

SITEMINDER SSO FOR EMC® DOCUMENTUM® REST

ABSTRACT

This white paper provides a detailed review of SiteMinder SSO integration with EMC Documentum REST Services by exploring the architecture, consumption workflow, deployment recommendations and alternatives, and the troubleshooting for this integration.

January, 2014

To learn more about how EMC products, services, and solutions can help solve your business and IT challenges, [contact](#) your local representative or authorized reseller, visit www.emc.com, or explore and compare products in the [EMC Store](#)

Copyright © 2014 EMC Corporation. All Rights Reserved.

EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

The information in this publication is provided “as is.” EMC Corporation makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose.

Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

For the most up-to-date listing of EMC product names, see EMC Corporation Trademarks on EMC.com.

VMware and <insert other VMware marks in alphabetical order; remove sentence if no VMware marks needed. Remove highlight and brackets> are registered trademarks or trademarks of VMware, Inc. in the United States and/or other jurisdictions. All other trademarks used herein are the property of their respective owners.

Part Number H13864

TABLE OF CONTENTS

EXECUTIVE SUMMARY (BODY SUBHEAD LEVEL 1 STYLE)	4
AUDIENCE	4
SITEMINDER ARCHITECTURE	5
SITEMINDER INTEGRATION WITH DOCUMENTUM REST SERVICES	7
Authentication Flow	7
SITEMINDER COMPONENTS INSTALLATION AND CONFIGURATION	9
Installing Microsoft Active Directory Server as Policy Store	10
Install SiteMinder Policy Server 15.5	11
Configuring Active Directory Server as Policy Store	12
Install SiteMinder Administrative UI 12.5	15
Installing Apache-based Web Agent	15
Configure Content Server	20
Configure Rest Service Server	23
Accessing Rest Services through SiteMinder Agent	23
TROUBLESHOOTING	26
Logging	26
APPENDIX	27
Configure ACS/BOCS with SiteMinder	27
Solution 1: change ACS/BOCS "Base URL" configuration	27
Solution 2: URL rewrite	27
CONCLUSION	34
REFERENCES	35

EXECUTIVE SUMMARY (BODY SUBHEAD LEVEL 1 STYLE)

CA SiteMinder Secure SSO & Flexible Access Management can provide your organization with enterprise-class secure single sign-on (SSO) and flexible access management so that your organization can authenticate users and control access to Web applications and portals. Across Internet, intranet and cloud applications, SiteMinder SSO helps enable the secure delivery of essential information and applications to your employees, partners, suppliers and customers via secure single sign-on. It also scales to help you meet your growing business needs with flexible administration tools that can support either centralized or distributed administration.

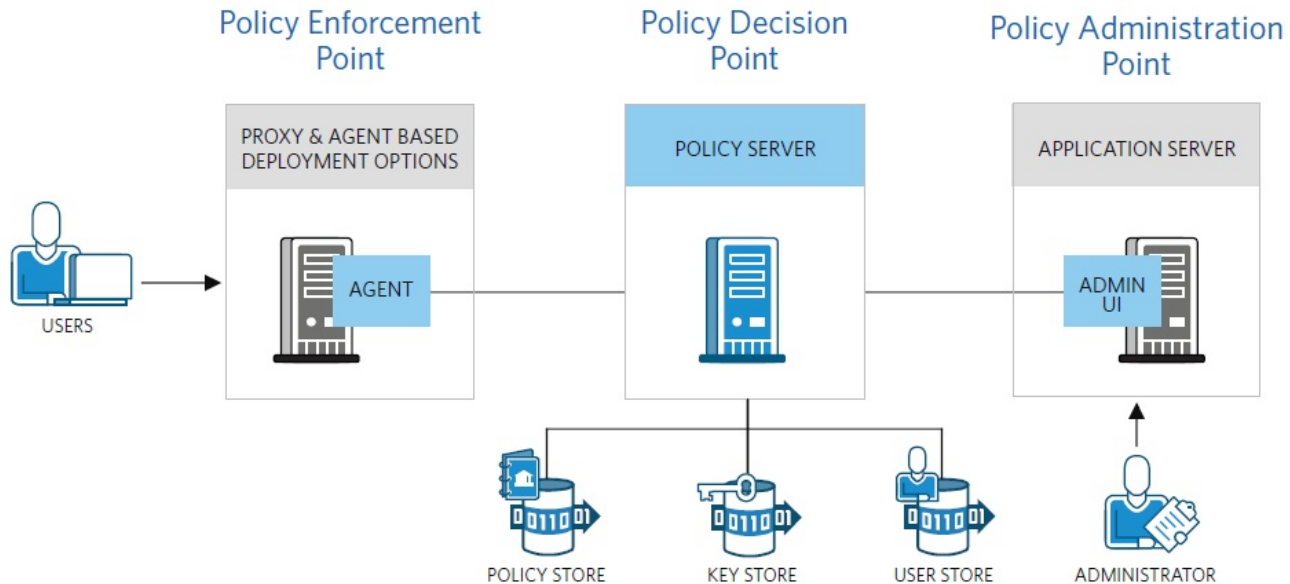
AUDIENCE

This white paper is intended for architects, engineers, support professionals and customers. It provides detailed understanding for enabling SiteMinder SSO for Documentum REST Services. This white paper is intended for architects, engineers, support professionals and customers. It provides detailed understanding for enabling SiteMinder SSO for Documentum REST Services.

SiteMinder Architecture

SiteMinder is a centralized web access management system that enables user authentication and single sign-on, policy-based authorization, identity federation, and auditing of access to Web applications and portals.

Figure 1 SiteMinder SSO architecture



- SiteMinder Policy Server
 - Acts as the Policy Decision Point (PDP). The Policy Server authenticates users on behalf of the PEP, evaluates security policies, and makes authorization decisions that are communicated back to the PEP. The Policy Server also audits each of these events.
- SiteMinder Policy Store
 - An entitlement store that resides in an LDAP directory server or ODBC database. The purpose of this component is to store all policy-related objects.
- SiteMinder Agent
 - Acts as a Policy Enforcement Point (PEP) and also performs the services of authentication management and single sign-on. Agents can also support optional requirements such as securely passing user entitlements to protected business applications.
- SiteMinder Administrative UI

- Serves as a secure Policy Administration Point (PAP). One instance of the Administrative UI server can connect to and manage multiple Policy Servers.

