.

Creating a Custom Extension Property File for Workflow Operations

To register a custom extension for workflow operations such as requests for enrollment, token, on-demand tokencode service, and user group membership, you must create a custom property file for workflow operations.

To register a custom extension for non-workflow operations, you must create a property file for non-workflow operations.

Non-workflow operations are operations that users can do such as changing user profiles, PINs, or passwords, putting tokens in emergency access mode, changing PINs, activating tokens, synchronizing tokens, and resetting PINs. For more information, see [“Creating a Custom Extension Property File for Non-Workflow Operations”](#) on page 296.

Guidelines for Property Files

The following property file guidelines are the same for workflow and non-workflow property files:

- The filename of the custom extension must be unique and cannot contain spaces.
- The maximum length of the custom extension filename, EXTENSION_NAME, in the property file is 32 characters.
- You can only include the three routine properties, PRE_REQUEST, PRE_PROCESSING, and POST_PROCESSING once in a property file.

- You must include at least one of the three routine properties, PRE_REQUEST, PRE_PROCESSING, or POST_PROCESSING in a property file.

The following property file is an example of how to add a custom extension to the second activity in an enrollment operation that has two approval steps.

```
SELF_SERVICE_ID=ENR
DEFINITION_ID=NEWTA
ACTIVITY_ID=4
EXTENSION_NAME=CUSTOM_EXTN
EXTENSION_DESC= Custom Extension description
PRE_REQUEST=CustomExtensionPreRequestRoutine
PRE_PROCESSING=CustomExtensionPreProcessingRoutine
POST_PROCESSING=CustomExtensionPostProcessingRoutine
```

Explanation of properties in the example:

- SELF_SERVICE_ID – Equals ENR, an enrollment only operation.
- DEFINITION_ID – Equals NEWTA because the enrollment only operation (ENR), in this example, has two approval steps.
- ACTIVITY_ID – Equals 4 because there are 2 approval steps and this example adds a custom extension to the second activity, the second approval step.
- EXTENSION_NAME – Name of the custom extension.
- EXTENSION_DESC – Description of the custom extension.
- PRE_REQUEST – This property corresponds to the prerequisite routine. The value for this property must correspond to the Java Bean ID. If the value is empty, the property is not processed.
- PRE_PROCESSING – This property corresponds to the preprocessing routine. The value for this property must correspond to the Java Bean ID. If the value is empty, the property is not processed.
- POST_PROCESSING – This property corresponds to the postprocessing routine. The value for this property must correspond to the Java Bean ID. If the value is empty, the property is not processed.

The following table shows the mapping between self-service operations, workflow definitions, and workflow activities to use when you create a custom extension property file for workflow operations.

Mapping Between Self-Service Operations, Workflow Definitions, and Workflow Activities

SELF_SERVICE_ID	Self Service Name	Workflow DEFINITION_ID	Workflow Definition Name	Workflow ACTIVITY_ID	Workflow Activity Name
ENR	Enrollment	NEWOA	USR-ENR 1 approval step	1	APPROVE_ENROLLMENT
ENR	Enrollment	NEWTA	USR-ENR 2 approval steps	3 4	APPROVE_ENROLLMENT APPROVE_ENROLLMENT



Mapping Between Self-Service Operations, Workflow Definitions, and Workflow Activities

SELF_SERVIC E_ID	Self Service Name	Workflow DEFINITION_ID	Workflow Definition Name	Workflow ACTIVITY _ID	Workflow Activity Name
ENRHWTKN	Enrollment and New SecurID Token	NEASTWOA	ENR-TKN 1 approval step	36	APPROVE_ENROLLMENT_S ECUREID_TOKEN
ENRHWTKN	Enrollment and New SecurID Token	NEASTWOOAOD	ENR-TKN 1 approval step and 1 distribution step	38	APPROVE_ENROLLMENT_S ECUREID_TOKEN
				39	DISTRIBUTE_SECUREID_TO KEN
ENRHWTKN	Enrollment and New SecurID Token	NEASTWTA	ENR-TKN 2 approvalsteps	46	APPROVE_ENROLLMENT_S ECUREID_TOKEN
				45	APPROVE_ENROLLMENT_S ECUREID_TOKEN
ENRHWTKN	Enrollment and New SecurID Token	NEASTWTAAOD	ENR-TKN 2 approval steps and 1 distribution step	41	APPROVE_ENROLLMENT_S ECUREID_TOKEN
				42	APPROVE_ENROLLMENT_S ECUREID_TOKEN
				43	DISTRIBUTE_SECUREID_TO KEN
ENRSWTKN	Enrollment and New SecurID Token	NEASTWOA	ENR-TKN 1 approval step	36	APPROVE_ENROLLMENT_S ECUREID_TOKEN
ENRSWTKN	Enrollment and New SecurID Token	NEASTWOOAOD	ENR-TKN 1 approval step and 1 distribution step	38	APPROVE_ENROLLMENT_S ECUREID_TOKEN
				39	DISTRIBUTE_SECUREID_TO KEN
ENRSWTKN	Enrollment and New SecurID Token	NEASTWTA	ENR-TKN 2 approval steps	46	APPROVE_ENROLLMENT_S ECUREID_TOKEN
				45	APPROVE_ENROLLMENT_S ECUREID_TOKEN
ENRSWTKN	Enrollment and New SecurID Token	NEASTWTAAOD	ENR-TKN 2 approval steps and 1 distribution step	41	APPROVE_ENROLLMENT_S ECUREID_TOKEN
				42	APPROVE_ENROLLMENT_S ECUREID_TOKEN
				43	DISTRIBUTE_SECUREID_TO KEN
HWTKN	New SecurID Token	NOASTWOA	USR-TKN 1 approval step	54	APPROVE_SECUREID_TOKE N

Mapping Between Self-Service Operations, Workflow Definitions, and Workflow Activities

SELF_SERVIC E_ID	Self Service Name	Workflow DEFINITION_ID	Workflow Definition Name	Workflow ACTIVITY _ID	Workflow Activity Name
HWTKN	New SecurID Token	NOASTWOOAOD	USR-TKN 1 approval step and 1 distribution step	56	APPROVE_SECUREID_TOKE N
				57	DISTRIBUTE_SECUREID_TO KEN
HWTKN	New SecurID Token	NOASTWTA	USR-TKN 2 approval steps	51	APPROVE_SECUREID_TOKE N
				52	APPROVE_SECUREID_TOKE N
HWTKN	New SecurID Token	NOASTWTAAOD	USR-TKN 2 approval stepS and 1 distribution step	49	APPROVE_SECUREID_TOKE N
				50	APPROVE_SECUREID_TOKE N
				47	DISTRIBUTE_SECUREID_TO KEN
SWTKN	New SecurID Token	NOASTWOA	USR-TKN 1 approval step	54	APPROVE_SECUREID_TOKE N
SWTKN	New SecurID Token	NOASTWOOAOD	USR-TKN 1 approval step and 1 distribution step	56	APPROVE_SECUREID_TOKE N
				57	DISTRIBUTE_SECUREID_TO KEN
SWTKN	New SecurID Token	NOASTWTA	USR-TKN 2 approval steps	51	APPROVE_SECUREID_TOKE N
				52	APPROVE_SECUREID_TOKE N
SWTKN	New SecurID Token	NOASTWTAAOD	USR-TKN 2 approval steps and 1 distribution step	49	APPROVE_SECUREID_TOKE N
				50	APPROVE_SECUREID_TOKE N
				47	DISTRIBUTE_SECUREID_TO KEN
HWREPEXPTKN	Replacement for Expired SecurID Token	RESTWOA	EXP-TKN 1 approvalstep	6	APPROVE_SECUREID_TOKE N_REPLACEMENT

Mapping Between Self-Service Operations, Workflow Definitions, and Workflow Activities

SELF_SERVIC E_ID	Self Service Name	Workflow DEFINITION_ID	Workflow Definition Name	Workflow ACTIVITY _ID	Workflow Activity Name
HWREPEXPTKN	Replacement for Expired SecurID Token	RESTWAAAOD	EXP-TKN 1 approval step and 1 distribution step	8	APPROVE_SECUREID_TOKEN_REPLACEMENT
				9	DISTRIBUTE_SECUREID_TOKEN
HWREPEXPTKN	Replacement for Expired SecurID Token	RESTWTA	EXP-TKN 2 approval steps	15	APPROVE_SECUREID_TOKEN_REPLACEMENT
				16	APPROVE_SECUREID_TOKEN_REPLACEMENT
HWREPEXPTKN	Replacement for Expired SecurID Token	RESTWTAAOD	EXP-TKN 2 approval steps and 1 distribution step	11	APPROVE_SECUREID_TOKEN_REPLACEMENT
				12	APPROVE_SECUREID_TOKEN_REPLACEMENT
				13	DISTRIBUTE_SECUREID_TOKEN
SWREPEXPTKN	Replacement for Expired SecurID Token	RESTWOA	EXP-TKN 1 approval step	6	APPROVE_SECUREID_TOKEN_REPLACEMENT
SWREPEXPTKN	Replacement for Expired SecurID Token	RESTWAAAOD	EXP-TKN 1 approval step and 1 distribution step	8	APPROVE_SECUREID_TOKEN_REPLACEMENT
				9	DISTRIBUTE_SECUREID_TOKEN
SWREPEXPTKN	Replacement for Expired SecurID Token	RESTWTA	EXP-TKN 2 approval steps	15	APPROVE_SECUREID_TOKEN_REPLACEMENT
				16	APPROVE_SECUREID_TOKEN_REPLACEMENT
SWREPEXPTKN	Replacement for Expired SecurID Token	RESTWTAAOD	EXP-TKN 2 approval steps and 1 distribution step	11	APPROVE_SECUREID_TOKEN_REPLACEMENT
				12	APPROVE_SECUREID_TOKEN_REPLACEMENT
				13	DISTRIBUTE_SECUREID_TOKEN
HWREPLOBRTKN	Replacement for Lost or Broken SecurID Token	RLOBSTWOA	LST-TKN 1 approval step	23	APPROVE_SECUREID_TOKEN_REPLACEMENT

Mapping Between Self-Service Operations, Workflow Definitions, and Workflow Activities

SELF_SERVIC E_ID	Self Service Name	Workflow DEFINITION_ID	Workflow Definition Name	Workflow ACTIVITY _ID	Workflow Activity Name
HWREPLOBRTKN	Replacement for Lost or Broken SecurID Token	RLOBSTWOOAOD	LST-TKN 1 approval step and 1 distribution step	25	APPROVE_SECUREID_TOKE N_REPLACEMENT
				26	DISTRIBUTE_SECUREID_TO KEN
HWREPLOBRTKN	Replacement for Lost or Broken SecurID Token	RLOBSTWTA	LST-TKN 2 approval steps	32	APPROVE_SECUREID_TOKE N_REPLACEMENT
				33	APPROVE_SECUREID_TOKE N_REPLACEMENT
HWREPLOBRTKN	Replacement for Lost or Broken SecurID Token	RLOBSTWTAAOD	LST-TKN 2 approval steps and 1 distribution step	29	APPROVE_SECUREID_TOKE N_REPLACEMENT
				30	APPROVE_SECUREID_TOKE N_REPLACEMENT
				31	DISTRIBUTE_SECUREID_TO KEN
SWREPLOBRTKN	Replacement for Lost or Broken SecurID Token	RLOBSTWOA	LST-TKN 1 approval step	23	APPROVE_SECUREID_TOKE N_REPLACEMENT
SWREPLOBRTKN	Replacement for Lost or Broken SecurID Token	RLOBSTWOOAOD	LST-TKN 1 approval step and 1 distribution step	25	APPROVE_SECUREID_TOKE N_REPLACEMENT
				26	DISTRIBUTE_SECUREID_TO KEN
SWREPLOBRTKN	Replacement for Lost or Broken SecurID Token	RLOBSTWTA	LST-TKN 2 approval steps	32	APPROVE_SECUREID_TOKE N_REPLACEMENT
				33	APPROVE_SECUREID_TOKE N_REPLACEMENT
SWREPLOBRTKN	Replacement for Lost or Broken SecurID Token	RLOBSTWTAAOD	LST-TKN 2 approval steps and 1 distribution step	29	APPROVE_SECUREID_TOKE N_REPLACEMENT
				30	APPROVE_SECUREID_TOKE N_REPLACEMENT
				31	DISTRIBUTE_SECUREID_TO KEN
UPDGRP	User Group Membership	ROCUGMWOA	USR-GRP 1 approval step	18	APPROVE_GROUP_MEMBE RSHIP

Mapping Between Self-Service Operations, Workflow Definitions, and Workflow Activities

SELF_SERVIC E_ID	Self Service Name	Workflow DEFINITION_ID	Workflow Definition Name	Workflow ACTIVITY _ID	Workflow Activity Name
UPDGRP	User Group Membership	ROCUGMWT	USR-GRP 2 approval steps	20	APPROVE_GROUP_MEMBE RSHIP
				21	APPROVE_GROUP_MEMBE RSHIP
ENRSMSTKN	Enrollment and On-Demand Tokencode Service	NEASMSWOA	ENR-SMS 1 approval step	60	APPROVE_ENROLLMENT_S MS_TOKEN
ENRSMSTKN	Enrollment and On-Demand Tokencode Service	NEASMSWTA	ENR-SMS 2 approval steps	62	APPROVE_ENROLLMENT_S MS_TOKEN
				63	APPROVE_ENROLLMENT_S MS_TOKEN
SMSTKN	On-Demand Tokencode Service	NOASMSWOA	SMS-TKN 1 approval step	65	APPROVE_SMS_TOKEN
SMSTKN	On-Demand Tokencode Service	NOASMSWTA	SMS-TKN 2 approval steps	67	APPROVE_SMS_TOKEN
				68	APPROVE_SMS_TOKEN

Creating a Custom Extension Property File for Non-Workflow Operations

A custom extension property file lets you add your own custom extension for non-workflow operations for Credential Manager operations such as changing user profiles, PINs, or passwords, putting tokens in emergency access mode, changing PINs, activating tokens, synchronizing tokens, and resetting PINs.

The following property file is an example of how to add a custom extension to the change password operation.

```

SELF_SERVICE_ID= UPDPWD
EXTENSION_NAME=CUSTOM_EXTN
EXTENSION_DESC= Custom Extension description
PRE_REQUEST=CustomExtensionPreRequestRoutine
PRE_PROCESSING=CustomExtensionPreProcessingRoutine
POST_PROCESSING=CustomExtensionPostProcessingRoutine
    
```

Explanation of properties in the example:

- **SELF_SERVICE_ID** – The self-service ID is UPDPWD, the change password operation, because this example adds a custom extension to the change password operation.
- **EXTENSION_NAME** – Name of the custom extension.
- **EXTENSION_DESC** – Description of the custom extension.

- **PRE_REQUEST** – This property corresponds to the prerequisite routine. The value for this property must correspond to the Java Bean ID. If the value is empty, the property is not processed.
- **PRE_PROCESSING** – This property corresponds to the preprocessing routine. The value for this property must correspond to the Java Bean ID. If the value is empty, the property is not processed.
- **POST_PROCESSING** – This property corresponds to the postprocessing routine. The value for this property must correspond to the Java Bean ID. If the value is empty, the property is not processed.

The following table lists the self-service IDs to use when you create a custom extension property file for non-workflow operations.

Non-Workflow Self-Service Operations

SELF_SERVICE_ID	Self-Service Name
UPDATT	Request for change attributes
EAMPLATKN	Request to place token in emergency access mode
CHGTKNPIN	Change token PIN
ACTTKN	Activate token
UPDPWD	Request for change password
ACTREPTKN	Activate replacement token
RESYNCTKN	Resynchronize SecurID token
RESTKNPIN	Reset token PIN
CHGSMSPIN	Change on-demand tokencode PIN delivered by mobile device
EASMSTKN	Emergency access on-demand tokencode delivered by mobile device
UPDSMSTKN	Update on-demand tokencode delivered by mobile device

Set Trace Utility

Use the Set Trace utility, `set-trace`, to enable and disable tracing at a more detailed level than you can using the Security Console. Because this utility is integrated with `log4j`, the Java-based logging utility, you can adjust the levels for third-party components, such as `hibernate`, `spring`, and `struts`.

By default, all trace information for Authentication Manager and third-party components is recorded in the default trace log file, **`imsTrace.log`**.

Important: The diagnostic monitors are resource intensive. Do not use all of them at the same time. Limit their use just to the area of the code that you are tracing, and remove the monitor as soon as you are finished.

To use `set-trace`:

1. Open a new command shell, and change directories to **`RSA_AM_HOME/utils`**.
2. Type:

```
rsutil set-trace options
```

For relevant options, see the following section, "[Options for `set-trace`.](#)"

Important: Although it is possible to enter the administrator user-id and password on the command line along with the other options, this creates a potential security vulnerability. RSA recommends that you enter the user-id and password only when the utility presents a prompt.

Use the indicated options to perform the following tasks:

- To enable node-level tracing, type:

```
rsutil set-trace --node
--category trace.com.rsa.ims.authn --level VERBOSE
```

where:

- `trace.com.rsa.ims.authn` is the name of the trace category.
- `VERBOSE` is the logging level.

- To enable instance-level tracing, type:

```
rsutil set-trace --category net.sf.hibernate
--level VERBOSE
```

where:

- `net.sf.hibernate` is the name of the trace category.
- `VERBOSE` is the logging level.

- To disable instance-level tracing, type:

```
rsutil set-trace --category net.sf.hibernate --remove
```

where `net.sf.hibernate` is the name of the trace category to disable.

- To enable tracing only on a server node, type:

```
rsutil set-trace --category org.springframework
--level VERBOSE --node
```

where:

- *org.springframework* is the name of the trace category.
- *VERBOSE* is the logging level.

Note: You cannot run set-trace on a replica instance to modify log categories. To modify log categories on a replica instance, run set-trace on the primary instance and specify the replica instance using the --instance option.

- To enable tracing on a replica instance.

```
rsutil set-trace --category org.springframework
--level VERBOSE --instance replica_instance_name --node
```

where:

- *org.springframework* is the name of the trace category.
- *VERBOSE* is the logging level.
- *replica_instance_name* is the name of the replica instance.

- To enable a diagnostic monitor, type:

```
rsutil set-trace --diagnostic AuthnBeforeMonitor
```

where *AuthnBeforeMonitor* is the name of the diagnostic monitor to enable.

- To disable a diagnostic monitor, type:

```
rsutil set-trace --diagnostic AuthnBeforeMonitor
--remove
```

where *AuthnBeforeMonitor* is the name of the diagnostic monitor to disable.

- To display current settings, type:

```
rsutil set-trace
```

- To list the categories that can be set, type:

```
rsutil set-trace --listCategory
```

- To list the diagnostic monitors that can be set, type

```
rsutil set-trace --listDiagnostic
```

- To set org.apache.commons.lang to the same level as the parent (org.apache) level, type:

```
rsutil set-trace --category org.apache.commons.lang
--level NONE --user-id user --password password
```

where:

- *user* is the administrator's user ID.
- *password* is the administrator's password.

Options for set-trace

The following table describes the options for this utility.

Flag	Alternate Flag	Description
-c	--category	Specifies the name of the trace category to set or remove.
-d	--diagnostic	Specifies the diagnostic monitor to use. For options, see “Diagnostic Monitors for set-trace” on page 301.
-h	--help	Displays help for this utility.
-i	--instance	Specifies the instance name where log categories are changed. For example, --instance <i>instance_name</i> .
-l	--level	Specify one of the following logging levels: <hr/> Note: The logging level can be either in uppercase or lowercase. For example, either VERBOSE or verbose. <hr/> VERBOSE. Captures all fatal, error, warning, and information messages, as well as log messages associated with minor and frequently occurring but otherwise normal events. INFO. Captures all fatal, error, and warning messages, as well as log messages for significant events in the normal life cycle of the application. WARN. Captures all fatal and error messages, as well as log messages for minor problems while the application is running. ERROR. Captures all fatal messages, as well as error conditions that must be addressed, but may not necessarily cause the application to crash. FATAL. Captures only log messages that imply the imminent crash of the application or the relevant subcomponent. These messages require immediate attention. NONE. Sets logging to the same level as the parent. For example, org.springframework is a parent of org.springframework.transaction. If org.springframework.transaction is set to none, it has the same logging level as org.springframework.
-n	--node	Set at the server node level instead of instance level.
-p	--password	Administrator's password.
-r	--remove	Disable the trace setting for the specified category.
-s	--listCategory	Lists the categories that can be set.

Flag	Alternate Flag	Description
-t	--listDiagnostic	Lists the diagnostic monitors that can be set.
-u	--user-id	Administrator's user ID.
-v	--version	Displays the version and copyright information.

Diagnostic Monitors for set-trace

The following table describes the diagnostic monitors for set-trace.

Note: Each monitor has a “before” and “after” version. The before version traces the entrance to the methods, and the after version traces the return from the methods.

Name	Instrumented Packages	Description
AuthnBeforeMonitor AuthnAfterMonitor	com.rsa.authn	The command and internal Authentication Manager components. Provides authentication features for use by clients, such as web agents and the Authentication Manager Configuration Management Service.
AppCoreCommandBeforeMonitor AppCoreCommandAfterMonitor	com.rsa.command	Handles the processing of commands (Public API) being delivered to the Authentication Manager (or product) system. This includes the parsing of the command and the invocation of the proper command object. Tracing is for the internal business tier APIs and the public APIs and their implementation.

User Groups and Token Bulk Requests Utility

Use the User Groups and Token Bulk Requests utility, `import-bulk-request`, to import bulk requests for new or additional token and user group membership requests. Credential Manager, a feature of Authentication Manager runs this task as a batch job.

Note: You need provisioning to execute the User Groups and Token Bulk Requests utility. The provisioning option is included with the Enterprise Server license. If you want to use this utility, you must upgrade to the Enterprise Server license.

When you run the User Groups and Token Bulk Requests utility, the time that it takes for bulk request operations to complete depends on the number of requests in the comma-separated value (CSV) input file. If you have a large number of bulk requests in your CSV input file, you may get time-out errors.

Note: When you submit a batch job, make sure that you write down the job ID to run the Manage Batchjob utility to view, cancel, delete, or view the status of an existing batch job. For more information, see [“Using the Manage Batchjob Utility”](#) on page 275.

Using the User Groups and Token Bulk Requests Utility

Note: You must run the `import-bulk-request` utility from the same system on which the input file resides.

To use `import-bulk-request`:

1. Open a new command shell, and change directories to `RSA_AM_HOME/utlils`.
2. Type:

```
rsautil import-bulk-request option flags
```

For relevant options, see the following section, [“Options for `import-bulk-request`.”](#)

Important: Although it is possible to enter the master password on the command line along with the other options, this creates a potential security vulnerability. RSA recommends that you enter the master password only when the utility presents a prompt.

Examples:

- To import bulk token requests, type:

```
rsautil import-bulk-request -u jdoe -p switz -m
skimtev -f C:\TokenRequests.csv -r TOKEN
```
- To import bulk on-demand tokencode service requests using the mobile device delivery method, type:

```
rsautil import-bulk-request -d SMS
```
- To import bulk user group membership change requests, type:

```
rsautil import-bulk-request -u jdoe -p switz -m
skimtev -f C:\GroupRequests.csv -r GROUP
```

Options for import-bulk-request

The following table describes the options for this utility.

Flag	Alternate Flag	Description
-h	--help	Optional. Displays help for this utility.
-I	--interactive	Optional. Runs the utility in interactive mode.
-m	--masterPassword	Master password.
-p	--password	Password of the administrator running the utility.
-u	--userID	User name of the administrator running the utility.
-v	--version	Optional. Displays the version and the copyright information.

Batch Options

-d	--deliveryMethod	<p>Optional. The delivery method, mobile device or e-mail, for on-demand tokencode service bulk requests.</p> <p>When you import on-demand tokencode service bulk requests, the delivery method follows these guidelines:</p> <ul style="list-style-type: none"> • You can use only one delivery method for each on-demand tokencode service bulk request. • If you configure one delivery method for the on-demand tokencode service, either mobile device or e-mail, then all on-demand tokencode service bulk requests use that delivery method. • If you configure both mobile device and e-mail as the delivery methods for the on-demand tokencode service, then you can set the delivery method for on-demand tokencode service bulk requests using the (-d) option flag. If you do not use the option flag, the default delivery method is e-mail. • If the CSV file does not contain any on-demand tokencode service bulk requests, the delivery method (-d) option flag is ignored.
----	------------------	---

Flag	Alternate Flag	Description
-f	--file	Request input file (CSV) with pathname and filename. Can have multiple --files.
-r	--requestType	The request type, token, or group.

Creating Input Files for Bulk Requests

You must create a CSV input file that lists all of the bulk requests that you want to input into Credential Manager using the User Groups and Token Bulk Requests utility. The CSV (also known as a comma-separated list or comma-separated variables) file format is a file type that stores tabular data.

To create a CSV input file:

- Do one of the following:
 - Create an ASCII text file. (Separate header information and request information by commas.)
 - Use spreadsheet software, such as Microsoft Excel, and enter the values in each column of the spreadsheet.

Note: You must create a separate CSV input file for token requests and a separate CSV input file for user group membership requests. You cannot put token requests and user group membership requests in the same CSV input file.

- Enter header information (attribute names) for requests, on the first line, separated by commas, or in different columns, if you use a spreadsheet.
- Enter request information (attribute values) in the second and following lines, separated by commas, or in different columns, if you use a spreadsheet. Each line, after the header, is a separate request.

Note: The request information (attribute values) must follow the same order as the header information (attribute names) in each line of the input file.

For information about attribute names and attribute values, see the following section, [“CSV Format for Token Requests Input File”](#) and [“CSV Format for User Group Membership Requests Input File”](#) on page 306.

- Do one of the following:
 - Save the text file with extension .csv.
 - If you use spreadsheet software, save the input file as a .csv file.
- Run the import-bulk-request utility. For information, see [“Using the User Groups and Token Bulk Requests Utility”](#) on page 302.

CSV Format for Token Requests Input File

The following table lists the attribute names (header information) and attributes values (request information) for the token requests input file.

Attribute Names	Attribute Values
USER_ID	Logon ID of the user for whom the bulk request is created.
IDENTITY_SOURCE_NAME	Identity source name of the user for whom the bulk request is created. Note: Do not use the user-friendly display name.
TOKEN_TYPE	Use the token type name that appears on the Security Console Self-Service Configuration - Manage Tokens page. Note: Use On-Demand Tokencodes to create bulk requests for the on-demand tokencode service. You cannot send on-demand tokencodes to users using bulk requests.

Sample CSV Input File for Token Requests

```

USER_ID, IDENTITY_SOURCE_NAME, TOKEN_TYPE
JSmith, InternalDatabase, Key Fob
SDoe, InternalDatabase, Standard Card
PPorter, InternalDatabase, PIN Pad
FKing, InternalDatabase, SecurID 800
LMathews, InternalDatabase, Desktop
RKapur, InternalDatabase, Blackberry
DCohen, InternalDatabase, Palm
RBouvier, InternalDatabase, Pocket PC
ADoyle, InternalDatabase, Toolbar
GWilliams, InternalDatabase, CT KIP
TKennedy, InternalDatabase, On Demand Tokencode Service

```

CSV Format for User Group Membership Requests Input File

The following table lists the attribute names (header information) and attributes values (request information) for the user group membership requests input file.

Attribute Names	Attribute Values
USER_ID	Logon ID of the user for whom the bulk request is created.
IDENTITY_SOURCE_NAME	Identity source name of the user for whom the bulk request is created. Note: Do not use the user-friendly display name.
USER_GROUP_NAME	User group name. You can request membership in more than one group by using “;” for a delimiter. Note: You must use the exact name for each user group member. Do not use the user-friendly display name.

Sample CSV Input File for User Group Membership Requests

```

USER_ID,IDENTITY_SOURCE_NAME,USER_GROUP_NAME
GWilliams,InternalDatabase,Managers
TKennedy,InternalDatabase,Managers; HR
DCohen,InternalDatabase,Doc
RBouvier,InternalDatabase,Customers
    
```

Log Files for Bulk Requests

A log file is created for each bulk request batch job. View the log file to verify that the batch job successfully created all requests in the input file.

The location of the log file is:

RSA_AM_HOME/server/logs/CLU

where ***RSA_AM_HOME*** is the name of the directory where you installed Authentication Manager.

The format for the log file is:

- For bulk token requests:
UCM Requests_BulkToken_YYYY_MM_DD_HH_MIN_SEC.log
- For bulk user group membership requests:
UCM Requests_BulkGroup_YYYY_MM_DD_HH_MIN_SEC.log

PIN and Protection of Distribution Files

The CSV output file stores PINs and passwords for protection of distribution files in text format. There is no protection of passwords or PINs in this file. Administrators can use this file to send PINs and passwords to users by e-mail. The CSV output file has the same name as the corresponding log file with extension .csv and will be generated in the same location as the log file.

For tokens with PINs, the system generates PINs to protect distribution files. The system uses the setting to password protect distribution files set by the administrator.

For tokens without PINs, that have one-factor authentication, the system protects distribution files with passwords (8 alphanumeric characters).

Verify Archive Log Utility

Use the Verify Archive Log utility, `verify-archive-log`, to verify that an archived log is intact and not changed. Use this utility to audit the integrity of an archived log file.

Using the Verify Archive Log Utility

To use `verify-archive-log`:

1. Open a new command shell, and change directories to `RSA_AM_HOME/utlils`.
2. Type:

```
rsautl verify-archive-log options
```

For relevant options, see the following section, "[Options for verify-archive-log.](#)"

To verify an archived log file, type:

```
rsautl verify-archive-log --file archive_log_filename
```

where `archive_log_filename` is the name of the archived log file to verify.

Important: Although it is possible to enter the master password on the command line along with the other options, this creates a potential security vulnerability. RSA recommends that you enter the master password only when the utility presents a prompt.

Options for `verify-archive-log`

The following table describes the options for this utility.

Flag	Alternate Flag	Description
-h	--help	Displays help for this utility.
-X	--debug	Displays debug messages.
-v	--version	Displays the version and copyright information.



Flag	Alternate Flag	Description
-m	--master-password	Master password of the encrypted properties file.
-f	--file	Archived log file to verify.

E

Updating Server IP Addresses and Names

This appendix contains information on changing the IP address or hostname of an RSA Authentication Manager server. You use the Update Instance Nodes utility to perform these tasks.

Update Instance Nodes Utility

If you have installed and configured a deployment in a lab environment and want to move that deployment to a corporate network, use the following procedures to change the IP addresses or fully qualified domain names (FQDN) to correspond to the corporate network scheme.

Use the Update Instance Nodes utility, `update-instance-node`, to register a changed IP address or FQDN on:

- The primary instance
- The primary instance when the internal database is hosted on a separate machine
- A server node
- A hardware load balancer
- The internal database for the primary instance when it is hosted on a separate machine
- A replica instance
- A replica instance when the internal database is hosted on a separate machine
- The internal database for a replica instance when it is hosted on a separate machine

Using the Update Instance Nodes Utility

To use `update-instance-node`:

1. On the appropriate machine, open a new command shell, and change directories to `RSA_AM_HOME/utlils`.
2. Type:

```
rsautl update-instance-node options
```

For relevant options, see the following section, "[Options for `update-instance-node`](#)."

For example, to register a changed IP address of an instance in the deployment on the primary instance, you specify the following options:

```
rsautil update-instance-node
--old-host Current_IP_Address
--new-host New_IP_Address
--instance primary
```

where:

- *Current_IP_Address* is the current IP address of the instance, for example, 192.168.1.1.
- *New_IP_Address* is the new IP address of the instance, for example, 192.168.200.245.

Important: Although it is possible to enter the master password on the command line along with the other options, this creates a potential security vulnerability. RSA recommends that you enter the master password only when the utility presents a prompt.

Options for update-instance-node

The following table lists the options for this utility.

Flag	Alternate Flag	Description
-h	--help	Displays help for this utility.
-i	--instance	Specifies the type of instance where the IP address or FQDN is being changed. Select one of the following: <ul style="list-style-type: none"> app-server-only. Specifies the application server instance. database-only. Specifies the internal database host when the internal database is hosted on a separate machine. primary. Specifies the application server and database server. If a server node contains a software proxy, this also specifies the proxy server. proxy-server-only. Specifies the hardware or software proxy. radius-only. Specifies the RADIUS server host when RADIUS is hosted on a separate machine.
-m	--master-password	Specifies the master password for the encrypted properties file.

Flag	Alternate Flag	Description
-n	--new-host	Specifies the new host address of the server node.
-o	--old-host	Specifies the current host address of the server node.
-v	--version	Displays the version and copyright information.

Changing an IP Address or a Fully Qualified Domain Name on a Standalone Deployment

A standalone deployment consists of the application server and internal database installed on the same host machine.

To change the IP address:

1. On the host machine, stop Authentication Manager. Do not stop the internal database.
2. Run the Update Instance Nodes utility to change the current IP address to a new IP address:
 - a. On the host machine, open a new command shell, and change directories to ***RSA_AM_HOME/utlils***.
 - b. Type:


```
rsautil update-instance-node
--old-host Current_IP_Address
--new-host New_IP_Address
--instance primary
```

 where:
 - *Current_IP_Address* is the current IP address of the host, for example, 192.168.1.1.
 - *New_IP_Address* is the new IP address of the host, for example, 192.168.200.245.
3. On the host machine, change the IP address in the host machine network configuration, and update any required DNS tables.
For more information, see your operating system documentation.
4. Restart the host machine.
5. On the host machine, start Authentication Manager.

To change the FQDN:

1. On the host machine, stop Authentication Manager. Do not stop the internal database.
2. Run the Update Instance Nodes utility to change the current FQDN to a new FQDN:
 - a. On the host machine, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:


```
rsautil update-instance-node
--old-host Current_FQDN
--new-host New_FQDN
--instance primary
```

 where:
 - *Current_FQDN* is the current FQDN of the host, for example, standalone.mydomain.com.
 - *New_FQDN* is the new FQDN of the host, for example, new.standalone.mydomain.com.
3. On the host machine, change the FQDN of the host machine.
For more information, see your operating system documentation.
4. Restart the host machine.
5. On the host machine, start Authentication Manager.

Changing the IP Address or a Fully Qualified Domain Name on a Cluster Deployment

A cluster deployment consists of the primary instance (the application server, the internal database server, and an optional RADIUS server) and one or more server nodes. RADIUS and the internal database can be installed on the same machine as the application server or each on a separate machine. A cluster deployment can also include a hardware load balancer.

