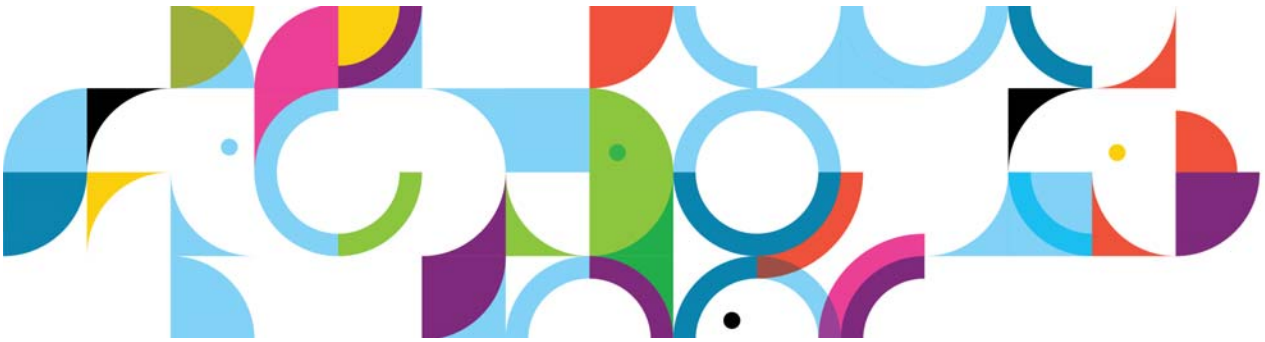
IBM Connections 4 Public Deployment Scenarios

Deployment Scenarios

ERC 1.0



Trademarks

IBM® and the IBM logo are registered trademarks of International Business Machines Corporation.

The following are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide:

AIX®	Cognos®	DB™
DB2 Universal Database™	DB2®	Domino®
Lotus®	LotusScript®	Notes®
Power®	Quickr®	Rational®
Sametime®	System z®	Tivoli®
WebSphere®	400®	

Intel and Xeon are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java™ and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Other product and service names might be trademarks of IBM or other companies.

January 2013 edition

IBM Connections 4: PDS SiteMinder and SPNEGO

About the author

Sophy Carroll is a lead member of the Lotus Connections System Verification Test (SVT) specializing in the area of product deployment and integration verification test. Sophy can be reached at sophy_carroll@ie.ibm.com.

Overview

This scenario explains how to deploy IBM® Connections 4.0 in a network deployment that involves multiple computers with one IBM WebSphere® cell that contains two nodes, both of which host IBM® Connections 4.0. This scenario is typical of an enterprise-level production deployment with SiteMinder and SPNEGO enabled.

Systems and naming conventions used throughout this document

Computer host name	Applications	Version#	OS/version	RAM/ CPU	VM or HW
connections.example.com	WebSphere Application Server Deployment Manager IBM HTTP Server	WebSphere Application Server v7.0.0.21 (64 bit) IBM HTTP Server v7.0.0.21 (32 bit)	SUSE 10 SP 4 (64 bit)	4 G /2 CPUs	VM
node1.example.com	Node1 (WebSphere Application Server)	WebSphere Application Server v7.0.0.21		8 G /2 CPUs	
node2.example.com	Node2 (WebSphere Application Server)	WebSphere Application Server v7.0.0.21		8 G /2 CPUs	
db2server.example.com	DB2 Tivoli Directory Integrator	DB2 v9.7+FP6 Tivoli Directory Integrator v9.1+FP5		4 G /2 CPUs	
msad2008.example.com	MS Active Directory 2008	2008	Win2008 R2 EE Server		
domino.example.com	Domino Mail-in server	Domino 8.5.3	Win2008 R2 EE Server	4 G /2 CPUs	VM

PreConnections installation work
Assuming: WebSphere Application Server and nodes are set up

PreConnections installation work

It is assumed that WebSphere Application Server and nodes are set up.

Deployment-specific information

1. Check the security on the admin console: that in Global security enable admin security and enable app security are checked. In the writer's VM image, enable app security is not checked.
2. Set the max JVM value for the DMGR or you get OOMs when configuring the remote http server.
   - In the Admin console, go to system admin `\deployment manager\Java process management\process definition\JVM`

Figure 1. Deployment manager > Process definition > Java Virtual Machine

   - Set the maximum heap size to 1024.

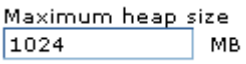


Figure 2. Maximum heap size

Contents

1. Cognos configuration

Requirements before you start the setup of Cognos

__ 1. Ensure that the Deployment Manager is running and that the time difference between the node that hosts Cognos BI Server and the Deployment Manager does not exceed 5 minutes so that the addNode action succeeds.

__ 2. The node that hosts Cognos BI Server must not already be federated to the Deployment Manager. It is done later.

__ 3. For Cognos and Connections to work, you must use an LDAP user as the admin on Cognos. A local WebSphere user, wasadmin, does not work. So, plan to use an LDAP user, for example `wpsbind`.

__ 4. Download both the Cognos BI Server and Cognos Transformer to your test systems where you install them. You might install them to the same system as the deployment manager. Here are the names and part numbers to download from Xpertise Library.

OS	BI Server	Transformer	Full eAssembly (includes BI Server and Transformer)
AIX	IBM Cognos Business Intelligence Server 64-bit 10.1.1 AIX Multilingual (CI5VTML)	IBM Cognos Business Intelligence Transformer 10.1.1 AIX Multilingual (CI2Q4ML)	IBM Cognos Business Intelligence 10.1.1 AIX Multilingual eAssembly (CRFY4ML)
Linux	IBM Cognos Business Intelligence Server 64-bit 10.1.1 x86 Multilingual (CI5W7ML)	IBM Cognos Business Intelligence Transformer 10.1.1 Linux x86 Multilingual (CI2Q6ML)	IBM Cognos Business Intelligence 10.1.1 Linux x86 Multilingual eAssembly (CRFY8ML)
Linux (System z)	IBM Cognos Business Intelligence Server 64-bit 10.1.1 Linux on System z Multilingual (CI5W5ML)	IBM Cognos Business Intelligence Transformer 10.1.1 Linux on System z Multilingual (CI2QHML)	IBM Cognos Business Intelligence 10.1.1 Linux on System z Multilingual eAssembly (CRFZ6ML)
Windows	IBM Cognos Business Intelligence Server 64-bit 10.1.1 Windows Multilingual (CI5VVML)	IBM Cognos Business Intelligence Transformer 10.1.1 Windows Multilingual (CI2Q1ML)	IBM Cognos Business Intelligence 10.1.1 Windows Multilingual eAssembly (CRFY3ML)

Create Connections databases on DB2 server

Linux

Before you can use the wizard to create databases for your IBM® Connections deployment, prepare the database server. Follow these steps.

__ 1. Log in to your database server as the root user:

__ a. export DISPLAY=<hostname:displaynumber.screennumber>.

__ b. echo $DISPLAY // Echo the value of DISPLAY under the root user. Ensure that the current user is qualified or else switch to a qualified user by running the following commands.

__ 2. Grant display authority to all users by running the following commands under the root user or system administrator:

__ a. xhost + // Grant display authority to other users

__ 3. su: db2user:

__ a. export DISPLAY=<hostname:displaynumber.screennumber> where <hostname:displaynumber.screennumber> represents the client system, monitor number, and window number.

__ b. xclock // Display the clock, confirming that the current user has display authority and can run the wizard

The creation of the Connections databases is done by a wizard.

__ 1. Copy the `Lotus_Connections_4.0_wizards_lin_aix.tar` to your computer and extract it. Do it as the db2user user on the OS and **not** root. Then, go into the Wizard folder and run `./dbWizard.sh`. The following screen is shown. Select **Next** to continue.

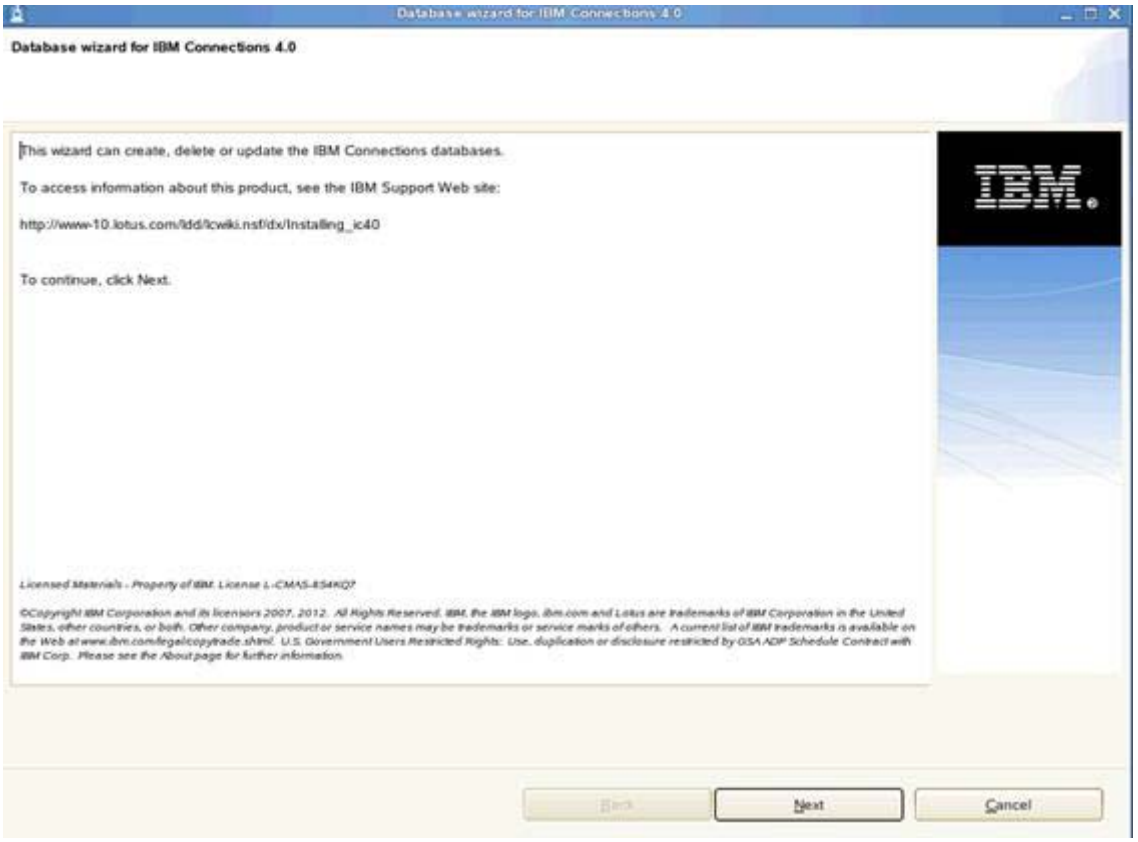


Figure 3. Database wizard for IBM Connections 4.0

__ 2. You are then asked what you want to do: Create, delete, or upgrade. Click **Create** and **Next** to continue.

Figure 4. Database task selection

__ 3. Select the path of your database installation location and the database instance name. Click **Next** to continue.

Figure 5. Database selection

__ 4. Ensure that all databases are selected and then click **Next** to continue.

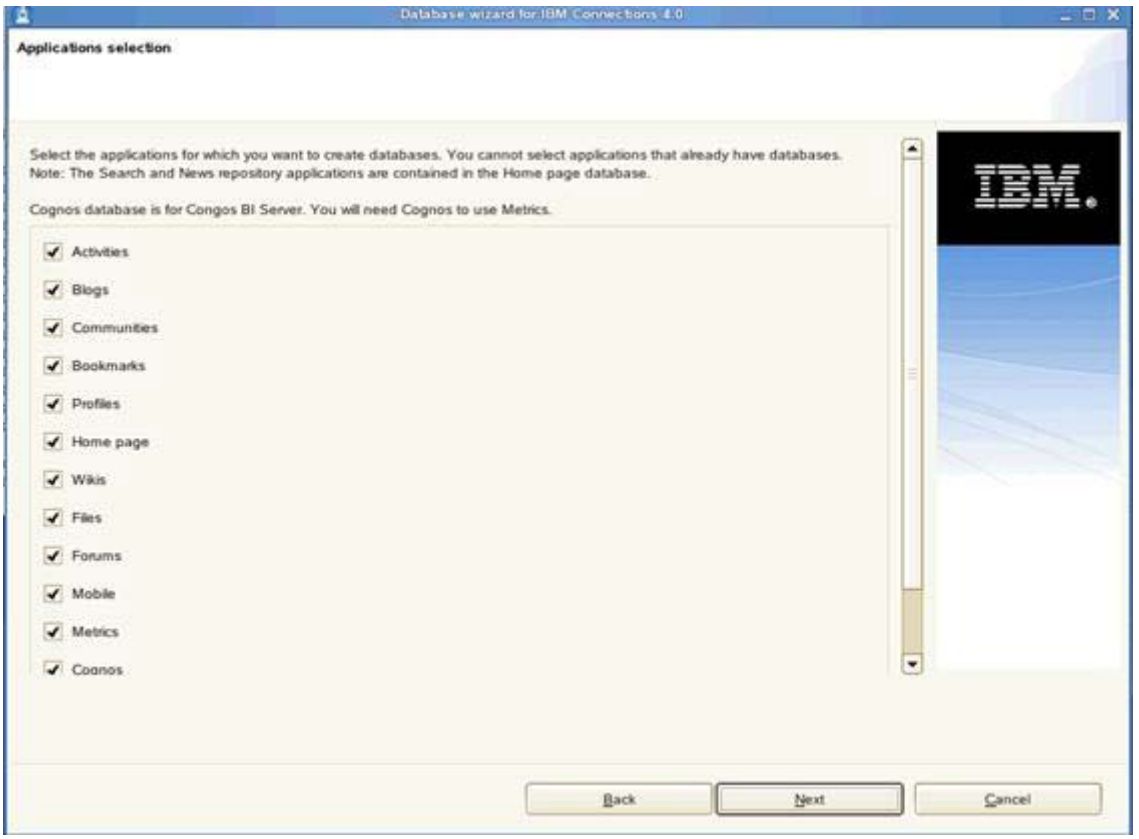


Figure 6. Applications selection

__ 5. Click **Create** from the summary screen.

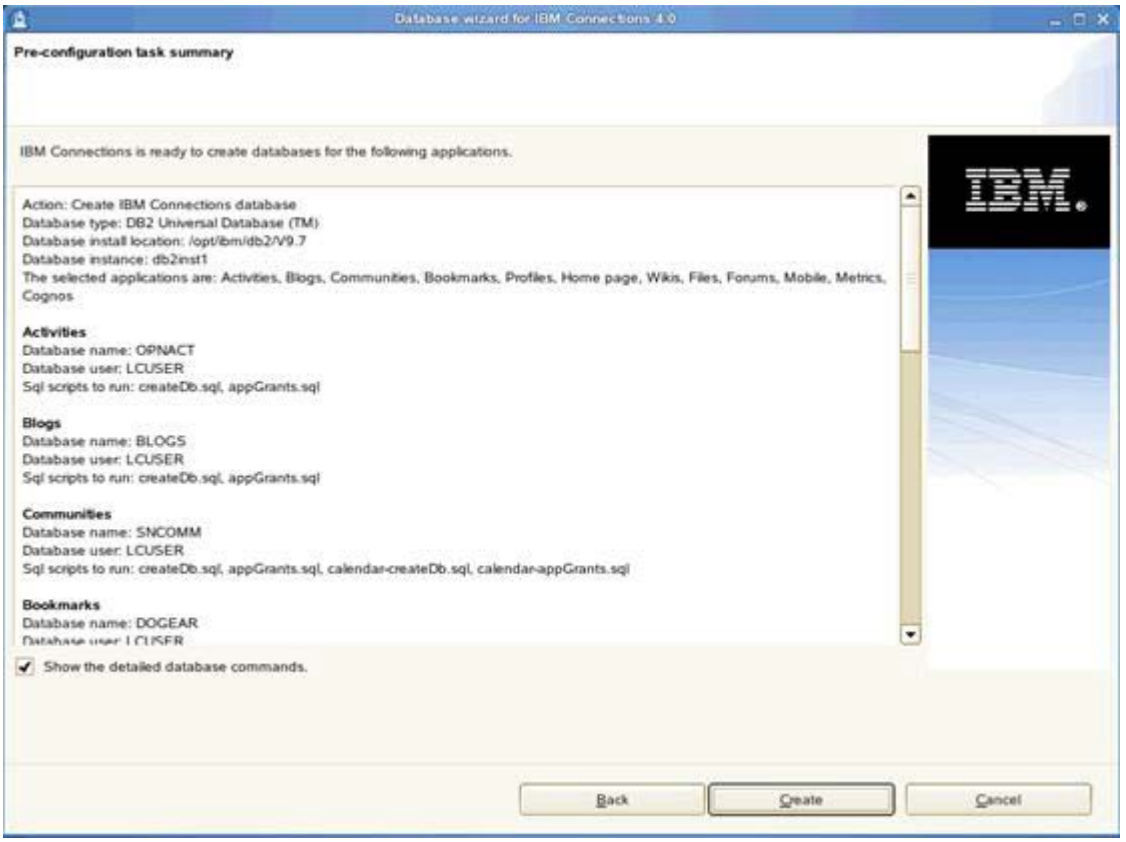


Figure 7. Pre-configuration task summary

__ 6. Finally, click **Execute** to create the DBs.

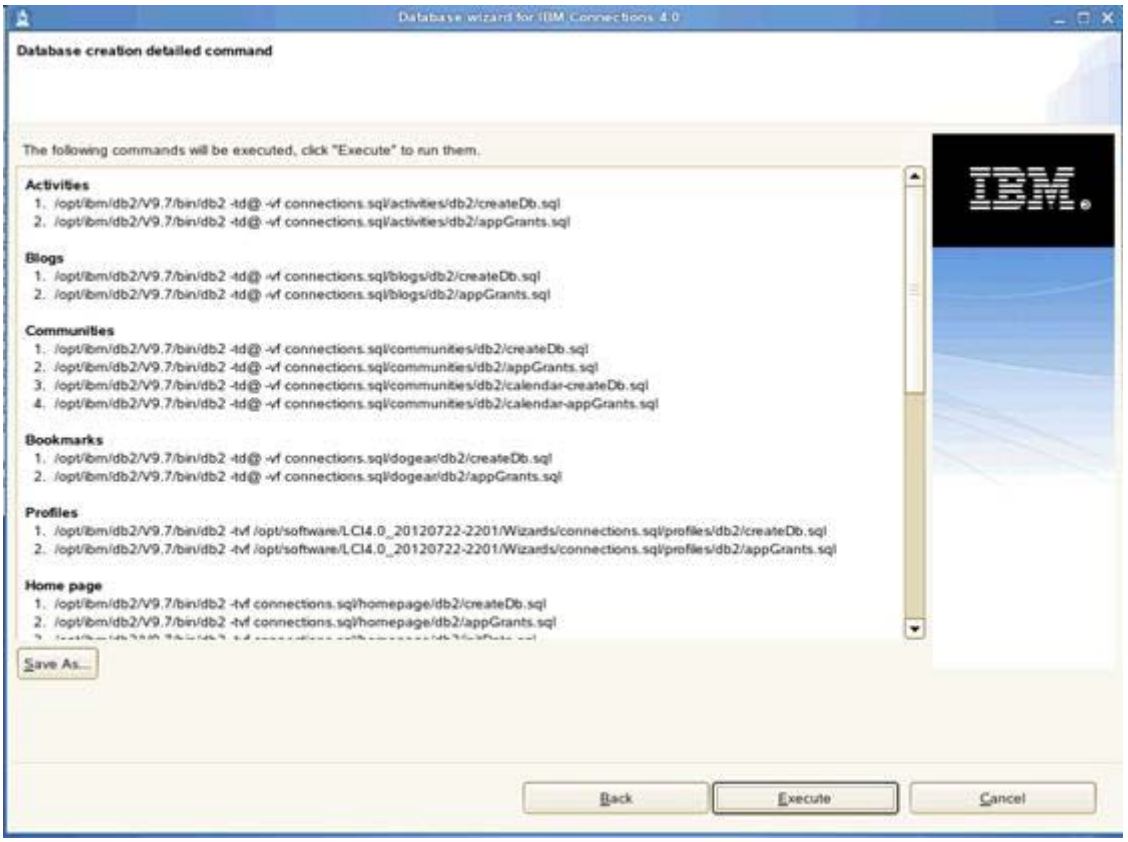


Figure 8. Database creation detailed command

The database is now created and you see the following:

Figure 9. Database creation task: Creation in progress

After some time, the databases are created successfully.

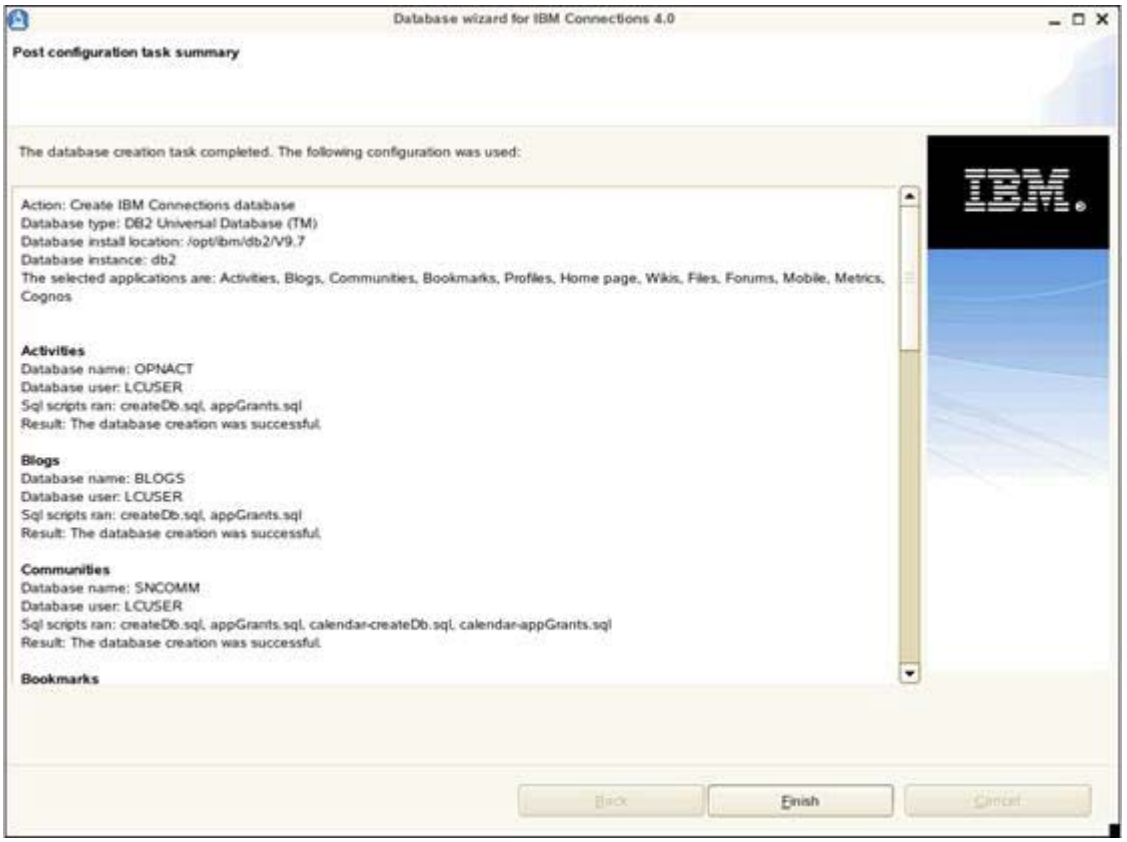


Figure 10. Post configuration task summary

The databases are now created. If you run db2 list database directory, you should see that each database is created.

Install the DB2 client on to your Cognos node

__ 1. Add the following lines to your `.profile` file to allow the root db2 commands:

```
if [ -f /home/db2user/sqllib/db2profile ]; then
 . /home/db2user/sqllib/db2profile
 fi
```

__ 2. Run the following commands to catalog the databases to your node:

```
db2 catalog tcpip node db2server remote db2server.example.com server 50001
db2 catalog database metrics at node db2server
db2 catalog database cognos at node db2server
```

__ 3. In the DB2 client installation directory, open the `/etc/ld.so.conf` file for editing.

__ 4. Add the library `/opt/ibm/db2/V9.7/lib32` to the file.

__ 5. Save and close the file.

__ 6. Run the ldconfig command to regenerate dynamic link libraries (DLLs).

Creating the WebSphere Application Server profile for Cognos server

Cognos needs its own dedicated WebSphere Application Server setup. You can either set up another server for it or create a profile under an existing WebSphere Application Server you set up. If you do choose to set up a new server, then make sure to install all the WebSphere Application Server fix packs, and so on, so it is at the same level as the Deployment Manager you plan to federate into later.

In this document, you create a profile on the existing node one application server that you use for Connections later. So, this node will then run node one for Connections and the cognos_server.

__ 1. On node one of your Application Server, run the following command from `/opt/IBM/WebSphere/AppServer/bin`:

```
./manageprofiles.sh -create -templatepath
/opt/IBM/WebSphere/AppServer/profileTemplates/default -adminUserName admin
-adminPassword password
```

You should see something like the following:

Figure 11. Running the command on the Application Server

Also, if you look under `/opt/IBM/WebSphere/AppServer/profiles` you should see AppSrv01 (your Connections Application server profile) and AppSrv02 (your cognos_server Application server profile).

Setup and configuration of Cognos BI Server and Cognos Transformer

The setup of both the Cognos BI Server and Cognos transformer is automated for Connections so you do not need to set them up manually.

__ 1. Using the Connections installers, look under `/opt/software/LCI4.0_Gold/IBM_Connections_Install` for example and you see a folder called Cognos. Under here, there is a `CognosConfig.tar/zip` (depending on your operating system) which is what is used to set up these pieces.

__ 2. However, before you do that you must copy the Cognos BI Server and Cognos Transformer to your system. Copy them into the WebSphere Application Server that you created the previous profile. Create the following and copy the installation files for each installation type (you must extract the installation files from what you downloaded from Xpertise Library in the first step previously):

```
/opt/software/cognos/BI
/opt/software/cognos/TF
```

When complete, you should see something like this:

```
dslvm768:/opt/software/Cognos # pwd
/opt/software/Cognos
dslvm768:/opt/software/Cognos # ls ./BI
bisrvr_linuxi8664h_10.1.1_ml.tar.gz documentation linuxi38664h zipfiles
dslvm768:/opt/software/Cognos # ls ./TF
bitrsfrmr_linuxi38632_10.1.1_ml.tar.gz documentation linuxi38632 zipfiles
dslvm768:/opt/software/Cognos #
```

__ 3. Copy the `Cognos.zip` from the virtual machine where you extracted `/opt/software/Cognos/GC`.

__ 4. Now, under `/opt/software/Cognos/Gold` extract the `CognosConfig.tar/zip`. When extracted, go to `/opt/software/Cognos/Gold/BI-Customization/JDBC` and copy the JDBC drivers for your database back end to this location. Copy all the .jar files from `/home/db2user/sqllib/java/` to this location. They are needed to make a database connection to the Cognos and metrics databases.

__ 5. Next, configure the `cognos-set-up.properties` file which is used to provide the settings that are needed to perform the installation of the Cognos server and Cognos transformer. Following are the settings that you must supply.

Settings that are needed for cognos-setup.properties

These settings are needed for cognos-setup.properties:

```
# ****************************************************************
#
# Licensed Materials: Property of IBM
#
# 5724-S68
#
# Copyright IBM Corp. 2012  All Rights Reserved.
#
# US Government Users Restricted Rights: Use, duplication or
# disclosure restricted by GSA ADP Schedule Contract with
# IBM Corp.
#
# ****************************************************************
# Location of the already installed WebSphere Application Server where you will
deploy Cognos Business Intelligence
# Examples: C:\Program Files\IBM\WebSphere\AppServer
#           /opt/IBM/WebSphere/AppServer
was.install.path=/opt/IBM/WebSphere/AppServer
# Profile name of the Application Server
# Important: This must not be the Deployment Manager profile
# Default profile is located here: <was.install.path>/profiles/<Profile_Name>
# Example: /opt/IBM/WebSphere/AppServer/profiles/AppSrv01 uses the profile name
AppSrv01
was.profile.name=AppSrv02
# Local WebSphere Application Server administrator username
was.local.admin.username=admin
# Local WebSphere Application Server administrator password
# Note: Password is stored in clear text; leave setting blank to supply it at run
time
was.local.admin.password=password
# The following property is only required for Windows systems.
# The fully qualified host name of this Application Server
# Example: host.example.com
was.fqdn.hostname=node1.example.com

# The WebSphere Application Server node where the Cognos BI server instance will be
created (this must be an existing node)
# The node name can be found in
<was.install.path>/profiles/<Profile_Name>/logs/AboutThisProfile.txt
cognos.was.node.name=Node1Node02
# The server instance name where Cognos BI EAR will be deployed; this server
instance will be created during installation
cognos.was.server.name=cognos_server
```



```
# Location of issetup installer for Cognos BI Server
# The installer is stored below the directory where you expanded the BI Server
package
# Note: Include installer in the path: issetup.exe for Windows; issetup for
non-Windows
# Examples: C:\biserver_10.1.1\winx64h\issetup.exe
#           /opt/biserver_10.1.1/linuxi38664h/issetup
cognos.biserver.issetup=/opt/software/Cognos/BI/linuxi38664h/issetup
# Location of issetup installer for Cognos Transformer
# The installer is stored below the directory where you expanded the Transformer
package
# Note: Include installer in the path: issetup.exe for Windows; issetup for
non-Windows
# Examples: C:\transformer_10.1.1\win32\issetup.exe
#           /opt/transformer_10.1.1/linuxi38632/issetup
cognos.transformer.issetup=/opt/software/Cognos/TF/linuxi38632/issetup

# To deploy and configure the product, fill in the desired install location
#
# Important: BI Server and Transformer cannot share the same install location
#
# Install location of Cognos BI Server
# Examples: C:\Program Files\IBM\Cognos
#           /opt/IBM/Cognos64
cognos.biserver.install.path=/opt/IBM/Cognos/BI
# Install location of Cognos Transformer
# Examples: C:/Program Files (x86)/IBM/Cognos
#           /opt/IBM/Cognos
cognos.transformer.install.path=/opt/IBM/Cognos/TF

# Cognos installation language
# Valid values:
# EN   : English (Default)
# ZH_CN: Chinese (PRC)
# ZH_TW: Chinese (Taiwan)
# FR   : French
# DE   : German
# IT   : Italian
# JA   : Japanese
# KO   : Korean
# PT_BR: Portuguese (Brazil)
# ES   : Spanish
cognos.locale=EN
# Context root of Cognos BI Server application; do not include leading '/'
cognos.contextroot=cognos

# The LDAP user name and password chosen to be the Cognos administrator
# Note: Password is stored in clear text; leave blank to supply at run time
```

```
cognos.admin.username=ldap_admin
cognos.admin.password=password
# The Cognos name space to be used by IBM Connections
cognos.namespace=IBMConnections

# Location where PowerCubes generated by the Transformer are stored
# Examples: C:\Program Files\IBM\Cognos\PowerCubes
#           /opt/IBM/Cognos/PowerCubes
cognos.cube.path=/opt/IBM/Cognos/PowerCubes

# Information for the Cognos Content Store database
# Supported database types:
#    DB Type    : Value
# ============================
#    DB2        : db2
#    Oracle     : oracle
#    SQL Server : sqlserver
cognos.db.type=db2
# Format the cognos.db.host property as: host_name:port
cognos.db.host=db2server.example.com
cognos.db.name=COGNOS
cognos.db.user=db2user
# Note: Password is stored in clear text; leave blank to supply at run time
cognos.db.password=password

# Information for the Metrics database
# Supported database types:
#    DB Type    : Value
# ============================
#    DB2        : db2
#    Oracle     : oracle
#    SQL Server : sqlserver
metrics.db.type=db2
# Format the metrics.db.host property as: host_name:port
metrics.db.host=db2server.example.com
metrics.db.name=METRICS
# The local database name is used by the database client on the Transformer server
to reference the Metrics database.
# For DB2, this is the Metrics database local catalog alias name.
# For Oracle, this is the Metrics database local TNS name.
# For SQL Server, this is the Metrics database instance name.
metrics.db.local.name=METRICS
metrics.db.user=db2user
# Note: Password is stored in clear text; leave blank to supply at run time
metrics.db.password=password
```


__ 6. When all of the settings are entered, run the following command to set up the Cognos server from within `/opt/software/Cognos/Gold`: `./cognos-set-up.sh`.

Assuming that it runs OK, you should see the following when it is finished.

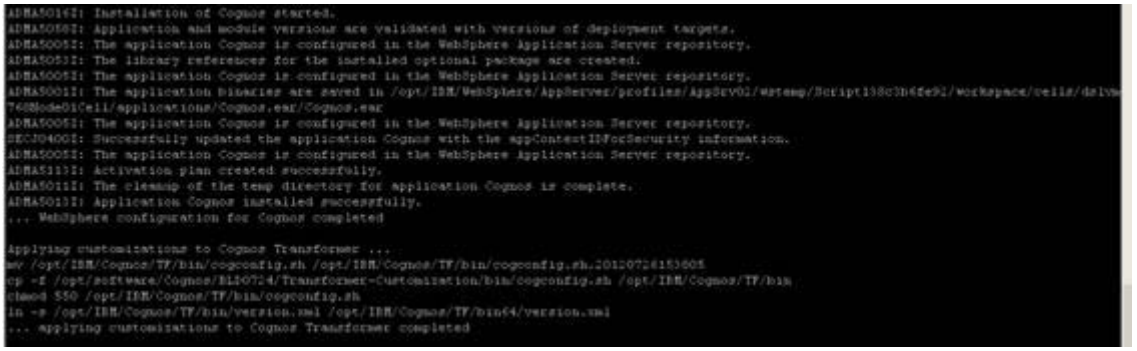


Figure 12. Command to set up the Cognos server

__ 7. Next, configure the cognos_server. Run the following within `/opt/software/Cognos/Gold`: `./cognos-configure.sh`.

Assuming that it runs OK, you should see the following when it is finished.

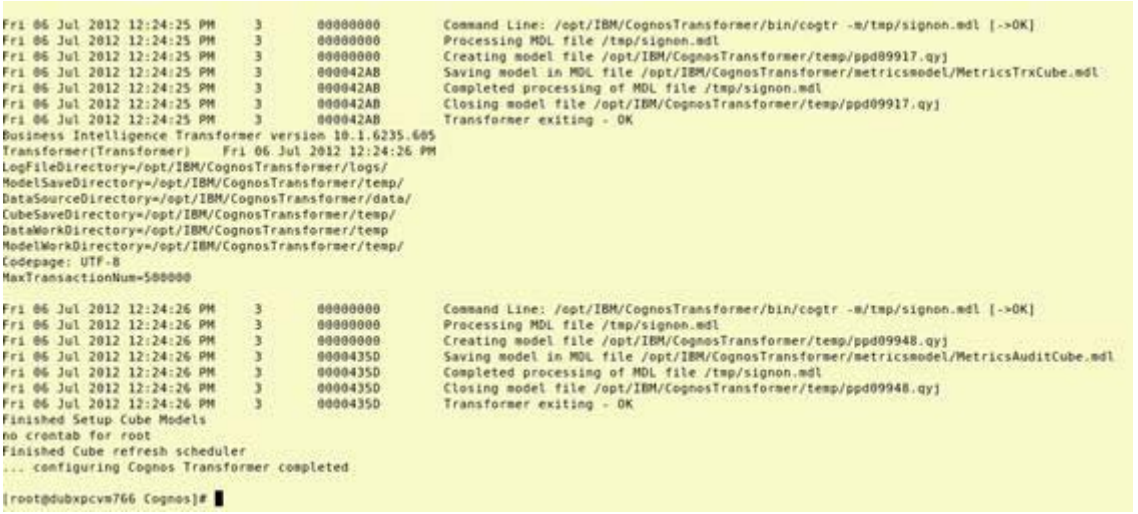


Figure 13. Configuring the Cognos server

Federate the Cognos application server into Deployment Manager

Next, federate the application server into the deployment manager, by running the following commands:

__ 1. Ensure that the clocks are in synch between your Deployment Manager and application server. Run `ntpdate clock.redhat.com` to on your Deployment Manager and application server.

__ 2. Make sure that the Deployment Manager is started and the application server is stopped.

__ 3. Then, from within your `/opt/IBM/WebSphere/AppServer/profiles/AppSrv02/bin` run the following command (make sure to use the `-includeapps` flag):

`./addNode.sh connections.example.com 8879 -includeapps -user admin -password password`

If all goes well, you should see something like this reported:

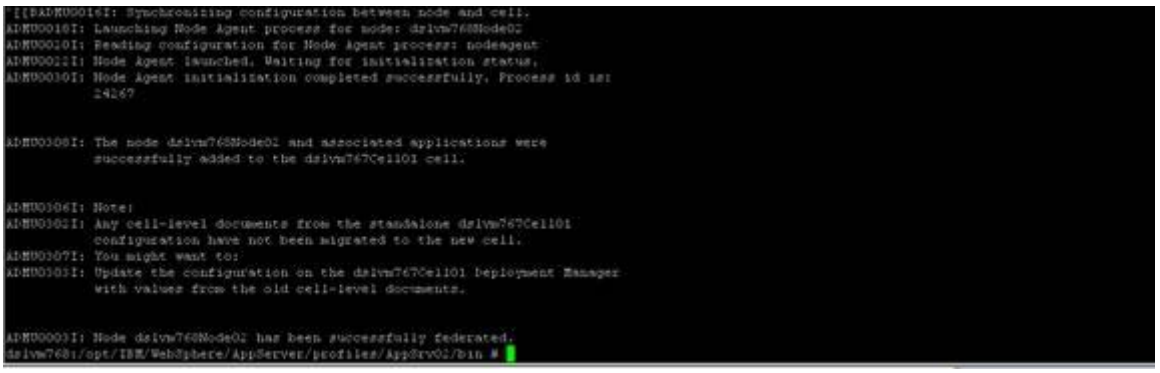


Figure 14. Federating the Cognos Application Server into Deployment Manager

If you log in to your Deployment Manager at `https://connections.example.com:9043/ibm/console/logon.jsp` and go to **Servers > Server Types > WebSphere Application Servers**, you should see something like this.

☐	cognos_server	dslvm768Node02	dslvm768.litbg02.swg.usma.ibm.com	dslvm768.example.com
☐	server1	dslvm771Node01	dslvm771.litbg02.swg.usma.ibm.com	dslvm771.example.com
☐	server1	dslvm768Node01	dslvm768.litbg02.swg.usma.ibm.com	dslvm768.example.com
☐	server1	dslvm768Node02	dslvm768.litbg02.swg.usma.ibm.com	dslvm768.example.com

Figure 15. WebSphere Application Servers

Configure Cognos LDAP security

__ 1. Next, add the LDAP security information into the Cognos configuration tool. Start the Cognos configuration tool.

Linux

On Linux, this tool is found under `/opt/IBM/CognosServer/bin64/cogconfig.sh` but you must export `JAVA_HOME` first before it can be run. Use "**export JAVA_HOME=/opt/IBM/WebSphere/AppServer/java**".