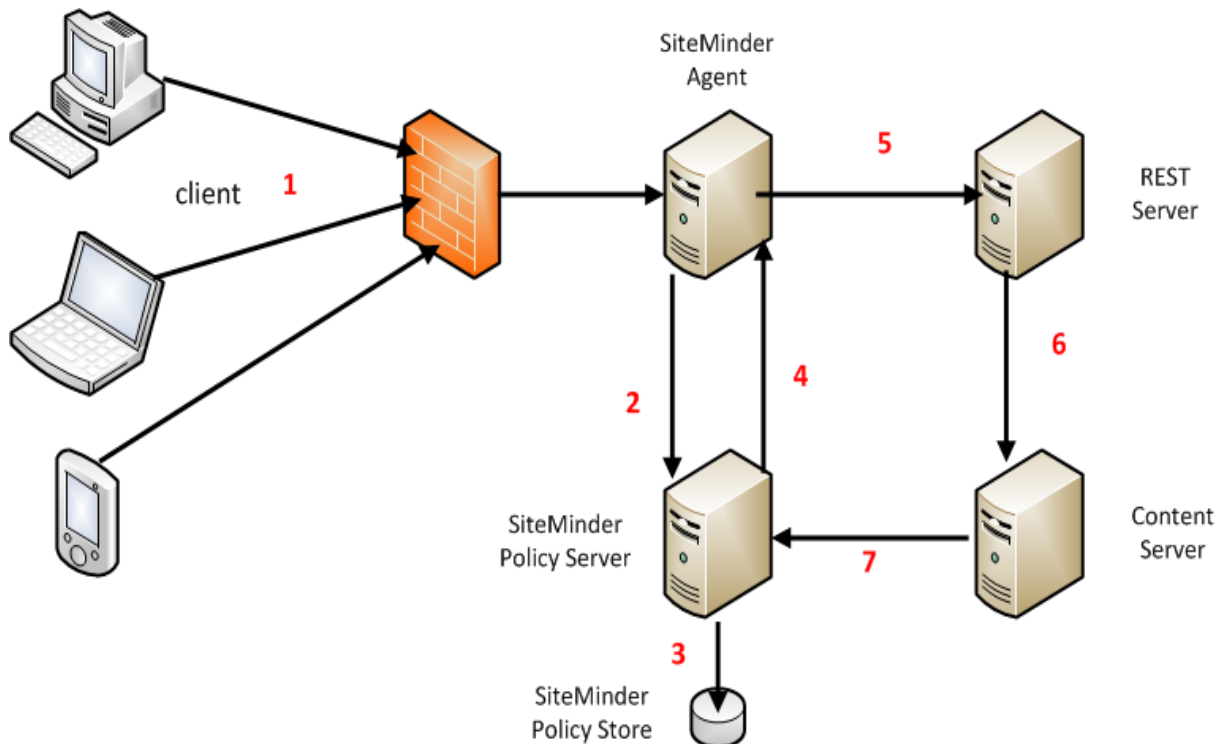
SiteMinder Integration with Documentum REST Services

Documentum REST Services extends the existing Content Server SiteMinder SSO authentication support. The following are required components:

- A user agent, e.g. web browser, generic HTTP client, etc.
- A SiteMinder Policy Server 15.5 with Policy Store
- A SiteMinder Policy Agent 15.5 with Web Server
- A Documentum REST Services server 7.1+
- A Content Server 7.1+

Documentum REST Services is protected by SiteMinder Agent. Any request should be sent to Agent directly.

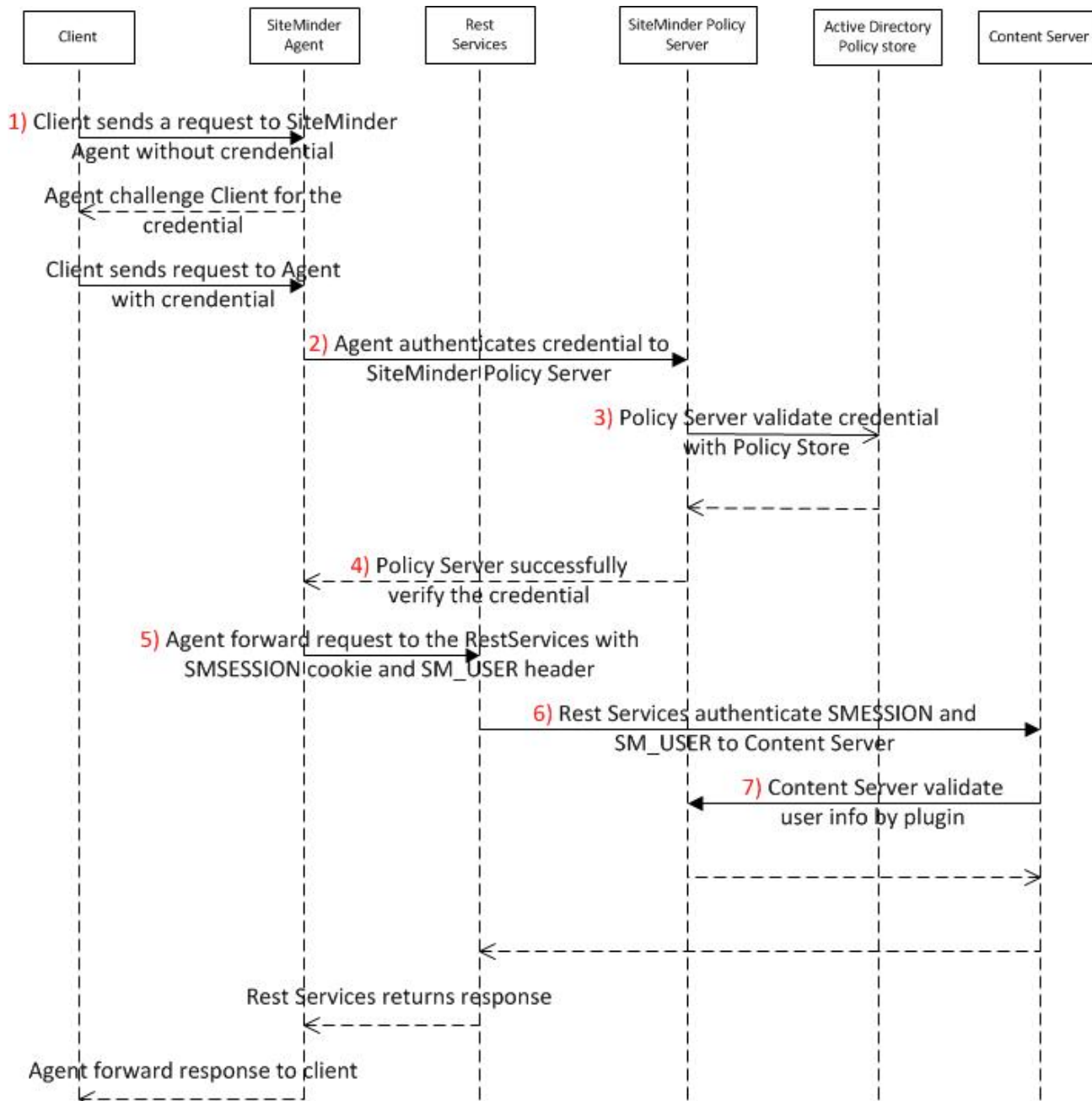
Figure 2 Integration architecture



Authentication Flow

In this section, we go through the authentication flow of the SiteMinder SSO.

