



White paper version: 13.0

Date: 24th February 2015

AUTHORS: Bhavya Natarajan

Krishna Chalamasandra

**A COMPLETE GUIDE FOR
THE INSTALLATION, CONFIGURATION, AND INTEGRATION OF
OPEN ACCESS MANAGER WITH
CISCO UNIFIED COMMUNICATIONS MANAGER 8.5/8.6 AND ABOVE,
CISCO UNITY CONNECTION 8.6 AND ABOVE , AND ACTIVE
DIRECTORY FOR SINGLE SIGN-ON**

Table of Contents

Preface	4
1 Introduction.....	5
2 Configuring Domain Controller on Windows 2003 Server	5
3 Configuring DNS on Windows 2003 Server Domain Controller	14
4 Configuring Windows Client Desktop as Domain Computer of Domain Controller	17
5 Brief History of OpenSSO and OpenAM	21
6 System Requirements for OpenAM Installation.....	21
6.1 OpenAM Installation on Linux Platform	21
6.2 OpenAM Installation on Windows Platform.....	23
7 Installation and Configuration of Apache Tomcat with HTTPS.....	24
7.1 Installation and configuration of Apache Tomcat on Linux platform	24
7.2 Installation and Configuration of Apache Tomcat on Windows Platform.....	25
8 Provisioning Active Directory for Single Sign-On	26
9 Deploying OpenSSO (OpenAM) Enterprise on Apache Tomcat	26
9.1 Deploying OpenSSO Enterprise War on Apache Tomcat over Linux platform	26
9.2 Deploying OpenSSO Enterprise War on Apache Tomcat over Windows Platform	27
10 Configuring OpenSSO Enterprise Using the GUI Configurator	27
10.1 Configuring Policies on OpenSSO Server	31
10.1.1 Configuring Policies on OpenSSO Server for Cisco Unified Communications Manager 8.5, 8.6	31
10.1.2 Configuring Policies on OpenSSO Server for Cisco Unity Connection 8.6	41
10.2 Configuring Windows Desktop SSO Authentication Module Instance	45
10.3 Configuring J2EE Agent Profile on OpenSSO Server.....	46
10.3.1 Configuring J2EE Agent Profile on OpenSSO Server for Cisco Unified Communications Manager 8.5, 8.6.....	46
10.3.2 Configuring J2EE Agent Profile on OpenSSO Server for Cisco Unity Connection 8.6	51
11 Undeploying/Removing OpenSSO Enterprise (OpenAM).....	55
11.1 Uninstalling OpenSSO Enterprise (OpenAM) Server Deployed on Linux Platform	55
11.2 Uninstalling OpenSSO Enterprise (OpenAM) Server Deployed on Windows Platform	55
12 Configuring Browser/Registry for SSO.....	56
12.1 Internet Explorer	56
12.2 Mozilla Firefox.....	56
12.3 Configuring Windows Registry for RTMT SSO	57
12.4 SSO Configurations Test with Browser	57
13 Configuring SSO on Cisco Unified Communications Manager 8.5	58
13.1 Unified CM 8.5 SSO CLI Commands	59
13.1.1 utils sso enable	59
13.1.2 utils sso disable	60
13.1.3 utils sso status	60
14 Configuring SSO on Cisco Unified Communications Manager 8.6	60
14.1 Unified CM 8.6 SSO CLI Commands	61
14.1.1 utils sso enable	61
14.1.2 utils sso disable	62
14.1.3 utils sso status	62
14.2 CUCM 8.6 SSO GUI	62
15 Configuring SSO on Cisco Unity Connection 8.6.....	63
15.1 Cisco Unity Connection 8.6 SSO CLI Commands	64
15.1.1 utils sso enable	64
15.1.2 utils sso disable	66

15.1.3	utils sso status	66
15.2	Cisco Unity Connection 8.6 SSO GUI.....	67
16	OpenSSO Enterprise Session Failover	68
16.1	Requirements for Access Manager Session Failover (AMSFO).....	69
16.2	Configuration of AMSFO components.....	69
16.2.1	Installation and Configuration of Load Balancer.....	69
16.2.1.1	Installation of Load Balancer on Linux Platform	69
16.2.1.2	Installation of Load Balancer on Windows Platform.....	72
16.2.1.3	Configuration of Load Balancer for HTTP Load Balancing	77
16.2.2	Installation and Configuration of Session Failover Components	86
16.2.2.1	Configuration of Session Failover Components on Linux Platform	86
16.2.2.2	Configuration of Session Failover Components of Windows Platform	92
16.2.3	Installation and configuration of OpenAM Enterprise Servers for Session Failover	98
16.2.3.1	Installation of OpenAM Enterprise Server 1	98
16.2.3.2	Installation of OpenAM Enterprise Server 2	104
16.2.3.3	Configure OpenSSO Enterprise for Session Failover.....	107
16.3	Configuring SSO on Cisco Unified Communications Manager with AMSFO Setup	107

Preface

This document covers the installation and configuration of all the required software that is essential for achieving a Single Sign-On solution with Cisco Unified Communications Manager 8.5/8.6 and Cisco Unity Connection 8.6. Here is a brief picture of coverage in each chapter.

Chapter 1: This chapter contains a brief introduction to the products that are installed and configured as part of this document.

Chapter 2: This chapter covers installation of Active Directory Service on domain controller. Skip this chapter if you already have the domain controller set up in your environment.

Chapter 3: This chapter covers installation of DNS on domain controller. Skip this chapter if you already have the DNS configured on your domain controller.

Chapter 4: This chapter covers configuring the desktop as the domain computer of the previously set-up domain controller. Skip this chapter if your desktop is already a domain computer of the domain controller that is under test.

Chapter 5: This chapter contains a brief introduction to OpenSSO and OpenAM.

Chapter 6: This chapter covers installation and configuration of Linux OS/Windows OS for OpenAM deployment.

Chapter 7: This chapter covers installation and configuration of Apache Tomcat with SSL on the Linux and Windows Platform that is set up in Chapter 6.

Chapter 8: This chapter covers provisioning Active Directory for Windows Desktop SSO Authentication.

Chapter 9: This chapter covers installing OpenSSO Enterprise on Linux/Windows platform.

Chapter 10: This chapter covers configuring OpenSSO with policies, agents, and authentication module instance.

Chapter 11: This chapter covers uninstalling OpenSSO Enterprise on Linux/Windows platform.

Chapter 12: This chapter covers configuring browsers for Single Sign-On.

Chapter 13: This chapter covers SSO Enable/Disable/Status on Cisco Unified Communications Manager 8.5.

Chapter 14: This chapter covers SSO Enable/Disable/Status on Cisco Unified Communications Manager 8.6

Chapter 15: This chapter covers SSO Enable/Disable/Status on Cisco Unity Connection 8.6.

Chapter 16: For the high-availability OpenSSO Server, OpenSSO Enterprise session failover can be implemented. This chapter explains how to configure OpenSSO Enterprise session failover. Ignore this chapter if you do not wish to have OpenSSO session failover.

1 Introduction

This document covers the installation and configuration of the required software components that are essential for achieving a Single Sign-On (SSO) solution with Cisco Unified Communications Manager 8.5/8.6 and Cisco Unity Connection 8.6.

Below is the list of products that are installed and configured as part of this guide:

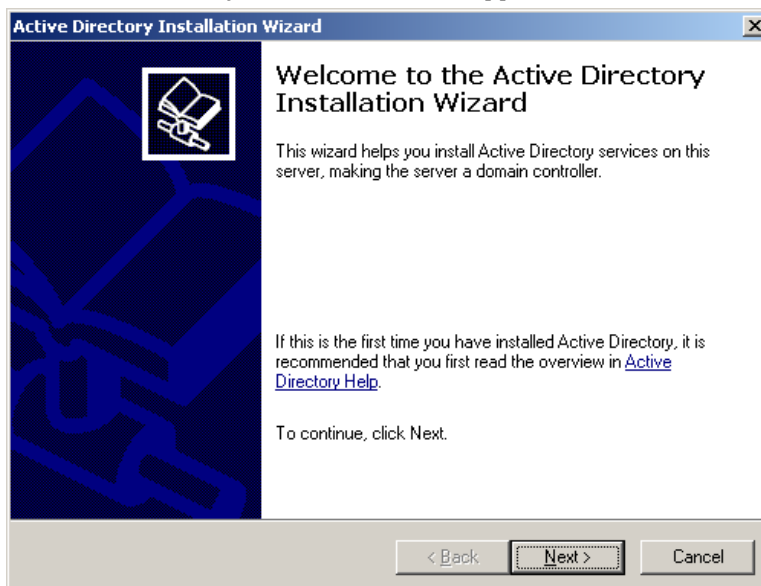
- Microsoft Windows 2003 server with SP2—For domain controller and DNS configurations. For instance, in this guide the domain controller is configured for the domain **vrajoli.com**.
- Redhat Enterprise Linux (RHEL) 5.5—For OpenAM deployment
- Microsoft Windows 2003 Server—For OpenAM deployment
- Windows XP with SP3—Client Desktop

Note: If you are planning to use Windows 2008 Server for configuring Active Directory, make sure Windows 2008 Server has SP2 installed.

2 Configuring Domain Controller on Windows 2003 Server

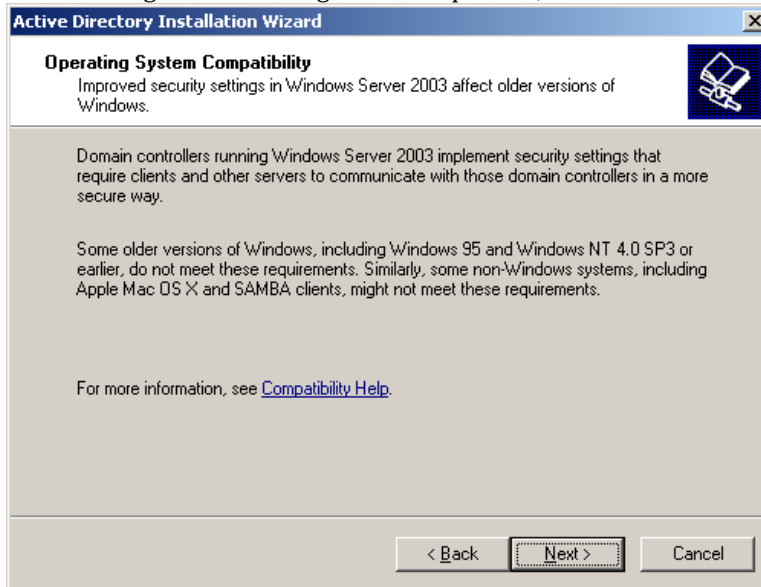
Setting up Active Directory using the run command **dcpromo** is a straightforward procedure. To begin, from your Windows 2003 Server desktop go to **Start**. Click **Run**, type **dcpromo**, and then click **Enter**.

The Active Directory Installation Wizard appears. Click **Next**.



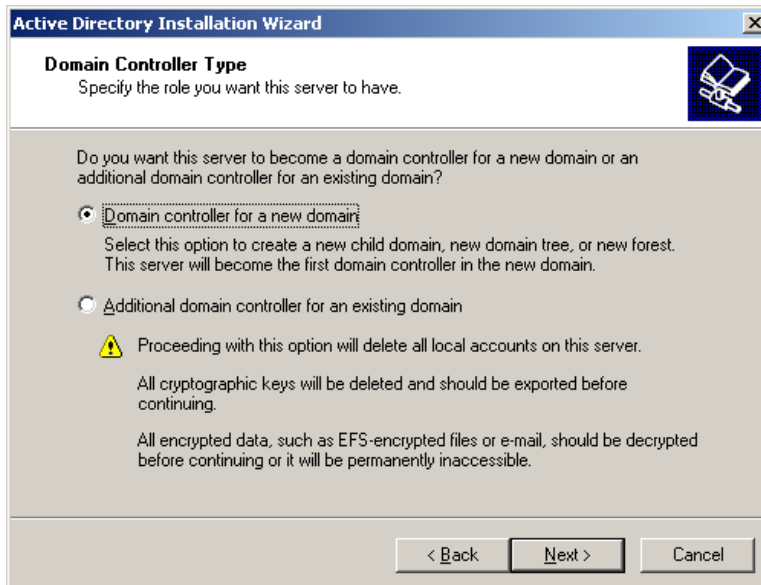
In the following window, you get a warning about compatibility issues with other Operating Systems. Improved security settings in Windows Server 2003 affect older versions of Windows.

After reading and evaluating the consequences, click **Next**.



In the next window, you see two options. The first option asks you if you want the server to become a domain controller for a new domain or if you want the server to be an additional domain controller for an existing domain.

Select the first option, "Domain controller for a new domain," and click **Next**.

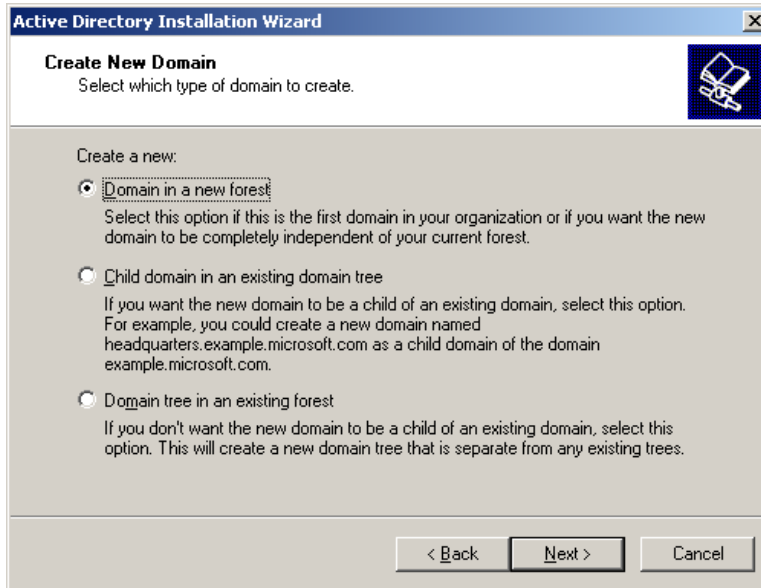


In the next window, you get three options:

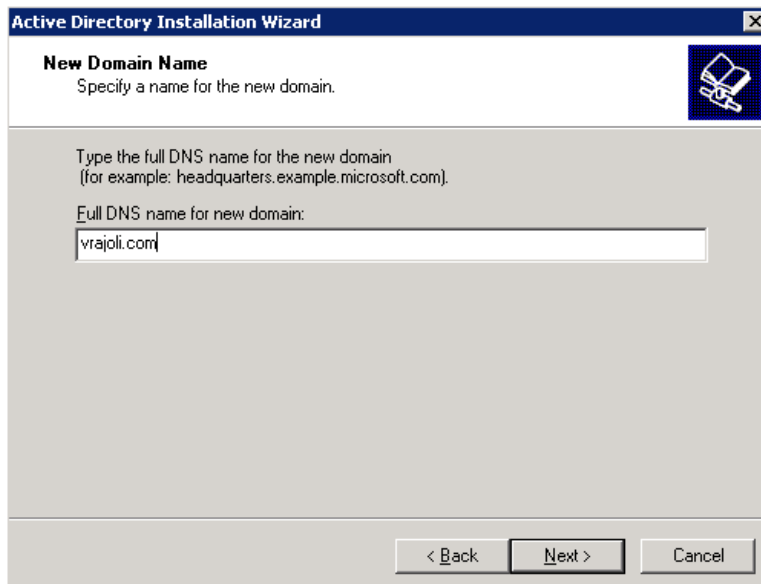
- 1) The first option is to set up the domain in a new forest. Select this option if this is the first domain controller in your organization, or if you want it to be totally independent of any forest.

- 2) The second option is to create a new child domain in an existing domain tree. Select this option if you want the domain to be a child domain from an existing domain.
- 3) The third option is to create a domain tree in an existing forest. If you do not want any of the above, select this option.

In our case, choose the first option, “Domain in a new forest,” and click **Next**.

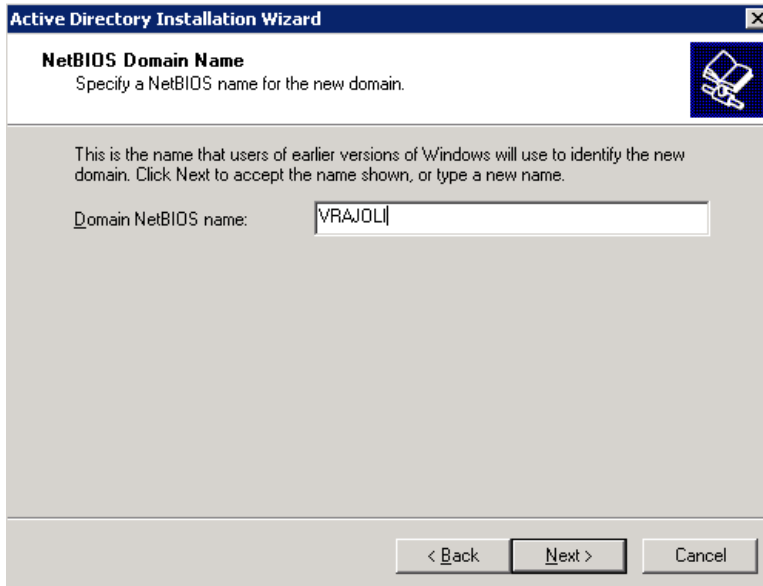


In the next window, type the full DNS name for the new domain (for example, “vrajoli.com”) and click **Next**.



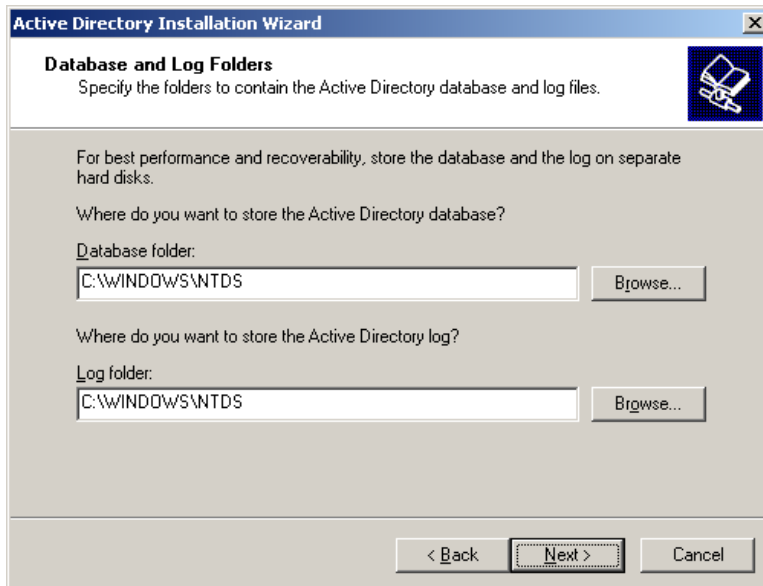
In the next window, you will see the Domain NetBIOS name field. This is the name that users of earlier version of Windows will use to identify the new domain.

Choose the Domain NetBIOS name, and click **Next**.



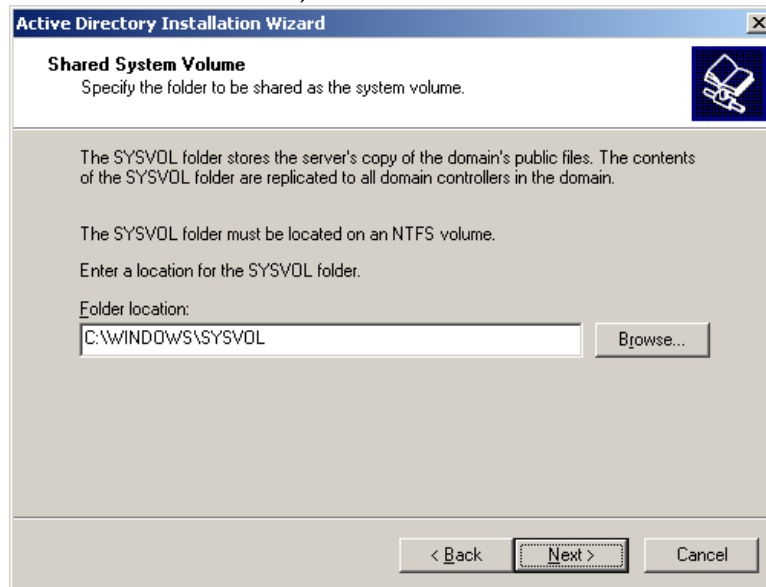
In the next window, you can choose the location where you want to store the database and log files. For best performance, store them on separate disks.

Choose the location where you want to store the Active Directory database and logs, and click **Next**.



The Shared System Volume window appears. Here, choose the location where you want to store the SYSVOL folder. This folder contains the domain public files and is replicated to all the domain controllers in the domain.

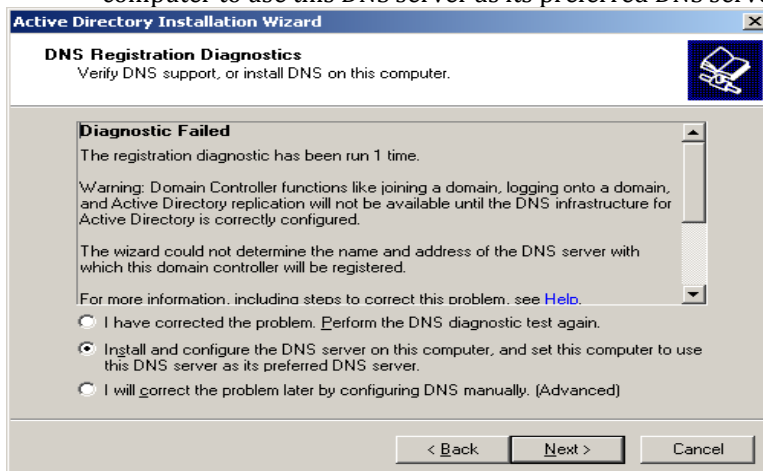
Choose the folder location, and click **Next**.



In the next window, the DNS registration diagnostics appear. Here, you will likely see the message **Diagnostic Failed**.

You have three options.

- 1) "I have corrected the problem. Perform the DNS diagnostic test again."
- 2) "Install and Configure the DNS server on this computer..."
Allow the Active Directory wizard to install and configure DNS for you, and use this DNS as the primary DNS for this server.
- 3) "I will correct the problem later by configuring DNS manually"
Bypass this window if you plan to correct the problem later on.
Choose option two, "Install and configure DNS server on this computer, and set this computer to use this DNS server as its preferred DNS server," and click **Next**.



In the next window, choose what type of permissions you want for users and group objects. Here you have two options:

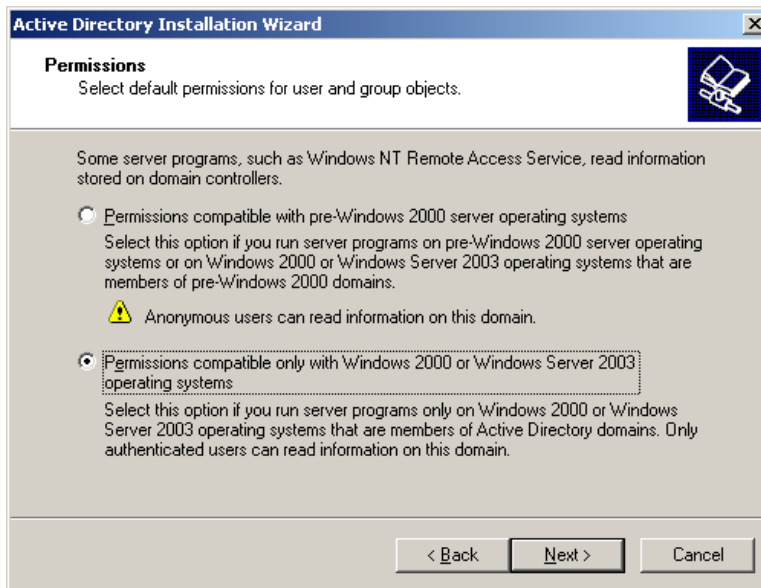
1) Permissions compatible with pre-Windows 2000 server operating systems

Select this option if you run server programs with a version of Windows earlier than Windows 2000.

2) Permissions compatible only with Windows 2000 or Windows 2003 Server operating systems

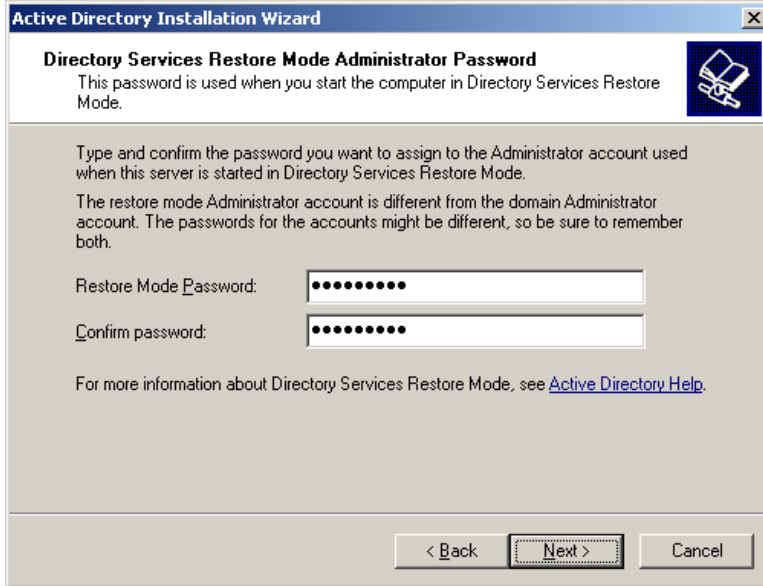
Select this option if you only run Windows Servers 2000 and Windows Servers 2003 on your domain.

Select the second option, "Permissions compatible only with Windows 2000 or Windows 2003 Server operating systems," and click **Next**.



In the next window, enter the **Directory services restore mode administrator password**. This password is used when you start the computer in directory services restore mode. This account is different from the domain administrator account.

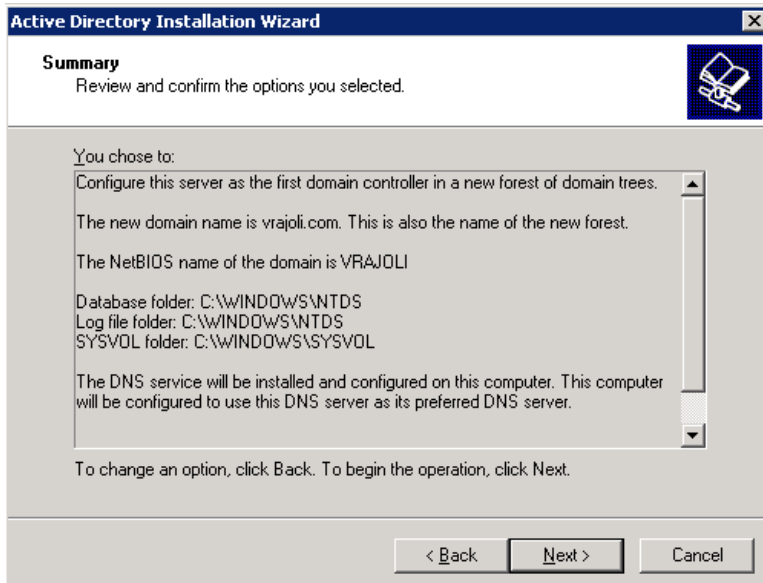
Type your chosen password, and click **Next**.



The screenshot shows the 'Active Directory Installation Wizard' window. The title bar reads 'Active Directory Installation Wizard'. The main heading is 'Directory Services Restore Mode Administrator Password'. Below the heading, there is a sub-heading: 'This password is used when you start the computer in Directory Services Restore Mode.' followed by a small icon of a computer with a key. The main text area contains instructions: 'Type and confirm the password you want to assign to the Administrator account used when this server is started in Directory Services Restore Mode. The restore mode Administrator account is different from the domain Administrator account. The passwords for the accounts might be different, so be sure to remember both.' Below this text are two password input fields. The first is labeled 'Restore Mode Password:' and the second is labeled 'Confirm password:'. Both fields contain a series of black dots. At the bottom of the window, there are three buttons: '< Back', 'Next >', and 'Cancel'. The 'Next >' button is highlighted with a dashed border.

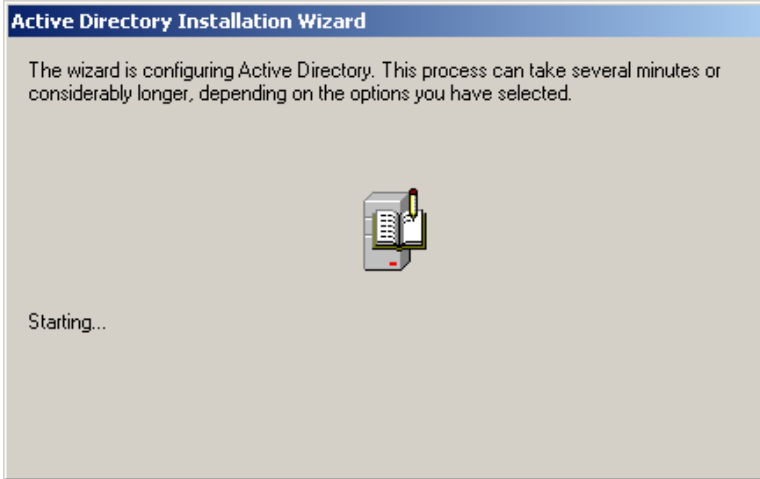
Next, you see the summary of all the options you have chosen in the Active Directory wizard. Remember, the domain administrator account password is the same as the current local administrator password.

Click **Next**.

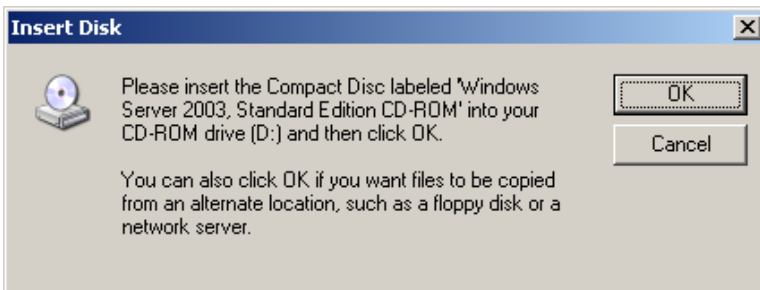


The screenshot shows the 'Active Directory Installation Wizard' window. The title bar reads 'Active Directory Installation Wizard'. The main heading is 'Summary'. Below the heading, there is a sub-heading: 'Review and confirm the options you selected.' followed by a small icon of a computer with a key. The main text area contains a list of options chosen: 'You chose to: Configure this server as the first domain controller in a new forest of domain trees. The new domain name is vrajoli.com. This is also the name of the new forest. The NetBIOS name of the domain is VRAJOLI Database folder: C:\WINDOWS\NTDS Log file folder: C:\WINDOWS\NTDS SYSVOL folder: C:\WINDOWS\SYSVOL The DNS service will be installed and configured on this computer. This computer will be configured to use this DNS server as its preferred DNS server.' Below this text is a scroll bar. At the bottom of the window, there are three buttons: '< Back', 'Next >', and 'Cancel'. The 'Next >' button is highlighted with a dashed border.

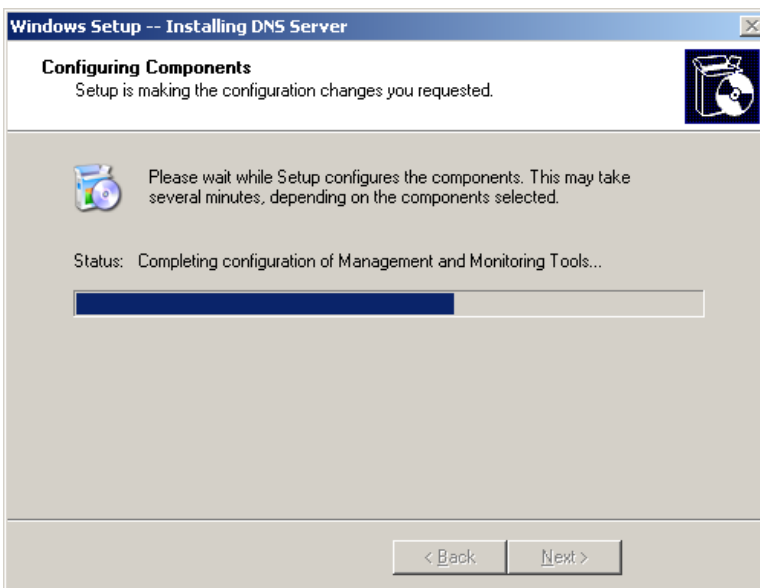
The Active Directory installation begins.



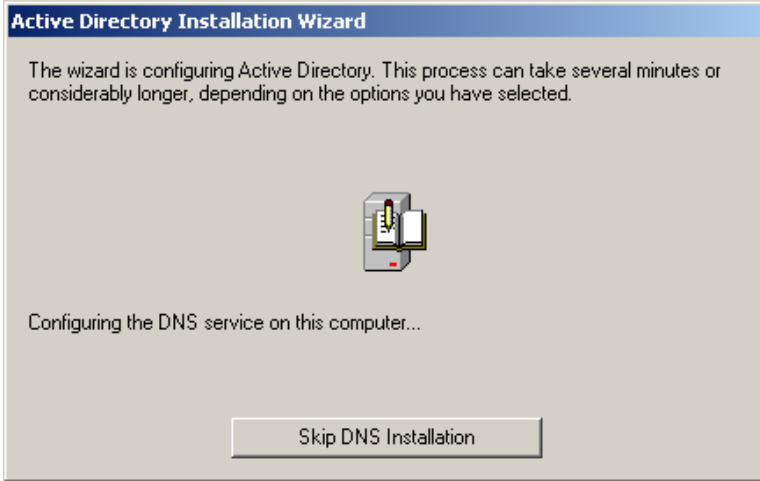
During this process, Active Directory starts installing DNS and prompts you to insert the Windows Server 2003 CD-ROM. Please insert the CD-ROM and click **OK**.



Installation of DNS server begins.

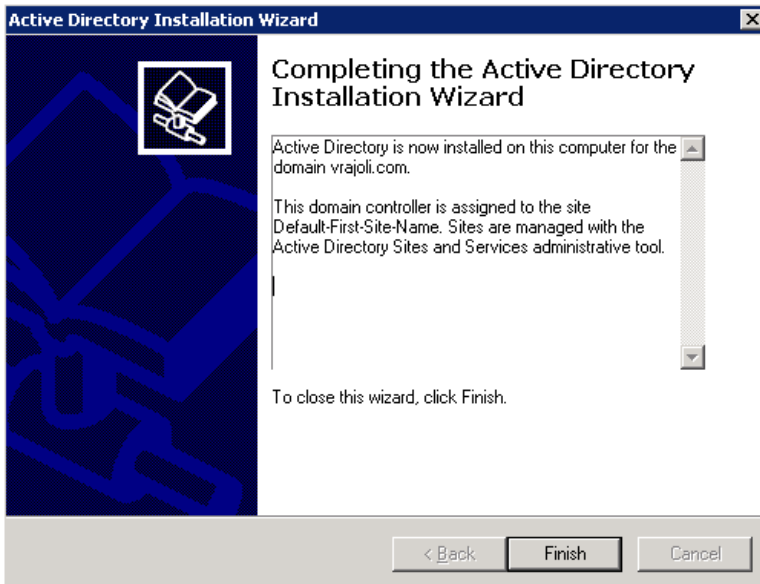


Then, the wizard configures Active Directory.



The following window appears.

Click **Finish**.



Click **Restart Now** to restart the computer.



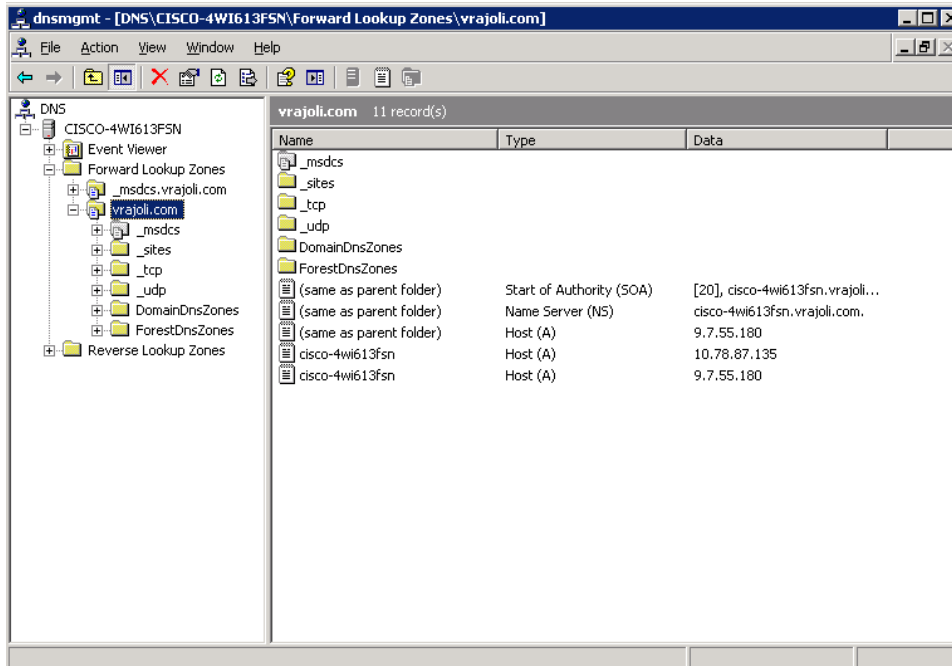
Active Directory is now installed.