Changing the IP Address on a Cluster Deployment

- To change the IP address of the primary instance:
- To change the IP address of a server node:
- To change the IP address of a hardware load balancer:
- To change the IP address of the primary instance with the internal database hosted on a separate machine:
- To change the IP address of a machine hosting the internal database:
- To change the IP address of the RADIUS primary server:

To change the IP address of the primary instance:

1. On each server node, stop Authentication Manager.
2. On the primary instance, stop Authentication Manager. Do not stop the internal database.
3. On the primary instance, run the Update Instance Nodes utility to change the current IP address of the primary instance to a new IP address:
 - a. On the primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:

```
rsautil update-instance-node
--old-host Current_Primary_Instance_IP_Address
--new-host New_Primary_Instance_IP_Address
--instance primary
```

where:
 - *Current_Primary_Instance_IP_Address* is the current IP address of the primary host, for example, 192.168.1.1.
 - *New_Primary_Instance_IP_Address* is the new IP address of the primary host, for example, 192.168.200.245.
4. On each server node, run the Update Instance Nodes utility to change the current IP address of the primary instance to a new IP address:
 - a. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:

```
rsautil update-instance-node
--old-host Current_Primary_Instance_IP_Address
--new-host New_Primary_Instance_IP_Address
--instance primary
```

where:
 - *Current_Primary_Instance_IP_Address* is the current IP address of the primary instance host, for example, 192.168.1.1.
 - *New_Primary_Instance_IP_Address* is the new IP address of the primary instance host, for example, 192.168.200.245.
5. On the primary instance, change the IP address in the host machine network configuration, and update any required DNS tables.
For more information, see your operating system documentation.
6. Restart each host machine.
7. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.
8. If your deployment includes a RADIUS server, see [“To change the IP address of the RADIUS primary server:”](#) on page 319.

To change the IP address of a server node:

1. On each server node, stop Authentication Manager.
2. On the primary instance, stop Authentication Manager. Do not stop the internal database.
3. On each server node, run the Update Instance Nodes utility to change the current IP address of the server node to a new IP address:

- a. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
- b. Type:

```
rsautil update-instance-node
--old-host Current_Server_Node_IP_Address
--new-host New_Server_Node_IP_Address
--instance app-server-only
```

where:

- *Current_Server_Node_IP_Address* is the current IP address of the server node host, for example, 192.168.1.1.
- *New_Server_Node_IP_Address* is the new IP address of the server node host, for example, 192.168.200.245.

4. On the primary instance, run the Update Instance Nodes utility to change the current IP address of the server node to the new IP address:
- a. On the primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.

- b. Type:

```
rsautil update-instance-node
--old-host Current_Server_Node_IP_Address
--new-host New_Server_Node_IP_Address
--instance app-server-only
```

where:

- *Current_Server_Node_IP_Address* is the current IP address of the server node host, for example, 192.168.1.1.
- *New_Server_Node_IP_Address* is the new IP address of the server node host, for example, 192.168.200.245.

5. On the server node that is getting a new IP address, change the IP address in the host machine network configuration, and update any required DNS tables.
For more information, see your operating system documentation.
6. Restart each host machine.
7. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.

To change the IP address of a hardware load balancer:

1. On each server node, stop Authentication Manager.
2. On the primary instance, stop Authentication Manager. Do not stop the internal database.
3. On the primary instance, run the Update Instance Nodes utility to change the current hardware load balancer IP address to a new IP address:
 - a. On the primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utls***.
 - b. Type:

```
rsautil update-instance-node
--old-host Current_Loadbalancer_IP_Address
--new-host New_Loadbalancer_IP_Address
--instance proxy-server-only
```

where:
 - *Current_Loadbalancer_IP_Address* is the current IP address of the load balancer, for example, 192.168.1.1.
 - *New_Loadbalancer_IP_Address* is the new IP address of the load balancer, for example, 192.168.200.245.
4. On the hardware load balancer machine, change the IP address in the machine network configuration, and update any required DNS tables. Restart the machine. For more information, see your hardware load balancer documentation.
5. Restart each host machine.
6. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.

To change the IP address of the primary instance with the internal database hosted on a separate machine:

1. On each server node, stop Authentication Manager
2. On the primary instance, stop Authentication Manager. Do not stop the internal database.
3. On the primary instance, run the Update Instance Nodes utility to change the current IP address of the primary instance to a new IP address:
 - a. On the primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.

b. Type:

```
rsautil update-instance-node
--old-host Current_Primary_Instance_IP_Address
--new-host New_Primary_Instance_IP_Address
--instance primary
```

where:

- *Current_Primary_Instance_IP_Address* is the current IP address of the primary instance host, for example, 192.168.1.1.
- *New_Primary_Instance_IP_Address* is the new IP address of the primary instance host, for example, 192.168.200.245.

4. On each server node, run the Update Instance Nodes utility to change the current IP address of the primary instance to a new IP address:
 - a. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.

b. Type:

```
rsautil update-instance-node
--old-host Current_Primary_Instance_IP_Address
--new-host New_Primary_Instance_IP_Address
--instance primary
```

where:

- *Current_Primary_Instance_IP_Address* is the current IP address of the primary host, for example, 192.168.1.1.
- *New_Primary_Instance_IP_Address* is the new IP address of the primary host, for example, 192.168.200.245.

5. On the internal database host, run the Update Instance Nodes utility to change the current IP address of the primary instance to a new IP address:
 - a. On the internal database host, open a new command shell, and change directories to ***RSA_AM_HOME/utils***.
 - b. Type:

```
rsautil update-instance-node
--old-host Current_Primary_Instance_IP_Address
--new-host New_Primary_Instance_IP_Address
--instance primary
```

where:
 - *Current_Primary_Instance_IP_Address* is the current IP address of the primary host, for example, 192.168.1.1.
 - *New_Primary_Instance_IP_Address* is the new IP address of the primary host, for example, 192.168.200.245.
6. On the primary instance, change the IP address in the host machine network configuration, and update any required DNS tables.
For more information, see your operating system documentation.
7. Restart each host machine.
8. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.

To change the IP address of a machine hosting the internal database:

1. On each server node, stop Authentication Manager.
2. On the primary instance, stop Authentication Manager. Do not stop the internal database.
3. On the internal database host, run the Update Instance Nodes utility to change the current IP address of the machine hosting the internal database to the new IP address:
 - a. On the internal database host, open a new command shell, and change directories to ***RSA_AM_HOME/utils***.
 - b. Type:

```
rsautil update-instance-node
--old-host Current_Database_Host_IP_Address
--new-host New_Database_Host_IP_Address
--instance database-only
```

where:
 - *Current_Database_Host_IP_Address* is the current IP address of the internal database host, for example, 192.168.1.1.
 - *New_Database_Host_IP_Address* is the new IP address of the internal database host, for example, 192.168.200.245.

4. On the primary instance, run the Update Instance Nodes utility to change the current IP address of the internal database host to the new IP address:
 - a. On the primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utls***.
 - b. Type:


```
rsautil update-instance-node
--old-host Current_Database_Host_IP_Address
--new-host New_Database_Host_IP_Address
--instance database-only
```

 where:
 - *Current_Database_Host_IP_Address* is the current IP address of the machine hosting the internal database, for example, 192.168.1.1.
 - *New_Database_Host_IP_Address* is the new IP address of the machine hosting the internal database, for example, 192.168.200.245.
5. On each server node, run the Update Instance Nodes utility to change the current IP address of the internal database host to the new IP address:
 - a. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utls***.
 - b. Type:


```
rsautil update-instance-node
--old-host Current_Database_Host_IP_Address
--new-host New_Database_Host_IP_Address
--instance database-only
```

 where:
 - *Current_Database_Host_IP_Address* is the current IP address of the machine hosting the internal database, for example, 192.168.1.1.
 - *New_Database_Host_IP_Address* is the new IP address of the machine hosting the internal database, for example, 192.168.200.245.
6. On the internal database host, change the IP address in the host machine network configuration, and update any required DNS tables.
For more information, see your operating system documentation.
7. Restart each host machine.
8. On the internal database host, start the database.
9. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.

To change the IP address of the RADIUS primary server:

Note: If your deployment includes a RADIUS primary server that is hosted on a separate machine, use the following procedure to change the IP address of the RADIUS server.

If your deployment includes a RADIUS primary server that is hosted on the primary instance, complete [step 8](#) through [step 9](#) in the following procedure after you have changed the IP address of the primary instance.

1. On each server node, stop Authentication Manager.
2. On the primary instance, stop Authentication Manager. Do not stop the internal database.
3. On the RADIUS server host, run the Update Instance Nodes utility to change the current IP address of the machine hosting the RADIUS server to the new IP address:
 - a. On the RADIUS server host, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:

```
rsautil update-instance-node
--old-host Current_Radius_Host_IP_Address
--new-host New_Radius_Host_IP_Address
--instance radius-only
```

where:

- *Current_Radius_Host_IP_Address* is the current IP address of the RADIUS server host, for example, 192.168.1.1.
- *New_Radius_Host_IP_Address* is the new IP address of the RADIUS server host, for example, 192.168.200.245.

4. On the primary instance, run the Update Instance Nodes utility to change the current IP address of the RADIUS server host to the new IP address:
 - a. On the primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:

```
rsautil update-instance-node
--old-host Current_Radius_Host_IP_Address
--new-host New_Radius_Host_IP_Address
--instance radius-only
```

where:

- *Current_Radius_Host_IP_Address* is the current IP address of the machine hosting the RADIUS server, for example, 192.168.1.1.
- *New_Radius_Host_IP_Address* is the new IP address of the machine hosting the RADIUS server, for example, 192.168.200.245.

5. On each server node, run the Update Instance Nodes utility to change the current IP address of the RADIUS server host to the new IP address:
 - a. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utls***.
 - b. Type:


```
rsautil update-instance-node
--old-host Current Radius Host IP Address
--new-host New Radius Host IP Address
--instance radius-only
```

 where:
 - *Current Radius Host IP Address* is the current IP address of the machine hosting the RADIUS server, for example, 192.168.1.1.
 - *New Radius Host IP Address* is the new IP address of the machine hosting the RADIUS server, for example, 192.168.200.245.
6. On the RADIUS server host, change the IP address in the host machine network configuration, and update any required DNS tables.
For more information, see your operating system documentation.
7. Restart each host machine.
8. On the RADIUS server host, perform the following steps to update the RADIUS server:
 - a. On the RADIUS server host, ensure that the RADIUS server is stopped.
 - b. On the RADIUS server host, open a new command shell, and change directories to one of the following:
 - On Windows: ***RSA_AM_HOME/radius/Service***.
 - On UNIX: ***RSA_AM_HOME/radius***.
 - c. Type:


```
sbrsetuptool -identity PRIMARY
-secret replication_secret
```

 where *replication_secret* is the replication secret that you are using for all of the replicas. You can use the same replication secret that you used before changing the IP address.
 - d. On the RADIUS server host, start the RADIUS server.
9. On each RADIUS replica server, perform the following steps:
 - a. On the RADIUS replica server host, ensure the RADIUS server is stopped.
 - a. On the primary instance host, open a new command shell, and change directories to one of the following:
 - On Windows: ***RSA_AM_HOME/radius/Service***.
 - On UNIX: ***RSA_AM_HOME/radius***.
 - b. Copy the **replica.ccpmkg** file from this location to the RADIUS replica server host.

- c. On the RADIUS replica server host, open a new command shell, and change directories to one of the following:
 - On Windows: *RSA_AM_HOME/radius/Service*.
 - On UNIX: *RSA_AM_HOME/radius*.
- d. Type:

```
sbrsetuptool -identity REPLICA
-radpath replica.ccpmkg_path
```

where *replica.ccpmkg_path* is the absolute path to the **replica.ccpmkg** file that you copied from the primary instance host to the RADIUS replica server host.
- e. Restart the RADIUS server on the replica host.

Changing the FQDN on a Cluster Deployment

- [To change the FQDN of the primary instance:](#)
- [To change the FQDN of a server node:](#)
- [To change the FQDN of a hardware load balancer:](#)
- [To change the FQDN of a machine hosting the internal database:](#)
- [To change the FQDN of the primary instance with the internal database hosted on a separate machine:](#)
- [To change the FQDN of the RADIUS primary server:](#)

To change the FQDN of the primary instance:

1. On each server node, stop Authentication Manager.
2. On the primary instance, stop Authentication Manager. Do not stop the internal database.
3. On the primary instance, run the Update Instance Nodes utility to change the current FQDN of the primary instance to the new FQDN:
 - a. On the primary instance, open a new command shell, and change directories to *RSA_AM_HOME/utills*.
 - b. Type:

```
rsautil update-instance-node
--old-host Current_Primary_Instance_FQDN
--new-host New_Primary_Instance_FQDN
--instance primary
```

where:

- *Current_Primary_Instance_FQDN* is the current FQDN of the primary instance host, for example, *primaryinstance.mydomain.com*.
- *New_Primary_Instance_FQDN* is the new FQDN of the primary instance host, for example, *newprimaryinstance.mydomain.com*.

4. On each server node, run the Update Instance Nodes utility to change the current FQDN of the primary instance to the new FQDN:
 - a. On each server node host, open a new command shell, and change directories to ***RSA_AM_HOME/utils***.
 - b. Type:


```
rsautil update-instance-node
--old-host Current_Primary_Instance_FQDN
--new-host New_Primary_Instance_FQDN
--instance primary
```

 where:
 - *Current_Primary_Instance_FQDN* is the current FQDN of the primary instance host, for example, primaryinstance.mydomain.com.
 - *New_Primary_Instance_FQDN* is the new FQDN of the primary instance host, for example, newprimaryinstance.mydomain.com.
5. On the primary instance host, change the FQDN of the host machine. For more information, see your operating system documentation.
6. Restart each host machine.
7. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.
8. If your deployment includes a RADIUS server, see [“To change the FQDN of the RADIUS primary server:”](#) on page 327.

To change the FQDN of a server node:

1. On each server node, stop Authentication Manager.
2. On the primary instance, stop Authentication Manager. Do not stop the internal database.
3. On each server node host, run the Update Instance Nodes utility to change the current FQDN of a server node to the new FQDN:
 - a. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utils***.
 - b. Type:


```
rsautil update-instance-node
--old-host Current_Server_Node_FQDN
--new-host New_Server_Node_FQDN
--instance app-server-only
```

 where:
 - *Current_Server_Node_FQDN* is the current FQDN of the server node host, for example, server1.mydomain.com.
 - *New_Server_Node_FQDN* is the new FQDN of the server node host, for example, new.server1.mydomain.com.

4. On the primary instance host, run the Update Instance Nodes utility to change the current FQDN of a server node to the new FQDN:
 - a. On the primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utils***.
 - b. Type:

```
rsautil update-instance-node
--old-host Current_Server_Node_FQDN
--new-host New_Server_Node_FQDN
--instance app-server-only
```

where:
 - *Current_Server_Node_FQDN* is the current FQDN of the server node host, for example, server1.mydomain.com.
 - *New_Server_Node_FQDN* is the new FQDN of the server node host, for example, new.server1.mydomain.com.
5. On the server node that is getting the new FQDN, change the FQDN of the host machine.
For more information, see your operating system documentation.
6. Restart each host machine.
7. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.

To change the FQDN of a hardware load balancer:

1. On each server node, stop Authentication Manager.
2. On the primary instance, stop Authentication Manager. Do not stop the internal database.
3. On the primary instance host, run the Update Instance Nodes utility to change the current hardware load balancer FQDN to the new FQDN:
 - a. On the primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utils***.
 - b. Type:

```
rsautil update-instance-node
--old-host Current_Loadbalancer_FQDN
--new-host New_Loadbalancer_FQDN
--instance proxy-server-only
```

where:
 - *Current_Loadbalancer_FQDN* is the current FQDN of the load balancer, for example, hw.loadbalancer.mydomain.com.
 - *New_Loadbalancer_FQDN* is the new FQDN of the load balancer, for example, new.hw.loadbalancer.mydomain.com.

4. On the hardware load balancer machine, change the FQDN of the machine. Restart the machine.
For more information, see your hardware load balancer documentation.
5. Restart each host machine.
6. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.

To change the FQDN of a machine hosting the internal database:

1. On each server node, stop Authentication Manager.
2. On the primary instance, stop Authentication Manager. Do not stop the internal database.
3. On the internal database host, run the Update Instance Nodes utility to change the current FQDN of the internal database host to the new FQDN:
 - a. On the internal database host, open a new command shell, and change directories to ***RSA_AM_HOME/utils***.
 - b. Type:

```
rsutil update-instance-node
--old-host Current_Database_Host_FQDN
--new-host New_Database_Host_FQDN
--instance database-only
```

where:

- *Current_Database_Host_FQDN* is the current FQDN of the database, for example, db.mydomain.com.
- *New_Database_Host_FQDN* is the new FQDN of the database, for example, new.db.mydomain.com.

4. On the primary instance, run the Update Instance Nodes utility to change the current FQDN of the internal database host to the new FQDN:
 - a. On the primary instance host, open a new command shell, and change directories to ***RSA_AM_HOME/utils***.
 - b. Type:

```
rsutil update-instance-node
--old-host Current_Database_Host_FQDN
--new-host New_Database_Host_FQDN
--instance database-only
```

where:

- *Current_Database_Host_FQDN* is the current FQDN of the database host, for example, db.mydomain.com.
- *New_Database_Host_FQDN* is the new FQDN of the database host, for example, new.db.mydomain.com.

5. On each server node, run the Update Instance Nodes utility to change the current FQDN of the internal database host to the new FQDN:
 - a. On each server node host, open a new command shell, and change directories to ***RSA_AM_HOME/utils***.
 - b. Type:

```
rsautil update-instance-node
--old-host Current_Database_Host_FQDN
--new-host New_Database_Host_FQDN
--instance database-only
```

where:
 - *Current_Database_Host_FQDN* is the current FQDN of the database host, for example, db.mydomain.com.
 - *New_Database_Host_FQDN* is the new FQDN of the database host, for example, new.db.mydomain.com.
6. On the internal database host, change the FQDN of the host machine. For more information, see your operating system documentation.
7. Restart each host machine.
8. On the internal database host, start the database.
9. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.

To change the FQDN of the primary instance with the internal database hosted on a separate machine:

1. On each server node, stop Authentication Manager.
2. On the primary instance, stop Authentication Manager. Do not stop the internal database.
3. On the primary instance, run the Update Instance Nodes utility to change the current FQDN of the primary instance to the new FQDN:
 - a. On the primary instance host, open a new command shell, and change directories to ***RSA_AM_HOME/utils***.
 - b. Type:

```
rsautil update-instance-node
--old-host Current_Primary_Instance_FQDN
--new-host New_Primary_Instance_FQDN
--instance primary
```

where:
 - *Current_Primary_Instance_FQDN* is the current FQDN of the primary instance host, for example, primaryinstanceapplicationserver.mydomain.com.
 - *New_Primary_Instance_FQDN* is the new FQDN of the primary instance host, for example, new.primaryinstanceapplicationserver.mydomain.com.

4. On each server node, run the Update Instance Nodes utility to change the current FQDN of the primary instance to the new FQDN:
 - a. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utls***.
 - b. Type:


```
rsautil update-instance-node
--old-host Current_Primary_Instance_FQDN
--new-host New_Primary_Instance_FQDN
--instance primary
```

 where:
 - *Current_Primary_Instance_FQDN* is the current FQDN of the primary instance host, for example, primaryinstanceapplicationserver.mydomain.com.
 - *New_Primary_Instance_FQDN* is the new FQDN of the primary instance host, for example, new.primaryinstanceapplicationserver.mydomain.com.
5. On the internal database host, run the Update Instance Nodes utility to change the current FQDN of the primary instance to the new FQDN:
 - a. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utls***.
 - b. Type:


```
rsautil update-instance-node
--old-host Current_Primary_Instance_FQDN
--new-host New_Primary_Instance_FQDN
--instance primary
```

 where:
 - *Current_Primary_Instance_FQDN* is the current FQDN of the primary instance host, for example, primaryinstanceapplicationserver.mydomain.com.
 - *New_Primary_Instance_FQDN* is the new FQDN of the primary instance host, for example, new.primaryinstanceapplicationserver.mydomain.com.
6. On the primary instance, change the FQDN of the host machine.
For more information, see your operating system documentation.
7. Restart each host machine.
8. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.

To change the FQDN of the RADIUS primary server:

Note: If your deployment includes a RADIUS primary server that is hosted on a separate machine, use the following procedure to change the FQDN of the RADIUS server.

If your deployment includes a RADIUS primary server that is hosted on the primary instance, complete [step 8](#) through [step 9](#) in the following procedure after you have changed the FQDN of the primary instance.

1. On each server node, stop Authentication Manager.
2. On the primary instance, stop Authentication Manager. Do not stop the internal database.
3. On the RADIUS server host, run the Update Instance Nodes utility to change the current FQDN of the machine hosting the RADIUS server to a new FQDN:

- a. On the RADIUS server host, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
- b. Type:

```
rsautil update-instance-node
--old-host Current_Radius_Host_FQDN
--new-host New_Radius_Host_FQDN
--instance radius-only
```

where:

- *Current_Radius_Host_IP_Address* is the current FQDN of the RADIUS server host, for example, radius.mydomain.com.
- *New_Radius_Host_IP_Address* is the new FQDN of the RADIUS server host, for example, new.radius.mydomain.com.

4. On the primary instance, run the Update Instance Nodes utility to change the current FQDN of the RADIUS server host to a new FQDN:
 - a. On the primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.

- b. Type:

```
rsautil update-instance-node
--old-host Current_Radius_Host_FQDN
--new-host New_Radius_Host_FQDN
--instance radius-only
```

where:

- *Current_Radius_Host_FQDN* is the current FQDN of the machine hosting the RADIUS server, for example, radius.mydomain.com.
- *New_Radius_Host_FQDN* is the new FQDN of the machine hosting the RADIUS server, for example, new.radius.mydomain.com.

5. On each server node, run the Update Instance Nodes utility to change the current FQDN of the RADIUS server host to a new FQDN:
 - a. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utls***.
 - b. Type:


```
rsautil update-instance-node
--old-host Current Radius Host FQDN
--new-host New Radius Host IP FQDN
--instance radius-only
```

 where:
 - *Current_Radius_Host_FQDN* is the current FQDN of the machine hosting the RADIUS server, for example, radius.mydomain.com.
 - *New_Radius_Host_FQDN* is the new FQDN of the machine hosting the RADIUS server, for example, new.radius.mydomain.com.
6. On the RADIUS server host, change the FQDN of the host machine. For more information, see your operating system documentation.
7. Restart each host machine.
8. On the RADIUS server host, perform the following steps to update the RADIUS server:
 - a. On the RADIUS server host, ensure that the RADIUS server is stopped.
 - b. On the RADIUS server host, open a new command shell, and change directories to one of the following:
 - On Windows: ***RSA_AM_HOME/radius/Service***.
 - On UNIX: ***RSA_AM_HOME/radius***.
 - c. Type:


```
sbrsetuptool -identity PRIMARY
-secret replication_secret
```

 where *replication_secret* is the replication secret that you are using for all of the replicas. You can use the same replication secret that you used before changing the FQDN.
 - d. On the RADIUS server host, start the RADIUS server.
9. For each RADIUS replica server, perform the following steps:
 - a. On the RADIUS replica server host, ensure that the RADIUS server is stopped.
 - b. On the primary instance host, open a new command shell, and change directories to one of the following:
 - On Windows: ***RSA_AM_HOME/radius/Service***.
 - On UNIX: ***RSA_AM_HOME/radius***.
 - c. Copy the **replica.ccpmkg** file from this location to the RADIUS replica server host.

- d. On the RADIUS replica server host, open a new command shell, and change directories to one of the following:
 - On Windows: *RSA_AM_HOME/radius/Service*.
 - On UNIX: *RSA_AM_HOME/radius*.
- e. Type:

```
sbrsetuptool -identity REPLICA  
-radpath replica.ccpmkg_path
```

where *replica.ccpmkg_path* is the absolute path to the **replica.ccpmkg** file that you copied from the primary instance host to the RADIUS replica server host.
- f. Restart the RADIUS server on the replica host.

Changing an IP Address or a Fully Qualified Domain Name in a Replicated Deployment

A replicated deployment consists of a primary instance and one or more replica instances. In the primary and replica instances, the internal database and an optional RADIUS server can be installed on the same machine as the application server. The internal database and RADIUS server can also be installed on separate machines.

Changing the IP Address in a Replicated Deployment

- To change the IP address of the primary instance in a replicated deployment:
- To change the IP address of the primary instance with the internal database hosted on a separate machine:
- To change the IP address of a machine hosting the internal database for the primary instance:
- To change the IP address of a replica instance in a replicated deployment:
- To change the IP address of a replica instance in a replicated deployment with the internal database hosted on a separate machine:
- To change the IP address of a machine hosting the internal database for a replica instance:
- To change the IP address of a RADIUS replica server:

To change the IP address of the primary instance in a replicated deployment:

1. On each server node, stop Authentication Manager.
2. On the primary instance, stop Authentication Manager. Do not stop the internal database.
3. On the primary instance, run the Update Instance Nodes utility to change the current IP address of the primary instance to the new IP address:
 - a. On the primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utlils***.

b. Type:

```
rsautil update-instance-node
--old-host Current_Primary_Instance_IP_Address
--new-host New_Primary_Instance_IP_Address
--instance primary
```

where:

- *Current_Primary_Instance_IP_Address* is the current IP address of the primary host, for example, 192.168.1.1.
- *New_Primary_Instance_IP_Address* is the new IP address of the primary host, for example, 192.168.200.245.

4. Run the Update Instance Nodes utility on each server node to change the current IP address of the primary instance to the new IP address:
 - a. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utlils***.

b. Type:

```
rsautil update-instance-node
--old-host Current_Primary_Instance_IP_Address
--new-host New_Primary_Instance_IP_Address
--instance primary
```

where:

- *Current_Primary_Instance_IP_Address* is the current IP address of the primary host, for example, 192.168.1.1.
- *New_Primary_Instance_IP_Address* is the new IP address of the primary host, for example, 192.168.200.245.

5. On the primary instance, change the IP address in the host machine network configuration, and update any required DNS tables.

For more information, see your operating system documentation.

6. Restart the primary instance.

7. Restart each server node.
8. Restart each replica instance in the replicated deployment.
9. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.

To change the IP address of the primary instance in a replicated deployment with the internal database hosted on a separate machine:

1. On each server node, stop Authentication Manager.
2. On the primary instance, stop Authentication Manager. Do not stop the internal database.
3. On the primary instance, run the Update Instance Nodes utility to change the current IP address of the primary instance to the new IP address:
 - a. On the primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.

- b. Type:

```
rsautil update-instance-node
--old-host Current_Primary_Instance_IP_Address
--new-host New_Primary_Instance_IP_Address
--instance primary
```

where:

- *Current_Primary_Instance_IP_Address* is the current IP address of the primary host, for example, 192.168.1.1.
- *New_Primary_Instance_IP_Address* is the new IP address of the primary host, for example, 192.168.200.245.

4. On each server node, run the Update Instance Nodes utility to change the current IP address of the primary instance to the new IP address:
 - a. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.

- b. Type:

```
rsautil update-instance-node
--old-host Current_Primary_Instance_IP_Address
--new-host New_Primary_Instance_IP_Address
--instance primary
```

where:

- *Current_Primary_Instance_IP_Address* is the current IP address of the primary host, for example, 192.168.1.1.
- *New_Primary_Instance_IP_Address* is the new IP address of the primary host, for example, 192.168.200.245.

5. On the machine hosting the internal database, run the Update Instance Nodes utility to change the current IP address of the primary instance to the new IP address:
 - a. On the machine hosting the internal database, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:


```
rsautil update-instance-node
--old-host Current_Primary_Instance_IP_Address
--new-host New_Primary_Instance_IP_Address
--instance primary
```

 where:
 - *Current_Primary_Instance_IP_Address* is the current IP address of the primary host, for example, 192.168.1.1.
 - *New_Primary_Instance_IP_Address* is the new IP address of the primary host, for example, 192.168.200.245.
6. On the primary instance, change the IP address in the host machine network configuration, and update any required DNS tables.
For more information, see your operating system documentation.
7. Restart the primary instance.
8. Restart each server node.
9. Restart the machine hosting the internal database.
10. Restart each replica instance in the replicated deployment.
11. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.

To change the IP address of a machine hosting the internal database for the primary instance:

1. On each server node, stop Authentication Manager.
2. On the primary instance, stop Authentication Manager. Do not stop the internal database.
3. Run the Update Instance Nodes utility on the machine hosting the internal database to change the current IP address of the internal database host to the new IP address:
 - a. On the machine hosting the internal database, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.

- b. Type:

```
rsautil update-instance-node
--old-host Current_Internal_Database_Host_IP_Address
--new-host New_Internal_Database_Host_IP_Address
--instance database-only
```

where:

- *Current_Internal_Database_Host_IP_Address* is the current IP address of the machine hosting the internal database, for example, 192.168.1.1.
- *New_Internal_Database_Host_IP_Address* is the new IP address of the machine hosting the internal database, for example, 192.168.200.245.

- a. On the primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.

- b. Type:

```
rsautil update-instance-node
--old-host Current_Internal_Database_Host_IP_Address
--new-host New_Internal_Database_Host_IP_Address
--instance database-only
```

where:

- *Current_Internal_Database_Host_IP_Address* is the current IP address of the machine hosting the internal database, for example, 192.168.1.1.
- *New_Internal_Database_Host_IP_Address* is the new IP address of the machine hosting the internal database, for example, 192.168.200.245.

5. Run the Update Instance Nodes utility on each server node to change the current IP address of the machine hosting the internal database to the new IP address:
 - a. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utls***.
 - b. Type:


```
rsautil update-instance-node
--old-host Current_Internal_Database_Host_IP_Address
--new-host New_Internal_Database_Host_IP_Address
--instance database-only
```

 where:
 - *Current_Internal_Database_Host_IP_Address* is the current IP address of the machine hosting the internal database, for example, 192.168.1.1.
 - *New_Internal_Database_Host_IP_Address* is the new IP address of the machine hosting the internal database, for example, 192.168.200.245.
6. On the machine hosting the internal database, change the IP address in the host machine network configuration, and update any required DNS tables.
For more information, see your operating system documentation.
7. Restart the primary instance.
8. Restart each server node.
9. Restart the machine hosting the internal database.
10. Restart each replica instance in the replicated deployment.
11. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.

To change the IP address of a replica instance in a replicated deployment:

1. On each server node, stop Authentication Manager.
2. On the replica instance, stop Authentication Manager. Do not stop the internal database.
3. On the replica instance, run the Update Instance Nodes utility to change the current IP address of the replica instance to the new IP address:
 - a. On the replica instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.

- b. Type:

```
rsautil update-instance-node
--old-host Current_Replica_Instance_IP_Address
--new-host New_Replica_Instance_IP_Address
--instance primary
```

where:

- *Current_Replica_Instance_IP_Address* is the current IP address of the replica host, for example, 192.168.1.1.
- *New_Replica_Instance_IP_Address* is the new IP address of the replica host, for example, 192.168.200.245.

- a. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.

- b. Type:

```
rsautil update-instance-node
--old-host Current_Replica_Instance_IP_Address
--new-host New_Replica_Instance_IP_Address
--instance primary
```

where:

- *Current_Replica_Instance_IP_Address* is the current IP address of the replica host, for example, 192.168.1.1.
- *New_Replica_Instance_IP_Address* is the new IP address of the replica host, for example, 192.168.200.245.

5. On the replica instance, change the IP address in the host machine network configuration, and update any required DNS tables.

For more information, see your operating system documentation.

6. Restart the replica instance.
7. Restart each server node.

8. Restart the primary instance in the replicated deployment.
9. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.

Note: You do not need to restart any other replica instances.

10. If your deployment includes a RADIUS server, see, "[To change the IP address of a RADIUS replica server:](#)" on page 339.

To change the IP address of a replica instance in a replicated deployment with the internal database hosted on a separate machine:

1. On each server node, stop Authentication Manager.
2. On the replica instance, stop Authentication Manager. Do not stop the internal database.
3. On the replica instance, run the Update Instance Nodes utility to change the current IP address of the replica instance to the new IP address:
 - a. On the replica instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:

```
rsautil update-instance-node
--old-host Current_Replica_Instance_IP_Address
--new-host New_Replica_Instance_IP_Address
--instance primary
```

where:

- *Current_Replica_Instance_IP_Address* is the current IP address of the replica host, for example, 192.168.1.1.
- *New_Replica_Instance_IP_Address* is the new IP address of the replica host, for example, 192.168.200.245.

4. On each server node, run the Update Instance Nodes utility to change the current IP address of the replica instance to the new IP address:
 - a. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:

```
rsautil update-instance-node
--old-host Current_Replica_Instance_IP_Address
--new-host New_Replica_Instance_IP_Address
--instance primary
```

where:
 - *Current_Replica_Instance_IP_Address* is the current IP address of the replica host, for example, 192.168.1.1.
 - *New_Replica_Instance_IP_Address* is the new IP address of the replica host, for example, 192.168.200.245.
5. On the internal database host, run the Update Instance Nodes utility to change the current IP address of the replica instance to the new IP address:
 - a. On the internal database host, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:

```
rsautil update-instance-node
--old-host Current_Replica_Instance_IP_Address
--new-host New_Replica_Instance_IP_Address
--instance primary
```

where:
 - *Current_Replica_Instance_IP_Address* is the current IP address of the replica host, for example, 192.168.1.1.
 - *New_Replica_Instance_IP_Address* is the new IP address of the replica host, for example, 192.168.200.245.
6. On the replica instance, change the IP address in the host machine network configuration, and update any required DNS tables.

For more information, see your operating system documentation.
7. Restart the replica instance.
8. Restart each server node.
9. Restart the primary instance in the replicated deployment.
10. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.

Note: You do not need to restart any other replica instances.

To change the IP address of a machine hosting the internal database for a replica instance:

1. On each server node, stop Authentication Manager.
2. On the replica instance, stop Authentication Manager. Do not stop the internal database.
3. On the internal database host, run the Update Instance Nodes utility to change the current IP address of the internal database host to the new IP address:
 - a. On the internal database host, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.

- b. Type:

```
rsautil update-instance-node
--old-host Current_Internal_DB_Host_IP_Address
--new-host New_Internal_DB_Host_IP_Address
--instance database-only
```

where:

- *Current_Internal_DB_Host_IP_Address* is the current IP address of the internal database host, for example, 192.168.1.1.
- *New_Internal_DB_Host_IP_Address* is the new IP address of the internal database host, for example, 192.168.200.245.

4. On the replica instance, run the Update Instance Nodes utility to change the current IP address of the internal database host to the new IP address:
 - a. On the replica instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.