Figure 3 Basic Authentication Flow



Step 1):

A client sends a Documentum REST request to the SiteMinder Agent without credential information. The client does not know the actual location of the REST Services Server. The request URI depends on the outer-inner URL mapping in the SiteMinder Agent (a reverse proxy server). A sample URI to Documentum repositories service looks like: <http://agent-host:8080/dctm-rest/repositories>.

The SiteMinder Agent challenges the client for the credential. The actual authentication type varies depending on the SiteMinder configuration, e.g. HTTP Basic authentication.

Then the client sends back with credential information.

Step 2):

The SiteMinder Agent validates the credential to the SiteMinder Policy Server.

Step 3):

The SiteMinder Policy Server queries credential from the Policy Store.

Step 4):

The SiteMinder Policy Server successfully validates the client credential and response to the Agent.

Step 5):

Upon a successful validation, the request will be forwarded to the REST Services. And the SiteMinder Agent will add a request header SM_USER and a cookie SMSESSION to the forwarded request. This header contains the username (distinguished name or short name depending on the configuration).

Step 6):

Before processing request, the REST Services will validate the client to Content Server, with the SM_USER and SMSESSION info.

Step 7):

The Content Server SiteMinder plugin connects to the SiteMinder Policy Server to validate credential passed by the REST Services.

Step 8):

Upon a successful validation, the REST Services will process the request, and send back the response to the SiteMinder Agent. Then the response is forwarded to the client. A SMSESSION cookie is sent together to the client so that the client can reuse it to achieve the Single Sign-on.

SiteMinder Components Installation and Configuration

Installing Microsoft Active Directory Server as Policy Store

The Policy Store is required by Policy Server. It can be a LDAP server or Database server. In this white paper, Active Directory Server is installed as Policy Store. Any other LDAP or Database can be used as well. But the configuration are different.

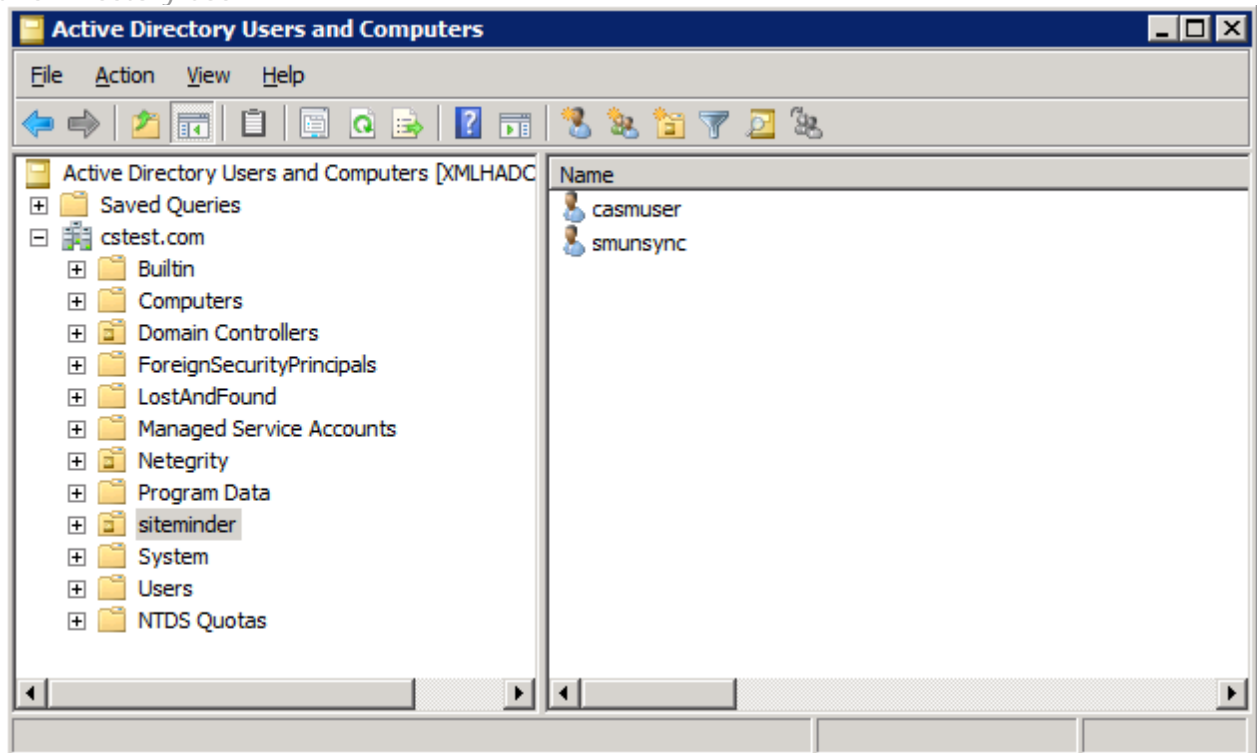
- 1) Setup a Windows Active Directory domain controller, the LDAP default port is 389(non-ssl) and 636(ssl).
- 2) Create a test Organizational Unit and a test user.

The distinguishedName of the created OU and user:

☐ ou=siteminder,dc=cstest,dc=com

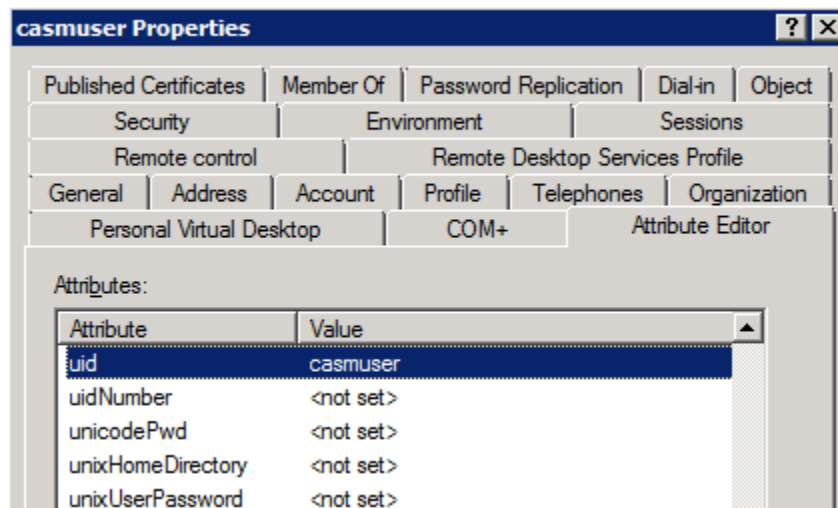
☐ cn=casmuser,ou=siteminder,dc=cstest,dc=com

Figure 4 Active Directory user



Since the attribute uid is mandatory by default in Content Server LDAP configuration, edit uid and set a value: right-click casmuser -> **properties** -> **Attribute Editor**, edit uid value to the user *casmuser*.