Then, run the `cogconfig.sh` from the same terminal window on VNC.

__ 2. Right-click **Local Configuration > Security > Authentication** and select **New resource > Namespace**.

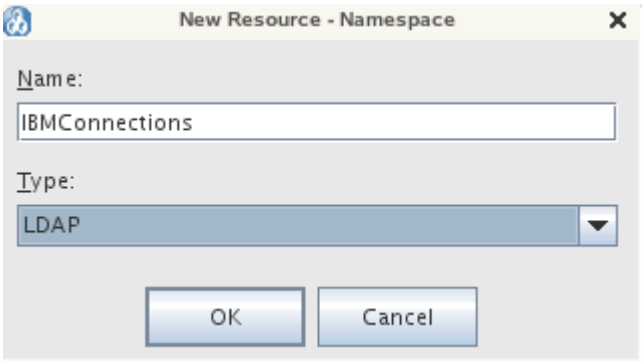


Figure 16. New Resource: Namespace

__ 3. Call it `IBMConnections` and select **LDAP** as the Type. Then, complete the LDAP information. In the following figure, you can see what you must set to get security enabled on Cognos by using MS Active Directory.

IBMConnections – Namespace – Resource Properties

	Name		Value
	Type		LDAP
*	Namespace ID		IBMConnections
*	Host and port		w2k8.example.com:389
*	Base Distinguished Name		OU=SharedLDAP,OU=Lotus,OU=Software G...
	User lookup		(sAMAccountName=${userID})
	Use external identity?		True
	External identity mapping		(sAMAccountName=${environment("REMOT...
	Bind user DN and password		**********
	Size limit		-1
	Time out in seconds		-1
	Use bind credentials for search?		True
	Allow empty password?		False
	Unique identifier		ObjectGUID
	Data encoding		UTF-8
	SSL certificate database		
	Advanced properties		<click the edit button>
	Folder mappings (Advanced)		
	Object class		organizationalunit,organization,container
	Description		description
	Name		ou,o,cn
	Group mappings (Advanced)		
	Object class		Group
	Description		description
	Member		Member
	Name		cn

Figure 17. IBM Connections: Namespace: Resource Properties

Account mappings (Advanced)		
Account object class		user
Business phone		telephonenumber
Content locale		
Description		description
Email		mail
Fax/Phone		facsimiletelephonenumber
Given name		givenname
Home phone		homephone
Mobile phone		mobile
Name		displayName
Pager phone		pager
Password		unicodePwd
Postal address		postaladdress
Product locale		
Surname		sn
User name		sAMAccountName
Custom properties		<click the edit button>

Figure 18. IBM Connections: Namespace: Resource Properties

__ 4. Save it now by selecting **File > Save**.

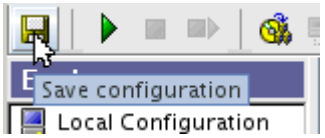


Figure 19. Saving properties

__ 5. Then, right-click **IBMConnections** and click **Test**.

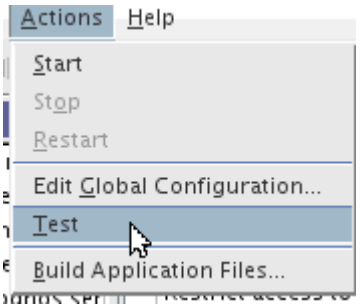


Figure 20. IBMConnections: Test

Tasks should be successful.

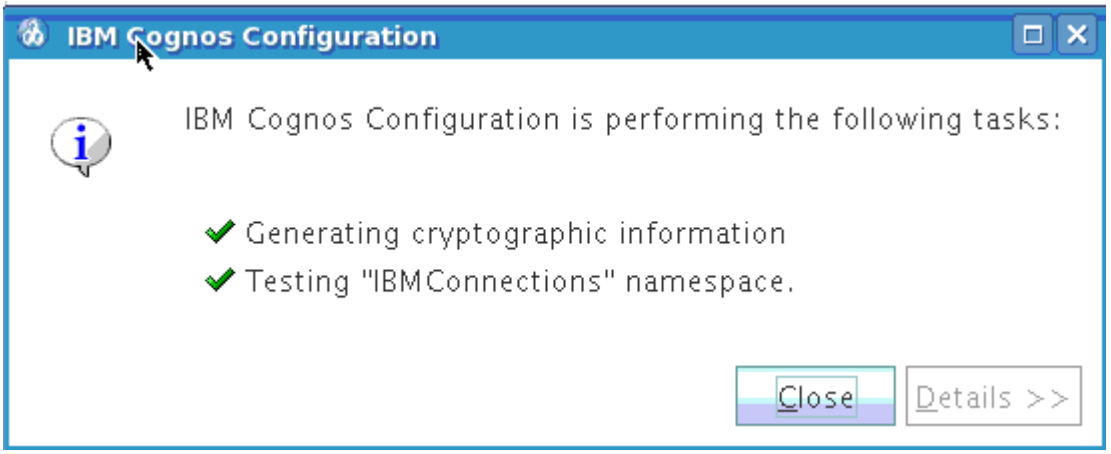


Figure 21. IBM Cognos Configuration

__ 6. Finally, click **Local Configuration > Security > Authentication > Cognos** and set “Allow anonymous access?” to **False**.

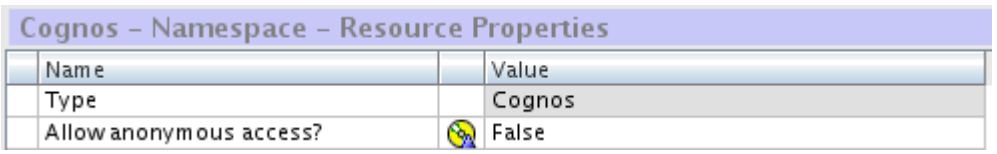


Figure 22. Cognos: Namespace: Resource Properties

__ 7. Save and close the configuration tool.

Note

When exiting Cognos Configuration tool, a message opens and asks you a question. Click **No**.

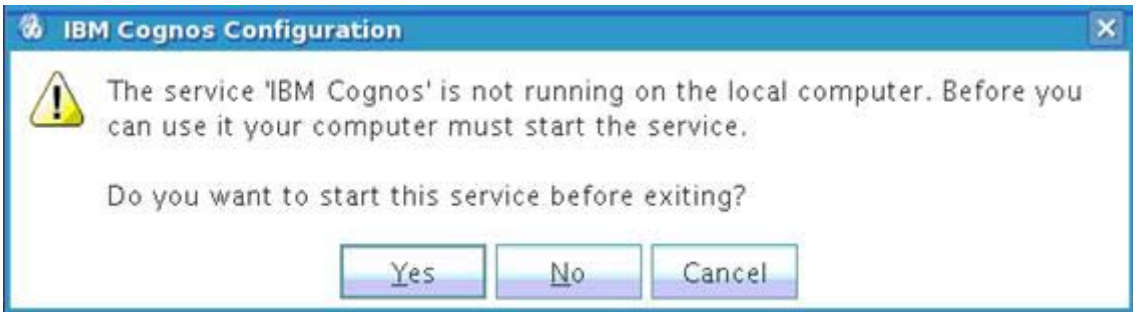


Figure 23. IBM Cognos Configuration warning message

Edit virtual hosts

__ 1. First, check what ports Cognos is using: in the WebSphere Application Server admin console that is, `https://connections.example.com:9043/ibm/console/logon.jsp`.

__ 2. Go to `servers\websphere applications servers\cognos-server\communications`.

__ 3. Click **+** at ports and look for the following hosts:

Figure 24. WC_defaulthost

Figure 25. WC_defaulthost_secure

__ 4. Go to `environment\Virtual hosts` and look for `node1.example.com 9082` and `node1.example.com 9445` and delete these entries.

__ 5. Add in two entries for * and 9082 and * + 9445.

__ 6. Click **Save**.

__ 7. For it to take effect, restart the nodes and the deployment manager.

Verification step

Cognos is now set up. You can now start the Cognos server and validate that it is working.

__ 1. Log in to your deployment manager and go to **Servers**/**Server Types**/**WebSphere Application Servers** and start up the cognos_server application. It should start cleanly. If you have an HTTP configured against your system, generate the plug-in, and start up the HTTP server.

__ 2. Go to the URL `https://node1.example.com:9445/cognos/servlet/` you should see the following which confirms that Cognos is set up.

Figure 26. IBM Cognos: Content Manager

__ 3. Enter `https://node1.example.com:9445/cognos/servlet/dispatch`. You should be able to log in to the BI Content Manager as user \ password: this proves that the LDAP security is configured correctly:

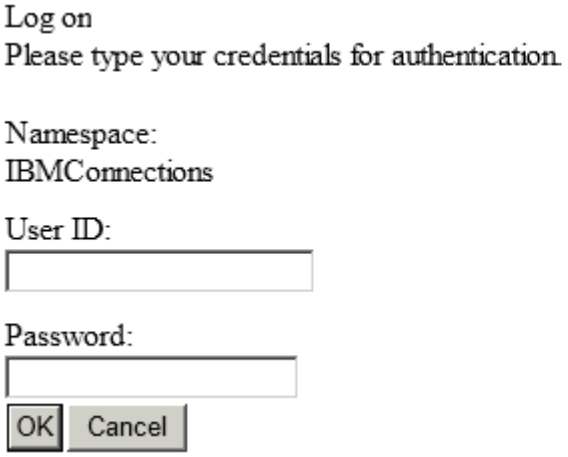


Figure 27. Logging in to the BI Content Manager

When you log in as the admin, you see:

Figure 28. Logging in to the BI Content Manager as the admin

Building the Power Cubes on Cognos

__ 1. To build the Power Cubes, go to the /opt/IBM/CognosTransformer/metricsmodel/ and run build-all.sh to build the cubes.

__ 2. When done, run the build-all.sh/bat to build the cubes.

__ 3. Check the file trxschelog.log file under /opt/IBM/CognosTransformer/metricsmodel for errors and success. You should see:

```
Thu 19 Jul 2012 11:47:10 AM 3 00000000 Command Line:
/opt/IBM/Cognos/TF/bin/cogtr -c -s -g
-f/opt/IBM/Cognos/TF/metricsmodel/promptStartBuild.xml
-m/opt/IBM/Cognos/TF/metricsmodel/MetricsAuditCube.mdl [->OK]
 Thu 19 Jul 2012 11:47:10 AM 3 00000000 Processing MDL file
/opt/IBM/Cognos/TF/metricsmodel/MetricsAuditCube.mdl
 Thu 19 Jul 2012 11:47:10 AM 3 00000000 Creating model file
/opt/IBM/Cognos/TF/temp/ppd31792.qyj
 Thu 19 Jul 2012 11:47:10 AM 3 0000435D Completed processing of MDL file
/opt/IBM/Cognos/TF/metricsmodel/MetricsAuditCube.mdl
 Thu 19 Jul 2012 11:47:10 AM 4 0000435D Start cube update.
 Thu 19 Jul 2012 11:47:10 AM 4 0000435D Initializing categories.
 Thu 19 Jul 2012 11:47:10 AM 4 0000435D Timing,

INITIALIZING CATEGORIES,00:00:00
```

If it goes well, you should see:

```
20120719114803 : build all data success
```

2. Connections installation

Installation of Connections 4.0

Note

The installation of Lotus Connections 4.0 is done on the Deployment Manager server and then synched with the nodes.

Make sure that your Deployment Manager and the nodes are started.

If you are installing the Metrics application, ensure that you installed and configured Cognos.

Ensure that the directory paths that you enter contain no spaces.

Ensure that the Open File Descriptor limit is 8192.

Follow these steps for how to set the limit.

__ 1. Open a command line and enter the following command to find the current open file limit: `ulimit -a`.

__ 2. Add the following line to the user's profile file: `ulimit -n 8192`.

__ 3. Check the previous item on Node 1 and Node 2 as well.

__ 4. Ensure that the GTK library is available on your system. If you are installing on a 64-bit system, you also need the 32-bit version of the GTK library. Check it on Deployment Manager, Node 1 and Node 2. To check, run `rom -qa | grep gtk`.

__ 5. You must install the interim fixes

- PM53930
- PM56596
- PM60895
- WebSphere Application Server PK 7.0.0.21-WS-WAS-TFPM65486.pak

__ 6. Copy the installation files to you server and extract the `Lotus_Connections_4.0_lin_aix.tar` file. Start the installation by running `./launchpad.sh` under `Lotus_Connections_Install`. The following wizard is displayed:

Figure 29. IBM Connections 4.0.0: Welcome

__ 7. In the left pane of the launchpad, click **Install IBM Connections 4.0** and then click **Launch the IBM Connections 4.0 install wizard** in the right pane.

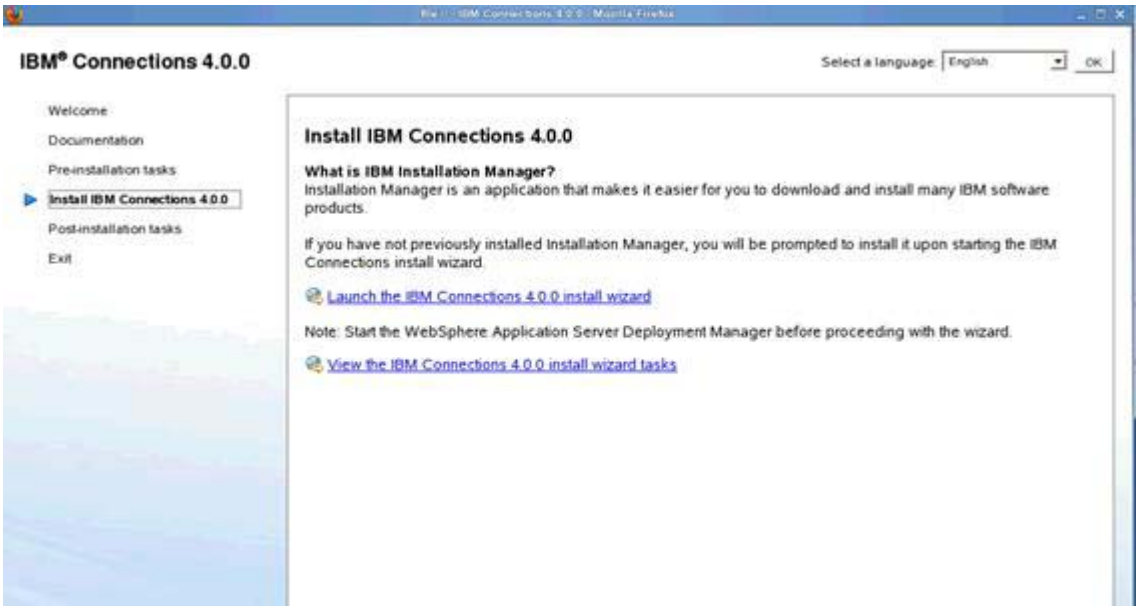


Figure 30. Install IBM Connections 4.0.0

__ 8. In the Select packages to install window, select the packages that you want to install and click **Next** to continue.

Figure 31. Install packages

__ 9. Review and accept the license agreement by clicking **I accept the terms in the license agreements**. Click **Next**.

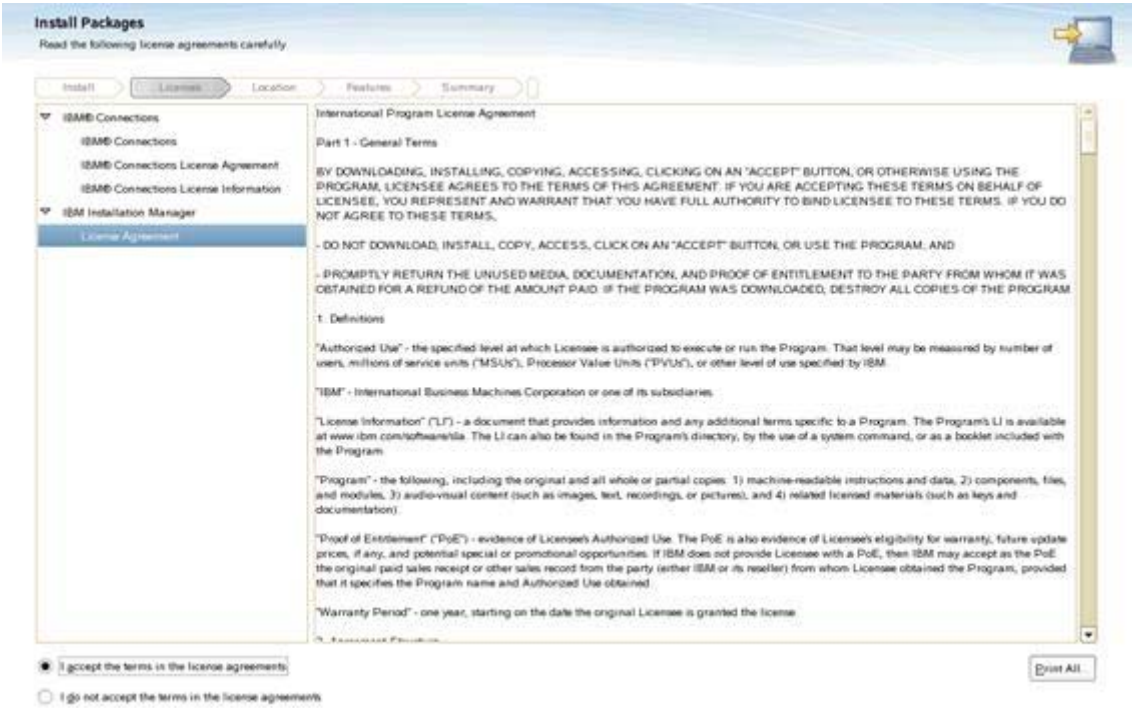


Figure 32. License agreement

__ 10. Specify the location of shared directories for IBM Installation Manager. Click **Next** to continue.

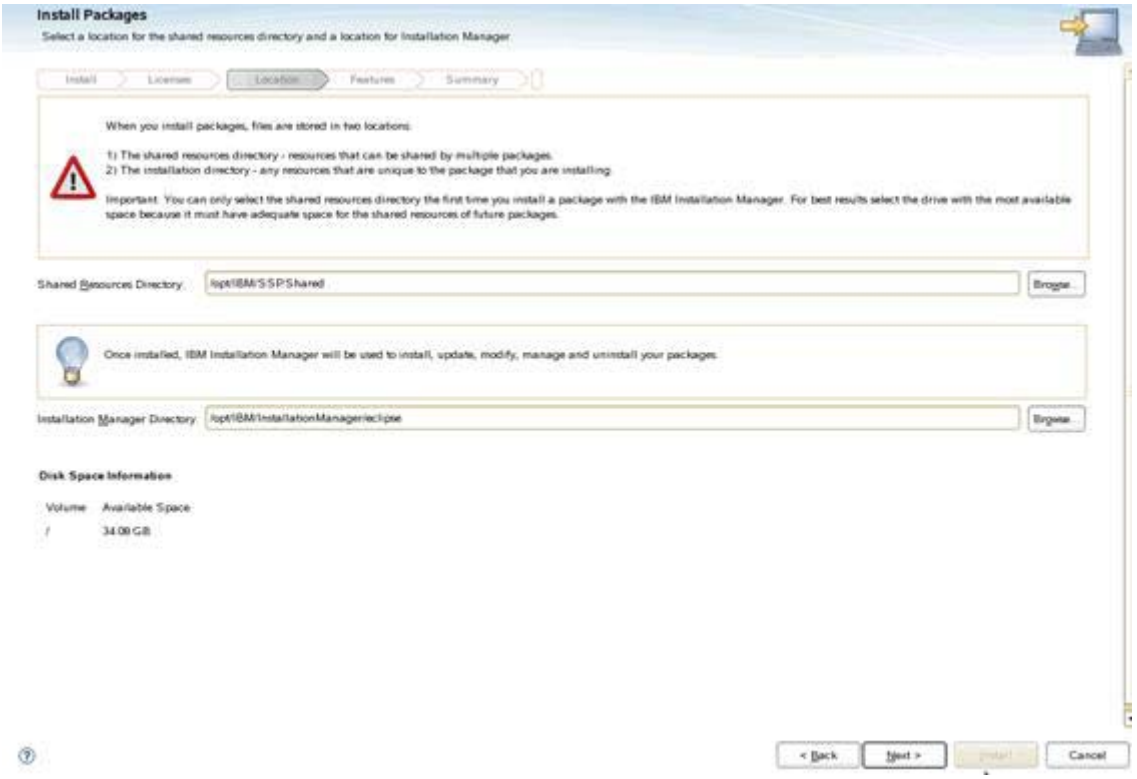


Figure 33. Shared directories location

__ 11. Choose to use the existing package group or to create a package group. Click **Next** to continue.

__ 12. Specify the location of the installation directory for IBM Connections. You can accept the default directory location, or enter a new directory name, or click **Browse** to select an existing directory. Click **Next**.

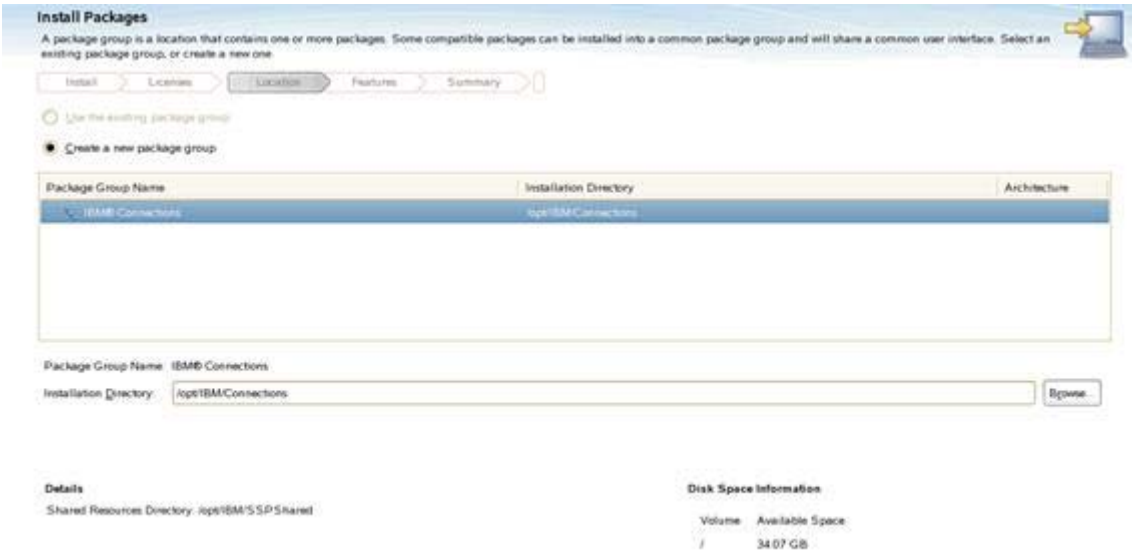


Figure 34. Installation directory location

__ 13. Confirm all the applications that you want to install. As Cognos is enabled, Metrics is selected. If Cognos is not enabled, clear Metrics and click **Next** to continue.

Figure 35. Confirming the applications to install

Figure 36. Confirming the applications to install

__ 14. Select the path to the WebSphere Application Server instance that is running on your deployment manager. For example, `/opt/IBM/WebSphere/DeploymentManager`. Enter the host name.

__ 15. The admin user name and password. Ensure that it is the admin user (that you use to log in to WebSphere Application Server) that you set when you enabled security previously and not the default user, wasadmin.

__ 16. Click **Validate** at the bottom.

__ 17. If you get an error on validation, check that you pointed to the correct Deployment ManagerGR, the user and password are correct, and administrative and application security boxes were checked when you enabled security.

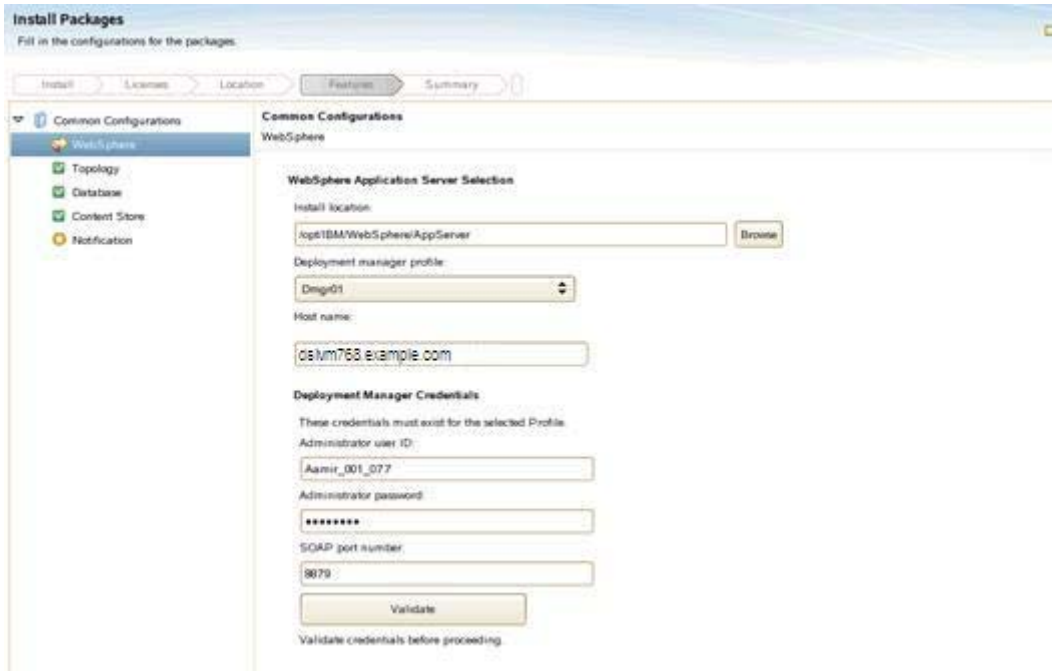


Figure 37. Configurations for the packages

__ 18. This retrieves the SSL certificate from the Deployment Manager and confirm if all is **OK**.

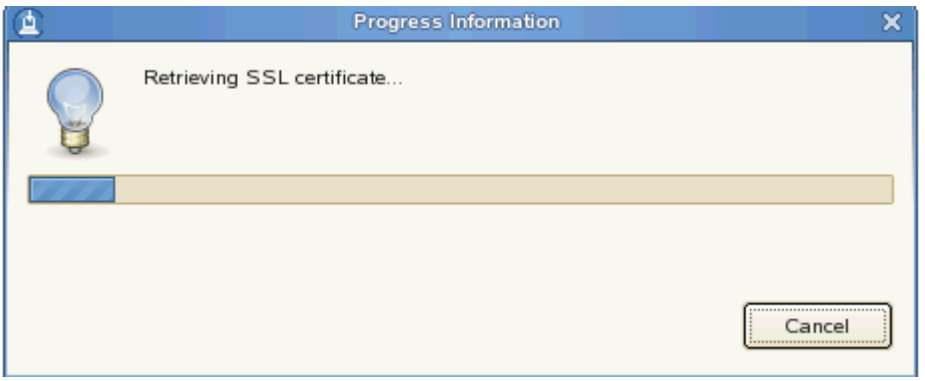


Figure 38. Retrieving SSL certificate

__ 19. Click **Topology Configuration** on the left side. Choose **Medium** configuration. Click **Next** when done.

Important

Ensure to click all the boxes at for Node1Node1 and Node2Node1.

Reminder

Remember that you must install metrics to the Connections server, on Node1Node1 and Node2Node1.

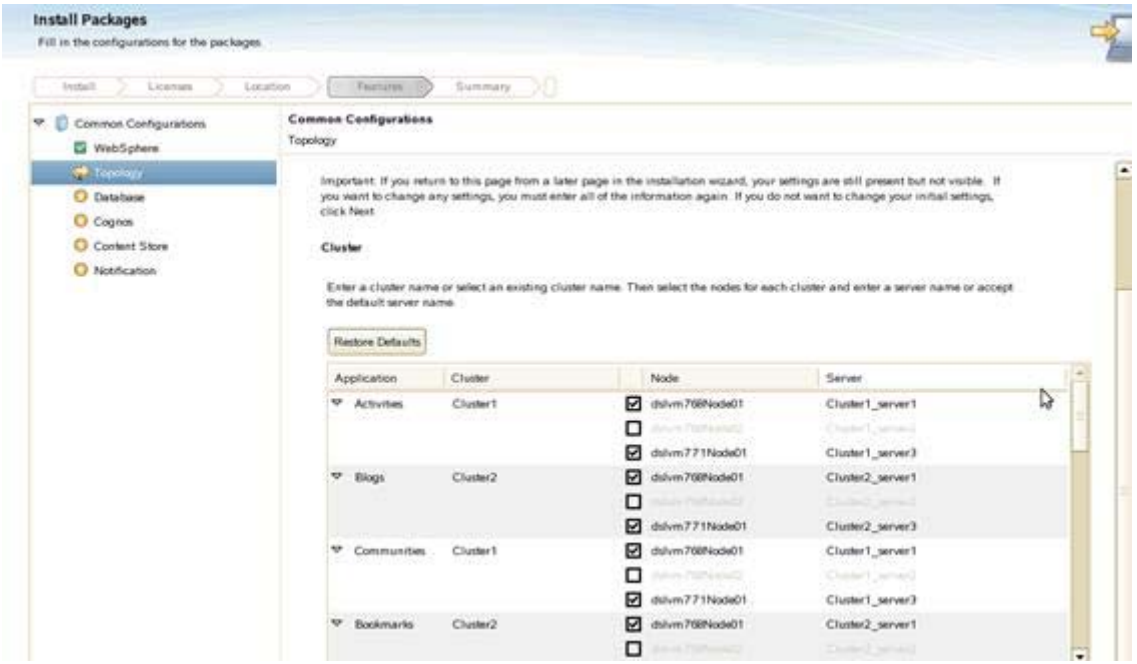


Figure 39. Topology configuration

__ 20. Next is the database configuration. Ensure that your database server is started. Click **Yes, the applications are on the same database instance**. Enter the host name and port of your database server.

__ 21. Then, scroll down and enter the JDBC driver location, `/opt/ibm/db2/V9.7/java` in this example. Create your databases as db2user so the user ID and password are db2user.

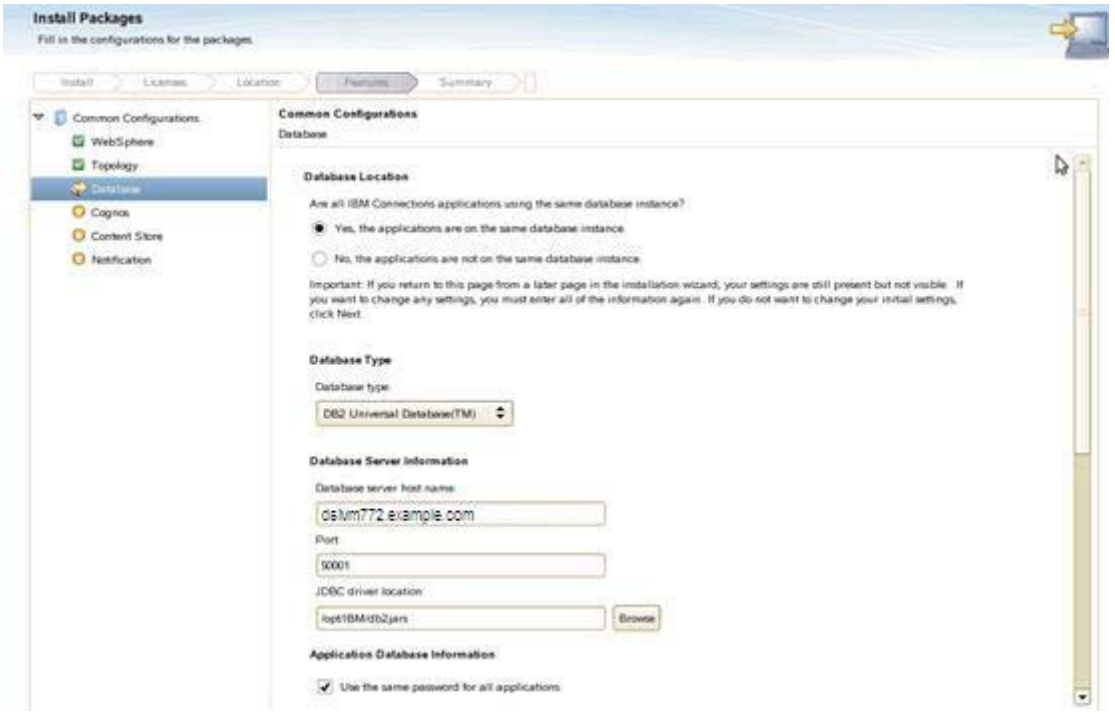


Figure 40. Configuring the packages

__ 22. Click **Validate**.

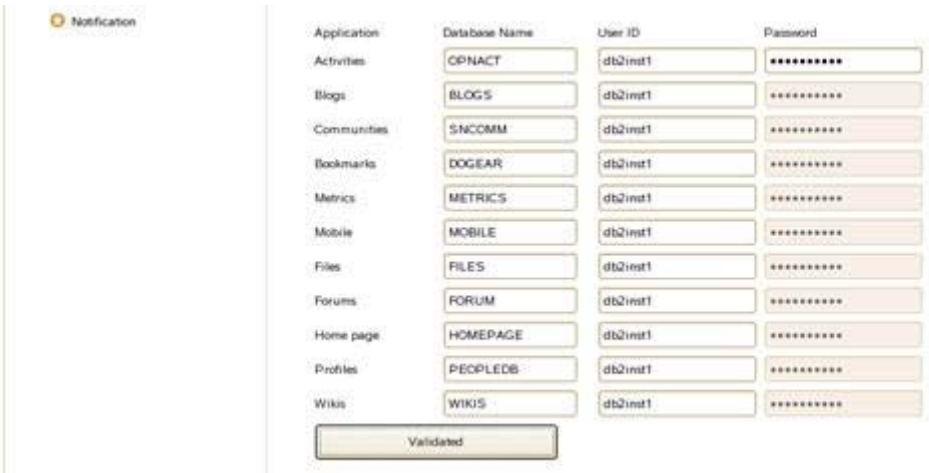


Figure 41. Validating the configuration for the packages

This is then displayed.

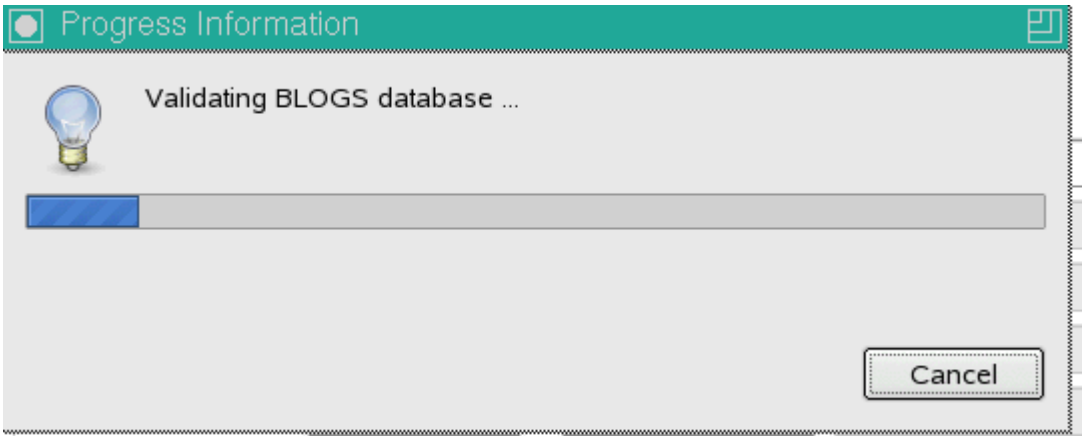


Figure 42. Validating BLOGS database

When the validation is complete, you should see the following.

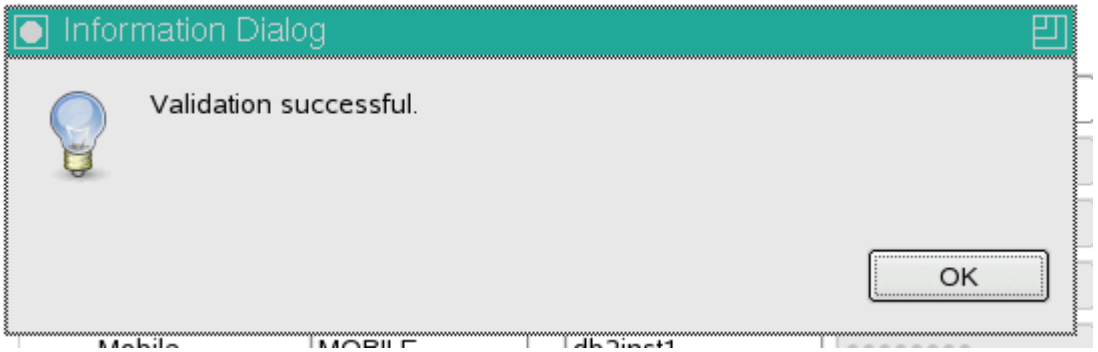


Figure 43. Validation successful

__ 23. Click **Next** to continue.

__ 24. Now you are asked for Cognos information. Enter your admin ID for Cognos, click **Load node info** to select the Cognos node and click **Validate**.

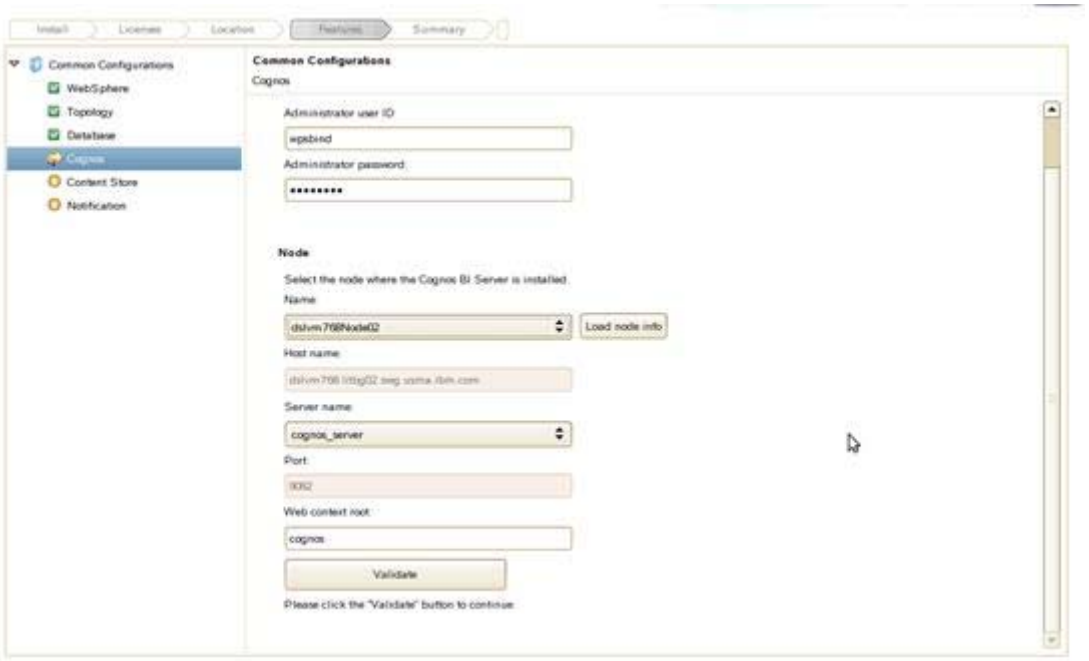


Figure 44. Entering ID for Cognos, loading node info and validating

__ 25. Now you are asked about the content store. In a cluster or where the Deployment Manager and Nodes are not installed on the same computer. It should be a shared location were full read/write access is granted.