- b. Type:

```
rsautil update-instance-node
--old-host Current_Internal_DB_Host_IP_Address
--new-host New_Internal_DB_Host_IP_Address
--instance database-only
```

where:

- *Current_Internal_DB_Host_IP_Address* is the current IP address of the internal database host, for example, 192.168.1.1.
- *New_Internal_DB_Host_IP_Address* is the new IP address of the internal database host, for example, 192.168.200.245.

5. On each server node, run the Update Instance Nodes utility to change the current IP address of the internal database host to the new IP address:
 - a. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utls***.
 - b. Type:

```
rsautil update-instance-node
--old-host Current_Internal_DB_Host_IP_Address
--new-host New_Internal_DB_Host_IP_Address
--instance database-only
```

where:
 - *Current_Internal_DB_Host_IP_Address* is the current IP address of the internal database host, for example, 192.168.1.1.
 - *New_Internal_DB_Host_IP_Address* is the new IP address of the internal database host, for example, 192.168.200.245.
6. On the internal database host, change the IP address in the host machine network configuration, and update any required DNS tables.
For more information, see your operating system documentation.
7. Restart the replica instance.
8. Restart each server node.
9. Restart the primary instance in the replicated deployment.
10. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.

Note: You do not need to restart any other replica instances.

To change the IP address of a RADIUS replica server:

Note: If your deployment includes a RADIUS replica server that is hosted on a separate machine, use the following procedure to change the IP address of the RADIUS server.

If your deployment includes a RADIUS replica server that is hosted on the replica instance, complete [step 9](#) through [step 10](#) in the following procedure after you have changed the IP address of the replica instance.

1. On each server node, stop Authentication Manager.
2. On the primary instance, stop Authentication Manager. Do not stop the internal database.

3. Run the Update Instance Nodes utility on the RADIUS replica host to change the current IP address of the RADIUS replica host to the new IP address:
 - a. On the RADIUS replica host, open a new command shell, and change directories to ***RSA_AM_HOME/utils***.
 - b. Type:


```
rsautl update-instance-node
--old-host Current_Radius_Host_IP_Address
--new-host New_Radius_Host_IP_Address
--instance radius-only
```

 where:
 - *Current_Radius_Host_IP_Address* is the current IP address of the RADIUS replica host, for example, 192.168.1.1.
 - *New_Internal_Radius_IP_Address* is the new IP address of the RADIUS replica host, for example, 192.168.200.245.
4. On the primary instance, run the Update Instance Nodes utility to change the current IP address of the RADIUS replica server host to a new IP address:
 - a. On the primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utils***.
 - b. Type:


```
rsautl update-instance-node
--old-host Current_Radius_Host_IP_Address
--new-host New_Radius_Host_IP_Address
--instance radius-only
```

 where:
 - *Current_Radius_Host_IP_Address* is the current IP address of the RADIUS replica host, for example, 192.168.1.1.
 - *New_Radius_Host_IP_Address* is the new IP address of the RADIUS replica host, for example, 192.168.200.245.
5. On the RADIUS replica host, change the IP address in the host machine network configuration, and update any required DNS tables.
For more information, see your operating system documentation.
6. Restart each host machine.
7. Restart each server node.
8. Restart the primary instance in the replicated deployment.

Note: You do not need to restart any other replica instances.

9. Perform the following steps to update the RADIUS replica server:
 - a. On the primary instance host, open a new command shell, and change directories to one of the following:
 - On Windows: *RSA_AM_HOME/radius/Service*.
 - On UNIX: *RSA_AM_HOME/radius*.
 - b. Copy the **replica.ccpmkg** file from this location to the RADIUS replica host.
 - c. On the RADIUS replica host, open a new command shell, and change directories to one of the following:
 - On Windows: *RSA_AM_HOME/radius/Service*.
 - On UNIX: *RSA_AM_HOME/radius*.
 - d. Type:

```
sbrsetuptool -identity REPLICA
-radpath replica.ccpmkg_path
```

where *replica.ccpmkg_path* is the absolute path to the **replica.ccpmkg** file that you copied from the primary instance host to the RADIUS replica host.
10. Restart the RADIUS server on the replica host.

Changing the FQDN in a Replicated Deployment

- [To change the FQDN of the primary instance in a replicated deployment:](#)
- [To change the FQDN of the primary instance in a replicated deployment with the internal database hosted on a separate machine.](#)
- [To change the FQDN of a machine hosting the internal database for the primary instance:](#)
- [To change the FQDN of a replica instance in a replicated deployment:](#)
- [To change the FQDN of a replica instance in a replicated deployment with the internal database hosted on a separate machine:](#)
- [To change the FQDN of a machine hosting the internal database for a replica instance:](#)
- [To change the FQDN of a RADIUS replica server:](#)

To change the FQDN of the primary instance in a replicated deployment:

Important: RSA recommends that you back up the primary instance before you begin the promotion process.

1. On all server nodes, stop Authentication Manager.
2. Use the Manage Replication utility to promote a replica instance to a primary instance:
 - a. On the replica instance host being promoted, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:


```
rsautil manage-replication --action promote
--planned-mode
```
 - c. When prompted, enter your master password for the encrypted properties file.
3. Use the Manage Replication utility to make sure that the replica instance is attached:
 - a. Type:


```
rsautil manage-replication --action attach-status
```
 - b. When prompted, enter your master password for the encrypted properties file.
4. On the current primary instance host, start Authentication Manager.
5. On the original primary instance, use the Manage Backups utility to make a full backup of the internal database and to transfer the backup files to the current primary instance:
 - a. On the original primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:


```
rsautil manage-backups --action export
--filename backup_filename_and_path
```

 where *backup_filename_and_path* is the absolute path and name of the database backup file.
 - c. When prompted, enter your master password for the encrypted properties file. The export action generates two files based on the name that you provide: **filename**. The database backup file. **filename.secrets**. The user credentials backup file.
 - d. On the current primary instance host, stop Authentication Manager.
 - e. On the current primary instance, copy the files generated in [step c](#) from the original primary instance.

- f. On the current primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - g. Type:

```
rsautil manage-backups --action import
--filename backup_filenames_and_path
```

where *backup_filenames_and_path* is the absolute path and names of the two backup files copied in [step e](#).
 - h. When prompted, enter your master password for the encrypted properties file.
6. Use the Update Instance Nodes utility to change the FQDN of the original primary instance host:
 - a. On the original primary instance host, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:

```
rsautil update-instance-node
--old-host Current_Primary_Instance_FQDN
--new-host New_Primary_Instance_FQDN
--instance primary
```

where:
 - *Current_Primary_Instance_FQDN* is the current FQDN of the original primary instance host, for example, primaryinstance.mydomain.com.
 - *New_Primary_Instance_FQDN* is the new FQDN of the original primary instance host, for example, newprimaryinstance.mydomain.com.
 - c. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - d. Type:

```
rsautil update-instance-node
--old-host Current_Primary_Instance_FQDN
--new-host New_Primary_Instance_FQDN
--instance primary
```

where:
 - *Current_Primary_Instance_FQDN* is the current FQDN of the original primary instance host, for example, primaryinstance.mydomain.com.
 - *New_Primary_Instance_FQDN* is the new FQDN of the original primary instance host, for example, newprimaryinstance.mydomain.com.
 7. On the original primary instance host, change the FQDN of the host machine. For more information, see your operating system documentation.
 8. Restart the original primary instance host.
 9. Restart each server node.

10. Begin the process to attach and promote the original primary instance. Copy the **primary.properties** file from the current primary instance to the original primary instance:
 - a. On the current primary instance host, stop Authentication Manager.
 - b. On the original primary instance host, stop Authentication Manager.
 - c. On the current primary instance, open a new command shell, and change directories to **RSA_AM_HOME/utills/etc**.
 - d. Copy the **primary.properties** file from this location to the **RSA_AM_HOME/utills/etc** directory on the original primary instance host.
 11. Use the Setup Replication utility to attach the original primary instance as a replica instance:
 - a. On the original primary instance host, open a new command shell, and change directories to **RSA_AM_HOME/utills**.
 - b. Type:


```
rsutil setup-replication --action attach-old-primary
```
 - c. When prompted, enter your master password for the encrypted properties file.
 - d. Verify the addition of the replica instance. Type:


```
rsutil setup-replication --action list
```
 - e. When prompted, enter your master password for the encrypted properties file. The displayed report shows the replica instance after the primary instance.
-
- Important:** RSA recommends that you back up the primary instance before you begin the promotion process.
-
12. Use the Manage Replication utility to promote the original primary instance that was attached as a replica instance in [step 11](#) to a primary instance:
 - a. On the replica instance host being promoted (the original primary instance), open a new command shell, and change directories to **RSA_AM_HOME/utills**.
 - b. Type:


```
rsutil manage-replication --action promote  
--planned-mode
```
 - c. When prompted, enter your master password for the encrypted properties file.
 13. Use the Manage Replication utility to make sure that the replica instance (the original primary instance) is promoted:
 - a. Type:


```
rsutil manage-replication --action attach-status
```
 - b. When prompted, enter your master password for the encrypted properties file.
 14. On the replica instance (the original primary instance) that was promoted in [step 12](#), start Authentication Manager.

15. On the former primary instance, use the Manage Backups utility to make a full backup of the internal database and to transfer the backup files to the machine hosting the current primary instance:
 - a. On the former primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:

```
rsautil manage-backups --action export
--filename backup_filename_and_path
```

where *backup_filename_and_path* is the absolute path and name of the database backup file.
 - c. When prompted, enter your master password for the encrypted properties file. The export action generates two files based on the name that you provide:
filename. The database backup file.
filename.secrets. The user credentials backup file.
 - d. On the current primary instance (the original primary instance), stop Authentication Manager.
 - e. On the current primary instance (the original primary instance), copy the files generated in [step c](#) from the former primary instance.
 - f. On the current primary instance (the original primary instance), open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - g. Type:

```
rsautil manage-backups --action import
--filename backup_filenames_and_path
```

where *backup_filenames_and_path* is the absolute path and names of the two backup files copied in [step e](#).
 - h. When prompted, enter your master password for the encrypted properties file.
 - i. On all server nodes, start Authentication Manager.
16. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.

17. Optional. Begin the process to attach the former primary instance as a replica. Copy the **primary.properties** file from the current primary instance to the demoted primary instance being attached:
 - a. On the current primary instance host, stop Authentication Manager.
 - b. On the demoted primary instance host being attached, stop Authentication Manager.
 - c. On the current primary instance, open a new command shell, and change directories to **RSA_AM_HOME/utills/etc**.
 - d. Copy the **primary.properties** file from this location to the **RSA_AM_HOME/utills/etc** directory on the demoted primary instance host being attached.
18. Optional. Use the Setup Replication utility to attach the demoted primary instance as a replica instance:
 - a. On the demoted primary instance host, open a new command shell, and change directories to **RSA_AM_HOME/utills**.
 - b. Type:


```
rsutil setup-replication --action attach-old-primary
```
 - c. When prompted, enter your master password for the encrypted properties file.
 - d. Verify the addition of the replica instance. Type:


```
rsutil setup-replication --action list
```
 - e. When prompted, enter your master password for the encrypted properties file. The displayed report shows the replica instance after the primary instance.

To change the FQDN of the primary instance in a replicated deployment with the internal database hosted on a separate machine.

1. On each server node, stop Authentication Manager.
2. On the primary instance host, stop Authentication Manager. Do not stop the internal database.
3. On the primary instance, use the Update Instance Nodes utility to change the FQDN of the primary instance host:
 - a. On the primary instance host, open a new command shell, and change directories to **RSA_AM_HOME/utills**.
 - b. Type:


```
rsutil update-instance-node
--old-host Current_Primary_Instance_FQDN
--new-host New_Primary_Instance_FQDN
--instance primary
```

 where:
 - *Current_Primary_Instance_FQDN* is the current FQDN of the primary instance host, for example, primaryinstance.mydomain.com.
 - *New_Primary_Instance_FQDN* is the new FQDN of the primary instance host, for example, newprimaryinstance.mydomain.com.

4. On each server node, use the Update Instance Nodes utility to change the FQDN of the primary instance host:
 - a. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:

```
rsautil update-instance-node
--old-host Current_Primary_Instance_FQDN
--new-host New_Primary_Instance_FQDN
--instance primary
```

where:
 - *Current_Primary_Instance_FQDN* is the current FQDN of the primary instance host, for example, primaryinstance.mydomain.com.
 - *New_Primary_Instance_FQDN* is the new FQDN of the primary instance host, for example, newprimaryinstance.mydomain.com.
5. On the machine hosting the internal database, use the Update Instance Nodes utility to change the FQDN of the primary instance host:
 - a. On the machine hosting the internal database, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:

```
rsautil update-instance-node
--old-host Current_Primary_Instance_FQDN
--new-host New_Primary_Instance_FQDN
--instance primary
```

where:
 - *Current_Primary_Instance_FQDN* is the current FQDN of the primary instance host, for example, primaryinstance.mydomain.com.
 - *New_Primary_Instance_FQDN* is the new FQDN of the primary instance host, for example, newprimaryinstance.mydomain.com.
6. On the primary instance host, change the FQDN of the host machine.
For more information, see your operating system documentation.
7. Restart the primary instance.
8. Restart each server node.
9. Restart each replica instance.
10. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.

To change the FQDN of a machine hosting the internal database for the primary instance:

Important: RSA recommends that you back up the primary instance before you begin the promotion process.

1. On all server nodes, stop Authentication Manager.
2. Use the Manage Replication utility to promote a replica instance to a primary instance:
 - a. On the replica instance host being promoted, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:


```
rsautil manage-replication --action promote
--planned-mode
```
 - c. When prompted, enter your master password for the encrypted properties file.
3. Use the Manage Replication utility to make sure that the replica instance is attached:
 - a. Type:


```
rsautil manage-replication --action attach-status
```
 - b. When prompted, enter your master password for the encrypted properties file.
4. On the current primary instance host, start Authentication Manager.
5. On the machine hosting the internal database for the original primary instance, use the Manage Backups utility to make a full backup of the internal database and to transfer the backup files to the machine hosting the internal database for the current primary instance:
 - a. On the machine hosting the internal database for the original primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:


```
rsautil manage-backups --action export
--filename backup_filename_and_path
```

 where *backup_filename_and_path* is the absolute path and name of the database backup file.
 - c. When prompted, enter your master password for the encrypted properties file. The export action generates two files based on the name that you provide: **filename**. The database backup file. **filename.secrets**. The user credentials backup file.
 - d. On the current primary instance host, stop Authentication Manager.
 - e. On the machine hosting the internal database for the current primary instance, copy the files generated in [step c](#) from the machine hosting the internal database for the original primary instance.

- f. On machine hosting the internal database for the current primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - g. Type:

```
rsautil manage-backups --action import
--filename backup_filenames_and_path
```

where *backup_filenames_and_path* is the absolute path and names of the two backup files copied in [step e](#).
 - h. When prompted, enter your master password for the encrypted properties file.
6. Use the Update Instance Nodes utility to change the FQDN of the machine hosting the internal database for the original primary instance:
 - a. On the original primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:

```
rsautil update-instance-node
--old-host Current_Internal_Database_Host_FQDN
--new-host New_Internal_Database_Host_FQDN
--instance database-only
```

where:
 - *Current_Internal_Database_Host_FQDN* is the current FQDN of the machine hosting the internal database for the original primary instance host, for example, primaryinstanceinternaldatabase.mydomain.com.
 - *New_Internal_Database_Host_FQDN* is the new FQDN of the machine hosting the internal database for the original primary instance host, for example, newprimaryinstanceinternaldatabase.mydomain.com.
 - c. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - d. Type:

```
rsautil update-instance-node
--old-host Current_Internal_Database_Host_FQDN
--new-host New_Internal_Database_Host_FQDN
--instance database-only
```

where:
 - *Current_Internal_Database_Host_FQDN* is the current FQDN of the machine hosting the internal database for the original primary instance host, for example, primaryinstanceinternaldatabase.mydomain.com.
 - *New_Internal_Database_Host_FQDN* is the new FQDN of the machine hosting the internal database for the original primary instance host, for example, newprimaryinstanceinternaldatabase.mydomain.com.

- e. On the machine hosting the internal database for the original primary instance host, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
- f. Type:


```
rsautil update-instance-node
--old-host Current_Internal_Database_Host_FQDN
--new-host New_Internal_Database_Host_FQDN
--instance database-only
```

 where:
 - *Current_Internal_Database_Host_FQDN* is the current FQDN of the machine hosting the internal database for the original primary instance host, for example, primaryinstanceinternaldatabase.mydomain.com.
 - *New_Internal_Database_Host_FQDN* is the new FQDN of the machine hosting the internal database for the original primary instance host, for example, newprimaryinstanceinternaldatabase.mydomain.com.
7. On the machine hosting the internal database for the original primary instance, change the FQDN of the host machine.
For more information, see your operating system documentation.
8. Restart the machine hosting the internal database for the original primary instance host.
9. Begin the process to attach and promote the original primary instance. Copy the **primary.properties** file from the current primary instance to the original primary instance:
 - a. On the current primary instance host, stop Authentication Manager.
 - b. On the original primary instance host, stop Authentication Manager.
 - c. On the current primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills/etc***.
 - d. Copy the **primary.properties** file from this location to the ***RSA_AM_HOME/utills/etc*** directory on the original primary instance host.
10. Use the Setup Replication utility to attach the original primary instance as a replica instance:
 - a. On the original primary instance host, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:


```
rsautil setup-replication --action attach-old-primary
```
 - c. When prompted, enter your master password for the encrypted properties file.
 - d. Verify the addition of the replica instance. Type:


```
rsautil setup-replication --action list
```
 - e. When prompted, enter your master password for the encrypted properties file. The displayed report shows the replica instance after the primary instance.

Important: RSA recommends that you back up the primary instance before you begin the promotion process.

11. Use the Manage Replication utility to promote the original primary instance that was attached as a replica instance in [step 10](#) to the primary instance:
 - a. On the replica instance host being promoted (the original primary instance), open a new command shell, and change directories to ***RSA_AM_HOME/utls***.
 - b. Type:

```
rsutil manage-replication --action promote
--planned-mode
```
 - c. When prompted, enter your master password for the encrypted properties file.
12. Use the Manage Replication utility to make sure that the replica instance (the original primary instance) is promoted:
 - a. Type:

```
rsutil manage-replication --action attach-status
```
 - b. When prompted, enter your master password for the encrypted properties file.
13. On the replica instance (the original primary instance) that was promoted in [step 11](#), start Authentication Manager.
14. On the machine hosting the internal database for the former primary instance, use the Manage Backups utility to make a full backup of the internal database and to transfer the backup files to the machine hosting the internal database for the current primary instance (the original primary instance):
 - a. On the machine hosting the internal database for the former primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utls***.
 - b. Type:

```
rsutil manage-backups --action export
--filename backup_filename_and_path
```

where *backup_filename_and_path* is the absolute path and name of the database backup file.
 - c. When prompted, enter your master password for the encrypted properties file. The export action generates two files based on the name that you provide: ***filename***. The database backup file. ***filename.secrets***. The user credentials backup file.
 - d. On the current primary instance host (the original primary instance), stop Authentication Manager.
 - e. On the machine hosting the internal database for the current primary instance (the original primary instance), copy the files generated in [step c](#) from the machine hosting the internal database for former primary instance.

- f. On the internal database host for the current primary instance (the original primary instance), open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - g. Type:


```
rsautil manage-backups --action import
--filename backup_filenames_and_path
```

 where *backup_filenames_and_path* is the absolute path and names of the two backup files copied in [step e](#).
 - h. When prompted, enter your master password for the encrypted properties file.
15. On all server nodes, start Authentication Manager.
 16. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.
 17. Optional. Begin the process to attach the former primary instance as a replica. Copy the **primary.properties** file from the current primary instance to the demoted primary instance being attached:
 - a. On the current primary instance host, stop Authentication Manager.
 - b. On the demoted primary instance host being attached, stop Authentication Manager.
 - c. On the current primary instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills/etc***.
 - d. Copy the **primary.properties** file from this location to the ***RSA_AM_HOME/utills/etc*** directory on the demoted primary instance host being attached.
 18. Optional. Use the Setup Replication utility to attach the demoted primary instance as a replica instance:
 - a. On the demoted primary instance host, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:


```
rsautil setup-replication --action attach-old-primary
```
 - c. When prompted, enter your master password for the encrypted properties file.
 - d. Verify the addition of the replica instance. Type:


```
rsautil setup-replication --action list
```
 - e. When prompted, enter your master password for the encrypted properties file. The displayed report shows the replica instance after the primary instance.

To change the FQDN of a replica instance in a replicated deployment:

1. Use the Manage Replication utility to identify and remove the replica instance:
 - a. On the primary instance host, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:

```
rsautil manage-replication --action list
```
 - c. When prompted, enter your master password for the encrypted properties file. This action displays the names of all of the replica instances in the deployment.
 - d. Type:

```
rsautil manage-replication --action remove-replica  
--name instance_name
```

where *instance_name* is the unique instance name of the replica instance that you are removing. You can find this name in the list displayed in [step c](#).
 - e. When prompted, enter your master password for the encrypted properties file.
2. Use the Update Instance Nodes utility to change the FQDN of the replica instance:
 - a. On each server node, stop Authentication Manager.
 - b. On the replica instance host, stop Authentication Manager. Do not stop the internal database.
 - c. On the replica instance host, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - d. Type:

```
rsautil update-instance-node  
--old-host Current_Replica_Instance_FQDN  
--new-host New_Replica_Instance_FQDN  
--instance primary
```

where:
 - *Current_Replica_Instance_FQDN* is the current FQDN of the replica host, for example, replicainstance.mydomain.com.
 - *New_Replica_Instance_FQDN* is the new FQDN of the replica host, for example, newreplicainstance.mydomain.com.
 - e. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.

- f. Type:


```
rsautil update-instance-node
--old-host Current_Replica_Instance_FQDN
--new-host New_Replica_Instance_FQDN
--instance primary
```

 where:
 - *Current_Replica_Instance_FQDN* is the current FQDN of the replica host, for example, replicainstance.mydomain.com.
 - *New_Replica_Instance_FQDN* is the new FQDN of the replica host, for example, newreplicainstance.mydomain.com.
- g. On the replica instance host, change the FQDN of the host machine. For more information, see your operating system documentation.
- h. Restart the replica instance.
- i. Restart each server node.
3. Run configUtil on the replica instance host to attach the replica instance:
 - a. On each server node, stop Authentication Manager.
 - b. On the replica instance host, stop Authentication Manager. Do not stop the internal database.
 - c. On the replica instance host, open a new command shell, and change directories to ***RSA_AM_HOME/config***. Do one of the following:
 - On Windows, type:


```
configUtil.cmd configure util-replica
```
 - On UNIX (as root), type:


```
configUtil.sh configure util-replica
```
 - d. On the replica instance host, start Authentication Manager.
 - e. On each server node, start Authentication Manager.
4. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.
5. If your deployment includes a RADIUS server, see [“To change the FQDN of a RADIUS replica server:”](#) on page 359.

To change the FQDN of a replica instance in a replicated deployment with the internal database hosted on a separate machine:

1. On each server node, stop Authentication Manager.
2. On the replica instance host, stop Authentication Manager. Do not stop the internal database.
3. On the replica instance, use the Update Instance Nodes utility to change the FQDN of the replica instance host:

- a. On the replica instance host, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.

- b. Type:

```
rsautil update-instance-node
--old-host Current_Replica_Instance_FQDN
--new-host New_Replica_Instance_FQDN
--instance primary
```

where:

- *Current_Replica_Instance_FQDN* is the current FQDN of the replica instance host, for example, replicainstance.mydomain.com.
- *New_Primary_Instance_FQDN* is the new FQDN of the replica instance host, for example, newreplicainstance.mydomain.com.

4. On each server node, use the Update Instance Nodes utility to change the FQDN of the replica instance host:

- a. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.

- b. Type:

```
rsautil update-instance-node
--old-host Current_Replica_Instance_FQDN
--new-host New_Replica_Instance_FQDN
--instance primary
```

where:

- *Current_Replica_Instance_FQDN* is the current FQDN of the replica instance host, for example, replicainstance.mydomain.com.
- *New_Primary_Instance_FQDN* is the new FQDN of the replica instance host, for example, newreplicainstance.mydomain.com.

5. On the machine hosting the internal database, use the Update Instance Nodes utility to change the FQDN of the replica instance host:
 - a. On the machine hosting the internal database for the replica instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:


```
rsautil update-instance-node
--old-host Current_Replica_Instance_FQDN
--new-host New_Replica_Instance_FQDN
--instance primary
```

 where:
 - *Current_Replica_Instance_FQDN* is the current FQDN of the replica instance host, for example, primaryinstance.mydomain.com.
 - *New_Replica_Instance_FQDN* is the new FQDN of the replica instance host, for example, newprimaryinstance.mydomain.com.
6. On the replica instance, change the FQDN of the host machine.
For more information, see your operating system documentation.
7. Restart the replica instance host.
8. Restart each server node.
9. Restart the machine hosting the internal database for the replica instance.
10. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.

To change the FQDN of a machine hosting the internal database for a replica instance:

1. Use the Manage Replication utility to identify and remove the replica instance:
 - a. On the primary instance host, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - b. Type:


```
rsautil manage-replication --action list
```
 - c. When prompted, enter your master password for the encrypted properties file.
This action displays the names of all of the replica instances in the deployment.
 - d. Type:


```
rsautil manage-replication --action remove-replica
--name instance_name
```

 where *instance_name* is the unique instance name of the replica instance that you are removing. You can find this name in the list displayed in [step c](#).
 - e. When prompted, enter your master password for the encrypted properties file.

2. Use the Update Instance Nodes utility to change the FQDN of the machine hosting the internal database for the replica instance:
 - a. On each server node, stop Authentication Manager.
 - b. On the replica instance host, stop Authentication Manager. Do not stop the internal database.
 - c. On the replica instance host, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - d. Type:

```
rsautil update-instance-node
--old-host Current_Internal_Database_Host_FQDN
--new-host New_Internal_Database_Host_FQDN
--instance database-only
```

where:

- *Current_Internal_Database_Host_FQDN* is the current FQDN of the machine hosting the internal database for the replica instance, for example, internaldatabase.mydomain.com.
 - *New_Internal_Database_Host_FQDN* is the new FQDN of the machine hosting the internal database for the replica instance, for example, newinternaldatabase.mydomain.com.
- e. On each server node, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.
 - f. Type:

```
rsautil update-instance-node
--old-host Current_Internal_Database_Host_FQDN
--new-host New_Internal_Database_Host_FQDN
--instance database-only
```

where:

- *Current_Internal_Database_Host_FQDN* is the current FQDN of the machine hosting the internal database for the replica instance, for example, internaldatabase.mydomain.com.
 - *New_Internal_Database_Host_FQDN* is the new FQDN of the machine hosting the internal database for the replica instance, for example, newinternaldatabase.mydomain.com.
- g. On the machine hosting the internal database for the replica instance, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.

h. Type:

```
rsautil update-instance-node
--old-host Current_Internal_Database_Host_FQDN
--new-host New_Internal_Database_Host_FQDN
--instance database-only
```

where:

- *Current_Internal_Database_Host_FQDN* is the current FQDN of the machine hosting the internal database for the replica instance, for example, internaldatabase.mydomain.com.
 - *New_Internal_Database_Host_FQDN* is the new FQDN of the machine hosting the internal database for the replica instance, for example, newinternaldatabase.mydomain.com.
3. On the machine hosting the internal database, change the FQDN of the internal database host machine.
For more information, see your operating system documentation.
 4. Restart each server node.
 5. Restart the replica instance.
 6. Restart the machine hosting the internal database.
 7. Run configUtil on the replica instance host to attach the replica instance:
 - a. On each server node, stop Authentication Manager.
 - b. On the replica instance host, stop Authentication Manager. Do not stop the internal database.
 - c. On the replica instance host, open a new command shell, and change directories to **RSA_AM_HOME/config**. Do one of the following:
 - On Windows, type:
`configUtil.cmd configure util-replica`
 - On UNIX (as root), type:
`configUtil.sh configure util-replica`
 - d. On the replica instance host, start Authentication Manager.
 - e. On each server node, start Authentication Manager.
 8. Rebalance the servers:
 - a. Open the Security Console.
 - b. Click **Access > Authentication Agents > Authentication Manager Contact List > Automatic Rebalance**.

To change the FQDN of a RADIUS replica server:

Note: If your deployment includes a RADIUS replica server that is hosted on a separate machine, use the following procedure to change the IP address of the RADIUS server.

If your deployment includes a RADIUS replica server that is hosted on the replica instance, complete [step 6](#) through [step 7](#) in the following procedure after you have changed the FQDN of the replica instance.

1. Use the Update Instance Nodes utility to change the FQDN of the machine hosting the RADIUS server for the replica instance:
 - a. On each server node, stop Authentication Manager.
 - b. On the primary instance host, stop Authentication Manager. Do not stop the internal database.
 - c. On the primary instance host, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.

d. Type:

```
rsutil update-instance-node
--old-host Current Radius Host FQDN
--new-host New Radius Host FQDN
--instance radius-only
```

where:

- *Current_Radius_Host_FQDN* is the current FQDN of the machine hosting the RADIUS server for the replica instance, for example, radius.mydomain.com.
 - *New_Radius_Host_FQDN* is the new FQDN of the machine hosting the RADIUS server for the replica instance, for example, newradius.mydomain.com.
- e. On the machine hosting the RADIUS server, open a new command shell, and change directories to ***RSA_AM_HOME/utills***.

f. Type:

```
rsutil update-instance-node
--old-host Current Radius Host FQDN
--new-host New Radius Host FQDN
--instance radius-only
```

where:

- *Current_Radius_Host_FQDN* is the current FQDN of the machine hosting the RADIUS server for the replica instance, for example, radius.mydomain.com.
 - *New_Radius_Host_FQDN* is the new FQDN of the machine hosting the RADIUS server for the replica instance, for example, newradius.mydomain.com.
2. On the machine hosting the RADIUS replica server, change the FQDN of the RADIUS server host machine.

For more information, see your operating system documentation.

3. Restart each server node.
4. Restart the replica instance.
5. Restart the primary instance in the replicated deployment.
6. Perform the following steps to update the RADIUS replica server:
 - a. On the primary instance host, open a new command shell, and change directories to one of the following:
 - On Windows: ***RSA_AM_HOME/radius/Service***.
 - On UNIX: ***RSA_AM_HOME/radius***.
 - b. Copy the **replica.ccpmkg** file from this location to the RADIUS replica host.
 - c. On the RADIUS replica host, open a new command shell, and change directories to one of the following:
 - On Windows: ***RSA_AM_HOME/radius/Service***.
 - On UNIX: ***RSA_AM_HOME/radius***.
 - d. Type:


```
sbrsetuptool -identity REPLICA
-radpath replica.ccpmkg_path
```

 where *replica.ccpmkg_path* is the absolute path to the **replica.ccpmkg** file you copied from the primary instance host to the RADIUS replica server host.
7. Restart the RADIUS server on the replica host.

F

RSA Authentication Manager Message IDs

The following tables in this appendix contain message IDs that can be filtered by a Network Management System (NMS).

The message IDs are grouped in the following tables:

Audit Log Messages. Contains error messages from the administrative audit log.

System Log Messages. Contains error messages from the system log.

RSA Authentication Manager Administrative Audit Log Messages. Contains authentication-specific failure and warning messages from the administrative audit log.

RSA Authentication Manager Runtime Audit Log Messages. Contains authentication-specific failure and warning messages from the runtime audit log.

RSA Authentication Manager System Log Messages. Contains authentication-specific failure and warning messages from the system log.

Audit Log Messages

The severity level of the following message IDs is ERROR.

Message ID	Cause
CLEANUP_IDENTITY_SOURCE	INSUFFICIENT_PRIVILEGE
CREATE_ADMIN_ROLE	INSUFFICIENT_PRIVILEGE
CREATE_ATTRIBUTE	INSUFFICIENT_PRIVILEGE
CREATE_ATTRIBUTE_MAPPING	INSUFFICIENT_PRIVILEGE
CREATE_AUTH_POLICY	INSUFFICIENT_PRIVILEGE
CREATE_GROUP	INSUFFICIENT_PRIVILEGE
CREATE_IDENTITY_SOURCE	INSUFFICIENT_PRIVILEGE
CREATE_LOCKOUT_POLICY	INSUFFICIENT_PRIVILEGE
CREATE_PRINCIPAL	UNEXPECTED_EXCEPTION
CREATE_PRINCIPAL	DUPLICATE_ENTRY_EXISTS
CREATE_PRINCIPAL	INSUFFICIENT_PRIVILEGE
CREATE_PRINCIPAL_PREFERENCES	INSUFFICIENT_PRIVILEGE
CREATE_PWD_POLICY	INSUFFICIENT_PRIVILEGE



Message ID	Cause
CREATE_REPORT_QUERY	DUPLICATE_ENTRY_EXISTS
CREATE_SELFSERVICE_POLICY	INSUFFICIENT_PRIVILEGE
CREATE_TRUST	INSUFFICIENT_PRIVILEGE
DELETE_ADMIN_ROLE	INSUFFICIENT_PRIVILEGE
DELETE_ATTRIBUTE	INSUFFICIENT_PRIVILEGE
DELETE_ATTRIBUTE_MAPPING	INSUFFICIENT_PRIVILEGE
DELETE_GROUP	INSUFFICIENT_PRIVILEGE
DELETE_IDENTITY_SOURCE	INSUFFICIENT_PRIVILEGE
DELETE_PRINCIPAL	INSUFFICIENT_PRIVILEGE
DELETE_PRINCIPAL_PREFERENCES	INSUFFICIENT_PRIVILEGE
DELETE_SECURITY_DOMAIN	INSUFFICIENT_PRIVILEGE
DELETE_TRUST	INSUFFICIENT_PRIVILEGE
DEREFERENCE_REALM	INSUFFICIENT_PRIVILEGE
FIND_ORPHANED_GROUPS	INSUFFICIENT_PRIVILEGE
FIND_ORPHANED_PRINCIPALS	INSUFFICIENT_PRIVILEGE
LINK_GROUP_GROUP	INSUFFICIENT_PRIVILEGE
LINK_GROUP_PRINCIPAL	INSUFFICIENT_PRIVILEGE
LINK_PRINCIPAL_ADMIN_ROLE	INSUFFICIENT_PRIVILEGE
MANAGE_ATTR_CATEGORY	INSUFFICIENT_PRIVILEGE
READ_ADMIN_ROLE	INSUFFICIENT_PRIVILEGE
READ_ATTRIBUTE	INSUFFICIENT_PRIVILEGE
READ_ATTRIBUTE_MAPPING	INSUFFICIENT_PRIVILEGE
READ_AUTH_POLICY	INSUFFICIENT_PRIVILEGE
READ_GROUP	INSUFFICIENT_PRIVILEGE
READ_LOCKOUT_POLICY	INSUFFICIENT_PRIVILEGE
READ_PRINCIPAL	INSUFFICIENT_PRIVILEGE
READ_PRINCIPAL_PREFERENCES	INSUFFICIENT_PRIVILEGE
READ_PWD_POLICY	INSUFFICIENT_PRIVILEGE
READ_REPORT_DATA	UNEXPECTED_EXCEPTION

Message ID	Cause
READ_REPORT_META_DATA	UNEXPECTED_EXCEPTION
READ_SECURITY_DOMAIN	INSUFFICIENT_PRIVILEGE
READ_SECURITY_QUESTIONS_POLICY	INSUFFICIENT_PRIVILEGE
READ_SELFSERVICE_POLICY	INSUFFICIENT_PRIVILEGE
READ_TRUST	INSUFFICIENT_PRIVILEGE
REGISTER_GROUP	INSUFFICIENT_PRIVILEGE
REGISTER_PRINCIPAL	NO_USER_ID
UNREGISTER_GROUP	INSUFFICIENT_PRIVILEGE
UPDATE_ADMIN_ROLE	INSUFFICIENT_PRIVILEGE
UPDATE_ATTRIBUTE	INSUFFICIENT_PRIVILEGE
UPDATE_GROUP	INSUFFICIENT_PRIVILEGE
UPDATE_IDENTITY_SOURCE	INSUFFICIENT_PRIVILEGE
UPDATE_PRINCIPAL	INSUFFICIENT_PRIVILEGE
UPDATE_PRINCIPAL_PREFERENCES	INSUFFICIENT_PRIVILEGE
UPDATE_REALM	INSUFFICIENT_PRIVILEGE
UPDATE_SECURITY_DOMAIN	INSUFFICIENT_PRIVILEGE
UPDATE_TRUST	INSUFFICIENT_PRIVILEGE

System Log Messages

The severity level of the following message IDs is ERROR.

