

CA SiteMinder®

Agent for SharePoint Guide

12.52 for SharePoint 2010 and 2013



This Documentation, which includes embedded help systems and electronically distributed materials, (hereinafter referred to as the "Documentation") is for your informational purposes only and is subject to change or withdrawal by CA at any time. This Documentation is proprietary information of CA and may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA.

If you are a licensed user of the software product(s) addressed in the Documentation, you may print or otherwise make available a reasonable number of copies of the Documentation for internal use by you and your employees in connection with that software, provided that all CA copyright notices and legends are affixed to each reproduced copy.

The right to print or otherwise make available copies of the Documentation is limited to the period during which the applicable license for such software remains in full force and effect. Should the license terminate for any reason, it is your responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or destroyed.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, LOST INVESTMENT, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED IN ADVANCE OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.

The use of any software product referenced in the Documentation is governed by the applicable license agreement and such license agreement is not modified in any way by the terms of this notice.

The manufacturer of this Documentation is CA.

Provided with "Restricted Rights." Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

Copyright © 2013 CA. All rights reserved. All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

CA Technologies Product References

This document references the following CA Technologies products:

- CA SiteMinder®
- CA DLP Content classification service

Documentation Changes

The following documentation updates have been made since the last release of this documentation:

- [Create a PowerShell Script to Update the Certificates](#) (see page 309)—Fixed error in text of PowerShell script.
- [Update the spsapachessl.properties File](#) (see page 376)—Topics added to describe how to update the spsapachessl.properties file for encrypted and unencrypted private keys. Resolves CQ 174928.
- [Enable SSL for an Encrypted Private Key](#) (see page 236)—Added topic to describe how to enable SSL for an encrypted private key. Resolves CQ 172531.
- [Configure Your CA SiteMinder® Policy Server](#) (see page 39)—Updated content to include information about ODBC databases. Resolves CQ 172529.
- [Create a PowerShell Script to Update the Certificates](#) (see page 309)—Corrected the included PowerShell script. Corrects CQ 175584 and STAR issue 21534651.
- [Grant Application Pool Identities for SharePoint Web Applications Permissions to the Client Certificate](#) (see page 218)—Updated material for better clarity. Resolves CQ 174725.
- [How to Enable SSL for the Agent for SharePoint](#) (see page 196)—Removed the "Modify the ConfigSSL.bat File" topic which was not required. Resolves CQ 172533.
- [Register the Claims Search Service Endpoint on all WFE Servers](#) (see page 220)—Corrected content. Resolves CQ 174721.
- [Export the Client Authentication Certificate from the Current User Certificate Store](#) (see page 215)—Corrected content. Resolves CQ 174734.
- [Update the SSLConfig.properties File](#) (see page 223)—Updated to include relative path to TrustStore.jceks in GenerateSSLConfig command. Resolves CQ 174687.
- [Enable Single Logout by Running the SharePoint Connection Wizard](#) (see page 275)—Corrected content. Resolves CQ 174146.
- [Create a Trusted Store for the Root Certificate Authority Certificate](#) (see page 222)—Replaces "Install the Client Authentication Certificate on Your Agent for SharePoint," which was incorrect. Resolves CQ 174147.

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At <http://ca.com/support>, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at <http://ca.com/docs>.

Contents

Chapter 1: Introduction 17

Purpose and Audience	17
New Architecture to Support SharePoint 2010.....	18
Major Differences between Agent for SharePoint Releases	18
CA SiteMinder® and Microsoft SharePoint	19
CA SiteMinder® Agent for SharePoint Components and Microsoft SharePoint	19
CA SiteMinder® Components used with SharePoint.....	20
Example SharePoint Farm Deployment with Single Web Front End.....	21
Example SharePoint Farm Deployment with Multiple Web Front Ends and Load Balancing	22
Load Balancers and Session Affinity	23

Chapter 2: Federation and Claims-based Authentication 25

Claims-based Authentication Overview	25
Claims	26
Tokens	27
Security Token Service (STS)	27
Identity Provider (IdP)	28
Claims Provider	28
Example Federation and Claims-based Authentication Scenario	28
How the SharePoint Connection Wizard Simplifies Deployment.....	29

Chapter 3: Migrating from SharePoint 2007 to SharePoint 2010 31

Upgrades to the CA SiteMinder® Agent for SharePoint	31
How to Migrate the CA SiteMinder® Agent for SharePoint from SharePoint 2007 to SharePoint 2010	31

Chapter 4: Prerequisites 33

Policy Server Prerequisites.....	33
CA SiteMinder Agent for SharePoint Prerequisites.....	34
Agent for SharePoint Prerequisites for Linux Operating Environments	34
Microsoft Prerequisites.....	37
Verify SharePoint Installation	37

Chapter 5: Configure Your CA SiteMinder® Policy Server 39

How to Configure your CA SiteMinder® Policy Server	39
Open the Administrative UI to Change Policy Server Objects.....	42

Create a Host Configuration Object	42
Configure Clusters.....	43
Create an Agent Object.....	44
(Optional) Create an Agent Group for Multiple Agent Objects.....	45
Create a 4.x Agent Object for the SharePoint Connection Wizard	46
Create an Agent Configuration Object.....	47
Create A User Directory Connection	51
Create Virtual Attribute Mappings for your User Claim.....	52
Create an Authentication Scheme for the Agent for SharePoint.....	57
Determine your policy model	57
Enable Paging for Searches of Active Directory User Stores (32-bit systems).....	70
Enable Paging for Searches of Active Directory User Stores (64-bit systems).....	71

Chapter 6: Request and Install a Policy Server Token Signing Certificate 73

How to Request and Install a Policy Server Token Signing Certificate	73
Token-Signing Certificate Locations in Your SharePoint Environment	76
Create a Certificate Request for a Server Certificate on an IIS Web Server.....	77
Submit Your Certificate Request to a Certificate Authority	78
Approve a Certificate Request using Active Directory Certificate Services	79
Complete Your Certificate Request.....	80
Verify Your Approval and Download Your Certificate and Certificate Chain	81
Export Your Policy Server Signing Certificate	82
Add a Policy Server Signing Certificate to Policy Servers and Create a Trust File	83
Provide the Policy Server Signing Certificate Files to Your Agent Owner	85
Provide the Certificate Files to Your SharePoint Administrator.....	85

Chapter 7: Install and Configure the CA SiteMinder® Agent for SharePoint 87

CA SiteMinder® Agent for SharePoint Configuration Overview.....	87
FIPS Support Overview.....	88
Install the CA SiteMinder® Agent for SharePoint.....	89
Install the CA SiteMinder Agent for SharePoint on Windows.....	90
Install the CA SiteMinder® Agent for SharePoint on UNIX.....	90
How to Configure the CA SiteMinder® Agent for SharePoint	91
Gather CA SiteMinder® Agent for SharePoint Configuration Wizard Information	92
Run the Configuration Wizard.....	94
Confirm that the Agent for SharePoint Is Functioning.....	96
Set a Basic Proxy Rule for the Agent for SharePoint	97
Enable Support for Dynamic Policy Server Clusters for the CA SiteMinder Agent for SharePoint	98
Install the Policy Server Signing Certificate on your CA SiteMinder Agent for SharePoint.....	99
Assign Permissions for Log Files and Directories on UNIX/Linux	99
Manage SharePoint Connections Using the SharePoint Connection Wizard.....	100

Prerequisites for Using the SharePoint Connection Wizard	103
Alternate Connection Wizard Method to Help Resolve Firewall Issues.....	104
SAML Autopost Frequency.....	104
Create a SharePoint Connection	106
How to Start and Stop the Agent for SharePoint	108
Change the Value of the EnableWebAgent Parameter	109
Change the States of the Services on your Agent for SharePoint	110

Chapter 8: Configure SharePoint 113

How to Configure SharePoint for the Agent for SharePoint	113
Permissions Required for Trusted Identity Provider and Claims Provider	114
How to Create Alternate Access Mappings.....	115
Alternate Access Mappings.....	116
Zones and Alternate Access Mappings	117
Obtain the Public and Internal URLs	119
Specify a Public URL for the Web Application.....	121
Specify an Internal URL for the Web Application	122
How to Configure the Trusted Identity Provider.....	123
Copy the Policy Server Signing certificate to the SharePoint Central Administration Server	123
Copy the Powershell Script to the SharePoint Central Administration Server	124
Modify the PowerShell Script.....	125
Add Additional Certificate Authority Certificates to the PowerShell Script.....	131
Run the Powershell Script to Create a Trusted Identity Provider	133
Verify That the Trusted Identity Provider Is Registered.....	134

Chapter 9: Adding Claims to Trusted Identity Providers 135

Verify that your Account has the Required Permissions.....	137
Open a SharePoint 2010 Management Shell Window on your SharePoint Central Administration Server	137
Identify your Trusted Identity Provider.....	137
Add a Claim to your Trusted Identity Provider	138
Verify the New Claim Exists.....	138
Add an Attribute Mapping for the New Claim	139
Update the Affiliate Domain with a Response Attribute	140
Search for and Add Users using the New Claim	142
Removing Claims from Trusted Identity Providers.....	143
Verify that your Account has the Required Permissions.....	144
Open a SharePoint 2010 Management Shell Window on your SharePoint Central Administration Server	144
Identify your Trusted Identity Provider.....	145
Remove the ClaimsMapping Identity from your Trusted Identity Provider.....	145

Remove the Claim Type from your Trusted Identity Provider	146
Update the Trusted Identity Token Issuer	146
Configure the Authentication Providers	146
Modify an Existing Classic Authentication to Claims-based Authentication.....	147
How to Disable Client Loopback.....	149
Disable Client Loopback	150
Add and Grant Permission to CA SiteMinder® Users	150
Manage User Profiles	151

Chapter 10: Features to Set Up Following Basic Installation and Configuration of the Agent for SharePoint 153

Additional SharePoint Configuration Options	153
Create a New Web Application with Claims based Authentication	153
Enable SSL on IIS for the Web Application	154
Enable SSL for the Web Application	155
Office Client Integration	156
How to Configure Office Client Integration for the Agent for SharePoint	157
How to Configure WebDAV to Accomodate Microsoft Hot Fixes 2563214 and 2647954	163
Claims Provider	165
Claims Provider Searches and Results.....	165
Agent for SharePoint Virtual Attribute Mappings.....	166
Install Claims Provider.....	181
How to Configure the Claims Provider.....	182
Extend Web Applications to Different Zones for CRAWL Service and Search Support	188
How to Set Log Files, and Command-line Help to Another Language.....	188
Determine the IANA Code for Your Language	190
Environment Variables.....	190

Chapter 11: Advanced Options 195

How to Enable SSL for the Agent for SharePoint	196
Verify the Prerequisites.....	199
Create the JCEKS Key Store and Private Key	200
Create a Certificate Signing Request and Submit It to a Certificate Authority	202
Generate the Certificates by Processing the Request at the Certificate Authority	204
Download and Import the Certificate Chain	205
Define the KeyStore and the SSL Ports	206
Generate an SSLConfig.properties File.....	207
Restart the Agent for SharePoint.....	207
Add a Trusted Root Authority to your SharePoint Farm	210
Request a Client Certificate.....	211
Generate the Client Authentication Certificate	213

Verify Your Certificate Approval and Install Your Client Authentication Certificate	214
Add the Certificate Snap-ins.....	215
Export the Client Authentication Certificate from the Current User Certificate Store	215
Import the Client Authentication Certificate into the Local Computer Certificate Store	217
Install the Client Authentication Certificate on your SharePoint Servers	218
Grant Application Pool Identities for SharePoint Web Applications Permissions to the Client Certificate.....	218
Register the Claims Search Service Endpoint on all WFE Servers	220
Create a Trusted Store for the Root Certificate Authority Certificate	222
Update the SSLConfig.properties File	223
Restart the Agent for SharePoint.....	223
Modify the SSL Configuration File for Your Agent for SharePoint	226
Generate a Private Key.....	228
Generate and Submit Certificate Signing Requests	232
Download and Install the Certificates from your Certificate Authority	233
Accommodate Your SSL Sites by Modifying the Proxy Rules	234
Enable SSL on Your Agent for SharePoint	235
Run the Connection Wizard	236
Create Alternate Access Mappings for Your Port-Based Virtual Sites.....	239
Modify Your Authentication Scheme.....	240
Restart the Agent for SharePoint.....	240
How to Configure Multiple User Directories.....	243
Open the Administrative UI to Change Policy Server Objects.....	245
Define Virtual Attribute Mappings.....	245
Add Directory Connections	247
Run the SharePoint Connection Wizard.....	248
Remove the Web Applications from the Trusted Identity Provider.....	250
Remove the Trusted Identity Provider.....	251
Copy the Powershell Script to the SharePoint Central Administration Server	251
Determine PowerShell Script Modifications	252
Run the PowerShell Script.....	260
Verify Trusted Identity Provider Registration	261
Disable Client Loopback	261
Add Users to Your Web Applications	262
How to Configure SLO for SharePoint 2013	263
Configure the JSP file on the system that is running your CA SiteMinder Agent for SharePoint.....	266
Edit the File of Each Web Front-End (WFE) Server in Your SharePoint Environment	267
Open the Administrative UI to Change Policy Server Objects.....	269
Enable Single Logout by Running the SharePoint Connection Wizard.....	275
How to Configure Single Logout on SharePoint 2010	278
Verify the Server Hosting Your Agent for SharePoint Has the Proper Files	279
Edit the File of Each Web Front-End (WFE) Server in Your SharePoint Environment	280

Open the Administrative UI to Change Policy Server Objects.....	281
Enable Single Logout by Running the SharePoint Connection Wizard.....	286
How to Monitor Data with CA Introscope.....	289
Configure Your EPAgent.....	291
Modify the server.conf File	293
Open the Administrative UI to Change Policy Server Objects.....	293
Update Your Agent Configuration Object (ACO)	295
Start Your EPAgent on Windows Operating Environments	296
Start Your EPAgent on UNIX Operating Environments	296
Restart the Agent for SharePoint.....	297
How to Use the Session Linker	299
Enable the SessionLinker.....	300
How to Replace the Certificates for your CA SiteMinder® Trusted Identity Provider.....	305
Replace the Certificates on your Servers	307
Verify that your Account has the Required Permissions.....	308
Open a SharePoint 2010 Management Shell Window on your SharePoint Central Administration Server	308
Identify your Trusted Identity Provider.....	308
Create a PowerShell Script to Update the Certificates	309
Add the New Certificates to your CA SiteMinder® Trusted Identity Provider	310
Virtual Hosts with the Agent for SharePoint.....	311
Virtual Host Configurations Supported by the Agent for SharePoint	311
Define Virtual Hosts for each Web Application	312
How to Configure Port-based Virtual Hosts	313
How to Configure Host-Header-Based Virtual Hosts	316
How to Configure Path-based Virtual Hosts	319
How to Modify the Sign-In URL of your CA SiteMinder® Trusted Identity Provider	322
Edit the Sign-In URL for the Affiliate Domain using the Sharepoint Connection Wizard	323
Verify that your Account has the Required Permissions.....	325
Open a SharePoint 2010 Management Shell Window on your SharePoint Central Administration Server	325
Identify your Trusted Identity Provider.....	325
Change the Sign-in URL of your CA SiteMinder® Trusted Identity Provider	326
Verify that the Sign-in URL has Changed.....	326
Configure the Agent for SharePoint for Web Applications That Use NTLM Authentication.....	327
How to replace the (WS-Fed) token signing certificates.....	328
Remove the expired certificate from your IIS web server	329
Add the new certificates	329
Install the Policy Server Signing Certificate on your CA SiteMinder Agent for SharePoint.....	330
Re-configure the trusted identity provider	330

Chapter 12: CA DLP Content Classification Service and the CA SiteMinder Agent for SharePoint **333**

Enable the DLP Plug-in.....	333
Set the Proxy Rules for the Agent for SharePoint when using CA DLP Content Classification Service with Multiple Authentication	335

Chapter 13: Log Files **339**

logger.properties File Overview	339
Modifying the logger.properties File.....	339
Logging Settings	340
SvrConsoleAppender Settings.....	340
SvrFileAppender Settings	341
Server Log Settings.....	341
Server Log Rolling Settings	342

Chapter 14: Troubleshooting **345**

Attributes Appear Truncated in SharePoint	346
Log Files Show Access Denied Due to BadURLChars Settings	347
Log Files Show Access Denied Because of SPAuthorizeUserAgent Settings	349
Enable Search of Custom Object Classes in Your LDAP Directory	350
REST API in Excel Services Does Not Work Due to CSSChecking ACO Parameter	351
Cannot Log Off Users from Sites and Subsites without Referring LogOffURI ACO (CQ 135854).....	352
Enable Paging for Searches of Active Directory User Stores (32-bit systems)	353
Enable Paging for Searches of Active Directory User Stores (64-bit systems)	354
Users Cannot Access Office Applications in Internet Explorer 7 when Office Client Integration Is Enabled	355
I Can Only View Read Only Copies of Documents on SharePoint Sites, but the Office Client Integration Is Enabled.....	357
SharePoint FedAuth Cookies and Office Client Integration Behavior	358
Registration Failed with Unknown Error 127	358
How to Reduce People Picker Timeouts with Large Databases on UNIX/Linux Operating Environments.....	359
Open the sm.registry File with a Text Editor	359
Add the Setting in the Registry File	360
Restart Your Policy Server.....	361
How to Reduce People Picker Timeouts with Large Databases on Windows Operating Environments.....	362
Open the Windows Registry Editor	362
Add the Registry Key	363
Restart Your Policy Server.....	364
Non-english Input Characters Contain Junk Characters.....	364

Appendix A: SessionLinker Reference 365

How the SessionLinker Works.....	365
What the SessionLinker Does Not Support.....	366
Working with Cookies	367
Single Session Cookie Enforcement	367
Enable Wildcard Cookie Names	368
Maintain Links to Multiple Cookies.....	368
Troubleshooting	369

Chapter 15: Upgrade Your CA SiteMinder Agent for SharePoint 371

How to Upgrade Your CA SiteMinder Agent for SharePoint	371
Install the New Version of the Agent	372
Install the New Version of the Claims Provider on Your SharePoint Central Administration Servers	374
Confirm the Name of Your Existing Trusted Identity Provider.....	375
Associate Your Existing Trusted Identity Provider with the Upgraded Claims Provider	375
Update the spsapachessl.properties File	376

Appendix B: Proxy Rule Configuration Reference 379

Proxy Rules Overview.....	379
Planning Routes for Incoming Requests	380
Proxy Rules Terminology.....	382
Establish a Proxy Rules Configuration File.....	383
Proxy Rules DTD	384
nete:proxyrules	384
nete:case.....	386
nete:cond.....	388
nete:default.....	390
nete:forward	391
nete:redirect	392
nete:local.....	393
nete:xprcond	393
How nete:xprcond Elements Works.....	395
Regular Expression Syntax.....	396
Regular Expression Examples in nete:rule and nete:result.....	399
Header Values in Forwards, Redirects, and Results Filters	400
Dynamic Header Value in a nete:forward.....	400
Dynamic Header Value in a nete:redirect.....	401
Dynamic Header Value in a nete:result	401
Response Handling.....	401
Modify Proxy Rules.....	402

Sample Proxy Rules Configuration Files	402
Proxy Rules Example—Routing Requests by Virtual Host.....	403
Proxy Rules Example—Routing Requests by Header Value	403
Proxy Rules Example—Routing Requests by Device Type	404
Proxy Rules Example—Routing Requests with URIs	405
Proxy Rules Example—Routing Requests by File Extension.....	405
Proxy Rules Example—Routing Requests with Nested Conditions.....	406
Proxy Rules Example—Using Regular Expression in Proxy Rules.....	407
Proxy Rules Example—Routing Requests by Cookie Existence.....	408
Proxy Rules Example—Routing Requests by Cookie Value.....	408
 Chapter 16: Remove CA SiteMinder® Agent for SharePoint	409
How to Remove the CA SiteMinder® Agent for SharePoint.....	409
Remove Claims Provider	409
Delete a SharePoint Connection	410
Remove the Trusted Identity Provider from any Web Applications Using it	412
Remove Trusted Identity Provider	413
Remove the Agent for SharePoint from Windows.....	413
Remove the Agent for SharePoint from UNIX.....	414
(Optional) Delete Policy Store Objects.....	414
 Appendix C: Agent for SharePoint Worksheets	417
CA SiteMinder Agent for SharePoint Configuration Wizard Information Worksheet.....	417
SharePoint Connection Wizard Information Worksheet.....	418
SharePoint 2010 Federation Worksheet.....	419
 Chapter 17: Platform Support and Installation Media	419
Locate the CA SiteMinder® Agent for SharePoint Platform Support Matrix.....	420
Locate the Bookshelf	420
Locate the Installation Media.....	421
 Index	423

Chapter 1: Introduction

This section contains the following topics:

[Purpose and Audience](#) (see page 17)

[New Architecture to Support SharePoint 2010](#) (see page 18)

[Major Differences between Agent for SharePoint Releases](#) (see page 18)

[CA SiteMinder® and Microsoft SharePoint](#) (see page 19)

[Example SharePoint Farm Deployment with Single Web Front End](#) (see page 21)

[Example SharePoint Farm Deployment with Multiple Web Front Ends and Load Balancing](#) (see page 22)

[Load Balancers and Session Affinity](#) (see page 23)

Purpose and Audience

The CA SiteMinder® Agent for SharePoint is a gateway or a proxy server-based solution that lets you protect resources in your Microsoft SharePoint environment with CA SiteMinder®.

This guide describes how to install and configure the CA SiteMinder® Agent for SharePoint so you can protect resources stored on SharePoint. This guide is intended for the following CA SiteMinder® and SharePoint personnel:

- SharePoint Administrators
- CA SiteMinder® Administrators

This guide assumes that SharePoint administrators can perform the following tasks:

- Create a SharePoint web application
- Add SharePoint web applications to site collections
- Manage SharePoint site collection administrators
- Work with web application access policies in SharePoint
- Add, modify, or remove files or other content to a SharePoint web application
- Manage SharePoint users and user profiles

This guide assumes that CA SiteMinder® administrators can perform the following tasks:

- Install and configure CA SiteMinder® Agent for SharePoint and Policy Servers
- Create CA SiteMinder® policies, realms, rules, and responses to protect resources
- Manage CA SiteMinder® user directories

New Architecture to Support SharePoint 2010

The CA SiteMinder® Agent for SharePoint 2010 features a new architecture designed to protect your SharePoint 2010 resources. This new architecture is based on industry standards and uses a proxy model to streamline enterprise deployments of the Agent for SharePoint, while supporting future growth.

This agent also includes a new SharePoint connection wizard which simplifies the process of creating connections between your SiteMinder objects and SharePoint resources. This wizard creates the CA SiteMinder® objects you need on the Policy Server and generates a PowerShell script that properly configures your SharePoint central administration server.

Major Differences between Agent for SharePoint Releases

The following table describes the major differences between the Agent for SharePoint releases:

Agent for SharePoint 2007	Agent for SharePoint 2010
Required installation of the following on each SharePoint 2007 server: <ul style="list-style-type: none">■ A CA SiteMinder® Web Agent■ A CA SiteMinder® Agent for SharePoint	Deployed as a proxy-server based solution in front of SharePoint 2010 for more centralized configuration and management.
Used one of two SharePoint 2007 authentication methods: <ul style="list-style-type: none">■ Windows Impersonation■ ASP.NET Forms-based authentication (FBA)	Uses the new SharePoint 2010 claims-based authentication option, which is based on industry-standard protocols (WS-Federation / SAML 1.1).
Used a CA SiteMinder® Management UI, installed into SharePoint, to configure protection of SharePoint resources. Included a Role and Membership Provider to facilitate People Picker access to SiteMinder user directories.	Configuration and administration enhancements include: <ul style="list-style-type: none">■ New Connection Wizard to automate the configuration of required SiteMinder objects and simplify the creation of a Trusted Identity Provider inside SharePoint 2010.■ Farm-wide configuration of various aspects of the SiteMinder integration using the new SharePoint 2010 PowerShell interface.■ Improved People Picker usability through a new Claims Provider component.

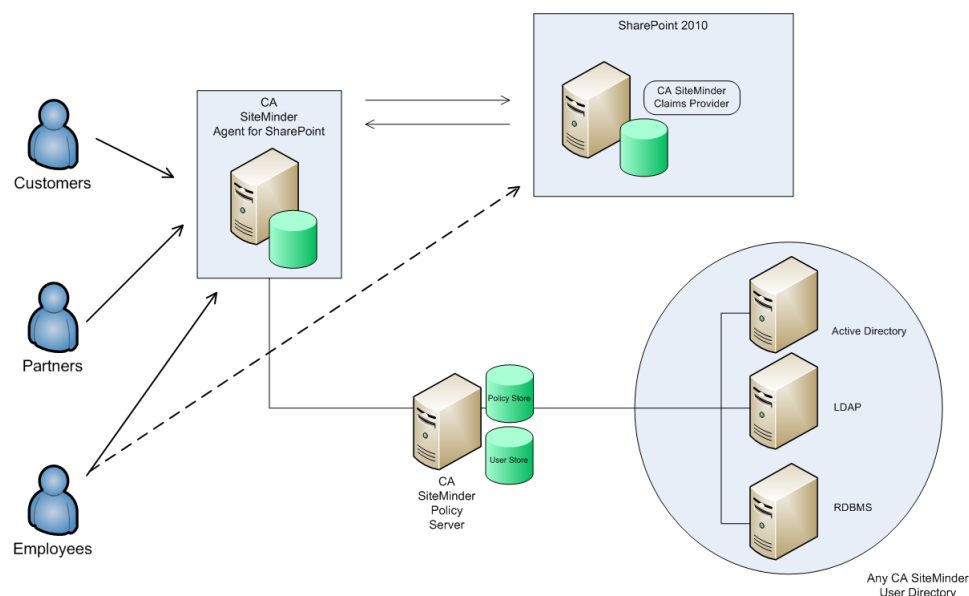
CA SiteMinder® and Microsoft SharePoint

The CA SiteMinder® Agent for SharePoint integrates Microsoft SharePoint 2010 into the SiteMinder web access management environment.

An access control solution uses policy decision points and policy enforcement points. The CA SiteMinder® Agent for SharePoint uses a gateway or proxy server policy enforcement point to protect resources in a Microsoft SharePoint environment. In the network topology, these enforcement points are physically placed between the user and the resource on SharePoint server.

CA SiteMinder® Agent for SharePoint Components and Microsoft SharePoint

The following illustration shows the relationship between the CA SiteMinder® components and the SharePoint server.



In the previous illustration, customers, partners, and employees request resources from SharePoint. The requests must pass through the CA SiteMinder® Agent for SharePoint. The agent provides authentication, policy enforcement, and federated single sign-on capabilities. The CA SiteMinder® Policy Server acts as the policy decision point for authentication. The CA SiteMinder® Policy Store which is connected to the Policy Server stores policies and other configuration objects. This solution enables external users to access protected SharePoint resources and internal users to access SharePoint resources.

CA SiteMinder® Components used with SharePoint

The CA SiteMinder® Agent for SharePoint solution contains the following SiteMinder components in a specific configuration designed to protect SharePoint resources.

Policy Server

The Policy Server acts as the Policy Decision Point (PDP). The Policy Server evaluates and enforces access control policies, for requests made to resources protected by agents, such as the CA SiteMinder® Agent for SharePoint.

Agent for SharePoint

The Agent for SharePoint is a stand-alone server that provides a proxy-based solution for access control. The agent acts as the policy enforcement point (PEP), standing in the network topology physically between the user and the resource on the SharePoint server.

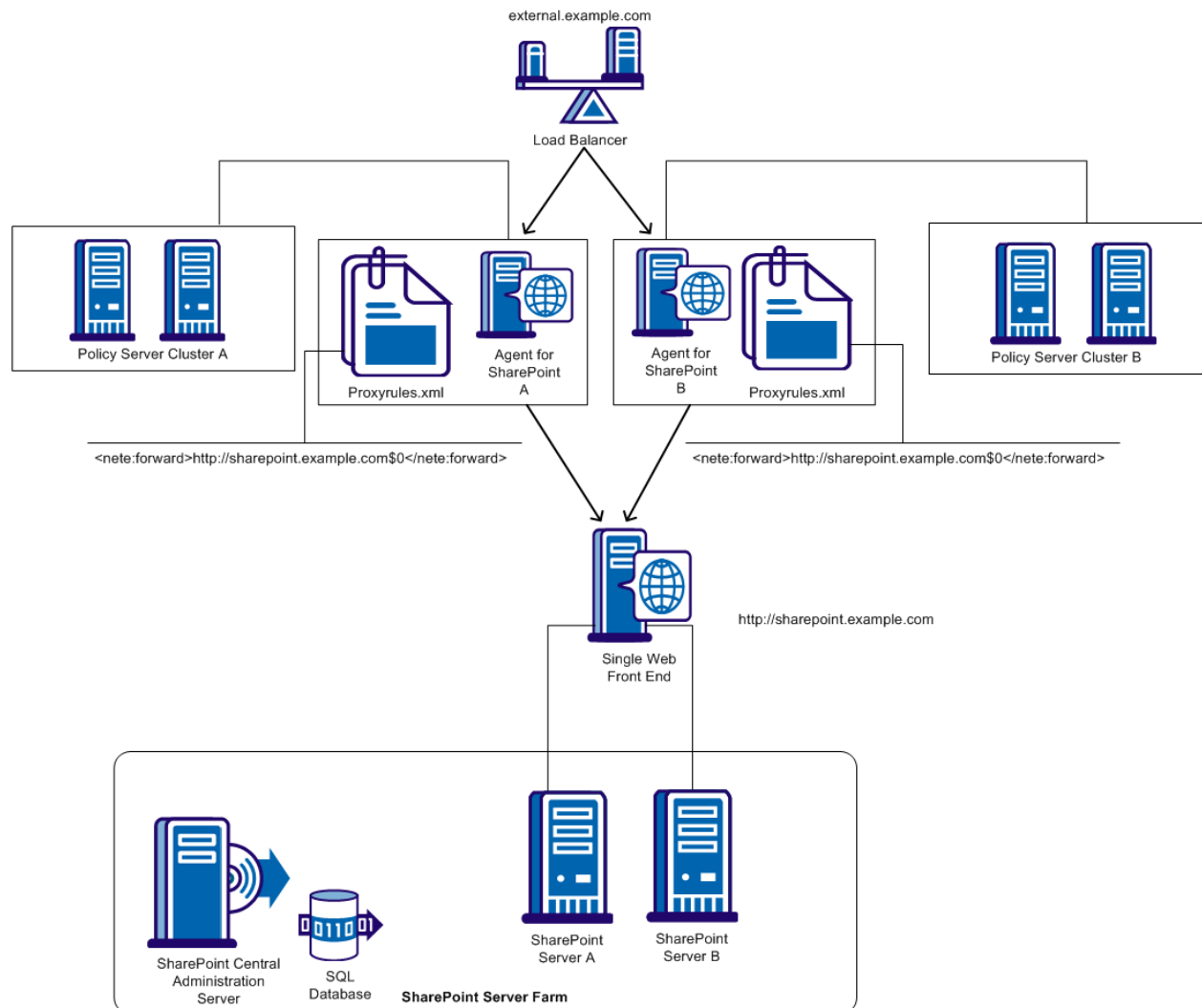
Claims Provider

The CA SiteMinder® Claims Provider is used for configuring particular claim values to grant permissions to SharePoint resources. The Claims Provider is packaged as a SharePoint solution (WSP file) with its feature receiver.

Note: Upgrade any SiteMinder components in your environment that do not meet the minimum versions.

Example SharePoint Farm Deployment with Single Web Front End

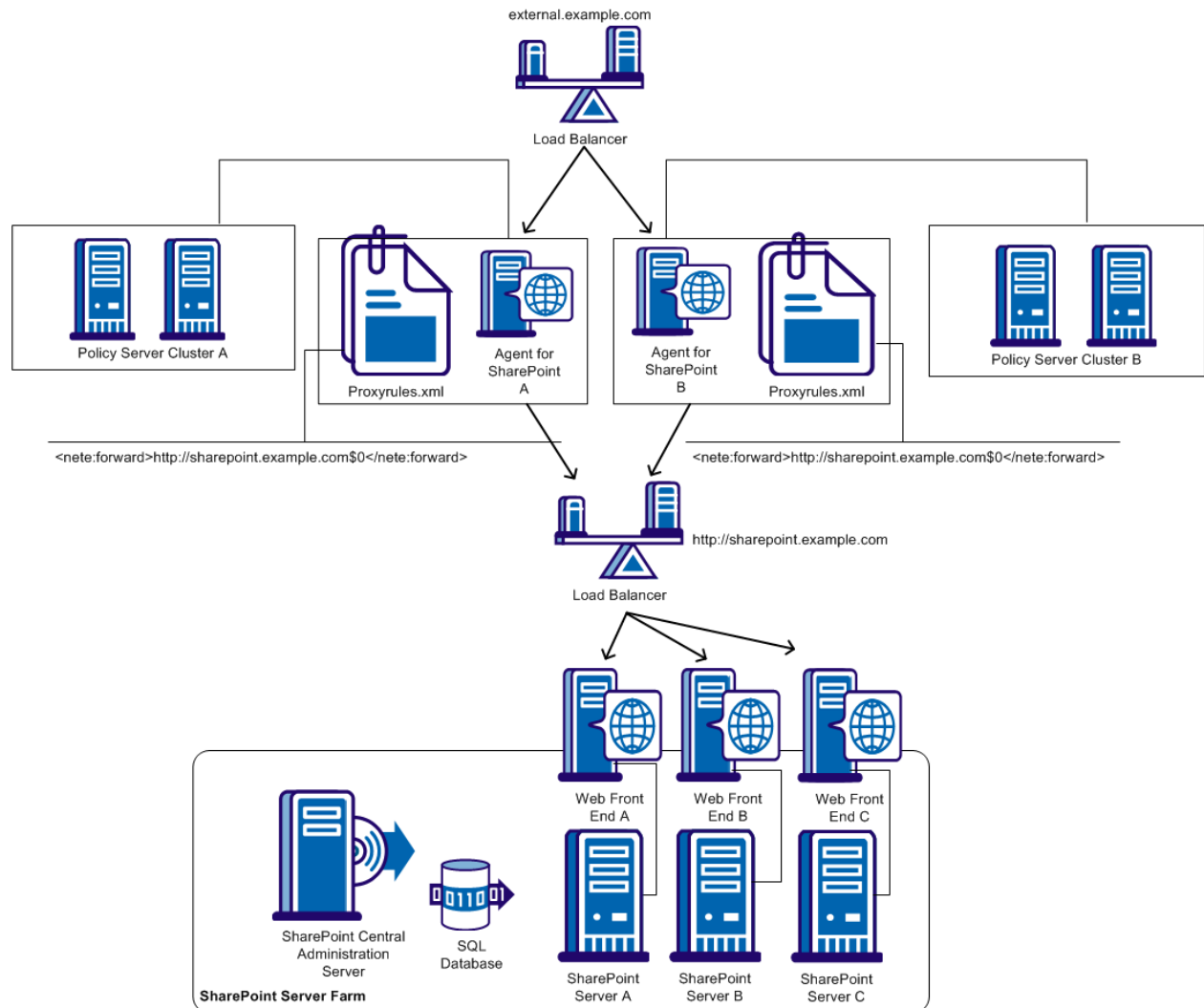
If the servers in your SharePoint farm are associated with a single web front end (WFE) server, the following illustration provides one possible deployment scenario:



In the previous example, your setting in the proxyrules.xml file is
`<nete:forward>http://sharepoint.example.com$0</nete:forward>`

Example SharePoint Farm Deployment with Multiple Web Front Ends and Load Balancing

If your SharePoint farm has servers associated with a multiple web front end (WFE) servers, the following illustration provides one possible deployment scenario:



In the previous example, your setting in the proxyrules.xml file is `<nete:forward>http://sharepoint.example.com$0</nete:forward>`

Load Balancers and Session Affinity

Load balancers that use session affinity dynamically select the best-performing server to which to send requests when establishing a session. The load balancers send subsequent requests for the same session back to the same server.

Configuring session affinity helps your load balancers operate more efficiently because the CA SiteMinder® caches are used to their full potential. For example, sessions are stored in the Web Agent cache when they are created. Since the session is cached, subsequent requests for resources during the same session are validated using the information from the Web Agent cache. The Policy Server is not contacted, and efficiency is increased.

Chapter 2: Federation and Claims-based Authentication

Enterprise applications and services are increasingly distributed across organizations. They have customers and partners who reside outside of the enterprise that need access to SharePoint applications within the enterprise. As a result, the need for secure but seamless access to SharePoint resources has increased.

CA SiteMinder® Agent for SharePoint lets you protect your SharePoint resources using CA SiteMinder® web access management capabilities. The federation capabilities allow partnering organizations to trust and share digital identities and attributes of employees, customers, and suppliers across trust domains. These trust domains can exist within one organization or between different organizations.

These federation capabilities also provide single sign-on across partner sites. The Agent for SharePoint provides a custom CA SiteMinder® solution which issues claims and packages claims into security tokens, used to validate and access SharePoint resources.

The following section gives an overview about federation and claims-based authentication used in this solution.

Claims-based Authentication Overview

Claims-based authentication enables applications to authenticate users with the minimum required information. Claims-based authentication allows applications to verify and validate user claims.

The following list explains the fundamental concepts of Claims-based authentication:

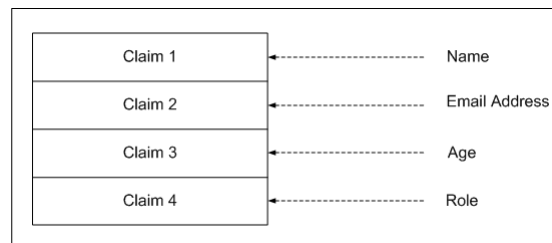
- Claims
- Tokens
- Security Token Service (STS)
- Identity Provider (IdP)
- Claims Provider

Claims

Claims represent any identity information about a user. In some instances, the user can be an application or a computer. A claim enables the user to gain access to multiple resources, such as applications and network resources, without entering credentials multiple times.

A claim is a statement about a user (for example, a name). The bits of identity information include, name, e-mail address, age, or organizational roles and responsibilities. A claim can also include the right of a user to perform something like access a file. Claims can also contain a restrictive right like the financial limit of a user.

A claim is given one or more values and then packaged in security tokens issued by a security token service (STS).



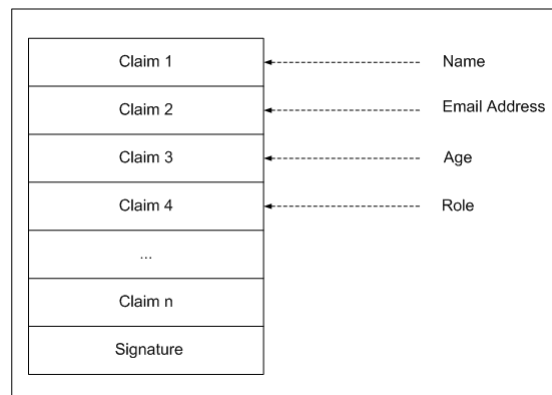
The previous illustration represents a claims token. The illustration shows claim values inside the token.

Tokens

Claims information is transferred in security tokens. Each token contains a set of one or more claims, and contains information about the user to whom this token applies. A security token service (STS) issues the token.

Tokens can be issued in different formats, such as Security Assertion Markup Language (SAML) tokens or WS-Federation (WS-FED) tokens. Security tokens can be signed with an X.509 certificate to protect the contents of the token in transit. The application that receives the token validates it before using the claims.

The Agent for SharePoint uses WS-FED tokens and X.509 certificates to protect its content.



The previous illustration represents a security token. This token contains claim values and a digital signature.

Security Token Service (STS)

The STS (Security Token Service) is a web service that issues, manages, and validates security tokens. STS makes assertions based on the evidence that it trusts, whoever trusts it.

Identity Provider (IdP)

An identity provider is a system that creates, maintains, and manages identity information and asserts identities to other service providers within a federation. For example, a user Adam, has an email address of adam@example.com and authenticated to this domain using a password mechanism.

An identity provider is also known as a SAML authority, asserting party, trusted identity provider, or source site, and is often abbreviated as IdP.

In the CA SiteMinder® Agent for SharePoint solution, the Agent for SharePoint is the IdP STS. The identity provider owns the STS and affirms the tokens created by the STS.

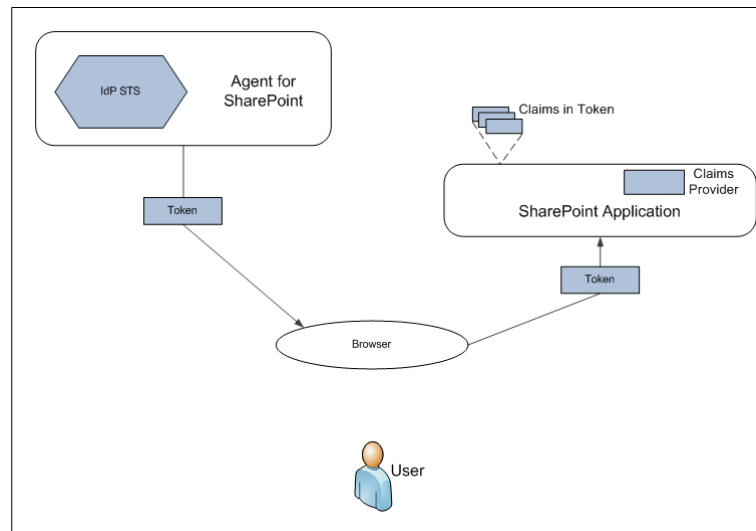
Claims Provider

A CA SiteMinder® claims provider uses virtual attribute mappings in your CA SiteMinder® directories to support searches of your CA SiteMinder® users with the SharePoint people picker.

The claims provider finds and selects user, group, and role-based claim values.

Example Federation and Claims-based Authentication Scenario

The following illustration provides a possible federation and claims-based authentication scenario.



In the illustration, a SharePoint application works on behalf of a user, such as a web browser or another client. This SharePoint application asks an IdP-STS (Agent for SharePoint) for a token containing claims for this user. An HTTP protocol makes the request, the IdP-STS authenticates the user in some way, such as verifying the password of the user. Therefore, the IdP-STS can be certain that the user is authentic.

The request sent to an IdP-STS typically contains a URI identifying the SharePoint application this user wishes to access. The IdP-STS asserts the identity of the user and the application. Once the STS finds account information and other attributes about the user and the application, it generates the token and returns it to the browser.

How the SharePoint Connection Wizard Simplifies Deployment

This release of the Agent for SharePoint includes a connection wizard that automatically creates the Federation objects it requires on your CA SiteMinder Policy Server. The connection wizard also creates a PowerShell script that you modify and run on your SharePoint central administration server. This PowerShell script creates the Trusted Identity provider (IdP).

Chapter 3: Migrating from SharePoint 2007 to SharePoint 2010

This section contains the following topics:

[Upgrades to the CA SiteMinder® Agent for SharePoint](#) (see page 31)

[How to Migrate the CA SiteMinder® Agent for SharePoint from SharePoint 2007 to SharePoint 2010](#) (see page 31)

Upgrades to the CA SiteMinder® Agent for SharePoint

No direct upgrade path exists for moving from the CA SiteMinder® Agent for SharePoint with SharePoint 2007 to the CA SiteMinder® Agent for SharePoint with SharePoint 2010. As discussed in the Overview chapter, the agent used with SharePoint 2010 uses a claims-based authentication model supported by SharePoint 2010. The previous agent for use with SharePoint 2007 used different authentication models.

How to Migrate the CA SiteMinder® Agent for SharePoint from SharePoint 2007 to SharePoint 2010

Your organization decides to move from a SharePoint 2007 environment to a SharePoint 2010 environment. You can repurpose your SharePoint 2007 hardware when you migrate to SharePoint 2010. If you decide to do so, consider the steps outlined in this topic.

To migrate the CA SiteMinder® Agent for SharePoint from SharePoint 2007 to SharePoint 2010 complete, the following steps:

1. Uninstall the CA SiteMinder® Agent for SharePoint running on the SharePoint 2007 system. The uninstall procedure is documented in the CA SiteMinder Agent for Microsoft SharePoint Guide r12.0.
2. Upgrade the SharePoint environment to SharePoint 2010.
3. Install the CA SiteMinder® Agent for SharePoint 12.52 as described in this guide.

Note: You may be able to reuse some of the sites from your SharePoint 2007 deployment in your SharePoint 2010 deployment. See Microsoft.com and SharePoint 2010 documentation for SharePoint migration recommendations, including the migration of existing user identities to the claims format.

Chapter 4: Prerequisites

This section contains the following topics:

[Policy Server Prerequisites](#) (see page 33)

[CA SiteMinder Agent for SharePoint Prerequisites](#) (see page 34)

[Microsoft Prerequisites](#) (see page 37)

Policy Server Prerequisites

The CA SiteMinder® Policy Server requires the following prerequisites to operate with the Agent for SharePoint:

- CA SiteMinder® Policy Server 12.5, 12.0 SP3 CR05 NIN Build 443 and above.
Important! Multiple directory connections are supported with Policy Server version 12.5 and above only.
- CA SiteMinder® Administrative UI 12.5 (any CR), 12.0 SP3 CR05 NIN Build 443 and above.
- (For Office Client Integration) HTTP methods for WebDAV defined in the CA SiteMinder® Agent type
- An SSL Certificate
- The following open ports:
 - Ports for accounting, authentication, and authorization requests (44441, 44442, 44443 respectively)
 - Port for Connection Wizard (44444)
 - Ports for directory server connections.

CA SiteMinder Agent for SharePoint Prerequisites

Release 12.52 of the CA SiteMinder Agent for SharePoint is the minimum version required.

The CA SiteMinder Agent for SharePoint also requires the following:

- A 32-bit Java Development Kit version 1.6.0_16 or higher is required on the CA SiteMinder Agent for SharePoint system.

Important! The CA SiteMinder Agent for SharePoint cannot be installed on a computer that hosts any other web server. The CA SiteMinder Agent for SharePoint operates as a stand-alone proxy-based solution.

- Open the following ports on the CA SiteMinder Agent for SharePoint:
 - Port 8009 (ajp13)
 - Port 8005 (Tomcat shutdown)
 - Port for HTTP requests on the embedded Apache web server
 - Port for HTTPS requests on the embedded Apache web server
 - Port for HTTP requests by the Claims search service
 - Port for HTTPS requests by the Claims search service

Agent for SharePoint Prerequisites for Linux Operating Environments

If you want to install your Agent for SharePoint on a Linux operating environment, verify that your computer meets the following prerequisites:

- [Required Linux patches](#) (see page 34).
- Required Linux libraries.
- [Required Linux tools](#) (see page 36).

More information:

[Registration Failed with Unknown Error 127](#) (see page 358)

Required Linux Patches

The following Linux patches are required:

For Web Agents running on 64-bit Linux systems

- Binutils 2.17
- GCC 4.1.0

Required Linux Libraries

Certain library files are required for components operating on Linux operating environments. Failure to install the correct libraries can cause the following error:

```
java.lang.UnsatisfiedLinkError
```

If you are installing, configuring, or upgrading a Linux version of this component, the following libraries are required on the host system:

Red Hat 5.x:

```
compat-gcc-34-c++-3.4.6-patch_version.i386
```

```
libstdc++-4.x.x-el5.i686.rpm
```

Red Hat 6.x:

libstdc++-4.x.x-x.el6.i686.rpm

Additionally, for Red Hat 6.x (64-bit):

Note: All the RPM packages that are required for 64-bit Red Hat 6.x are *32-bit* packages.

libXau-1.0.5-1.el6.i686.rpm

libxcb-1.5-1.el6.i686.rpm

compat-db42-4.2.52-15.el6.i686.rpm

compat-db43-4.3.29-15.el6.i686.rpm

libX11-1.3-2.el6.i686.rpm

libXrender-0.9.5-1.el6.i686.rpm

libexpat.so.1 (provided by expat-2.0.1-11.el6_2.i686.rpm)

libfreetype.so.6 (provided by freetype-2.3.11-6.el6_2.9.i686.rpm)

libfontconfig.so.1 (provided by fontconfig-2.8.0-3.el6.i686.rpm)

libICE-1.0.6-1.el6.i686.rpm

libuuid-2.17.2-12.7.el6.i686.rpm

libSM-1.1.0-7.1.el6.i686.rpm

libXext-1.1-3.el6.i686.rpm

compat-libstdc++-33-3.2.3-69.el6.i686.rpm

compat-db-4.6.21-15.el6.i686.rpm

libXi-1.3-3.el6.i686.rpm

libXtst-1.0.99.2-3.el6.i686.rpm

libXft-2.1.13-4.1.el6.i686.rpm

libXt-1.0.7-1.el6.i686.rpm

libXp-1.0.0-15.1.el6.i686.rpm

Linux Tools Required

Before installing a CA SiteMinder® Agent on a Red Hat Apache 2.2 web server running on the Red Hat Enterprise Linux operating environment, install all the items included in the Red Hat Legacy Software Development tools package.

Microsoft Prerequisites

The CA SiteMinder® Agent for SharePoint is designed for Microsoft SharePoint 2010.

Verify that your SharePoint servers have the following prerequisites:

- (For Office Client Integration) use Office 2007 SP2 or higher.
- Open Ports for your SharePoint resources (set during SharePoint installation or configuration)

Note: For more information about specific patches or service packs, and the latest version information, see the Platform Support Matrix.

More information:

[Locate the CA SiteMinder® Agent for SharePoint Platform Support Matrix](#) (see page 420)

Verify SharePoint Installation

Use the following process to verify that SharePoint is installed correctly before configuring SharePoint with the CA SiteMinder® Agent for SharePoint.

Follow these steps:

1. Log on to SharePoint 2010 Central Administration and create a SharePoint site with any template.

Note: Verify that the Windows user has administrator privileges.

2. Log on to the newly created SharePoint site.
3. Perform various actions like uploading documents and adding contacts.

Chapter 5: Configure Your CA SiteMinder® Policy Server

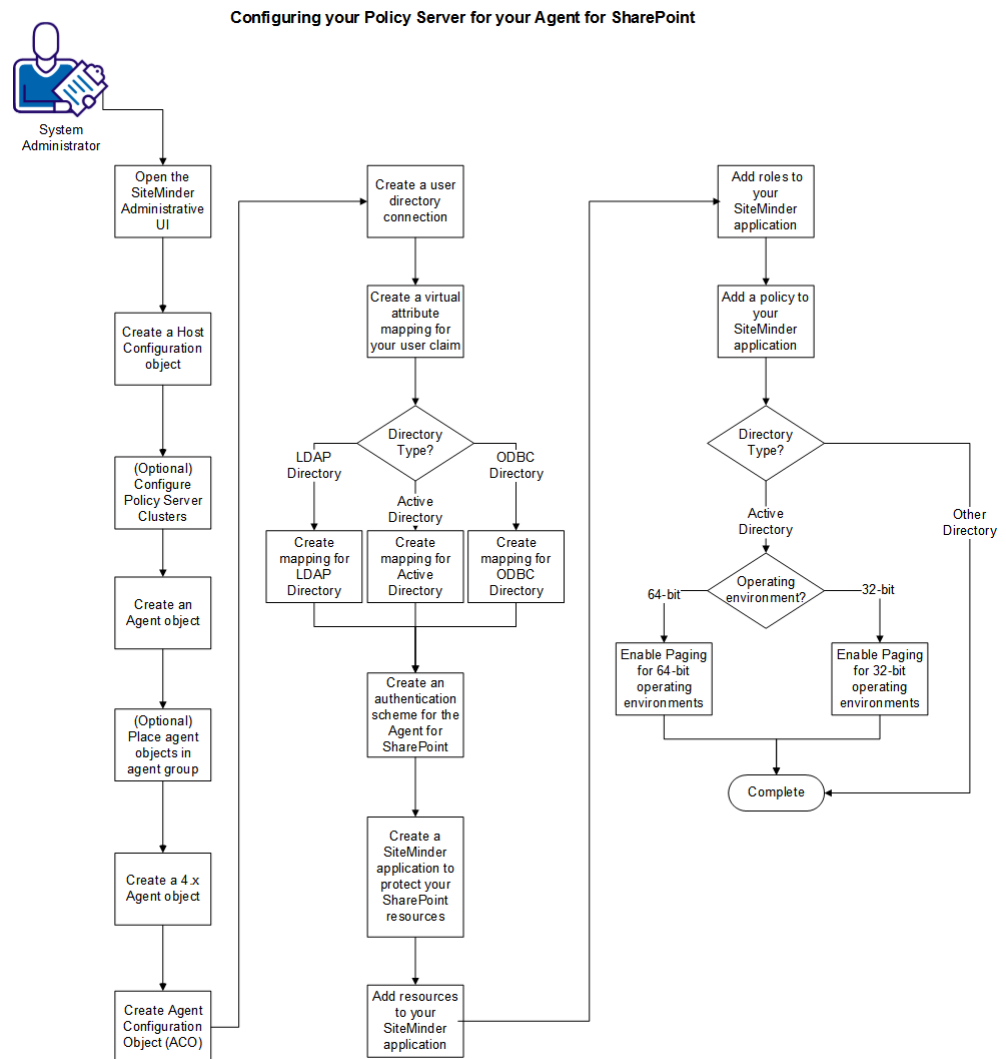
This section contains the following topics:

[How to Configure your CA SiteMinder® Policy Server](#) (see page 39)

How to Configure your CA SiteMinder® Policy Server

The Policy Server authenticates and authorizes users who request access to the resources in your SharePoint environment. The Policy Server stores items that you create to define the users in your SharePoint environment and the resources that you want to protect with CA SiteMinder®.

The following illustration describes the configuration process that prepares your Policy Server for use with the Agent for SharePoint:



Follow these steps:

1. [Open the CA SiteMinder® Administrative UI](#) (see page 42).
2. [Create a host configuration object](#) (see page 42).
3. [\(Optional\) Configure Policy Server clusters](#) (see page 43).
4. [Create an Agent Object](#) (see page 44).
5. [\(Optional\) Create agent groups for multiple agent objects](#) (see page 45).
6. [Create a 4.x agent object for the SharePoint Connection wizard](#) (see page 46).
7. [Create an Agent Configuration Object](#) (see page 47).
8. [Create a user directory connection](#) (see page 51).
9. [Create virtual attribute mappings for your user claim](#) (see page 52).
10. [Create an authentication scheme for the Agent for SharePoint](#) (see page 57).
11. [Determine your policy model](#) (see page 57), and then do *one* of the following steps:
 - [Create a Policy Domain](#) (see page 58).
 - [Create a CA SiteMinder® application to protect your SharePoint resources](#) (see page 64).
12. For Active Directory user directories only, enable paging on the system hosting your Policy Server. Use the appropriate procedure for your operating environment:
 - [Enable paging on 32-bit operating environments](#) (see page 70).
 - [Enable paging on 64-bit operating environments](#) (see page 71).

Open the Administrative UI to Change Policy Server Objects

Change the objects on your Policy Server by opening the Administrative UI.

Follow these steps:

1. Open the following URL in a browser.

`https://host_name:8443/iam/siteminder/adminui`

host_name

Specifies the fully qualified Administrative UI host system name.

2. Enter your CA SiteMinder® superuser name in the User Name field.
3. Enter the CA SiteMinder® superuser account password in the Password field.
Note: If your superuser account password contains dollar-sign (\$) characters, replace each instance of the dollar-sign character with \$DOLLAR\$. For example, if the CA SiteMinder® superuser account password is \$password, enter \$DOLLAR\$password in the Password field.
4. Verify that the proper server name or IP address appears in the Server drop-down list.
5. Select Log In.

Create a Host Configuration Object

You can create a new Host Configuration object or duplicate an existing object.

To create a host configuration object

1. Click Infrastructure, Hosts.
2. Click Host Configuration Objects.
The Host Configuration Objects page appears.
3. Click Create Host Configuration.

4. Do one of the following:
 - (Recommended) Create a copy of an existing Host Configuration object and modify its properties. You can copy the DefaultHostSettings object and use its settings as a template for the new object. The Policy Server installation program installs the DefaultHostSettings object.
Important! Do not directly modify and use the DefaultHostSettings object. Always copy this object and then modify it.
 - Create a new object.
5. Click OK.

The Create Host Configuration page appears.

Note: Click Help for descriptions of settings and controls, including their respective requirements and limits.
6. Type the name and a description.
7. In Configuration Values, specify the Host Configuration settings.
8. Click Submit.

The Host Configuration Object is created.

Configure Clusters

Policy Server clusters are defined as part of a Host Configuration Object. When a CA SiteMinder® agent initializes, the settings from the Host Configuration Object are used to setup communication with Policy Servers.

Note: For more information about Host Configuration Objects, see the *Web Agent Configuration Guide* and the *Policy Server Configuration Guide*.

Follow these steps:

1. Select the Infrastructure, Hosts. Host Configuration Objects.
2. Click Create Host Configuration.
3. In the Clusters section, click Add.

The Cluster Setup section opens.

Note: You can click Help for a description of fields, controls, and their respective requirements.
4. Enter the IP address and the port number of the Policy Server in the Host and Port fields respectively.
5. Click Add to Cluster.

The Policy Server appears in the servers list in the Current Setup section.

6. Repeat these steps to add other Policy Servers to the cluster.
7. Click OK to save your changes.

Your return to the Host Configuration dialog The Policy Server cluster is listed in a table.

8. In the Failover Threshold Percent field, enter a percentage of the number of Policy Servers that must be active and click Apply.

If the percentage of active servers in the cluster falls below the percentage you specify, the cluster fails over to the next available cluster in the list of clusters. This setting applies to all clusters that use the Host Configuration Object.

Important! The Policy Server specified in the Configuration Values section is overwritten by the Policy Servers specified in a cluster. This Policy Server is no longer used because a cluster is configured. For the value of the Policy Server parameter in the Configuration Values section to apply, do not specify any Policy Servers in a cluster. If clusters are configured, and you decide to remove the clusters in favor of a simple failover configuration delete all Policy Server information from the cluster.

9. Click Submit to save your changes.

Create an Agent Object

Agent act as policy-enforcement points (PEPs), by intercepting user requests for SharePoint resources and communicating with the Policy Server. Agent objects associate the protected resources on your SharePoint servers with the CA SiteMinder® policies that protect those resources.

Follow these steps:

1. Click Infrastructure, Agent, Agents.
2. Click Create Agent.

The Create Agent screen appears.

3. Click OK.

The Create Agent: screen appears.

4. Enter a distinctive name and description.
5. Verify that the CA SiteMinder® option button is selected and that Web Agent appears in the Agent Type drop-down list.
6. Click Submit.

The agent object is created and a confirmation screen appears.

(Optional) Create an Agent Group for Multiple Agent Objects

If you have multiple Agent Objects in your CA SiteMinder® environment, you can place them in agent groups. Agent groups make managing large numbers of agent objects easier.

Follow these steps:

Note: If you are an experienced CA SiteMinder® user, you can add your agent objects to an existing agent group instead of creating a group.

1. Click Infrastructure, Agent.
2. Click Agent Groups.
The Agent Groups page appears.
3. Click Create an agent group.
The Create agent group screen appears.
4. Click Create a new object of type Agent Group, and then click OK.
The Create Agent Group: screen appears.
5. Enter a distinctive name and description.
6. Verify that the CA SiteMinder® option button is selected and that Web Agent appears in the Agent Type drop-down list.
7. Click Add/Remove.
The Agent Group members screen appears.
8. Click the arrows to move the agent objects you want into the selected members column, and then click OK.
The Create Agent Group screen reappears. The agent objects in the group appear in the Group Members list.
9. Click Submit.
The agent group is created and a confirmation screen appears.

Create a 4.x Agent Object for the SharePoint Connection Wizard

The SharePoint connection wizard requires an Agent Object that supports CA SiteMinder® 4.x functionality. Define this agent object on your Policy Servers.

Important: Do not add the 4.x agent object to any agent group, realm, or policy. This agent object exists only to support the internal operations of the Agent for SharePoint.

Follow these steps:

1. Click Infrastructure, Agent, Agents.
2. Click Create Agent.
The Create Agent screen appears.
3. Click OK.
The Create Agent: screen appears.
4. Enter a distinctive name and description.
5. Verify that the CA SiteMinder® option button is selected and that Web Agent appears in the Agent Type drop-down list.
6. Click the Supports 4.x agents check box.

The trust settings fields appear.

7. Complete the following fields:

IP Address

Specifies the IP Address of the Policy Server.

Shared Secret

Specifies a password that is associated with the 4.x Agent object. The SharePoint Connection Wizard also requires this password.

Confirm Secret

Confirms a password that is associated with the 4.x Agent object. The SharePoint Connection Wizard also requires confirmation of this password.

8. Click Submit.

The agent object is created and a confirmation screen appears.

Create an Agent Configuration Object

An embedded Apache web server is part of the Agent for SharePoint. An Agent Configuration Object (ACO) on the Policy Server contains configuration parameters that control the behavior of the agent running on the embedded web server.

Agents need values in certain parameters to start. For example, all agents need one value in either of the following parameters:

- AgentName
- DefaultAgentName

Other parameters control optional functions that you can set anytime. For example, if you decide to store agent logs on your web server, you can set those parameters later. Agents do not need values in logging parameters to start.

Note: For more information about other parameters in your ACO that are not listed here, see the CA SiteMinder® *Web Agent Configuration Guide*.

Follow these steps:

1. Click Infrastructure, Agent Configuration, Create Agent Configuration.

The Create Agent Configuration: Search pane opens.

2. Click the following buttons:

- Create a copy of an object of type Agent Configuration.
- SharePoint2010DefaultSettings.

Important! Only copy the SharePoint2010DefaultSettings ACO object. Do not copy any other object in the list.

3. Click OK.

4. Type the name and a description for the agent configuration object.
5. If you have multiple virtual hosts and plan to assign different Agent identities to each virtual host, use the AgentName parameter. Use the DefaultAgentName parameter, if different Agent identities for virtual hosts are not required. Remove any # character in front of the parameter name, and then change the value of *one* of the following parameters (*not* both):

AgentName

Defines the identity of the web agent. This identity links the name and the IP address or FQDN of each web server instance hosting an Agent.

The value of the DefaultAgentName is used instead of the AgentName parameter if any of the following events occur:

- The AgentName parameter is disabled.
- The value of AgentName parameter is empty.
- The values of the AgentName parameter do *not* match any existing agent object.

Note: This parameter can have more than one value. Use the multivalue option when setting this parameter in an Agent Configuration Object. For local configuration files, add each value to a separate line in the file.

Default: No default

Limit: Multiple values are allowed.

Limits: Must contain 7-bit ASCII characters in the range of 32-127, and include one or more printable characters. Cannot contain the ampersand (&) and asterisk (*) characters. The value is not case-sensitive. For example, the names MyAgent and myagent are treated the same.

Example: myagent1,192.168.0.0 (IPv4)

Example: myagent2, 2001:DB8::/32 (IPv6)

Example: myagent,www.example.com

DefaultAgentName

Defines a name that the agent uses to process requests. The value for DefaultAgentName is used for requests on an IP address or interface when no agent name value exists in the AgentName parameter.

If you are using virtual servers, you can set up your CA SiteMinder® environment quickly by using a DefaultAgentName. Using DefaultAgentName means that you do not need to define a separate agent for each virtual server.

Important! If you do not specify a value for the DefaultAgentName parameter, then the value of the AgentName parameter requires every agent identity in its list. Otherwise, the Policy Server cannot tie policies to the agent.

Default: No default.

Limit: Multiple values are allowed.

Limits: Must contain 7-bit ASCII characters in the range of 32-127, and include one or more printable characters. Cannot contain the ampersand (&) and asterisk (*) characters. The value is not case-sensitive. For example, the names MyAgent and myagent are treated the same.

6. Change the value of the following parameter:

LogOffUri

Enables full log-out and displays a confirmation page after users are successfully logged off. Configure this page so that it *cannot* be stored in a browser cache. If a cached page is used, session hijacking by unauthorized users is possible.

When the SharePoint users click the Sign out link, the following URI is used:

- `/_layouts/SignOut.aspx`

When the SharePoint users click the Sign in as another user link, the following URI is used:

- `/_layouts/accessdenied.aspx?loginasanotheruser=true`

If you have multiple SharePoint web sites below a top-level SharePoint website, add the URIs of the lower-level sites to the LogOffUri parameter.

Note: When the CookiePath parameter is set, the value of the LogOffUri parameter must point to the same cookie path. For example, if the value of your CookiePath parameter is set to example.com, then your LogOffUri must point to example.com/logoff.html

Default: `/_layouts/SignOut.aspx,`
`/_layouts/accessdenied.aspx?loginasanotheruser=true`

Limits: Multiple URI values permitted. Do *not* use a fully qualified URL. Use a relative URI.

Example: (for a parent site of www.example.com with two lower-level sites named finance and hr respectively) `/finance/_layouts/SignOut.aspx,`
`finance/_layouts/accessdenied.aspx?loginasanotheruser=true`
`/hr/_layouts/SignOut.aspx,`
`/hr/_layouts/accessdenied.aspx?loginasanotheruser=true`

7. Click OK.

The new values appear next to the parameters in the list.

8. Click Submit.

The Create Agent Configuration Task is submitted for processing and the confirmation message appears.

Create A User Directory Connection

The Policy Server communicates with a user directory to authenticate users. The user directory needs a connection defined in the CA SiteMinder® Administrative UI. Create a connection for your directory that contains users who require access to SharePoint resources.

Note: Only the directory vendors that CA SiteMinder® supports operate with the Agent for SharePoint. For more information, see the Platform Support Matrix at www.support.ca.com.

Follow these steps:

1. Click Infrastructure, Directory, User Directory, Create User Directory.
The Create User Directory pane appears.
2. Enter the Name and an optional description.
3. Select the Directory type from the Namespace list and complete the required connection information under the Directory Setup.
4. If your directory server requires credentials for searches, do the following steps:
 - a. Click the Require Credentials check box.
 - b. Type the user name and password of an authorized account.

Note: The Require Credentials setting is required for LDAP directories which support anonymous search. This setting supports queries that the CA SiteMinder® Claims Provider makes to the user directory to support the SharePoint People Picker. For more information about these credentials, see the administrator of your directory server.

5. (Optional) In the User Attributes fields, specify the user directory profile attributes that are reserved for CA SiteMinder®.
6. Click Submit.

The Create User Directory task is submitted for processing, and the confirmation message appears.

More information:

[How to Configure Multiple User Directories](#) (see page 243)

Create Virtual Attribute Mappings for your User Claim

Integration with SharePoint requires at least one claim that contains an identifier that uniquely identifies the user. These claims often appear in the people picker as cryptic values, such as the following example:

`uid=e123456`

Such claims are difficult to associate with the intended user. The Agent for SharePoint uses a special attribute mapping which retrieves the display name of the user. This user name appears next to the related identifier claim in the people picker. After this user mapping is configured, the previous example appears in the people picker like the following one:

`uid=e123456 associated_user_name`

To create a virtual attribute mapping for your user claim, select the procedure corresponding to your type of directory server from the following list:

- Create an attribute mapping for user claims in LDAP directories.
- Create an attribute mapping for user claims in Active Directory servers.
- [Create attribute mappings for user claims in an ODBC directory](#) (see page 54).

Create an Attribute Mapping for User Claims in an LDAP Directory

The Agent for SharePoint requires an attribute mapping that is based on an attribute with a unique value for each user. Use the Administrative UI to create a pair of attribute mappings that defines how CA SiteMinder® searches for user claims through the SharePoint people picker.

Follow these steps:

1. Log on to the CA SiteMinder® Administrative UI.
2. Click Infrastructure, Directory, User Directory.
A list of user directory connections appears.
3. Click the name of the user directory that you want to modify.
The View User Directory screen appears. All fields and controls are inactive.
4. Scroll to the bottom of the page and click Modify.
The Modify User Directory page appears. All fields and controls are active.
5. In the Attribute Mapping List section, click Create.
The create attribute mapping page appears.
6. Verify that the Create a new object of type Attribute Mapping option button is selected, and then click OK.

7. Click the name field, and enter the following name:

useridentifier

8. Verify that the Alias option button is selected, and then click the Definition field.

9. Enter the following definition:

uid

10. Click OK.

The Modify User directory page appears.

11. To create the second mapping, repeat Steps 4 through 5.

12. Click the name field, and then enter the following name:

smuserdisplayname

13. Verify that the Alias option button is selected, and then click the Definition field.

14. Enter the following definition:

displayName

15. Click OK.

The Modify User directory page appears.

16. Click Submit.

The attribute mappings are created.

Create Attribute Mappings for User Claims in a Microsoft Active Directory Server

The Agent for SharePoint requires an attribute mapping that is based on an attribute with a unique value for each user. Use the Administrative UI to create a pair of attribute mappings that defines how CA SiteMinder® searches for user claims through the SharePoint people picker.

Follow these steps:

1. Log on to the CA SiteMinder® Administrative UI.

2. Click Infrastructure, Directory, User Directory.

A list of user directory connections appears.

3. Click the name of the user directory that you want to modify.

The View User Directory screen appears. All fields and controls are inactive.