3 Configuring DNS on Windows 2003 Server Domain Controller

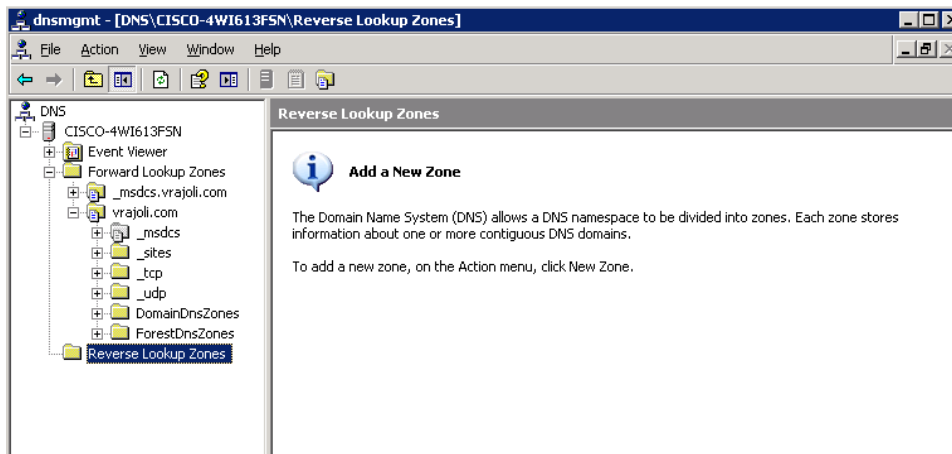
DNS is installed along with Active Directory configuration, only the configuration must be done on the DNS Server.

To configure DNS, go to **Start Menu > Programs > Administrative Tools > DNS**.

The following window appears. In the Forward Lookup Zones, you see the domain controller configured.



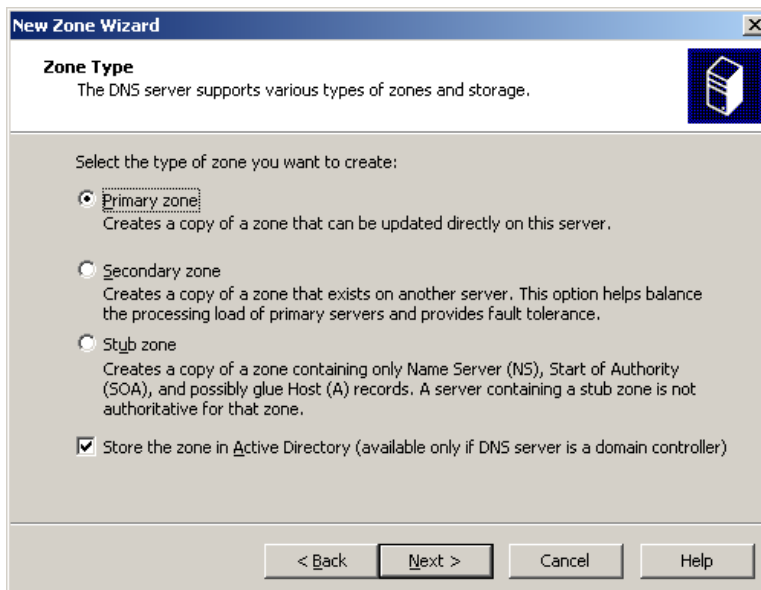
Now you must create the Reverse Lookup Zone. Right-click on Reverse Lookup Zones and click **New Zone**.



The following window appears. Click **Next**.



The Zone Type window appears. Select **Primary zone**.



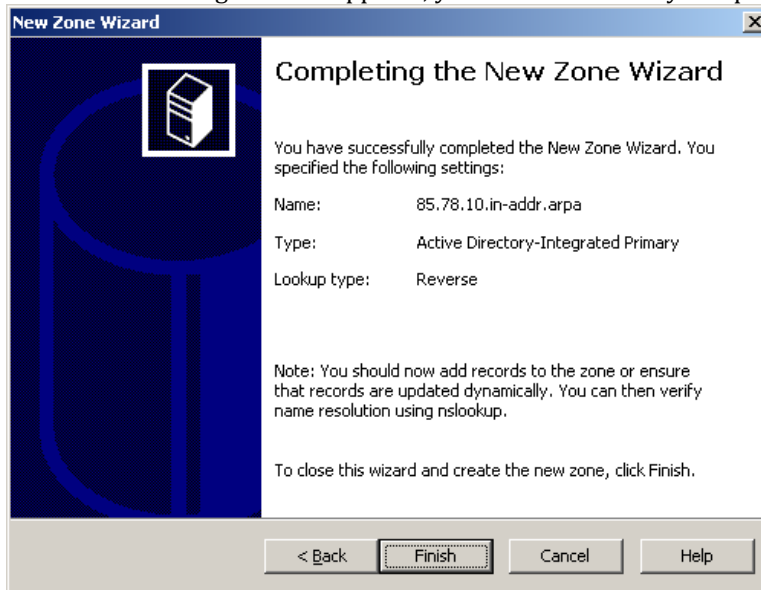
You are asked to configure a Reverse Lookup Zone name. Enter the network ID and then click **Next**.

The screenshot shows the 'New Zone Wizard' dialog box with the title 'Reverse Lookup Zone Name'. Below the title is a sub-header 'Reverse Lookup Zone Name' and a description: 'A reverse lookup zone translates IP addresses into DNS names.' To the right is a blue icon of a book. The main area contains instructions: 'To identify the reverse lookup zone, type the network ID or the name of the zone.' There are two radio buttons: 'Network ID:' (selected) and 'Reverse lookup zone name:'. The 'Network ID' field contains '10 .78 .85 .'. Below it is explanatory text: 'The network ID is the portion of the IP addresses that belongs to this zone. Enter the network ID in its normal (not reversed) order. If you use a zero in the network ID, it will appear in the zone name. For example, network ID 10 would create zone 10.in-addr.arpa, and network ID 10.0 would create zone 0.10.in-addr.arpa.' The 'Reverse lookup zone name' field contains '85.78.10.in-addr.arpa'. At the bottom are four buttons: '< Back', 'Next >', 'Cancel', and 'Help'.

Select the type of dynamic updates that DNS Zone accepts. Click **Allow only Secure dynamic updates** and then click **Next**.

The screenshot shows the 'New Zone Wizard' dialog box with the title 'Dynamic Update'. Below the title is a sub-header 'Dynamic Update' and a description: 'You can specify that this DNS zone accepts secure, nonsecure, or no dynamic updates.' To the right is a blue icon of a book. The main area contains instructions: 'Dynamic updates enable DNS client computers to register and dynamically update their resource records with a DNS server whenever changes occur. Select the type of dynamic updates you want to allow:'. There are three radio buttons: 'Allow only secure dynamic updates (recommended for Active Directory):' (selected), 'Allow both nonsecure and secure dynamic updates', and 'Do not allow dynamic updates'. The first option has a note: 'This option is available only for Active Directory-integrated zones.' The second option has a warning icon and note: 'This option is a significant security vulnerability because updates can be accepted from untrusted sources.' The third option has a note: 'Dynamic updates of resource records are not accepted by this zone. You must update these records manually.' At the bottom are four buttons: '< Back', 'Next >', 'Cancel', and 'Help'.

When the following window appears, you have successfully completed the new zone creation.



Now configure the Local Area Connection properties for this server.

- In the **Local Area Connection Properties** dialog box, click **Internet Protocol (TCP/IP)**, and then click **Properties**.
- In the **Internet Protocols (TCP/IP) Properties** dialog box, click **Use the following IP address**, and then type the static IP address, subnet mask, and default gateway for this server.
- In **Preferred DNS**, type the IP address of this server.
- In **Alternate DNS**, type the IP address of another internal DNS server, or leave this box blank.
- When you finish setting up the static addresses for your DNS, click **OK**, and then click **Close**.

4 Configuring Windows Client Desktop as Domain Computer of Domain Controller

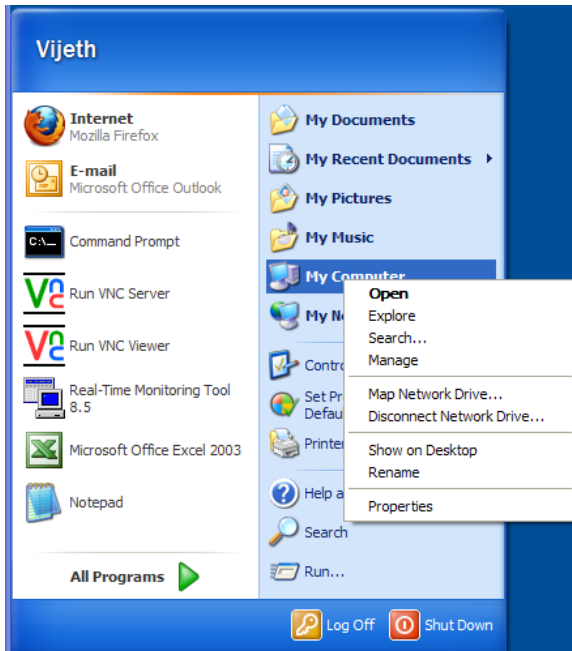
In this example, we use Windows XP desktop for joining the vrajoli.com domain. Create a DNS entry on DNS server for this client host (Windows XP).

Configuring LAN properties for this client desktop

- In the **Local Area Connection Properties** dialog box, click **Internet Protocol (TCP/IP)**, and then click **Properties**.
- In the **Internet Protocols (TCP/IP) Properties** dialog box, click **Use the following IP address**, and then type the static IP address, subnet mask, and default gateway for this server.
- In **Preferred DNS**, type the IP address of DNS server (vrajoli).
- In **Alternate DNS**, type the IP address of another internal DNS server, or leave this box blank.
- When you finish setting up the static addresses for your DNS, click **OK**, and then click **Close**.

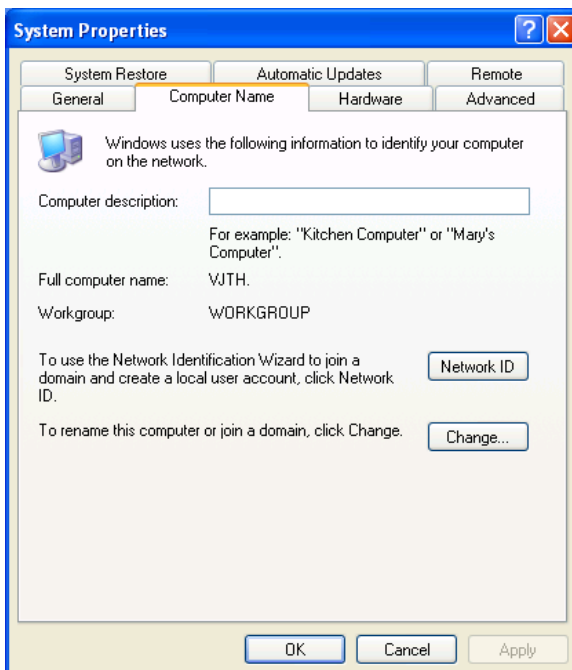
Make a DNS entry for this Client desktop. Open the DNS utility on domain controller (**Administrative Tools > DNS**), in the forward lookup zone, create a new host under the domain name (vrajoli.com). Click the check box to create associated reverse pointer. Verify that Client desktop host was added to DNS.

Right-click **My Computer > Properties**.



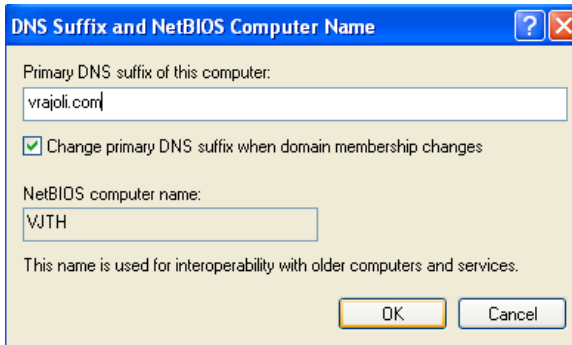
Go to Computer Name tab and click on the **Change** button.

Change the computer name.

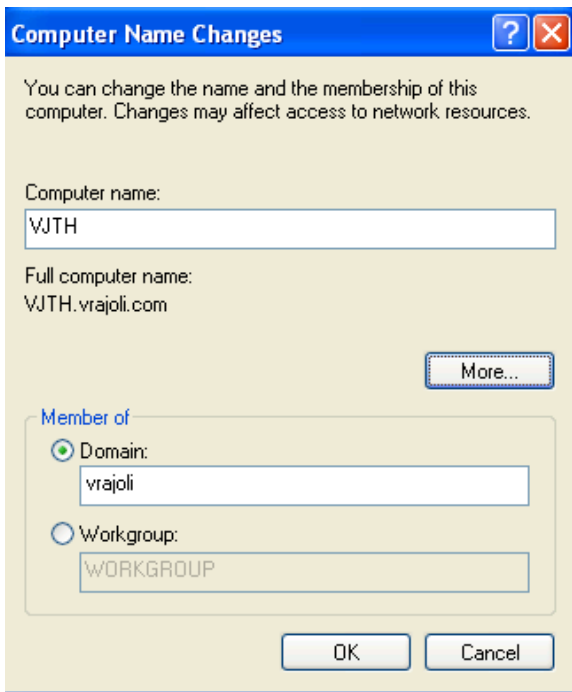


Click the **More** button on the Computer Name Changed window, and enter Primary DNS Suffix of this computer with the DNS name (DNS running on domain controller). In our case the DNS name is "vrajoli.com."

Click **OK**.



Click the Domain radio button then put in your domain name, not including the extension (in this example we use the domain "vrajoli" but when joining the computer to a domain, we will only type "vrajoli").



Then you will be presented with a username and password prompt. Enter the username and password of a domain administrator.

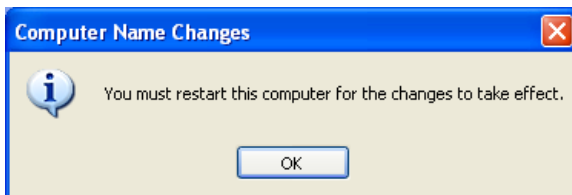
Click **OK**



After a minute or two, you receive a message welcoming you to the domain. Click **OK**.



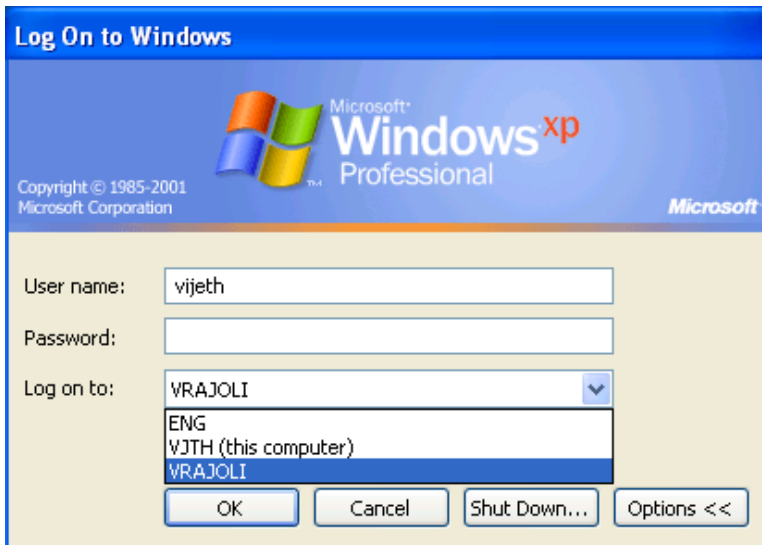
Then you receive a message telling you that a restart is required; click **OK** to restart.



The procedure is complete. You now know how to add a Windows XP computer to a Windows Server 2003 domain.

After the XP computer boots to Control-Alt-Delete, you may need to change it from logging on to itself (which *will* use the local info) to logging on to the domain.

To change to logging on to the domain, press **Ctrl-Alt-Del**, and then click the **Options** button in the logon window. Then select the domain from the drop-down box.



After these steps, you can log on using domain credentials.

5 Brief History of OpenSSO and OpenAM

OpenSSO is an open source access management and federation server platform originally created by Sun Microsystems. The main purpose of OpenSSO is to provide an easy and powerful way to enable using Single Sign-On with many legacy software products. Oracle completed the acquisition of Sun Microsystems in February 2010 and announced that OpenSSO would no longer be their strategic product. OpenSSO will continue to be developed and supported by ForgeRock under the name of OpenAM.

More information about OpenAM is available at <http://www.forgerock.com/openam.html>.

6 System Requirements for OpenAM Installation

Active Directory, domain controller and DNS should be installed and configured before getting started with OpenAM.

6.1 OpenAM Installation on Linux Platform

- Get the compatible hardware for installing RHEL 3/4/5 version (32bit/64bit).
- Install the RHEL and make sure the installation is successful.
- After the successful installation, configure the network settings and get the machine up in the network.
- Configure the hostname for this OpenAM host machine and create a DNS entry for this host on the above DNS setup.

- Configure DNS and domain name, which should point to the above configured AD and DNS setup.
- Install JAVA on this machine. The latest version of JAVA can be downloaded from <http://www.oracle.com/technetwork/java/javase/downloads/index.html>. The latest version of JAVA as of September 13, 2010 is JDK 6 Update 21.
- **Note:** The latest version of java (JDK 6 Update 21) has some issue with Kerberos; please visit <http://forums.sun.com/thread.jspa?threadID=5448003> to know more.
- You can use JDK 6 Update 20, which can be downloaded from <http://java.sun.com/products/archive/j2se/6u20/index.html>.
- Define JAVA_HOME, JRE_HOME environment variables in your user profile (.bash_profile) as below:
 - `export JAVA_HOME=/usr/java/jdk1.6.0_20`
 - `export JRE_HOME=/usr/java/jdk1.6.0_20/jre`
- Create java keystore, which is required for enabling SSL on Tomcat, which is installed and configured in section 7.1.

Execute '**\$JAVA_HOME/bin/keytool -genkey -alias tomcat -keyalg RSA -validity 1825**' command on the terminal, default keystore password is **changeit**.

Next you are prompted to input certain requests. When prompted to "Enter the first name and last name," enter the FQDN(hostname.domainname) of your OpenAM host (ex:vrajlnx.vrajoli.com).

Example:

```
[root@vrajlnx ~]# $JAVA_HOME/bin/keytool -genkey -alias tomcat -keyalg RSA -
validity 1825
```

Enter keystore password:

What is your first and last name?

[Unknown]: **vrajlnx.vrajoli.com**

What is the name of your organizational unit?

[Unknown]: **IPCBU**

What is the name of your organization?

[Unknown]: **CSIPL**

What is the name of your City or Locality?

[Unknown]: **Bengaluru**

What is the name of your State or Province?

[Unknown]: **Karnataka**

What is the two-letter country code for this unit?

[Unknown]: **IN**

Is CN=vrajlnx.vrajoli.com, OU=IPCBU, O=CSIPL, L=Bengaluru, ST=Karnataka, C=IN correct?

[no]: **yes**

Enter key password for <tomcat>

(RETURN if same as keystore password):

[root@vrajlnx ~]#

6.2 OpenAM Installation on Windows Platform

- Install Windows OS (XP/2003 Server/2008 Server/Windows7/Vista).
- After successful installation, configure the network settings and get the machine up in the network.
- Join this computer to the domain controller (ex: vrajoli.com), refer to the section 4 for the procedure to be followed for joining the computer to the domain controller.
- Install JAVA on this machine. The latest version of JAVA can be downloaded from <http://www.oracle.com/technetwork/java/javase/downloads/index.html>. The latest version of JAVA as of September 13, 2010 is JDK 6 Update 21.
- **Note:** The latest version of java (JDK 6 Update 21) has some issue with Kerberos; please visit <http://forums.sun.com/thread.jspa?threadID=5448003> to know more.
- You can use JDK 6 Update 20, which can be downloaded from <http://java.sun.com/products/archive/j2se/6u20/index.html>.
- Create java keystore, which is required for enabling SSL on Tomcat, which is installed and configured in section 7.2.
- Open the command prompt and execute the following command. In this test setup, JAVA is installed under *c:\Program Files\Java*. Please enter the right path of keytool.exe in your setup when executing this command. The default keystore password is **changeit**

```
C:\>"c:\Program Files\Java\jdk1.6.0_20\bin\keytool.exe" -genkey -alias tomcat  
-keyalg RSA -validity 1825 -keystore c:\keystore
```

Enter keystore password:

What is your first and last name?

[Unknown]: vrajlnx.vrajoli.com

What is the name of your organizational unit?

[Unknown]: IPCBU

What is the name of your organization?

[Unknown]: CSIPL

What is the name of your City or Locality?

[Unknown]: Bengaluru

What is the name of your State or Province?

[Unknown]: Karnataka

What is the two-letter country code for this unit?

[Unknown]: IN

*Is CN=vrajlnx.vrajoli.com, OU=IPCBU, O=CSIPL, L=Bengaluru, ST=Karnataka, C=IN
corr*

ect?

[no]: yes

Enter key password for <tomcat>

(RETURN if same as keystore password):

Keystore will be created under c:\>.

7 Installation and Configuration of Apache Tomcat with HTTPS

7.1 Installation and configuration of Apache Tomcat on Linux platform

- Download the latest version of Apache Tomcat; refer to <http://tomcat.apache.org/index.html> for the latest version; download the zip/tar archives specific to your processor architecture (32bit/64bit). We use apache-tomcat-7.0.0 in this guide.
- Copy the downloaded apache-tomcat-7.0.0.tar.gz to the specific location on the OpenAM server that was set up in section 6.1.
- Extract the apache-tomcat-7.0.0.tar.gz archive. In this guide, we are extracting it under root home directory (/root).
- Increase the JVM heap size on Tomcat by setting `JAVA_OPTS="$JAVA_OPTS -Xmx1024m -XX:MaxPermSize=256m -Xms512m` property in the catalina.sh under /root /apache-tomcat-7.0.0/bin directory.

Example: JAVA_OPTS="\$JAVA_OPTS -Xmx1024m -XX:MaxPermSize=256m -Xms512m -Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager"

- Open the **server.xml** file under /root/apache-tomcat-7.0.0/conf directory,
 - Comment the 8080 connector port: Make the code read as below

```
<!-- <Connector port="8080" protocol="HTTP/1.1"
connectionTimeout="20000"
redirectPort="8443" /> -->
```
 - Uncomment the 8443 connector port: Remove <!-- code at the beginning and --> at the end of 8443 connector, make the code read as below

```
<Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"
maxThreads="150" scheme="https" secure="true"
clientAuth="false" sslProtocol="TLS" />
```

Save the **server.xml** file after making the above changes.

- Start the Tomcat by executing startup.sh under /root/ apache-tomcat-7.0.0/bin directory.
- Launch a browser and go to: **https://localhost:8443/tomcat.gif**. If your certificate is self-signed, your browser warns you. Instruct the browser to import the

certificate and proceed. You should then see the little Tomcat logo. If you do, Tomcat is configured.

7.2 Installation and Configuration of Apache Tomcat on Windows Platform

- Download the latest version of Apache Tomcat for windows platform; refer to <http://tomcat.apache.org/index.html> for the latest version. Download the Tomcat service installer (32-bit/64-bit Windows Service Installer - apache-tomcat-7.0.0.exe). We use apache-tomcat-7.0.0 in this guide.
- Install the apache-tomcat-7.0.0.exe. In this guide, Tomcat is installed under *c:\Program Files\Apache Software Foundation\Tomcat 7.0*.
- Set the JAVA_HOME, JRE_HOME and JAVA_OPTS environment variables by creating a file called **setenv.bat** under *c:\Program Files\Apache Software Foundation\Tomcat 7.0\bin* and set the above variables.

Content of **setenv.bat** in testing this guide:

```
set JAVA_HOME=c:\Program Files\Java\jdk1.6.0_20
set JRE_HOME=c:\Program Files\Java\jdk1.6.0_20\jre
set JAVA_OPTS=%JAVA_OPTS% -Xms512m -Xmx1024m
```

- Open the **server.xml** file under **c:\Program Files\Apache Software Foundation\Tomcat 7.0\conf** folder:

- Comment the 8080 connector port:

Make the code read as below

```
<!-- <Connector port="8080" protocol="HTTP/1.1"
connectionTimeout="20000"
redirectPort="8443" /> -->
```

- Uncomment the 8443 connector port:

Remove **<!--** code at the beginning and **-->** at the end of 8443 connector, in this 8443 connector we have added two more attributes: **keystoreFile** (location of the keystore file that was created in section 6.2, in this test it was created under C:\keystore) and **keystoreType**. Because we have keystore created with default password **changeit**, no need to set keystorePass attribute. Make the code read as below:

```
<Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"
maxThreads="150" scheme="https" secure="true"
clientAuth="false" sslProtocol="TLS"
keystoreFile="C:\keystore"
keystoreType="JKS" />
```

Save the **server.xml** file after making the above changes.

- Start the Tomcat service from services.msc utility or from Administrative Tools > Services > Apache Tomcat 7 > Start.

- Launch a browser and go to **https://localhost:8443/tomcat.gif**. If your certificate is self-signed, your browser warns you. Instruct the browser to import the certificate and proceed. You should then see the little Tomcat logo. If you do, Tomcat is configured.

8 Provisioning Active Directory for Single Sign-On

- Log on to AD Server.
- From the Start menu, go to **Programs > Administration Tools**. Select **Active Directory Users and Computers**.
- Go to **Users > New > Users** and create a new user with the OpenSSO Enterprise host name as the User ID (example: *vrajlnx*).
- The OpenSSO Enterprise hostname should not include the domain name.
- Create a keytab file on the AD server using the following command from the command prompt.

```
ktpass -princ HTTP/hostname.domainname@DCDOMAIN -pass password -mapuser userName-out hostname.HTTP.keytab -ptype KRB5_NT_PRINCIPAL -target DCDOMAIN
```

Example:

```
ktpass -princ HTTP/vrajlnx.vrajoli.com@VRAJOLI.COM -pass <password> -mapuser vrajlnx -out vrajlnx.HTTP.keytab -ptype KRB5_NT_PRINCIPAL -target VRAJOLI.COM
```

- After successful creation of the keytab file, copy the keytab file to a location on the OpenAM server; this path will later be specified in OpenAM configuration.

For OpenAM configured on Linux, you can create a directory under root, and copy the above keytab file. *Example: /root/keytab/ vrajlnx.HTTP.keytab*

For OpenAM configured on Windows, you can create a directory under C:\>, and copy the above keytab file. *Example: c:/keytab/ vrajlnx.HTTP.keytab*

9 Deploying OpenSSO (OpenAM) Enterprise on Apache Tomcat

9.1 Deploying OpenSSO Enterprise War on Apache Tomcat over Linux platform

- Go to the ForgeRock site <http://www.forgerock.org/openam.html> and download the stable release of any OpenAM version supported.

Tomcat Versions supported:	TomcatV6, TomcatV7
OpenAM Versions supported:	OpenAM 9.5.4, OpenAM 10.0

- Copy the openam .war file to OpenAM server on any location and unzip it.
- Stop the Tomcat service if it is running on this OpenAM server.

- After unzipping, copy the file named opensso.war or openam.war under opensso/deployable-war directory and paste it under /root /apache-tomcat-7.0.0/webapps directory.
- Start the Tomcat by executing startup.sh under /root/ apache-tomcat-7.0.0/bin directory.

9.2 Deploying OpenSSO Enterprise War on Apache Tomcat over Windows Platform

- Go to the ForgeRock site <http://www.forgerock.org/openam.html> and download the stable release of any OpenAM version supported.

Tomcat Versions supported:	TomcatV6, TomcatV7
OpenAM Versions supported:	OpenAM 9.5.4, OpenAM 10.0

- Copy the openam .war file to OpenAM server on any location and unzip it.
- Stop the Tomcat service if running on this OpenAM server. (Administrative Tools > Services > Apache Tomcat 7 > Stop)
- After unzipping, copy opensso.war file under opensso\deployable-war folder and paste it under *c:\Program Files\Apache Software Foundation\Tomcat 7.0\webapps* folder.
- Start the Tomcat from Administrative Tools > Services > Apache Tomcat 7 > Start

10 Configuring OpenSSO Enterprise Using the GUI Configurator

OpenAM server and policy agents require FQDNs for the hostname of the machines where you will do your installations. You can **NOT** use a host name like "localhost" and can **NOT** use numeric IP addresses like "192.168.1.2" as hostnames either, because this will **cause problems** in installation, configuration and usage.

When accessing the OpenAM for the first time, you should use FQDN of OpenAM server in the URL (<https://vrajlnx.vrajoli.com:8443/opensso>). When you access OpenSSO Enterprise for the first time, you are directed to the Configurator to perform the OpenSSO Enterprise initial configuration.

The Configuration Options window appears when you access the OpenSSO for the first time.

Select the configuration option:

- **Default Configuration:** You specify and confirm passwords for the OpenSSO Enterprise administrator (amAdmin) and the default policy agent user (UrlAccessAgent), which is the policy agent user that connects to OpenSSO Enterprise server. The Configurator uses default values for the other configuration settings.

The default policy agent user is also referred to as an application user. This user can connect to OpenSSO Enterprise server from a client such as the client SDK or a distributed authentication UI server.

Choose Default Configuration for development environments or simple demonstration purposes when you just want to evaluate OpenSSO Enterprise features. Click **Create Default Configuration** and continue with Configuring OpenSSO Enterprise with the Default Configuration.

OR

- **Custom Configuration:** You specify the configuration settings that meet the specific requirements for your deployment (or accept the default settings). Choose Custom Configuration for production and more complex environments, for example, a multiserver installation with several OpenSSO Enterprise instances behind a load balancer. Click **Create New Configuration** and continue with Configuring OpenSSO Enterprise with a Custom Configuration.

For Custom configuration, refer to Chapter 14.



In this section, we select Default Configuration. When the configuration is complete, the Configurator displays a link to the OpenSSO.

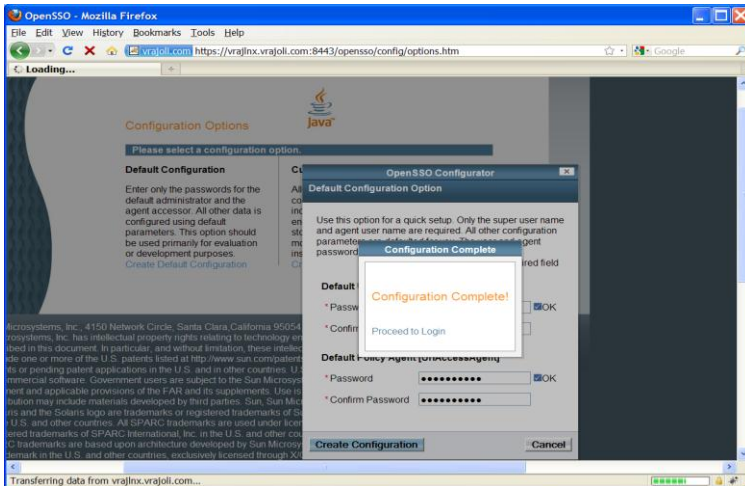
Use the Enterprise Administration Console to perform any additional configuration required for your deployment.

If a problem occurred during the configuration, the Configurator displays an error message. If you can, correct the error and retry the configuration.

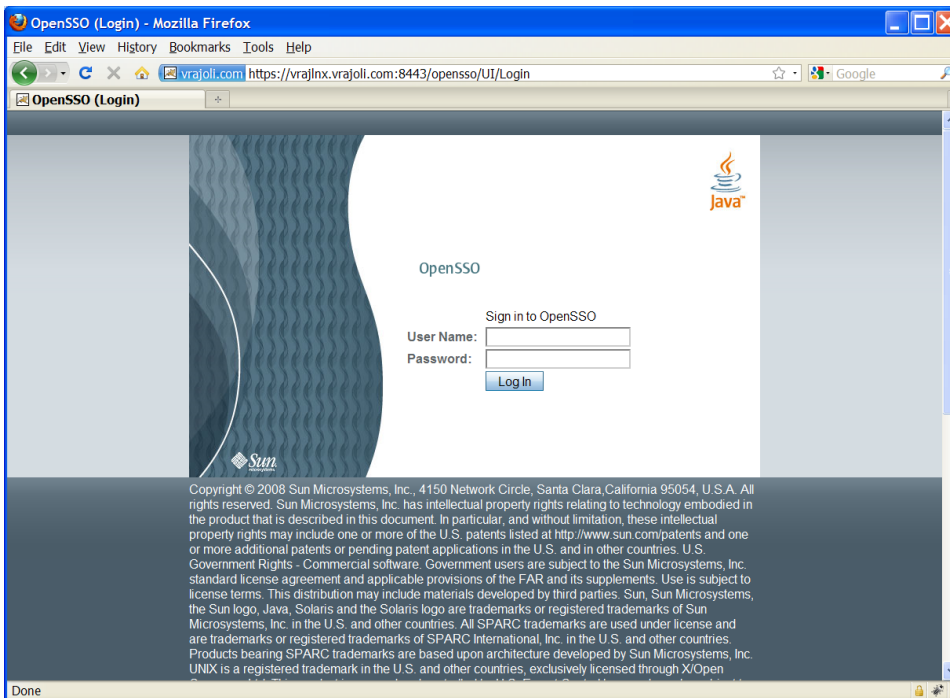
Also, check the web container log files and the install.log, which if created, will be in the configuration directory (default /opensso). These logs might contain information about the cause of a configuration problem.

By default, OpenSSO is deployed under /root/opensso directory on Linux platform; on Windows platform OpenSSO is deployed under C:\opensso.

Click **Proceed** to log in.

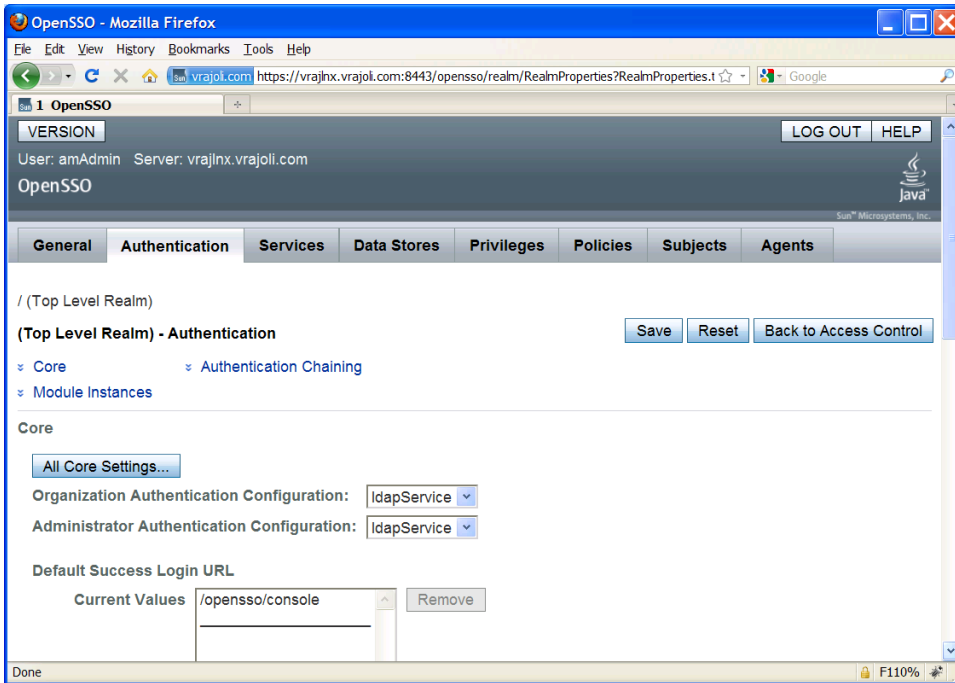


The following window appears. Log in to the OpenSSO server with the amAdmin username and password you created during the default configuration.

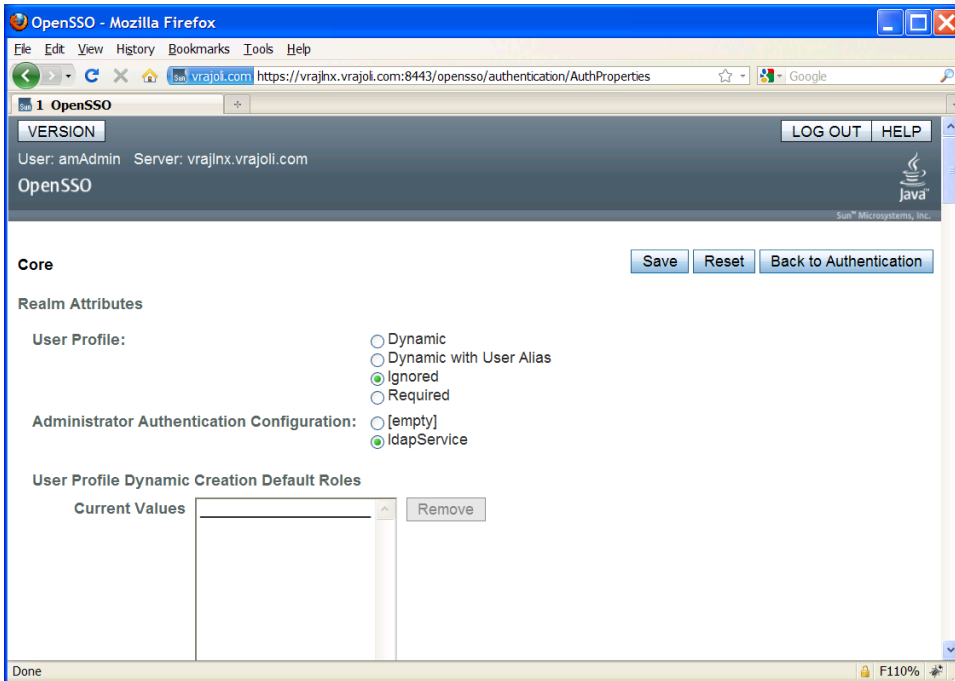


Go to Access Control tab and click on / **(Top Level Realm)**, then go to the Authentication tab as shown below.

Click the **All Core Settings** button.



Set the User Profile to **Ignored**. Click the **Save** button to save the configuration.



10.1 Configuring Policies on OpenSSO Server

10.1.1 Configuring Policies on OpenSSO Server for Cisco Unified Communications Manager 8.5, 8.6

- Log in to OpenSSO server with amAdmin username and password
- Enable ssoadm tool from UI

Go to Configuration-> Servers and Sites -> click on the listed server name.