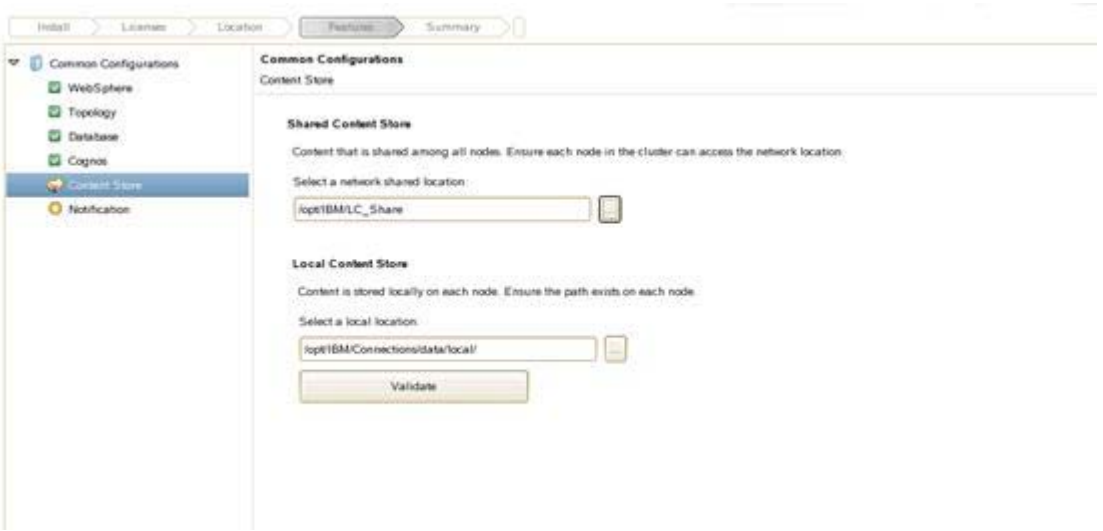


Figure 45. Content store information

__ 26. Change the shared content store to `/opt/IBM/LC_Share` which both nodes have access to. Click **OK** when ready.

__ 27. Click **Validate**, **OK**, and then **Next**.

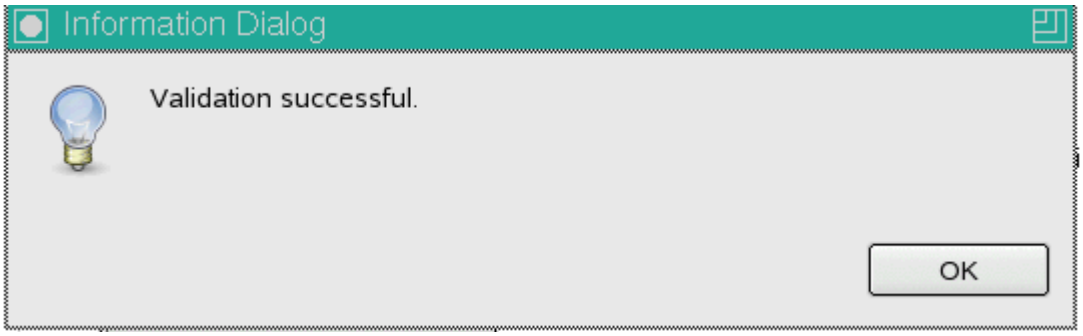


Figure 46. Validation successful

__ 28. Finally, if mail is not configured, click **None** to not enable notification from the notification configuration screen. Otherwise, click **Enable notification and ReplyTo**.

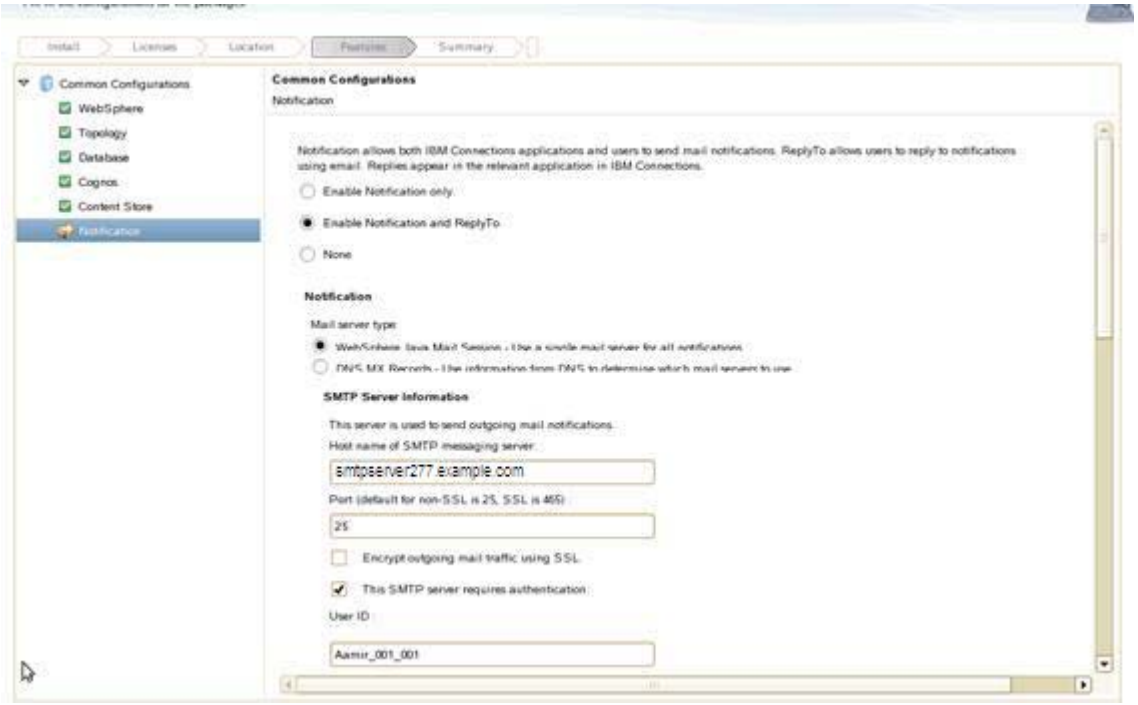


Figure 47. Enabling notification

Figure 48. Enabling notification

__ 29. Lastly, the summary screen. When ready, click **Install**.

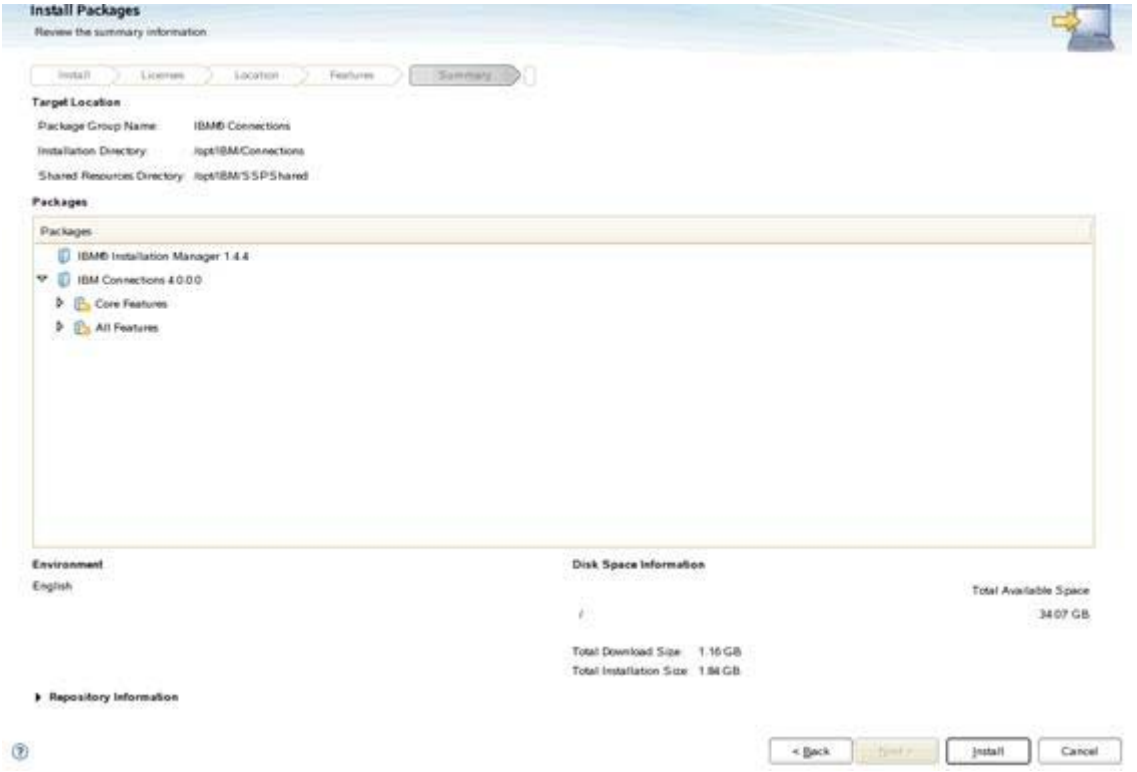


Figure 49. Summary information

The installation starts.

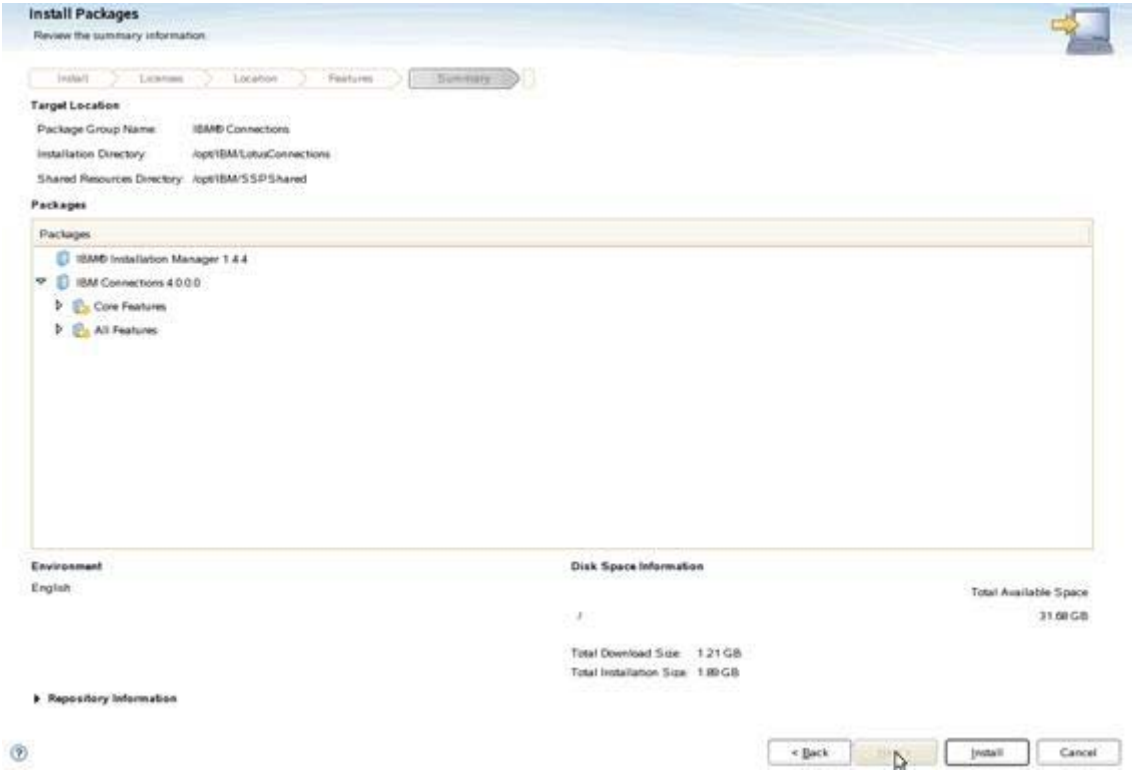


Figure 50. Installation in progress

When complete, you should see the following result:

Figure 51. Install successful

__ 30. Click **Finish** to close the Connections installer.

Quick check of your Connections 4.0 installation and server definitions

__ 1. Stop your Node Agents and Deployment Manager.

__ 2. Then, start up your Deployment Manager, and when it is started, start up your Node Agents.

__ 3. Wait sometime for synchronization to complete.

__ 4. After your Node Agents start up, check their logs, and you should see many messages:

```
[7/19/12 14:15:52:850 IST] 00000031 NodeSyncTask  A   ADMS0003I: The
configuration synchronization completed successfully.
 [7/19/12 14:15:55:183 IST] 00000032 AppBinaryProc I   ADMA7021I:
Distribution of application Common completed successfully.
 [7/19/12 14:15:58:811 IST] 00000032 AppBinaryProc I   ADMA7021I:
Distribution of application Mobile completed successfully.
 [7/19/12 14:16:01:010 IST] 00000032 AppBinaryProc I   ADMA7021I:
Distribution of application Mobile Administration completed successfully.
 [7/19/12 14:16:25:335 IST] 00000032 AppBinaryProc I   ADMA7021I:
Distribution of application WidgetContainer completed successfully.
 [7/19/12 14:16:27:581 IST] 00000032 AppBinaryProc I   ADMA7021I:
Distribution of application Metrics completed successfully.
 [7/19/12 14:16:31:021 IST] 00000032 AppBinaryProc I   ADMA7021I:
Distribution of application Search completed successfully.
 [7/19/12 14:16:34:469 IST] 00000032 AppBinaryProc I   ADMA7021I:
Distribution of application Profiles completed successfully.
 [7/19/12 14:16:39:191 IST] 00000032 AppBinaryProc I   ADMA7021I:
Distribution of application Activities completed successfully.
 [7/19/12 14:16:40:561 IST] 00000032 AppBinaryProc I   ADMA7021I:
Distribution of application Moderation completed successfully.
 [7/19/12 14:16:43:496 IST] 00000032 AppBinaryProc I   ADMA7021I:
Distribution of application Files completed successfully.
 [7/19/12 14:16:49:322 IST] 00000032 AppBinaryProc I   ADMA7021I:
Distribution of application Communities completed successfully.
 [7/19/12 14:16:49:448 IST] 00000032 AppBinaryProc I   ADMA7021I:
Distribution of application ibmasyncrsp completed successfully.
 [7/19/12 14:16:52:061 IST] 00000032 AppBinaryProc I   ADMA7021I:
Distribution of application News completed successfully.
 [7/19/12 14:16:53:075 IST] 0000003b NodeSyncTask  A   ADMS0003I: The
configuration synchronization completed successfully.
 [7/19/12 14:16:55:061 IST] 00000032 AppBinaryProc I   ADMA7021I:
Distribution of application Homepage completed successfully.
 [7/19/12 14:16:58:410 IST] 00000032 AppBinaryProc I   ADMA7021I:
Distribution of application Forums completed successfully.
 [7/19/12 14:17:00:692 IST] 00000032 AppBinaryProc I   ADMA7021I:
Distribution of application Wikis completed successfully.
 [7/19/12 14:17:05:040 IST] 00000032 AppBinaryProc I   ADMA7021I:
Distribution of application Blogs completed successfully.
 [7/19/12 14:17:14:570 IST] 00000032 AppBinaryProc I   ADMA7021I:
```



```
Distribution of application Help completed successfully.
 [7/19/12 14:17:18:073 IST] 00000032 AppBinaryProc I   ADMA7021I:
Distribution of application Dogear completed successfully.
 [7/19/12 14:17:52:981 IST] 00000040 NodeSyncTask  A   ADMS0003I: The
configuration synchronization completed successfully.
 [7/19/12 14:18:52:989 IST] 00000042 NodeSyncTask  A   ADMS0003I: The
configuration synchronization completed successfully.
```

__ 5. Then, start the Connections servers.

Setting path variables for search

During the installation, you set `/opt/IBM/LC_Share` which then set `/opt/IBM/LC_Share/search/stellent/dcs/oiexport` as the location for the `stellent` converters. In a multi-node cluster, it is recommended to run it on the nodes themselves and not the shared area.

__ 1. Copy the folder `/opt/IBM/LC_Share/search/stellent` to `/opt/IBM/Connections/stellent` on both nodes in your cluster. Change the rights on the folder to 777.

__ 2. In `/opt/IBM/Connections/stellent`, run `cp -rf /opt/IBM/LC_Share/search/stellent/*`.

__ 3. Run `chmod 777 -R *`.

__ 4. Set up that share and then goto **Environment > WebSphere Variables and FILE_CONTENT_CONVERSION**. Change the path from the shared area to the local area on your nodes. This should be the same across both nodes.

☐	FILES_EVENT_CONTENT_DIR	${FILES_CONTENT_DIR}	Cell=dslvm767Cell01
☐	FILE_CONTENT_CONVERSION	/opt/IBM/Connections/stellent/dcs/oiexport /exporter	Cell=dslvm767Cell01
☐	FORUM_CONTENT_DIR	/opt/IBM/LC_Share/forums/content	Cell=dslvm767Cell01
☐	FORUM_HOME	/opt/IBM/Connections/forum/forum/forum	Cell=dslvm767Cell01

Figure 52. Environment > WebSphere Variables and FILE_CONTENT_CONVERSION

__ 5. Then, add `/opt/IBM/Connections/stellent/dcs/oiexport` to your PATH variable in `.profile` for the root user.

__ 6. Either add export `LD_LIBRARY_PATH=/opt/IBM/Connections/stellent/dcs/oiexport` to `/opt/IBM/WebSphere/AppServer/bin/set-upCmdLine.sh` and run `. ./set-upCmdLine.sh` before you start the nodes or add export `LD_LIBRARY_PATH=/opt/IBM/Connections/stellent/dcs/oiexport` and add the line to the PATH in `.profile`.

```
export LD_LIBRARY_PATH=/usr/local/staf/lib:/opt/IBM/Connections/stellent/dcs/oiexport

ulimit -n 8192

PATH=/opt/IBM/WebSphere/AppServer/java/jre/bin:/opt/IBM/Connections/stellent/dcs/oiexport:$PATH:$HOME/bin

export PATH
```

Figure 53. Adding the PATH variable in .profile

__ 7. To check that `LD_LIBRARY_PATH` is checked, enter echo `$LD_LIBRARY_PATH`.

```
dslvm771:~ # echo $LD_LIBRARY_PATH
/usr/local/staf/lib:/opt/IBM/Connections/stellent/dcs/oiexport
dslvm771:~ #
```

Figure 54. Checking that LD_LIBRARY_PATH is checked

Note

Do it on all nodes of your cluster.

__ 8. Restart the server. Then, to make sure that the variables take effect or in /root/ folder run `.profile`.

Information

For more information about this extra step, see `http://www-10.lotus.com/ldd/lcwiki.nsf/xpDocViewer.xsp?lookupName=IBM+Connections+4.0+documentation#action=openDocument&res_title=Copying_Search_conversion_tools_to_local_nodes_ic40&content=pdcontent`.

Populate the PROFILES database with LDAP user information

Note

How to avoid an OOM when populating PROFILES database with 300K users

Previously the defect 59044: Profile Tivoli Directory Integrator Population OOM against IC 4.0 Builds to fix an OOM issue with profiles when populating large LDAP directories. Unfortunately this defect is deferred as there is a work-around. The work-around is to increase the JVM size of the Tivoli Directory Integrator process that does the population into the PROFILES database. This is done by adding `-Xms256M -Xmx3072M` to `ibmdisrv/ibmdisrv.bat` on your Tivoli Directory Integrator server.

This is documented for customers at the following link, so if you have any problems with populating your systems, this is how to do it:

`http://www-10.lotus.com/ldd/lcwiki.nsf/xpDocViewer.xsp?lookupName=IBM+Connections+4.0+documentation#action=openDocument&res_title=Configuring_Tivoli_Directory_Integrator_ic40&content=pdcontent`.

This example uses a 3 GB heap size. If you have more available memory on your systems, then you should increase the -XmxXXXXM to reflect your own memory.

Do it on the server where you installed Tivoli Directory Integrator. In this example, it is on the DB2 server.

Populating the PROFILES database with LDAP user can now be done by a wizard.

__ 1. Copy the `Lotus_Connections_4.0_wizards_lin_aix.tar` to your server and extract it. Then, go into the Wizard folder and run `./populationWizard.sh`. The following screen is shown. Click **Next** to continue.

__ 2. On the Welcome page of the wizard, click **Launch Information Center** to open the IBM Connections Information Center in a browser window. Click **Next** to continue.

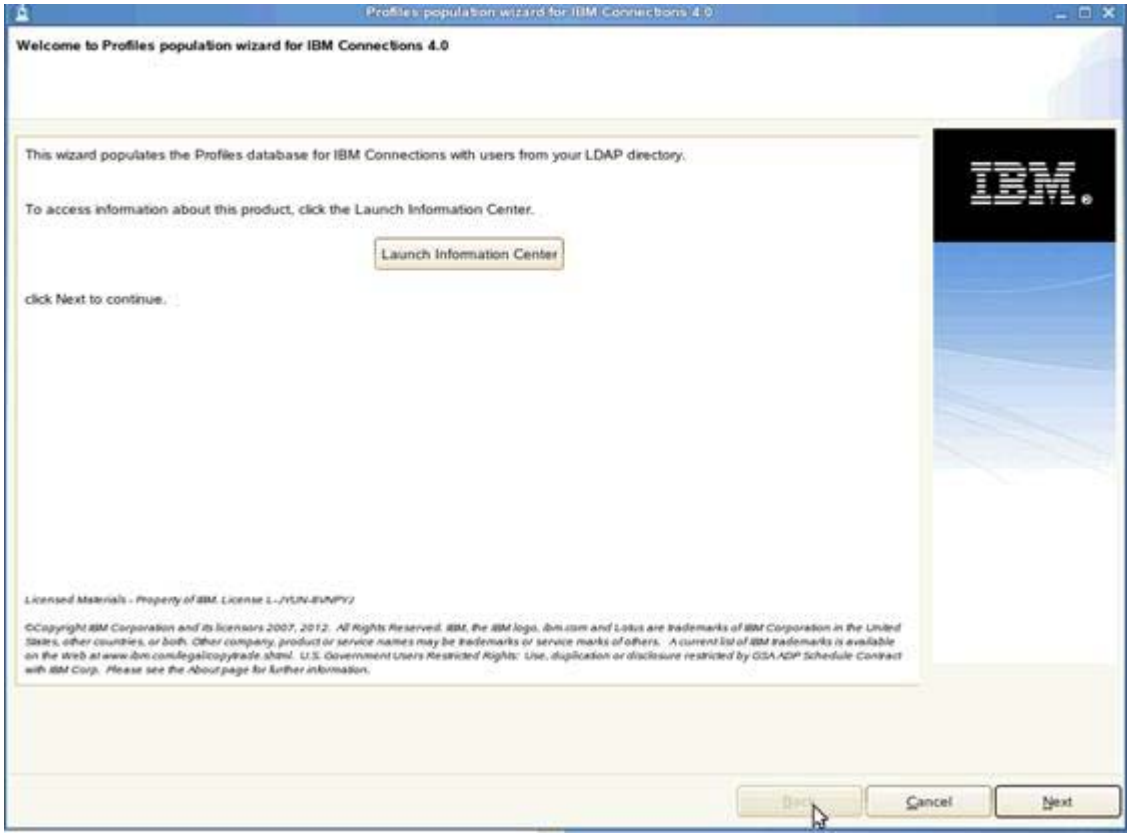


Figure 55. Profiles population wizard for IBM Connections 4.0

__ 3. Click Default settings or, if you are resuming an earlier session, click **Last successful default settings** and click **Next**.