4. Scroll to the bottom of the page and click Modify.

The Modify User Directory page appears. All fields and controls are active.

5. In the Attribute Mapping List section, click Create.

The create attribute mapping page appears.

6. Verify that the Create a new object of type Attribute Mapping option button is selected, and then click OK.
7. Click the name field, and enter the following name:
`useridentifier`
8. Verify that the Alias option button is selected, and then click the Definition field.
9. Enter the following definition:
`sMAAccountName`
10. Click OK.
The Modify User directory page appears.
11. To create the second mapping, repeat Steps 4 through 5.
12. Click the name field, and then enter the following name:
`smuserdisplayname`
13. Verify that the Alias option button is selected, and then click the Definition field.
14. Enter the following definition:
`displayName`
15. Click OK.
The Modify User directory page appears.
16. Click Submit.
The attribute mappings are created.

Create Attribute Mappings for User Claims in an ODBC Directory

The Agent for SharePoint requires attribute mappings that are based on attributes with a unique value for each user. To create three attribute mappings that define how CA SiteMinder® searches for user claims through the SharePoint people picker, use the Administrative UI .

Follow these steps:

1. Log on to the CA SiteMinder® Administrative UI.
2. Click Infrastructure, Directory, User Directory.
A list of user directory connections appears.
3. Click the name of the user directory that you want to modify.
The View User Directory screen appears. All fields and controls are inactive.
4. Scroll to the bottom of the page and click Modify.
The Modify User Directory page appears. All fields and controls are active.

5. To create the first mapping, do the following steps :
 - a. In the Attribute Mapping List section, click Create.
The create attribute mapping page appears.
 - b. Verify that the Create a new object of type Attribute Mapping option button is selected, and then click OK.
 - c. Click the name field, and enter the following name:
`useridentifier`
 - d. Verify that the Alias option button is selected, and then click the Definition field.
 - e. Enter the following definition:
`UserID`
 - f. Click OK.
The user mapping is created and the Modify User Directory page reappears.
6. To create the second mapping, do the following steps:
 - a. In the Attribute Mapping List section, click Create.
The create attribute mapping page appears.
 - b. Verify that the Create a new object of type Attribute Mapping option button is selected, and then click OK.
 - c. Click the name field, and enter the following name:
`smuserdisplayname`
 - d. Verify that the Alias option button is selected, and then click the Definition field.
 - e. Enter the following definition:
`Name`
 - f. Click OK.
The user mapping is created and the Modify User Directory page reappears.

7. To create the third mapping, do the following steps:
 - a. In the Attribute Mapping List section, click Create.
The create attribute mapping page appears.
 - b. Verify that the Create a new object of type Attribute Mapping option button is selected, and then click OK.
 - c. Click the name field, and enter the following name:
smusergroups
 - d. Verify that the Alias option button is selected, and then click the Definition field.
 - e. Enter the following definition:
Name
 - f. Click OK.
The user mapping is created and the Modify User Directory page reappears.
8. Click Submit.
The user directory modifications are committed.

Create an Authentication Scheme for the Agent for SharePoint

CA SiteMinder® uses authentication schemes to collect credentials and determine the identity of a user. During authentication, the agent communicates with the Policy Server to determine the proper credentials to retrieve from a user who is requesting resources.

If you are an experienced CA SiteMinder® user, you can use an existing authentication scheme instead of creating one.

Follow these steps:

1. Click Infrastructure, Authentication, Authentication Schemes, Create Authentication Scheme.

The Create Authentication Scheme pane appears.

2. Select Create a new object of type Authentication Scheme option, and then click OK.

The Create Authentication Scheme: Pane appears.

3. Enter a distinctive name, and (optional) description.
4. Select the type of Authentication Scheme from the Authentication Scheme Type list.

The options for your chosen Authentication Scheme appear.

5. Complete the fields for your Authentication Scheme.
6. Click Submit.

The Create Authentication Scheme task is submitted for processing and the confirmation screen appears.

Determine your policy model

CA SiteMinder® supports the following policy models:

- [Policy Domain](#) (see page 58) (for protecting resources *without* CA DLP)
- [Application \(EPM\)](#) (see page 64) (for protecting resources *with* CA DLP)

For example, you can protect the SharePoint root site with a policy domain. If you have document libraries that you want to protect with CA DLP, create applications for those libraries. In this situation, use the same agent or agent group in both the policy domain and the application.

Create a Policy Domain

A policy domain is a logical grouping of resources associated with one or more user directories. Policy domains contain realms, rules, responses, and policies (and optionally, rule groups and response groups).

The resources in a policy domain can be grouped in one or more realms. Rules control access to resources, that are associated with the realm that contains the resource. By grouping realms and rules in a policy domain, you can provide a secure domain for your resources.

For example, on a SharePoint site, some resources require a higher level of security than other resources. Define a realm with a higher level of security than uses an authentication scheme such as a certificate-based scheme. Use a realm with basic authentication for the less sensitive resources. For example, a common set of users wants to access both types of resources. You can group both realms in the same policy domain.

Follow these steps:

1. Click Policies, Domains.
2. Click Domain, Create Domain.

The Create Domain pane opens.

Note: Click Help for descriptions of settings and controls, including their respective requirements and limits.

3. Type the name and a description of the policy domain.
4. Add User Directories and [Realms](#) (see page 59).
5. Click Submit.

The Create Domain Task is submitted for processing.

Note: For more information about Policy Domains, see the *CA SiteMinder® Policy Server Configuration Guide*.

Assign User Directories

Add your user directories to a policy domain. The Policy Server authenticates users by comparing the credentials to the credentials that are stored in the user directories.

Follow these steps:

1. Under User Directories, click Add/Remove.
The Choose user directories pane opens.
2. Select a user directory from the list of Available Members, and click the right-facing arrows.
The user directory is removed from the list of Available Members and added to the list of Selected Members.
3. Click OK.
The selected user directory is added to the domain.

Note: To create a user directory and add it to the domain, click New... under User Directories.

Configure Realms

Realms are groupings of resources in a specific location on your network. CA SiteMinder® <agents> protect the resources in a realm. When users request resources within a realm, the associated Agent for SharePoint authenticates the user. The realm uses the authentication scheme you configured. The SharePoint server authorizes the user.

Because most SharePoint resources are URL-based, define the URLs of your SharePoint resources that you want to protect. Use the following examples as guides:

- <http://intranet.example.com>
- <http://intranet.example.com/finance>
- <http://intranet.example.com/investors>

Follow these steps:

1. Click Policies, Domains.
2. Click Realm, Create Realm.
The Create Realm: Select Domain pane appears.
3. Select the domain you created for your SharePoint resources from the Domain list, and then click Next.
The Create Realm: Define Realm pane appears.
4. Complete the name and description fields.

5. Click the ellipsis option button.

The Select an Agent screen appears.

6. Click the option button next to the Agent object you created for your SharePoint resources, and then click OK.

Important: Do not add the 4.x agent object to any agent group, realm, or policy. This agent object exists only to support the internal operations of the Agent for SharePoint.

7. Click the Resource filter field, and then enter the URL of a SharePoint resource that you want to protect. The realms meet the minimum requirements to enable basic authentication:

- Create a realm to protect the authentication URL:

`/affwebservices/redirect.jsp/redirect.jsp`

- Create a realm to leave the ClaimsWS unprotected:

`/ClaimsWS/services/WSSharePointClaimsServiceImpl`

Note: We recommend protecting only URLs on SharePoint systems, not lists, or specific documents.

8. Under rules, create new rules or delete existing rules.
9. Under Sub realms, create new sub realms or delete existing sub realms.
10. Under Session, specify the session properties.
11. Under Advanced, specify the following:
 - Registration schemes.
 - Authorization directory mappings.
 - Types of events you want the realm to process.
12. Click Finish.

The Create Realm Task is submitted for processing.

Create a Rule for Web Agent Actions

You can create a rule that fires in response to specified Web agent actions. The rule allows or denies access to the resource it is protecting.

To create a rule

1. Click Policies, Domains.
2. Click Rule, Create Rule.

The Create Rule: Select Domain pane opens.

3. Select a domain from the Domain list, and click Next.
The Create Rule: Select Realm pane opens.
4. Select the realm that includes the resources that you want the rule to protect, and click Next.
The Create Rule: Define Rule pane opens.
Note: If a realm does not exist for the resources that you want to protect, a rule cannot be created to protect those resources.
5. Type the name and a description of the rule in the fields on the General group box.
Note: Click Help for descriptions of settings and controls, including their respective requirements and limits.
6. Type the resource that you want the rule to protect in the Resource field.
The Effective Resource updates to include the resource.
7. Specify whether the rules allow or deny access to the protected resource in the Allow/Deny and Enable/Disable sections.
8. Select the Web agent actions option button in the Action section.
The Action List is populated with HTTP actions.
9. Select one or more HTTP actions from the Action list.
10. (Optional) Specify time restrictions, an active rule, or both in the Advanced section.
11. Click Finish.
The Create Rule task is submitted for processing.

Create a Policy

You can create a policy by adding it to a new or existing domain. Policies define relationships between users and resources.

Follow these steps:

1. Click the Policies, Domains.
2. Click Domain, Modify Domain.
The Modify Domain pane opens.
3. Specify search criteria, and click Search.
A list of domains that match the search criteria opens.
4. Select a domain, and click Select.
The Modify Domain: *Name* pane opens.
5. Click the Policies tab on the Domain pane.
The Policies dialog opens.

6. Click Create.

The Create Policy: *Name* pane opens.

7. Type the name and a description of the policy.

8. Click the Users tab.

The User Directories dialog opens.

9. Add users, user groups, or both to the policy, and click OK.

The Modify Domain: *Name* pane reopens.

10. Click Submit.

The Modify Domain Task is submitted for processing.

Add Users to a Policy

You can add individual users, user groups, or both to a policy and can create a policy binding between the added users and the policy. When a user tries to access a protected resource, the policy verifies that the user is part of its policy binding. Then the policy fires the rules included in the policy to see if the user is allowed to access the resource.

Follow these steps:

1. Click Policies, Domains.

The Domain pane appears.

2. Click Policy, Modify Policy.

The Modify Policy page appears.

3. Select the policy to change from the search results and click Select.

The Modify Policy:*Name* page appears.

4. Click the Users tab on the Policy pane.

The User Directories pane opens and contains group boxes for each user directory that is associated with the policy domain.

5. Add users or groups from the user directory to the policy.

In each user directory section, you can select Add Members, Add Entry, Add All. Depending on which method you use to add users to the policy, a dialog opens to let you add users.

Note: If you select Add Members, the User/Groups pane opens. Individual users are not displayed automatically. Use the search utility to find a specific user within one of the directories.

You can edit or delete a user or group by clicking the right arrow (>) or minus sign (-), respectively.

6. Select individual users, user groups, or both using whatever method and click OK.

The User Directories pane reopens and lists the new users for the policy on the section of the user directory. The task of binding users to the policy is complete.

Add Rules to a Policy

Rules indicate the specific resources included in a policy and whether to allow or deny access to the resources when the rule fires. Responses indicate the actions you want to occur when the rule fires.

Note: Add at least one rule or rule group to a policy.

Follow these steps:

1. Navigate to Policy, Rules.
The Rules dialog opens.
2. Click Add Rule.
The Available Rules pane opens.
3. Select the individual rules, rule groups, or both that you want to add to the policy, and click OK.
The Rules section lists the added rules and groups.
4. (Optional) Associate the rule with a response or response group.

Note: To remove a rule or rule group from a policy, click the minus sign (-) to the right of the rule on the Rules section. To create a rule, click New Rule on the Available Rules pane.

Create a CA SiteMinder® Application to Protect SharePoint Resources that CA DLP also Protects

CA SiteMinder® applications protect resources by combining access privileges with specific conditions. Users who have the privileges and meet the conditions are granted access to the resources they request.

This section describes creating an application with the following components:

- A resource filter to protect the authentication URL.
- A resource filter to leave the claims web service (ClaimsWS) unprotected by CA SiteMinder®.
- A connection to the user directory that contains your SharePoint users.

These components meet the minimum requirements of the CA SiteMinder Agent for SharePoint. We recommend creating few applications and components during evaluation, testing, or initial-deployment environments. You can add more applications and components at any time.

Note: Resources protected with CA DLP [require applications](#) (see page 57). Do *not* use policy domains.

Follow these steps:

1. Click Policies, Applications.
The applications screen appears.
2. Click Create Application.
The Create Application: screen appears, with the General tab selected.
3. Enter a distinctive name and optional description.
4. Create the component for the authentication URL by doing the following steps:
 - a. Click the Component Name field, and type a distinctive name describing the SharePoint resources that you want to protect, such as, "Protected SharePoint Resources."
 - b. Verify that Web Agent appears in the Agent Type drop-down list.
 - c. Click Lookup Agent/Agent Group.
The Select Agent or Agent Group screen appears.
 - d. Click the option button that corresponds to your Agent Object, and then click OK.
Important: Do not add the 4.x agent object to any agent group, application, or component. This agent object exists only to support the internal operations of the Agent for SharePoint.
 - e. Click the Resource Filter field, and then enter the following value:
`affwebservices/redirectjsp/redirect.jsp`

Verify that the field begins with *one* forward slash as shown in the following example:

`/affwebservices/redirectjsp/redirect.jsp`

- f. Click the Authentication Scheme drop-down list, and then select the authentication scheme that you want.
 - g. Click OK.
5. Create the component for the ClaimsWS by doing the following steps:
- a. Click Create Component.
The Create Component screen appears, with the cursor in the Component Name field.
 - b. Type a distinctive name describing the SharePoint resources that you want to protect, such as, "Claims Web Service."
 - c. Verify that Web Agent appears in the Agent Type drop-down list.
 - d. Click Lookup Agent/Agent Group.
The Select Agent or Agent Group screen appears.
 - e. Click the option button that corresponds to your Agent Object, and then click OK.

Important: Do not add the 4.x agent object to any agent group, application, or component. This agent object exists only to support the internal operations of the Agent for SharePoint.
 - f. Click the Resource Filter field, and then enter the following value:
`ClaimsWS/services/WSSharePointClaimsServiceImpl`
 - g. Verify that the field begins with *one* forward slash as shown in the following example:
`/ClaimsWS/services/WSSharePointClaimsServiceImpl`
 - h. Click the Unprotected option button.
 - i. Click OK.
6. Add your user directory connection by doing the following steps:
- a. Click Add/Remove.
The Choose user directories screen appears.
 - b. Under the Available Members, click the directory connections that you want, and then click the arrow icon between the lists.
Your directory connections move to the Selected Members list.
 - c. Click OK.
The Choose user directories screen closes, and the Create Application: screen appears.

Note: The components in Steps 5 and 6 are the basic components the Agent for SharePoint requires to operate. For testing or production environments, create components for the other SharePoint URLs resources you want to protect. Possible examples of components include the following items:

- `http://intranet.example.com`
- `http://intranet.example.com/finance`
- `http://intranet.example.com/investors`

7. Click Submit.

The application is created and a confirmation message appears.

Add Resources to your Application

CA SiteMinder® applications use resources to protect items in your SharePoint environment. These resources for CA SiteMinder® applications consist of the following parts:

- Rules.
- Authentication or authorization actions which occur when a rule fires.

Note: In the previous context, "resources" refers only to the rules and actions that are associated with CA SiteMinder® applications. Generally, resources indicate the protected items on a SharePoint server, such as URLs.

Follow these steps:

1. Click Policies, Applications.

The applications screen appears, showing a list of applications.

2. Locate the application that you created to protect your SharePoint sites, and then click the Edit icon.

The Modify Application: screen appears.

3. Click the Resources tab.

The Resources screen appears.

4. Click the Select a context root drop-down list, and then select the resource filter that you previously created for your SharePoint authorization URL. See the following example:

`/affwebservices/redirectjsp/redirect.jsp`

5. Click Create.

The General screen appears.

6. Enter a distinctive name, and an optional description.

7. Verify that the Web Agent actions option button is selected, and then Ctrl-click the following items in the Action list:

- Get
- Head
- Options
- Post
- Put

8. Click OK.

The General screen closes and the Resources screen appears.

9. Click Submit.

The application resources are created and a confirmation message appears.

Add Roles to your Application

CA SiteMinder® applications use roles to define the users or groups or organizations to which you wish to grant access to your SharePoint resources.

Follow these steps:

1. Click Policies, Applications.

The applications screen appears, showing a list of applications.

2. Locate the application that you created to protect your SharePoint sites, and then click the Edit icon.

The Modify Application: screen appears.

3. Click the Roles tab.

The Roles screen appears.

4. Click Create Role.

5. Verify that the Create a new object of type Role option button is selected, and then click OK.

The Create Role: screen appears.

6. Enter a distinctive name and optional description.

7. Create any of the following roles:

- Roles that are based on membership in a group (member groups).
- Roles that are based on membership in an organization (member organizations).
- Roles that are based on user attributes (Member attributes, such as users who match a particular attribute in your user directory).

8. Click OK.

The Create Role: screen closes, and the Modify Application: screen appears.

9. Click Submit.

The Role is created and a confirmation message appears.

Add a Policy to your Application

Policies combine application resources and roles to protect your SharePoint environment.

Follow these steps:

1. Click Policies, Applications.

The applications screen appears, showing a list of applications.

2. Locate the application that you created to protect your SharePoint sites, and then click the Edit icon.

The Modify Application: screen appears.

3. Click the Policies tab.

The Policies screen appears.

4. Click the Select a context root drop-down list, and then select the resource filter that you previously created for your SharePoint authorization URL. See the following example:

`/affwebservices/redirect.jsp/redirect.jsp`

5. Click the check boxes of the roles that you want to associate with your rules for the resource from Step 4.
6. Click the check boxes of the responses that you want to associate with your rules for the resource from Step 4.
7. Click Submit.

The Policies screen closes. The Modify Application screen appears with a confirmation message.

Enable Paging for Searches of Active Directory User Stores (32-bit systems)

Valid for Policy Servers that are installed on Windows 32-bit operating environments that are connected to Active Directory servers.

Symptom:

I cannot use the SharePoint people picker to search my Active Directory user store.

Solution:

The Active Directory namespace does not support paging, causing searches of more than 1000 users to fail. To support searches of large numbers of users in the Active Directory namespace, set the EnablePagingADNameSpace registry key to one.

To enable paging for searches on your Windows Policy Server:

1. Open the Windows registry editor.
2. Locate the following registry key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Netegrity\SiteMinder\CurrentVersion\Ds\LDAPProvider\EnablePagingADNameSpace
```

3. Set the value of the key to 1.

To enable paging for searches on your UNIX Policy Server:

1. Navigate to *policy_server_installation_directory/siteminder/registry*
2. Open sm.registry in a text editor.
3. Locate the following text in the file:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Netegrity\SiteMinder\CurrentVersion\Ds\LDAPProvider\EnablePagingADNameSpace
```

4. Set the value of the key to 1.

Enable Paging for Searches of Active Directory User Stores (64-bit systems)

Valid for Policy Servers that are installed on Windows 64-bit operating environments (using WoW64 mode) that are connected to Active Directory servers.

Symptom:

I cannot use the SharePoint people picker to search my Active Directory user store.

Solution:

The Active Directory namespace does not support paging, causing searches of more than 1000 users to fail. To support searches of large numbers of users in the Active Directory namespace, set the EnablePagingADNameSpace registry key to one.

To enable paging for searches on your Windows Policy Server:

1. Open the Windows registry editor.
2. Locate the following registry key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Netegrity\SiteMinder\CurrentVersion\Ds\LDAPProvider\EnablePagingADNameSpace
```

3. Set the value of the key to 1.

Chapter 6: Request and Install a Policy Server Token Signing Certificate

This section contains the following topics:

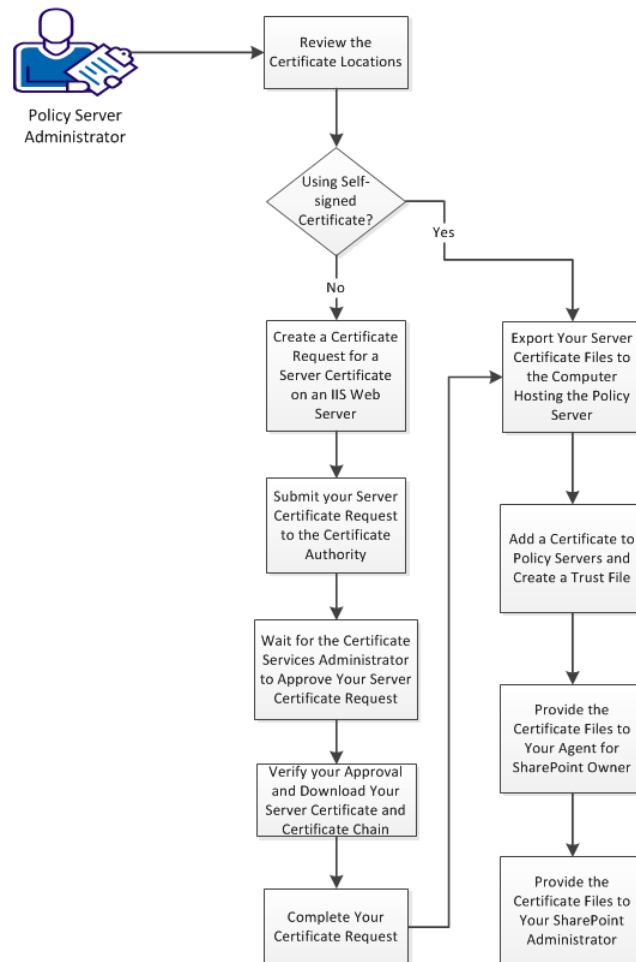
[How to Request and Install a Policy Server Token Signing Certificate](#) (see page 73)

How to Request and Install a Policy Server Token Signing Certificate

The Policy Server requires an SSL certificate to sign the WS-Fed token it sends to the SharePoint claims provider. This certificate verifies that the WS-Fed token is from the Policy Server and not an unauthorized third party.

The following graphic describes the process for requesting and installing a Policy Server signing certificate:

How to Configure Your Policy Server to Sign WS-Fed Tokens



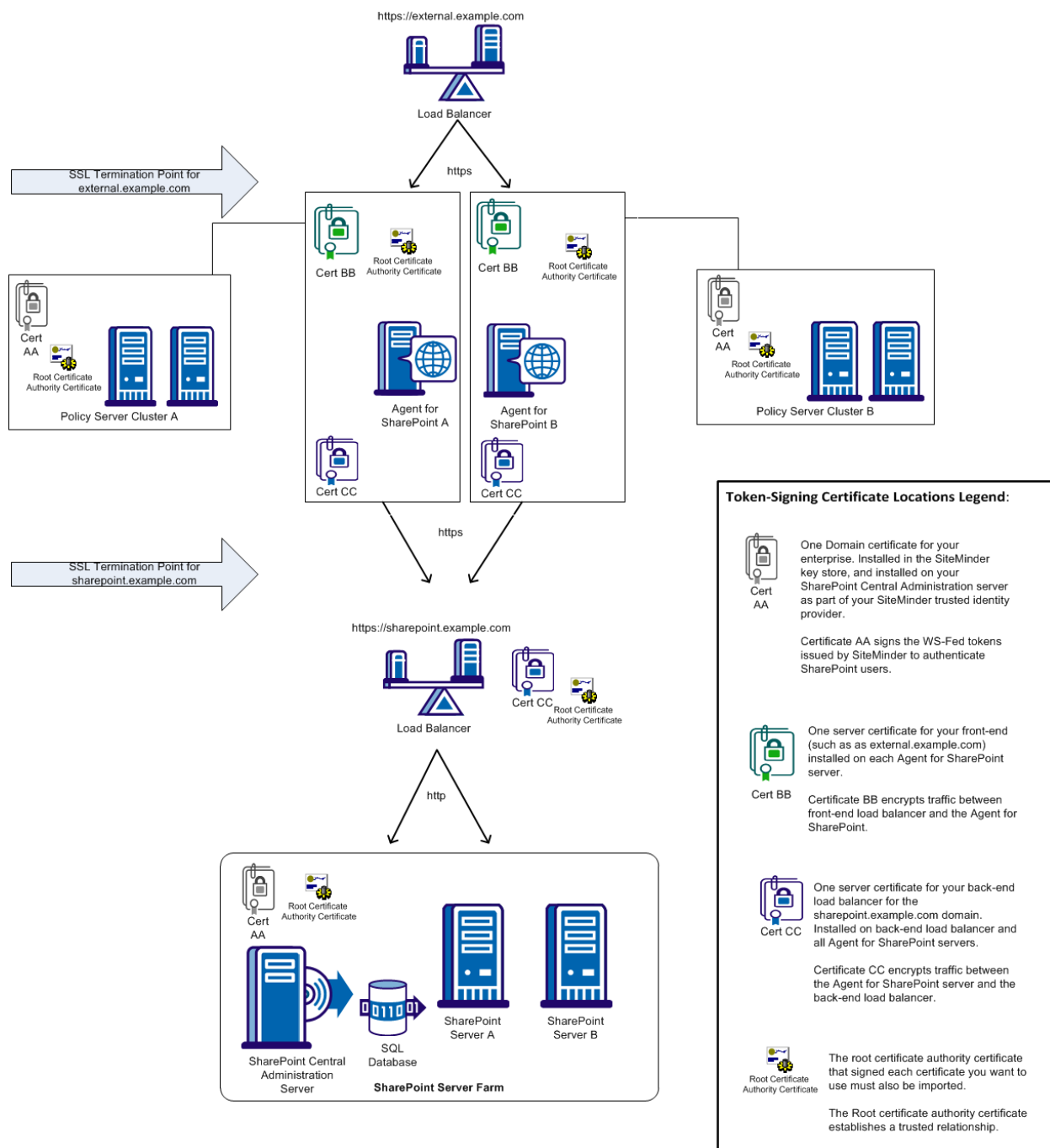
Follow these steps:

Note: This procedure provides one possible example of how to configure this feature using third-party tools. CA Technologies did *not* develop nor provide these tools. These tools are subject to change at any time by the third party without notice. Use this procedure as a guide for configuring this feature in your specific environment. The actual steps that are required in your situation could be different from the steps that are shown here.

1. Review the [certificate locations](#) (see page 76).
2. If you are using a self-signed certificate, go to Step 8.
Important! Do not use self-signed certificates in production environments. We recommend using self-signed certificates in test environments only.
3. [Create a certificate request for a server certificate on an IIS web server](#) (see page 77).
4. [Submit your server certificate request to the certificate authority](#) (see page 78).
5. Wait for the Certificate Services administrator to [approve your server certificate request](#) (see page 79).
6. [Verify your approval and download your server certificate and certificate chain](#) (see page 81).
7. [Complete your certificate request \(using the same IIS web server and browser from Step 3\)](#) (see page 80).
8. [Export your server certificate files to the computer hosting the Policy Server](#) (see page 82).
9. [Add a certificate to Policy Servers and create a trust file](#) (see page 83).
10. [Provide the certificate files to your CA SiteMinder Agent for SharePoint owner](#) (see page 85).
11. [Provide the certificate files to your SharePoint administrator](#) (see page 85).

Token-Signing Certificate Locations in Your SharePoint Environment

The following illustration shows the typical locations of the certificates that sign your WS-Fed tokens in your SharePoint environment:



Create a Certificate Request for a Server Certificate on an IIS Web Server

Requesting a certificate is the first step in the process of creating a Policy Server signing certificate. Any IIS web server in your organization can request a certificate. Using an IIS web server hosted on your Policy Server is more convenient, because it eliminates exporting the certificates to the Policy Server.

Follow these steps:

Note: This procedure provides one possible example of how to configure this feature using third-party tools. CA Technologies did *not* develop nor provide these tools. These tools are subject to change at any time by the third party without notice. Use this procedure as a guide for configuring this feature in your specific environment. The actual steps that are required in your situation could be different from the steps that are shown here.

1. Open Internet Information Services (IIS) Manager.
2. Under Connections, click the web server.
3. Double-click Server Certificates.

A list of certificates appears.

4. Under Actions, click Create Certificate request...

The Create Certificate wizard appears.

5. Complete the wizard. Save the certificate request to a local file. We recommend using a distinctive name that is easy to remember. For example, `ps_wsfd_signing_certificate_request.txt`

The certificate request is created.

Submit Your Certificate Request to a Certificate Authority

After generating your certificate request on an IIS web server, request a certificate from the web server in your organization hosting Active Directory Certificate Services.

Skip this procedure in any of the following situations:

- If you do not use Active Directory Certificate services in your organization.
- You typically submit your certificate requests to an independent, third-party certificate authority.

In any of the previous situations, follow your typical procedures instead.

Follow these steps:

Note: This procedure provides one possible example of how to configure this feature using third-party tools. CA Technologies did *not* develop nor provide these tools. These tools are subject to change at any time by the third party without notice. Use this procedure as a guide for configuring this feature in your specific environment. The actual steps that are required in your situation could be different from the steps that are shown here.

1. Open your web browser.
2. Navigate to the following URL:

`https://fully_qualilfied_domain_name_of_server_running_active_directory_certificate_services/certsrv`

An example of such a URL is `http://certificateauthority.example.com/certsrv`.

3. Click Request a certificate.

The Request a certificate screen appears.

4. Click the advanced certificate request link.

5. Click Submit a certificate request by using a base-64-encoded CMC or PKCS #10 file, or submit a renewal request by using a base-64-encoded PKCS #7 file.

The Submit a Certificate Request or Renewal Request screen appears.

6. Open the text file containing your certificate request with a text editor. Copy and paste the entire contents of the file into the Saved request field on the screen.

7. Click Submit.

The certificate pending screen appears.

8. Note the following items for future reference:

- Your request ID.
- Use the same browser to verify the status of your request within ten days.

The request is submitted.

Approve a Certificate Request using Active Directory Certificate Services

Certificate administrators approve or reject certificate requests. Certificate administrator privileges are separate from Administrator privileges. Not all users who have accounts on the computer hosting Active Directory Certificate services have sufficient privileges to approve or reject certificates.

If you have certificate administrator privileges on the web server to which your certificate was submitted, use this procedure. Otherwise, ask the certificate administrator to do this approval for you.

Follow these steps:

Note: This procedure provides one possible example of how to configure this feature using third-party tools. CA Technologies did *not* develop nor provide these tools. These tools are subject to change at any time by the third party without notice. Use this procedure as a guide for configuring this feature in your specific environment. The actual steps that are required in your situation could be different from the steps that are shown here.

1. Log in to the web server hosting the Active Directory Certificate services using an account with Certificate administrator privileges.
2. Click Start, Administrative Tools, Certification Authority
The certsrv snap-in appears.
3. Click the name of the certification authority, and then click the pending request folder.
A list of pending certificate requests appears.
4. Right-click the request ID associated with the request for the Policy Server Signing certificate.
5. From the context menu, select All Tasks, Issue.
The certificate is issued.

Complete Your Certificate Request

After downloading your certificate (*.cer) file, complete your certificate request by adding the certificate to your IIS web server. Use the same IIS server from which you originally requested the certificate.

Note: This procedure provides one possible example of how to configure this feature using third-party tools. CA Technologies did *not* develop nor provide these tools. These tools are subject to change at any time by the third party without notice. Use this procedure as a guide for configuring this feature in your specific environment. The actual steps that are required in your situation could be different from the steps that are shown here.

Follow these steps:

1. Open Internet Information Services (IIS) Manager.
The Start page appears.
2. Under Connections, click the web server.
3. Double-click Server Certificates.
A list of certificates appears.
4. Under Actions, click Complete Certificate Request...
The Complete Certificate Request wizard appears.
5. Complete the wizard by doing the following tasks:
 - a. Navigate to the *.cer file you downloaded previously.
 - b. Create a friendly name for the *.cer file.The new certificate appears in the list of certificates.

Verify Your Approval and Download Your Certificate and Certificate Chain

Use the same IIS web server and web browser from which you submitted the request to verify the status of your certificate request. If your certificate is approved, download both the certificate and the certificate chain to your IIS web server.

Follow these steps:

Note: This procedure provides one possible example of how to configure this feature using third-party tools. CA Technologies did *not* develop nor provide these tools. These tools are subject to change at any time by the third party without notice. Use this procedure as a guide for configuring this feature in your specific environment. The actual steps that are required in your situation could be different from the steps that are shown here.

1. Open your web browser you used to request your certificate.
2. Navigate to the following URL:

`https://fully_qualilfied_domain_name_of_server_running_active_directory_certificate_services/certsrv`

An example of such a URL is `http://certificateauthority.example.com/certsrv`.

3. Click View the status of a pending certificate request.

A list of your certificate requests appears.

4. Click the link for your certificate request.

The Certificate Issued screen appears. If it does not, contact the certificate administrator in your organization for more information.

5. Click the Base 64 Encoded option button.
6. Click all the following links and save the files to your web server:
 - Copy Certificate. (downloads a *.cer file)
 - Copy Certificate Chain (downloads a *.p7b file)

Your certificate is downloaded.

Export Your Policy Server Signing Certificate

Export your Policy Server Signing certificate with IIS manager. This export process creates a certificate file that you add to your Policy Server.

Note: This procedure provides one possible example of how to configure this feature using third-party tools. CA Technologies did *not* develop nor provide these tools. These tools are subject to change at any time by the third party without notice. Use this procedure as a guide for configuring this feature in your specific environment. The actual steps that are required in your situation could be different from the steps that are shown here.

Follow these steps:

1. Open Internet Information Services (IIS) Manager.
The Start page appears.
2. Under Connections, click the web server.
3. Double-click Server Certificates.
A list of certificates appears.
4. Click your Policy Server signing certificate.
Your Policy Server signing certificate is selected.
5. Under Actions, click Export.
The Export Certificate dialog appears.
6. Do the following steps:
 - a. Click the ellipsis button and select a directory for your exported certificate.
A browse dialog appears.
 - b. Enter a file name for your exported certificate.
 - c. Click Open.
The browse dialog closes.
 - d. Enter a password for the exported certificate and confirm it.
Note: You need this password to import this certificate into the central key store shared by the Policy Servers.
 - e. Click OK.
The Export Certificate dialog closes and the certificate is exported.
7. Close the Internet Information Services (IIS) Manager.

Add a Policy Server Signing Certificate to Policy Servers and Create a Trust File

CA SiteMinder® requires a certificate to complete signing the WS-Token. CA SiteMinder® signs the WS-Token and sends it to SharePoint. To create a certificate for the WS-Token, import an existing certificate that contains both a private and a public key. After the certificate has been imported to the key store and been assigned an alias, export the certificate to your SharePoint Central Administration server to create a trust certificate.

This certificate often uses the Public-Key Cryptography Standards #12 (PKCS) format. In the following example, the password protects the PKCS#12 file.

Note: On Windows operating environments, a .pfx file is equivalent to a .p12 file.

Follow these steps:

1. Log on to the Administrative UI.
2. Add the Policy Server signing certificate to the Policy Servers with the following steps:
 - a. Click Infrastructure, X509 Certificate Management, Trusted Certificates and Private Keys.
The trusted certificates and private keys screen appears.
 - b. Click Import New.
The Import Certificate/Private key wizard starts.
 - c. Click the Browse button, navigate to the certificate that you want to import, and then click Next.
 - d. Enter the password with which you previously exported the certificate, and then click Next.
 - e. Highlight the text in the Alias field, and then type a new Alias for the certificate.
 - f. Click Next.
 - g. Review the information that is shown on the confirmation screen, and then click Finish.
The Policy Server signing certificate is added to the central key store on the Policy Servers. The Policy Server signing certificate appears in the list that is shown on the Administrative UI.
3. Create a trust certificate for your SharePoint central administration server with the following steps:
 - a. Locate the certificate from Step 2g in the list.
 - b. Click the Action drop-down list, and then choose Export.
The Export Key Store Entry screen appears.
 - c. Verify that the following value appears in the format drop-down list:
X509-DER
 - d. Click Export.
 - e. Save the certificate to another location.
The trust certificate for your SharePoint central administration server is created.
4. Copy the certificate from Step 3e to a directory on your SharePoint central administration server. This certificate is the trust certificate.
5. Copy any Certificate Authority Certificates in the certificate chain to a directory on your SharePoint central administration server.

Note: The Powershell script (which the SharePoint connection wizard creates) requires the paths to the following certificates on your SharePoint central administration server:

- The *exported_certificate_file_name.cer* (certificate) file.
- Any Certificate Authority certificates in the certificate chain.

More information:

[Modify the PowerShell Script](#) (see page 125)

Provide the Policy Server Signing Certificate Files to Your Agent Owner

The system hosting the CA SiteMinder Agent for SharePoint needs a copy of Policy Server signing certificate. This copy helps the CA SiteMinder Agent for SharePoint validate the WS-Fed tokens that the Policy Server sends. The certificate chain validates the Policy Server signing certificate.

Provide the following files to the administrator of the system that hosts the CA SiteMinder Agent for SharePoint:

- The Policy Server signing certificate file (.cer file) exported from the Policy Server.
- Any Certificate Authority certificates in the certificate chain.

More information:

[Install the Policy Server Signing Certificate on your CA SiteMinder Agent for SharePoint](#) (see page 99)

Provide the Certificate Files to Your SharePoint Administrator

The SharePoint central administration server needs a copy of Policy Server signing certificate. This copy helps the central administration server validate the WS-Fed tokens that the CA SiteMinder Agent for SharePoint forwards from the Policy Server. The certificate chain validates the Policy Server signing certificate.

The SharePoint administrator must edit the PowerShell script that the SharePoint connection wizard generates to include references to these certificate files.

Provide the following files to the SharePoint administrator:

- The Policy Server signing certificate file (.cer file) exported from the Policy Server.
- Any Certificate Authority certificates in the certificate chain.

More information:

[How to Configure the Trusted Identity Provider](#) (see page 123)

Chapter 7: Install and Configure the CA SiteMinder® Agent for SharePoint

This section contains the following topics:

[CA SiteMinder® Agent for SharePoint Configuration Overview](#) (see page 87)

[FIPS Support Overview](#) (see page 88)

[Install the CA SiteMinder® Agent for SharePoint](#) (see page 89)

[How to Configure the CA SiteMinder® Agent for SharePoint](#) (see page 91)

[Assign Permissions for Log Files and Directories on UNIX/Linux](#) (see page 99)

[Manage SharePoint Connections Using the SharePoint Connection Wizard](#) (see page 100)

[How to Start and Stop the Agent for SharePoint](#) (see page 108)

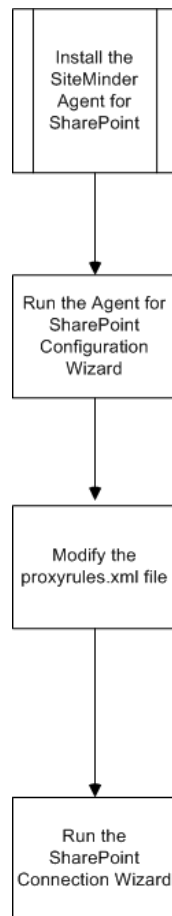
CA SiteMinder® Agent for SharePoint Configuration Overview

The CA SiteMinder® Agent for SharePoint authenticates the identities of users who request access to SharePoint resources using CA SiteMinder®. After CA SiteMinder® authenticates the user, the Agent for SharePoint creates a token, which is forwarded to the SharePoint server. SharePoint then receives and validates the token, it maps the assertions in it to internal SharePoint variables that are used for authorization.

The CA SiteMinder® Claims Provider module lets you search your CA SiteMinder® directories using the SharePoint people picker.

Installing and configuring the Agent for SharePoint involves several separate procedures.

The following illustration describes the tasks you perform when configuring the CA SiteMinder® Agent for SharePoint:



More information:

[Install the CA SiteMinder® Agent for SharePoint](#) (see page 89)

[Run the Configuration Wizard](#) (see page 94)

FIPS Support Overview

The Agent for SharePoint supports the requirements for cryptographic modules specified in the Federal Information Processing Standards (FIPS) 140-2 standard. When you install the agent, a dialog appears that prompts you to select the level of FIPS support your operating configuration requires.

During a new installation, you can select one of these three FIPS modes:

- COMPAT — Specifies that the installation is not FIPS-compliant. Select this mode when interacting with clients running earlier versions of the Agent for SharePoint.
- MIGRATE — Specifies that the Agent for SharePoint operates both with FIPS-compliant algorithms and algorithms used in earlier version of the agent simultaneously while the data is migrated.
- ONLY — Specifies that the Agent for SharePoint only uses or accepts FIPS-compliant algorithms. When you install in this mode, additional manual configuration is required.

The FIPS mode you select during installation usually is the same as the FIPS mode configured on the Policy Server. When the Policy Server is in Migrate mode, it can operate with the Agent for SharePoint in any mode.

Note: For more information about FIPS, refer the *CA SiteMinder® Policy Server Installation Guide*.

Install the CA SiteMinder® Agent for SharePoint

To use the Agent for SharePoint, the system where you plan to install it must have at least 256 MB of RAM. Other prerequisites differ based on the server system.

For detailed information, see the CA SiteMinder® Agent for SharePoint Support Matrix at <http://ca.com/support>.

Note: Installation prerequisites pertain to the system on which you run the Agent for SharePoint, not the destination servers to which the Agent for SharePoint routes incoming requests.

The Agent for SharePoint installation consists of two tasks:

1. Install the software.
2. Run the configuration tool.

Note: Throughout the installation instructions, there are references to *Agent-for-SharePoint_home* in directory paths. This variable represents the installation directory of the Agent for SharePoint.

Install the CA SiteMinder Agent for SharePoint on Windows

The default installation location for the agent on 32-bit Windows operating environments is: C:\Program Files\CA\Agent-for-SharePoint. On 64-bit Windows operating environments, the default installation location is C:\CA\Agent-for-SharePoint.

Important! The CA SiteMinder Agent for SharePoint cannot be installed on a computer that hosts any other web server. The CA SiteMinder Agent for SharePoint operates as a stand-alone proxy-based solution.

To run the agent installer on Windows operating environments, you need local Administrator privileges.

Note: We recommend installing the agent on an NTFS file-system partition.

Follow these steps:

1. Copy the installation program from the Download location on the CA Support site.
2. Right-click the following executable, and then select Run as administrator:

`ca-sp2010agent-version-operating_environment.exe`

The installation program starts.

3. Follow the instructions from the installation wizard.

Note: The installer displays all Java executables that are installed in the system. Pick a Java component and version that is equal to or greater than the one shown by the installer. If the installer does not detect any Java executables by default, then browse and select the appropriate path. For more information about the required Java executables or other third-party software requirements, see the platform support matrix.

4. Restart your system after the installation finishes.

Install the CA SiteMinder® Agent for SharePoint on UNIX

The default installation location is `user_home/CA/Agent-for-SharePoint`. The folder where you install the agent requires sufficient permissions (755). Do *not* install the agent under the `/root` folder, because its default permissions (750) are insufficient.

Important! The CA SiteMinder Agent for SharePoint cannot be installed on a computer that hosts any other web server. The CA SiteMinder Agent for SharePoint operates as a stand-alone proxy-based solution.

Note: On the Solaris or Linux operating environments, the agent runs under the "nobody" user account. If you prefer not to run the agent under this user account, create an alternate user and assign the necessary permissions. Do *not* run this program as a root user.

Follow these steps:

1. Copy the appropriate file for your operating environment from the download location on the CA Support site to a temporary directory:
 - Solaris operating environment: `ca-sp2010agent-version-sol.bin`
 - Linux operating environment: `ca-sp2010agent-version-linux.exe`
2. Enter the appropriate command for your operating environment from the following list:
 - Solaris: `sh ./ca-sp2010agent-version-sol.bin`
 - Linux: `sh ./ca-sp2010agent-version-linux.exe`
3. Follow the prompts that the installation wizard provides.

Note: The installer displays all Java executables that are installed in the system. Pick a Java component and version that is equal to or greater than the one shown by the installer. If the installer does not detect any Java executables by default, then browse and select the appropriate path. For more information about the required Java executables or other third-party software requirements, see the platform support matrix.

How to Configure the CA SiteMinder® Agent for SharePoint

After you install the Agent for SharePoint, configure the agent for the requirements of your SharePoint environment. Configuring the agent requires several separate procedures, which are described in the following process:

1. [Gather the information for your configuration wizard](#) (see page 92).
2. [Run the Agent for SharePoint Configuration Wizard](#) (see page 94).
3. [Confirm that the Agent for SharePoint is functioning](#) (see page 96).
4. Review the following example deployment diagrams:
 - [Deployment with a single web front end \(farms or stand-alone SharePoint servers\)](#) (see page 21).
 - [Deployment with multiple web front ends \(farms only\)](#) (see page 22).
5. [Set your proxy rule according to the deployment model you want](#) (see page 97).

Note: To operate the Agent for SharePoint with the CA DLP content classification service (CCS), [configure different proxy rules instead](#) (see page 335).
6. [\(Optional\) Enable support for dynamic Policy Server clusters](#) (see page 98).
7. Run the SharePoint connection wizard.

More information:

[CA SiteMinder Agent for SharePoint Configuration Wizard Information Worksheet](#) (see page 417)

[SharePoint Connection Wizard Information Worksheet](#) (see page 418)

Gather CA SiteMinder® Agent for SharePoint Configuration Wizard Information

The Agent for SharePoint configuration wizard helps you register a trusted host, configure the embedded Apache web server

To establish a connection between the Agent for SharePoint and the Policy Server, register a trusted host with the Policy Server. After registration is complete, the registration tool creates the SmHost.conf file. When this file is created successfully, the Agent for SharePoint is allowed to communicate with the Policy Server.

The following lists the required host registration information:

SiteMinder administrator name

Name of a SiteMinder administrator who has privileges to create a trusted host.

SiteMinder administrator password

Password of the SiteMinder administrator.

Trusted host name

Name of the trusted host assigned during configuration.

Note: The name you enter for the trusted host must be unique.

Host Configuration Object

Name of a host configuration object already defined in the Policy Server administrative UI.

Agent Configuration Object

Name of an existing Agent Configuration Object defined in the Policy Server administrative UI.

IP address of the Policy Server where the host is registered

Note: Include a port number for the Policy Server. For example, 121.111.12.11:44442.

Host Configuration File name and location

Identifies the SmHost.conf file, which Web Agents and custom Agents use to act on behalf of the trusted host. Using this file, the host can find a Policy Server and establish a connection. The wizard lists the default location.

Email address of the Agent for SharePoint administrator

The email address for the administrator Default: admin@example.com.

Fully qualified host name of the server

Specifies the hostname of the Agent for SharePoint, this hostname is the address users enter in their web browser:

spagent.example.com

Port number for HTTP requests

The port listening for HTTP requests Default: 80.

Port number for SSL requests

The port listening for SSL requests Default: 443.

Port number for HTTP Claims web service

The HTTP port used for Claims web service.

Port number for SSL Claims web service

The SSL port used for Claims web service.

Note: No default values are provided for the Claims WS HTTP and SSL Ports. However, use a port that is free, which Tomcat can use to host the web application. For UNIX and Linux, the ports must be greater than 1024. Nobody account works with ports above 1024.

Webagent Enable option

Indicates if the configuration wizard enables (starts) the agent automatically. This setting produces the same results as editing the EnableWebAgent parameter value in the WebAgent.conf file with a text editor.

Default: No (clear check box)

More information:

[CA SiteMinder Agent for SharePoint Configuration Wizard Information Worksheet](#) (see page 417)

Run the Configuration Wizard

After you install the Agent for SharePoint, run the configuration wizard. The configuration wizard registers the Agent for SharePoint with the Policy Server and performs some administrative tasks for the embedded Apache web server.

Before you run the wizard, verify that the required Policy Server objects exist.

Note: Tomcat uses the nobody user by default because it is the least privileged user.

Important! If you have previously run the configuration wizard on your Agent for SharePoint, create a backup copy of your proxyrules.xml file. The configuration wizard creates a default proxyrules.xml file each time it runs on a computer.

Follow these steps:

1. Open a console window and navigate to the directory *Agent-for-SharePoint_home/*
2. Enter one of the following commands:
 - Windows: ca-spagent-config.cmd
 - UNIX or Linux: ca-spagent-config.sh

The wizard starts. The Host Registration screen appears.

Note: In Windows, you can alternatively navigate to *Agent-for-SharePoint_home/install_config_info* and double-click *ca-spagent-config.exe*.

3. Select Yes option to perform host registration if the computer is not registered as a trusted host.
4. As part of the trusted host registration process, respond to the prompts as follows:
 - a. Specify the name and password of the SiteMinder administrator and click Next.

The information you enter must be defined at the Policy Server where the trusted host registers. This screen also includes an optional check box for enabling shared secret rollover.
 - b. Specify the name of the Trusted Host and the Host Configuration Object and click Next.

The name you enter for the trusted host must be unique. The name for the Host Configuration Object must already be defined at the Policy Server where the trusted host is registered.
 - c. Enter the IP address of the Policy Server where you want to register the trusted host and click Add. Click Next.

Note: Include a port number for the Policy Server. For example, 121.111.12.11:44442.
 - d. Specify the name and location of the host configuration file, SmHost.conf. The wizard lists the default location. Click Next.

- e. Specify the name of the Agent Configuration Object and click Next. The Agent Configuration Object that you enter must already be defined at the Policy Server where the trusted host is registered.

Enter the name of the ACO built from the SharePoint2010DefaultSettings ACO template defined in the Policy Server.

- f. Specify the name and location of the Agent Configuration file. The wizard lists the default location. Click Next.

5. Enter the following information for the Apache web server:

- Server name, in the form *server_name.example.com*.
- Web server administrator email address, in the form *admin@example.com*.
- HTTP port number. The default is 80.
- HTTPS (SSL) port number. The default is 443.

Note: On Solaris or Linux, an additional screen prompts for the name of the user under which Tomcat and Apache run. This user cannot be root.

6. Enter the following configuration information for the Claims Search Web Service:Claims WS HTTP Port.

- Claims WS SSL Port.

Note: No default values are provided for the Claims WS HTTP and SSL Ports. However, use a port that is free, which Tomcat can use to host the web application. For UNIX and Linux, the ports must be greater than 1024. Nobody account works with ports above 1024.

7. (Optional) Select the check box if you want to enable the Agent for SharePoint.

8. Review the Configuration Summary.

9. Click Install.

The files are installed.

10. Click Done to exit the wizard.

Note: If you run the Configuration Wizard again for any reason, SSL must be reinitialized. The installer contacts the Policy Server and attempts to register the Trusted Host to create the host configuration file. If trusted host registration does not succeed, the Agent for SharePoint cannot contact the Policy Server and operate properly.

More information:

[Gather CA SiteMinder® Agent for SharePoint Configuration Wizard Information](#) (see page 92)

[CA SiteMinder Agent for SharePoint Configuration Wizard Information Worksheet](#) (see page 417)

Confirm that the Agent for SharePoint Is Functioning

After you install the Agent for SharePoint, but before changing the proxy rules, you can verify that the server is functioning. You can request index.html file by using the server and port number you specified during installation. For example, if you installed the Agent for SharePoint on server1.example.com and selected port 88 for HTTP communication, you can request the following URL with a browser:

`http://server1.example.com:88`

If the Agent for SharePoint is working properly, the request redirects to the main CA website (www.ca.com). The default proxy rules file specifies this URL for all redirects.

Set a Basic Proxy Rule for the Agent for SharePoint

The Agent for SharePoint operates as a proxy-based solution. To protect your SharePoint resources, edit the default proxy rules file so that the Agent for SharePoint points to one of the following locations:

- [A hardware load balancer that redirects incoming requests to multiple web front ends associated with multiple SharePoint servers in a SharePoint server farm](#) (see page 22).
- [A single web front end associated with multiple SharePoint servers in a SharePoint server farm](#) (see page 21).

Follow these steps:

1. Open the following file on your Agent for SharePoint with a text editor:

Agent-for-SharePoint_home\proxy-engine\conf\proxyrules.xml

Important: Do not modify any other configuration files or settings unless explicitly told to do so by CA support personnel.

2. Locate the following line:

```
<nete:forward>http://www.ca.com$0</nete:forward>
```

3. Edit the previous line with one of the following values:

- The URL of your hardware load balancer. This hardware load balancer operates between your Agent for SharePoint server and the SharePoint servers.
- The URL of your single web front end. In this context, this web front end (WFE) refers a web server that operates in front of your "back end" SharePoint servers.

If the URL is sharepoint.example.com, edit the line to match the following example:

```
<nete:forward>http://www.sharepoint.example.com$0</nete:forward>
```

Note: The proxyrules.xml file used by the Agent for SharePoint supports redirection to one URL. The Agent for SharePoint does *not* provide any built-in load-balancing functions.

4. Save the file and close your text editor.

The proxy rule is set.

More information:

[Virtual Host Configurations Supported by the Agent for SharePoint](#) (see page 311)

Enable Support for Dynamic Policy Server Clusters for the CA SiteMinder Agent for SharePoint

The CA SiteMinder Agent for SharePoint supports dynamic Policy Server clusters. These dynamic Policy Server clusters automatically report when individual Policy Servers are added to or removed from a cluster. A restart of the CA SiteMinder Agent for SharePoint is not required.

Follow these steps:

1. Use a text editor to open the following file:

Agent - for - SharePoint_home\proxy-engine\conf\defaultagent\SmHost.conf

2. Locate the following line:

enabledynamichco="NO"

3. Change the previous line to match the following example:

enabledynamichco="YES"

4. Save the file and close the text editor.
5. Restart the CA SiteMinder Agent for SharePoint.

Support for dynamic policy servers is enabled. The CA SiteMinder Agent for SharePoint automatically detects changes to Policy Server clusters.

More information:

[How to Start and Stop the Agent for SharePoint](#) (see page 108)

Install the Policy Server Signing Certificate on your CA SiteMinder Agent for SharePoint

The CA SiteMinder Agent for SharePoint uses an embedded Apache web server. Install the Policy Server signing certificate you want to use for your SharePoint Connection. We recommend using a certificate signed by a Certificate Authority. After copying the certificate and related key files to your CA SiteMinder Agent for SharePoint, edit the configuration file for the embedded Apache web server.

Follow these steps:

1. Copy the certificate files and related key files to your Agent for SharePoint.
2. Open the following file with a text editor:

Agent-for-SharePoint_home\httpd\conf\extra\httpd-ssl.conf

3. Edit the following directives in the file to point to your certificate and related key files (respectively):
 - SSLCertificateFile
 - SSLCertificateKeyFile

Assign Permissions for Log Files and Directories on UNIX/Linux

On the UNIX or Linux operating environments, the user account under which the Agent for SharePoint runs requires permissions to create log files.

After running the Agent for SharePoint configuration wizard on a UNIX or Linux operating environment, grant the user account the permissions shown in the following table:

Grant these permissions:	To these directories:
Read, Execute	<i>Agent-for-SharePoint_home</i> directory and all subdirectories
Write	<i>Agent-for-SharePoint_home</i> /proxy-engine/logs

Manage SharePoint Connections Using the SharePoint Connection Wizard

The SharePoint connection wizard takes you through the process of configuring and managing SharePoint connections with CA SiteMinder®.

Before running the wizard, gather the following information:

Policy Server Name

Specifies the Policy Server name or IP address.

Example: *host_name:port_number*

Note: Specify the Administration port number if the port number is different from the default port number 44444.

Username

Specifies the Policy Server administrator username.

Password

Specifies the Policy Server administrator password.

Agent Name

Specifies the name of the 4.x-compatible Agent object on your Policy Server. The connection with the Policy Server is established using the details given in the Agent Name.

Shared Secret Key

Specifies the shared secret key that is associated with the 4.x-compatible Agent object on your Policy Server.

Select a domain

Specifies the name of the policy domain you created in the Policy Server to protect your SharePoint resources.

Name

Specifies a name for the SharePoint connection. This name is also used as the file name of the PowerShell script that the wizard creates.

Note: Use a unique name across all Resource Partners and SharePoint connections.

Authentication URL

Specifies the *port number* that is associated with the predefined protected URL which the SharePoint connection wizard adds automatically. When users try accessing a protected SharePoint resource without a SiteMinder session, they are redirected to the Authentication URL.

If you are using a default port number (such as 80 for HTTP or 443 for HTTPS), delete the <port> setting from this field.

Note: We recommend using HTTPS on production environments and pages which handle user credentials, such as login pages.

SharePoint Realm

Specifies a name for a SharePoint realm that uniquely identifies this connection between SiteMinder and SharePoint. This name is used to create the trusted identity provider.

Limits: Unique value across all SharePoint servers, farms and within the SiteMinder environment. This value cannot be used with any other identity providers.

Skew Time

Specifies the number of seconds used as a time difference between the Policy Server (token producer) and the SharePoint server (token consumer). This skew time accommodates for SharePoint connections using clocks that are acting as an account partner but are not synchronized with the Policy Server.

Note: This setting also affects the frequency of the [SAML autopost operation](#) (see page 104).

Limits: Positive integers.

Validity Duration

Specifies the number of seconds for which a session remains valid. If the validity duration expires, a logout message is generated. The user that is associated with the invalid session is logged out.

Note: This setting also affects the frequency of the [SAML autopost operation](#) (see page 104).

Signing Alias

Specifies the alias that the key store uses to identify the private key that is associated with the certificate your Policy Server uses to sign the tokens.

Note: We recommended verifying that the private key exists in the central key store before you specify its associated alias in this field. Open the Administrative UI, and then click Infrastructure, X.509 Certificate Management, Trusted Certificates and Private Keys for a list of certificates and their aliases.

Protection Level

Specifies the protection level that is assigned to the resource partner object the connection wizard creates. This protection level setting must be equal to or lower than the protection level assigned to the authentication scheme that protects your SharePoint resources.

Limits: 1-1000 (higher numbers indicate a higher protection level).

Identifier Claim Name

Specifies name of the attribute mapping in your user directory which identifies the unique value that is associated with each user.

Example: useridentifier

Directory Attribute

Specifies the directory attribute in your directory that is associated with the specified Identifier Claim name.

Example: (LDAP directory) uid

Example: (Active directory) sAMAccountName

Attribute

Specifies an attribute name for one of the following claim types:

- Group based
- Role based

For multi-valued attributes, prefix *FMATTR*:

Example: (group-based claim) smusergroups

Example: (role-based claim) userrole

Example: (multiv-alued attributes) FMATTR:LastName

Claim Type

Specifies an attribute value that is associated with the specified attribute name.

For group-based claims, use the friendly role of your groups. The people picker in SharePoint displays the description and distinguished name (DN) of the group. Permissions are tied to the DN of the group, not the friendly name.

Example: (LDAP directory group-based claim) description

Example: (LDAP directory role-based claim) employeeType

Example: (Active Directory group-based claim) name

Example: (Active Directory role-based claim) countryCode

Enabled SignOut

Indicates if the single log out feature is enabled for the associated cleanup URLs and the associated confirm URLs.

CleanUp URL

Specifies the URLs of the cleanup pages for the single log out feature.

Limits: Separate multiple URLs with a semicolon (;)

Confirm URL

Specifies the URLs of the confirmation pages for the single log out feature.

Limits: Separate multiple URLs with a semicolon (;)

Prerequisites for Using the SharePoint Connection Wizard

Before you run the SharePoint Connection Wizard, perform the following steps:

- Verify that you are using a version of the Policy Server that supports the CA SiteMinder® Agent for SharePoint.
- Create a 4.x Agent in the Policy Server Administration UI to enable the Connection Wizard to communicate with the Policy Server.
- The default port number that the SharePoint Connection Wizard uses to contact the Policy Server is 44444. Specify the Administration Service Port number in the Policy Server Management Console if different from the default port number.
- Verify that a policy domain exists on your Policy for the SharePoint resources you want to protect. Verify that the directory containing your SharePoint users is associated with the same policy domain. The SharePoint connection wizard requires the name of the policy domain.
- Identify the required inputs for the SharePoint Connection Wizard by using the Information worksheet.
- Verify that the certificate you want to use for the SharePoint Claims provider is installed on your Agent for SharePoint.

More information:

[Locate the CA SiteMinder® Agent for SharePoint Platform Support Matrix](#) (see page 420)
[SharePoint Connection Wizard Information Worksheet](#) (see page 418)

Alternate Connection Wizard Method to Help Resolve Firewall Issues

If you experience firewall issues when you try to run the connection wizard, verify that port 44444 is open on your Policy Server.

If your Policy Server uses the *same operating environment* as your Agent for SharePoint, you can copy the SharePoint connection wizard executable file to your Policy Server. Then execute the connection wizard on your Policy Server instead.

Copy the appropriate connection wizard executable file for the operating environment of your Policy Server from the following list:

- (Windows) ca-spconnect-12.0-version-win32.exe
- (Solaris) ca-spconnect-version-sol.bin
- (Linux) ca-spconnect-version-rhel30.bin

More information:

[Edit a SharePoint Connection using the SharePoint Connection Wizard](#) (see page 107)

[Delete a SharePoint Connection](#) (see page 410)

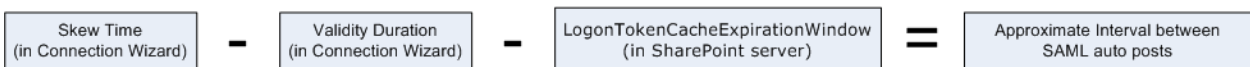
SAML Autopost Frequency

The following settings determine the frequency at which a SAML autopost operation occurs in your SiteMinder and SharePoint environments:

- Skew time (set in the SharePoint Connection wizard)
- Validity duration (set in the SharePoint Connection wizard)
- Logon Token Cache Expiration window (set in SharePoint)

If these settings create a short interval, pop-up windows related to the autopost operation appear. If these settings create a longer interval, inactive users remain logged in for longer periods than the security policies of your organization prefer.

The following illustration describes the relationships among components that affect how often the SAML autopost occurs:



The following table provides some examples of how changes in the Login Cache Token value on SharePoint change how often the SAML autopost occurs:

SiteMinder				SharePoint	Approximate Time Between SAML Auto Post Operations
Realm Idle Timeout	Realm Max Timeout	Validity Period	Skew Time	Logon Token Cache Expiration Window	
1 hour	1 hour	4400 seconds (1 hour 13 minutes)	10 seconds	10 minutes	63 minutes
1 hour	1 hour	4400 seconds (1 hour 13 minutes)	10 seconds	5 minutes	68 minutes

When the Logon Token Cache Expiration Window setting in SharePoint is lower, the SAML autopost operation occurs less often. However, inactive users could possibly remain logged in.

Note: For more information about how to disable FedAuth cookies in SharePoint 2010, go to the [technet blogs](#) website, and then search for the following phrase:

"Setting the Login Token Expiration Correctly for SharePoint 2010 SAML Claims Users"

Create a SharePoint Connection

The Agent for SharePoint uses a connection wizard to define the connection parameters that are used when CA SiteMinder® communicates with your SharePoint server. The connection wizard does following tasks:

- Configures the connection between your Agent for SharePoint and the Policy Server.
- Creates a Windows PowerShell script that you modify and run on your SharePoint central administration server to create a trusted identity provider.

Follow these steps:

1. Perform the following:

- (Windows)

- a. Navigate to the following directory:

Agent - for - SharePoint_home/sharepoint_connection_wizard

- b. Right-click the executable and select Run as administrator.

The SharePoint Connection wizard starts.

- (Unix)

- a. Navigate to the following directory:

Agent - for - SharePoint_home/sharepoint_connection_wizard

- b. Enter one of the following commands:

- Solaris: `sh ./ca-spconnect-12.0-sp3-sol.bin`

- Linux: `sh ./ca-spconnect-12.0-sp3-rhel30.bin`

The SharePoint Connection wizard starts.

2. Complete the wizard using the information you gathered.
3. Click Install.

The Save Complete screen appears and shows location of your PowerShell script. The PowerShell script is created in the following directory:

Agent - for - SharePoint_home/sharepoint_connection_wizard/

The connection wizard uses the connection name that you specified (in Step 8) as the name of the PowerShell script. For example, if you specify `my_sharepoint_connection` for a connection name in the connection wizard, then name of the PowerShell script is `my_sharepoint_connection.ps1`.

4. Click Done.

The connection wizard closes.

More information:

[SAML Autopost Frequency](#) (see page 104)

[Alternate Connection Wizard Method to Help Resolve Firewall Issues](#) (see page 104)

Edit a SharePoint Connection using the SharePoint Connection Wizard

Follow these steps:

1. Perform the following:
 - (Windows)
 - a. Navigate to the following directory:
Agent - for - SharePoint_home/sharepoint_connection_wizard
 - b. Right-click the executable and select Run as administrator.
The SharePoint Connection wizard starts.
 - (Unix)
 - a. Navigate to the following directory:
Agent - for - SharePoint_home/sharepoint_connection_wizard
 - b. Enter one of the following commands:
 - Solaris: `sh ./ca-spconnect-12.0-sp3-sol.bin`
 - Linux: `sh ./ca-spconnect-12.0-sp3-rhel30.bin`The SharePoint Connection wizard starts.
2. Click Next.
The Login Details screen appears.
3. Enter the following login details to connect to the Policy Server.

Policy Server Name

Specifies the Policy Server name or IP address.

Username

Specifies the Policy Server administrator username.

Password

Specifies the Policy Server administrator password.

Agent Name

Specifies the Agent-4x. The connection with the Policy Server is established using the details given in the Agent Name.

Shared Secret Key

Specifies the shared secret key associated with the Agent.

4. Click Next

The Select Action screen appears.

5. Select Edit a SharePoint Connection option.

6. Click Next.

The SharePoint Connection Properties screen appears.

7. Make the required changes in SharePoint Connection Properties, Name IDs, and Add Attributes screen.

8. Click Install in the Commit Details screen.

The Save Complete screen appears.

9. Click Done.

The partnership details are saved, the SharePoint Connection is modified, and the wizard closes.

More information:

[SharePoint Connection Wizard Information Worksheet](#) (see page 418)

[Alternate Connection Wizard Method to Help Resolve Firewall Issues](#) (see page 104)

How to Start and Stop the Agent for SharePoint

Starting or stopping the Agent for SharePoint involves the following separate procedures:

1. [Changing the value of EnableWebAgent in the WebAgent.conf file](#) (see page 109).
2. [Changing the state of the related services on the computer running the Agent for SharePoint](#) (see page 110).

Change the Value of the EnableWebAgent Parameter

Change the value of the EnableWebAgent parameter to accomplish either of the following tasks:

- Start the Agent for SharePoint when the related services start.
- Stop the Agent for SharePoint when the related services start.

Follow these steps:

1. Open the following file with a text editor:

Agent - for - SharePoint_home\proxy-engine\conf\defaultagent\WebAgent.conf

2. Locate the following line:

EnableWebAgent="NO"

3. Change the value inside the quotation marks to *one* of the following values:

- YES to start the Agent for SharePoint after the services start. Your resources are protected.
- NO to stop the Agent for SharePoint after the services start. Your resources are *not* protected.

4. [Change the state of the related services on your Agent for SharePoint](#) (see page 110).

Change the States of the Services on your Agent for SharePoint

You can change the states of the related services on your Agent for SharePoint.

Note: To start or stop your Agent for SharePoint, [change the value of the EnableWebAgent parameter first](#) (see page 109).

Follow these steps:

1. To change the states of the related services, select *one* of the following procedures:
 - For Windows operating environments, go to Step 2.
 - To *start* the Agent for SharePoint on UNIX operating environments, go to Step 3.
 - To *stop* the Agent for SharePoint on UNIX operating environments, go to Step 4.
2. For Windows operating environments, do the following steps:
 - a. From the Windows Start menu navigate to Administrative Tools, Services.
The Services dialog appears.
 - b. Scroll down the list of services and select SiteMinder Agent for SharePoint.
 - c. From the Action menu, select All Tasks and select the command that you want.
 - d. Repeat Step b for SiteMinder Agent for SharePoint Proxy Engine.
The states of the services and Agent for SharePoint are changed.
3. To start the Agent for SharePoint on UNIX operating environments, do the following steps.
 - a. Log in as a root user.
 - b. Navigate to the following directory:
Agent-for-SharePoint_home/proxy-engine
 - c. Run the following command:

`./sps-ctl start`
The service and the Agent for SharePoint start. The Agent for SharePoint stops or starts according to the [value you set in the EnableWebAgent parameter](#) (see page 109).
4. To stop the Agent for SharePoint on a system running UNIX, do the following steps:
 - a. Navigate to the following directory:
Agent-for-SharePoint_home/proxy-engine
 - b. Run the following command:

`./sps-ctl stop`
The service and the Agent for SharePoint stop.

Chapter 8: Configure SharePoint

This section contains the following topics:

[How to Configure SharePoint for the Agent for SharePoint](#) (see page 113)
[Permissions Required for Trusted Identity Provider and Claims Provider](#) (see page 114)
[How to Create Alternate Access Mappings](#) (see page 115)
[How to Configure the Trusted Identity Provider](#) (see page 123)
[Adding Claims to Trusted Identity Providers](#) (see page 135)
[Removing Claims from Trusted Identity Providers](#) (see page 143)
[Configure the Authentication Providers](#) (see page 146)
[How to Disable Client Loopback](#) (see page 149)
[Add and Grant Permission to CA SiteMinder® Users](#) (see page 150)
[Manage User Profiles](#) (see page 151)

How to Configure SharePoint for the Agent for SharePoint

Configuring your SharePoint servers for the Agent for SharePoint involves several separate procedures.

Follow these steps:

1. Configure Alternate Access Mappings.
2. Configure the Trusted Identity Provider.
3. [Configure Authentication Providers](#) (see page 146).
4. [Disable Client Loopback](#) (see page 149).
5. [Add SiteMinder Users to SharePoint.](#) (see page 150)
6. [Manage User Profiles.](#) (see page 151)

Permissions Required for Trusted Identity Provider and Claims Provider

Users who create the trusted identity provider and install or configure the SharePoint claims provider need the following permissions:

User account permissions

User accounts require the following privileges:

- Domain user account.
- Member of Local administrator group on each SharePoint server in the farm (except for the SQL Server and SMTP server)
- Access to the SharePoint 2010 server databases.

Setup User Account

The setup user account requires the following permissions:

- Member of the WSS_ADMIN_WPG Windows security group.
- Member of the IIS_WPG role group.

Database permissions

The following database permissions are required:

- db_owner on the SharePoint Server 2010 server farm configuration database.
- db_owner on the SharePoint Server 2010 Central Administration content database.

PowerShell scripts for Claims Provider

Running the PowerShell scripts for the Claims Provider requires the following permissions:

- Local administrator on all SharePoint web front end (WFE) servers.
- Access (read/write) to the configuration database.

Note: The preceding permissions apply when the user is not an Administrator or not part of an Administrator group.

How to Create Alternate Access Mappings

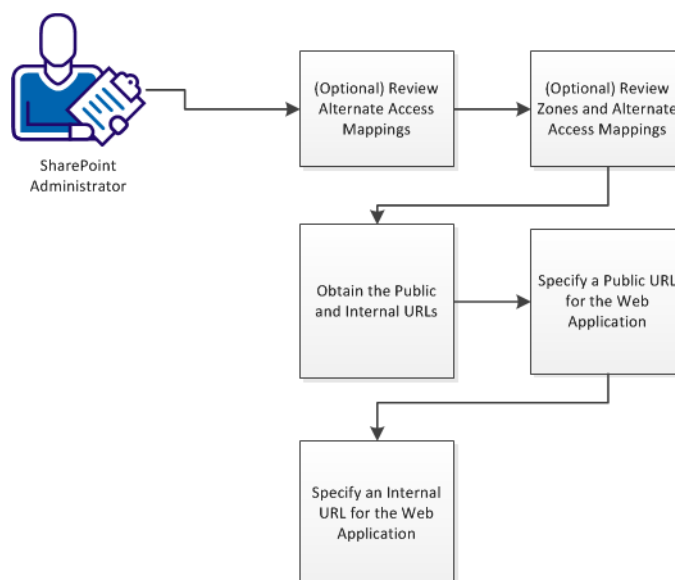
Alternate access mappings can direct users who request an external URL to a specific web application on your SharePoint servers. Create alternate access mappings between your external URLs and the web applications on your SharePoint servers.

The Agent for SharePoint uses proxy rules in a similar fashion. Users who authenticate through the Agent for SharePoint are redirected to the internal web application hosted in SharePoint.

Important! The proxy rules in the Agent for SharePoint must match the alternate access mappings for your SharePoint web application.

The following graphic describes how to create alternate access mappings:

How to Create Alternate Access Mappings



Follow these steps:

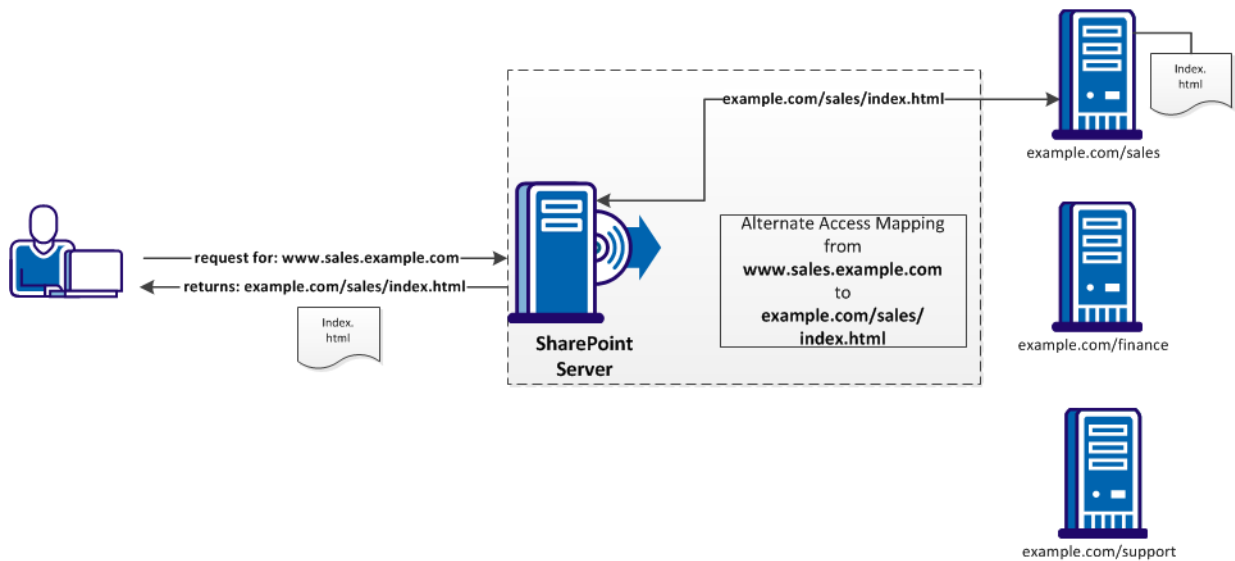
1. (Optional) Review the following topics that are related to SharePoint administration:
 - [Alternate access mappings](#) (see page 116).
 - [Zones and alternate access mappings](#) (see page 117).
2. Obtain the [public and internal URLs](#) (see page 119).
3. [Specify a public URL for the web application](#) (see page 121)
4. [Specify an internal URL for the web application](#) (see page 122).

Alternate Access Mappings

SharePoint central administration servers let you create alternate access mappings between external and internal URLs.

- External URLs are those URLs that your customers, partners, or people outside of your organization access. For example, your customers and partners could log in to your network using `www.login.example.com`.
- Internal URLs correspond to the location of the web application in your SharePoint environment. For example, the login server that processes logins could be named `login123.example.com`.

An alternate access mapping creates an association in SharePoint between your external login URL and the login server in the back end. For example, the SharePoint server directs all the requests for `www.example.com` to the `login123.example.com` server as shown in the following graphic:

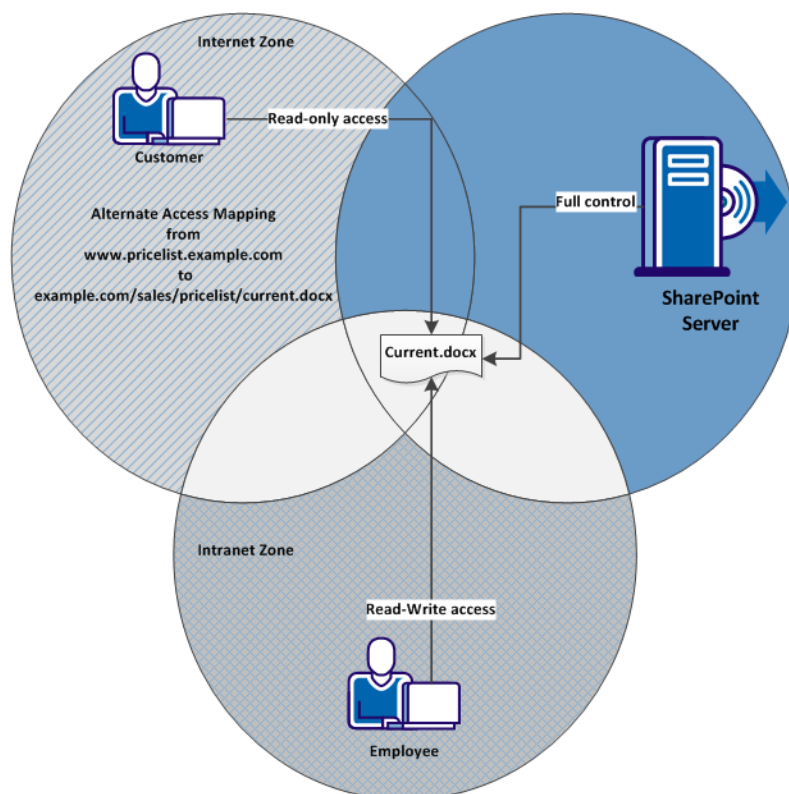


Zones and Alternate Access Mappings

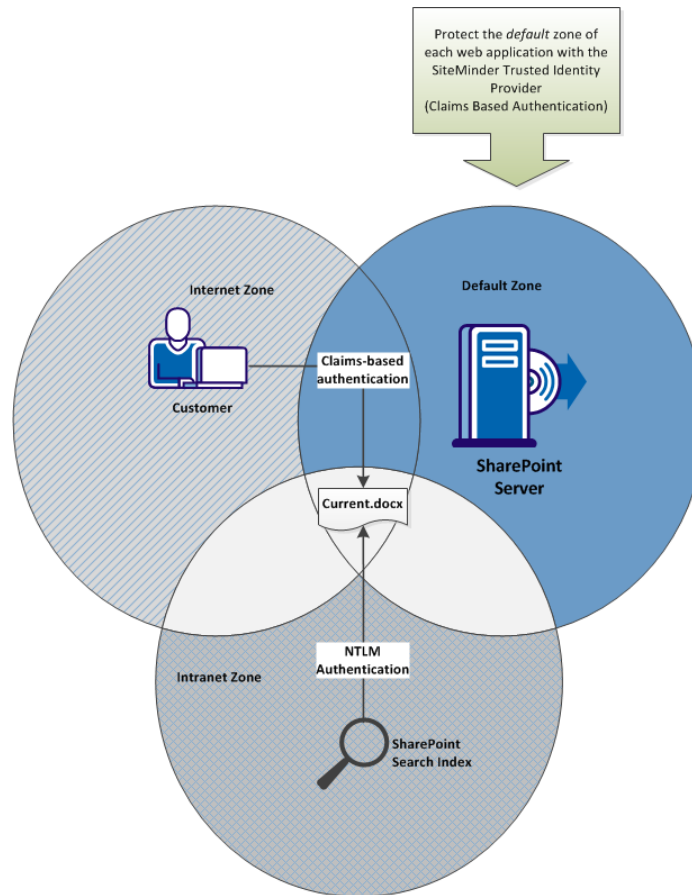
Alternate access mappings also support zones. Zones let you configure different access paths to a single web application on your SharePoint server. Creating alternate access mappings across different zones can accomplish the following goals:

- Create different URLs for the same web application. For example, you could have one URL for external users and a different URL for internal users that both point to the same web application.
- Allow customers read-only access to documents that are hosted on your SharePoint server, while granting full access to your employees.
- Require secure (HTTPS) connections to a web application for external visitors, while allowing employee access to the web application using HTTP.
- Index the content of your web application using the SharePoint search index (which requires NTLM access), while requiring another authentication method for users.

The following graphic describes how different zones permit different levels of access to the same document for external customers and internal employees:



The following graphic describes how multiple authentication methods apply to the same document by extending the associated web application to multiple zones:



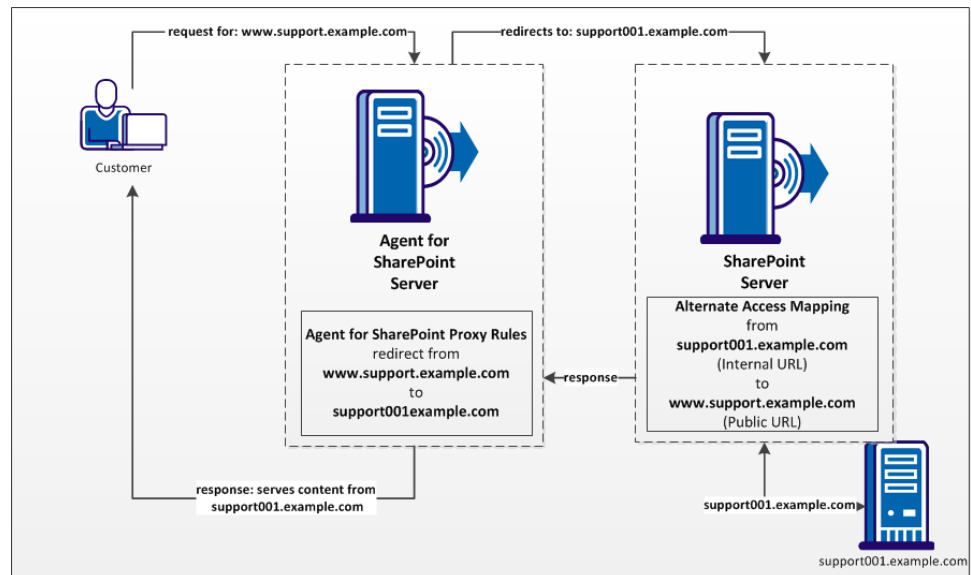
To accommodate the SharePoint search index, the web application must be extended into one zone that uses NTLM authentication.

Obtain the Public and Internal URLs

The Agent for SharePoint runs on a proxy-server. The Agent for SharePoint forwards requests to the web applications in your SharePoint environment using proxy rules. These proxy rules direct traffic from the public URL (the server hosting Agent for SharePoint) to your SharePoint web applications (the internal URLs).

For example, customers who access support.example.com are authenticated by the Agent for SharePoint. Next the user is redirected to a SharePoint web application hosted on a server named support001.example.com. The web application serves *the content* from support001.example.com back to the user who requested the support.example.com page.

The following graphic describes the relationship between proxy rules and alternate access mappings from the previous example:



Follow these steps:

1. Obtain the external URLs that are hosted on your Agent for SharePoint server from your network administrator. In this scenario, the URL `www.support.example.com` is hosted on the Agent for SharePoint server.
2. Log in to the server hosting the Agent for SharePoint.
3. Create *a copy* of the following file:

Agent-for-SharePoint_home\proxy-engine\conf\proxyrules.xml

4. Open the copy that you created in Step 3 with a text editor.
5. Locate the line containing the `nete:forward` tags, as shown in the following example:

```
<nete:forward>http://server2.company.com$1</nete:forward>
```

Note: In a typical environment, the URL in the Step 5 matches the Internal URL for your SharePoint web application.

6. Record the public and internal URLs for future reference. You need these public and internal URLs to create your alternate access mappings.
7. Repeat Steps 4 through 6 for to obtain any additional Internal URLs for other web applications.

Specify a Public URL for the Web Application

The public URL is an external URL through which your customers or external users connect to your organization. The public URL appears in the web browsers of your users.

When you use the Agent for SharePoint in front of your SharePoint server farm, use the URL of the server hosting your Agent for SharePoint as the public URL.

Important! The proxy rule settings of the Agent for SharePoint must match your alternate access mappings.

This procedure describes creating alternate access mappings for the default zone. Adding another type of authentication to a single internal URL with an alternate access mapping is described in a separate scenario.

Follow these steps:

1. Click Start, Programs, Microsoft SharePoint 2010 Products, SharePoint 2010 Central Administration.

The Central Administration home page appears.

2. Click Application Management.

The Application Management page appears.

3. Click Configure alternate access mappings.

The Alternate Access Mappings page appears with a list of available web applications.

Note: If the web application that you want is not listed, click the Alternate Access Mapping Collection drop-down list. Pick the web application that you want.

4. Click Edit Public URLs.

The Edit Public URLs page appears.

5. Locate the field for the zone that contains the internal URL for your web application. For example, if you created a web application named `http://support001:27975` in the default zone, then locate the Default (zone) field with that URL.

6. Replace the internal URL in Step 5 with the public URL that you want. For example, if you are mapping from the internal URL `http://supportp001:2975` to `support.example.com`, then replace the internal URL in the field with `support.example.com`.

7. Click Save.

Specify an Internal URL for the Web Application

This procedure allows the SharePoint Administrator to map the public URL (<http://support.example.com>) to the SharePoint internal URL (<http://support001.example.com>).

Follow these steps:

1. Click Start, Programs, Microsoft SharePoint 2010 Products, SharePoint 2010 Central Administration.

The Central Administration home page appears.

2. Click Application Management.

The Application Management page appears

3. Click Configure alternate access mappings.

The Alternate Access Mappings page appears with a list of available web applications.

4. Click Add Internal URLs.

The Add Internal URLs page appears.

Note: If the mapping collection that you want edit does not appear, then select one from the Alternate Access Mapping Collection list.

5. Enter the internal URL as <http://support001.example.com> in the Add Internal URL section, in the URL protocol, host, and port field.

6. Click Save.

The Alternate Access Mappings page appears with the saved settings. The following table describes how the alternate access mappings appear in SharePoint using the examples in this procedure:

Internal URL	Zone	Public URL for the Zone
http://support001.example.com	Default	http://support.example.com
http://support.example.com	Default	http://support.example.com

How to Configure the Trusted Identity Provider

The Windows Identity Framework in SharePoint 2010 supports multiple authentication providers. Create a Trusted Identity Provider in SharePoint to establish runtime integration with CA SiteMinder® Agent for SharePoint. To configure the trusted identity provider follow these steps:

1. [Copy the Policy Server signing certificate to the SharePoint central administration server](#) (see page 123).
2. [Copy the PowerShell script to the SharePoint central administration server](#) (see page 124).
3. [Modify the PowerShell script](#) (see page 125).
4. [\(Optional\) Add additional certificate authority certificates to the PowerShell script](#) (see page 131).
5. [Create the trusted identity provider](#) (see page 133).
6. [\(Optional\) Verify that the Trusted Identity Provider is registered](#) (see page 134).

More information:

[SharePoint 2010 Federation Worksheet](#) (see page 419)

Copy the Policy Server Signing certificate to the SharePoint Central Administration Server

The Policy Server signing certificate that you exported from your key store on a Policy Server is required to create a trusted identity provider. This certificate lets the SharePoint claims provider verify the authentication claims (tokens) that the Policy Server sends.

Follow these steps:

1. Navigate to the directory on your Policy Server to which you exported your certificate from the central key store.
2. Locate the Policy Server signing certificate file that you exported, and then copy it to a directory on your SharePoint central administration server.

Copy the Powershell Script to the SharePoint Central Administration Server

The PowerShell script created by the SharePoint connection wizard on your Agent for SharePoint host is required to create a trusted identity provider. Copy it from your Agent for SharePoint host to your SharePoint central administration server.

Follow these steps:

1. Navigate to the following directory on your Agent for SharePoint server:
Agent-for-SharePoint_home\sharepoint_connection_wizard
2. Locate the PowerShell script created by the SharePoint connection wizard. The script uses the connection name you chose while running the wizard as the file name. For example, if your connection name was my_connection, the name of the script is my_connection.ps1.
3. Copy the PowerShell script to a directory on your SharePoint central administration server.

Modify the PowerShell Script

To create a trusted identity provider on your SharePoint central administration server, edit the PowerShell script to include the following information about your SharePoint environment:

- The full path to the root certificate (typically from a third-party Certificate Authority) that signed your certificate.
- Create a trusted root authority in SharePoint for the certificate authority which signed your certificate.
- The full path to your signing certificate.
- Friendly names for each of the claim mappings.
- The SharePoint realm name (to identify the trusted identity provider).

Note: This value appears in SharePoint Central Administration under the list of available trusted identity providers.

- A friendly description for the trusted identity provider.

The specific modifications to the PowerShell script vary according to the type of certificates you want to use with your CA SiteMinder® trusted identity provider. The following scenarios exist:

- You are using a certificate that is signed by an external certificate authority, and the certificate authority is *not* trusted by your SharePoint server.
- You are using a self-signed certificate and the certificate authority is *not* trusted by your SharePoint server.
- You are using a certificate, and the certificate authority is trusted by your SharePoint server. Check with your SharePoint administrator to confirm that the proper certificate authority is trusted.

Follow these steps:

1. Use the previous list to determine which scenario applies to your situation.
2. Perform the appropriate procedure from the following list:
 - [Modify the PowerShell script for certificates that are signed by an external certificate authority](#) (see page 126).
 - [Modify the PowerShell script for un-trusted self-signed certificates](#) (see page 128).
 - [Modify the PowerShell script for certificates that are issued by a trusted certificate authority](#) (see page 130).

Modify the PowerShell Script for Certificates Signed by an Un-Trusted External Certificate Authority

If your signing certificate is signed by an external certificate authority, modify the PowerShell script to do the following tasks:

- Import the certificate authority certificate (root certificate) into SharePoint.
- Create a SharePoint trusted root authority that is based on the certificate authority certificate.
- Import the signing certificate.

Follow these steps:

1. Open the PowerShell script with any text editor.
2. Locate the following text:

`"<full path to Root certificate file>"`
3. Replace the previous text with the full path to your root certificate. For example, if the full path to your certificate is
`C:\certificates\sharepoint\certificate_authority_certificate.cer`, the updated line matches the following example:

`"C:\certificates\sharepoint\certificate_authority_certificate.cer"`
4. Locate the first occurrence of the following text:

`<Trusted root authority name>`
5. Replace the previous text with a friendly name for the new trusted root authority in SharePoint. For example, if the name you want is SPCAAuth, the updated line matches the following example:

`"SPCAAuth"`
6. Locate the following text:

`"<full path to Signing certificate file>"`
7. Replace the previous text with the full path to your Signing certificate. For example, if the full path to your certificate is
`C:\certificates\sharepoint\signing_certificate.cer`, the updated line matches the following example:

`"C:\certificates\sharepoint\signing_certificate.cer"`
8. Locate the second occurrence of the following text:

`<Trusted root authority name>`
9. Replace the previous text with a friendly name for the new trusted root authority in SharePoint. For example, if the name you want is SPSigningAuth, the updated line matches the following example:

`"SPSigningAuth"`

10. Locate the following text:

"<Name of the trusted identity provider>"

11. Replace the previous text with the name of your SharePoint realm (the realm name follows \$realm = in the PowerShell script). For example, if the name of your SharePoint realm is \$realm="urn:moss2010-wsfed1-casm", the updated line could match the following example:

"moss2010-wsfed1-casm"

12. Locate the following text:

"<Description for the Trusted Identity Provider>"

13. Replace the previous text with a description for your trusted identity provider. For example, if you want to describe the trusted identity provider as "SiteMinder Provider," the updated line could match the following example:

"SiteMinder Provider"

Note: The LDAP directory and Active Directory charts contain additional examples of possible names.

14. If your certificate chain contains *more than one* certificate authority certificate, [add the other certificate authority certificates to the script](#) (see page 131). If your script contains *one* certificate authority certificate, go to the next step.
15. Save your changes and close your text editor.
- The PowerShell script is modified.
16. [Create a trusted identity provider](#) (see page 133).

Modify the PowerShell Script for Un-Trusted Self-Signed Certificates

If you are using a self-signed certificate that is issued by a certificate authority which is not explicitly trusted by your SharePoint server, modify the PowerShell script to do the following tasks:

- Import the certificate authority certificate (root certificate) into SharePoint.
- Create a SharePoint trusted root authority that is based on the certificate authority certificate.
- Import the signing certificate.

Follow these steps:

1. Open the PowerShell script with any text editor.
2. Locate the following text:
`"<full path to Root certificate file>"`
3. Replace the previous text with the full path to your root certificate. For example, if the full path to your certificate is `C:\certificates\sharepoint\certificate_authority_certificate.cer`, the updated line matches the following example:
`"C:\certificates\sharepoint\certificate_authority_certificate.cer"`
4. Locate the first occurrence of the following text:
`<Trusted root authority name>`
5. Replace the previous text with a friendly name for the new trusted root authority in SharePoint. For example, if the name you want is `SPCAAuth`, the updated line matches the following example:
`"SPCAAuth"`
6. Locate the following text:
`"<full path to Signing certificate file>"`
7. Replace the previous text with the full path to your Signing certificate. For example, if the full path to your certificate is `C:\certificates\sharepoint\signing_certificate.cer`, the updated line matches the following example:
`"C:\certificates\sharepoint\signing_certificate.cer"`
8. Locate the second occurrence of the following text:
`<Trusted root authority name>`
9. Replace the previous text with a friendly name for the new trusted root authority in SharePoint. For example, if the name you want is `SPSigningAuth`, the updated line matches the following example:
`"SPSigningAuth"`

10. Locate the following text:

"<Name of the trusted identity provider>"

11. Replace the previous text with the name of your SharePoint realm (the realm name follows \$realm = in the PowerShell script). For example, if the name of your SharePoint realm is \$realm="urn:moss2010-wsfed1-casm", the updated line could match the following example:

"moss2010-wsfed1-casm"

12. Locate the following text:

"<Description for the Trusted Identity Provider>"

13. Replace the previous text with a description for your trusted identity provider. For example, if you want to describe the trusted identity provider as "SiteMinder Provider," the updated line could match the following example:

"SiteMinder Provider"

Note: The LDAP directory and Active Directory charts contain additional examples of possible names.

14. If your certificate chain contains *more than one* certificate authority certificate, [add the other certificate authority certificates to the script](#) (see page 131). If your script contains *one* certificate authority certificate, go to the next step.
15. Save your changes and close your text editor.
- The PowerShell script is modified.
16. [Create a trusted identity provider](#) (see page 133).

Modify the PowerShell Script for Certificates Issued by a Trusted Certificate Authority

If you are using a certificate signed by a certificate authority that is trusted by the SharePoint server, modify the PowerShell script to do the following tasks:

- Skip the step to import the certificate authority certificate.
- Skip the step to create a new SharePoint trusted root authority.
- Import only the signing certificate.

Follow these steps:

1. Open the PowerShell script with any text editor.
2. Comment the first two lines in the PowerShell script, as shown in the following example:

```
$rootcert = New-Object  
System.Security.Cryptography.X509Certificates.X509Certificate2("<full  
path to Root certificate file>")  
#New-SPTtrustedRootAuthority -Name "<Trusted root authority name>"  
-Certificate $rootcert
```

3. Locate the following text:

```
"<full path to Signing certificate file>"
```

4. Replace the previous text with the full path to your Signing certificate. For example, if the full path to your certificate is C:\certificates\sharepoint\signing_certificate.cer, the updated line matches the following example:

```
"C:\certificates\sharepoint\signing_certificate.cer"
```

5. Locate the second occurrence of the following text:

```
<Trusted root authority name>
```

6. Replace the previous text with a friendly name for the new trusted root authority in SharePoint. For example, if the name you want is SPSigningAuth, the updated line matches the following example:

```
"SPSigningAuth"
```

7. Locate the following text:

```
"<Name of the trusted identity provider>"
```

8. Replace the previous text with the name of your SharePoint realm (the realm name follows \$realm = in the PowerShell script). For example, if the name of your SharePoint realm is \$realm="urn:moss2010-wsfed1-casm", the updated line could match the following example:

```
"moss2010-wsfed1-casm"
```

9. Locate the following text:

```
"<Description for the Trusted Identity Provider>"
```

10. Replace the previous text with a description for your trusted identity provider. For example, if you want to describe the trusted identity provider as "SiteMinder Provider," the updated line could match the following example:

"SiteMinder Provider"

Note: The LDAP directory and Active Directory charts contain additional examples of possible names.

11. Save your changes and close your text editor.

The PowerShell script is modified.

12. [Create a trusted identity provider](#) (see page 133).

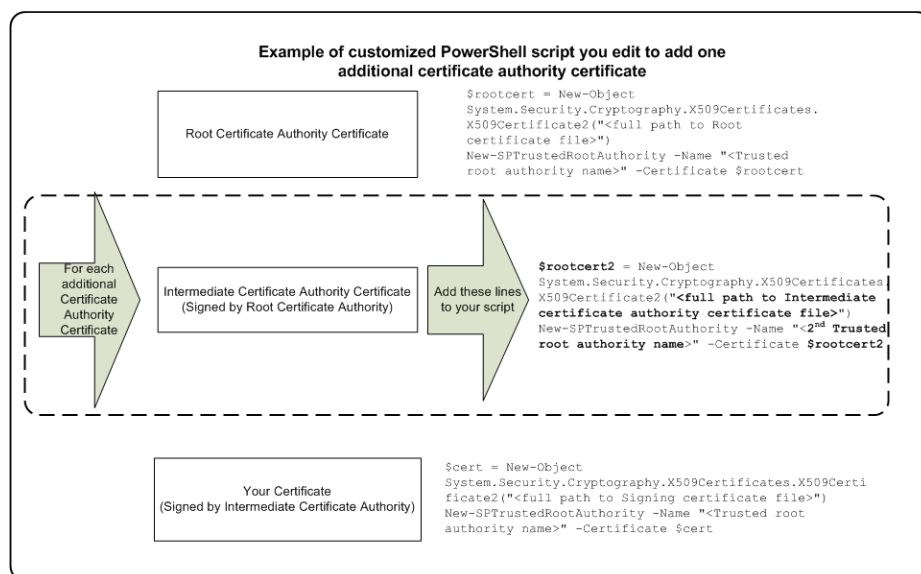
Add Additional Certificate Authority Certificates to the PowerShell Script

The PowerShell script created by the SharePoint connection wizard accommodates the following certificates:

- A certificate authority certificate (also named a root certificate)
- One SSL certificate.

The trusted identity provider requires that all certificates in the certificate chain are included. If an intermediate certificate authority signed your certificate instead, modify the PowerShell script to include both certificate authority certificates.

The following illustration describes the differences between the default PowerShell script, and a PowerShell script that accommodates multiple certificate-authority certificates:



Follow these steps:

1. Copy the following section from your PowerShell script:

```
$rootcert = New-Object  
System.Security.Cryptography.X509Certificates.X509Certificate2("<full path to  
Root certificate file>")  
New-SPTrustedRootAuthority -Name "<Trusted root authority name>" -Certificate  
$rootcert
```
2. Copy the following section from your PowerShell script:
3. Add a new line after the section you copied, and then paste the copied into the new line.
4. Edit the pasted section using the changes shown in the following table as a guide:

Change this value:

\$rootcert

<full path to Root certificate file>

<Trusted root authority name>

To this value:

\$rootcert2

<full path to additional certificate
authority certificate file>Name of the additional trusted root
authority

5. To add additional certificate authority certificates, repeat Steps 1 through 4.
6. Save your changes and close your text editor.
The PowerShell script is modified.
7. [Create a trusted identity provider](#) (see page 133).

Run the Powershell Script to Create a Trusted Identity Provider

Run the modified PowerShell script to create a trusted identity provider on your SharePoint central administration server.

Follow these steps:

1. Click Start, All Programs, Microsoft SharePoint 2010 Products, SharePoint 2010 Management Shell
2. Navigate to the directory containing your edited PowerShell script.
3. Run the script with the following command:

```
.\your_connection_name.ps1
```

For example, if you named your connection "my_sharepoint" when you ran the connection wizard, the command would be `.\my_sharepoint.ps1`.

The trusted identity provider is created.

Verify That the Trusted Identity Provider Is Registered

After running the PowerShell script to create your trusted identity provider, verify that it is registered in your SharePoint central administration server.

Follow these steps:

1. From your SharePoint central administration server, click Start, All Programs, Microsoft SharePoint 2010 Products, SharePoint 2010 Management Shell.

The Microsoft PowerShell command prompt appears.

2. Enter the following command:

```
Get-SPTtrustedIdentityTokenIssuer
```

A list of the trusted identity providers that are configured on the SharePoint central administration server appears.

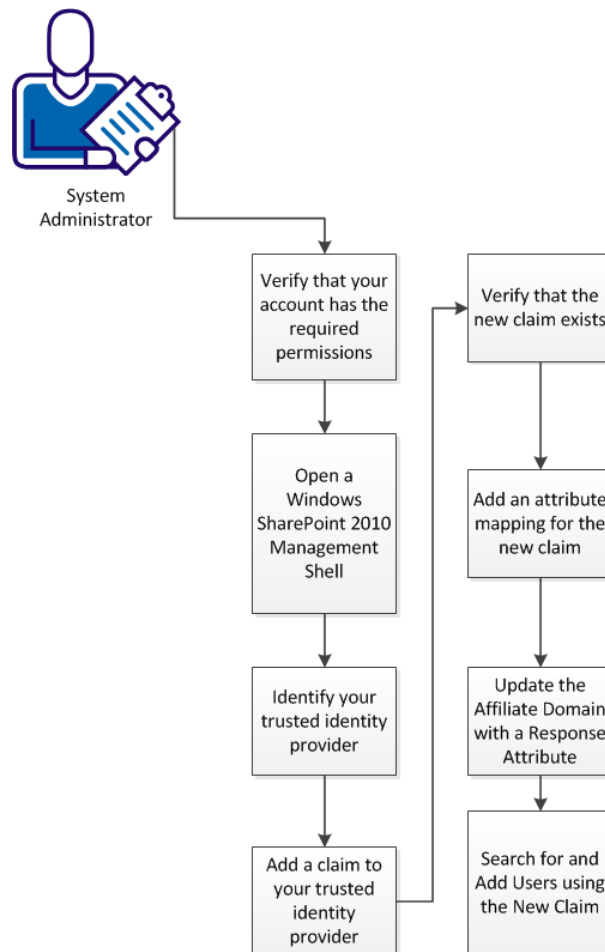
Chapter 9: Adding Claims to Trusted Identity Providers

SharePoint 2010 supports third-party identity providers. These identity providers authenticate and authorize users who request SharePoint resources. A SharePoint administrator configures a trusted identity provider for a SharePoint environment.

Claims are a form of attribute or role, that a user has. Each claim has a name to identify it, and a value that the trusted identity provider verifies by connecting to a user directory.

For example, you can configure claims that correspond to the SamAccountName attribute of an Active Directory server or a uid of an LDAP directory server.

You can add a claim to a CA SiteMinder® trusted identity provider at any time. The following illustration describes the process:



To add a claim to a CA SiteMinder® trusted identity provider, follow these steps:

1. [Verify that your account has the required permissions](#) (see page 137).
2. [Open a SharePoint 2010 Management Shell window on your SharePoint Central Administration server](#) (see page 137).
3. [Identify your CA SiteMinder® trusted identity provider](#) (see page 137).
4. [Add a claim to your trusted identity provider](#) (see page 138).
5. [Verify that the new claim exists.](#) (see page 138)
6. [Add an attribute mapping for the new claim](#) (see page 139).
7. [Update the affiliate domain with a response attribute](#) (see page 140).
8. [Search for and add users using the new claim](#) (see page 142).

Verify that your Account has the Required Permissions

The user account with which you want to modify the CA SiteMinder® trusted identity provider requires certain permissions. Modify the permissions of your user account if it does *not* meet the following conditions:

- An Administrator account.
- A member of the Administrators group.

Add the following privileges to your account:

- Local administrator on all SharePoint web front end (WFE) servers.
- Read/Write access to the configuration database.

Open a SharePoint 2010 Management Shell Window on your SharePoint Central Administration Server

Add claims to your CA SiteMinder® trusted identity provider using the SharePoint 2010 Management shell.

Follow these steps:

1. Log in to your SharePoint Central Administration server.
2. Click Start, All Programs, Microsoft SharePoint 2010 Products, SharePoint 2010 Management Shell.

A SharePoint 2010 management shell command-line window appears.

Identify your Trusted Identity Provider

A SharePoint 2010 environment can have multiple trusted identity providers. Identify your CA SiteMinder® trusted identity provider before modifying any claims that are associated with it.

Follow these steps:

1. Enter the following command to list all of the trusted identity providers:

```
Get-SPTtrustedIdentityTokenIssuer
```

A list of trusted identity providers appears.

2. Locate your CA SiteMinder® trusted identity provider in the list.

Your CA SiteMinder® trusted identity provider is identified.

Add a Claim to your Trusted Identity Provider

Adding a claim to your CA SiteMinder® trusted identity provider involves several steps using the SharePoint 2010 Management Console. This example adds a claim for the last name of a user to the CA SiteMinder® trusted identity provider. Use this example as a guide to add any claim you want to your CA SiteMinder® trusted identity provider.

Follow these steps:

1. Enter the following command to assign the name of your CA SiteMinder® trusted identity provider to a variable:

```
$trusted_identity_provider_variable_name = Get-SPTrustedIdentityTokenIssuer  
-Identity "name_of_siteminder_trusted_identity_provider"
```

2. Enter the following command to add a claim type that is based on the last name of a user:

```
$map2 = New-SPClaimTypeMapping -IncomingClaimType  
"http://schemas.xmlsoap.org/claims/lastname" -IncomingClaimTypeDisplayName  
"role" -LocalClaimType "http://schemas.xmlsoap.org/claims/lastname"
```

3. Enter the following command to associate the new claim type with your CA SiteMinder® trusted identity provider:

```
$map2 | Add-SPClaimTypeMapping -TrustedIdentityTokenIssuer  
$trusted_identity_provider_variable_name
```

The new claim is added to your trusted identity provider.

Verify the New Claim Exists

You can verify the addition of the new claim to your CA SiteMinder® trusted identity provider. This example verifies the addition of a claim for the last name of a user.

Follow these steps:

1. Enter the following command to verify the presence of your new claim:

```
Get-SPTrustedIdentityTokenIssuer
```

A list of trusted identity providers appears.

2. Verify that new claim for your CA SiteMinder® trusted identity provider appears.

Add an Attribute Mapping for the New Claim

Add an attribute mapping for the new claim using the CA SiteMinder® Administrative UI. For this example, an attribute mapping links the claim, such as last name, to a specific attribute in your user directory. For both Active Directory servers and LDAP directories, map the Last Name claim to the sn attribute in your directory.

Follow these steps:

1. Log on to the CA SiteMinder® Administrative UI.
2. Click Infrastructure, Directory, User Directory, Modify User Directory.
A list of user directory connections appears.
3. Click the option button for your user directory, and then click Select.
The Modify User directory page appears.
4. Click Create.
The create attribute mapping page appears.
5. Verify that the Create a new object of type Attribute Mapping option button is selected, and then click OK.
6. Click the name field, and enter the name of the new claim. For example, if your new claim is Last Name, as shown in this example, enter the following text:

Last Name
7. Verify that the Alias option button is selected, and then click the Definition field.
8. Enter the directory attribute that you want to associate with the claim you added. For example, if your new claim is Last Name, as shown in this example, enter the following text:

sn
9. Click OK.
The Modify User directory page appears.
10. Click Submit.
The attribute mapping for the new claim is created.

Update the Affiliate Domain with a Response Attribute

Update the affiliate domain with a response attribute for your new claim. This update requires running the SharePoint connection wizard on the computer hosting your CA SiteMinder® Agent for SharePoint.

This procedure adds the mapping of the new claim to your CA SiteMinder® Policy Server.

Follow these steps:

1. Navigate to the following directory:
`Agent-for-SharePoint_home/sharepoint_connection_wizard`
2. Do *one* of the following procedures:
 - For Windows operating environments, right-click the executable and then select Run as administrator.
 - For Solaris operating environments, enter the following command:
`Solaris: sh ./ca-spconnect-version-sol.bin`
 - For Linux operating environments, enter the following command:
`Linux: sh ./ca-spconnect-version-rhel30.bin`

The wizard starts.

3. Click Next.
The Login Details screen appears.
4. Complete the following fields with the information from your existing CA SiteMinder® settings:

Policy Server Name

Specifies the Policy Server name or IP address.

Username

Specifies the Policy Server administrator username.

Password

Specifies the Policy Server administrator password.

Agent Name

Specifies the Agent-4x. The connection with the Policy Server is established using the details given in the Agent Name.

Shared Secret Key

Specifies the shared secret key that is associated with the Agent.

5. Click Next

The Select Action screen appears.

6. Select Edit a SharePoint Connection option.

7. Click Next.

The SharePoint Connection Properties screen appears.

8. Click Next until the Add Attributes screen appears.

9. Click the drop-down arrows and select the values for the new claim from the following lists:

Attribute

Specifies an attribute name for one of the following claim types:

- Group based
- Role based

For multivalued attributes, prefix FMATTR, as shown in the following example:

Example: (multivalued attributes) FMATTR:LastName

Claim Type

Specifies an attribute value in your directory that is associated with the specified attribute name.

Example: (Active Directory attribute value for LastName) sn.

Example: (LDAP Directory role-based claim) sn.

10. Click Add, and then click Next.

The attribute details are saved and the Commit Details screen appears.

11. Click Install in the Commit Details screen.

The Save Complete screen appears.

12. Click Done.

The partnership details are saved, the SharePoint Connection is modified, and the wizard closes.

Search for and Add Users using the New Claim

You can search for users to add to your SharePoint Policy for web application using the new claim. For example, if you added a claim for the Last Name attribute, you can search for users by entering their last names in the SharePoint people picker.

Follow these steps:

1. Click Start, Programs, Microsoft SharePoint 2010 Products.
The Central Administration home page appears.
2. Click Manage web applications, in the Application Management section.
The Web Applications Management page appears with a list of available web applications.
3. Click the web application name for which you want to add users.
The buttons on the ribbon become available.
4. Click User Policy on the ribbon.
The Policy for Web Application dialog appears.
5. Click Add Users.
The Select Zone dialog appears.
6. Verify that the Zone you want appears in the drop-down list, and then Click Next.
The Add Users dialog appears.
7. Click the Browse button, in the Choose Users section, below the Users text box.
The Select People and Groups – Webpage Dialog appears.
8. Enter a value that corresponds to the new claim. For example, if your new claim is Last Name, enter the last name of a user.
The right pane displays the search results with a list of users whose attributes match the value on which you searched.
9. Select the user and click Add.
The selected user is added.
10. (Optional) Repeat steps 8 and 9 to select additional users.
11. Click OK.
The Add Users dialog appears and displays the selected user.
12. Under Choose Permissions, click the permissions that you want to grant to the users.
13. Click Finish.
The selected users and permissions are added.

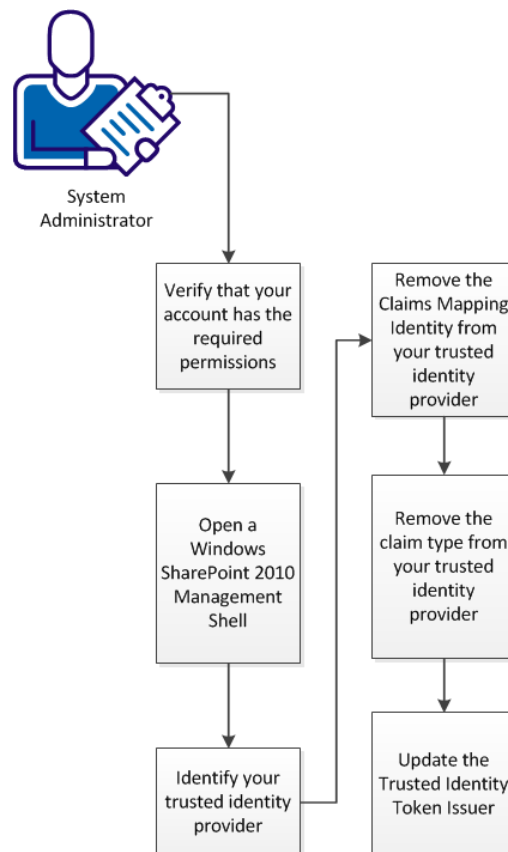
Removing Claims from Trusted Identity Providers

SharePoint 2010 supports third-party identity providers. These identity providers authenticate and authorize users who request SharePoint resources. A SharePoint administrator configures a trusted identity provider for a SharePoint environment.

Claims are a form of attribute or role, that a user has. Each claim has a name to identify it, and a value that the trusted identity provider verifies by connecting to a user directory.

For example, you can configure claims that correspond to the SamAccountName attribute of an Active Directory server or a uid of an LDAP directory server.

You can remove a claim to a CA SiteMinder® trusted identity provider at any time. The following illustration describes the process:



To remove a claim from a CA SiteMinder® trusted identity provider, follow these steps:

1. [Verify that your account has the required permissions](#) (see page 137).
2. [Open a SharePoint 2010 Management Shell window on your SharePoint Central Administration server](#) (see page 137).
3. [Identify your trusted identity provider](#) (see page 137).
4. [Remove the claims mapping identity from your trusted identity provider](#) (see page 145).
5. [Remove the claim type from your trusted identity provider](#) (see page 146).
6. [Update the trusted identity token issuer](#) (see page 146).

Verify that your Account has the Required Permissions

The user account with which you want to modify the CA SiteMinder® trusted identity provider requires certain permissions. Modify the permissions of your user account if it does *not* meet the following conditions:

- An Administrator account.
- A member of the Administrators group.

Add the following privileges to your account:

- Local administrator on all SharePoint web front end (WFE) servers.
- Read/Write access to the configuration database.

Open a SharePoint 2010 Management Shell Window on your SharePoint Central Administration Server

Add claims to your CA SiteMinder® trusted identity provider using the SharePoint 2010 Management shell.

Follow these steps:

1. Log in to your SharePoint Central Administration server.
2. Click Start, All Programs, Microsoft SharePoint 2010 Products, SharePoint 2010 Management Shell.

A SharePoint 2010 management shell command-line window appears.

Identify your Trusted Identity Provider

A SharePoint 2010 environment can have multiple trusted identity providers. Identify your CA SiteMinder® trusted identity provider before modifying any claims that are associated with it.

Follow these steps:

1. Enter the following command to list all of the trusted identity providers:

```
Get-SPTrustedIdentityTokenIssuer
```

A list of trusted identity providers appears.

2. Locate your CA SiteMinder® trusted identity provider in the list.

Your CA SiteMinder® trusted identity provider is identified.

Remove the ClaimsMapping Identity from your Trusted Identity Provider

Removing a claim from your CA SiteMinder® trusted identity provider involves several steps using the SharePoint 2010 Management Console. This example removes a claim for the last name of a user from the CA SiteMinder® trusted identity provider. Use this example as a guide to remove any claim you want from your CA SiteMinder® trusted identity provider.

Follow these steps:

1. Enter the following command to assign the name of your <stmdnr> trusted identity provider to a variable:

```
$trusted_identity_provider_variable_name = Get-SPTrustedIdentityTokenIssuer  
-Identity "name_of_siteminder_trusted_identity_provider"
```

2. Enter the following command to verify that the correct item is assigned to the variable:

```
echo $trusted_identity_provider_variable_name
```

3. Enter the following command to remove the claim from the CA SiteMinder® trusted identity provider. The command shown in the following example removes a claim for the last name of a user:

```
Remove-SPClaimTypeMapping -Identity  
"http://schemas.xmlsoap.org/claims/lastname" -TrustedIdentityTokenIssuer  
$trusted_identity_provider_variable_name
```

4. Repeat Step 1 to refresh the variable for the CA SiteMinder® trusted identity provider.

Remove the Claim Type from your Trusted Identity Provider

Remove the claim type from your CA SiteMinder® trusted identity provider.

Follow these steps:

1. Enter the following command to list the claim types contained in the variable for your CA SiteMinder® trusted identity provider:

```
$trutsed_identity_provider_variable_name.ClaimTypes
```

2. From the previous list, locate the claim type that is associated with the claim identity you want to remove.
3. Enter the following command to remove the claim type:

```
$trutsed_identity_provider_variable_name.ClaimTypes.Remove("http://schemas.xmlsoap.org/claims/lastname")
```

For example, the previous command removes the claim type for the last name of a user.

Update the Trusted Identity Token Issuer

Update the CA SiteMinder® trusted identity provider after removing the claim identity and the claim type.

Follow these steps:

1. Enter the following command to update the CA SiteMinder® trusted identity provider:

```
$trutsed_identity_provider_variable_name.Update
```

The trusted identity provider is updated.

Configure the Authentication Providers

You can create a web application that uses Claims-based authentication type by using the SharePoint Central Administration user interface or Windows PowerShell. Use the Central Administration to create a web application.

If you want to automate the task of creating a web application, which is common in enterprises, use Windows PowerShell. You can also modify the authentication type of an existing classic based authentication to claims-based authentication using the PowerShell script.

Modify an Existing Classic Authentication to Claims-based Authentication

You can update a web application that uses classic authentication to claims-based authentication using a PowerShell script. The following procedure helps you migrate existing web applications configured to use classic authentication, to use claims-based authentication.

Important! You cannot reverse this process. After you convert the web application authentication type to a Claims-based authentication, you cannot reconvert the authentication to the previous type.

Follow these steps:

1. Open the SharePoint 2010 Management Shell command prompt.

The command prompt appears.

2. Enter the following command to change the authentication mode to claims-based authentication:

```
$WebAppName = "http:// yourWebAppUrl"
$account = "yourDomain\yourUser"
$wa = get-SPWebApplication $WebAppName
```

```
Set-SPwebApplication $wa -AuthenticationProvider
(New-SPAuthenticationProvider) -Zone Default
```

The authentication mode is changed to claims-based authentication and the migration prompt is displayed.

Note: The preceding command modifies an existing classic authentication web application to claims-based authentication. Associate this web application with the Trusted Identity Provider in the SharePoint Central Administration user interface.

3. Click Yes to continue, at the migration prompt.
 4. Enter the following command to set the user as an administrator for the site:
- ```
$wa = get-SPWebApplication $WebAppName
$account = (New-SPClaimsPrincipal -identity $account -identitytype
1).ToEncodedString()
```
- The user is set as the administrator for the site.
5. Enter the following command to configure the policy to enable the user to have full access:

```
$zp = $wa.ZonePolicies("Default")
$p = $zp.Add($account,"PSPolicy")
$fc=$wa.PolicyRoles.GetSpecialRole("FullControl")
$p.PolicyRoleBindings.Add($fc)
$wa.Update()
```

The user obtains full access.

6. Enter the following command to configure the policy to perform user migration:

```
$wa = get-SPWebApplication $WebAppName
$wa.MigrateUsers($true)
```

The user migration process is completed.

7. Start SharePoint 2010 Central Administration from Start, Programs, Microsoft SharePoint 2010 Products.

The Central Administration Home page appears.

8. Click Manage web applications, in the Application Management section.

The Web Applications Management page appears with a list of available web applications.

9. Select the web application that has been updated and click Authentication Providers on the ribbon.

The Authentication Providers dialog shows that the authentication type has been updated to claims-based authentication.

**Note:** For information about claims-based authentication and for using the Windows PowerShell, see the *SharePoint Server 2010 Deployment Guide* from the Microsoft TechNet website.

## How to Disable Client Loopback

The Agent for SharePoint has a client loopback feature that lets you create policies in your SharePoint environment using directory attribute values that do not yet exist.

For example, suppose that your directory server contains an attribute named `employeeType`, and the `employeeType` attribute uses one of the following values for each user:

- Employee
- Contractor
- Manager
- Executive

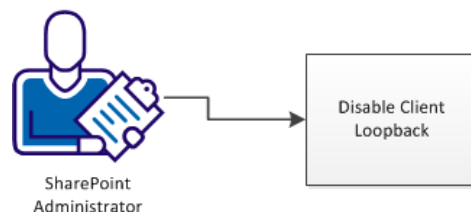
For example, suppose you want to create an attribute value for the `employeeType` attribute named `Vendor` in your directory servers to use with SharePoint.

If a different group in your organization manages the directory servers, that task is beyond your control. The Claims Provider creates placeholders for the new attribute values using the loopback feature.

In this example, use the loopback feature so that the `Vendor` attribute value exists in your SharePoint environment it appears in the directory servers. New attribute values let you create SharePoint policies whenever you want, without waiting for your administrator to add the actual attribute values to your directory.

If you do not need to add attributes before they exist in your directory, disable the client loopback feature.

### How to Disable Client Loopback



**Follow these steps:**

1. [Disable client loopback](#) (see page 150).

## Disable Client Loopback

If you do *not* need to add attributes using the SharePoint people picker before they exist in your user directories, disable the client loopback feature. Leaving client loopback enabled when the directory attributes exist returns duplicates in the SharePoint people picker.

**Follow these steps:**

1. Log in to your SharePoint central administration server.
2. Click Start, All Programs, Microsoft SharePoint 2010 Products, SharePoint 2010 Management Shell.

The management shell command-line window opens.

3. Navigate to the following directory:

C:\Program Files\CA\SharePointClaimsProvider\scripts

4. Enter the following command:

```
.\Set-SMClaimProviderConfiguration.ps1 -DisableLoopBackSearch
```

Loopback search is disabled.

## Add and Grant Permission to CA SiteMinder® Users

Add your users to SharePoint and assign permission levels depending on their roles. Permission levels allow users to perform a set of related tasks.

**Follow these steps:**

1. From your SharePoint central administration server, click, Start SharePoint 2010 Central Administration from Start, Programs, Microsoft SharePoint 2010 Products.

The Central Administration home page appears.

2. Click Manage web applications, in the Application Management section.

The Web Applications Management page appears with a list of available web applications.

3. Click the web application name for which you want to add users.

The buttons on the ribbon become available.

4. Click User Policy on the ribbon.  
The Policy for Web Application dialog appears.
5. Click Add Users.  
The Select Zone dialog appears.
6. Verify that the Zone you want appears in the drop-down list, and then Click Next.  
The Add Users dialog appears.
7. Click the Browse button, in the Choose Users section, below the Users text box.  
The Select People and Groups – Webpage Dialog appears.
8. Browse and select the user group to search for the user.  
The right pane displays the search results with the list of users.
9. Select the user and click Add.  
SharePoint adds the selected user.
10. (Optional) Repeat steps 8 and 9 to select additional users.
11. Click OK.  
The Add Users dialog appears and displays the selected users.
12. Select the required permissions for the users, in the Choose Permissions section.
13. Click Finish.  
SharePoint adds the selected users and assigns the selected permissions to the users.

## Manage User Profiles

The CA SiteMinder® Agent for SharePoint 12.52 does not support User Profile Import or User Migration. However, you can use the Microsoft SharePoint User Profile Synchronization Service to import user information from external directory sources. The User Profile Synchronization Service lets you extract the additional data from the external directory and augments the user records with this data. Data can also be written to the directory source (such as Active Directory or an LDAP directory), provided appropriate permissions are present.

The User Profile Service in SharePoint stores information about users in a central location, that allows multiple SharePoint applications to manage user profiles. Enable the User Profile service using SharePoint Central Administration.

You can configure SharePoint to use the User Profile Synchronization Service to import CA SiteMinder® users. And you can use the CA SiteMinder® Agent for SharePoint solution to protect the web applications.

**Note:** For more information about User Profile Synchronization, see the *Configure profile synchronization* article from the Microsoft TechNet website.



# Chapter 10: Features to Set Up Following Basic Installation and Configuration of the Agent for SharePoint

---

This section contains the following topics:

[Additional SharePoint Configuration Options](#) (see page 153)

[Office Client Integration](#) (see page 156)

[Claims Provider](#) (see page 165)

[Extend Web Applications to Different Zones for CRAWL Service and Search Support](#) (see page 188)

[How to Set Log Files, and Command-line Help to Another Language](#) (see page 188)

## Additional SharePoint Configuration Options

Perform any of these additional configuration steps at any time:

- [Create a web application that uses claims-based authentication](#) (see page 153).
- [Enable SSL on your IIS web server for a web application](#) (see page 154).
- [Enable SSL for the web application](#) (see page 155).

## Create a New Web Application with Claims based Authentication

**Follow these steps:**

1. Start SharePoint 2010 Central Administration from Start, Programs, Microsoft SharePoint 2010 Products.

The Central Administration home page appears.

2. Click Manage web applications, in the Application Management section.

The Web Applications Management page appears with a list of available web applications.

3. Click New, on the ribbon.

Create New Web Application dialog appears.

4. Select Claims Based Authentication option, in the Authentication section.
5. Select Yes option for Use Secure Sockets Layer (SSL), in the Security Configuration section.
6. Select the Trusted Identity Provider option, in the Claims Authentication Types.

**Note:** This option is already selected if you have set up Trusted Identity Provider authentication in Windows PowerShell.

**Important!** Verify that the options for all other authentication types in the Claims Authentication Types section are cleared.

7. Complete the remaining appropriate sections.
8. Click OK.

A new web application with claims authentication is created.

**Note:** For information about Claims-based authentication, see [www.microsoft.com](http://www.microsoft.com).

## Enable SSL on IIS for the Web Application

A Secure Sockets Layer (SSL) encryption is required for a web application as it provides greater security. Remote clients access the web application using URLs starting with https:// when SSL is used.

The following procedure describes how to enable SSL on IIS Manager.

### Follow these steps:

1. Click Start, Administrative Tools, Internet Information Services (IIS) Manager.  
The IIS Manager dialog appears.
2. Navigate to and select the Windows Claims-based authentication web application site that requires SSL encryption, in the Connections pane.
3. Click Edit Bindings in the Actions pane.  
The Site Bindings dialog appears.
4. Select the https entry, and then click Edit.  
The Edit Site Binding dialog appears.

5. Select the certificate for the server hosting your Agent for SharePoint from the list, in the SSL Certificate field.

6. Click OK.

The Site Bindings dialog appears.

7. Click Close.

The web application is enabled for SSL encryption.

**Note:** If you do not find an appropriate certificate, you cannot bypass the certificate warning screen. You can import the certificate that is issued to the URL into the client to bypass the certificate warning. For more information about configuring a Secure Sockets Layer, refer the *IIS 7 Operations Guide* from [www.microsoft.com](http://www.microsoft.com).

**More information:**

[Enable SSL for the Web Application](#) (see page 155)

[Create a New Web Application with Claims based Authentication](#) (see page 153)

## Enable SSL for the Web Application

You can configure the web application to use SSL when you create a web application. See *Create a Web Application with Claims-based Authentication* for the procedure to enable SSL when creating a web application. Alternatively, you can extend the SLL capability of a web application by performing the following procedure.

**Follow these steps:**

1. Click Start, Programs, Microsoft SharePoint 2010 Products, Start SharePoint 2010 Central Administration.

The Central Administration home page appears.

2. Click Application Management, Configure alternate access mappings section.

The Central Administration> Alternate Access Mappings page appears with a list of available web applications.

3. Click Add Internal URLs button.

The Central Administration> Add Internal URLs page appears.

4. Select an Alternate Access Mapping Collection from the Alternate Access Mapping Collection list.
5. Enter the URL with HTTPS in the Add Internal URL field and select a zone.  
For example, enter <https://spserver.example.com>.
6. Click OK.

The Central Administration> Alternate Access Mappings page appears with the modified URL.

**Note:** If you do not find an appropriate certificate, you cannot bypass the certificate warning screen. If you have a certificate that is issued to the URL, you can import the certificate in to the client to bypass the certificate warning. For more information about enabling SSL, for the web application in SharePoint refer [www.microsoft.com](http://www.microsoft.com).

**More information:**

[Create a New Web Application with Claims based Authentication](#) (see page 153)

[Enable SSL on IIS for the Web Application](#) (see page 154)

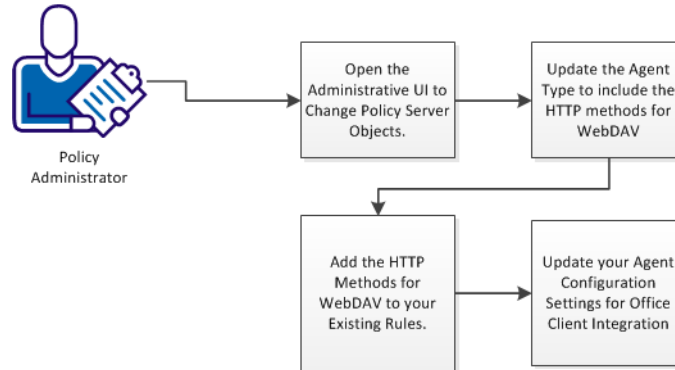
## Office Client Integration

Office Client Integration lets users edit and update documents stored on SharePoint with the respective Microsoft Office applications. For example, someone who has Microsoft Word can revise a Word document stored SharePoint.

## How to Configure Office Client Integration for the Agent for SharePoint

Office Client Integration lets users collaborate on Microsoft Office documents stored in SharePoint. When users open a Microsoft Office document on SharePoint, they use the related Microsoft Office application to edit the document.

### How to Configure Office Client Integration



#### Follow these steps:

1. [Open the Administrative UI to change Policy Server objects](#) (see page 42).
2. [Update the Agent Type to include the HTTP methods for WebDAV](#) (see page 158).
3. [Add the HTTP methods for WebDAV to your existing rules](#) (see page 159).
4. [Update your Agent Configuration Settings for Office Client Integration](#) (see page 160).

## Open the Administrative UI to Change Policy Server Objects

Change the objects on your Policy Server by opening the Administrative UI.

### Follow these steps:

1. Open the following URL in a browser.

`https://host_name:8443/iam/siteminder/adminui`

#### **host\_name**

Specifies the fully qualified Administrative UI host system name.

2. Enter your CA SiteMinder® superuser name in the User Name field.
3. Enter the CA SiteMinder® superuser account password in the Password field.  
**Note:** If your superuser account password contains dollar-sign (\$) characters, replace each instance of the dollar-sign character with \$DOLLAR\$. For example, if the CA SiteMinder® superuser account password is \$password, enter \$DOLLAR\$password in the Password field.
4. Verify that the proper server name or IP address appears in the Server drop-down list.
5. Select Log In.

## Update the Agent Type to Include the HTTP Methods for WebDAV

To use the Office Client Integration feature, modify the Agent type to include the methods for WebDAV.

### Follow these steps:

1. Click Infrastructure, Agents, Agent Type, Modify Agent Type.  
The Create Agent Type search pane appears.
2. Highlight the text in the search field, and then type the following:  
Web Agent
3. Click Search.  
The Web Agent type appears in the list.
4. Click Select.  
The Modify Agent Type: *Web Agent* pane appears.
5. Scroll to the bottom of the Actions section, and then click Create.  
A new action field appears at the end of the list.

6. Highlight the text in the New Action field, and then enter the following:

Head

7. Scroll to the bottom of the Actions section, and then click Create.

A new action field appears at the end of the list.

8. Repeat Steps 6 and 7 until all of the following methods are added:

OPTIONS

PROPFIND

PROPPATCH

COPY

DELETE

MOVE

LOCK

UNLOCK

9. Click Submit.

The Modify Agent type task is submitted for processing. A confirmation screen appears.

10. Click OK.

The Agent type settings for your SharePoint resources are updated.

## Add the HTTP Methods for WebDAV to Your Existing Rules

To use the Office Client Integration feature with the Agent for SharePoint, update the web agent actions in any rules protecting SharePoint sites.

### Follow these steps:

1. Click Policies, Domains, Rule, Modify Rule.

The Modify Rule screen appears.

2. Click the option button of the domain that contains the rule you want, and then click Select.

Modify Rule: Name screen appears.

3. In the Action drop-down list, press and hold Ctrl and click the following items:

- HEAD
- OPTIONS
- PROPFIND
- PROPPATCH
- COPY
- DELETE
- MOVE
- LOCK
- UNLOCK

4. Click Submit.

5. Repeat Steps 2 through 4 for any additional rules that you want.

The rule is updated, and the confirmation screen appears.

## Update your Agent Configuration Settings for Office Client Integration

The parameter settings in the Agent Configuration Object that is associated with your Agent for SharePoint control how Office Client Integration operates on your Agent for SharePoint.

### Follow these steps:

1. Click Infrastructure, Agent Configuration, Modify Agent Configuration.
2. Click the edit button for the Agent Configuration object of your Agent for SharePoint.

The Modify Agent Configuration: Name pane opens.



3. Change the values of the following parameters:

**SPClientIntegration**

Specifies the hostnames of the SharePoint servers that the Agent for SharePoint protects on which you want to permit Office Client Integration. The default parameter is blank and listed as plain. If there are multiple host entries, use the multivalue option button to add multiple hosts.

Add a port number to the value if the Agent for SharePoint operates on a nondefault port (any port except 80 or 443).

To use this parameter, verify that the SharePoint resources that CA SiteMinder® protects also have their Office Client Integration enabled on the SharePoint central administration server.

Because Office Client Integration requires a persistent FedAuth cookie, verify that your SharePoint server is *not* configured to use session cookies. By default, UseSessionCookies in SharePoint is set to NO.

**Default:** None

**Limits:** Multiple values are allowed. Use fully qualified domain names for all values.

**Example:** *agent\_for\_sharepoint\_host\_name.example.com* (default ports of 80 or 443)

**Example:** *agent\_for\_sharepoint\_host\_name.example.com:81* (with a nondefault port number for HTTP)

**Example:** *agent\_for\_sharepoint\_host\_name.example.com:4343* (with a nondefault port number for HTTPS)

**SPDisableClientIntegration**

Specifies the hostnames of the SharePoint servers that the Agent for SharePoint protects on which you want to prohibit Office Client Integration. The default parameter is blank and listed as plain. If there are multiple host entries, then switch over to a multi—value parameter. The URL in this parameter *requires* a port number (even for a default port such as 80 or 443).

This setting prevents SharePoint administrators from circumventing CA SiteMinder® settings regarding Office Client integration.

**Limit:** Multiple values are allowed.

**Example:** *agent\_for\_sharepoint\_host\_name:port\_number*

4. The following parameter describes the user agent values to which the Agent for SharePoint permits access:

**SPAuthorizeUserAgent**

Specifies a list of Microsoft Office user-agent strings for which the Agent for SharePoint allows access. This list is populated automatically with the default values when the Agent for SharePoint starts. The user-agent strings in this parameter act as a whitelist. Changes to this parameter override the default settings. Access is denied to clients whose user-agent string does not appear in the list.

For example, setting the value to Microsoft Office allows access to all versions of Microsoft Office products that are associated with that user-agent string. Conversely, setting the value to Microsoft Office/12.0 allows access to only those versions of Microsoft Office products that are associated with that user-agent string.

**Default:** Microsoft Office, MS FrontPage, MSFrontPage, Microsoft Data Access Internet Publishing Provider Protocol Discovery, Test for Web Form Existence, Microsoft-WebDAV-MiniRedir

**Limits:** Multiple values are allowed.

5. Examine the default values of the previous parameter. Ask your SharePoint or IIS web server administrator if more user-agent values are required.

**Note:** Microsoft (*not* CA Technologies) defined the user-agent strings in the previous parameter. For more information about these user-strings, search the [Microsoft Developer Network \(MSDN\) library](#) website for information about the user-string that you want.

6. Change the value of the CSSChecking parameter to no.

**Note:** Because the Agent for SharePoint is a proxy-based solution, this setting is required for Office Client Integration.

7. Click OK.

The new values appear next to the parameters in the list.

8. Click Submit.

The Create Agent Configuration Task is submitted for processing and the confirmation message appears.

## How to Configure WebDAV to Accomodate Microsoft Hot Fixes 2563214 and 2647954

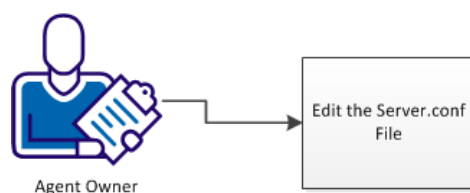
The following hot fixes from Microsoft affect the behavior of the Agent for SharePoint:

- [KB2563214](#)
- [KB2647954](#)

These hotfixes return Web browser error messages with code 500 to users who try opening Microsoft Office documents.

As a work-around, modify the server.conf file on any servers running the Agent for SharePoint that have any of the previous hot fixes installed.

### How to Accommodate Microsoft Hot Fixes KB2563214 and KB2647954



#### Follow these steps:

1. Modify the [server.conf file on the server running your Agent for SharePoint](#) (see page 164).

## Modify the server.conf File

Adding new directives to the server.conf file on each Agent for SharePoint eliminates the error messages that the Microsoft hot fixes cause.

### Follow these steps:

1. Log on to the server hosting your Agent for SharePoint.
2. Open the following file with a text editor:

*Agent-for-SharePoint\_Home/secure-proxy/proxy-engine/conf/server.conf*

#### **Agent-for-SharePoint\_Home**

Indicates the directory where the CA SiteMinder Agent for SharePoint is installed.

**Default:** (Windows) [32-bit] C:\Program Files\CA\Agent-for-SharePoint

**Default:** (Windows) [64-bit] C:\CA\Agent-for-SharePoint

**Default:** (UNIX/Linux) /opt/CA/Agent-for-SharePoint

3. Search the file for the following tag:

<SharePoint>

4. Do *one* of the following tasks:

- If the tag in Step 3 is *already* in the file, remove any comment marks in the section to accommodate the hotfixes. Go to Step 5.
- If the tag does *not* exist in the file, then go to Step 5.

5. Add the following section:

```
<SharePoint>
 allowedClientMethods="PROPFIND,OPTIONS"
 allowedUserAgents="WebDAV"
</SharePoint>
```

6. Save the file and close the text editor.

The server.conf file is modified to accommodate the Microsoft hot fixes.

7. Repeat Steps 1 through 6 on all servers running the Agent for SharePoint.

## Claims Provider

The Claims Provider in the Agent for SharePoint is used for configuring particular claim values to grant permissions to SharePoint resources using the SharePoint people picker. The Claims Provider is packaged as a SharePoint solution (WSP file) with its feature receiver.

The Claims Provider requires Directory Attribute Mappings that you configure using the CA SiteMinder® Administrative UI. The Claims Provider uses these mappings to display the results of your searches in the SharePoint people picker.

Using the Claims Provider involves several separate procedures. Use the following process.

1. [Create virtual attribute mappings](#) (see page 166).
2. [Install the Claims Provider](#) (see page 181).
3. [Configure the Claims Provider](#) (see page 182).

**Note:** The Claims Provider for SharePoint installer supports Windows 64-bit Operating Systems.

## Claims Provider Searches and Results

The SharePoint claims provider lets you search your CA SiteMinder® directories with the SharePoint people picker.

The following table describes the relationships between the search criteria you enter in the people picker and the search results that appear:

<b>When you search for this attribute in the SharePoint people picker:</b>	<b>The SharePoint people picker returns the following results:</b>
User identifier or display name.	The user identifier or the display name of the user
Group name	The friendly name associated with the smusergroup attribute
Other attributes (such as claim names based on a role)	The attribute value you associated with the role.

## Agent for SharePoint Virtual Attribute Mappings

Virtual attribute mappings create relationships between the attributes from your SiteMinder user directories and the SiteMinder claims provider. These mappings allow SiteMinder to search your user directories for claims, and display the results in the SharePoint people picker.

The following types of claims are supported:

- User claims (one is required)
- Group claims
- Role claims

**Note:** Configuring this feature requires information from several systems or administrators in your organization. Work with the administrators for your SharePoint environment and with administrators for the user directories in your organization.

## Virtual Attribute Mapping Examples for an LDAP Directory

To search the user directory in your CA SiteMinder® environment using the SharePoint people picker, create virtual attribute mappings. The Agent for SharePoint requires at least *one* attribute mapping for claims that are based on the ID of a user. Create additional mappings to accommodate your needs.

**Important!** The Agent for SharePoint supports only one CA SiteMinder® user directory.

Each additional mapping creates another association between a specific attribute in your user directory and the Agent for SharePoint. The people picker in SharePoint uses these associations to search your user directory using the values you specify. For example, you can create an attribute mapping that lets you search by user name, group name or email address.

The following table identifies the typical LDAP directory attribute mappings and describes how they are used in your CA SiteMinder® and SharePoint environments:

For LDAP Directories:	Create a CA SiteMinder® virtual attribute to search for this claim with the people picker.		Create a CA SiteMinder® virtual attribute so the friendly names appear in the people picker next to the corresponding claim values.		Enter these corresponding values in the SharePoint Connection wizard.		(Optional) Customize the display name for the people picker
Purpose	1. Use this name for your virtual attribute.	2. Enter the name of the directory attribute you want to use for the claim value.	3. Use this name for the CA SiteMinder® virtual attribute.	4. Use this name for the directory attribute you want to use as a claim value.	5. To define the claim in the connection wizard:	6. To define the attribute value for the claim in the connection wizard:	7. Replace the string following the -IncomingClaimTypeDisplayName with this value:
Mandatory User claim that uniquely identifies the user.	useridentifier	uid	smuserdisplayname	displayName	Enter the following value in the Identifier Claim Name field: useridentifier	Enter the following value in the Directory Attribute field: uid	User ID

(Optional) Group-based user-claim that is based on a DN in the directory.	smusergroups	Description  (use the friendly name of your groups).	Not required for group-based claims.	Click the Attribute drop-down list and then select the following value: smusergroups	Not required. The connection wizard configures this setting automatically.	Group
(Optional) Role-based user claim	userrole	employeeType	Not supported.	<p>1. Click the Attribute drop-down list and then select the following value: NameValue</p> <p>2. Click the Claim type drop-down list and select the following value: User Attribute</p> <p>3. Click the Claim Name field and enter the following value: userrole</p>	Enter the following value in the Directory Attribute field: employeeType	Role



## Virtual Attribute Mapping Examples for a Microsoft Active Directory Server

To search the user directory in your CA SiteMinder® environment using the SharePoint people picker, create virtual attribute mappings. The Agent for SharePoint requires at least *one* attribute mapping for claims that are based on the ID of a user. Create additional mappings to accommodate your needs.

**Important!** The Agent for SharePoint supports only one CA SiteMinder® user directory.

Each additional mapping creates another association between a specific attribute in your user directory and the Agent for SharePoint. The people picker in SharePoint uses these associations to search your user directories using the values you specify. For example, you can create an attribute mapping that lets you search by user name, group name or email address.

The following table identifies the typical Microsoft Active Directory attribute mappings and describes how they are used in your CA SiteMinder® and SharePoint environments:

For Active Directories:	Create a CA SiteMinder® virtual attribute to search for this claim with the people picker.		Create a CA SiteMinder® virtual attribute so the friendly names appear in the people picker next to the corresponding claim values.		Enter these corresponding values in the SharePoint Connection wizard.		(Optional) Customize the display name for the people picker
Purpose	1. Use this name for your virtual attribute.	2. Enter the name of the directory attribute you want to use for the claim value.	3. Use this name for the CA SiteMinder® virtual attribute.	4. Use this name for the directory attribute you want to use as a claim value.	5. To define the claim in the connection wizard:	6. To define the attribute value for the claim in the connection wizard:	7. Replace the string following the -Incoming ClaimType DisplayName with this value:
Mandatory User claim that uniquely identifies the user.	useridentifier	sAMAccount Name	smuserdisplay name	displayName	Enter the following value in the Identifier Claim Name field: useridentifier	Enter the following value in the Directory Attribute field: sAMAccount Name	User ID

(Optional) A group-based user-claim corresponding to a DN in the directory.	smusergroups	name (use the friendly name of your groups).	Not required for group-based claims.	Click the Attribute drop-down list and then select the following value: smusergroups	Not required. The connection wizard automatically configures this setting.	Group
(Optional) Role-based user claim	userrole	countryCode	Not supported.	<p>1. Click the Attribute drop-down list and then select the following value: NameValue</p> <p>2. Click the Claim type drop-down list and select the following value: User Attribute</p> <p>3. Click the Claim Name field and enter the following value: userrole</p>	Enter the following value in the Directory Attribute field: countryCode	Role

## User Claims

Integration with SharePoint requires at least one claim that contains an identifier that uniquely identifies the user. These claims often appear in the people picker as cryptic values, such as the following example:

```
uid=e123456
```

Such claims are difficult to associate with the intended user. The Agent for SharePoint uses a special attribute mapping which retrieves the display name of the user. This user name appears next to the related identifier claim in the people picker. After this user mapping is configured, the previous example appears in the people picker like the following one:

```
uid=e123456 associated_user_name
```

## Create an Attribute Mapping for User Claims in an LDAP Directory

The Agent for SharePoint requires an attribute mapping based on an attribute with a unique value for each user. Use the Administrative UI to create a pair of attribute mappings that defines how SiteMinder searches for user claims through the SharePoint people picker.

**Important!** The Agent for SharePoint supports only one CA SiteMinder® user directory.

**Note:** For more information about the relationships between attribute mappings in an LDAP directory and the other components of your environment, [see the LDAP examples chart](#) (see page 167).

### Follow these steps:

1. Log on to the SiteMinder Administrative UI.
2. Click Infrastructure, Directory, User Directory, Modify User Directory.  
A list of user directory connections appears.
3. Click the option button for your user directory, and then click Select.  
The Modify User directory page appears.
4. Click Create.  
The create attribute mapping page appears.
5. Verify that the Create a new object of type Attribute Mapping option button is selected, and then click OK.
6. Click the name field, and enter the following name:  
`useridentifier`
7. Verify that the Alias option button is selected, and then click the Definition field.
8. Enter the following definition:  
`uid`
9. Click OK.  
The Modify User directory page appears.
10. To create the second mapping, repeat Steps 4 through 5.
11. Click the name field, and then enter the following name:  
`smuserdisplayname`
12. Verify that the Alias option button is selected, and then click the Definition field.
13. Enter the following definition:  
`displayName`
14. Click OK.

The Modify User directory page appears.

15. Click Submit.

The attribute mappings are created.

## Create an Attribute Mapping for User Claims in a Microsoft Active Directory Server

The Agent for SharePoint requires an attribute mapping that is based on an attribute with a unique value for each user. Use the Administrative UI to create a pair of attribute mappings that defines how SiteMinder searches for user claims through the SharePoint people picker.

**Important!** The Agent for SharePoint supports only one CA SiteMinder® user directory.

**Note:** For more information about relationships between attribute mappings in an Active Directory server and other components of your environment, see the [Active Directory examples table](#) (see page 169).

### Follow these steps:

1. Log in to the SiteMinder Administrative UI.
2. Click Infrastructure, Directory, User Directory, Modify User Directory.  
A list of user directory connections appears.
3. Click the option button for your user directory, and then click Select.  
The Modify User directory page appears.
4. Click Create.  
The create attribute mapping page appears.
5. Verify that the Create a new object of type Attribute Mapping option button is selected, and then click OK.
6. Click the name field, and enter the following name:  
`useridentifier`
7. Verify that the Alias option button is selected, and then click the Definition field.
8. Enter the following definition:  
`sAMAccountName`
9. Click OK.  
The Modify User directory page appears.
10. To create the second mapping, repeat Steps 4 through 5.
11. Click the name field, and then enter the following name:  
`smuserdisplayname`
12. Verify that the Alias option button is selected, and then click the Definition field.
13. Enter the following definition:  
`displayName`
14. Click OK.

The Modify User directory page appears.

15. Click Submit.

The attribute mappings are created.

## Group Claims

You can also configure a claim that uses the groups to which the user belongs. Group mappings assign SharePoint permissions based on groups of users rather than individuals.

Some user directories define the groups of users by including an attribute in the record that contains the distinguished name (DN) of each group. The DN also appears as a cryptic value such as the following example:

```
entryDN=cn=grp12345,ou=Groups,dc=example,dc=com
```

Such claims are difficult to identify the name of the group associated with the value in the people picker.

The Agent for SharePoint uses two attribute mappings and the groups setting you specify in the SharePoint connection wizard to search for groups by their display name. The Agent for SharePoint retrieves both the display name of the group and DN of the group.

Both the display name and the DN of the group then appear in the people picker, for as shown in the following example:

```
cn=grp12345,ou=Groups,dc=example,dc=com(Sales Managers).
```

## Create Attribute Mappings for Group-based Claims in LDAP Directories

You can also create attribute mappings based on a group of users. Use the Administrative UI to create an attribute mapping that defines how SiteMinder searches for group claims through the SharePoint people picker.

**Note:** For more information about the relationships between attribute mappings in an LDAP directory and the other components of your environment, [see the LDAP examples chart](#) (see page 167).

### Follow these steps:

1. Log on to the SiteMinder Administrative UI.
2. Click Infrastructure, Directory, User Directory, Modify User Directory.  
A list of user directory connections appears.
3. Click the option button for your user directory, and then click Select.  
The Modify User directory page appears.
4. Click Create.  
The create attribute mapping page appears.
5. Verify that the Create a new object of type Attribute Mapping option button is selected, and then click OK.
6. Click the name field, and enter the following name:  
`smusergroups`
7. Verify that the Alias option button is selected, and then click the Definition field.
8. Enter the following definition:  
`description`
9. Click OK.  
The Modify User directory page appears.
10. Click Submit.  
The attribute mapping is created.



## Create Attribute Mappings for Group-based Claims in Active Directory

You can also create attribute mappings based on a group of users. Use the Administrative UI to create an attribute mapping that defines how SiteMinder searches for group claims through the SharePoint people picker.

**Note:** For more information about relationships between attribute mappings in an Active Directory server and other components of your environment, see the [Active Directory examples table](#) (see page 169).

### Follow these steps:

1. Log on to the SiteMinder Administrative UI.
2. Click Infrastructure, Directory, User Directory, Modify User Directory.  
A list of user directory connections appears.
3. Click the option button for your user directory, and then click Select.  
The Modify User directory page appears.
4. Click Create.  
The create attribute mapping page appears.
5. Verify that the Create a new object of type Attribute Mapping option button is selected, and then click OK.
6. Click the name field, and enter the following name:  
`smusergroups`
7. Verify that the Alias option button is selected, and then click the Definition field.
8. Enter the following definition:  
`name`
9. Click OK.  
The Modify User directory page appears.
10. Click Submit.  
The attribute mapping is created.

## Role Claims

You can also configure any number of claims in Name=Value format. These name/value pairs are often named *role claims*.

Role claims are found by reading a configurable attribute on the user record in your user directory. You can then assign any name you want for the claim. For example, you can name a claim "userrole" and configure it to point to the "employeeType" attribute in your LDAP directory.

After authentication the Agent for SharePoint creates a name/value pair such as "userrole=manager" for the claim. If the "employeeType" attribute for the authenticated user contains the value named manager, SharePoint allows the user access to the resource.

## Create an Attribute Mapping for a Role-based Claims in LDAP Directories

You can also create attribute mappings based on user roles. Use the Administrative UI to create an attribute mapping that defines how SiteMinder searches for role-based claims through the SharePoint people picker.

**Note:** For more information about the relationships between attribute mappings in an LDAP directory and the other components of your environment, [see the LDAP examples chart](#) (see page 167).

### Follow these steps:

1. Log on to the SiteMinder Administrative UI.
2. Click Infrastructure, Directory, User Directory, Modify User Directory.  
A list of user directory connections appears.
3. Click the option button for your user directory, and then click Select.  
The Modify User directory page appears.
4. Click Create.  
The create attribute mapping page appears.
5. Verify that the Create a new object of type Attribute Mapping option button is selected, and then click OK.
6. Click the name field, and enter the following name:  
`userrole`
7. Verify that the Alias option button is selected, and then click the Definition field.
8. Enter the following definition:  
`employeeType`
9. Click OK.  
The Modify User directory page appears.
10. Click Submit.  
The attribute mapping is created.
11. (Optional) Create more role-based mappings to suit your needs.

## Create an Attribute Mapping for a Role-based Claims in Active Directory

You can also create attribute mappings based on user roles. Use the Administrative UI to create an attribute mapping that defines how SiteMinder searches for role-based claims through the SharePoint people picker.

**Note:** For more information about relationships between attribute mappings in an Active Directory server and other components of your environment, see the [Active Directory examples table](#) (see page 169).

### Follow these steps:

1. Log on to the SiteMinder Administrative UI.
2. Click Infrastructure, Directory, User Directory, Modify User Directory.  
A list of user directory connections appears.
3. Click the option button for your user directory, and then click Select.  
The Modify User directory page appears.
4. Click Create.  
The create attribute mapping page appears.
5. Verify that the Create a new object of type Attribute Mapping option button is selected, and then click OK.
6. Click the name field, and enter the following name:  
userrole
7. Verify that the Alias option button is selected, and then click the Definition field.
8. Enter the following definition:  
countryCode
9. Click OK.  
The Modify User directory page appears.
10. Click Submit.  
The attribute mapping is created.
11. (Optional) Create more role-based mappings to suit your needs.

## Install Claims Provider

If you are not the user who installed or configured SharePoint, you need *one* of the following privileges to run the Claims Provider installer:

- Administrator for the local server
- Administrator for the group
- Farm Administrator (for any SharePoint farms)

If you are installing your Claims provider on a new SharePoint farm, install the claims provider on your SharePoint central administration server. If you add any additional SharePoint servers to your farm later, install the claims provider on each SharePoint server you add.

### Follow these steps:

1. Log on to your SharePoint central administration server.
2. Copy the installation program from the download location on the CA Support site.
3. Locate the following executable:

`ca-spcclaims-version-win64.exe`

4. Right-click the executable, and then select Run as administrator.

The installation program starts.

5. Follow the installation wizard.
6. Restart your system after the installation finishes.

The Claims provider is successfully installed.

### More information:

[Locate the CA SiteMinder® Agent for SharePoint Platform Support Matrix](#) (see page 420)

[Locate the Installation Media](#) (see page 421)

## Verify Claims Provider Installation

### Follow these steps:

1. Start SharePoint 2010 Central Administration from Start, Programs, Microsoft SharePoint 2010 Products.
2. Click System Settings.

The Central Administration>System Settings page appears.

3. Click Manage Farm Solutions, in the Farm Management section.

The Central Administration>Solution Management page appears and the status of the Claims Provider is shown as Deployed.

## How to Configure the Claims Provider

After you install the CA SiteMinder® Claims provider, add the claims search service and update the claims provider of the trusted identity token issuer:

**Follow these steps:**

1. [Update the claims provider of the trusted identity token issuer](#) (see page 182).
2. [Add the Claims search service](#) (see page 183).

After you add the Claims Search service, you can also configure the Claims Provider to suit your needs with any of the following optional procedures:

- [Create SharePoint policies with place holders for expected directory attribute values](#). (see page 185)
- [Change how directory attributes appear in the SharePoint people picker](#) (see page 186).

## Update the Claims Provider of the Trusted Identity Token Issuer

The Update-SMTrustedIdentityTokenIssuer command updates the claims provider of a trusted identity token issuer to CASiteMinderClaimProvider.

**Follow these steps:**

1. Click Start, All Programs, Microsoft SharePoint 2010 Products, the SharePoint 2010 Management Shell.

The SharePoint 2010 Management Shell command prompt appears.

2. Navigate to the following directory:

```
C:\Program Files\CA\SharePointClaimsProvider\scripts
```

3. Enter the update command. This command has the following format:

```
Update-SMTrustedIdentityTokenIssuer.ps1 -TrustedIdentityTokenIssuer
"Name_of_Trusted_Identity_Provider_registered_with_SharePoint"
```

### **TrustedIdentityTokenIssuer**

Specifies the name of the CA SiteMinder® trusted identity token issuer (trusted login provider) to update.

**Example:**

```
.\Update-SMTrustedIdentityTokenIssuer.ps1 -TrustedIdentityTokenIssuer
"SiteMinder Federation"
```

The SharePoint central administration server is updated with the new claims provider of the trusted identity token issuer.

## Add Claims Search Web Service

Add the claims search web service used in the Agent for SharePoint to specific SharePoint web applications by executing the Add-SMClaimSearchService command. The changes made by this script are reflected across the SharePoint Farm.

### Follow these steps:

1. Click Start, All Programs, Microsoft SharePoint 2010 Products, the SharePoint 2010 Management Shell.

The SharePoint 2010 Management Shell command prompt appears.

2. Navigate to the following directory:

C:\Program Files\CA\SharePointClaimsProvider\scripts

3. Enter the add command. This command has the following format:

```
ADD-SMClaimSearchService.ps1 -WebApplication <URL_of_web_application>
-claimSearchService <URL_of_claim_search_service_in_spagent>
```

#### WebApplication

Specifies the URL of the web application.

#### claimSearchService

Specifies the URL of the claim search service running in Agent for SharePoint.

#### Example:

```
.\ADD-SMClaimSearchService.ps1 -WebApplication http://myhostname:1234
-claimSearchService
http://spagent.ca.com:2345/ClaimsWS/services/WSSharePointClaimsServiceImp
l
```

The claims search web service is added to the web.conf file of the web application.

4. Enter the add command again, to add the claims web search service to the web.conf file of the SharePoint Central Administration.

```
ADD-SMClaimSearchService.ps1 -WebApplication <Central_Administration_URL>
-claimSearchService <URL_of_claim_search_service_in_spagent>
```

**WebApplication**

Specifies the URL of the SharePoint Central Administration website.

**claimSearchService**

Specifies the URL of the claim search service running in the Agent for SharePoint. Add the port number you specified for the Claims WS of the Agent for SharePoint when you ran the Configuration wizard to the end of the URL.

**Example:**

```
.\ADD-SMClaimSearchService.ps1 -WebApplication
http://SharePoint_server_name:1221 -claimSearchService
http://spagent.ca.com:2345/ClaimsWS/services/WSSharePointClaimsServiceImp
l
```

The claims search web service is added to the web.conf file of the SharePoint Central Administration.

**More information:**

[CA SiteMinder Agent for SharePoint Configuration Wizard Information Worksheet](#) (see page 417)



## Create SharePoint Policies with Placeholders for Expected Directory Attributes

The Agent for SharePoint has a client loopback feature that lets you create policies in your SharePoint environment using directory attribute values that do not yet exist.

For example, suppose that your directory server contains an attribute named `employeeType`, and the `employeeType` attribute uses one of the following values for each user:

- Employee
- Contractor
- Manager
- Executive

For example, suppose you want to create an attribute value for the `employeeType` attribute named `Vendor` in your directory servers to use with SharePoint.

If a different group in your organization manages the directory servers, that task is beyond your control. The Claims Provider creates placeholders for the new attribute values using the loopback feature.

In this example, use the loopback feature so that the `Vendor` attribute value exists in your SharePoint environment it appears in the directory servers. New attribute values let you create SharePoint policies whenever you want, without waiting for your administrator to add the actual attribute values to your directory.

### Follow these steps:

1. Click Start, All Programs, Microsoft SharePoint 2010 Products, SharePoint 2010 Management Shell.  
The management shell command line window opens.
2. Navigate to the following directory:  
`C:\Program Files\CA\SharePointClaimsProvider\scripts`
3. Enter the following command:  
`.\Set-SMClaimProviderConfiguration.ps1 -EnableLoopBackSearch`  
Loopback search is enabled.
4. Use the SharePoint people picker to search the new attribute values you want.  
A placeholder for the new attribute value is added to SharePoint using the loopback search function.
5. Repeat Step 4 to add additional placeholders for more attribute values.
6. (Optional) After adding your placeholders, disable support for the loopback search function by doing the following steps:
  - a. Repeat Steps 1 and 2.

- b. Enter the following command:

```
.\Set-SMClaimProviderConfiguration.ps1 -DisableLoopBackSearch
```

Loopback search is disabled.

## Change How Directory Attributes Appear in the SharePoint People Picker

You can customize how certain directory attributes from your SiteMinder user directories appear in the SharePoint people picker.

### Change how directory attributes appear in the SharePoint people picker

1. Click Start, All Programs, Microsoft SharePoint 2010 Products, SharePoint 2010 Management Shell.

The management shell command line window opens.

2. Navigate to the following directory:

```
C:\Program Files\CA\SharePointClaimsProvider\scripts
```

3. Enter the `.\Set-SMClaimProviderConfiguration.ps1` command with one of the following options:

#### **-UserNameFormat**

Specifies how the user names for which you search appear in the SharePoint people picker. Use one of the following options:

##### **ValueOnly**

Displays only the value of the identifier claim attribute in your directory server associated with the user. For example, if your uid is `user_number`, then only `user_number` appears in your search results.

**Example:** `user_0001`

##### **DisplaynameOnly**

Displays only the name of the user, using the format specified in your SiteMinder directory.

**Example:** `last_name_of_user, first_name_of_user`

##### **DisplaynameAppended**

Displays the name of the user, and the value of the identifier claim attribute in your directory server associated with the user.

**Example:** `user_0001 (last_name_of_user, first_name_of_user)`

**-GroupNameFormat**

Specifies how the group names for which you search appear in the SharePoint people picker. Use one of the following options:

**ValueOnly**

Displays only the domain name (DN) value of the group claim attribute in your directory server associated with the user.

**Example:** OU=group\_0001, DC=example, DC=COM

**DisplaynameOnly**

Displays only the name of the group, using the format specified in your SiteMinder directory.

**Example:** *group\_name*

**DisplaynameAppended**

Displays the name of the group, and the value of the group claim attribute in your directory server associated with the user.

**Example:** *group\_name* OU=group\_0001, DC=example, DC=COM

The appearance of the directory attributes is changed.

## Remove Claims Search Web Service

The Remove-SMClaimSearchService command removes the changes made in the web.config file. The script identifies the modifications made by the user from the *CASiteMinderSharePoint2010Agent\_ClaimsSearchServiceEndpoint* file.

**Follow these steps:**

1. Click Start, All Programs, Microsoft SharePoint 2010 Products, the SharePoint 2010 Management Shell.

The SharePoint 2010 Management Shell command prompt appears.

2. Navigate to the following directory:

C:\Program Files\CA\SharePointClaimsProvider\scripts

3. Enter the remove command. This command has the following format:

```
Remove-SMClaimSearchService.ps1 -WebApplication <URL_of_web_application>
```

**WebApplication**

Specifies the URL of the web application.

**Example:**

```
.\Remove-SMClaimSearchService.ps1 -WebApplication http://myhostname:1234
```

The changes made in the web.config file are removed.

## Extend Web Applications to Different Zones for CRAWL Service and Search Support

The Agent for SharePoint does not support CRAWL services because the service does not use CA SiteMinder® cookies. The SharePoint CRAWL service uses Windows authentication, and the Agent for SharePoint uses claims authentication. Because the SharePoint CRAWL service cannot respond to the authentication challenge the Agent for SharePoint makes, the Agent for SharePoint denies the request. When this denial occurs, the connection to the CRAWL service or the search times out.

### Follow these steps:

1. Extend the SharePoint web application with which you want to use the crawl service to a different zone.
2. Configure the extended web application (from Step 1) to use Integrated Windows (IWA or NTLM) authentication.
3. Configure the CRAWL service to use the URL of the extended SharePoint web application (from Step 1).

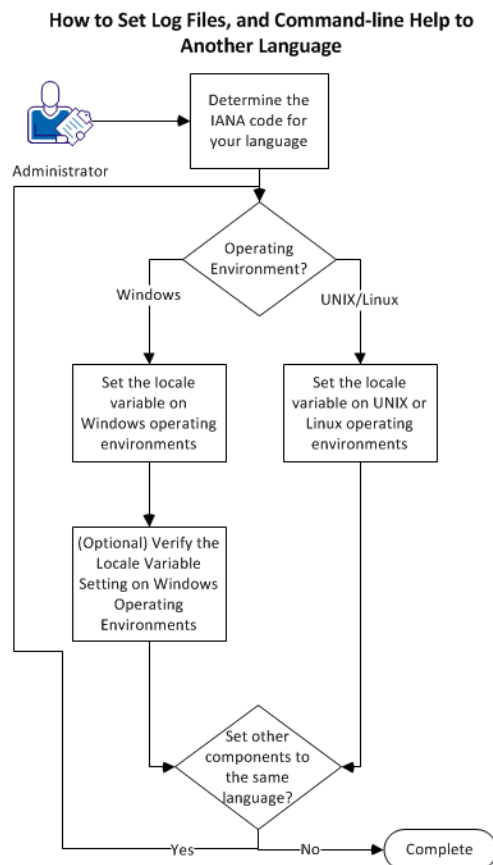
Extending the web application to another zone provides protection of the web application with the Agent for SharePoint while supporting the CRAWL service and search functions.

## How to Set Log Files, and Command-line Help to Another Language

The following components support log files, and command-line help in other languages:

- The Policy Server
- The Web Agent
- The Report Server
- The CA SiteMinder Agent for SharePoint
- The CA SiteMinder Agent for SharePoint
- Agent for SharePoint
- Any custom software that is created with the CA SiteMinder® SDK.

The following graphic describes the work flow for setting log files, and command-line help to another language:



**Follow these steps:**

1. [Determine the IANA code for your language](#) (see page 190).
2. Create the environment variable for your operating environment using one of the following procedures:
  - [Set the locale variable on Windows operating environments](#) (see page 191).
  - [Set the locale variable on UNIX or Linux operating environments](#) (see page 193).
3. (Optional) [Verify the locale variable setting on windows operating environments](#) (see page 192).
4. (Optional) Repeat Steps 1 through 3 to set any other components in your environment to the same language.

## Determine the IANA Code for Your Language

Each language has a unique code. The Internet Assigned Numbers Authority (IANA) assigns these language codes. Adding a language code to a locale variable changes the language that the software displays. Determine the proper code for the language that you want before creating the locale variable.

The following table lists the IANA codes that correspond to the languages supported by the software:

Language	IANA Code
Brazilian Portuguese	pt_BR
French	fr
German	de
Italian	it
Japanese	ja
Korean	ko
Simplified Chinese	zh-Hans
Spanish	es

**Note:** A list of IANA language codes is available from this [third-party website](#).

## Environment Variables

The environment variables are settings by which users can customize a computer to suit their needs. Examples of environment variables include the following items:

- A default directory for searching or storing downloaded files.
- A username.
- A list of locations to search for executable files (path).

Windows operating environments allow global environment variables, which apply to all users of a computer. The environment variables on UNIX or Linux operating environments must be set for each user or program.

To set the locale variable, pick the procedure for your operating environment from the following list:

- [Set the locale variable on Windows operating environments](#) (see page 191).
- [Set the locale variable on UNIX or Linux operating environments](#) (see page 193).

## Set the Locale Variable on Windows Operating Environments

The following locale variable specifies the language settings for the software:

`SM_ADMIN_LOCALE`

Create this variable and set it to the language that you want. Set this variable on *each* component for which you want to use another language. For example, suppose you want to have a Policy Server and an agent that is set to French. Set this variable on both of those components to French.

**Note:** The installation or configuration programs do *not* set this variable.

**Follow these steps:**

1. Click Start, Control Panel, System, Advanced system settings.

The system properties dialog appears.

2. Click the Advanced tab.
3. Click Environment Variables.
4. Locate the System variables section, and then click New.

The New System Variable dialog opens with the cursor in the Variable name: field.

5. Type the following text:

`SM_ADMIN_LOCALE`

6. Click the Variable name: field, and then type the [IANA language code](#) (see page 190) that you want.
7. Click OK.

The New System Variable dialog closes and the `SM_ADMIN_LOCALE` variable appears in the list.

8. Click OK *twice*.

The locale variable is set.

9. (Optional) Repeat Steps 1 through 8 to set other components to the same language.

## Verify the Locale Variable Value on Windows Operating Environments

You can verify the value to which the locale variable is set at any time. You can do this procedure after setting the variable to confirm that it is set correctly.

**Note:** Instructions for verifying the variable value on UNIX and Linux are in the [setting procedure](#) (see page 193).

### Follow these steps:

1. Open a command-line window with the following steps:
  - a. Click Start, Run.
  - b. Type the following command:  
`cmd`
  - c. Click OK.

A command-line window opens.

2. Enter the following command:

```
echo %SM_ADMIN_LOCALE%
```

The locale appears on the next line. For example, when the language is set to German, the following code appears:

```
de
```

The value of the locale variable is verified.



## Set the Locale Variable on UNIX or Linux Operating Environments

The following locale variable specifies the language settings for the software:

`SM_ADMIN_LOCALE`

Create this variable and set it to the language that you want. Set this variable on *each* component for which you want to use another language. For example, suppose you want to have a Policy Server and an agent that is set to French. Set this variable on both of those components to French.

**Note:** The installation or configuration programs do *not* set this variable.

### Follow these steps:

1. Log in to the computer that is running the component that you want.
2. Open a console (command-line) window.
3. Enter the following command:

```
export SM_ADMIN_LOCALE=IANA_language_code
```

The command in the following example sets the language to French:

```
export SM_ADMIN_LOCALE=fr
```

The locale variable is set.

4. (Optional) Verify that the locale variable is set properly by entering the following command:

```
echo $SM_ADMIN_LOCALE
```

The locale appears on the next line. For example, when the language is set to German, the following code appears:

```
de
```

5. (Optional) Repeat Steps 1 through 4 to set other components to the same language.



# Chapter 11: Advanced Options

---

This section contains the following topics:

[How to Enable SSL for the Agent for SharePoint](#) (see page 196)

[How to Configure Multiple User Directories](#) (see page 243)

[How to Configure SLO for SharePoint 2013](#) (see page 263)

[How to Configure Single Logout on SharePoint 2010](#) (see page 278)

[How to Monitor Data with CA Introscope](#) (see page 289)

[How to Use the Session Linker](#) (see page 299)

[How to Replace the Certificates for your CA SiteMinder® Trusted Identity Provider](#) (see page 305)

[Virtual Hosts with the Agent for SharePoint](#) (see page 311)

[How to Modify the Sign-In URL of your CA SiteMinder® Trusted Identity Provider](#) (see page 322)

[Configure the Agent for SharePoint for Web Applications That Use NTLM Authentication](#) (see page 327)

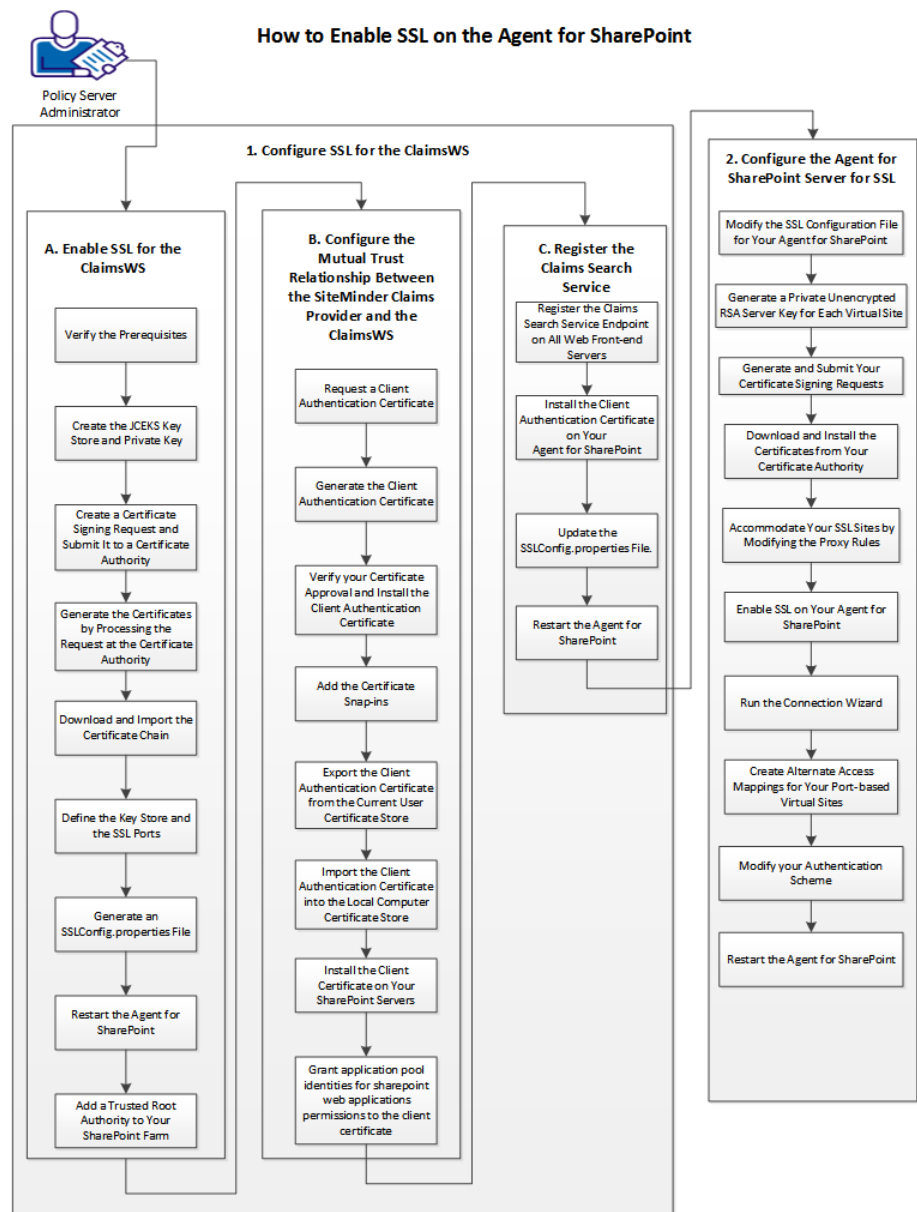
[How to replace the \(WS-Fed\) token signing certificates](#) (see page 328)

## How to Enable SSL for the Agent for SharePoint

The procedure for enabling Secure Sockets Layer (SSL) communications on the Agent for SharePoint has the following parts:

- Configuring SSL for the ClaimsWS
  - Protecting the ClaimsWS service with SSL
  - Configuring the mutual trust relationship between the ClaimsWS and the CA SiteMinder® claims provider.
  - Registering the Claim Search Service
- Configuring the Agent for SharePoint (reverse proxy) server for SSL.

The following graphic describes these procedures:



**Follow these steps:**

1. Enable SSL for the ClaimsWS service with the following steps:
  - a. [Verify the prerequisites](#) (see page 199).
  - b. [Create the JCEKS key store and private key](#) (see page 200).
  - c. [Create a certificate signing request and submit it to a certificate authority](#) (see page 202).
  - d. [Generate the certificates by processing the request at the certificate authority](#) (see page 204).
  - e. [Download and import the certificate chain](#) (see page 205).
  - f. [Define the Key Store and the SSL ports](#) (see page 206).
  - g. [Generate an SSLConfig.properties file](#) (see page 207).
  - h. [Restart the Agent for SharePoint](#) (see page 207).
  - i. [Add a trusted root authority to your SharePoint farm](#) (see page 210).
2. Configure the mutual trust relationship between the CA SiteMinder® claims provider and the ClaimsWS service with the following steps:
  - a. [Request a client authentication certificate](#) (see page 211).
  - b. [Generate the client authentication certificate](#) (see page 213).
  - c. [Verify your certificate approval and install the client authentication certificate](#) (see page 214).
  - d. [Add the certificate snap-ins](#) (see page 215).
  - e. [Export the client authentication certificate from the current user certificate store](#) (see page 215).
  - f. [Import the client authentication certificate into the local computer certificate store](#) (see page 217).
  - g. [Install the client certificate on your SharePoint servers](#) (see page 218).
  - h. [Grant application pool identities for sharepoint web applications permissions to the client certificate](#) (see page 218).
3. Register the Claims WS service with the following steps:
  - a. [Register the claims search service end point on all web front-end \(WFE\) servers](#) (see page 220).
  - b. Install the client authentication certificate on your Agent for SharePoint.
  - c. [Update the SSLConfig.properties file](#) (see page 223).
  - d. [Restart the Agent for SharePoint](#) (see page 207).

4. Configure the Agent for SharePoint server for SSL with the following steps:
  - a. [Modify the SSL configuration file for your Agent for SharePoint](#) (see page 226).
  - b. [Generate a private unencrypted RSA server key for each virtual site](#) (see page 229).
  - c. [Generate and submit certificate signing requests](#) (see page 232).
  - d. [Download and install the certificates from your certificate authority](#) (see page 233).
  - e. [Accommodate your SSL sites by modifying the proxy rules](#) (see page 234).
  - f. [Enable SSL on your Agent for SharePoint](#) (see page 235).
  - g. [Run the connection wizard](#) (see page 236).
  - h. [Create alternate access mappings for your port-based virtual sites](#) (see page 239).
  - i. [Modify your authentication scheme](#) (see page 240).
  - j. [Restart the Agent for SharePoint](#) (see page 207).

## Verify the Prerequisites

The first step in protecting the ClaimsWS service is verifying the prerequisites.

Verify the following prerequisites before protecting the Claims WS service with SSL:

- Farm administrator privileges and local administrator privileges for each SharePoint server in the farm.
- The *java\_home* variable in your environment points to the proper JDK installation directory.  
For example, if you are using Java 1.6, your *java\_home* variable must point to the installation directory for the Java 1.6 JDK.
- For UNIX/Linux operating environments, verify the following conditions:
  - The Agent for SharePoint environment variables are exported to your environment. Run the following script:

```
Agent - for -SharePoint_home\ca_sps_env.sh
```

## Create the JCEKS Key Store and Private Key

The next step in protecting the ClaimsWS service is creating a JCEKS key store and private key.

The JCEKS key store is a repository for the certificates and their related private keys. The certificates that you create are stored in the JCEKS key store. Creating a key store also creates a server certificate. This process requires the following information:

- An alias (nickname) for the server certificate you are requesting.
- A password for the JCEKS key store.
- The fully qualified domain name of the server hosting your Agent for SharePoint
- The name of your organizational unit (department or group)
- The name of your organization.
- The locality of your organization.
- The two-letter state and country codes for your organization.

### Follow these steps:

1. Log in to the system hosting your Agent for SharePoint.
2. Open a command-line window.
3. Navigate to the following directory:

*Agent\_for\_SharePoint\_home\SSL\keys*

#### **Agent-for-SharePoint\_Home**

Indicates the directory where the CA SiteMinder Agent for SharePoint is installed.

**Default:** (Windows) [32-bit] C:\Program Files\CA\Agent-for-SharePoint

**Default:** (Windows) [64-bit] C:\CA\Agent-for-SharePoint

**Default:** (UNIX/Linux) /opt/CA/Agent-for-SharePoint

4. Run the following command:

```
keytool -genkeypair -keyalg RSA -keystore .\ServerCert.jceks -alias Alias_Name
-storetype JCEKS -storepass keystore_password
```

The following table lists the prompts from the JCEKS keytool utility and sample responses:

Keytool Prompt:	Sample Response:	Purpose:
What is your First and Last Name?	agentforsharepointserver.example.com	Fully qualified domain name (FQDN) of the server hosting your Agent for SharePoint.



What is your Organizational Unit?	support	Department or group name
What is your Organization?	example	Name of your organization
What is your City or Locality?	Your City	City or Town
What is your State?	YS	Two-letter state or province abbreviation
What is your Country Code?	YC	Two-letter country code

The keytool utility displays a confirmation resembling the following example:

Is the following correct:

```
cn=agentforsharepointserver.example.com,ou=support,o=example,l=Your
City,st=YS,c=YC
```

5. Enter yes.

The keystore and private key are created.

6. Leave the command-line window *open*, and continue with the next step of creating a certificate request.

## Create a Certificate Signing Request and Submit It to a Certificate Authority

The next step in protecting the ClaimsWS service involves creating a certificate signing request for the server certificate in your JCEKS key store.

A signing request submits the certificate to a certificate authority. The certificate authority validates (signs) the certificate. Certificates that are signed third-party certificate authorities are considered more secure than self-signed certificates.

Self-signed certificates are acceptable for evaluation or testing environments.

To submit a certificate signing request, you need the following information:

- The alias of your server certificate for the Agent for SharePoint.
- A file name for your certificate request (.csr file).
- The password for your JCEKS key store.

### Follow these steps:

1. Create a certificate signing request with the following command:

```
keytool -certreq -v -alias Alias_Name -sigalg MD5withRSA -file
.\file_name_of_certificate_request.csr -keypass keystore_password -keystore
ServerCert.jceks -storepass keystore_password -storetype JCEKS
```

The keytool utility produces a certificate signing request similar to the following example:

```
-----BEGIN NEW CERTIFICATE REQUEST-----
MIIBrzCCARgCAQAwbzELMAkGA1UEBhMCVVMxCzAJBgNVBAGTAk1BMRMwEQYDVQQHEwpGcmFtaW5n
aGFtMQswCQYDVQQKEwJDQTEPMA0GA1UECxMGU01URVNUSAwHgYDVQQDExdzbXNwcziwMTAuc210
...
...
...
dsrZKqtNaqym7DrkSql7LsUGcsACUp1K4PU6t3P16CKvagspJ18zwTqTRpkGtbu6emvEwpcQveuW
k27YooCZ4XDzFxtAnv9EI17L4N4QHHxXCa8kIUL0dGtJ4vD
-----END NEW CERTIFICATE REQUEST-----
```

2. Copy the entire certificate signing request.
3. Close the command-line window.

4. Submit the certificate signing request to a certificate authority with the following steps:

**Note:** This procedure demonstrates submitting a request to a Microsoft Active Directory Certificate Services certificate authority.

- a. Open your Web browser, and then navigate to the following URL:

`https://fully_qualilfied_domain_name_of_server_running_active_directory_certificate_services/certsrv`

**Note:** An example of such a URL is  
`http://certificateauthority.example.com/certsrv.`

- b. Click Request a certificate.
- c. Click the advanced certificate request link.
- d. Click the Create and submit a request to this CA.
- e. An Advanced Certificate Request form appears.
- f. Complete the form by doing the following tasks:

- Submitting a request for a PKCS # 7 file.
- Copying your certificate signing request into the field

**Note:** Under the type of certificate needed drop-down list, verify that Client Authentication Certificate appears.

- g. Click Submit.

A confirmation dialog appears.

- h. Click Yes.

The request is submitted. Note your request ID for future reference.

## Generate the Certificates by Processing the Request at the Certificate Authority

The next step in protecting the ClaimsWS service is having a certificate authority process your request.

After the certificate authority receives your certificate signing request, they will process the request and will return the signed certificate.

Some organizations use third-party certificate authorities to sign their certificate requests. Other organizations could possibly have an internal group that operates a certificate authority.

The following procedure demonstrates the process for approving a certificate with Microsoft Active Directory Certificate services:

### **Follow these steps:**

Certificate administrators approve or reject certificate requests. Certificate administrator privileges are separate from the Administrator privileges in the Windows operating environment. Not all users who have accounts on the computer hosting Active Directory Certificate services have sufficient privileges to approve or reject certificates.

Use this procedure if you have certificate administrator privileges. Otherwise, ask the certificate administrator in your organization to issue the certificate for you.

### **Follow these steps:**

1. Log in to the web server hosting the Active Directory Certificate services using an account with Certificate administrator privileges.
2. Click Start, Administrative Tools, Certification Authority.

The certsrv snap-in appears.

3. Click the name of the certification authority, and then click the pending request folder.

A list of pending certificate requests appears.

4. Right-click the request ID associated with the request for the client certificate.
5. From the context menu, select All Tasks, Issue.

The certificate is issued.

Continue with the next step of downloading and importing the certificate.

## Download and Import the Certificate Chain

The next step in protecting the ClaimsWS service is downloading and importing the certificate chain.

After your certificate has been signed, download and install the following items to the server hosting your Agent for SharePoint:

- The signed certificate.
- The certificate chain (any additional certificate-authority certificates that your certificate-authority issued).

The certificate chain validates your certificate to the web browsers of your users.

This process requires the following information:

- The alias (nickname) of the server certificate you are requesting.
- The password for the JCEKS key store.

### Follow these steps:

1. Log in to the server hosting your Agent for SharePoint.
2. Download the following files with the *same* Web browser from which sent the certificate signing request:
  - certnew.cer (your signed certificate)
  - certnew.p7b (the certificate chain)
3. Move the files that you downloaded in Step 2 to the following directory:  
Agent\_for\_SharePoint\_home/SSL/keys
4. Import the certificate chain into the keystore with the following command;  

```
keytool -importcert -v -noprompt -alias Alias_Name -file .\certnew.p7b -keypass
keystore_password -keystore ServerCert.jceks -storepass keystore_password
-storetype JCEKS
```
5. Continue with the next step of defining the claims store and the SSL ports.

## Define the KeyStore and the SSL Ports

The next step in protecting the ClaimsWS service is defining the key store and SSL ports.

After downloading and importing the certificate chain to the server hosting the Agent for SharePoint, add the following settings:

- The local SSL port number (defined when you ran the SharePoint Connection wizard).
- The path to the key store on the server that is hosting the Agent for SharePoint.

These settings are defined in the `server.conf` file.

### Follow these steps:

1. Open the following file with a text editor:

`Agent_for_SharePoint_home\proxy-engine\conf\server.conf`

Locate the following section of the file:

`<localapp>`

2. In the `<localapp>` section, locate the following line:

`#local.https.port=port_number`

3. Remove the `#` from the beginning of the previous line.
4. Verify that the port number following the equal sign matches what you entered for the Claims WS service SSL port in the SharePoint connection wizard. If you defined port number 2525 for your connection, the edited line would match the following example:

`local.https.port=2525`

5. Locate the following line:

`#local.https.keyStoreFileName="tomcat.keystore"`

6. Remove the `#` from the beginning of the previous line.
7. Replace the `tomcat.keystore` with the relative path to the keystore you created for the keys and certificates that are associated with the Claims WS service. If the relative path to your keystore is `ServerCert.jceks`, then the edited line would match the following example:

`local.https.keyStoreFileName="ServerCert.jceks"`

8. Save the file and close text editor.
9. Continue with the next step of generating an `SSLConfig.properties` file.

## Generate an SSLConfig.properties File

The next step of protecting the ClaimsWS service involves generating an SSLConfig.properties file for the keystore.

### Follow these steps:

1. On the server hosting your Agent for SharePoint, open a command-line window.
2. If you have not yet created the TrustStore, run the following command:

```
GenerateSSLConfig -keystorepass keystore_password
```

3. When prompted, enter the following values:

- *keystore\_password* (keystore password)
- false (Enable Client Authentication)

**Important!** Do not enable client authentication yet.

## Restart the Agent for SharePoint

Starting or stopping the Agent for SharePoint involves the following separate procedures:

1. [Changing the value of EnableWebAgent in the WebAgent.conf file](#) (see page 109).
2. [Changing the state of the related services on the computer running the Agent for SharePoint](#) (see page 110).

## Change the Value of the EnableWebAgent Parameter

Change the value of the EnableWebAgent parameter to accomplish either of the following tasks:

- Start the Agent for SharePoint when the related services start.
- Stop the Agent for SharePoint when the related services start.

### Follow these steps:

1. Open the following file with a text editor:

Agent - for - SharePoint\_home\proxy-engine\conf\defaultagent\WebAgent.conf

2. Locate the following line:

EnableWebAgent="NO"

3. Change the value inside the quotation marks to *one* of the following values:

- YES to start the Agent for SharePoint after the services start. Your resources are protected.
- NO to stop the Agent for SharePoint after the services start. Your resources are *not* protected.

4. [Change the state of the related services on your Agent for SharePoint](#) (see page 110).



## Change the States of the Services on your Agent for SharePoint

You can change the states of the related services on your Agent for SharePoint.

**Note:** To start or stop your Agent for SharePoint, [change the value of the EnableWebAgent parameter first](#) (see page 109).

**Follow these steps:**

1. To change the states of the related services, select *one* of the following procedures:
  - For Windows operating environments, go to Step 2.
  - To *start* the Agent for SharePoint on UNIX operating environments, go to Step 3.
  - To *stop* the Agent for SharePoint on UNIX operating environments, go to Step 4.
2. For Windows operating environments, do the following steps:
  - a. From the Windows Start menu navigate to Administrative Tools, Services.  
The Services dialog appears.
  - b. Scroll down the list of services and select SiteMinder Agent for SharePoint.
  - c. From the Action menu, select All Tasks and select the command that you want.
  - d. Repeat Step b for SiteMinder Agent for SharePoint Proxy Engine.  
The states of the services and Agent for SharePoint are changed.
3. To start the Agent for SharePoint on UNIX operating environments, do the following steps.
  - a. Log in as a root user.
  - b. Navigate to the following directory:  
*Agent-for-SharePoint\_home/proxy-engine*
  - c. Run the following command:  
  
`./sps-ctl start`  
  
The service and the Agent for SharePoint start. The Agent for SharePoint stops or starts according to the [value you set in the EnableWebAgent parameter](#) (see page 109).
4. To stop the Agent for SharePoint on a system running UNIX, do the following steps:
  - a. Navigate to the following directory:  
*Agent-for-SharePoint\_home/proxy-engine*
  - b. Run the following command:  
  
`./sps-ctl stop`  
  
The service and the Agent for SharePoint stop.

## Add a Trusted Root Authority to your SharePoint Farm

The next step in protecting the ClaimsWS service is adding a trusted root authority to your SharePoint farm.

Your SharePoint farm requires a new trusted root authority to identify and authenticate the information that it receives from the claims service. Create a trusted root authority on your SharePoint 2010 central administration server.

### Follow these steps:

1. Copy the certificates for the ClaimsWS service from the system hosting your Agent for SharePoint, to a directory on your SharePoint central administration server. Include the signed certificate that you downloaded from your certificate authority (certnew.cer file) and all the certificates in the certificate chain (certnew.p7b).
2. Open the SharePoint 2010 central administration site.
3. Click Security.
4. Under General Security, click Manage trust.
5. Click New.

The Create Trusted Relationship dialog appears.

6. Enter a name for the trust relationship.
7. Click the Browse button next to the Root Authority Certificate, and then locate the certificate that you copied over in Step 1.
8. Click OK.
9. Repeat Steps 1 through 8 for each Certificate Authority certificate in your certificate chain. For example, if your certificate chain includes three certificates, repeat this step three times.

The trusted root authority is created.

10. Continue by configuring the mutual trust relationship between the CA SiteMinder® claims provider and the ClaimsWS.

## Request a Client Certificate

A mutual trust relationship between the following components is required for secure communications:

- The SharePoint claims search service.
- The CA SiteMinder® claims provider.

The first step in creating this relationship is requesting a client authenticate certificate. This certificate is installed on all SharePoint web front-end (WFE) servers. The client authentication certificate allows the ClaimsWS service to verify the identities of the WFE servers.

Several third-party tools are available for creating certificates. This procedure provides one possible example using Active Directory Certificate services and IIS 7.

If your organization uses different tools or procedures to create client certificates, use those tools or procedures instead.

If you already have a client authentication certificate, skip this procedure.

### Follow these steps:

1. Open a Web browser (from a system running an IIS web server).
2. Navigate to the following URL:

`https://fully_qualilfied_domain_name_of_server_running_active_directory_certificate_services/certsrv`

An example of such a URL is `http://certificateauthority.example.com/certsrv`.

3. Click Request a certificate.

The Request a certificate screen appears.

4. Click the advanced certificate request link.
5. Click the Create and submit a request to this CA.

An Advanced Certificate Request form appears.

6. Complete the form, using the following examples as a guide:

Name: SiteMinderClaimsProvider  
E-Mail: admin@support.example.com  
Company: Example  
Department: Support  
City: your\_city  
State: your\_state  
Country/Region your\_country  
Type of Certificate Needed: Client Authentication Certificate  
Mark keys as exportable: ENABLED  
Friendly Name: SiteMinderClaimsProvider

**Note:** Under the type of certificate needed drop-down list, verify that Client Authentication Certificate appears.

7. Click Submit.

A confirmation dialog appears.

8. Click Yes.

The request is submitted.

9. Note the following items for future reference:

- Your request ID.
- Verify the status of your request using the same browser within ten days.

## Generate the Client Authentication Certificate

The next step in configuring a mutual trust relationship between the claims search service and the claims provider is generating the client authentication certificate.

The next step in protecting the ClaimsWS service is having a certificate authority process your request.

After the certificate authority receives your certificate signing request, they will process the request and will return the signed certificate.

Some organizations use third-party certificate authorities to sign their certificate requests. Other organizations could possibly have an internal group that operates a certificate authority.

The following procedure demonstrates the process for approving a certificate with Microsoft Active Directory Certificate services:

### **Follow these steps:**

Certificate administrators approve or reject certificate requests. Certificate administrator privileges are separate from the Administrator privileges in the Windows operating environment. Not all users who have accounts on the computer hosting Active Directory Certificate services have sufficient privileges to approve or reject certificates.

Use this procedure if you have certificate administrator privileges. Otherwise, ask the certificate administrator in your organization to issue the certificate for you.

### **Follow these steps:**

1. Log in to the web server hosting the Active Directory Certificate services using an account with Certificate administrator privileges.
2. Click Start, Administrative Tools, Certification Authority.

The certsrv snap-in appears.

3. Click the name of the certification authority, and then click the pending request folder.

A list of pending certificate requests appears.

4. Right-click the request ID associated with the request for the client certificate.
5. From the context menu, select All Tasks, Issue.

The certificate is issued.

Continue with the next step of downloading and importing the certificate.

## Verify Your Certificate Approval and Install Your Client Authentication Certificate

The next step in creating a mutual trust relationship is verifying your approval and installing your client authentication certificate. Your IIS web server must have the client authentication certificate installed first before installing it on any SharePoint central administration or web front-end (WFE) servers.

Verify the status of your certificate request using the *same* IIS web server *and* Web browser from which you submitted the request. If your certificate is approved, install the certificate on your IIS web server first.

### Follow these steps:

1. Open the same Web browser that you used to request your certificate on your system hosting an IIS web server.
2. Navigate to the following URL:  
  
`https://fully_qualilfied_domain_name_of_server_running_active_directory_certificate_services/certsrv`  
  
An example of such a URL is `https://certificateauthority.example.com/certsrv`.
3. Click View the status of a pending certificate request.  
  
A list of your certificate requests appears.
4. Click the link for your certificate request.  
  
The Certificate Issued screen appears. If it does not, contact the certificate administrator in your organization for more information.
5. Click the Install Certificate link.  
  
A confirmation dialog appears.
6. Click Yes.  
  
The certificate is installed under My User Account on your IIS web server. Continue with the next step of installing the certificate snap-ins on your IIS web server.

## Add the Certificate Snap-ins

The next step for creating a mutual trust relationship between the Claims WS and the CA SiteMinder® claims provider is adding the certificate snap-ins.

The following accounts on your IIS web server require the certificate snap-in:

- Local computer
- My user account

**Follow these steps:**

1. Click Start, Run.  
The Run dialog appears.
2. Type mmc in the Open field, and then click OK.  
The Microsoft Management console appears.
3. Click File, Add/Remove Snap-in.  
The Add or Remove Snap-ins dialog appears.
4. In the Available snap-ins list, click Certificates, and then click Add.  
The Certificates snap-in dialog appears.
5. Select the Computer account option button, and then click Next.
6. Select the Local computer option button, and then click Finish.  
The Certificates snap-in dialog closes. The Certificates snap-in appears in the Selected snap-ins list.
7. Click Certificates in the Available snap-ins list, and then click Add.  
The Certificates snap-in dialog appears.
8. Select the My User Account option button, and then click Finish.
9. Click OK.  
The Add or Remove Snap-ins dialog closes. The certificate snap-ins are added.
10. Save your instance of the console for future use. Otherwise, the snap-ins do not appear in the future.

## Export the Client Authentication Certificate from the Current User Certificate Store

The next step for creating the mutual trust relationship is exporting the client certificate from the current user certificate store.

The Windows operating environment uses several different locations within the same computer to store certificates. These locations vary depending on the user account type. Installing your client authentication certificate on your IIS web server placed it in the following store:

- Certificates, Current User, Personal, Certificates

Export the certificate from the current user certificate store so it can be added to the other certificate stores on the computer.

**Follow these steps:**

1. Click Start, Run.  
The Run dialog appears.
2. Type mmc In the Open field, and then click OK.  
The Microsoft Management console appears.
3. Expand the console root folder, and then click "Certificates - Current User".
4. Expand "Certificates - Current User/Personal", and then double-click the 'Certificates' folder corresponding to where the certificate is stored.  
A list of certificates appears.
5. Right-click your client authentication certificate, and then select All Tasks, Export.  
The certificate export wizard opens.
6. Export the certificate using the Personal Information Exchange - PKCS #12 (.PFX) format with the following options:
  - Include all certificates in the certification path if possible
  - Export all extended properties
7. Click Next.
8. Enter and confirm a password to protect the private key and click Next.
9. Specify the name of the certificate file to export and click Next.
10. Note the certificate filename and verify the other settings and click Finish.

The client certificate is exported. Continue with the next step of importing the certificate into the local computer certificate store.



## Import the Client Authentication Certificate into the Local Computer Certificate Store

The next step for creating the mutual trust relationship is importing the client authentication certificate into the local computer certificate store.

Import the client authentication certificate into the following certificate store on your IIS web server.

- Certificates, Local computer

### Follow these steps:

1. Copy the client authentication certificate that you exported from the current user store to a directory on your IIS web server.
2. Click Start, Run.  
The Run dialog appears.
3. Type mmc in the Open field, and then click OK.
4. Expand Certificates (LocalComputer)
5. Expand Personal.  
The certificates folder appears.
6. Right-click the certificates folder, and then click All Tasks, Import.
7. Import the certificate.  
The certificate appears.
8. Double-click the client certificate. Verify that the General tab is selected.
9. Note the value in the Issued to field. You need this name to register the endpoint for the claims search service.

## Install the Client Authentication Certificate on your SharePoint Servers

The next step in establishing the mutual trust relationship is installing the client-authentication certificate on more servers.

Install the client authentication certificate that you exported from your IIS web server on the following servers in your SharePoint environment:

- Your SharePoint central administration server.
- All web front-end (WFE) servers in your SharePoint farm.

### Follow these steps:

1. Copy the exported client authentication certificate to a directory on your server.
2. Click Start, Run.  
The Run dialog appears.
3. In the Open field, type mmc and then click OK.
4. Expand Certificates — Local Computer.
5. Expand Personal.
6. The certificates folder appears.  
Right-click the certificates folder, and then click All Tasks, Import.
7. Import the client certificate.  
The certificate appears.
8. Double-click the client certificate. Verify that the General tab is selected.
9. Note the value in the Issued to field. You need this name to register the endpoint for the claims search service.
10. Repeat Steps 1 through 9 on each server in your environment (your SharePoint central administration server and on *each* WFE server). For example, if you have one SharePoint central administration server and five WFE servers, perform this procedure six times.

The client authentication certificate is installed. Continue with the next step of granting permissions to the application pools.

## Grant Application Pool Identities for SharePoint Web Applications Permissions to the Client Certificate

All application pool identities that are associated with protected SharePoint web applications need read-only permissions to the client authentication certificate. Perform this procedure on all the following servers in your environment:

- Your SharePoint central administration server.
- All web front end (WFE) servers in your SharePoint farm.

**Follow these steps:**

1. To obtain the application pool identities, do the following steps:
  - a. In IIS Manager, go to the Application Pools section and click SharePoint Web Application Pool.
  - b. Click Advanced Settings and locate the application pool identity.
  - c. Make a note of the application pool identities.
2. To start the Microsoft Management console, do the following steps:
  - a. Click Start, Run.  
The Run dialog appears.
  - b. In the Open field, type mmc and then click OK.  
The Microsoft Management console appears.
3. Expand the console root folder, and then click Certificates — Local Computer.
4. Locate and right-click your client certificate, then select All tasks, Manage Private keys.  
The permissions dialog appears.
5. Grant each application pool identity that you noted in Step 1 read access to the client certificate.
6. Repeat Steps 1 through 5 on the SharePoint central administration server and all the WFE servers in your SharePoint farm. For example, if you have one SharePoint central administration server and five WFE servers, perform this procedure six times.

The permissions are granted. Continue with the next step of registering the claims search service endpoint on all WFE servers.

## Register the Claims Search Service Endpoint on all WFE Servers

The next step in establishing the mutual trust relationship is registering the claims search service endpoint on all WFE servers in your SharePoint farm.

Registering a new end point for the claims search service associates the secure connection with the client authentication certificate. A PowerShell script that is installed with the claims provider automates the registration process. Register the new end point for all of the web front end (WFE) servers in your SharePoint environment.

### Follow these steps:

1. Remove any previously registered CA SiteMinder® claims services from the WFE server by running the following script:

```
SharePointClaimsProvider_directory\scripts\Remove-SMClaimSearchService.ps1
-WebApplication "url_of_SharePoint_web_application"
```

The following example describes removing the registration of a previous claims search service endpoint for the following web applications:

- SharePoint\_webapplication.support.example.com:8189/ (runs on port 8189)
- SharePoint\_webapplication.support.example.com:8286/ (runs on port 8286)

```
.\Remove-SMClaimSearchService.ps1 -WebApplication
"http://SharePoint_webapplication.support.example.com:8189/"
```

```
.\Remove-SMClaimSearchService.ps1 -WebApplication
"http://SharePoint_webapplication.support.example.com:8286/"
```

2. Repeat Step 1 for each SharePoint web application on the WFE server
3. Gather the following information:

#### **-WebApplication url\_of\_SharePoint\_web application**

Specifies the URL associated with a SharePoint web application.

**Example:** http://SharePoint\_webapplication.support.example.com:/ (runs on the default port).

**Example:** http://SharePoint\_webapplication.support.example.com:81/ (runs on port 81).

**Example:** http://SharePoint\_webapplication.support.example.com:82/ (runs on port 82).

#### **-ClaimSearchService claims\_search\_service\_URL**

Specifies the URL of the claims search service.

**Limits:** If the claim search service uses SSL, specify the https: protocol.

**Example:**

https://claim\_search\_service.support.example.com:8002/ClaimsWS/services/  
WSSharePointClaimsServiceImpl

#### **-ClientCertificateName**

Specifies the value in the Issued To: field of your client authentication certificate. This client certificate protects the Claims WS (web service).

**Example:** SiteminderClaimsProvider

4. Open the SharePoint 2010 Management Shell.
5. Navigate to the following directory:

*SharePointClaimsProvider\_directory\scripts*

6. Enter the following command for your first web application:

```
.\Add-SMClaimSearchService.ps1 -WebApplication url_of_web_application url
-ClaimSearchService https://claims_search_service_url
-EnableSSLClientAuthentication -ClientCertificateName
name_in_Issued-To:_field_of_Certificate
```

The first end point is registered.

7. Repeat Step 4 for *each* SharePoint web application on the WFE server. The following example describes registering a claims search service endpoint for the following web applications:

- SharePoint\_webapplication.support.example.com:81 (runs on port 81)
- SharePoint\_webapplication.support.example.com:82 (runs on port 82)

```
.\Add-SMClaimSearchService.ps1 -WebApplication
http://SharePoint_webapplication.support.example.com:81/
-ClaimSearchService

https://claim_search_service.support.example.com:8002/ClaimsWS/services/W
SSharePointClaimsServiceImpl -EnableSSLClientAuthentication

-ClientCertificateName SiteminderClaimsProvider

. \Add-SMClaimSearchService.ps1 -WebApplication
http://SharePoint_webapplication.support.example.com:82/
-ClaimSearchService

https://claim_search_service.support.example.com:8002/ClaimsWS/services/W
SSharePointClaimsServiceImpl -EnableSSLClientAuthentication

-ClientCertificateName SiteminderClaimsProvider
```

8. Restart your WFE server.
9. Repeat Steps 1 through 8 on all of the web front end (WFE) servers in your SharePoint environment.

The claims search service endpoint is registered. Continue with the next step of creating a trusted store for the root certificate authority certificate.

## Create a Trusted Store for the Root Certificate Authority Certificate

The server on which your Agent for SharePoint runs also requires a separate trusted store for the root certificate authority certificates. If you use certificates signed by a third-party certificate authority, import the certificate authority certificate signed by the third party into this trusted store. If you are using a self-signed certificate import either the self-signed certificate or the associated public key into this trusted store.

**Important!** Do not use self-signed certificates in production environments. We recommend using self-signed certificates in test environments only.

### Follow these steps:

**Note:** This procedure provides one possible example of how to configure this feature using third-party tools. CA Technologies did *not* develop nor provide these tools. These tools are subject to change at any time by the third party without notice. Use this procedure as a guide for configuring this feature in your specific environment. The actual steps that are required in your situation could be different from the steps that are shown here.

1. Copy your certificate to the server on which your Agent for SharePoint runs.
2. Open a Command Prompt window.
3. Create a trusted store with the following command:

```
Keytool -importcert -alias alias_name -file path_to_root_certificate
-trustcacerts -keystore relative_path_to_trusted_store -storepass
trusted_store_password -storetype JCEKS
```

**Note:** We recommend using a relative location under the *Agent-for-SharePoint\_home\SSL\keys* directory

## Update the SSLConfig.properties File

The next step of the process of creating a mutual trust relationship is updating the SSLConfig.properties file.

The server that runs your Agent for SharePoint requires a password-protected location (trust store) for the client authentication certificate. Specify a password for the trust store when creating it.

### Follow these steps:

1. Run the following command on the server that runs your Agent for SharePoint:

```
GenerateSSLConfig -keystorepass keystore_password -truststore TrustStore.jceks
-truststorepass truststore_password
```

A confirmation prompt for your trust store password appears.

2. Re—enter your trust store password.

A confirmation prompt for client authentication appears.

3. Enter yes.

The SSLConfig.properties file is updated. Continue with the next step of restarting your Agent for SharePoint.

## Restart the Agent for SharePoint

Starting or stopping the Agent for SharePoint involves the following separate procedures:

1. [Changing the value of EnableWebAgent in the WebAgent.conf file](#) (see page 109).
2. [Changing the state of the related services on the computer running the Agent for SharePoint](#) (see page 110).

## Change the Value of the EnableWebAgent Parameter

Change the value of the EnableWebAgent parameter to accomplish either of the following tasks:

- Start the Agent for SharePoint when the related services start.
- Stop the Agent for SharePoint when the related services start.

### Follow these steps:

1. Open the following file with a text editor:

Agent - for - SharePoint\_home\proxy-engine\conf\defaultagent\WebAgent.conf

2. Locate the following line:

EnableWebAgent="NO"

3. Change the value inside the quotation marks to *one* of the following values:

- YES to start the Agent for SharePoint after the services start. Your resources are protected.
- NO to stop the Agent for SharePoint after the services start. Your resources are *not* protected.

4. [Change the state of the related services on your Agent for SharePoint](#) (see page 110).



## Change the States of the Services on your Agent for SharePoint

You can change the states of the related services on your Agent for SharePoint.

**Note:** To start or stop your Agent for SharePoint, [change the value of the EnableWebAgent parameter first](#) (see page 109).

**Follow these steps:**

1. To change the states of the related services, select *one* of the following procedures:
  - For Windows operating environments, go to Step 2.
  - To *start* the Agent for SharePoint on UNIX operating environments, go to Step 3.
  - To *stop* the Agent for SharePoint on UNIX operating environments, go to Step 4.
2. For Windows operating environments, do the following steps:
  - a. From the Windows Start menu navigate to Administrative Tools, Services.  
The Services dialog appears.
  - b. Scroll down the list of services and select SiteMinder Agent for SharePoint.
  - c. From the Action menu, select All Tasks and select the command that you want.
  - d. Repeat Step b for SiteMinder Agent for SharePoint Proxy Engine.  
The states of the services and Agent for SharePoint are changed.
3. To start the Agent for SharePoint on UNIX operating environments, do the following steps.
  - a. Log in as a root user.
  - b. Navigate to the following directory:  
*Agent-for-SharePoint\_home/proxy-engine*
  - c. Run the following command:  
  
`./sps-ctl start`  
  
The service and the Agent for SharePoint start. The Agent for SharePoint stops or starts according to the [value you set in the EnableWebAgent parameter](#) (see page 109).
4. To stop the Agent for SharePoint on a system running UNIX, do the following steps:
  - a. Navigate to the following directory:  
*Agent-for-SharePoint\_home/proxy-engine*
  - b. Run the following command:  
  
`./sps-ctl stop`  
  
The service and the Agent for SharePoint stop.

## Modify the SSL Configuration File for Your Agent for SharePoint

This section describes configuring secure communications between your Agent for SharePoint reverse proxy and the Public URLs of your SharePoint web applications.

The first step in configuring the reverse proxy for secure communications is modifying the SSL configuration file.

The SSL configuration file requires the following modifications:

- Add listening directives for each SSL port.
- Add virtual host sections for each port-based virtual host.

### Follow these steps:

1. Log in to the server hosting your Agent for SharePoint:
2. Open the following file with a text editor:

*Agent-for-SharePoint\_home\httpd\conf\extra\httpd-ssl.conf*

#### **Agent-for-SharePoint\_Home**

Indicates the directory where the CA SiteMinder Agent for SharePoint is installed.

**Default:** (Windows) [32-bit] C:\Program Files\CA\Agent-for-SharePoint

**Default:** (Windows) [64-bit] C:\CA\Agent-for-SharePoint

**Default:** (UNIX/Linux) /opt/CA/Agent-for-SharePoint

3. Add the appropriate number of 'Listen' directives for your environment. Use the following examples as a guide:
  - Listen 443 #(for the default http port 80)
  - Listen 481 #(for http port 81)
  - Listen 482 #(for http port 82)

The previous example assumes that you already have three web applications listening for HTTP requests on ports 80, 81 and 82. The previous example shows how to add HTTPS ports 443, 481 and 482 respectively.

4. Add a section for each port-based virtual host, using the following examples as a guide:

```
<VirtualHost _default_:443>
General setup for the virtual host
DocumentRoot "C:/CA/Agent-for-SharePoint/httpd/htdocs"
ServerName SMSPA2010.sptest.ca.com:443
ServerAdmin Admin@sptest.ca.com
ErrorLog logs/error_log.log
TransferLog logs/access_log.log
SSLEngine on
SSLCertificateFile
"C:/CA/Agent-for-SharePoint/SSL/certs/smspa2010.sptest.ca.com.cer"
SSLCertificateKeyFile
"C:/CA/Agent-for-SharePoint/SSL/keys/smspa2010.sptest.ca.com.key"
</VirtualHost>

<VirtualHost *:481>
DocumentRoot "C:/CA/Agent-for-SharePoint/httpd/htdocs/481smspa2010"
ServerName smspa2010.sptest.ca.com
ServerAdmin Admin@sptest.ca.com
ErrorLog logs/481smspa2010_error_log.log
TransferLog logs/481smspa2010_access_log.log
SSLEngine on
SSLCertificateFile
C:/CA/Agent-for-SharePoint/SSL/certs/smspa2010.sptest.ca.com.cer
SSLCertificateKeyFile
C:/CA/Agent-for-SharePoint/SSL/keys/smspa2010.sptest.ca.com.key
CustomLog logs/cipher_log_481smspa2010 \
"%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \"%r\" %b"
</VirtualHost>

<VirtualHost *:482>
DocumentRoot "C:/CA/Agent-for-SharePoint/httpd/htdocs/482smspa2010"
ServerName smspa2010.sptest.ca.com
ServerAdmin Admin@sptest.ca.com
ErrorLog logs/482smspa2010_error_log.log
TransferLog logs/482smspa2010_access_log.log
SSLEngine on
SSLCertificateFile
C:/CA/Agent-for-SharePoint/SSL/certs/smspa2010.sptest.ca.com.cer
SSLCertificateKeyFile
C:/CA/Agent-for-SharePoint/SSL/keys/smspa2010.sptest.ca.com.key
CustomLog logs/cipher_log_482smspa2010 \
"%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \"%r\" %b"
</VirtualHost>
```

The previous example describes the virtual host entries that are created to match the port settings in Step 2.

5. Save the file and close the text editor.

The SSL Configuration file is modified. Continue with the next step of generating certificates and keys for each unique server (FQDN) in your environment.

## Generate a Private Key

The next step in configuring the reverse proxy for secure communications is to generate a private (Windows) RSA Key (server key) for each virtual site with a fully qualified domain name (FQDN). Do one of the following procedures:

- [Generate a private unencrypted RSA server key for each virtual site](#) (see page 229)
- [Generate a private encrypted RSA server key for each virtual site](#) (see page 230)

## Generate a Private Unencrypted RSA Server Key for Each Virtual Site

Generate a private key for each virtual site with a fully qualified domain name (FQDN). This procedure describes how to generate an unencrypted private key.

### Follow these steps:

1. Open a command-line window.
2. Navigate to the following directory

*Agent-for-Sharepoint\_home\SSL\bin*

#### **Agent-for-SharePoint\_Home**

Indicates the directory where the CA SiteMinder Agent for SharePoint is installed.

**Default:** (Windows) [32-bit] C:\Program Files\CA\Agent-for-SharePoint

**Default:** (Windows) [64-bit] C:\CA\Agent-for-SharePoint

**Default:** (UNIX/Linux) /opt/CA/Agent-for-SharePoint

3. Generate the keys by running the following commands:

```
.\openssl genrsa -out ..\keys\server_FQDN.key [numbits]
```

#### ***server\_FQDN***

Specifies the fully qualified domain name of the server.

#### ***numbits***

(Optional) Specifies the size of the private key to generate in bits.

**Default:** 512

**Range:** 512 - 2048

The following example describes creating a 2048-bit key for a server named smspa2010:

```
.\openssl genrsa -out ..\keys\smspa2010.example.com.key 2048
```

4. Repeat Step 3 for each virtual server.

The private unencrypted server keys are created. Continue with the next step of generating a certificate signing request.

## Generate a Private Encrypted RSA Server Key for Each Virtual Site

Generate a private key for each virtual site with a fully qualified domain name (FQDN). This procedure describes how to generate an encrypted private key.

### Follow these steps:

1. Open a command-line window.
2. Navigate to the following directory

*Agent-for-Sharepoint\_home\SSL\bin*

#### **Agent-for-SharePoint\_Home**

Indicates the directory where the CA SiteMinder Agent for SharePoint is installed.

**Default:** (Windows) [32-bit] C:\Program Files\CA\Agent-for-SharePoint

**Default:** (Windows) [64-bit] C:\CA\Agent-for-SharePoint

**Default:** (UNIX/Linux) /opt/CA/Agent-for-SharePoint

3. Generate the keys by running the following commands:

```
.\openssl genrsa -des3 -out ..\keys\server_FQDN.key [numbits]
```

**server\_FQDN**

Specifies the fully qualified domain name of the server.

**numbits**

(Optional) Specifies the size of the private key to generate in bits.

**Default:** 512

**Range:** 512 - 2048

The following example describes creating a 1024-bit key for a server named smspa2010:

```
.\openssl genrsa-des3 -out ..\keys\smspa2010.example.com.key 1024
```

4. Repeat Step 3 for each virtual server.

The private encrypted server keys are created and written to the specified key output file.

The key output file will be in encrypted ASCII PEM (from "Privacy Enhanced Mail") format.

Because the file is encrypted, you will be prompted for a pass-phrase to protect it and decrypt it later if you want. If you do not want your key to be protected, do not use the `-des3` argument in the command line.

To view the details of this RSA key, enter the following command:

```
openssl rsa -noout -text -in server.key
```

## Generate and Submit Certificate Signing Requests

The next step in configuring the reverse proxy for secure communications is generating the certificate signing requests for each of the virtual servers.

### Follow these steps:

1. Open a command-line window.
2. Generate the certificate signing requests by running the following command:

```
.\openssl req -config .\openssl.cnf -new -key ..\keys\server_FQDN.key -out
..\keys\server_FQDN.csr
```

The following example describes creating a certificate request for a server named smspa2010 on the support.example.com domain:

```
.\openssl req -config .\openssl.cnf -new -key
..\keys\smspa2010.support.example.com.key -out
..\keys\smspa2010.support.example.com.csr
```

3. Create your certificate request by adding the information at each prompt, as shown in the following example:

```
Country: Your_Country
State: Your_State
Locality: Your_Town
Organization: Example
Org. Unit: support
CN: smspa2010.support.example.com
E-Mail: admin@support.ca.com
Challenge Pwd: firewall
Optional name: blank
```

**Note:** The value for the common name (CN) must match the fully qualified domain name (FQDN) of the web server.

The system generates a certificate request with the certificate file name and a request number, as shown in the following example:

```
smspa2010.support.example.com.csr 8
```

4. Record the file name and certificate signing request for future reference.
5. Repeat Steps 2 through 4 for the other virtual servers.
6. Submit your certificate signing requests to the certificate authority that your organization uses.

The certificate signing requests are generated and submitted. Continue with the next step of downloading your certificates from your certificate authority.



## Download and Install the Certificates from your Certificate Authority

The next step in configuring the reverse proxy for secure communications is downloading the signed certificates from the certificate authority.

The virtual host sections in your SSL configuration file specify a certificate location for each virtual host. The SSLCertificateFile line in the following example specifies the location for the spa2010.support.example.com server:

```
SSLCertificateFile
"Agent-for-SharePoint_home/SSL/certs/smspa2010.support.example.com.cer
```

### **Agent-for-SharePoint\_Home**

Indicates the directory where the CA SiteMinder Agent for SharePoint is installed.

**Default:** (Windows) [32-bit] C:\Program Files\CA\Agent-for-SharePoint

**Default:** (Windows) [64-bit] C:\CA\Agent-for-SharePoint

**Default:** (UNIX/Linux) /opt/CA/Agent-for-SharePoint

### **Follow these steps:**

1. Log in to your Agent for SharePoint server from which you issued the certificate requests.
2. Review the SSL configuration file for the SSLCertificateFile lines.
3. Copy a certificate file to its respective location that is specified in the SSL Configuration file.
4. Repeat Step 3 for each unique server running a virtual host.

The certificates are downloaded. Continue with the next step of accommodating your SSL sites by modifying the proxy rules.

## Accommodate Your SSL Sites by Modifying the Proxy Rules

The next step in configuring the reverse proxy for secure communication is modifying the proxy rules for the server on which your Agent for SharePoint runs.

**Note:** Even if you are using only SSL, the proxy rules files require rules for both HTTP and HTTPS protocols.

### Follow these steps:

1. Open the following file with a text editor:

*Agent-for-SharePoint\_home\proxy-engine\conf\proxyrules.xml*

#### **Agent-for-SharePoint\_Home**

Indicates the directory where the CA SiteMinder Agent for SharePoint is installed.

**Default:** (Windows) [32-bit] C:\Program Files\CA\Agent-for-SharePoint

**Default:** (Windows) [64-bit] C:\CA\Agent-for-SharePoint

**Default:** (UNIX/Linux) /opt/CA/Agent-for-SharePoint

2. Modify the ProxyRules.xml file for the SSL sites by adding proxy rules that include the SSL port and the related web application. The following example shows the new rules in bold:

```
<nete:proxyrules xmlns:nete="http://smspa2010.sptest.ca.com/" debug="yes">
<nete:cond type="host" criteria="endswith">
<nete:case value="81">
<nete:forward>http://w2k8r2.sptest.ca.com:14056$0</nete:forward>
</nete:case>
<nete:case value="82">
<nete:forward>http://w2k8r2.sptest.ca.com:31415$0</nete:forward>
</nete:case>
<nete:case value="481">
<nete:forward>http://w2k8r2.sptest.ca.com:14056$0</nete:forward>
</nete:case>
<nete:case value="482">
<nete:forward>http://w2k8r2.sptest.ca.com:31415$0</nete:forward>
</nete:case>
<nete:default>
<nete:forward>http://w2k8r2.sptest.ca.com:31567$0</nete:forward>
</nete:default>
</nete:cond>
</nete:proxyrules>
```

3. Save the file and close the text editor.

The proxy rules are modified. Continue with the next step of enabling SSL on your Agent for SharePoint.

## Enable SSL on Your Agent for SharePoint

The next step in configuring the reverse proxy for secure communication is enabling SSL on the server that runs your Agent for SharePoint.

To enable SSL on your Agent for SharePoint, do one of the following procedures, as appropriate.

- [Enable SSL for an unencrypted private key on Windows](#) (see page 235)
- [Enable SSL for an unencrypted private key on UNIX](#) (see page 235)
- [Enable SSL for an encrypted private key](#) (see page 236)

### Enable SSL for an Unencrypted Private Key on Windows

To enable SSL for an unencrypted private key on Windows, generate an `spsapachessl.properties` file.

**Follow these steps:**

1. Open a command-line window with administrative privileges. Navigate to the following directory:

*Agent-for-SharePoint\_home\httpd\bin*

2. Run the following script file:

`configssl.bat -enable`

**Note:** If an overwrite warning appears, confirm that you want to overwrite the existing `spsapachessl.properties` file.

### Enable SSL for an Unencrypted Private Key on UNIX

To enable SSL for an unencrypted private key on UNIX, edit the `spsapachessl.properties` located in the following location:

*Agent-for-Sharepoint\_home/httpd/conf/spsapachessl.properties*

**Follow these steps:**

1. Open the `spsapachessl.properties` file in a text editor.
2. Search for the following line:  
`apache.ssl.enabled=`
3. Do one of the following tasks:
  - If the previous line does not exist, add it to the file. Then go to step 4.
  - If the previous line exists, go to Step 4.

4. Edit the line as follows:  
`apache.ssl.enabled=Y`
5. Save the changes to the `spsapachessl.properties` file and close the text editor.

## Enable SSL for an Encrypted Private Key

To enable SSL for an encrypted private key, generate an `spsapachessl.properties` file.

### Follow these steps:

1. Open a command-line window with administrative privileges.  
Navigate to the following directory:
  - (Windows) *Agent-for-SharePoint\_home*\httpd\bin
  - (UNIX) *Agent-for-SharePoint\_home*/httpd/bin
2. Run one of the following script files:
  - (Windows) `configssl.bat -enable passphrase`
  - (UNIX) `configssl.sh passphrase`

**Note:** If an overwrite warning appears, confirm that you want to overwrite the existing `spsapachessl.properties` file.

## Run the Connection Wizard

The next steps in configuring the reverse proxy for secure communications involve the following tasks:

- Running the connection wizard to change the protocol of the Authentication URL to HTTPS.
- Changing the SignIn URL on your SharePoint central administration server using several PowerShell commands.

### Follow these steps:

1. Edit the existing connection using the Connection Wizard with the following steps:
  - a. Log in to the server that runs your Agent for SharePoint.
  - b. Navigate to the following directory:  
*Agent-for-SharePoint\_home*/sharepoint\_connection\_wizard

- c. Do the appropriate step for your operating environment:
- Windows: Right-click the executable and then select Run as administrator.
  - Solaris: `sh ./ca-spconnect-12.0-sp3-sol.bin`
  - Linux: `sh ./ca-spconnect-12.0-sp3-rhel30.bin`

The SharePoint Connection wizard starts.

- d. Click Next.

The Login Details screen appears.

- e. Enter the following login for the Policy Server.

**Policy Server Name**

Specifies the Policy Server name or IP address.

**Username**

Specifies the Policy Server administrator username.

**Password**

Specifies the Policy Server administrator password.

**Agent Name**

Specifies the Agent-4x. The connection with the Policy Server is established using the details given in the Agent Name.

**Shared Secret Key**

Specifies the shared secret key that is associated with the Agent.

- f. Click Next

The Select Action screen appears.

- g. Select Edit a SharePoint Connection option.

- h. Click Next.

The SharePoint Connection Properties screen appears.

- i. Change the protocol of the Authentication URL to HTTPS in the SharePoint Connection Properties screen.

- j. Click Install in the Commit Details screen.

The Save Complete screen appears.

- k. Click Done.

The partnership details are saved, the SharePoint Connection is modified, and the wizard closes.

2. Modify the SignInUrl of the CA SiteMinder® Trusted Identity Token Issuer with the following steps:
  - a. Log in to your SharePoint central administration server.
  - b. Click Start, All Programs, Microsoft SharePoint 2010 Products, SharePoint 2010 Management Shell.
  - c. Verify the following settings by running the Get-SPTtrustedIdentityTokenIssuer command:
    - The name of the provider  
(such as *LDAP-Claims*)
    - The current SignInUrl  
(such as *http://smspa2010.support.example.com/affwebservices/public/wsfeddispatch*  
*er*).
  - d. Run the Set-SPTtrustedIdentityTokenIssuer command as shown in the following example:

```
Set-SPTtrustedIdentityTokenIssuer "LDAP-Claims" -SignInUrl
https://smspa2010.support.example.com/affwebservices/public/wsfeddispatch
er
```
  - e. Run the Get-SPTtrustedIdentityTokenIssuer command again to verify the change to the SignInUrl.

**Note:** For more information about the Set-SPTtrustedIdentityTokenIssuer command, see <http://technet.microsoft.com/en-us/library/ff607792.aspx>

The protocol is changed. Continue with the next step of creating alternate access mappings for your port-based virtual sites.

## Create Alternate Access Mappings for Your Port-Based Virtual Sites

The next step in configuring the reverse proxy for secure communication is creating alternate access mappings on your SharePoint server for the port-based virtual hosts on your Agent for SharePoint.

Port-based proxy rules require the following alternate access mappings on your SharePoint central administration server:

- Set the public URL for the zone to the URL of your virtual host on your Agent for SharePoint that is associated with the web application.
- Set the internal URL to the SharePoint server to which the requests from the virtual host on the Agent for SharePoint are forwarded.

**Follow these steps:**

1. Open your SharePoint central administration site
2. Click Application Management.
3. Under Web Applications, click Configure Alternate Access Mappings..
4. Use the examples in the following table as a guide to edit your public URLs and Add Internal URLs:

<b>Public URL</b> <b>(URL of your virtual site on your Agent for SharePoint)</b>	<b>Internal URL</b> <b>(URL of web application on your SharePoint server)</b>
https://support.example.com	https://spa2010.support.example.com \443

The alternate access mappings are created. Continue with the next step of modifying the ConfigSSL.bat file.

## Modify Your Authentication Scheme

The next step in configuring the reverse proxy for secure communication is modifying your CA SiteMinder® authentication scheme to use SSL.

Authentication schemes use HTTP unless you specify HTTPS when creating the authentication scheme.

**Follow these steps:**

1. Login to the Administrative UI.
2. Click Infrastructure, Authentication, Authentication Schemes.
3. Click the link of the authentication scheme that you want.
4. Click Modify.
5. Select the Use SSL Connection check box.
6. Click Submit.

A confirmation screen appears.

7. Click OK.

The authentication scheme is modified. Continue with the next step of restarting your Agent for SharePoint.

## Restart the Agent for SharePoint

Starting or stopping the Agent for SharePoint involves the following separate procedures:

1. [Changing the value of EnableWebAgent in the WebAgent.conf file](#) (see page 109).
2. [Changing the state of the related services on the computer running the Agent for SharePoint](#) (see page 110).



## Change the Value of the EnableWebAgent Parameter

Change the value of the EnableWebAgent parameter to accomplish either of the following tasks:

- Start the Agent for SharePoint when the related services start.
- Stop the Agent for SharePoint when the related services start.

### Follow these steps:

1. Open the following file with a text editor:

Agent - for - SharePoint\_home\proxy-engine\conf\defaultagent\WebAgent.conf

2. Locate the following line:

EnableWebAgent="NO"

3. Change the value inside the quotation marks to *one* of the following values:

- YES to start the Agent for SharePoint after the services start. Your resources are protected.
- NO to stop the Agent for SharePoint after the services start. Your resources are *not* protected.

4. [Change the state of the related services on your Agent for SharePoint](#) (see page 110).

## Change the States of the Services on your Agent for SharePoint

You can change the states of the related services on your Agent for SharePoint.

**Note:** To start or stop your Agent for SharePoint, [change the value of the EnableWebAgent parameter first](#) (see page 109).

### Follow these steps:

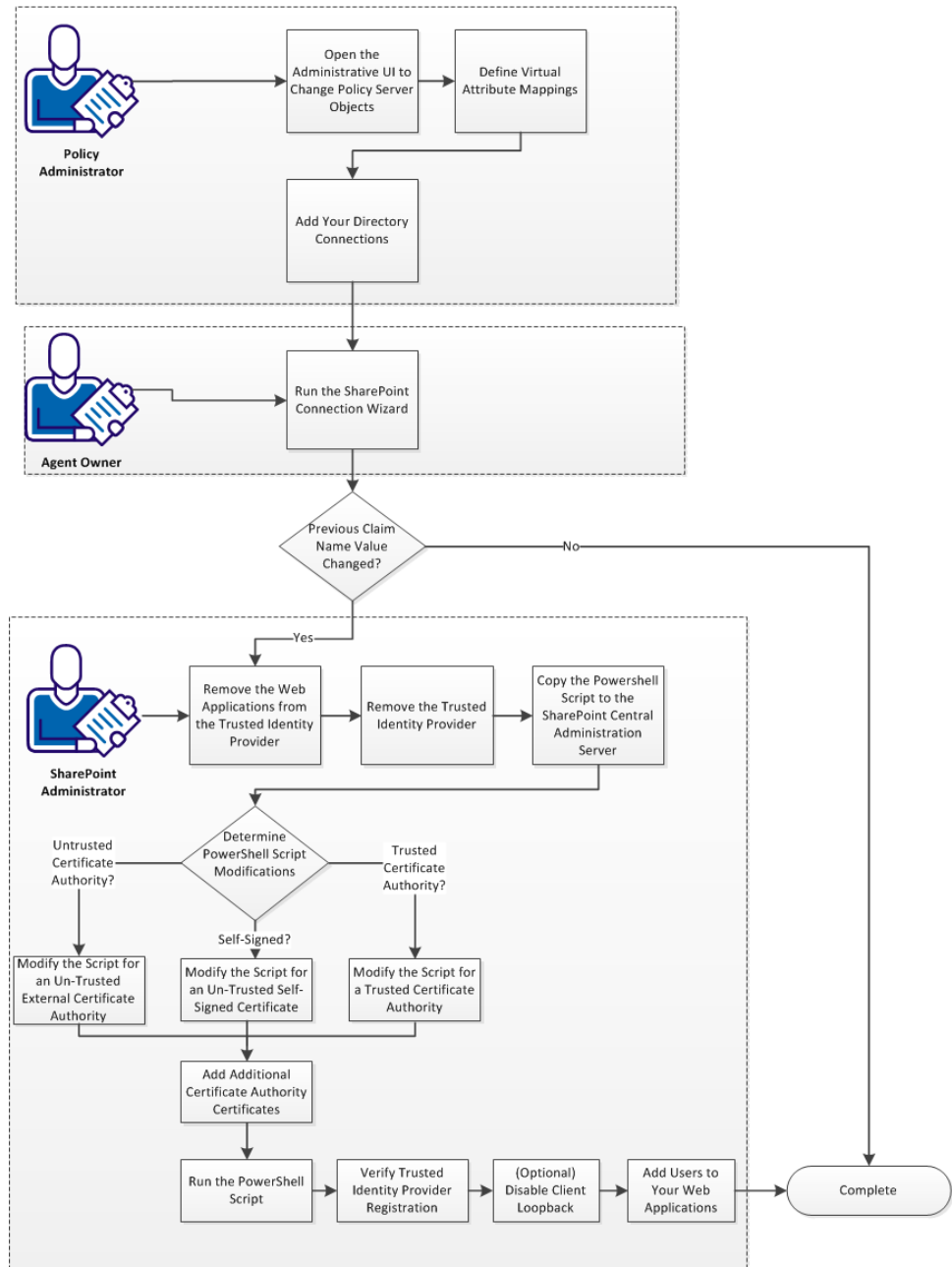
1. To change the states of the related services, select *one* of the following procedures:
  - For Windows operating environments, go to Step 2.
  - To *start* the Agent for SharePoint on UNIX operating environments, go to Step 3.
  - To *stop* the Agent for SharePoint on UNIX operating environments, go to Step 4.
2. For Windows operating environments, do the following steps:
  - a. From the Windows Start menu navigate to Administrative Tools, Services.  
The Services dialog appears.
  - b. Scroll down the list of services and select SiteMinder Agent for SharePoint.
  - c. From the Action menu, select All Tasks and select the command that you want.
  - d. Repeat Step b for SiteMinder Agent for SharePoint Proxy Engine.  
The states of the services and Agent for SharePoint are changed.
3. To start the Agent for SharePoint on UNIX operating environments, do the following steps.
  - a. Log in as a root user.
  - b. Navigate to the following directory:  
*Agent-for-SharePoint\_home/proxy-engine*
  - c. Run the following command:  
  
`./sps-ctl start`  
  
The service and the Agent for SharePoint start. The Agent for SharePoint stops or starts according to the [value you set in the EnableWebAgent parameter](#) (see page 109).
4. To stop the Agent for SharePoint on a system running UNIX, do the following steps:
  - a. Navigate to the following directory:  
*Agent-for-SharePoint\_home/proxy-engine*
  - b. Run the following command:  
  
`./sps-ctl stop`  
  
The service and the Agent for SharePoint stop.

## How to Configure Multiple User Directories

If the users who access your protected SharePoint web applications are stored in more than one user directory, configure multiple user directories.

**Important!** Multiple directory connections are supported with Policy Server version 12.5 and above only.

### How to Configure Multiple User Directories



**Follow these steps:**

1. [Open the Administrative UI to change Policy Server objects](#) (see page 42).
2. [Define virtual attribute mappings](#) (see page 245).
3. [Add directory connections](#) (see page 247).
4. [Run the SharePoint connection wizard](#) (see page 248).
5. If you *changed* the value of an *existing* Claim Name (attribute), do the following steps:
  - a. [Remove the web applications from the trusted identity provider](#) (see page 250).
  - b. [Remove the trusted identity provider](#) (see page 251).
  - c. [Copy the PowerShell script to the SharePoint central administration server](#) (see page 251).
  - d. [Determine the PowerShell script modifications](#) (see page 252) (pick *one* of the following procedures):
    - [Modify the script for an un-trusted certificate authority](#) (see page 253).
    - [Modify the script for an un-trusted self-signed certificate](#) (see page 255).
    - [Modify the script for a trusted certificate authority](#) (see page 257).
  - e. [Add certificate authority certificates](#) (see page 258).
  - f. [Run the PowerShell script](#) (see page 260).
  - g. [Verify the trusted identity provider registration](#) (see page 261).
  - h. [\(Optional\) Disable client loopback](#) (see page 150).
  - i. [Add users to your web applications](#) (see page 262).

## Open the Administrative UI to Change Policy Server Objects

Change the objects on your Policy Server by opening the Administrative UI.

### Follow these steps:

1. Open the following URL in a browser.

`https://host_name:8443/iam/siteminder/adminui`

#### **host\_name**

Specifies the fully qualified Administrative UI host system name.

2. Enter your CA SiteMinder® superuser name in the User Name field.
3. Enter the CA SiteMinder® superuser account password in the Password field.  
**Note:** If your superuser account password contains dollar-sign (\$) characters, replace each instance of the dollar-sign character with \$DOLLAR\$. For example, if the CA SiteMinder® superuser account password is \$password, enter \$DOLLAR\$password in the Password field.
4. Verify that the proper server name or IP address appears in the Server drop-down list.
5. Select Log In.

## Define Virtual Attribute Mappings

The next step in configuring multiple user directories is defining the virtual attribute mappings in each user directory. For example, suppose that some users exist in an Active Directory server while others exist in an LDAP directory server. Defining *virtual* attribute mappings, or aliases for each directory allows CA SiteMinder® access to both directories.

The following table provides examples of typical attribute mappings for an Active Directory server, an LDAP directory server, and an ODBC database:

User Attribute Field (in Administrative UI)	Active Directory Attribute Name	LDAP Directory Attribute Name	ODBC Attribute Name
UID	sAMAccountname	cn	Name
AliasID	sAMAccountname	cn	Name
mail	userPrincipalName	Mail	EmailAddress
smusergroups	Manager	Name	Name

**Follow these steps:**

1. Click Infrastructure, Directories, User Directories.
2. Click the Edit icon of a user directory that you want.
3. Create an attribute mapping with the following steps:
  - a. Scroll to the Attribute Mapping list, and then click Create.  
The Create Attribute Mapping pane opens.
  - b. Verify that Create a new object is selected, and click OK.  
The Create Attribute Mapping: Name pane opens.
  - c. Type a name and an optional description for the attribute mapping. For example, to create an attribute mapping for the UID, type UID.
  - d. Select the Alias option button.
  - e. In the Definition field, type the attribute name that you want to add, as shown in the following examples:
    - (Active Directory) sAMAccountname
    - (LDAP directory server) cn
    - (ODBC database) Name
  - f. Click OK.
4. Repeat Steps 3a through 3f until all of the attributes have been added to the user directory.
5. Click Submit.  
The attribute mappings are added to your directory.
6. Repeat Steps 2 through 6 to add attributes to another user directory.  
The attribute mappings are defined.

## Add Directory Connections

The next step in configuring multiple user directories is adding the user directory connections that contain the attribute mappings to the following items:

- Policy domains.
- Policy applications (EPM).

**Follow these steps:**

1. Pick the appropriate procedure for your type of policy from the following list:
  - If you use policy domains, go to Step 2.
  - If you use application policies (EPM), go to Step 4.
2. Add directory connections to your policy domain with the following steps:
  - a. Click Policies, Domain, Domains.
  - b. Click the edit icon of the domain that protects your SharePoint web applications.

The Modify Domain: screen appears with the General tab selected.
  - c. If the user directories to which you defined the attribute mappings do *not* appear in the list, go to Step 2d. Otherwise, click Cancel and go to Step 3.
  - d. Click Add/Remove.
  - e. Click the directory connection that you want from the Available Members list, and then click the right arrow.
  - f. Repeat Step 2f to add other directories.
  - g. Click OK.
  - h. Click Submit.
3. Repeat Steps 2a through 2h for any other policy domains on which you want to add directory connections.
4. Add directory connections your application policy (EPM) with the following steps:
  - a. Click Policies, Application, Applications.
  - b. Click the edit icon of the application that protects your SharePoint web applications.

The Modify Application: screen appears with the General tab selected.
  - c. If the user directories to which you defined the attribute mappings do *not* appear in the list, go to Step 4d. Otherwise, click Cancel and go to Step 5.
  - d. Click Add/Remove.
  - e. Click the directory connection that you want from the Available Members list, and then click the right arrow.
  - f. Repeat Step 4f to add other directories.

- g. Click OK.
  - h. Click Submit.
5. Repeat Steps 4a through 4h for any other application policies (EPM) on which you want to add directory connections.

The directory connections are added. Have your agent owner continue with the next step of running the SharePoint connection wizard.

## Run the SharePoint Connection Wizard

As an agent owner who is responsible for running the server hosting the Agent for SharePoint, run the SharePoint connection wizard to finish configuring multiple user directories.

### Follow these steps:

1. Log in to the server that runs your Agent for SharePoint.
2. Navigate to the following directory:

*Agent - for - SharePoint \_home/sharepoint\_connection\_wizard*

3. Do the appropriate step for your operating environment:
  - Windows: Right-click the executable and then select Run as administrator.
  - Solaris: sh ./ca-spconnect-12.0-sp3-sol.bin
  - Linux: sh ./ca-spconnect-12.0-sp3-rhel30.bin

The SharePoint Connection wizard starts.

4. Click Next.

The Login Details screen appears.

5. Enter the following login for the Policy Server.

#### **Policy Server Name**

Specifies the Policy Server name or IP address.

#### **Username**

Specifies the Policy Server administrator username.

#### **Password**

Specifies the Policy Server administrator password.

#### **Agent Name**

Specifies the Agent-4x. The connection with the Policy Server is established using the details given in the Agent Name.



### Shared Secret Key

Specifies the shared secret key that is associated with the Agent.

6. Click Next

The Select Action screen appears.

7. Select Edit a SharePoint Connection option.

8. Click Next.

The SharePoint Connection Properties screen appears.

9. Click through the wizard until you reach the Define Additional Claims screen.

10. Verify that Name Value Pair appears in the Attribute drop-down list.

11. Verify that User Attribute appears in the Claim Type drop-down list.

12. Click the Claim Name field. Type the name of the user attribute that is defined in one of your directory connections in the Administrative UI. For example, if your policy administrator defined UID as a user attribute in the Administrative UI, then type UID as the Claim Name.

13. Type the *alias* name of the attribute from your directory that your policy administrator defined in the Administrative UI. For example, if the alias name for the user attribute is userid then type userid as the directory attribute.

14. Click Add.

15. Repeat Steps 10 through 14 to add the attributes for your other directories.

16. Click through the wizard until the Commit Details screen appears.

17. Click Install.

The Save Complete screen appears.

18. Click Done.

The SharePoint connection wizard closes.

## Remove the Web Applications from the Trusted Identity Provider

A trusted identity provider cannot be removed from SharePoint while any web applications are using it. Before you remove the trusted identity provider itself, remove the association between the CA SiteMinder® trusted identity provider and any of your web agents using it.

**Follow these steps:**

1. Log in to your SharePoint central administration server.
2. Click Start, Microsoft SharePoint 2010 Products, SharePoint 2010 Central Administration.  
The Central Administration home page opens.
3. Under Application Management, click Manage web applications.  
The web application management page opens.
4. Click the line corresponding to the name of a web application using the CA SiteMinder® trusted identity provider.  
The web application is selected.
5. On the ribbon, click Authentication Providers.  
The Authentication Providers dialog appears.
6. In the Authentication Providers dialog, click the link that corresponds to the zone of your web application. For example, if the web application using the CA SiteMinder® trusted identity provider is in the Intranet zone, click the Intranet link.  
The Edit Authentication page appears.
7. Under Claims Authentication types, clear all Trusted Identity provider check boxes.
8. Click Save.  
The CA SiteMinder® trusted identity provider is removed from the web application in the zone.
9. Repeat Steps 3 through 8 for all web applications and the zones using the CA SiteMinder® trusted identity provider.  
The trusted identity provider is removed from all web applications and their respective zones.

## Remove the Trusted Identity Provider

You can perform the following procedure to remove the trusted identity provider for SharePoint using Windows PowerShell.

**Follow these steps:**

1. Log in to your SharePoint central administration server.
2. Select Start, All Programs, Microsoft SharePoint 2010 Products, SharePoint 2010 Management Shell.

The Microsoft PowerShell command prompt appears.

3. Enter the following command:

```
Remove-SPTrustedIdentityTokenIssuer -Identity
```

**-Identity**

Specifies the name of the identity provider to remove.

**Example:** Remove-SPTrustedIdentityTokenIssuer TestSTS

The trusted identity provider for SharePoint is removed.

## Copy the Powershell Script to the SharePoint Central Administration Server

Extra configuration steps are required if you changed the value of an existing Claim Name when you configured multiple user directories. The SharePoint connection wizard creates a PowerShell script that contains the new Claim Name. Copy this PowerShell script from your Agent for SharePoint host to your SharePoint central administration server.

**Follow these steps:**

1. Navigate to the following directory on the server running your Agent for SharePoint:  
  
*Agent-for-SharePoint\_home\sharepoint\_connection\_wizard\*
2. Locate the PowerShell script that the SharePoint connection wizard created. The script uses the connection name that you chose while running the wizard as the file name. For example, if your connection name was *my\_connection*, the name of the script is *my\_connection.ps1*.
3. Copy the PowerShell script to a directory on your SharePoint central administration server.

## Determine PowerShell Script Modifications

To create a trusted identity provider on your SharePoint central administration server, edit the PowerShell script to include the following information about your SharePoint environment:

- The full path to the root certificate (typically from a third-party Certificate Authority) that signed your certificate.
- Create a trusted root authority in SharePoint for the certificate authority which signed your certificate.
- The full path to your signing certificate.
- Friendly names for each of the claim mappings.
- The SharePoint realm name (to identify the trusted identity provider).

**Note:** This value appears in SharePoint Central Administration under the list of available trusted identity providers.

- A friendly description for the trusted identity provider.

The specific modifications to the PowerShell script vary according to the type of certificates you want to use with your CA SiteMinder® trusted identity provider.

Find the proper procedure for your situation in the following table:

If your certificates fit this situation:	Then use this procedure to modify your script:
You are using a certificate that is signed by an external certificate authority, and the certificate authority is not trusted by your SharePoint server.	<a href="#">Modify the script for an un-trusted external certificate authority</a> (see page 253).
You are using a self-signed certificate and the certificate authority is <i>not</i> trusted by your SharePoint server.	<a href="#">Modify the script for an un-trusted self-signed certificate</a> (see page 255).
You are using a certificate, and the certificate authority is trusted by your SharePoint server. Verify with your SharePoint administrator to confirm that the proper certificate authority is trusted.	<a href="#">Modify the script for a trusted certificate authority</a> (see page 257).

## Modify the Script for an Un-Trusted External Certificate Authority

If your signing certificate is signed by an external certificate authority, modify the PowerShell script to do the following tasks:

- Import the certificate authority certificate (root certificate) into SharePoint.
- Create a SharePoint trusted root authority that is based on the certificate authority certificate.
- Import the signing certificate.

### Follow these steps:

1. Open the PowerShell script with any text editor.
2. Locate the following text:  
`"<full path to Root certificate file>"`
3. Replace the previous text with the full path to your root certificate. For example, if the full path to your certificate is `C:\certificates\sharepoint\certificate_authority_certificate.cer`, the updated line matches the following example:  
`"C:\certificates\sharepoint\certificate_authority_certificate.cer"`
4. Locate the first occurrence of the following text:  
`<Trusted root authority name>`
5. Replace the previous text with a friendly name for the new trusted root authority in SharePoint. For example, if the name you want is `SPCAAuth`, the updated line matches the following example:  
`"SPCAAuth"`
6. Locate the following text:  
`"<full path to Signing certificate file>"`
7. Replace the previous text with the full path to your Signing certificate. For example, if the full path to your certificate is `C:\certificates\sharepoint\signing_certificate.cer`, the updated line matches the following example:  
`"C:\certificates\sharepoint\signing_certificate.cer"`
8. Locate the second occurrence of the following text:  
`<Trusted root authority name>`
9. Replace the previous text with a friendly name for the new trusted root authority in SharePoint. For example, if the name you want is `SPSigningAuth`, the updated line matches the following example:  
`"SPSigningAuth"`
10. Locate the following text:

"<Name of the trusted identity provider>"

11. Replace the previous text with the name of your SharePoint realm (the realm name follows \$realm = in the PowerShell script). For example, if the name of your SharePoint realm is \$realm="urn:moss2010-wsfed1-casm", the updated line could match the following example:

"moss2010-wsfed1-casm"

12. Locate the following text:

"<Description for the Trusted Identity Provider>"

13. Replace the previous text with a description for your trusted identity provider. For example, if you want to describe the trusted identity provider as "SiteMinder Provider," the updated line could match the following example:

"SiteMinder Provider"

14. If your certificate chain contains *more than one* certificate authority certificate, [add the other certificate authority certificates to the script](#) (see page 258). If your script contains *one* certificate authority certificate, go to the next step.

15. Save your changes and close your text editor.

The PowerShell script is modified.

16. [Run the PowerShell script](#) (see page 260).

## Modify the Script for an Un-Trusted Self-Signed Certificate

If you are using a self-signed certificate that is issued by a certificate authority which is not explicitly trusted by your SharePoint server, modify the PowerShell script to do the following tasks:

- Import the certificate authority certificate (root certificate) into SharePoint.
- Create a SharePoint trusted root authority that is based on the certificate authority certificate.
- Import the signing certificate.

### Follow these steps:

1. Open the PowerShell script with any text editor.
2. Locate the following text:  
`"<full path to Root certificate file>"`
3. Replace the previous text with the full path to your root certificate. For example, if the full path to your certificate is `C:\certificates\sharepoint\certificate_authority_certificate.cer`, the updated line matches the following example:  
`"C:\certificates\sharepoint\certificate_authority_certificate.cer"`
4. Locate the first occurrence of the following text:  
`<Trusted root authority name>`
5. Replace the previous text with a friendly name for the new trusted root authority in SharePoint. For example, if the name you want is `SPCAAuth`, the updated line matches the following example:  
`"SPCAAuth"`
6. Locate the following text:  
`"<full path to Signing certificate file>"`
7. Replace the previous text with the full path to your Signing certificate. For example, if the full path to your certificate is `C:\certificates\sharepoint\signing_certificate.cer`, the updated line matches the following example:  
`"C:\certificates\sharepoint\signing_certificate.cer"`
8. Locate the second occurrence of the following text:  
`<Trusted root authority name>`
9. Replace the previous text with a friendly name for the new trusted root authority in SharePoint. For example, if the name you want is `SPSigningAuth`, the updated line matches the following example:  
`"SPSigningAuth"`

10. Locate the following text:

"<Name of the trusted identity provider>"

11. Replace the previous text with the name of your SharePoint realm (the realm name follows \$realm = in the PowerShell script). For example, if the name of your SharePoint realm is \$realm="urn:moss2010-wsfed1-casm", the updated line could match the following example:

"moss2010-wsfed1-casm"

12. Locate the following text:

"<Description for the Trusted Identity Provider>"

13. Replace the previous text with a description for your trusted identity provider. For example, if you want to describe the trusted identity provider as "SiteMinder Provider," the updated line could match the following example:

"SiteMinder Provider"

**Note:** The LDAP directory and Active Directory charts contain additional examples of possible names.

14. If your certificate chain contains *more than one* certificate authority certificate, [add the other certificate authority certificates to the script](#) (see page 258). If your script contains *one* certificate authority certificate, go to the next step.

15. Save your changes and close your text editor.

The PowerShell script is modified.

16. [Run the PowerShell script](#) (see page 260).



## Modify the Script for a Trusted Certificate Authority

If you are using a certificate signed by a certificate authority that is trusted by the SharePoint server, modify the PowerShell script to do the following tasks:

- Skip the step to import the certificate authority certificate.
- Skip the step to create a new SharePoint trusted root authority.
- Import only the signing certificate.

### Follow these steps:

1. Open the PowerShell script with any text editor.
2. Comment the first two lines in the PowerShell script, as shown in the following example:

```
#$rootcert = New-Object
#System.Security.Cryptography.X509Certificates.X509Certificate2("<full
#path to Root certificate file>")
#New-SPTtrustedRootAuthority -Name "<Trusted root authority name>"
-Certificate $rootcert
```

3. Locate the following text:

```
"<full path to Signing certificate file>"
```

4. Replace the previous text with the full path to your Signing certificate. For example, if the full path to your certificate is C:\certificates\sharepoint\signing\_certificate.cer, the updated line matches the following example:

```
"C:\certificates\sharepoint\signing_certificate.cer"
```

5. Locate the second occurrence of the following text:

```
<Trusted root authority name>
```

6. Replace the previous text with a friendly name for the new trusted root authority in SharePoint. For example, if the name you want is SPSigningAuth, the updated line matches the following example:

```
"SPSigningAuth"
```

7. Locate the following text:

```
"<Name of the trusted identity provider>"
```

8. Replace the previous text with the name of your SharePoint realm (the realm name follows \$realm = in the PowerShell script). For example, if the name of your SharePoint realm is \$realm="urn:moss2010-wsfed1-casm", the updated line could match the following example:

```
"moss2010-wsfed1-casm"
```

9. Locate the following text:

```
"<Description for the Trusted Identity Provider>"
```

10. Replace the previous text with a description for your trusted identity provider. For example, if you want to describe the trusted identity provider as "SiteMinder Provider," the updated line could match the following example:

"SiteMinder Provider"

11. Save your changes and close your text editor.

The PowerShell script is modified.

12. [Run the PowerShell script](#) (see page 260).

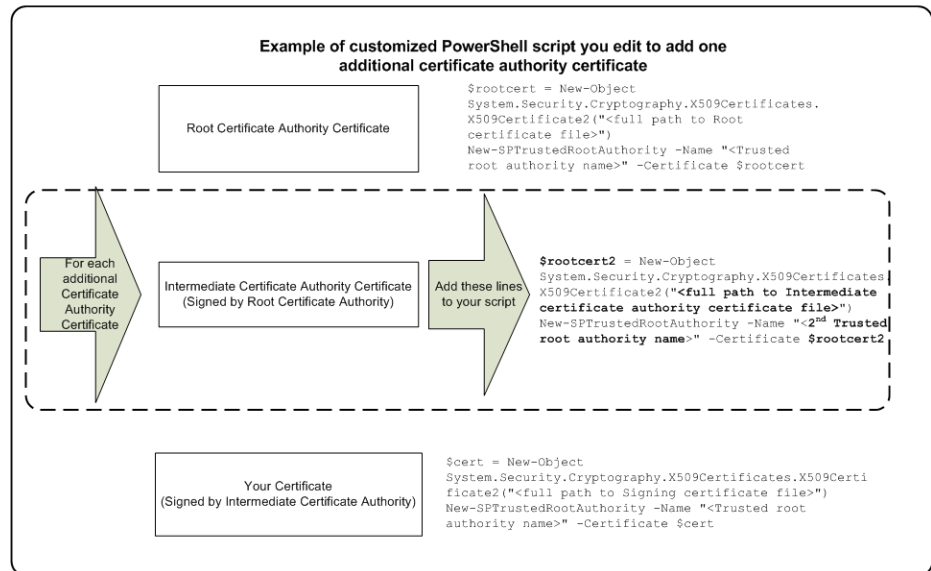
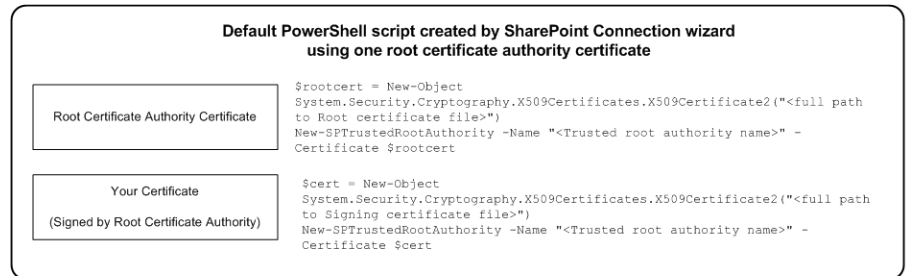
## Add Additional Certificate Authority Certificates

The PowerShell script created by the SharePoint connection wizard accommodates the following certificates:

- A certificate authority certificate (also named a root certificate)
- One SSL certificate.

The trusted identity provider requires that all certificates in the certificate chain are included. If an intermediate certificate authority signed your certificate instead, modify the PowerShell script to include both certificate authority certificates.

The following graphic describes the differences between the default PowerShell script, and a PowerShell script that accommodates multiple certificate-authority certificates:



**Follow these steps:**

1. Copy the following section from your PowerShell script:

```
$rootcert = New-Object
System.Security.Cryptography.X509Certificates.X509Certificate2("<full path to Root certificate file>")
New-SPTrustedRootAuthority -Name "<Trusted root authority name>" -Certificate $rootcert
```

2. Copy the following section from your PowerShell script:
3. Add a new line after the section you copied, and then paste the copied into the new line.
4. Edit the pasted section using the changes shown in the following table as a guide:

Change this value:	To this value:
\$rootcert	\$rootcert2

<full path to Root certificate file>	<full path to additional certificate authority certificate file>
<Trusted root authority name>	Name of the additional trusted root authority

5. To add additional certificate authority certificates, repeat Steps 1 through 4.
6. Save your changes and close your text editor.  
The additional certificate authority certificates are added.
7. [Run the PowerShell script](#) (see page 260).

## Run the PowerShell Script

Run the PowerShell script that contains the updated Claim Name value on your SharePoint central administration server.

### Follow these steps:

1. Click Start, All Programs, Microsoft SharePoint 2010 Products, SharePoint 2010 Management Shell.
2. Navigate to the directory containing the modified PowerShell script.
3. Run the script with the following command:

```
.\your_connection_name.ps1
```

For example, if you named your connection *my\_sharepoint* when you ran the connection wizard, the command would be `.\my_sharepoint.ps1`.

The trusted identity provider is modified.

## Verify Trusted Identity Provider Registration

After running the PowerShell script to create your trusted identity provider, verify that it is registered in your SharePoint central administration server.

**Follow these steps:**

1. From your SharePoint central administration server, click Start, All Programs, Microsoft SharePoint 2010 Products, SharePoint 2010 Management Shell.

The Microsoft PowerShell command prompt appears.

2. Enter the following command:

```
Get-SPTrustedIdentityTokenIssuer
```

A list of the trusted identity providers that are configured on the SharePoint central administration server appears.

## Disable Client Loopback

If you do *not* need to add attributes using the SharePoint people picker before they exist in your user directories, disable the client loopback feature. Leaving client loopback enabled when the directory attributes exist returns duplicates in the SharePoint people picker.

**Follow these steps:**

1. Log in to your SharePoint central administration server.
2. Click Start, All Programs, Microsoft SharePoint 2010 Products, SharePoint 2010 Management Shell.

The management shell command-line window opens.

3. Navigate to the following directory:

```
C:\Program Files\CA\SharePointClaimsProvider\scripts
```

4. Enter the following command:

```
.\Set-SMClaimProviderConfiguration.ps1 -DisableLoopBackSearch
```

Loopback search is disabled.

## Add Users to Your Web Applications

Add your users to SharePoint and assign permission levels depending on their roles. Permission levels allow users to perform a set of related tasks.

**Follow these steps:**

1. From your SharePoint central administration server, click, Start SharePoint 2010 Central Administration from Start, Programs, Microsoft SharePoint 2010 Products.  
The Central Administration home page appears.
2. Click Manage web applications, in the Application Management section.  
The Web Applications Management page appears with a list of available web applications.
3. Click the web application name for which you want to add users.  
The buttons on the ribbon become available.
4. Click User Policy on the ribbon.  
The Policy for Web Application dialog appears.
5. Click Add Users.  
The Select Zone dialog appears.
6. Verify that the Zone you want appears in the drop-down list, and then Click Next.  
The Add Users dialog appears.
7. Click the Browse button, in the Choose Users section, below the Users text box.  
The Select People and Groups – Webpage Dialog appears.
8. Browse and select the user group to search for the user.  
The right pane displays the search results with the list of users.
9. Select the user and click Add.  
SharePoint adds the selected user.
10. (Optional) Repeat steps 8 and 9 to select additional users.
11. Click OK.  
The Add Users dialog appears and displays the selected users.
12. Select the required permissions for the users, in the Choose Permissions section.
13. Click Finish.  
SharePoint adds the selected users and assigns the selected permissions to the users.

## How to Configure SLO for SharePoint 2013

Users visiting multiple web sites that the CA SiteMinder Agent for SharePoint protects have a Fedauth cookie from each site in their browsers. The SLO feature removes these Fedauth cookies when the users log out.

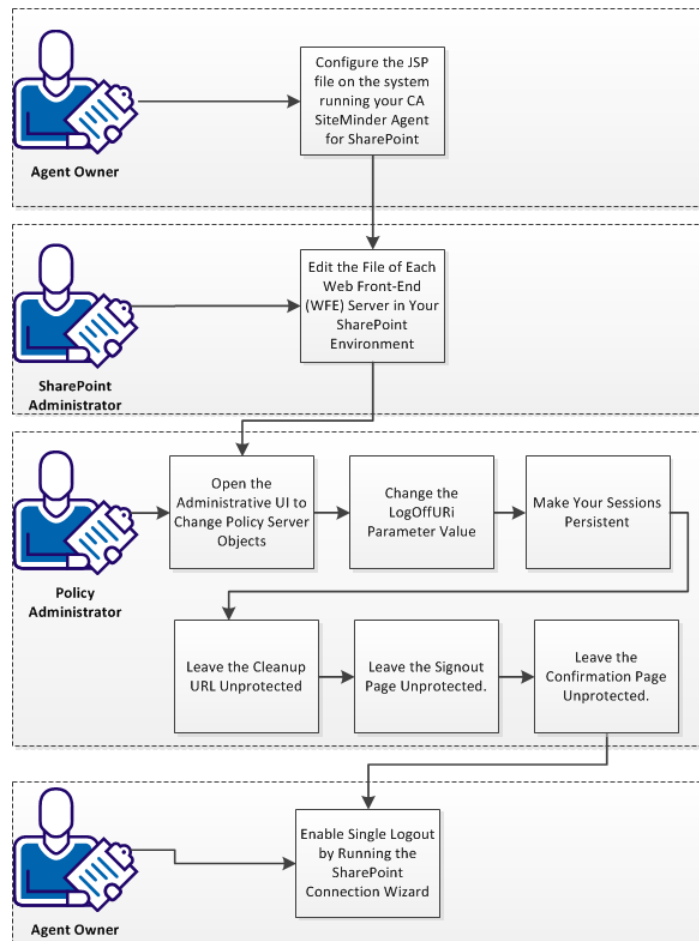
Configuring the single log-out feature of CA SiteMinder Agent for SharePoint for SharePoint 2010 to support SharePoint 2013 involves several separate procedures.

This scenario assumes that the following prerequisites are met:

- At least one Policy Server is installed, configured, and running.
- An Administrative UI installed, configured, and running.
- The CA SiteMinder Agent for SharePoint is installed and configured.
- A session store is installed and configured.

The following graphic describes how to configure SLO for SharePoint 2013:

### How to Configure Single Logout for SharePoint 2013





**Follow these steps:**

1. [Configure the JSP file on the system running your CA SiteMinder Agent for SharePoint](#) (see page 266).
2. [Edit the file of each Web Front-End \(WFE\) Server in Your SharePoint environment](#) (see page 267).
3. [Open the Administrative UI to change the Policy Server objects](#) (see page 42).
4. [Change the LogOffURI parameter value](#) (see page 270).
5. [Make your sessions persistent](#) (see page 271).
6. [Leave the cleanup URL unprotected](#) (see page 272).
7. [Leave the sign-out service URL unprotected](#) (see page 273).
8. [Leave the confirmation page unprotected](#) (see page 274).
9. [Enable single log-out by running the SharePoint connection wizard](#) (see page 275).

## Configure the JSP file on the system that is running your CA SiteMinder Agent for SharePoint

As an agent owner who is responsible for running the server hosting the CA SiteMinder Agent for SharePoint, configure the following file:

`spsignout.jsp`

### Follow these steps:

1. Log in to the system hosting your CA SiteMinder Agent for SharePoint.
2. Navigate to the following directory:

*Agent-for-SharePoint\_Home\Tomcat\webapps\affwebservices*

### **Agent-for-SharePoint\_Home**

Indicates the directory where the CA SiteMinder Agent for SharePoint is installed.

**Default:** (Windows) [32-bit] C:\Program Files\CA\Agent-for-SharePoint

**Default:** (Windows) [64-bit] C:\CA\Agent-for-SharePoint

**Default:** (UNIX/Linux) /opt/CA/Agent-for-SharePoint

3. Open the following file with a text editor:

`spsignout.jsp`

4. Locate the following line:

```
<%response.sendRedirect("http://SharePointServerHostName>Port>/affwebservices/public/wsfedsignout?wa=wsignout1.0");%>
```

5. Replace the URL shown in the previous line URL of the protected SharePoint application. If the URL of your protected SharePoint application is example.com, then edit the line to match the following example:

```
<%response.sendRedirect("http://example.com/affwebservices/public/wsfedsignout?wa=wsignout1.0");%>
```

6. Save the file and close the text editor.

The JSP file is configured.

## Edit the File of Each Web Front-End (WFE) Server in Your SharePoint Environment

As a SharePoint administrator who is responsible for running the SharePoint environment, edit the Welcome.ascx file on your WFE servers to accommodate SharePoint 2013. Editing the file replaces the SharePoint signout URL with the URL of the CA SiteMinder® signout page.

### Follow these steps:

1. Log in to your WFE server.
2. Make a backup copy of the following file:  
  
    %ProgramFiles%\Common Files\Microsoft Shared\Web Server  
    Extensions\15\TEMPLATE\CONTROLTEMPLATES\Welcome.ascx
3. Open the original version of the Welcome.ascx file with a text editor:

**Important!** Do not use Notepad, Wordpad (or any other text editor with line-length limitations) to edit the .config (XML) files. A text editor that is designed for writing programming source code typically does not have such line-length limitations. For more information, see the documentation or online help for your respective editor.

4. Locate the following line:

```
<SharePoint:MenuItemTemplate runat="server" id="ID_Logout"
```

5. Change ID\_Logout to ID\_Logout2, as shown in the following example:

```
<SharePoint:MenuItemTemplate runat="server" id="ID_Logout2"
```

6. Locate the following line:

```
UseShortID="true"
```

7. Add a line following the previous line (shown in Step 6).

8. Add the following settings to the new line:

```
ClientOnClickNavigateurl="http://example.com/affwebservice/spsignout.jsp"
```

9. Replace the example.com text in the previous line with the domain of your SharePoint web application. For example, if the domain of your SharePoint web application is support.example.com, then the text in Step 8 would resemble the following example:

```
ClientOnClickNavigateurl="http://support.example.com/affwebservice/spsignout.jsp"
```

**Note:** If the realm or component protecting the directory of the spsignout.jsp page is set to /\*, create a realm or component to leave /affwebservice/spsignout.jsp *unprotected*.

10. Verify that the edited Welcome.ascx file resembles the following example:

```
<SharePoint:MenuItemTemplate runat="server" id="ID_Logout2"
 Text="<%=Resources:wss,personalactions_logout%>"
 Description="<%=Resource3s:wss,personalactions_logutdescription%>"
 MenuGroupID="100"
 UseShortID="true"
 ClientOnClickNavigateUrl="http://example.com/affwebservice/spsignout.jsp"
```

11. Save the file and close the text editor.
12. Restart the Internet Information Services (IIS) on your WFE server.
13. Repeat Steps 1 through 12 on all of your WFE servers.

The files of each WFE servers are edited. Have your policy administrator perform the next steps by opening the Administrative UI.

## Open the Administrative UI to Change Policy Server Objects

Change the objects on your Policy Server by opening the Administrative UI.

**Follow these steps:**

1. Open the following URL in a browser.

`https://host_name:8443/iam/siteminder/adminui`

**host\_name**

Specifies the fully qualified Administrative UI host system name.

2. Enter your CA SiteMinder® superuser name in the User Name field.
3. Enter the CA SiteMinder® superuser account password in the Password field.  
**Note:** If your superuser account password contains dollar-sign (\$) characters, replace each instance of the dollar-sign character with \$DOLLAR\$. For example, if the CA SiteMinder® superuser account password is \$password, enter \$DOLLAR\$password in the Password field.
4. Verify that the proper server name or IP address appears in the Server drop-down list.
5. Select Log In.

## Change the LogOffUri parameter value

The SLO feature requires the following value in the LogOffUri agent configuration parameter:

`/_layouts/15/SignOut.aspx`

### Follow these steps:

1. From the Administrative UI, click Infrastructure, Agent, Agent Configuration Objects.
2. Click the edit icon in the line Agent Configuration Object that protects your SharePoint 2010 resources.

**Note:** This agent configuration object must be based on the SharePoint 2010 default settings template.

3. Locate the following parameter:

#### LogOffUri

Enables full log-out and displays a confirmation page after users are successfully logged off. Configure this page so that it *cannot* be stored in a browser cache. If a cached page is used, session hijacking by unauthorized users is possible.

When the SharePoint users click the Sign out link, the following URI is used:

- `/_layouts/SignOut.aspx`

When the SharePoint users click the Sign in as another user link, the following URI is used:

- `/_layouts/accessdenied.aspx?loginasanotheruser=true`

If you have multiple SharePoint web sites below a top-level SharePoint website, add the URIs of the lower-level sites to the LogOffUri parameter.

**Note:** When the CookiePath parameter is set, the value of the LogOffUri parameter must point to the same cookie path. For example, if the value of your CookiePath parameter is set to example.com, then your LogOffUri must point to example.com/logoff.html

**Default:** `/_layouts/SignOut.aspx,`  
`/_layouts/accessdenied.aspx?loginasanotheruser=true`

**Limits:** Multiple URI values permitted. Do *not* use a fully qualified URL. Use a relative URI.

**Example:** (for a parent site of www.example.com with two lower-level sites named finance and hr respectively) `/finance/_layouts/SignOut.aspx,`  
`finance/_layouts/accessdenied.aspx?loginasanotheruser=true`  
`/hr/_layouts/SignOut.aspx,`  
`/hr/_layouts/accessdenied.aspx?loginasanotheruser=true`

4. Click the edit icon next to the previous parameter, and then add the following value:

/\_layouts/15/SignOut.aspx

5. Click OK.
6. Click Submit.

The value of the LogOffURi parameter has changed.

## Make Your Sessions Persistent

As a policy administrator who manages the policies on the Policy Server, the next step in configuring single logout is making your sessions persistent.

### Follow these steps:

1. Pick the appropriate procedure for your type of policy from the following list:
  - If you use policy domains, go to Step 2.
  - If you use application policies (EPM), go to Step 4.
2. Make the sessions in your policy domain persistent with the following steps:
  - a. Click Policies, Domain, Realms.
  - b. Click the edit icon of the realm that protects your SharePoint web applications.
  - c. Click the Persistent option button (in the Session section).
  - d. Click Submit.
3. Repeat Steps 2a through 2d for any other policy domains on which you want to configure single logout.
4. Make the sessions in your application policy (EPM) persistent with the following steps:
  - a. Click Policies, Application, Applications.
  - b. Click the edit icon of the application that protects your SharePoint web applications.
  - c. Verify that the General tab is selected, and then click Advanced Settings...
  - d. Click the Persistent option button (in the Session section).
  - e. Click OK.
  - f. Click Submit.
5. Repeat Steps 4a through 4f for any other policy applications (EPM) on which you want to configure single logout.

The sessions are persistent. Have your policy administrator continue with the next step of leaving the cleanup URL unprotected.

## Leave the Clean Up URL Unprotected

As a policy administrator who manages the policies on the Policy Server, leave the cleanup URL unprotected.

Leaving the cleanup URL unprotected prevents a security challenge from appearing during the single logout process.

### Follow these steps:

1. Pick the appropriate procedure for your type of policy from the following list:
  - If you use policy domains, go to Step 2.
  - If you use application policies (EPM), go to Step 4.
2. Leave the cleanup URL unprotected in your policy domain with the following steps:
  - a. Click Policies, Domain, Realms.
  - b. Click Create Realm
  - c. Verify that the domain with your SharePoint web applications is selected and then click Next.
  - d. Enter a name and optional description for the new realm.
  - e. Click the Lookup Agent/Agent Group button, and then add the agent object that protects your SharePoint web applications.
  - f. Click the resource filter field, and then add the following text:  
`_trust?wa=wsignoutcleanup1.0`
  - g. Click the Unprotected option button.
  - h. Click Finish.
3. Repeat Steps 2a through 2h for each policy domain protecting your SharePoint web applications.
4. Leave the cleanup URL unprotected in your application policy (EPM) with the following steps:
  - a. Click Policies, Application, Applications.
  - b. Click the edit icon of the application that protects your SharePoint web applications.
  - c. Verify that the General tab is selected, and then click Create Component.
  - d. Enter a name for the component.
  - e. Click the Lookup Agent/Agent Group button, and then add the agent object that protects your SharePoint web applications.
  - f. Click the resource filter field, and then add the following text:  
`_trust?wa=wsignoutcleanup1.0`



- g. Click the Unprotected option button.
  - h. Click OK.
  - i. Click Submit.
5. Repeat Steps 4a through 4i for each application policy (EPM) protecting your SharePoint web applications.

The cleanup URLs are unprotected. Have your policy administrator continue with the next step of leaving the sign-out service URL unprotected.

## Leave the Sign-out Service URL Unprotected

As a policy administrator who manages the policies on the Policy Server, leave the sign-out service URL unprotected. Leaving the sign-out service URL unprotected prevents a security challenge from appearing during the single logout process.

### Follow these steps:

1. Click Infrastructure, Agent, Agent Groups.
2. Select the FederationWebServicesAgentGroup name in the agent groups list and click Modify.
3. Click Add/Remove.
4. Select the agent you want to add to the agent group from the list of Available Members, and click the right-facing arrows.
5. Click OK, Submit.
6. Click Policies, Domain.
7. Select FederationWebServicesDomain, and click Modify.
8. Click Realms.
9. Select the public realm and click Modify.
10. Ensure that the Unprotected option is selected in the Default Resource Protection field. If not, select the Unprotected option.
11. Click the Resource Filter field and add the following text:  
`/affwebservices/spsignout.jsp`
12. Click Finish.
13. Repeat Steps 1 through 12 for each policy domain or application policy (EPM) protecting your SharePoint web applications.

The sign-out service URLs are unprotected. Continue with the next step of leaving the confirmation URL unprotected.

## Leave the Confirmation Page Unprotected

As a policy administrator who manages the policies on the Policy Server, the next step in configuring single logout is leaving the confirmation page unprotected.

Leaving the confirmation page unprotected prevents a security challenge from appearing during the single logout process.

### Follow these steps:

1. Pick the appropriate procedure for your type of policy from the following list:
  - If you use policy domains, go to Step 2.
  - If you use application policies (EPM), go to Step 4.
2. Leave the confirmation page unprotected in your policy domain with the following steps:
  - a. Click Policies, Domain, Realms.
  - b. Click Create Realm
  - c. Verify that the domain with your SharePoint web applications is selected and then click Next.
  - d. Enter a name and optional description for the new realm.
  - e. Click the Lookup Agent/Agent Group button, and then add the agent object that protects your SharePoint web applications.
  - f. Click the resource filter field, and then add the following text:  
`affwebservices/spsignoutconfirmurl.jsp`
  - g. Click the Unprotected option button.
  - h. Click Finish.
3. Repeat Steps 2a through 2h for each policy domain protecting your SharePoint web applications.
4. Leave the confirmation page unprotected in your application policy (EPM) with the following steps:
  - a. Click Policies, Application, Applications.
  - b. Click the edit icon of the application that protects your SharePoint web applications.
  - c. Verify that the General tab is selected, and then click Create Component.
  - d. Enter a name for the component.
  - e. Click the Lookup Agent/Agent Group button, and then add the agent object that protects your SharePoint web applications.
  - f. Click the resource filter field, and then add the following text:  
`affwebservices/spsignoutconfirmurl.jsp`

- g. Click the Unprotected option button.
  - h. Click OK.
  - i. Click Submit.
5. Repeat Steps 4a through 4i for each application policy (EPM) protecting your SharePoint web applications.

The confirmation pages are unprotected. Have your SharePoint administrator continue with the next step of enabling single logout by running the SharePoint connection wizard.

## Enable Single Logout by Running the SharePoint Connection Wizard

As an agent owner who is responsible for running the system that hosts the CA SiteMinder Agent for SharePoint, run the SharePoint connection wizard to finish enabling single logout.

### Follow these steps:

1. Edit the existing connection using the Connection Wizard with the following steps:
  - a. Log in to the server that runs your Agent for SharePoint.
  - b. Navigate to the following directory:

*Agent-for-SharePoint\_home/sharepoint\_connection\_wizard*

### **Agent-for-SharePoint\_Home**

Indicates the directory where the CA SiteMinder Agent for SharePoint is installed.

**Default:** (Windows) [32-bit] C:\Program Files\CA\Agent-for-SharePoint

**Default:** (Windows) [64-bit] C:\CA\Agent-for-SharePoint

**Default:** (UNIX/Linux) /opt/CA/Agent-for-SharePoint

- a. Do the appropriate step for your operating environment:
  - Windows: Right-click the executable and then pick Run as administrator.
  - Solaris: sh ./ca-spconnect-version\_number-sol.bin
  - Linux: sh ./ca-spconnect-version\_number-rhel30.bin

The SharePoint Connection wizard starts.

- b. Click Next.

The Login Details screen appears.

- c. Enter the following login for the Policy Server.

**Policy Server Name**

Specifies the Policy Server name or IP address.

**Username**

Specifies the Policy Server administrator username.

**Password**

Specifies the Policy Server administrator password.

**Agent Name**

Specifies the Agent-4x. The connection with the Policy Server is established using the details given in the Agent Name.

**Shared Secret Key**

Specifies the shared secret key that is associated with the Agent.

- d. Click Next

The Select Action screen appears.

- e. Select Edit a SharePoint Connection option.

- f. Click Next.

The SharePoint Connection Properties screen appears.

- g. Click through the wizard until you reach the Single Logout Configuration screen.
- h. Select the Enabled SignOut check box.
- i. Click the CleanUp URL field and then type the cleanup URLs from all of your protected web applications.

**Note:** Separate multiple URLs with semi-colons.

Use the following example as a guide:

```
http://SharePoint_web_application_one_page_URL/_trust?wa=wsignoutcleanup1.0;http://SharePoint_web_application_two_page_URL/_trust?wa=wsignoutcleanup1.0
```

- j. Click the Confirm URL field and type a single confirmation page URL. Use the following example as a guide:

```
http://SharePoint_web_application_one_page_URL/affwebservices/spsignoutconfirmurl.jsp
```

**Note:** If you have multiple public web applications.

*SharePoint\_web\_application\_one\_URL* can be the fully qualified domain that is named of any of those applications.

For example, if you have the following public SharePoint URLs:

http://marketing.example.com and http://developers.example.com,

*SharePoint\_web\_application\_one\_page\_URL* can be

http://marketing.example.com or http://developers.example.com.

- k. Click through the wizard until the Commit Details screen appears.
- l. Click Install.

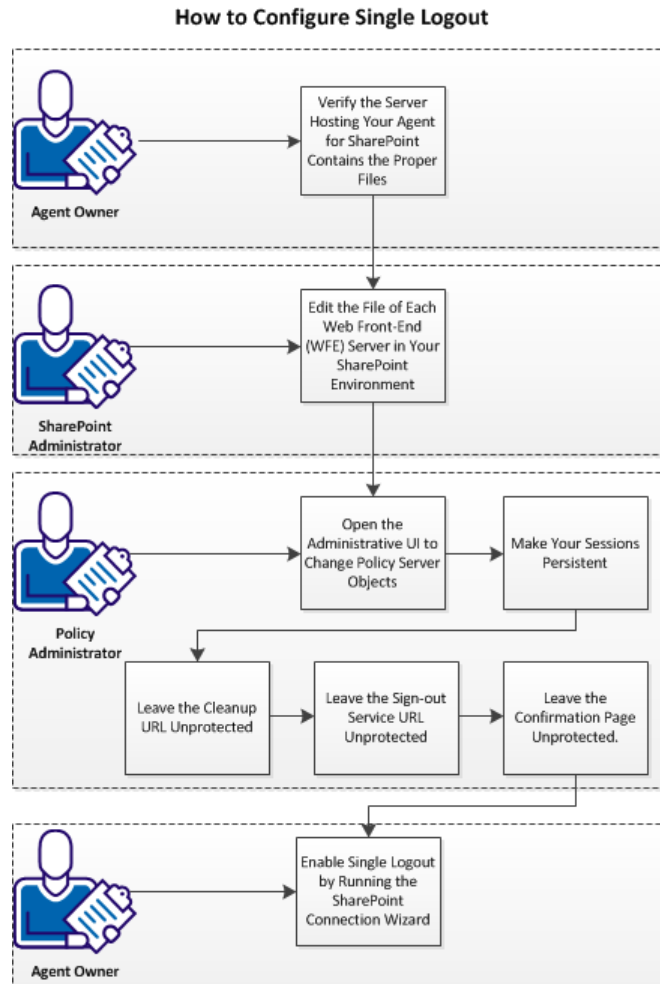
The Save Complete screen appears.

- m. Click Done.

The SharePoint connection wizard closes. Single logout is enabled.

## How to Configure Single Logout on SharePoint 2010

Users who visit multiple websites that the Agent for SharePoint protects have a Fedauth browser cookie for each website. Configuring the single logout verifies that these Fedauth cookies are removed from the browser of the user upon logout.



**Follow these steps:**

1. [Verify that the server hosting your Agent for SharePoint contains the proper files](#) (see page 279).
2. [Edit the file of each web front-end \(WFE\) server in your SharePoint environment](#) (see page 280).
3. [Open the Administrative UI](#) (see page 42), and then perform the following tasks:
  - a. [Make your sessions persistent](#) (see page 271).
  - b. [Leave the cleanup URL unprotected](#) (see page 283).
  - c. Leave the sign-out service URL unprotected.
  - d. [Leave the confirmation page unprotected](#) (see page 274).
4. [Enable single logout by running the SharePoint Connection wizard](#) (see page 286).

## Verify the Server Hosting Your Agent for SharePoint Has the Proper Files

As an agent owner who is responsible for running the server hosting the Agent for SharePoint, verify that the server contains the correct .jsp file. This step is the first step in configuring the single log-out feature.

**Follow these steps:**

1. Log in to the system hosting your Agent for SharePoint.
2. Navigate to the following directory:

*Agent-for-SharePoint\_Home\Tomcat\webapps\affwebservices*

**Agent-for-SharePoint\_Home**

Indicates the directory where the CA SiteMinder Agent for SharePoint is installed.

**Default:** (Windows) [32-bit] C:\Program Files\CA\Agent-for-SharePoint

**Default:** (Windows) [64-bit] C:\CA\Agent-for-SharePoint

**Default:** (UNIX/Linux) /opt/CA/Agent-for-SharePoint

3. Verify that the following files exist:

- signoutconfirmurl.jsp
- spsignoutconfirmurl.jsp

**Note:** If the previous file does not exist, verify that the proper version of the Agent for SharePoint is installed on your server.

The presence of the proper file is verified. Have your SharePoint administrator continue with the next step of editing the files on your web front-end (WFE) servers.

## Edit the File of Each Web Front-End (WFE) Server in Your SharePoint Environment

As a SharePoint administrator who is responsible for running the SharePoint environment, edit the Welcome.ascx file on your WFE servers. Editing the file replaces the SharePoint signout URL with the URL of the <stmdnr> signout page. This step is the next step in configuring the single logout feature.

### Follow these steps:

1. Log in to your WFE server.

2. Make a backup copy of the following file:

```
%ProgramFiles%\Common Files\Microsoft Shared\Web Server
Extensions\14\TEMPLATE\CONTROLTEMPLATES\Welcome.ascx
```

3. Open the original version of the Welcome.ascx file with a text editor:

**Important!** Do not use Notepad, Wordpad (or any other text editor with line-length limitations) to edit the .config (XML) files. A text editor that is designed for writing programming source code typically does not have such line-length limitations. For more information, see the documentation or online help for your respective editor.

4. Locate the following line:

```
<SharePoint:MenuItemTemplate runat="server" id="ID_Logout"
```

5. Change ID\_Logout to ID\_Logout2, as shown in the following example:

```
<SharePoint:MenuItemTemplate runat="server" id="ID_Logout2"
```

6. Locate the following line:

```
UseShortID="true"
```

7. Add a line following the previous line (shown in Step 6).

8. Add the following settings to the new line:

```
ClientonClickNavigateurl="http://example.com/affwebservices/public/wsfedsigno
ut?wa=wsignout1.0"
```

9. Replace the example.com text in the previous line with the domain of your SharePoint web application. For example, if the domain of your SharePoint web application is support.example.com, then the text in Step 8 would resemble the following example:

```
ClientonClickNavigateurl="http://support.example.com/affwebservices/public/ws
fedsignout?wa=wsignout1.0"
```

10. Save the file and close the text editor.

11. Restart the Internet Information Services (IIS) on your WFE server.

12. Repeat Steps 1 through 11 on all of your WFE servers.

The files of each WFE servers are edited. Have your policy administrator perform the next steps by opening the Administrative UI.



## Open the Administrative UI to Change Policy Server Objects

Change the objects on your Policy Server by opening the Administrative UI.

**Follow these steps:**

1. Open the following URL in a browser.

`https://host_name:8443/iam/siteminder/adminui`

**host\_name**

Specifies the fully qualified Administrative UI host system name.

2. Enter your CA SiteMinder® superuser name in the User Name field.
3. Enter the CA SiteMinder® superuser account password in the Password field.  
**Note:** If your superuser account password contains dollar-sign (\$) characters, replace each instance of the dollar-sign character with \$DOLLAR\$. For example, if the CA SiteMinder® superuser account password is \$password, enter \$DOLLAR\$password in the Password field.
4. Verify that the proper server name or IP address appears in the Server drop-down list.
5. Select Log In.

## Make Your Sessions Persistent

As a policy administrator who manages the policies on the Policy Server, the next step in configuring single logout is making your sessions persistent.

### Follow these steps:

1. Pick the appropriate procedure for your type of policy from the following list:
  - If you use policy domains, go to Step 2.
  - If you use application policies (EPM), go to Step 4.
2. Make the sessions in your policy domain persistent with the following steps:
  - a. Click Policies, Domain, Realms.
  - b. Click the edit icon of the realm that protects your SharePoint web applications.
  - c. Click the Persistent option button (in the Session section).
  - d. Click Submit.
3. Repeat Steps 2a through 2d for any other policy domains on which you want to configure single logout.
4. Make the sessions in your application policy (EPM) persistent with the following steps:
  - a. Click Policies, Application, Applications.
  - b. Click the edit icon of the application that protects your SharePoint web applications.
  - c. Verify that the General tab is selected, and then click Advanced Settings...
  - d. Click the Persistent option button (in the Session section).
  - e. Click OK.
  - f. Click Submit.
5. Repeat Steps 4a through 4f for any other policy applications (EPM) on which you want to configure single logout.

The sessions are persistent. Have your policy administrator continue with the next step of leaving the cleanup URL unprotected.

## Leave the Clean Up URL Unprotected

As a policy administrator who manages the policies on the Policy Server, the next step in configuring single logout is leaving the cleanup URL unprotected.

Leaving the cleanup URL unprotected prevents a security challenge from appearing during the single logout process.

### Follow these steps:

1. Pick the appropriate procedure for your type of policy from the following list:
  - If you use policy domains, go to Step 2.
  - If you use application policies (EPM), go to Step 4.
2. Leave the cleanup URL unprotected in your policy domain with the following steps:
  - a. Click Policies, Domain, Realms.
  - b. Click Create Realm
  - c. Verify that the domain with your SharePoint web applications is selected and then click Next.
  - d. Enter a name and optional description for the new realm.
  - e. Click the Lookup Agent/Agent Group button, and then add the agent object that protects your SharePoint web applications.
  - f. Click the resource filter field, and then add the following text:  
`_trust?wa=wsignoutcleanup1.0`
  - g. Click the Unprotected option button.
  - h. Click Finish.
3. Repeat Steps 2a through 2h for each policy domain protecting your SharePoint web applications.
4. Leave the cleanup URL unprotected in your application policy (EPM) with the following steps:
  - a. Click Policies, Application, Applications.
  - b. Click the edit icon of the application that protects your SharePoint web applications.
  - c. Verify that the General tab is selected, and then click Create Component.
  - d. Enter a name for the component.
  - e. Click the Lookup Agent/Agent Group button, and then add the agent object that protects your SharePoint web applications.
  - f. Click the resource filter field, and then add the following text:  
`_trust?wa=wsignoutcleanup1.0`

- g. Click the Unprotected option button.
  - h. Click OK.
  - i. Click Submit.
5. Repeat Steps 4a through 4i for each application policy (EPM) protecting your SharePoint web applications.

The cleanup URLs are unprotected. Have your policy administrator continue with the next step of leaving the sign-out service URL unprotected.

## Leave the Sign-out Service URL Unprotected

As a policy administrator who manages the policies on the Policy Server, the next step in configuring single logout is leaving the sign-out service URL unprotected.

Leaving the sign-out service URL unprotected prevents a security challenge from appearing during the single logout process.

### Follow these steps:

1. Click Infrastructure, Agent, Agent Groups.
2. Select the FederationWebServicesAgentGroup name in the agent groups list and click Modify.
3. Click Add/Remove.
4. Select the agent you want to add to the agent group from the list of Available Members, and click the right-facing arrows.
5. Click OK, Submit.
6. Click Policies, Domain.
7. Select FederationWebServicesDomain, and click Modify.
8. Click Realms.
9. Select the public realm and click Modify.
10. Ensure that the Unprotected option is selected in the Default Resource Protection field. If not, select the Unprotected option.
11. Click the Resource Filter field and add the following text:  
`/affwebservices/public/wsfedsignout?wa=wsignout1.0`
12. Click Finish.
13. Repeat Steps 1 through 12 for each policy domain or application policy (EPM) protecting your SharePoint web applications.

The sign-out service URLs are unprotected. Have your policy administrator continue with the next step of leaving the confirmation URL unprotected.

## Leave the Confirmation Page Unprotected

As a policy administrator who manages the policies on the Policy Server, the next step in configuring single logout is leaving the confirmation page unprotected.

Leaving the confirmation page unprotected prevents a security challenge from appearing during the single logout process.

### Follow these steps:

1. Pick the appropriate procedure for your type of policy from the following list:
  - If you use policy domains, go to Step 2.
  - If you use application policies (EPM), go to Step 4.
2. Leave the confirmation page unprotected in your policy domain with the following steps:
  - a. Click Policies, Domain, Realms.
  - b. Click Create Realm
  - c. Verify that the domain with your SharePoint web applications is selected and then click Next.
  - d. Enter a name and optional description for the new realm.
  - e. Click the Lookup Agent/Agent Group button, and then add the agent object that protects your SharePoint web applications.
  - f. Click the resource filter field, and then add the following text:  
`affwebservices/spsignoutconfirmurl.jsp`
  - g. Click the Unprotected option button.
  - h. Click Finish.
3. Repeat Steps 2a through 2h for each policy domain protecting your SharePoint web applications.
4. Leave the confirmation page unprotected in your application policy (EPM) with the following steps:
  - a. Click Policies, Application, Applications.
  - b. Click the edit icon of the application that protects your SharePoint web applications.
  - c. Verify that the General tab is selected, and then click Create Component.
  - d. Enter a name for the component.
  - e. Click the Lookup Agent/Agent Group button, and then add the agent object that protects your SharePoint web applications.
  - f. Click the resource filter field, and then add the following text:  
`affwebservices/spsignoutconfirmurl.jsp`

- g. Click the Unprotected option button.
  - h. Click OK.
  - i. Click Submit.
5. Repeat Steps 4a through 4i for each application policy (EPM) protecting your SharePoint web applications.

The confirmation pages are unprotected. Have your SharePoint administrator continue with the next step of enabling single logout by running the SharePoint connection wizard.

## Enable Single Logout by Running the SharePoint Connection Wizard

As an agent owner who is responsible for running the server hosting the Agent for SharePoint, run the SharePoint connection wizard to finish enabling single logout.

### Follow these steps:

1. Edit the existing connection using the Connection Wizard with the following steps:
  - a. Log in to the server that runs your Agent for SharePoint.
  - b. Navigate to the following directory:

*Agent-for-SharePoint\_home/sharepoint\_connection\_wizard*

### Agent-for-SharePoint\_Home

Indicates the directory where the CA SiteMinder Agent for SharePoint is installed.

**Default:** (Windows) [32-bit] C:\Program Files\CA\Agent-for-SharePoint

**Default:** (Windows) [64-bit] C:\CA\Agent-for-SharePoint

**Default:** (UNIX/Linux) /opt/CA/Agent-for-SharePoint

- a. Do the appropriate step for your operating environment:
  - Windows: Right-click the executable and then pick Run as administrator.
  - Solaris: sh ./ca-spconnect-version\_number-sol.bin
  - Linux: sh ./ca-spconnect-version\_number-rhel30.bin

The SharePoint Connection wizard starts.

- b. Click Next.

The Login Details screen appears.

- c. Enter the following login for the Policy Server.

**Policy Server Name**

Specifies the Policy Server name or IP address.

**Username**

Specifies the Policy Server administrator username.

**Password**

Specifies the Policy Server administrator password.

**Agent Name**

Specifies the Agent-4x. The connection with the Policy Server is established using the details that are given in the Agent Name.

**Shared Secret Key**

Specifies the shared secret key that is associated with the Agent.

- d. Click Next

The Select Action screen appears.

- e. Select Edit a SharePoint Connection option.

- f. Click Next.

The SharePoint Connection Properties screen appears.

- g. Click through the wizard until you reach the Single Logout Configuration screen.
- h. Select the Enabled SignOut check box.
- i. Click the CleanUp URL field and then type the cleanup URLs from all of your protected web applications.

**Note:** Separate multiple URLs with semi-colons.

Use the following example as a guide:

`http://SharePoint_web_application_one_page_URL/_trust?wa=wsignoutcleanup1.0;http://SharePoint_web_application_two_page_URL/_trust?wa=wsignoutcleanup1.0`

- j. Click the Confirm URL field and type a single confirmation page URL. Use the following example as a guide:

`http://SharePoint_web_application_one_page_URL/affwebservices/spsignoutconfirmurl.jsp`

**Note:** If you have multiple public web applications.

*SharePoint\_web\_application\_one\_URL* can be the fully qualified domain that is named of any of those applications.

For example, if you have the following public SharePoint URLs:

`http://marketing.example.com` and `http://developers.example.com`,

*SharePoint\_web\_application\_one\_page\_URL* can be

`http://marketing.example.com` or `http://developers.example.com`.

- k. Click through the wizard until the Commit Details screen appears.
- l. Click Install.

The Save Complete screen appears.

- m. Click Done.

The SharePoint connection wizard closes. Single logout is enabled.

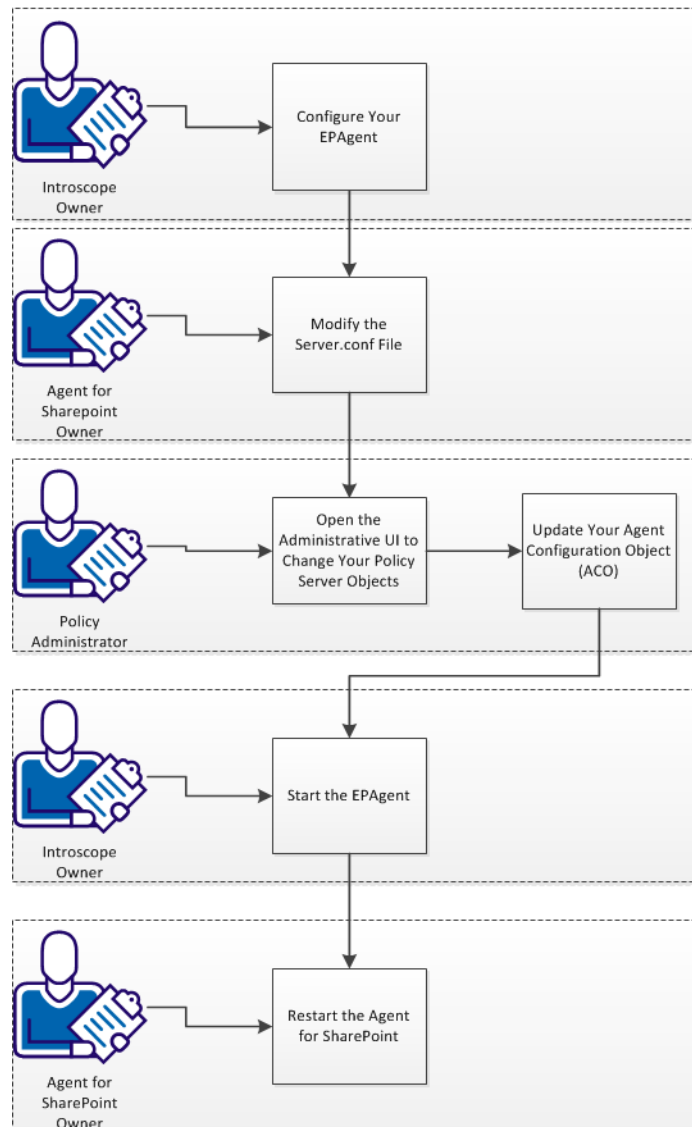


## How to Monitor Data with CA Introscope

CA Introscope® can monitor the following statistics of the server that hosts the Agent for SharePoint:

- The average response time for each of the following Agent for SharePoint components:
  - Session Discovery
  - Java Web Agent
  - Post Agent Session Writer
  - Proxy Rules Filter
  - Noodle Servlet
  - HTTP Client
- The average response time for each backend server.
- The wait times for the Agent for SharePoint requests.
- Number of hits for each proxy rule.
- The health data for the Agent for SharePoint framework instances.

### How to Monitor Data with CA Introscope®



**Follow these steps:**

1. [Configure your EPAgent](#) (see page 291).
2. [Modify the server.conf file](#) (see page 293).
3. [Open the Administrative UI to change Policy Server objects](#) (see page 42).
4. [Update your Agent Configuration object \(ACO\)](#) (see page 295).
5. Start the EPAgent using the appropriate procedure for your operating environment:
  - [Start your EPAgent on Windows operating environments](#) (see page 296).
  - [Start your EPAgent on UNIX operating environments](#) (see page 296).
6. [Restart the Agent for SharePoint](#) (see page 207).

## Configure Your EPAgent

You can configure the following items on your EPAgent:

- [Properties](#) (see page 291)
- [Logging options](#) (see page 292)
- [Plug-ins](#) (see page 292)
- [Network data sources](#) (see page 292)

## Configure EPAgent Properties

The properties used by the EPAgent are the same as the properties that are used for the Java agent.

**Follow these steps:**

1. Configure the EPAgent settings in the *IntroscopeEPAgent.properties* file. The settings for the EPAgent are the same type as found in the [assign the value for wisc in your book] Agent profile.

**Note:** For more information about properties, see the *[assign the value for wapm in your book] Java Agent Implementation Guide*.

2. If you change the name or location of the *IntroscopeEPAgent.properties* file, you can set it with the Java system property:

`-Dcom.wily.introscope.epagent.properties=filename`

**Note:** This system property should immediately follow "java" in the command line. If it is placed later on the command line—for example, after `-jar`—it will not work.

## Configure EPAgent Logging Options

By default, the EPAgent sends message and error output to the command console. You can configure the EPAgent to send message and error output to a log file also.

**Follow these steps:**

1. Open the file `<EPAgent_Home>/epagent/IntroscopeEPAgent.properties`.
2. Modify the properties.

## Configuring EPAgent Plug-ins

To run EPAgent using the default plug-ins, you simply need to uncomment certain properties in `IntroscopeEPAgent.properties` file.

However, you may want to remove plug-ins you don't need from the default plug-ins in the `IntroscopeEPAgent.properties` file, or add additional plug-ins.

EPAgent plug-ins are separated into two sections in the `IntroscopeEPAgent.properties` file, stateful and stateless.

## Configuring the EPAgent for Network Data Sources

You can configure the EPAgent to accept data from network sources.

- Configure the EPAgent for simple or XML network input.
- Configure the EPAgent for HTTP GET input.

## Modify the server.conf File

After the Introscope owner configures the EPAgent, the next step is modifying the server.conf file of the server on which the Agent for SharePoint runs.

### Follow these steps:

1. Log in to the server hosting your Agent for SharePoint.
2. Open the following file with a text editor:

*Agent-for-SharePoint\_Home/proxy-engine/conf/server.conf*

#### **Agent-for-SharePoint\_Home**

Indicates the directory where the CA SiteMinder Agent for SharePoint is installed.

**Default:** (Windows) [32-bit] C:\Program Files\CA\Agent-for-SharePoint

**Default:** (Windows) [64-bit] C:\CA\Agent-for-SharePoint

**Default:** (UNIX/Linux) /opt/CA/Agent-for-SharePoint

3. Locate the following section:  

```
<metric-reporter name="WilyMetricReporter">
```
4. Change the value on the enabled line (in the previous section) to "yes", as shown in the following example:  

```
enabled="yes"
```
5. Save the file and close the text editor.

The server.conf file is modified. Continue with the next steps of having your policy administrator open the Administrative UI and update your Agent Configuration object (ACO).

## Open the Administrative UI to Change Policy Server Objects

Change the objects on your Policy Server by opening the Administrative UI.

### Follow these steps:

1. Open the following URL in a browser.

`https://host_name:8443/iam/siteminder/adminui`

#### **host\_name**

Specifies the fully qualified Administrative UI host system name.

2. Enter your CA SiteMinder® superuser name in the User Name field.
3. Enter the CA SiteMinder® superuser account password in the Password field.

**Note:** If your superuser account password contains dollar-sign (\$) characters, replace each instance of the dollar-sign character with \$DOLLAR\$. For example, if the CA SiteMinder® superuser account password is \$password, enter \$DOLLAR\$password in the Password field.

4. Verify that the proper server name or IP address appears in the Server drop-down list.
5. Select Log In.

## Update Your Agent Configuration Object (ACO)

The next step in monitoring data with CA Introscope® involves updating a configuration parameter in your Agent Configuration Object (ACO).

### Follow these steps:

1. Click Infrastructure, Agent, Agent Configuration Objects.
2. Click the edit icon in the line Agent Configuration Object you want.
3. Locate the following parameter:

#### **EnableIntroscopeApiSupport**

Collects information about the agent and sends it to CA Introscope® using a plug-in. This parameter uses the following settings:

- When set to yes, the plug-in calls an API to collect the data.
- When set to no, the plug-in creates an HTTP header with the data.
- When set to both, the plug-in calls the API to collect the data *and* creates an HTTP header with the data.
- When set to none, data not collected.

**Default:** No.

**Limits:** Yes, Both, No, None.

**Example:** (HTTP header) sm-wa-perf-counters =  
`server_name.example.com:6180,86117203,86118343,1,0,0,1,0,0,1,0,0,0,0,0,1,0,0,0,0,0,0,1125,0,15,1,1,750,750,`

4. Click the edit icon next to the previous parameter, and then add the settings that you want.
5. (Optional) To *disable* automatic monitoring of the agent instance metrics that are configured with the Agent for SharePoint, change the value of the following parameter to no:

#### **EnableMonitoring**

Specifies whether the agent sends monitoring information to the Policy Server.

**Default:** Yes.

6. Click OK.
7. Click Submit.

Your Agent Configuration object is updated. Continue with the next step of having your Introscope administrator start the EPAGENT.

## Start Your EPAgent on Windows Operating Environments

You can run the EPAgent as either a standalone .jar file, or a Java application.

### To run the EPAgent as a standalone .jar file

- Run a Java command-line with the appropriate *-jar* flag, as in the following example:

```
java -jar <EPAgent_Home>/epagent/lib/EPAgent.jar
-Dcom.wily.introscope.epagent.properties=<EPAgent_Home>/epagent/IntroscopeEP
Agent.properties"
```

### To run the EPAgent as a Java application

- Add the EPAgent files to the appropriate *classpath* as in the following example:

```
java -classpath "<EPAgent_Home>/epagent/lib/EPAgent.jar"
-Dcom.wily.introscope.epagent.properties=<EPAgent_Home>/epagent/IntroscopeEP
Agent.properties"
```

## Start Your EPAgent on UNIX Operating Environments

You can use a control script (shell script) to run the Introscope EPAgent on a UNIX operating system.

### Follow these steps:

1. Open a command prompt.
2. Navigate to the directory that has the control script. For example:

```
cd Introscope<version_number>/bin
```

3. Run the command that corresponds to the action you want:

#### **EPACtrl.sh start**

Starts the EPAgent.

#### **EPACtrl.sh status**

Shows the status of EPAgent process whether it is running or stopped.

#### **EPACtrl.sh stop**

Stops the EPAgent process

#### **EPACtrl.sh help**

Displays the help menu.



## Restart the Agent for SharePoint

Starting or stopping the Agent for SharePoint involves the following separate procedures:

1. [Changing the value of EnableWebAgent in the WebAgent.conf file](#) (see page 109).
2. [Changing the state of the related services on the computer running the Agent for SharePoint](#) (see page 110).

## Change the Value of the EnableWebAgent Parameter

Change the value of the EnableWebAgent parameter to accomplish either of the following tasks:

- Start the Agent for SharePoint when the related services start.
- Stop the Agent for SharePoint when the related services start.

### Follow these steps:

1. Open the following file with a text editor:  
`Agent-for-SharePoint_home\proxy-engine\conf\defaultagent\WebAgent.conf`
2. Locate the following line:  
`EnableWebAgent="NO"`
3. Change the value inside the quotation marks to *one* of the following values:
  - YES to start the Agent for SharePoint after the services start. Your resources are protected.
  - NO to stop the Agent for SharePoint after the services start. Your resources are *not* protected.
4. [Change the state of the related services on your Agent for SharePoint](#) (see page 110).

## Change the States of the Services on your Agent for SharePoint

You can change the states of the related services on your Agent for SharePoint.

**Note:** To start or stop your Agent for SharePoint, [change the value of the EnableWebAgent parameter first](#) (see page 109).

### Follow these steps:

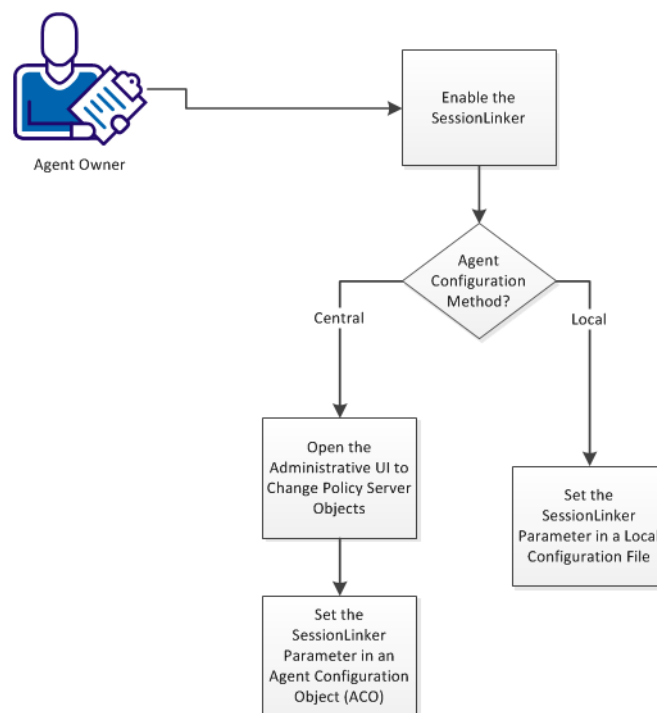
1. To change the states of the related services, select *one* of the following procedures:
  - For Windows operating environments, go to Step 2.
  - To *start* the Agent for SharePoint on UNIX operating environments, go to Step 3.
  - To *stop* the Agent for SharePoint on UNIX operating environments, go to Step 4.
2. For Windows operating environments, do the following steps:
  - a. From the Windows Start menu navigate to Administrative Tools, Services.  
The Services dialog appears.
  - b. Scroll down the list of services and select SiteMinder Agent for SharePoint.
  - c. From the Action menu, select All Tasks and select the command that you want.
  - d. Repeat Step b for SiteMinder Agent for SharePoint Proxy Engine.  
The states of the services and Agent for SharePoint are changed.
3. To start the Agent for SharePoint on UNIX operating environments, do the following steps.
  - a. Log in as a root user.
  - b. Navigate to the following directory:  
*Agent-for-SharePoint\_home/proxy-engine*
  - c. Run the following command:  
  
`./sps-ctl start`  
  
The service and the Agent for SharePoint start. The Agent for SharePoint stops or starts according to the [value you set in the EnableWebAgent parameter](#) (see page 109).
4. To stop the Agent for SharePoint on a system running UNIX, do the following steps:
  - a. Navigate to the following directory:  
*Agent-for-SharePoint\_home/proxy-engine*
  - b. Run the following command:  
  
`./sps-ctl stop`  
  
The service and the Agent for SharePoint stop.

## How to Use the Session Linker

The SessionLinker synchronizes a SiteMinder session with a third-party application session (such as SharePoint) for better security. If a user logs out from SiteMinder, the SessionLinker invalidates the related session of the third-party application.

Part of this synchronization process uses cookies from the third-party application. The SessionLinker requires certain information about these third-party cookies to link the sessions.

### How to Use the SessionLinker



**Follow these steps:**

1. Enable the session linker.
2. Pick the procedure that matches your agent configuration method from the following list:
  - For agents using an Agent Configuration object (ACO) on a Policy Server, follow these steps:
    - a. [Open the Administrative UI](#) (see page 42).
    - b. Set the [SessionLinker parameter in an Agent Configuration object](#) (see page 301).
  - For agents using a local configuration file on the, [set the Session parameter in a local configuration file](#) (see page 303).

## Enable the SessionLinker

Because the SessionLinker operates on a server, enable it on the server first.

**Follow these steps:**

1. Log in to the server that runs your Agent for SharePoint.
2. Open the following file with a text editor:

`Agent-for-SharePoint_Home\proxy-engine\conf\defaultagent\WebAgent.conf`

**Agent-for-SharePoint\_Home**

Indicates the directory where the CA SiteMinder Agent for SharePoint is installed.

**Default:** (Windows) [32-bit] C:\Program Files\CA\Agent-for-SharePoint

**Default:** (Windows) [64-bit] C:\CA\Agent-for-SharePoint

**Default:** (UNIX/Linux) /opt/CA/Agent-for-SharePoint

3. Locate the following line that applies to your operating environment::

(Windows)

`#LoadPlugin="Agent-for-SharePoint_Home\agentframework\bin\SessionLinkerPlugin.dll"`

(UNIX/Linux)

`#LoadPlugin="Agent-for-SharePoint_Home\agentframework\bin\LibSessionLinkerPlugin.so"`

4. Delete the # character at the beginning of the line.
5. Save the file and close the text editor.

The SessionLinker is enabled.

## Open the Administrative UI to Change Policy Server Objects

Change the objects on your Policy Server by opening the Administrative UI.

### Follow these steps:

1. Open the following URL in a browser.

`https://host_name:8443/iam/siteminder/adminui`

#### **host\_name**

Specifies the fully qualified Administrative UI host system name.

2. Enter your CA SiteMinder® superuser name in the User Name field.
3. Enter the CA SiteMinder® superuser account password in the Password field.  
**Note:** If your superuser account password contains dollar-sign (\$) characters, replace each instance of the dollar-sign character with \$DOLLAR\$. For example, if the CA SiteMinder® superuser account password is \$password, enter \$DOLLAR\$password in the Password field.
4. Verify that the proper server name or IP address appears in the Server drop-down list.
5. Select Log In.

## Set the SessionLinker Parameter in an Agent Configuration Object

A configuration parameter controls the SessionLinker. Add the SessionLinker parameter to your Agent Configuration object (ACO) if the configuration settings for your agents are centrally managed on a Policy Server.

### Follow these steps:

1. From the Administrative UI, click the Infrastructure, Agent Configuration Objects.
2. Click the edit icon in the line Agent Configuration Object you want.
3. Click Add.

The Create Parameter dialog appears.

4. Type the following text in the Name field:

SessionLinker

5. Click the Value field, and then add the following settings (on one line):

**Important!** Use semicolons (;) to separate each SessionLinker setting. For example, `Cookie=cookie_name;NOBLOT;URL=url_value;`

**COOKIE=cookie\_name;**

Specifies the name of the cookie from the third-party (foreign) application. If cookie names change, use an asterisk as a wildcard character. For example, if the cookies from your third party begin with APSESSION, use APPSESSION for the value of this setting.

**Examples: Cookie Names**

- `COOKIE=APSESSION;`
- `COOKIE=APP*;`
- `COOKIE=APPLICATION*;`

**BLOT | NOBLOT;**

(Optional) Specifies how the SessionLinker responds to invalid sessions. If the value of this parameter is set to BLOT, the user is granted access. The third party (foreign) session cookie is *not* passed through the web server to the target page. If the value of this parameter is set to NOBLOT, the user is redirected to URL specified in the URL setting. If the value of this setting is NOBLOT, set the URL parameter.

**Default:** BLOT

**Limits:** BLOT, NOBLOT

**URL=url\_value;**

Specifies a URL to where users are redirected when the value of the SessionLinker parameter contains NOBLOT. Users are directed to this URL and no the target page.

**Example:** `URL=/InvalidSessionWarning.jsp`

**ORPHANTIMEOUT=seconds**

Specifies the number of seconds that the SessionLinker maintains orphaned sessions.

**Default:** 86400 (24 hours)

**Limits:** Cannot be *less* than the *maximum number of seconds* that cookies from the third party (foreign) application are accepted.

**COOKIESCOPE=number\_of\_characters;**

(Optional) Specifies the number of characters in a URL, so that cookies used in more than area of a website can be distinguished. Suppose different applications use the same 15-character URL string as a prefix for naming its cookies. Use a larger value for the cookiescope setting. The larger number distinguishes between specific resources in other locations.

**Examples of URLs and corresponding values:**

- /scripts/wgate/ (15-character prefix string)
- /scripts/wgate/abc (18-character string)
- /scripts/wgate/xyz (18-character string)

**OPTIONS=USE\_HOST\_LINKS;**

Instructs the SessionLinker to link sessions for each virtual host defined in the server.conf file of your Agent for SharePoint.

**Default:** USE\_HOST\_LINKS

**Example:**

Cookie=cookie\_value;BLOT;Orphantimeout=1440;OPTIONS=USE\_HOST\_LINKS;

6. Click OK.
7. Click Submit.

The SessionLinker parameter is added to your Agent Configuration Object.

## Set the SessionLinker Parameter in a Local Configuration File

A configuration parameter controls the SessionLinker. Add the SessionLinker parameter to your local configuration file if the configuration settings for your agents are stored on each server.

**Follow these steps:**

1. Log in to the server that runs your Agent for SharePoint.
2. Open the following file with a text editor:

*Agent - for - SharePoint\_Home\proxy-engine\conf\defaultagent\LocalConfig.conf*

3. Locate the following line:

SessionGracePeriod="30"

4. Add line after the previous line.

**Note:** The order of the parameters in the LocalConfig.conf file does *not* matter, but having them in alphabetical order makes them easier to find.

5. Type the following text:

SessionLinker="

6. Add the following settings (on one line):

**Important!** Use semicolons (;) to separate each SessionLinker setting. For example, `Cookie=cookie_name;NOBLOT;URL=url_value;`

**COOKIE=cookie\_name;**

Specifies the name of the cookie from the third-party (foreign) application. If cookie names change, use an asterisk as a wildcard character. For example, if the cookies from your third party begin with APSESSION, use APPSESSION for the value of this setting.

**Examples: Cookie Names**

- `COOKIE=APSESSION;`
- `COOKIE=APP*;`
- `COOKIE=APPLICATION*;`

**BLOT | NOBLOT;**

(Optional) Specifies how the SessionLinker responds to invalid sessions. If the value of this parameter is set to BLOT, the user is granted access. The third party (foreign) session cookie is *not* passed through the web server to the target page. If the value of this parameter is set to NOBLOT, the user is redirected to URL specified in the URL setting. If the value of this setting is NOBLOT, set the URL parameter.

**Default:** BLOT

**Limits:** BLOT, NOBLOT

**URL=url\_value;**

Specifies a URL to where users are redirected when the value of the SessionLinker parameter contains NOBLOT. Users are directed to this URL and no the target page.

**Example:** `URL=/InvalidSessionWarning.jsp`

**ORPHANTIMEOUT=seconds**

Specifies the number of seconds that the SessionLinker maintains orphaned sessions.

**Default:** 86400 (24 hours)

**Limits:** Cannot be *less* than the *maximum number of seconds* that cookies from the third party (foreign) application are accepted.



**COOKIESCOPE=*number\_of\_characters*;**

(Optional) Specifies the number of characters in a URL, so that cookies used in more than area of a website can be distinguished. Suppose different applications use the same 15-character URL string as a prefix for naming its cookies. Use a larger value for the cookiescope setting. The larger number distinguishes between specific resources in other locations.

**Examples of URLs and corresponding values:**

- /scripts/wgate/ (15-character prefix string)
- /scripts/wgate/abc (18-character string)
- /scripts/wgate/xyz (18-character string)

**OPTIONS=USE\_HOST\_LINKS;**

Instructs the SessionLinker to link sessions for each virtual host defined in the server.conf file of your Agent for SharePoint.

**Default:** USE\_HOST\_LINKS

**Example:**

Cookie=*cookie\_value*;BLOT;Orphantimeout=1440;OPTIONS=USE\_HOST\_LINKS;

7. At the end of the line, type a double-quotation mark (").
8. Save the file, and then close the text editor.

The SessionLinker parameter is added to your local configuration file.

9. Repeat Steps 1 through 8 on all servers running your Agent for SharePoint that use local configuration.

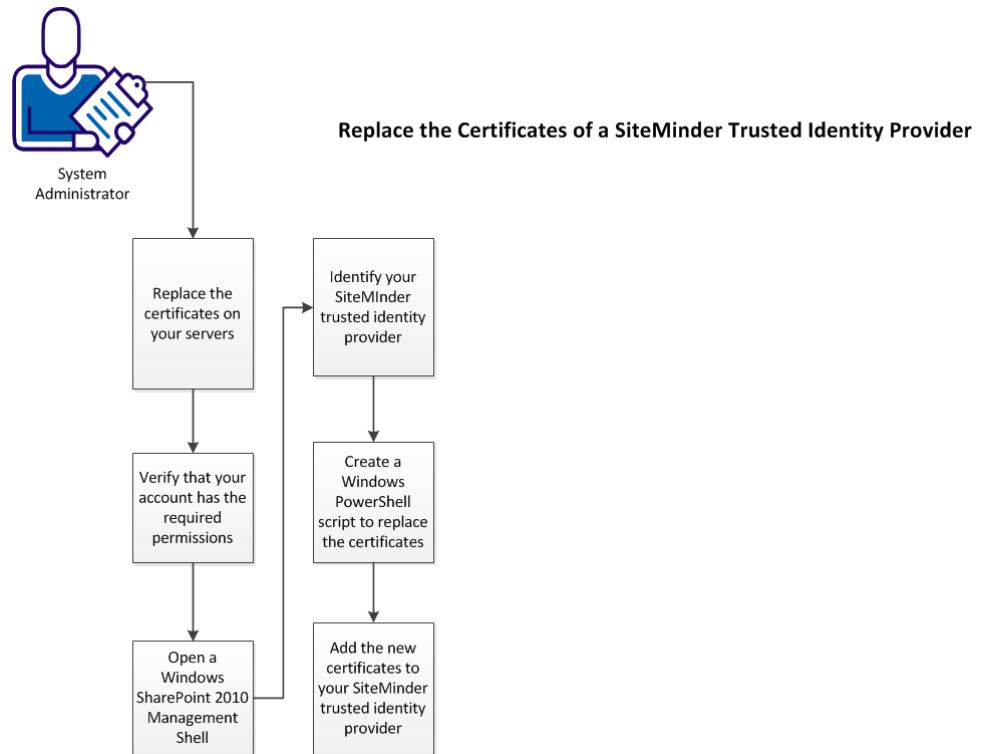
## How to Replace the Certificates for your CA SiteMinder® Trusted Identity Provider

CA SiteMinder® trusted identity providers use the following SSL certificates to encrypt their communications with the CA SiteMinder® Policy Server:

- A certificate authority certificate (CA-certificate or root certificate).
- An x.509 certificate (signing certificate).

When any of the previous certificates expire, you can replace them with valid certificates.

The following illustration describes how to replace the certificates of your CA SiteMinder® trusted identity provider:



**Follow these steps:**

1. [Replace the certificates on your servers](#) (see page 307).
2. [Verify that your account has the required permissions](#) (see page 308).
3. [Open a SharePoint 2010 management shell window on your SharePoint central administration server](#) (see page 308).
4. [Identify your CA SiteMinder® trusted identity provider](#) (see page 308).
5. [Create a Windows PowerShell script to update the certificates](#) (see page 309).
6. [Add the new certificates to your CA SiteMinder® trusted identity provider](#) (see page 310).

## Replace the Certificates on your Servers

Replace the expired certificates on the following computers:

- The computer hosting your SharePoint central administration server.
- Any computers hosting a web front end (WFE) for your SharePoint environment.

### Follow these steps:

1. Perform the following steps on the computer hosting your SharePoint central administration server:
    - a. Remove the expired CA-certificate (root certificate) from the computer.
    - b. Copy your new CA-certificate (root certificate) to the computer.

**Note:** Record this information for future use in your Windows PowerShell script.
    - c. Remove the expired signing certificate from the computer.
    - d. Copy your new signing certificate to the computer.

**Note:** Record this information for future use in your Windows PowerShell script.
  2. Perform the following steps on a computer hosting a web front end (WFE) server in your SharePoint environment:
    - a. Remove the expired CA-certificate (root certificate) from the computer.
    - b. Copy your new CA-certificate (root certificate) to the computer.

**Note:** Record this information for future use in your Windows PowerShell script.
    - c. Remove the expired signing certificate from the computer.
    - d. Copy your new signing certificate to the computer.

**Note:** Record this information for future use in your Windows PowerShell script.
  3. Repeat Step 2 for all web front end (WFE) servers in your SharePoint environment.
- The certificates on your computers have been replaced.

## Verify that your Account has the Required Permissions

The user account with which you want to modify the CA SiteMinder® trusted identity provider requires certain permissions. Modify the permissions of your user account if it does *not* meet the following conditions:

- An Administrator account.
- A member of the Administrators group.

Add the following privileges to your account:

- Local administrator on all SharePoint web front end (WFE) servers.
- Read/Write access to the configuration database.

## Open a SharePoint 2010 Management Shell Window on your SharePoint Central Administration Server

Add claims to your CA SiteMinder® trusted identity provider using the SharePoint 2010 Management shell.

### Follow these steps:

1. Log in to your SharePoint Central Administration server.
2. Click Start, All Programs, Microsoft SharePoint 2010 Products, SharePoint 2010 Management Shell.

A SharePoint 2010 management shell command-line window appears.

## Identify your Trusted Identity Provider

A SharePoint 2010 environment can have multiple trusted identity providers. Identify your CA SiteMinder® trusted identity provider before modifying any claims that are associated with it.

### Follow these steps:

1. Enter the following command to list all of the trusted identity providers:

```
Get-SPTtrustedIdentityTokenIssuer
```

A list of trusted identity providers appears.

2. Locate your CA SiteMinder® trusted identity provider in the list.

Your CA SiteMinder® trusted identity provider is identified.

## Create a PowerShell Script to Update the Certificates

Adding the new certificates to your CA SiteMinder® trusted identity provider involves several steps using the SharePoint 2010 Management shell.

We recommend using a PowerShell script that contains all of the commands, such as the one shown in the following example:

```
Remove-SPTrustedRootAuthority CASigningRootCert
Remove-SPTrustedRootAuthority CASigningCert

$rootcert = New-Object
System.Security.Cryptography.X509Certificates.X509Certificate2("full_path_to_updated_certificate_authority_certificate.cer")
$cert = New-Object
System.Security.Cryptography.X509Certificates.X509Certificate2("full_path_to_signing_certificate.cer")
$tip = Get-SPTrustedIdentityTokenIssuer
name_of_siteminder_trusted_identity_provider
$tip.SigningCertificate = $cert
$tip.Update()
New-SPTrustedRootAuthority -Name "CASigningRootCert" -Certificate $rootcert
New-SPTrustedRootAuthority -Name "CASigningCert" -Certificate $cert
```

**Follow these steps:**

1. Copy the example script shown previous and save it on your SharePoint central administration server as a .ps1 file.
2. Open the .ps1 file with a text editor.
3. Edit the .ps1 file to suit your environment with the following steps:

- a. Locate the following text:

*full\_path\_to\_updated\_certificate\_authority\_certificate*

- b. Replace the previous text with the full path to your new certificate authority (root) certificate.

**Example:** C:\exampleserver\certificates\rootcertificate.cer

- c. Locate the following text:

*full\_path\_to\_signing\_certificate*

- d. Replace the previous text with the full path to your new signing certificate.

**Example:**

C:\exampleserver\certificates\signingcertificates\sharepointsigningcertificate.cer

- e. Locate the following text:

*name\_of\_siteminder\_trusted\_identity\_provider*

- f. Replace the previous text with the name of your CA SiteMinder® trusted identity provider.

**Example:** SiteMinder\_TIP

4. Save the .ps1 file and close the text editor.

The Windows PowerShell script is created.

## Add the New Certificates to your CA SiteMinder® Trusted Identity Provider

Add the new certificates to your CA SiteMinder® trusted identity provider by running the PowerShell script on your SharePoint Central administration server.

**Follow these steps:**

1. Change the directory of your SharePoint 2010 Management shell window to the directory that contains your .ps1 file.
2. Execute your .ps1 file with the following command.

*\.name\_of\_your\_.ps1\_file.ps1*

The new certificates are added to the trusted identity provider.

## Virtual Hosts with the Agent for SharePoint

The following sections describe using virtual hosts with your Agent for SharePoint.

### Virtual Host Configurations Supported by the Agent for SharePoint

The CA SiteMinder® Agent for SharePoint supports virtual hosts. Virtual hosts conserve hardware resources by operating different websites on a single server.

The Agent for SharePoint supports virtual hosts that use the following configuration methods:

#### **Port-based virtual hosts**

Indicates a virtual host on your Agent for SharePoint server that operates on a unique port number.

#### **Host-header-based virtual hosts**

Indicates a virtual host on your Agent for SharePoint server that uses unique host header values.

#### **Path-based virtual hosts**

Indicates a virtual host on your Agent for SharePoint server using unique URI values.

#### **More information:**

[Set a Basic Proxy Rule for the Agent for SharePoint](#) (see page 97)

## Define Virtual Hosts for each Web Application

Virtual hosts are required for each SharePoint web application you want to protect. Define a virtual host for each SharePoint web application on the Agent for SharePoint server. A single virtual host definition on the Agent for SharePoint server accommodates the following types of proxy rules:

- Port-based forwarding
- Host-header-based forwarding
- Path-based forwarding

### Follow these steps:

1. Use a text editor to open the following file:

*Agent-for-SharePoint\_home\proxy-engine\conf\server.conf*

2. Locate the following line:

```
hostnames="default_SharePoint_URL"
```

3. Change the value of previous line to include a default URL to which you want to forward any requests that are *not* for your web applications. Any requests that are not for your web applications are forwarded to this default URL. For example, a generic SharePoint page can appear to users who do not request a specific resource.

4. Copy the following section:

```
<VirtualHost name="default">
 #addresses="192.168.1.100"
 hostnames="default_SharePoint_URL"
 defaultsessionscheme="default"

 # specify the block size for request and response in KBs
 requestblocksize="4"
 responseblocksize="4"

 #The defaults can be overridden
 #not only for the Virtual Host
 #but for the WebAgent for that
 #virtual host as well
 #<WebAgent>
 #</WebAgent>
</VirtualHost>
```

5. Add a new line below the </VirtualHost> tag.
6. Copy the section from Step 4 and paste it into the new line you created in Step 5.
7. Do the following steps:
  - a. Replace the word default in the <VirtualHost name= tag with a unique name you want.



- b. Replace the URL in the <hostnames= tag with the URL of your web application.
8. Save your changes to the file.
9. Repeat Steps 5 through 8 until virtual hosts are defined for all your web applications.

## How to Configure Port-based Virtual Hosts

Configuring port-based virtual hosts on your Agent for SharePoint is a process that involves several separate procedures. Some procedures involve different components in your environment. To configure port-based virtual hosts on your Agent for SharePoint server, use the following process:

1. [Define a virtual host for each web application](#) (see page 312).
2. Have your network administrator [update your DNS server with the virtual host settings](#) (see page 313).
3. [Create proxy rules for your port-based virtual hosts](#) (see page 314).
4. [On your SharePoint central administration server, do the following](#) (see page 315):
  - a. Change the public URL of the web application to the virtual host defined in the Agent for SharePoint.
  - b. Change the internal URL of the web application to the actual URL of your SharePoint resource.

## Update the DNS Tables with your Port-based Virtual Hosts

The virtual host names defined on your Agent for SharePoint require an association with the IP address of the Agent for SharePoint in the DNS servers of your organization. Have your network administrator update the DNS tables in your organization accordingly.

## Create Proxy Rules for your Port-based Virtual Hosts

Port-based virtual hosts require different settings than the default proxy rule file used by the Agent for SharePoint. After defining virtual hosts for your web applications, create proxy rules for your port-based virtual hosts.

### Follow these steps:

1. To preserve your current proxy rules, rename your existing proxyrules.xml file in the following directory:

*Agent - for - SharePoint\_home*\proxy-engine\conf

2. Open the following file with a text editor:

*Agent - for - SharePoint\_home*\proxy-engine\examples\proxyrules\proxyrules\_example1.xml

3. Save a copy of the previous file using the following path and file name:

*Agent - for - SharePoint\_home*\proxy-engine\conf\proxyrules.xml

4. Locate the following line:

```
<nete:proxyrules xmlns:nete="http://www.company.com/">
```

5. Replace the http://www.company.com/ with the name of your virtual host, as shown in the following example:

```
<nete:proxyrules xmlns:nete="http://www.example.com/">
```

6. Locate the following line:

```
<nete:case value="banking.company.com:80">
```

7. Replace the banking.company.com:80 with the domain name, suffix and port number for your SharePoint web application, as shown in the following example:

```
sharepoint.example.com:8606
```

8. Add your other web applications to the proxy rules file by repeating Steps 5 through 7 in the following section:

```
<!-- replace bondtrading.company.com with a virtual host defined in the
server.conf file -->
 <nete:case value="bondtrading.company.com:80">
 <!-- replace http://server2.company.com with the appropriate destination
server -->
 <nete:forward>http://server2.company.com$1</nete:forward>
 </nete:case>
```

9. Duplicate the previous section and modify it until all your port-based web applications have proxy rules.

10. Locate the following line:

```
<nete:forward>http://home.company.com$1</nete:forward>
```

11. Replace the `http://home.company.com` in the previous line with the URL of a default site you want to use for requests *not* matching your web applications.
12. Save the file and close the text editor.
13. [Restart the Agent for SharePoint](#) (see page 108).

## Add Public and Internal URLs on your SharePoint Server for your Port-Based Hosts

Port-based proxy rules require the following alternate access mappings on your SharePoint central administration server:

- Set the public URL for the zone to the URL of your virtual host associated with the web application.
- Set the internal URL to the server to which the requests from the virtual host are forwarded.

The following table describes an example of the alternate access mappings for port-based proxy rules:

Internal URL	Zone	Public URL for Zone
<code>http://www.sharepoint.example.com:8606</code>	Default	<code>http://sharepoint.example.com</code>

### Follow these steps:

1. Open your SharePoint central administration site.
2. Click Application Management.
3. Under Web Applications, click Configure Alternate Access Mappings.
4. Use the examples in the previous table as a guide to edit your public URLs and [Add Internal URLs](#) (see page 122).

## How to Configure Host-Header-Based Virtual Hosts

Configuring host-header-based virtual hosts on your Agent for SharePoint is a process that involves several separate procedures. Some procedures involve different components in your environment. To configure host-header-based virtual hosts on your Agent for SharePoint server, use the following process:

1. [Define a virtual host for each web application](#) (see page 312).
2. Have your network administrator [update your DNS server with the virtual host settings](#) (see page 316).
3. [Create proxy rules for your host-header-based virtual hosts](#) (see page 317).
4. [On your SharePoint central administration server, do the following](#) (see page 318):
  - a. Change the public URL of the web application to the virtual host defined in the Agent for SharePoint.
  - b. Change the internal URL of the web application to the actual URL of your SharePoint resource.

## Update the DNS Tables with your Host-Header-Based Virtual Hosts

The virtual host names defined on your Agent for SharePoint require an association with the IP address of the Agent for SharePoint in the DNS servers of your organization. Have your network administrator update the DNS tables in your organization accordingly.

## Create Proxy Rules for your Host-Header-Based Virtual Hosts

Host-header-based virtual hosts require different settings than the default proxy rule file used by the Agent for SharePoint. After defining your virtual hosts for your web applications, create proxy rules for your host-header-based virtual hosts.

### Follow these steps:

1. To preserve your current proxy rules, rename your existing proxyrules.xml file in the following directory:

*Agent-for-SharePoint\_home*\proxy-engine\conf

2. Open the following file with a text editor:

*Agent-for-SharePoint\_home*\proxy-engine\examples\proxyrules\proxyrules\_example2.xml

3. Save a copy of the previous file using the following path and file name:

*Agent-for-SharePoint\_home*\proxy-engine\conf\proxyrules.xml

4. Locate the following line:

```
<nete:proxyrules xmlns:nete="http://www.company.com/">
```

5. Replace `http://www.company.com/` in the previous line with the name of your virtual host, as shown in the following example:

```
<nete:proxyrules xmlns:nete="http://www.example.com/">
```

6. Locate the following line:

```
<nete:cond type="header" criteria="equals" headername="HEADER">
```

7. Replace `HEADER` in the previous line with the following:

```
HOST
```

8. Locate the following line:

```
<nete:case value="value1">
```

9. Replace `value1` in the previous line with the value of a host header you want, as shown in the following example:

```
<nete:case value="sharepoint.example.com">
```

10. Locate the following line:

```
<nete:forward>http://server1.company.com</nete:forward>
```

11. Replace the `http://server1.company.com` with the URL of the server to which you want to forward requests that use the header value from Step 8. Use the following example as a guide:

```
<nete:forward>http://sharepointserver1.example.com</nete:forward>
```

12. Add additional header values and destination servers by repeating Steps 8 through 11 on the following respective lines in the file:

```
<nete:case value="value2">
```

```
<nete:forward>http://server2.company.com</nete:forward>
```

13. Locate the following line:

```
<nete:forward>http://home.company.com$1</nete:forward>
```

14. Replace the `http://home.company.com` in the previous line with the URL of a default site you want to use for requests *not* matching your web applications.

15. Save the file and close the text editor.

16. [Restart the Agent for SharePoint](#) (see page 108).

## Add Public and Internal URLs on your SharePoint server for your Host-Header-Based Hosts

Host-header-based proxy rules require the following alternate access mappings on your SharePoint central administration server:

- Set the public URL for the zone to the URL of your virtual host associated with the web application.
- Set the internal URL to the server to which the requests from the virtual host are forwarded.

The following table describes an example of the alternate access mappings for host-header-based proxy rules:

Internal URL	Zone	Public URL for Zone
http://www.sharepointserver1.example.com	Default	http://sharepoint.example.com

### Follow these steps:

1. Open your SharePoint central administration site.
2. Click Application Management.
3. Under Web Applications, click Configure Alternate Access Mappings.
4. Use the examples in the previous table as a guide to edit your public URLs and [Add Internal URLs](#) (see page 122).

## How to Configure Path-based Virtual Hosts

If your web applications share the same ports and use host-header values to separate traffic, you can configure path-based virtual hosts.

Configuring path-based virtual hosts on your Agent for SharePoint is a process that involves several separate procedures. Some procedures involve different components in your environment. To configure path-based virtual hosts on your Agent for SharePoint server, use the following process:

1. [Define a virtual host for each web application](#) (see page 312).
2. Have your network administrator [update your DNS server with the virtual host settings](#) (see page 319).
3. [Create proxy rules for your path-based virtual hosts](#) (see page 320).
4. [On your SharePoint central administration server, do the following](#) (see page 321):
  - a. Change the public URL of the web application to the virtual host defined in the Agent for SharePoint.
  - b. Change the internal URL of the web application to the actual URL of your SharePoint resource.

## Update the DNS Tables with your Path-based Virtual Hosts

The virtual host names defined on your Agent for SharePoint require an association with the IP address of the Agent for SharePoint in the DNS servers of your organization. Have your network administrator update the DNS tables in your organization accordingly.

## Create Proxy Rules for your Path-based Virtual Hosts

Path-based virtual hosts require different settings than the default proxy rule file used by the Agent for SharePoint. After defining your virtual hosts for your web applications, create proxy rules for your path-based virtual hosts.

### Follow these steps:

1. To preserve your current proxy rules, rename your existing proxyrules.xml file in the following directory:

*Agent-for-SharePoint\_home*\proxy-engine\conf

2. Open the following file with a text editor:

*Agent-for-SharePoint\_home*\proxy-engine\examples\proxyrules\proxyrules\_example2.xml

3. Save a copy of the previous file using the following path and file name:

*Agent-for-SharePoint\_home*\proxy-engine\conf\proxyrules.xml

4. Locate the following line:

```
<nete:proxyrules xmlns:nete="http://www.netegrity.com/">
```

5. Replace `http://www.netegrity.com/` in the previous line with the name of your virtual host, as shown in the following example:

```
<nete:proxyrules xmlns:nete="http://www.example.com/">
```

6. Locate the following line:

```
<nete:case value="/dir1">
```

7. Replace the `/dir1` in the previous line with the path (URI) for which you want the request redirected. For example, if the path is `/sales`, all URLs containing `/sales` are redirected to the resource you specify.

8. Locate the following line:

```
<nete:forward>http://server1.company.com$2</nete:forward>
```

9. Replace the `http://server1.company.com` with the URL of the server to which you want to forward requests that use the path (URI) value from Step 8. Use the following example as a guide:

```
<nete:forward>http://sharepointserver1.example.com</nete:forward>
```

10. Add additional header values and destination servers by repeating Steps 6 through 9 on the following respective lines in the file:

```
<nete:case value="/dir2">
```

```
<nete:forward>http://server2.company.com$2</nete:forward>
```

11. Locate the following line:

```
<nete:forward>http://home.company.com$1</nete:forward>
```



12. Replace the `http://home.company.com` in the previous line with the URL of a default site you want to use for requests *not* matching your web applications.
13. Save the file and close the text editor.
14. [Restart the Agent for SharePoint](#) (see page 108).

## Add Public and Internal URLs on your SharePoint server for Path-Based Virtual Hosts

Path-based proxy rules require the following alternate access mappings on your SharePoint central administration server:

- Set the public URL for the zone to the URL of your virtual host associated with the web application.
- Set the internal URL to the server to which the requests from the virtual host are forwarded.

The following table describes an example of the alternate access mappings for path-based proxy rules:

Internal URL	Zone	Public URL for Zone
<code>http://www.sharepointserver1.example.com</code>	Default	<code>http://sharepoint.example.com</code>

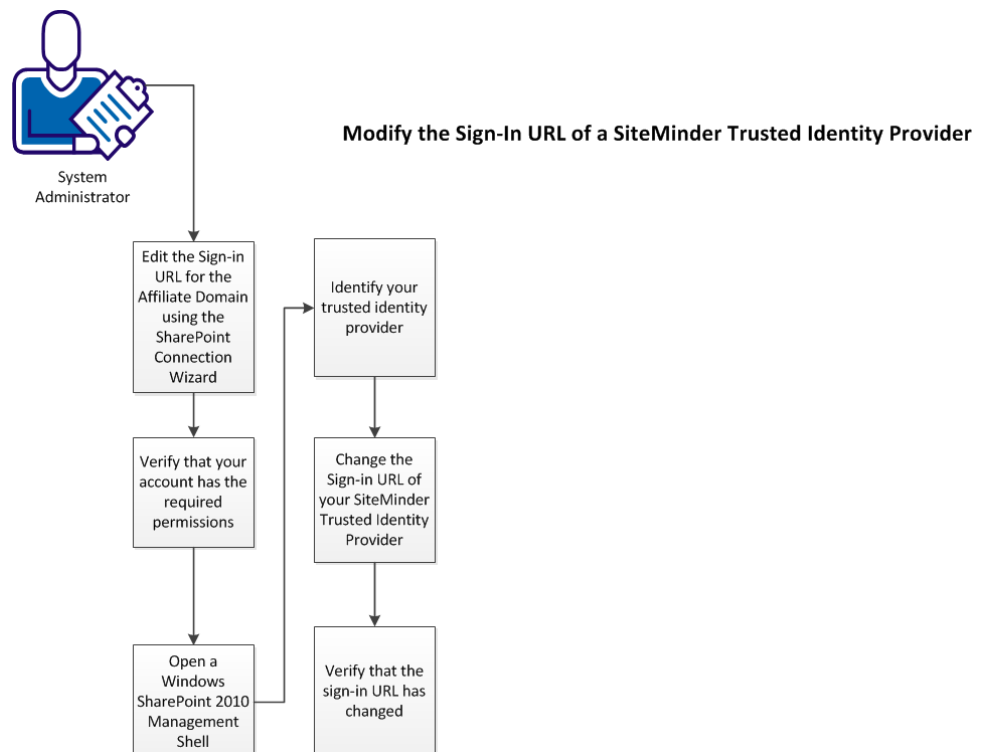
### Follow these steps:

1. Open your SharePoint central administration site.
2. Click Application Management.
3. Under Web Applications, click Configure Alternate Access Mappings.
4. Use the examples in the previous table as a guide to edit your public URLs and [Add Internal URLs](#) (see page 122).

## How to Modify the Sign-In URL of your CA SiteMinder® Trusted Identity Provider

This scenario describes changing the sign-in URL of your CA SiteMinder® trusted identity provider of an existing CA SiteMinder® environment. For example, update the URL if you change the protocol of your sign-in URL from HTTP to HTTPS.

The following illustration describes the process of modifying the sign-in URL of your CA SiteMinder® trusted identity provider:



To modify the sign-in URL of your CA SiteMinder® identity provider, follow these steps:

1. [Edit the sign-in URL for the affiliate domain using the SharePoint connection wizard](#) (see page 323).
2. [Verify that your account has the required permissions](#) (see page 137).
3. [Open a SharePoint 2010 Management Shell window on your SharePoint Central Administration server](#) (see page 137).
4. [Identify your CA SiteMinder® trusted identity provider](#) (see page 137).
5. [Change the sign-in URL of your CA SiteMinder® trusted identity provider](#) (see page 326).
6. [Verify that the sign-in URL has changed](#) (see page 326).

## Edit the Sign-In URL for the Affilliate Domain using the Sharepoint Connection Wizard

You can update the affiliate domain with a new sign-in URL for your CA SiteMinder® trusted identity provider. This update requires running the SharePoint connection wizard on the computer hosting your CA SiteMinder® Agent for SharePoint.

This procedure adds the new sign-in URL of your CA SiteMinder® trusted identity provider on your CA SiteMinder® Policy Server.

### Follow these steps:

1. Navigate to the following directory:

Agent-for-SharePoint\_home/sharepoint\_connection\_wizard

2. Do *one* of the following procedures:

- For Windows operating environments, right-click the executable and then select Run as administrator.

- For Solaris operating environments, enter the following command:

Solaris: `sh ./ca-spconnect-version-sol.bin`

- For Linux operating environments, enter the following command:

Linux: `sh ./ca-spconnect-version-rhel30.bin`

The wizard starts.

3. Click Next.

The Login Details screen appears.

4. Complete the following fields with the information from your existing CA SiteMinder® settings:

#### Policy Server Name

Specifies the Policy Server name or IP address.

#### Username

Specifies the Policy Server administrator username.

#### Password

Specifies the Policy Server administrator password.

#### Agent Name

Specifies the Agent-4x. The connection with the Policy Server is established using the details given in the Agent Name.

### Shared Secret Key

Specifies the shared secret key that is associated with the Agent.

5. Click Next

The Select Action screen appears.

6. Select Edit a SharePoint Connection option.

7. Click Next.

The SharePoint Connection Properties screen appears.

8. Click Next until the SharePoint Connection Properties screen appears.

9. Locate the following field:

### Authentication URL

Specifies the *port number that is associated* with the predefined protected URL which the SharePoint connection wizard adds automatically. When users try accessing a protected SharePoint resource without a SiteMinder session, they are redirected to the Authentication URL.

If you are using a default port number (such as 80 for HTTP or 443 for HTTPS), delete the <port> setting from this field.

**Note:** We recommend using HTTPS on production environments and pages which handle user credentials, such as login pages.

10. Change the protocol (such as HTTP or HTTPS) or the port number.

11. Click Next.

The attribute details are saved and the Commit Details screen appears.

12. Click Install in the Commit Details screen.

The Save Complete screen appears.

13. Click Done.

The partnership details are saved, the SharePoint Connection is modified, and the wizard closes.

## Verify that your Account has the Required Permissions

The user account with which you want to modify the CA SiteMinder® trusted identity provider requires certain permissions. Modify the permissions of your user account if it does *not* meet the following conditions:

- An Administrator account.
- A member of the Administrators group.

Add the following privileges to your account:

- Local administrator on all SharePoint web front end (WFE) servers.
- Read/Write access to the configuration database.

## Open a SharePoint 2010 Management Shell Window on your SharePoint Central Administration Server

Add claims to your CA SiteMinder® trusted identity provider using the SharePoint 2010 Management shell.

### Follow these steps:

1. Log in to your SharePoint Central Administration server.
2. Click Start, All Programs, Microsoft SharePoint 2010 Products, SharePoint 2010 Management Shell.

A SharePoint 2010 management shell command-line window appears.

## Identify your Trusted Identity Provider

A SharePoint 2010 environment can have multiple trusted identity providers. Identify your CA SiteMinder® trusted identity provider before modifying any claims that are associated with it.

### Follow these steps:

1. Enter the following command to list all of the trusted identity providers:

```
Get-SPTtrustedIdentityTokenIssuer
```

A list of trusted identity providers appears.

2. Locate your CA SiteMinder® trusted identity provider in the list.

Your CA SiteMinder® trusted identity provider is identified.

## Change the Sign-in URL of your CA SiteMinder® Trusted Identity Provider

Use the SharePoint 2010 Management Console to Changing the sign-in URL of your CA SiteMinder® trusted identity provider.

### Follow these steps:

1. Enter the following command to change the sign-in URL of your CA SiteMinder® trusted identity provider:

```
Set-SPTrustedIdentityTokenIssuer
"name_of_your_siteminder_trusted_identity_provider" -SignInUrl new_sign-in_URL
```

### Example: Changing Sign-in URL

This example shows how to change a sign-in URL for a trusted identity provider named SMTIP.