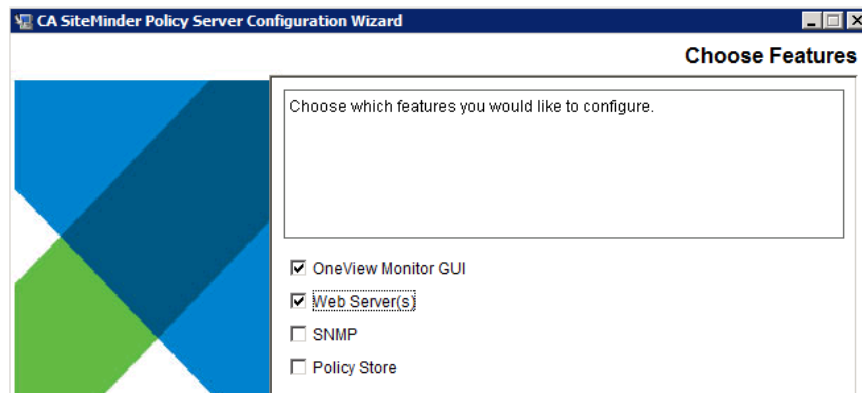
Figure 5 Set UID attribute



Install SiteMinder Policy Server 15.5

- 1) Before you install the Policy Server, accomplish following tasks:
 - a. ☐ Install JDK 6 or later
 - b. ☐ Turn off Windows Firewall
 - c. ☐ Install IIS web server 7.5 or later, add the following Role Services
 - i. ASP.NET
 - ii. CGI
 - iii. ISAPI Extensions
 - iv. ISAPI Filters
 - v. IIS Management Console
 - vi. Windows Authentication (for the SiteMinder Windows Authentication Scheme)
 - d. ☐ Install servletexec in thirdparty-tools folder
- 2) Run the Policy Server Installer as Administrator; choose OneView Monitor GUI and Web Server(S), we will configure Policy Store manually later.

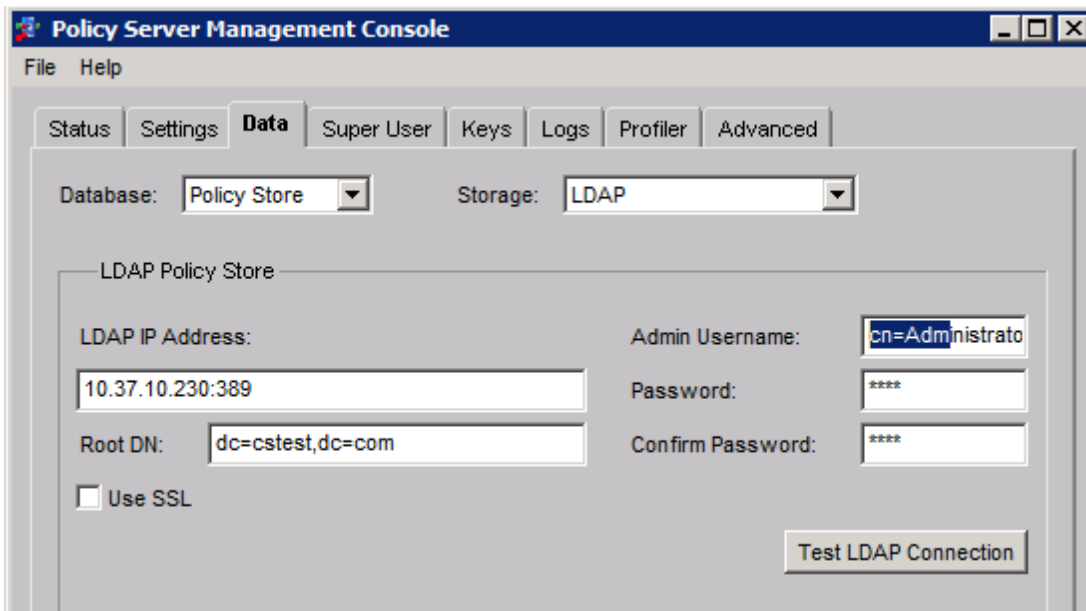
Figure 6 Installing Policy Server



Configuring Active Directory Server as Policy Store

- 1) Point the Policy Server to the directory server.
 - a. Start -> All Programs -> CA -> SiteMinder -> SiteMinder Policy Server Management Console.
 - b. Click the Data tab.
 - c. Select the following value from the Database list: Policy Store.
 - d. Select the following value from the Storage list: LDAP.
 - e. Configure the following settings in the LDAP Policy Store group box.
 - i. LDAP IP Address
 - ii. Admin Username
Note: the distinguishedName of the domain Administrator user
 - iii. Password
 - iv. Confirm Password
 - v. Root DN
 - f. Click Apply.
 - g. Click Test LDAP Connection to verify that the Policy Server can access the policy store.
 - h. Select the following value from the Database list: Key Store.
 - i. Select the following value from the Storage list: LDAP.
 - j. Select the following option: Use Policy Store database.
 - k. Click OK.

Figure 7 Policy Server Management Console



2) Create the Policy Store schema.

- a. Run the following command from the Policy Server host system:
`smldapsetup ldgen -ffile_name`
`file_name`: Specifies the name of the LDIF file you are creating. An LDIF file with the SiteMinder schema is created.
- b. Run the following command:
`smldapsetup ldmod -ffile_name`
`file_name`: Specifies the name of the LDIF you created. The utility imports the policy store schema.
- c. Navigate to `policy_server_home\xps\db` and open the following file: `ActiveDirectory.ldif`
- d. Manually replace each instance of `<RootDN>` with the DN that represents the policy store schema location, not the policy store object location.
 Example: If the following root DN represents the policy store object: `ou=policystore,dc=cstest,dc=com`
 Replace each instance of `<RootDN>` with the following DN: `dc=cstest,dc=com`
- e. Run the following command:
`smldapsetup ldmod -fpolicy_server_home\xps\db\ActiveDirectory.ldif`
`policy_server_home`: Specifies the Policy Server installation path.
 The policy store schema is extended. You have created the policy store schema.

3) Set the SiteMinder superuser password.

- a. The default SiteMinder administrator account is named: siteminder. The account has maximum permissions.
Copy the smreg utility to siteminder_home\bin.
siteminder_home: Specifies the Policy Server installation path.
Note: The utility is at the top level of the Policy Server installation kit.
 - b. Run the following command: smreg -su password
Specifies the password for the default SiteMinder administrator.
- 4) Import the Policy Store data definitions.
 - a. Open a command window and navigate to siteminder_home\xps\dd.
siteminder_home: Specifies the Policy Server installation path.
 - b. Run the following command:
XPSDDInstall SmMaster.xdd
Imports the required data definitions.
- 5) Import the default Policy Store objects.
Before running a SiteMinder utility or executable on Windows Server 2008, open the command line window with Run as administrator.
 - a. Open a command window and navigate to siteminder_home\db.
 - b. Import one of the following files:
 - To import smpolicy.xml, run the following command:
XPSImport smpolicy.xml -npass
 - To import smpolicy-secure.xml, run the following command:
XPSImport smpolicy-secure.xml -npass
 npass: Specifies that no passphrase is required. The default policy store objects do not contain encrypted data.
- 6) Restart the Policy Server.
 - a. Open the Policy Server Management Console.
 - b. Click the Status tab, and click Stop in the Policy Server group box. The Policy Server stops as indicated by a red stoplight.
 - c. Click Start. The Policy Server starts as indicated by a green stoplight.
- 7) Prepare for the Administrative UI Registration.
 - a. Log into the Policy Server host system.
 - b. Run the following command:
XPSRegClient siteminder[:passphrase] -adminui-setup
passphrase: Specifies the password for the default SiteMinder super user account (siteminder).
- 8) Enable ObjectCategory attribute support.
 - a. Launch the Windows Registry Editor.
 - b. Locate the key
HKLM\Software\Netegrity\SiteMinder\CurrentVersion\DS\LDAP
Provider.
To enable support, set the EnableObjectCategory value to 1.