The screenshot shows the OpenAM user interface. At the top, there is a header with "VERSION" on the left and "LOG OUT" on the right. Below the header, the user information "User: amAdmin Server: openamlnx.cucmsso.com" is displayed. The OpenAM logo is visible. A navigation menu is shown with tabs for "Common Tasks", "Access Control", "Federation", "Web Services", "Configuration", and "Sessions". Under the "Configuration" tab, there are sub-tabs for "Authentication", "Console", "System", "Global", and "Servers and Sites".

Servers and Sites

The screenshot shows the "Servers and Sites" configuration page. It starts with a "Default Server Settings" button. Below that, there is a section for "Servers (1 Item(s))" with "New...", "Delete", and "Clone..." buttons. A table lists the server with columns for "Server Name" and "Site Name". The server name is "https://openamlnx.cucmsso.com:8443/openam". Below the servers section, there is a section for "Sites (0 Item(s))" with "New..." and "Delete" buttons. A table with columns "Site Name", "Primary URL", and "Assigned Servers" is shown, with the message "There are no sites." below it.

Click on Advanced tab. Now add Advance Properties

The screenshot shows the OpenAM user interface with the "Advanced" configuration tab selected. The header and user information are the same as in the previous screenshot. The navigation menu shows "General", "Security", "Session", "SDK", "Directory Configuration", and "Advanced".

Edit <https://openamlnx.cucmsso.com:8443/openam>

[Save](#) [Reset](#) [Back to Servers and Sites](#)


The screenshot shows the "Advanced Properties (10 Item(s))" configuration page. It has "Add..." and "Delete" buttons. A table lists the properties with columns for "Property Name" and "Property Value".

Property Name	Property Value
<input type="checkbox"/> com.ipplanetam.lbcookie.value	01
<input type="checkbox"/> com.ipplanetam.security.SSLSocketFactoryImpl	com.sun.identity.shared.idap.factory.JSSESocketFa
<input type="checkbox"/> com.sun.embedded.replicationport	
<input type="checkbox"/> com.sun.embedded.sync.servers	0n
<input type="checkbox"/> com.sun.identity.common.systemtimerpool.size	3
<input type="checkbox"/> com.sun.identity.sm.sms_object_class_name	com.sun.identity.sm.Idap.SMSEmbeddedLdapObj
<input type="checkbox"/> com.sun.identity.urlconnection.useCache	false
<input type="checkbox"/> opensso.protocol.handler.pkgs	
<input type="checkbox"/> org.forgerock.embedded.dsadminport	4444

add Advance Properties ssoadm.disabled = false

VERSION LOG OUT

User: amAdmin Server: openamlnx.cucmsso.com



General Security Session SDK Directory Configuration **Advanced**

Edit <https://openamlnx.cucmsso.com:8443/openam>

Advanced Properties (10 Item(s))

Property Name	Property Value
<input type="checkbox"/> com.iplanet.am.lbcookie.value	01
<input type="checkbox"/> com.iplanet.security.SSLSocketFactoryImpl	com.sun.identity.shared.Idap.factory.JSSESocketFa
<input type="checkbox"/> com.sun.embedded.replicationport	
<input type="checkbox"/> com.sun.embedded.sync.servers	on
<input type="checkbox"/> com.sun.identity.common.systemtimerpool.size	3
<input type="checkbox"/> com.sun.identity.sm.sms_object_class_name	com.sun.identity.sm.Idap.SMSEmbeddedLdapObj
<input type="checkbox"/> com.sun.identity.urlconnection.useCache	false
<input type="checkbox"/> opensso.protocol.handler.pkgs	
<input type="checkbox"/> org.forgerock.embedded.dsadminport	4444
<input type="checkbox"/> ssoadm.disabled	false

- Browse for <https://<host FQDN>:8443/opensso/ssoadm.jsp> link
- select sub-command "add-attrs"



```
add-agent-to-grp
    Add agents to a agent group.

add-amsdk-idrepo-plugin
    Create AMSDK IdRepo Plug-in

add-app-priv
    Add an application privilege to delegate resources of a given
    application.

add-attr-defs
    Add default attribute values in schema.

add-attrs
    Add attribute schema to an existing service.

add-auth-cfg-entr
    Add authentication configuration entry

add-cot-member
    Add a member to a circle of trust.

add-member
    Add an identity as member of another identity

add-plugin-interface
    Add Plug-in interface to service.

add-plugin-schema
    Add Plug-in schema to service.
```

Give value as Service Name= iPlanetAMWebAgentService , Schema Type=policy ,Attribute Schema XML =

```
-----
    <ServicesConfiguration>
<Service name="iPlanetAMWebAgentService" version="1.0">
<Schema
i18nFileName="amWebAgent"
i18nKey="iplanet-am-web-agent-service-description">
<Policy>
    <AttributeSchema name="PUT"
        type="single"
        syntax="boolean"
        uitype="radio"
        i18nKey="PUT">
    <IsResourceNameAllowed/>
    <BooleanValues>
        <BooleanTrueValue i18nKey="allow">allow</BooleanTrueValue>
        <BooleanFalseValue i18nKey="deny">deny</BooleanFalseValue>
    </BooleanValues>
    </AttributeSchema>
    <AttributeSchema name="DELETE"
        type="single"
        syntax="boolean"
        uitype="radio"
        i18nKey="DELETE">
    <IsResourceNameAllowed/>
    <BooleanValues>
        <BooleanTrueValue i18nKey="allow">allow</BooleanTrueValue>
        <BooleanFalseValue i18nKey="deny">deny</BooleanFalseValue>
```

```
</BooleanValues>
</AttributeSchema>
</Policy>
</Schema>
</Service>
</ServicesConfiguration>
```



[Back to main page.](#)

Sub Command, add-attrs
Add attribute schema to an existing service.

Service Name*:

Schema Type*:

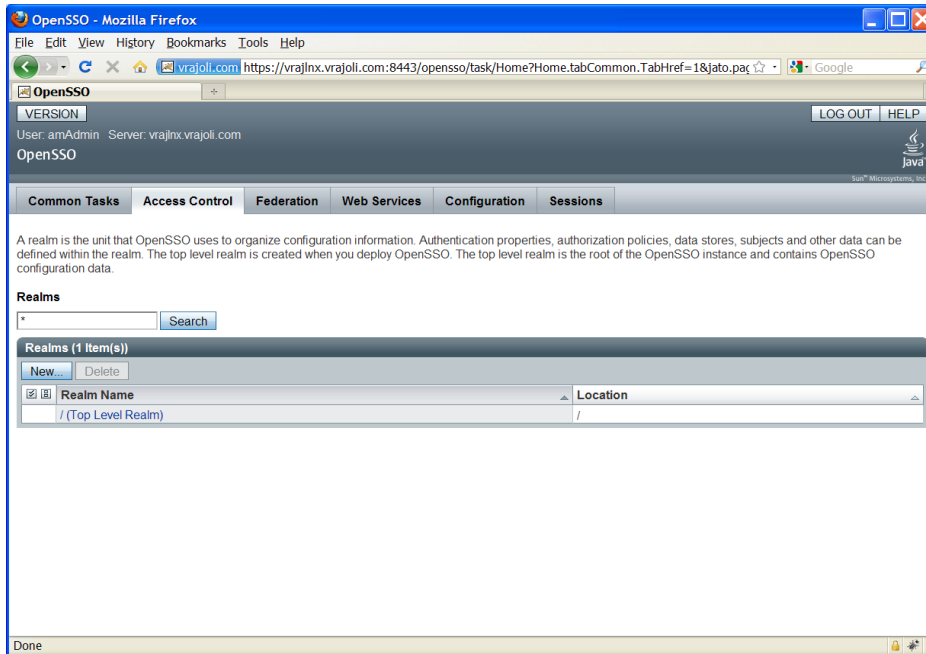
Attribute Schema XML*:

```
<ServicesConfiguration>
  <Service name="iPlanetAMWebAgentService" version="1.0">
    <Schema
      i18nFileName="amWebAgent"
      i18nKey="iPlanetAMWebAgentServiceDescription">
      <Policy>
        <AttributeSchema name="PUT"
          type="single"
          syntax="boolean"
          uiType="radio"
          i18nKey="PUT">
          <IsResourceNameAllowed/>
          <BooleanValues>
            <BooleanTrueValue
              i18nKey="allow">allow</BooleanTrueValue>
            <BooleanFalseValue
              i18nKey="deny">deny</BooleanFalseValue>
          </BooleanValues>
        </AttributeSchema>
        <AttributeSchema name="DELETE"
          type="single"
          syntax="boolean"
          uiType="radio"
          i18nKey="DELETE">
          <IsResourceNameAllowed/>
          <BooleanValues>
            <BooleanTrueValue
              i18nKey="allow">allow</BooleanTrueValue>
            <BooleanFalseValue
              i18nKey="deny">deny</BooleanFalseValue>
          </BooleanValues>
        </AttributeSchema>
      </Policy>
    </Schema>
  </Service>
</ServicesConfiguration>
```

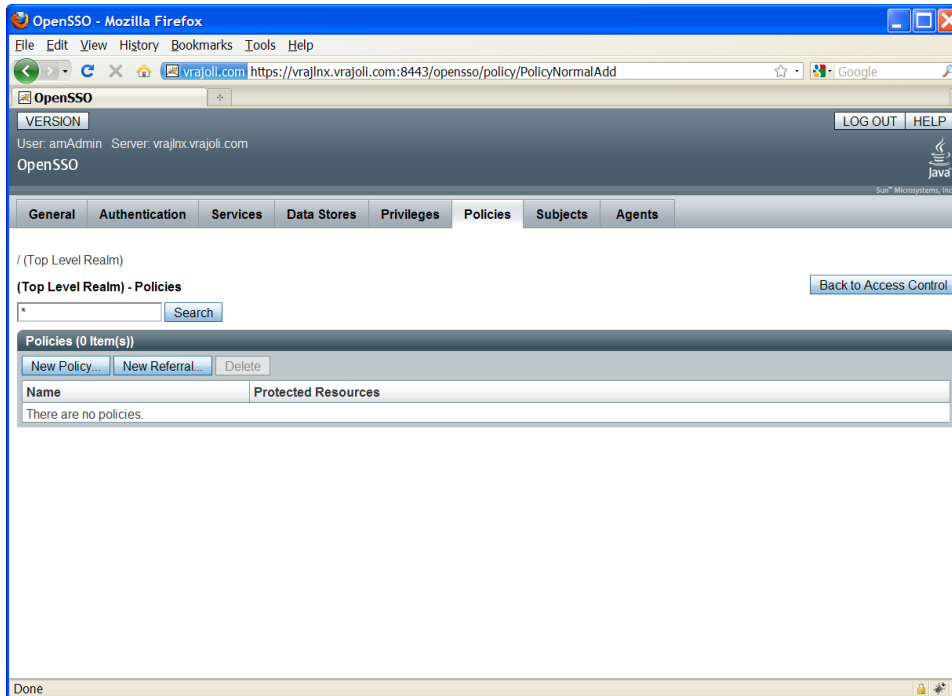
Name of sub schema:

- Restart Tomcat.

Go to Access Control tab and click on / **(Top Level Realm)**. The following window appears.



Go to the Policies tab and add a new policy; enter the PolicyName.



Create a new rule from the Policy Configuration window.

In the window that appears, select service type as **URL Policy Agent (with resource name)**.

Enter the Rule Name and Resource URL as Web Application URL. In our case it is CUCMUser application URL (https://<CUCM FQDN>:8443/*).

Check the **GET** , **POST** , **PUT** and **DELETE** check boxes and then click **Finish**.

OpenSSO - Mozilla Firefox

File Edit View History Bookmarks Tools Help

vrjoli.com https://vrajlnx.vrajoli.com:8443/opensso/policy/SelectServiceType Google

OpenSSO

VERSION LOG OUT HELP

User: amAdmin Server: vrajlnx.vrajoli.com

OpenSSO

Sun Microsystems, Inc.

Step 2 of 2: New Rule

Back Finish Cancel

* Indicates required field

* Service Type: URL Policy Agent

* Name: CUCMUser

* Resource Name: https://vrajolicucm1.vrajoli.com:8443/*

Actions

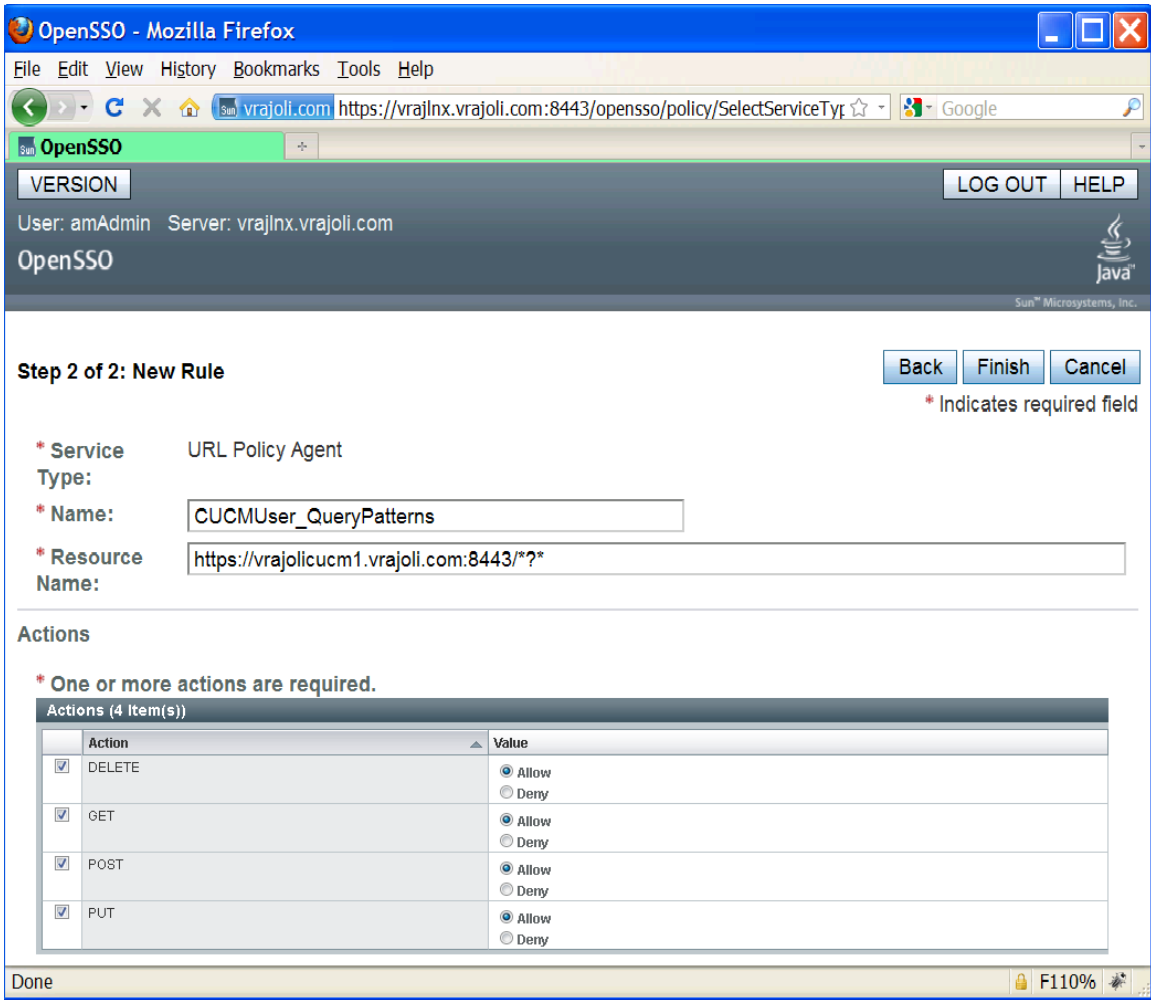
* One or more actions are required.

Actions (4 Item(s))

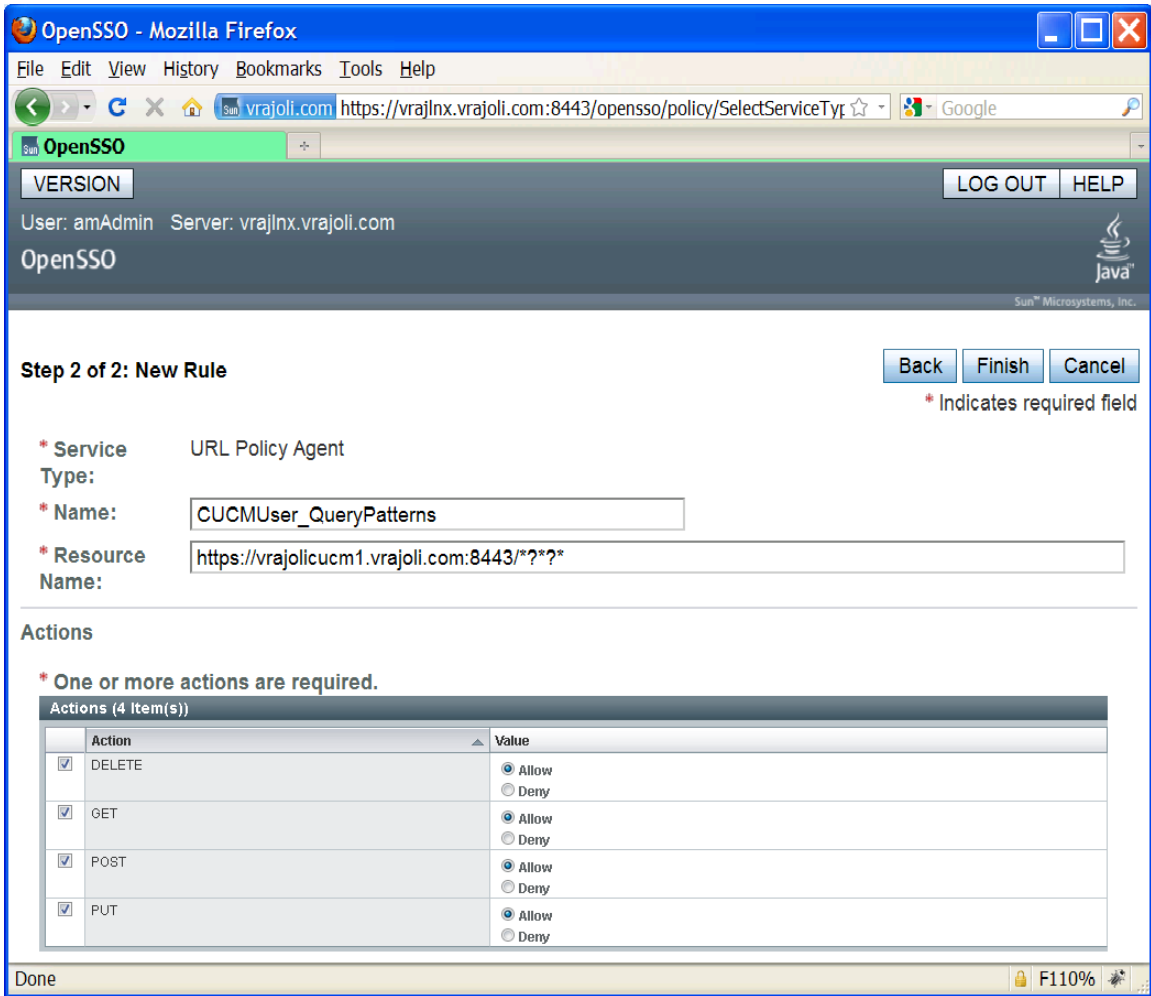
Action	Value
<input checked="" type="checkbox"/> DELETE	<input checked="" type="radio"/> Allow <input type="radio"/> Deny
<input checked="" type="checkbox"/> GET	<input checked="" type="radio"/> Allow <input type="radio"/> Deny
<input checked="" type="checkbox"/> POST	<input checked="" type="radio"/> Allow <input type="radio"/> Deny
<input checked="" type="checkbox"/> PUT	<input checked="" type="radio"/> Allow <input type="radio"/> Deny

Done F110%

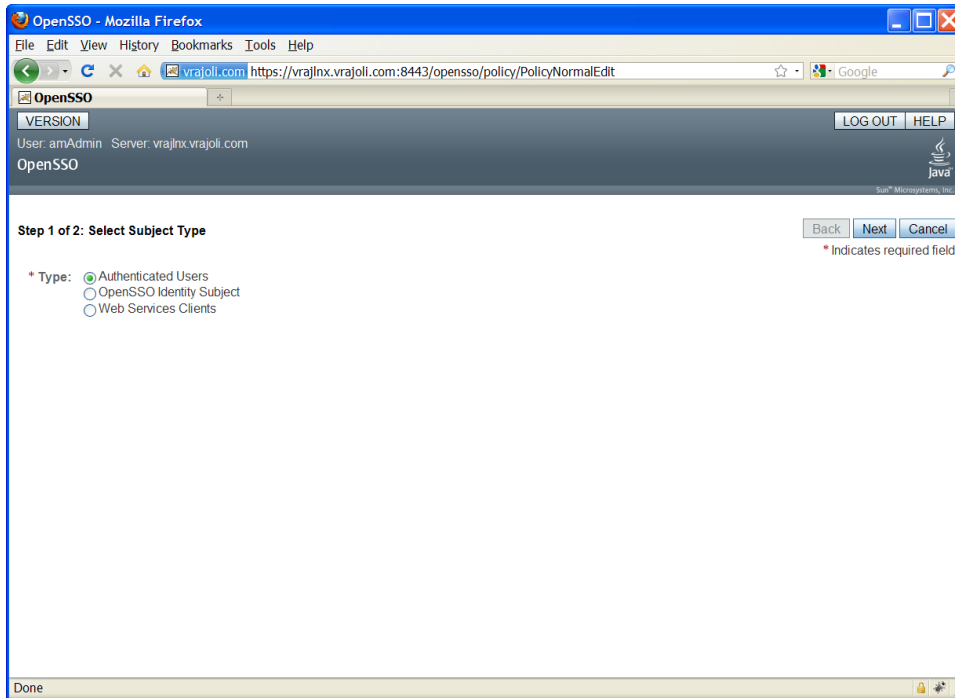
Create another rule for the requests involving query patterns (*?*) in Find and List windows of CUCMUser application.



From Cisco Unified Communications Manager Release 8.6, SSO support is provided for RTMT application as well. To achieve SSO for RTMT, along with the above policy rules, you should create one more new rule for requests involving RTMT query patterns (*?*?*).



Click the **New** button under Subjects on the Policy Configuration window. Select subject type as **Authenticated Users**.



Enter the Subject Name and Click **Finish**.

(Optional) If 443 re-director port is used to access CUCM web applications

Create a new rule from the Policy Configuration window.

In the window that appears, select service type as **URL Policy Agent (with resource name)**.

Enter the Rule Name and Resource URL as Web Application URL `https://<CUCM:FQDN>/*`.

Check the **GET , POST , PUT and DELETE** check boxes and then click **Finish**.

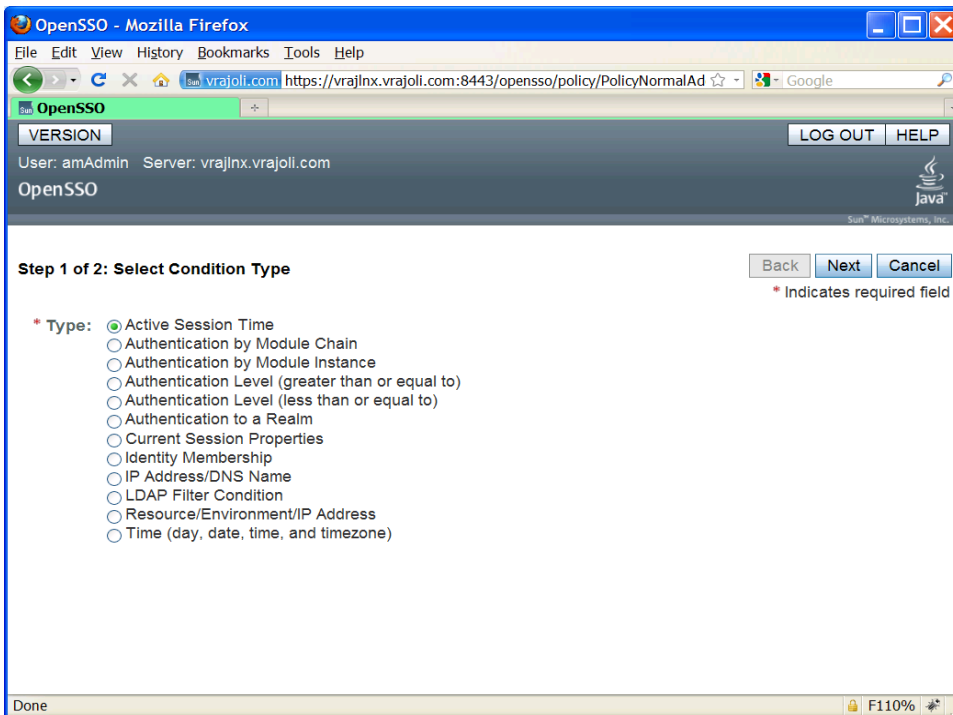
Repeat the same steps for creating two more policies with Web Application URL `https://<CUCM:FQDN>/?**` and `https://<CUCM:FQDN>/?*?**`.

In our case, policies mentioned below needs to be created.

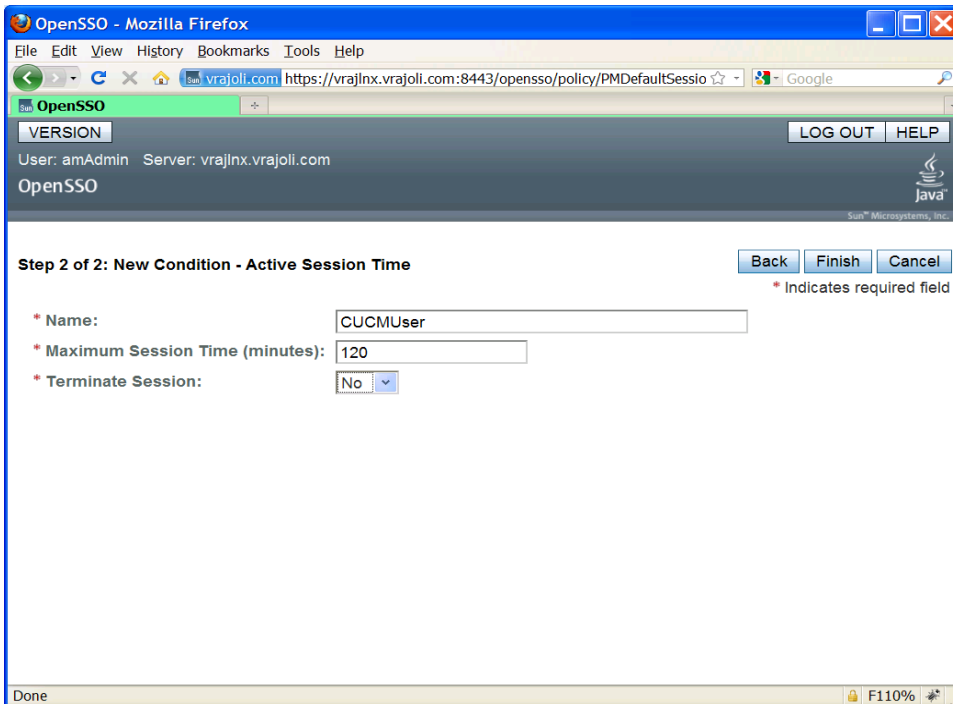
<https://vrajolicucm1.vrajoli.com/>
https://vrajolicucm1.vrajoli.com/?*
https://vrajolicucm1.vrajoli.com/?*?*

Now Policy is created with defining Rules and Subjects.

Click the **New** button under Conditions. Under select condition type, select **Active Session Time** and then click **Next**.

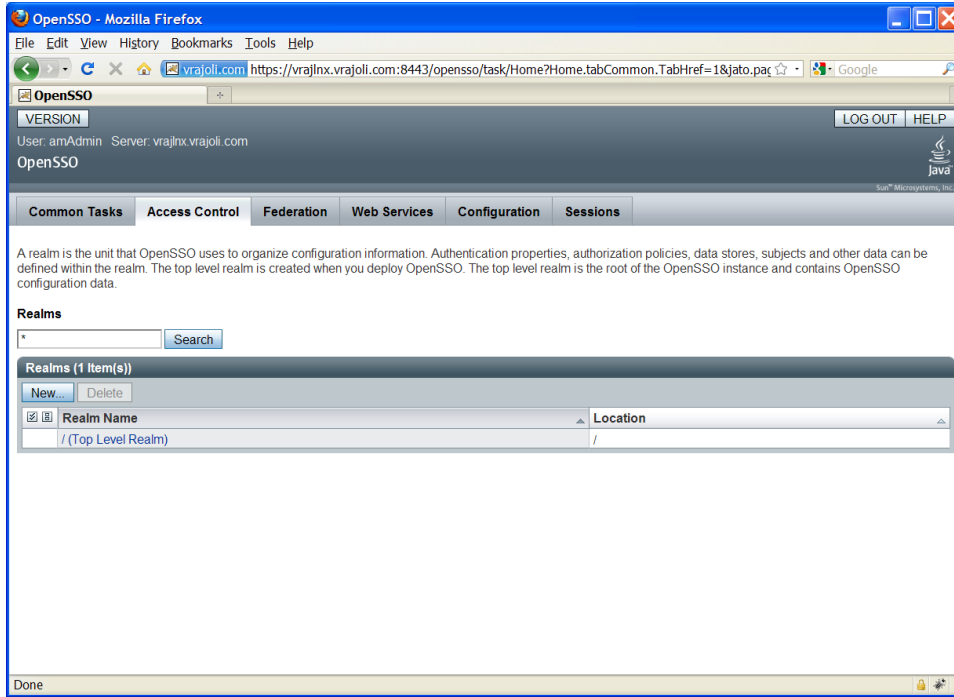


Configure active session timeout as 120 minutes.

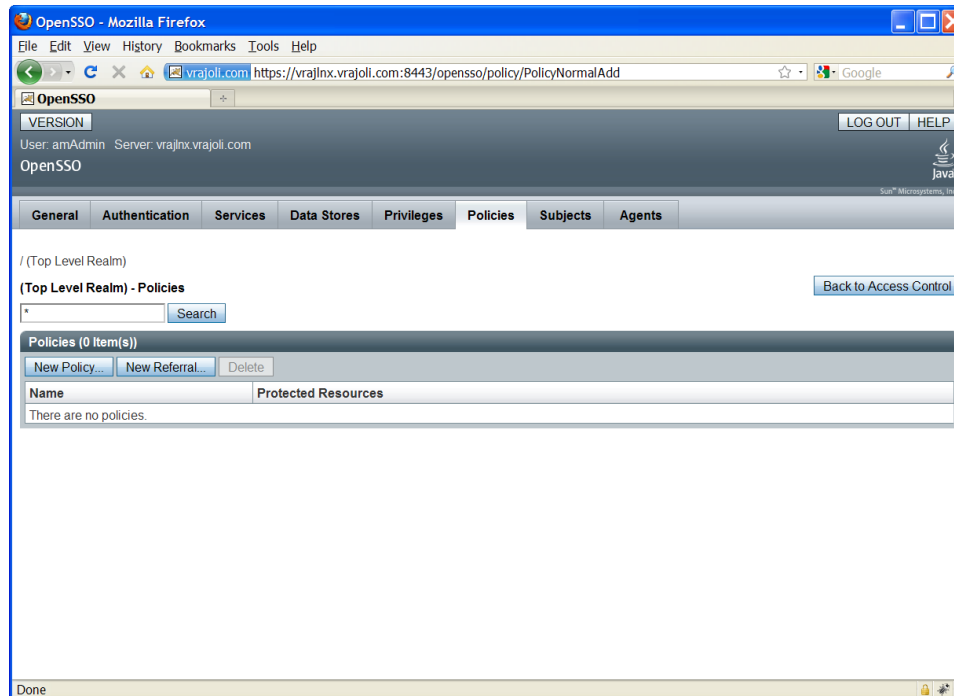


10.1.2 Configuring Policies on OpenSSO Server for Cisco Unity Connection 8.6

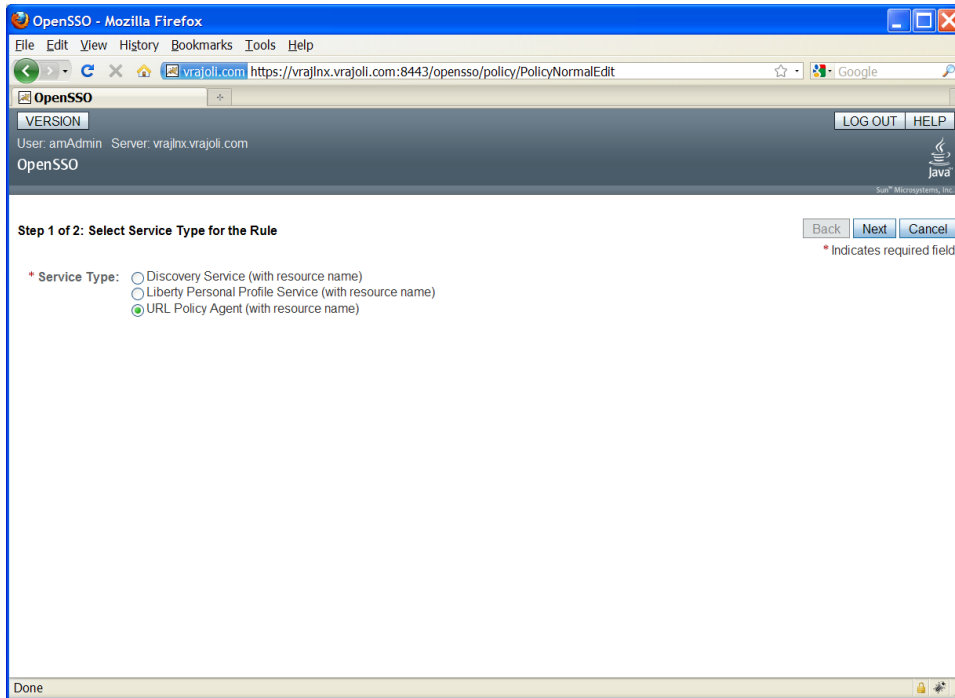
Log in to OpenSSO server with amAdmin username and password. Go to Access Control tab and click on / (**Top Level Realm**). The following window appears.



Go to policies tab and add a new policy. Enter the PolicyName.



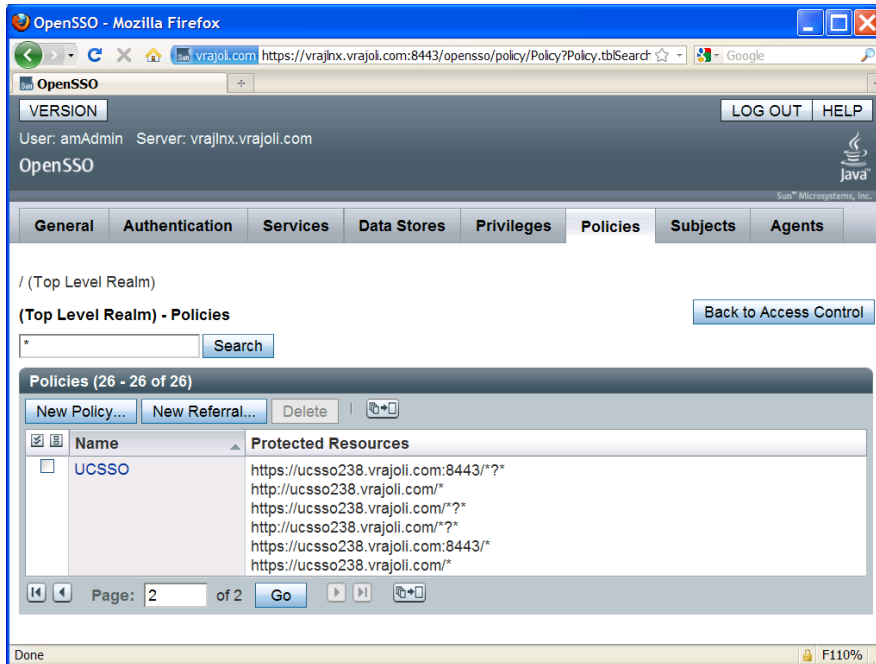
Create a new Rule from the Policy Configuration window. The following window appears. Select service type **URL Policy Agent (with resource name)**.



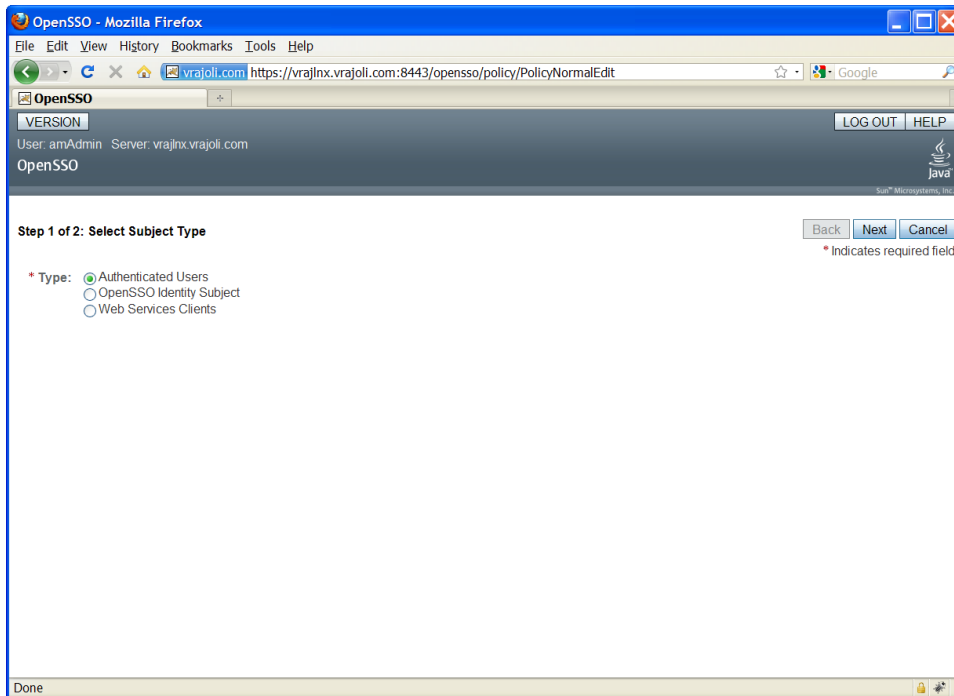
Then create rules for Cisco Unity Connection application(s) with below resource URLs configured in protected resources.