```
Set-SPTrustedIdentityTokenIssuer "SMTIP" -SignInUrl
https://sharepoint.example.com
```

The sign-in URL is changed.

## Verify that the Sign-in URL has Changed

You can verify the new sign-in URL for your CA SiteMinder® trusted identity provider.

### Follow these steps:

1. Enter the following command to verify the presence the new sign-in URL:

```
Get-SPTrustedIdentityTokenIssuer
```

A list of trusted identity providers and their respective settings appears.

2. Verify that the sign-in URL for your CA SiteMinder® trusted identity provider is correct.

## Configure the Agent for SharePoint for Web Applications That Use NTLM Authentication

If the web server uses a connection-oriented authentication scheme, configure a connection-oriented connection pool for secure forward request processing.

**Important!** We highly recommend that you do not configure a connection-oriented connection pool.

### Follow these steps:

1. Verify that the value for the JK environment variable REMOTE\_PORT is set in the httpd.conf file.
2. Open server.conf and add the following lines in <Service name="forward"> section:

```
Pool configuraiton for connection oriented authentication backend
connections eg: NTLM.
<connection-pool name="connection oriented authentication">
 connection-timeout="connection_timeout_value"
 max-size="maximum_connections"
 enabled="yes|no"
</connection-pool>
```

#### ***connection\_timeout\_value***

Defines the time in seconds the connection times out. We recommend that you set a lower value.

**Default:** 5

#### ***maximum\_connections***

Defines the number of connections in the connection pool.

**Default:** 50

#### ***yes/no***

Specifies the status of the connection-oriented connection pools. Set the value to yes to enable the connection-oriented connection pools.

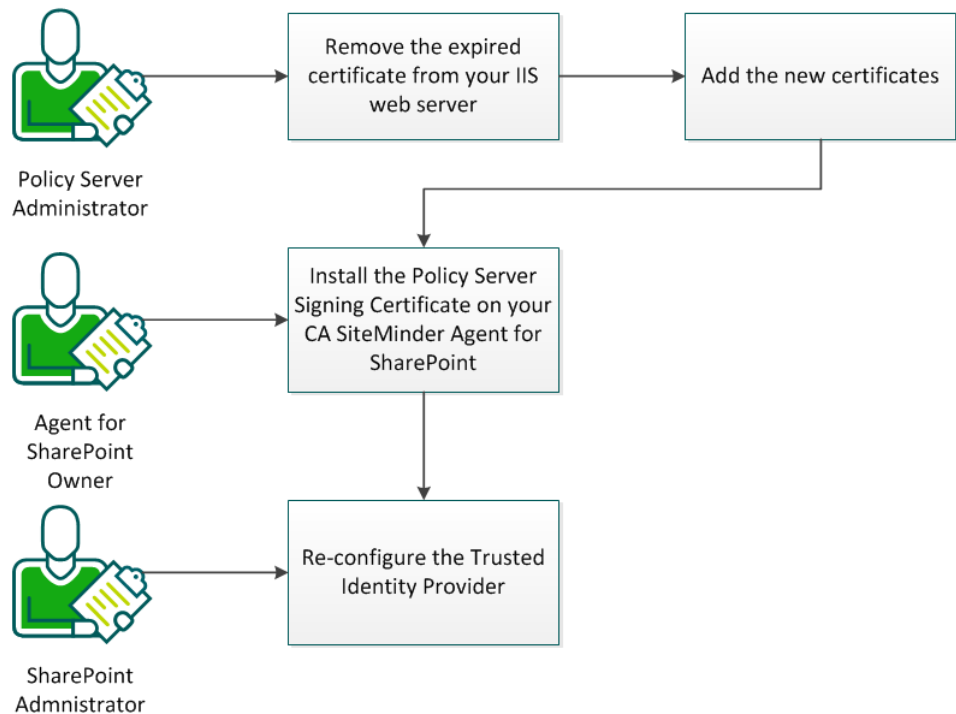
**Default:** yes

3. Open proxyrules.xml and add the connection-auth attribute to the forward rule.  
Example: <nete:forward connection-auth="yes">hostname:port\$1</nete:forward>

## How to replace the (WS-Fed) token signing certificates

You can replace your (WS-Fed) token signing certificates when they expire or if they have been revoked.

### How to replace the (WS-Fed) token signing certificates



#### Follow these steps:

1. Remove the expired certificate from your IIS web server.
2. Add the new certificates.
3. Install the Policy Server signing certificate on your CA SiteMinder Agent for SharePoint.
4. Re-configure the trusted identity provider.



## Remove the expired certificate from your IIS web server

When the token-signing certificate expires, remove it from your IIS web server.

**Follow these steps:**

1. Log in to your IIS web server.
2. Click Start, Administrative Tools, Internet Information Services (IIS).
3. Under Connections, expand your web server, and then double-click Server Certificates.

A list of certificates appears.

4. Right-click the expired certificate, and then pick Delete.
5. Close the IIS Manager.
6. Open a command-prompt window that has administrator privileges.
7. Run the following command:

```
iisreset
```

The expired certificate is removed.

## Add the new certificates

Add a new certificate to replace the expired one in your CA SiteMinder Agent for SharePoint environment.

**Follow these steps:**

1. [Create a certificate request for a server certificate on an IIS web server](#) (see page 77).
2. [Submit your certificate request to a certificate authority](#) (see page 78).
3. [Approve a certificate request using active directory certificate services](#) (see page 79).
4. [Complete your certificate request](#) (see page 80).
5. [Verify your approval and download your certificate and certificate chain](#) (see page 81).
6. [Export your policy server signing certificate](#) (see page 82).
7. [Add a policy server signing certificate to policy servers and create a trust file](#) (see page 83).

## Install the Policy Server Signing Certificate on your CA SiteMinder Agent for SharePoint

The CA SiteMinder Agent for SharePoint uses an embedded Apache web server. Install the Policy Server signing certificate you want to use for your SharePoint Connection. We recommend using a certificate signed by a Certificate Authority. After copying the certificate and related key files to your CA SiteMinder Agent for SharePoint, edit the configuration file for the embedded Apache web server.

### Follow these steps:

1. Copy the certificate files and related key files to your Agent for SharePoint.
2. Open the following file with a text editor:  
*Agent - for - SharePoint\_home\httpd\conf\extra\httpd-ssl.conf*
3. Edit the following directives in the file to point to your certificate and related key files (respectively):
  - SSLCertificateFile
  - SSLCertificateKeyFile

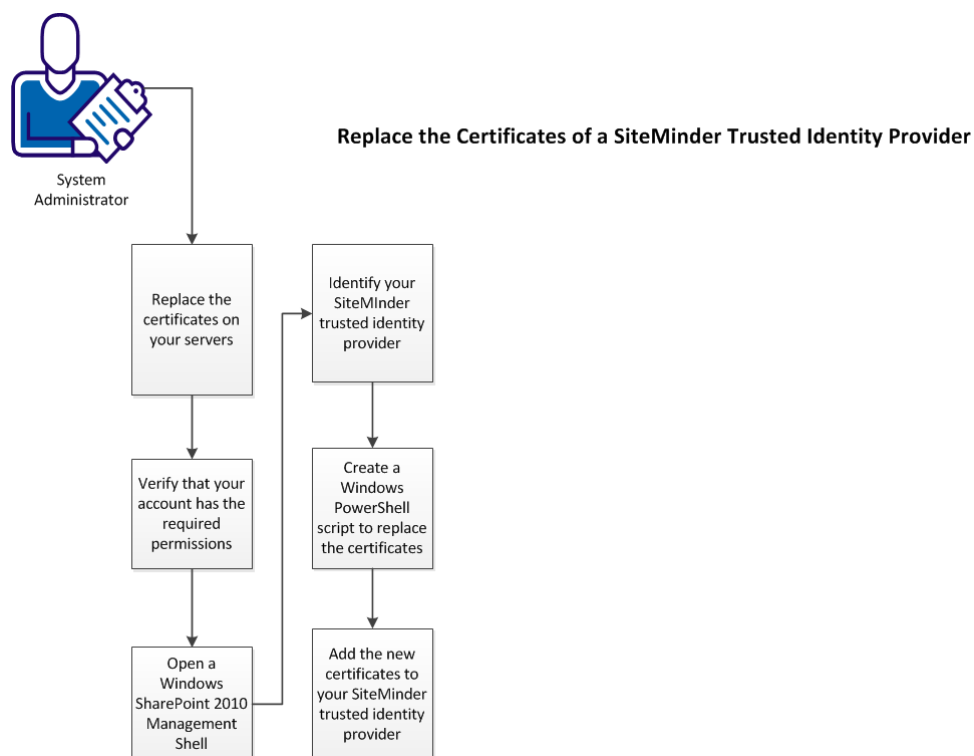
## Re-configure the trusted identity provider

CA SiteMinder® trusted identity providers use the following SSL certificates to encrypt their communications with the CA SiteMinder® Policy Server:

- A certificate authority certificate (CA-certificate or root certificate).
- An x.509 certificate (signing certificate).

When any of the previous certificates expire, you can replace them with valid certificates.

The following illustration describes how to replace the certificates of your CA SiteMinder® trusted identity provider:



**Follow these steps:**

1. [Replace the certificates on your servers](#) (see page 307).
2. [Verify that your account has the required permissions](#) (see page 308).
3. [Open a SharePoint 2010 management shell window on your SharePoint central administration server](#) (see page 308).
4. [Identify your CA SiteMinder® trusted identity provider](#) (see page 308).
5. [Create a Windows PowerShell script to update the certificates](#) (see page 309).
6. [Add the new certificates to your CA SiteMinder® trusted identity provider](#) (see page 310).



# Chapter 12: CA DLP Content Classification Service and the CA SiteMinder Agent for SharePoint

---

## Enable the DLP Plug-in

Enabling the DLP plug-in configures the agent to extract the resource information from the protected document. The agent passes the information to the Policy Server as part of the authorization process.

**Important!** A separate procedure is required in the application tier to enable the integration. Do not modify the web agent configuration file before the SharePoint agent configuration object is modified. The CA SiteMinder® administrator is responsible for completing the task.

### Follow these steps:

1. Log in to the system hosting your CA SiteMinder Agent for SharePoint.
2. Go to the following location:

*Agent-for-SharePoint\_Home\proxy-engine\conf\defaultagent*

#### **Agent-for-SharePoint\_Home**

Indicates the directory where the CA SiteMinder Agent for SharePoint is installed.

**Default:** (Windows) [32-bit] C:\Program Files\CA\Agent-for-SharePoint

**Default:** (Windows) [64-bit] C:\CA\Agent-for-SharePoint

**Default:** (UNIX/Linux) /opt/CA/Agent-for-SharePoint

3. Open the following file:

`WebAgent.conf`

4. Uncomment (remove the # sign to the left of) the line that loads the disambiguation plug-in.

**Example:** (Windows [32-bit]) LoadPlugin="C:\Program Files\CA\Agent-for-SharePoint\agentframework\bin\DisambiguatePlugin.dll"

**Example:** (Windows [64-bit])  
LoadPlugin="C:\CA\Agent-for-SharePoint\agentframework\bin\DisambiguatePlugin.dll"

**Example:** (UNIX/Linux)  
LoadPlugin="/opt/CA/Agent-for-SharePoint/agentframework/bin/DisambiguatePlugin.so"

5. Save the file.
6. Restart the web server.

The CA SiteMinder Agent for SharePoint is configured for the CA DLP integration.

## Set the Proxy Rules for the Agent for SharePoint when using CA DLP Content Classification Service with Multiple Authentication

The CA SiteMinder Agent for SharePoint operates as a proxy-based solution. To protect your SharePoint resources, edit the proxy rules file so that the Agent for SharePoint forwards requests to one of the following destinations:

- A hardware load balancer that redirects incoming requests to multiple web front ends associated with multiple SharePoint servers in a SharePoint server farm.
- A single web front end that is associated with multiple SharePoint servers in a SharePoint server farm.

When using the CA SiteMinder Agent for SharePoint, and the CA DLP content classification services together with multiple authentication, specific proxy rules are required for the proper protection of resources.

**Important!** Do not use any other proxy rule settings with the Agent for SharePoint, the CA DLP content classification service, and multi-authentication. Resources that the CA DLP content classification service classifies use an HTTP request header for proper forwarding by the Agent for SharePoint. If the Agent for SharePoint does not properly forward these requests using these rules (as they are shown here), unauthorized access or disclosure is possible.

### Follow these steps:

1. Locate the following file on your CA SiteMinder Agent for SharePoint:

*Agent-for-SharePoint\_home\proxy-engine\conf\proxyrules.xml*

#### **Agent-for-SharePoint\_Home**

Indicates the directory where the CA SiteMinder Agent for SharePoint is installed.

**Default:** (Windows) [32-bit] C:\Program Files\CA\Agent-for-SharePoint

**Default:** (Windows) [64-bit] C:\CA\Agent-for-SharePoint

**Default:** (UNIX/Linux) /opt/CA/Agent-for-SharePoint

2. Rename the previous file using a name similar to the following example:

*proxyrules\_xml\_default.txt*

3. Open the following file on your CA SiteMinder Agent for SharePoint with a text editor:

*Agent-for-SharePoint\_home\proxy-engine\examples\proxyrules\proxyrules\_example2.xml*

4. Save the previous file as a new file in the following location:

*Agent-for-SharePoint\_home\proxy-engine\conf\proxyrules.xml*

5. Locate the following text in the updated proxyrules.xml file:

```
:///$$PROXY_RULES_DTD$$"
```

6. Replace the previous text with the appropriate line for your operating environment:

- For Windows use the following line:

```
:///C:\Program
Files\CA\Agent-for-SharePoint\proxy-engine\conf\dtd\proxyrules.dtd"
```

- For UNIX/Linux, use the following line:

```
:///opt/CA/Agent-for-SharePoint/proxy-engine/conf/dtd/proxyrules.dtd"
```

**Note:** The previous examples indicate the default installation directory for the product. If you installed the product in a different directory, edit the examples to point to your installation directory instead.

7. Locate the following text:

```
http://www.company.com
```

8. Change the previous text to the domain of your organization. Use the following example as a guide:

```
http://www.example.com
```

9. Locate the following line:

```
<nete:cond type="header" criteria="equals" headername="HEADER">
```

10. Edit the previous line so that it matches the following line:

```
<nete:cond type="header" criteria="equals" headername="SMSERVICETOKEN">
```

11. Locate the following line:

```
<nete:case value="value1">
```

12. Edit the previous line so that it matches the following line:

```
<nete:case value="DLP">
```

13. Add a line after the previous line.

14. Copy and paste the following xml syntax onto the new line:

```
<nete:xprcond>
<nete:xpr>

<nete:rule>^/_login/default.aspx\?ReturnUrl=(.*)</nete:rule>
<nete:result>http://sharepoint.example.com:port_number/_trust/default.aspx?tr
ust=name_of_siteminder_trusted_identity_provider&ReturnUrl=$1</nete:resul
t>
</nete:xpr>

<nete:xpr-default>

<nete:forward>http://sharepoint.example:port_number$0</nete:forward>
</nete:xpr-default>
```



</nete:xprcond>

15. Replace both instances of the **sharepoint.example:port\_number** in the previous section with *one* of the following values:
  - The host name, domain, and port number of your hardware load balancer. This hardware load balancer operates between your CA SiteMinder Agent for SharePoint server and the SharePoint servers.
  - host name, domain, and port number of your single web front end. In this context, this web front end (WFE) refers a web server that operates in front of your "back end" SharePoint servers.
16. Replace the instance of *name\_of\_siteminder\_trusted\_identity\_provider* in the previous section with the name of your CA SiteMinder® trusted identity provider.
17. Locate the following line in the file:

<nete:forward>http://**home.company.com**\$0</nete:forward>

18. Replace the **home.company.com** in the previous line with *one* of the following values:
    - The host name, domain, and port number of your hardware load balancer. This hardware load balancer operates between your CA SiteMinder Agent for SharePoint server and the SharePoint servers.
    - host name, domain, and port number of your single web front end. In this context, this web front end (WFE) refers a web server that operates in front of your "back end" SharePoint servers.
  19. Save the file and close your text editor.
- The proxy rules are set.



# Chapter 13: Log Files

---

This section contains the following topics:

[logger.properties File Overview](#) (see page 339)

[Modifying the logger.properties File](#) (see page 339)

[Logging Settings](#) (see page 340)

## logger.properties File Overview

The CA SiteMinder Agent for SharePoint log settings are configured through the `logger.properties` file. These settings in the file are groups of name/value pairs or directives that the CA SiteMinder Agent for SharePoint reads at run time. You can update the `logger.properties` file without restarting the CA SiteMinder Agent for SharePoint.

The `logger.properties` file is located in the following (default) directory:

*Agent-for-SharePoint\_home/Tomcat/properties*

## Modifying the logger.properties File

The log settings for the CA SiteMinder Agent for SharePoint are maintained in the `logger.properties` file that is located in the following directory:

*Agent-for-SharePoint\_home/Tomcat/properties*

### **Follow these steps:**

1. Open the file in a text editor.
2. Edit the directives, as necessary.
3. Save the file.

The log settings are changed.

## Logging Settings

The logger.properties file contents are grouped into the following sections:

- The SvrConsoleAppender settings
- The SvrFileAppender settings
- The Server log settings
- The Server log rolling settings

The directives that are contained in this file follow the format name=value. Any lines beginning with the # symbol are comments, and are not read when the CA SiteMinder Agent for SharePoint loads configuration settings.

**Note:** Pathnames on Windows operating environments use double backslashes (\\).

### SvrConsoleAppender Settings

The SvrConsoleAppender Settings section contains settings for logging events on to a console. This section has the following format:

```
SvrConsoleAppender is set to be a ConsoleAppender.
log4j.appender.SvrConsoleAppender=org.apache.log4j.ConsoleAppender
log4j.appender.SvrConsoleAppender.layout=org.apache.log4j.PatternLayout
log4j.appender.SvrConsoleAppender.layout.ConversionPattern=<log_message_display_f
ormat_on_console>
```

#### ***log\_message\_display\_format\_on\_console***

Specifies the display format of a log message on the console. The CA SiteMinder Agent for SharePoint supports all the log4j date pattern strings.

**Default:** [%d{dd/MMM/yyyy:HH:mm:ss-SSS}] [%p] - %m%n

## SvrFileAppender Settings

The SvrFileAppender Settings section contains settings for logging events in a file. This section has the following format:

```
SvrFileAppender is set to be a FileAppender.
log4j.appender.SvrFileAppender=org.apache.log4j.FileAppender
log4j.appender.SvrFileAppender.layout=org.apache.log4j.PatternLayout
log4j.appender.SvrFileAppender.layout.ConversionPattern=<log_message_display_format_in_file>
```

### ***log\_message\_display\_format\_in\_file***

Specifies the display format of a log message in the file. The CA SiteMinder Agent for SharePoint supports all the log4j date pattern strings.

**Default:** [%d{dd/MMM/yyyy:HH:mm:ss-SSS}] [%p] - %m%n

## Server Log Settings

The server log settings section contains settings for enabling and disabling logging, setting logging level, and setting the output format of the log messages. This section has the following format:

```
Server.conf settings:
details of setting "log4j.rootCategory":
For First attribute:
Depending on the logging level needed, set the appropriate level
Possible values : OFF, FATAL, ERROR, WARN, INFO, DEBUG, ALL
For Second attribute:
if you want to enable log console, then add SvrConsoleAppender, else don't add this.
For Third attribute:
if you want to enable logging into file, theb add SvrFileAppender, else don't add this.
log4j.rootCategory=<log_level>,<output_format>
```

### ***log level***

Specifies the log level of a message. The following list displays values in starting with the lowest priority and moving to the highest:

- OFF
- FATAL
- ERROR
- WARN
- INFO
- DEBUG
- ALL

If the value is set to OFF, logging is disabled. If the value is set to any other value, logging is enabled.

**Default:** INFO

### ***output format***

Specifies how a log message is displayed. You can display a log message on a console, or can store it in a file, or both.

**Default:** SvrFileAppender

**Example:** The following command specifies the log level of INFO, and displays a log message on the console and stores it in a file:

```
log4j.rootCategory=INFO,SvrConsoleAppender,SvrFileAppender
```

## Server Log Rolling Settings

The server log rolling Settings section contains settings to enable the creation of a new log file at certain intervals. You can enable the log rolling based on *one* of the following mechanisms:

- Log rolling based on the file size
- Log rolling based on the age of the file

## Log Rolling Based on the File Size

The Log Rolling Based on the file size section contains settings for enabling the log rolling based on a file size. This section has the following format:

# Enable the below setting only if file logging is enabled above. if not make it as an comment by adding "#" at the begging of the line.

log4j.appender.SvrFileAppender.File=<logfile\_path>

# Enable this only if file logging is enabled above.

# set vale to "true" if messages are to be appended to the existing file. else set to "false"

log4j.appender.SvrFileAppender.Append=**true|false**

#Configurations to rollover server log file based on file size

log4j.appender.SvrFileAppender=org.apache.log4j.RollingFileAppender

log4j.appender.SvrFileAppender.MaxFileSize=<maximum\_logfile\_size>

log4j.appender.SvrFileAppender.MaxBackupIndex=<maximum\_number\_of\_logfile>

### **logfile path**

Specifies the name and path of the log file.

**Default Name:** server.log

**Default Path:** *Agent-for-SharePoint\_home*/secure-proxy/proxy-engine/logs/

### **true|false**

Specifies how the CA SiteMinder Agent for SharePoint manages the log file. When this value is set to true, the CA SiteMinder Agent for SharePoint appends new log messages to the existing log file. When this value is set to false, the CA SiteMinder Agent for SharePoint rolls over the existing log file by creating a log file for new messages.

**Default:** true

### **MaxFileSize**

Specifies the maximum size of the log file after which the CA SiteMinder Agent for SharePoint creates a log file.

**Default:** 1 MB

### **MaxBackupIndex**

Specifies the maximum number of log files that the CA SiteMinder Agent for SharePoint creates. When the number of log files exceeds this number, the CA SiteMinder Agent for SharePoint deletes the oldest log file. The CA SiteMinder Agent for SharePoint creates another log file.

**Default:** 10

## Log Rolling Based on the Age of the File

The Log Rolling Based on the file age section contains settings for enabling the log rolling based on the age of a file. This section has the following format:

```
#Configurations to rollover server log file based on time
#log4j.appender.SvrFileAppender=org.apache.log4j.DailyRollingFileAppender
#log4j.appender.SvrFileAppender.DatePattern=<date_pattern>
```

### ***date\_pattern***

Specifies the date when the CA SiteMinder Agent for SharePoint creates a log file.

**Default:** yyyy-MM-dd

The CA SiteMinder Agent for SharePoint creates another logfile in the following format:

```
<logfile_name>.<date_format>
```

### ***logfile\_name***

Specifies the name of the log file.

**Default:** server.log

### ***date\_format***

Specifies the date when the CA SiteMinder Agent for SharePoint created a logfile. The CA SiteMinder Agent for SharePoint supports all the log4j date pattern strings.

**Default:** yyyy-MM-dd



# Chapter 14: Troubleshooting

---

This section contains the following topics:

- [Attributes Appear Truncated in SharePoint](#) (see page 346)
- [Log Files Show Access Denied Due to BadURLChars Settings](#) (see page 347)
- [Log Files Show Access Denied Because of SPAuthorizeUserAgent Settings](#) (see page 349)
- [Enable Search of Custom Object Classes in Your LDAP Directory](#) (see page 350)
- [REST API in Excel Services Does Not Work Due to CSSChecking ACO Parameter](#) (see page 351)
- [Cannot Log Off Users from Sites and Subsites without Referring LogOffURI ACO \(CQ 135854\)](#) (see page 352)
- [Enable Paging for Searches of Active Directory User Stores \(32-bit systems\)](#) (see page 353)
- [Enable Paging for Searches of Active Directory User Stores \(64-bit systems\)](#) (see page 354)
- [Users Cannot Access Office Applications in Internet Explorer 7 when Office Client Integration Is Enabled](#) (see page 355)
- [I Can Only View Read Only Copies of Documents on SharePoint Sites, but the Office Client Integration Is Enabled](#) (see page 357)
- [SharePoint FedAuth Cookies and Office Client Integration Behavior](#) (see page 358)
- [Registration Failed with Unknown Error 127](#) (see page 358)
- [How to Reduce People Picker Timeouts with Large Databases on UNIX/Linux Operating Environments](#) (see page 359)
- [How to Reduce People Picker Timeouts with Large Databases on Windows Operating Environments](#) (see page 362)
- [Open the Windows Registry Editor](#) (see page 362)
- [Add the Registry Key](#) (see page 363)
- [Restart Your Policy Server](#) (see page 364)
- [Non-english Input Characters Contain Junk Characters](#) (see page 364)

## Attributes Appear Truncated in SharePoint

### Symptom:

I have noticed the following problems occurring:

- My directory attributes appear truncated in SharePoint.
- I see the following message in my log files:

```
[WARNING: Response attribute will be trimmed. [attr = FMATTR:memberOf] [actual
attr len = number] [response attr len = number]]
```

### Solution:

Do the following tasks:

1. Open the following file on your Policy Server:

```
policy_server_home\config\properties\EntitlementGenerator.properties
```

2. Locate the following line:

```
com.netegrity.assertiongenerator.wsfed.MaxUserAttributeLength=1024
```

3. Change the value 1024 (at the end of the line) to a larger number. We recommend using multiples of 1024.

## Log Files Show Access Denied Due to BadURLChars Settings

### Symptom:

The log files of my Agent for SharePoint show users were denied access to resources because of the settings in the BadURLChars parameter.

### Solution:

#### Follow these steps:

1. Examine the request to determine which character from the URL appears in the list of values for the following parameter:

#### BadUrlChars

Specifies the character sequences that cannot be used in URL requests. The Agent for SharePoint examines the characters in the URL that occur before the "?" character against those characters specified by this parameter. If any of the specified characters are found, the Agent for SharePoint rejects the request.

You can specify the following characters:

- a backward slash (\)
- two forward slashes (//)
- period and a forward slash (./)
- forward slash and a period (/.)
- forward slash and an asterisk (/\*)
- an asterisk and a period (\*.)
- a tilde (~)
- %2D
- %20
- %00-%1f
- %25 (do *not* add this value to the list if the URLs of your protected SharePoint resources contain blank spaces [%20])

Separate multiple characters with commas. Do *not* use spaces.

You can use the bad URL characters in CGI parameters if the question mark (?) precedes the bad URL characters.

**Default:** (Agent for SharePoint) //,./,./,/\*,\*,~,\\,%00-%1f

#### Limits:

- The default hexadecimal numbers apply to English characters. For other languages, remove any hexadecimal values that correspond to the characters of the language that you want to allow. Examples of such languages include (but are not limited to), Brazilian Portuguese, French, Japanese, and Chinese.
  - You can specify characters literally. You can also enter the URL-encoded form of that character. For example, you can enter the letter a, or you can enter the encoded equivalent of %61.
  - You can specify a maximum number of 4096 characters (including commas that are used for separating characters).
  - You can specify ranges of characters that are separated with hyphens. The syntax is: *starting\_character-ending\_character*. For example, you can enter a-z as a range of characters.
  - Specify any quotation marks (") with the URL-encoded equivalent of %22. Do *not* use ASCII.
2. Remove the character in your URL from the list of values in the previous parameter.

## Log Files Show Access Denied Because of SPAuthorizeUserAgent Settings

### Symptom:

The trace log files of my Agent for SharePoint show users were denied access to resources because of the settings in the SPAuthorizeUserAgent parameter.

### Solution:

#### Follow these steps:

1. Examine the request shown in the trace log file to determine which User Agent string value was denied access. The following example shows typical trace log file results for this parameter:

```
spauthorizeuseragent=Microsoft Office Protocol Discovery
spauthorizeuseragent=Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; WOW64;
Trident/4.0; SLCC2; .NET CLR 2.0.50727; FDM; .NET CLR 3.5.30729; .NET CLR
3.0.30729; .NET4.0C; .NET4.0E)
spauthorizeuseragent=Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1; WOW64;
Trident/4.0; SLCC2; .NET CLR 2.0.50727; FDM; .NET CLR 3.5.30729; .NET CLR
3.0.30729; .NET4.0C; .NET4.0E)
spauthorizeuseragent=Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US;
rv:1.9.2.13) Gecko/20101203 Firefox/3.6.13
spauthorizeuseragent=Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; WOW64;
Trident/4.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729;
InfoPath.2; MS-RTC LM8; .NET4.0C)
spauthorizeuseragent=Microsoft Office/12.0
spauthorizeuseragent=Microsoft Office/12.0 (Windows NT 6.1; Microsoft Office
Word 12.0.6545; Pro)
spauthorizeuseragent=Microsoft Office Existence Discovery
spauthorizeuseragent=Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; WOW64;
Trident/4.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729;
InfoPath.2; MS-RTC LM8; .NET4.0C; MSOffice 12)
spauthorizeuseragent=MSFrontPage/12.0
spauthorizeuseragent=Mozilla/4.0 (compatible; MS FrontPage 12.0)
spauthorizeuseragent=Microsoft-WebDAV-MiniRedir/6.1.7600
```

2. Add the user agent string from your trace log to the list of values in the following parameter:

#### SPAuthorizeUserAgent

Specifies a list of Microsoft Office user-agent strings for which the Agent for SharePoint allows access. This list is populated automatically with the default values when the Agent for SharePoint starts. The user-agent strings in this parameter act as a whitelist. Changes to this parameter override the default settings. Access is denied to clients whose user-agent string does not appear in the list.

For example, setting the value to Microsoft Office allows access to all versions of Microsoft Office products that are associated with that user-agent string. Conversely, setting the value to Microsoft Office/12.0 allows access to only those versions of Microsoft Office products that are associated with that user-agent string.

**Default:** Microsoft Office, MS FrontPage, MSFrontPage, Microsoft Data Access Internet Publishing Provider Protocol Discovery, Test for Web Form Existence, Microsoft-WebDAV-MiniRedir

**Limits:** Multiple values are allowed.

## Enable Search of Custom Object Classes in Your LDAP Directory

### Symptom:

My LDAP directory contains custom object classes, but SiteMinder does not find them during searches.

### Solution:

#### Follow these steps:

**Note:** For UNIX and Linux environments, navigate to the /registry directory, and then locate the previous setting in the sm.registry file.

1. Open the following registry location on each Policy Server:

HKLM\SOFTWARE\Netegrity\SiteMinder\CurrentVersion\Ds\GroupClassFilters

2. Locate the following key:

LDAP:

3. Change the value of the data to the following:

\*

4. Navigate to the following key:

HKLM\SOFTWARE\Netegrity\SiteMinder\CurrentVersion\  
Ds\ClassFilters

5. Locate the following key:

LDAP:

6. Change the value of the data to the following:

\*

## REST API in Excel Services Does Not Work Due to CSSChecking ACO Parameter

### Symptom:

REST API in Excel Services does not work when a SharePoint web application using Claims-based Authentication is protected with CA SiteMinder®.

### Solution:

The REST API in Excel Services does not work because the CSSChecking ACO parameter is enabled by default. CSSChecking verifies URLs for escaped and unescaped characters defined in the BadCSSChars parameter and returns with an Access Denied message.

Disable the CSSChecking ACO parameter. This change allows the REST API in Excel Services to work when a SharePoint web application using Claims-based Authentication is protected with CA SiteMinder®.

### Follow these steps:

1. Log on to the SiteMinder Administrative UI.  
The relevant tabs for your administrator privileges appear.
2. Click Infrastructure, Agents, Agent Configuration, Modify Agent Configuration.  
The Modify Agent Configuration: Search screen opens.
3. Select the Agent Configuration object from the list and click Select.  
The Modify Agent Configuration: ACO\_Name dialog appears.
4. Click the Edit button on the CsxChecking Parameter.  
The Edit Parameter dialog appears.
5. Enter No in the Value field and click OK.  
The Modify Agent Configuration: ACO\_Name dialog appears with the General and Parameters group boxes.
6. Click Submit.  
The Modify Agent Configuration Object task is submitted for processing and the confirmation message appears.

## Cannot Log Off Users from Sites and Subsites without Referring LogOffURI ACO (CQ 135854)

**Symptom:**

I cannot log off from a SharePoint site or a subsite.

**Solution:**

You cannot log off from a SharePoint site or a subsite because the actual logoff URL is different. Verify the signoff URL for each of the sites and subsites and add them to the LogOffURI ACO parameter.

For example, assume <http://example.ca.com/> is the main site and <http://example.ca.com/hr> and <http://example.ca.com/finance> are subsites. The logoff URI is different for each of the sites and the subsites. Configure all of the sign-out pages as logoff URIs in the LogOffURI ACO parameter.



## Enable Paging for Searches of Active Directory User Stores (32-bit systems)

**Valid for Policy Servers that are installed on Windows 32-bit operating environments that are connected to Active Directory servers.**

**Symptom:**

I cannot use the SharePoint people picker to search my Active Directory user store.

**Solution:**

The Active Directory namespace does not support paging, causing searches of more than 1000 users to fail. To support searches of large numbers of users in the Active Directory namespace, set the EnablePagingADNameSpace registry key to one.

**To enable paging for searches on your Windows Policy Server:**

1. Open the Windows registry editor.
2. Locate the following registry key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Netegrity\SiteMinder\CurrentVersion\Ds\LDAPProvider\EnablePagingADNameSpace
```

3. Set the value of the key to 1.

**To enable paging for searches on your UNIX Policy Server:**

1. Navigate to *policy\_server\_installation\_directory/siteminder/registry*
2. Open sm.registry in a text editor.
3. Locate the following text in the file:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Netegrity\SiteMinder\CurrentVersion\Ds\LDAPProvider\EnablePagingADNameSpace
```

4. Set the value of the key to 1.

## Enable Paging for Searches of Active Directory User Stores (64-bit systems)

**Valid for Policy Servers that are installed on Windows 64-bit operating environments (using WoW64 mode) that are connected to Active Directory servers.**

**Symptom:**

I cannot use the SharePoint people picker to search my Active Directory user store.

**Solution:**

The Active Directory namespace does not support paging, causing searches of more than 1000 users to fail. To support searches of large numbers of users in the Active Directory namespace, set the EnablePagingADNameSpace registry key to one.

**To enable paging for searches on your Windows Policy Server:**

1. Open the Windows registry editor.
2. Locate the following registry key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Netegrity\SiteMinder\CurrentVersion\Ds\LDAPProvider\EnablePagingADNameSpace
```

3. Set the value of the key to 1.

## Users Cannot Access Office Applications in Internet Explorer 7 when Office Client Integration Is Enabled

### Valid for Windows Vista

#### Symptom:

I cannot access Office applications in Internet Explorer 7 when Office Client Integration is enabled.

#### Solution:

This error is the result of a known Microsoft issue. Persistent cookies are not shared between Internet Explorer 7 and Office applications in Windows Vista. Internet Explorer 7 has an isolated cache location where files saved by web pages and persistent cookies are saved.

To access Office applications, add the SharePoint site to the list of trusted sites. This change enables the Web to save persistent cookies and temporary files to the regular cache. In this location, persistent cookies and temporary files are available to Office applications.

The following procedure shows how to add the SharePoint site (<http://spagent.example:port>) to the list of trusted sites in Internet Explorer 7.

#### Follow these steps:

1. Open Internet Explorer 7 browser.
2. Click Internet Options in the Tools menu.  
The Tools menu opens.
3. Click on the Security tab.  
The Security tab opens.
4. Click on Trusted Sites.  
The Trusted Sites icon is selected and the description appears.
5. Click on the Sites button.
6. Type the SharePoint site <http://spagent.example:port> into the text box and click the Add button.
7. (Optional) Clear the Require server verification (<https://>) option.  
**Note:** Clear the Require server verification (<https://>) option to add sites to the zone that do not use the <https://> protocol. This setting protects your information while it is being transferred to the server that the site is hosted on.
8. Click the Close button.  
The Trusted Sites dialog opens.

9. Click OK.

The Internet Options dialog opens.

**Note:** For more information about this issue, see the KB article 932118 on Microsoft Support site.

## I Can Only View Read Only Copies of Documents on SharePoint Sites, but the Office Client Integration Is Enabled

### Symptom:

My SharePoint servers have the Office Client Integration feature enabled, but I cannot open any of the documents for editing. I can only open read-only files.

Sometimes I also see the following error message:

An error (1502) occurred during the action Open File. File not found.

### Solution:

Verify that the host names in the following Agent for SharePoint configuration parameter do *not* contain port numbers:

#### SPClientIntegration

Specifies the hostnames of the SharePoint servers that the Agent for SharePoint protects on which you want to permit Office Client Integration. The default parameter is blank and listed as plain. If there are multiple host entries, use the multivalue option button to add multiple hosts.

Add a port number to the value if the Agent for SharePoint operates on a nondefault port (any port except 80 or 443).

To use this parameter, verify that the SharePoint resources that CA SiteMinder® protects also have their Office Client Integration enabled on the SharePoint central administration server.

Because Office Client Integration requires a persistent FedAuth cookie, verify that your SharePoint server is *not* configured to use session cookies. By default, UseSessionCookies in SharePoint is set to NO.

**Default:** None

**Limits:** Multiple values are allowed. Use fully qualified domain names for all values.

**Example:** *agent\_for\_sharepoint\_host\_name.example.com* (default ports of 80 or 443)

**Example:** *agent\_for\_sharepoint\_host\_name.example.com:81* (with a nondefault port number for HTTP)

**Example:** *agent\_for\_sharepoint\_host\_name.example.com:4343* (with a nondefault port number for HTTPS)

## SharePoint FedAuth Cookies and Office Client Integration Behavior

**Symptom:**

SharePoint stores a persistent FedAuth cookie on the hard drives of authenticated users. I do not want the SharePoint server to use these persistent cookies.

**Solution:**

You can configure SharePoint so a persistent FedAuth cookie is not stored. However, disabling the persistent FedAuth cookie also disables the single-sign on function of Office Client Integration. Users who try to open files on the SharePoint server are challenged for their credentials.

**Note:** For more information about how to disable FedAuth cookies in SharePoint 2010, go to the [technet blogs](#) website, and then search for the following phrase:

"Setting the Login Token Expiration Correctly for SharePoint 2010 SAML Claims Users"

## Registration Failed with Unknown Error 127

**Valid on Linux operating environments**

**Symptom:**

I received the following error:

Registration Failed: Unknown Error 127

**Solution:**

Verify that your Linux operating environment meets the proper prerequisites.

**More information:**

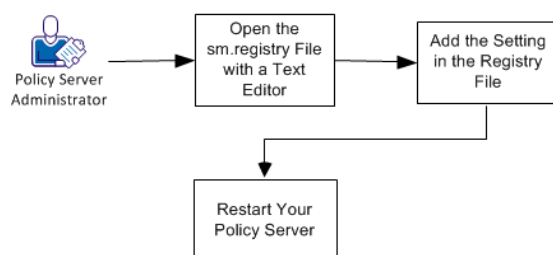
[Agent for SharePoint Prerequisites for Linux Operating Environments](#) (see page 34)

## How to Reduce People Picker Timeouts with Large Databases on UNIX/Linux Operating Environments

Sometimes the SharePoint people picker times out if the user database is large. This situation occurs because the CA SiteMinder Agent for SharePoint is searching a large group of users in a database.

To reduce these timeouts, you can change a setting in the registry.

### How to Reduce People Picker Time Outs with Large Databases



#### Follow these steps:

1. [Open the sm.registry file with a text editor](#) (see page 359).
2. [Add the setting in the registry file](#) (see page 360).
3. [Restart Your Policy Server](#) (see page 361).
4. Repeat Steps 1 through 4 for all your Policy Servers.

## Open the sm.registry File with a Text Editor

Change this setting on UNIX or Linux operating environments by opening the sm.registry file with a text editor. The sm.registry file is stored on your Policy Server.

#### Follow these steps:

1. Navigate to the following directory:

*Installation\_Directory/registry*

***installation\_directory***

Specifies the location in the file system where the Policy Server is installed.

**Default:** /opt/CA/siteminder

2. Open the following file with a text editor:

sm.registry

You can now change the settings.

## Add the Setting in the Registry File

The following registry key controls people picker sorting and reduces timeouts:

### **EnableSorting**

Specifies the sorting method the product uses when searching your Active Directory database. The following methods are available:

- Disabled. Set the value to 0.
- Enabled for unknown directories. Set the value to 1.
- Enabled for all directories. Set the value to 2.

**Default:** 2

### **Follow these steps:**

1. Locate the following section of the sm.registry file:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Netegrity\SiteMinder\CurrentVersion\Ds\LDAPProvider=
```

2. Add the following line in the LDAPProvider section:

```
EnableSorting= 0; REG_DWORD
```

3. Change the zero in the previous line to *one* of the following values:

- 1 (enables sorting for unknown directories)
- 2 (enables sorting for all directories)

4. Verify that the line in your sm.registry file matches *one* of the following examples:

```
EnableSorting= 1; REG_DWORD
```

```
EnableSorting= 2; REG_DWORD
```

5. Save the changes to the sm.registry file, and then close the text editor.

The setting is added to the registry file.



## Restart Your Policy Server

Apply the changes that you made previously by restarting your Policy Server.

### Follow these steps:

1. Log in to the system hosting the Policy Server with the same user account that installed the Policy Server originally.
2. Stop all Policy Server processes, with *one* of the following actions:
  - Open the Management Console, click the Status tab, and then click the Stop buttons.
  - Use the following script. This script also stops the UNIX executive so that the processes do not restart automatically.

```
installation_path/siteminder/stop-all
```

- The following script can stop the Policy Server processes. The UNIX executive starts with the script (if it is not already running). The script can be invoked using the command line with the following options:

```
installation_path/siteminder/smpolsrv -stop
```

The Policy Server logs all UNIX executive activity in the `installation_directory/log/smexec.log` file. Log entries are always appended to the existing log file.

3. Start all Policy Server processes, with *one* of the following actions:
  - Open the Management Console, click the Status tab, and then click the Start buttons.
  - Use the following script. This script also starts the UNIX executive.

```
installation_path/siteminder/start-all
```

- The following script can start the Policy Server processes. The UNIX executive starts with the script (if it is not already running). The script can be invoked using the command line with the following options:

```
installation_path/siteminder/smpolsrv -start
```

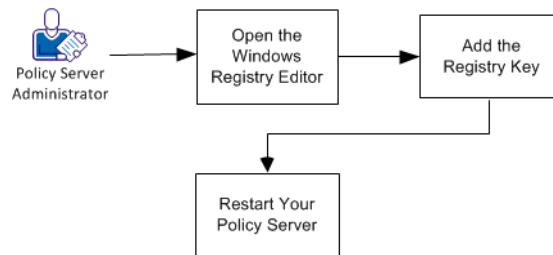
The Policy Server logs all UNIX executive activity in the `installation_directory/log/smexec.log` file. Log entries are always appended to the existing log file.

## How to Reduce People Picker Timeouts with Large Databases on Windows Operating Environments

Sometimes the SharePoint people picker times out if the user database is large. This situation occurs because the CA SiteMinder Agent for SharePoint is searching a large group of users in a database.

To reduce these timeouts, you can change a setting in the registry.

### How to Reduce People Picker Time Outs with Large Databases



#### Follow these steps:

1. [Open the Windows registry editor](#) (see page 362).
2. [Add the registry key](#) (see page 363).
3. [Restart your Policy Server](#) (see page 364).
4. Repeat Steps 1 through 4 for all your Policy Servers.

## Open the Windows Registry Editor

Change this setting by opening the Windows registry editor on the system hosting your Policy Server.

#### Follow these steps:

1. Click Start, Run.
2. Type the following text in the Open: Field.  
regedit
3. Click OK.

The Windows registry editor opens.

## Add the Registry Key

The following registry key controls people picker sorting and reduces timeouts:

### **EnableSorting**

Specifies the sorting method the product uses when searching your Active Directory database. The following methods are available:

- Disabled. Set the value to 0.
- Enabled for unknown directories. Set the value to 1.
- Enabled for all directories. Set the value to 2.

**Default:** 2

### **Follow these steps:**

1. In the registry editor, expand the following item:  
`HKEY_LOCAL_MACHINE`
2. Click Software, Netegrity, SiteMinder, Currentversion, Ds, LDAPProvider.
3. Locate the following registry key:  
`EnableSorting`
4. Right-click in the right pane, and then pick New, DWORD.  
A new DWORD named NewValue #1 appears in the list.
5. Right-click the new DWORD, and then do the following steps:
  - a. Pick Rename.
  - b. Type the following text:  
`EnableSorting`
6. Right-click the EnableSorting DWORD, and then pick Modify.  
The Edit DWORD dialog appears. The Value data: field is selected.
7. Do *one* of the following tasks:
  - To disable sorting, type 0.
  - To enable sorting for unknown directories, type 1.
  - To enable sorting for all directories, type 2.
8. Click OK.
9. Close the registry editor.  
The value of the EnableSorting registry key is changed.

## Restart Your Policy Server

For your new settings to take effect restart your Policy Server using the Management Console. You do *not* need to restart the computer on which the Policy Server runs.

**Follow these steps:**

1. Click Start, All Programs, CA, SiteMinder, SiteMinder Policy Server Management Console.

The console opens with the Status tab selected.

2. Click the following Stop buttons.

- Policy Server
- OneView Monitor Service.

3. The Policy Server stops.

4. Click the following Start buttons:

- Policy Server
- OneView Monitor Service.

The Policy Server starts.

5. Click OK.

The Policy Server is restarted.

## Non-english Input Characters Contain Junk Characters

**Symptom:**

When I install or configure SiteMinder components in the console mode on UNIX machines, few non-English input characters are not displayed correctly in the console window.

**Solution:**

Verify terminal settings of your console window and confirm that the console does not clear high (8th) bit of input characters by executing the following command:

```
stty -istrip
```

# Appendix A: SessionLinker Reference

---

This section contains the following topics:

[How the SessionLinker Works](#) (see page 365)

[Working with Cookies](#) (see page 367)

[Troubleshooting](#) (see page 369)

## How the SessionLinker Works

The SessionLinker synchronizes a SiteMinder session with a third-party application session for better security. If a user logs out of SiteMinder, the SessionLinker invalidates the related session of the third-party application.

When a user authenticates, SiteMinder assigns a unique session identifier to that user session. The session identifier, called the SiteMinder Session ID, remains constant for that user for the life of the user session. If the user logs out of SiteMinder through the Logout URL, SiteMinder deletes the SMSESSION cookie that SiteMinder uses to track the SiteMinder Session ID.

The SessionLinker module takes application session cookies and associates them, one by one, with a SiteMinder session. Once associated, the application cookie (referred to here as the foreign cookie) can only be used in conjunction with that particular SiteMinder session. The SessionLinker prevents attempts by other SiteMinder sessions to use the same foreign session.

To understand the SessionLinker operation, associate the SiteMinder session and corresponding foreign cookies that SiteMinder tracks together in a table, as shown in the following example:

SiteMinder Session ID	Foreign Cookie
ONE	ABCD
TWO	LMNO
THREE	PQRST
FOUR	VWXY

The SessionLinker uses the following process:

1. The SessionLinker receives a request from a web server.
2. The SessionLinker extracts the SiteMinder Session ID from the HTTP headers and the Foreign Cookie from all the incoming HTTP cookies.
3. The SessionLinker compares the values that are presented from the web server against the contents of the table to determine whether the request must be allowed, as shown in the following examples:
  - a. If the Session ID is FIVE and the Foreign Cookie is RSTU, SessionLinker inserts these values into the table.
  - b. If the Session ID is SIX and the Foreign Cookie is ABCD, SessionLinker blocks the request because the Foreign Cookie ABCD is already associated with Session ONE.
  - c. If the Session ID is ONE and the Foreign Cookie is HIJK, the old session is orphaned and SessionLinker updates the table to associate Session ID ONE with HIJK. When a session is orphaned, the Foreign Cookie can no longer be presented by anyone. This feature allows the SessionLinker to support applications that update the cookie during the user session.

The entire process is repeated for each Foreign Cookie. The resulting table may appear as follows:

SiteMinder Session ID	Foreign Cookie
***Orphaned***	ABCD
ONE	HIJK
TWO	LMNO
THREE	PQRST
FOUR	VWXY
FIVE	RSTU

## What the SessionLinker Does Not Support

The SessionLinker does *not* do any of the following tasks:

- Track cookies issued to the user throughout the CA SiteMinder® environment. Doing so would require a persistent data store that could be read from and written to by every web server employing SessionLinker. The massive number of reads and writes necessary to support this tracking would require substantial processing power and bandwidth, and is thus unmanageable.

- Destroy the cookies of an existing user when the user logs out of CA SiteMinder®. Because the cookies are not being tracked centrally, no mechanism knows which cookies to destroy. In addition, because of the way different web browsers handle cookies, the logout page cannot always determine which cookies the user has received. Finally, SessionLinker does not actually integrate with the CA SiteMinder® logout process.
- Terminate the session of an underlying application. To support this function, the SessionLinker would need to know how to terminate sessions in each of the applications – many of which do not have an exposed API to manage sessions. Because applications can be configured to terminate sessions after some amount of idle time, and there is little the overhead in leaving a session active, this function has not been implemented.

SessionLinker accomplishes the linking by preventing the user from presenting an invalid Foreign Session cookie.

## Working with Cookies

### Single Session Cookie Enforcement

In most cases, an application has a specific name that is always used for an associated session cookie. In other cases, the name of the cookie begins with a known string, such as ASPSESSIONID or MYAPPSSESSION, and ends with a random or unpredictable suffix. In such cases, the SessionLinker prevents users from presenting more than one of these cookies and enforces the expected session linking.

If the SessionLinker detects multiple potential session cookies, it performs the following steps:

1. Blocks access to sessions
2. Destroys all the cookies
3. Redirects the user to a URL that you specify. If you do not specify a URL, the internal server error is displayed.

## Enable Wildcard Cookie Names

You can add the following parameters of the ACO configured on the Policy Server to the configuration settings already selected:

### COOKIE

Specifies that a cookies beginning with the specified name must be considered as a potential foreign session cookie. The cookie value may end in an asterisk (\*). If you specify a cookie value other than a wildcard syntax, you must specify COOKIEPATH and COOKIEDOMAIN values that determine how to destroy the incoming cookies.

### COOKIEPATH

Specifies the cookie path. If you specified a wildcard syntax for the COOKIE parameter, do not specify this parameter. The COOKIEPATH value depends on the session cookie, and has the following format:

`COOKIEPATH=<Path for outbound cookies or cookies>`

**Default Value:** /

**Example:** COOKIEPATH=/

### COOKIEDOMAIN

Specifies the cookie domain. If you specified a wildcard syntax for the COOKIE parameter, you can specify this value in the following format:

`COOKIEDOMAIN=<domain name for outbound cookie or cookies>`

**Default Value:** Blank

**Example:** COOKIEDOMAIN=.ca.com

## Maintain Links to Multiple Cookies

Some web applications use more than one cookie simultaneously within the same area of the site. You can configure the SessionLinker to maintain links from a single CA SiteMinder® session to a number of cookies. A maximum of ten foreign session cookies can be linked to a single CA SiteMinder® session.

### Follow these steps:

1. Determine the correct configuration string for each cookie.

**Note:** Each configuration string requires at least a COOKIE directive, but any of the directives can be combined.

2. Assign an integer from 0 through 9 to each cookie.
3. Append the selected number to the directive name.

**Note:** You can use any number for each set of directives but the settings for a single cookie require the same number.



4. Concatenate the separate configuration strings into a single string.

## Troubleshooting

If an error occurs, consider the following possibilities to troubleshoot the error:

- Verify that the valid SMSESSION cookie and FOREIGN SESSION cookie are set at the user and are passed to the SPS.
- If you enabled the SessionLinker using the webagent.conf file, verify that the web agent is enabled.
- Verify that the SessionLinker ACO syntax is correct.
- If agent tracing is enabled in the SessionLinker ACO, verify the logs and trace messages in the agent logs and trace.
- Verify that the SPS loaded the SessionLinker plug-in binary properly. Check the agents.log file for log messages. If there are any errors, check if any dependent libraries exist for the SessionLinker plug-in library on the SPS.
- If a request is rejected, verify that the session identifiers on CA SiteMinder® Policy Server (SMSESSION) and application web server (FOREIGN SESSION) are linked are the same user.



# Chapter 15: Upgrade Your CA SiteMinder Agent for SharePoint

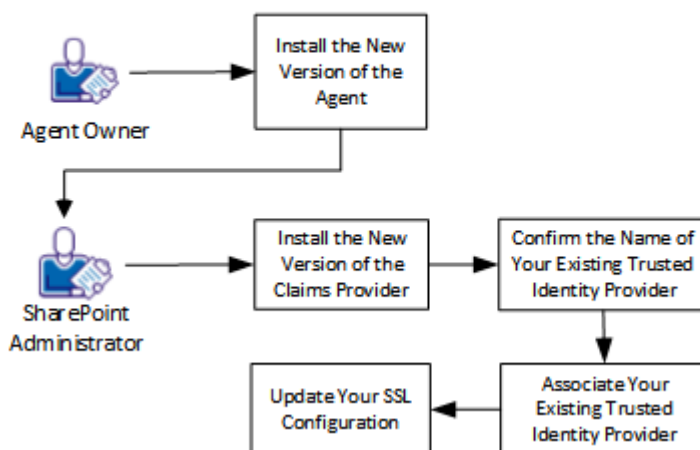
---

## How to Upgrade Your CA SiteMinder Agent for SharePoint

Upgrading your CA SiteMinder Agent for SharePoint from a previous version to 12.52 involves installing the new version of the software on your existing components.

**Note:** The upgraded product uses any existing SharePoint connection and the trusted identity provider from your previous version. You do *not* have to re-create these items. Associate the upgraded claims provider with your existing trusted identity provider.

### How to Upgrade Your CA SiteMinder Agent for SharePoint



**Follow these steps:**

1. Install the new version of the agent using *one* of the following procedures:
  - [Upgrade the agent on Windows operating environments](#) (see page 372).
  - [Upgrade the agent on UNIX operating environments](#) (see page 373).
2. [Install the new version of the claims provider on your SharePoint central administration servers](#) (see page 374).
3. [Confirm the name of your existing trusted identity provider](#) (see page 375).
4. [Associate your existing trusted identity provider with the new claims provider](#) (see page 375).
5. Update your SSL configuration.
6. Repeat Steps 1 through 5 for any other components running the CA SiteMinder Agent for SharePoint.

## Install the New Version of the Agent

The first step of the upgrade process is installing the new version of the Agent. Pick the appropriate procedure for your operating environment from the following list:

- [Upgrade the agent on Windows operating environments](#) (see page 372).
- [Upgrade the agent on UNIX operating environments](#) (see page 373).

## Upgrade the agent on Windows Operating Environments

Installing a new version of the agent upgrades the product from the previous version.

The default installation location for the agent on 32-bit Windows operating environments is: C:\Program Files\CA\Agent-for-SharePoint. On 64-bit Windows operating environments, the default installation location is C:\CA\Agent-for-SharePoint.

**Important!** The CA SiteMinder Agent for SharePoint cannot be installed on a computer that hosts any other web server. The CA SiteMinder Agent for SharePoint operates as a stand-alone proxy-based solution.

To run the agent installer on Windows operating environments, you need local Administrator privileges.

**Note:** We recommend installing the agent on an NTFS file-system partition.

**Follow these steps:**

1. Copy the installation program from the Download location on the CA Support site.
2. Right-click the following executable, and then select Run as administrator:

`ca-sp2010agent-version-operating_environment.exe`

The installation program starts.

3. Follow the instructions from the installation wizard.

**Note:** The installer displays all Java executables that are installed in the system. Pick a Java component and version that is equal to or greater than the one shown by the installer. If the installer does not detect any Java executables by default, then browse and select the appropriate path. For more information about the required Java executables or other third-party software requirements, see the platform support matrix.

4. Restart your system after the installation finishes.
5. [Continue with the next step of installing the new version of the claims provider](#) (see page 374).

## Upgrade the Agent on UNIX Operating Environments

Installing a new version of the agent upgrades the product from the previous version.

The default installation location is `user_home/CA/Agent-for-SharePoint`. The folder where you install the agent requires sufficient permissions (755). Do *not* install the agent under the `/root` folder, because its default permissions (750) are insufficient.

**Important!** The CA SiteMinder Agent for SharePoint cannot be installed on a computer that hosts any other web server. The CA SiteMinder Agent for SharePoint operates as a stand-alone proxy-based solution.

**Note:** On the Solaris or Linux operating environments, the agent runs under the "nobody" user account. If you prefer not to run the agent under this user account, create an alternate user and assign the necessary permissions. Do *not* run this program as a root user.

**Follow these steps:**

1. Copy the appropriate file for your operating environment from the download location on the CA Support site to a temporary directory:
  - Solaris operating environment: `ca-sp2010agent-version-sol.bin`
  - Linux operating environment: `ca-sp2010agent-version-linux.exe`

2. Enter the appropriate command for your operating environment from the following list:

- Solaris: `sh ./ca-sp2010agent-version-sol.bin`
- Linux: `sh ./ca-sp2010agent-version-linux.exe`

3. Follow the prompts that the installation wizard provides.

**Note:** The installer displays all Java executables that are installed in the system. Pick a Java component and version that is equal to or greater than the one shown by the installer. If the installer does not detect any Java executables by default, then browse and select the appropriate path. For more information about the required Java executables or other third-party software requirements, see the platform support matrix.

4. [Continue with the next step of installing the new version of the claims provider](#) (see page 374).

## Install the New Version of the Claims Provider on Your SharePoint Central Administration Servers

The next step of the upgrade process is installing a new version of the claims provider on your SharePoint central administration server. This installation upgrades the claims provider from the previous version.

If you are not the user who installed or configured SharePoint, you need *one* of the following privileges to run the Claims Provider installer:

- Administrator for the local server
- Administrator for the group
- Farm Administrator (for any SharePoint farms)

If you are installing your Claims provider on a new SharePoint farm, install the claims provider on your SharePoint central administration server. If you add any additional SharePoint servers to your farm later, install the claims provider on each SharePoint server you add.

### Follow these steps:

1. Log on to your SharePoint central administration server.
2. Copy the installation program from the download location on the CA Support site.
3. Locate the following executable:

`ca-spclaims-version-win64.exe`

4. Right-click the executable, and then select Run as administrator.

The installation program starts.

5. Follow the installation wizard.
6. Restart your system after the installation finishes.

The Claims provider is successfully installed.

## Confirm the Name of Your Existing Trusted Identity Provider

Confirm the name of your existing trusted identity provider before associating it with your upgraded claims provider.

### Follow these steps:

1. Log on to the computer hosting your SharePoint central administration server.
2. Click Start, All Programs, Microsoft SharePoint 2010 Products, the SharePoint 2010 Management Shell.

3. Enter the following command;

```
Get-SPTrustedIdentityTokenIssuer
```

The name of your existing trusted identity provider appears. This provider is the one which you want to associate with your upgraded claims provider.

4. Continue with the next step of associating your existing trusted identity provider with the [upgraded claims provider](#) (see page 375).

## Associate Your Existing Trusted Identity Provider with the Upgraded Claims Provider

The next step of upgrading is associating the trusted identity provider with the upgraded claims provider.

**Note:** The upgraded product uses any existing SharePoint connection and the trusted identity provider from your previous version. You do *not* have to re-create these items. Associate the upgraded claims provider with your existing trusted identity provider.

The Update-SMTrustedIdentityTokenIssuer command updates the claims provider of a trusted identity token issuer to CASiteMinderClaimProvider.

### Follow these steps:

1. Click Start, All Programs, Microsoft SharePoint 2010 Products, the SharePoint 2010 Management Shell.

The SharePoint 2010 Management Shell command prompt appears.

2. Navigate to the following directory:

C:\Program Files\CA\SharePointClaimsProvider\scripts

3. Enter the update command. This command has the following format:

```
Update-SMTrustedIdentityTokenIssuer.ps1 -TrustedIdentityTokenIssuer
"Name_of_Trusted_Identity_Provider_registered_with_SharePoint"
```

**TrustedIdentityTokenIssuer**

Specifies the name of the CA SiteMinder® trusted identity token issuer (trusted login provider) to update.

**Example:**

```
.\Update-SMTrustedIdentityTokenIssuer.ps1 -TrustedIdentityTokenIssuer
"SiteMinder Federation"
```

The SharePoint central administration server is updated with the new claims provider of the trusted identity token issuer.

## Update the spsapachessl.properties File

Update the spsapachessl.properties file after upgrading the agent.

Do one of the following procedures, as appropriate:

- [Generate an spsapachessl.properties file for an unencrypted private key on Windows](#) (see page 377)
- [Update the spsapachessl.properties file for an unencrypted private key on UNIX](#) (see page 377)
- [Generate an spsapachessl.properties file for an encrypted private key](#) (see page 378)



## Generate an `spsapachessl.properties` File for an Unencrypted Private Key on Windows

Generate an updated `spsapachessl.properties` file for an unencrypted private key on Windows.

### Follow these steps:

1. Open a command-line window with administrative privileges. Navigate to the following directory:

*Agent-for-SharePoint\_home\httpd\bin*

2. Run the following command:

`configssl.bat -enable`

**Note:** If an overwrite warning appears, confirm that you want to overwrite the existing `spsapachessl.properties` file.

The `spsapachessl.properties` file is generated.

## Update the `spsapachessl.properties` File for an Unencrypted Private Key on UNIX

Update the existing `spsapachessl.properties` file for an unencrypted private key after upgrading the Agent on UNIX. The `spsapachessl.properties` is located in the following location:

*Agent-for-Sharepoint\_home/httpd/conf/spsapachessl.properties*

### Follow these steps:

1. Open the `spsapachessl.properties` file in a text editor.
2. Search for the following line:  
`apache.ssl.enabled=`
3. Do one of the following tasks:
  - If the previous line does not exist, add it to the file. Then go to step 4.
  - If the previous line exists, go to Step 4.
4. Confirm that the value after the equal sign matches the setting that you want. Use the following examples:  
`apache.ssl.enabled=Y`  
`apache.ssl.enabled=N`
5. For example, if you were using SSL before your upgrade, verify that the value after the equal sign is Y.
6. Save the changes to the `spsapachessl.properties` file and close the text editor.

## Generate an `spsapachessl.properties` File for an Encrypted Private Key

Generate an updated `spsapachessl.properties` file for an encrypted private key.

### Follow these steps:

1. Open a command-line window with administrative privileges.  
Navigate to the following directory:
  - (Windows) *Agent-for-SharePoint\_home\httpd\bin*
  - (UNIX) *Agent-for-SharePoint\_home/httpd/bin*
2. Run one of the following script files:
  - (Windows) `configssl.bat -enable passphrase`
  - (UNIX) `configssl.sh passphrase`

**Note:** If an overwrite warning appears, confirm that you want to overwrite the existing `spsapachessl.properties` file.

The `spsapachessl.properties` file is generated.

# Appendix B: Proxy Rule Configuration Reference

---

This section contains the following topics:

[Proxy Rules Overview](#) (see page 379)  
[Establish a Proxy Rules Configuration File](#) (see page 383)  
[Proxy Rules DTD](#) (see page 384)  
[How nete:xprcond Elements Works](#) (see page 395)  
[Regular Expression Syntax](#) (see page 396)  
[Header Values in Forwards, Redirects, and Results Filters](#) (see page 400)  
[Response Handling](#) (see page 401)  
[Modify Proxy Rules](#) (see page 402)  
[Sample Proxy Rules Configuration Files](#) (see page 402)

## Proxy Rules Overview

The primary purpose of the CA SiteMinder Agent for SharePoint is to route requests to the appropriate destination servers in the enterprise. The CA SiteMinder Agent for SharePoint routes requests using the proxy engine, which is built into the server. The proxy engine interprets proxy rules and provides both a forward and a redirect service to handle the disposition of all user requests for back end resources.

Proxy rules define the details of how the CA SiteMinder Agent for SharePoint forwards or redirects requests to resources located on destination servers within the enterprise. A set of proxy rules is defined in an XML configuration file according to the proxy rules DTD, which is installed with the SPS.

**Note:** The proxyrules.xml file contains a default rule that forwards requests to `http://www.ca.com$0`, where `$0` appends the entire URI from the original request to the destination, which is `www.ca.com` in this case. You must modify this rule to suit your environment.

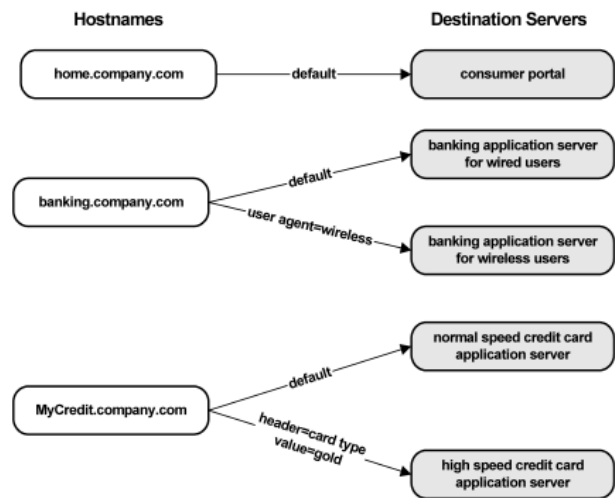
### More information:

[Proxy Rules DTD](#) (see page 384)

## Planning Routes for Incoming Requests

Before you set up the proxyrules.xml file, you should map out the routing for incoming requests. Depending on the virtual host that contains the requested resource, the proxy rules can use a default destination; forward a request based on the user agent type, or uses a HTTP header value to determine the destination server that will fulfill the request. The CA SiteMinder Agent for SharePoint can provide access to a number of virtual hosts.

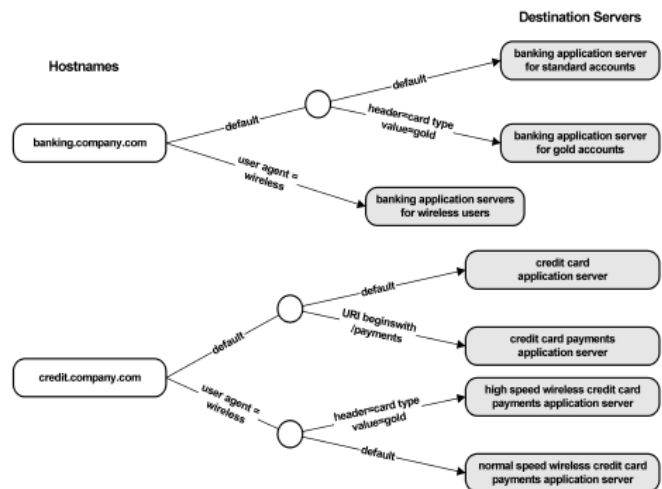
The following illustration shows how proxy rules can be used to route requests to appropriate destination servers.



**Note:** A proxy rules configuration file is processed from top to bottom by the SPS. The first condition that is met by an incoming request determines the behavior of the proxy engine. For example, if a set of proxy rules has two conditions based on a string contained in the URI of a request, and part of the URI of an incoming request matches both of the strings, the first condition listed in the proxy rules file will be used to route the request.

Proxy rules can also be used to control more complex conditions for directing requests to destination servers.

The following illustration shows how proxy rules can be used with a second level of conditions nested within parent conditions.



## Proxy Rules Terminology

The proxy rules configuration file is a description of the XML elements that the CA SiteMinder Agent for SharePoint uses when routing user requests. The following terms are used to describe the proxy rules:

### Destinations

A destination is a URL that fulfills a request. The CA SiteMinder Agent for SharePoint forwards a request to a destination, or sends a redirect response to a user that specifies a destination. A set of proxy rules must contain destinations that can be reached according to the conditions and cases defined in the proxy rules.

### Conditions

A condition is an attribute of a request that allows the CA SiteMinder Agent for SharePoint to determine the destination of a request. Each condition may have many cases that the CA SiteMinder Agent for SharePoint evaluates to direct a request to the appropriate destination. Each condition must contain a default element that defines the behavior if a request does not match any of the cases defined in the condition.

The conditions may include any *one* or *more* of the following:

#### URI

The CA SiteMinder Agent for SharePoint uses the portion of the requested URL after the hostname to determine how to route a request. Using criteria described in the DTD, a portion of a URI, such as the file extension of the requested resource, can be used to route requests.

#### Query String

The CA SiteMinder Agent for SharePoint uses the query string portion of a URI to determine how to route a request. The query string includes all the characters following a '?' in the request.

#### Host

The CA SiteMinder Agent for SharePoint uses the requested server hostname to determine how to route a request. The port number of the hostname can also be used as criteria for routing requests. This condition is used when the proxy has more than one virtual server.

**Header**

The CA SiteMinder Agent for SharePoint uses the value of any HTTP header to determine how to route a request. To route requests based on the type of device being used to access resources, requests may be routed according to the USER\_AGENT HTTP header.

**Note:** HTTP headers derived from SiteMinder responses may be used to determine how to route a request.

**Cookie**

The CA SiteMinder Agent for SharePoint uses the existence or value of a cookie to determine how to route a request. If a cookie value is encoded, specify the encoding scheme in encoding parameter. The CA SiteMinder Agent for SharePoint supports only the base64 encoding scheme.

**Cases**

A case is a set of specific values for conditions that provide the information the CA SiteMinder Agent for SharePoint needs to determine the ultimate destination for a request. For example, if a set of proxy rules uses the host condition, cases include the virtual hosts configured for the system, such as home.company.com and banking.company.com.

## Establish a Proxy Rules Configuration File

The proxy rules configuration file is an XML configuration that is identified by the rules\_file directive of the <ServiceDispatcher> element in the server.conf file. The rules\_file directive indicates the relative path from the installation directory to the proxy rules configuration file. At installation, the relative path to the default proxy rules configuration file is generated automatically and inserted into the rules\_file directive for the default virtual host.

The generated path and proxy rules file name is the following:

*sps\_home/secure-proxy/proxy-engine/conf/proxyrules.xml*

All the entries in the proxyrules.xml file must be well-formed and satisfy the syntax rules as per the XML specifications. Changes to the proxy rules configuration file do not require a server restart to take affect; the CA SiteMinder Agent for SharePoint detects when changes are made to the file and loads the new proxy rules file.

If the CA SiteMinder Agent for SharePoint detects an error in the proxy rules when parsing the rules, the CA SiteMinder Agent for SharePoint records an error in the server log, ignores the changes, and uses the existing proxy rules. The server log file location is specified in the logger.properties file.

## Proxy Rules DTD

You must create the proxyrules.xml file using the proxy rules DTD. To view the proxy rules DTD, go to the following directory:

`sps_home\secure-proxy\proxy-engine\conf\dtd`

The following elements are configurable in the DTD:

- `nete:proxyrules`
- `nete:description`
- `nete:case`
- `nete:cond`
- `nete:default`
- `nete:forward`
- `nete:redirect`
- `nete:local`
- `nete:xprcond`

### **nete:proxyrules**

The full definition for the `nete:proxyrules` element is:

```
<!ELEMENT nete:proxyrules (nete:description?,(nete:cond | nete:forward |
nete:redirect | nete:local)) >
```

This element is the root element for a proxy rules XML configuration file. It contains an optional `nete:description` element and one of the following elements:

- `nete:cond`
- `nete:xprcond`
- `nete:forward`
- `nete:redirect`
- `nete:local`

The `nete:proxyrules` element must be present in a proxy rule configuration file.



## Debug Attribute

The nete:proxyrules element has the following attribute:

```
<ATTLIST nete:proxyrules
 debug (yes|no) "no"
```

This attribute enables or disables logging that aids in debugging proxy rules. The default for this attribute is no. To enable logging, set this attribute to yes.

For example:

```
<nete:proxyrules xmlns:nete="http://www.company.com/" debug="yes">
```

When enabled, trace logging information for the CA SiteMinder Agent for SharePoint is included in trace logs. The location of the log file is determined by the TraceFileName parameter in the agent configuration object that you specified during the CA SiteMinder Agent for SharePoint installation process. The TraceConfigFile parameter in the same agent configuration object must point to the Secure Proxy-specific trace logging configuration file.

By default this file is located in:

```
<install-dir>\proxy-engine\conf\defaultagent\SecureProxyTrace.conf
```

**Note:** This change does not require a CA SiteMinder Agent for SharePoint restart for the changes to take effect.

## nete:description

The full definition for the nete:description element is:

```
<!ELEMENT nete:description (#PCDATA)>
```

This is an optional element that contains a parsed character data (PCDATA) description of another element.

**Note:** PCDATA is any text that is not markup text.

A nete:description element can be a child of the nete:proxyrules element and may contain a description of your choosing.

## nete:case

The nete:case element provides the destination associated with a specific value for a condition defined in the nete:cond parent element. The SPS uses the value of the nete:case element to evaluate a case.

The definition for the nete:case element is:

```
<!ELEMENT nete:case (nete:cond | nete:xprcond | nete:forward | nete:redirect |
nete:local)>
```

All nete:case elements must contain one of the following child elements:

### nete:cond

nete:case elements can contain additional nete:cond elements. This allows you to nest multiple conditions in a set of proxy rules.

### nete:xprcond

nete:case elements can contain additional nete:xprcond elements for regular expression matching of URIs. This allows you to nest a regular expression condition in a set of other conditions.

### nete:forward

Provides a destination to which requests that fulfill the nete:case comparison will be forwarded.

### nete:redirect

Provides a destination to which requests that fulfill the nete:case comparison will be redirected. Once redirected, requests are fulfilled directly by the destination server, rather than through the SPS.

In the following example, a nete:cond element specifies URI matching, and the nete:case elements demonstrate possible uses of the comparison type attribute.