Note: The default value is 0.

Install SiteMinder Administrative UI 12.5

Unzip the prerequisite installer and the Administrative UI installer, be sure that both executables are in the same location. The prerequisite installer automatically starts the Administrative UI installer to complete the installation.

- 1) Double-click adminui-pre-req-12.5-win32.exe.
- 2) Enter the required values. Click Install.
- 3) The required components are installed. Click Done.
- 4) The Administrative UI installer starts. Follow the prompts and click Install.
- 5) The Administrative UI is installed. The SiteMinder Administrative UI login screen appears.
- 6) Type siteminder in the User Name field.
- 7) Type the siteminder account password in the Password field.
- 8) Type the fully qualified Policy Server host name in the Server field.
- 9) The Administrative UI opens and is registered with the Policy Server.

To start or stop Administrative UI application server, operate SiteMinder Administrative UI service in Services.

To open Administrative UI, **Start -> All Programs -> CA -> SiteMinder -> SiteMinder Administrative User Interface.**

Installing Apache-based Web Agent

Install Web Agent on a dedicated machine which is different from the machine where Policy Server is installed.

- 1) Installing and configuring the Apache server.
 - a. Install the Apache server as a service for all users.
 - b. Configure Apache server as a proxy for the REST server, e.g. `http://dfs-0x10/dctm-rest`.
 - i. Open `/conf/httpd.conf`, uncomment the following lines:
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_connect_module modules/mod_proxy_connect.so
LoadModule proxy_http_module modules/mod_proxy_http.so
Include conf/extra/httpd-vhosts.conf

- ii. Edit `/conf/extra/httpd-vhosts.conf` as shown in the following example:

```
<VirtualHost _default_:80>
    ServerAdmin someone@emc.com
    DocumentRoot "C:/Apache2.2/htdocs"
    ServerName restsmwa2:80

    SSLProxyEngine on
    ProxyRequests Off
    ProxyPass /dctm-rest http://dfs-0x10/dctm-rest
    ProxyPassReverse /dctm-rest http://dfs-0x10/dctm-rest
    ProxyPreserveHost on
</VirtualHost>
```
- iii. Restart Apache server.
- iv. Test with `"http://localhost/dctm-rest/services"`, the request can be forward successfully.

2) Configure objects in Administrative UI.

- a. Infrastructure -> Agent -> Agents -> Create Agent -> Create a new object of type Agent.

Name: restsmwa

IP Address: Host IP address where install Web Agent

Shared Secret: password

Confirm Secret: password

Figure 8 Configure Agent in Administrative UI

General	
•Name	restsmwa
Description	

Agent Type Settings	
Select an agent type	<input checked="" type="radio"/> SiteMinder <input type="radio"/> RADIUS
Agent Type	Web Agent
Supports 4.x agents	<input checked="" type="checkbox"/>

Trust Settings	
•IP Address	10.32.122.51
•Shared Secret	••••••••
•Confirm Secret	••••••••

- b. Infrastructure -> Agent -> Agent Configuration Objects -> Create Agent Configuration -> Create a copy of an object of type Agent Configuration -> ApacheDefaultSettings.

- i. Set DefaultAgentName, uncomment #DefaultAgentName by removing # and enter Web Agent name created in previous step.
- ii. Set AllowCacheHeaders to yes.
- iii. Set RequireCookies to no.
- iv. Set CssChecking to no.
- v. Uncomment BadQueryChars and BadUrlChars
Name: REST_ApacheDefaultSettings

Figure 9 Create Agent Configuration in Administrative UI

General

•Name Description

- c. Infrastructure -> Hosts -> Host Configuration Objects -> Create Host Configuration -> Create a copy of an object of type Host Configuration -> DefaultHostSettings.
Name: REST_DefaultHostSettings
Host: Policy Server IP address

Figure 10 Create Host Configuration in Administrative UI

General

•Name Description

Configuration Values

	Host	Accounting Port	Authentication Port	Authorization Port
Policy Server	10.32.122.40	44441	44442	44443

Add

Enable Failover ☒

Maximum Sockets Per Port

Minimum Sockets Per Port

New Socket Step

Request Timeout

- d. Infrastructure -> Directory -> User Directories -> Create User Directory
Name: REST_LDAP
Server: LDAP URL
UserName: Domain Administrator user
Password: password
Root: dc=cstest,dc=com
Start: cn=
End: ,ou=siteminder,dc=cstest,dc=com

Figure 11 Create User Directory in Administrative UI

General

•Name Description

Directory Setup

Namespace LDAP:

Server

Use authenticated user's security context ☐

Secure Connection ☐

Administrator Credentials

Require Credentials ☒

Username

Password

Confirm Password

LDAP Settings

LDAP Search

Root

Scope ☐ One Level ☒ Sub-Tree

Max Time

Max Results

LDAP User DN Lookup

Start

End

Effective Lookup Specifies the text string used in the search expression or filter.

- e. Policies -> Domain -> Domains -> Create Domain.
Name: REST_DOMAIN
- f. User Directories -> Add/Remove -> Add the created User Directory to Selected Members.
- g. Create a realm
Name: REST_REALM
Agent: restsmwa

Figure 12 Create Realm in Administrative UI

General

•Name Description

Domain REST_DOMAIN

Resource

•Agent

Resource Filter

Effective Resource

Default Resource Protection ☒ Protected ☐ Unprotected

Authentication Scheme

- h. Create a rule, select all options in Action.

Figure 13 Create Rule in Administrative UI

Attributes

Realm and Resource

Resource: *

Effective Resource: REST_AgentGroup/*

Regular Expression: ☒

Allow/Deny and Enable/Disable

☒ Allow Access

☐ Deny Access

Enabled: ☒

Action

☒ Web Agent actions

☐ Authentication events

☐ Authorization events

☐ Impersonation events

• Action

- Trace
- Options
- Head
- Delete

i. Create a policy, select Users tab, Add Members.