Message ID	Cause
AA_PROCESS_REQUEST	UNEXPECTED_EXCEPTION
AA_PROCESS_REQUEST	AA_BAD_AUTHNREQUEST
ACCESS_DATABASE	UNEXPECTED_EXCEPTION
ACCESS_DATABASE	UNEXPECTED_DATABASE_OPERATION_FAILURE
ACCESS_DIRECTORY	IDENTITY_SOURCE_IS_READONLY
ACCESS_DIRECTORY	UNEXPECTED_LDAP_EXCEPTION
ADD_BATCH_JOB	UNEXPECTED_EXCEPTION



Message ID	Cause
ADD_BATCH_JOB	INSUFFICIENT_PRIVILEGE
ARCHIVE_LOG	UNEXPECTED_EXCEPTION
ARCHIVE_LOG_FILE_SIGN	UNEXPECTED_EXCEPTION
CANCEL_BATCH_JOB	INSUFFICIENT_PRIVILEGE
CANCEL_SCHEDULE_JOB	INSUFFICIENT_PRIVILEGE
CLU_AUDIT_LOG_COPY	UNEXPECTED_EXCEPTION
CLU_AUDIT_LOG_COPY	SIGNATURE_MISMATCH
CONDITION_EVALUATION	UNEXPECTED_EXCEPTION
CONF_METADATA_CHANGED	INSUFFICIENT_PRIVILEGE
CONF_METADATA_INSTALLED	INSUFFICIENT_PRIVILEGE
CONF_METADATA_INSTALLED	UNEXPECTED_EXCEPTION
CONF_METADATA_REMOVED	INSUFFICIENT_PRIVILEGE
CONF_METADATA_REMOVED	UNEXPECTED_EXCEPTION
CONF_READ	INSUFFICIENT_PRIVILEGE
CONF_READ	CONFIG_VALUE_INVALID
CONF_READ	CONFIG_VALUE_NOT_DEFINED
CONF_READ	CONFIGURATION_SERVICE_NULL
CONF_VALUE_UPDATED	INSUFFICIENT_PRIVILEGE
CONN_POOL_GET_CONNECTION	UNEXPECTED_DATABASE_OPERATION_FAILURE
CONSOLE_INTEGRATION_INIT	UNEXPECTED_EXCEPTION
CONSOLE_INTEGRATION_REGCONSOLECONFIG	UNEXPECTED_EXCEPTION
CONSOLE_INTEGRATION_REGDOMENU	UNEXPECTED_EXCEPTION
CONSOLE_INTEGRATION_REGMENU	UNEXPECTED_EXCEPTION
CONSOLE_INTEGRATION_REGPAGEHELP	UNEXPECTED_EXCEPTION
CREATE_ADMIN_ROLE	UNEXPECTED_EXCEPTION
CREATE_ATTRIBUTE	JMS_UNABLE_TO_PUBLISH_ADMIN_EVENT
CREATE_ATTRIBUTE_CATEGORY	UNEXPECTED_EXCEPTION
CREATE_COMMAND_TARGET	CREATE_COMMAND_TARGET_FAILED_UNEXPECTEDLY
CREATE_GROUP	UNEXPECTED_LDAP_EXCEPTION

Message ID	Cause
CREATE_GROUP	JMS_UNABLE_TO_PUBLISH_ADMIN_EVENT
CREATE_GROUP	UNEXPECTED_EXCEPTION
CREATE_GROUP_PARTIAL_FAILURE	UNEXPECTED_DATABASE_OPERATION_FAILURE
CREATE_IDENTITY_SOURCE	JMS_UNABLE_TO_PUBLISH_ADMIN_EVENT
CREATE_PRINCIPAL	UNEXPECTED_LDAP_EXCEPTION
CREATE_PRINCIPAL	UNEXPECTED_DATABASE_OPERATION_FAILURE
CREATE_PRINCIPAL	UNEXPECTED_EXCEPTION
CREATE_PRINCIPAL_PREFERENCES	UNEXPECTED_EXCEPTION
CREATE_REALM	INSUFFICIENT_PRIVILEGE
CREATE_REALM_PREFERENCES	UNEXPECTED_EXCEPTION
CREATE_SECURITY_DOMAIN	INSUFFICIENT_PRIVILEGE
CREATE_TRUST	UNEXPECTED_EXCEPTION
CREATE_TRUST_DOMAIN	UNEXPECTED_EXCEPTION
CREATE_TRUST_INSTANCE	UNEXPECTED_EXCEPTION
DB_SPACE_USAGE_MANAGEMENT	UNEXPECTED_EXCEPTION
DELETE_ADMIN_ROLE	UNEXPECTED_EXCEPTION
DELETE_AGED_JOB	UNEXPECTED_EXCEPTION
DELETE_ATTRIBUTE	JMS_UNABLE_TO_PUBLISH_ADMIN_EVENT
DELETE_AUTH_POLICY	INSUFFICIENT_PRIVILEGE
DELETE_BATCH_JOB	INSUFFICIENT_PRIVILEGE
DELETE_GROUP	UNEXPECTED_LDAP_EXCEPTION
DELETE_GROUP	JMS_UNABLE_TO_PUBLISH_ADMIN_EVENT
DELETE_GROUP	UNEXPECTED_EXCEPTION
DELETE_IDENTITY_SOURCE	JMS_UNABLE_TO_PUBLISH_ADMIN_EVENT
DELETE_IDENTITY_SOURCE	UNEXPECTED_EXCEPTION
DELETE_IDENTITY_SOURCE	INSUFFICIENT_PRIVILEGE
DELETE_PRINCIPAL	UNEXPECTED_LDAP_EXCEPTION
DELETE_PRINCIPAL	JMS_UNABLE_TO_PUBLISH_ADMIN_EVENT
DELETE_PRINCIPAL	UNEXPECTED_EXCEPTION

Message ID	Cause
DELETE_PRINCIPAL_PREFERENCES	UNEXPECTED_EXCEPTION
DELETE_REALM	UNEXPECTED_EXCEPTION
DELETE_REALM	INSUFFICIENT_PRIVILEGE
DELETE_REALM_PREFERENCES	UNEXPECTED_EXCEPTION
DELETE_SCHEDULE_JOB	INSUFFICIENT_PRIVILEGE
DELETE_SECURITY_DOMAIN	UNEXPECTED_EXCEPTION
DELETE_SECURITY_DOMAIN	INSUFFICIENT_PRIVILEGE
DELETE_SIGNING_KEY	UNEXPECTED_EXCEPTION
DELETE_TRUST	UNEXPECTED_EXCEPTION
DELETE_TRUST_DOMAIN	UNEXPECTED_EXCEPTION
DELETE_TRUST_INSTANCE	UNEXPECTED_EXCEPTION
EXECUTE_BATCH_JOB	UNEXPECTED_EXCEPTION
EXECUTE_BATCH_JOB	UNEXPECTED_EXCEPTION
EXECUTE_COMMAND	UNEXPECTED_EXCEPTION
EXECUTE_SQL_SCRIPT	UNEXPECTED_DATABASE_OPERATION_FAILURE
EXTENSION_LOADED	UNEXPECTED_EXCEPTION
FEATURE_LICENSE_CHECK	LICENSE_MISSING_FOR_PRODUCT
FEATURE_LICENSE_CHECK	FEATURE_MISSING_STRATEGIES
FEATURE_LICENSE_CHECK	LICENSE_STRATEGY_FAILED
FEATURE_LICENSE_CHECK	LICENSE_UNABLE_TO_LOAD_STRATEGY
GENERATE_SCRIPT	SCRIPT_GENERATION_FAILED
INITIALIZE_PERMISSIONS	UNEXPECTED_EXCEPTION
JMS_CONTENTS_DOWNLOAD	UNEXPECTED_EXCEPTION
JMS_HANDLE_EVENT	JMS_INCORRECT_MESSAGE_TYPE
JMS_HANDLE_EVENT	UNEXPECTED_EXCEPTION
JMS_HANDLE_EVENT	JMS_UNABLE_TO_VALIDATE_SIGNATURE
JMS_HANDLE_EVENT	JMS_UNABLE_TO_PROCESS_EVENT
JMS_HANDLE_EVENT	JMS_UNABLE_TO_RECREATE_EVENT
JMS_INIT	CONFIG_VALUE_NOT_DEFINED

Message ID	Cause
JMS_INIT	JMS_UNABLE_TO_ACCESS_BEAN_INSTANCE
JMS_INIT	JMS_UNABLE_TO_CREATE_INITIAL_CONTEXT
JMS_INIT	JMS_JNDI_LOOKUP_FAILED
JMS_INIT	JMS_OPERATION_FAILED
JMS_PUBLISH_EVENT	JMS_UNABLE_TO_PROCESS_EVENT
JMS_PUBLISH_EVENT	UNEXPECTED_EXCEPTION
JMS_SCHEDULE_DEFERRED_CONTENTS_DOWNLOAD	UNEXPECTED_EXCEPTION
JMS_SEND_BATCH_NOTIFICATION	UNEXPECTED_EXCEPTION
KM_KEY_BIND	KM_KEY_NAME_ALREADY_BOUND
KM_KEY_BIND	UNEXPECTED_DATABASE_OPERATION_FAILURE
KM_KEY_BIND	INSUFFICIENT_PRIVILEGE
KM_KEY_FETCH	KM_KEY_NOT_FOUND
KM_KEY_FETCH	UNEXPECTED_DATABASE_OPERATION_FAILURE
KM_KEY_FETCH	INSUFFICIENT_PRIVILEGE
KM_KEY_PASSWORD_RESET	KM_KEY_NOT_FOUND
KM_KEY_PASSWORD_RESET	UNEXPECTED_DATABASE_OPERATION_FAILURE
KM_KEY_PASSWORD_RESET	INSUFFICIENT_PRIVILEGE
KM_KEY_UNBIND	KM_KEY_NOT_FOUND
KM_KEY_UNBIND	UNEXPECTED_DATABASE_OPERATION_FAILURE
KM_KEY_UNBIND	INSUFFICIENT_PRIVILEGE
KM_KEY_UPDATE	KM_KEY_NOT_FOUND
KM_KEY_UPDATE	UNEXPECTED_DATABASE_OPERATION_FAILURE
KM_KEY_UPDATE	INSUFFICIENT_PRIVILEGE
LICENSE_CHECK	LICENSE_MISSING_FOR_PRODUCT
LICENSE_CHECK	LICENSE_MISSING_STRATEGIES
LICENSE_CHECK	LICENSE_STRATEGY_FAILED
LICENSE_CHECK	LICENSE_UNABLE_TO_LOAD_STRATEGY
LICENSE_CHECK	LICENSE_CHECK_DISABLED
LICENSE_GET	LICENSE_INVALID



Message ID	Cause
LICENSE_INSTALL	INSUFFICIENT_PRIVILEGE
LICENSE_REPLACEMENT	INSUFFICIENT_PRIVILEGE
LICENSE_UNINSTALL	INSUFFICIENT_PRIVILEGE
LINK_GROUP_GROUP	JMS_UNABLE_TO_PUBLISH_ADMIN_EVENT
LINK_GROUP_GROUP	UNEXPECTED_LDAP_EXCEPTION
LINK_GROUP_GROUP	UNEXPECTED_EXCEPTION
LINK_GROUP_PRINCIPAL	JMS_UNABLE_TO_PUBLISH_ADMIN_EVENT
LINK_GROUP_PRINCIPAL	UNEXPECTED_LDAP_EXCEPTION
LINK_GROUP_PRINCIPAL	UNEXPECTED_EXCEPTION
LINK_IDENTITY_SOURCES	UNEXPECTED_EXCEPTION
LINK_PRINCIPAL_ADMIN_ROLE	UNEXPECTED_EXCEPTION
LINK_SECURITY_DOMAIN_POLICIES	INSUFFICIENT_PRIVILEGE
PROCESS_SECURITY_QUESTIONS	UNEXPECTED_EXCEPTION
PROCESS_SECURITY_QUESTIONS	INVALID_SECURITY_QUES_ANSWER
PROCESS_SECURITY_QUESTIONS	MALFORMED_SECURITY_QUES_STRING
PULL_FROM_REPLICAS	UNEXPECTED_EXCEPTION
READ_ADMIN_ROLE	UNEXPECTED_EXCEPTION
READ_ATTRIBUTE	UNEXPECTED_EXCEPTION
READ_AUTHENTICATORS	NOT_FOUND
READ_AUTHENTICATORS	INSUFFICIENT_PRIVILEGE
READ_AUTHENTICATORS	UNEXPECTED_EXCEPTION
READ_BATCH_JOB	NOT_FOUND
READ_BATCH_JOB	UNEXPECTED_EXCEPTION
READ_BATCH_JOB	INSUFFICIENT_PRIVILEGE
READ_GROUP	UNEXPECTED_LDAP_EXCEPTION
READ_GROUP	INVALID_GROUP_OBJECT_CLASSES
READ_GROUP	UNEXPECTED_EXCEPTION
READ_IDENTITY_SOURCE	UNEXPECTED_LDAP_EXCEPTION
READ_IDENTITY_SOURCE	UNEXPECTED_EXCEPTION

Message ID	Cause
READ_PRINCIPAL	INVALID_PRINCIPAL_OBJECT_CLASSES
READ_PRINCIPAL	INVALID_ATTRIBUTE_VALUE
READ_PRINCIPAL	UNEXPECTED_LDAP_EXCEPTION
READ_PRINCIPAL	UNEXPECTED_EXCEPTION
READ_PRINCIPAL	MULTIPLE_PRINCIPALS_HAVE_SAME_EXUID
READ_PRINCIPAL_PREFERENCES	UNEXPECTED_EXCEPTION
READ_REALM	UNEXPECTED_EXCEPTION
READ_REALM_PREFERENCES	UNEXPECTED_EXCEPTION
READ_REALM_PREFERENCES	INSUFFICIENT_PRIVILEGE
READ_SCHEDULE_JOB	INSUFFICIENT_PRIVILEGE
READ_SECURITY_DOMAIN	UNEXPECTED_EXCEPTION
READ_TRUST	UNEXPECTED_EXCEPTION
READ_TRUST_DOMAIN	UNEXPECTED_EXCEPTION
READ_TRUST_INSTANCE	UNEXPECTED_EXCEPTION
REGISTER_PRINCIPAL	UNEXPECTED_EXCEPTION
REGISTRY_COMPONENT_DEREGISTER	UNEXPECTED_EXCEPTION
REGISTRY_COMPONENT_DEREGISTER	INSUFFICIENT_PRIVILEGE
REGISTRY_COMPONENT_LOOKUP	UNEXPECTED_EXCEPTION
REGISTRY_COMPONENT_REGISTRATION	UNEXPECTED_EXCEPTION
REGISTRY_COMPONENT_REGISTRATION	INSUFFICIENT_PRIVILEGE
REGISTRY_COMPONENT_UPDATE	INSUFFICIENT_PRIVILEGE
REGISTRY_COMPONENT_UPDATE	UNEXPECTED_EXCEPTION
REGISTRY_INIT_DEPLOYMENT_UUID	UNEXPECTED_EXCEPTION
REGISTRY_INITIALIZATION	HOME_INSTANCE_UNKNOWN
REGISTRY_INITIALIZATION	UNEXPECTED_EXCEPTION
REGISTRY_INSTANCE_DEREGISTER	UNEXPECTED_EXCEPTION
REGISTRY_INSTANCE_DEREGISTER	JMS_UNABLE_TO_PUBLISH_ADMIN_EVENT
REGISTRY_INSTANCE_DEREGISTER	INSUFFICIENT_PRIVILEGE
REGISTRY_INSTANCE_LOOKUP	UNEXPECTED_EXCEPTION



Message ID	Cause
REGISTRY_INSTANCE_REGISTRATION	UNEXPECTED_EXCEPTION
REGISTRY_INSTANCE_UPDATE	UNEXPECTED_EXCEPTION
REGISTRY_INSTANCE_UPDATE	INSUFFICIENT_PRIVILEGE
REGISTRY_PATCH_ADD	INSUFFICIENT_PRIVILEGE
REGISTRY_PATCH_ADD	UNEXPECTED_EXCEPTION
REGISTRY_PATCH_REMOVE	INSUFFICIENT_PRIVILEGE
REGISTRY_PATCH_REMOVE	UNEXPECTED_EXCEPTION
REGISTRY_TOPOLOGY_READ	UNEXPECTED_EXCEPTION
REGISTRY_TOPOLOGY_READ	INSUFFICIENT_PRIVILEGE
REGISTRY_TOPOLOGY_WRITE	UNEXPECTED_EXCEPTION
REGISTRY_TOPOLOGY_WRITE	JMS_UNABLE_TO_PUBLISH_ADMIN_EVENT
REMOVE_ORPHANED_PRINCIPALS	JMS_UNABLE_TO_PUBLISH_ADMIN_EVENT
REPLICATION_SYSTEM_SETUP	FAILED_REMOVE_CONFIGURATION
REPLICATION_SYSTEM_SETUP	FAILED_GENERATING_DATA
REPLICATION_SYSTEM_SETUP	FAILED_ADD_REPLICA
REPLICATION_SYSTEM_SETUP	SITE_NOT_IN_ATTACHMENT_PROCESS
REPLICATION_SYSTEM_SETUP	FAILED_SETUP_PRIMARY
REPLICATION_SYSTEM_SETUP	PRIMARY_DOES_NOT_EXISTS
REPLICATION_SYSTEM_SETUP	SITE_IN_PROMOTED_MODE
REPLICATION_SYSTEM_SETUP	SYSTEM_DOES_NOT_EXISTS
REPLICATION_SYSTEM_SETUP	FAILED_SYNCHRONIZE_REPLICA
REPLICATION_SYSTEM_SETUP	SITE_IN_ATTACHMENT_MODE
REPLICATION_SYSTEM_SETUP	SITE_NOT_IN_ATTACHMENT_MODE
REPLICATION_SYSTEM_SETUP	CANT_ATTACH_PRIMARY_SITE
REPLICATION_SYSTEM_SETUP	FAILED_ATTACH_REPLICA
REPLICATION_SYSTEM_SETUP	FAILED_PROMOTION
SCHEDULE_BATCH_JOB	INSUFFICIENT_PRIVILEGE
SEND_SMTP_MESSAGE	SMTP_INVALID_ARGUMENTS
SEND_SMTP_MESSAGE	SMTP_NOT_CONFIGURED

Message ID	Cause
SEND_SMTP_MESSAGE	UNEXPECTED_EXCEPTION
SEND_SMTP_MESSAGE	INVALID_RECIPIENT_ADDRESSES
SESSION_ADD_CONFIGURATION	INSUFFICIENT_PRIVILEGE
SESSION_ADD_LIFETIME_CONFIGURATION	INSUFFICIENT_PRIVILEGE
SESSION_DELETE_CONFIGURATION	INSUFFICIENT_PRIVILEGE
SESSION_DELETE_LIFETIME_CONFIGURATION	INSUFFICIENT_PRIVILEGE
SESSION_READ_CONFIGURATION	INSUFFICIENT_PRIVILEGE
SESSION_READ_LIFETIME_CONFIGURATION	INSUFFICIENT_PRIVILEGE
SESSION_SEARCH_CONFIGURATION	INSUFFICIENT_PRIVILEGE
SESSION_SEARCH_LIFETIME_CONFIGURATION	INSUFFICIENT_PRIVILEGE
SESSION_UPDATE_CONFIGURATION	INSUFFICIENT_PRIVILEGE
SESSION_UPDATE_LIFETIME_CONFIGURATION	INSUFFICIENT_PRIVILEGE
SMTP_CONNECT	MISSING_SMTP_CREDENTIALS
SMTP_CONNECT	INVALID_MAIL_PROTOCOL
SMTP_CONNECT	UNEXPECTED_EXCEPTION
SNMP_AGENT_START	UNEXPECTED_EXCEPTION
SNMP_READ_CONFIG	UNEXPECTED_EXCEPTION
SNMP_READ_CONFIG	CONFIG_VALUE_NOT_DEFINED
UNLINK_GROUP_GROUP	JMS_UNABLE_TO_PUBLISH_ADMIN_EVENT
UNLINK_GROUP_GROUP	UNEXPECTED_EXCEPTION
UNLINK_GROUP_PRINCIPAL	JMS_UNABLE_TO_PUBLISH_ADMIN_EVENT
UNLINK_GROUP_PRINCIPAL	UNEXPECTED_EXCEPTION
UNLINK_IDENTITY_SOURCES	UNEXPECTED_EXCEPTION
UNLINK_PRINCIPAL_ADMIN_ROLE	UNEXPECTED_EXCEPTION
UNREGISTER_GROUP	UNEXPECTED_DATABASE_OPERATION_FAILURE
UNREGISTER_GROUP	JMS_UNABLE_TO_PUBLISH_ADMIN_EVENT
UNREGISTER_PRINCIPAL	JMS_UNABLE_TO_PUBLISH_ADMIN_EVENT
UPDATE_ADMIN_ROLE	UNEXPECTED_DATABASE_OPERATION_FAILURE
UPDATE_ADMIN_ROLE	UNEXPECTED_EXCEPTION



Message ID	Cause
UPDATE_ATTRIBUTE	JMS_UNABLE_TO_PUBLISH_ADMIN_EVENT
UPDATE_ATTRIBUTE	UNEXPECTED_EXCEPTION
UPDATE_ATTRIBUTE_CATEGORY	UNEXPECTED_EXCEPTION
UPDATE_AUTHENTICATORS	UNEXPECTED_EXCEPTION
UPDATE_GROUP	UNEXPECTED_LDAP_EXCEPTION
UPDATE_GROUP	ERROR_LINK_AD_GROUPS
UPDATE_GROUP	JMS_UNABLE_TO_PUBLISH_ADMIN_EVENT
UPDATE_GROUP	UNEXPECTED_EXCEPTION
UPDATE_IDENTITY_SOURCE	JMS_UNABLE_TO_PUBLISH_ADMIN_EVENT
UPDATE_PRINCIPAL	UNEXPECTED_LDAP_EXCEPTION
UPDATE_PRINCIPAL	CONCURRENT_UPDATE_FKEY
UPDATE_PRINCIPAL	UNEXPECTED_EXCEPTION
UPDATE_PRINCIPAL	JMS_UNABLE_TO_PUBLISH_ADMIN_EVENT
UPDATE_PRINCIPAL	NOT_FOUND
UPDATE_PRINCIPAL_PREFERENCES	UNEXPECTED_EXCEPTION
UPDATE_REALM	INSUFFICIENT_PRIVILEGE
UPDATE_REALM_PREFERENCES	INSUFFICIENT_PRIVILEGE
UPDATE_REALM_PREFERENCES	UNEXPECTED_EXCEPTION
UPDATE_SECURITY_DOMAIN	UNEXPECTED_EXCEPTION
UPDATE_SECURITY_DOMAIN	INSUFFICIENT_PRIVILEGE
UPDATE_TRUST	UNEXPECTED_EXCEPTION
UPDATE_TRUST_DOMAIN	UNEXPECTED_EXCEPTION
UPDATE_TRUST_INSTANCE	UNEXPECTED_EXCEPTION
XML_SERIALIZER_BEAN_PARSING	UNEXPECTED_EXCEPTION
XML_SERIALIZER_CLASS_LOADING	UNEXPECTED_EXCEPTION
XML_SERIALIZER_CONFIGURATION	UNEXPECTED_EXCEPTION
XML_SERIALIZER_PARSING	UNEXPECTED_EXCEPTION

RSA Authentication Manager Administrative Audit Log Messages

The severity level of the following message IDs is FAILURE and WARNING.

Message ID	Cause
AUTHMGR_AGENT_GROUP_LINK_UNLINK	The user did not have sufficient privilege to link or unlink an agent and a group.
AUTHMGR_AGENT_CREATE	The user did not have sufficient privilege to create an agent.
AUTHMGR_AGENT_DELETE	The agent could not be deleted because it was in use.
AUTHMGR_AGENT_ENABLE	The agent could not be enabled because more than one administrator tried to edit it at the same time.
AUTHMGR_AGENT_ENABLE	The user did not have sufficient privilege to enable the agent.
AUTHMGR_AGENT_READ	The user did not have sufficient privilege to view the agent.
AUTHMGR_AGENT_CLEAR_NODESECRET	The user did not have sufficient privilege to clear the agent's node secret.
AUTHMGR_AGENT_UPDATE	The agent could not be updated because more than one administrator tried to edit it at the same time.
AUTHMGR_AGENT_CREATE	The user did not have sufficient privilege to create an agent.
AUTHMGR_AGENT_DELETE	The user did not have sufficient privilege to delete an agent.
AUTHMGR_AGENT_UPDATE	The user did not have sufficient privilege to update an agent.
AUTHMGR_AGENT_READ	The user did not have sufficient privilege to view an agent.
AUTHMGR_AGENT_READ	The agent data could not be found.
AUTHMGR_AGENT_UPDATE	An unexpected exception occurred while attempting to update an agent.
AUTHMGR_AGENT_LINK_APSLIST	The agent could not be linked to an agent protocol server list because more than one administrator tried to edit it at the same time.
AUTHMGR_AGENT_LINK_APSLIST	The agent could not be linked to an agent protocol server list because it could not be found.
AM_RADIUS_ATTRDEF_CREATE	The user did not have sufficient privilege to create a radius attribute.
AM_TOKEN_ATTRDEF_CREATE	The user did not have sufficient privilege to create a token attribute definition.
AM_RADIUS_ATTRDEF_DELETE	The user did not have sufficient privilege to delete a radius attribute definition.
AM_TOKEN_ATTRDEF_DELETE	The user did not have sufficient privilege to delete a token attribute definition.
AM_RADIUS_ATTRDEF_UPDATE	The user did not have sufficient privilege to update a radius attribute definition.



Message ID	Cause
AM_TOKEN_ATTRDEF_UPDATE	The user did not have sufficient privilege to update a token attribute definition.
AM_RADIUS_ATTRDEF_READ	The user did not have sufficient privilege to view a radius attribute definition.
AM_TOKEN_ATTRDEF_READ	The user did not have sufficient privilege to view a token attribute definition.
REMOTE_PRINCIPAL_ATTR_VALUE_CREATE	The user did not have sufficient privilege to create a remote principal attribute value.
REMOTE_PRINCIPAL_ATTR_VALUE_DELETE	The user did not have sufficient privilege to delete a remote principal attribute value.
REMOTE_PRINCIPAL_ATTR_VALUE_UPDATE	The user did not have sufficient privilege to update a remote principal attribute value.
REMOTE_PRINCIPAL_ATTR_VALUE_READ	The user did not have sufficient privilege to view a remote principal attribute value.
CREATE	The user did not have sufficient privilege to create an agent protocol server.
CREATE	The user did not have sufficient privilege to create an agent protocol server list.
CREATE	The user did not have sufficient privilege to create self service token information.
CREATE	The user did not have sufficient privilege to add a software token device type.
CREATE	The user did not have sufficient privilege to add a token type.
CREATE	The user did not have sufficient privilege to import software token device type definitions.
DELETE	The user did not have sufficient privilege to delete an agent protocol server list.
DELETE	The user did not have sufficient privilege to self service token information.
DELETE	The user did not have sufficient privilege to delete an agent protocol server.
DELETE	The user did not have sufficient privilege to delete a software token device type.
DELETE	The user did not have sufficient privilege to delete a token type.
READ	The user did not have sufficient privilege to view an agent protocol server.
READ	The user did not have sufficient privilege to view an agent protocol server list.

Message ID	Cause
AUTHMGR_REALM_ADD	The current license does not support multiple realms.
AUTHMGR_REALM_ADD	An unexpected exception occurred while attempting to check license support for multiple realms.
AM_CONFIGURATION_UPDATE_FAILED	The user did not have sufficient privilege to update realm configuration.
AM_CONFIGURATION_UPDATE_FAILED	The realm configuration data could not be found.
LINK_UNLINK	The user did not have sufficient privilege to linking or unlinking an agent with a trusted user group.
LINK_UNLINK	The user did not have sufficient privilege to linking or unlinking an agent with a user.
TRUSTED_USER_GROUP_CREATE	The user did not have sufficient privilege to creating a trusted user group.
TRUSTED_USER_GROUP_DELETE	The user did not have sufficient privilege to deleting a trusted user group.
TRUSTED_USER_GROUP_UPDATE	The user did not have sufficient privilege to updating a trusted user group.
TRUSTED_USER_GROUP_READ	The user did not have sufficient privilege to viewing a trusted user group.
TRUSTED_USER_GROUP_CREATE	A trusted group already exists with the supplied name.
TRUSTED_USER_GROUP_CREATE	An unexpected exception occurred while attempting to create trusted user group.
TRUSTED_USER_GROUP_DELETE	The trusted user group could not be found.
TRUSTED_USER_GROUP_DELETE	An unexpected exception occurred while attempting to delete the trusted user group.
TRUSTED_USER_GROUP_DELETE	The trusted user group was in use when an administrator tried to delete it.
TRUSTED_USER_GROUP_AGENT_LINK	An unexpected exception occurred while attempting to link an agent to a trusted user group.
TRUSTED_USER_GROUP_REMOTE_PRINCIPAL_LINK	An unexpected exception occurred while attempting to add a remote principal to a trusted user group.
TRUSTED_USER_GROUP_READ	The trusted user group could not be found.
TRUSTED_USER_GROUP_READ	An unexpected exception occurred while attempting to view a trusted user group.
TRUSTED_USER_GROUP_AGENT_UNLINK	The trusted user group or agent could not be found.
TRUSTED_USER_GROUP_REMOTE_PRINCIPAL_UNLINK	The trusted user group or trusted user could not be found.
TRUSTED_USER_GROUP_UPDATE	The trusted user group could not be found.



Message ID	Cause
TRUSTED_USER_GROUP_UPDATE	An unexpected exception occurred while attempting to update the trusted user group.
AUTHMGR_CTKIP_AUTHCODE_DELETE	The user did not have sufficient privilege to delete the ctkip authorization code.
AUTHMGR_CTKIP_AUTHCODE_DELETE	The ctkip authorization code was in use when the administrator tried to delete it.
AUTHMGR_CTKIP_MANAGEMENT	The user did not have sufficient privilege to manage a ctkip authorization code.
AUTHMGR_FILE_UPDATE	The user did not have sufficient privilege to update the file data.
AUTHMGR_CREATE_GROUP_RESTRICTED_ACCESS_HOURS	The user did not have sufficient privilege to add new restricted access times for a group.
AUTHMGR_UPDATE_GROUP_RESTRICTED_ACCESS_HOURS	The user did not have sufficient privilege to update the restricted access times for a group.
AUTHMGR_READ_GROUP_RESTRICTED_ACCESS_HOURS	The user did not have sufficient privilege to view the restricted access times for a group.
ACTIVATE_BCO	The user did not have sufficient privilege to enable On-Demand authentication for the user.
AUTHMGR_OFFLINE_AUTHN_POLICY_CREATE	The user did not have sufficient privilege to creating an offline authentication policy.
AUTHMGR_OFFLINE_AUTHN_POLICY_DELETE	The user did not have sufficient privilege to deleting an offline authentication policy.
AUTHMGR_OFFLINE_AUTHN_POLICY_UPDATE	The user did not have sufficient privilege to updating an offline authentication policy.
AUTHMGR_OFFLINE_AUTHN_POLICY_READ	The user did not have sufficient privilege to viewing an offline authentication policy.
AUTHMGR_OFFLINE_AUTHN_POLICY_UPDATE	An unexpected exception occurred while attempting to update an offline authentication policy.
AUTHMGR_OFFLINE_AUTHN_POLICY_UPDATE	Could not update the offline authentication policy because more than one administrator tried to edit it at the same time.
AUTHMGR_OFFLINE_AUTHN_POLICY_READ	An unexpected exception occurred while attempting to viewing an offline authentication policy.
CREATE	The user did not have sufficient privilege to add a token policy.
DELETE	The user did not have sufficient privilege to delete a token policy.
UPDATE	The user did not have sufficient privilege to update a token policy.
AUTHMGR_TOKEN_POLICY_READ	The user did not have sufficient privilege to read a token policy.
AUTHMGR_TOKEN_POLICY_UPDATE	The user did not have sufficient privilege to update a token policy.
AUTHMGR_TOKEN_POLICY_DELETE	The user did not have sufficient privilege to delete a token policy.