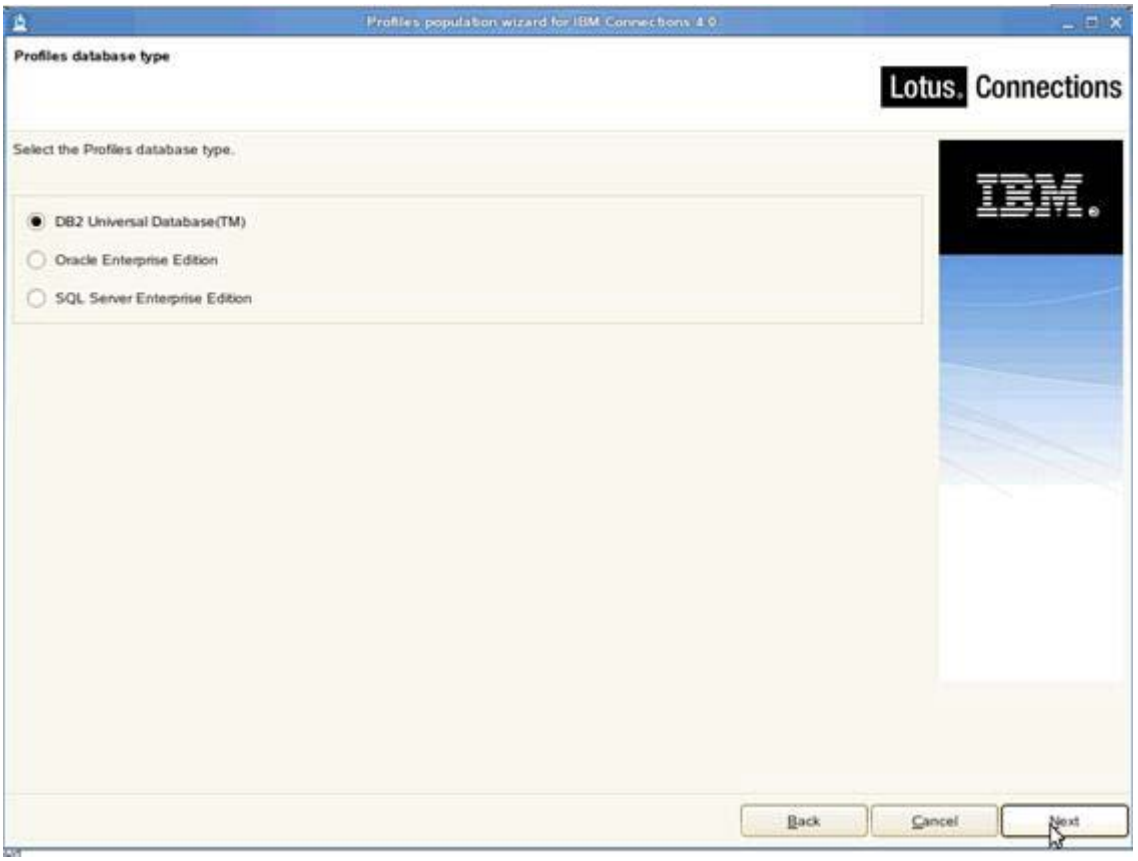


Figure 56. Profiles database type

__ 4. Next, enter the database information for where your PEOPLEDB database is located and click **Next** to continue.

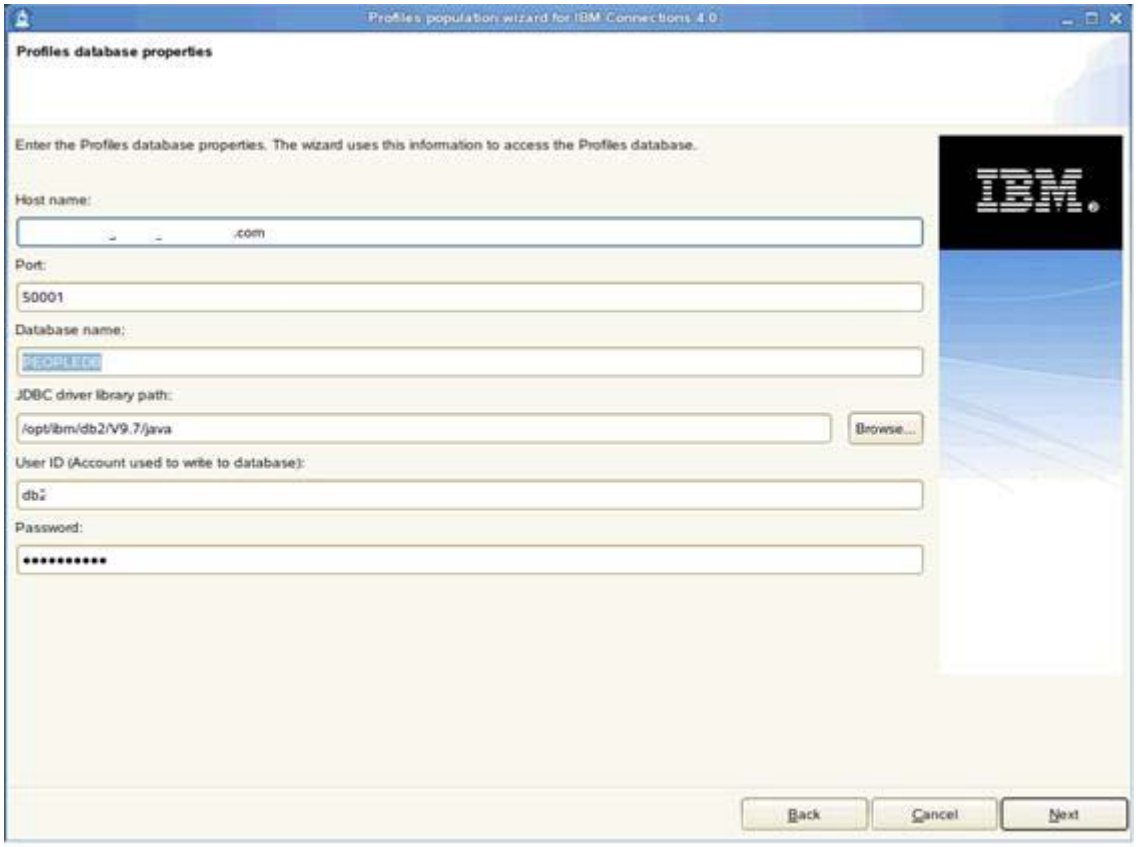


Figure 57. Profile database properties

__ 5. Enter your LDAP server and port and then click **Next** to continue.

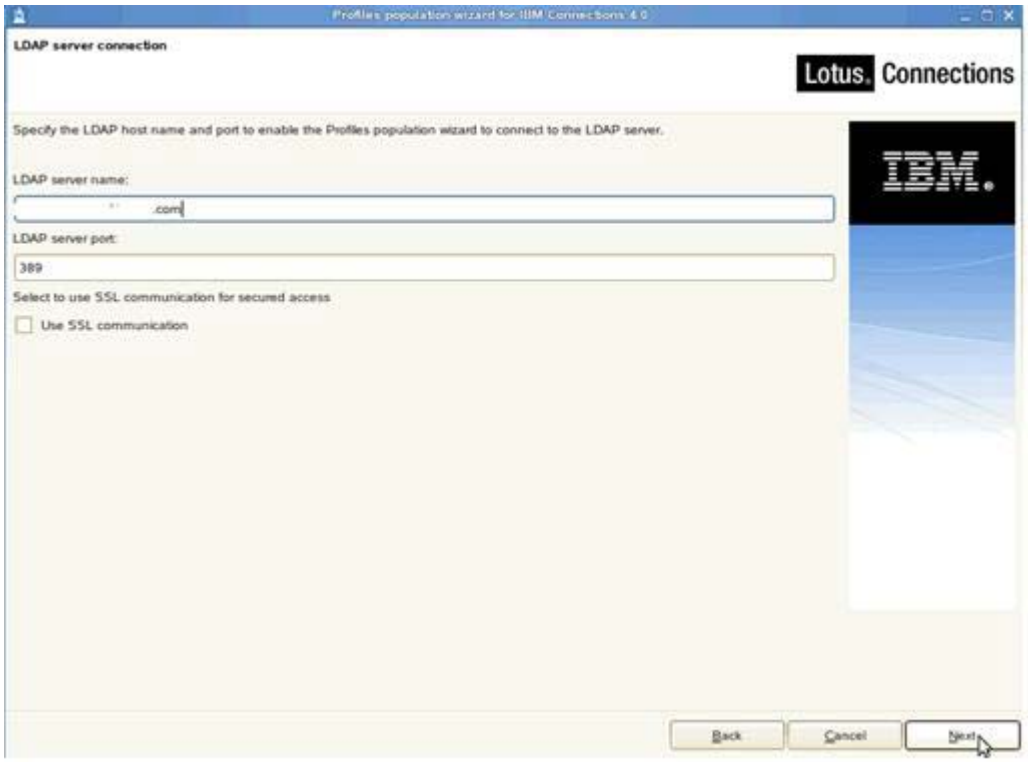


Figure 58. LDAP server connection

__ 6. You are then asked about your bind user and Bind password and select **Next** to continue.

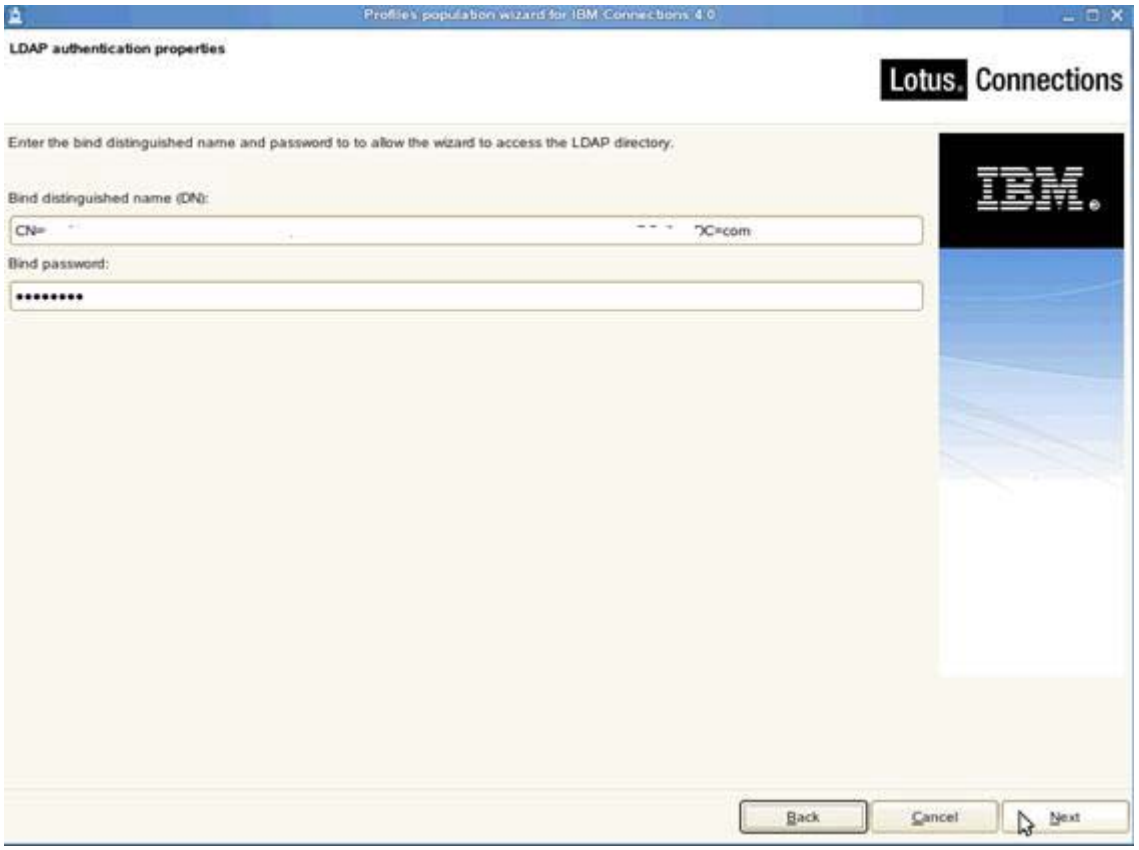


Figure 59. LDAP authentication properties

__ 7. Enter the search base and search filter. Click **Next** to continue.

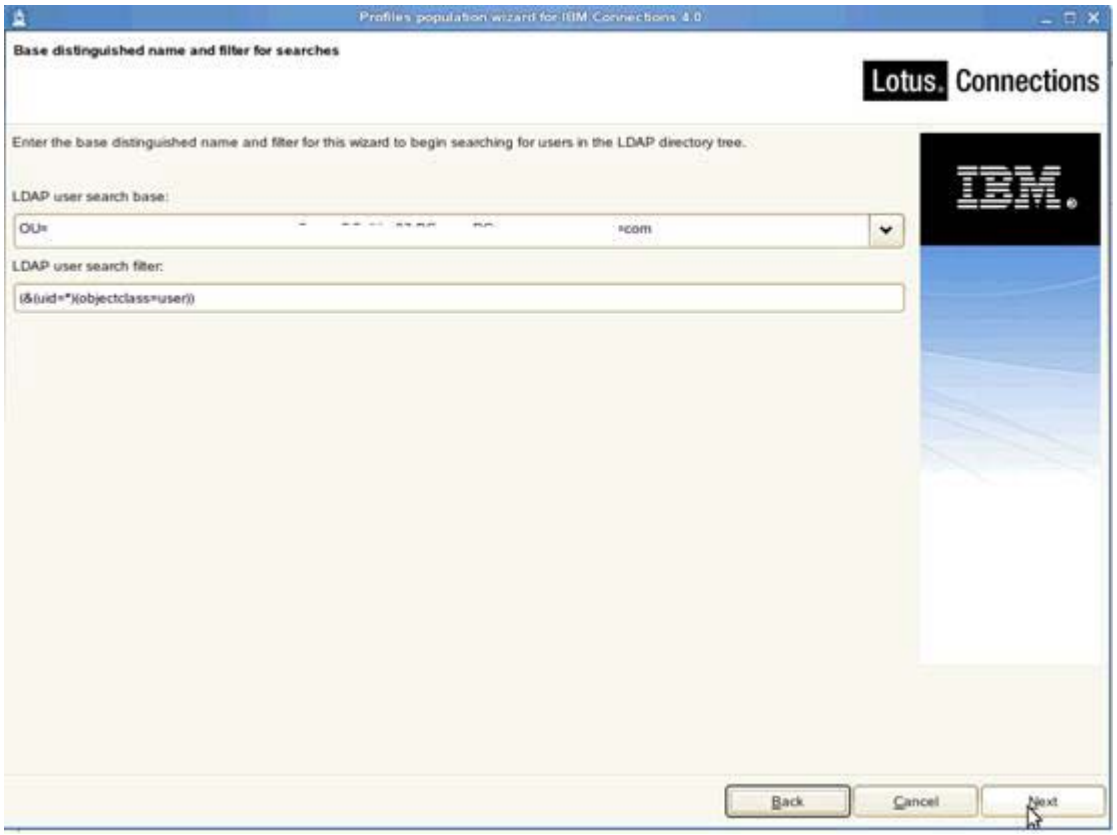


Figure 60. Base distinguished name and filter for searches

__ 8. Select the default database mapping for this example. Click **Next** to continue.

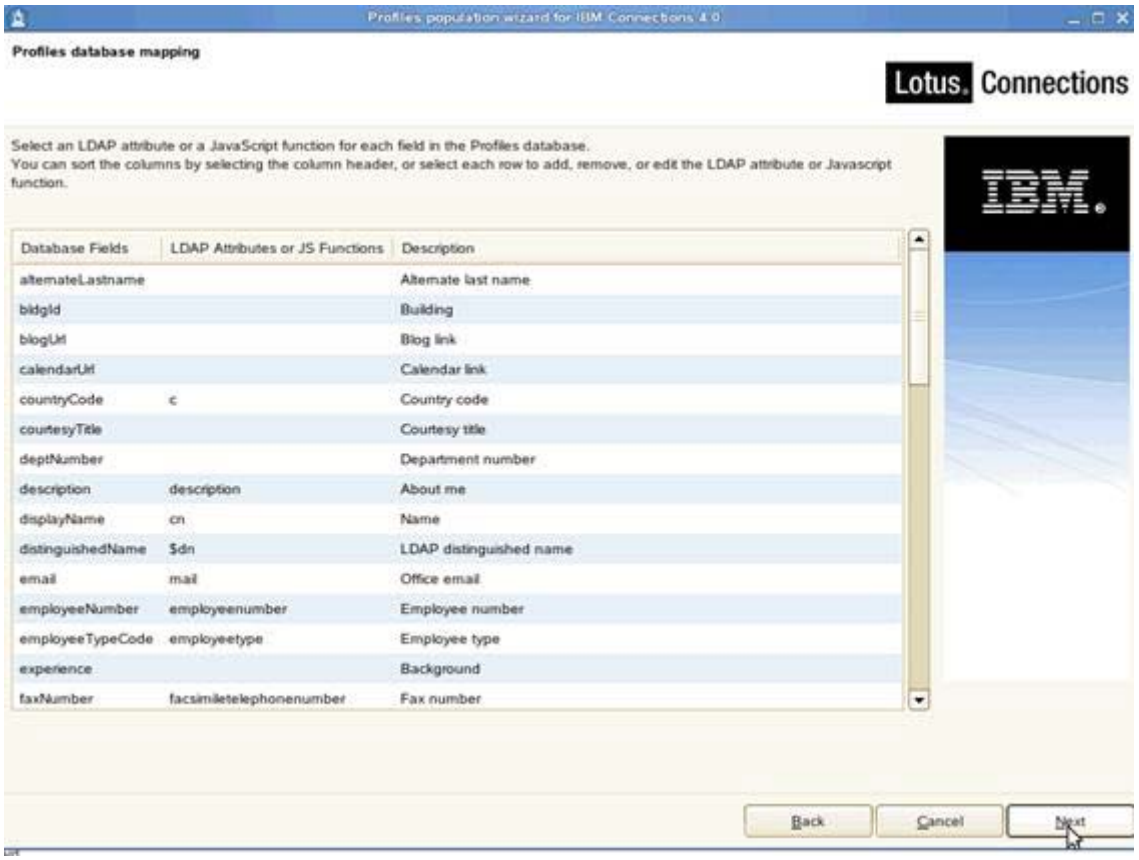


Figure 61. Profiles database mapping

__ 9. Do not select any of the optional database tasks. Then, click **Next** to continue.

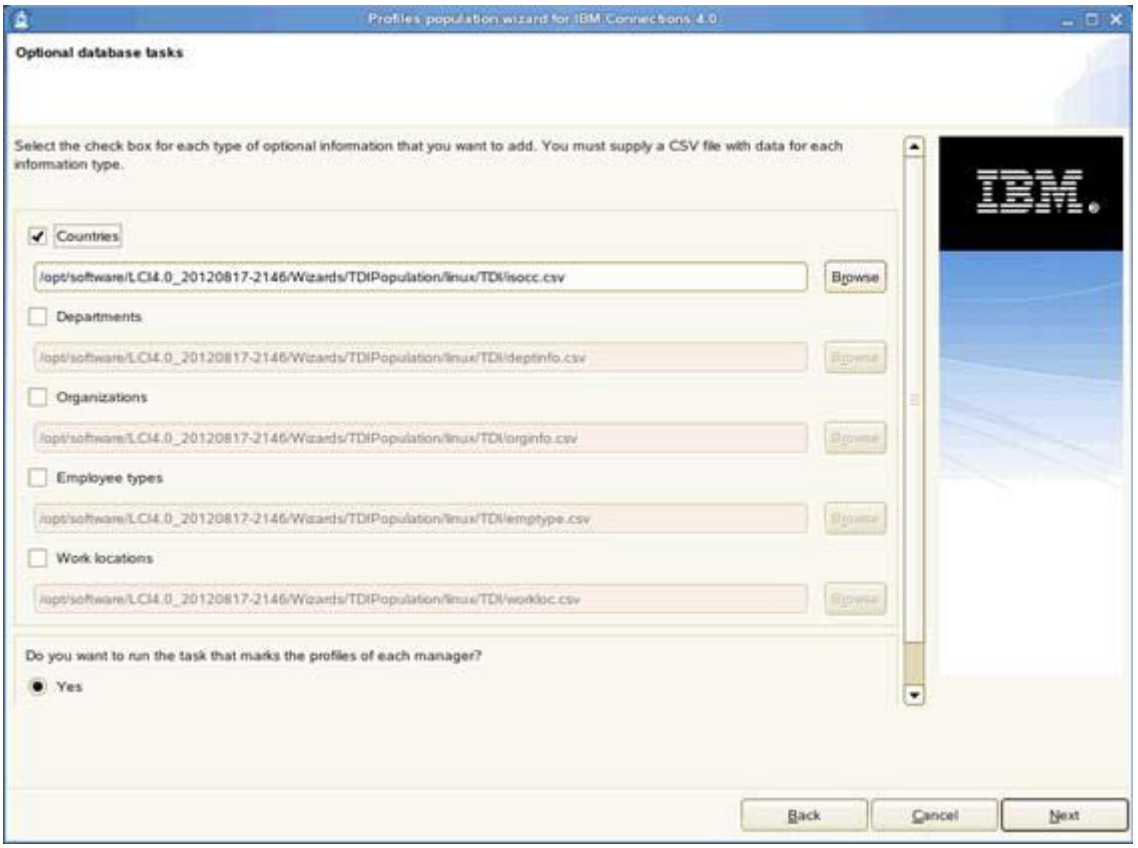


Figure 62. Optional database tasks

__ 10. Review the summary page to ensure that the information you entered in the previous panels is correct. To make changes, click **Back** to return to the relevant page and edit the information. Otherwise, click **Configure** to begin populating the database.

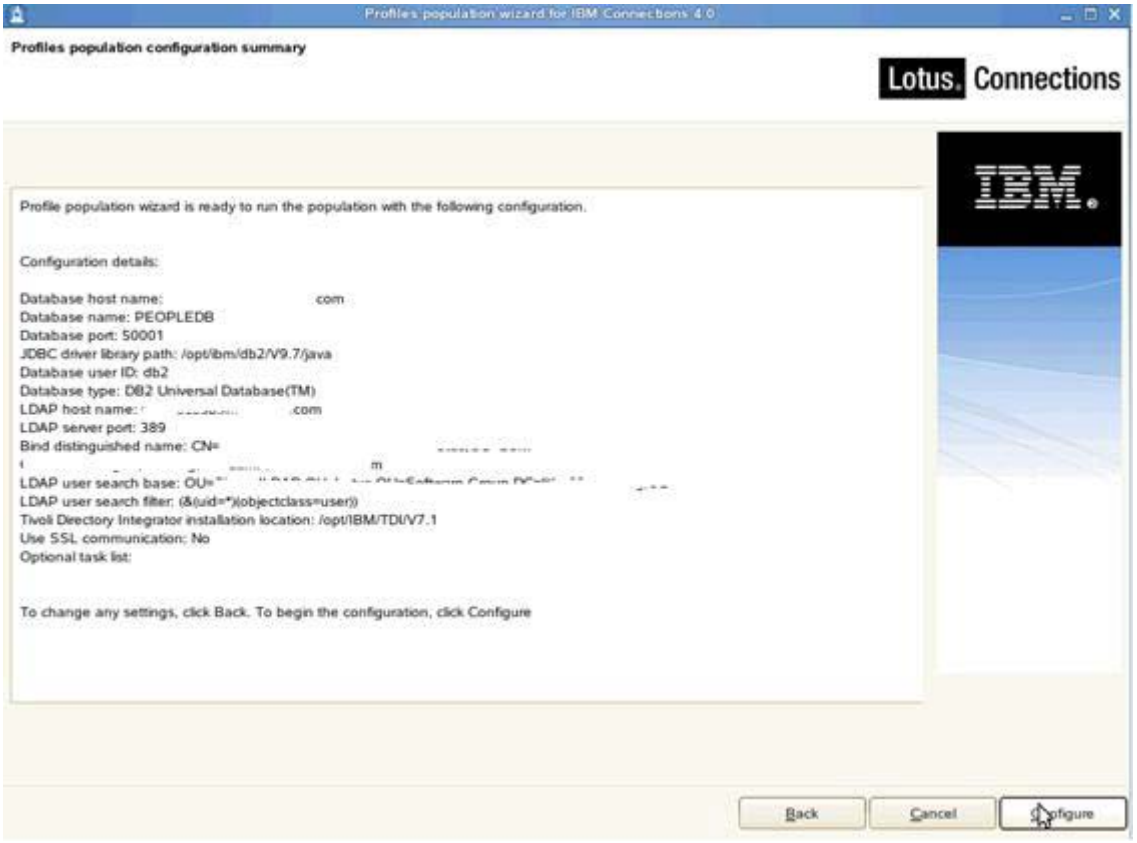


Figure 63. Profiles population configuration summary

Then, you see the execution of the population task:

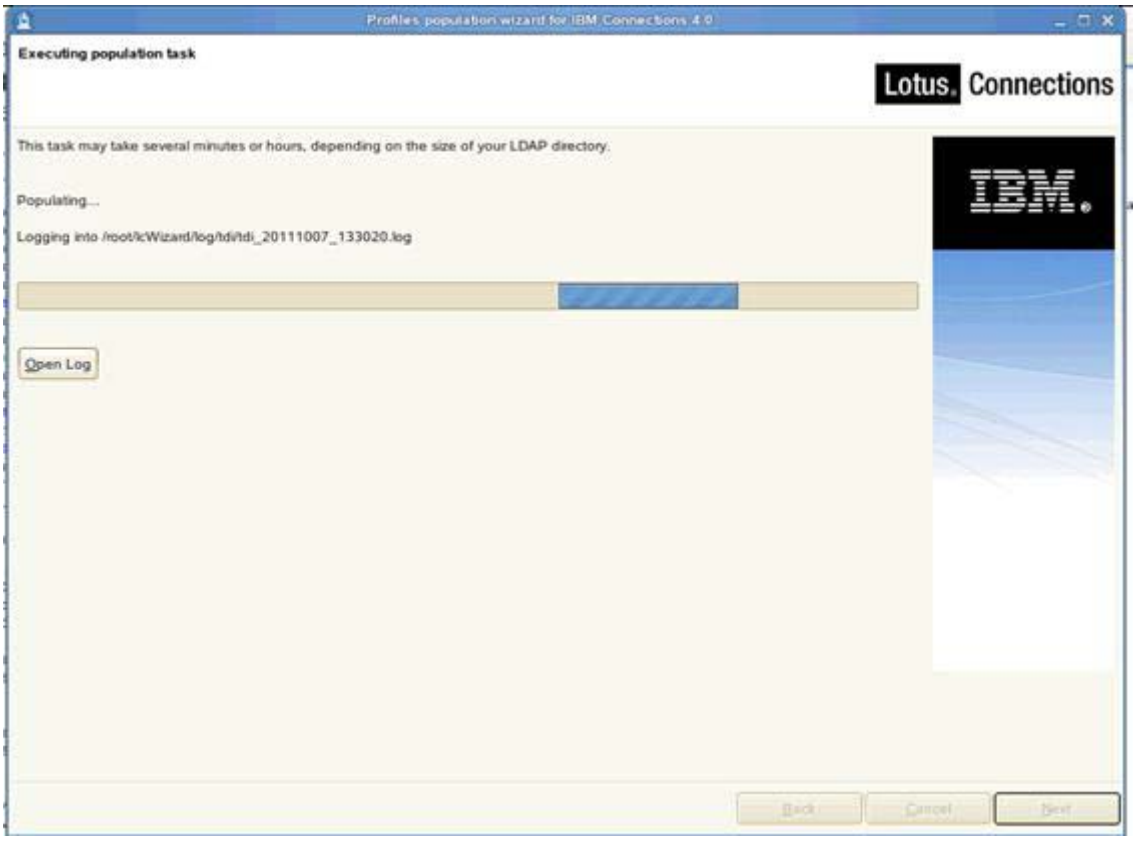


Figure 64. Executing population task

This task can take a long time (3 - 4 hours) so tail the previous log which is being referenced. Click **Finish** to exit the wizard. When the installation completes, you should see the following summary:

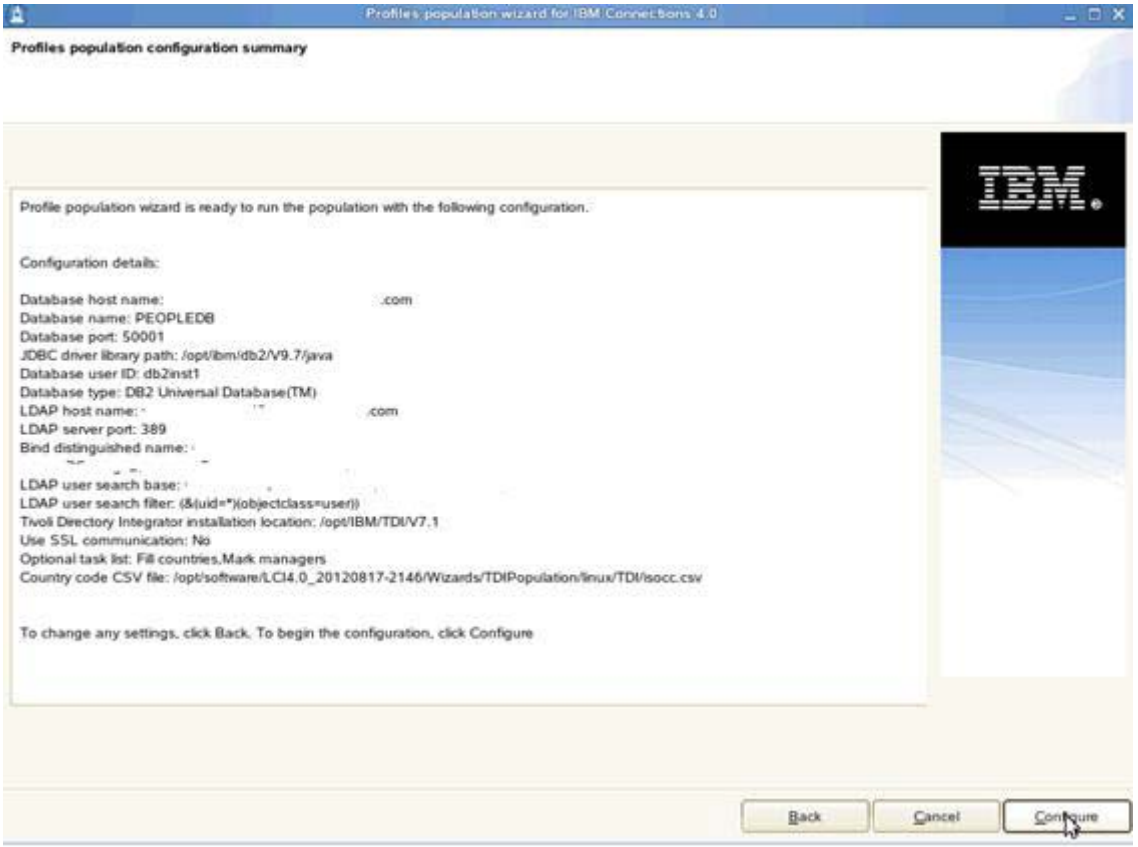


Figure 65. Profiles population configuration summary

__ 11. When the installation completes, click **Finish** to exit the wizard.

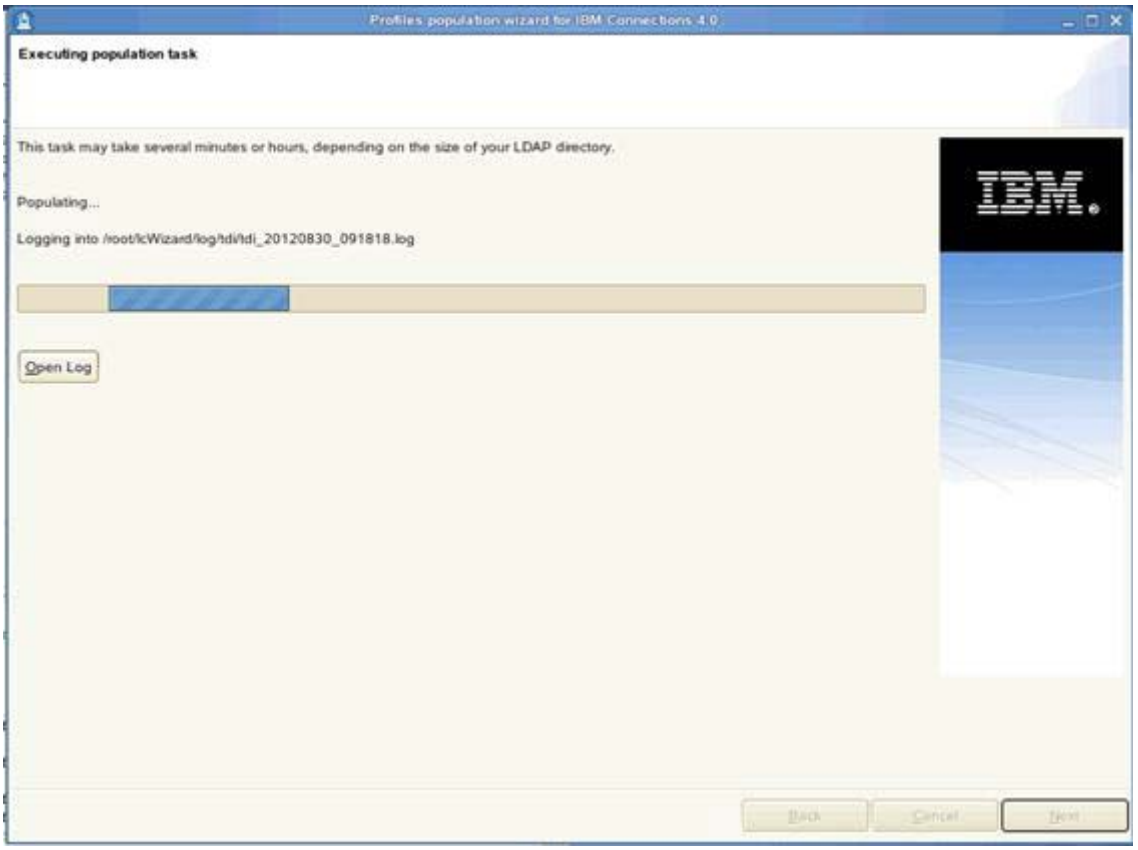


Figure 66. Executing population task

__ 12. When populating, you must check whether the users are in the PROFILES database. To do so, check on the DB2 server check by running:

```
db2 connect to PEOPLEDB
db2 SELECT PROF_UID FROM EMPINST."EMPLOYEE" ORDER BY PROF_UID FETCH FIRST 20
ROWS ONLY
```

__ 13. When Connections is running, run `http://connections.example.com/profiles` in a browser so a search for some users. A list should come back.

__ 14. Finally, start the Connections servers. Check for errors in the **Logs > /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/logs** on each node.

3. Configuring the remote HTTP server

The next section is about HTTP configuration and must be completed as HTTP is required for login by default on Connections.

To start the administration server, go to HTTPServer/bin directory and issue the command `./adminctl` start.

Add web server as unmanaged node

__ 1. After the administration server is started, open the Deployment Manager and add the web server to the cell as an unmanaged node. Open the administrative console at `https://connections.example.com:9043/admin`.

__ 2. Go to **System Administration > Nodes** and click **Add Node**.

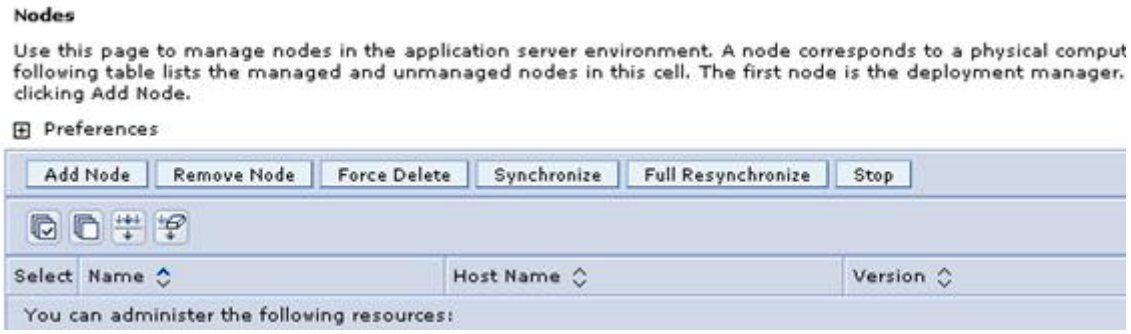


Figure 67. Adding a node

__ 3. Click **Unmanaged node** and click **Next**.

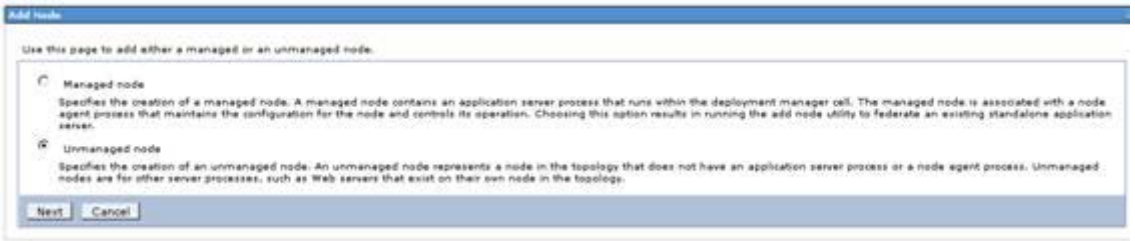


Figure 68. Adding an unmanaged node

__ 4. Provide a name and host name of the HTTP server and click **OK.**

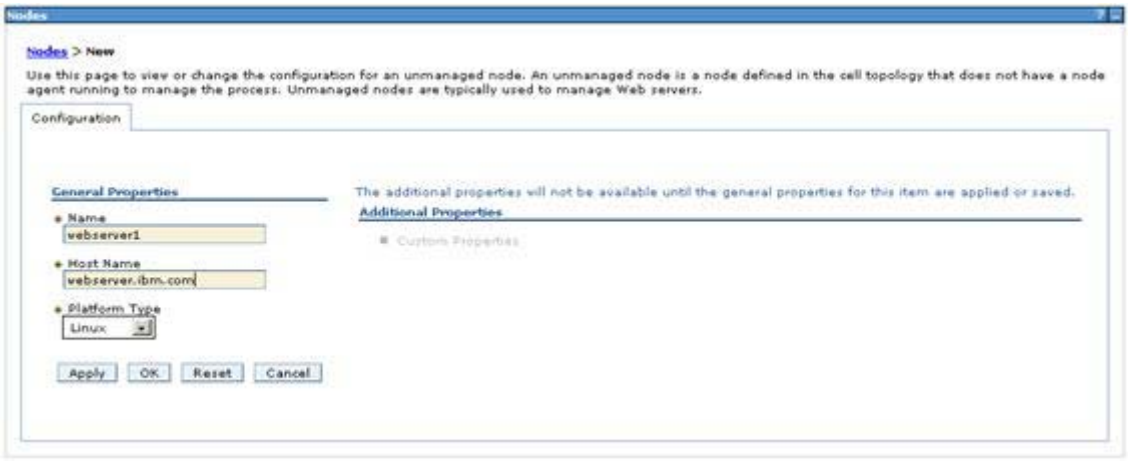


Figure 69. Entering the node's general properties

__ 5. Click **Save**.

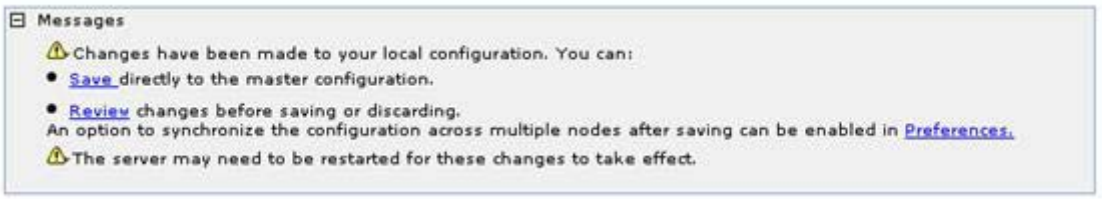


Figure 70. Saving the node's properties

On the nodes panel, the web server is displayed in the list.

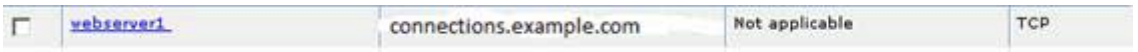


Figure 71. Display of the new web server

Add web server as a server

Next, add the web server as a server in the figuration. To do so, do the following:

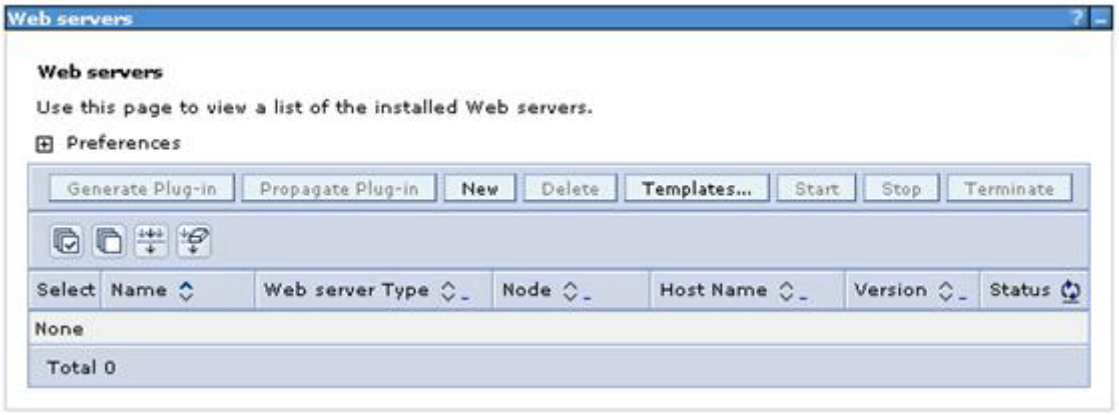


Figure 72. Adding a web server as a server in the figuration

__ 6. From **Servers > Server Types > Web Servers**, click **New**.

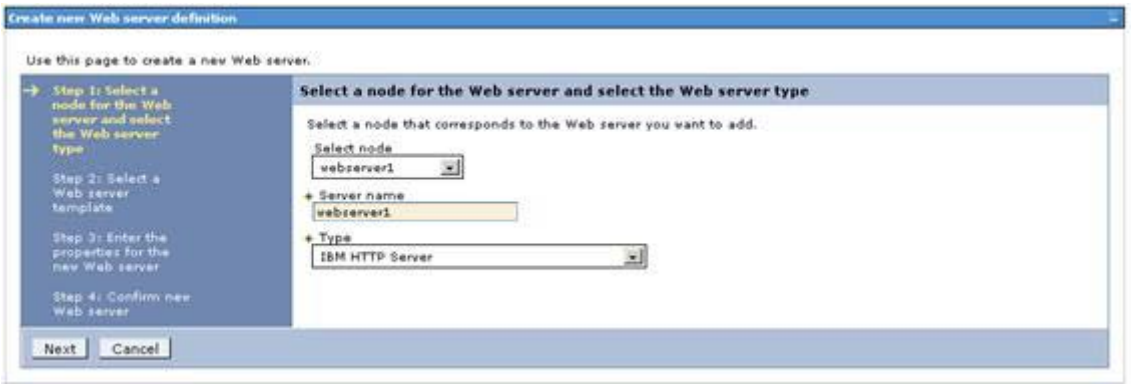


Figure 73. Creating new web server definition

__ 7. Select the web server node and provide the name of this server as `webserver1`. This is the same name that is provided during the plug-ins installation on the web server. Click **Next** to continue.