Figure 14 Add user members for policy in Administrative UI

General **Users** **Rules** **Expression**

• = Required

User Directories

REST_LDAP

Allow Nested Groups: ☒

AND Users/Groups: ☒

Name	User Class	Exclude
OU=siteminder,DC=cstest,DC=com	organizationalUnit	Exclude

Add Members Add Entry Add All

j. Run a functionality test.

Start -> **All Programs** -> **CA** -> **SiteMinder** -> **SiteMinder Test Tool** -> *fill the fields* -> **Connect** -> **IsProtected** -> **IsAuthenticated** -> **IsAuthorized** -> **DoAccounting** -> **DoManagement**, all commands should pass. Details about the tool please refer to: [link](#).

Figure 15 Running test tool to verify configuration

3) Install Web Agent using Web Agent installer.

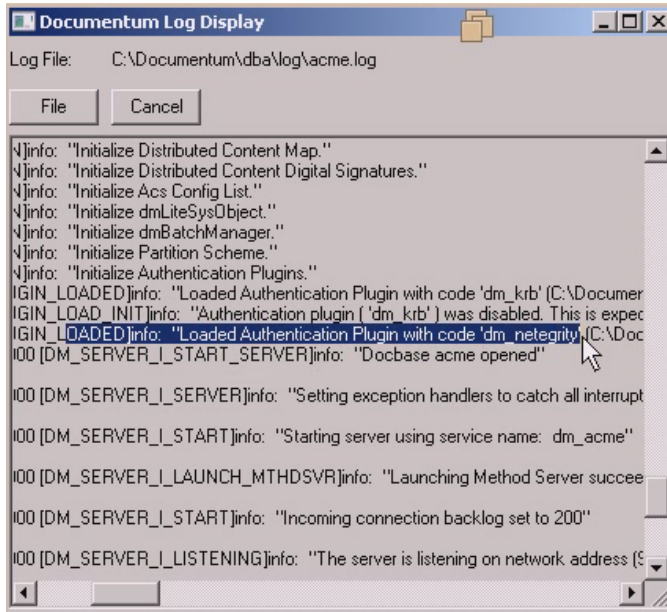
4) Configure Web Agent

- a. Yes, I would like to do Host Registration now.
- b. Admin User Name: siteminder.
- c. Admin Password: siteminder's password.
- d. Trusted Host Name: any value.
- e. Host Configuration Object: REST_DefaultHostSetting which created in previous step.
- f. IP Address: Policy Server IP Address.
- g. FIPS Compatibility Mode.
- h. File name: default value.
- i. Select a location: default value.
- j. Apache 2.2.22.
- k. Default Web Site.
- l. Agent Configuration Object: REST_ApacheDefaultSettings which created in previous step.
- m. Check Yes to enable Web Agent.
- n. Click Install.

Configure Content Server

- 1) Install the netegrity plugin.
 Navigate to
`%DM_HOME%\install\external_apps\authplugins\netegrity,` follow
 the steps in readme.txt
- 2) Restart Content Server and check the log file to see whether the
 netegrity plugin is loaded
 "Loaded Authentication Plugin with code 'dm_netegrity'..."

Figure 16 Content server log with dm_netegrity plugin loaded



- 3) Create an LDAP configuration object
 Logon DA -> Basic Configuration -> LDAP Servers -> File -> New
 LDAP Server Configuration
HostName/IP Address: LDAP Server URL
Port: LDAP Server port

Figure 17 Configure LDAP in DA of content server

LDAP Server Configuration Properties

Info Sync & Authentication Mapping Failover

LDAP Server Configuration : MSAD

*Name : MSAD

Status : ☒ Enable this LDAP Configuration

LDAP Directory

Directory Type : Microsoft Active Directory

*Hostname / IP Address : 10.37.10.230

*Port : 389

*Binding Name : cctest\Administrator

*Binding Password : [Set](#)

Figure 18 Configure LDAP mapping in DA of content server

LDAP Server Configuration Properties

Info Sync & Authentication Mapping Failover

LDAP Server Configuration : MSAD

User Mapping

*User Object Class : user

*User Search Base : dc=cctest,dc=com

User Search Filter : (cn=)

[Search Builder...](#) (Requires a valid User Object Class)

Group Mapping

*Group Object Class : group

*Group Search Base : dc=cctest,dc=com

Group Search Filter : (cn=)

[Search Builder...](#) (Requires a valid Group Object Class)

Property Mapping

[Add](#) [Edit](#) [Delete](#) Show Items 10

Repository Property	Type	Map To	Map Type	Mandatory
user_name	dm_user	cn	ATTRIBUTE	Yes
user_login_name	dm_user	uid	ATTRIBUTE	Yes
user_address	dm_user	mail	ATTRIBUTE	No
group_name	dm_group	cn	ATTRIBUTE	Yes

- 4) Right-click the LDAP Server Configuration object -> Synchronize Server, go to User Management, make sure casuser is synchronized.

Configure Rest Service Server

- 1) Open the configuration file /WEB-INF/classes/rest-api-runtime.properties
- 2) Set rest.security.auth.mode to siteminder
rest.security.auth.mode=siteminder
- 3) Locate rest.security.siteminder.cookie.name, and uncomment it
rest.security.siteminder.cookie.name=
- 4) Locate rest.security.siteminder.user.header, and uncomment it
rest.security.siteminder.user.header=
- 5) Restart the Rest server

Figure 19 Rest Service configurations

```
# The default mode is basic. For more information, refer to the Reference Guide for details.
rest.security.auth.mode=siteminder

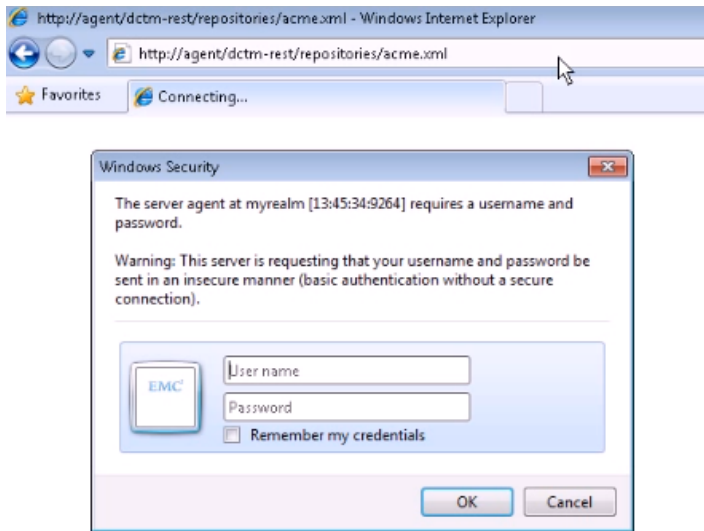
# Specify the SiteMinder cookie name for the request set by the SiteMinder web agent after the successful SSO
authentication
# This property CAN be left as empty, where the actual value is "SMSESSION"
rest.security.siteminder.cookie.name=

# Specify the HTTP header representing the authenticated principals set by the SiteMinder web agent after the
successful SSO authentication
# This property CAN be left as empty, where the actual value is "SM_USER".
rest.security.siteminder.user.header=
```