- `https://<FQDN>:8443/*`
- `https://<FQDN>:8443/*?*`
- `https://<FQDN>/*`
- `https://<FQDN>/*?*`
- `http://<FQDN>/*`
- `http://<FQDN>/*?*`

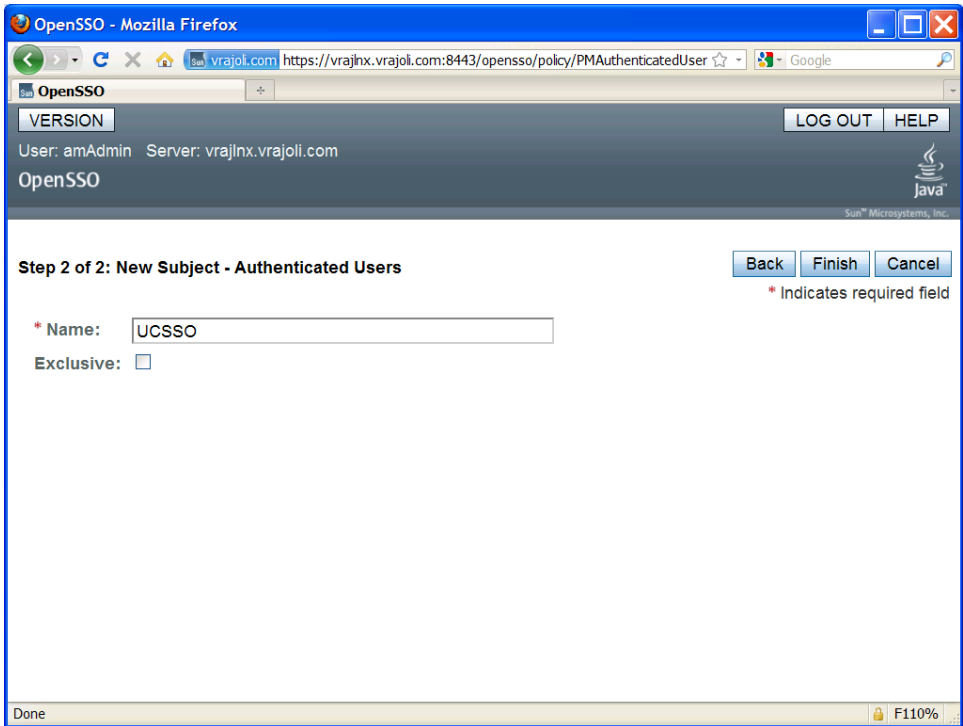
where FQDN is the fully qualified domain name of Cisco Unity Connection server. The following window shows OpenSSO Policy configured for Cisco Unity Connection server.



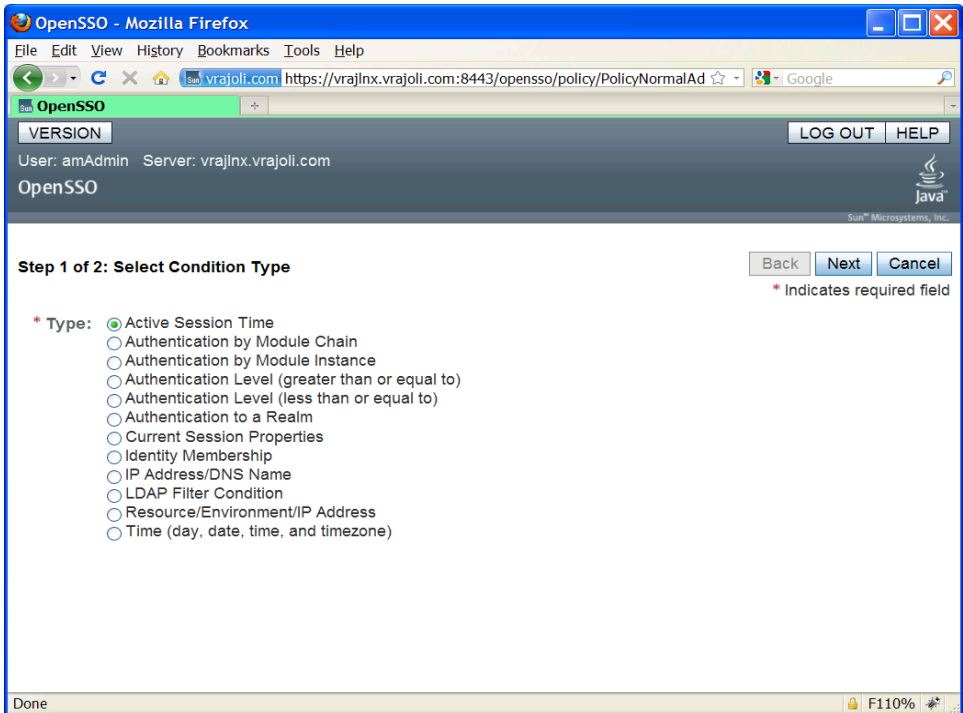
Click the **New** Button under Subjects on the Policy Configuration window. Select subject type **Authenticated Users**.



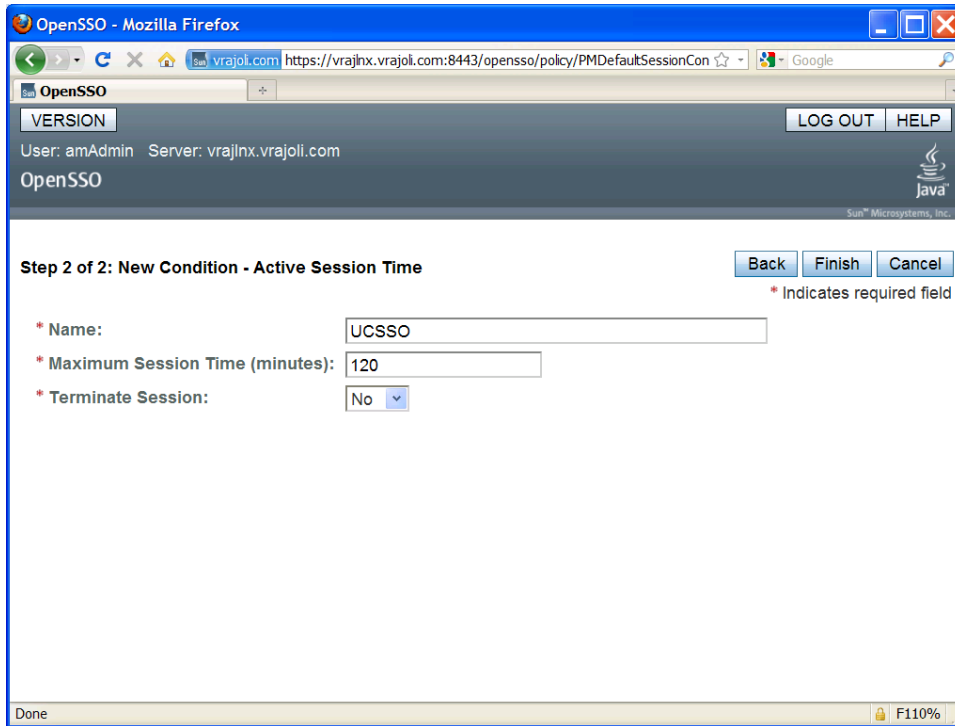
Enter the Subject Name and click **Finish**. Now Policy is created with defining Rules and Subjects.



Click the **New** button under Conditions. Under select condition type, select **Active Session Time** and then click **Next**.



Configure active session timeout as 120 minutes.

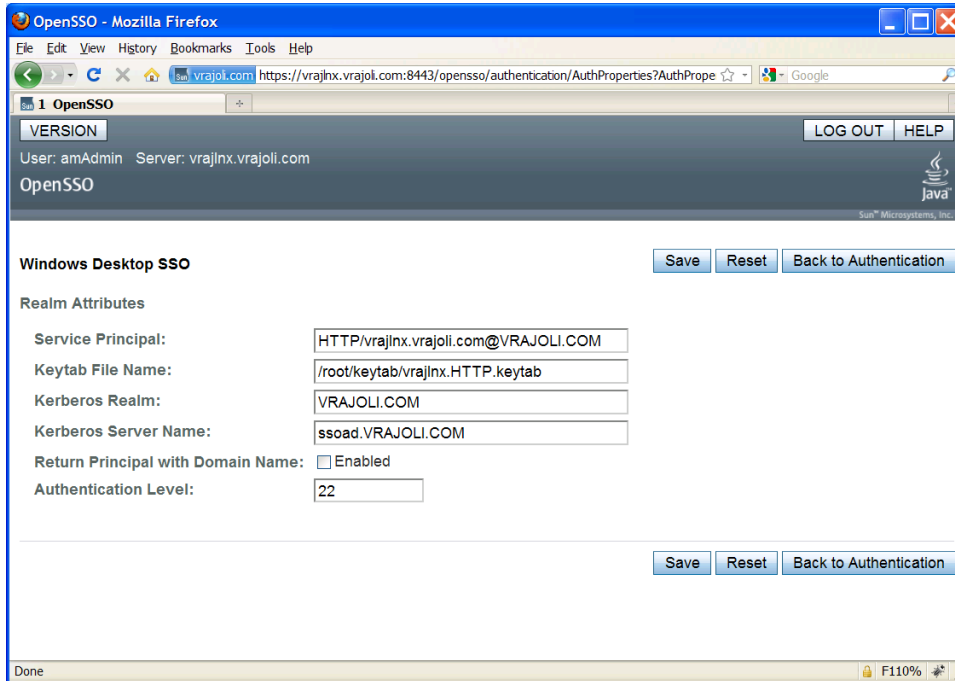


10.2 Configuring Windows Desktop SSO Authentication Module Instance

The Windows Desktop SSO Authentication Module enables OpenSSO Enterprise to work with Kerberos tokens. The user presents the Kerberos token, previously issued by a Kerberos Distribution Center, to OpenSSO Enterprise using the SPNEGO protocol. The client browser sends back a SPNEGO token embedded with a Kerberos token. The OpenSSO Windows Desktop SSO Authentication module retrieves the Kerberos token and authenticates the user using the Java GSS API. If authentication is successful, the OpenSSO Windows Desktop SSO Authentication module returns an SSOToken to the client.

- Copy the keytab files to OpenAM server, which was created in Chapter 8.
- Log in to the OpenSSO Enterprise administration console as amAdmin.
- Go to **Access Control > Default Realm > Authentication**.
- On the Module Instances window, click **New**.
- Enter a name for the new login module, and then select **Windows Desktop SSO**. Click **OK**.
In this test, Module instance with name CUCMUser is created.
- On the Module Instances window, click the name of the new login module (Example: CUCMUser) and provide the following information:
 - Service Principal: HTTP/ openAMhost.example.com@EXAMPLE.COM
 - Keytab File Name: /root/keytab/openAMhost.HTTP.keytab
 - Kerberos Realm: EXAMPLE.COM
 - Kerberos Server Name: Kerberos.example.com

- If multiple Kerberos Domain Controllers exist for failover purposes, all Kerberos Domain Controllers can be set using a colon (:) as the separator.
- Return Principal with Domain Name: False
- Authentication Level: 22
- Restart the OpenSSO Enterprise server.



10.3 Configuring J2EE Agent Profile on OpenSSO Server

10.3.1 Configuring J2EE Agent Profile on OpenSSO Server for Cisco Unified Communications Manager 8.5, 8.6

Perform the following task in OpenSSO Enterprise Console. The key steps of this task involve creating an agent name (ID) and an agent password.

- Log in to OpenSSO Enterprise Console as a user with AgentAdmin privileges, such as amAdmin.
- Click the **Access Control** tab.
- Click the name of the realm to which the agent will belong, such as the following: /(Top Level Realm).
- Click the **Agents** tab.
- Click the **J2EE** tab.
- Click **New** in the agent section.
- Enter values for the following fields:

- Name: Enter the name or identity of the agent. This is the agent profile name, which is the name the agent uses to log in to OpenSSO Enterprise. Multibyte names are not accepted.

Note: While enabling SSO on Cisco Unified Communications Manager or Cisco Unity Connection, when prompted to “Enter the name of the profile configured for this policy agent,” enter the above configured agent name.

- Password: Enter the agent password. However, it must be the same password entered in the agent profile password file that is used by the agentadmin utility to install the agent.

Note: While enabling SSO on Cisco Unified Communications Manager or Cisco Unity Connection, when prompted to “Enter the password of the profile name,” enter the above configured password.

- Reenter Password: Confirm the password.

- In the Server URL field, enter the OpenSSO Enterprise server URL.

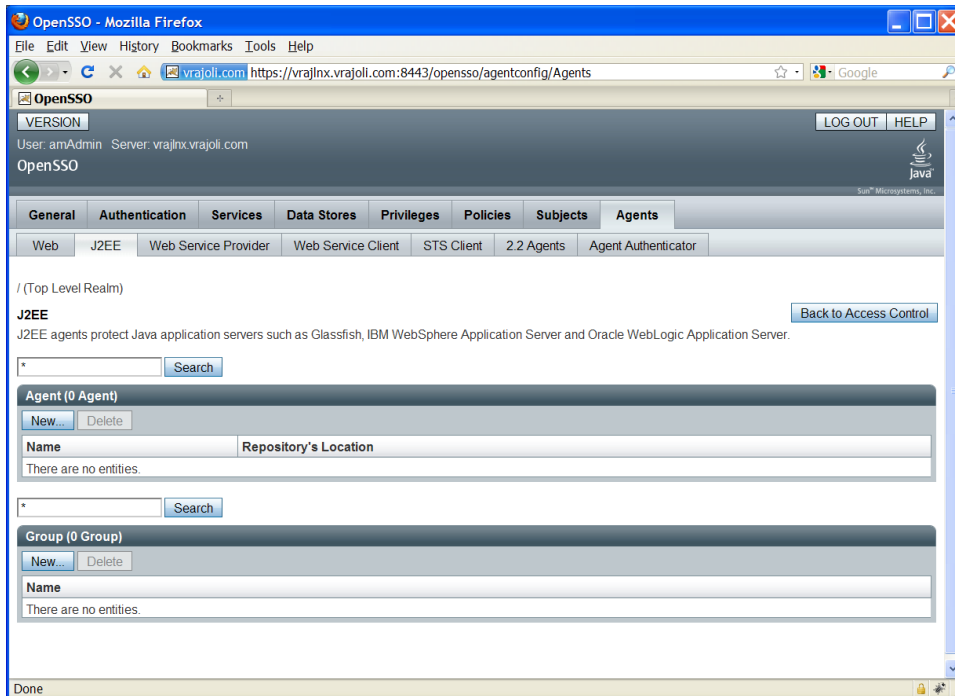
For example: https://<OpenAM FQDN>:8443/opensso

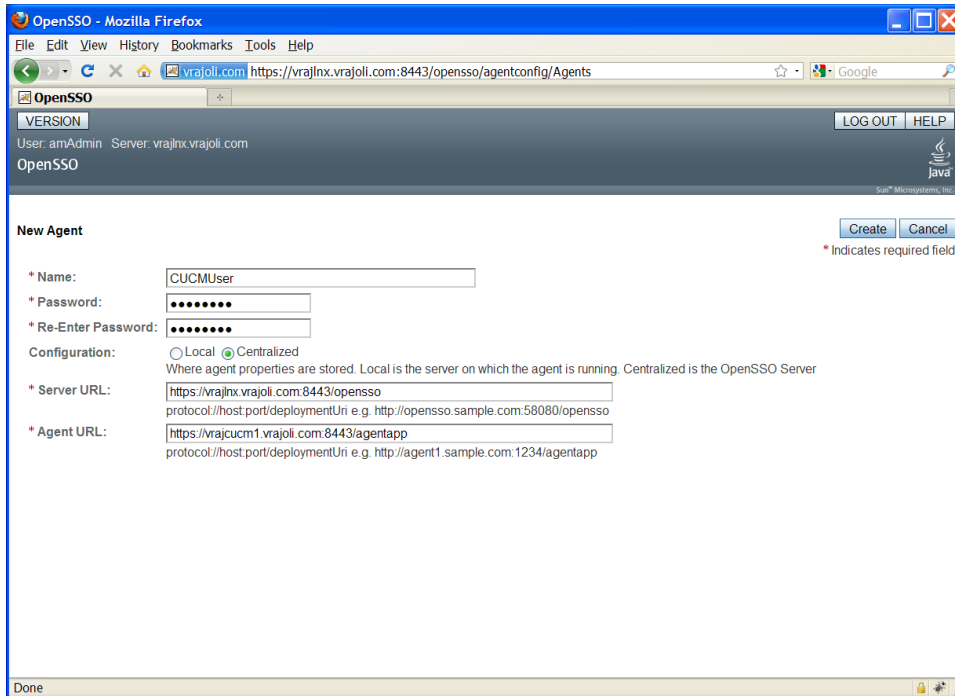
- In the Agent URL field, enter the URL for the agent application.

For example: https://<Cisco Unified Communications Manager FQDN>:8443/agentapp

- Click **Create**.

The Console creates the agent profile and displays the J2EE Agent window again with a link to the new agent profile.





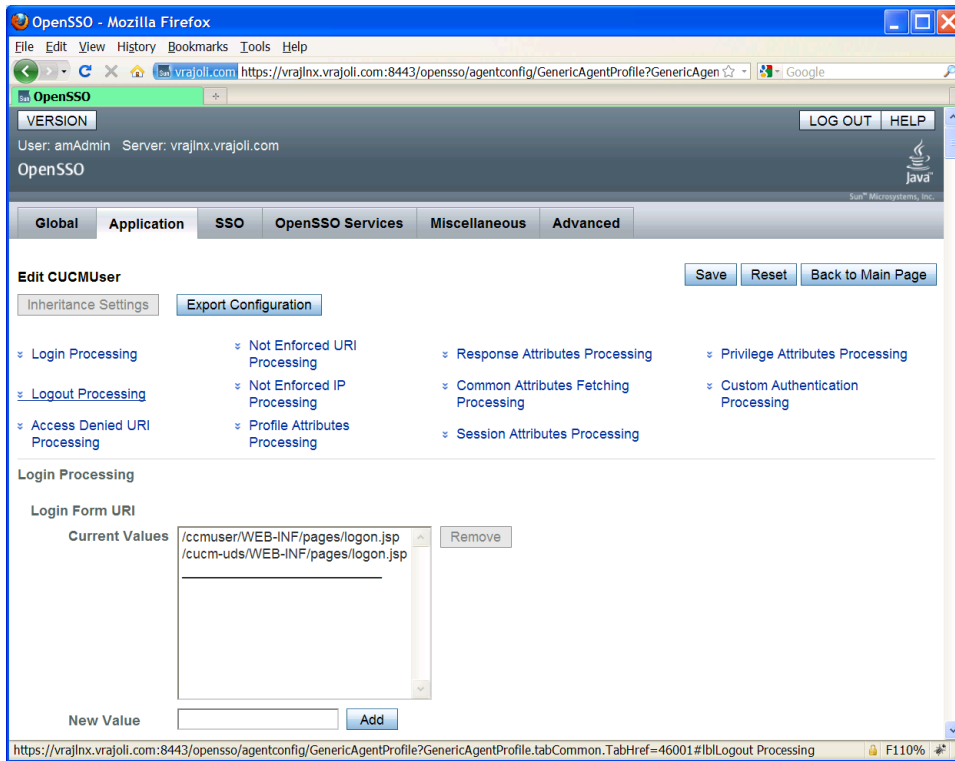
Click on the J2EE agent created above, go to Application tab, and under Login processing enter new Login Form URIs.

- For CCMUser webapp: ***/ccmuser/WEB-INF/pages/logon.jsp***
Note :- For CUCM 10.0 and above, the URL is /ucmuser/dojo/entry/Login.jsp
- For Cisco UC Integration for Microsoft Office Communicator:

/cucm-uds/WEB-INF/pages/logon.jsp

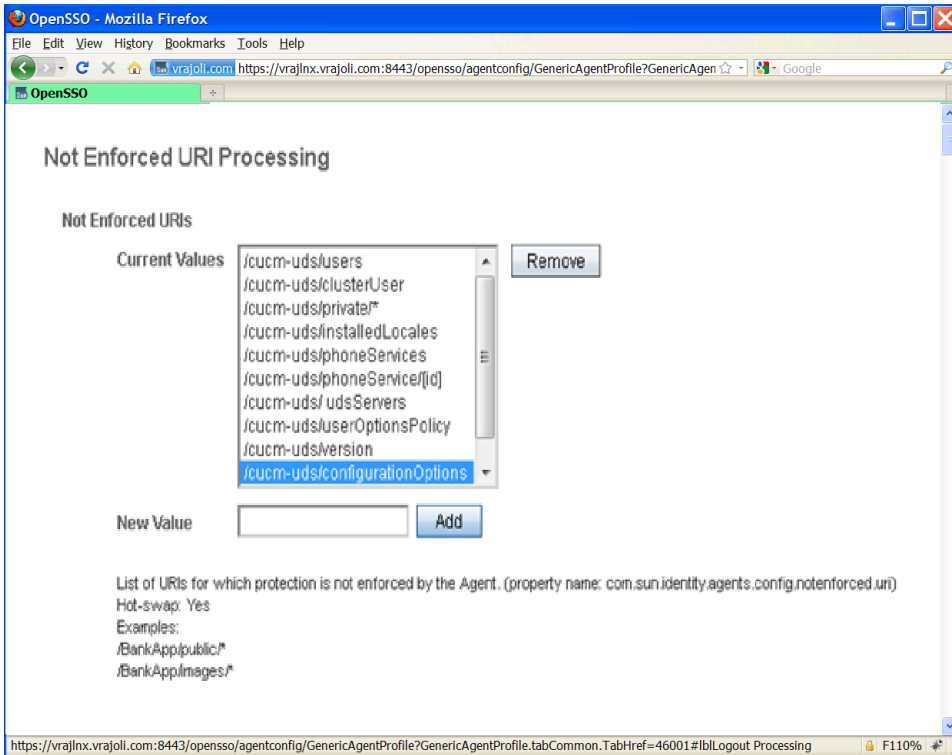
Starting from Cisco Unified Communications Manager Release 8.6, SSO support is provided for other applications like Cisco Unified CM Administration, Cisco Unified Serviceability, Cisco Unified Reporting, Cisco Unified OS Administration, Disaster Recovery System and RTMT along with Cisco Unified CM User Options and Cisco UC Integration for Microsoft Office Communicator. For these new applications, you must configure below Login Form URIs.

- For Cisco Unified CM Administration: ***/ccmadmin/WEB-INF/pages/logon.jsp***
- For Cisco Unified Serviceability: ***/ccmservice/WEB-INF/pages/logon.jsp***
- For Cisco Unified Reporting: ***/cucreports/WEB-INF/pages/logon.jsp***
- For Cisco Unified OS Administration: ***/cmplatform/WEB-INF/pages/logon.jsp***
- For Disaster Recovery System: ***/drf/WEB-INF/pages/logon.jsp***
- For Real Time Monitoring Tool (RTMT): ***/ast/WEB-INF/pages/logon.jsp***



Click on **Not Enforced URI Processing** . Enter the following URI's for which authentication is not required in this section.

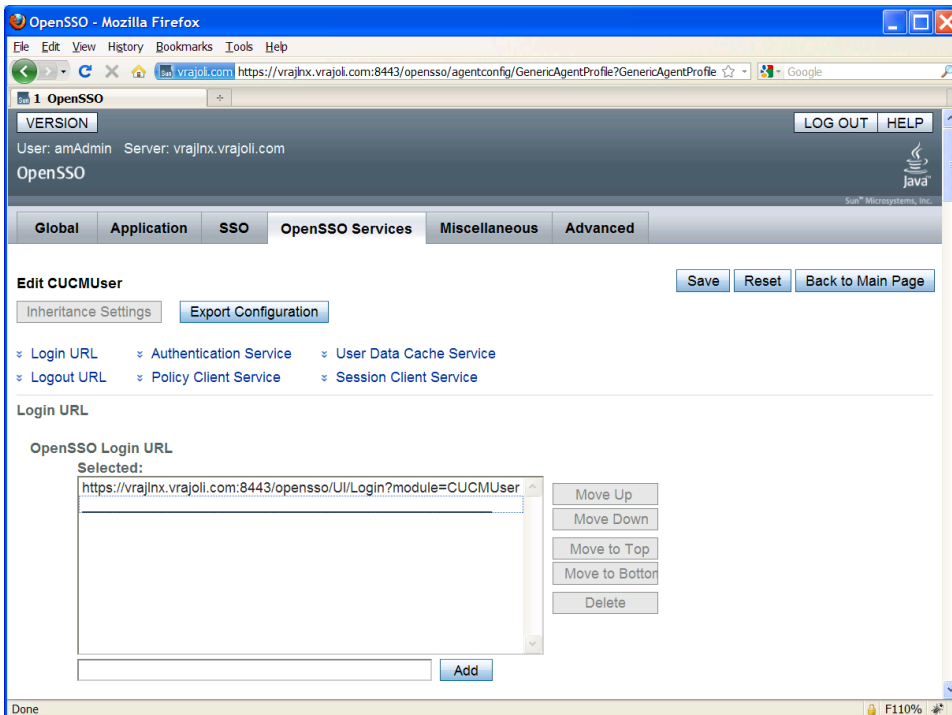
- /cucm-uds/users
- /cucm-uds/clusterUser
- /cucm-uds/configurationOptions
- /cucm-uds/installedLocales
- /cucm-uds/phoneServices
- /cucm-uds/phoneService/[id]
- /cucm-uds/ udsServers
- /cucm-uds/userOptionsPolicy
- /cucm-uds/version
- /cucm-uds/private/*



Go to OpenSSO Services tab, under Login URL add OpenSSO Login URL as **https://<OpenSSO FQDN>:8443/opensso/UI/Login?module=<WindowsDesktopSSO_Module>**.

WindowsDesktopSSO_Module should be same as the one created in section 10.2.

Ex: *https://<OpenAM FQDN>:8443/opensso/UI/Login?module=CUCMUser*

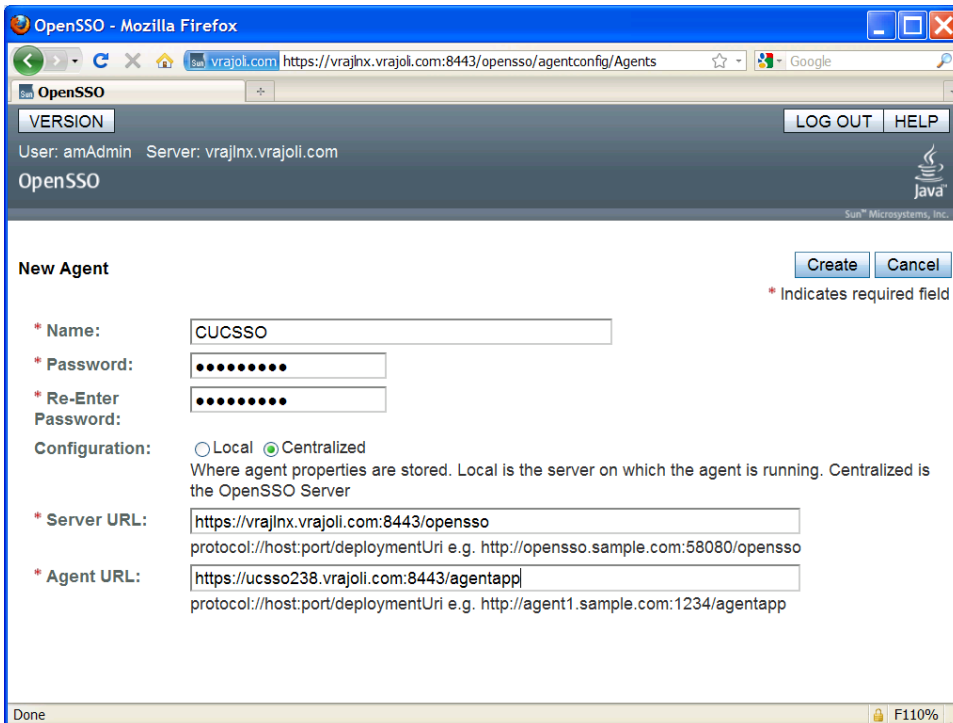
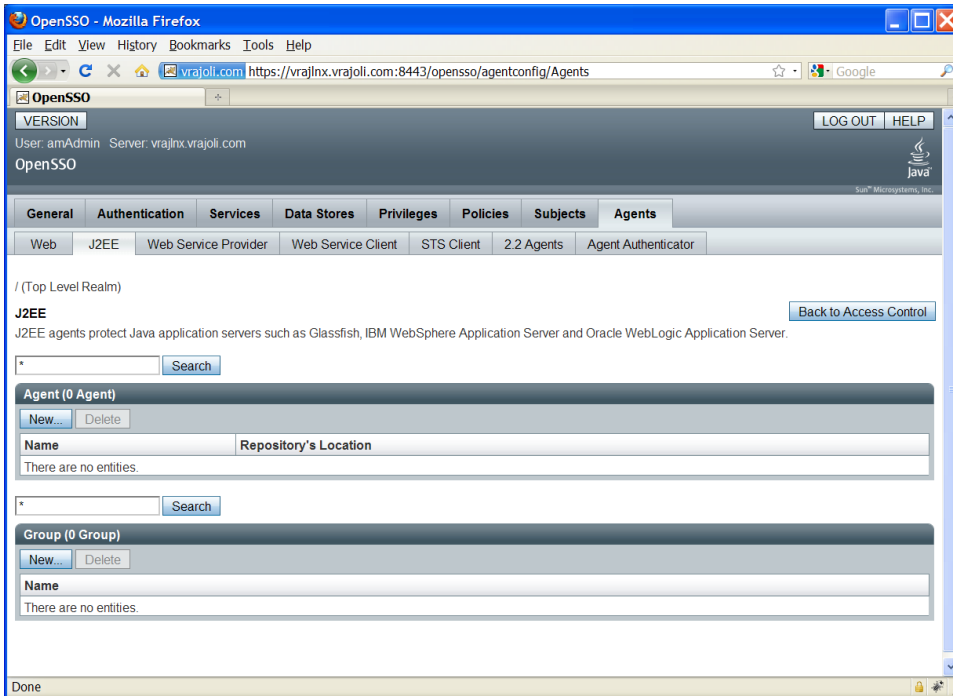


10.3.2 Configuring J2EE Agent Profile on OpenSSO Server for Cisco Unity Connection 8.6

Perform the following task in OpenSSO Enterprise Console. The key steps of this task involve creating an agent name (ID) and an agent password.

- Log in to OpenSSO Enterprise Console as a user with AgentAdmin privileges, such as amAdmin.
- Click the **Access Control** tab.
- Click the name of the realm to which the agent will belong, such as the following: /(Top Level Realm).
- Click the **Agents** tab.
- Click the **J2EE** tab.
- Click **New** in the agent section.
- Enter values for the following fields:
 - Name: Enter the name or identity of the agent. This is the agent profile name, which is the name the agent uses to log in to OpenSSO Enterprise. Multibyte names are not accepted.
Note: While enabling SSO on Cisco Unified Communications Manager or Cisco Unity Connection, when prompted to “Enter the name of the profile configured for this policy agent,” enter the above configured agent name.
 - Password: Enter the agent password. However, it must be the same password entered in the agent profile password file that is used by the agentadmin utility to install the agent.
Note: While enabling SSO on Cisco Unified Communications Manager or Cisco Unity Connection, when prompted to “Enter the password of the profile name,” enter the above configured.
 - Reenter Password: Confirm the password.
- In the Server URL field, enter the OpenSSO Enterprise server URL.
For example: https://<OpenAM FQDN>:8443/opensso
- In the Agent URL field, enter the URL for the agent application.
For example: https://<Cisco Unity Connection FQDN>:8443/agentapp
- Click **Create**.

The Console creates the agent profile and displays the J2EE Agent window again with a link to the new agent profile.

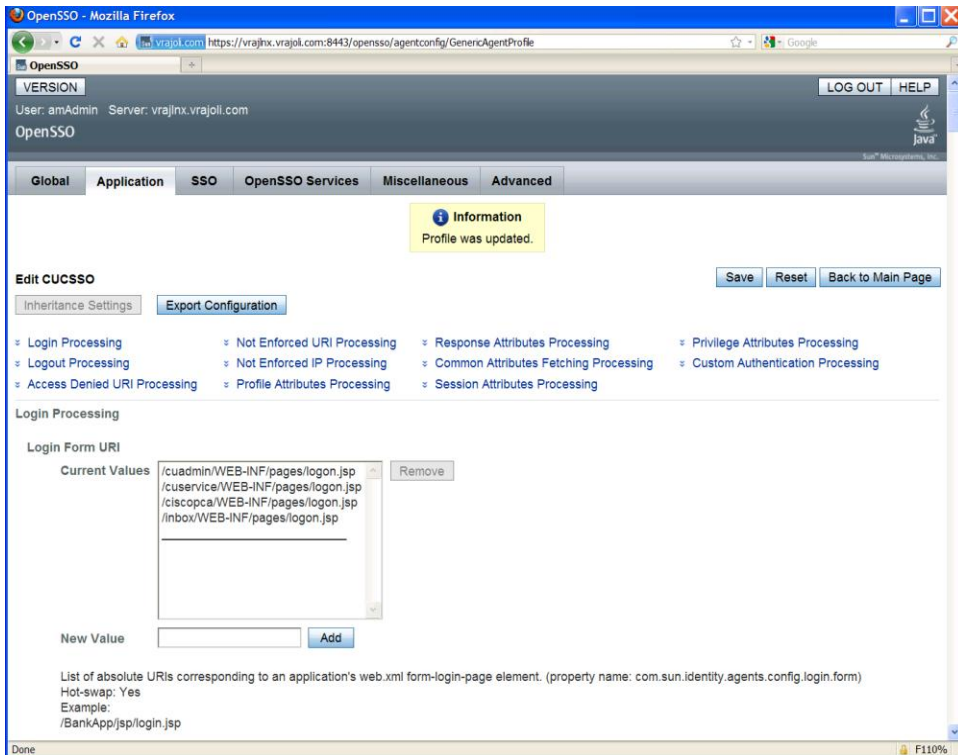


Click on the J2EE agent created above, go to Application tab, and under Login processing enter new Login Form URIs.

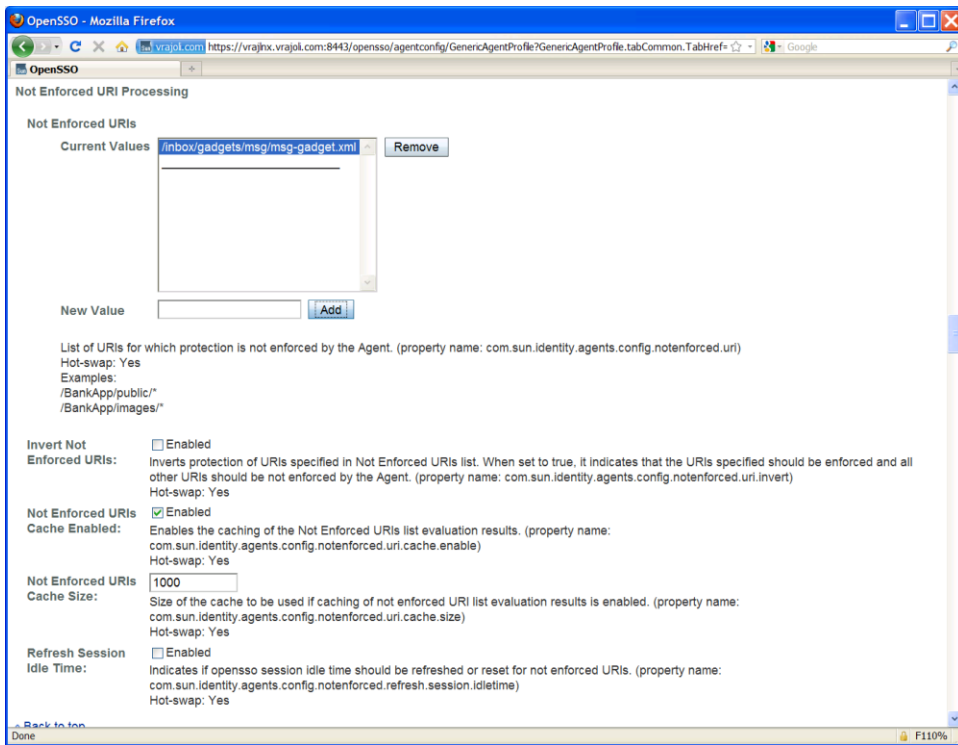
- For Cisco Unity Connection Administration: */cuadmin/WEB-INF/pages/logon.jsp*
- For Cisco Unity Connection Serviceability: */cuservice/WEB-INF/pages/logon.jsp*

- For Cisco Personal Communications Assistant: */ciscopca/WEB-INF/pages/logon.jsp*
- For Cisco Unity Connection Web Inbox: */inbox/WEB-INF/pages/logon.jsp*
- For Cisco Unified CM User option: */ccmuser/WEB-INF/pages/logon.jsp*
- For Cisco Unity Connection REST APIs: */vmrest/WEB-INF/pages/logon.jsp*
- For Cisco UC Integration for Microsoft Office Communicator: */cucm-uds/WEB-INF/pages/logon.jsp*
- For Cisco Unified CM Administration: */ccmadmin/WEB-INF/pages/logon.jsp*
- For Cisco Unified Serviceability: */ccmservice/WEB-INF/pages/logon.jsp*
- For Cisco Unified Reporting: */cucreports/WEB-INF/pages/logon.jsp*
- For Cisco Unified OS Administration: */cmplatform/WEB-INF/pages/logon.jsp*
- For Disaster Recovery System: */drf/WEB-INF/pages/logon.jsp*
- For Real Time Monitoring Tool (RTMT): */ast/WEB-INF/pages/logon.jsp*

Note :- In CUCM 10.0 and above, the URL for Cisco Unified CM User option is /ucmuser/dojo/entry/Login.jsp



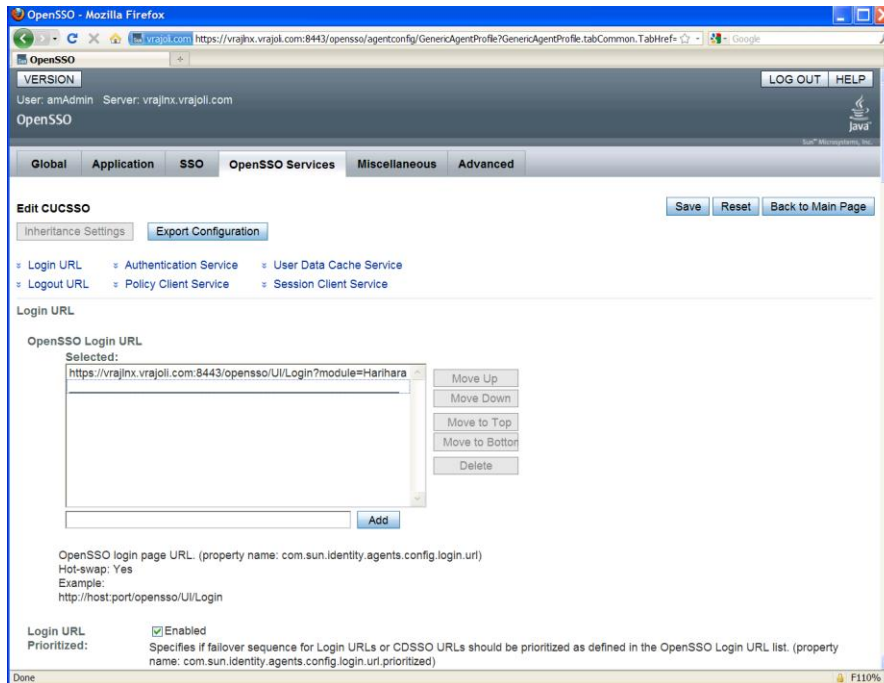
Under the Application tab, in the section titled “Not Enforced URI Processing” add the following URI: **/inbox/gadgets/msg/msg-gadget.xml**.