__ 8. The IHS option is selected, click **Next**.

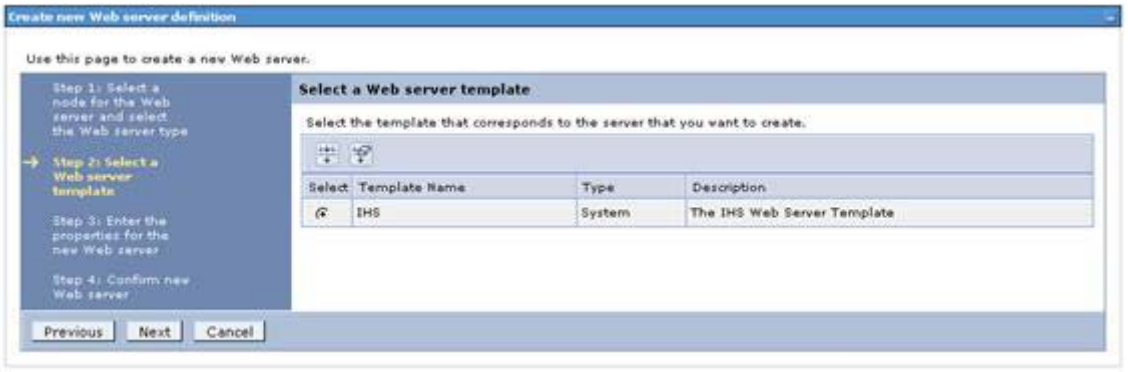


Figure 74. Selecting a Web server template

__ 9. Provide all of the web server details as previously and click **Next**.

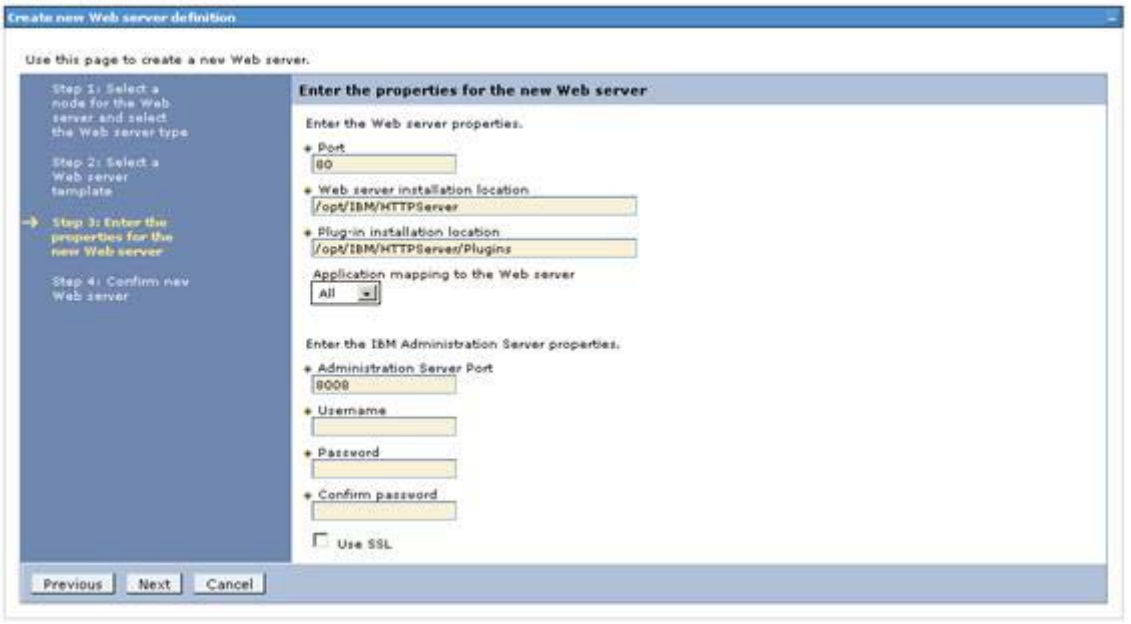


Figure 75. Entering the properties for the new web server

__ 10. Confirm the new web server and click **Finish**.

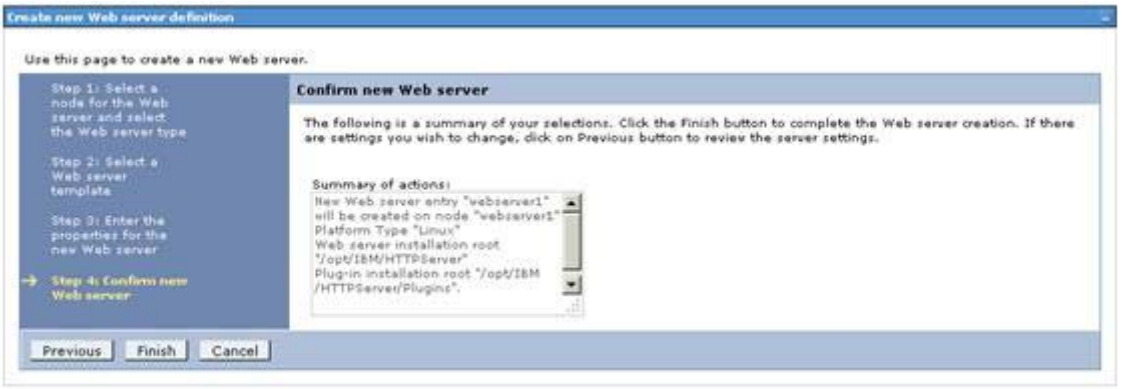


Figure 76. Confirming the new web server

__ 11. Save this change. Before proceeding, do a full synchronization between nodes in the deployment.

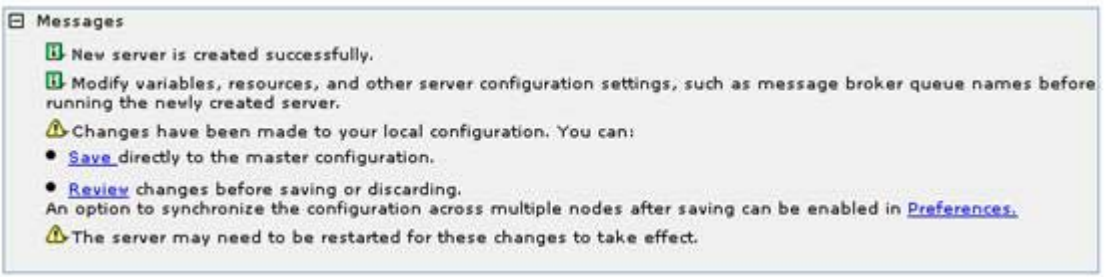


Figure 77. Saving the changes in the new web server

__ 12. Return to **Servers > Server Types > Web Servers**. Generate and propagate the plug-in file to the web server.

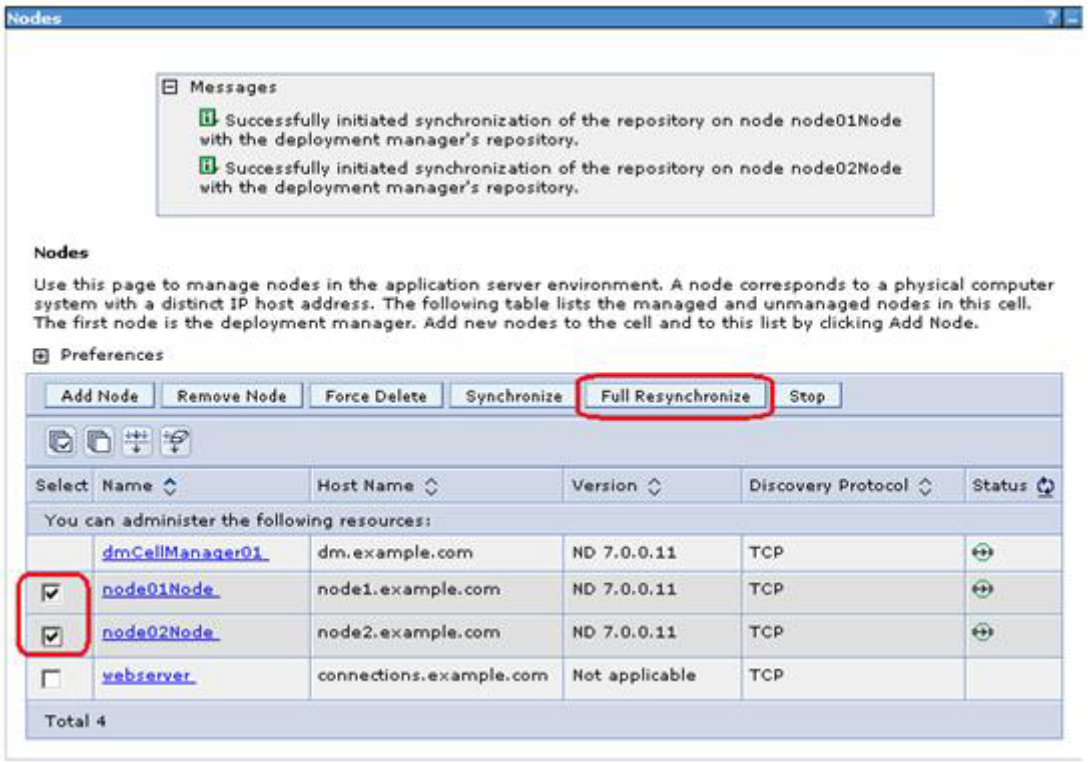


Figure 78. Generating and propagating the plug-in file to the web server

__ 13. To do so, select the check box beside webserver1 and click **Generate Plug-in**.

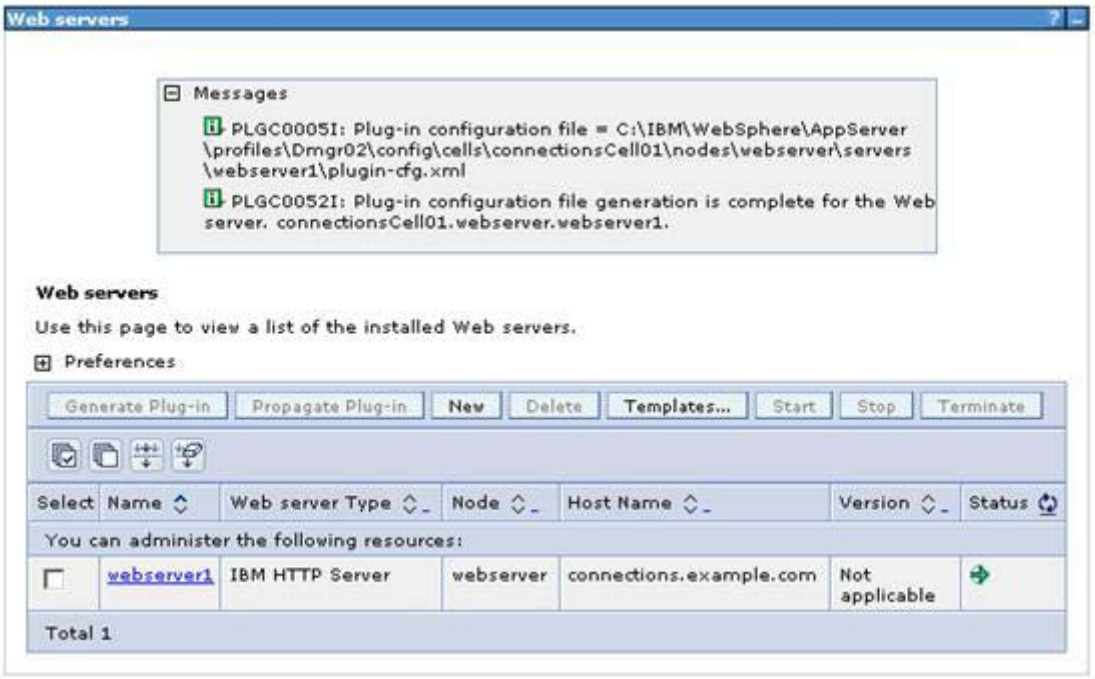


Figure 79. Generating plug-in

__ 14. Select the check box again and click **Propagate Plug-in**.

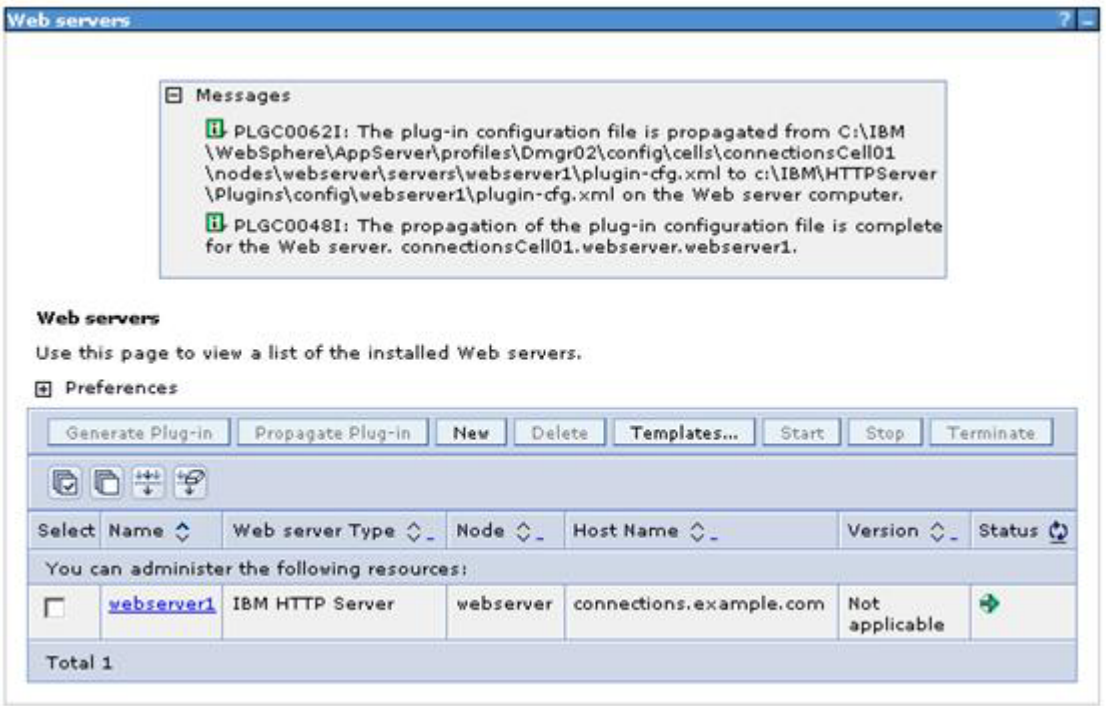


Figure 80. Propagating plug-in

__ 15. Click **webserver1** and click Plug-in properties.

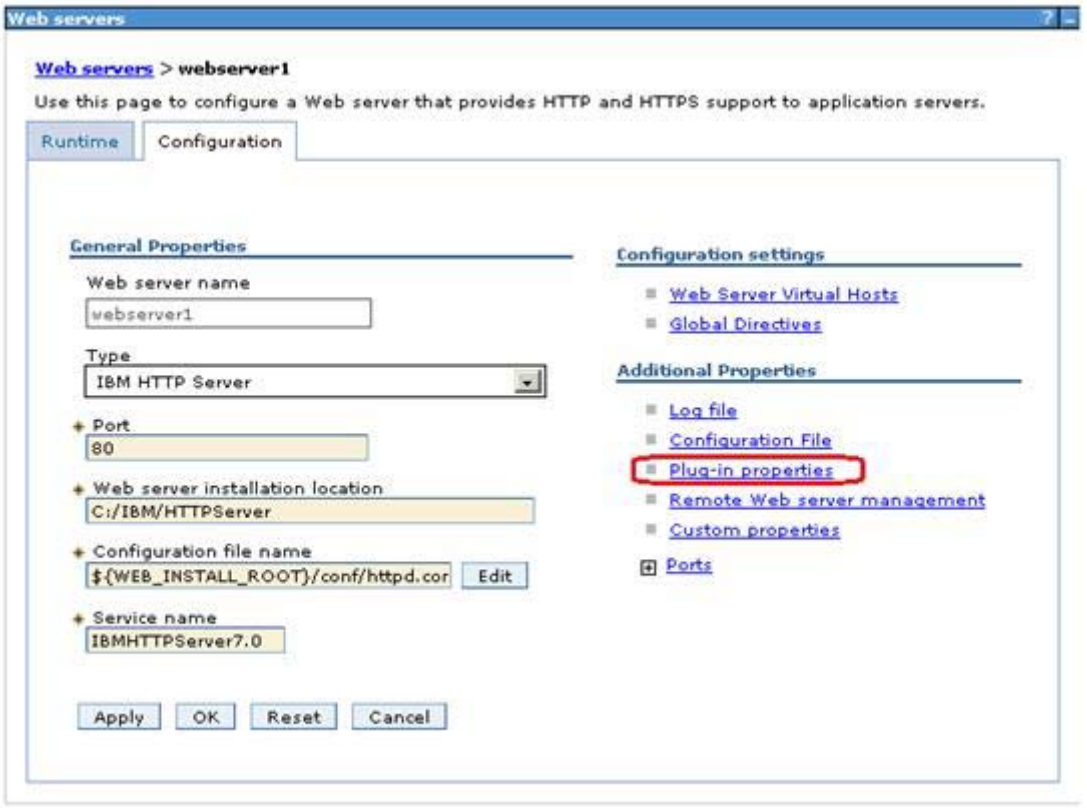


Figure 81. Plug-in properties

__ 16. From the repository copy of web server plug-in files section, click **Copy to Web server key store directory**.

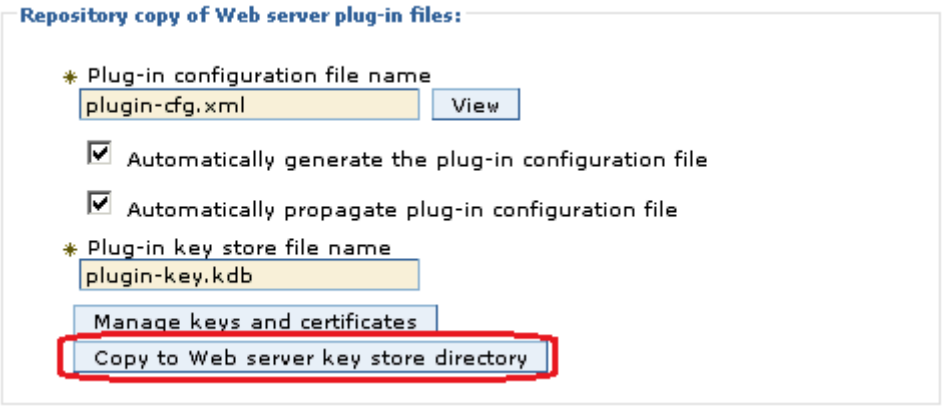


Figure 82. Copying to web server key store directory

__ 17. The following message is displayed to indicate the successful copying of these keys. Again, restart the web server for the plug-in changes to take effect.

Figure 83. Message indicating the successful copying of the keys

Configuring IBM HTTP Server for SSL

To support SSL, create a self-signed certificate and then configure IBM HTTP Server for SSL traffic. If you use this certificate in production, users might receive warning messages from their browsers.

In a typical production deployment, you would use a certificate from a trusted certificate authority.

The first step is to create a key file.

__ 1. Start the iKeyman utility by `ikeyman.sh` from `/opt/IBM/HTTPServer/bin`. The following panel is displayed when you run this utility.

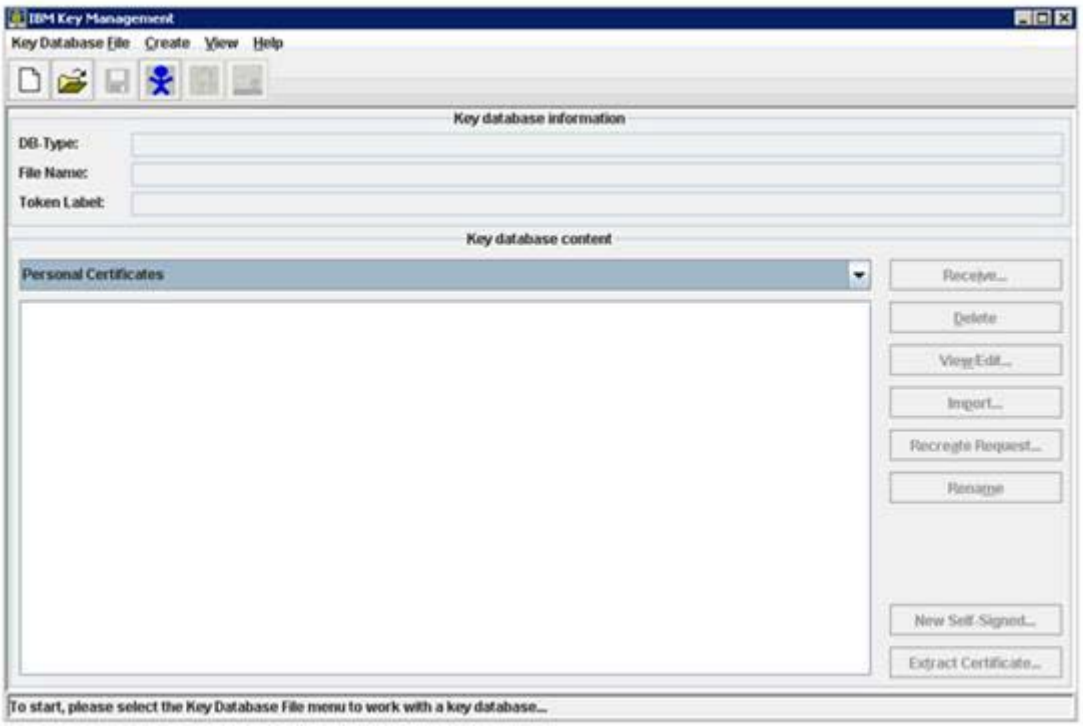


Figure 84. IBM Key Management

__ 2. Click Key **Database File > New...**

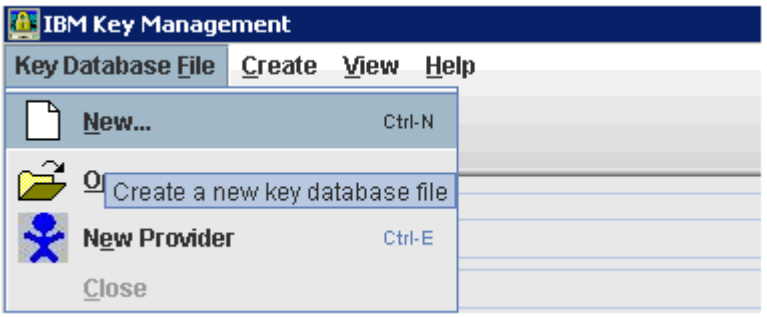


Figure 85. Creating a database file

__ 3. Ensure that the key database type is selected as CMS. Input a name for the key file and location to store it.

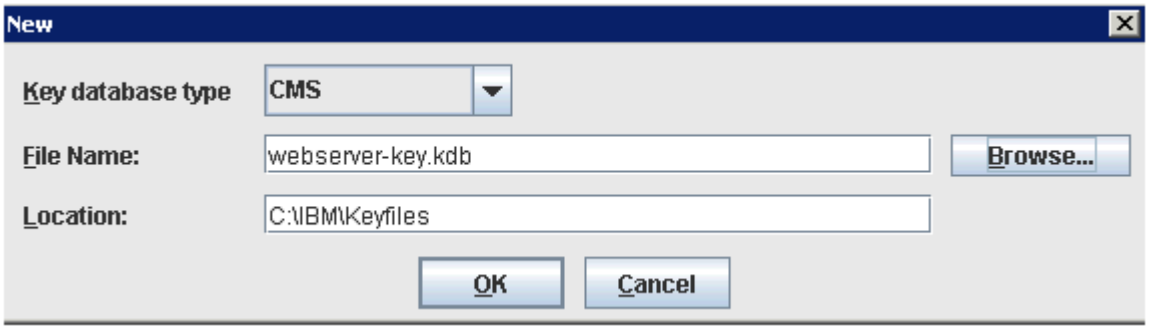


Figure 86. Providing a name and location for the key file

__ 4. Enter a password and check **Stash password to a file**.

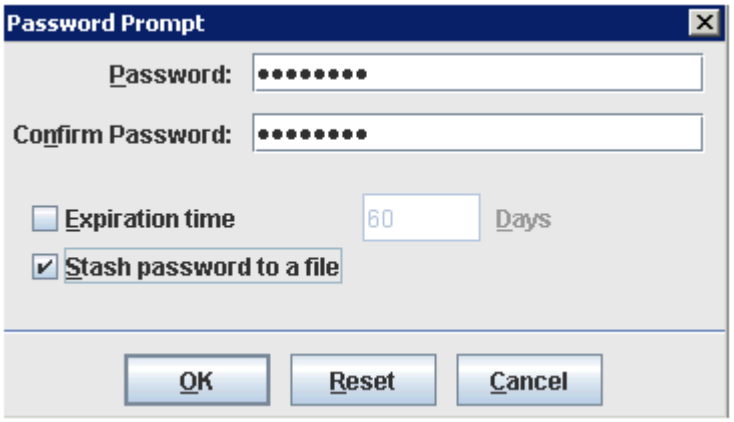


Figure 87. Password prompt

You are returned to the iKeyman panel with the webserver-key.kdb opened.

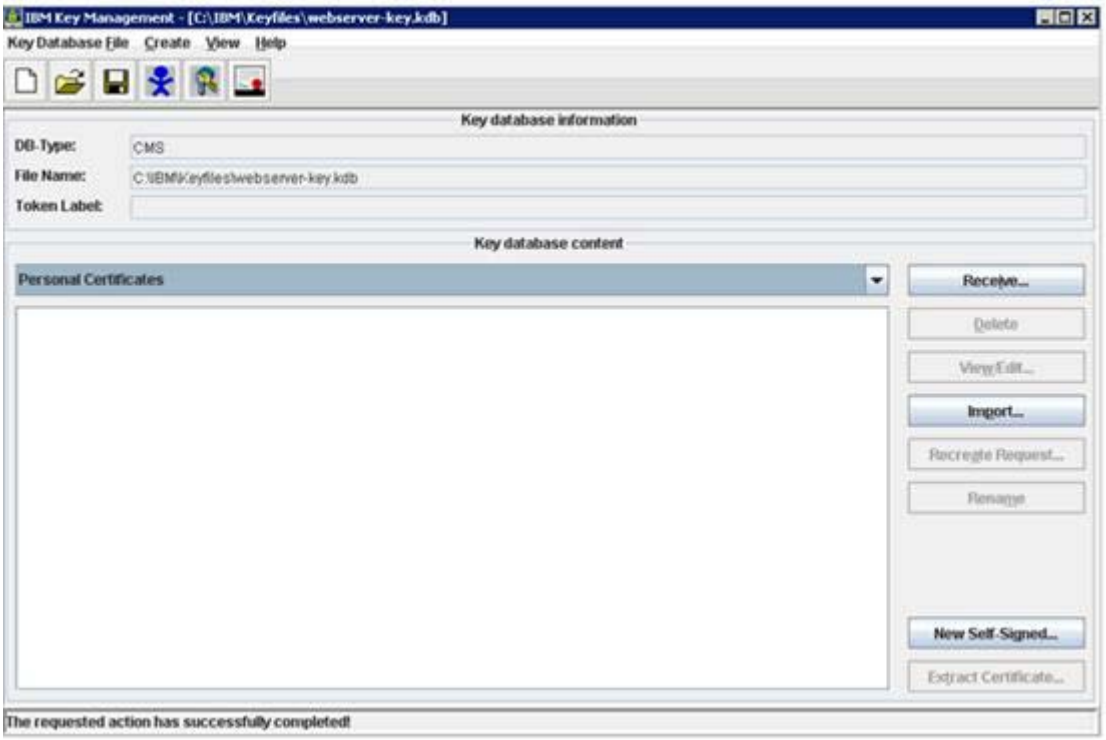


Figure 88. IBM Key Management

__ 5. Create a self-signed certificate by using **Create > New Self-Signed Certificate**.

Figure 89. Creating a New Self-Signed Certificate

__ 6. Input the label and other details as appropriate. Click **OK** to save the certificate.

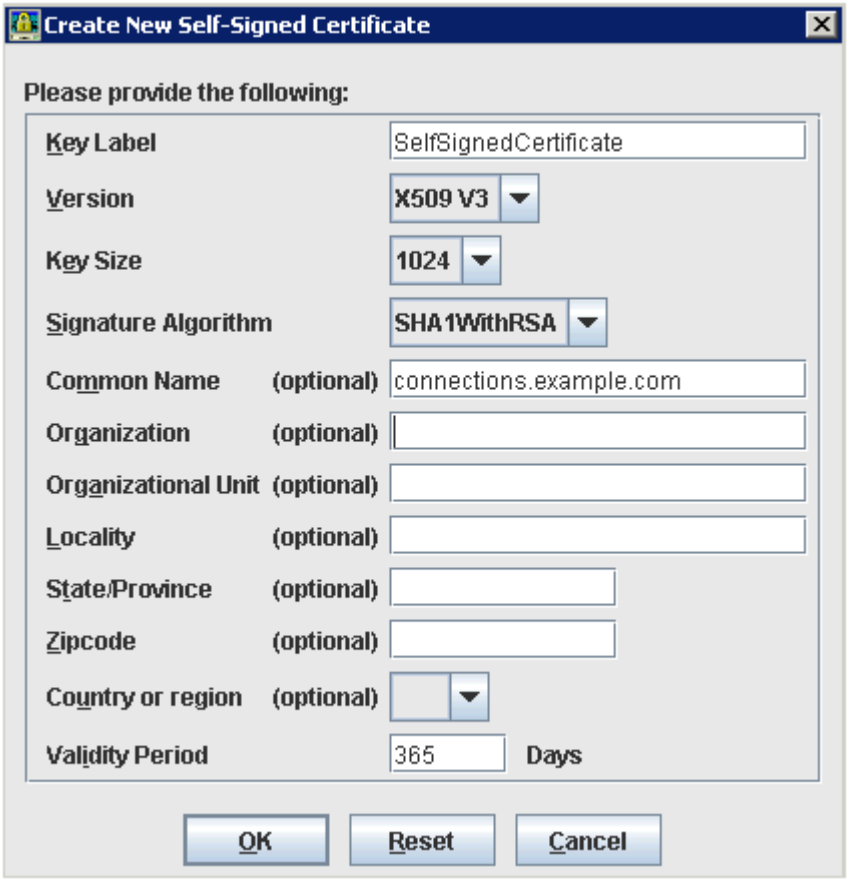


Figure 90. Create New Self-Signed Certificate: Details

The certificate now appears in the key file.

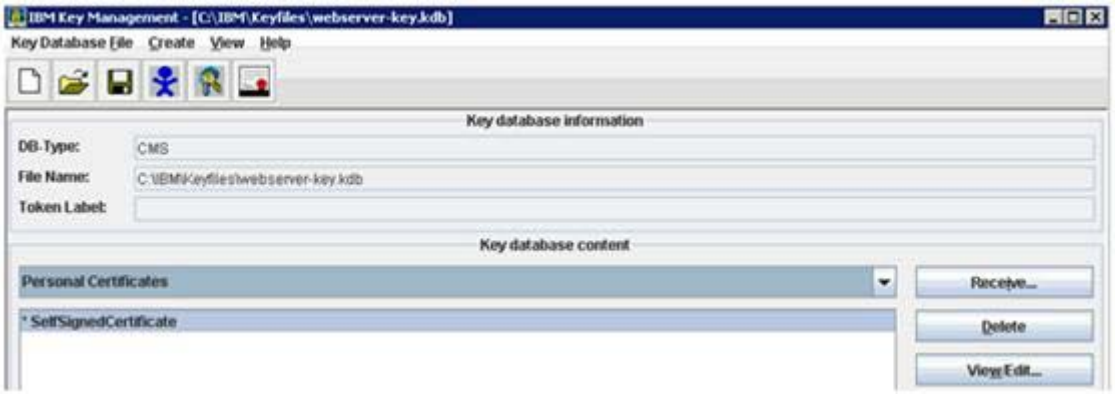


Figure 91. IBM Key Management showing the certificate created

__ 7. Stop the IBM HTTP Server, if started. When verified as stopped, log in to the administrative console and configure the web server for SSL. From the Web servers panel, click **webserver1**.

Figure 92. Configuring the web server for SSL

__ 8. Click **Configuration File** to open the `httpd.conf` from the administrative console.

Figure 93. Configuration File

The `httpd.conf` opens in the browser as shown in the figure.

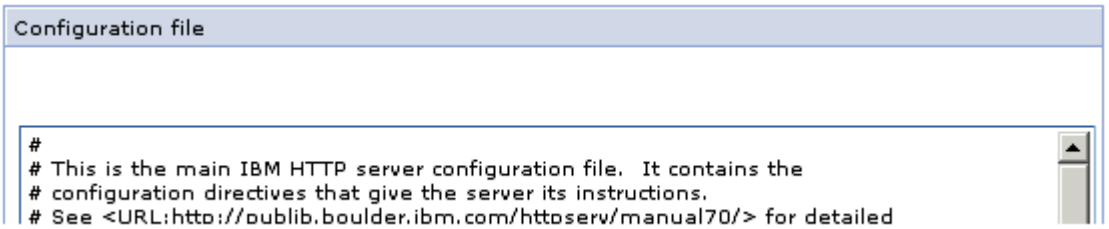


Figure 94. httpd.conf

__ 9. Scroll to the bottom of the configuration file. At the end of the `httpd.conf`, add the previous lines to load the SSL module by using the newly created key file:

```
LoadModule ibm_ssl_module modules/mod_ibm_ssl.so
<IfModule mod_ibm_ssl.c>
Listen 0.0.0.0:443
<VirtualHost *:443>
ServerName connections.example.com
SSLEnable
AllowEncodedSlashes On
</VirtualHost>
</IfModule>
SSLDisable
Keyfile "/opt/IBM/Keyfiles/webserver-key.kdb"
SSLStashFile "/opt/IBM/Keyfiles/webserver-key.sth"
```

__ 10. Click **OK** to save this change.

__ 11. Next, start the IBM HTTP Server. To verify that the SSL settings took effect correctly, type `https://connections.example.com` into a browser. If the IBM HTTP Server page appears over https, then this step was successful. You might need to accept the certificate to your browser as it is not signed.

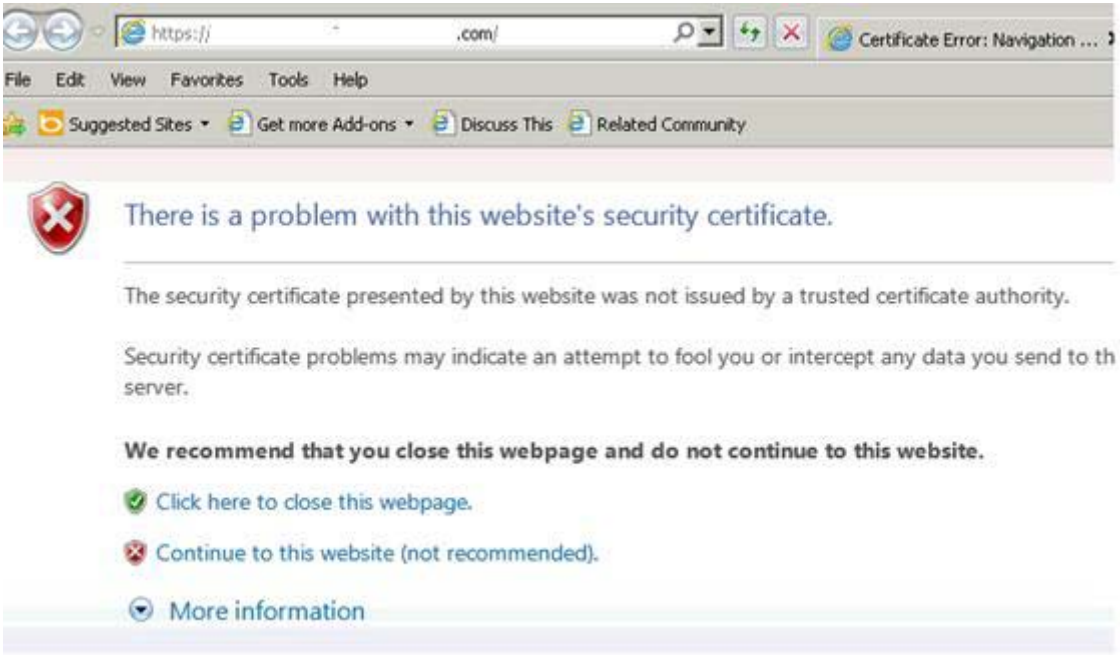


Figure 95. Website's security certificate

__ 12. Click **Continue to this website (not recommended)**. The WebSphere software start page is displayed.

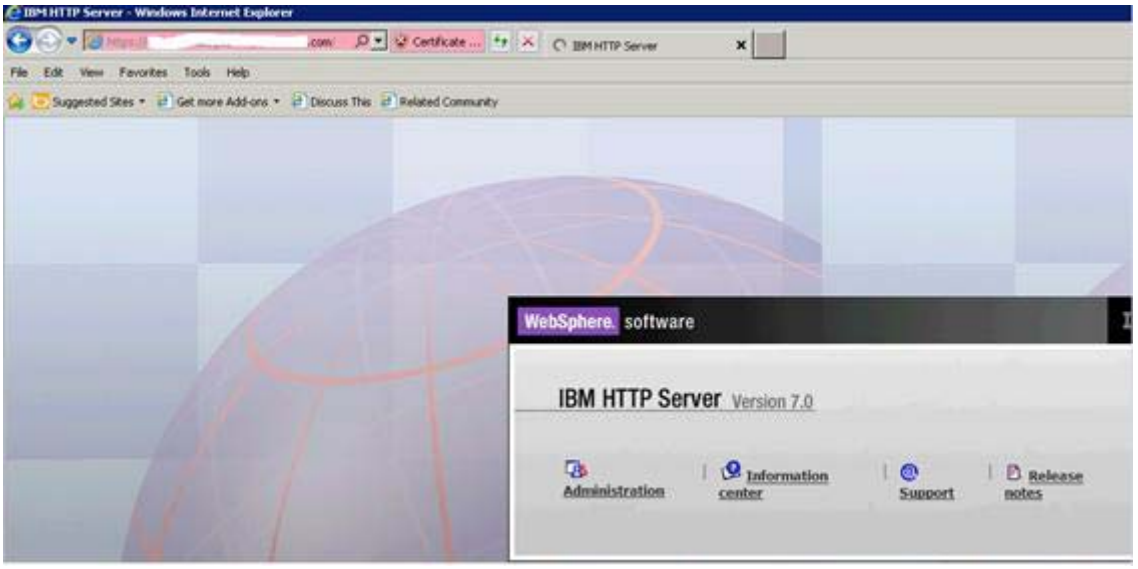


Figure 96. WebSphere software start page

Adding certificates to the WebSphere truststore

__ 1. On the administrative console go to **Security > SSL Certificate and Key Management**. Click **CellDefaultTrustStore** as shown in the figure.

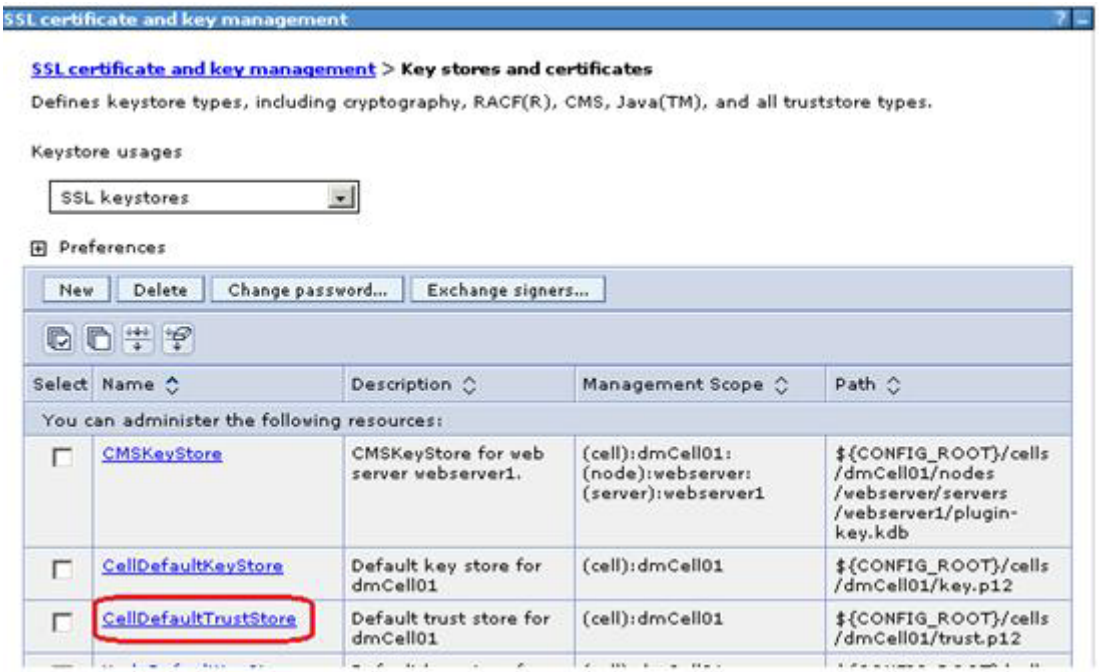


Figure 97. SSL certificate and key management

__ 2. From within CellDefaultTrustStore, click **Signer Certificates** from the right side.

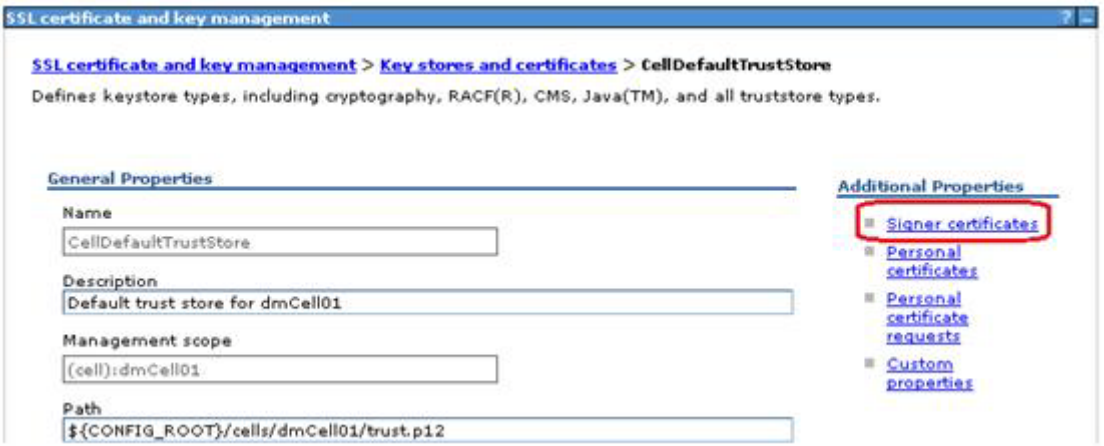


Figure 98. Additional Properties > Signer certificates

__ 3. To add the webservers signer to the truststore, click **Retrieve from Port**.

Figure 99. Retrieving from port

__ 4. Enter the host name of the web server and its SSL port (typically 443). Then, click **Retrieve Signer Information**, which retrieves the information that is shown at the bottom of the screen capture. Provide an alias for this signer certificate and click **OK** to add this certificate to the list of signers.

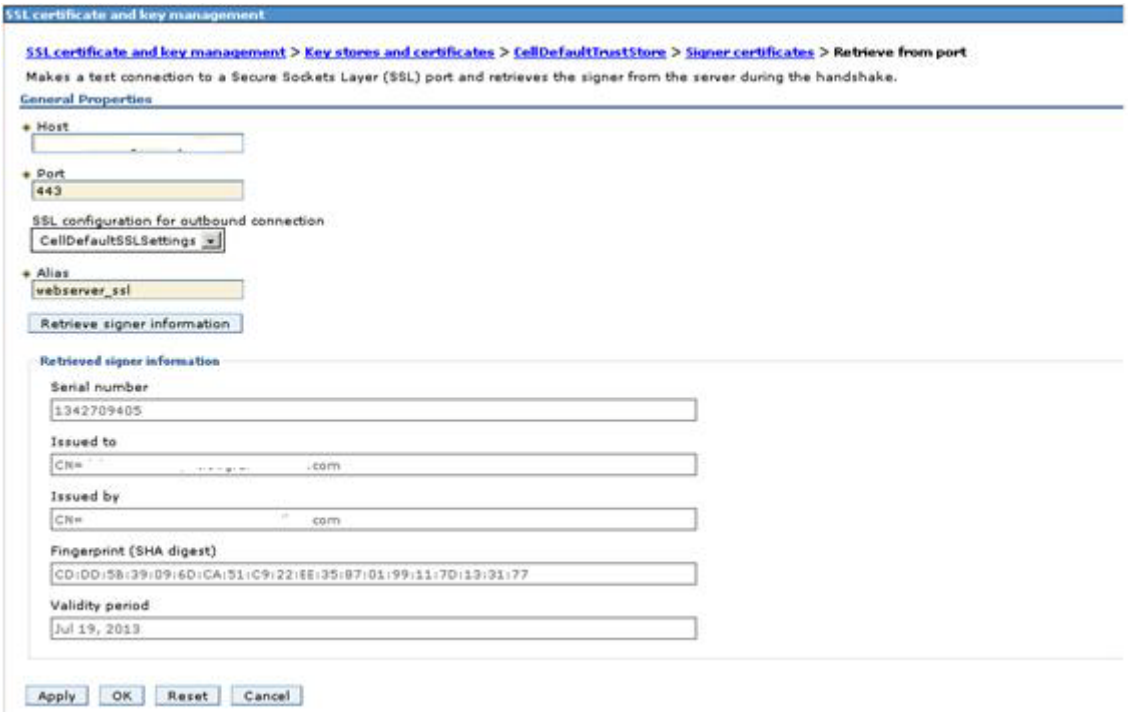


Figure 100. Retrieved signer information

__ 5. Save this change and restart the HTTP server to apply the changes.

Update web addresses used by Lotus Connections to access content

__ 1. Using the wsadmin client, check out the `LotusConnections-config.xml` to a temporary directory. From this directory, this file must be edited so that all `href` and `ssl_href` values are updated to reflect the host name of the HTTP Server and do not include any port numbers. An example is as follows:

```
<sloc:serviceReference enabled="true" serviceName="bookmarklet" ssl_enabled="true">
    <sloc:href>
        <sloc:hrefPathPrefix>/connections/bookmarklet</sloc:hrefPathPrefix>
        <sloc:static href="http://connections.example.com:9081" ssl_href="https://connections.example.com:9444"/>
        <sloc:interService href="https://connections.example.com:9444"/>
    </sloc:href>
</sloc:serviceReference>
```

Figure 101. LotusConnections-config.xml

__ 2. Convert the following original values of the `hrefs ssl_hrefs` from their previous default values to their new values. In this case, all that is done is to drop the port numbers 9081 and 9044 from these URLs.

```
<sloc:serviceReference enabled="true" serviceName="bookmarklet" ssl_enabled="true">
    <sloc:href>
        <sloc:hrefPathPrefix>/connections/bookmarklet</sloc:hrefPathPrefix>
        <sloc:static href="http://connections.example.com" ssl_href="https://connections.example.com"/>
        <sloc:interService href="https://connections.example.com"/>
    </sloc:href>
</sloc:serviceReference>
```

Figure 102. LotusConnections-config.xml

__ 3. Repeat this process for all `href` and `ssl_hrefs` that are currently set to `connections.example.com`.

Note

For the metrics and Cognos entries:

The metrics interlock settings change as per all the other applications, the Cognos settings keep the port numbers. This is because there are the port numbers to point to the BI Cognos server.

```
<sloc:serviceReference bootstrapHost="" bootstrapPort="" clusterName="Cluster1" enabled="true" serviceName="metrics" ssl_enabled="true">
    <sloc:href>
        <sloc:hrefPathPrefix>/metrics</sloc:hrefPathPrefix>
        <sloc:static href="http://connections.example.com" ssl_href="https://connections.example.com"/>
        <sloc:interService href="https://connections.example.com"/>
    </sloc:href>
</sloc:serviceReference>

<sloc:serviceReference bootstrapHost="" bootstrapPort="" clusterName="admin_replace" enabled="true" serviceName="cognos" ssl_enabled="true">
    <sloc:href>
        <sloc:hrefPathPrefix>/cognos</sloc:hrefPathPrefix>
        <sloc:static href="http://node1.example.com:9082" ssl_href="https://node1.example.com:9445"/>
        <sloc:interService href="https://node1.example.com:9445"/>
    </sloc:href>
</sloc:serviceReference>
```

Figure 103. LotusConnections-config.xml

__ 4. After this process is complete, save the file and check the file back in using the wsadmin client. After the file is checked back in, resynchronize the node so that this change is pushed out.