Accessing Rest Services through SiteMinder Agent

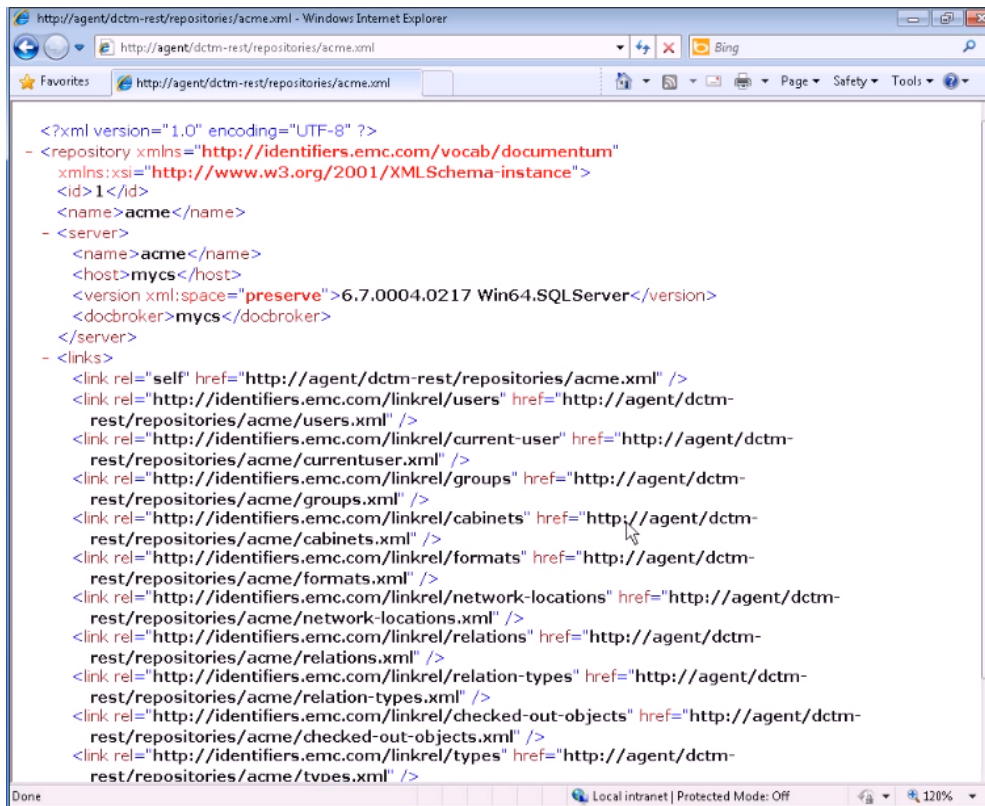
- 1) Get the Repository resource through Internet Explorer without credential
The request is sent to a SiteMinder Agent host.
- 2) The SiteMinder Agent challenge the client for credential

Figure 20 Get Repository resource without credential



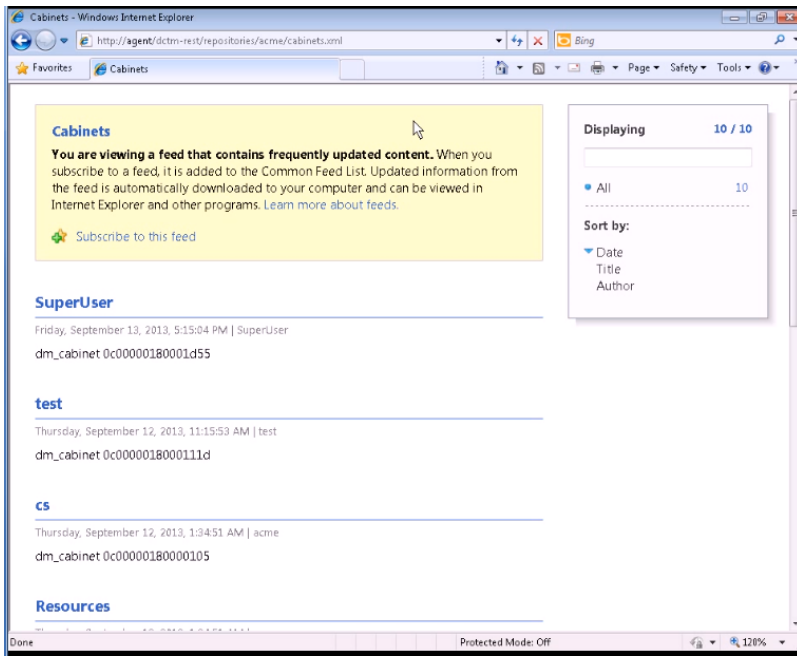
3) After the client inputs credential information, the response is sent back

Figure 21 Response with correct credential



4) The client continues getting other resources, no credential required, e.g. cabinets resource.

Figure 22 Get Cabinets resource



Troubleshooting

Logging

- 1) Enable logging of Siteminder products, Content Server, Proxy Server
Please refer relative documentation.
- 2) Enable log of REST server
 - a. Open <dctm-rest>\WEB-INF\classes\log4j.properties
 - b. Add loggers for packages,
log4j.logger.com.emc.documentum.rest.security=TRACE
log4j.logger.com.emc.documentum.rest.log=DEBUG

Figure 23 log4j.properties

```
log4j.logger.com.emc.documentum.rest.log=DEBUG
log4j.logger.com.emc.documentum.rest.dfc=TRACE
log4j.logger.com.emc.documentum.rest.security=TRACE
log4j.logger.org.jasig.cas=TRACE
log4j.logger.net.sf.ehcache=TRACE
```

- c. Open <dctm-rest>\WEB-INF\classes\rest-api-runtime.properties
Set the property rest.message.logging.enabled to true

Figure 24 rest-api-runtime.properties

```
# Determines whether or not to enable the REST request and response message logging on the server
side.
# To enable the message logging, set this property to TRUE, and enable DEBUG logging level for the
package 'com.emc.documentum.rest.log' in log4j.
# The default value is false.
rest.message.logging.enabled=true

# Specifies the logging buffer size in byte for requests and responses when the message logging is
enabled.
# The value MUST be a non-negative integer.
# The default value is 1048567.
rest.message.logging.buffer=10240
```

- d. The output log file is defined by the log4j as well,
which defaults to target/rest-api.log
 - e. Dump rest-api-runtime.properties setting

Configure ACS/BOCS with SiteMinder

When accessing a REST server with SiteMinder, the link generated by the REST server should use the SiteMinder Agent host. But the ACS/BOCS links currently are directly pointing to the ACS/BOCS server. They may be required to be protected by SiteMinder as well. The following are two sample solutions which may work without changing the code.