Go to OpenSSO Services tab, Under Login URL add OpenSSO Login URL as **https://<OpenSSO FQDN>:8443/opensso/UI/Login?module=<WindowsDesktopSSO_Module>**.

WindowsDesktopSSO_Module should be same as the one created in section 10.2.

Ex: *https://<OpenAM FQDN>:8443/opensso/UI/Login?module=CUCMUser*



11 Undeploying/Removing OpenSSO Enterprise (OpenAM)

11.1 Uninstalling OpenSSO Enterprise (OpenAM) Server Deployed on Linux Platform

- Stop the Tomcat running on OpenAM server by executing shutdown.sh under /root/ apache-tomcat-7.0.0/bin directory.
- Remove the following directories and all of their contents:
 - ConfigurationDirectory is the directory created when the OpenSSO Enterprise instance is initially configured using the Configurator. The default directory is **opensso** in the home directory of the user running the Configurator. If the Configurator is run by root, ConfigurationDirectory is created in the root home directory (/root).
 - user-home-directory.openssocfg where user-home-directory is the home directory of the user who deployed the opensso.war file. If this user is root, the directory is **/.openssocfg**.
- Remove the opensso.war file from webapps directory of the Tomcat.
Example: /root/ apache-tomcat-7.0.0/webapps
- Start the Tomcat on OpenAM Server, by executing startup.sh under /root/ apache-tomcat-7.0.0/bin directory.

11.2 Uninstalling OpenSSO Enterprise (OpenAM) Server Deployed on Windows Platform

- Stop the Tomcat service if running on this OpenAM server. (**Administrative Tools > Services > Apache Tomcat 7 > Stop**)
- Delete the **opensso** and **.openssocfg** folder from the user home directory.

- Delete the **opensso.war** file from the webapps folder of Tomcat.
Example: c:\Program Files\Apache Software Foundation\Tomcat 7.0\webapps
- Start the Tomcat from **Administrative Tools > Services > Apache Tomcat 7 > Start**.

12 Configuring Browser/Registry for SSO

Standard browser clients like Internet Explorer, Mozilla Firefox and Safari have the capability to handle HTTP 401: Negotiate. The steps to enable this capability for Internet Explorer 6/7/8 and Firefox is explained in this section.

12.1 Internet Explorer

Steps to set up Internet Explorer for SSO:

- Supported version 6.X onwards.
- In the Tool menu, go to **Internet Options > Advanced > Security**.
- Select the check box for Integrated Windows Authentication option.
- Go to **Tools > Internet Options > Security > Local Intranet**.
- Select Custom Level. In the User Authentication/Logon panel/option, select the **Automatic Logon Only in Intranet Zone** option.
- Go to **Sites** and select all of the options.
- Click **Advanced** and add the OpenSSO Enterprise to the local zone (if it is not added already).
- Additionally for IE7 and IE8 browsers, go to **Tools > Internet Options > Security** tab. Uncheck the **Enable Protected Mode** check box (requires restarting Internet Explorer).
- For Windows machines (Windows7/Windows 2008 and other higher versions) with extended Protection for Authentication enabled, disable extended Protection for Authentication by creating registry entry under registry key HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\LSA\. Add DWORD value SuppressExtendedProtection - 0x02.

12.2 Mozilla Firefox

Steps to setup Firefox for SSO:

- Supported version 3.x onwards.
- Open Firefox browser.
- In the address field, type **about:config**.
- In the filter, type **network.n**.
- Double-click on **network.negotiate-auth.trusted-uris**. This preference lists the sites that are permitted to engage in SPNEGO Authentication with the browser. Enter a comma-delimited list of trusted domains or URLs, for example, vrajoli.com.

12.3 Configuring Windows Registry for RTMT SSO

To achieve RTMT SSO, a new registry key '*allowtgtsessionkey*' of type *REG_DWORD* with value set to '*1*' should be created on desktop client (Windows XP/Windows 7) at below location corresponding the respective OS distribution.

Windows XP →

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\Kerberos\

Windows Vista/Windows 7 →

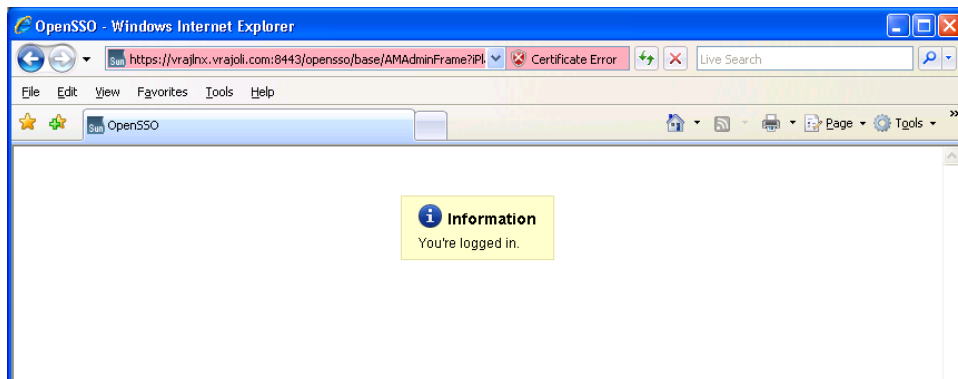
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\Kerberos\Parameters

12.4 SSO Configurations Test with Browser

- Log in to a domain computer that is a member of domain controller. (*Example: vrajoli.com*)
- Configure the browser capable for SSO, described in sections 12.1 and 12.2.
- Browse for `https://<OpenAM FQDN>:8443/opensso/UI/Login?module=<WindowsDesktopSSO_Module>`. `WindowsDesktopSSO_Module` should be same as that configured in section 10.2.

In this test, it will be: `https://vrajlnx.vrajoli.com:8443/opensso/UI/Login?module=CUCMUser`

- You should see the message "You're logged in" as shown in the following figure. This message indicates that the configurations you made are correct.



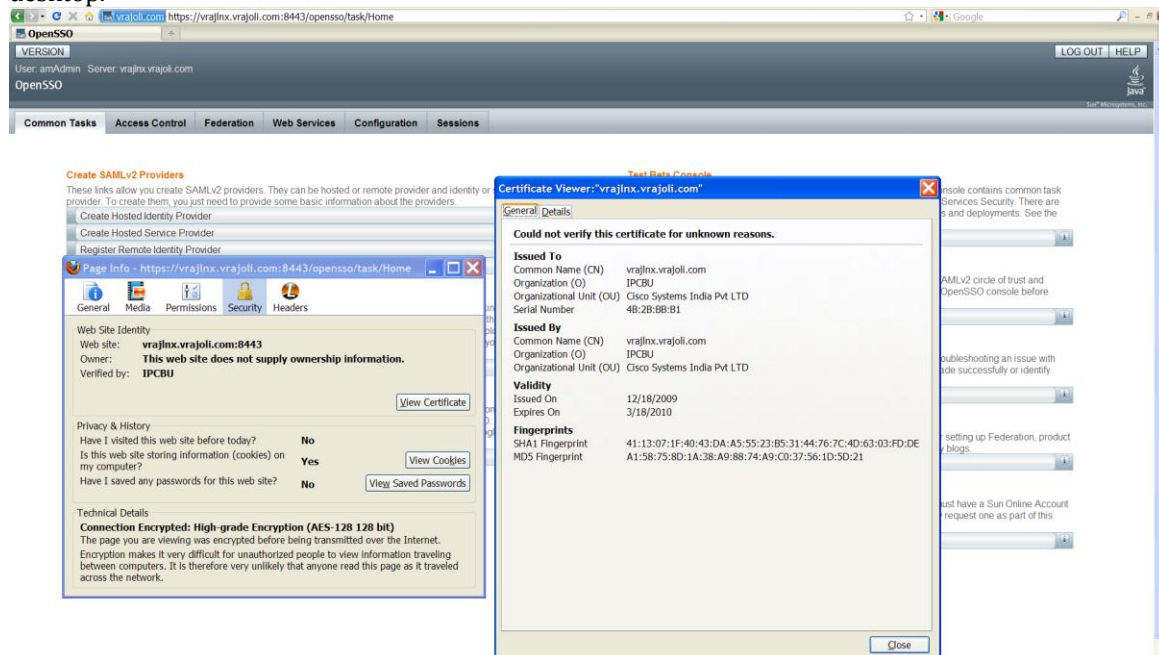
13 Configuring SSO on Cisco Unified Communications Manager 8.5

Before you enable SSO on Unified CM 8.5, you must complete the following steps:

- Log in to Cisco Unified Serviceability and activate Cisco DirSync service.
- Log in to Cisco Unified CM Administration, go to **System > LDAP > LDAP System**, check the **Enable Synchronizing from LDAP Server** check box and save.
- Go to **System > LDAP > LDAP Directory**, create a new directory agreement with configuring LDAP Directory information and LDAP server information, save the configuration and click the **Perform Full Sync** button. Upon completion of directory sync, AD users are imported to Unified CM.
- On Cisco Unified CM Administration window, go to **User Management > End User**, verify all the AD users are available and LDAP Sync Status is Active.
- Associate respective User Groups to the end users and save.

Because we configure Unified CM to talk to OpenAM over HTTPS while enabling SSO, we must import OpenAM server certificate into CallManager tomcat-trust store before enabling the SSO on Unified CM.

To get the OpenAM server certificate, log in to OpenAM URL (<https://<OpenAM FQDN>:8443/opensso>), click the **Security** icon at the bottom right corner. Click the **Details** tab of Certificate Viewer window, click the **Export** button and save it your desktop.



After getting OpenAM server certificate, log on to Unified CM OS Administration window, go to **Security > Certificate Management**, and click the **Upload Certificate** button.

In the Upload Certificate window, select certificate name as **tomcat-trust** and browse for the saved OpenAM certificate and upload it.

OpenAM server certificate has been added into the Unified CM Tomcat-trust store.

13.1 Unified CM 8.5 SSO CLI Commands

Three CLI command are available for managing SSO on Unified CM 8.5:

- `utils sso enable`
- `utils sso disable`
- `utils sso status`

13.1.1 `utils sso enable`

Use this command to enable SSO-based authentication.

This command starts the wizard for enabling SSO on Unified CM. It asks for the following:

- OpenAM server URL
- Relative path for policy agent deployment
- Profile name configured for this policy agent on the AM server
- Password for the above
- Module name configured for Windows Desktop SSO

```
admin:utils sso enable
**** W A R N I N G ****
This command will restart Tomcat for successful completion.
This command needs to be executed on all the nodes in the cluster.
Do you want to continue (yes/no): yes
Enter URL of the Open Access Manager (OpenAM) server: https://vrajlrx.vrajoli.com:8443/opensso
Enter the relative path where the policy agent should be deployed: agentapp
Enter the name of the profile configured for this policy agent: CUCMUser
Enter the password of the profile name: *****
Enter the login module instance name configured for Windows Desktop SSO: CUCMUser
Validating connectivity and profile with Open Access Manager (OpenAM) Server: https://vrajlrx.vrajoli.com:8443/opensso
Valid profile
Enabling SSO ... This will take upto 5 minutes
SSO Enable Success

Please make sure to execute this command on all the nodes in the cluster.
admin:
```

13.1.2 utils sso disable

This command disables SSO-based authentication. No parameters are required.

```
admin:utils sso disable

***** W A R N I N G *****
This command will restart Tomcat for successful completion.

This command needs to be executed on all the nodes in the cluster.

Do you want to continue (yes/no): yes
Disabling SSO configuration. This will take upto 5 minutes

Disable SSO Success

Please make sure to execute this command on all the nodes in the cluster.
```

13.1.3 utils sso status

This command provides the status of SSO on Unified CM 8.5. No parameters are required.

When SSO is disabled:

```
admin:utils sso status
SSO Status: Disabled
```

When SSO is enabled:

```
admin:utils sso status
SSO Status: Enabled

Primary Open Access Manager (OpenAM) server URL: https://vrajlnx.vrajoli.com:8443/opensso

Profile name: CUCMUser

Login module name: CUCMUser
admin:
```

14 Configuring SSO on Cisco Unified Communications Manager 8.6

With Unified CM 8.6, SSO is extended to the following Unified CM applications along with CCMUser and Cisco UC Integration for Microsoft Office Communicator:

- Cisco Unified CM Administration
- Cisco Unified Serviceability
- Cisco Unified Reporting
- Cisco Unified OS Administration
- Disaster Recovery System

- RTMT

Before you enable SSO on Unified CM 8.6, you must complete the following steps:

- Log in to Cisco Unified Serviceability and activate Cisco DirSync service.
- Log in to Cisco Unified CM Administration, go to **System > LDAP > LDAP System**, check the **Enable Synchronizing from LDAP Server** check box and save.
- Go to **System > LDAP > LDAP Directory**, create a new directory agreement with configuring LDAP Directory information and LDAP server information, save the configuration and click the **Perform Full Sync** button. Upon completion of directory sync, AD users are imported to Unified CM.
- On Cisco Unified CM Administration window, go to **User Management > End User**, and verify that all the AD users are available and LDAP Sync Status is Active.
- Associate respective User Groups to the end users and save. For an end user to access SSO-enabled applications like Cisco Unified CM Administration, Cisco Unified Serviceability, Cisco Unified Reporting, Cisco Unified OS Administration, and RTMT, end user should have **Standard Audit Users** and **Standard CCM Super Users** User Groups associated.

Because we configure Unified CM to talk to OpenAM over HTTPS while enabling SSO, we must import OpenAM server certificate into CallManager tomcat-trust store before enabling the SSO on Unified CM.

14.1 Unified CM 8.6 SSO CLI Commands

14.1.1 `utils sso enable`

This command enables SSO for the applications that you choose from the following list:

- **Cisco Unified CM Administration (CUCM Admin, CU Serviceability, CU Reporting)**
- **Cisco Unified CM User Options (CUCM End User options)**
- **Cisco Unified Operating System Administration (CUCM OS Admin, DRF)**
- **Cisco Unified Data Service (CUCiMOC)**
- **RTMT**

This command starts the wizard for enabling SSO on the Unified CM. It asks for the following:

- OpenAM server URL
- Relative path for policy agent deployment
- Profile name configured for this policy agent on the AM server
- Password for the above
- Module name configured for Windows Desktop SSO

```

admin:utils sso enable
***** W A R N I N G *****
This command will restart Tomcat for successful completion.
This command needs to be executed on all the nodes in the cluster.
Do you want to continue (yes/no): y

List of apps for which SSO can be enabled
1) Cisco Unified CM Administration (CUCM Admin, CU Serviceability, CU Reporting)
2) Cisco Unified CM User Options (CUCM End User options)
3) Cisco Unified Operating System Administration (CUCM OS Admin, DRF)
4) Cisco Unified Data Service (CUCiMOC)
5) RTMT
Do you want to enable SSO for Cisco Unified CM Administration (CUCM Admin, CU Serviceability, CU Reporting) (yes/no):y
Do you want to enable SSO for Cisco Unified CM User Options (CUCM End User options) (yes/no):y
Do you want to enable SSO for Cisco Unified Operating System Administration (CUCM OS Admin, DRF) (yes/no):y
Do you want to enable SSO for Cisco Unified Data Service (CUCiMOC) (yes/no):y
Do you want to enable SSO for RTMT (yes/no):y

Enter URL of the Open Access Manager (OpenAM) server: https://vrajlnx.vrajoli.com:8443/opensso
Enter the relative path where the policy agent should be deployed: agentapp
Enter the name of the profile configured for this policy agent: cucmssso238
Enter the password of the profile name: *****
Enter the login module instance name configured for Windows Desktop SSO: Harihara
Validating connectivity and profile with Open Access Manager (OpenAM) Server: https://vrajlnx.vrajoli.com:8443/opensso
Valid profile
Valid module name
Enabling SSO ... This will take upto 5 minutes
SSO Enable Success

Please make sure to execute this command on all the nodes in the cluster.
admin:

```

14.1.2 utils sso disable

This command disables SSO for SSO-enabled options.

14.1.3 utils sso status

This command displays the list of SSO-enabled applications.

```

admin:utils sso status
SSO Status: Disabled

```

14.2 CUCM 8.6 SSO GUI

From Unified CM Release 8.6, SSO configurations (SSO enable/disable/status) are supported from GUI as well.

To configure SSO from GUI, log in to Cisco Unified OS Administration.

Go to **Security > Single Sign on** and you will see the following configuration window.

The screenshot shows the Cisco Unified Operating System Administration interface for SSO Applications Configuration. At the top, there is a navigation menu with options: Show, Settings, Security, Software Upgrades, Services, and Help. Below the menu is a 'Save' button. A warning message states: 'Warning: Changing the SSO settings causes an immediate Tomcat restart'. The 'Server Settings' section contains five input fields with asterisks indicating they are required: 'Enter URL of the Open Access Manager (OpenAM) server*', 'Enter the relative path where the policy agent should be deployed*', 'Enter the name of the profile configured for this policy agent*', 'Enter the password of the profile name*', and 'Enter the login module instance name configured for Windows Desktop SSO*'. The 'Select Applications' section features a table with columns for 'Select All', 'Application name', and 'SSO Status'. The table lists five applications, all with 'Disabled' status and unchecked 'Select All' boxes. Below the table is another 'Save' button. At the bottom, an information icon is followed by the text '*- indicates required item.'

Select All	Application name	SSO Status
<input type="checkbox"/>	Cisco Unified CM Administration (CUCM Admin, CU Serviceability, CU Reporting)	Disabled
<input type="checkbox"/>	Cisco Unified CM User Options (CUCM End User options)	Disabled
<input type="checkbox"/>	Cisco Unified Operating System Administration (CUCM OS Admin, DRF)	Disabled
<input type="checkbox"/>	Cisco Unified Data Service (CUCIMOC)	Disabled
<input type="checkbox"/>	RTMT	Disabled

For enabling SSO from GUI, the same parameters are requested as requested from CLI. All the inputs are validated before enabling SSO to a selected application. A check box is provided for selecting the application(s); check the check box to select the applications on which you want to enable SSO and then click **Save**. After successful validation of all the inputs, a popup window is displayed saying "Enabling/Disabling SSO for the applications will cause Tomcat to restart. Do you want to continue?" Click **OK** if you want to proceed or click **Cancel** to cancel.

To disable SSO-enabled application(s), uncheck the check box of the SSO-enabled application(s), then click **Save**. A popup window is displayed saying "Enabling/Disabling SSO for the applications will cause Tomcat to restart. Do you want to continue?" Click **OK** if you want to proceed or click **Cancel** to cancel.

15 Configuring SSO on Cisco Unity Connection 8.6

Before you enable SSO on Cisco Unity Connection 8.6, you must complete the following steps:

- Import users to Cisco Unity Connection either directly from LDAP server **OR** from Unified CM; however, users imported from Unified CM must first be imported from LDAP to Unified CM). Users must be configured with the appropriate roles to log in to Cisco Unity Connection Administration, or Cisco Unity Connection Serviceability.
- For a coresident and standalone Cisco Unity Connection server, user accounts that will access Cisco Unity Connection Administration and Cisco Unity Connection Serviceability must have the System Administrator role. To give a user the System Administrator role, select the user in Cisco Unity Connection Administration, choose **Edit/Roles** from the menu. Select **System Administrator** from the available roles, and add it to the user's Assigned Roles.

Because we configure Cisco Unity Connection to talk to OpenAM over HTTPS while enabling SSO, we must import OpenAM server certificate into CallManager tomcat-trust store before enabling the SSO on Cisco Unity Connection.

15.1 Cisco Unity Connection 8.6 SSO CLI Commands

There are three CLI command available for managing SSO on Cisco Unity Connection 8.6:

- `utils sso enable`
- `utils sso disable`
- `utils sso status`

15.1.1 utils sso enable

This command enables SSO-based authentication. Cisco Unity Connection offers the following options for enabling SSO:

- **Cisco Unified CM Administration (CUCM Admin, CU Serviceability, CU Reporting)**
- **Cisco Unified CM User Options (CUCM End User options)**
- **Cisco Unified Operating System Administration (CUCM OS Admin, DRF)**
- **Cisco Unity Connection REST APIs**
- **Cisco Unity Connection PCA and Web Inbox**
- **Cisco Unity Connection Administration**
- **Cisco Unified Data Service (CUCiMOC)**
- **RTMT**

This command starts the wizard for enabling SSO on Cisco Unity Connection. It asks for the following:

- OpenAM server URL
- Relative path for policy agent deployment
- Profile name configured for this policy agent on the AM server
- Password for the above
- Module name configured for Windows Desktop SSO


```
admin:utils sso enable
      ***** W A R N I N G *****
This command will restart Tomcat for successful completion.

This command needs to be executed on all the nodes in the cluster.

Do you want to continue (yes/no): y

List of apps for which SSO can be enabled

1) Cisco Unified CM Administration (CUCM Admin, CU Serviceability, CU Reporting)
2) Cisco Unified Operating System Administration (CUCM OS Admin, DRF)
3) Cisco Unity Connection Rest APIs
4) Cisco Unity Connection PCA and Web Inbox
5) Cisco Unity Connection Administration
6) Cisco Unified Data Service (CUCiMOC)
7) RTMT
8) Cisco Unified CM User Options (Cisco Unified CM User options)

Do you want to enable SSO for Cisco Unified CM Administration (CUCM Admin, CU Serviceability, CU Reporting) (yes/no):y
Do you want to enable SSO for Cisco Unified Operating System Administration (CUCM OS Admin, DRF) (yes/no):y
Do you want to enable SSO for Cisco Unity Connection Rest APIs (yes/no):y
Do you want to enable SSO for Cisco Unity Connection PCA and Web Inbox (yes/no):y
Do you want to enable SSO for Cisco Unity Connection Administration (yes/no):y
Do you want to enable SSO for Cisco Unified Data Service (CUCiMOC) (yes/no):y
Do you want to enable SSO for RTMT (yes/no):y
Do you want to enable SSO for Cisco Unified CM User Options (Cisco Unified CM User options) (yes/no):y
```

15.1.2 utils sso disable

This command disables SSO-based authentication. No parameters are required.

```
***** W A R N I N G *****
This command will restart Tomcat for successful completion.

This command needs to be executed on all the nodes in the cluster.

Do you want to continue (yes/no): y

List of apps for which SSO can be disabled

1) Cisco Unified CM Administration (CUCM Admin, CU Serviceability, CU Reporting)
2) Cisco Unified Operating System Administration (CUCM OS Admin, DRF)
3) Cisco Unity Connection Rest APIs
4) Cisco Unity Connection PCA and Web Inbox
5) Cisco Unity Connection Administration
6) Cisco Unified Data Service (CUCiMOC)
7) RTMT
8) Cisco Unified CM User Options (Cisco Unified CM User options)

Do you want to disable SSO for Cisco Unified CM Administration (CUCM Admin, CU Serviceability, CU Reporting) (yes/no):y
Do you want to disable SSO for Cisco Unified Operating System Administration (CUCM OS Admin, DRF) (yes/no):y
Do you want to disable SSO for Cisco Unity Connection Rest APIs (yes/no):y
Do you want to disable SSO for Cisco Unity Connection PCA and Web Inbox (yes/no):y
Do you want to disable SSO for Cisco Unity Connection Administration (yes/no):y
Do you want to disable SSO for Cisco Unified Data Service (CUCiMOC) (yes/no):y
Do you want to disable SSO for RTMT (yes/no):y
Do you want to disable SSO for Cisco Unified CM User Options (Cisco Unified CM User options) (yes/no):y
```

15.1.3 utils sso status

This command provides the status of SSO on Cisco Unity Connection 8.6. No parameters are required.

```
admin:utils sso status
SSO Status: Disabled
```

15.2 Cisco Unity Connection 8.6 SSO GUI

To configure SSO from GUI, log in to Cisco Unified OS Administration. Go to **Security > Single Sign on**. The following configuration window appears.

The screenshot shows the Cisco Unified Operating System Administration GUI. The page title is "Cisco Unified Operating System Administration" with the Cisco logo. The navigation bar includes "Navigation" and "Cisco Unified OS Administration". The user is logged in as "admin". The main menu includes "Show", "Settings", "Security", "Software Upgrades", "Services", and "Help". The current page is "SSO Applications Configuration".

There is a "Save" button at the top left of the configuration area. Below it are five input fields:

- Enter URL of the Open Access Manager(OpenAM) server*
- Enter the relative path where the policy agent should be deployed*
- Enter the name of the profile configured for this policy agent*
- Enter the password of the profile name*
- Enter the login module instance name configured for Windows Desktop SSO*

Below the input fields is a section titled "Select Applications" with a table of applications and their SSO status. All applications are currently disabled.

<input type="checkbox"/> Select All	Application name	SSO Status
<input type="checkbox"/>	Cisco Unified CM Administration (CUCM Admin, CU Serviceability, CU Reporting)	Disabled
<input type="checkbox"/>	Cisco Unified Operating System Administration (CUCM OS Admin, DRF)	Disabled
<input type="checkbox"/>	Cisco Unity Connection Rest APIs	Disabled
<input type="checkbox"/>	Cisco Unity Connection PCA and Web Inbox	Disabled
<input type="checkbox"/>	Cisco Unity Connection Administration	Disabled
<input type="checkbox"/>	Cisco Unified Data Service (CUCIMOC)	Disabled
<input type="checkbox"/>	RTMT	Disabled
<input type="checkbox"/>	Cisco Unified CM User Options (Cisco Unified CM User options)	Disabled

At the bottom of the configuration area is another "Save" button.

For enabling SSO from GUI, the same parameters are requested as requested from CLI. All the inputs are validated before enabling SSO to a selected application. A check box is provided for selecting the application(s); check the check box to select the applications on which you want to enable SSO and then click **Save**. After successful validation of all the inputs, a popup window is displayed saying "Enabling/Disabling SSO for the applications will cause Tomcat to restart. Do you want to continue?" Click **OK** if you want to proceed or click **Cancel** to cancel.

In the following figure, SSO is enabled for all applications.

Cisco Unified Operating System Administration
For Cisco Unified Communications Solutions

Navigation Cisco Unified OS Administration
akaagarwssso Search Document

Show Settings Security Software Upgrades Services Help

SSO Applications Configuration

Save

Status

Warning: Changing the SSO settings causes an immediate Tomcat restart

Server Settings:

Enter URL of the Open Access Manager (OpenAM) server*

Enter the relative path where the policy agent should be deployed*

Enter the name of the profile configured for this policy agent*

Enter the password of the profile name*

Enter the login module instance name configured for Windows Desktop SSO*

Select Applications

<input type="checkbox"/> Select All	Application name	SSO Status
<input checked="" type="checkbox"/>	Cisco Unified CM Administration (CUCM Admin, CU Serviceability, CU Reporting)	Enabled
<input checked="" type="checkbox"/>	Cisco Unified Operating System Administration (CUCM OS Admin, DRF)	Enabled
<input checked="" type="checkbox"/>	Cisco Unity Connection Rest APIs	Enabled
<input checked="" type="checkbox"/>	Cisco Unity Connection PCA and Web Inbox	Enabled
<input checked="" type="checkbox"/>	Cisco Unity Connection Administration	Enabled
<input checked="" type="checkbox"/>	Cisco Unified Data Service (CUCIMOC)	Enabled
<input checked="" type="checkbox"/>	RTMT	Enabled
<input checked="" type="checkbox"/>	Cisco Unified CM User Options (Cisco Unified CM User options)	Enabled

Save

16 OpenSSO Enterprise Session Failover

Note: Refer this chapter if you wish to configure the session failover

Session failover ensures that session data remains accessible to OpenSSO Enterprise servers and OpenSSO Enterprise Policy Agents. Service requests are routed to a failover server, the user's session continues uninterrupted, and no user data is lost. The OpenSSO Enterprise Session Service maintains authenticated session states and continues processing new client requests subsequent to the failure. In most cases, without session failover, after system failure and subsequent service recovery, the user would have to re-authenticate.

Session failover is critical when end users' transactions involve financial data or other sensitive information that is difficult to recover when a system failure occurs. With session failover, when a system failover occurs, the user's transaction can proceed uninterrupted. Session failover is less important if end users are, for example, reading but not writing data.

When you configure OpenSSO Enterprise for session failover, the user's authenticated session state is stored in the Berkeley Database in the event of a single hardware or software failure. In session failover deployments, you configure the OpenSSO Enterprise servers to communicate with Message Queue brokers which manage session state persistence in the Berkeley Database. This configuration enables the user's session to fail over to a backup OpenSSO Enterprise server without losing any session state information. The user does not have to login again. The backup OpenSSO Enterprise server is determined among the available servers in the configuration list by an internal algorithm.

This type of deployment ensures the state availability even if one of the OpenSSO Enterprise servers is inaccessible due to scheduled maintenance, hardware failure, or software failure. However, the single load balancer can be a single point of failure. When this load balancer is inaccessible, no OpenSSO Enterprise services or session data are available to the Policy Agents.

16.1 Requirements for Access Manager Session Failover (AMSFO)

Key components that are required for basic session failover in an OpenSSO Enterprise deployment for high availability are:

- A single load balancer distributes the workload among multiple OpenSSO Enterprise servers. This increases transaction throughput, and ensures failover when a system failure occurs.
- Multiple OpenSSO Enterprise servers with respective embedded Directory Servers act as backups when system failure occurs. Embedded Directory Servers ensure that replicated configuration data is always available even during system failure.
- When OpenSSO Enterprise is configured for session failover, a Java Message Queue Broker Cluster replicates session data and stores it in the Berkeley Database. When a system failure occurs, the replicated session data is made available to Policy Agents so that the end user does not lose data and does not have to re-authenticate after system recovery.
- Multiple Berkeley Databases are used to store session data, and are configured for session failover. If one Berkeley Database fails, the working Berkeley Database can provide session data to the OpenSSO Enterprise servers for session validation.

In all examples in this chapter, load balancers represent the only access points to OpenSSO Enterprise servers. An access point can be any hardware or software that acts as a load balancer, and is associated with a site, that is installed in front of OpenSSO Enterprise servers. Policy Agents interact with OpenSSO Enterprise servers through these access points.

16.2 Configuration of AMSFO components

16.2.1 Installation and Configuration of Load Balancer

16.2.1.1 Installation of Load Balancer on Linux Platform

To configure load balancer on Linux platform, install the Red Hat Enterprise Linux 5.5 (lower version of RHEL can also be used). After RHEL is installed, configure network settings and create an entry in DNS servers for this RHEL host. After everything is configured on this RHEL server, we will install Sun Java System Web Server application for load balancing.

In this guide we will be configuring Sun Java System Web Server as the load balancer; you can get the Sun Java System Web Server setup file from https://cds.sun.com/is-bin/INTERSHOP.enfinity/WFS/CDS-CDS_SMI-Site/en_US/-/USD/ViewProductDetail-Start?ProductRef=SJWS-7-TechPrvw-OTH-G-Beta@CDS-CDS_SMI URL.

On the above download window, select the platform as Linux and download the installation file.

Copy the installation file to the above RHEL server at any location and run the setup.

```
+++++  
[root@ssoloadbal Sun Java System Web Server]# ./setup
```

Welcome to the Oracle iPlanet Web Server 7.0.9 installation wizard.

Copyright (c) 2007, 2010, Oracle and/or its affiliates. All rights reserved.

Oracle and Java are registered trademarks of Oracle and/or its affiliates.

Other names may be trademarks of their respective owners.

You will be asked to specify preferences that determine how Oracle iPlanet Web Server 7.0.9 is installed and configured.

The installation program pauses as questions are presented so you can read the information and make your choice. When you are ready to continue, press Enter (Return on some keyboards).

<Press ENTER to Continue>

Some questions require that you provide more detailed information. Some questions also display default values in brackets []. For example, yes is the default answer to the following question:

Are you sure? [yes]

To accept the default, press Enter.

To provide a different answer, type the information at the command prompt and then press Enter.

<Press ENTER to Continue>

Oracle iPlanet Web Server components will be installed in the directory listed below, referred to as the installation directory. To use the specified directory, press Enter. To use a different directory, enter the full path of the directory and press Enter.

Oracle iPlanet Web Server Installation Directory [/opt/oracle/webserver7]

{"<" goes back, "!" exits}:

Specified directory /opt/oracle/webserver7 does not exist

Create Directory? [Yes/No] [yes] {"<" goes back, "!" exits} yes

Select the Type of Installation

- 1. Express*
- 2. Custom*
- 3. Exit*

What would you like to do [1] {"<" goes back, "!" exits}? 1

Choose a user name and password. You must remember this user name and password to administer the Web Server after installation.

Administrator User Name [admin] {"<" goes back, "!" exits}

Administrator Password:

Retype Password:

Product : Oracle iPlanet Web Server

Location : /opt/oracle/webserver7

Disk Space : 231.37 MB

Administration Command Line Interface

Server Core

Start Administration Server [yes/no] [yes] {"<" goes back, "!" exits}: yes

Ready to Install

- 1. Install Now*
- 2. Start Over*
- 3. Exit Installation*

What would you like to do [1] {"<" goes back, "!" exits}? 1

Installing Oracle iPlanet Web Server

| -1%-----25%-----50%-----75%-----100%|

Installation Successful.

Refer to the installation log file at:

/opt/oracle/webserver7/setup/install.log for more details.

Next Steps:

- You can access the Administration Console by accessing the following URL:

https://ssoloadbal.vrajoli.com:8989

[root@ssoloadbal Sun Java System Web Server]#

+++++

16.2.1.2 Installation of Load Balancer on Windows Platform

To configure load balancer on windows platform, install Windows XP SP2/Windows Server 2003. After the Windows OS is installed, configure network settings and create an entry in DNS servers for this host. After everything is configured on this Windows machine, we will install Sun Java System Web Server application for load balancing.

In this guide we will be configuring Sun Java System Web Server as the load balancer; you can get the Sun Java System Web Server setup file from https://cds.sun.com/is-bin/INTERSHOP.enfinity/WFS/CDS-CDS_SMI-Site/en_US/-/USD/ViewProductDetail-Start?ProductRef=SIWS-7-TechPrvw-OTH-G-Beta@CDS-CDS_SMI URL.

On the above download window, select the platform as Windows xp/Windows Server 2003 and download the installation file.

Copy the installation file to the above Windows server at any location and run the setup.

Below is the example of setup done in this guide. The accompanying figures show the default settings, which you can use during installation.