This completes the web server, SSL, and certificate configuration for this scenario. Now, when the application is started it can be accessed at `https://connections.example.com/<component`, where <component represents any of the Connections applications. The commands to do all of the above are shown below (the above updates take place after the check out command):

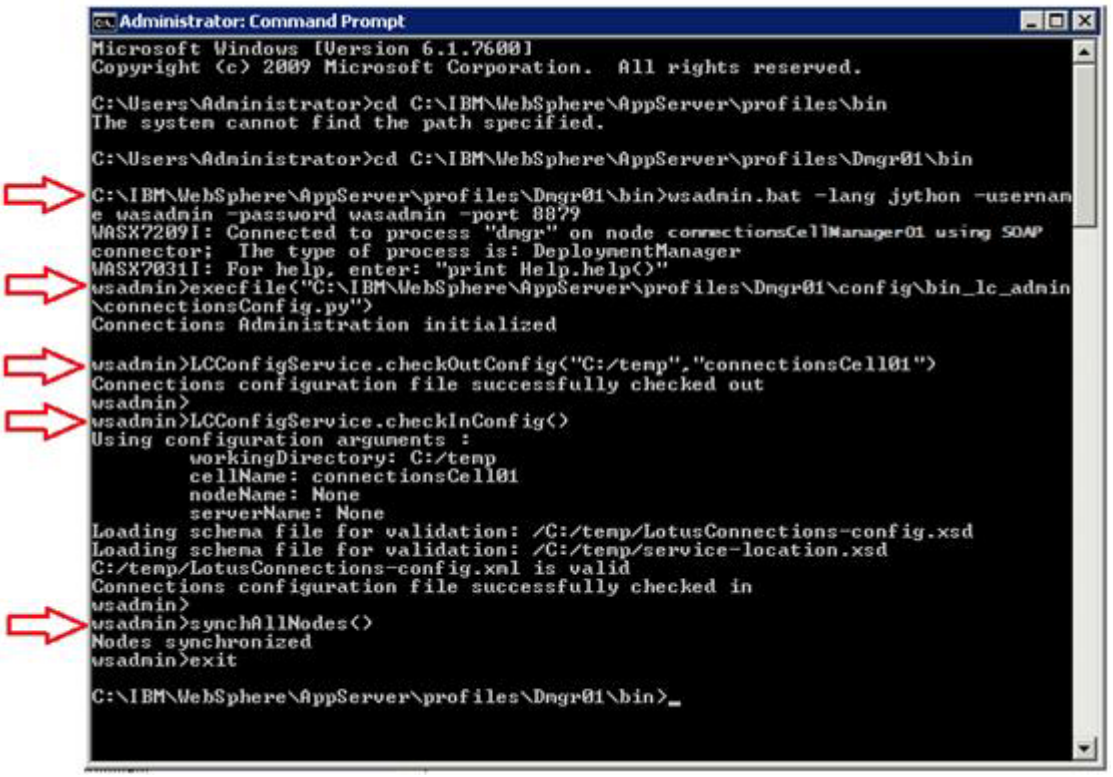


Figure 104. Administrator: Command Prompt

The following list provides the previous commands in a text format so that they can be copied and used again in your own deployment:

```
1: wsadmin.bat -lang jython -username wasadmin -password wasadmin -port 8879
2:
execfile("C:\IBM\WebSphere\AppServer\profiles\Dmgr01\config\bin_lc_admin\connec
tionsConfig.py")
3. LCConfiguService.checkOutConfig("C:/temp","connectionsCell01")
<Make changes to the checked out file>
4: LCConfigService.checkInConfig()
5: synchAllNodes()
```


Add users or all authenticated in application realm to metrics application

Now you add users who can generate metrics.

__ 1. Log in to your admin console and select **Applications > Application Types > Websphere enterprise applications** and then click **Metrics**.

__ 2. Then, select **Security role to user/group mapping** and add the users to the admin and metrics-report-run roles.

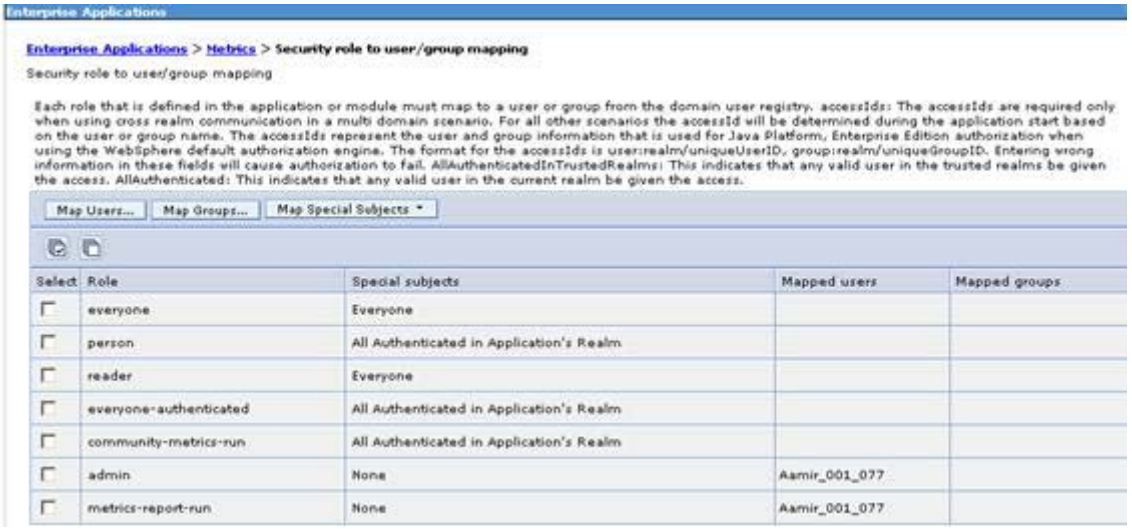


Figure 105. Security role to user/group mapping

__ 3. Save the application and synch the nodes.

Enabling fast downloads for files and wikis

The last item is to enable fast download for files and wikis.

__ 1. On your deployment manager, go to `/opt/IBM/Connections/plugins/ihs/mod_ibm_local_redirect/linux_ia32-ap22`. You see a file that is called `mod_ibm_local_redirect.so` located there.

__ 2. Copy this file to your HTTP server under `/opt/IBM/HTTPServer/modules/`.

Reminder

You must remember to do this; otherwise when you download files the file size is 0.

__ 3. Now edit the `httpd.conf` under `/opt/IBM/HTTPServer/conf`:

`LoadModule ibm_local_redirect_module modules/mod_ibm_local_redirect.so`

`LoadModule env_module modules/mod_env.so` (it might already exit to check your existing file).

__ 4. Also, add the following sections. Paths must change based on installation.

```
Alias /downloadfiles /opt/IBM/LC_Share/files/upload/
Alias /downloadwikis /opt/IBM/LC_Share/wikis/upload/
<Directory /opt/IBM/LC_Share/files/upload/>
Order Deny,Allow
Deny from all
Allow from env=REDIRECT_FILES_CONTENT
</Directory>
<Directory /opt/IBM/LC_Share/wikis/upload/>
Order Deny,Allow
Deny from all
Allow from env=REDIRECT_WIKIS_CONTENT
</Directory>
<Location /files>
IBMLocalRedirect On
IBMLocalRedirectKeepHeaders
X-LConn-Auth,Cache-Control,Content-Type,Content-Disposition,Last-Modified,ET
ag,Content-Language,Set-Cookie
SetEnv FILES_CONTENT true
</Location>
<Location /wikis>
IBMLocalRedirect On
IBMLocalRedirectKeepHeadErs
X-LConn-Auth,Cache-Control,Content-Type,Content-Disposition,Last-Modified,ET
ag,Content-Language,Set-Cookie
SetEnv WIKIS_CONTENT true
</Location>
```


__ 5. Finally, edit the `files-config.xml` and `wikis-config.xml` files under `/opt/IBM/WebSphere/DeploymentManager/profiles/Dmgr01/config/cells/connectionsCell01/LotusConnections-config/` on your deployment manager and change:

```
<download>
<modIBMLocalRedirect enabled="true"
hrefPathPrefix="/downloadfiles" />
<stats>
```

and:

```
<download>
<modIBMLocalRedirect enabled="true"
hrefPathPrefix="/downloadwikis" />
<stats>
```

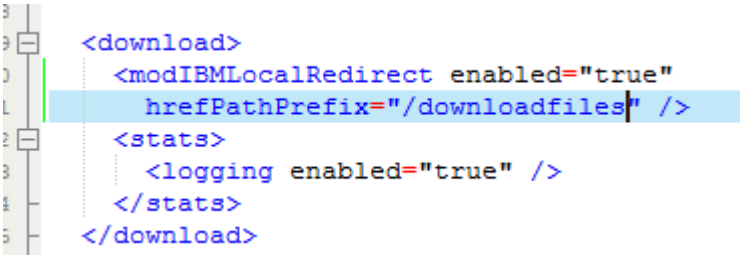


Figure 106. files-config.xml

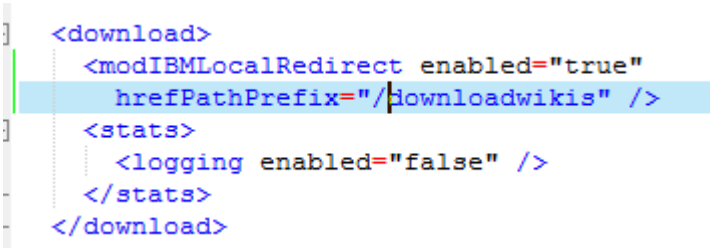


Figure 107. wikis-config.xml

__ 6. When changed, make sure to synch the changes to your nodes.

__ 7. Restart HTTP server and Connections cluster servers.

Verification checkpoint

__ 1. Restart the configuration. Verify that you can log on to the home page and you can access all applications as an admin and as a non-admin user: and do things: create a community, blog, wiki, forum, upload files, and so on.

```
https://connections.example.com/activities
https://connections.example.com/communities
https://connections.example.com/forums
https://connections.example.com/profiles
https://connections.example.com/blogs
https://connections.example.com/dogear
https://connections.example.com/files
https://connections.example.com/wikis
https://connections.example.com/homepage
https://connections.example.com/mobile
https://connections.example.com/search
https://connections.example.com/news
```


4. SiteMinder setup

Important

Be sure to use the installation directories in the screen captures and not the default folders.

This section explains how to enable Computer Associates SiteMinder with an already deployed Connections 4 system. Before beginning the SiteMinder installation and enablement work, make sure that the following prerequisites are completed:

- Lotus Connections 4.0 is set up and working with the IBM HTTP Server without issue.
- The J2C Authentication Alias `connectionsAdmin` is a user who exists on the LDAP and has administrative rights on the administrative console.

Information

See the information center on how to change this post-installation if not implemented during the installation:
`http://www-10.lotus.com/ldd/lcwiki.nsf/xpDocViewer.xsp?lookupName=IBM+Connections+4.0+documentation#action=openDocument&res_title=Changing_references_to_administrative_credentials_ic40&content=pdcontent`

Installing the SiteMinder Agents

This document describes a configuration that uses SiteMinder Policy Server 6.0 SP6, SiteMinder ASA 6.0 Agent for WebSphere Application Server (with CR00011 test fix), and SiteMinder Web Agent v6qmr6-cr007. The following sections detail how to install the web agent on the HTTP Server and the application server agents on all of the nodes in your configuration.

Information

For SiteMinder Policy server setup, see the information center.

Preparing WebSphere Application Server for SiteMinder

__ 1. If not already done, you must ensure that single sign-on is enabled on the Deployment Manager. On the deployment manager, go to **Security > Global Security > Web and SIP Security > Sign Sign-On (SSO)**. Ensure that the following is set:

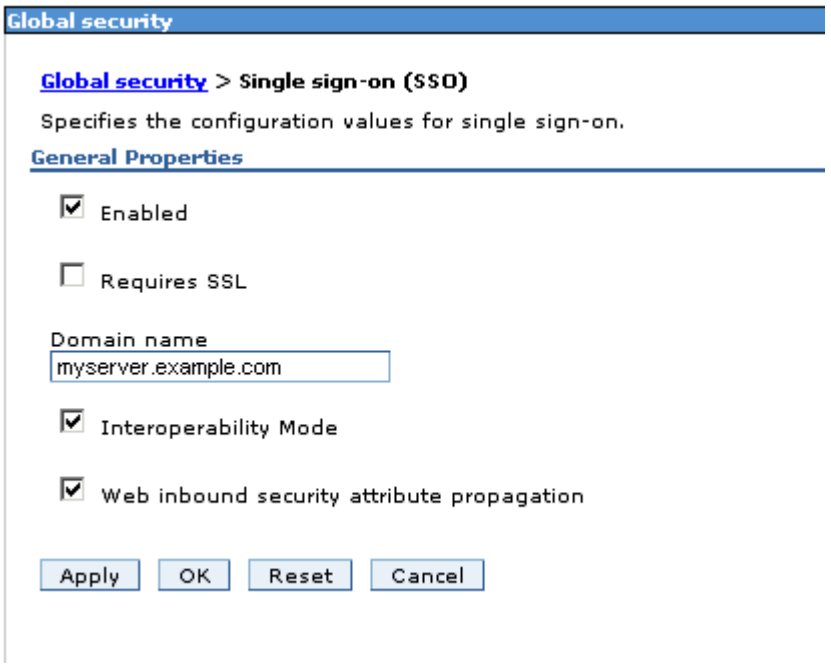


Figure 108. Global security

Copy unrestricted JCE policy files to WebSphere Application Server

Download and apply the Unrestricted JCE policy files:

__ 1. Go to the J2SE 5 SDK Security information web page (`https://www14.software.ibm.com/webapp/iwm/web/preLogin.do?source=jcesdk`).

__ 2. Authenticate with your universal IBM user ID and password.

__ 3. Download the Unrestricted JCE Policy files for SDK for all newer versions package.

__ 4. Extract the files from the downloaded package.

__ 5. Back up your existing copies (if any) of the `US_export_policy.jar` and `local_policy.jar` files, in the `app_server_root/java/jre/lib/security` directory.

__ 6. Copy the new JAR files from the extracted package to the same directory, overwriting any existing files.

__ 7. All servers, node agents, and deployment manager's must be restarted in order for this change to take effect.

Set up SiteMinder policy server

__ 1. Create agents on the SiteMinder Policy Server, including Web Agents for IBM HTTP Server and Microsoft IIS, and an Application Server Agent for WebSphere Application Server:

__ a. Open the SiteMinder Administration console.

__ b. Right click **Agents** and click **Create Agent**.

__ c. Enter details of the Name and Description of the Web Agent for IBM HTTP Server.

__ d. Repeat these steps for the Web Agent for IIS.

__ e. Repeat these steps for the Application Server Agent.

__ 2. Create Agent Configuration Objects on the SiteMinder Policy Server. In the SiteMinder Administration Console, open the Agent Configuration Objects pane and complete the following steps:

__ a. Configure the Web Agent for IBM HTTP Server:

i. Right click **Apache Default Settings Agent** and click **Duplicate Configuration Object**.

ii. Enter the Name and description of the Agent Configuration Object.

iii. Update the following parameters to match your environment:

- **DefaultAgentName**

  Name of the Apache Agent that was created earlier

- **CookieDomain**

  your_domain

where `your_domain` is your IBM Connections domain. If, for example, the URL is `http://activities.example.com/activities`, your host name is `activities.example.com` and your domain is `example.com`. In this example, you would set `CookieDomain=example.com`.

- **RequireCookies** NO

  This parameter configures the Web Agent to support basic authentication but without requiring all API client programs to support cookies.

- **BadCSSChars** <,>

  This parameter enables the Invite colleagues function in Profiles.

- **LogOffUri** URI

  Configure SiteMinder to recognize only one web address as the logout web address. Uncomment one of the following URIs by removing the number sign (#) character:

```
#LogOffUri="/activities/service/html/ibm_security_logout"
#LogOffUri="/blogs/ibm_security_logout"
#LogOffUri="/communities/communities/ibm_security_logout"
#LogOffUri="/dogear/ibm_security_logout"
#LogOffUri="/files/ibm_security_logout"
#LogOffUri="/forums/ibm_security_logout"
```

```
#LogOffUri="/homepage/web/ibm_security_logout"
#LogOffUri="/moderation/ibm_security_logout"
#LogOffUri="/news/ibm_security_logout"
#LogOffUri="/profiles/ibm_security_logout"
#LogOffUri="/search/ibm_security_logout"
#LogOffUri="/wikis/ibm_security_logout"
```

__ b. Under the System tab, update the Agent Configuration Object with the following value: `FCCCompatMode: NO`.

__ c. Configure the Web Agent for IIS:

i. Right-click IIS Default Settings Agent and select Duplicate Configuration Object.

ii. Enter the Name and description of the Agent Configuration Object.

iii. Update the following parameters to match your environment:

- **DefaultAgentName**

  Name of the Apache Agent that was created earlier

- **CookieDomain**

  your_domain

  where your_domain is your IBM Connections domain. If, for example, the URL is `http://activities.example.com/activities`, your host name is `activities.example.com` and your domain is `example.com`. In this example, you would set `CookieDomain=example.com`.

- **RequireCookies** NO

  This parameter configures the Web Agent to support basic authentication but without requiring all API client programs to support cookies.

- **BadCSSChars** <,>

  This parameter enables the Invite colleagues function in Profiles.

__ d. Configure the Application Server Agent:

i. Right click **Apache Default Settings Agent** and select **Duplicate Configuration Object**.

ii. Enter the name and description of the Agent Configuration Object.

iii. Update the following parameters to match your environment:

- **DefaultAgentName**

  Name of the Apache Agent that was created earlier

- **CookieDomain**

  your_domain

  where your_domain is your IBM Connections domain. If, for example, the URL is `http://activities.example.com/activities`, your host name is `activities.example.com` and your domain is `example.com`. In this example, you would set `CookieDomain=example.com`.

- **AssertionAuthResource**

  /siteminderassertion

- **AssertbyUserID**

  True

Note

1. When activated, the LogOffUri parameter clears the SMSESSION cookie and ensures that the user is logged out of all IBM Connections browser sessions.

2. To add parameters, edit the Agent Configuration Object on the SiteMinder Policy Server. Alternatively, you can edit the LocalConfig.conf file on the HTTP server if the Web Agent is configured to use it.

3. If you are editing the SiteMinder configuration file directly, you must surround the values of SiteMinder configuration parameters with quotation marks ("); for example: `BadCSSChars="<,>"`. If you are changing these parameters within the SiteMinder Policy Server, do not use quotation marks.

__ 3. Specify your SiteMinder Authentication Scheme configuration:

__ a. Open the SiteMinder Administration Console and go to the Authentication Scheme Properties dialog box.

__ b. From the Authentication Scheme type list, click **Windows Authentication template**.

__ c. Clear the **Use Relative Target** check box.

__ d. Enter the URL of your IIS server in the web Server Name field.

__ e. Complete the User DN Lookup field with the appropriate information for your domain. For example, `(sAMAccountName=%{UID})`.

__ 4. On the SiteMinder Policy Server, create a domain for the IBM HTTP Server web agent.

__ 5. Create protected realms under the IBM HTTP Server Web Agent domain:

__ a. Using the IBM HTTP Server Agent Object and Windows Authentication Scheme that you created earlier, create SiteMinder realms that Windows forms authentication protects.

Table 1: Realms that require forms authentication

Application	Protected URL resource
ConnectionsDefaultRealm	/
Activities	/activities/follow/atomfba /activities/service/atom2/forms /activities/service/atom2/communityEvent /activities/service/download/forms /activities/service/getnonce/forms

Blogs	/blogs/api_form /blogs/atom_form /blogs/follow/atomfba /blogs/roller-ui/blog /blogs/roller-ui/feed_form /blogs/roller-ui/rendering/api_form /blogs/roller-ui/rendering/feed_form /blogs/services/atom_form
Bookmarks	/dogear/atom_fba
Common resources	/connections/opensocial/rest
Communities	/communities/calendar/atom_form /communities/follow/atomfba /communities/forum/service/atom/forms /communities/recomm/ajax /communities/recomm/atom_form /communities/service/atom/forms
Files	/files/follow/atomfba /files/form/cmis/repository
Forums	/forums/atom/forms /forums/follow/atomfba
Metrics	/metrics /cognos
Profiles	/profiles/atom/forms /profiles/atom2/forms /profiles/follow/atomfba
Wikis	/wikis/follow/atomfba

__ 6. Using the IBM HTTP Server Agent Object that you created earlier, create SiteMinder realms that basic authentication protects.

Table 2: Realms that require basic authentication

Application	Protected URL resource
Activities	/activities/follow/atom /activities/service/download /activities/service/html/autocompleteactivityname /activities/service/html/autocompleteentryname /activities/service/html/autocompletemembers /activities/service/atom /activities/service/getnonce
Blogs	/blogs/api /blogs/atom /blogs/follow/atom /blogs/issuecategories /blogs/roller-ui/BlogsWidgetEventHandler.do /blogs/roller-ui/feed /blogs/roller-ui/rendering/api /blogs/roller-ui/rendering/feed /blogs/services/atom
Bookmarks	/dogear/api/app /dogear/api/deleted /dogear/api/notify /dogear/atom
Common resources	/connections/opensocial/basic/rest

Communities	/communities/calendar/atom /communities/calendar/handleEvent /communities/calendar/ical /communities/follow/atom /communities/forum/service/atom /communities/recomm/atom /communities/recomm/handleEvent /communities/service/atom /communities/service/json
Files	/files/basic/api /files/basic/cmis /files/basic/opensocial /files/follow/atom
Forums	/forums/atom /forums/follow/atom
Home page	/homepage/atom/search /homepage/atom/mysearch
News	/news/atom/service /news/atom/stories/newsfeed /news/atom/stories/public /news/atom/stories/saved /news/atom/stories/statusupdates /news/atom/stories/top /news/atom/watchlist /news/atomfba/stories/public
Profiles	/profiles/atom /profiles/atom2 /profiles/audio.do /profiles/follow/atom /profiles/json /profiles/photo.do /profiles/vcard
Wikis	/wikis/basic/api /wikis/follow/atom

Optional

Protect login credentials with encryption

Using the Basic over SSL Template scheme, create a SiteMinder Authentication Scheme and apply the new Authentication Scheme to all the SiteMinder realms that require basic authentication.

__ 7. Create Delete and Head actions for the Web Agent. By default, the Web Agent has only the Get, Post, and Put actions available. To add the Delete and Head actions, complete the following steps:

__ a. In the SiteMinder Administration Console, click **View** and click **Agent Types**.

__ b. Click **Agent Types** in the Systems pane.

__ c. Double-click **Web Agent** in the Agent Type list.

__ d. In the Agent Type Properties dialog box, click **Create**.

__ e. Enter `Delete` in the New Agent Action dialog box and click **OK**.

__ f. Enter Head in the New Agent Action dialog box and click **OK**.

__ g. Click **OK** again to save the new action.

__ 8. Create the following rules for each realm:

Table 3: Rules for the IBM HTTP Server realmsGetPostPutDelHead rule

GetPostPutDelHead rule	OnAuthAccept rule
Realm: CurrentRealm	Realm: CurrentRealm
Resource: * (not /*)	Resource: * (not /*)
Action: Web Agent actions > Get,Post,Put,Delete,Head	Action: Authentication events > OnAuthAccept
When this Rule fires: Allow Access	When this Rule fires: Allow Access
Enable or Disable this Rule: Enabled	Enable or Disable this Rule: Enabled

__ 9. Create a policy and add the users who can access the server to the policy. You can allow all users in the LDAP directory or a subset of users; for example: an LDAP branch, individual users, or groups of users.

__ 10. Add the new rules to the new policy.

__ 11. Specify realms that SiteMinder does not protect.

Note

You must configure notification templates and some Atom feeds as unprotected URLs. The Blogs footer page must also be unprotected because Blogs uses the Velocity template to extract footer pages.

Table 4: Realms that do not require authentication

Application	Unprotected URL resource
Activities	/activities/auth /activities/images /activities/oauth /activities/service/html/images /activities/service/html/mainpage /activities/service/html/styles /activities/service/html/themes /activities/service/html/servermetrics /activities/service/html/serverstats /activities/serviceconfigs /activities/static/
Blogs	/blogs/oauth /blogs/serviceconfigs /blogs/static/
Bookmarks	/dogear/oauth /dogear/peoplelike /dogear/serviceconfigs /dogear/static/

Common resources	/connections/bookmarklet/tools/blet.js /connections/bookmarklet/tools/discussThis.js /connections/bookmarklet/tools/rlet.js /connections/core/oauth /connections/oauth /connections/resources/ic /connections/resources/socmail-client /connections/resources/socpim /connections/resources/web /nav/common
Communities	/communities/calendar/Calendar.xml /communities/calendar/oauth /communities/comm.widget /communities/images /communities/nav /communities/recomm/oauth /communities/resourceStrings.do /communities/service/atom/oauth /communities/service/html/communityview /communities/service/html/community/autoCompleteMembers.do /communities/service/html/singleas /communities/service/opensocial/oauth /communities/serviceconfigs /communities/static/ /communities/stylesheet /communities/tools/embedAS.html /communities/widgets
Files	/files/app /files/basic/anonymous/api /files/basic/anonymous/cmis /files/basic/anonymous/opensocial /files/form/anonymous/api /files/form/anonymous/cmis /files/form/anonymous/opensocial /files/oauth /files/static/
Forums	/forums/oauth /forums/serviceconfigs /forums/static/
Home page	/homepage/oauth /homepage/search /homepage/serviceconfigs /homepage/static/ /homepage/web/updates/
Metrics	/metrics/service/eventTracker /metrics/service/oauth /cognos/servlet
Moderation	/moderation/app /moderation/oauth /moderation/static
News	/help /news/microblogging/isPermitted.action /news/follow/oauth /news/oauth /news/serviceconfigs /news/sharebox/config.action /news/static/
OAuth Provider	/oauth2

Profiles	/profiles/atom/forms/connections.do /profiles/images /profiles/oauth /profiles/serviceconfigs /profiles/static/
Search	/search/atom/search /search/oauth /search/static/
Widget container	/connections/opensocial/anonymous/rest /connections/opensocial/common /connections/opensocial/gadgets /connections/opensocial/ic /connections/opensocial/oauth /connections/opensocial/rpc /connections/opensocial/social /connections/opensocial/xrds /connections/opensocial/xpc
Wikis	/wikis/basic/anonymous/api /wikis/form/anonymous/api /wikis/home /wikis/js /wikis/oauth /wikis/static/

__ 12. On the SiteMinder Policy Server, create a domain for the Application Server Agent.

__ 13. Add the following realm to the new WebSphere Application Server domain:

Table 5: SiteMinder realms for WebSphere Application Server

Realm name	Protected resource
SM TAI Validation	/siteminderasssertion

Note

You must configure the Protected Resource of this realm to match the AssertionAuthResource parameter that you configured earlier for the Application Server Agent.

__ 14. On the SiteMinder Policy Server, create a domain for the IIS Server Agent.

__ 15. Using the IIS Agent Object and Windows Authentication Scheme that you created earlier, create a SiteMinder realm that Windows authentication protects.

Table 6: SiteMinder realms that require Windows authentication

Realm name	Protected resource
IIS_Realm	/

__ 16. Create the following rules for this realm:

Table 7: Rules for the IIS realm

GetPostPutDelHead rule	OnAuthAccept rule
Realm: CurrentRealm	Realm: CurrentRealm
Resource: * (not /*)	Resource: * (not /*)
Action: Web Agent actions -> Get,Post,Put,Delete,Head	Action: Authentication events -> OnAuthAccept
When this Rule fires: Allow Access	When this Rule fires: Allow Access
Enable or Disable this Rule: Enabled	Enable or Disable this Rule: Enabled

__ 17. Set the timeout value of the session for each realm.

__ a. In the SiteMinder Policy Server, open the **Realm Dialog** and click **Session**.

__ b. In the Session Timeouts Group Box, enter timeouts for each realm. Enter the following values, if they are not already present:

Maximum Timeout Enabled

2 Hours 0 Minutes

Idle Timeout Enabled

1 Hours 0 Minutes

Note

The maximum timeout and the idle timeout must be longer than the LTPA token timeout, which is defined in WebSphere Application Server. The LTPA token timeout is set to 120 minutes by default.

Install the web agent on the IBM HTTP server

You must uninstall and reinstall the web agent to enable SiteMinder. After uninstallation, delete the `/opt/netegrity` folder and install to the same place.

__ 1. Extract the web agent files to a folder on your directory. Run `chmod 777 nete..` to make it executable on Linux to run the installation.

__ 2. Wait for it to complete.

__ 3. Close the installation.

__ 4. Run the web agent installation wizard from the files that are downloaded from WTI.

Figure 109. CA SiteMinder Web Agent

__ 5. Click **Next** at the following panel.

Figure 110. CA SiteMinder Web Agent: Introduction

__ 6. Accept the license agreement and click **Next**.

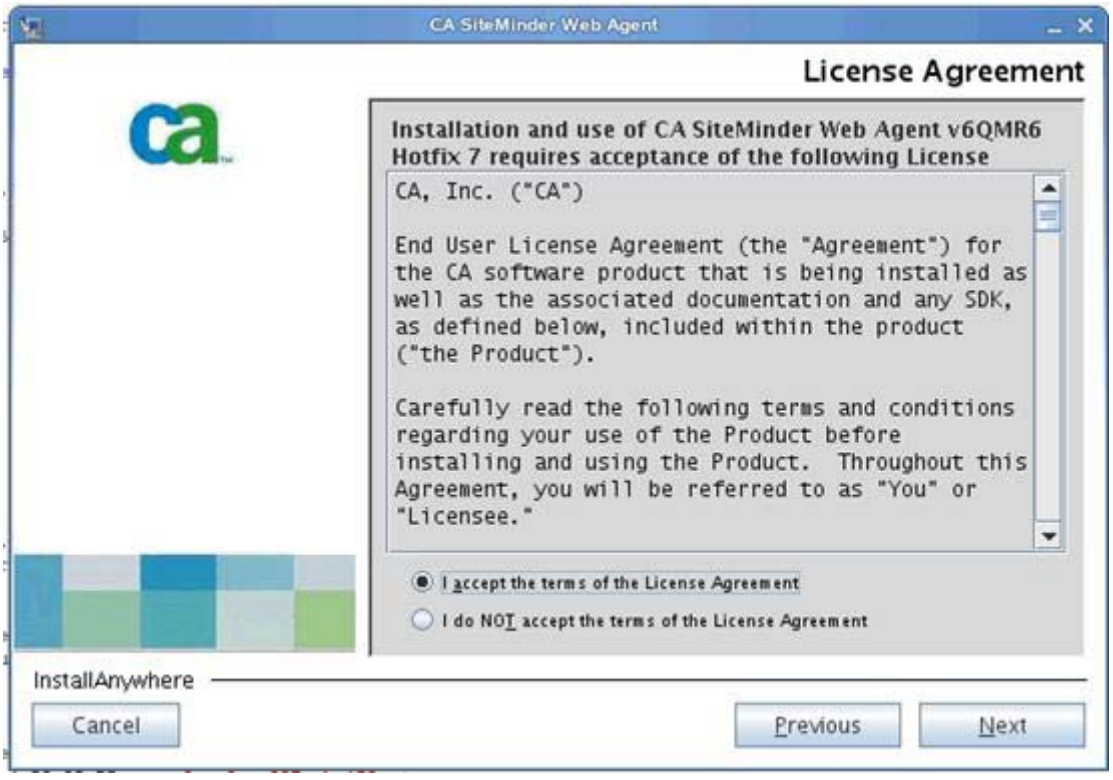


Figure 111. CA SiteMinder Web Agent: License Agreement

__ 7. Click **Next** at the information panel to continue.

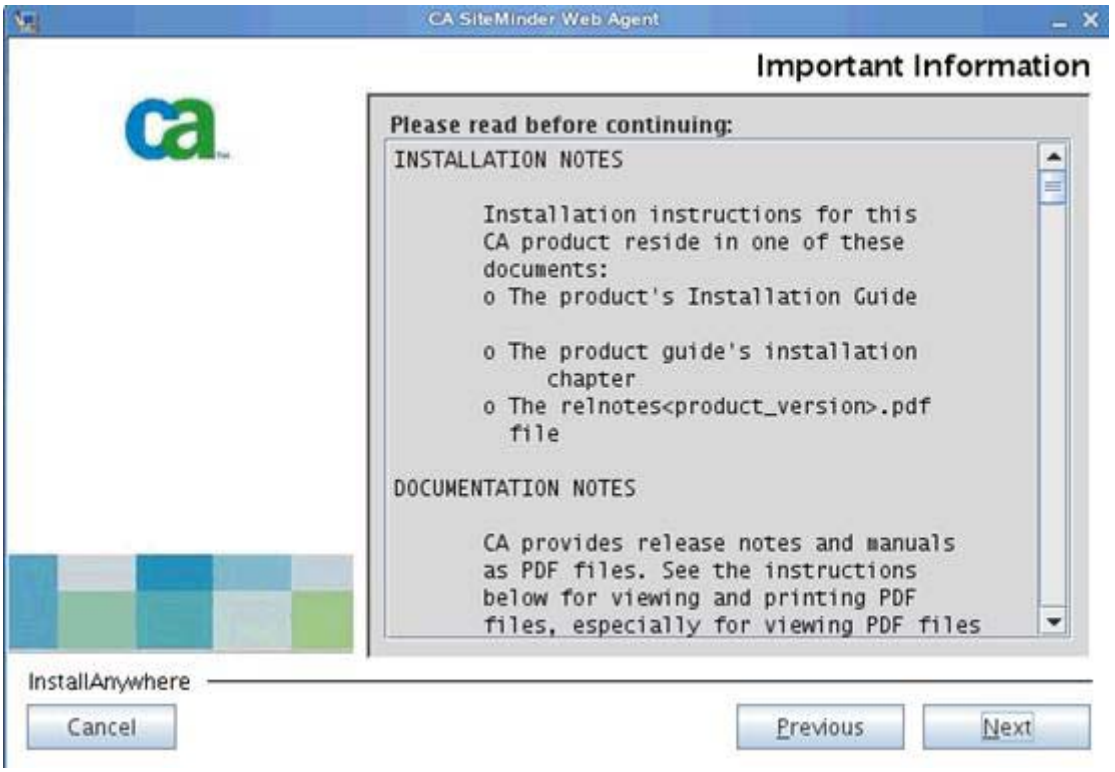


Figure 112. CA SiteMinder Web Agent: Important information

__ 8. Select a path to install the web agent and click **Next** to continue.

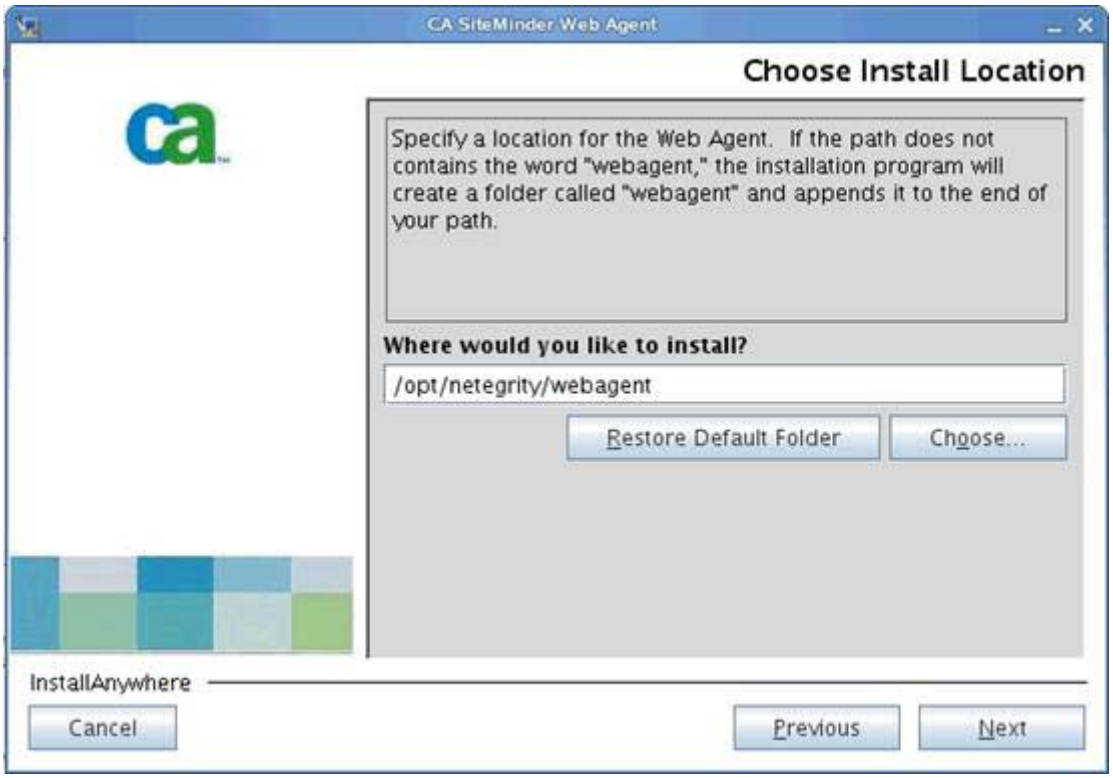


Figure 113. CA SiteMinder Web Agent: Choose Install Location

The Web Agent starts configuring for your system.