Message ID	Cause
AUTHMGR_TOKEN_POLICY_CREATE	The user did not have sufficient privilege to add a token policy.
UPDATE_AM_PRINCIPAL	The supplied static passcode does not meet policy requirements.
UPDATE_AM_PRINCIPAL	The user did not have sufficient privilege to update a user.
CLEARBADPASSCODES	The user did not have sufficient privilege to clear a principals bad passcodes.
MANAGE_PUK_IGNORE	A duplicate PUK was found and was not replaced.
PUK_INVALID_ARGUMENT_EXCEPTION	An unexpected exception occurred while attempting to add a new PUK.
PUK_INVALID_ARGUMENT_EXCEPTION	The user did not have sufficient privilege to add a new PUK.
MANAGE_PUK_LOOKUP	The user did not have sufficient privilege to view a PUK.
PUK_DUPLICATE_DATA_EXCEPTION	While updating a PUK, an existing PUK was found with the same value as the new values.
PUK_INVALID_ARGUMENT_EXCEPTION	An unexpected exception occurred while attempting to update a PUK.
PUK_OBJECT_IN_USE_EXCEPTION	The PUK could not be updated because it was in use.
MANAGE_PUK_IMPORT	No principal was found, and so permission to manage a PUK import was denied.
REALM_SETTINGS_READ	An unexpected exception occurred while attempting to read the realm settings.
REALM_SETTINGS_CREATE	The user did not have sufficient privilege to add new realm settings.
REALM_SETTINGS_DELETE	The user did not have sufficient privilege to delete realm settings.
REALM_SETTINGS_UPDATE	The user did not have sufficient privilege to update realm settings.
REALM_SETTINGS_READ	The user did not have sufficient privilege to view realm settings.
REMOTE_PRINCIPAL_CREATE	The user did not have sufficient privilege to create a new remote principal.
REMOTE_PRINCIPAL_DELETE	The user did not have sufficient privilege to delete a remote principal.
REMOTE_PRINCIPAL_UPDATE	The user did not have sufficient privilege to update a remote principal.
REMOTE_PRINCIPAL_READ	The user did not have sufficient privilege to view a remote principal.
REMOTE_PRINCIPAL_READ	An unexpected exception occurred while attempting to search remote principals.
REMOTE_PRINCIPAL_READ	The remote principal could not be found.
AUTHMGR_APS_UPDATE	Moving agent protocol servers to different hosts is not allowed.
AUTHMGR_APS_UPDATE	Changing which instance an agent protocol server is associated with is not allowed.



Message ID	Cause
AUTHMGR_APS_SYNCHRONIZATION	While rebalancing server lists, some servers did not get included in any list.
AUTHMGR_APS_SYNCHRONIZATION	The Agent Protocol Server List is local only because there were too many servers in the list.
AUTHMGR_AGENT_READ	The user did not have sufficient privilege to view an agent.
READ_AM_PRINCIPAL	The user did not have sufficient privilege to view a user.
CREATE_AM_TOKEN	The user did not have sufficient privilege to add a token.
DELETE_AM_TOKEN	The user did not have sufficient privilege to delete a token.
UPDATE_AM_TOKEN	The user did not have sufficient privilege to edit a token.
READ_AM_TOKEN_OFFLINE_EMERGENCY_ACCESS_INFO	The user did not have sufficient privilege to view a token's offline emergency access information.
READ_AM_TOKEN_ONLINE_EMERGENCY_ACCESS_INFO	The user did not have sufficient privilege to view a token's online emergency access information.
SYNC_TOKENS	The user did not have sufficient privilege to resynchronize a token.
AM_RESET_PIN	The user did not have sufficient privilege to reset a token's PIN.
AM_TOKEN_ENABLED	The user did not have sufficient privilege to enable or disable a token.
READ_TOKEN	The user did not have sufficient privilege to view a token.
LINK_UNLINK_TOKEN_PRINCIPAL	Assign a token to a user.
AM_TOKEN_GENERATE_ONLINE_EA	The user did not have sufficient privilege to manage online emergency access for the token.
AM_IMPORT_TOKEN_SKIPPED_IGNORE_MODE	While importing tokens, a duplicate was found. Importing of the duplicate token was skipped.
IMPORT_TOKEN	The token was skipped while importing due to a database transaction rollback.
EXPORT_SOFT_TOKEN	The user did not have sufficient privilege to export software tokens.
DISTRIBUTE_SOFT_TOKEN_CTKIP	The user did not have sufficient privilege to manage ctkip tokens.
SYNC_TOKENS	The user did not have sufficient privilege to view the tokens in the synchronization batch job.
SYNC_TOKENS	The user did not have sufficient privilege to view the associated principal as part of a synchronization batch job.
SYNC_TOKENS	The user did not have sufficient privilege to update the associated principal as part of a synchronization batch job.
SYNC_TOKENS	The user did not have sufficient privilege to update the tokens in the synchronization batch job.

Message ID	Cause
NEXT_AVAILABLE_AM_TOKEN	While trying to assign the next available token to a user, the user's security domain could not be found.
SEARCH_AM_TOKEN	The search domain is not valid.
AUTHMGR_TOKENTYPES_MANAGEMENT	The user did not have sufficient privilege to manage ctkip authcodes.
MIGRATION_INSUFFICIENT_PRIVILEGE	The user did not have sufficient privilege to perform a 6.1 migration.
AM_RADIUS_CREATE_PROFILE	An unexpected exception occurred while attempting to add a new radius profile.
AM_RADIUS_DELETE_PROFILE	An unexpected exception occurred while attempting to delete a radius profile.
AM_RADIUS_UPDATE_PROFILE	An unexpected exception occurred while attempting to update a radius profile.
AM_RADIUS_CREATE_SERVER	The user did not have sufficient privilege to add a radius server.
AM_RADIUS_DELETE_SERVER	The radius server object was in use and could not be deleted.
AM_RADIUS_UPDATE_SERVER	The user did not have sufficient privilege to update a radius server.
AM_RADIUS_CREATE_SERVER	The user did not have sufficient privilege to add a radius server.
AM_RADIUS_DELETE_SERVER	The user did not have sufficient privilege to delete a radius server.
AM_RADIUS_VIEW_SERVER	The user did not have sufficient privilege to view a radius server.
AM_RADIUS_CREATE_CLIENT	The user did not have sufficient privilege to create a radius client.
AM_RADIUS_UPDATE_CLIENT	The user did not have sufficient privilege to update a radius client.
AM_RADIUS_VIEW_CLIENT	The user did not have sufficient privilege to view a radius client.
AM_RADIUS_DELETE_CLIENT	The user did not have sufficient privilege to delete a radius client.
AM_RADIUS_CREATE_PROFILE	The user did not have sufficient privilege to create a radius profile.
AM_RADIUS_UPDATE_PROFILE	The user did not have sufficient privilege to update a radius profile.
AM_RADIUS_UPDATE_PROFILE	The user did not have sufficient privilege to view a radius profile.
AM_RADIUS_DELETE_PROFILE	The user did not have sufficient privilege to delete a radius profile.
AM_RADIUS_CREATE_POLICY	The user did not have sufficient privilege to create a radius policy.
AM_RADIUS_UPDATE_POLICY	The user did not have sufficient privilege to update a radius policy.
AM_RADIUS_VIEW_POLICY	The user did not have sufficient privilege to view a radius policy.
AM_RADIUS_DELETE_POLICY	The user did not have sufficient privilege to delete a radius policy.
AM_REPORT_SECDOMAIN_LOOKUP	The user did not have sufficient privilege to view the specified security domain while running a report on tokens and users.

Message ID	Cause
AM_REPORT_IDENTITY_LOOKUP	The user did not have sufficient privilege to view the specified identity source while running a report on tokens and users.
MANAGE_SMS_AUTHENTICATOR	The user did not have sufficient privilege to manage On-Demand authenticators.
MANAGE_SMS_PIN	The user did not have sufficient privilege to manage On-Demand authentication PINs.
AM_ON_DEMAND_CONFIGURATION_UPDATE	The admin must be system or super-admin to manage SMS configuration.
TRANSMIT_TEST_TXT_MSG_SMS	The test On-Demand authentication text message failed to send.

RSA Authentication Manager Runtime Audit Log Messages

The severity level of the following message IDs is FAILURE and WARNING.

Message ID	Cause
TR_R_REMOTE_PRINCIPAL_NOT_DISCOVERED	The user could not be discovered on any Trusted Realms.
TR_R_REALM_DISABLED	The remote Trusted Realm is disabled for authentication.
TR_R_LOCAL_REALM_DISABLED	The local realm is disabled for remote authentications.
ACM_NEW_PIN_REJECTED	The remote realm rejected the new pin.
ACM_ACCESS_DENIED	The authentication attempt failed in the remote realm.
AUTH_AGENT_LOOKUP	No agent could be found for the IP address.
AUTH_UDP_PACKET_PROCESSING	An unexpected exception occurred while processing the authentication packet.
AUTH_LOG_REQUEST	A log request was sent directly by the agent to the server.
AUTH_LOG_REQUEST	An invalid log request was sent.
AUTH_UNSUPPORTED_PROTOCOL	An authentication was requested using an unsupported protocol.
AUTH_AGENT_LOG_REQUEST_FAIL	An unexpected exception occurred while processing an agent log request.
AUTH_AGENT_ACCESS_CHECK	The agent was disabled in the middle of a two step authentication.
AUTH_AGENT_ACCESS_CHECK	Could not lookup the agent by the IP provided in the authentication packet.
AUTH_AGENT_TRUSTED_USER_ACCESS_CHECK	The agent is restricted and the trusted user does not belong to a trusted user group that has been activated for the agent.
AUTH_AGENT_ACCESS_CHECK	The agent is disabled.

Message ID	Cause
AUTH_NODE_VERIFICATION	The agent's node secret stored on the server is not the node secret used to encrypt the packet.
AUTH_PRINCIPAL_RESOLUTION	The trusted user could not be found.
AUTH_SESSION_OPEATION_FAILURE	An unexpected exception occurred while retrieving the session data for the second step of an authentication.
AUTH_SECONDARY_SEGMENT_PROCESSING_FAILURE	There was an error while processing the secondary segments of an authentication request.
AUTHMGR_PASSCODE_REUSE	Authentication failed because the passcode had already been used.
AUTOREG_UPDATE_FAILED	Automatically updating agent IPs has been disabled for the realm.
AGENT_AUTO_REG_START	Auto registration was not started on an SSL socket.
AGENT_AUTO_REG_START	Auto registration did not start as it disabled on the realm.
AUTOREG_VERIFY_NODESECRET	During agent auto registration, the node secret sent by the agent did not match the secret on the server.
AUTOREG_UNASSIGN_IP	Auto registration attempted to register a new agent that had an existing agent's IP, but the existing agent's IP is protected.
AUTOREG_UNASSIGN_IP	Auto registration attempted to register a new agent that has the same IP as a server instance.
AUTOREG_UNASSIGN_IP	Auto registration attempted to register a new agent that had an existing agent's IP, but the existing agent is in a different realm.
AUTOREG_DHCP_ERROR	While looking up an agent as part of auto registration, an unrecoverable DHCP configuration error was encountered.
AUTOREG_AGENT_NOT_FOUND	The agent sent a node secret during an auto registration but the agent record could not be found.
AUTO_REG_DUPLICATE_AGENT	Auto registration tried to update an agent that was never used for authentication.
AUTO_REG_DUPLICATE_IP	An agent with no node secret tried to register as a new agent but an existing agent with the same IP was found.
NO_MORE_OTT	The principal trying to authenticate does not have any more one time tokencodes to use.
TFT_EXPIRED	A temporary fixed tokencode was deleted because it had expired.
OTTS_EXPIRED	A one time tokencode was deleted because it had expired.
TOKEN_EXPIRED	Authentication failed because the token had expired.
AUTHMGR_PASSCODE_REUSE	Authentication failed because the passcode was already used.
AUTH_FAILED_BAD_TOKENCODE_GOOD_PIN	Authentication failed because the pin was correct but the tokencode was incorrect.



Message ID	Cause
AUTH_FAILED_BAD_PIN_GOOD_TOKENCODE	Authentication failed because the tokencode was correct but the pin was not.
AUTH_FAILED_BAD_PIN_PREVIOUS_TOKENCODE	Authentication failed because the pin was bad and the tokencode was previously used.
AUTHN_LOGIN_EVENT	The peek ahead limit was exceeding while generating an On-Demand tokencode.
AUTHN_LOGIN_EVENT	The On-Demand tokencode destination is invalid.
AUTHN_LOGIN_EVENT	The On-Demand authenticator does not have a PIN set.
TR_H_AUTHMGR_PIN_CHANGE	Pin change attempt failed.
AUTHMGR_PIN_CHANGE	Pin change attempt failed.
AUTHMGR_SMS_PIN_CHANGE	Pin change attempt failed.
OA_DATA_DOWNLOAD_FAILED	Offline authentication data was not sent to the agent because it could not provide valid proof of authentication.
OA_DATA_DOWNLOAD_FAILED	An unexpected exception occurred while attempting to send the agent its offline authentication data.
OA_DATA_DOWNLOAD_FAILED	The ticket supplied by the agent was invalid.

RSA Authentication Manager System Log Messages

The severity level of the following message IDs is FAILURE and WARNING.

Message ID	Cause
CREATE_TRUST	An unexpected exception occurred while attempting to handle a request to insert a new legacy realm.
ADJUDICATOR_FAILOVER	The adjudicator was set to offline mode.
ADJUDICATOR_PROCESS	The adjudicator encountered an error while attempting to adjudicate an authentication.
ADJUDICATOR_SERVICE_SHUTDOWN	The adjudicator service was shutdown.
ADJUDICATOR_CONFIGURATION	Invalid adjudicator manager configuration.
ADJUDICATOR_CONFIGURATION	IPE while checking if the adjudicator is running on a primary server.
ADJUDICATOR_CONFIGURATION	An unexpected datastore exception occurred.
ADJUDICATOR_CONFIGURATION	IPE while trying to access the adjudicator configuration.
ADJUDICATOR_CONFIGURATION	An unexpected exception occurred while reloading the adjudicator configuration.

Message ID	Cause
ADJUDICATOR_REHOMING	An unexpected exception occurred while scheduling a rehomining operation.
ADJUDICATOR_INSTANCE_CONFIGURATION	Attempting to fetch the configuration during a replica add/remove.
ADJUDICATOR_CONFIGURATION	Attempting to create a user home definition with an invalid node.
ADJUDICATOR_CONFIGURATION	An unexpected exception while gathering rehomining data.
UPDATE_AM_PRINCIPAL	Failed to update the principal while performing a rehomining operation.
ADJUDICATOR_CLOCK_SETBACK	The system time was set back.
ADJUDICATOR_TIME_CHECK	The adjudicator's set time process failed.
ADJUDICATOR_TIME_CHECK	An unexpected exception while handling a time check request.
AUTHMGR_BEAN_CONVERT	An unexpected exception occurred while copying properties from one object to another.
AUTHMGR_BEAN_CONVERT	There was a problem with the database while attempting to add an agent.
AUTHMGR_AGENT_READ	An unexpected datastore exception occurred while looking up the groups belonging to an agent.
AUTHMGR_AGENT_READ	An unexpected datastore exception occurred while trying to determine the access privileges of an authenticating principal attempting to authenticate.
AUTHMGR_AGENT_UPDATE	More than one administrator attempted to update an agent.
AUTHMGR_AGENT_READ	An unexpected datastore exception occurred while looking up the trusted users that can authenticate on an agent.
AUTHMGR_AGENT_READ	An invalid GUID was supplied while searching for an agent.
AUTHMGR_AGENT_READ	An unexpected datastore exception occurred while searching agents.
AUTHMGR_REALM_PREDELETE_AGENT_DELETE	An exception occurred while deleting agents as part of a realm removal.
AUTHMGR_AGENT_READ	IPE while looking up an agent's server list.
AUTHMGR_AGENT_READ	The agent's server list could not be found.
AUTHMGR_AGENT_READ	An unexpected exception occurred while looking up an agent's server list.
AUTHMGR_AGENT_LINK_APSLIST	An unexpected exception occurred while modifying the agent/server list association.
SYNC_TOKENS	Could not find the sync tokens job report file.
FEATURE_LICENSE_CHECK	An unexpected exception occurred while checking the product license.
AUTHMGR_SD_PREDELETE_VALIDATION	Attempting to delete a realm that still contains domain objects.



Message ID	Cause
AUTHMGR_BEAN_CONVERT	An unexpected exception occurred while attempting to copy properties from one object to another.
PROCESS_REFERENTIAL_INTEGRITY_MESSAGES	No referential integrity event has been received in the last window.
PROCESS_REFERENTIAL_INTEGRITY_MESSAGES	An unexpected exception occurred while checking referential integrity of the database.
TRUSTED_USER_GROUP_CREATE	An unexpected datastore exception occurred while adding a trusted user group.
TRUSTED_USER_GROUP_CREATE	An unexpected datastore exception occurred while deleting a trusted user group.
TRUSTED_USER_GROUP_UPDATE	An unexpected datastore exception occurred while linking a trusted user group to an agent.
TRUSTED_USER_GROUP_UPDATE	An unexpected datastore exception occurred while linking a trusted user group to a remote principal.
TRUSTED_USER_GROUP_READ	An unexpected datastore exception occurred while looking up a trusted user group.
TRUSTED_USER_GROUP_UPDATE	An unexpected datastore exception occurred while unlinking an agent from a trusted user group.
TRUSTED_USER_GROUP_UPDATE	An unexpected datastore exception occurred while unlinking a remote principal from a trusted user group.
TRUSTED_USER_GROUP_UPDATE	An unexpected datastore exception occurred while updating a trusted user group.
TRUSTED_USER_GROUP_UPDATE	An unexpected datastore exception occurred while updating the restricted access times for a trusted user group.
AUTHMGR_HOST_DELETE	One administrator attempted to delete the server list while another attempted to delete a host referenced by the server list.
AUTHMGR_APS_DELETE	An unexpected datastore exception occurred while deleting a server list.
FEATURE_LICENSE_CHECK	An unexpected exception occurred while checking the product license.
LOCATE_SMS_AUTHENTICATOR_COUNT	The system configuration does not contain the number of on demand tokencode users.
GET_REG_USERS	The system configuration does not contain the number of registered users.
LOCATE_REALM_DEFAULT_OA_POLICY	No default offline authentication policy could be found for a realm.q
CREATE_REALM	Attempting to make more than one policy of the same type the default policy for a realm.
LOCATE_REALM_DEFAULT_TOKEN_POLICY	Could not locate a realm's default token authentication policy.
READ_OA_POLICY	An unexpected exception occurred while retrieving the offline authentication policy applying to a principal.

Message ID	Cause
CREATE_AM_PRINCIPAL	Attempting to add or edit a user when the active user limit for your license has been exceeded.
CREATE_AM_PRINCIPAL	An unexpected exception occurred while assigning a user a token.
AUTHMGR_TRUST_PREDELETE_VALIDATION	Attempting to delete a trust that still contains remote principals.
REMOTE_PRINCIPAL_READ	An unexpected datastore exception occurred while searching for remote principals belonging to a trusted user group.
REMOTE_PRINCIPAL_READ	An unexpected datastore exception occurred while searching for trusted user groups that a remote principal belongs to.
AUTHMGR_APS_LIST_CONFIG_READ	An unexpected exception occurred while trying to read agent protocol server lists.
AUTHMGR_APS_LIST_UPDATE	An unexpected exception occurred while refreshing the server list.
READ_SERVER_LIST	More than one agent protocol server list was found.
AUTHMGR_APS_SYNCHRONIZATION	The primary server could not be determined while performing agent protocol server synchronization.
AUTHMGR_APS_SYNCHRONIZATION	An agent protocol server list was added that shares a hostname with an existing agent.
AUTHMGR_APS_SYNCHRONIZATION	One of the IP addresses in an agent protocol server list could not be resolved to a hostname.
PROCESS_REFERENTIAL_INTEGRITY_MESSAGES	An unexpected exception occurred while handling a referential integrity message.
IMPORT_TOKEN	The token xml file has been corrupted.
IMPORT_TOKEN	An unexpected exception occurred while overwriting tokens during a token import.
IMPORT_TOKEN	An unexpected exception occurred while importing tokens.
EXPORT_SOFT_TOKEN	Could not locate the principal assigned to a software token during distribution.
SYNC_TOKENS	Could not find the output file for a token synchronization job report.
SYNC_TOKENS	The token's security domain ID did not resolve correctly to a security domain.
SYNC_TOKENS	An unexpected exception occurred while looking up principals as part of a token synchronization batch job.
SYNC_TOKENS	The input file supplying token serial numbers to a batch token synchronization job is malformed.
READ_AM_TOKEN_EMERGENCY_ACCESS_INFO	The token's offline authentication policy could not be found.
UPDATE_AM_TOKEN_OFFLINE_EMERGENCY_ACCESS_INFO	An unexpected exception occurred while updating the token's offline authentication policy.



Message ID	Cause
AM_LINK_TOKEN_PRINCIPAL	An error occurred while checking the product license.
AM_LINK_TOKEN_PRINCIPAL	The active user limit was exceeded while attempting to assign a token to a user.
AUTHMGR_REALM_PREDELETE_TOKEN_DELETE	The token was in use when attempting to delete it in order to delete the security domain.
AUTHMGR_REALM_PREDELETE_TOKEN_DELETE	The token could not be found when attempting to delete it in order to delete the security domain.
AUTHMGR_REALM_PREDELETE_TOKEN_DELETE	An unexpected exception occurred while attempting to delete the token in order to delete the security domain.
ARCHIVE_LOG	An unexpected exception occurred while attempting to perform a log archive batch job.
AGENT_REQUEST_HANDLE	An error occurred while attempting to retransmit data.
AGENT_REQUEST_HANDLE	An error occurred while trying to process the request from the client.
AGENT_RESPONSE_SEND	The adjudicator operation did not complete while processing the response.
AGENT_RESPONSE_SEND	An unexpected exception occurred while attempting to process the response.
AGENT_RESPONSE_SEND	An unexpected exception occurred while attempting to send the response.
READ_TOKEN	An unexpected exception occurred while attempting to read the principal's tokens.
READ_TOKEN	IPE to view the requested token.
READ_TOKEN	An unexpected exception occurred while trying to view the requested token.
READ_REPLACEMENT_TOKEN	An unexpected exception occurred while attempting to find a replacement token.
READ_REPLACEMENT_TOKEN	IPE to search for a replacement token.
READ_AGENT	IPE to view agents.
READ_AGENT	An unexpected exception occurred while attempting to search for an agent by IP address.
READ_AGENT	An unexpected exception occurred while trying to lookup an agent.
AUTHMGR_AGENT_UPDATE	An unexpected exception occurred while attempting to update an agent.
AUTHMGR_AGENT_UPDATE	IPE to edit agents.
AUTHMGR_AGENT_UPDATE	IPE to edit agents or agent protocol server lists.
AUTHMGR_AGENT_UPDATE	An unexpected exception occurred while trying to update the agent's agent protocol server list.

Message ID	Cause
READ_AGENT_ACTIVATED_GROUPS	IPE to view agents.
READ_AGENT_ACTIVATED_GROUPS	An unexpected exception occurred while attempting to get agents activated for a group.
READ_TOKEN_POLICY	An unexpected exception occurred while retrieving the token policy for a principal.
READ_SERVER_LIST	An unexpected exception occurred while attempting to retrieve the home instance's agent protocol server list.
READ_AM_PRINCIPAL	IPE to view principals.
READ_AM_PRINCIPAL	An unexpected exception occurred while attempting to view the principal.
READ_OA_POLICY	An unexpected exception occurred while attempting to search for the principal's offline authentication policy.
READ_SERVER_CONFIG	An unexpected exception occurred while attempting to retrieve the server's configuration.
LICENSE_CHECK	Could not find the authenticator in the database.
VALIDATE_PASSCODE	An unexpected exception occurred while attempting to authenticate via SecurID.
VALIDATE_NEW_STATIC_PASSCODE	An unexpected exception occurred while attempting to authenticate with a static passcode.
VALIDATE_NEW_PIN	An unexpected exception occurred while attempting to validate the user supplied new pin.
VALIDATE_NEXT_TOKENCODE	An unexpected exception occurred while attempting to validate the second tokencode during authentication.
READ_AM_PRINCIPAL	Failed to look up the principal during a SecurID authentication.
ON_DEMAND_LOGIN_FAILURE	An unexpected exception occurred while processing an On-Demand authentication.
AUTH_AGENT_LOOKUP	Could not find the requested agent while attempting to authenticate.
AUTHMGR_SERVER_STARTUP	An unexpected exception occurred while starting the server.
AUTHMGR_SERVER_STARTUP	While starting the server, the license was determined to be in violation.
FEATURE_LICENSE_CHECK	An unexpected exception occurred while checking the license during server startup.
OA_SERVER_START	The server was interrupted during startup.
OA_SERVER_START	The server could not obtain the requested socket to listen on.
OA_SERVER_START	An unexpected exception occurred while starting the authentication server.
TCP_SERVER_SHUTDOWN	The server was stopped.



Message ID	Cause
OA_SERVER_START	The server did not start because the license check failed.
FEATURE_LICENSE_CHECK	An unexpected exception occurred while checking the license in order to start the authentication server.
CONNECTION_ERROR	A connection request was rejected by the authentication server.
CONNECTION_ERROR	The server failed to accept a connection because the SSL peer could not be verified.
TCP_SERVER_STARTUP	The server was started.
APS_SERVER_START	The server was interrupted during startup.
APS_SERVER_START	The server could not obtain the requested socket to listen on.
UDP_SERVER_SHUTDOWN	The server was stopped.
UDP_SERVER_STARTUP	The server was started.
UDP_SERVER_SHUTDOWN	The server shutdown because the socket was closed.
CONNECTION_ERROR	The connection was rejected because too many requests were pending.
REGISTRY_TOPOLOGY_READ	While checking if a host is a primary server, the hostname could not be found.
READ_GROUP	Failed to lookup group by id while logging.
LOOKUP_OBJECT	Failed to lookup the requested object while logging.
SESSION_CREATE	Could not create a new session.
CTKIP_SERVICE_PROCESS_REQUEST	The CTKIP client sent an invalid request, it will not be processed.
AUTHMGR_BEAN_CONVERT	Failed to copy properties from one object to another during migration.
CONNECTION_ERROR	A socket timed out while reading for an authentication.
CREATE_RADIUS_SERVER	An unexpected datastore exception occurred while attempting to create a RADIUS server.
FEATURE_LICENSE_CHECK	An unexpected exception occurred while checking the RADIUS license.
CREATE_RADIUS_SERVER	License check failed while creating a radius server.
CREATE_RADIUS_CLIENT	License check failed while creating a radius client.
CREATE_RADIUS_POLICY	License check failed while creating a radius policy.
CREATE_RADIUS_PROFILE	License check failed while creating a radius profile.
DELETE_RADIUS_SERVER	License check failed while deleting a radius server.
DELETE_RADIUS_CLIENT	License check failed while creating a radius client.
DELETE_RADIUS_POLICY	License check failed while creating a radius policy.

Message ID	Cause
DELETE_RADIUS_PROFILE	License check failed while creating a radius profile.
CREATE_RADIUS_CLIENT	License check failed while retrieving the list of all radius client's make/model.
CREATE_RADIUS_PROFILE	License check failed while getting a list of all radius attributes.
VIEW_RADIUS_POLICY	License check failed while getting a list of all radius policies.
VIEW_RADIUS_CLIENT	License check failed while getting a list of radius clients in a security domain.
LIST_RADIUS_CLIENTS_STATS	License check failed while fetching radius client statistics.
VIEW_RADIUS_PROFILE	License check failed while getting a list of all radius profiles.
CREATE_RADIUS_PROFILE	License check failed while getting a list of all radius attributes.
LIST_RADIUS_SERVER_STATS	License check failed while getting systemwide radius statistics.
LINK_RADIUS_CLIENT_AND_AGENT	License check failed while linking an agent to a radius client.
VIEW_RADIUS_SERVER	License check failed while looking up a radius server.
VIEW_RADIUS_CLIENT	License check failed while looking up a radius client.
VIEW_RADIUS_POLICY	License check failed while looking up a radius policy.
VIEW_RADIUS_PROFILE	License check failed while looking up a radius profile.
RADIUS_NOTIFY_REPLICATION_PKG	License check failed while notifying a radius replica about the replica package.
RADIUS_PUBLISH_REPLICATION_PKG	License check failed while forcing replication to radius servers.
LINK_RADIUS_PROFILE_WITH_USER	License check failed while unlinking a radius profile and a user.
UPDATE_RADIUS_SERVER	License check failed while updating a radius server.
LINK_RADIUS_PROFILE_WITH_AGENT	License check failed while linking a radius profile with an agent.
LINK_RADIUS_PROFILE_WITH_USER	License check failed while linking a radius profile with a user.
UPDATE_RADIUS_CLIENT	License check failed while updating a radius client.
UPDATE_RADIUS_POLICY	License check failed while updating a radius policy.
UPDATE_RADIUS_PROFILE	License check failed while updating a radius profile.
UPDATE_RADIUS_SERVER	License check failed while updating a radius server.
GENERATE_REPORT	An unexpected exception occurred while generating a report.
GENERATE_REPORT	A report could not find a resource bundle.
GENERATE_REPORT	Unable to clean up temporary records from the database.



Message ID	Cause
ADD_BATCH_JOB	An unexpected exception occurred while trying to start a On-Demand authenticator cleanup batch job.
CLEANUP_EXPIRED_AUTHENTICATORS_JOB_FAILURE	An unexpected exception occurred while attempting to delete an On-Demand authenticator for a user.
FEATURE_LICENSE_CHECK	An unexpected exception occurred while performing an On-Demand authentication license check.
SMS_FEATURE_ID	The license check had an unexpected exception or was found to be in violation for On-Demand authentication.
SMS_AUTHENTICATOR_COUNT_FEATURE_ID	The license check had an unexpected exception or there were too many users enabled for On-Demand authentication.
DISPATCH_MESSAGE	The requested On-Demand authentication transmission mechanism was not recognized.
DISPATCH_MESSAGE	An unexpected exception occurred while sending an On-Demand authentication message.
MESSAGE_HANDLER_HANDLE_MESSAGE	An unexpected exception occurred while sending an On-Demand authentication message.
MESSAGE_PROCESSOR_START	The On-Demand message processor failed to start.
MESSAGE_PROCESSOR_STOP	The On-Demand message processor failed to stop.
MESSAGE_PROCESSOR_DROP_MESSAGE	An On-Demand authentication message was dropped because the queue was full.
TRANSMIT_TXT_MSG_SMS	Unable to send the On-Demand authentication message to Clickatell.
SMS_CLICKATELL_SSL_INITIALIZATION_FAILURE	The Clickatell trust store could not be found.
SMS_CLICKATELL_SSL_INITIALIZATION_FAILURE	The Clickatell root certificate could not be found.
SMS_CLICKATELL_SSL_INITIALIZATION_FAILURE	An unexpected exception occurred while loading the Clickatell root certificate.
REFRESH_TRANSMISSION_PLUGINS	An unexpected exception occurred while refreshing the list of On-Demand authenticator droppable plugins.
TRANSMIT_TXT_MSG_SMTP	An unexpected exception occurred while sending the On-Demand authentication message over SMTP.



Troubleshooting

This appendix contains the following information on some common RSA Authentication Manager issues and their corresponding solutions:

- [Common Problems and Resolutions](#)
- [General Troubleshooting Tips](#)
- [User and Token-Related Resolutions](#)
- [System-Related Resolutions](#)

Common Problems and Resolutions

The following table lists common problems, their possible causes, and the corresponding resolutions. Topics are broken down into these categories:

- User and token-related
- System-related
- Identity source or LDAP
- RSA Credential Manager
- Microsoft Management Console (MMC).

Problem	Possible Cause	Resolution
User and Token-Related		
User cannot authenticate or user is getting an access denied message.	User is locked out of Authentication Manager for violating the lockout policy.	Assisting Users Who Have Been Locked Out of the System on page 94.
	User did not violate the Authentication Manager lockout policy, but did violate the external identity source lockout policy (for example, an Active Directory lockout policy).	Check the identity source policy, and unlock the user in the identity source if necessary.
	Token is out of sync with Authentication Manager.	Resynchronizing Tokens on page 103.