+++++

```
C:\Documents and Settings\Administrator\Desktop\sjsws-7_0u1-windows-i586>dir
```

```
Volume in drive C has no label.
```

```
Volume Serial Number is 1C23-BFE7
```

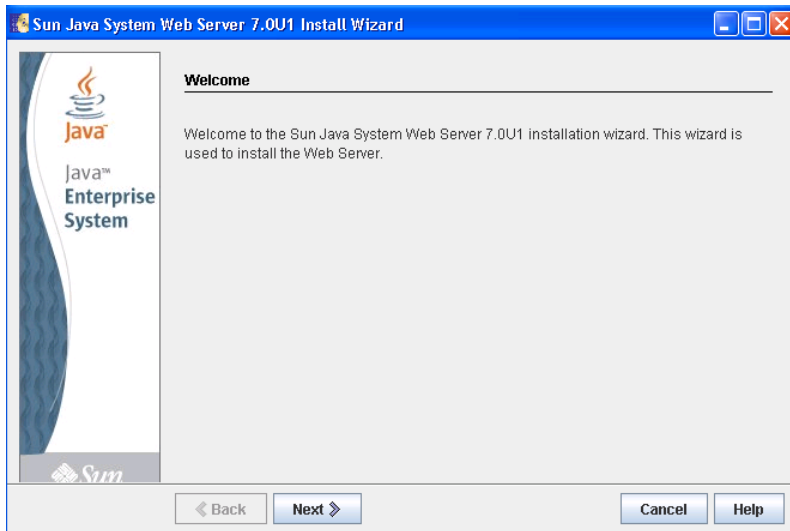
```
Directory of C:\Documents and Settings\Administrator\Desktop\sjsws-7_0u1-windows-i586
```

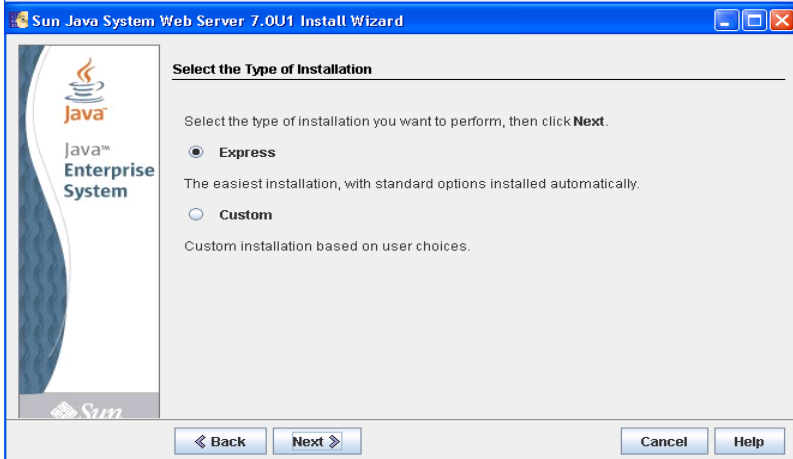
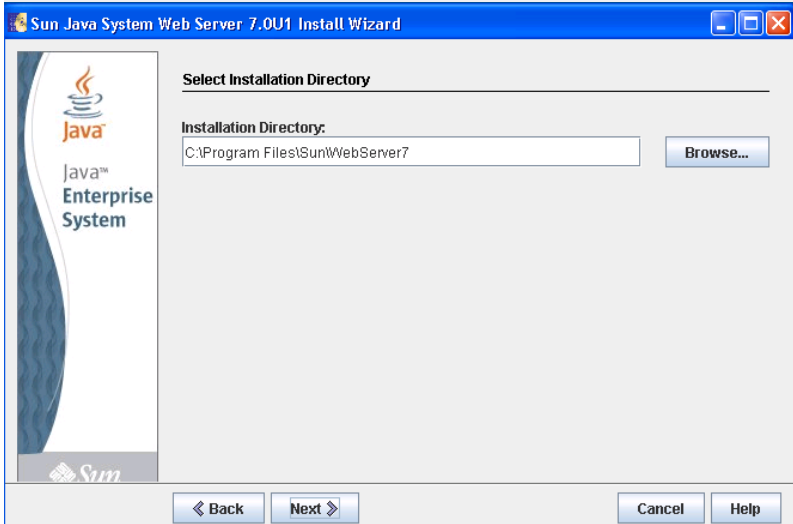
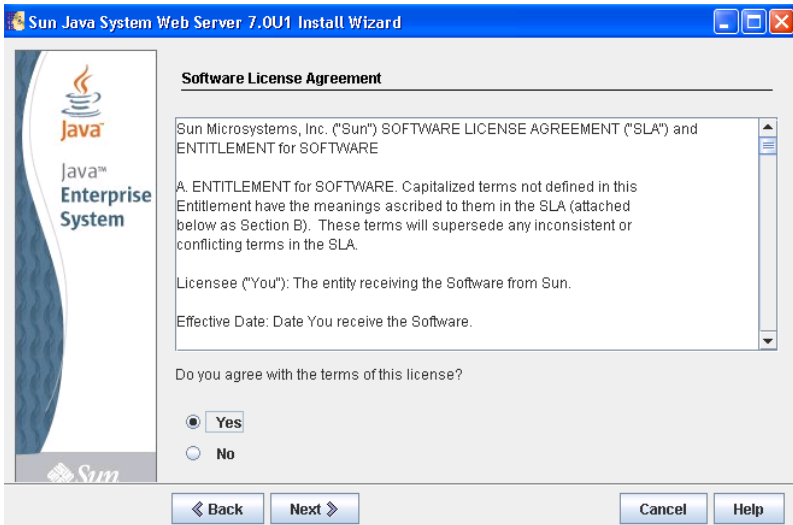
```
09/25/2010 10:35 PM <DIR>      .
09/25/2010 10:35 PM <DIR>      ..
09/25/2010 10:35 PM <DIR>      Legal
09/25/2010 01:20 PM          751 README.txt
09/25/2010 01:22 PM          20,603 setup.exe
09/25/2010 10:34 PM <DIR>      WebServer
```

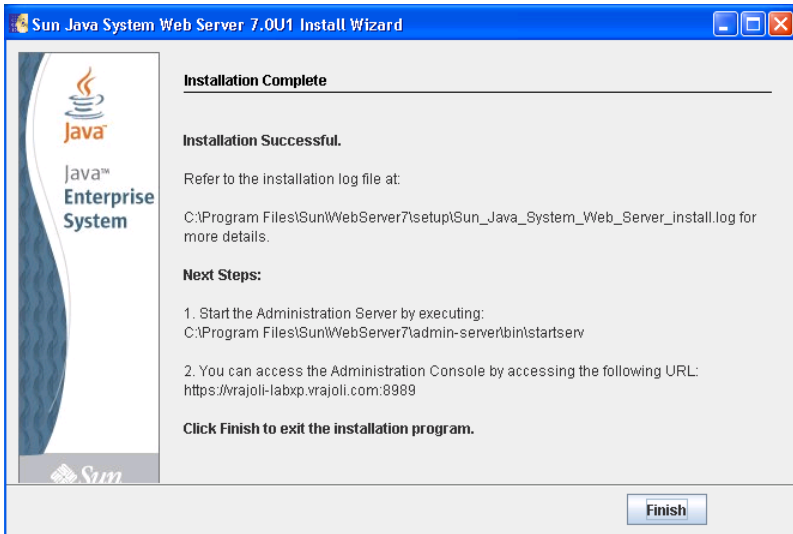
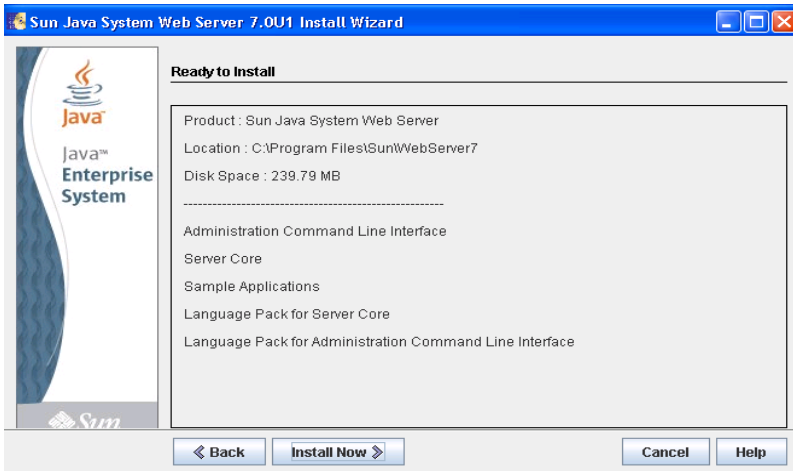
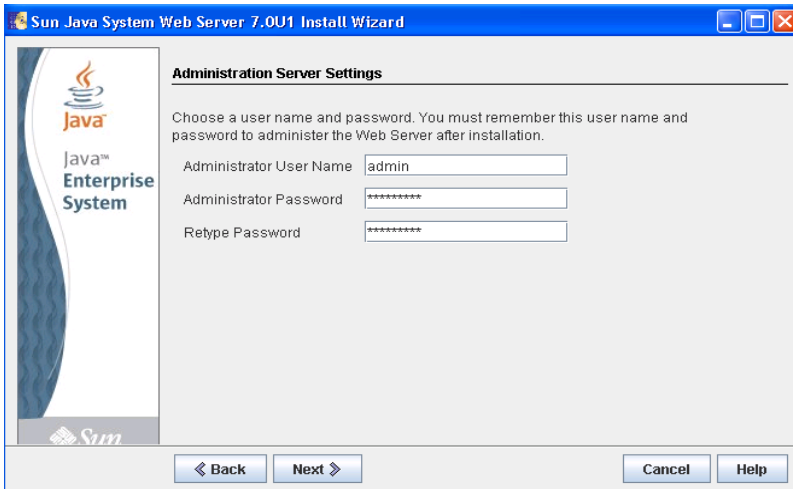
```
2 File(s)    21,354 bytes
```

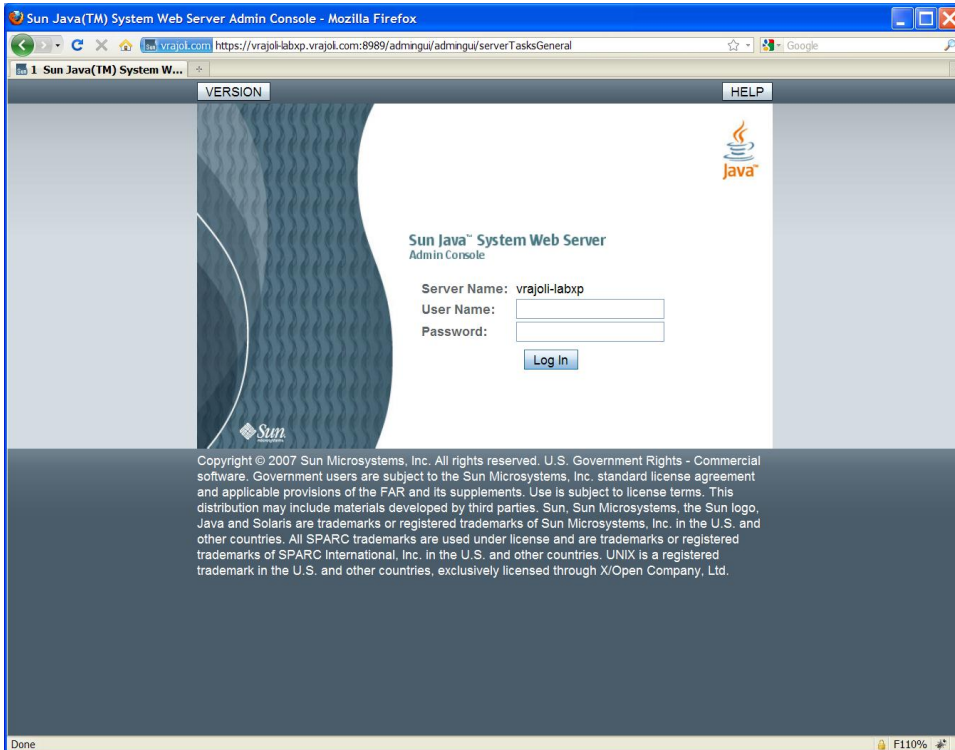
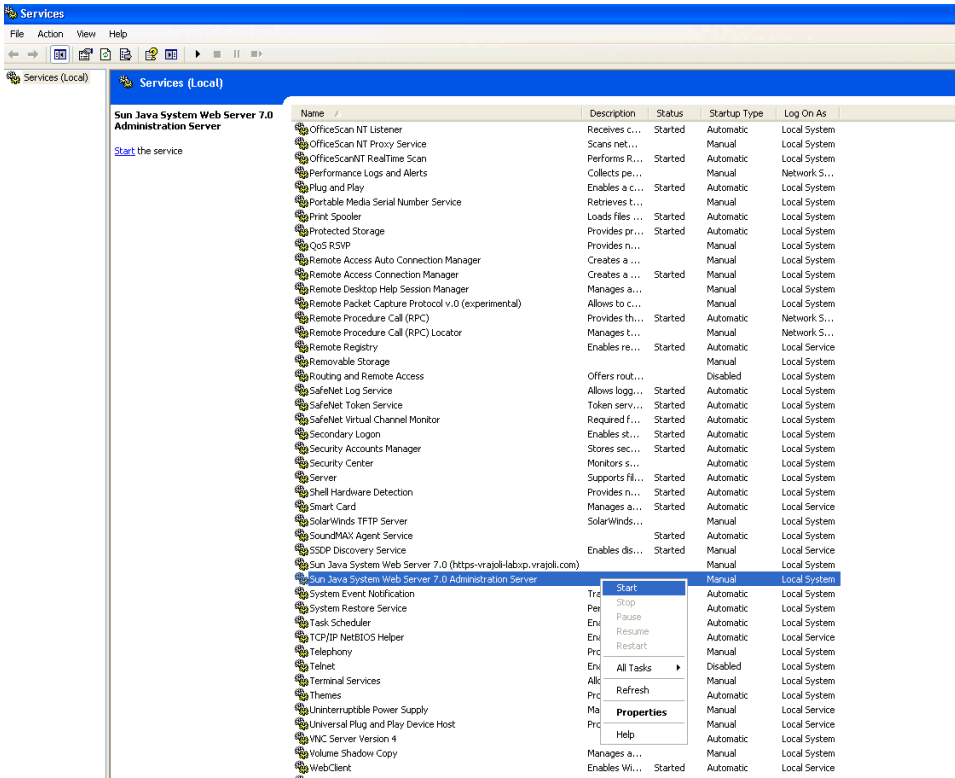

4 Dir(s) 8,806,412,288 bytes free

C:\Documents and Settings\Administrator\Desktop\sjsws-7_0u1-windows-i586>setup.exe





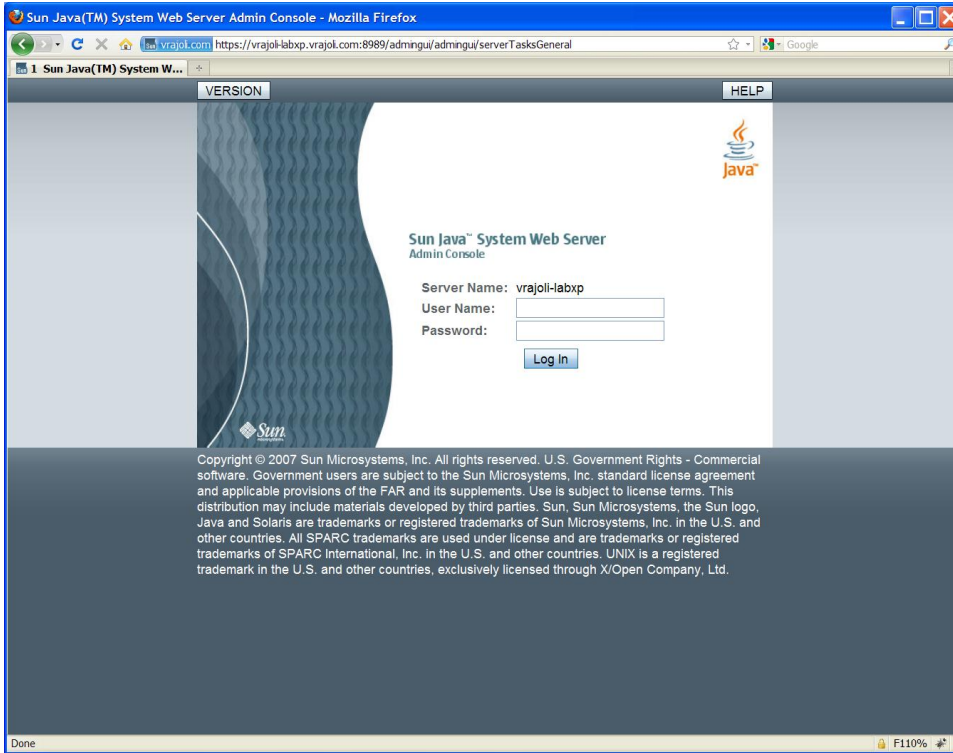




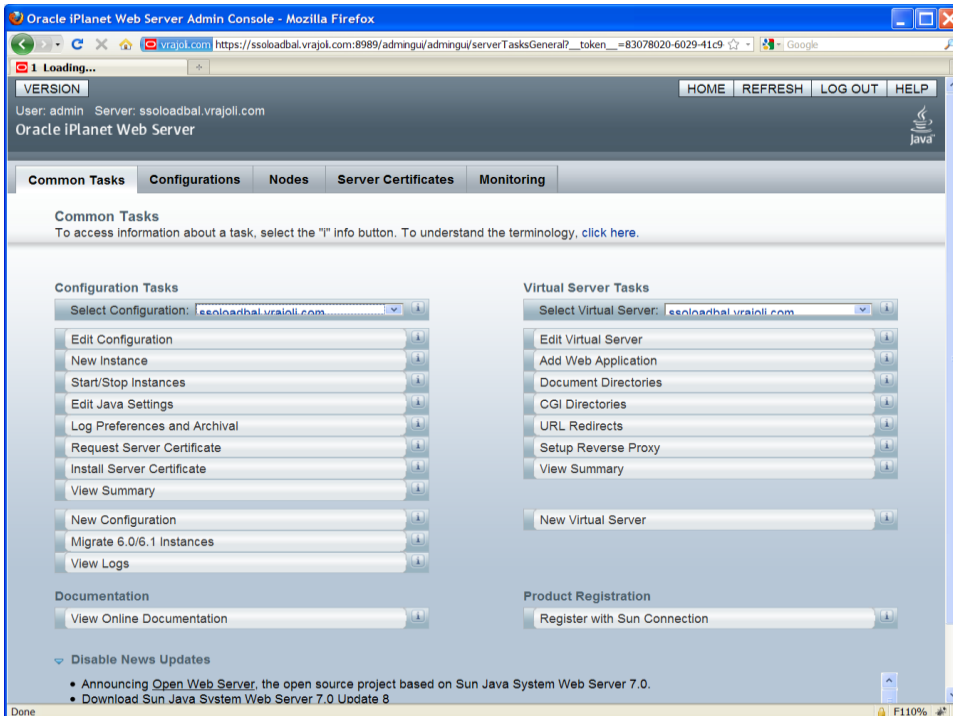
+++++

16.2.1.3 Configuration of Load Balancer for HTTP Load Balancing

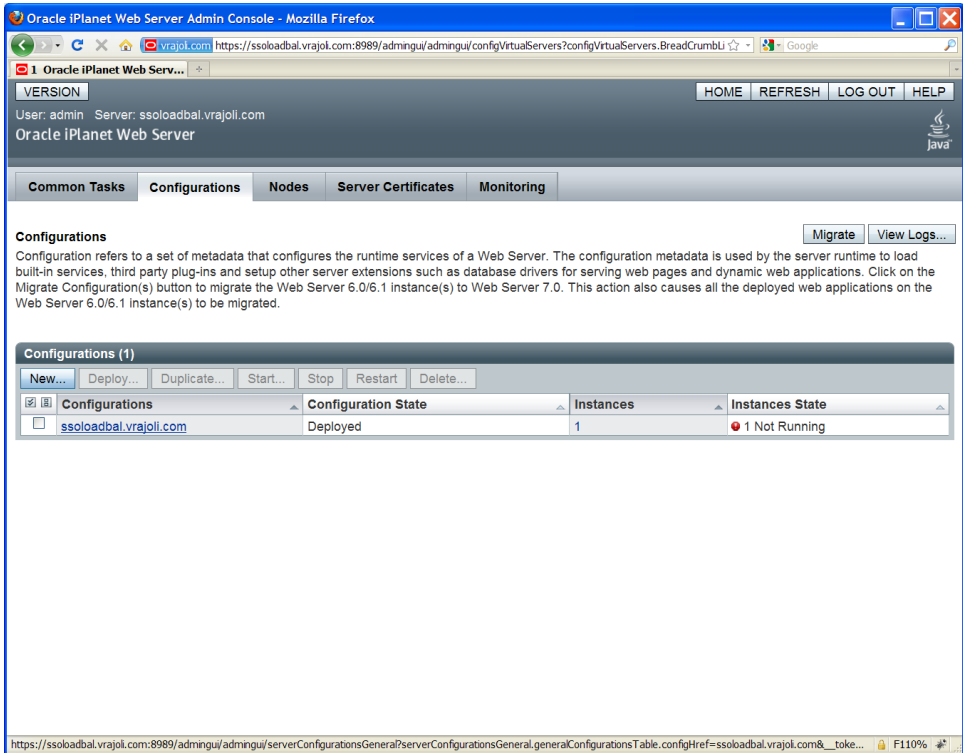
Browse the Load Balancer URL, <https://ssoloadbal.vrajoli.com:8989>; you will be presented with following login window. Log in to the admin console.



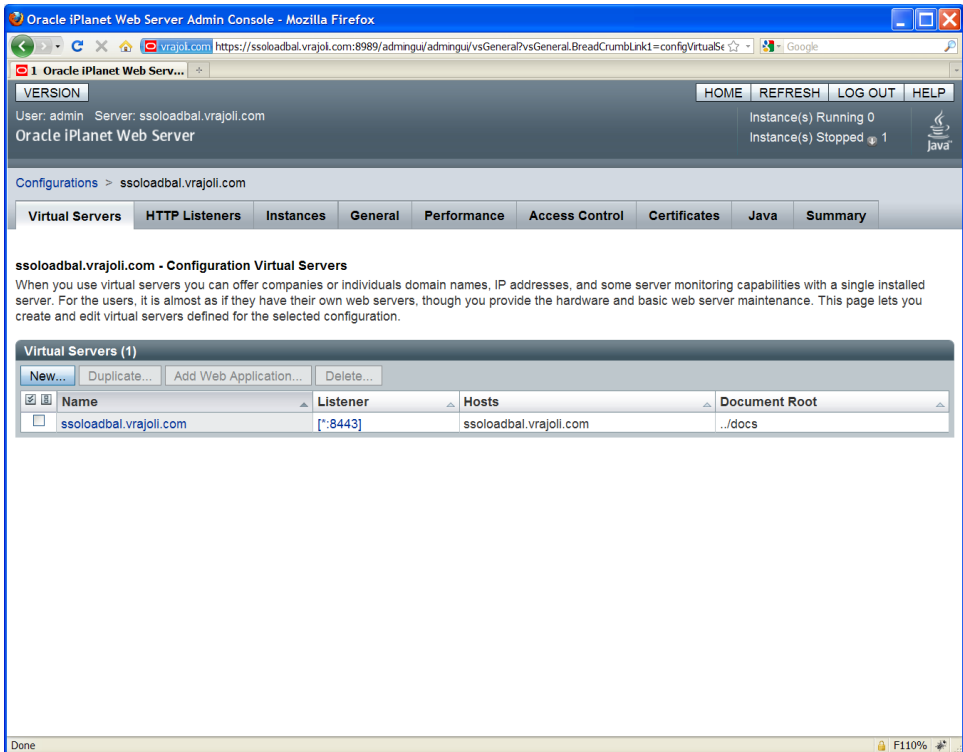
After login, the following window appears:



Click the **Configurations** tab, and then click on the load balancer name (ssoloadbal.vrajoli.com) as shown in the following figure (Configurations > Load Balancer Name).

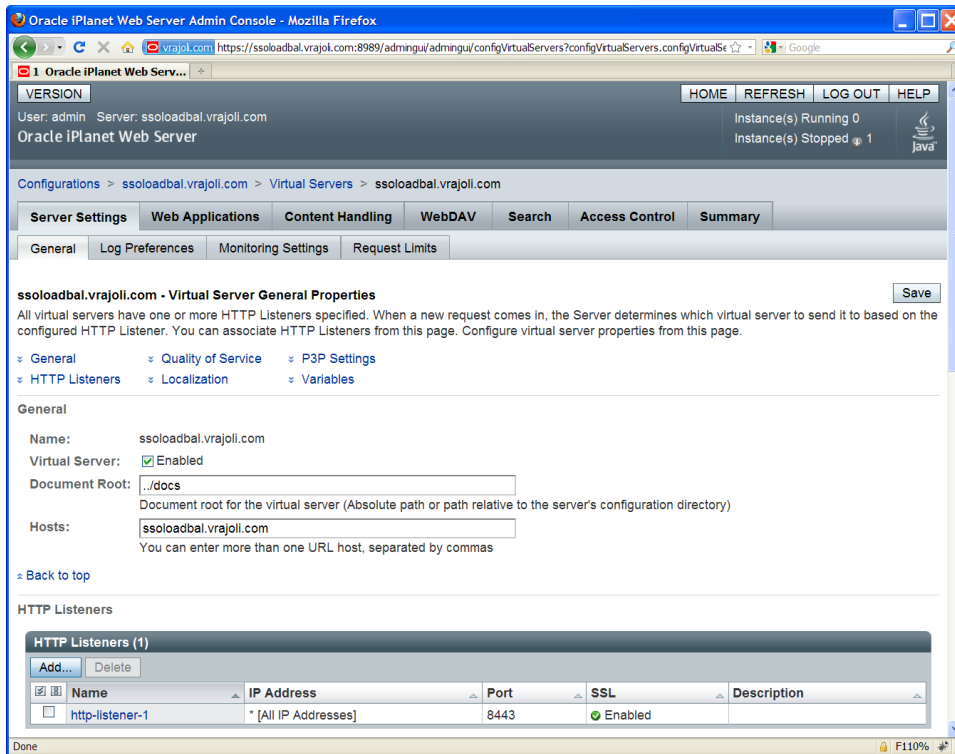


After you click on the load balancer name, the following window appears.

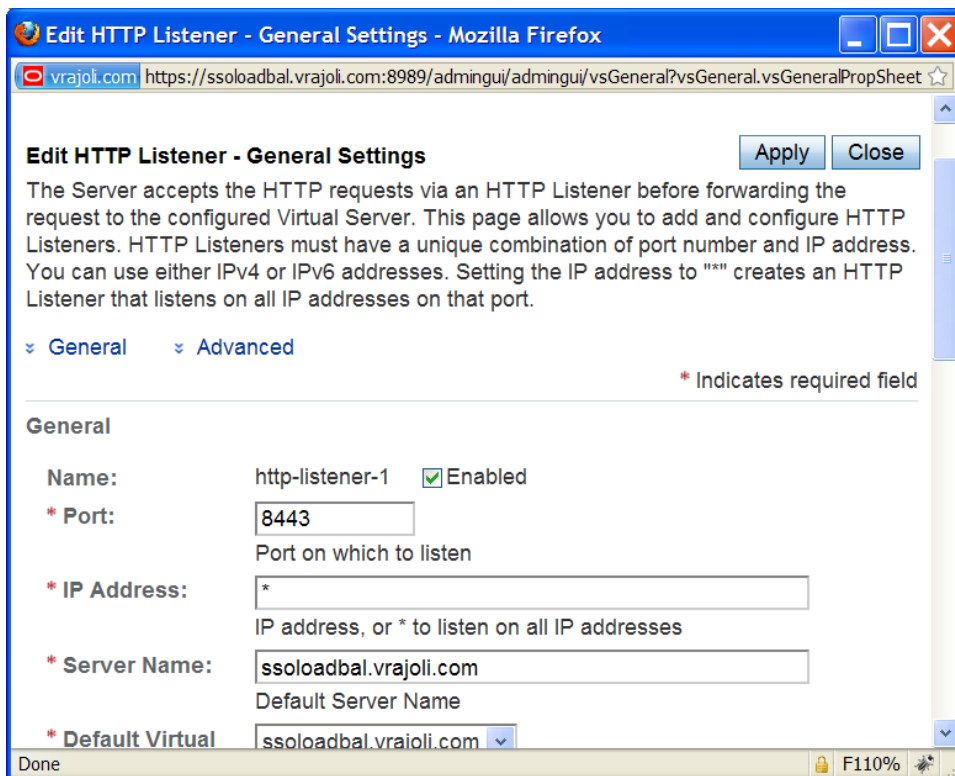


Click on the Load Balancer Name (ssoloadbal.vrajoli.com) under Virtual Servers.

The following window appears.



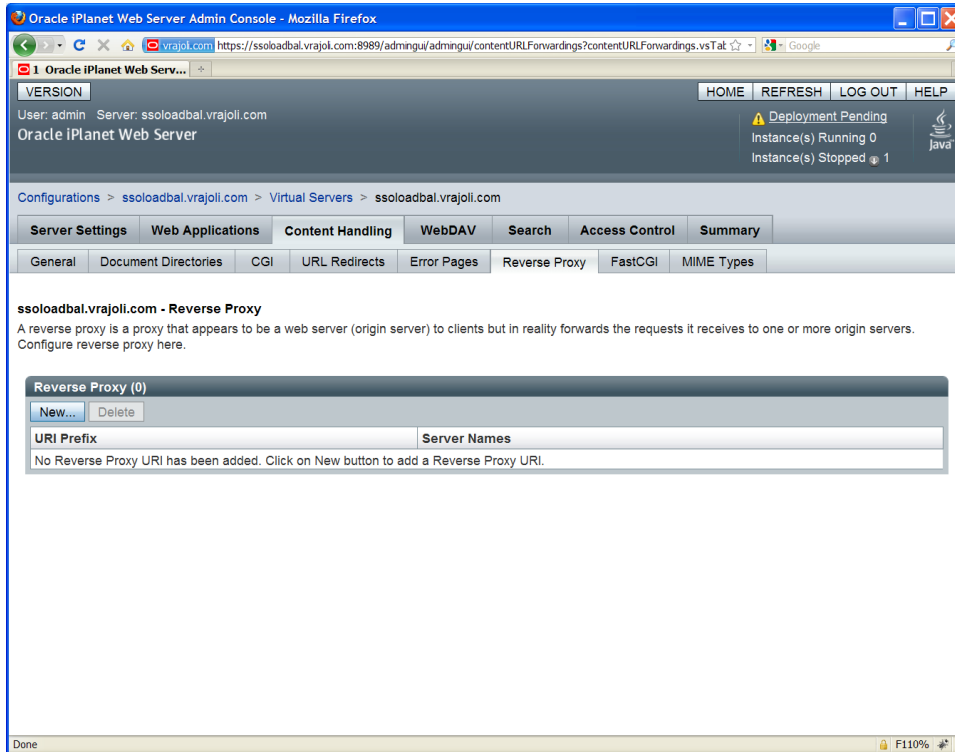
Click on **HTTP Listeners**, and then click on **http-listener-1** and set the port value to 8443 as shown below. Click **Apply** and then click **Close**.



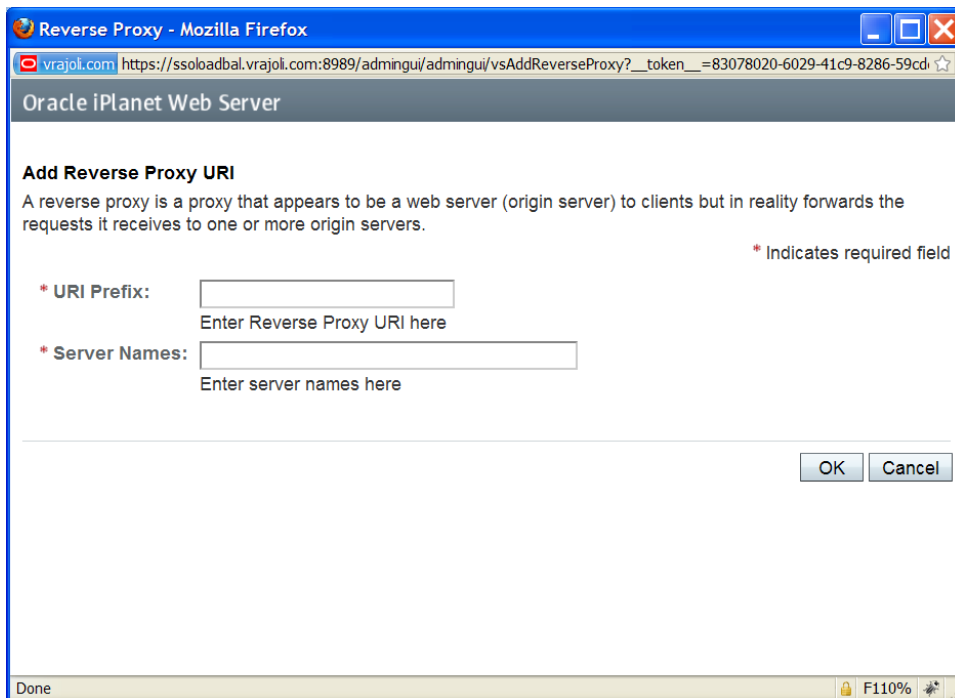
Then click the **Content Handling** tab under **Configurations > ssoloadbal.vrajoli.com > Virtual Servers > ssoloadbal.vrajoli.com**.

In the Content handling tab, select the **Reverse Proxy** tab.

Click **New**.



The following window appears.

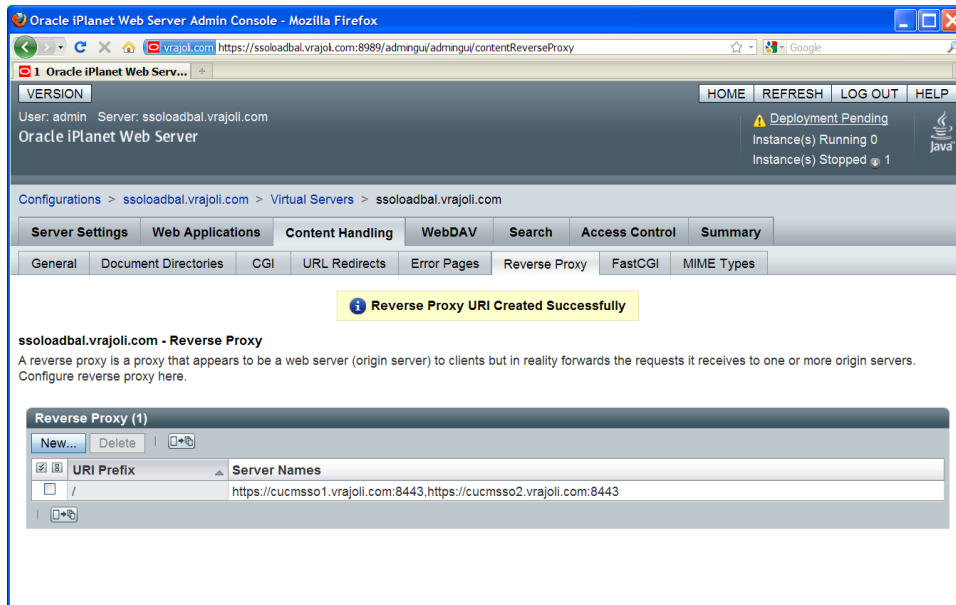


Under URI Prefix, enter / and under Server Names enter the OpenAM Enterprise server hostname. In this guide, we have two OpenAM Enterprise servers, namely cucmssso1.vrajoli.com and cucmssso2.vrajoli.com. Click **OK**.

URI Prefix: /

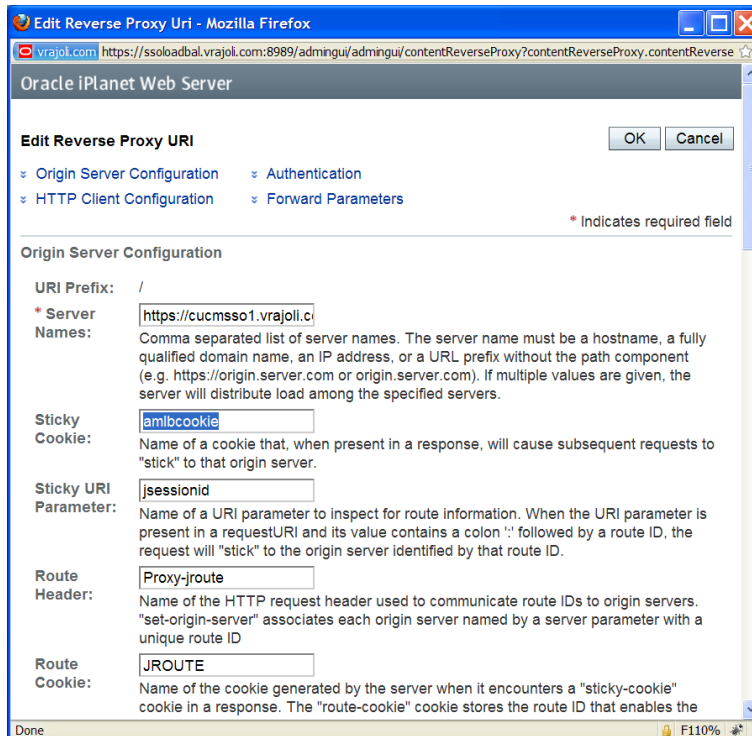
Server Names: https://cucmssso1.vrajoli.com:8443,https://cucmssso2.vrajoli.com:8443

Later you see the message “Reverse Proxy URI Created Successfully.”



Click on / URI Prefix, and modify the sticky cookie value to **amlbcookie** instead of JSESSIONID.

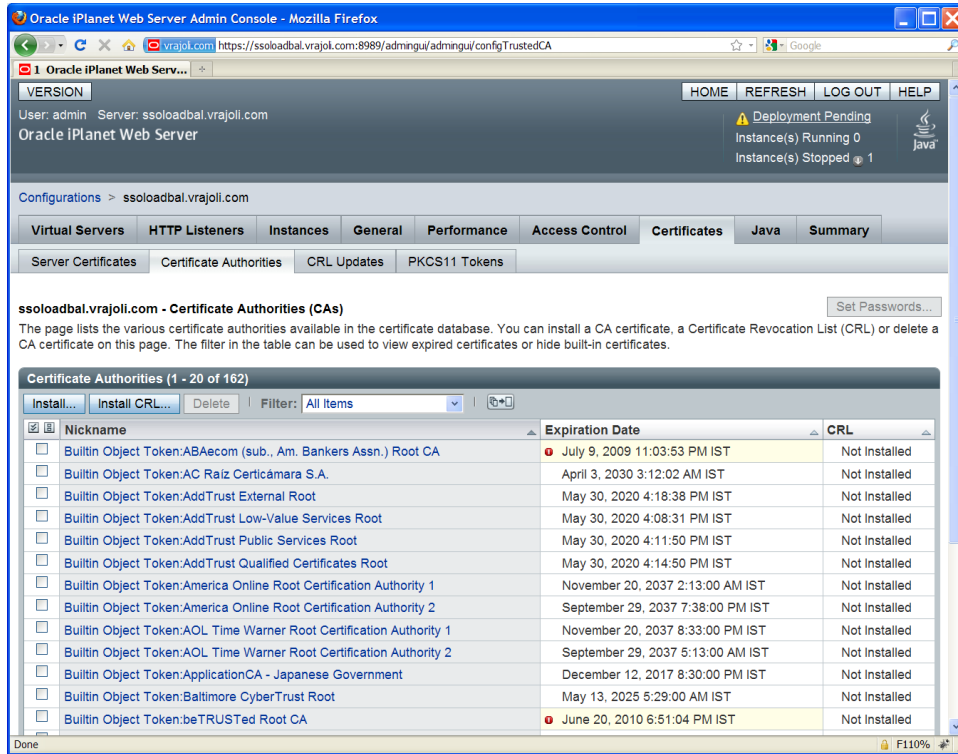
Click **OK**.