```
<nete:cond type="uri" criteria="beginswith">
 <nete:case value="/hr">
 <nete:forward>http://hr.company.com$0</nete:forward>
 </nete:case>
 <nete:case value="/employee">
 <nete:forward>http://employees.company.com$1
 </nete:forward>
 </nete:case>
</nete:cond>
```

**Note:** The <nete:forward>URL</nete:forward> elements must be located on the same line. In the example, the </nete:forward> closing tags sometimes appear on a separate line due to space constraints, however, a line break in an actual proxy rules file causes an error. The SPS interprets line breaks before the </nete:forward> closing tag as characters that are part of the URL contained in the nete:forward element.

## Forward and Redirect Syntax

When forwarding or redirecting a request, the SPS uses a system for maintaining some part or all of the Universal Resource Indicator (URI) specified by a user. This URI points to a resource that lies on a destination server and must be interpreted by the SPS to fulfill a request.

Either of the following may be appended to a URL specified in a forward or redirect destination:

### \$0

Appends the entire URI string from a user's request to the destination specified in the proxy rule.

For example, if a proxy rule forwards all user requests for `www.company.com` to `proxy.company.com$0` and a user request for `proxy.company.com/employees/hr/index.html`, that request is forwarded to `www.company.com/employees/hr/index.html`.

### \$1

May only be used in `nete:case` elements where the parent `nete:cond` element specifies a URI substring match using the `beginswith` comparison. `$1` indicates that everything to the right of the matching text is appended to the forwarded or redirected request.

For example, consider a proxy rules configuration file that has a `nete:cond` element of:

```
<nete:cond type="uri" criteria="beginswith">
```

Assume this condition is the child of a condition that is evaluating URIs for a hostname of `www.company.com` and a `nete:case` element of:

```
<nete:case value="/hr">
 <nete:forward>http://hr.company.com$1</nete:forward>
</nete:case>
```

If a user requests:

```
http://www.company.com/hr/employees/index.html
```

The request is forwarded to:

```
http://hr.company.com/employees/index.html
```

**Note:** Because this example specifies the `$1` parameter, the `/hr` portion of the URI is omitted when the request is forwarded to `hr.company.com`.

## nete:cond

The definition for the nete:cond element is:

```
<!ELEMENT nete:cond (nete:case+,nete:default)>
```

In addition, the nete:cond element has the following attributes:

```
<!ATTLIST nete:cond type (header | host | uri | query | cookie) #REQUIRED
criteria (equals | beginswith | endswith | contains | exists) "equals"
headername CDATA #IMPLIED>
```

The nete:cond element specifies a condition that will be evaluated to determine how the CA SiteMinder Agent for SharePoint handles incoming requests. This element must include an attribute to be evaluated by the SPS.

Possible attributes, as defined in the ATTLIST element, include:

### header

Specifies an HTTP header. HTTP headers are name-value pairs which can be retrieved from a user directory by SiteMinder. If you select header as the type, you must also specify the name of the header. The following is an example of a nete:cond element using header as the type:

```
<nete:cond type="header" headername="USER_AGENT">
```

This element indicates that a header will be used to determine the destination of a request, and that the header to be evaluated by the CA SiteMinder Agent for SharePoint is USER\_AGENT. The actual destinations for requests are determined by nete:case elements which are children of nete:cond elements as described in the next section.

Conditions that use headers as the comparison require an additional argument of headername="*value*" where *value* is the name of the HTTP or SiteMinder header.

**Note:** HTTP headers generated by SiteMinder responses may be specified in nete:cond elements.

### host

Specifies a host name in deployments where the CA SiteMinder Agent for SharePoint proxies for multiple virtual hosts.

Port numbers are considered part of the host, so you can use the endswith criteria, described later, in conjunction with the host condition to route requests by port number.

**query**

Specifies the query string portion of the URI that follows the '?' character. This is similar to a `nete:cond` that makes use of the URI as follows:

**URI**

Specifies universal resource indicator, which is the portion of a requested URL that follows the server name.

You can use the `endswith` criteria in conjunction with the URI condition to route requests by file extension.

**cookie**

Specifies a cookie attribute to determine how to route a request. If a cookie value is encoded, specify the encoding scheme in encoding parameter. The CA SiteMinder Agent for SharePoint supports only the base64 encoding scheme and does not support cookie creation. The following are the possible cases of an encoded cookie:

- If a cookie is encoded using base64, specify the cookie value in the `value` attribute and `base64` as the encoding parameter in the `nete:case` element. The CA SiteMinder Agent for SharePoint uses the base64 encoding algorithm to decode the cookie value received from a `httprequest` and compares the result the decoded value with the value specified in the `value` attribute.
- If a cookie is not encoded, enter the cookie value in plain text in the `value` attribute and you can specify the encoding parameter as `blank` in the `nete:case` element. The CA SiteMinder Agent for SharePoint accepts the specified plain text as the cookie value and verifies the `nete:cond`.

If a cookie is encoded using a different encoding scheme, enter the encoded cookie value in the `value` attribute and specify the encoding parameter as `blank` in the `nete:case` element. The CA SiteMinder Agent for SharePoint accepts the specified encoded cookie value as plain text, and uses the plain text cookie value to verify the `nete:cond`.

One of the type attributes described above must be included in the `nete:cond` element. In addition, the `nete:cond` element must specify a criteria that defines the comparison that the proxy engine executes on the value of the condition for an incoming request. Possible criteria are:

**equals**

Indicates that the value of the type attribute of the `nete:cond` parent element must equal the contents of the `value` attribute of the `nete:case` element to act on a request.

**beginswith**

Indicates that the value of the type attribute of the `nete:cond` parent element must begin with the contents of the `value` attribute of the `nete:case` element to act on a request.

#### **endswith**

Indicates that the value of the type attribute of the nete:cond parent element must end with the contents of the value attribute of the nete:case element to act on a request.

#### **contains**

Indicates that the value of the type attribute of the nete:cond parent element must contain the contents of the value attribute of the nete:case element to act on a request.

#### **exists**

Indicates that the nete:cond parent element must exist and the value attribute of the nete:case element must be true to act on a request. You can use the exists criteria with only header and cookie attributes.

**Note:** If no criteria are specified, the CA SiteMinder Agent for SharePoint assumes the default criteria of equals.

Each nete:cond element must have one or more nete:case child elements. The nete:case children provide the unique values that the CA SiteMinder Agent for SharePoint uses to route requests to appropriate destinations.

## **nete:default**

The definition of the nete:default element is:

```
<!ELEMENT nete:default (nete:cond | nete:xprcond | nete:forward | nete:redirect | nete:local)>
```

This element is required and must be a child element of each nete:cond element. If a request does not meet the requirements of any nete:case elements contained in nete:cond elements, the nete:default element determines how to handle the request.

The possible child elements associated with the nete:default element are identical to the elements available for a nete:case element. If you create nete:cond elements as children to a nete:default, be careful to take into account a default case so that all possible client requests may be handled by the SPS.

In the following example, the `nete:default` element forwards all requests that do not meet the criteria of any other proxy rules to a home page of general information.

```
<nete:default>
 <nete:forward>http://home.company.com/index.html
</nete:forward>
</nete:default>
```

The opening and closing tags, `<nete:forward>URL</nete:forward>`, elements must be located on the same line. In the example, the `</nete:forward>` closing tags sometimes appear on a separate line due to space constraints, however, a line break in an actual proxy rules file causes an error. The CA SiteMinder Agent for SharePoint interprets line breaks before the `</nete:forward>` closing tag as characters that are part of the URL contained in the `nete:forward` element.

## nete:forward

The definition of the `nete:forward` element is:

```
<!ELEMENT nete:forward (#PCDATA)>
```

The `nete:forward` element forwards a request to a specified URL.

**Note:** The `<nete:forward>` and `</nete:forward>` tags must be located on a single line in the proxy rules file. If they are not located on the same line, the CA SiteMinder Agent for SharePoint interprets line breaks as part of the URL contained in the element. This causes the forward service to fail.

In the following example, the `nete:forward` element forwards requests while maintaining the URI requested by the user.

```
<nete:forward>http://home.company.com$0</nete:forward>
```

If the user's request satisfies the `nete:case` parent element's criteria, that request is forwarded to `home.company.com`. Therefore, a request for `http://server.company.com/hr/benefits/index.html` that is forwarded by the previous `nete:forward` element is fulfilled by forwarding the request to `http://home.company.com/hr/benefits/index.html`.

If you want to forward a request over SSL, be sure to use `https` instead of `http` when defining the destination contained in the `<nete:forward>` element.

The `nete:forward` element contains the following attribute:

```
<!ATTLIST nete:forward filter CDATA #IMPLIED>
```

This attribute allows you to specify the name of a Java filter class that can be invoked during a forward from the CA SiteMinder Agent for SharePoint to a destination server. Filters can be written using the Filter API.

**More information:**

[Forward and Redirect Syntax](#) (see page 387)

The nete:forward element contains the following attribute:

```
<!ATTLIST nete:forward filter CDATA #IMPLIED>
```

This attribute allows you to specify the name of a java filter class that can be invoked during a forward from the CA SiteMinder Agent for SharePoint to a destination server. Filters can be written according to the Filter API.

## nete:redirect

The definition of the nete:redirect element is:

```
<!ELEMENT nete:redirect (#PCDATA)>
```

The nete:redirect element specifies a response that is returned to a user which redirects the user request to an appropriate destination server. The PCDATA follows the standard forward and redirect syntax. Once redirected, the CA SiteMinder Agent for SharePoint does not handle the completion of the request. Instead, the request is handled by the server that is the target of the redirection.

The <nete:redirect> and </nete:redirect> tags must be located on a single line in the proxy rules file. If they are not located on the same line, the CA SiteMinder Agent for SharePoint interprets line breaks as part of the URL contained in the element. This causes the redirect service to fail.

In the following example, the nete:redirect element redirects requests while maintaining the URI requested by the user. Unlike a nete:forward element, once the request has been redirected, the CA SiteMinder Agent for SharePoint is removed from the transaction, and the destination server provides resources directly to the user.

```
<nete:redirect>http://home.company.com$0</nete:redirect>
```

If a user's request for http://www.company.com/hr/index.html meets a parent nete:case element's criteria and is redirected by the above example, the user is redirected and the user's browser displays the URL of the destination server fulfilling the request:

```
http://home.company.com/hr/index.html
```

**Note:** If you want to redirect a request over SSL, be sure to use https instead of http when defining the destination contained in the <nete:redirect> element.



**More information:**

[Forward and Redirect Syntax](#) (see page 387)

## nete:local

The nete:local element is included to support future functionality, and should not be included in proxy rules configuration files.

## nete:xprcond

The nete:xprcond element may be used like a nete:cond element in situations where you want to apply regular expressions as conditions in your proxy rules. Regular expressions can be used to evaluate the URI string and any attached query string in your proxy rules.

The definition of the nete:xprcond element is:

```
<!ELEMENT nete:xprcond (nete:xpr+, nete:xpr-default)>
```

This element must contain one or more nete:xpr elements and a single nete:xpr-default element.

## nete:xpr and nete:xpr-default

A nete:xpr element is similar to a nete:cond element, and contains other elements for a rule based on a regular expression as well as a resulting behavior when the CA SiteMinder Agent for SharePoint finds a match for the expression. A nete:xpr-default element contains the default behavior for URI or query string combinations that do not match any of the regular expressions contained in the nete:xpr elements within a nete:xprcond element.

The definition of a nete:xpr element is:

```
<!ELEMENT nete:xpr (nete:rule, nete:result)>
```

A `nete:xpr` element contains a `nete:rule` element that defines a regular expression, and a `nete:result` element that specifies the behavior for strings that match the regular expression.

The definition of a `nete:xpr-default` element is:

```
<!ELEMENT nete:xpr-default (nete:forward|nete:redirect|nete:local)>
```

The `nete:xpr-default` element specifies a forward or redirect that should be completed if the URI or query string being evaluated by the CA SiteMinder Agent for SharePoint does not match the conditions stated in any of the `nete:xpr` elements contained in the parent `nete:xprcond` element.

## **nete:rule and nete:result**

The `nete:rule` and `nete:result` elements must be contained in a `nete:xpr` element. The `nete:rule` element specifies the regular expression that the SPS evaluates against incoming requests. The `nete:result` element determines the behavior for matching requests.

The definition of the `nete:rule` element is:

```
<!ELEMENT nete:rule (#PCDATA)>
```

This element contains a string that is a regular expression. The URI and query string of a request is matched against this regular expression to determine if a request satisfies the `nete:xpr` condition.

The definition of the `nete:result` element is:

```
<!ELEMENT nete:result (#PCDATA)>
```

`nete:result` elements may have the following attributes:

```
<!ATTLIST nete:result service (forward|redirect) "forward">
```

This element contains a string consisting of the substitution string (URL) by which the SPS creates a URL to pass to a service (forward or redirect). The `service` attribute is used to specify the appropriate service that will receive the URL. The default service is the forward service defined in the `server.conf` configuration file.

The substitution URL in the `nete:result` element should be a URL that optionally contains `$#`, where `#` is 0, 1, 2, etc.

- `$0` is the entire URI and query string being evaluated.
- `$n` is the part of the requested URI that matched the `n`th set of parentheses in the regular expression described in the associated `nete:rule` element.

For example, a set of proxy rules can contain the following:

```
<nete:xprcond>
 <nete:xpr>
 <nete:rule>^/realma(.*)</nete:rule>
 <nete:result>http://server1.company.com$1</nete:result>
 </nete:xpr>

 <nete:xpr-default>
 <nete:forward>http://www.company.com$0</nete:forward>
 </nete:xpr-default>
</nete:xprcond>
```

The `<nete:result>URL</nete:result>` tags must be located on a single line in the proxy rules file. If they are not located on the same line, CA SiteMinder Agent for SharePoint interprets line breaks as part of the URL contained in the element. This causes the result service to fail.

In the previous nete:xprcond proxy rule example, a request for:

`http://www.company.com/realma/index.html`

will be forwarded to:

`http://server1.company.com/index.html`

## How nete:xprcond Elements Works

The processing of a nete:xprcond element is similar to the processing of all other nete:cond elements. As the CA SiteMinder Agent for SharePoint processes requests, and it encounters a nete:xprcond element in the proxy rules configuration file, the following occurs:

1. The CA SiteMinder Agent for SharePoint examines the first nete:xpr element in the nete:xprcond element.
2. The proxy engine evaluates the regular expression described in the nete:rule element against the URI and query string of the request.
3. If the requested URI and query string matches the regular expression specified in the nete:rule element, then the CA SiteMinder Agent for SharePoint resolves the result string using the results of the match, and the request is forwarded to the specified service with the URL derived from the nete:result element.

4. If the requested URI and query string do not match the regular expression in the first `nete:xpr` element, the proxy rules engine evaluates the next `nete:xpr` element (repeat steps 2 and 3). This process continues until the rules engine finds a match or reaches the `nete:xpr-default` element.
5. If no match is found before reaching the `nete:xpr-default` element, then the contents of the `nete:xpr-default` element determine how to route the request.

## Regular Expression Syntax

This section describes the syntax that should be used to construct regular expressions for `nete:rule` elements. A `nete:xprcond` element takes the following form:

```
<nete:xprcond>
 <nete:xpr>
 <nete:rule>regular_expression</nete:rule>
 <nete:result>result</nete:result>
 </nete:xpr>
 <nete:xpr-default>forward_destination</nete:xpr-default>
</nete:xprcond>
```

In the `nete:xpr` element, the `nete:rule` element must consist of a regular expression that uses the syntax described in the following table. This syntax is consistent with the regular expression syntax supported by Apache and described at <http://www.apache.org>.

Characters	Results
unicode character	Matches any identical unicode character
\	Used to quote a meta-character like '*'
\\	Matches a single '\' character
\Onnn	Matches a given octal character
\xhh	Matches a given 8-bit hexadecimal character
\\uhhhh	Matches a given 16-bit hexadecimal character
\t	Matches an ASCII tab character
\n	Matches an ASCII newline character
\r	Matches an ASCII return character
\f	Matches an ASCII form feed character
[abc]	Simple character class
[a-zA-Z]	Character class with ranges

Characters	Results
[^abc]	Negated character class
[:alnum:]	Alphanumeric characters
[:alpha:]	Alphabetic characters
[:blank:]	Space and tab characters
[:cntrl:]	Control characters
[:digit:]	Numeric characters
[:graph:]	Characters that are printable and are also visible (A space is printable, but not visible, while an 'a' is both)
[:lower:]	Lower-case alphabetic characters
[:print:]	Printable characters (characters that are not control characters)
[:punct:]	Punctuation characters (characters that are not letter, digits, control characters, or space characters)
[:space:]	Space characters (such as space, tab, and formfeed)
[:upper:]	Upper-case alphabetic characters
[:xdigit:]	Characters that are hexadecimal digits
[:javastart:]	Start of a Java identifier
[:javapart:]	Part of a Java identifier
.	Matches any character other than newline
\w	Matches a "word" character (alphanumeric plus "_")
\W	Matches a non-word character
\s	Matches a whitespace character
\S	Matches a non-whitespace character
\d	Matches a digit character
\D	Matches a non-digit character
^	Matches only at the beginning of a line
\$	Matches only at the end of a line
\b	Matches only at a word boundary
\B	Matches only at a non-word boundary
A*	Matches A 0 or more times (greedy)
A+	Matches A 1 or more times (greedy)

Characters	Results
A?	Matches A 1 or 0 times (greedy)
/ { /	Matches A exactly n times (greedy)
/ { /	Matches A at least n times (greedy)
/ { /	Matches A at least n but not more than m times (greedy)
A*?	Matches A 0 or more times (reluctant)
A+?	Matches A 1 or more times (reluctant)
A??	Matches A 0 or 1 times (reluctant)
AB	Matches A followed by B
A B	Matches either A or B
(A)	Used for subexpression grouping
\1	Backreference to 1st parenthesized subexpression
\	Backreference to nth parenthesized subexpression

All closure operators (+, \*, ?, {m,n}) are greedy by default, meaning that they match as many elements of the string as possible without causing the overall match to fail. If you want a closure to be reluctant (non-greedy), you can simply follow it with a '?'. A reluctant closure will match as few elements of the string as possible when finding matches. {m,n} closures don't currently support reluctance.

## Regular Expression Examples in nete:rule and nete:result

Regular expressions offer a very flexible and powerful tool that can be employed in CA SiteMinder Agent for SharePoint proxy rules. This section provides a few examples nete:rule elements in proxy rules. In addition, the examples also contain various uses of the nete:result element to show how groupings in a nete:rule can be used to affect the destination generated by the children of a nete:xprcond element.

### Map Single Rule to Many Destination Servers

In the following example, a nete:rule element contains a regular expression that can be used to forward requests to many different destinations. This example assumes that the CA SiteMinder Agent for SharePoint will receive URIs that take the following form:

*/GOTO=some path and or filename*

Consider a nete:xprcond element contains the following child elements:

```
<nete:xpr>
 <nete:rule>/GOTO=(.*)/(.*)</nete:rule>
 <nete:result>http://$1/$2</nete:result>
</nete:xpr>
```

The regular expression in the nete:rule element and the associated nete:result element match URIs that produce a /GOTO=string. Upon finding a match, the CA SiteMinder Agent for SharePoint uses the first string after the = symbol in the URI as the \$1 value of the result, and the value following the first / symbol that appears after the = symbol as the \$2 result. The nete:result element combines these elements to create a URL. By default, nete:result elements use the CA SiteMinder Agent for SharePoint forward service.

For example, if the URI of a request evaluated by the nete:xpr element described above were as follows:

*/GOTO=server1.company.com/index.html*

Then the regular expression in the nete:rule element would find a match and assign the value of \$1 as server1.company.com and the value of \$2 as index.html. The nete:result element assembles these values into the following URL:

*http://server1.company.com/index.html*

This URL is the target which the CA SiteMinder Agent for SharePoint uses to resolve the request.

## Regular Expressions to Redirect Users

The `nete:result` element can also be used to create a redirect response that is returned to the user requesting the resource. This forces the fulfillment of a request to be handled by a server other than the CA SiteMinder Agent for SharePoint after authentication and authorization. The following is an example of a `nete:xpr` element that specifies a redirect in the `nete:result` child element.

```
<nete:xpr>
 <nete:rule>/REDIR=(.*)/(.*)</nete:rule>
 <nete:result service="redirect">http://$1/$2</nete:result>
</nete:xpr>
```

**Note:** The `service` attribute instructs the CA SiteMinder Agent for SharePoint to use the redirect service in place of the default forward service.

## Header Values in Forwards, Redirects, and Results Filters

The value of an HTTP header or a SiteMinder response header can be substituted directly into a `nete:forward`, `nete:redirect`, or `nete:result` element. When a URI in a forward or redirect element, or a rule in a result filter element contains `{{HEADER_NAME}}`, the proxy engine searches for a header in a request that matches the specified header and substitutes the header value before resolving the forward, redirect, or result. If no matching header is found in a request, the proxy engine substitutes an empty string in place of the header value.

**Note:** Header names are case sensitive.

## Dynamic Header Value in a `nete:forward`

To use a dynamic header value as part of a `nete:forward` element, simply insert `{{HEADER_NAME}}` into the URL portion of the forward. For example:

```
<nete:forward>http://www.company.com/{{RESPONSE1}}$1</nete:forward>
```

You can use multiple headers in a single `nete:forward` element. For example:

```
<nete:forward>http://www.company.com/{{RESPONSE1}}/{{RESPONSE2}}$1
</nete:forward>
```



## Dynamic Header Value in a nete:redirect

To use a dynamic header value as part of a nete:redirect element, simply insert `{{HEADER_NAME}}` into the URL portion of the redirect. For example:

```
<nete:redirect>http://www.company.com/{{RESPONSE1}}$1</nete:redirect>
```

You can use multiple headers in a single nete:redirect element. For example:

```
<nete:redirect>http://www.company.com/{{RESPONSE1}}/{{RESPONSE2}}$1</nete:redirect>
```

## Dynamic Header Value in a nete:result

To use a dynamic header value as part of a nete:result element, simply insert `{{HEADER_NAME}}` into the URL portion of the result. For example:

```
<nete:result>http://www.company.com/{{HEADER_NAME}}$1</nete:result>
```

You can use other features of proxy rules, such as filters, in conjunction with a dynamic header value. For example:

```
<nete:result filter="filter1">http://$1/$2{{HEADER_NAME}}</nete:result>
```

# Response Handling

The CA SiteMinder Agent for SharePoint uses SiteMinder responses to determine a destination for a request. Since transactions that are routed through the CA SiteMinder Agent for SharePoint include an interaction between the CA SiteMinder Agent for SharePoint web agent and SiteMinder, any SiteMinder responses gathered during the authentication and authorization process may be used by the CA SiteMinder Agent for SharePoint to determine the destination of a request.

For example, if a user directory contains information about the account type for a banking web site, the CA SiteMinder Agent for SharePoint can proxy users with different types of accounts to different destinations. This enables an enterprise to provide a higher quality of service to its best customers. Customers with standard accounts can be handled by one set of destination servers, while customers with premium accounts can be handled by a separate set of high performance destination servers.

## Modify Proxy Rules

To modify proxy rules you must edit the proxy rules XML configuration file using a text editor. Since proxy rules are XML files, your proxy rules configuration file must be well-formed and valid. Remember that the tags in a well-formed XML file must all consist of opening and closing tags. To be valid, the file must adhere to the guidelines laid out in the proxyrules.dtd.

Changes to the proxy rules XML configuration file are picked up automatically by the SPS. When the CA SiteMinder Agent for SharePoint receives a request, it checks whether or not the proxy rules have changed. If the file has changed, the rules are reloaded before fulfilling the request.

**Note:** If you change the name of the proxy rules XML configuration file in the rules\_file directive in the <ServiceDispatcher> element of the server.conf file, you must restart the CA SiteMinder Agent for SharePoint.

## Sample Proxy Rules Configuration Files

The CA SiteMinder Agent for SharePoint installs several examples of proxy rules configuration files. You can use these example XML files as the basis for your own proxy rules files.

You can find these example files in the directory `sps_home\secure-proxy\proxy-engine\examples\proxyrules`. We recommend you look at the example file as you are reading the descriptions in this guide.

You may copy and customize a file to suit your needs.

### To customize and deploy a proxy rules file

1. Navigate to the directory `sps_home\secure-proxy\proxy-engine\examples\proxyrules`.
2. Make a copy of the example file you want to use.
3. Customize the content and save the new file under a unique name.
4. Copy the modified file to the directory `sps_home/secure-proxy/proxy-engine/conf`.
5. Open up the server.conf file to modify the proxy rules section of the file to point to the customized file.

## Proxy Rules Example—Routing Requests by Virtual Host

The example file `proxyrules_example1.xml` file routes requests based on the hostname specified in the request. The example file `proxyrules_example10.xml` file also routes CA SiteMinder Agent for SharePoint requests based on the hostname specified in the request, CA SiteMinder Agent for SharePoint uses the PID in the proxy rule to count of the number of times the proxy rule has been triggered. If you configured CA Wily Introscope to monitor CA SiteMinder Agent for SharePoint, the count is displayed in CA Wily Introscope data metrics.

In this file, a simple set of proxy rules routes user requests based on the virtual host specified in the requested resource. All requests to the `bondtrading.company.com` server are forwarded to `server2`, all requests to `banking.company.com` are forwarded to `server1`, and all other requests are forwarded to the companies home server, which is the default for requests that do not match the criteria in any other `nete:cond` element.

**Note:** The `nete:case` elements must specify a port, since the port number is considered as part of the virtual host requested by the user. Use the `beginswith` criteria to avoid needing port numbers.

The following table illustrates the results of requests using the proxy rules based on virtual hosts.

Requested URL	Forwarded URL
<code>http://banking.company.com/index.html</code>	<code>http://server1.company.com/index.html</code>
<code>http://bondtrading.company.com/index.html</code>	<code>http://server2.company.com/index.html</code>
<code>http://www.company.com/index.html</code>	<code>http://home.company.com/index.html</code>

## Proxy Rules Example—Routing Requests by Header Value

The example file `proxyrules_example2.xml` file routes CA SiteMinder Agent for SharePoint requests based on the value of an HTTP header. The HTTP header can be a standard header or one created using a SiteMinder response.

**Note:** For information about SiteMinder responses, see the *CA SiteMinder Policy Design*.

In this example, assume that the CA SiteMinder Agent for SharePoint routes requests made to a default virtual host of `www.company.com`.

In this file, the value of the HTTP header variable "HEADER" determines the destination for the request.

The following table illustrates the results of requests using the proxy rules based on an HTTP header.

Requested URL	Forwarded URL
http://www.company.com/index.html HTTP_HEADER has the following value: HTTP_HEADER="value1"	http://server1.company.com/index.html
http://www.company.com/index.html HTTP_HEADER has the following value: HTTP_HEADER="value2"	http://server2.company.com/index.html
http://www.company.com/index.html HTTP_HEADER has a value other than value1 or value2.	http://home.company.com/index.html

**Note:** You do not need to include the HTTP\_ of the header variable name in the nete:cond element. CA SiteMinder Agent for SharePoint assumes HTTP\_ for header variable names.

Proxy rules that use header values are an excellent way to forward requests based on a desired level of service. For example, you can use the value of an HTTP header variable that contains a user account types to distribute requests to high performance servers for customers with premium accounts.

## Proxy Rules Example—Routing Requests by Device Type

The example file proxyrules\_example3.xml file routes CA SiteMinder Agent for SharePoint requests based on the type of device used to access the resource.

**Note:** The user-agent HTTP header value is used to determine how to route requests.

In the file, users who access resources using a browser (user agent contains Mozilla for Web browsers) are forwarded to a Web server, while all other users are forwarded to a wireless server.

The following table illustrates the results of requests using the proxy rules based on a device type.

Requested URL	Forwarded URL
http://www.company.com/index.html User access resource via a Web browser.	http://home.company.com/index.html
http://www.company.com/index.wml User access resource via a wireless device.	http://wireless.company.com/index.wml

## Proxy Rules Example—Routing Requests with URIs

The example file proxyrules\_example4.xml file routes CA SiteMinder Agent for SharePoint requests based on the URI specified in the user request.

The following table illustrates the results of requests using the proxy rules based on URIs.

Requested URL	Forwarded URL
http://www.company.com/dir1/index.html	http://server1.company.com/index.html
http://www.company.com/dir2/index.html	http://server2.company.com/index.html
http://www.company.com/index.html	http://home.company.com/index.html

## Proxy Rules Example—Routing Requests by File Extension

The example file proxyrules\_example5.xml file routes CA SiteMinder Agent for SharePoint requests based on the file extension requested by the user. This is achieved by using the URI condition in combination with the endsuffix criteria.

In the file, the <nete:forward> and </nete:forward> tags appear on separate lines due to space constraints. However, in your proxy rules configuration files, the opening and closing tags for a <nete:forward> element must appear on the same line. If they do not, the CA SiteMinder Agent for SharePoint interprets the line break as part of the forward URL, which causes requests to be forwarded incorrectly.

In the previous example, users who access .jsp resources are forwarded to an application server, while wireless users are forwarded to the wireless server. All other users are forwarded to the home server.

The following table illustrates the results of requests using the proxy rules based on file extensions.

Requested URL	Forwarded URL
http://www.company.com/app.jsp	http://application.company.com/app.jsp
http://www.company.com/index.wml	http://wireless.company.com/index.wml
http://www.company.com/index.html	http://home.company.com/index.html

## Proxy Rules Example—Routing Requests with Nested Conditions

The example file proxyrules\_example6.xml file routes CA SiteMinder Agent for SharePoint requests based on the hostname, certain headers, and device types. This file demonstrates how the CA SiteMinder Agent for SharePoint can handle complex relationships in a single configuration file.

In the file, the `<nete:forward>URL</nete:forward>` elements must be located on the same line. In the example, the `</nete:forward>` closing tags sometimes appear on a separate line due to space constraints, however, a line break in an actual proxy rules file causes an error. The CA SiteMinder Agent for SharePoint interprets line breaks before the `</nete:forward>` closing tag as characters that are part of the URL contained in the `nete:forward` element.

The following table illustrates the results of requests using proxy rules with nested conditions.

Requested URL	Forwarded URL
http://banking.company.com/index.wml	http://wireless.company.com/banking/index.wml
http://banking.company.com/index.html	http://server1.company.com/banking/index.html
http://bondtrading.company.com/index.html with a header value of GOLD_USER="yes"	http://fast.company.com/bondtrading/index.html
http://bondtrading.company.com/index.html with a header value of GOLD_USER="no"	http://server2.company.com/bondtrading/index.html

Requested URL	Forwarded URL
http://www.company.com/index.wml with a USER_AGENT header value that contains a wireless device name	http://home.company.com/ wireless/index.wml
http://www.company.com/index.html with a USER_AGENT header value that does not contains a wireless device name	http://home.company.com/index.html

## Proxy Rules Example—Using Regular Expression in Proxy Rules

The example file proxyrules\_example7.xml file routes CA SiteMinder Agent for SharePoint requests based on nete:xprcond elements that contain regular expressions. Regular expressions are evaluated based on the URI and query string of a request.

In the file, the URI and query string of the request are evaluated against the three regular expressions defined in the nete:xpr elements. If a match is not found against the first nete:xpr element, the CA SiteMinder Agent for SharePoint tries to match it against the second, and finally the third regular expression. If no matches are found, the nete:xpr-default condition is used to handle the request.

The following table lists the results of requests using the regular expression proxy rules.

Requested URL	Forwarded URL
http://server.company.com/realma/hr/index.html	http://server1.company.com/hr/index.html
http://server.company.com/GOTO=server2.company.com/index.html	http://server2.company.com/index.html
http://server.company.com/REDIR=server2.company.com/index.html	http://server2.company.com/index.html User is redirected so that server2.company.com must directly fulfill the user's request.
http://server.company.com/index.html	http://www.company.com/index.html

## Proxy Rules Example—Routing Requests by Cookie Existence

The example file `proxyrules_example8.xml` file routes CA SiteMinder Agent for SharePoint requests based on the existence of a cookie.

In this example, if a request contains a cookie header "mycookie", CA SiteMinder Agent for SharePoint routes the request to `www.company.com`.

## Proxy Rules Example—Routing Requests by Cookie Value

The example file `proxyrules_example9.xml` file routes CA SiteMinder Agent for SharePoint requests based on the value of a cookie.

In this example, if a request contains a cookie header "mycookie" and the request does not specify encoding mechanism, CA SiteMinder Agent for SharePoint routes the request to `www.abcd.com`. If a request contains a cookie header "mycookie" and the base64 encoding mechanism, CA SiteMinder Agent for SharePoint routes the request to `www.xyz.com`.



# Chapter 16: Remove CA SiteMinder® Agent for SharePoint

---

This section contains the following topics:

[How to Remove the CA SiteMinder® Agent for SharePoint](#) (see page 409)

## How to Remove the CA SiteMinder® Agent for SharePoint

To remove the CA SiteMinder® Agent for SharePoint, complete the following procedures:

1. [Remove the Claims Provider from SharePoint](#) (see page 409).
2. [Run the SharePoint Connection wizard to delete your SharePoint Connection](#) (see page 410).
3. Perform the following steps on your SharePoint central administration server:
  - a. [Remove the trusted identity provider from any web applications using it](#) (see page 412).
  - b. [Remove the Trusted Identity Provider from SharePoint](#) (see page 413).
4. Remove the Agent for SharePoint.
5. [\(Optional\) Remove Policy Server Objects from the Policy Store](#) (see page 414).

## Remove Claims Provider

You can remove the Claims Provider from the computer hosting SharePoint Central Administrative by completing the following procedure.

### Follow these steps:

1. Select Start, Control Panel, Programs, Uninstall a program.  
The Uninstall or change a program page appears.
2. Select CA CA SiteMinder® Claims Provider for SharePoint.
3. Click Uninstall.
4. Read the confirmation information and click Uninstall.
5. Click Done.

The Claims Provider is removed from your system.

## Delete a SharePoint Connection

### Follow these steps:

1. Perform the following:
  - (Windows)
    - a. Navigate to the following directory:  
*Agent-for-SharePoint\_home/sharepoint\_connection\_wizard*
    - b. Right-click the executable and select Run as administrator.  
The SharePoint Connection wizard starts.
  - (Unix)
    - a. Navigate to the following directory:  
*Agent-for-SharePoint\_home/sharepoint\_connection\_wizard*
    - b. Enter one of the following commands:
      - Solaris: `sh ./ca-spconnect-12.0-sp3-sol.bin`
      - Linux: `sh ./ca-spconnect-12.0-sp3-rhel30.bin`The SharePoint Connection wizard starts.
2. Click Next.  
The Login Details screen appears.
3. Enter the following login details to connect to the Policy Server.

#### Policy Server Name

Specifies the Policy Server name or IP address.

#### Username

Specifies the Policy Server administrator username.

#### Password

Specifies the Policy Server administrator password.

#### Agent Name

Specifies the Agent-4x. The connection with the Policy Server is established using the details given in the Agent Name.

#### Shared Secret Key

Specifies the shared secret key associated with the Agent.

4. Click Next  
The Select Action screen appears.
5. Select Delete a SharePoint connection option.
6. Click Next.

The Delete from list screen appears.

7. Select the items from the list and click Delete.
8. Click Next.

The Commit details screen appears.

9. Click Install.

The Save complete screen appears.

10. Click Done.

The partnership details are saved, the SharePoint Connection is deleted, and the wizard closes.

**More information:**

[SharePoint Connection Wizard Information Worksheet](#) (see page 418)

## Remove the Trusted Identity Provider from any Web Applications Using it

A trusted identity provider cannot be removed from SharePoint while any web applications are using it. Before you remove the trusted identity provider itself, remove the association between the CA SiteMinder® trusted identity provider and any of your web agents using it.

### Follow these steps:

1. Log in to your SharePoint central administration server.
2. Click Start, Microsoft SharePoint 2010 Products, SharePoint 2010 Central Administration.  
The Central Administration home page opens.
3. Under Application Management, click Manage web applications.  
The web application management page opens.
4. Click the line corresponding to the name of a web application using the CA SiteMinder® trusted identity provider.  
The web application is selected.
5. On the ribbon, click Authentication Providers.  
The Authentication Providers dialog appears.
6. In the Authentication Providers dialog, click the link that corresponds to the zone of your web application. For example, if the web application using the CA SiteMinder® trusted identity provider is in the Intranet zone, click the Intranet link.  
The Edit Authentication page appears.
7. Under Claims Authentication types, clear all Trusted Identity provider check boxes.
8. Click Save.  
The CA SiteMinder® trusted identity provider is removed from the web application in the zone.
9. Repeat Steps 3 through 8 for all web applications and the zones using the CA SiteMinder® trusted identity provider.  
The trusted identity provider is removed from all web applications and their respective zones.

### More information:

[Alternate Connection Wizard Method to Help Resolve Firewall Issues](#) (see page 104)

## Remove Trusted Identity Provider

You can perform the following procedure to remove the trusted identity provider for SharePoint using Windows PowerShell.

### Follow these steps:

1. Select Start, All Programs, Microsoft SharePoint 2010 Products, SharePoint 2010 Management Shell.

The Microsoft PowerShell command prompt appears.

2. Enter following command:

```
Remove-SPTtrustedIdentityTokenIssuer -Identity
```

#### **-Identity**

Specifies the name of the identity provider to remove.

**Example:** Remove-SPTtrustedIdentityTokenIssuer TestSTS

The trusted identity provider for SharePoint is removed.

**Note:** If you re-create a Trusted Identity Provider, verify that a hash precedes the 'New-SPTtrustedRootAuthority' line in the powershell script. As the certificates (signing, root CA, and intermediate CA) are not removed, modify the powershell script by adding hash to avoid certificate errors.

## Remove the Agent for SharePoint from Windows

You can remove the Agent for SharePoint from your Windows system by performing the following procedure.

### Follow these steps:

1. Select Start, Control Panel.
2. Select Programs, Uninstall a program.
3. Select CA SiteMinder® Agent for SharePoint *version*.
4. Click Uninstall/Change.

5. Read the confirmation information and click Uninstall.
6. Click Finish.  
The uninstall confirmation screen appears.
7. Select one of the following options:
  - Yes, restart my system
  - No, I will restart my system myself
8. Click Done.

**Note:** If you have modified any of the Agent for SharePoint files such as server.conf, the uninstall program does not remove these files or their parent folders.

## Remove the Agent for SharePoint from UNIX

Use the following procedure to uninstall Agent for SharePoint from a UNIX system.

**Follow these steps:**

1. Open a console window.
2. Navigate to the root installation directory.
3. Run the following program at the command prompt:  
`./ca-spagent-uninstall.sh`

**Note:** If you have modified any Agent for SharePoint files, such as server.conf, the uninstall program does not remove these files or their parent folders automatically. Remove any files and folders for files you have changed.

## (Optional) Delete Policy Store Objects

If you do not intend to use the Policy Store objects after removing the CA SiteMinder Agent for SharePoint, delete the objects using the CA SiteMinder® Administrative UI.

**Note:** Your administrative privileges determine the objects you can access.

**Follow these steps:**

1. Click <tab>, <Policy Server category>.

**Example:** Click Infrastructure, Authentication.

2. Click <Policy Server object>, Delete <Policy Server object>.

The Delete Object pane opens.

**Example:** Click Authentication Scheme, Delete Authentication Scheme.

The Delete Authentication Scheme pane opens.

3. Specify search criteria, and click Search.

A list of objects that match the search criteria opens.

4. Select an object from the list, and click Select.

A confirmation pane opens.

**Note:** You can select more than one object at a time.

5. Click Yes.

The Delete Object task is submitted for processing.





# Appendix C: Agent for SharePoint Worksheets

---

This section contains the following topics:

[CA SiteMinder Agent for SharePoint Configuration Wizard Information Worksheet](#) (see page 417)

[SharePoint Connection Wizard Information Worksheet](#) (see page 418)

[SharePoint 2010 Federation Worksheet](#) (see page 419)

## CA SiteMinder Agent for SharePoint Configuration Wizard Information Worksheet

Use this worksheet to gather the required information to configure the CA SiteMinder Agent for SharePoint.

Information Required	Your Value
SiteMinder administrator name	
SiteMinder administrator password	
Trusted host name	
Host Configuration Object	
Agent Configuration Object	
IP address of the Policy Server where the host is registered	
Host Configuration File name and location	
Name and location of the Web Agent configuration file	
Email address of the Apache web server administrator	
Fully qualified host name of the server	
Port number for HTTP requests	
Port number for SSL requests	
Port number for HTTP Claims web service	
Port number for SSL Claims web service	

**More information:**[Run the Configuration Wizard](#) (see page 94)

## SharePoint Connection Wizard Information Worksheet

Use this worksheet to gather the required information to configure the SharePoint Connection Wizard.

**Important!** The SharePoint connection wizard automatically creates federation objects (resource partners) in your Policy Servers. Use only the SharePoint connection wizard to create or manage these objects. If you have a Federation Security Services license, these objects also appear in the FSS Administrative UI. Advise your Federation Security Services Administrator not to modify these objects with the FSS Administrative UI unless explicitly told to do so by CA support personnel.

Information Needed	Your Value
Policy Server name	
Policy Server administrator username	
Policy Server administrator password	
Agent-4x name	
Shared Secret Key of the Agent-4x	
Domain associated with the SharePoint connection	
Name of the SharePoint connection	
Authentication URL	
SharePoint Realm Name	
Skew Time	
Validity Duration	
Signing Alias	
Protection level	
Identifier Claim Name	
Directory Attribute	
Attribute	(group-based claims) smusergroups (role-based claims) userrole
Claim Type	

## SharePoint 2010 Federation Worksheet

Use this worksheet to gather the required information to configure SharePoint for CA SiteMinder®.

Information Needed	Your Value
Trusted Identity Provider name	
Certificate authority certificate	
Certificate-Authority Certificates	
Claims Mappings	
Claims Identifier	
Realm	
SignInUrl	
UseWReply	
Name ID	
Account Partner ID	
Signing Certificate	
Security Token Consuming Service	

**More information:**

[Configure SharePoint](#) (see page 113)

## Chapter 17: Platform Support and Installation Media

## Locate the CA SiteMinder® Agent for SharePoint Platform Support Matrix

You can find a comprehensive list of the CA and third-party components supported by CA SiteMinder® on the Technical Support site.

**Follow these steps:**

1. Log in to the [Technical Support site](#).  
The Support home page appears.
2. Under Support, click Support By Product.
3. In the Select a Product Page field, enter CA SiteMinder® and press Enter.  
The CA SiteMinder® product page appears.
4. Scroll to the Product Status section and click CA SiteMinder Family of Products Platform Support Matrices.  
The CA SiteMinder Platform Support Matrices section appears.
5. Under CA SiteMinder Agent for SharePoint, click the PDF link.  
The CA SiteMinder Agent for SharePoint Platform Support Matrix opens in a new tab.

## Locate the Bookshelf

The CA SiteMinder® Agent for SharePoint bookshelf is available on the Technical Support site.

**Follow these steps:**

1. Log in to the [Technical Support site](#).  
The Support home page appears.
2. Under Support, Under Support, click Documentation.
3. In the Select a Bookshelf field select CA SiteMinder® Agent for SharePoint *version* and click Go.  
The CA SiteMinder® Agent for SharePoint bookshelf page appears.

## Locate the Installation Media

You can find a comprehensive list of the CA SiteMinder® installation media on the Technical Support site.

**Follow these steps:**

1. Log in to the [Technical Support site](#).
2. Under Support, click Download Center, Products.  
The Download Center screen appears.
3. Enter CA SiteMinder® Agent for SharePoint in the Select a Product field.
4. Select a release from the Select a Release list.
5. Select a service pack from the Select a Gen Level list.
6. Click Go.

The Product Downloads screen appears. All CA SiteMinder® Agent for SharePoint installation executables are listed.



# Index

---

## (

- (Optional) Create an Agent Group for Multiple Agent Objects • 45
- (Optional) Delete Policy Store Objects • 414

## A

- Accommodate Your SSL Sites by Modifying the Proxy Rules • 234
- Add a Claim to your Trusted Identity Provider • 138
- Add a Policy Server Signing Certificate to Policy Servers and Create a Trust File • 83
- Add a Policy to your Application • 69
- Add a Trusted Root Authority to your SharePoint Farm • 210
- Add Additional Certificate Authority Certificates • 258
- Add Additional Certificate Authority Certificates to the PowerShell Script • 131
- Add an Attribute Mapping for the New Claim • 139
- Add and Grant Permission to CA SiteMinder® Users • 150
- Add Claims Search Web Service • 183
- Add Directory Connections • 247
- Add Public and Internal URLs on your SharePoint server for Path-Based Virtual Hosts • 321
- Add Public and Internal URLs on your SharePoint server for your Host-Header-Based Hosts • 318
- Add Public and Internal URLs on your SharePoint Server for your Port-Based Hosts • 315
- Add Resources to your Application • 67
- Add Roles to your Application • 68
- Add Rules to a Policy • 63
- Add the Certificate Snap-ins • 215
- Add the HTTP Methods for WebDAV to Your Existing Rules • 159
- Add the new certificates • 329
- Add the New Certificates to your CA SiteMinder® Trusted Identity Provider • 310
- Add the Registry Key • 363
- Add the Setting in the Registry File • 360
- Add Users to a Policy • 62
- Add Users to Your Web Applications • 262
- Adding Claims to Trusted Identity Providers • 135
- Additional SharePoint Configuration Options • 153

- Advanced Options • 195
- Agent for SharePoint Prerequisites for Linux Operating Environments • 34
- Agent for SharePoint Virtual Attribute Mappings • 166
- Agent for SharePoint Worksheets • 417
- Alternate Access Mappings • 116
- Alternate Connection Wizard Method to Help Resolve Firewall Issues • 104
- Approve a Certificate Request using Active Directory Certificate Services • 79
- Assign Permissions for Log Files and Directories on UNIX/Linux • 99
- Assign User Directories • 59
- Associate Your Existing Trusted Identity Provider with the Upgraded Claims Provider • 375
- Attributes Appear Truncated in SharePoint • 346

## C

- CA DLP Content Classification Service and the CA SiteMinder Agent for SharePoint • 333
- CA SiteMinder Agent for SharePoint Configuration Wizard Information Worksheet • 417
- CA SiteMinder Agent for SharePoint Prerequisites • 34
- CA SiteMinder® Agent for SharePoint Components and Microsoft SharePoint • 19
- CA SiteMinder® Agent for SharePoint Configuration Overview • 87
- CA SiteMinder® and Microsoft SharePoint • 19
- CA SiteMinder® Components used with SharePoint • 20
- CA Technologies Product References • 3
- Cannot Log Off Users from Sites and Subsites without Referring LogOffURI ACO (CQ 135854) • 352
- Change How Directory Attributes Appear in the SharePoint People Picker • 186
- Change the LogOffURI parameter value • 270
- Change the Sign-in URL of your CA SiteMinder® Trusted Identity Provider • 326
- Change the States of the Services on your Agent for SharePoint • 110, 209, 225, 242, 298
- Change the Value of the EnableWebAgent Parameter • 109, 208, 224, 241, 297

---

- Claims • 26
- Claims Provider • 28, 165
- Claims Provider Searches and Results • 165
- Claims-based Authentication Overview • 25
- Complete Your Certificate Request • 80
- Configure Clusters • 43
- Configure EPAgent Logging Options • 292
- Configure EPAgent Properties • 291
- Configure Realms • 59
- Configure SharePoint • 113
- Configure the Agent for SharePoint for Web Applications That Use NTLM Authentication • 327
- Configure the Authentication Providers • 146
- Configure the JSP file on the system that is running your CA SiteMinder Agent for SharePoint • 266
- Configure Your CA SiteMinder® Policy Server • 39
- Configure Your EPAgent • 291
- Configuring EPAgent Plug-ins • 292
- Configuring the EPAgent for Network Data Sources • 292
- Confirm that the Agent for SharePoint Is Functioning • 96
- Confirm the Name of Your Existing Trusted Identity Provider • 375
- Contact CA Technologies • 5
- Copy the Policy Server Signing certificate to the SharePoint Central Administration Server • 123
- Copy the Powershell Script to the SharePoint Central Administration Server • 124, 251
- Create a 4.x Agent Object for the SharePoint Connection Wizard • 46
- Create a CA SiteMinder® Application to Protect SharePoint Resources that CA DLP also Protects • 64
- Create a Certificate Request for a Server Certificate on an IIS Web Server • 77
- Create a Certificate Signing Request and Submit It to a Certificate Authority • 202
- Create a Host Configuration Object • 42
- Create a New Web Application with Claims based Authentication • 153
- Create a Policy • 61
- Create a Policy Domain • 58
- Create a PowerShell Script to Update the Certificates • 309
- Create a Rule for Web Agent Actions • 60
- Create a SharePoint Connection • 106
- Create a Trusted Store for the Root Certificate Authority Certificate • 222

- Create A User Directory Connection • 51
- Create Alternate Access Mappings for Your Port-Based Virtual Sites • 239
- Create an Agent Configuration Object • 47
- Create an Agent Object • 44
- Create an Attribute Mapping for a Role-based Claims in Active Directory • 180
- Create an Attribute Mapping for a Role-based Claims in LDAP Directories • 179
- Create an Attribute Mapping for User Claims in a Microsoft Active Directory Server • 174
- Create an Attribute Mapping for User Claims in an LDAP Directory • 52, 172
- Create an Authentication Scheme for the Agent for SharePoint • 57
- Create Attribute Mappings for Group-based Claims in Active Directory • 177
- Create Attribute Mappings for Group-based Claims in LDAP Directories • 176
- Create Attribute Mappings for User Claims in a Microsoft Active Directory Server • 53
- Create Attribute Mappings for User Claims in an ODBC Directory • 54
- Create Proxy Rules for your Host-Header-Based Virtual Hosts • 317
- Create Proxy Rules for your Path-based Virtual Hosts • 320
- Create Proxy Rules for your Port-based Virtual Hosts • 314
- Create SharePoint Policies with Placeholders for Expected Directory Attributes • 185
- Create the JCEKS Key Store and Private Key • 200
- Create Virtual Attribute Mappings for your User Claim • 52

## D

- Debug Attribute • 385
- Define the KeyStore and the SSL Ports • 206
- Define Virtual Attribute Mappings • 245
- Define Virtual Hosts for each Web Application • 312
- Delete a SharePoint Connection • 410
- Determine PowerShell Script Modifications • 252
- Determine the IANA Code for Your Language • 190
- Determine your policy model • 57
- Disable Client Loopback • 150, 261
- Documentation Changes • 4
- Download and Import the Certificate Chain • 205



---

Download and Install the Certificates from your Certificate Authority • 233

Dynamic Header Value in a nete  
forward • 400  
redirect • 401  
result • 401

## E

Edit a SharePoint Connection using the SharePoint Connection Wizard • 107  
Edit the File of Each Web Front-End (WFE) Server in Your SharePoint Environment • 267, 280  
Edit the Sign-In URL for the Affiliate Domain using the Sharepoint Connection Wizard • 323  
Enable Paging for Searches of Active Directory User Stores (32-bit systems) • 70, 353  
Enable Paging for Searches of Active Directory User Stores (64-bit systems) • 71, 354  
Enable Search of Custom Object Classes in Your LDAP Directory • 350  
Enable Single Logout by Running the SharePoint Connection Wizard • 275, 286  
Enable SSL for an Encrypted Private Key • 236  
Enable SSL for an Unencrypted Private Key on UNIX • 235  
Enable SSL for an Unencrypted Private Key on Windows • 235  
Enable SSL for the Web Application • 155  
Enable SSL on IIS for the Web Application • 154  
Enable SSL on Your Agent for SharePoint • 235  
Enable Support for Dynamic Policy Server Clusters for the CA SiteMinder Agent for SharePoint • 98  
Enable the DLP Plug-in • 333  
Enable the SessionLinker • 300  
Enable Wildcard Cookie Names • 368  
Environment Variables • 190  
Establish a Proxy Rules Configuration File • 383  
Example Federation and Claims-based Authentication Scenario • 28  
Example SharePoint Farm Deployment with Multiple Web Front Ends and Load Balancing • 22  
Example SharePoint Farm Deployment with Single Web Front End • 21  
Export the Client Authentication Certificate from the Current User Certificate Store • 215  
Export Your Policy Server Signing Certificate • 82  
Extend Web Applications to Different Zones for CRAWL Service and Search Support • 188

## F

Features to Set Up Following Basic Installation and Configuration of the Agent for SharePoint • 153  
Federation and Claims-based Authentication • 25  
FIPS Support Overview • 88  
Forward and Redirect Syntax • 387

## G

Gather CA SiteMinder® Agent for SharePoint Configuration Wizard Information • 92  
Generate a Private Encrypted RSA Server Key for Each Virtual Site • 230  
Generate a Private Key • 228  
Generate a Private Unencrypted RSA Server Key for Each Virtual Site • 229  
Generate an spsapachessl.properties File for an Encrypted Private Key • 378  
Generate an spsapachessl.properties File for an Unencrypted Private Key on Windows • 377  
Generate an SSLConfig.properties File • 207  
Generate and Submit Certificate Signing Requests • 232  
Generate the Certificates by Processing the Request at the Certificate Authority • 204  
Generate the Client Authentication Certificate • 213  
Grant Application Pool Identities for SharePoint Web Applications Permissions to the Client Certificate • 218  
Group Claims • 175

## H

Header Values in Forwards, Redirects, and Results Filters • 400  
How nete  
xprcond Elements Works • 395  
How the SessionLinker Works • 365  
How the SharePoint Connection Wizard Simplifies Deployment • 29  
How to Configure Host-Header-Based Virtual Hosts • 316  
How to Configure Multiple User Directories • 243  
How to Configure Office Client Integration for the Agent for SharePoint • 157  
How to Configure Path-based Virtual Hosts • 319  
How to Configure Port-based Virtual Hosts • 313  
How to Configure SharePoint for the Agent for SharePoint • 113

---

How to Configure Single Logout on SharePoint 2010 • 278

How to Configure SLO for SharePoint 2013 • 263

How to Configure the CA SiteMinder® Agent for SharePoint • 91

How to Configure the Claims Provider • 182

How to Configure the Trusted Identity Provider • 123

How to Configure WebDAV to Accomodate Microsoft Hot Fixes 2563214 and 2647954 • 163

How to Configure your CA SiteMinder® Policy Server • 39

How to Create Alternate Access Mappings • 115

How to Disable Client Loopback • 149

How to Enable SSL for the Agent for SharePoint • 196

How to Migrate the CA SiteMinder® Agent for SharePoint from SharePoint 2007 to SharePoint 2010 • 31

How to Modify the Sign-In URL of your CA SiteMinder® Trusted Identity Provider • 322

How to Monitor Data with CA Introscope • 289

How to Reduce People Picker Timeouts with Large Databases on UNIX/Linux Operating Environments • 359

How to Reduce People Picker Timeouts with Large Databases on Windows Operating Environments • 362

How to Remove the CA SiteMinder® Agent for SharePoint • 409

How to replace the (WS-Fed) token signing certificates • 328

How to Replace the Certificates for your CA SiteMinder® Trusted Identity Provider • 305

How to Request and Install a Policy Server Token Signing Certificate • 73

How to Set Log Files, and Command-line Help to Another Language • 188

How to Start and Stop the Agent for SharePoint • 108

How to Upgrade Your CA SiteMinder Agent for SharePoint • 371

How to Use the Session Linker • 299

## I

I Can Only View Read Only Copies of Documents on SharePoint Sites, but the Office Client Integration Is Enabled • 357

Identify your Trusted Identity Provider • 137, 145, 308, 325

Identity Provider (IdP) • 28

Import the Client Authentication Certificate into the Local Computer Certificate Store • 217

Install and Configure the CA SiteMinder® Agent for SharePoint • 87

Install Claims Provider • 181

Install the CA SiteMinder Agent for SharePoint on Windows • 90

Install the CA SiteMinder® Agent for SharePoint • 89

Install the CA SiteMinder® Agent for SharePoint on UNIX • 90

Install the Client Authentication Certificate on your SharePoint Servers • 218

Install the New Version of the Agent • 372

Install the New Version of the Claims Provider on Your SharePoint Central Administration Servers • 374

Install the Policy Server Signing Certificate on your CA SiteMinder Agent for SharePoint • 99, 330

Introduction • 17

## L

Leave the Clean Up URL Unprotected • 272, 283

Leave the Confirmation Page Unprotected • 274, 285

Leave the Sign-out Service URL Unprotected • 273, 284

Linux Tools Required • 36

Load Balancers and Session Affinity • 23

Locate the Bookshelf • 420

Locate the CA SiteMinder® Agent for SharePoint Platform Support Matrix • 420

Locate the Installation Media • 421

Log Files • 339

Log Files Show Access Denied Because of SPAuthorizeUserAgent Settings • 349

Log Files Show Access Denied Due to BadURLChars Settings • 347

Log Rolling Based on the Age of the File • 344

Log Rolling Based on the File Size • 343

logger.properties File Overview • 339

Logging Settings • 340

## M

Maintain Links to Multiple Cookies • 368

Major Differences between Agent for SharePoint Releases • 18

---

Make Your Sessions Persistent • 271, 282  
Manage SharePoint Connections Using the  
    SharePoint Connection Wizard • 100  
Manage User Profiles • 151  
Map Single Rule to Many Destination Servers • 399  
Microsoft Prerequisites • 37  
Migrating from SharePoint 2007 to SharePoint 2010  
    • 31  
Modify an Existing Classic Authentication to  
    Claims-based Authentication • 147  
Modify Proxy Rules • 402  
Modify the PowerShell Script • 125  
Modify the PowerShell Script for Certificates Issued  
    by a Trusted Certificate Authority • 130  
Modify the PowerShell Script for Certificates Signed  
    by an Un-Trusted External Certificate Authority •  
    126  
Modify the PowerShell Script for Un-Trusted  
    Self-Signed Certificates • 128  
Modify the Script for a Trusted Certificate Authority  
    • 257  
Modify the Script for an Un-Trusted External  
    Certificate Authority • 253  
Modify the Script for an Un-Trusted Self-Signed  
    Certificate • 255  
Modify the server.conf File • 164, 293  
Modify the SSL Configuration File for Your Agent for  
    SharePoint • 226  
Modify Your Authentication Scheme • 240  
Modifying the logger.properties File • 339

## N

nete

- case • 386
- cond • 388
- default • 390
- description • 385
- forward • 391
- local • 393
- proxyrules • 384
- redirect • 392
- rule and nete
  - result • 394
- xpr and nete
  - xpr-default • 393
- xprcond • 393

New Architecture to Support SharePoint 2010 • 18

Non-english Input Characters Contain Junk  
Characters • 364

## O

Obtain the Public and Internal URLs • 119  
Office Client Integration • 156  
Open a SharePoint 2010 Management Shell Window  
    on your SharePoint Central Administration Server  
    • 137, 144, 308, 325  
Open the Administrative UI to Change Policy Server  
    Objects • 42, 158, 245, 269, 281, 293, 301  
Open the sm.registry File with a Text Editor • 359  
Open the Windows Registry Editor • 362

## P

Permissions Required for Trusted Identity Provider  
    and Claims Provider • 114  
Planning Routes for Incoming Requests • 380  
Platform Support and Installation Media • 419  
Policy Server Prerequisites • 33  
Prerequisites • 33  
Prerequisites for Using the SharePoint Connection  
    Wizard • 103  
Provide the Certificate Files to Your SharePoint  
    Administrator • 85  
Provide the Policy Server Signing Certificate Files to  
    Your Agent Owner • 85  
Proxy Rule Configuration Reference • 379  
Proxy Rules DTD • 384  
Proxy Rules Example—Routing Requests by Cookie  
    Existence • 408  
Proxy Rules Example—Routing Requests by Cookie  
    Value • 408  
Proxy Rules Example—Routing Requests by Device  
    Type • 404  
Proxy Rules Example—Routing Requests by File  
    Extension • 405  
Proxy Rules Example—Routing Requests by Header  
    Value • 403  
Proxy Rules Example—Routing Requests by Virtual  
    Host • 403  
Proxy Rules Example—Routing Requests with Nested  
    Conditions • 406  
Proxy Rules Example—Routing Requests with URIs •  
    405  
Proxy Rules Example—Using Regular Expression in  
    Proxy Rules • 407  
Proxy Rules Overview • 379

---

Proxy Rules Terminology • 382

Purpose and Audience • 17

## R

Re-configure the trusted identity provider • 330

Register the Claims Search Service Endpoint on all WFE Servers • 220

Registration Failed with Unknown Error 127 • 358

Regular Expression Examples in nete rule and nete result • 399

Regular Expression Syntax • 396

Regular Expressions to Redirect Users • 400

Remove CA SiteMinder® Agent for SharePoint • 409

Remove Claims Provider • 409

Remove Claims Search Web Service • 187

Remove the Agent for SharePoint from UNIX • 414

Remove the Agent for SharePoint from Windows • 413

Remove the Claim Type from your Trusted Identity Provider • 146

Remove the ClaimsMapping Identity from your Trusted Identity Provider • 145

Remove the expired certificate from your IIS web server • 329

Remove the Trusted Identity Provider • 251

Remove the Trusted Identity Provider from any Web Applications Using it • 412

Remove the Web Applications from the Trusted Identity Provider • 250

Remove Trusted Identity Provider • 413

Removing Claims from Trusted Identity Providers • 143

Replace the Certificates on your Servers • 307

Request a Client Certificate • 211

Request and Install a Policy Server Token Signing Certificate • 73

Required Linux Libraries • 35

Required Linux Patches • 34

Response Handling • 401

REST API in Excel Services Does Not Work Due to CSSChecking ACO Parameter • 351

Restart the Agent for SharePoint • 207, 223, 240, 297

Restart Your Policy Server • 361, 364

Role Claims • 178

Run the Configuration Wizard • 94

Run the Connection Wizard • 236

Run the PowerShell Script • 260

Run the Powershell Script to Create a Trusted Identity Provider • 133

Run the SharePoint Connection Wizard • 248

## S

SAML Autopost Frequency • 104

Sample Proxy Rules Configuration Files • 402

Search for and Add Users using the New Claim • 142

Security Token Service (STS) • 27

Server Log Rolling Settings • 342

Server Log Settings • 341

SessionLinker Reference • 365

Set a Basic Proxy Rule for the Agent for SharePoint • 97

Set the Locale Variable on UNIX or Linux Operating Environments • 193

Set the Locale Variable on Windows Operating Environments • 191

Set the Proxy Rules for the Agent for SharePoint when using CA DLP Content Classification Service with Multiple Authentication • 335

Set the SessionLinker Parameter in a Local Configuration File • 303

Set the SessionLinker Parameter in an Agent Configuration Object • 301

SharePoint 2010 Federation Worksheet • 419

SharePoint Connection Wizard Information Worksheet • 418

SharePoint FedAuth Cookies and Office Client Integration Behavior • 358

Single Session Cookie Enforcement • 367

Specify a Public URL for the Web Application • 121

Specify an Internal URL for the Web Application • 122

Start Your EPAgent on UNIX Operating Environments • 296

Start Your EPAgent on Windows Operating Environments • 296

Submit Your Certificate Request to a Certificate Authority • 78

SvrConsoleAppender Settings • 340

SvrFileAppender Settings • 341

## T

Tokens • 27

Token-Signing Certificate Locations in Your SharePoint Environment • 76

---

Troubleshooting • 345, 369

## U

Update the Affiliate Domain with a Response Attribute • 140  
Update the Agent Type to Include the HTTP Methods for WebDAV • 158  
Update the Claims Provider of the Trusted Identity Token Issuer • 182  
Update the DNS Tables with your Host-Header-Based Virtual Hosts • 316  
Update the DNS Tables with your Path-based Virtual Hosts • 319  
Update the DNS Tables with your Port-based Virtual Hosts • 313  
Update the spsapachessl.properties File • 376  
Update the spsapachessl.properties File for an Unencrypted Private Key on UNIX • 377  
Update the SSLConfig.properties File • 223  
Update the Trusted Identity Token Issuer • 146  
Update Your Agent Configuration Object (ACO) • 295  
Update your Agent Configuration Settings for Office Client Integration • 160  
Upgrade the Agent on UNIX Operating Environments • 373  
Upgrade the agent on Windows Operating Environments • 372  
Upgrade Your CA SiteMinder Agent for SharePoint • 371  
Upgrades to the CA SiteMinder® Agent for SharePoint • 31  
User Claims • 171  
Users Cannot Access Office Applications in Internet Explorer 7 when Office Client Integration Is Enabled • 355

## V

Verify Claims Provider Installation • 181  
Verify SharePoint Installation • 37  
Verify that the Sign-in URL has Changed • 326  
Verify That the Trusted Identity Provider Is Registered • 134  
Verify that your Account has the Required Permissions • 137, 144, 308, 325  
Verify the Locale Variable Value on Windows Operating Environments • 192  
Verify the New Claim Exists • 138  
Verify the Prerequisites • 199

Verify the Server Hosting Your Agent for SharePoint Has the Proper Files • 279

Verify Trusted Identity Provider Registration • 261  
Verify Your Approval and Download Your Certificate and Certificate Chain • 81  
Verify Your Certificate Approval and Install Your Client Authentication Certificate • 214  
Virtual Attribute Mapping Examples for a Microsoft Active Directory Server • 169  
Virtual Attribute Mapping Examples for an LDAP Directory • 167  
Virtual Host Configurations Supported by the Agent for SharePoint • 311  
Virtual Hosts with the Agent for SharePoint • 311

## W

What the SessionLinker Does Not Support • 366  
Working with Cookies • 367

## Z

Zones and Alternate Access Mappings • 117