Problem	Possible Cause	Resolution
	<p>Token has expired.</p> <p>Note: If the token has expired, you see a log message in the audit log.</p>	<p>Assign a new token, and provide emergency access if necessary. See Providing Emergency Access on page 417.</p> <p>Note: To avoid having users with expired tokens, schedule a recurring report that shows tokens close to expiration. Be proactive and replace tokens before they expire.</p>
	<p>The user ID is too long.</p>	<p>Do not create a user ID longer than 48 characters.</p>
	<p>The firewall is not configured properly or the appropriate ports are not open.</p>	<p>Assessing the Impact of Firewalls on RSA Authentication Manager on page 412.</p>
	<p>IP name resolution or agent host name is entered incorrectly.</p>	<p>Name and IP Address Resolution in RSA Authentication Manager on page 420.</p>
	<p>The agent configuration file is corrupt or invalid.</p>	<p>Updating an Agent Configuration File on page 422.</p>
	<p>If the user is trying to access a restricted agent, and your deployment uses an Active Directory forest or Global Catalog for authentication, the default group type must be set to Universal.</p>	<p>Specify the default group type on the Add Identity Source page in the Operations Console. For more information, see “Adding an Identity Source in RSA Authentication Manager” on page 25.</p> <p>For more information on using Active Directory and Global Catalogs, see Appendix A, “Integrating Active Directory Forests.”</p>

Problem	Possible Cause	Resolution
	<p>If you have configured Authentication Manager to allow system-generated PINs, and your deployment includes RSA RADIUS, you must configure RADIUS to allow system-generated PINs.</p> <p>Authentication Manager is out of sync with Coordinated Universal Time (UTC). Note: If Authentication Manager is out of sync with UTC, all of the users are unable to authenticate.</p>	<p>By default, RADIUS does not allow system-generated PINs. Edit the RADIUS configuration file, securid.ini, to allow system-generated PINs. For more information, see “Using System-Generated PINs with RSA RADIUS” on page 48.</p> <p>Resynchronizing RSA Authentication Manager with Coordinated Universal Time on page 421.</p>
<p>User is being prompted to enter a second tokencode.</p>	<p>User has violated the token policy and incorrect passcodes must be cleared.</p> <p>Token is out of sync with Authentication Manager.</p> <p>Authentication Manager is out of sync with Coordinated Universal Time (UTC). Note: If Authentication Manager is out of sync with UTC, all of the users are prompted to enter a second tokencode or are unable to authenticate. The behavior they experience is based on the time discrepancy.</p>	<p>Clearing Incorrect Passcodes on page 106.</p> <p>Resynchronizing Tokens on page 103.</p> <p>Resynchronizing RSA Authentication Manager with Coordinated Universal Time on page 421.</p>
<p>User is being prompted to create a new PIN.</p>	<p>PIN has been cleared.</p> <p>User has a new token.</p>	<p>Instruct user how to create a new PIN.</p> <p>Instruct user how to create a PIN.</p>

Problem	Possible Cause	Resolution
An existing user (user exists in Active Directory) gets an error message saying that he or she cannot enroll in RSA Credential Manager.	The user account is disabled, locked, expired, or set to require users to change their password during the next logon in the Active Directory.	<p>If Active Directory has the “change password during next logon” option set, then users cannot enroll in Credential Manager. Clear this option so that users can enroll in Credential Manager.</p> <p>Resolve disabled, locked, or expired users in Active Directory to allow users to enroll in Credential Manager.</p>
When administering RSA Credential Manager, if you approve a request with two approval steps and then click the Submit and Continue button, an error message appears stating that the request is completed.	If a custom extension is added to a request with two approval steps to automatically approve the second approval step, there is a time delay that causes the error message to appear.	Wait and then refresh the RSA Security Console.
System-Related		
<p>Authentication Manager does not start.</p> <p>When Authentication Manager does not start, you may get a start-up error message telling you that the service failed to start.</p>	<p>The Authentication Manager server may fail to start for a variety of reasons (for example, minimum system requirements not met).</p>	<p>RSA Authentication Manager Does Not Start on page 418.</p>
	A required port may be in use.	<p>Shut down all of the Authentication Manager services and ensure that none of the ports are in use. See “Assessing the Impact of Firewalls on RSA Authentication Manager” on page 412.</p> <p>If you configured SNMP, ensure that the SNMP Adaptor Port you chose is not already in use. See “Configuring SNMP” on page 203.</p>

Problem	Possible Cause	Resolution
<p>“Unable to start Authentication Manager” message and log displays “Reached EOF” message in the Managed Server Log.</p>	<p>Minimum system requirements not met or the system was running other services.</p>	<p>Making Sure the RSA Authentication Manager Machine Meets Minimum System Requirements on page 407.</p> <p>Make sure CPU-intensive services are not running on the Authentication Manager server.</p>
<p>RSA Security Console does not start.</p>	<p>The RSA Security Console may fail to start for a variety of reasons. For example:</p> <ul style="list-style-type: none"> • The URL is incorrect. • Service start-up has been initiated, but the service has not been given the appropriate amount of time to complete the start-up process. In this case, either the message “page cannot be displayed” appears, or you get a message that the back-end servers are unavailable. 	<p>Allow at least five minutes for the service to start.</p> <p>RSA Security Console Does Not Start on page 419.</p>
<p>RSA Security Console does not start and browser displays a blank screen.</p>	<p>Browser is not configured properly.</p>	<p>Configuring Browser Settings for the RSA Security Console, RSA Operations Console, and RSA Self-Service Console on page 411.</p>
<p>The Microsoft Management Console snap-in does not start.</p>	<p>The Microsoft Management Console may fail to start for a variety of reasons. For example:</p> <ul style="list-style-type: none"> • IP address or name resolution issue • Network connectivity issue 	<p>RSA Authentication Manager Microsoft Management Console Snap-in Does Not Start on page 419.</p>
<p>When accessing the Security Console, an Internet Explorer message appears, warning you that there is a problem with the web site’s security certificate, and advises you not to continue to the web site.</p>	<p>Internet Explorer 7.0 is warning you that the certificate is not signed.</p>	<p>Add the self-signed root certificate generated at installation to the trusted root list.</p>

Problem	Possible Cause	Resolution
Security Console URL has red background; "Certificate Error" message appears.	Internet Explorer 7.0 is warning you that the certificate is not signed.	Add the self-signed root certificate generated at installation to the trusted root list.
Authentication Manager is very slow or poor performance.	Minimum system requirements are not met.	Making Sure the RSA Authentication Manager Machine Meets Minimum System Requirements on page 407.
Disk space is low or filling quickly.	The system logs need to be archived, the time between archives is too long, or both.	Change the log archiving settings. See " Archiving Log Files " on page 197.
	Additional logging or tracing has been enabled.	Be careful when configuring instance logging, as large logs take up a lot of disk space. The default values should be sufficient.
	The Oracle database archive logs are taking up too much space.	Call RSA Customer Support.
After rebooting and restarting the primary, the Oracle server appears to be running, but it has not started yet. When manually trying to restart the proxy and managed servers, an error occurs.		After rebooting the primary, stop and restart the Oracle server, and then start the other services.
Lower-level security domains do not have the parent domain attributes.	Lower-level security domains do not inherit attributes from parents.	As designed.
Cannot read the trace file.	The trace file is obfuscated.	Call RSA Customer Support.
"Error 404 - Not Found" encountered while accessing the Security Console after running the Manage Nodes utility.	Did not restart Authentication Manager services after server node removal.	Restart the Authentication Manager services after removing a server node using the Manage Nodes utility.

Problem	Possible Cause	Resolution
Trusted realm authentication fails.	If you clear the node secret in the trusted realm, trusted realm authentications fail.	Perform a local authentication in the trusted realm to reestablish the node secret. Subsequent trusted realm authentications should succeed.
When you enable or disable the Authentication Agent auto-registration feature, the change is not reflected when you view the Security Console on a replica instance.	Outdated information is stored in the system cache.	See the Operations Console Help topic "Flush the Cache."
SNMP gets for the total number of authentication policies are off by one count.	SNMP counts the system default policies that exist only in the internal database. For example, if the Security Console displays three password policies, the SNMP get value is four.	Subtract one from the SNMP count.
Newly added replicas are not appearing on the List Instances page in the Operations Console.		Log off of the Operations Console and then log on again to see the new instances.
When you enable or disable the automatic deletion of replaced tokens feature, the change is not reflected when you view the Security Console on a replica instance.	Outdated information is stored in the system cache.	See the Operations Console Help topic "Flush the Cache."

Problem	Possible Cause	Resolution
Identity Source or LDAP		
Users added using the directory console (for example, Active Directory) are not visible through the Security Console.	The new information does not immediately display in the Security Console because of the cache layer.	The information displays in the Security Console after a short delay.
User data that exists on the Authentication Manager primary does not appear in the replica or users cannot authenticate through the replica machine.	The changes made on the primary have not yet propagated to the replica due to the cache refresh interval.	All changes should be propagated within 10 minutes. You can flush the cache if you want to see the changes immediately. For more information, see the Operations Console Help topic "Flush the Cache."
If you attempt to add an identity source right after you have added a new replica, the directory connection settings do not appear for the new replica.		Log off of the Operations Console and then log on again. The replica directory settings are now listed.
If the server exception error PRINCIPAL_SEARCH_TIME_EXCEEDED appears, you must re-index your Sun Java System Directory Server, and increase the cache size to 100 MB.	Sun Java System Directory Server needs to be re-indexed, and the cache size needs to be increased.	Re-index the Sun Java System Directory Server, and increase the cache size. For instructions, see " RSA Security Console Times Out When Searching for Users " on page 419.
LDAP queries are failing after primary directory server failover.	When configured for failover, the system does not switch back to the primary directory server after failover, even if the primary is restored. Therefore, if the secondary directory server fails after the primary directory server is restored, the system does not attempt to access the primary directory server, causing queries to fail.	Restart the application server to switch back to the primary directory server. To avoid future service interruptions, restart the application server as soon as the primary directory server becomes operational.

Problem	Possible Cause	Resolution
RSA Credential Manager		
An existing user (user exists in Active Directory) gets an error message saying that he or she cannot enroll in RSA Credential Manager.	The user account is disabled, locked, expired, or set to require users to change their password during the next logon in the Active Directory.	If Active Directory has the “change password during next logon” option set, then users cannot enroll in Credential Manager. Clear this option so that users can enroll in Credential Manager. Resolve disabled, locked, or expired users in Active Directory to allow users to enroll in Credential Manager.
When administering RSA Credential Manager, if you approve a request with two approval steps and then click the Submit and Continue button, an error message appears stating that the request is completed.	If a custom extension is added to a request with two approval steps to automatically approve the second approval step, there is a time delay that causes the error message to appear.	Wait and then refresh the RSA Security Console.
Microsoft Management Console (MMC)		
The Microsoft Management Console (MMC) snap-in does not start.	The Microsoft Management Console may fail to start for a variety of reasons. For example: <ul style="list-style-type: none"> • IP address or name resolution issue • Network connectivity issue 	RSA Authentication Manager Microsoft Management Console Snap-in Does Not Start on page 419.
Unable to open the Microsoft Management Console (MMC) snap-in after unlinking Active Directory from a realm, and then linking it to a new realm.	Unlinking Active Directory from the realm revokes the MMC administrator permissions.	After linking the identity source to new realm, assign the Super Admin role to the Active Directory administrator.

General Troubleshooting Tips

This section provides general information for troubleshooting RSA Authentication Manager and configuring and managing the system to prevent future issues. The following issues are included:

- [Using the Activity Monitor and Log Messages for Troubleshooting](#)
- [Making Sure the RSA Authentication Manager Machine Meets Minimum System Requirements](#)
- [Supported Browsers](#)
- [Configuring Browser Settings for the RSA Security Console, RSA Operations Console, and RSA Self-Service Console](#)
- [Assessing the Impact of Firewalls on RSA Authentication Manager](#)
- [Configuring the Cache for Improved Performance](#)
- [Test User Access to Restricted Agent](#)

Using the Activity Monitor and Log Messages for Troubleshooting

Viewing Log Messages in Real Time

Activity Monitors allow you to view log messages in real time. Authentication Manager has three different Activity Monitors: Authentication Activity, System Activity, and Administrator Activity. These Activity Monitors allow you to view messages written to the audit and system logs.

You can configure the types of messages displayed on the Activity Monitor. For example, you can choose to display fatal error messages only, or you can choose to only view activity for a certain administrator. By sorting and filtering data, you can use the Activity Monitor to monitor the system and troubleshoot existing issues.

For detailed information on the Activity Monitor, see [“Using the Activity Monitor”](#) on page 206.

For instructions, see the Security Console Help topic “Manage the Activity Monitor.”

Viewing Archived Log Messages

As the Activity Monitor populates with new log messages, the older log messages are written to the internal database. You can use these messages for troubleshooting problems with Authentication Manager.

To view messages written to the internal database (this includes data from the Administrative Audit, Runtime Audit, and System Logs), run a report. Use the Reporting menu in the RSA Security Console to run reports such as:

- Administrator Activity
- Authentication Activity
- System Log Report

Reports can be customized to meet your exact troubleshooting needs. For more information on creating and running reports, see [“Generating Reports”](#) on page 198.

You can also use the trace file for troubleshooting. You can increase the logging level of the trace file to “information” to capture more descriptive messages so that you can further monitor the system. Do not increase trace logging levels for extended periods of time, as trace logs can take up large amounts of disk space.

Important: If you increase the trace logging levels for troubleshooting purposes, do not set them to “verbose” unless instructed to do so by Customer Support. This logging level takes up large amounts of disk space and can impact system performance.

Data from the Trace Log is written to a file in:
RSA_AM_HOME/server/logs.

For more information on logging, see [“Configuring RSA Authentication Manager Logging”](#) on page 195.

Viewing Log Files

The log messages that display in the Security Console are also written to files in your Authentication Manager directory. In addition, your Authentication Manager directory contains other logs for your servers, including your WebLogic and proxy servers. The information in these logs can be very valuable for troubleshooting Authentication Manager.

The following table lists the log files available in each location.

Log File Name	Description of Contents
<i>RSA_AM_HOME/server/logs</i>	
imsAdminAudit.log	Contains the information written to the Administrative Audit log. Captures log messages that record any administrative action, such as adding and editing users.
	Note: Only contains data not written to the Administrative Audit log stored in the internal database.
imsRuntimeAudit.log	Contains the information written to the Runtime Audit log. Captures log messages that record any runtime activity, such as authentication and authorization of users.
	Note: Only contains data not written to the Runtime Audit log stored in the internal database.



Log File Name	Description of Contents
imsSystem.log	Contains the information written to the System log. Captures log messages that record system level messages, such as “Authentication Manager Server started,” and “Connection Manager lost db connection.” <hr/> Note: Only contains data not written to the System log stored in the internal database. <hr/>
imsTrace.log	Contains the information written to the Trace log. Captures log messages that you can use to debug your system.
AdminServer.log	Contains information on the main WebLogic admin server.
AdminServer_access.log	Contains information on the main WebLogic admin server HTTP interface.
<server name>_server.log	Contains information on the primary Authentication Manager server, where <i>server name</i> is the name of the machine. Includes information such as server start up, environment, which server components are running, networking services, and server errors.
<server name>_server_access.log	Contains information on the primary Authentication Manager HTTP interface. <hr/> Note: Unless the server is accessed directly, all entries will appear to come from the proxy server. <hr/>
proxy_server.log	Contains information on the proxy server. Includes information such as server start up, environment, which server components are running, networking services, and server errors.
proxy_server_access.log	Contains information on the proxy server HTTP interface.
rsa-console.log	Contains information on the Security Console.
RSA_AM_HOME/server/servers/<server name>/logs	
<server name>_winservice.log	Contains Windows Services information for the primary Authentication Manager server.

Log File Name	Description of Contents
<i>RSA_AM_HOME/server/servers/AdminServer/logs</i>	
admin_winservice.log	Contains Windows Services information for the primary WebLogic admin server.
<i>RSA_AM_HOME/server/servers/proxy_server/logs</i>	
proxy_winservice.log	Contains Windows Services information for the proxy server.
<i>RSA_AM_HOME/imsoc/logs</i>	
imsAdminAudit.log	Contains the information written to the Administrative Audit log in the Operations Console. Captures log messages that record any administrative action, such as adding and editing users.
imsRuntimeAudit.log	Contains the information written to the Runtime Audit log in the Operations Console. Captures log messages that record any runtime activity, such as authentication and authorization of users.
imsSystem.log	Contains the information written to the System log in the Operations Console. Captures log messages that record system level messages, such as "Authentication Manager Server started," and "Connection Manager lost db connection."
imsTrace.log	Contains the information written to the Trace log in the Operations Console. Captures log messages that you can use to debug your system.
ops-console.log	Contains information on the Operations Console.
<i>RSA_AM_HOME/imsoc/servers/AdminServer/logs</i>	
AdminServer.log	Contains information on the main WebLogic admin server for the Operations Console.
Access.log	Contains information on the main WebLogic admin server for the Operations Console.

Log File Name	Description of Contents
<i>RSA_AM_HOME/radiusoc/logs</i> ¹	
imsAdminAudit.log	Contains the information written to the Administrative Audit log in the Operations Console. Captures log messages that record any administrative action, such as adding and editing users.
imsRuntimeAudit.log	Contains the information written to the Runtime Audit log in the Operations Console. Captures log messages that record any runtime activity, such as authentication and authorization of users.
imsSystem.log	Contains the information written to the System log in the Operations Console. Captures log messages that record system level messages, such as "Authentication Manager Server started," and "Connection Manager lost db connection."
imsTrace.log	Contains the information written to the Trace log in the Operations Console. Captures log messages that you can use to debug your system.
ops-console.log	Contains information on the Operations Console, including command failures, login failures, and session time outs.
<i>RSA_AM_HOME/db/admin/<database instance>/bdump</i>	
alert_ <i><database instance></i> .log	Contains an ongoing log of low-level failures such as, "out of disk space" or "disk hardware error." <i><database instance></i> is the "com.rsa.db.instance" property in <i>RSA_AM_HOME/etc/jndi.properties</i> .

¹RADIUS log files are generated for standalone RADIUS servers only.

Enabling Trace Logging for Troubleshooting

You can enable a more detailed logging level for troubleshooting purposes. Enabling detailed trace logging allows you to view log messages at a component level, providing more detailed information than the Security Console.

Important: Do not enable trace logging without the assistance of RSA Customer Support.

Use the Set Trace utility to enable trace logging. For more information, see "[Set Trace Utility](#)" on page 298.

Changing the Logging Level of a Replica Instance

Use the Store utility, `store`, to set the type of log and the logging level for a replica instance. If the primary instance is unavailable, you cannot set the logging level for a replica instance from the Operations Console on the replica instance. Use the Store utility to work around this limitation when you need to perform troubleshooting tasks on a replica instance.

To change the logging level:

1. On the replica instance that you want to change, open a new command shell, and change directories to `RSA_AM_HOME/utills`.

2. Type:

```
rsautil store options
```

For relevant options, see the following section, "[Options for store](#)."

Important: Although it is possible to enter the Super Admin's password on the command line along with the other options, this creates a potential security vulnerability. RSA recommends that you enter the Super Admin's password only when the utility presents a prompt.

Use the indicated options to perform the following tasks:

- To list all of the currently configured instances to obtain the name of a replica instance, type:

```
rsautil manage-replication --action list
```

- To set the type of log and the logging level on the replica instance, type:

```
rsautil store --action config name value instance
```

where:

- `name` is the type of log to set.
- `value` is the logging level to set.
- `instance` is the name of the replica instance on which you want to set the log type and logging level.

After you set the type of log and logging level on the replica instance, you must restart the application server to make the change effective.

For example, to set the logging level of the trace log to verbose on the replica instance named `a_instance`, type:

```
rsautil store --action config ims.logging.level.trace  
4 a_instance
```

Options for store

The following table describes the options for this utility.

Flag	Alternate Flag	Description
-a	--action	<p>Specifies the action to perform. Select:</p> <p>config name value instance. Specifies the log type and logging level to set, and the replica instance on which this is set.</p> <p><i>name</i> (choose one):</p> <p>ims.logging.level.audit.admin. Administrative audit log.</p> <p>ims.logging.level.audit.runtime. Runtime audit log.</p> <p>ims.logging.level.system. System log.</p> <p>ims.logging.level.trace. Trace log.</p> <p><i>value</i> (choose one):</p> <ul style="list-style-type: none"> 0. None 1. Error 2. Warning 3. Success 4. Verbose <p><i>instance</i> (specify the name of the replica instance)</p>
-h	--help	Displays help for this utility.
-m	--master-password	Master password of the encrypted properties file.
-V	--verbose	Optional. Displays more output messages.
-v	--version	Displays the version and copyright information.

Making Sure the RSA Authentication Manager Machine Meets Minimum System Requirements

Make sure that your system meets these minimum requirements for supported platform and system components. The requirements listed in this section serve only as guidelines. Hardware requirements vary depending on a number of factors, including authentication rates, number of users, frequency of reporting, and log retention. For more information, see the *Performance and Scalability Guide*.

The values listed for RSA RADIUS disk space and memory are in addition to those for Authentication Manager when RADIUS is installed on the same machine with Authentication Manager. When RADIUS is installed on a standalone machine, the values listed for Authentication Manager are sufficient.

Note: You must install all of your Authentication Manager and RADIUS software on the same system types. For example, do not configure Authentication Manager on Solaris and then configure RADIUS on Windows.

For CPU speed, memory, and disk speed, RSA recommends that the database server be significantly more powerful than every other machine in your Authentication Manager deployment.

RSA recommends that you deploy Authentication Manager on machines to which only authorized users have access. For example, avoid deploying Authentication Manager on machines that host other applications to which non-administrative users have access.

Important: Be sure that your UNIX and Windows servers are known by their fully qualified domain names. If you are installing the Authentication Manager database on a separate machine, configure your DNS server to resolve by both fully qualified name and short name.



Windows System Requirements

Operating System	Microsoft Windows Server 2003 Enterprise R2 SP2 (32-bit) Microsoft Windows Server 2003 Enterprise SP2 (32-bit) Microsoft Windows Server 2003 Enterprise R2 SP2 (64-bit) Microsoft Windows Server 2003 Enterprise SP2 (64-bit) Note: RADIUS is not supported on 64-bit Windows and Linux systems.
Hardware	Intel Xeon 2.8 GHz or equivalent (32-bit) Intel Xeon 2.8 GHz or equivalent (64-bit)
Disk Space	RSA Authentication Manager: 60 GB free space recommended Important: Do not allow all disk space to become consumed. At that point, Authentication Manager may stop operating and be difficult to restore. RSA RADIUS: Add 125 MB of free space
Memory Requirements	RSA Authentication Manager: 2 GB RSA RADIUS: Add 512 MB
Page File	2 GB

Linux System Requirements

Operating System	Red Hat Enterprise Linux 4.0-1 ES (32-bit) Red Hat Enterprise Linux 4.0-1 ES (64-bit) Red Hat Enterprise Linux 4.0-1 AS (32-bit) Red Hat Enterprise Linux 4.0-1 AS (64-bit) Note: RADIUS is not supported on 64-bit Windows and Linux systems.
Hardware	Intel Xeon 2.8 GHz or equivalent (32-bit) Intel EM64T 2.8 GHz or AMD Operon 1.8 GHz, or equivalent (64-bit)
Disk Space	RSA Authentication Manager: 60 GB free space recommended Important: Do not allow all disk space to become consumed. At that point, Authentication Manager may stop operating and be difficult to restore. RSA RADIUS: Add 470 MB of free space

Memory Requirements	RSA Authentication Manager: 2 GB RSA RADIUS: Add 512 MB
Swap Space	2 GB
Kernel Version	2.6.9-22.EL and later
Kernel Parameters	Maximum shared memory must be at least 256 MB
Packages (RPM) 32-bit	<p>The following packages must be installed:</p> <p>binutils-2.15.92.0.2-12 bog1-0.1.18-4 compat-db-4.1.25-9 compat-libstdc++-296.2.9.6-132.7.2 compat-openldap coreutils 5.2.1-31.2 or later control-center-2.8.0-12 cyrus-sasl-gssapi-2.1.19-5 cyrus-sasl-ntlm-2.1.19-5 cyrus-sasl-sql-2.1.19-5 fribidi-0.10.4-6 gcc-3.4.3-22.1 gcc-c++-3.4.3-22.1 gnome-libs-1.4.1.2.90-44.1 glibc-common-2.3.4-2.19 glibc-2.3.2-95.20 gsl-1.5-2 gtkspell-2.0.7-2 kdelibs initscripts 7.93.20 or later libstdc++-3.4.3-22.1 libaio-0.3.105-2.i386 libavc1394-0.4.1-4 libdbi-0.6.5-10 make-3.80-5 libstdc++-devel-3.4.3-22.1 pdksh-5.2.14-30 setarch-1.6-1 sysstat-5.0.5-1 xscreensaver-4.18-5</p> <p>Note: To check your RPM versions on Linux, use the command, <code>rpm -q package name</code>.</p>



Packages (RPM) 64-bit	Install the following packages: Compatibility Arch Development Support Compatibility Arch Support Note: Make sure that all components in each package are selected.
-----------------------	---

Solaris System Requirements

Operating System	Solaris 10 (64-bit)
Hardware	UltraSPARC 1.5 GHz, or equivalent For improved performance, use Sun 6 or 8 core UltraSPARC T1 servers. Note: On Sun UltraSPARC systems, Authentication Manager start-up and migration processes can take considerable time. For example, restarting Authentication Manager can take 15 minutes or more. Migration of a large database can take 12 hours or more. In general, Sun UltraSPARC systems with faster processors will yield better start-up and migration performance.
Disk Space	RSA Authentication Manager: 60 GB free space recommended 20 GB free space minimum RSA RADIUS: Add 650 MB of free space
Memory Requirements	RSA Authentication Manager: 4 GB RSA RADIUS: Add 512 MB
Swap Space	4 GB
Packages	SUNWarc SUNWbtool SUNWhea SUNWlibm SUNWlibms SUNWsprt SUNWtoo SUNWi1of SUNWi1cs SUNWi15cs SUNWxwft

Supported Browsers

This section describes the browsers supported for the RSA Security Console. Browser support differs between Windows and Linux platforms.

On Windows

- Internet Explorer 6.0 with SP2
- Internet Explorer 7.0
- Firefox 1.0.7 and above

On Linux

- Firefox 1.0.7 and above

On Solaris

- Firefox 1.0.7 and above

Note: On all browsers, JavaScript must be enabled. Internet Explorer may require configuration depending on your security level. See [“Logging On to the RSA Security Console”](#) on page 15.

Configuring Browser Settings for the RSA Security Console, RSA Operations Console, and RSA Self-Service Console

You must have the appropriate browser settings in order for the Security Console and Operations Console to operate properly. You must enable JavaScript and configure the correct security settings in the Security Console and Operations Console.

Your users must also configure their browsers if they will be using RSA Credential Manager and the Self-Service Console.

Enabling JavaScript

Make sure that JavaScript is enabled in your browser. If JavaScript is not enabled, you may see alert messages.

To enable JavaScript for Internet Explorer:

1. In the browser, select **Tools > Internet Options > Security**.
2. Select the appropriate Web content zone.
3. If you use the default security level, JavaScript is enabled. If you use a custom security setting, click **Custom Level**, and do the following:
 - Scroll down to **Miscellaneous > Use Pop-up Blocker**, and select **Disable**.
 - Scroll down to **Scripting > Active Scripting**, and select **Enable**.
 - Scroll down to **Scripting > Allow paste operations via script**, and select **Enable**.
 - Scroll down to **Scripting > Scripting of Java Applets**, and select **Enable**.

To enable JavaScript for Mozilla FireFox:

You do not need to enable JavaScript for Firefox, however if JavaScript is disabled, perform these steps:

1. Open the Firefox browser.
2. Click **Tools > Options > Content**.
3. Select **Enable JavaScript**.
4. Click **OK**.
5. In the browser, select **Edit > Preferences > Advanced > Enable JavaScript**.

Adding the RSA Security Console, RSA Operations Console, and the RSA Self-Service Console to Trusted Sites

If Internet Explorer is configured for enhanced security levels, you must add the Security Console and Operations Console URLs to the list of trusted sites.

You must also instruct your users who use Credential Manager to add the URL to their trusted sites.

Important: Do not lower your browser security level in lieu of adding the RSA Console URLs to your trusted sites. This compromises security.

To add an RSA Console to the list of trusted sites:

1. In Internet Explorer, select **Tools > Internet Options > Security**.
2. Select the Trusted Sites icon, and click **Sites**.
3. Enter the URL for the appropriate RSA Console.
4. Clear **Require server verification (https:) for all sites**.
5. Click **Add**.

Assessing the Impact of Firewalls on RSA Authentication Manager

Your system must be configured so that Authentication Manager can communicate freely, yet securely, with the agent, authentication, and identity source servers. To do this, there are certain ports that must remain open. If the appropriate ports are not open, you may encounter LDAP, replication, and authentication failures.

For example, if the appropriate ports are not open, users experience an authentication failure. In this situation, the system times out, the users experience a long delay, and then the users see a message telling them that access is denied. When this happens, you see error messages in the system log.

To ensure that Authentication Manager communicates securely with the other servers and system components, it is important that you make sure the appropriate ports are open between the following:

- Agents and authentication servers.
- Authentication servers and LDAP servers.
- Authentication servers and other authentication servers.

The following port numbers must be available to enable authentication, administration, replication, and other services on the network. RSA recommends that you reserve these ports for Authentication Manager, and make sure that no other applications or services are configured to use them.

Port Number	Protocol	Service	Description
1161	UDP	SNMP agent	Used to communicate with a Network Management Server using the Simple Network Management Protocol.
1162	UDP	SNMP agent	Used to communicate with a Network Management Server using the Simple Network Management Protocol.
1645	UDP	RADIUS authentication (legacy port)	Used for authentication requests from RADIUS clients.
1812	UDP	RADIUS authentication (standard port)	Used for RADIUS authentication and accounting.
1646	UDP	RADIUS accounting (legacy port)	Used for requests for accounting data.
1813	TCP	RADIUS (standard port)	Used for RADIUS administration and replication.
2334	TCP	RSA Authentication Manager database listener	Used to replicate data between instances.
5500	UDP	Agent authentication	Used for communication with authentication agents. This service receives authentication requests from agents and sends replies.
5550	TCP	Agent auto-registration	Used for communication with authentication agents that are attempting to register with Authentication Manager.
5556	TCP	RSA Authentication Manager node manager	Used to monitor and manage various services.



Port Number	Protocol	Service	Description
5580	TCP	Offline authentication service	Used to receive requests for additional offline authentication data, and send the offline data to agents. Also used to update server lists on agents.
7002	TCP	RSA Authentication Manager	Used for SSL-encrypted administration connections.
		RSA Authentication Manager Microsoft Management Console snap-in	Used for SSL-encrypted connections.
7004	TCP	RSA Authentication Manager proxy server	Used for load balancing of administration in an instance with multiple server nodes. This port is used for SSL connections.
		RSA Self-Service Console proxy server/SSL	Used for communication from users to Authentication Manager for requests and maintenance tasks. This port is used for SSL connections.
		RSA Authentication Manager Microsoft Management Console snap-in proxy server	Used for load balancing of administration in an instance with multiple server nodes. This port is used for SSL connections.
7006	TCP	RSA Authentication Manager cluster administration channel	Internal use only.
7008	TCP	RSA Authentication Manager cluster administration server	Internal use only.
7012	TCP	RSA Authentication Manager administration channel	Internal use only.
7014	TCP	RSA Authentication Manager proxy server administration channel	Internal use only.

Port Number	Protocol	Service	Description
7022	TCP	Network access point	Used for mutually authenticated SSL-encrypted trusted realm connections.
7071	TCP	RSA Operations Console	Used for non-SSL connection.
7072	TCP	RSA Operations Console	Used for SSL connections.

Configuring the Cache for Improved Performance

You can configure the system cache for improved system performance. For example, for each cached object, you can designate the cache size limits, view the size of the current cache, and designate the amount of time between cache refreshes.

For detailed information on how you can configure the cache for improved system performance, see [“Configuring the System Cache for Improved Performance”](#) on page 118.

Test User Access to Restricted Agent

If your users experience trouble accessing restricted agents, you can use the Security Console to test the user configuration.

To test user access to a restricted agent:

1. In the Security Console, click **Access > Test Access**.
2. Enter the name of the **Restricted Authentication Agent**.
3. Enter the **User ID**.
4. Click **Test**.

The test result displays whether the user is configured on the restricted agent.

User and Token-Related Resolutions

As an administrator, you receive telephone calls from users who need assistance. For example, users may call because they cannot authenticate, or because they have lost or damaged their token. This section provides information on the following user-related situations that you may encounter as an administrator:

- [Unlocking a User](#)
- [Assisting Users with Lost, Stolen, Damaged or Expired Tokens](#)
- [Providing Emergency Access](#)
- [Clearing PINs](#)
- [Forcing PIN Changes](#)
- [Clearing Incorrect Passcodes](#)
- [Resynchronizing a Token](#)

Unlocking a User

Users are locked out of Authentication Manager for the following reasons:

- The user violated the lockout policy specified by the security domain to which he or she belongs.
For this situation, see [Assisting Users Who Have Been Locked Out of the System](#) on page 94.
- The user violated the lockout policy as specified by the external identity source to which he or she belongs. Some identity sources, Microsoft Active Directory and Sun Java System Directory Server, for example, have their own lockout policies. If a user violates the identity source lockout policy, the user profile in the Security Console does not indicate that the user is locked out, but the user is unable to authenticate. Check your identity source to see if the user has violated the identity source lockout policy. If so, unlock the user.

Assisting Users with Lost, Stolen, Damaged or Expired Tokens

You may occasionally encounter users who are unable to use their tokens because the tokens are either damaged, lost, temporarily misplaced, stolen, or expired. In these situations, replace the token (if applicable) and provide temporary emergency access if necessary.

Important: Encourage your users to report lost or stolen tokens as soon as possible.

See [“Assisting Users Whose Tokens Are Lost, Stolen, Damaged, or Expired”](#) on page 95.

Providing Emergency Access

Users may occasionally require temporary emergency access to Authentication Manager if their token is temporarily unavailable, or if they are waiting for a replacement for their lost, stolen, damaged, or expired token.

You can provide temporary emergency access to Authentication Manager for the following scenarios:

Online authentication. Provides emergency access for users with misplaced, lost, stolen, or damaged tokens. Temporary emergency access is available using an Online Emergency Access Tokencode.

Offline authentication. Provides emergency access for RSA SecurID for Windows users who require emergency access while authenticating offline. Temporary emergency access is available using an Offline Emergency Access Tokencode or an Offline Emergency Passcode.

See [“Providing Users with Temporary Emergency Access”](#) on page 96.

Clearing PINs

You need to clear a user's PIN if the user has forgotten it. When you clear a PIN, the current PIN is deleted so that the user can create a new one.

See [“Clearing PINs”](#) on page 104.

Forcing PIN Changes

You can force users to change their PINs if there is concern that the PIN has been compromised. A compromised PIN puts the resources protected by Authentication Manager at risk.

Important: Instruct users to report compromised PINs as soon as possible, as they pose a significant security risk.

See [“Requiring Users to Change Their PINs”](#) on page 105.

Clearing Incorrect Passcodes

The system counts each time the assigned user enters an incorrect passcode, clearing this count automatically with each correct passcode. If a user enters more incorrect passcodes than allowed by the token policy, and then enters a correct passcode, the user is prompted for his or her next tokencode.

If you do not want a user to be prompted for the next tokencode, you can clear the incorrect passcodes so the user does not violate the token policy.

Note: If the user has violated both the lockout policy and the token policy, you must unlock the user account after clearing the incorrect passcodes. For more information, see [“Unlocking a User”](#) on page 416.

See [“Clearing Incorrect Passcodes”](#) on page 106.

Resynchronizing a Token

A token needs to be resynchronized when the following occurs:

- For time-based tokens, resynchronization is necessary when the token clock and the Authentication Manager system clock do not match. When the clocks do not match, the tokencodes are not the same. If the tokencodes are not the same, authentication attempts fail.
- For event-based tokens, resynchronization is necessary when the token's tokencode count and the Authentication Manager tokencode count are not the same. When the tokencode counts are different, authentication attempts fail.

See [“Resynchronizing Tokens”](#) on page 103.

System-Related Resolutions

In Authentication Manager, you may experience system-related issues. System-related issues could involve one or more of your servers, your identity source, or your system configuration. This section provides information on the following system-related situations that you may encounter as an administrator:

- [RSA Authentication Manager Does Not Start](#)
- [RSA Security Console Does Not Start](#)
- [RSA Authentication Manager Microsoft Management Console Snap-in Does Not Start](#)
- [Name and IP Address Resolution in RSA Authentication Manager](#)
- [Managing the Node Secret](#)
- [Resynchronizing RSA Authentication Manager with Coordinated Universal Time](#)
- [Updating an Agent Configuration File](#)
- [Reconfiguring CT-KIP After Promoting a Replica Instance](#)
- [Changing the IP Address or Hostname of a Server](#)

RSA Authentication Manager Does Not Start

Note: The Authentication Manager server may fail to start if the minimum system requirements are not met. When troubleshooting servers, check the system requirements first. See [“Making Sure the RSA Authentication Manager Machine Meets Minimum System Requirements”](#) on page 407.