Now you must import the OpenAM Enterprise Server 1 (cucmss01.vrajoli.com) certificate and OpenAM Enterprise Server 2 (cucmss02.vrajoli.com) certificate to the Load Balancer.

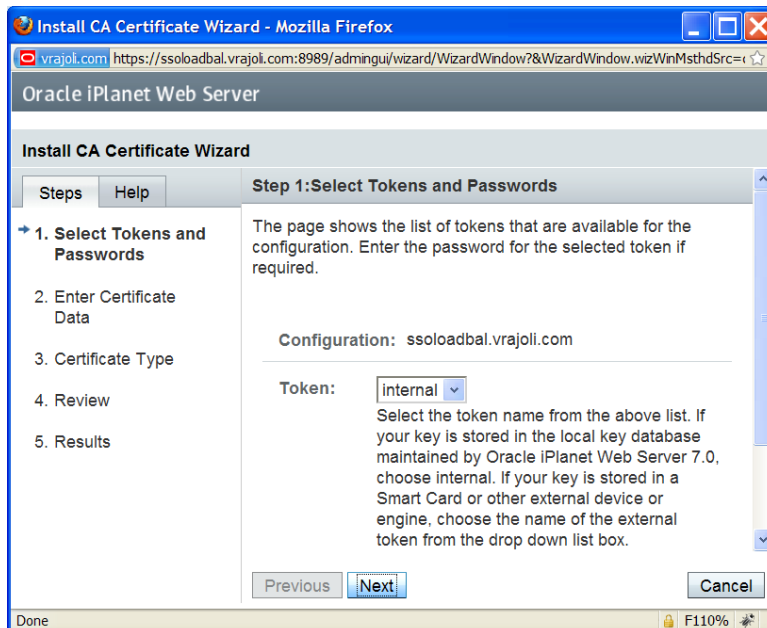
Copy OpenAM Enterprise Server 1(cucmss01.vrajoli.com) and OpenAM Enterprise Server 2 (cucmss02.vrajoli.com) certificate to Load Balancer box to any location.

Now, on the Load Balancer, Go to **Configurations > ssoloadbal.vrajoli.com > Certificates > Certificate Authorities** tab. The following window appears.



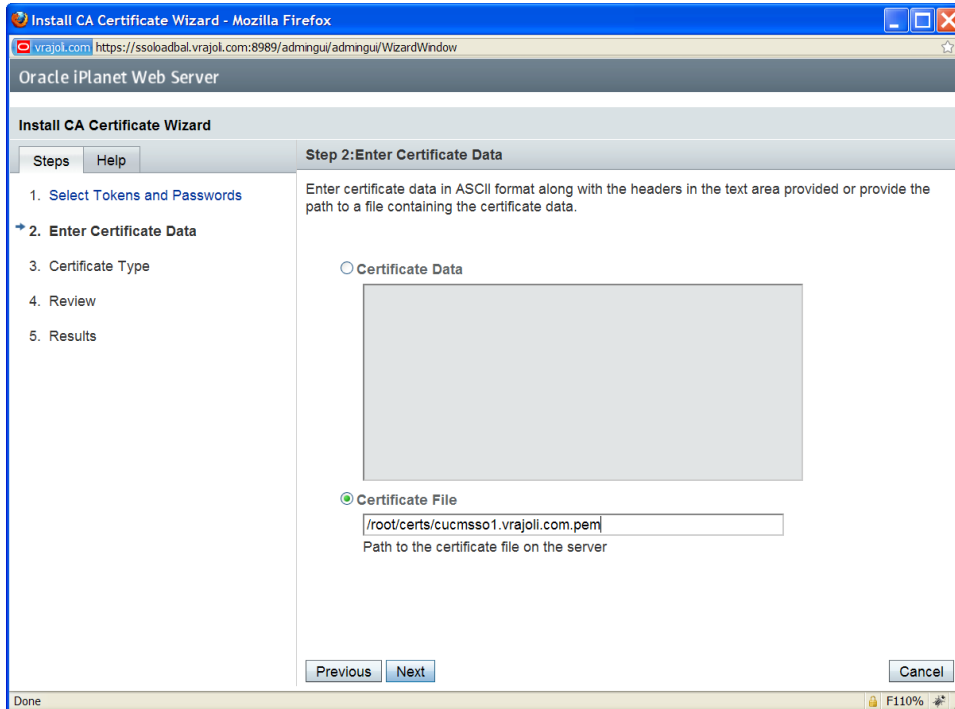
Click the **Install** button. The Install CA Certificate Wizard appears.

Click **Next**.



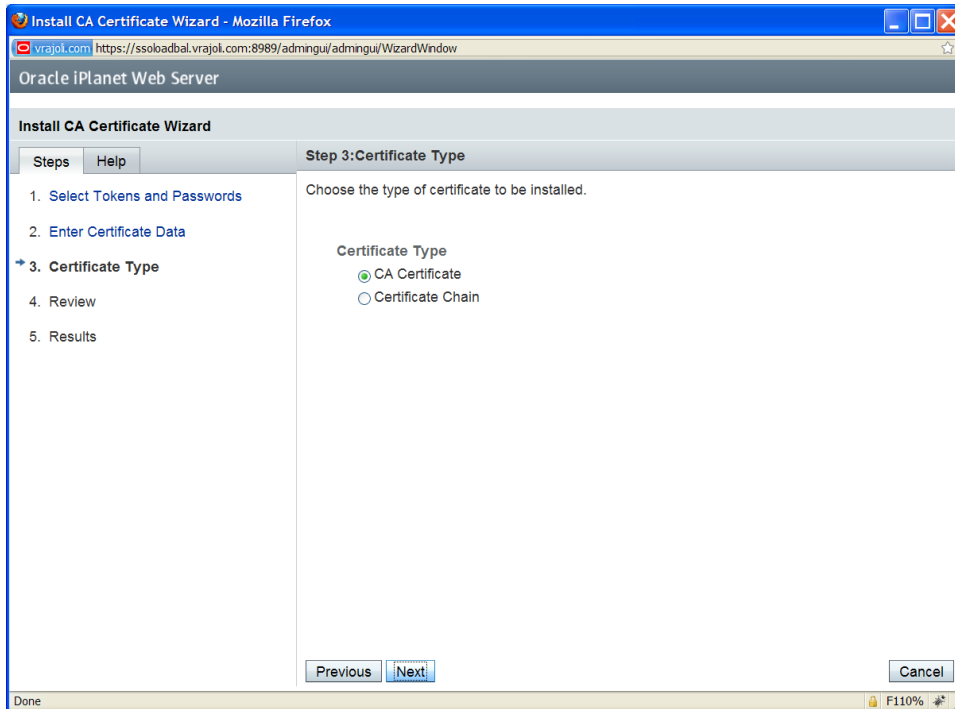
Click the **Certificate File** radio button and specify the path where you have stored the OpenAM Enterprise Server 1 certificate.

Click **Next**.

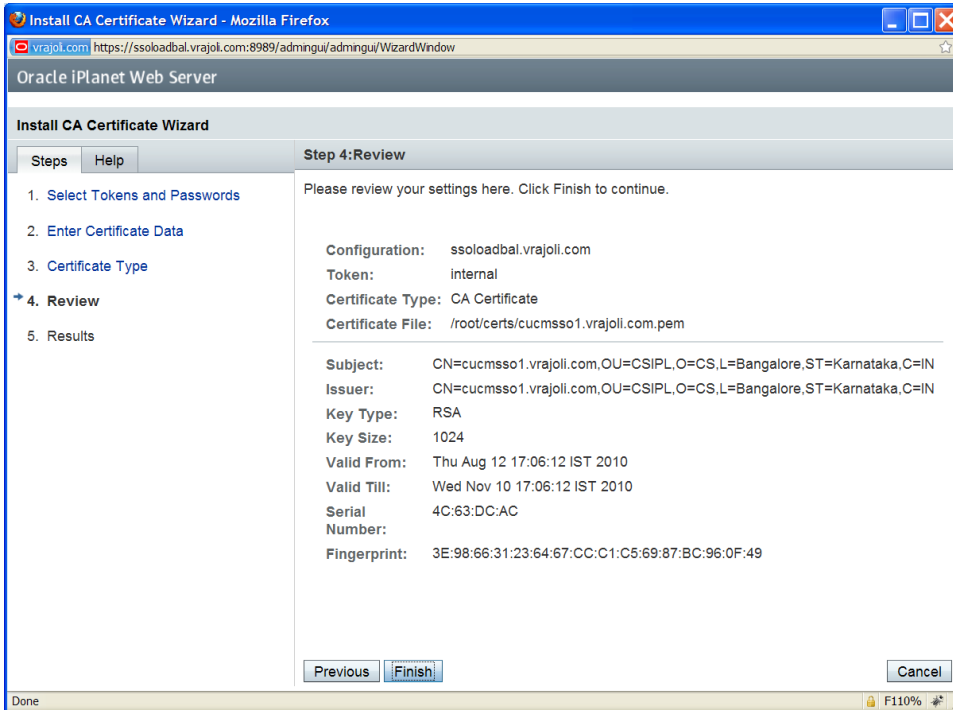


The following window appears.

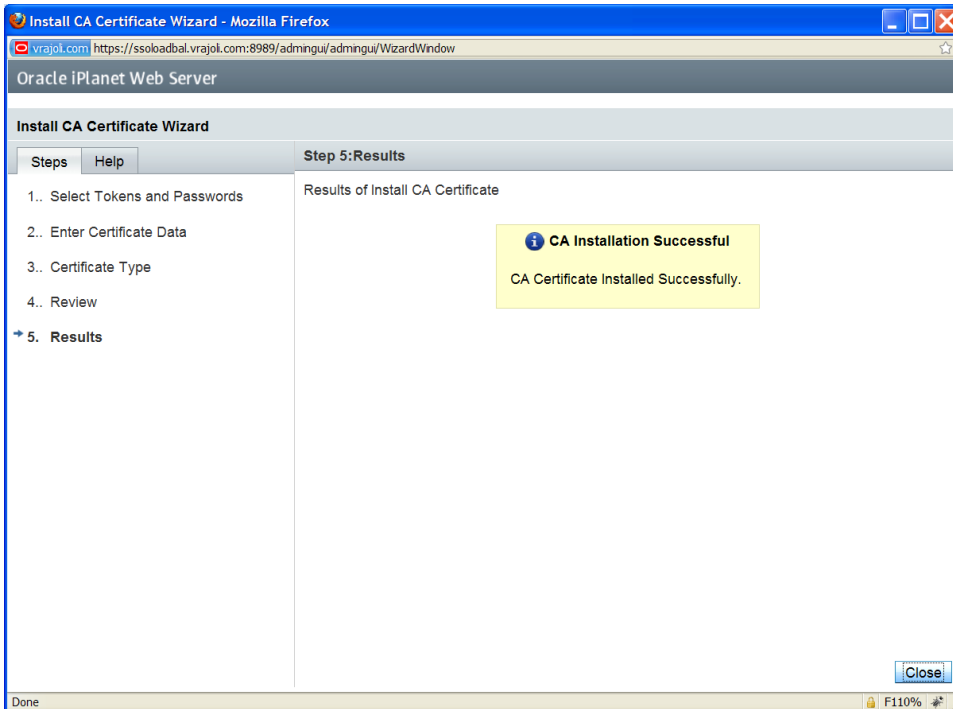
Click **Next**.



Click **Finish**.

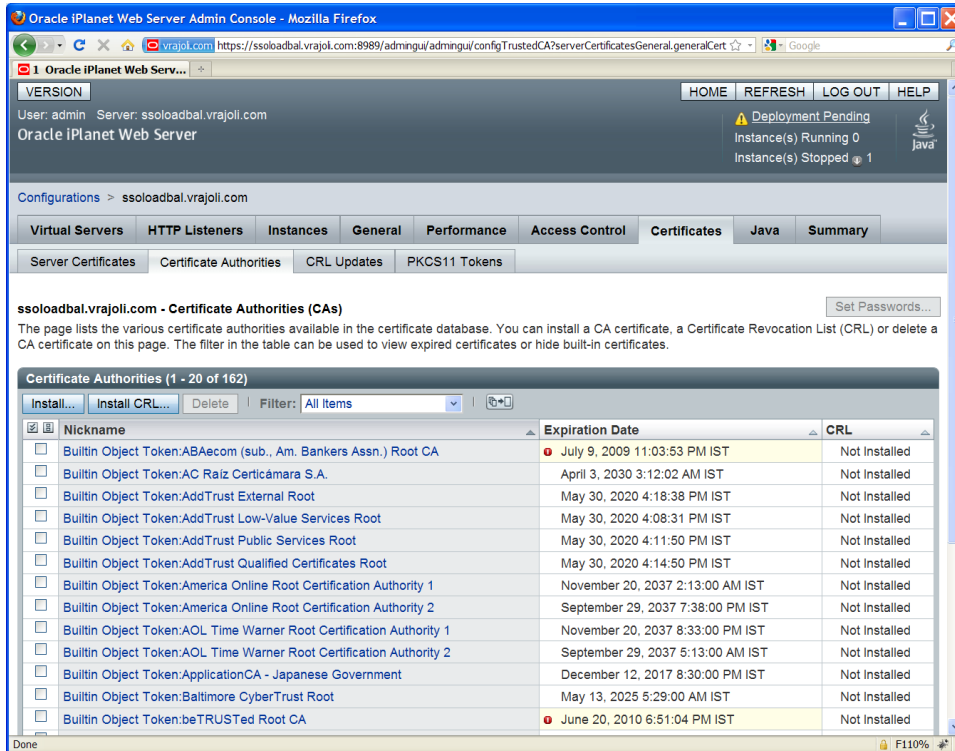


The following window appears, confirming you successfully installed the CA Certificate.



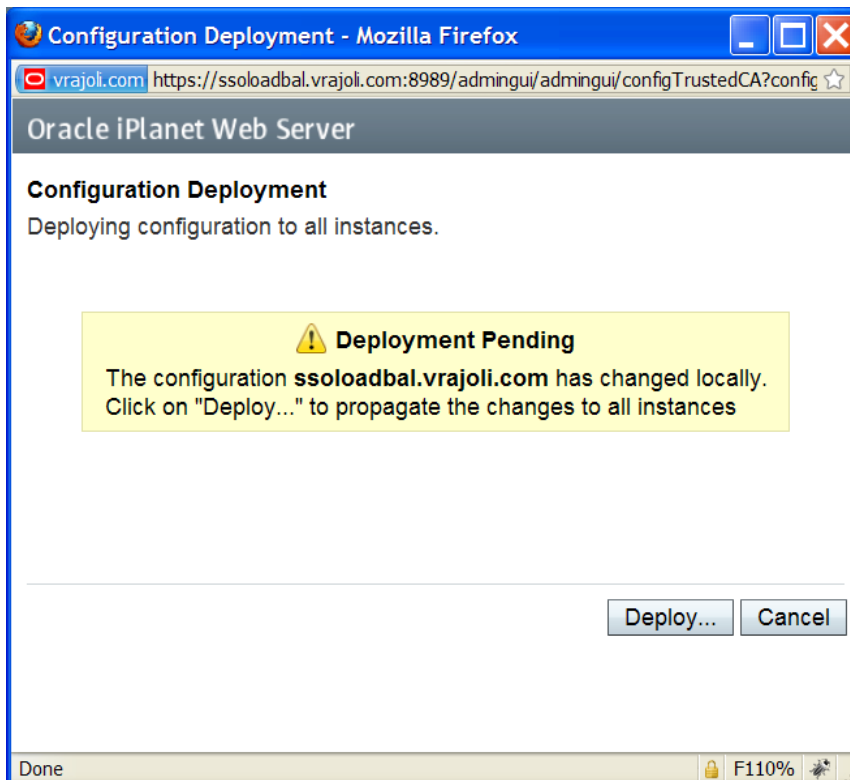
Use the same procedure to install the OpenAM Enterprise Server 2 (cucmsso2.vrajoli.com) certificate.

After importing all the OpenAM Enterprise server certificates to Load Balancer, click the **Deployment Pending** link on the top right corner.



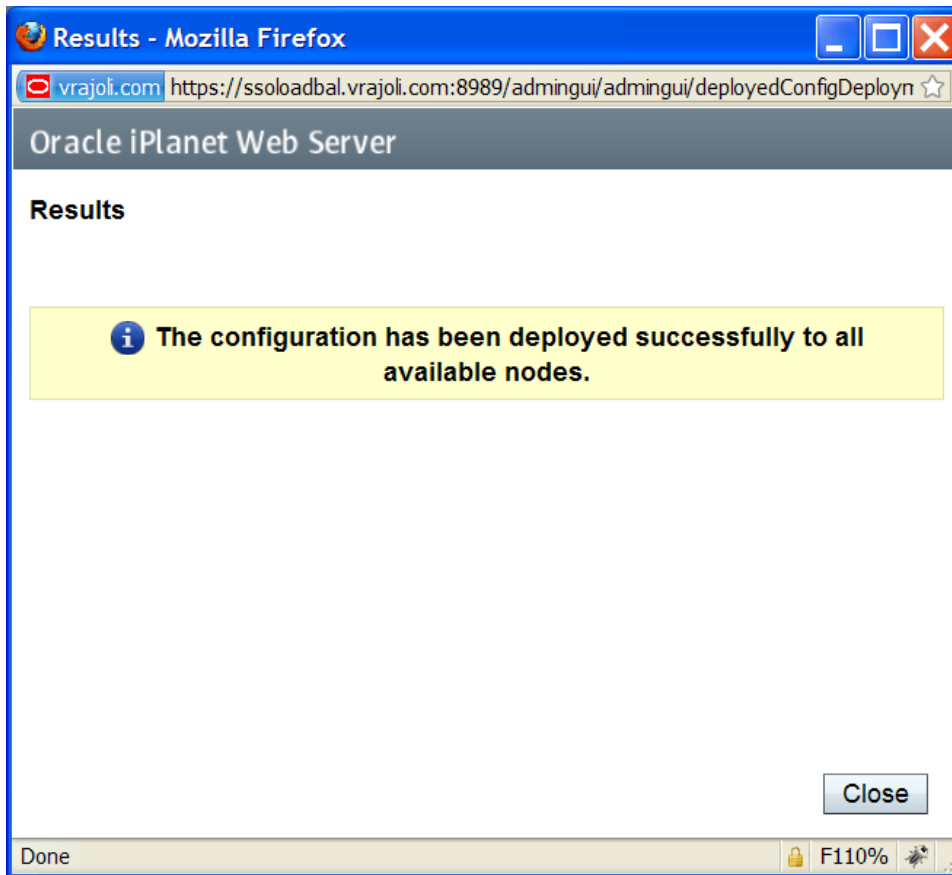
The following window appears.

Click the **Deploy** button.



The following window appears, indicating successful deployment.

Click **Close**.



16.2.2 Installation and Configuration of Session Failover Components

16.2.2.1 Configuration of Session Failover Components on Linux Platform

Prerequisites: one or more OpenAM Enterprise servers (for session failover) installed and configured on Linux platform. For installing OpenAM Enterprise on Linux platform, please refer to sections 6.1, 7.1 and 9.1.

In this guide, we have two OpenAM Enterprise servers for session failover.

OpenAM Enterprise Server 1 > cucmssso1.vrajoli.com

OpenAM Enterprise Server 2 > cucmssso2.vrajoli.com

Install the OpenSSO Enterprise session failover components on the cucmssso1.vrajoli.com host machine and the cucmssso2.vrajoli.com host machine.

To Install Session Failover Components on cucmssso1.vrajoli.com OpenAM Enterprise server on Linux

As a root user, log in to the cucmssso1.vrajoli.com host machine.

1. Create a directory into which the MessageQueue and BerkeleyDatabase bits can be downloaded and change into it.

```
# mkdir -p /export/SFO
```

```
# cd /export/SFO
```

2. Copy ssoSessionTools.zip to the cucmsso1.vrajoli.com host machine; ssoSessionTools.zip is included in the openam_release9_20100207.zip file under the tools directory. (openam_release9_20100207/opensso/tools/ssoSessionTools.zip)
3. Unzip ssoSessionTools.zip.

```
# cd /export/SFO
```

```
# unzip ssoSessionTools.zip -d ssoSessionTools
```

4. Modify the permissions on the setup script and run it to initialize the session failover tools.

```
# cd /export/SFO/ssoSessionTools
```

```
# chmod +x setup
```

```
# ./setup
```

5. When prompted, enter **opensso** as the directory to install the scripts (example: opensso).

Note: The directory location should be relative to the current directory.

When the script is finished, the following messages are displayed:

```
The scripts are properly setup under directory
```

```
/export/SFO/ssoSessionTools/opensso
```

```
JMQ is properly setup under directory
```

```
/export/SFO/ssoSessionTools/jmq
```

6. Change to the bin directory.

```
# cd /export/SFO/ssoSessionTools/jmq/imq/bin
```

7. Run the **imqbrokerd** command to create a new broker instance named msgqbroker.

```
# ./imqbrokerd -name msgqbroker -port 7777 &
```

8. Run netstat to verify that the newMessageQueue broker instance is up and running.

```
# netstat -an | grep 7777
```

```
*.7777 *.* 0 0 49152 0 LISTEN
```

9. Add a new user named msgquser.

This user will connect to the Message Queue broker instance on servers where Message Queue is installed. This user will be used only for session failover purposes, and does not assume the privileges of the guest user. It is a good practice to create a custom user for such purposes, and not to rely on the known user accounts or default user accounts to help prevent brute force or DOS attacks.

```
# ./imqusermgr add -u msgquser -g admin -p m5gqu5er -i msgqbroker
```

```
User repository for broker instance: msgqbroker
```

```
User msgquser successfully added.
```

10. Disable the guest user.

This step ensures that the guest user will not be able to access the OpenSSO Enterprise server.

```
# ./imqusermgr update -u guest -a false -i msgqbroker
```

User repository for broker instance: msgqbroker

Are you sure you want to update user guest? (y/n) y

User guest successfully updated.

11. Modify the amsfo.conf file.

amsfo.conf has parameters that are consumed by the OpenSSO Enterprise session failover startup script, amsfo.

- Change to the lib directory.

```
# cd /export/SFO/ssoSessionTools/opensso/config/lib
```

- Set the following properties:

```
CLUSTER_LIST=cucmsso1.vrajoli.com:7777,cucmsso2.vrajoli.com:7777
```

```
BROKER_INSTANCE_NAME=msgqbroker
```

```
USER_NAME=msgquser
```

```
BROKER_PORT=7777
```

Note: The port used for BROKER_PORT should be the same as the one used in the value of the CLUSTER_LIST.

- Save the file and close it.

12. Generate an encrypted password in a .password file with the following subprocedure.

- Change to the bin directory.

```
# cd /export/SFO/ssoSessionTools/opensso/bin
```

- Run **amsfopassword**.

This command generates an encrypted password, creates a new file named .password, and stores the encrypted password in the new file.

Caution: amsfopassword creates the .password file in a default location based on where the scripts were installed. If a different location is used, the PASSWORDFILE property in amsfo.conf should be changed accordingly.

```
# ./amsfopassword -e m5gqu5er -f
```

```
/export/SFO/ssoSessionTools/opensso/.password
```

```
os.name=SunOS
```

```
SUCCESSFUL
```

- (Optional)View the encrypted password for verification.

```
# more /export/SFO/ssoSessionTools/opensso/.password
```

```
M270Gb6U4ufRu+oWAZBdWw==
```

13. (Optional) Modify the amsessiondb script if necessary.

The amsessiondb script (located in the directory) starts the BerkeleyDatabase client, creates the database, and sets specific database values. It is called when the amsfo script is run for the first time. The amsessiondb script contains variables that specify default paths and directories. If any of the following components are not installed in their default directories, edit the amsessiondb script to set the variables to the correct locations.

```
IMQ_JAR_PATH=/export/SFO/ssoSessionTools/jmq/imq/lib
```

```
JMS_JAR_PATH=/export/SFO/ssoSessionTools/jmq/imq/lib
```

```
AM_HOME=/export/SFO/ssoSessionTools
```

Tip: Back up amsessiondb before you modify it.

14. Restart the session failover components with the following subprocedure.

- a. Change to the bin directory.
cd /export/SFO/ssoSessionTools/jmq/imq/bin
- b. Stop the MessageQueue instance using the product's command line interface.
 See the Message Queue documentation for more information.
- c. Run the **netstat** command to verify that the cucmssso1.vrajoli.com broker instance is stopped.
netstat -an | grep 7777
 If netstat returns no result, the cucmssso1.vrajoli.com broker instance is stopped.

Tip: If the cucmssso1.vrajoli.com broker instance is not stopped, kill the process using the following procedure.

- a. Get the Java process IDs.
ps -ef | grep java
- b. Kill the Java process IDs that were returned.
kill -9
- c. Run **netstat** again.
- d. Restart the cucmssso1.vrajoli.com broker instance.
cd /export/SFO/ssoSessionTools/opensso/bin
./amfso start
- e. Run the netstat command to verify that the MessageQueue port is open and listening.
netstat -an | grep 7777
***.7777 *.* 0 0 49152 0 LISTEN**

15. Log out of the cucmssso1.vrajoli.com host machine.

To Install Session Failover Components on cucmssso2.vrajoli.com

1. As a root user, log in to the mq-2 host machine.
2. Create a directory into which the MessageQueue and BerkeleyDatabase bits can be downloaded and change into it.
mkdir /export/SFO
cd /export/SFO
3. Copy ssoSessionTools.zip to the cucmssso1.vrajoli.com host machine, ssoSessionTools.zip is included in the openam_release9_20100207.zip file under the tools directory.
 (openam_release9_20100207/opensso/tools/ssoSessionTools.zip)
4. Unzip ssoSessionTools.zip.
cd /export/SFO
unzip ssoSessionTools.zip -d ssoSessionTools
5. Modify the permissions on the setup script and run it to initialize the session failover tools.
cd /export/SFO/ssoSessionTools
chmod +x setup
./setup
6. When prompted, enter **opensso** as the *Directory to install the scripts (example: opensso)*.
Note: The directory location should be relative to the current directory.
 When the script is finished, the following messages are displayed:
The scripts are properly setup under directory

```
/export/SFO/ssoSessionTools/opensso  
JMQ is properly setup under directory  
/export/SFO/ssoSessionTools/jmq
```

7. Change to the bin directory.

```
# cd /export/SFO/ssoSessionTools/jmq/imq/bin
```

8. Run the **imqbrokerd** command to create a new broker instance named msgqbroker.

```
# ./imqbrokerd -name msgqbroker -port 7777 &
```

9. Run **netstat** to verify that the newMessageQueue broker instance is up and running.

```
# netstat -an | grep 7777  
*.7777 *.* 0 0 49152 0 LISTEN
```

10. Add a new user named msgquser.

This user will connect to the Message Queue broker instance on servers where Message Queue is installed. This user will be used only for session failover purposes, and does not assume the privileges of the guest user. It is a good practice to create a custom user for such purposes, and not to rely on the known user accounts or default user accounts to help prevent brute force or DOS attacks.

```
# ./imqusermgr add -u msgquser -g admin -p m5gqu5er -i msgqbroker  
User repository for broker instance: msgqbroker  
User msgquser successfully added.
```

11. Disable the guest user.

This step ensures that the guest user will not be able to access the OpenSSO Enterprise server.

```
# ./imqusermgr update -u guest -a false -i msgqbroker  
User repository for broker instance: msgqbroker  
Are you sure you want to update user guest? (y/n) y  
User guest successfully updated.
```

12. Modify the amsfo.conf file with the following subprocedure.

amsfo.conf has parameters that are consumed by the OpenSSO Enterprise session failover startup script, amsfo.

- a. Change to the lib directory.

```
# cd /export/SFO/ssoSessionTools/opensso/config/lib
```

Tip: Back up amsfo.conf before you modify it.

- b. Set the following properties:

```
CLUSTER_LIST=msg-1.example.com:7777,cucmssso2.vrajoli.com.example.com:7777
```

BROKER_INSTANCE_NAME=msgqbroker

USER_NAME=msgquser

BROKER_PORT=7777

Note: The port used for BROKER_PORT should be the same as the one used in the value of the CLUSTER_LIST.

c. Save the file and close it.

13. Generate an encrypted password in a .password file with the following sub procedure.

a. Change to the bin directory.

cd /export/SFO/ssoSessionTools/opensso/bin

b. Run amsfopassword.

This command generates an encrypted password, creates a new file named .password, and stores the encrypted password in the new file.

Caution: amsfopassword creates the .password file in a default location based on where the scripts were installed. If a different location is used, the PASSWORDFILE property in amsfo.conf should be changed accordingly.

./amsfopassword -e m5gqu5er -f /export/SFO/ssoSessionTools/opensso/.password

os.name=SunOS

SUCCESSFUL

c. (Optional) View the encrypted password for verification.

more /export/SFO/ssoSessionTools/opensso/.password

M270Gb6U4ufRu+oWAZBdWw==

14. (Optional) Modify the amsessiondb script if necessary.

The amsessiondb script (located in the /export/SFO/ssoSessionTools/opensso/bin directory) starts the BerkeleyDatabase client, creates the database, and sets specific database values. It is called when the amsfo script is run for the first time. The amsessiondb script contains variables that specify default paths and directories. If any of the following components are not installed in their default directories, edit the amsessiondb script to set the variables to the correct locations.

IMQ_JAR_PATH=/export/SFO/ssoSessionTools/jmq/imq/lib

JMS_JAR_PATH=/export/SFO/ssoSessionTools/jmq/imq/lib

AM_HOME=/export/SFO/ssoSessionTools

Tip: Back up amsessiondb before you modify it.

15. Restart the session failover components.

a. Change to the bin directory.

cd /export/SFO/ssoSessionTools/jmq/imq/bin

b. Stop the MessageQueue instance using the product's command line interface.

See the Message Queue documentation for more information.

c. Run the netstat command to verify that the cucmssso2.vrajoli.com broker instance is stopped.

```
# netstat -an | grep 7777
```

If netstat returns no result, the cucmssso2.vrajoli.com broker instance is stopped.

Tip: If the cucmssso2.vrajoli.com broker instance is not stopped, kill the process using the following procedure.

a. Get the Java process IDs.

```
# ps -ef | grep java
```

b. Kill the Java process IDs that were returned.

```
# kill -9 #####
```

c. Run **netstat** again.

d. Restart the cucmssso2.vrajoli.com broker instance.

```
# cd /export/SFO/ssoSessionTools/opensso/bin
```

```
# ./amfso start
```

e. Run the netstat command to verify that the MessageQueue port is open and listening.

```
# netstat -an | grep 7777
```

```
*.7777 *.* 0 0 49152 0 LISTEN
```

16. Log out of the cucmssso2.vrajoli.com host machine.

16.2.2.2 Configuration of Session Failover Components of Windows Platform

To Install Session Failover Components on cucmssso1.vrajoli.com OpenAM Enterprise Server on Windows

1. Log in to the server where you want to install and configure the session failover components (cucmssso1.vrajoli.com).
2. Copy ssoSessionTools.zip to the cucmssso1.vrajoli.com host machine; ssoSessionTools.zip is included in the openam_release9_20100207.zip file under the tools directory (openam_release9_20100207/opensso/tools/ssoSessionTools.zip).
3. Unzip ssoSessionTools.zip to ssoSessionTools folder and cd to ssoSessionTools folder.

```
C:\>cd ssoSessionTools
```

4. Run the setup.bat script to install the session tools on Windows systems.

```
C:\ssoSessionTools>setup.bat
```

Name of the directory to install the scripts (example: sfoScripts): sfoScripts

The scripts are properly set up under directory: C:\ssoSessionTools\sfoScripts.

JMQ is properly set up under directory C:\ssoSessionTools\jmq

5. Change to bin folder.
6. Run the **imqbrokerd** command to create a new broker instance named msgqbroker.

```
C:\ssoSessionTools\jmq\imq\bin>imqbrokerd.exe -name msgqbroker -port 7777 &
```

[25/Sep/2010:23:38:40 IST]

=====
=====

Sun GlassFish(tm) Message Queue 4.4

Sun Microsystems, Inc.

Version: 4.4 (Build 16-a)

Compile: Thu 08/27/2009

Copyright (c) 2009 Sun Microsystems, Inc. All rights reserved. Use is subject to license terms.

=====
=====

Java Runtime: 1.6.0_21 Sun Microsystems Inc. c:\Program Files\Java\jdk1.6.0_21\jre

[25/Sep/2010:23:38:40 IST] IMQ_HOME=C:\ssoSessionTools\jmq\imq

[25/Sep/2010:23:38:40 IST] IMQ_VARHOME=C:\ssoSessionTools\jmq\imq\var

**[25/Sep/2010:23:38:40 IST] Windows Server 2008 6.0 x86 cucmss01.vrajoli.com (1 cp
u) Administrator**

[25/Sep/2010:23:38:40 IST] Java Heap Size: max=190080k, current=15872k

[25/Sep/2010:23:38:40 IST] Arguments: -name msgqbroker -port 7777

[25/Sep/2010:23:38:40 IST] [B1060]: Loading persistent data...

**[25/Sep/2010:23:38:40 IST] Using built-in file-based persistent store: C:\ssoSes
sionTools\jmq\imq\var\instances\msgqbroker**

**[25/Sep/2010:23:38:40 IST] [B1039]: Broker
"msgqbroker@cucmss01.vrajoli.com:7777" ready.**

7. Run **netstat** to verify that the newMessageQueue broker instance is up and running.

C:\ssoSessionTools\jmq\imq\bin>netstat -an | findstr 7777

TCP 0.0.0.0:7777 0.0.0.0 LISTENING

TCP 10.78.85.131:60787 10.78.85.131:7777 TIME_WAIT

TCP [::]:7777 [::]:0 LISTENING

8. Add a new user named msgquser.

This user will connect to theMessage Queue broker instance on servers where Message Queue is installed. This user will be used only for session failover purposes, and does not assume the privileges of the guest user. It is a good practice to create a custom user for such purposes, and not to rely on the known user accounts or default user accounts to help prevent brute force or DOS attacks.

**C:\ssoSessionTools\jmq\imq\bin>imqusermgr.exe add -u msgquser -g admin -p
m5gqu5er -i msgqbroker**

User repository for broker instance: msgqbroker

User msgquser successfully added.

9. Disable the guest user. This step ensures that the guest user will not be able to access the OpenSSO Enterprise server.

```
C:\ssoSessionTools\jmq\imq\bin>imqusermgr.exe update -u guest -a false -i msgqbroker
```

User repository for broker instance: msgqbroker

Are you sure you want to update user guest? (y/n)[n] y

User guest successfully updated.

10. Modify the amsfo.conf file.

amsfo.conf has parameters that are consumed by the OpenSSO Enterprise session failover startup script, amsfo.

- Change to the lib directory.

```
# cd C:\ssoSessionTools\sfoconfigs\config\lib
```

- Set the following properties:

```
CLUSTER_LIST=cucmsso1.vrajoli.com.example.com:7777,cucmsso2.vrajoli.com.example.com:7777
```

```
BROKER_INSTANCE_NAME=msgqbroker
```

```
USER_NAME=msgquser
```

```
BROKER_PORT=7777
```

Note: The port used for BROKER_PORT should be the same as the one used in the value of the CLUSTER_LIST.

- Save the file and close it.

11. Generate an encrypted password in a .password file with the following subprocedure.

Change to the bin directory.

```
# cd C:\ssoSessionTools\sfoconfigs\bin
```

- Run amsfopassword.bat

This command generates an encrypted password, creates a new file named .password, and stores the encrypted password in the new file.

Caution: amsfopassword creates the .password file in a default location based on where the scripts were installed. If a different location is used, change the PASSWORDFILE property in amsfo.conf accordingly.

```
C:\SSOSES~1\SFOSCR~1\bin>amsfopassword.bat -e m5gqu5er -f
c:\ssoSessionTools\sfoscripts\password
```

os.name=Windows Server 2008

SUCCESSFUL

12. Start the cucmssso1.vrajoli.com broker instance. To start the amsfo.pl you need to have Perl installed on OpenAM host. In this guide ActivePerl 5.12.2 Build 1202 has been installed on Windows machine where OpenAM is installed.

```
C:\ssoSessionTools\sfoscripts\bin>amsfo.pl
c:\ssoSessionTools\sfoscripts\config\lib\amsfo.conf start
```

starting JMQ Broker

```
C:/ssoSessionTools/jmq/imq/bin/imqbrokerd.exe -bgnd -silent -vmargs "-Xms256m -Xmx512m" -name msgqbroker -port 7777 -cluster
cucmssso1.vrajoli.com:7777,cucmssso2.vrajoli.com:7777
```

starting amsessiondb client

```
c:/Program Files/Java/jdk1.6.0_21/jre/bin/java.exe -classpath
"C:/ssoSessionTools/jmq/imq/lib/imq.jar;C:/ssoSessionTools/jmq/imq/lib/jms.jar;C:/ss
oSession
```

```
Tools/ext/je.jar;C:/ssoSessionTools/locale;C:/ssoSessionTools/lib/am_sessiondb.jar;"
com.sun.identity.ha.jmqdb.client.FAMHaDB -a cucmssso1.vrajoli.com:7777,
```

```
cucmssso2.vrajoli.com:7777 -u msgquser -f C:/ssoSessionTools/sfoscripts/password -b
/tmp/amsession/amsessiondb -m c:\ssoSessionTools\sfoscripts\config\l
```

ib\amsfo.conf

Initializing and connecting to the Message Queue server ...

Successfully started.