Figure 114. CA SiteMinder Web Agent: Wait

__ 9. Click **Install** to begin the web agent installation.

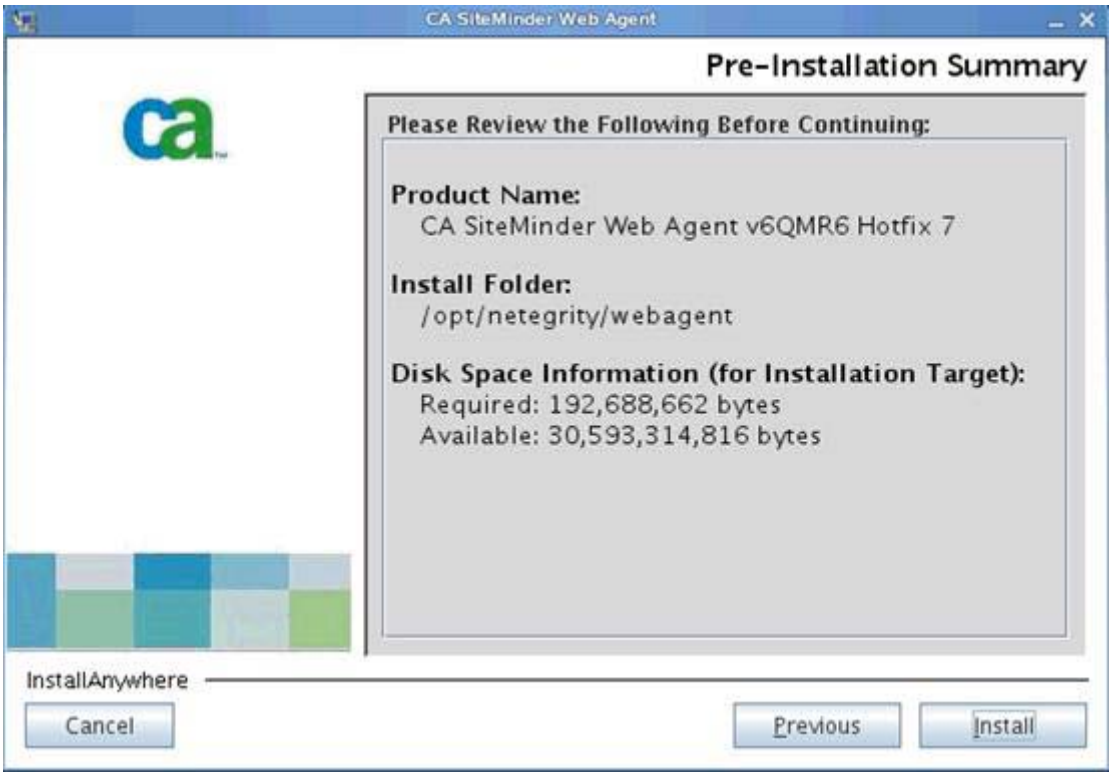


Figure 115. CA SiteMinder Web Agent: Pre-Installation Summary

The CA SiteMinder web agent starts installing.

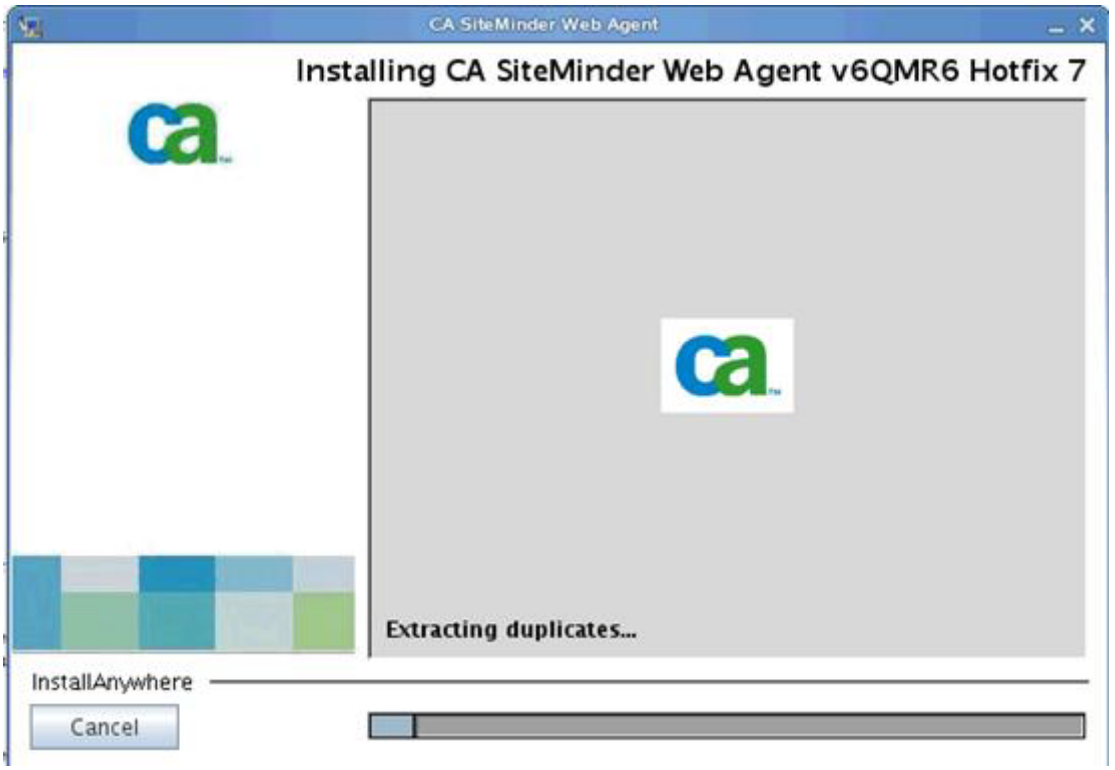


Figure 116. CA SiteMinder Web Agent: Installation in progress

__ 10. Click **Done** when installation completes.

In this case the installation log at `/opt/netegrity/webagent/install_config_info/CA_SiteMinder_Web_Agent_v6QMR6_InstallLog.log` reports the following, there are no unrecoverable errors so it is safe to proceed:

```
Installation: Successful.
474 Successes
0 Warnings
0 NonFatalErrors
0 FatalErrors
```

Registration

__ 1. CD to `/opt/netegrity/webagent`.

__ 2. Run `. ./nete_wa_env.sh`.

__ 3. Register the web agent with the policy server where < `./smreghost -i SM policy server -u admin_id -p admin_pwd -hn webagent_hostname -hc hostconfig_object`> for example < `./smreghost -i SM_Policy_Server -u RegHost -p RegHost -hn connections -hc host_connections`>.

__ 4. Check the `webagent.config` file in `/opt/IBM/HTTPServer/conf` that `EnableWebAgent="YES"`.

__ 5. Start the HTTP server:

__ a. Run `. ./envars-std`

__ b. ./apachectl start

__ c. SiteMinder should prompt you to link to the HTTP web server page.

Note

You can check that SiteMinder is enabled only if the SiteMinder policy server is enabled for SiteMinder. If SPNEGO is enabled on the policy server, then you do not see a SiteMinder screen.

__ 6. After configuring the web agent as previously, find the `WebAgent.conf` in the `HTTPServer/conf` directory. Open this file and edit it so `EnableWebAgent=YES`. Now restart your HTTP Server. When attempting to access the HTTP Server root, you should now see the SiteMinder login screen and be able to log in to get the IBM HTTP Server Splash Screen. It indicates that SiteMinder is set up correctly with the WebAgent.

Figure 117. IBM HTTP Server: Login

Install Application Server Agent

Install the Application Server Agent on both nodes `-node1.example.com` and `node2.example.com`.

__ 1. Run the TAI agent installation by using the following JAR command: `<java: jar ca-asa-6.0-cr11-was.jar >` for the application server agent. Click **Next** to continue.

Figure 118. eTrust SiteMinder: Application Server Agent v6.0 for WebSphere

__ 2. Click **Next** to continue.

Figure 119. eTrust SiteMinder: Application Server Agent v6.0 for WebSphere: Introduction

__ 3. Accept the license agreement, click **Next** to continue.

Figure 120. eTrust SiteMinder: Application Server Agent v6.0 for WebSphere: License Agreement

__ 4. Choose an installation location and click **Next** to continue.

Figure 121. eTrust SiteMinder: Application Server Agent v6.0 for WebSphere: Choose Install Folder

__ 5. Specify where WebSphere is installed.

Figure 122. eTrust SiteMinder: Application Server Agent v6.0 for WebSphere: Choose WebSphere Folder

__ 6. Click **Yes, create trusted host** to create a trusted host.

Figure 123. eTrust SiteMinder: Application Server Agent v6.0 for WebSphere: Host Registration

__ 7. Enter the information of the SiteMinder server. Click **Next** to continue.

Figure 124. eTrust SiteMinder: Application Server Agent v6.0 for WebSphere: Host Registration

__ 8. Allow the wizard time to register the host.

Figure 125. eTrust SiteMinder: Application Server Agent v6.0 for WebSphere: Registering the host

__ 9. Enter the agent configuration object name and click **Next**.

Figure 126. eTrust SiteMinder: Application Server Agent v6.0 for WebSphere: Agent Configuration

__ 10. Review any errors messages in the installation log. In this case, there are benign errors. Click **Done** to exit the wizard.

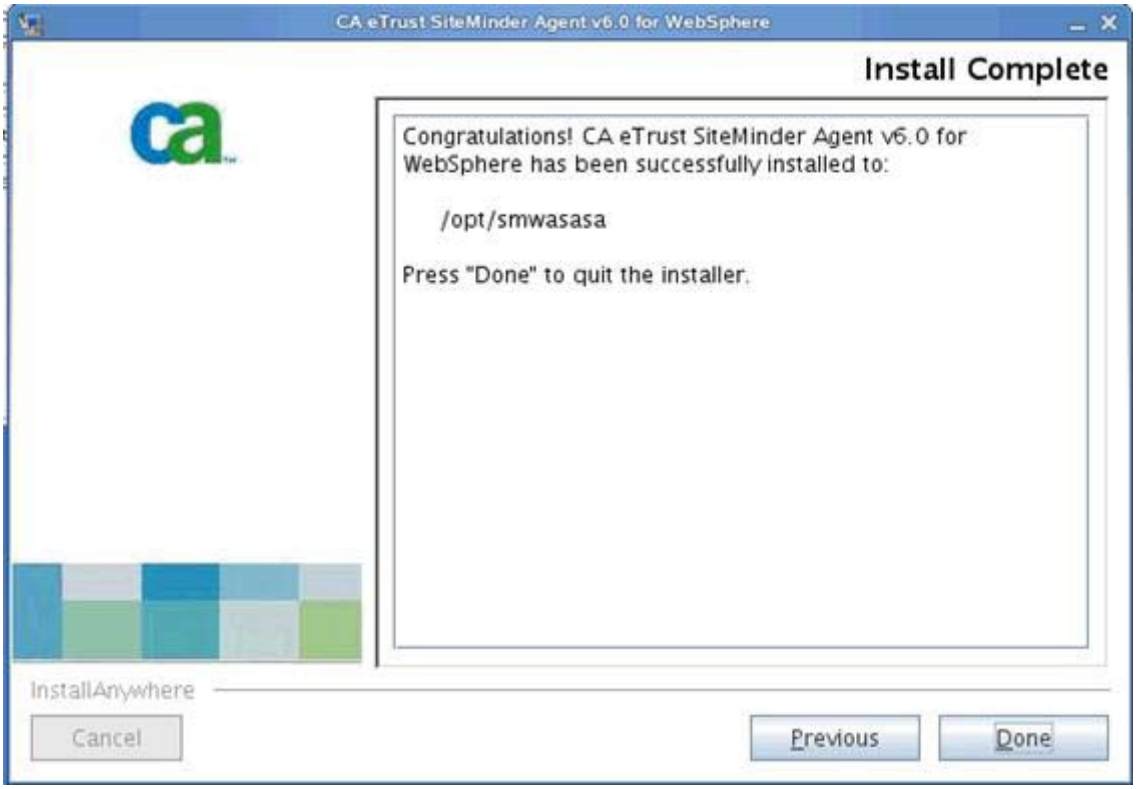


Figure 127. eTrust SiteMinder: Application Server Agent v6.0 for WebSphere: Install Complete

In this case the installation log at `/opt/smwasasa/log/CA_eTrust_SiteMinder_Agent_v6.0_for_WebSphere_InstallLog.log` reports the following, again there are no unrecoverable errors so it is safe to proceed:

```
Summary
-------
Installation: Successful.
54 Successes
0 Warnings
0 NonFatalErrors
0 FatalErrors
```

5. Post-agent installation actions

After you installed the various SiteMinder agents on your nodes and web server, turn your attention to the following tasks. The trust association interceptor must be enabled from the deployment manager and various rules must be put in place on the web server to handle logging out from SiteMinder correctly. Here, the SiteMinder authenticator is not being set, because you are enabling SPNEGO. SiteMinder and SPNEGO configuration use the default authenticator. In a stand-alone SiteMinder configuration, you would normally set the SiteMinder authenticator.

Actions on WebSphere Application Server post-agent installation

__ 1. When the Application Server Agent is configured ensure to copy `smagent.properties` from the agent installation directory: `smwasasa\conf` to `AppServer\profiles\AppSrv01\properties` on each node. That is, to the nodes and to the Cognos node.

```
        :/opt/smwasasa/conf # ls -l
total 20
-rwxr--r-- 1 root root 273 Jul 20 09:49 AsaAgent-assertion.conf
-rwxr--r-- 1 root root 273 Jul 20 09:49 AsaAgent-auth.conf
-rwxr--r-- 1 root root 273 Jul 20 09:49 AsaAgent-az.conf
-rw-r--r-- 1 root root 706 Jul 20 09:49 SmHost.conf
-rwxr--r-- 1 root root 459 Jul 20 09:49 smagent.properties
        :/opt/smwasasa/conf # cp smagent.properties /opt/IBM/WebSphere/AppServer
/profiles/AppSrv01/properties/
        :/opt/smwasasa/conf # cp smagent.properties /opt/IBM/WebSphere/AppServer
/profiles/AppSrv02/properties/
        :/opt/smwasasa/conf #
```

```
Terminal
File  Edit  View  Terminal  Tabs  Help
-rwxr-xr-x 1 root root  5738 Mar 21 20:29 ipc.client.props
drwxr-xr-x 5 root root  4096 Mar 21 20:55 linuxMenu
-rw-r--r-- 1 root root   520 Mar 21 20:55 portdef.props
-rw-r--r-- 1 root root   304 Mar 21 20:55 profileKey.metadata
-rwxr-xr-x 1 root root   291 Mar 21 20:40 rrdSecurity.props
-rw-r--r-- 1 root root 12147 Mar 21 20:54 sas.client.props
-rwxr-xr-x 1 root root  8437 Mar 21 20:32 sas.server.props
-rw-r--r-- 1 root root  7482 Mar 21 20:54 sas.stdclient.properties
-rw-r--r-- 1 root root  7482 Mar 21 20:54 sas.tools.properties
drwxr-xr-x 2 root root  4096 Mar 21 20:55 script
-rwxr-xr-x 1 root root  1703 Mar 21 20:32 server.policy
-rwxr-xr-x 1 root root   999 Mar 21 20:45 sib.client.ssl.properties
-rwxr--r-- 1 root root   459 Jul 20 09:52 smagent.properties
-rw-r--r-- 1 root root  5667 Mar 21 20:54 soap.client.props
-rw-r--r-- 1 root root  5050 Mar 21 20:55 ssl.client.props
-rwxr-xr-x 1 root root  3127 Mar 21 20:32 sslbitsizes.properties
drwxr-xr-x 2 root root  4096 Mar 21 20:54 version
-rwxr-xr-x 1 root root   663 Mar 21 20:32 was.policy
-rwxr-xr-x 1 root root   103 Mar 21 20:45 wmq.client.props
-rw-r--r-- 1 root root  7000 Mar 22 11:18 wsadmin.properties
-rwxr-xr-x 1 root root  7105 Mar 21 20:32 wsjaas.conf
-rwxr-xr-x 1 root root  3161 Mar 21 20:32 wsjaas_client.conf
-rwxr-xr-x 1 root root   846 Mar 21 20:32 wsserver.key
dslvm768:/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/properties #
```

Figure 128. Copying smagent.properties from the agent installation directory to the application server

__ 2. Next, on the Deployment Manager configure Trust Association Interceptor on WebSphere Application Server, from the deployment manager administrative console for WebSphere Application Server, click **Security > Global security > Web and SIP security**, click **Trust association**. Click **Enable Trust Association** and then click **Save**.

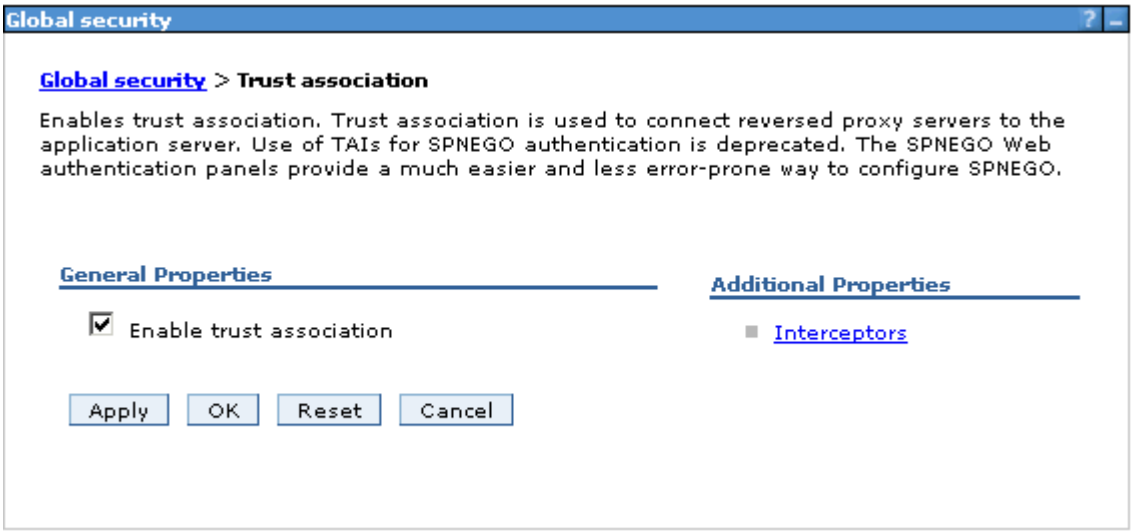


Figure 129. Enabling trust association

__ 3. Next, back in the trust association screen, click **Interceptors**. Click **New** and add an interceptor with the following name (`com.netegrity.SiteMinder.websphere.auth.SmTrustAssociationInterceptor`). Click **OK** and save the change.

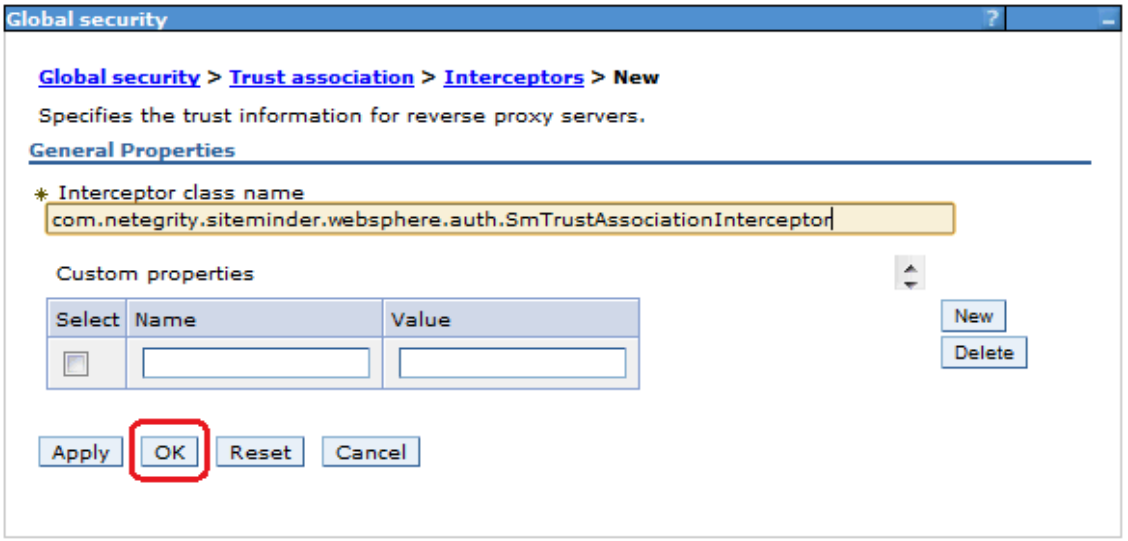


Figure 130. Creating an interceptor

__ 4. It is OK to delete Tivoli Access Manager and SPNEGO interceptors. Leaving these interceptors in place causes no issues but results in error messages in the logs during startup so it makes sense to delete these interceptors. Click **Delete** and save this change.

Note

You must not delete the oauth interceptor (`com.ibm.ws.security.oauth20.tai.OAuthTAI`) it is required for oauth to work properly. After this step you have two interceptors for oauth and for SiteMinder.

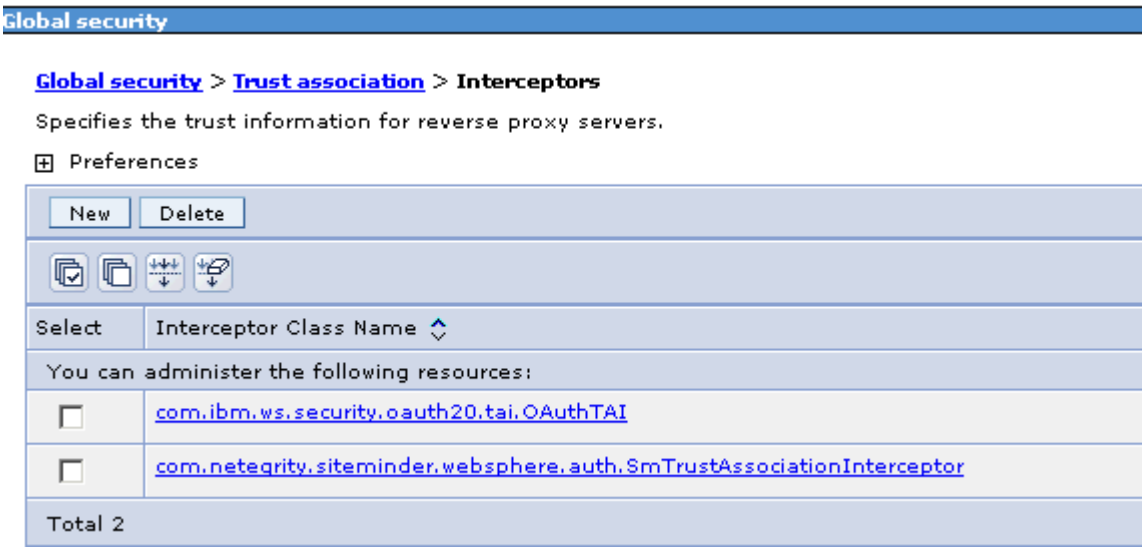


Figure 131. Global security > Trust association > Interceptors

Actions on HTTP server after the agent installation

Create rewrite rules to remap Atom API requests and to redirect URLs when users log out of Lotus Connections.

__ 1. Open the IBM HTTP Server `httpd.conf` configuration file. The file is stored in the `C:\IBM\HTTPServer\conf` directory on the web server.

__ 2. The extracted section of the following `httpd.conf` file shows these rules being implemented in both the HTTP and HTTPS sections of this file. In this extract, the logout rules redirect users to the home page logout screen and when they are logged out they are redirected to the page at `home.example.com`.

__ 3. When this change is made, save and close the `httpd.conf` file.

__ 4. Restart the IBM HTTP Server.

Note

Uncommon LoadModule `rewrite_module modules/mod_rewrite.so` line in the `httpd.conf` file.

This line is commented out by default. When the line is commented out, the web server does not start.

```
RewriteEngine On
RewriteCond %{REQUEST_URI} /(.*)/ibm_security_logout(.*)
RewriteCond %{QUERY_STRING}
!=logoutExitPage=https://connections.example.com/homepage
RewriteRule /(.*)/ibm_security_logout(.*)
/homepage/web/ibm_security_logout?logoutExitPage=https://connections.example
.com/homepage [noescape,L,R]
RewriteCond %{REQUEST_URI} !^/blogs/roller-ui/rendering/(.*)
RewriteRule ^/blogs/(.*)/api/(.*) /blogs/roller-ui/rendering/api/$1/api/$2
[R,L]
RewriteCond %{REQUEST_URI} !^/blogs/roller-ui/rendering/(.*)
RewriteRule ^/blogs/(.*)/feed/tags/atom(.*)
/blogs/roller-ui/rendering/feed/$1/tags/atom/ [R,L]
RewriteCond %{REQUEST_URI} !^/blogs/roller-ui/rendering/(.*)
RewriteRule ^/blogs/(.*)/feed/entries/atom(.*)
/blogs/roller-ui/rendering/feed/$1/entries/atom/ [R,L]
RewriteCond %{REQUEST_URI} !^/blogs/roller-ui/rendering/(.*)
RewriteRule ^/blogs/(.*)/feed/comments/atom(.*)
/blogs/roller-ui/rendering/feed/$1/comments/atom/ [R,L]
RewriteCond %{REQUEST_URI} !^/blogs/roller-ui/rendering/(.*)
RewriteRule ^/blogs/(.*)/feed/blogs/atom(.*)
/blogs/roller-ui/rendering/feed/$1/blogs/atom/ [R,L]
LoadModule ibm_ssl_module modules/mod_ibm_ssl.so
<IfModule mod_ibm_ssl.c>
Listen 0.0.0.0:443
<VirtualHost *:443>
ServerName connections.example.com
```

```
SSLEnable
AllowEncodedSlashes On
RewriteEngine On
RewriteCond %{REQUEST_URI} /(.*)/ibm_security_logout(.*)
RewriteCond %{QUERY_STRING}
!=logoutExitPage=https://connections.example.com/homepage
RewriteRule /(.*)/ibm_security_logout(.*)
/homepage/web/ibm_security_logout?logoutExitPage=https://connections.example
.com/homepage [noescape,L,R]
RewriteCond %{REQUEST_URI} !^/blogs/roller-ui/rendering/(.*)
RewriteRule ^/blogs/(.*)/api/(.*) /blogs/roller-ui/rendering/api/$1/api/$2
[R,L]
RewriteCond %{REQUEST_URI} !^/blogs/roller-ui/rendering/(.*)
RewriteRule ^/blogs/(.*)/feed/tags/atom(.*)
/blogs/roller-ui/rendering/feed/$1/tags/atom/ [R,L]
RewriteCond %{REQUEST_URI} !^/blogs/roller-ui/rendering/(.*)
RewriteRule ^/blogs/(.*)/feed/entries/atom(.*)
/blogs/roller-ui/rendering/feed/$1/entries/atom/ [R,L]
RewriteCond %{REQUEST_URI} !^/blogs/roller-ui/rendering/(.*)
RewriteRule ^/blogs/(.*)/feed/comments/atom(.*)
/blogs/roller-ui/rendering/feed/$1/comments/atom/ [R,L]
RewriteCond %{REQUEST_URI} !^/blogs/roller-ui/rendering/(.*)
RewriteRule ^/blogs/(.*)/feed/blogs/atom(.*)
/blogs/roller-ui/rendering/feed/$1/blogs/atom/ [R,L]
</VirtualHost>
</IfModule>
SSLDisable
```


Enabling and disabling SiteMinder and other troubleshooting steps

When attempting to debug any SiteMinder issues, a good tactic is first to disable SiteMinder to verify that the problem is not an underlying configuration problem on Connections that SiteMinder masks. When you verify that everything works correctly outside the SiteMinder environment, you can be confident that the introduction of SiteMinder caused the problems that are experienced. The following steps describe how to enable or disable SiteMinder and detail some other common troubleshooting techniques in this environment.

Enabling and Disabling SiteMinder

If you need to enable or disable SiteMinder at any point, do the following steps:

__ 1. Edit the `WebAgent.conf` on the HTTP Server (`HTTPServer/conf/WebAgent.conf`) and set "EnableWebAgent=NO". Restart the IBM HTTP Server.

__ 2. Change the custom authenticator back to the default authenticator in the `LotusConnections-config.xml`.

__ 3. Edit `smwasasa/conf/AsaAgent-assertion.conf` on both nodes and set `EnableWebAgent=NO`.

__ 4. Resynchronize nodes and restart Lotus Connections.

__ 5. Repeat this process to enable SiteMinder and instead set `EnableWebAgent=YES` where you previously set it to NO. You must also re-enable the custom authenticator in the `LotusConnections-config.xml` and restart the deployment.

__ 6. When SiteMinder is enabled, the following message should appear in the `SystemOut.log` for all Lotus Connections application servers to indicate SiteMinder loaded correctly with the configuration:

```
[10/11/10 12:45:23:225 EDT] 00000000 TrustAssociat A SECJ0121I: Trust
Association Init class
com.netegrity.SiteMinder.websphere.auth.SmTrustAssociationInterceptor loaded
successfully
```

Troubleshooting SiteMinder issues

Enable trace

Most errors that are encountered in this environment are typically interservice issues: communication errors between the back-end servers often because of authorization issues caused by the introduction of SiteMinder to the configuration. The following trace is appropriate in this circumstance to help diagnose issues: `com.ibm.connections.httpClient.*=all`.

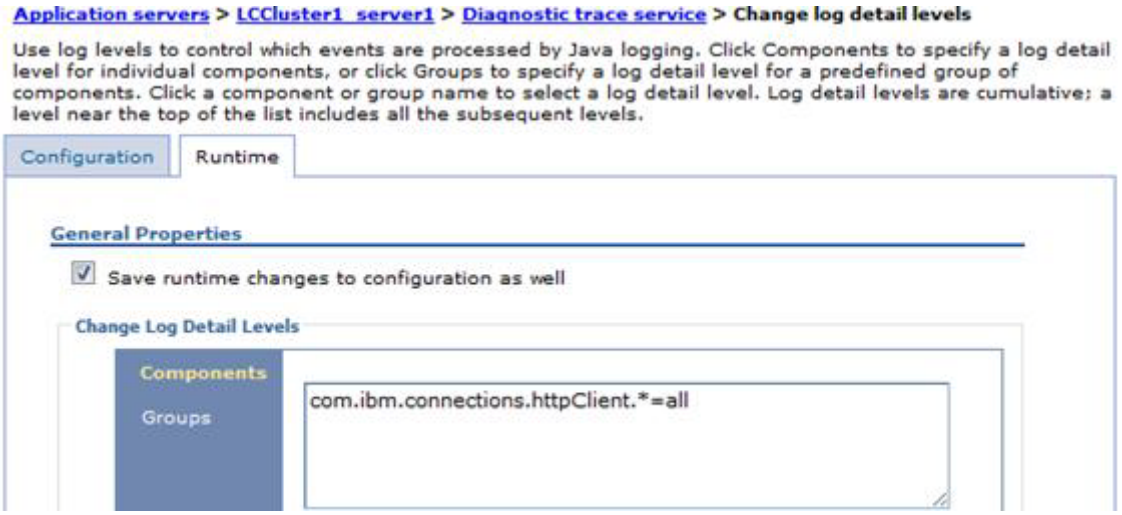


Figure 132. Change Log Detail Levels

Log files to help diagnose issues

To get a complete overview of any issues on the system with SiteMinder enabled, consult the following log files:

__ 1. Lotus Connections Server log files:

- __ a. SystemOut.log
- __ b. trace.log (if applicable)

__ 2. SiteMinder log files (on Nodes):

- __ a. smwasasa/log/smasa.log
- __ b. smwasasa/log/sm_tai.log

__ 3. SiteMinder log files (on web server):

- __ a. nnetegrity/webagent/log/wa.log
- __ b. netegrity/webagent/log/wa_trace.log

__ 4. SiteMinder Server log files:

- __ a. Consult the SiteMinder documentation to uncover what traces and logs can be enabled / referenced on the SiteMinder server side.

SiteMinder configuration files created by Web Agent and TAI/ASA

Here is a sample of the key configuration files on the nodes which are correctly configured. Note the relationship between all of the following files. Changes to these files require a restart to the web server in case of web agent and application server in case of ASA/TAI.

WebAgent.conf

WebAgent.conf is found in <HTTPServer_Root>/conf/WebAgent.conf and refers to the AgentConfigObject and SmHost.conf (which contains the policy server connection details). Also, note the EnableWebAgent parameter.

```
# WebAgent.conf: configuration file for SiteMinder Web Agent
# Web Agent Version = 6QMR6, Build = 667, Update = 0
#agentname="<AgentName>, <IPAddress>"
HostConfigFile="/opt/netegrity/webagent/config/SmHost.conf"
AgentConfigObject="connections_wa_conf"
EnableWebAgent="YES"
ServerPath="/opt/IBM/HTTPServer/conf"
localconfigfile="/opt/IBM/HTTPServer/conf/LocalConfig.conf"
LoadPlugin="/opt/netegrity/webagent/bin/libHttpPlugin.so"
#LoadPlugin="/opt/netegrity/webagent/bin/libAffiliate10Plugin.so"
#LoadPlugin="/opt/netegrity/webagent/bin/libSAMLAffiliatePlugin.so"
#LoadPlugin="/opt/netegrity/webagent/bin/libeTSSOPlugin.so"
#LoadPlugin="/opt/netegrity/webagent/bin/libIntroscopePlugin.so"
```

SmHost.conf

SmHost.conf is found at <SiteMinder ASA Home>/bin/SmHost.conf, refers to the policy server by IP address. It also contains the host name and hostconfigobject reference.

```
# Host Registration File: SmHost.conf
 #
 # This file contains bootstrap information required by
 # the SiteMinder Agent API to connect to Policy Servers
 # at startup. Be sure the IP addresses and ports below
 # identify valid listening Policy Servers. Please do not
 # hand edit the encrypted SharedSecret entry.
 #
 hostname="node1.example.com"

sharedsecret="{RC2}8DqJaGN/EnhNuEEqLiCieN/NHfSFKGAESra62kiN7B9az9Gni68XKbOgB
yaYNVNK7qsLUezwlimpMsViG/gfPZee7PYMl9A+LfcQkmDbhWsBOe1uNfEScvSyH7ysfiryHd5YU
fOVMNnGEjEjOhQioTwf7h2N26KqeuS0I6lZswv1KQTBBw7UXCPnlENF8DWl"
 sharedsecrettime="0"
 enabledynamichco="NO"
 hostconfigobject="host_node_TAI"
 # Add additional bootstrap policy servers here for fault tolerance.
 policyserver="policy_server_ip.40,44441,44442,44443"
 requesttimeout="60"
 cryptoprovider="BSAFE"
 # <EOF>
```

AsaAgent-assertion.conf

AsaAgent-assertion.conf, found at <SiteMinder ASA Home>/conf/AsaAgent-assertion.conf, contains an EnableWebAgent flag and references SmHost.conf and holding the value of the agent configuration object.

```
############################################################
 ## SiteMinder IBM WebSphere Application Server Agent
 ############################################################
 EnableWebAgent="YES"
 HostConfigFile="/opt/smwasasa/bin/SmHost.conf"
 AgentConfigObject="node_TAI_conf"
```

Smagent.properties

SmAgent.properties, found at <SiteMinder ASA Home>/conf/smagent.properties, is created when the ASA is registered. It contains the location of the AsaAgent-assertion.conf and is copied to <Application Server Home>/profiles/AppSrv01/properties on both nodes during the SiteMinder configuration.

```
#####################################################################
 # SiteMinder Generic Application Server Agent Properties File
 #####################################################################
 logfilename="/opt/smwasasa/log/smasa.log"
 loglevel="4"
 logappend="NO"
 logfile="YES"
 logconsole="NO"
 smazconf="/opt/smwasasa/conf/AsaAgent-az.conf"
 smauthconf="/opt/smwasasa/conf/AsaAgent-auth.conf"
 smassertionconf="/opt/smwasasa/conf/AsaAgent-assertion.conf"
```

Following are a few issues which occurred in the production of this document and might help in the resolution of other issues that are encountered in subsequent deployments.

Linux 64-bit issues

In addition to the windows known common issues above the following were found when integrating SiteMinder on a RedHat 5 64-bit configuration.

HTTP server does not start

If you already updated the httpd.conf file to point to `/opt/netegrity/webagent/bin/libmod_sm22.so` from `/opt/netegrity/webagent/bin/libmod_sm20.so` and the http server still does not start. Here is what to do:

__ 1. Edit the envvars-std file in `/opt/IBM/HTTPServer/bin` and add the location of the SiteMinder agent to the library path:

   __ a. `LD_LIBRARY_PATH="/opt/IBM/HTTPServer/lib:/opt/IBM/HTTPServer/gsk7/lib:/opt/netegrity/webagent/bin:$LD_LIBRARY_PATH"`.

   __ b. Export `LD_LIBRARY_PATH`.

   __ c. Save and close

   __ d. Run `<. ./envars-std>`.

   __ e. HTTP server should now start.

LLAWP error in the error_log

Note

```
[17/Jun/2011:14:23:01] [Error] SiteMinder Agent

Failed to send close message to LLAWP.
Execlp failed: 'Invalid argument'. LLAWP.exe must be callable from the system path.
CSmLowLevelAgent: No such file or directory
```

If an LLAWP error (similar to previously) appears in error_log, check your netegrity paths, as they might not be set. To set them:

__ 1. Change directory to `/opt/netegrity/webagent`.

__ 2. Run `nete_wa_env.sh`.

__ 3. Check they are set by typing each of the following:

   __ a. NETE_WA_ROOT

   __ b. PATH

   __ c. NETE_WA_PATH

   __ d. LD_LIBRARY_PATH

6. SPNEGO setup

How to configure SPNEGO over HTTPS

__ 1. Install the Web Agent on IIS:

__ a. Download the latest version of the Web Agent from the CA website (`http://www.ca.com/us/default.aspx`).

__ b. Install the Web Agent. For instructions, go to the SiteMinder BookShelf (`https://support.ca.com/cadocs/0/CA%20SiteMinder%20r6%200%20SP6-ENU/Bookshelf.html`).

__ c. When you are prompted for the Agent Configuration details, specify the Agent Configuration Object that you created earlier.

__ 2. Stop the Connections servers. Leave the deployment manager and the nodes running.

__ 3. In `/opt/IBM/HTTPServer/conf`, edit `http.conf` and add the following lines to the bottom of the file:

```
Listen 444
<VirtualHost *:444>
ServerName connections.example.com
SSLEnable
#KeyFile /local/IBM/HTTPServer/conf/wildcard/key.kdb
Keyfile "/opt/IBM/KeyFiles/webserver-key.kdb"
SSLStashFile "/opt/IBM/KeyFiles/webserver-key.sth"
</VirtualHost>
```


__ 4. In the admin console, go to WebSphere Application Server: **Environment > Virtual Hosts > default_host > Host Aliases > New**, and enter the host name and Port.