If one or all of your Authentication Manager servers fails to start, do the following:

View the system logs to check for error messages and strange activity. Use the following logs:

Trace. Captures log messages that you can use to debug your system.

System. Captures log messages that record system level messages.

RSA Security Console Does Not Start

Note: The Security Console may take a considerable time to start on its initial startup. This may extend to ten minutes in some cases.

If the RSA Security Console does not start, check the following:

- Make sure the URL is correct.
- Make sure the network connections are working.
- Make sure the server is reachable from the administrator workstation (name resolution).

If the problem continues, call RSA Customer Support.

RSA Authentication Manager Microsoft Management Console Snap-in Does Not Start

The Authentication Manager Microsoft Management Console snap-in may fail to start for various reasons, including:

- IP address or name resolutions issues
- Network connectivity issues

If the Authentication Manager Microsoft Management Console snap-in fails to start, try to locate the source of the problem. Do the following:

1. Try to start the Security Console in a web browser to see if you can log on and perform a standard operation, such as listing a user assigned token.
2. Try to use the Windows user account to log on to the Security Console. If this fails, the appropriate administrative role is not assigned to the Windows user.
3. Check whether the current Windows user account used to launch the MMC is Domain and Local administrator. If not, assign the appropriate privilege to the Windows user, and restart the MMC.
4. Open the Windows registry to see whether you have read or modify permission to the registry key:
HKEY_LOCAL_MACHINE\SOFTWARE\RSASecurity\
AuthenticationManager\MMC
5. (Remote access users only) Check whether the client machine is part of the Active Directory domain.

RSA Security Console Times Out When Searching for Users

If the server exception error PRINCIPAL_SEARCH_TIME_EXCEEDED appears, you must re-index your Sun Java System Directory Server, and increase the cache size to 100 MB.

For more information and instructions, see [“Re-Indexing your Directory for Improved Searches”](#) on page 24.

Name and IP Address Resolution in RSA Authentication Manager

Authentication Manager is dependent on server name and IP address resolution. The system must be able to use the server Fully Qualified Host Name (FQHN) to obtain the corresponding IP address. The system must also be able to take the IP address and find the corresponding FQHN.

To ensure proper communication between all of the Authentication Manager components, and to avoid authentication failures, do the following:

- Make sure that the authentication servers, agent hosts, and directory servers are all registered in the Domain Name System (DNS).
- Make sure that your system is configured to do forward and reverse name and IP address lookups.
- Make sure that all agents are entered correctly, for example, the name and IP address match what is stored in DNS. See the following section, [“Resolving Agent Hostnames and IP Addresses.”](#)

Resolving Agent Hostnames and IP Addresses

When adding authentication agents to Authentication Manager, it is very important that you make sure the agent hostname and corresponding IP address are entered correctly. If the hostname and IP address are entered incorrectly, authentication attempts fail.

To use the Security Console to check the hostnames and IP addresses of your Authentication Agents:

1. In the Security Console, click **Access > Authentication Agents > Manage Existing**.
2. Select the appropriate agent, and click **Edit** in the agent Context menu.
3. Verify the **Hostname** and **IP Address** of the Authentication Agent, and edit if necessary.
4. Click **Save**.

See the Security Console Help topic “Edit Authentication Agents.”

Resolving Agent IP Addresses When Using Network Address Translation (NAT)

If your system uses Network Address Translation (NAT), you need to configure alias IP addresses for each of your authentication agents. It is very important that the alias IP addresses are entered correctly, as incorrect addresses can result in authentication failures.

To use the Security Console to check the alias IP addresses of your authentication agents:

1. In the Security Console, click **Access > Authentication Agents > Manage Existing**.
2. Select the appropriate agent, and click **Edit** in the agent Context menu.
3. Verify the **Alternate IP Address** of the authentication agent, and edit if necessary.
4. Click **Save**.

Managing the Node Secret

The node secret is a shared secret known only to the authentication agent and the Authentication Manager. Authentication agents use the node secret to encrypt authentication requests that they send to Authentication Manager. Authentication Manager automatically creates and sends the node secret to the agent in response to the first successful authentication on the agent.

The node secret rarely needs to be refreshed, however there are times when it is necessary (for example, you changed the server IP address). Problems with the node secret can result in authentication or node verification errors.

To refresh the node secret, see [“Securing Communications Between the Authentication Agent and RSA Authentication Manager”](#) on page 112.

Resynchronizing RSA Authentication Manager with Coordinated Universal Time

Authentication Manager relies on standard time settings known as Coordinated Universal Time (UTC). The time, date, and time zone settings on computers running Authentication Manager must always be correct in relation to UTC.

When Authentication Manager is out-of-sync with UTC, all users experience the same behavior (for example, authentication failures). If the symptoms are not present to all users, see [“Common Problems and Resolutions”](#) on page 391 to look for other potential causes.

Make sure the time on the computer on which you are installing Authentication Manager is set to the local time and corresponds to the UTC. For example, if UTC is 11:43 a.m. and the Authentication Manager is installed on a computer in the Eastern Standard Time Zone in the United States, make sure the computer clock is set to 6:43 a.m. This differs during daylight saving time.

To obtain the correct UTC, go to www.time.gov.

Important: Contact RSA Customer Support before making any changes to the time or date settings.

Updating an Agent Configuration File

Occasionally, it may be necessary to generate a new Authentication Manager agent configuration file, **sdconf.rec**. Corrupt agent configuration files can cause communication failures between the agent and the server, and users cannot authenticate.

To use the Security Console to generate the agent configuration file:

1. In the Security Console, click **Access > Authentication Agents > Generate Configuration File**.
2. Select **Maximum Retries**.
3. Select **Maximum Time Between Each Retry**.
4. Click **Generate Configuration File**.
The Download Configuration File page opens.
5. Click **Download Now**.
6. When prompted, click **Save to Disk**, and save the ZIP file to your machine.
7. Unzip the file, and replace the existing **sdconf.rec** file installed on the agent.

Reconfiguring CT-KIP After Promoting a Replica Instance

When you configured Authentication Manager, you likely configured the following URLs necessary for Remote Token-Key Generation (CT-KIP):

Token-Key Generation. This is the URL to invoke the CT-KIP application that interacts with the Authentication Manager CT-KIP server for remote token-key generation.

Service Address. This is the server-side address of the CT-KIP service.

After you promote a replica instance to a primary instance, use the RSA Security Console to modify these URLs to point to the new primary instance. For instructions, see the Security Console Help topic “Configuring RSA Authentication Manager.”

Changing the IP Address or Hostname of a Server

There may be a time when you need to change the IP address or hostname of your authentication server after it has been installed and configured.

For example, assume that you are using a server in one of your labs. That server has an IP address for the lab network. If you decide to move that server so it is part of the corporate network, you must change the server IP address to correspond with the corporate network IP scheme.

For information on changing the IP address or hostname of an Authentication Manager or RSA RADIUS server, see Appendix E, [“Updating Server IP Addresses and Names.”](#)

Glossary

Term	Definition
Active Directory	The directory service that is included with Microsoft Windows Server 2003 and Microsoft Windows 2000 Server.
Active Directory forest	A federation of identity servers for Windows Server environments. All identity servers share a common schema, configuration, and Global Catalog.
AD	See Active Directory.
adjudicator	A component that defends Authentication Manager against replay attacks in which an intruder attempts to reuse an old passcode or acquires the current passcode for a token and sets the system clock back to use the captured passcode.
administrative command	A command other than a system-generated command.
administrative role	A collection of permissions and the scope within which those permissions apply.
administrator	Any user with one or more administrative roles that grants administrative permission to manage administrative resources.
Advanced Encryption Standard (AES)	The current cryptographic standard, adopted by the National Institute of Standards and Technology (NIST) in November, 2001. AES replaces Data Encryption Standard (DES) because it is considered to be more secure.
AES	See Advanced Encryption Standard.
agent	A software application installed on a device, such as a domain server, web server, or desktop computer, that enables authentication communication with Authentication Manager on the network server.
agent auto-registration utility	A utility included in the RSA Authentication Agent software that enables you to automatically register new authentication agents in the internal database, and updates the IP addresses for existing agents.
agent host	The machine on which an agent is installed.

Term	Definition
Agent Protocol Server	The Authentication Manager component that manages the ACE protocol packet traffic to and from agents. The inbound request packets are routed to the appropriate message handler. The response packets are sent to the originating agent.
approver	A Request Approver or an administrator with approver permissions.
attribute	A characteristic that defines the state, appearance, value, or setting of something. In Authentication Manager, attributes are values associated with users and user groups. For example, each user group has three standard attributes called Name, Identity Source, and Security Domain.
attribute mapping	The process of relating a user or user group attribute, such as User ID or Last Name, to one or more identity sources linked to a given realm. No attribute mapping is required in a deployment where the internal database is the primary identity source.
audit information	Data found in the audit log representing a history of system events or activity including changes to policy or configuration, authentications, authorizations, and so on.
audit log	A system-generated file that is a record of system events or activity. The system includes four such files, called the Trace, Administrative, Runtime Audit, and System logs.
authentication	The process of reliably determining the identity of a user or process.
authentication authority	The central entry point for authentication services.
authentication broker	A component that handles the authentication process and issuance of authentication tickets.
authentication method	The type of procedure required for obtaining authentication, such as a one-step procedure, a multiple-option procedure (user name and password), or a chained procedure.
authentication policy	A collection of rules that specify the authentication requirements. An authentication policy may be associated with one or more resources.
authentication protocol	The convention used to transfer credentials of a user during authentication. For example, HTTP-BASIC/DIGEST, NTLM, Kerberos, and SPNEGO.

Term	Definition
Authentication Server	An Authentication Manager component made up of services that handle authentication requests, database operations, and connections to the RSA Security Console.
authenticator	A device used to verify a user's identity to Authentication Manager. This can be a hardware token (for example, a key fob) or a software token.
authorization	The process of determining if a user is allowed to perform an operation on a resource.
authorization data	Information defined by the provisioning server, which is necessary to complete the provisioning of a CT-KIP-enabled token. Authorization data includes the appropriate serial number and places the new token credentials in the Authentication Manager internal database.
auto-registration	A setting which, if enabled, permits unregistered users to become registered upon a successful authentication to a system-managed resource. If auto-registration is disabled, only an administrative action can register users. Also see registered user and unregistered user.
Base Server license	Authentication Manager license that allows one primary instance and one replica instance. (Multiple replica instances and server nodes are not allowed.) Includes RSA Credential Manager self-service. Credential Manager provisioning can be added.
Business Continuity option	Authentication Manager option that allows you to temporarily increase the number of users allowed into your system and the number of users allowed to use on-demand authentication.
certificate	An asymmetric public key that corresponds with a private key. It is either self-signed or signed with the private key of another certificate.
certificate DN	The distinguished name of the certificate issued to the user for authentication.
chained authentication	The process of creating a strong form of authentication by combining two weaker forms. For example, the user is required to use a PIN and a tokencode.
client time-out	The amount of time (in seconds) that the user's desktop can be inactive before reauthentication is required.
CLU	See command line utility.

Term	Definition
cluster	An instance consisting of a database server and one or more server nodes.
command line utility (CLU)	A utility that provides a command line user interface.
connection pool	A named group of identical connections to a data store.
contact list	A list of server nodes provided by the Authentication Manager to the agent, to which the agent can direct authentication requests.
context-based authentication	An authentication sequence in which the system presents the user with only the authentication options that are appropriate for the User ID entered. The options are based on policy requirements and the authenticators that the user owns.
core attributes	The fixed set of attributes commonly used by all RSA products to create a user. These attributes are always part of the primary user record, whether the deployment is in an LDAP or RDBMS environment. You cannot exclude core attributes from a view, but they are available for delegation.
Credential Manager Provisioning	An option that automates the token deployment process and provides user self-service options.
cryptographic algorithm	A mathematical function that uses plain text as the input and produces cipher text as the output and vice-versa. It is used for encryption and decryption.
CT-KIP	Cryptographic Token-Key Initialization Protocol.
CT-KIP-capable token	A token that is capable of storing the authorization data and seed generated as a result of CT-KIP operations between a CT-KIP 1.0 client and an Authentication Manager CT-KIP server.
CT-KIP client	A program that implements the CT-KIP client-side protocol and interacts with a CT-KIP server for the secure initialization of CT-KIP-capable tokens.
CT-KIP server	A software component of Authentication Manager that implements the CT-KIP server-side protocol and interacts with a CT-KIP client application for the secure initialization of CT-KIP-capable tokens.
CT-KIP toolkit	An implementation of the CT-KIP client-server protocol. It provides the API for creating CT-KIP server or client applications.

Term	Definition
customer name	The name of the enterprise to which the license is issued.
data encryption standard (DES)	The cryptographic standard prior to November 2001, when the National Institute of Standards and Technology (NIST) adopted the Advanced Encryption Standard (AES).
data store	A data source such as a relational database (Oracle or DB2) or directory server (Sun Java System Directory Server or Microsoft Active Directory). Each type of data source manages and accesses data differently.
data transfer object	Simple object used to pass data between tiers. It does not contain business logic.
database server	The server where the database is installed.
delegated administration	A scheme for defining the scope and responsibilities of a set of administrators. It permits administrators to delegate a portion of their responsibilities to another administrator.
denial of service	The process of making a system or application unavailable. For example, the result of barraging a server with requests that consume all the available system resources, or of passing malformed input data that can cause the system to stop responding.
delivery address	The e-mail address or the cell phone number where the on-demand token codes will be delivered.
deployment	The arrangement of Authentication Manager instances into appropriate locations in a network to perform authentication.
DES	See data encryption standard.
distribution file	A shared secret between a hardware or software authenticator and an authentication server. The authenticator, sometimes called a token, and the server work together in a time synchronous, or time dependent mode to provide a one-time passcode that the token holder enters at logon.
distribution file password	A password used to protect the distribution file when the distribution file is sent by e-mail to the user.
distributor	A Token Distributor or an administrator with distributor permissions.
DTO	See data transfer object.

Term	Definition
dump	An RSA ACE/Server format used to back up, restore, and merge database information. A dump file is a binary data file that contains all database tables and columns in table-dependency order.
EAP	See extensible authentication protocol.
EAP-POTP	An RSA-proposed IETF (Internet Engineering Task Force) standard that defines the method for one-time password (RSA SecurID) authentication. It provides capabilities, such as end-to-end protection of one-time passwords and support for token exception cases (New PIN, Next Tokencode, and others).
EAP-POTP client	Client that supports the EAP-POTP method.
e-mail notifications	Contain status information about requests for user enrollment, tokens, and user group membership are sent to users who initiated the request. For token requests, e-mail notifications also contain information about how to download and activate tokens. Request Approvers and Token Distributors receive e-mail notifications about requests that require their action. See e-mail templates.
e-mail templates	Templates that administrators can use to customize e-mail notifications about user requests for user enrollment, tokens, user group membership, or the on-demand tokencode service. See e-mail notifications.
emergency access	The process for enabling a token for a user whose token is not available or is not functioning. Used in connection with offline authentication access.
emergency access passcode	A complete authentication code that, if enabled, can be used by a user to perform an offline authentication without an authenticator or PIN.
emergency access tokencode	A partial authentication code that, if enabled, can be used by a user to perform an offline authentication without an authenticator. The user is required to provide his or her PIN.
Enterprise Server license	Authentication Manager license that allows a primary instance, multiple replica instances, and multiple server nodes.
Evaluation license	Authorizes an evaluation copy of the product at a customer site.
event-based token	A hardware token that displays a tokencode whenever the user presses the button on the token.

Term	Definition
excluded words dictionary	A dictionary containing a record of words that users cannot use as passwords. It includes several thousand commonly used words that are likely to be included as part of any dictionary attacks on the system, for example, "password." The excluded words dictionary prevents users from using common, and therefore, easily guessed words as passwords.
extensible authentication protocol (EAP)	An authentication framework that supports multiple authentication methods.
failover mode	The state in which the connection pool management service has to use the secondary connection pools for serving the connection requests, because the primary connection pools are not available due to the failed primary data servers.
four-pass CT-KIP	The exchange of two protocol data units (PDUs) between the client and server.
Global Catalog	A read-only, replicated repository of a subset of the attributes of all entries in an Active Directory forest.
graded authentication	A mechanism for noting the relative strengths of authentication methods (either individually or as combinations). For example, an RSA SecurID token is stronger than a user name and password. Equivalently ranked methods may be used interchangeably.
group membership	See user group.
hardware token	A physical device, such as an RSA SecurID standard card, key fob, or PINPad that displays a tokencode.
high-water mark	The highest numbered interval used by a user to authenticate.
identity attribute definition	Customer-defined attributes that are mapped to an existing customer-defined schema element. They are always stored in the same physical repository as the user's or user group's core attribute data. You can search, query, and report on these attributes. Each identity attribute definition must map to an existing attribute in the LDAP or RDBMS.
Identity Management Services	The set of shared components, toolkits, and services used to build RSA products, for example, Authentication Manager.
identity source	A data store containing user and user group data. The data store can be the internal database or an external directory server, such as Sun Java System Directory Server or Microsoft Active Directory.

Term	Definition
IMS	See Identity Management Services.
initial time-out	The wait time, in seconds, before the initial remote access prompt appears. (The term is used in relation to remote RSA SecurID authentication.)
instance	One single database server, or a database server and one or more server nodes, acting as a single cohesive processing unit. An instance does not have to be a cluster, but a cluster is an instance.
instance ID	This ID identifies a single logical installation of a product or component. For example, in a non-clustered environment, it identifies the database server. In a clustered environment, it identifies the database server and the entire cluster of server nodes. Likewise for web agents, a single agent may have a unique instance ID or an entire server cluster may share a single instance ID.
instance name	The name assigned to an instance. It is either the hostname where a single server node is installed or the cluster name where the clustered instance is installed.
interval	A value used to represent a specific time-based PRN code being generated by an authenticator.
internal database	The Authentication Manager proprietary data source.
J2EE	See Java 2 Enterprise Edition.
Java 2 Enterprise Edition	A framework for building enterprise applications using Java technology.
Java Cryptographic Architecture (JCA)	The set of APIs provided by the Java 2 platform that establishes the architecture and encapsulates limited cryptographic functionality from various cryptographic providers.
Java Cryptographic Extensions (JCE)	The set of APIs provided by the Java 2 platform that encapsulates additional cryptographic functionality from various cryptographic providers.
Java keystore (JKS)	The Java 2 platform implementation of a keystore provided by Sun Microsystems.
Java Management Extensions (JMX)	The set of APIs provided by the Java 2 platform that enables building distributed, web-based, dynamic, and modular solutions for managing and monitoring devices, applications, and service-driven networks.

Term	Definition
Java Messaging Service (JMS)	A standard Java interface for interacting with message queues and topics.
Java Server Pages (JSP)	A commonly used technology for dynamic web content.
JCA	See Java Cryptographic Architecture.
JCE	See Java Cryptographic Extensions.
JKS	See Java keystore.
JMS	See Java Messaging Service.
JMX	See Java Management Extensions.
JSP	See Java Server Pages.
keystore	The Java 2 platform facility for storing keys and certificates.
Key Management services	The management of the generation, use, storage, security, exchange, and replacement of cryptographic keys.
Key Management encryption key	The key used for encryption or decryption operations of keys managed by Key Management services.
license	A verifiable piece of information that represents permission from RSA to use Authentication Manager, its features, or both. A license is a component of the License Management Service.
license category	A way of grouping different types of licenses. The license categories for Authentication Manager are Base Server, Enterprise Server, and Evaluation.
license creation date	The date when the license file is created.
license deployment	Specifies either a server or floating license.
license file	An XML file containing license data that is common across all IMS-based products. The categories of data are: client, product, and feature. A license file is a component of LMS.
license file version	The version of the license schema to which the generated license conforms.
license ID	An internal identifier associated with the license. RSA Manufacturing assigns the license ID.
License Management Service (LMS)	A service responsible for managing and validating product licenses.

Term	Definition
license.rec	A license record file containing the database key needed to extract critical information from the dump file.
LMS	See License Management Service.
local authentication client component	An RSA Authentication Agent component that requires users to enter valid RSA SecurID passcodes to access their Microsoft Windows desktops.
locked license	A license limited to a specific server instance. See server license.
lockout policy	A set of conditions specifying when an account will be locked and whether the account must be unlocked by an administrator or will unlock on its own after a designated amount of time. Lockout policies are applied to security domains. Each realm has a default lockout policy.
log archival	Creates a backup copy of the log for noncurrent, permanent storage.
logging service	A component responsible for recording system, audit, and trace events.
lower-level security domain	In a security domain hierarchy, a security domain that is nested within another security domain.
Management Information Base (MIB)	A type of virtual database used to manage the devices (switches and routers, for example) in a communication network. For example, SNMP uses MIB to specify the data in a device subsystem.
MD5	An algorithm that produces a 128-bit message digest.
member user	A user who is a member of a member user group.
member user group	A user group that is a member of another user group. For example, an organization might define a Sales Managers user group within a North America user group. All member user groups must belong to the same identity source as the parent group, with one exception: any user group from any identity source can be assigned to a parent group that is stored in the internal database.
MIB	See Management Information Base.
Microsoft Management Console (MMC)	A user interface through which system administrators can configure and monitor the system.
MMC	See Microsoft Management Console.

Term	Definition
namespace	A set of names. A namespace defines a scope for a collection of names.
Network Management System (NMS)	Software used to manage and administer a network. The NMS uses SNMP to monitor networked devices and is responsible for polling and receiving SNMP traps from agents in the network.
NMS	See Network Management System.
NMS administrator	The person monitoring the network (through the NMS) for significant events. Also known as a network administrator.
node secret	<p>A long-lived symmetric key that the agent uses to encrypt the data in the authentication request.</p> <p>Authentication Manager generates the authentication request when a user makes a successful authentication attempt. The node secret is known only to the Authentication Manager and the agent.</p>
offline emergency tokencode	Provides emergency access for RSA SecurID for Windows users who require emergency access while authenticating offline. Use this option if the user has a temporarily misplaced, lost, or stolen token. The Offline Emergency Access Tokencode is used with the user's PIN.
offline emergency passcode	Provides emergency access for RSA SecurID for Windows users who require emergency access while authenticating offline. Use this option if the user has forgotten his or her PIN. The Offline Emergency Passcode is used in place of the user's PIN and tokencode.
object	Describes the following: security domains, identity sources, attributes, users, user groups, administrative roles, and policies.
offset	A value used to represent the amount of time an authenticator's internal clock has drifted over time.
on-demand tokencode	<p>Tokencodes delivered by SMS or SMTP. They require the user to enter a PIN to achieve two-factor authentication. On-demand tokencodes are user-initiated, as Authentication Manager only sends a tokencode to the user when it receives a user request.</p> <p>An on-demand tokencode can only be used once, and you configure the lifetime of an on-demand tokencode.</p> <p>See on-demand tokencode service.</p>

Term	Definition
on-demand tokencode service	A service that allows users to request on-demand tokencodes delivered by text message or e-mail, instead of tokens. You configure the on-demand tokencode service for requests using the Security Console. Users must be enabled to receive on-demand tokencodes before they can request them.
one-time tokencode set	Used for online emergency access. A set of tokencodes, each of which can be used only once, and is used with the user's PIN to create a passcode. The administrator can specify how many tokencodes are in the set.
PAM	See Pluggable Authentication Modules.
passcode	A code entered by a user to authenticate. The passcode is a combination of a PIN and a tokencode.
password-based encryption	The process of obscuring information so that it is unreadable without knowledge of the password.
password policy	A set of specifications that define what constitutes a valid password and the conditions under which the password expires. Password policies are applied to security domains.
PDU	See Protocol Data Unit.
permissions	Specifies which tasks an administrator is allowed to perform.
Pluggable Authentication Modules (PAM)	Mechanisms that allow the integration of new authentication methods into an API, independent of the existing API authentication scheme.
primary connection pool	Refers to the connection pools containing the connections to the primary instance database server.
primary instance	The machine with the installation of Authentication Manager at which authentication and all administrative actions occur.
private key	In asymmetric key cryptography, the cryptographic key that corresponds to the public key. The private key is usually protected by some external mechanism (for example, smart card, password encrypted, and so on).
PRN	See pseudorandom number.
Protocol Data Unit	A packet of data exchanged between two application programs across a network.
provisioning	See token provisioning.

Term	Definition
provisioning data	The provisioning server-defined data. This is a container of information necessary to complete the provisioning of a token device. Its format is not specified by CT-KIP because it is outside the realm of CT-KIP, but it is necessary for provisioning.
pseudorandom number (PRN)	A random number or sequence of numbers derived from a single seed value.
public key	In asymmetric key cryptography, the cryptographic key that corresponds with the private key. The public key is usually encapsulated within a certificate.
RADIUS	See Remote Authentication Dial-In User Service.
realm	An entire security domain hierarchy consisting of a top-level security domain and all of its lower-level security domains. A realm includes all of the objects managed within the security domain hierarchy (users, tokens, and password policies, for example). Each realm manages users and user groups in one or more identity sources.
regular time-out	The number of seconds before remote access prompts time out. The term is used in relation to remote RSA SecurID authentication.
Remote Authentication Dial-In User Service (RADIUS)	A UDP-based protocol for administering and securing remote access to a network.
remote EAP (extensible authentication protocol)	A remote authentication feature that requires users to submit RSA SecurID passcodes in order to open remote connections to the network. EAP has a graphical user interface and enhanced security and is supported in both Point-to-Point Protocol (PPP) authentication environments and non-PPP authentication environments, including Point-to-Point Tunneling Protocol (PPTP) VPN connections, 802.1x wired, and 802.11 wireless connections, and other specialized network media.
remote post-dial	Refers to the dial-in Point-to-Point Protocol (PPP) authentication support. With a post-dial terminal-based connection, when remote users dial in, a terminal-like character interface presents a simple user name and passcode prompt. If the right passcode is entered, the PPP connection is established. If the wrong passcode is entered, the dial-up connection is severed.

Term	Definition
replica instance	The machine with the installation of Authentication Manager at which authentication occurs and at which an administrator can view the administrative data. No administrative actions are performed on the replica instance. All administrative actions are performed on the primary instance.
requests	Allows users to enroll, as well as request tokens, the on-demand tokencode service, and user group membership.
Request Approver	A predefined administrative role that grants permission to approve requests from users for user enrollment, tokens, or user group membership.
RSA Credential Manager	A component of Authentication Manager that allows users to request, maintain, and troubleshoot tokens.
RSA EAP	The RSA Security implementation of the EAP 15 authentication protocol that facilitates RSA SecurID authentication to networks in PPP, PPTP (VPN), and 802.1x (wireless or port access) environments.
RSA Operations Console	An administrative user interface through which the user configures and sets up Authentication Manager, for example, adding and managing identity sources, adding and managing instances, and disaster recovery.
RSA Protected OTP	The RSA implementation of the EAP 32 authentication protocol that facilitates RSA SecurID authentication to networks in PPP, PPTP (VPN), and 802.1x (wireless or port access) environments.
RSA Security Console	An administrative user interface through which the user performs most of the day-to-day administrative activities.
RSA Self-Service Console	A user interface through which the user requests, maintains, and troubleshoots tokens.
runtime	Describes automated processing behavior—behavior that occurs without direct administrator interaction.
runtime command	A logon or logoff command.
runtime identity source	The runtime representation of the identity source. Runtime identity sources are used during runtime operations, such as authentication and group membership resolution instead of the corresponding administrative source, which is used for all other operations. This is an integral part of Active Directory forest support, which uses the Global Catalog during runtime operations.

Term	Definition
scope	In a realm, the security domain or domains within which a role's permissions apply.
secondary connection pool	The connection pools containing the connections to the secondary data stores.
Secure Sockets Layer (SSL)	A protocol that uses cryptography to enable secure communication over the Internet. SSL is widely supported by leading web browsers and web servers.
security domain	A container that defines an area of administrative management responsibility, typically in terms of business units, departments, partners, and so on. Security domains establish ownership and namespaces for objects (users, roles, permissions, and so on) within the system. They are hierarchical.
security questions	A way of allowing users to authenticate without using their standard method. To use this service, a user must answer a number of security questions. To authenticate using this service, the user must correctly answer all or a subset of the original questions. The answers to security questions are case sensitive.
self-service	Allows users to perform maintenance tasks and troubleshoot tokens themselves, instead of calling the Help Desk. See also Token Provisioning.
Self-Service Console	See RSA Self-Service Console.
self-service requests	See requests.
self-service troubleshooting policy	Provides an emergency form of authentication that allows users to log on to the RSA Self-Service Console to perform troubleshooting tasks.
server node	An installation of Authentication Manager on a single server host. Each instance has one server node that contains the internal database. You can add additional server nodes to an instance, if your license allows. The additional server nodes cannot operate alone because they do not contain the internal database.
session	An encounter between a user and a software application that contains data pertaining to the user's interaction with the application. A session begins when the user logs on to the software application and ends when the user logs off of the software application.

Term	Definition
session policy	A set of specifications designating the restrictions on overall session lifetime and multiple session handling. Session policies are applied to an instance.
SHA1	A secure hash algorithm function that produces a 160-bit hash result.
shipping address	An address used by distributors to distribute hardware tokens.
Short Message Service (SMS)	A mechanism of delivery of short messages over mobile networks. It is often called text messaging. In Authentication Manager, it is a means of sending tokencodes to a cell phone. Tokencodes delivered by SMS are called on-demand tokencodes.
Simple Mail Transfer Protocol (SMTP)	A TCP/IP protocol used in sending and receiving e-mail. In Authentication Manager, it is a means of sending tokencodes to e-mail accounts. Tokencodes delivered by SMTP are called on-demand tokencodes.
Simple Network Management Protocol (SNMP)	A protocol for exchanging information about networked devices and processes. SNMP uses MIBs to specify the management data, and then uses the User Datagram Protocol (UDP) to pass the data between SNMP management stations and the SNMP agents.
single sign-on (SSO)	The process of requiring only a single user authentication event in order to access multiple applications and resources.
SMS	See Short Message Service.
SMTP	See Simple Mail Transfer Protocol.
snap-in	A software program designed to function as a modular component of another software application. For example, the MMC has a variety of snap-ins that offer different functionality (for example, Device Manager).
SNMP	See Simple Network Management Protocol.
SNMP agent	Software module that performs the network management functions requested by network management stations.
SNMP trap	An asynchronous event that is generated by the agent to tell the NMS that a significant event has occurred. SNMP traps are designed to capture errors and reveal their locations.
SSL	See Secure Sockets Layer.
SSO	See single sign-on.

Term	Definition
Super Admin	<p>An administrator who has all permissions within the system. A Super Admin:</p> <ul style="list-style-type: none"> • Can create and delete realms • Can link identity sources to realms • Has full permissions within any realm • Can assign administrative roles within any realm
symmetric key	A key that allows the same key value for the encryption and decryption of data.
system event	System-generated information related to nonfunctional system events such as server startup and shutdown, failover events, replication events, and so on.
system log	Persistable store for recording system events.
TACACS+	See Terminal Access Controller Access Control System+.
temporary fixed tokencode	Used for online emergency access. This temporary tokencode is used in conjunction with the user's PIN to create a passcode. The user can use this tokencode more than once. The administrator can configure the expiration date and other Temporary Fixed Tokencode attributes.
Terminal Access Controller Access Control System+ (TACACS+)	A remote authentication protocol that is used to communicate with an authentication server. Allows a remote access server to communicate with an authentication server to determine if a user has access to the network.
time-based token	A hardware token that always displays a tokencode and the tokencode changes automatically every 60 seconds.
token	A hardware device or software program that generates a pseudorandom number that is used in authentication procedures to verify a user's identity.
Token Distributor	A predefined administrative role that grants permission to act upon requests from users for tokens. Distributors record how they plan to deliver tokens to users and close requests.
token provisioning	The automation of all the steps required to provide enrollment, user group membership, RSA SecurID tokens, and the on-demand tokencode service to users. See also self-service.
tokencode	The random number displayed on the front of a user's RSA SecurID token. Tokencodes change at a specified time interval, typically every 60 seconds.

Term	Definition
top-level security domain	The top-level security domain is the first security domain in the security domain hierarchy (realm). The top-level security domain is unique in that it links to the identity source or sources and manages password, locking, and authentication policy for the entire realm.
trace log	Persistable store for trace information.
trusted realm	A trusted realm is a realm that meets these criteria: <ul style="list-style-type: none"> • It is located in a different deployment than your realm. • It has exchanged configuration settings with your realm. The settings are in an XML file called a trust package.
trust package	An XML file that contains configuration information about the realm.
two-factor authentication	An authentication protocol requiring two different ways of establishing and proving identity, for example, something you have (such as an authenticator) and something you know (such as a PIN).
two-pass CT-KIP	The exchange of one protocol data unit (PDU) between the client and server.
UDP	See User Datagram Protocol.
user	An account managed by the system that is usually a person, but may be a computer or a web service.
User Datagram Protocol (UDP)	A protocol that allows programs on networked computers to communicate with one another by sending short messages called datagrams.
user group	A collection of users, other user groups, or both. Members of the user group must belong to the same identity source. User group membership determines access permission in some applications.
User ID	A character string that the system uses to identify a user attempting to authenticate. Typically a User ID is the user's first initial followed by the last name. For example, Jane Doe's User ID might be <i>jdoe</i> .
workflow	The movement of information or tasks through a work or business process. A workflow can consist of one or two approval steps and a distribution step for different requests from users.

Term	Definition
workflow participant	Either approvers or distributors. Approvers review, approve, or defer user requests. Distributors determine the distribution method for token requests and record the method for each request. See also workflow.