13. Run the **netstat** command to verify that the MessageQueue port is open and listening.

```
C:\ssoSessionTools\sfoscripts\bin>netstat -an | findstr 7777
```

```
TCP 0.0.0.0:7777 0.0.0.0 LISTENING
```

```
TCP 10.78.85.131:60787 10.78.85.131:7777 TIME_WAIT
```

```
TCP [::]:7777 [::]:0 LISTENING
```

To Install Session Failover Components on cucmssso2.vrajoli.com OpenAM Enterprise Server on Windows

1. Log in to the server where you want to install and configure the session failover components (cucmssso1.vrajoli.com)
2. Copy ssoSessionTools.zip to the cucmssso1.vrajoli.com host machine, ssoSessionTools.zip is included in the openam_release9_20100207.zip file under the tools directory. (openam_release9_20100207/opensso/tools/ssoSessionTools.zip)
3. Unzip ssoSessionTools.zip to ssoSessionTools folder and cd to ssoSessionTools folder

```
C:\>cd ssoSessionTools
```

4. Run the setup.bat script to install the session tools on Windows systems.

C:\ssoSessionTools>setup.bat

Name of the directory to install the scripts (example: sfoascripts):sfoascripts

The scripts are properly set up under directory: C:\ssoSessionTools\sfoascripts

JMQ is properly set up under directory C:\ssoSessionTools\jmq

5. Change to bin folder.
6. Run the **imqbrokerd** command to create a new broker instance named msgqbroker.

C:\ssoSessionTools\jmq\imq\bin>imqbrokerd.exe -name msgqbroker -port 7777 &

[25/Sep/2010:23:38:40 IST]

=====
=====

Sun GlassFish(tm) Message Queue 4.4

Sun Microsystems, Inc.

Version: 4.4 (Build 16-a)

Compile: Thu 08/27/2009

Copyright (c) 2009 Sun Microsystems, Inc. All rights reserved. Use is
subject to license terms.

=====
=====

Java Runtime: 1.6.0_21 Sun Microsystems Inc. c:\Program Files\Java\jdk1.6.0_21\jre

[25/Sep/2010:23:38:40 IST] IMQ_HOME=C:\ssoSessionTools\jmq\imq

[25/Sep/2010:23:38:40 IST] IMQ_VARHOME=C:\ssoSessionTools\jmq\imq\var

[25/Sep/2010:23:38:40 IST] Windows Server 2008 6.0 x86 cucmssso1.vrajoli.com (1 cp
u) Administrator

[25/Sep/2010:23:38:40 IST] Java Heap Size: max=190080k, current=15872k

[25/Sep/2010:23:38:40 IST] Arguments: -name msgqbroker -port 7777

[25/Sep/2010:23:38:40 IST] [B1060]: Loading persistent data...

[25/Sep/2010:23:38:40 IST] Using built-in file-based persistent store: C:\ssoSes
sionTools\jmq\imq\var\instances\msgqbroker\

[25/Sep/2010:23:38:40 IST] [B1039]: Broker "msgqbroker@cucmssso1.vrajoli.com:7777"
ready.

7. Run netstat to verify that the newMessageQueue broker instance is up and running.

C:\ssoSessionTools\jmq\imq\bin>netstat -an | findstr 7777

TCP 0.0.0.0:7777 0.0.0.0 LISTENING

TCP 10.78.85.131:60787 10.78.85.131:7777 TIME_WAIT

TCP [::]:7777 [::]:0 LISTENING

8. Add a new user named msgquser.

This user will connect to the Message Queue broker instance on servers where Message Queue is installed. This user will be used only for session failover purposes, and does not assume the privileges of the guest user. It is a good practice to create a custom user for such purposes, and not to rely on the known user accounts or default user accounts to help prevent brute force or DOS attacks.

```
C:\ssoSessionTools\jmq\imq\bin>imqusermgr.exe add -u msgquser -g admin -p m5gqu5er -i msgqbroker
```

User repository for broker instance: msgqbroker

User msgquser successfully added.

9. **Disable the guest user. This step ensures** that the guest user will not be able to access the OpenSSO Enterprise server.

```
C:\ssoSessionTools\jmq\imq\bin>imqusermgr.exe update -u guest -a false -i msgqbroker
```

User repository for broker instance: msgqbroker

Are you sure you want to update user guest? (y/n)[n] y

User guest successfully updated.

10. Modify the amsfo.conf file.

amsfo.conf has parameters that are consumed by the OpenSSO Enterprise session failover startup script, amsfo.

- Change to the lib directory.

```
# cd C:\ssoSessionTools\sfoascripts\config\lib
```

- Set the following properties:

```
CLUSTER_LIST=cucmssso1.vrajoli.com.example.com:7777,cucmssso2.vrajoli.com.example.com:7777
```

```
BROKER_INSTANCE_NAME=msgqbroker
```

```
USER_NAME=msgquser
```

```
BROKER_PORT=7777
```

Note: The port used for BROKER_PORT should be the same as the one used in the value of the CLUSTER_LIST.

- Save the file and close it.

11. Generate an encrypted password in a .password file with the following subprocedure.

Change to the bin directory.

```
# cd C:\ssoSessionTools\sfoascripts\bin
```

- Run **amsfopassword.bat**.

This command generates an encrypted password, creates a new file named .password, and stores the encrypted password in the new file.

Caution: amsfopassword creates the .password file in a default location based on where the scripts were installed. If a different location is used, the PASSWORDFILE property in amsfo.conf should be changed accordingly.

```
C:\SSOSES~1\SFOSCR~1\bin>amsfopassword.bat -e m5gqu5er -f
c:\ssoSessionTools\sfoscripts\.password
```

os.name=Windows Server 2008

SUCCESSFUL

12. Start the cucmssso2.vrajoli.com broker instance. To start the amsfo.pl you need to have Perl installed on OpenAM host. In this guide ActivePerl 5.12.2 Build 1202 has been installed on Windows machine where OpenAM is installed.

```
C:\ssoSessionTools\sfoscripts\bin>amsfo.pl
c:\ssoSessionTools\sfoscripts\config\lib\amsfo.conf start
```

starting JMQ Broker

```
C:/ssoSessionTools/jmq/imq/bin/imqbrokerd.exe -bgnd -silent -vmargs "-Xms256m -Xmx512m" -name msgqbroker -port 7777 -cluster
cucmssso1.vrajoli.com:7777,cucmssso2.vrajoli.com:7777
```

starting amsessiondb client

```
c:/Program Files/Java/jdk1.6.0_21/jre/bin/java.exe -classpath
"C:/ssoSessionTools/jmq/imq/lib/imq.jar;C:/ssoSessionTools/jmq/imq/lib/jms.jar;C:/ss
oSession
```

```
Tools/ext/je.jar;C:/ssoSessionTools/locale;C:/ssoSessionTools/lib/am_sessiondb.jar;."
com.sun.identity.ha.jmqdb.client.FAMHaDB -a cucmssso1.vrajoli.com:7777,
```

```
cucmssso2.vrajoli.com:7777 -u msgquser -f C:/ssoSessionTools/sfoscripts/.password -b
/tmp/amsession/amsessiondb -m c:\ssoSessionTools\sfoscripts\config\l
```

ib\amsfo.conf

Initializing and connecting to the Message Queue server ...

Successfully started.

13. Run the **netstat** command to verify that the MessageQueue port is open and listening.

```
C:\ssoSessionTools\sfoscripts\bin>netstat -an |findstr 7777
```

```
TCP 0.0.0.0:7777 0.0.0.0 LISTENING
```

```
TCP 10.78.85.131:60787 10.78.85.131:7777 TIME_WAIT
```

```
TCP [::]:7777 [::]:0 LISTENING
```

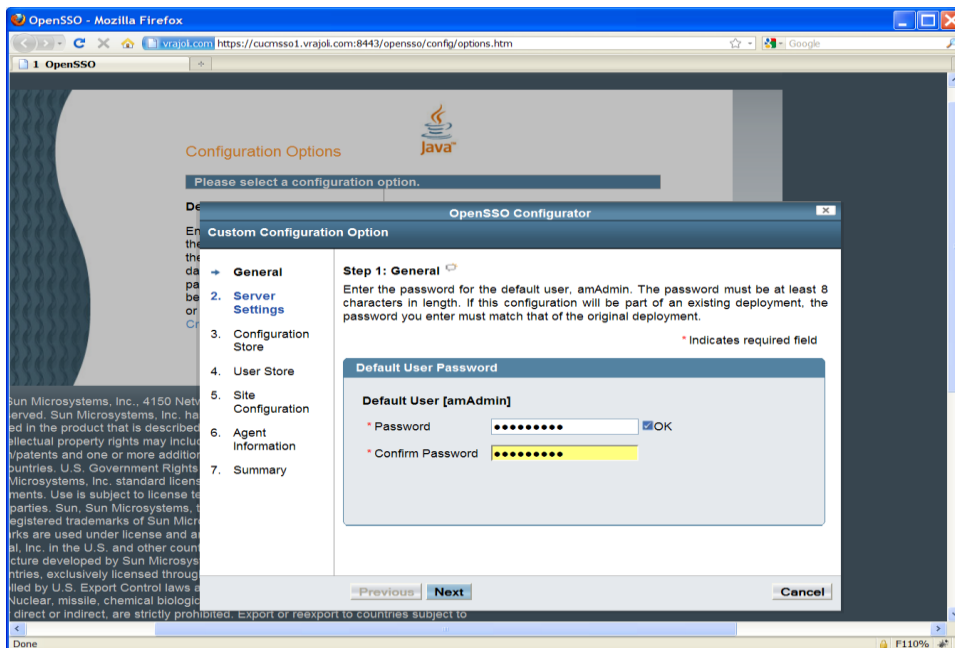
16.2.3 Installation and configuration of OpenAM Enterprise Servers for Session Failover

16.2.3.1 Installation of OpenAM Enterprise Server 1

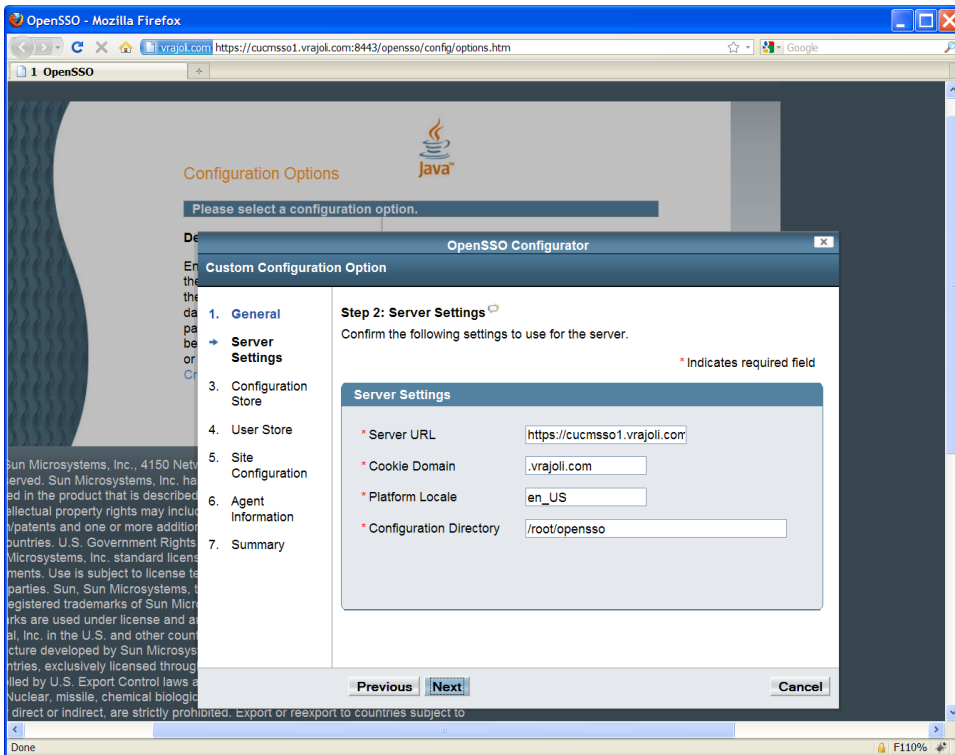
Browse the OpenAM URL: <https://cucmss01.vrajoli.com:8443/opensso>, you will see the following Configurator. Click on **Create New Configuration** under Custom Configuration.



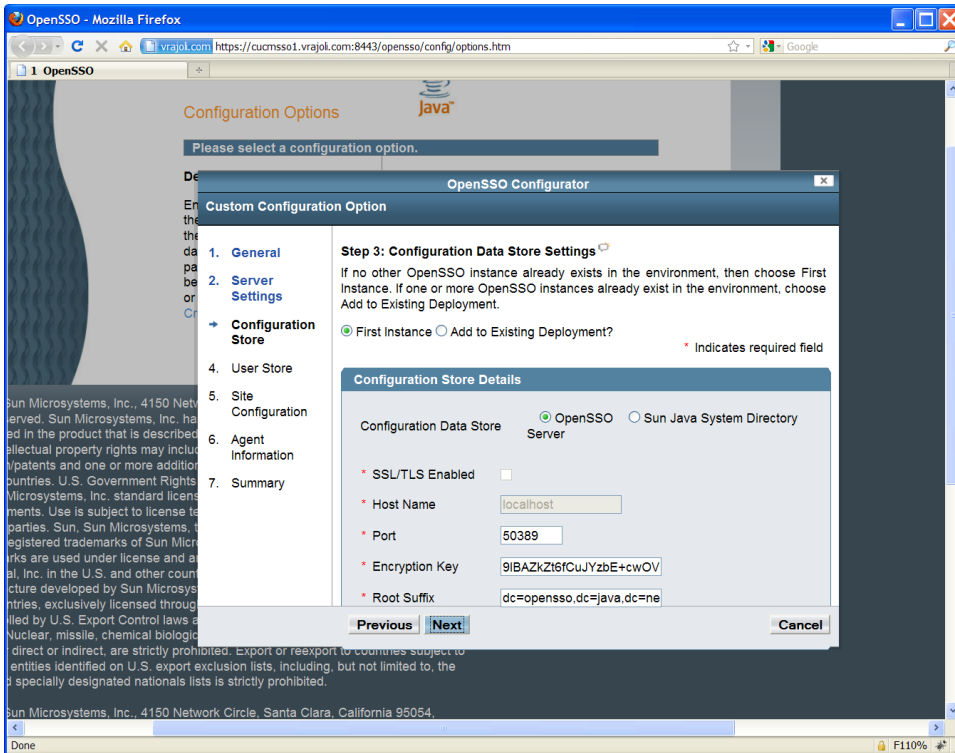
Create a new password for default user [amAdmin]. Click **Next**.



Click **Next**.



Click Next.

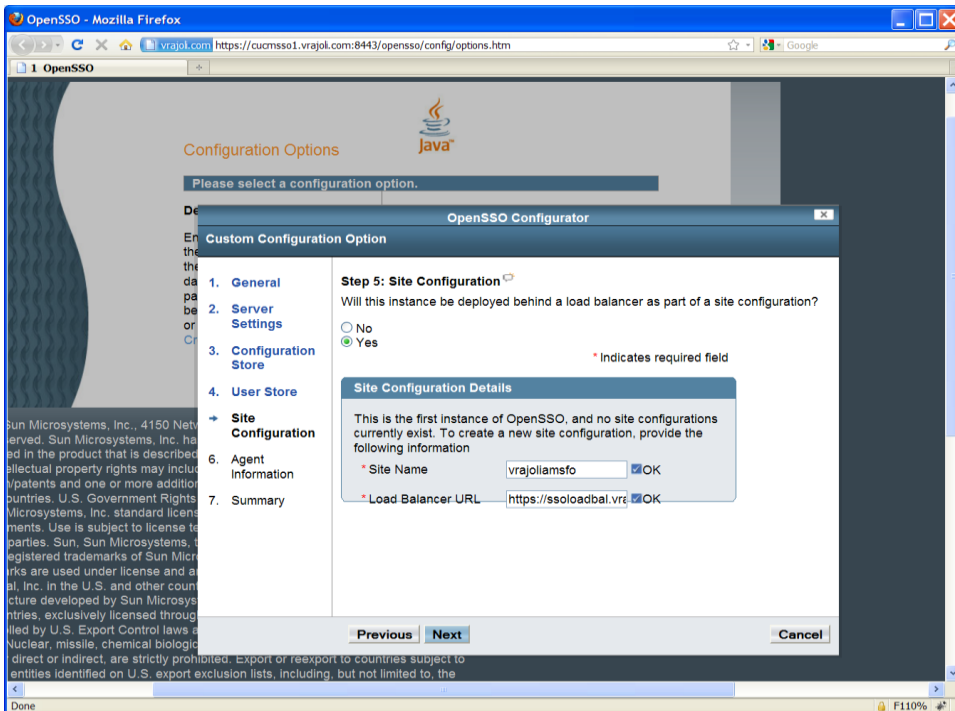


Select the **OpenSSO User Data Store** radio button and click **Next**.

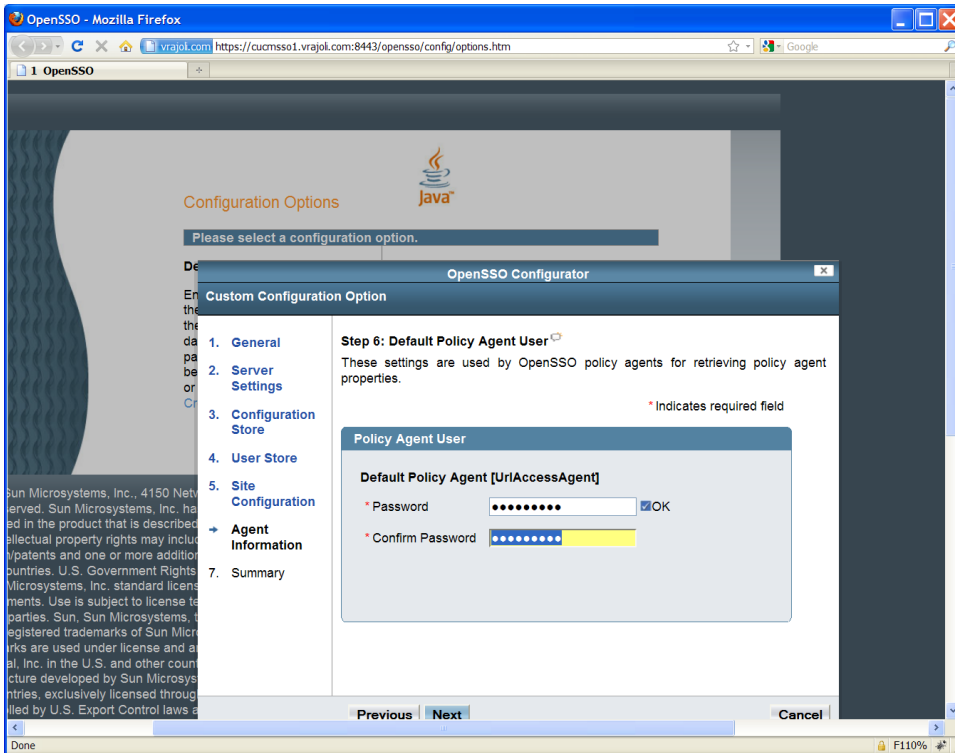


Under Site Configuration, click the **Yes** radio button and enter a site name and provide the Load Balancer URL (which was set up in section 3.1). Click **Next**.

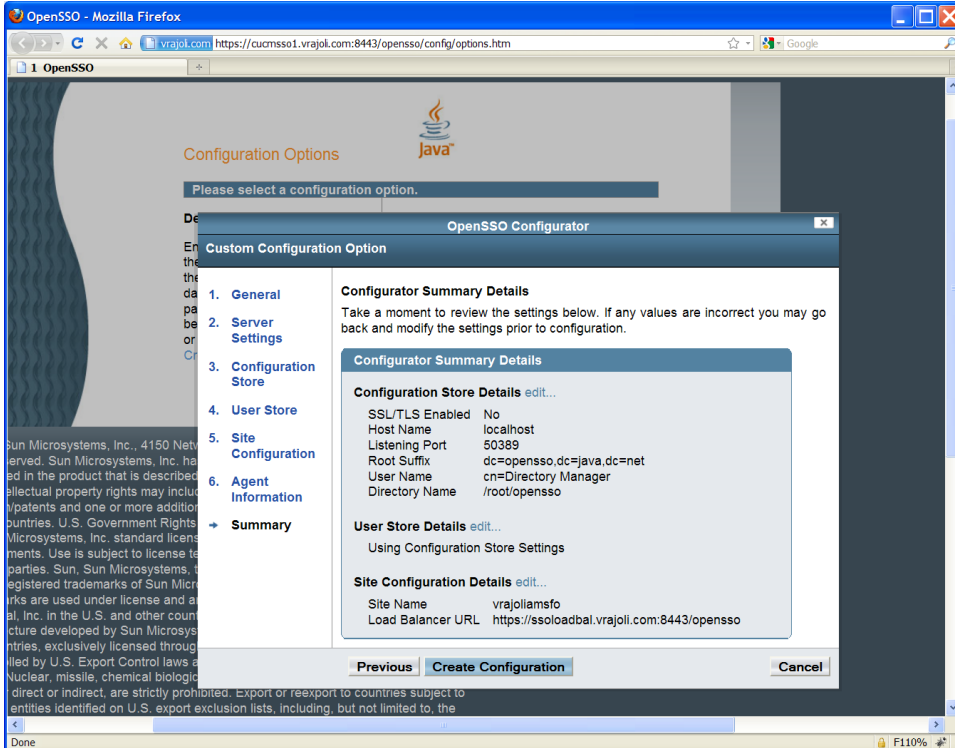
Example for Load Balancer URL: <https://ssoloadbal.vrajoli.com:8443/opensso>



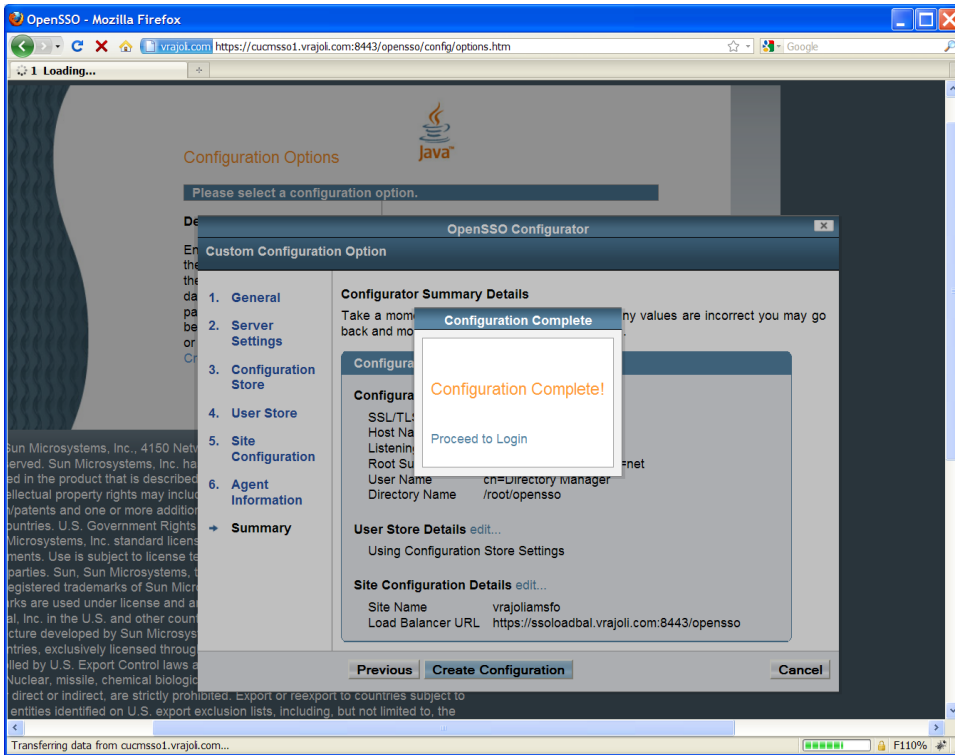
Create a password for Default Policy Agent and click **Next**.



Click the **Create Configuration** button.



Click on **Proceed to Login** link.



The OpenAM login window appears.

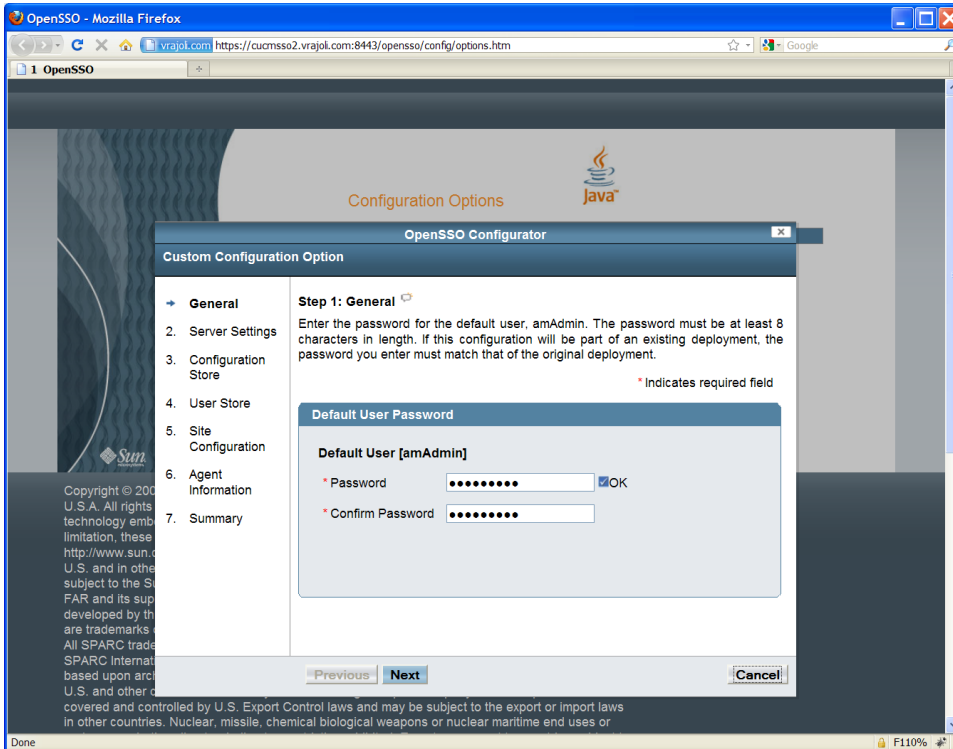


16.2.3.2 Installation of OpenAM Enterprise Server 2

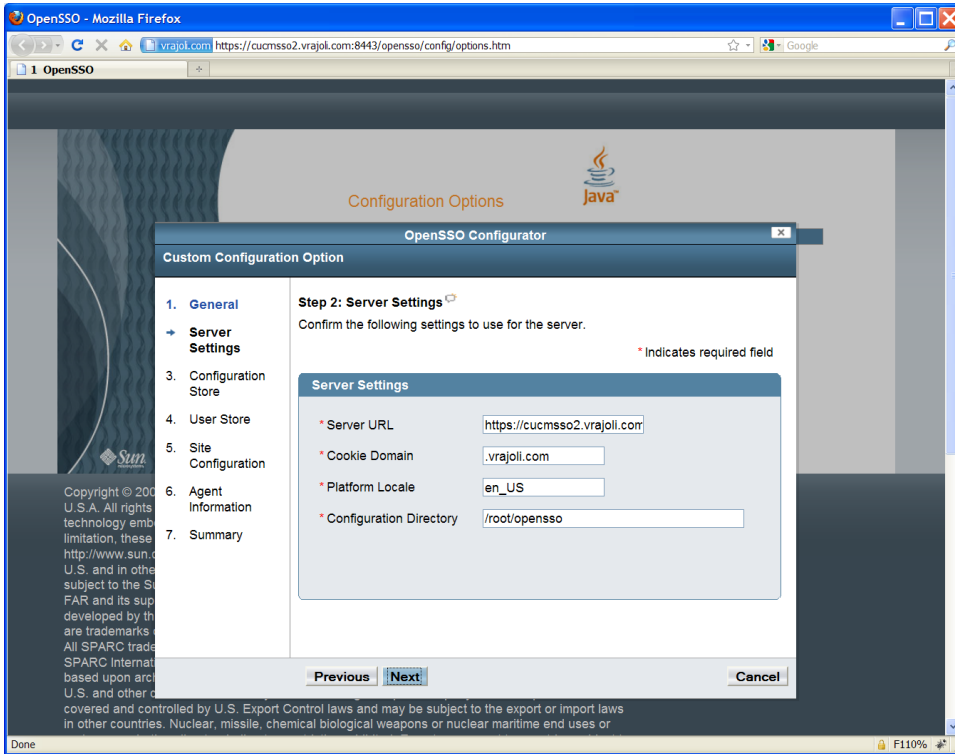
Browse the OpenAM URL: <https://cucmssso2.vrajoli.com:8443/opensso>, you will see the following Configurator. Click on **Create New Configuration** under Custom Configuration.



Create a password for default user [amAdmin] and click **Next**.

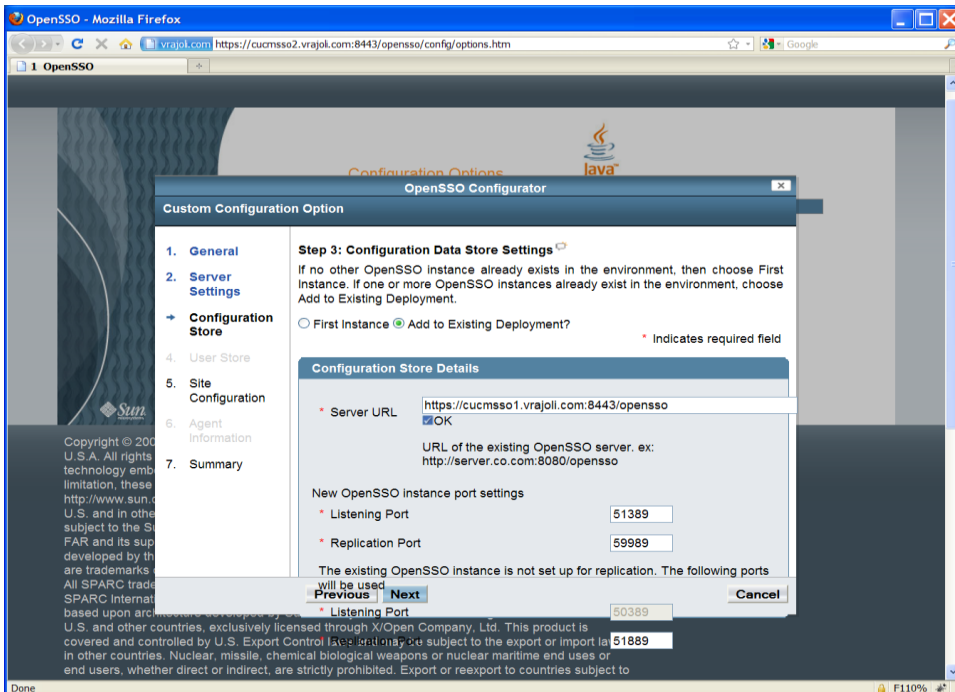


Click Next.



Click **Add to Existing Deployment** radio button and enter the OpenAM Enterprise Server 1 URL in the Server URL text box. Click **Next**.

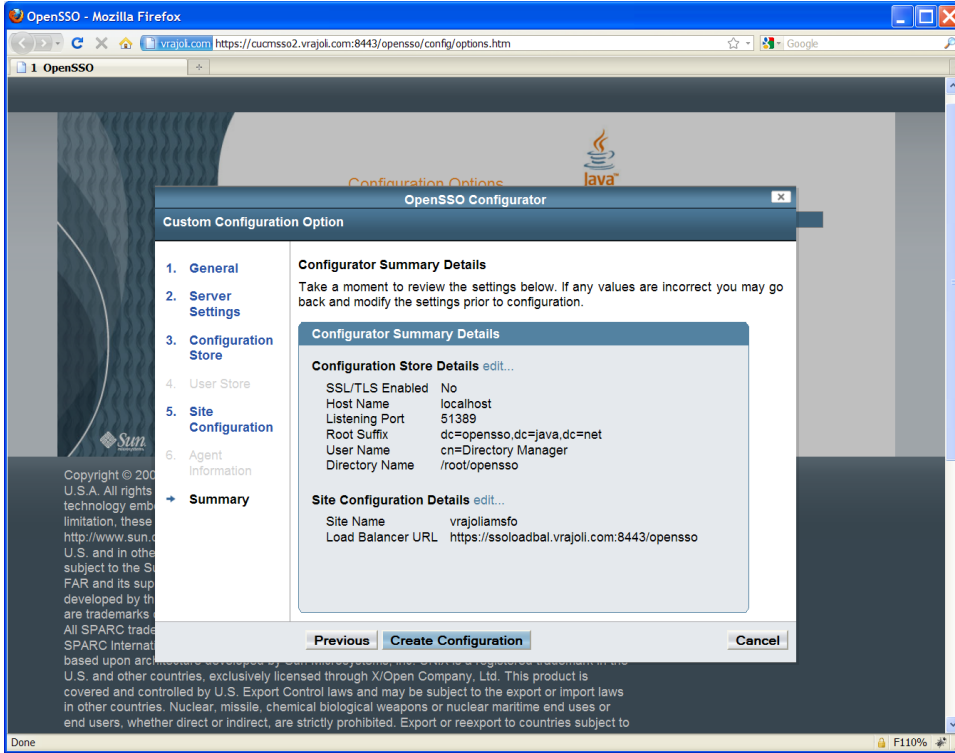
Note: Before you complete this step you must import the OpenAM Enterprise Server 1 certificate to the OpenAM Enterprise Server 2 trust store.



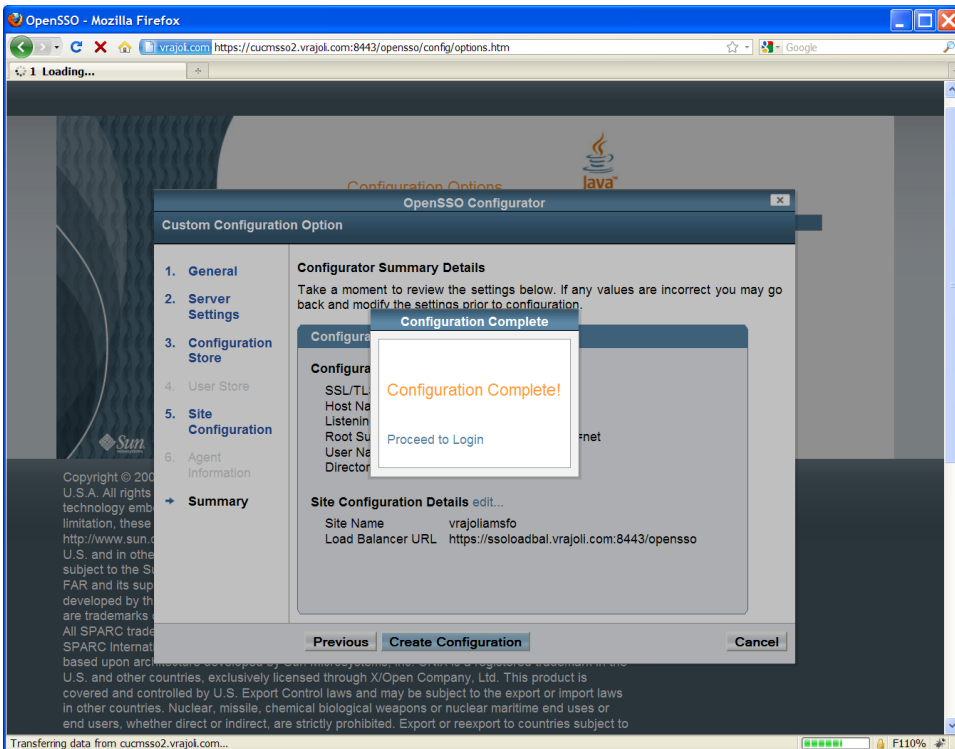
Under Site Configuration, click the **Yes** radio button and enter the same site name that was used in OpenAM Enterprise Server 1. Click **Next**.

Enter the Load Balancer URL that was set up in section 3.1, for example:
`https://ssoloadbal.vrajoli.com:8443/opensso`.

Click the **Create Configuration** button.



Click the **Proceed to Login** link.



16.2.3.3 Configure OpenSSO Enterprise for Session Failover

Access <https://cucmssso1.vrajoli.com:8443/opensso> from a web browser.

Log in to OpenAM Enterprise console.

Click the **Configuration** tab.

Under Global properties, click **Session**.

Under Secondary Configuration Instance, click **New**.

In the Add Sub Configuration window, provide the following information.

Name: Select **External**

Session Store User: Enter **msgquser**

Session Store Password: Enter **m5gqu5er**

Session Store Password (confirm): Enter **m5gqu5er**

Maximum Wait Time: Keep the default value of 5000.

Database URL: Enter **cucmssso1.vrajoli.com:7777,cucmssso2.vrajoli.com:7777**.

This is the Message Queue broker address list. Enter multiple values using a comma and no space.

Click **Add**.

Click **Save**.

Log out of the OpenSSO Enterprise console.

16.3 Configuring SSO on Cisco Unified Communications Manager with AMSFO Setup

Access Load Balancer URL (<https://ssoloadbal.vrajoli.com:8989>) from the web browser. Click the **Lock** icon on the right bottom corner, and click the **View Certificate** button. Go to the **Details** tab and export the certificate to your local machine from where you are browsing.

Because a replication setup is running on all the OpenSSO Enterprise servers in AMSFO environment, Policies, Authentication Module instances, and J2EE Agents that are created on one OpenSSO Enterprise server get replicated on the rest of the OpenSSO Enterprise servers.

For the Authentication Module instance, you must create a keytab for the Load Balancer host and not for the OpenSSO Enterprise server hosts. Place the Load Balancer keytab file on both the OpenSSO Enterprise host file systems.

Create OpenAM Policies, Authentication Module instance, and J2EE agent profile on any one of the OpenSSO Enterprise servers. For creating Policies, Authentication Module instances, and J2EE agent profile, refer to sections 10.1, 10.2 and 10.3.

Log in to Unified CM OS Administration; go to **Security > Certificate Management > Upload Certificate**.

Select certificate type as tomcat-trust, and browse for the above Load Balancer certificate that you saved in the previous step. When you find the certificate, click the **Upload** button.

Now log in to the CLI of Unified CM 8.5, and execute the command **utils sso enable** with OpenSSO URL as your Load Balancer host (<https://ssoloadbal.vrajoli.com:8443/opensso>).