Solution 1: change ACS/BOCS "Base URL" configuration

1) Change the ACS/BOCS configuration

The base url of ACS/BOCS should be changed to SiteMinder Agent's host and port. Then every time when ACS/BOCS generates a binary link, it will start with Agent's host and port.

One limitation is, if there are other applications/components which depend on ACS/BOCS links, these applications/components should be protected by SiteMinder as well, just like the REST server. Otherwise, the ACS/BOCS link is not accessible since it requires SiteMinder credential.

If there is more than one base URLs configured in ACS/BOCS, the result is not predictable.

2) SiteMinder Agent reverse proxy configuration

The following is the sample for an Apache server:

```
ProxyPass /dctm-rest http://rest-host:8080/dctm-rest
ProxyPass /ACS http://cs-host:9080/ACS
```

```
ProxyPassReverse /dctm-rest http://rest-host:8080/dctm-rest
ProxyPassReverse /ACS http://cs-host:9080/ACS
```

- 3) Any requests sent to rest with http(s)://agent-host/dctm-rest/xxx will be forwarded to the REST server, while requests sent to http(s)://agent-host/ACS/xxx will be forwarded to the ACS(content server) host.
- 4) Restart Apache after change.

Solution 2: URL rewrite

- 1) Change the reverse proxy's configuration (where the SiteMinder Agent is deployed) to modify the response from the REST server, and rewrite the links to the correct proxy server.
 - a. Apache server

i. load relative modules

```
LoadModule proxy_module modules/mod_proxy.so
LoadModule substitute_module modules/mod_substitute.so
```

ii. update the configuration (virtual host used here)

```
<VirtualHost _default_:80 _default_:10080>
    ServerAdmin admin@emc.com
    ServerName agent
    ProxyRequests Off
    <Proxy *>
        Order deny,allow
        Allow from all
    </Proxy>

    ProxyPass /dctm-rest http://rest-host:8080/dctm-rest
    ProxyPass /ACS http://content-server-host:9080/ACS

    ProxyPassReverse /dctm-rest http://rest-host:8080/dctm-rest
    ProxyPassReverse /ACS http://content-server-host:9080/ACS

    ProxyPreserveHost on

    <Location />
        Order allow,deny
        Allow from all
    </Location>

    <Location />
        AddOutputFilterByType SUBSTITUTE application/xml
        application/json application/vnd.emc.documentum+xml
        application/vnd.emc.documentum+json application/home+json
        application/home+xml
        Substitute s|content-server-host:9080/ACS|agent/ACS|ni
    </Location>
</VirtualHost>
```

b. IIS server

i. Install relative modules

The APR (Application Request Routing) and URL Rewrite module need be installed within IIS.

ii. APR configuration

Figure 25 Configure Application Request Routing

Application Request Routing

Use this feature to configure proxy settings for Application Request Routing.

☒ Enable proxy

Proxy Setting

HTTP version:

Pass through

☒ Keep alive

Time-out (seconds):

120

☒ Reverse rewrite host in response headers

Custom Headers

Preserve client IP in the following header:

X-Forwarded-For

☒ Include TCP port from client IP

Forwarding proxy header value:

Cache Setting

Memory cache duration (seconds):

60

☒ Enable disk cache

☐ Enable request consolidation

Query string support:

Ignore query string

Buffer Setting

Response buffer (KB):

4096

Response buffer threshold (KB):

256

Proxy Type

☒ Use URL Rewrite to inspect incoming requests

☒ Enable SSL offloading

Reverse proxy:

CNRDRUANW1L1C

iii. URL Rewriting configuration

Figure 26 Configure URL rewriting

Edit Outbound Rule

Name:

rest

Precondition:

<None>

Edit...

Match

Matching scope:
Response

Match the content within:
Link

Custom tags:

Content:
Matches the Pattern

Using:
Wildcards

Pattern:
http://CNRDRUANW1L1C:8080/dctm-rest/*

Test pattern...

☒ Ignore case

Conditions

Action

Action type:
Rewrite

Action Properties

Value:
http://agent/dctm-rest/{R:1}

☐ Stop processing of subsequent rules

- iv. inbound rule for the rest server (this is automatically created by the APR configuration, just modify it)

Figure 27 Configure inbound rule for Rest Server



Edit Inbound Rule

Name:

ARR_server_proxy

Match URL

Requested URL:

Matches the Pattern

Using:

Wildcards

Pattern:

dctm-rest/*

Test pattern...

☒ Ignore case

Conditions

Server Variables

Action

Action type:

Rewrite

Action Properties

Rewrite URL:

http://rest-host:8080/{R:0}

☒ Append query string

☒ Stop processing of subsequent rules


- v. Create a new inbound rule for the ACS server

Figure 28 Configure inbound rule for ACS

Edit Inbound Rule

Name:

acs

Match URL 


Requested URL: Using:

Pattern: [Test pattern...](#)

☒ Ignore case

Conditions 

Server Variables 

Action 

Action type:

Action Properties

Rewrite URL:

☒ Append query string

☒ Stop processing of subsequent rules

vi. Create a new outbound rule for the REST server.

Figure 29 Configure outbound rule for Rest Server

Edit Outbound Rule

Name:

rest full

Precondition:

<None>

Edit...

Match

Matching scope:
Response

Match the content within:

Custom tags:

Content:
Matches the Pattern

Using:
Exact Match

Pattern:
http://rest-server-host:8080/dctm-rest

Test pattern...

☒ Ignore case

Conditions

Action

Action type:
Rewrite

Action Properties

Value:
http://agent/dctm-rest/

☐ Stop processing of subsequent rules

vii. Create a new outbound rule for ACS.

Figure 30 Configure outbound rule for ACS



Edit Outbound Rule

Name:

acs.json

Precondition:

<None>

Edit...

Match

Matching scope:

Response

Match the content within:

Custom tags:

Content:

Matches the Pattern

Using:

Exact Match

Pattern:

http://content-server-host:9080/ACS/

Test pattern...

☒ Ignore case

Conditions

Action

Action type:

Rewrite

Action Properties

Value:

http://agent/ACS/

☐ Stop processing of subsequent rules

Conclusion

This white paper explains the architecture of SiteMinder SSO, and how to integrate SiteMinder SSO with Documentum REST Services. For further information on SiteMinder protocol and

configuration, please refer to SiteMinder project site. For feature requests and successful stories on the Documentum REST Services, please contact EMC product manager for Documentum REST Services. For further information on the SiteMinder SSO integration for Documentum REST Services, please contact EMC support.

References

- EMC Support: <http://support.emc.com>
 - Documentum Platform REST Services 7.1 Development Guide
 - Documentum Platform REST Services 7.1 Release Notes
 - Documentum Content Server 7.1 Administration and Configuration Guide
 - DOCUMENTUM CONTENT SERVER CENTRAL AUTHENTICATION SERVICE (SiteMinder) SSO A Detailed Review
- SiteMinder project site: <http://www.ca.com/us/secure-sso.aspx>
- SiteMinder resources: <http://www.ca.com/us/secure-sso-resources.aspx>