Index

A

- account.ini, 177
- Acct-Authentic, 186
- Acct-Delay-Time, 186
- Acct-Status-Type, 186
- Acct-Termination-Cause, 187
- Active Directory
 - definition, 425
 - Global Catalog options, 119
 - group membership, 240
 - password policy, 240
- Active Directory forest
 - definition, 425
- Activity Monitor, 206
 - Administrator Activity type, 206
 - Authentication Activity type, 206
 - clearing messages from, 206
 - displaying failure events, 206
 - displaying successful events, 206
 - displaying warning events, 206
 - pausing, 206
 - System Activity type, 206
 - types, 206
 - using for troubleshooting, 400
 - viewing event details with, 207
- AD. *See* Active Directory
- adadmreg.exe, 68
- adjudicator
 - definition, 425
- administrative
 - scope, 38
- administrative command
 - definition, 425
- administrative role
 - definition, 425
- administrative roles, 32
 - assigning, 42
 - predefined, 32
 - provisioning, 135
 - scope, 38
- administrator
 - adding, 32
 - definition, 425
 - modifying permissions, 109
- Advanced Encryption Standard
 - definition, 425
- AES. *See* Advanced Encryption Standard
- agent
 - definition, 425
 - agent auto-registration, 68
 - agent auto-registration utility
 - definition, 425
 - agent configuration file
 - updating, 422
 - agent host
 - definition, 425
 - Agent Protocol Server
 - definition, 426
 - agent record
 - create, 66
 - agent types
 - standard, 67
 - web, 67
 - approval steps, 134
 - approver
 - definition, 426
 - Archive Requests utility, 141, 265
 - archive-ucm-request command, 265
 - attribute
 - definition, 426
 - attribute exclusions, 41
 - attribute mapping, 23, 240
 - definition, 426
 - for provisioning, 142
 - audit information
 - definition, 426
 - audit log
 - definition, 426
 - authentication
 - definition, 426
 - offline, 59
 - RSA Self-Service Console, 129
 - authentication agent
 - hostname resolution, 420
 - IP address resolution, 420
 - testing user access to restricted agent, 415
 - updating agent configuration file, 422

- authentication agents
 - create, 66
 - download, 66
 - embedded, 66
 - enabling for trusted realms, 147, 152
 - restricted, 73
 - restricting user access with, 73–74
 - standard, 67
 - unrestricted, 73
 - web, 67
- authentication authority
 - definition, 426
- authentication broker
 - definition, 426
- Authentication Manager
 - browser settings, 411
 - common problems and solutions, 391
 - fails to start, 418
 - ports used by, 412
 - resynchronizing with Universal Coordinated Time (UCT), 421
 - supported browsers, 411
- authentication method
 - definition, 426
- authentication methods for Self-Service Console, 129
- authentication policy
 - definition, 426
- authentication protocol
 - definition, 426
- Authentication Server
 - definition, 427
- authenticator
 - definition, 427
- authorization
 - definition, 427
- authorization data
 - definition, 427
- Automated Agent Registration and Update utility, 68
- automated backups, 211
- automatic synchronization, 229
- auto-registration
 - definition, 427

B

- backing up the internal database, 209
- Base Server license, 121, 122
 - definition, 427
 - self-service, 125
- binding a software token to a device, 82, 84
- browser
 - adding Security Console URL to trusted sites, 412
 - enabling JavaScript, 411
 - requirements, 411
 - settings, 411
- browser security, 411
- browser support, 411
- browsers
 - supported by Authentication Manager, 411
- Business Continuity option
 - definition, 427

C

- cache
 - configuring for improved performance, 118
 - setting cached object limits, 118
 - setting refresh interval, 118
 - troubleshooting, 415
 - viewing current utilization, 118
- certificate
 - definition, 427
- certificate DN
 - definition, 427
- chained authentication
 - definition, 427
- changing replica instance logging level, 405
- clearing
 - incorrect passcodes, 106
 - PINs, 104
- clearing incorrect passcodes, 417
- clearing SecurID PINs, 417
- client time-out
 - definition, 427

- CLU
 - Archive Requests, 265
 - Collect Product Information, 268
 - Database Storage Management, 277
 - Import PIN Unlocking Key, 269
 - Manage Backups, 271
 - Manage Batchjob, 275
 - Manage Operations Console
 - Administrators, 282
 - Manage Secrets, 285
 - Register Custom Extension, 288
 - Restore Admin, 235
 - Set Trace, 298
 - store, 405
 - Update Instance Nodes, 309
 - User Groups and Token Bulk Requests, 302
 - Verify Archive Log, 307
- CLU command
 - archive-ucm-request, 265
 - collect-product-info, 269
 - import-bulk-request, 302
 - import-puk, 269
 - manage-backups, 274
 - manage-batchjob, 275
 - manage-database, 280
 - manage-oc-administrators, 282
 - manage-secrets, 287
 - register-custom-extension, 288
 - restore-admin, 236
 - set-trace, 298, 300
 - store, 405
 - update-instance-nodes, 310
 - verify-archive-log, 307
- CLU. *See* command line utility
- cluster
 - definition, 428
- Collect Product Information utility, 268
- collecting shipping addresses, 141
- collect-product-info command, 269
- command line utility
 - definition, 428
- communication
 - port usage, 413
- conditional statements
 - e-mail templates, 243, 247
 - examples, 243
- configuration files
 - authentication manager, 71
- configuring
 - emergency access for provisioning, 139
 - provisioning, 134
 - RSA Credential Manager, 128, 129
- confirmation code
 - confirming and exchanging, 150
- connection pool
 - definition, 428
- Console. *See* RSA Operations Console, RSA Security Console, RSA Self Service Console
- contact list
 - definition, 428
- contact lists
 - automatic, 72
 - manual, 72
- context-based authentication
 - definition, 428
- core attributes
 - definition, 428
- Credential Manager Configuration - Home page, 127
- Credential Manager Provisioning
 - definition, 428
 - license, 122
- Cryptographic Token-Key Initialization Protocol
 - client, 428
 - enabled token, 428
 - server, 428
 - toolkit, 428
- cryptographic token-key initialization protocol (CT-KIP), 84
- CT-KIP, 84
 - reconfiguring after promoting a replica, 422
- CT-KIP. *See* Cryptographic Token-Key Initialization Protocol
- custom attributes, 23, 132
- customer name
 - definition, 429
- customizing
 - e-mail notifications, 243, 244
 - non-workflow operations, 254
 - RSA Self-Service Console Help, 126, 252
 - RSA Self-Service landing page, 126
 - token graphics, 252
 - user profiles, 132
 - workflow operations, 254

D

- data encryption standard
 - definition, 429
- data store
 - definition, 429
- data transfer object
 - definition, 429
- database server
 - definition, 429
 - system requirements, 407
- Database Storage Management utility, 277
- default agent settings, 68
- delegated administration
 - definition, 429
- deleting self-service troubleshooting
 - policies, 133
- delivery address
 - definition, 429
- denial of service
 - definition, 429
- deploying software tokens
 - remote token-key generation, 84
- deployment
 - definition, 429
- DES. *See* data encryption standard
- directories, 21
- disabling
 - e-mail notifications, 136
- disabling tokens, 102
- disabling users, 93
- disaster recovery
 - promoting a replica, 221
- disk space, 408
- Displaying administrative roles with view or none permission for attribute, 111
- distribution file
 - definition, 429
- distribution file password
 - definition, 429
- distribution reports, 142
- distribution step, 134
- distribution, third-party, 141
- distributor
 - definition, 429
- DN
 - modifying, 29
- DNS
 - IP address resolution, 420
 - name resolution, 420

- Dropped Packet, 173, 175, 176
- DTO. *See* Data Transfer Object
- dump file
 - definition, 430
- duplicating
 - self-service troubleshooting
 - policies, 133
- Dynamic Host Configuration Protocol, 68

E

- EAP
 - definition, 430
- EAP-POTP
 - client, 430
 - definition, 430
- editing self-service troubleshooting
 - policies, 133
- e-mail notification
 - definition, 430
- e-mail notifications
 - configuring, 136
 - customizing, 243
 - customizing for proxy servers, 245
 - customizing guidelines, 244
 - enabling and disabling, 136
 - recipients, 136
- e-mail servers, 136
- e-mail template
 - definition, 430
- e-mail templates
 - conditional statement guidelines, 250
 - conditionalizing, 243
 - guidelines for customizing, 243
 - tags, 247
- emergency access, 132, 138
 - definition, 430
 - for offline authentication, 99
 - for online authentication, 97
 - one-time-tokencode, 138
 - provisioning, 138
 - troubleshooting, 138
- emergency access codes
 - setting formats, 57
- emergency access for provisioning
 - configuring, 139
- emergency access passcode
 - definition, 430

- emergency access tokencode
 - assigning for offline authentication, 100
 - definition, 430
 - for offline authentication, 96, 99, 100, 417
 - for online authentication, 96, 97, 417
 - for troubleshooting, 138
 - lifetime, 98
 - emergency passcode
 - assigning for offline authentication, 100
 - for offline authentication, 96, 99, 100, 417
 - enabling
 - e-mail notifications, 136
 - enabling JavaScript in the browser, 411
 - enabling tokens, 102
 - enabling users, 93
 - enrollment
 - customizing user profiles for, 132
 - identity sources for, 129
 - provisioning, 128
 - security domains for, 131
 - self-service, 128
 - Enterprise Server license, 121, 122
 - definition, 430
 - provisioning, 125
 - Evaluation license
 - definition, 430
 - event-based token
 - definition, 430
 - event-based tokens, 77
 - setting tokencode ranges for, 53
 - excluded words dictionary, 50
 - definition, 431
 - existing Authentication Manager
 - migrating data, 31
 - extensible authentication protocol
 - definition, 431
 - external directories, 21
 - External Unique Identifier (EXUID), 29
- F**
- Failed Authentication, 174
 - Failed on Checklist, 174
 - failover mode
 - definition, 431
 - fatal log messages, 195
 - Firefox, 411
- firewall
 - required open ports, 413
 - firewalls
 - assessing the impact of, 412
 - fixed passcodes
 - requiring periodic changes, 55
 - restricting reuse, 56
 - setting character requirements, 57
 - forcing SecurID PIN changes, 105, 417
 - four-pass CT-KIP
 - definition, 431
 - fully qualified domain name (FQDN)
 - changing on a cluster deployment, 312
 - changing on a replicated deployment, 329
 - changing on a standalone deployment, 311
- G**
- Global Catalog
 - definition, 431
 - mapping to identity source, 237
 - graded authentication
 - definition, 431
 - group membership
 - definition, 431
 - group membership for provisioning, 135
- H**
- hardware requirements, 408
 - hardware token
 - definition, 431
 - hardware tokens
 - assigning, 79
 - delivering to users, 80
 - mailing, 80
 - unassigning, 79
 - Help Desk calls, 143
 - high-water mark
 - definition, 431
- I**
- identity attribute
 - attribute mapping, 142
 - definition, 431
 - identity attribute definitions, 23
 - mapping, 23, 240
 - Identity Management Services
 - definition, 431

- identity source
 - attribute mapping, 119
 - definition, 431
 - deleting, 29
 - edit search results timeout, 119
 - Global Catalog options, 119
 - lockout policies, 416
 - re-indexing your directory, 24
 - selecting for Credential Manager, 129
 - updating Active Directory options, 119
 - updating attributes of, 119
- identity sources, 21
- Import PIN Unlocking Key utility, 269
 - option flags, 270
 - running, 270
- import-bulk-request command, 302
- import-puk command, 269
- IMS. *See* Identity Management Services
- incorrect passcodes
 - limiting the number of incorrect passcodes allowed, 52
- incorrect self-service troubleshooting
 - authentication attempts, 133
- information log messages, 196
- initial time-out
 - definition, 432
- instance
 - definition, 432
- instance ID
 - definition, 432
- instance name
 - definition, 432
- Insufficient Resources, 174, 175, 176
- internal database
 - backing up and restoring, 209
 - definition, 432
 - restoring, 273
 - transferring, 272
- Internet Explorer, 411
- interval
 - definition, 432
- Invalid Request, 173, 175, 176
- IP address
 - changing on a cluster deployment, 312
 - changing on a replicated deployment, 329
 - changing on a standalone deployment, 311
 - resolution of, 420

J

- J2EE. *See* Java 2 Enterprise Edition
- Java 2 Enterprise Edition
 - definition, 432
- Java Cryptographic Architecture
 - definition, 432
- Java Cryptographic Extensions
 - definition, 432
- Java keystore
 - definition, 432
- Java Management Extensions
 - definition, 432
- Java Messaging Service
 - definition, 433
- Java Server Pages
 - definition, 433
- JavaScript, 411
- JCA. *See* Java Cryptographic Architecture
- JCE. *See* Java Cryptographic Extensions
- JKS. *See* Java keystore
- JMS. *See* Java Messaging Service
- JMX. *See* Java Management Extensions
- JSP. *See* Java Server Pages

K

- Key Management encryption key
 - definition, 433
- Key Management services
 - definition, 433
- keyboard shortcuts
 - setting in the Security Console, 121
- keystore
 - definition, 433

L

- landing page. *See* Welcome, what would you like to do? page
- language
 - of Security Console, 121
- LDAP, 21
 - orphaned users, 29
 - SSL setup, 23
- license
 - Base Server, 122, 427
 - Business Continuity option, 122
 - definition, 433
 - Enterprise Server, 122, 430
 - Evaluation, 430
 - RSA Credential Manager provisioning option, 122

- license category
 - definition, 433
 - license creation date
 - definition, 433
 - license deployment
 - definition, 433
 - license file
 - definition, 433
 - license file version
 - definition, 433
 - license ID
 - definition, 433
 - determining, 14
 - License Management Service
 - definition, 433
 - license.rec
 - definition, 434
 - licenses, 121
 - Base Server, 121
 - category, 123
 - Enterprise Server, 121
 - installation date, 123
 - installed, 123
 - issue date, 123
 - license ID, 123
 - limit, 122
 - limitation type, 122
 - number in use, 122
 - status, 122
 - types, 121
 - upgrading, 123
 - viewing, 122, 123
 - limiting incorrect authentication attempts, 133
 - link identity source to realm, 28
 - Linux
 - requirements, 408
 - LMS. *See* License Management Service
 - Local Authentication Client
 - definition, 434
 - locked license
 - definition, 434
 - locking users out of the system, 57
 - lockout policy, 57
 - definition, 434
 - unlocking users, 94
 - log archival
 - definition, 434
 - log messages
 - archived, 400
 - flat files, 401
 - viewing in real time, 400
 - LogAccept, 183
 - logging
 - administrative audit log, 195
 - archiving log files, 197
 - configuring log settings, 195
 - enabling trace logging, 404
 - error messages, 195
 - fatal level messages, 195
 - flat files and descriptions, 401
 - information level messages, 196
 - levels, 195
 - runtime audit log, 195
 - storing log data, 196
 - system log, 195
 - trace log, 195
 - types of logs, 195
 - verbose level messages, 196
 - warning level messages, 195
 - logging service
 - definition, 434
 - LogLevel, 183
 - logon aliases, 107
 - logon methods, 133, 143
 - LogReject, 183
 - lower-level security domain
 - definition, 434
- M**
- Manage Backups utility, 211, 271
 - Manage Batchjob utility, 275
 - cancelling a running job, 276
 - deleting a completed job, 276
 - option flags, 277
 - running, 275
 - viewing debug messages, 276
 - viewing job status, 276
 - Manage Database utility. *See* Database Storage Management utility
 - Manage Operations Console Administrators utility, 282
 - Manage Secrets utility, 285
 - manage-backups command, 274
 - manage-batchjob command, 275
 - manage-database command, 280
 - Management Information Base
 - definition, 434
 - manage-oc-administrators command, 282

- manage-secrets command, 287
- managing internal database backups, 271
- manual synchronization, 229
- member user
 - definition, 434
- member user group
 - definition, 434
- memory requirements, 408
- Merging offline authentication policies, 60
- message IDs, 361
- MIB. *See* Management Information Base
- Microsoft Management Console
 - definition, 434
- Microsoft Management Console (MMC)
 - accessing help for Authentication Manager snap-in, 257
 - assigning fixed passcodes, 259
 - assigning tokens, 258
 - changing token security domain, 259
 - clearing incorrect passcodes, 259
 - clearing PINs, 260
 - disabling tokens, 258, 259
 - editing authentication attributes, 259
 - editing token properties, 259
 - emergency access for offline authentication, 260, 262
 - emergency access for online authentication, 260, 261
 - emergency access tokencode for offline authentication, 260, 262
 - emergency access tokencode for online authentication, 260, 261
 - emergency access tokencode lifetime, 261
 - enabling tokens, 258
 - forcing PIN changes, 260
 - launching the Authentication Manager snap-in, 255
 - logon aliases, 259
 - managing PINs, 260
 - providing emergency access for users, 260
 - replacing tokens, 259
 - requiring PINs, 260
 - snap-in fails to start, 419
 - unassigning tokens, 258
 - viewing token information, 259
 - Windows password integration, 259

- Microsoft Windows password
 - integration, 61
- migrating users and tokens, 31
- MMC. *See* Microsoft Management Console

N

- namespace
 - definition, 435
- NAT. *See* Network Address Translation
- Network Address Translation
 - resolving IP addresses, 421
- Network Management System
 - definition, 435
- NMS
 - message IDs, 361
- NMS administrator
 - definition, 435
- NMS. *See* Network Management System
- node secret, 112
 - definition, 435
 - managing, 421
 - manual delivery of, 112
 - Node Secret Load utility, 112
 - refreshing using the Node Secret Load utility, 113
- Node Secret Load utility, 112
 - platform versions, 113
 - refreshing the node secret, 113
- Node Secret Utility, 112
- non-workflow operations, 254

O

- object
 - definition, 435
- offline authentication, 59, 61
 - emergency access tokencode, 96, 417
 - emergency passcode, 96, 99, 417
 - policies, 59
 - providing emergency access for, 96, 99, 417
- offline authentication data
 - refreshing users' supplies, 63
- offline authentication policies
 - handling devices that do not meet security recommendations, 62
 - merging, 60
- offline authentication requirements
 - setting, 59
- offline days
 - recharging, 63
- offline emergency codes, 62

- offline emergency passcodes, 62
 - offline emergency tokencodes, 62
 - offset
 - definition, 435
 - on-demand tokencode
 - definition, 435
 - on-demand tokencode service, 85, 138, 142
 - definition, 436
 - on-demand tokencodes, 85
 - changing service providers, 89
 - configuring Authentication Manager for, 87
 - configuring for e-mail delivery, 88
 - configuring for text message delivery, 87
 - configuring message text, 88
 - configuring the plug-in, 89
 - configuring tokencode lifetime, 88
 - enabling users for, 89
 - setting PIN for, 90
 - one-time tokencode, 138
 - definition, 436
 - online authentication
 - emergency access tokencode, 96, 417
 - providing emergency access for, 96, 97, 417
 - online passcodes
 - setting minimum lengths, 61
 - Operations Console
 - definition, 438
 - logging on, 16
 - Super Admin credentials, 17
 - URL, 16
 - user name and password, 16
 - options
 - Business Continuity, 122, 427
 - provisioning, 122
 - Short Message Service, 122
 - orphaned LDAP users, 29
- P**
- PAM. *See* Pluggable Authentication Module
 - parent security domain, 136
 - passcode
 - definition, 436
 - fixed, 106
 - passcodes
 - clearing incorrect, 106, 417
 - emergency access, 96, 99, 417
 - password
 - Active Directory policy, 240
 - encrypted properties file, 285
 - password policy
 - definition, 436
 - setting, 47
 - password-based encryption
 - definition, 436
 - passwords
 - limiting lengths, 50
 - requiring periodic changes, 49
 - requiring system-generated, 48
 - resetting, 132
 - restricting reuse, 49
 - self-service troubleshooting, 133
 - setting character requirements, 51
 - setting requirements, 47
 - software token, 137
 - using an excluded words dictionary, 50
 - permissions
 - administrative, 37
 - definition, 436
 - modifying, 109
 - PIN
 - clearing, 104
 - forcing the change of, 105
 - setting for on-demand tokencodes, 90
 - setting requirements for, 104, 105
 - troubleshooting, 132
 - Pluggable Authentication Module
 - definition, 436
 - ports, 413
 - predefined administrative roles for
 - provisioning, 135
 - primary connection pool
 - definition, 436
 - primary instance
 - definition, 436
 - recovering, 220
 - private key
 - definition, 436
 - PRN. *See* pseudorandom number
 - promoting a replica instance, 221
 - properties file, 285
 - protecting tokens, 137
 - Protocol Data Unit
 - definition, 436

- provisioning
 - administrative roles, 135
 - attribute mapping, 142
 - configuring, 134, 136
 - customizing e-mail notifications, 243
 - customizing token graphics, 252
 - definition, 125, 436
 - emergency access, 138
 - enrolling, 128
 - license, 122
 - one-time tokencode, 138
 - protecting tokens, 137
 - selecting on-demand tokencode service, 142
 - selecting tokens, 137
- provisioning data
 - definition, 437
- proxy server replacement tags, 245
- proxy servers
 - customizing e-mail notifications, 245
- pseudorandom number
 - definition, 437
- public key
 - definition, 437
- R**
- RADIUS**
 - accounting log file, 183
 - accounting statistics, 174
 - associating profiles with users, 161
 - authentication log files, 178
 - backing up a server, 189
 - client authentication and accounting statistics, 176
 - default profiles, 167
 - EAP-POTP configuration, 172
 - editing dictionary files, 192
 - editing server configuration files, 192
 - log retention period, 182
 - managing, 160
 - managing clients, 169
 - managing replication, 171
 - managing servers, 170
 - managing user attributes, 169
 - monitoring system usage, 173
 - overview, 159
 - profile assignments, 168
 - profiles, 161
 - promoting a replica, 191
 - replication, 160
 - restoring a server, 189
 - server log file, 182
 - single-value and multiple-value user attributes, 165
 - trusted realms, 164
 - trusted users, 164
 - usage statistics, 173
 - user attributes, 163
- RADIUS clients**
 - managing, 169
- RADIUS profile**
 - assignments, 168
 - associating with users, 161
 - default, 167
 - definition, 161
 - managing checklist and return lists, 167
- RADIUS replication, 171**
- RADIUS servers**
 - adding, 170
 - backing up, 189
 - changing the IP address, 193
 - changing the name, 193
 - editing, 171
 - editing configuration files, 192
 - editing dictionary files, 192
 - managing, 170
 - promoting a replica, 191
 - restoring from backup, 189
 - starting and stopping, 170
- RADIUS user attributes**
 - definition, 163
 - managing, 169
 - single-value and multiple-value, 165
- RADIUS. *See* Remote Authentication Dial-In User Service**
- read-only or read/write access, 130, 132
- realm
 - definition, 437
- realm certificate
 - generating, 150
- realms
 - creating, 17
- reattaching a replica instance, 229
- recharging offline days, 63
- Red Hat Package Manager
 - versions required, 409, 410
- Register Custom Extension utility, 288
- register-custom-extension command, 288
- regular time-out
 - definition, 437

- Remote Authentication Dial-In User Service
 - definition, 437
 - platform requirements, 407
- remote EAP
 - definition, 437
- remote post-dial
 - definition, 437
- remote token-key generation (CT-KIP), 84
- replacing tokens, 101
 - choosing the replacement token, 101
 - using RSA Self-Service Console, 132
 - with next available token, 101
- replica instance
 - cause of failure, 219
 - changing logging level, 405
 - definition, 438
 - detecting failed, 219
 - promoting, 221
 - reattaching, 229
 - recovering, 220
 - removing, 231
 - resynchronizing diverged, 232
 - synchronizing with primary instance, 229
- reports, 198
 - changing ownership of, 202
 - creating, 198
 - download formats, 203
 - downloading, 203
 - jobs, 201
 - running, 200
 - scheduling recurring, 201
 - scope of, 202
 - templates, 198
 - viewing, 203
- Request Approver
 - default permissions, 135
 - definition, 438
 - tasks, 134
- requesting emergency access, 132
- requests, 140
 - definition, 438
 - emergency access, 132
 - token replacement, 139
 - viewing, 140
 - workflow definitions, 134
- requirements
 - system, 407
- Restore Admin utility, 235
- restore-admin command, 236
- restoring the internal database, 212
- restricted agents
 - providing access to, 107
 - testing user access to, 415
- resynchronizing RSA Authentication Manager with Universal Coordinated Time (UCT), 421
- resynchronizing tokens, 103, 418
- roles
 - modifying, 109
- RPM. *See* Red Hat Package Manager
- RSA ACE/Server
 - migrating from, 31
- RSA Credential Manager
 - configuring, 136
 - Credential Manager Configuration - Home page, 127
 - customizing, 243, 252
 - definition, 438
 - described, 125
 - selecting user groups, 135
 - Welcome, what would you like to do? page, 126
- RSA EAP
 - definition, 438
- RSA Operations Console
 - definition, 438
 - logging on, 16
 - Super Admin credentials, 17
 - URL, 16
 - user name and password, 16
- RSA password, 129
- RSA Protected OTP
 - definition, 438
- RSA SecurID authentication
 - overview, 65
- RSA SecurID for Windows, 61
- RSA SecurID PINs
 - requiring periodic changes, 53
 - restricting length, 54
 - restricting reuse, 54
- RSA SecurID tokens
 - selecting for provisioning, 137

- RSA Security Console
 - approving requests, 140
 - configuring Credential Manager, 128, 129, 136, 139
 - customizing Credential Manager, 132, 243, 252
 - definition, 438
 - distributing tokens, 141
 - editing users with, 29
 - logging on, 15
 - protecting, 45
 - protecting tokens for provisioning, 137
 - selecting identity sources for Credential Manager, 129
 - selecting security domains for Credential Manager, 131
 - selecting tokens for provisioning, 137
 - URL, 16
- RSA Self-Service Console, 126, 143
 - customizing Help, 126
 - definition, 438
 - emergency access, 132, 138
 - Help Desk calls, 143
 - impact of read-only or read/write access, 130
 - logon methods, 143
 - provisioning, 128
 - replacing tokens, 132
 - self-service, 128
 - tasks, 128, 130
 - troubleshooting, 132, 139
 - URL, 143
- RSA Self-Service Console Frequently Asked Questions, 252
- RSA Self-Service Console Help, 252
- runtime
 - definition, 438
- runtime command
 - definition, 438
- runtime identity source
 - definition, 438
- S**
- scope
 - administrative, 38
 - definition, 439
 - exceptions for Credential Manager, 140
- sdconf.rec, 71
- SDTID file format, 137
- search results
 - setting number per page, 121
- secondary connection pool
 - definition, 439
- Secure Sockets Layer
 - definition, 439
- SecurID for Microsoft Windows, 61
- SecurID PIN
 - clearing, 417
 - forcing the change of, 417
- SecurID PINs
 - requiring periodic changes, 53
 - restricting length, 54
 - restricting reuse, 54
 - setting character requirements, 55
- Security Console
 - adding to browser trusted sites, 412
 - browser settings for, 411
 - closing administrator sessions, 117
 - configuring preferences, 121
 - definition, 438
 - fails to start, 419
 - keyboard shortcuts, 121
 - language, 121
 - length of sessions, 116
 - limiting concurrent sessions, 115
 - search results, 121
 - session data, 114
 - session handling, 114
 - supported browsers, 411
 - timeout periods, 116
 - URL, 16
 - viewing administrator sessions, 117
- Security Console. *See* RSA Security Console
- security domain
 - definition, 439
- security domains, 18, 19
 - adding, 20
- security questions
 - definition, 439
 - restrictions, 132
 - self-service troubleshooting, 132
- security when changing token requests, 139
- selecting
 - identity sources for enrollment, 129
 - security domains for enrollment, 131
 - user groups for provisioning, 135
- self-service
 - definition, 125, 439
 - enrolling, 128
- Self-Service Console
 - definition, 438

- Self-Service Console. *See* RSA Self-Service Console
- self-service requests
 - definition, 439
- self-service troubleshooting, 132, 133
- self-service troubleshooting policy
 - definition, 439
- server
 - changing the IP address of, 423
- server node
 - definition, 439
- services
 - defined, 413
 - protocols used, 413
- session
 - definition, 439
- session handling, 114
 - closing administrator sessions, 117
 - concurrent sessions, 115
 - length of sessions, 116
 - timeout periods, 116
 - viewing administrator sessions, 117
- session policy
 - definition, 440
- Set Trace utility, 298
- set-trace command, 298, 300
- set-trace diagnostic monitors, 301
- shipping address
 - definition, 440
- shipping addresses, 141
- Short Message Service
 - definition, 440
- Simple Network Management Protocol
 - definition, 440
- Simple Network Management Protocol. *See* SNMP
- single sign-on
 - definition, 440
- SMS
 - changing service providers, 89
 - configuring, 87
 - definition, 440
 - enabling users for, 89
 - license, 122
 - sending tokencodes using, 85
 - service provider, 86
 - setting PIN for, 90
- SMTP
 - changing service providers, 89
 - configuring, 87
 - definition, 440
 - enabling users for, 89
 - sending tokencodes using, 85
 - setting PIN for, 90
- snap-in
 - definition, 440
- SNMP
 - definition, 440
 - gets and traps, 203
 - interpreting trap values, 205
 - message IDs, 361
 - trapping, 203
- SNMP agent
 - definition, 440
- SNMP trap
 - definition, 440
- SNMP trapping
 - configuring the Security Console for, 203
- software token files
 - guidelines, 137
 - passwords, 137
- software token types
 - adding, 120
- software tokens
 - binding to a device, 82, 84
- SSL
 - LDAP, 23
- SSL. *See* Secure Sockets Layer
- SSO. *See* single sign-on
- standard agents, 67
- status, viewing request, 140
- store command, 405
- Store utility
 - changing replica instance logging level, 405
 - displaying administrative roles with view or none permission for attribute, 111
- Sun Java System Directory Server
 - re-indexing, 24
- Super Admin, 33
 - definition, 441
- symmetric key
 - definition, 441
- synchronizing data between instances, 229

- system
 - fingerprint, 285
 - required packages, 409, 410
- system event
 - definition, 441
- system log
 - definition, 441
- system requirements
 - Linux, 408
 - Microsoft Windows, 408
 - Solaris, 410
 - supported browsers, 411
- systemfields.properties, 285
- T**
- TACACS+. *See* Terminal Access Controller Access Control System+
- templates
 - reporting, 198
- temporary fixed tokencode, 138
 - definition, 441
- Test Access, 415
- time-based token
 - definition, 441
- time-based tokens, 76
- Token Distributor
 - default permissions, 135
 - definition, 441
 - tasks, 134
- token graphics, 252
- token policy
 - emergency access tokencode format, 97
 - setting, 51
 - setting PIN requirements, 104, 105
- token provisioning
 - definition, 441
- token records
 - importing, 78
 - transferring to other security domains, 79
- token requests, 139
- tokencode
 - definition, 441
 - emergency access troubleshooting, 138
- tokencodes
 - delivering using cell phone, 85
 - delivering using email, 85
 - emergency access, 96, 417
 - ranges for event-based tokens, 53
- tokens
 - damaged, 95, 416
 - definition, 441
 - deploying to users, 75–91
 - disabling, 102
 - distributing, 141
 - enabling, 102
 - event-based, 77
 - expired, 95, 139, 416
 - guidelines for protecting, 137
 - lost, 95, 416
 - provisioning, 139
 - replacement, 132
 - replacing, 101
 - resynchronizing, 103, 132, 418
 - security when changing types, 139
 - selecting for provisioning, 137, 142
 - stolen, 95, 416
 - time-based, 76
- top-level security domain
 - definition, 442
- trace log
 - definition, 442
- TraceLevel, 183
- troubleshooting, 132, 133
 - Collect Product Information utility, 268
 - RSA Self-Service Console, 132
 - Set Trace utility, 298
 - Store utility, 405
- trust
 - configuring, 149
 - confirmation code, 150
 - creating, 149
 - editing, 152
 - generating a realm certificate, 150
 - generating and uploading the trust package, 150
 - one-way, 148
- trust package
 - definition, 442
 - generating and uploading, 150
- trusted realm
 - definition, 442

- trusted realms
 - adding authentication agents for, 152
 - administrator tasks, 147
 - configuring, 151
 - confirming and exchanging the confirmation code, 150
 - creating a trust, 148
 - creating and configuring a trust, 149
 - creating trusted user groups, 156
 - creating trusted users, 147, 156
 - editing a trust, 152
 - enabling authentication agents for, 147, 152
 - enabling duplicate agents, 154
 - enabling duplicate agents for access control, 154
 - enabling duplicate agents for alias resolution, 155
 - generating a realm certificate, 150
 - generating and uploading the trust package, 150
 - node secret, 397
 - one-way, 146
 - one-way trust, 148
 - overview, 145
 - RADIUS, 157
 - steps to create, 147
 - trusted user groups, 155
 - trusted user name identifier, 151
 - trusted users, 155
 - two-way, 146
 - trusted user groups
 - creating, 156
 - trusted user name identifier, 151
 - trusted users, 146
 - allowing RADIUS authentication, 157
 - creating, 147, 156
 - trusts
 - creating, 147, 148
 - one-way, 146
 - two-way, 146
 - two-factor authentication
 - definition, 442
 - two-pass CT-KIP
 - definition, 442
- ## U
- UDP. *See* User Datagram Protocol
 - Universal Coordinated Time (UCT), 421
 - UNIX
 - designating a default shell for, 106
 - unlink identity source from realm, 30
 - unlocking users, 94, 133, 416
 - Update Instance Nodes utility, 309
 - update-instance-nodes command, 310
 - updating an agent configuration file, 422
 - URL, 16, 143
 - User Datagram Protocol
 - definition, 442
 - user groups, 135
 - definition, 442
 - for provisioning, 135
 - providing access to restricted agents, 107
 - User Groups and Token Bulk Requests utility, 141, 302
 - User ID
 - definition, 442
 - user profiles, 132
 - users
 - clearing a PIN for, 104
 - clearing a SecurID PIN for, 417
 - default shell for UNIX, 106
 - definition, 442
 - disabling, 93
 - editing with the RSA Security Console, 29
 - enabling, 93
 - forcing a PIN change for, 105
 - forcing a SecurID PIN change for, 417
 - group membership, 107
 - locking out, 58
 - logon aliases, 107
 - providing access to restricted agents, 107
 - providing emergency access for, 96, 417
 - transferring to new security domains, 44
 - unlocking, 94, 416
 - with lost, stolen, or expired tokens, 95, 416

utility

- Archive Requests, 265
- Collect Product Information, 268
- Database Storage Management, 277
- Import PIN Unlocking Key, 269
- Manage Backups, 271
- Manage Batchjob, 275
- Manage Database. See Database Storage Management
- Manage Operations Console
 - Administrators, 282
- Manage Secrets, 285
- Node Secret, 112
- Node Secret Load, 112, 113
- Register Custom Extension, 288
- Restore Admin, 235
- Set Trace, 298
- Store, 405
- Update Instance Nodes, 309
- User Groups and Token Bulk Requests, 302
- Verify Archive Log, 307

V

- Velocity syntax rules, 243, 250
- verbose log messages, 196
- Verify Archive Log utility, 307

verify-archive-log command, 307

version number, determining, 14

viewing

- requests, 140
- status, 140

W

- warning log messages, 195
- web agents, 67
- Welcome, what would you like to do?
 - page, 126
- Windows password
 - integrating with RSA SecurID, 61
- Windows requirements, 408
- workflow
 - definition, 442
- workflow definitions
 - configuring, 135
 - described, 134
- workflow participant
 - definition, 443
- workflows, 134, 254

Z

- ZIP file format, 137