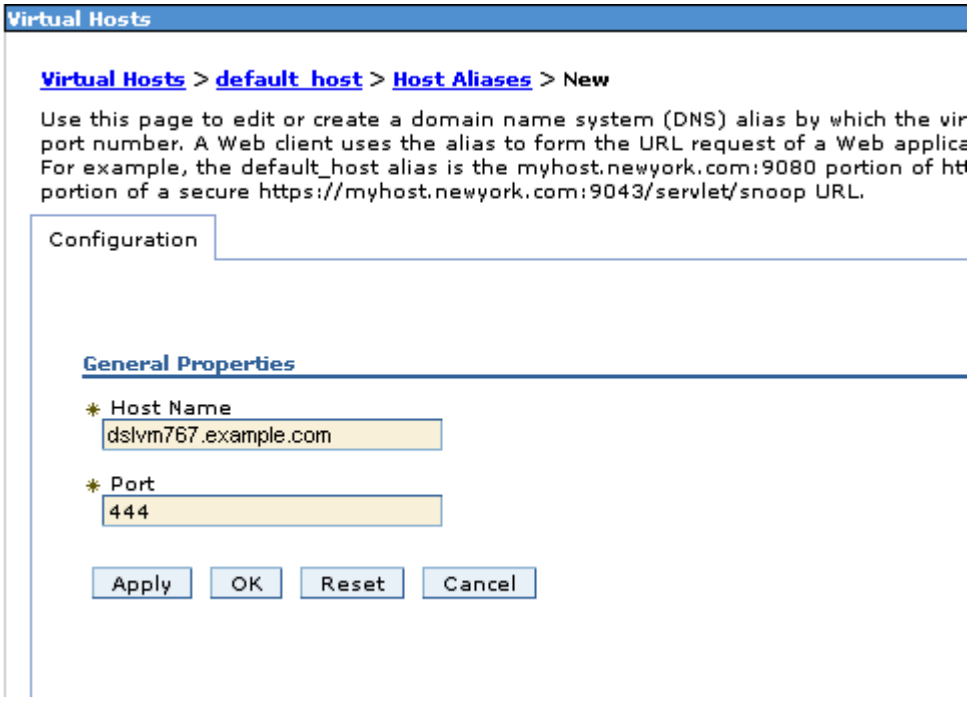


Figure 133. Virtual Hosts

This setting is added:

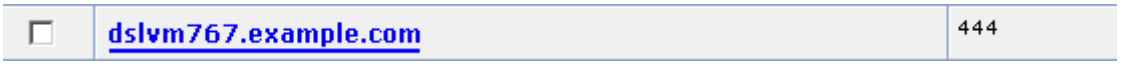


Figure 134. Setting to be added

Note

This causes a new `plugin-cfg.xml` to be generated as there is a change to the Virtual Hosts. Make sure that you have a backup of the `plugin-cfg.xml`.

__ 5. Go to system administration \nodes a. Highlight the nodes \. Click **Full resynchronize**.

Figure 135. Messages

__ 6. If the HTTP server administrator is running, the updated plugin-cfg.xml is copied from `/opt/IBM/WebSphere/AppServer/profiles/Dmgr01/config/cells/connectionsCell01/nodes/webserver1/servers/webserver1` to

/opt/IBM/HTTPServer/Plugins/config/webserver1. If the plugin-cfg.xml was not copied, copy it now.

__ 7. In /opt/IBM/HTTPServer/conf/WebAgent.conf:

__ a. Copy to make a backup of this file.

__ b. Uncomment the LocalConfig.conf location:

localconfigfile="/opt/IBM/HTTPServer/conf/LocalConfig.conf"

__ c. Save and close.

__ 8. In /opt/IBM/HTTPServer/conf/LocalConfig.conf:

__ a. Copy to make a backup of this file.

__ b. Uncomment IgnoreHost="connections.example.com:444". SiteMinder then ignores any traffic through this virtual host to it.

__ c. Comment out ALL other entries in the file. If you do not, the entries might cause confusion with what is set on the SiteMinder policy server.

__ d. Save and close.

__ 9. Start the HTTP server.

__ 10. Verification point:

__ a. If you go to https://connections.example.com:444 you get the HTTP landing page not the SiteMinder page.

__ b. If you go to https://connections.example.com you get the SiteMinder page, only if SPNEGO is not enabled on the SiteMinder policy server.

__ 11. The LotusConnections-config.xml must be updated to update the interservice url and to set the Authenticator:

__ a. On the Deployment Manager, go to /opt/IBM/WebSphere/AppServer/profiles/Dmgr01/config/cells/ConnectionsCell01/LotusConnections-config and edit LotusConnections-config.xml.

__ b. Add “444” to all entries of interService URL.

Your entries are changed from <sloc:interService href="https://connections.example.com"/>:

Figure 136. Changing the entries

To <sloc:interService href="https://connections.example.com:444"/>:

Figure 137. Changing the entries

__ c. Check the customAuthenticator is set to default: <customAuthenticator name="DefaultAuthenticator"/>.

__ d. Save and close.

__ 12. Stop the config: Connections, nodes and the Deployment Manager.

__ 13. Restart the Deployment Manager and the nodes. Allow them to synch to copy the LotusConnections-config.xml to the nodes.

__ 14. Start Connections.

__ 15. Verification point:

__ a. Check a user can access the config. Try to go to https://connections.example.com/homepage. You should be able to log in through SiteMinder and do some testing.

Config is now enabled for SiteMinder and SPNEGO over https.

__ b. After all that, here is an example of working HTTP files:

LocalConfig.conf:

```
# LocalConfig.conf: sample local configuration file for SiteMinder Web
Agents
#
# Make a copy of this file and modify that copy with desired local
configuration settings.
# '#' is used as a comment character at the beginning of a line.
Values commented out
# can be uncommented once proper values are specified.  Many such
values in this
# sample file are verbose explanations of what values should be used
and not the
# values themselves.  To uncomment a line simply remove the '#' from
the beginning
# of the line.
#
# Most parameters in this file are also valid in an Agent Configuration
Object.
# The exceptions are AgentConfigObject, EnableWebAgent, and
HostConfigFile.
#
AcceptTPCookie="NO"
#AgentName="<Agent Name>,<IPAddress>"
#AppendIISServerLog="NO"For IIS and SharePoint
#BadCSSChars="<,>,',;"
#BadQueryChars=""
#BadUrlChars="//,./,/.,/*,*.,~,\,%00-%1f,%7f-%ff,%25"
#BadFormChars="<,>,&,%22"
#CacheAnonymous="NO"
#CCCExt=".ccc"
#CookieDomain=""
#CookieDomainScope="0"
#CookieProvider="<cookie provider URL>"
#CSSChecking="YES"
#CSSErrorFile="<File path to error text, or URL to redirect to>"
#DecodeQueryData="NO"
#DefaultAgentName=""
#DefaultPassword="NO"For IIS and SharePoint
#DisableAuthSrcVars="NO"
#DisableDotDotRule=NO
#DisablePostDataLimit="NO"For IIS and SharePoint
#DisableSessionVars="NO"
#DisableUserNameVars="NO"
#EnableAuditing="NO"
#EnableFormCache="YES"
#EnableMonitoring="NO"
#EnforceRealmTimeouts="NO"
```



```
#ExpireForProxy="NO"
#FccCompatMode="NO"
#FCCExt=".fcc"
#FCCForceIsProtected="YES"
#ForceCookieDomain="NO"
#ForceFQHost="NO"
#ForceIISProxyUser="NO" IIS ONLY
#FormCacheTimeout="600"
#HTTPHeaderEncodingSpec=""
#IgnoreExt=".ccc"
#IgnoreQueryData="NO"
IgnoreHost="connections.example.com:444"
#IgnoreUrl="<URL to ignore>"
#LegacyVariables="NO"
#LogAppend="NO"
#LogFile="NO"
#LogFileName="<File Path to write log to>"
#LogOffUri="<Your Logoff Uri>"
#MaxResourceCacheSize="700"
#MaxSessionCacheSize="700"
#MaxUrlSize="4096"
#NTCExt=".ntc"
#OverrideIgnoreExtFilter=""
#P3PCompactPolicy=""IIS ONLY
#PersistentCookies="NO"
#PersistentIPCheck="YES"
#PreserveHeaders="NO"
#ProxyAgent="NO"Apache 2.0 ONLY
#ProxyTrust="NO"
#ProxyTimeout="NO"Apache 2.0 ONLY
#PSPollInterval="30"
#RemoteUserVar=""
#ReqCookieErrorFile="<File path to error text, or URL to redirect to>"
#RequireCookies="YES"
#ResourceCacheTimeout="600"
#SaveCredsTimeout="720"
#SCCExt=".scc"
#ServerErrorFile="<File path to error text, or URL to redirect to>"
#SPAuthenticatedGroup="SMAuthenticatedGroup"SharePoint only
#SPCacheEntryExpireMinute="30"SharePoint only
#SPDisambiguateGroup="NO"SharePoint only
#SPDisambiguateGroupRule="$groupname{$directoryname}" SharePoint only
#SPDisambiguateUser="YES"SharePoint only
#SPDisambiguateUserRule="$username{$directoryname}" SharePoint only
#SPEnableImpersonation=SharePoint only
#SPFormsTimeOut="30"SharePoint only
#SPImpersonateResponseVarName=SharePoint only
#SPIncludeMySiteSSP=SharePoint only
```

```
#SPNumCacheItem="1000"SharePoint only
#SPPersonalSiteTemplate=SharePoint only
#SPSortVirtualAttribute="UniversalID"SharePoint only
#SPToolsLogLocation=SharePoint only
#SPVirtualAttributeMapList="email=Emailgroup=GroupIDusername=Universal
IDdisplayname=DisplayName" SharePoint only
#SessionGracePeriod="30"
#SessionUpdatePeriod="60"
#SetRemoteUser="NO"
#SFCCExt=".sfcc"
#SSOZoneName="SM"
#SSOTrustedZone="SM"
#TraceAppend="NO"
#TraceConfigFile="<Path to WebAgentTrace.conf file>"
#TraceFile="NO"
#TraceFileName="<File Path to write trace log to>"
#TransientIDCookies="NO"
#TransientIPCheck="NO"
#UseAnonAccess="NO"For IIS and SharePoint
#UseSecureCookies="NO"
##############################################################################

#Newly Added Parameters: Mar 23rd, 2010

#AllowCacheHeaders="NO"
#ConstructFullPwsvcURL="NO"
#EnforcePolicies="YES"
#LogFileSize="0"
#SecureApps="NO"
#TargetAsRelativeURI="NO"
#TraceDelimiter=""
#TraceFileSize="0"
#TraceFormat="default"
#CookiePath="/"
#CookiePathScope="0"
#CookieValidationPeriod=""
#Custom401ErrorFile=""
#CustomIpHeader=""
#EncryptAgentName="YES"
#ExpiredCookieURL="<URL to redirect to>"
#FCCCompatMode=""
#IdleTimeoutURL="<URL to redirect to>"
#IgnoreCPFornotprotected="NO"
#LegacyCookieProvider="NO"
#LegacyEncoding=""
#LogLocalTime="YES"
#MasterCookiePath="/"
#MaxTimeoutURL="<URL to redirect to>"
```



```
#OverlookSessionForMethods=""
#OverlookSessionForMethodUri=""
#OverlookSessionForUrls=""
#PreservePostData="YES"
#SecureURLs="NO"
#UseServerRequestIp="NO"
#ValidTargetDomain=""
#ConformToRFC2047="YES"
#AgentNamesAreFQHostNames="NO"
#4xcompatmode=""
#autoauthorizeoptions=""
#defaulthostname=""
#DisableDNSLookups=""
#disallowutf8noncanonical=""
#enableaccounting=""
#enablentchallengeresponse=""
#forcegetsessiondata=""
#httpserviceprincipal=""
#IgnoreXMLSDK=""
#kccext=""
#LegacyPostPreservationEncoding="NO"
#legacytransferencoding=""
#legalhostnamechars=""
#LowerCaseHTTP="YES"
#LowerCaseProtocolSpecifier="NO"
#LogFilesToKeep="0"
#PostPreservationFile=""
#ProxyDefinition=""
#ProxyHeadersAutoAuth=""
#ProxyHeadersAutoAuth10=""
#ProxyHeadersDefaultTime=""
#ProxyHeadersProtected=""
#ProxyHeadersProtected10=""
#ProxyHeadersTimeoutPercentage=""
#ProxyHeadersUnprotected=""
#ProxyHeadersUnprotected10=""
#ServerPath=""
#smpsserviceprincipal=""
#sharedsecret=""
#StoreSessioninServer="NO"
#TraceFilesToKeep="0"
#TrackSessionDomain="NO"
#UseHTTPOnlyCookies="NO"
#UseSecureCPCookies="NO"
#usesessionforanonymous=""


#For Apache
```

```
#DeleteCerts="NO"
#GetPortFromHeaders="NO"
#HttpsPorts=""

#For IIS

#DefaultUserName=""
#InsecureServer="NO"


#For Domino

#DominoDefaultUser=""
#DominoSuperUser=""
#SkipDominoAuth=""
#UseDominoUserForUnprotected=""
#dominoautoauthnsfresources=""
#dominofinalizefilter=""
#DominoLegacyDocumentSupport="NO"
#DominoLookUpHeaderForLogin="NO"
#DominoMapUrlForRedirect="YES"
#DominoNormalizeUrls="YES"
#DominoUseHeaderForLogin=""
```


httpd.conf:

```
# This is the main IBM HTTP server configuration file.  It contains the
# configuration directives that give the server its instructions.
# See <URL:http://publib.boulder.ibm.com/httpserv/manual70/> for
detailed
# information about the Apache directives.
#
# The instructions provided in this configuration file are only hints
or
# reminders.  Consult the online docs for definitive information.
#
# The configuration directives are grouped into three basic sections:
#  1. Directives that control the operation of the web server process
as a
#     whole (the 'global environment').
#  2. Directives that define the parameters of the 'main' or 'default'
server,
#     which responds to requests that aren't handled by a virtual host.
#     These directives also provide default values for the settings
#     of all virtual hosts.
#  3. Settings for virtual hosts, which allow Web requests to be sent
to
#     different IP addresses or hostnames and have them handled by the
#     same web server process.
#
# Configuration and logfile names: If the filenames you specify for
many
# of the server's control files begin with "/" (or "drive:/" for
Win32), the
# server will use that explicit path.  If the filenames do *not* begin
# with "/", the value of ServerRoot is prepended -- so "logs/foo.log"
# with ServerRoot set to "/opt/IBM/HTTPServer" will be interpreted by
the
# server as "/opt/IBM/HTTPServer/logs/foo.log".
#

### Section 1: Global Environment
#
# The directives in this section affect the overall operation of IBM
HTTP
# Server, such as the number of concurrent requests it can handle or
where
# it can find its configuration files.
#

#
# ServerRoot: The top of the directory tree under which the server's
```

```
# configuration, error, and log files are kept.
#
# Do NOT add a slash at the end of the directory path.
#
ServerRoot "/opt/IBM/HTTPServer"

#
# The accept serialization lock file MUST BE STORED ON A LOCAL DISK.
#
#LockFile logs/accept.lock

#
# PidFile: The file in which the server should record its process
# identification number when it starts.
#
PidFile logs/httpd.pid

#
# Timeout: The number of seconds before receives and sends time out.
#
Timeout 300

#
# KeepAlive: Whether or not to allow persistent connections (more than
# one request per connection). Set to "Off" to deactivate.
#
KeepAlive On

#
# MaxKeepAliveRequests: The maximum number of requests to allow
# during a persistent connection. Set to 0 to allow an unlimited
amount.
# We recommend you leave this number high, for maximum performance.
#
MaxKeepAliveRequests 100

#
# KeepAliveTimeout: Number of seconds to wait for the next request from
the
# same client on the same connection.
#
KeepAliveTimeout 10

##
## Server-Pool Size Regulation (MPM specific)
##

# worker MPM
```



```
#
# For tuning recommendations, refer to <NEWINFOCENTERURL>.
#
# ThreadLimit: maximum setting of ThreadsPerChild
# ServerLimit: maximum setting of StartServers
# StartServers: initial number of server processes to start
# MaxClients: maximum number of simultaneous client connections
# MinSpareThreads: minimum number of worker threads which are kept
spare
# MaxSpareThreads: maximum number of worker threads which are kept
spare
# ThreadsPerChild: constant number of worker threads in each server
process
# MaxRequestsPerChild: maximum number of requests a server process
serves
<IfModule worker.c>
ThreadLimit         25
ServerLimit         64
StartServers         1
MaxClients         600
MinSpareThreads     25
MaxSpareThreads     75
ThreadsPerChild     25
MaxRequestsPerChild  0
</IfModule>

#
# Listen: Allows you to bind the web server to specific IP addresses
# and/or ports, in addition to the default. See also the <VirtualHost>
# directive.
#
# Change this to Listen on specific IP addresses as shown below to
# prevent the web server from accepting connections on all interfaces
# (0.0.0.0)
#
# Change this to "Listen 0.0.0.0:port" to restrict the server to
# IPv4.
#
#Listen 12.34.56.78:80
Listen 80

#
# Dynamic Shared Object (DSO) Support
#
# To be able to use the functionality of a module which was built as a
DSO you
# have to place corresponding `LoadModule' lines at this location so
the
```

```
# directives contained in it are actually available _before_ they are
used.
# Statically compiled modules (those listed by `httpd -l') do not need
# to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
LoadModule sm_module "/opt/netegrity/webagent/bin/libmod_sm22.so"
SmInitFile "/opt/IBM/HTTPServer/conf/WebAgent.conf"
LoadModule authz_host_module modules/mod_authz_host.so
LoadModule auth_basic_module modules/mod_auth_basic.so
LoadModule authn_file_module modules/mod_authn_file.so
LoadModule authz_user_module modules/mod_authz_user.so
#LoadModule authz_groupfile_module modules/mod_authz_groupfile.so
LoadModule include_module modules/mod_include.so
LoadModule log_config_module modules/mod_log_config.so
LoadModule ibm_local_redirect_module modules/mod_ibm_local_redirect.so
LoadModule env_module modules/mod_env.so
#LoadModule mime_magic_module modules/mod_mime_magic.so
#LoadModule expires_module modules/mod_expires.so
#LoadModule headers_module modules/mod_headers.so
LoadModule unique_id_module modules/mod_unique_id.so
LoadModule setenvif_module modules/mod_setenvif.so
#LoadModule proxy_module modules/mod_proxy.so
#LoadModule proxy_connect_module modules/mod_proxy_connect.so
#LoadModule proxy_ftp_module modules/mod_proxy_ftp.so
#LoadModule proxy_http_module modules/mod_proxy_http.so
LoadModule mime_module modules/mod_mime.so
#LoadModule dav_module modules/mod_dav.so
#LoadModule dav_fs_module modules/mod_dav_fs.so
LoadModule autoindex_module modules/mod_autoindex.so
#LoadModule asis_module modules/mod_asis.so
#LoadModule info_module modules/mod_info.so
LoadModule cgid_module modules/mod_cgid.so
LoadModule dir_module modules/mod_dir.so
LoadModule actions_module modules/mod_actions.so
#LoadModule speling_module modules/mod_speling.so
#LoadModule userdir_module modules/mod_userdir.so
LoadModule alias_module modules/mod_alias.so
LoadModule rewrite_module modules/mod_rewrite.so
#LoadModule deflate_module modules/mod_deflate.so

#
# ExtendedStatus controls whether the web server will generate "full"
# status information (ExtendedStatus On) or just basic information
# (ExtendedStatus Off) when the server status page is formatted or
# when IBM HTTP Server diagnostic modules report information.  The
# default is Off.
```



```
#
LoadModule status_module modules/mod_status.so
<IfModule mod_status.c>
ExtendedStatus On
</IfModule>

### Section 2: 'Main' server configuration
#
# The directives in this section set up the values used by the 'main'
# server, which responds to any requests that aren't handled by a
# <VirtualHost> definition.  These values also provide defaults for
# any <VirtualHost> containers you may define later in the file.
#
# All of these directives may appear inside <VirtualHost> containers,
# in which case these default settings will be overridden for the
# virtual host being defined.
#

#
# If you wish httpd to run as a different user or group, you must run
# httpd as root initially and it will switch.
#
User nobody
Group nobody

#
# ServerAdmin: Your address, where problems with the server should be
# e-mailed.  This address appears on some server-generated pages, such
# as error documents.  e.g. admin@your-domain.com
#
ServerAdmin you@your.address

#
# ServerName gives the name and port that the server uses to identify
itself.
# This can often be determined automatically, but we recommend you
specify
# it explicitly to prevent problems during startup.
#
# If this is not set to valid DNS name for your host, server-generated
# redirections will not work.  See also the UseCanonicalName directive.
#
# If your host doesn't have a registered DNS name, enter its IP address
here.
# You will have to access it by its address anyway, and this will make
# redirections work in a sensible way.
#
ServerName connections.example.com:80
```

```
#
# UseCanonicalName: Determines how the web server constructs self-
# referencing URLs and the SERVER_NAME and SERVER_PORT variables.
# When set "Off", the web server will use the Hostname and Port
supplied
# by the client.  When set "On", it will use the value of the
ServerName
# directive.
#
UseCanonicalName Off

#
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory,
but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "/opt/IBM/HTTPServer/htdocs"

#
# Each directory to which the web server has access can be configured
# with respect to which services and features are allowed and/or
disabled
# in that directory (and its subdirectories).
#
# First, we configure the "default" to be a very restrictive set of
# features.
#
<Directory />
    Options FollowSymLinks
    AllowOverride None
</Directory>

#
# Note that from this point forward you must specifically allow
# particular features to be enabled: so if something's not working as
# you might expect, make sure that you have specifically enabled it
# below.
#

#
# This should be changed to whatever you set DocumentRoot to.
#
<Directory "/opt/IBM/HTTPServer/htdocs">

#
# Possible values for the Options directive are "None", "All",
```



```
# or any combination of:
#   Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI
Multiviews
#
# Note that "MultiViews" must be named *explicitly* --- "Options All"
# doesn't give it to you.
#
# The Options directive is both complicated and important.  Please see
#
http://publib.boulder.ibm.com/httpserv/manual70/mod/core.html#options
# for more information.
#
    Options Indexes FollowSymLinks

#
# AllowOverride controls what directives may be placed in .htaccess
files.
# It can be "All", "None", or any combination of the keywords:
#   Options FileInfo AuthConfig Limit
#
    AllowOverride None

#
# Controls who can get stuff from this server.
#
    Order allow,deny
    Allow from all

</Directory>

#
# UserDir: The name of the directory that is appended onto a user's
home
# directory if a ~user request is received.
#
<IfModule mod_userdir.c>
UserDir public_html

#
# Control access to UserDir directories.  The following is an example
# for a site where these directories are restricted to read-only.
#
#<Directory /home/*/public_html>
#    AllowOverride FileInfo AuthConfig Limit Indexes
#    Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec
#    <Limit GET POST OPTIONS PROPFIND>
#        Order allow,deny
#        Allow from all
```

```
#    </Limit>
#    <LimitExcept GET POST OPTIONS PROPFIND>
#        Order deny,allow
#        Deny from all
#    </LimitExcept>
#</Directory>
</IfModule>

#
# DirectoryIndex: sets the file that the web server will serve if a
# directory is requested.
#
# The index.html.var file (a type-map) is used to deliver content-
# negotiated documents.  The MultiViews Option can be used for the
# same purpose, but it is much slower.
#
DirectoryIndex index.html index.html.var

#
# AccessFileName: The name of the file to look for in each directory
# for additional configuration directives.  See also the AllowOverride
# directive.
#
AccessFileName .htaccess

#
# The following lines prevent .htaccess and .htpasswd files from being
# viewed by Web clients.
#
<Files ~ "^\.ht">
    Order allow,deny
    Deny from all
</Files>

#
# TypesConfig describes where the mime.types file (or equivalent) is
# to be found.
#
TypesConfig conf/mime.types

#
# DefaultType is the default MIME type the server will use for a
document
# if it cannot otherwise determine one, such as from filename
extensions.
# If your server contains mostly text or HTML documents, "text/plain"
is
# a good value.  If most of your content is binary, such as
```



```
applications
# or images, you may want to use "application/octet-stream" instead to
# keep browsers from trying to display binary files as though they are
# text.
#
DefaultType text/plain

#
# The mod_mime_magic module allows the server to use various hints from
the
# contents of the file itself to determine its type.  The MIMEMagicFile
# directive tells the module where the hint definitions are located.
#
<IfModule mod_mime_magic.c>
    MIMEMagicFile conf/magic
</IfModule>

#
# HostnameLookups: Log the names of clients or just their IP addresses
# e.g., www.apache.org (on) or 204.62.129.132 (off).
# The default is off because it'd be overall better for the net if
people
# had to knowingly turn this feature on, since enabling it means that
# each client request will result in AT LEAST one lookup request to the
# nameserver.
#
HostnameLookups Off

#
# EnableMMAP: Control whether memory-mapping is used to deliver
# files (assuming that the underlying OS supports it).
# The default is on; turn this off if you serve from NFS-mounted
# filesystems.  On some systems, turning it off (regardless of
# filesystem) can improve performance; for details, please see
# http://httpd.apache.org/docs/2.2/mod/core.html#enablemmap
#
# EnableMMAP off

#
# EnableSendfile: Control whether the sendfile kernel support is
# used  to deliver files (assuming that the OS supports it).
# The default is on; turn this off if you serve from NFS-mounted
# filesystems.  Please see
# http://httpd.apache.org/docs/2.2/mod/core.html#enablesendfile
#
EnableSendfile off

#
```

```
# ErrorLog: The location of the error log file.
# If you do not specify an ErrorLog directive within a <VirtualHost>
# container, error messages relating to that virtual host will be
# logged here.  If you *do* define an error logfile for a <VirtualHost>
# container, that host's errors will be logged there and not here.
#
ErrorLog logs/error_log

#
# LogLevel: Control the number of messages logged to the error log.
# Possible values include: debug, info, notice, warn, error, crit,
# alert, emerg.
#
LogLevel warn

#
# The following directives define some format nicknames for use with
# a CustomLog directive (see below).
#
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\"
\"%{User-Agent}i\"" combined
LogFormat "%h %l %u %t \"%r\" %>s %b" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent

#
# The location and format of the access logfile (Common Logfile
Format).
# If you do not define any access logfiles within a <VirtualHost>
# container, they will be logged here.  Contrariwise, if you *do*
# define per-<VirtualHost> access logfiles, transactions will be
# logged therein and *not* in this file.
#
CustomLog logs/access_log common

#
# If you would like to have agent and referer logfiles, uncomment the
# following directives.
#
#CustomLog logs/referer_log referer
#CustomLog logs/agent_log agent

#
# If you prefer a single logfile with access, agent, and referer
information
# (Combined Logfile Format) you can use the following directive.
#
#CustomLog logs/access_log combined
```



```
#
# AddServerHeader
# This directive provides a means to enable or disable ServerHeader
values.
# The default value is "On". This provides a server header according to
the
# values specified in the ServerTokens and ServerSignature directives.
# Setting this directive to "Off" results in no server header
information
# being returned to clients.
# Set to one of: On | Off
#
#AddServerHeader Off

#
# ServerTokens
# This directive configures what you return as the Server HTTP response
# Header. The built-in default is 'Full' which sends information about
# the OS-type and compiled in modules.  The recommended value is 'Prod'
# which sends the least information.
# Set to one of:  Full | OS | Minor | Minimal | Major | Prod
# where Full conveys the most information, and Prod the least.
#
ServerTokens Prod

#
# Optionally add a line containing the server version and virtual host
# name to server-generated pages (internal error documents, FTP
directory
# listings, mod_status and mod_info output etc., but not CGI generated
# documents or custom error documents).
# Set to "EMail" to also include a mailto: link to the ServerAdmin.
# Set to one of:  On | Off | EMail
#
ServerSignature On

#
# Aliases: Add here as many aliases as you need (with no limit). The
format is
# Alias fakename realname
#
# Note that if you include a trailing / on fakename then the server
will
# require it to be present in the URL.  So "/icons" isn't aliased in
this
# example, only "/icons/".  If the fakename is slash-terminated, then
the
```

```
# realname must also be slash terminated, and if the fakename omits the
# trailing slash, the realname must also omit it.
#
# We include the /icons/ alias for FancyIndexed directory listings.  If
you
# do not use FancyIndexing, you may comment this out.
#
Alias /siteminderagent/pwcgi/ "/opt/netegrity/webagent/pw/"
<Directory "/opt/netegrity/webagent/pw/">
Options Indexes MultiViews ExecCGI
AllowOverride None
Order allow,deny
Allow from all
</Directory>
Alias /siteminderagent/pw/ "/opt/netegrity/webagent/pw/"
<Directory "/opt/netegrity/webagent/pw/">
Options Indexes MultiViews ExecCGI
AllowOverride None
Order allow,deny
Allow from all
</Directory>
Alias /siteminderagent/ "/opt/netegrity/webagent/samples/"
<Directory "/opt/netegrity/webagent/samples/">
Options Indexes MultiViews
AllowOverride None
Order allow,deny
Allow from all
</Directory>
Alias /icons/ "/opt/IBM/HTTPServer/icons/"

<Directory "/opt/IBM/HTTPServer/icons">
    Options Indexes MultiViews
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>

#
# ScriptAlias: This controls which directories contain server scripts.
# ScriptAliases are essentially the same as Aliases, except that
# documents in the realname directory are treated as applications and
# run by the server when requested rather than as documents sent to the
client.
# The same rules about trailing "/" apply to ScriptAlias directives as
to
# Alias.
#
ScriptAlias /cgi-bin/ "/opt/IBM/HTTPServer/cgi-bin/"
```



```
<IfModule mod_cgid.c>
#
# Additional to mod_cgid.c settings, mod_cgid has Scriptsock <path>
# for setting UNIX socket for communicating with cgid.
#
#Scriptsock            logs/cgisock
</IfModule>

#
# "/opt/IBM/HTTPServer/cgi-bin" should be changed to whatever your
ScriptAliased
# CGI directory exists, if you have that configured.
#
<Directory "/opt/IBM/HTTPServer/cgi-bin">
    AllowOverride None
    Options None
    Order allow,deny
    Allow from all
</Directory>

#
# Redirect allows you to tell clients about documents which used to
exist in
# your server's namespace, but do not anymore. This allows you to tell
the
# clients where to look for the relocated document.
# Example:
# Redirect permanent /foo http://www.example.com/bar

#
# Directives controlling the display of server-generated directory
listings.
#

#
# IndexOptions: Controls the appearance of server-generated directory
# listings.
#
IndexOptions FancyIndexing VersionSort

#
# AddIcon* directives tell the server which icon to show for different
# files or filename extensions.  These are only displayed for
# FancyIndexed directories.
#
AddIconByEncoding (CMP,/icons/compressed.gif) x-compress x-gzip
```

```
AddIconByType (TXT,/icons/text.gif) text/*
AddIconByType (IMG,/icons/image2.gif) image/*
AddIconByType (SND,/icons/sound2.gif) audio/*
AddIconByType (VID,/icons/movie.gif) video/*

AddIcon /icons/binary.gif .bin .exe
AddIcon /icons/binhex.gif .hqx
AddIcon /icons/tar.gif .tar
AddIcon /icons/world2.gif .wrl .wrl.gz .vrml .vrm .iv
AddIcon /icons/compressed.gif .Z .z .tgz .gz .zip
AddIcon /icons/a.gif .ps .ai .eps
AddIcon /icons/layout.gif .html .shtml .htm .pdf
AddIcon /icons/text.gif .txt
AddIcon /icons/c.gif .c
AddIcon /icons/p.gif .pl .py
AddIcon /icons/f.gif .for
AddIcon /icons/dvi.gif .dvi
AddIcon /icons/uuencoded.gif .uu
AddIcon /icons/script.gif .conf .sh .shar .csh .ksh .tcl
AddIcon /icons/tex.gif .tex
AddIcon /icons/bomb.gif core

AddIcon /icons/back.gif ..
AddIcon /icons/hand.right.gif README
AddIcon /icons/folder.gif ^^DIRECTORY^^
AddIcon /icons/blank.gif ^^BLANKICON^^

#
# DefaultIcon is which icon to show for files which do not have an icon
# explicitly set.
#
DefaultIcon /icons/unknown.gif

#
# AddDescription allows you to place a short description after a file
in
# server-generated indexes.  These are only displayed for FancyIndexed
# directories.
# Format: AddDescription "description" filename
#
#AddDescription "GZIP compressed document" .gz
#AddDescription "tar archive" .tar
#AddDescription "GZIP compressed tar archive" .tgz

#
# ReadmeName is the name of the README file the server will look for by
# default, and append to directory listings.
#
```



```
# HeaderName is the name of a file which should be prepended to
# directory indexes.
ReadmeName README.html
HeaderName HEADER.html

#
# IndexIgnore is a set of filenames which directory indexing should
ignore
# and not include in the listing.  Shell-style wildcarding is
permitted.
#
IndexIgnore .??* *~ *# HEADER* README* RCS CVS *,v *,t

#
# AddEncoding allows you to have certain browsers (Mosaic/X 2.1+)
uncompress
# information on the fly. Note: Not all browsers support this.
# Despite the name similarity, the following Add* directives have
nothing
# to do with the FancyIndexing customization directives above.
#
AddEncoding x-compress Z
AddEncoding x-gzip gz tgz

#
# AddType allows you to add to or override the MIME configuration
# file mime.types for specific file types.
#
AddType application/x-tar .tgz
AddType image/x-icon .ico

#
# AddHandler allows you to map certain file extensions to "handlers":
# actions unrelated to filetype. These can be either built into the
server
# or added with the Action directive (see below)
#
# To use CGI scripts outside of ScriptAliased directories:
# (You will also need to add "ExecCGI" to the "Options" directive.)
#
#AddHandler cgi-script .cgi

#
# For files that include their own HTTP headers:
#
#AddHandler send-as-is asis

#
```

```
# For server-parsed imagemap files:
#
#AddHandler imap-file map

#
# For type maps (negotiated resources):
#
#AddHandler type-map var

#
# Filters allow you to process content before it is sent to the client.
#
# To parse .shtml files for server-side includes (SSI):
# (You will also need to add "Includes" to the "Options" directive.)
#
#AddType text/html .shtml
#AddOutputFilter INCLUDES .shtml

#
# Action lets you define media types that will execute a script
whenever
# a matching file is called. This eliminates the need for repeated URL
# pathnames for oft-used CGI file processors.
# Format: Action media/type /cgi-script/location
# Format: Action handler-name /cgi-script/location
#

#
# Customizable error responses come in three flavors:
# 1) plain text 2) local redirects 3) external redirects
#
# Some examples:
#ErrorDocument 500 "The server made a boo boo."
#ErrorDocument 404 /missing.html
#ErrorDocument 404 "/cgi-bin/missing_handler.pl"
#ErrorDocument 402 http://www.example.com/subscription_info.html
#

#
# Putting this all together, we can internationalize error responses.
#
# We use Alias to redirect any /error/HTTP_<error>.html.var response
to
# our collection of by-error message multi-language collections.  We
use
# includes to substitute the appropriate text.
#
# You can modify the messages' appearance without changing any of the
```



```
# default HTTP_<error>.html.var files by adding the line:
#
#   Alias /error/include/ "/your/include/path/"
#
# which allows you to create your own set of files by starting with the
# /opt/IBM/HTTPServer/error/include/ files and copying them to
/your/include/path/,
# even on a per-VirtualHost basis.  The default include files will
display
# your IBM HTTP Server version number and your ServerAdmin email
address
# regardless of the setting of ServerSignature.
#
# The internationalized error documents require mod_alias, mod_include
# and mod_negotiation.  To activate them, uncomment the following 30
lines.

#    Alias /error/ "/opt/IBM/HTTPServer/error/"
#
#    <Directory "/opt/IBM/HTTPServer/error">
#        AllowOverride None
#        Options IncludesNoExec
#        AddOutputFilter Includes html
#        AddHandler type-map var
#        Order allow,deny
#        Allow from all
#        LanguagePriority en de es fr it nl sv
#        ForceLanguagePriority Prefer Fallback
#    </Directory>
#
#    ErrorDocument 400 /error/HTTP_BAD_REQUEST.html.var
#    ErrorDocument 401 /error/HTTP_UNAUTHORIZED.html.var
#    ErrorDocument 403 /error/HTTP_FORBIDDEN.html.var
#    ErrorDocument 404 /error/HTTP_NOT_FOUND.html.var
#    ErrorDocument 405 /error/HTTP_METHOD_NOT_ALLOWED.html.var
#    ErrorDocument 408 /error/HTTP_REQUEST_TIME_OUT.html.var
#    ErrorDocument 410 /error/HTTP_GONE.html.var
#    ErrorDocument 411 /error/HTTP_LENGTH_REQUIRED.html.var
#    ErrorDocument 412 /error/HTTP_PRECONDITION_FAILED.html.var
#    ErrorDocument 413 /error/HTTP_REQUEST_ENTITY_TOO_LARGE.html.var
#    ErrorDocument 414 /error/HTTP_REQUEST_URI_TOO_LARGE.html.var
#    ErrorDocument 415 /error/HTTP_SERVICE_UNAVAILABLE.html.var
#    ErrorDocument 500 /error/HTTP_INTERNAL_SERVER_ERROR.html.var
#    ErrorDocument 501 /error/HTTP_NOT_IMPLEMENTED.html.var
#    ErrorDocument 502 /error/HTTP_BAD_GATEWAY.html.var
#    ErrorDocument 503 /error/HTTP_SERVICE_UNAVAILABLE.html.var
#    ErrorDocument 506 /error/HTTP_VARIANT_ALSO_VARIES.html.var
```

```
#
# The following directives modify normal HTTP response behavior to
# handle known problems with browser implementations.
#
BrowserMatch "Mozilla/2" nokeepalive
BrowserMatch "MSIE 4\.0b2;" nokeepalive downgrade-1.0
force-response-1.0
BrowserMatch "RealPlayer 4\.0" force-response-1.0
BrowserMatch "Java/1\.0" force-response-1.0
BrowserMatch "JDK/1\.0" force-response-1.0

#
# The following directive disables redirects on non-GET requests for
# a directory that does not include the trailing slash.  This fixes a
# problem with Microsoft WebFolders which does not appropriately
handle
# redirects for folders with DAV methods.
# Same deal with Apple's DAV filesystem and Gnome VFS support for DAV.
#
BrowserMatch "Microsoft Data Access Internet Publishing Provider"
redirect-carefully
BrowserMatch "^WebDrive" redirect-carefully
BrowserMatch "^WebDAVFS/1.[012]" redirect-carefully
BrowserMatch "^gnome-vfs" redirect-carefully

#
# Allow server status reports generated by mod_status,
# with the URL of http://servername/server-status
# Change the ".example.com" to match your domain to enable.
#
<IfModule mod_status.c>
<Location /server-status>
    SetHandler server-status
    Order deny,allow
    Deny from all
# Add an "Allow from" directive to provide access to the server status
page.
#
# Examples:
#
# 1. Allow any client with hostname *.example.com to view the page.
#
# Allow from .example.com
#
# 2. Allow the local machine to view the page using the loopback
address.
#
# Allow from 127.0.0.1
```



```
#
# 3. Allow any machine on the local network to view the page.
#
# Allow from 192.168.1
</Location>
</IfModule>

#
# Allow remote server configuration reports, with the URL of
#  http://servername/server-info (requires that mod_info.c be loaded).
# Change the ".example.com" to match your domain to enable.
#
#<Location /server-info>
#    SetHandler server-info
#    Order deny,allow
#    Deny from all
#    Allow from .example.com
#</Location>

#
# Proxy Server directives. Uncomment the following lines to
# enable the proxy server:
#
#<IfModule mod_proxy.c>

#Enable the forward proxy server. Note: Do not use the ProxyRequests
directive if
#all you require is reverse proxy.
#
#ProxyRequests On
#
#<Proxy *>
#    Order deny,allow
#    Deny from all
#    Allow from .example.com
#</Proxy>

#
# Enable/disable the handling of HTTP/1.1 "Via:" headers.
# ("Full" adds the server version; "Block" removes all outgoing Via:
headers)
# Set to one of: Off | On | Full | Block
#
#ProxyVia On

#</IfModule>
# End of proxy directives.
```

```
### Section 3: Virtual Hosts
#
# VirtualHost: If you want to maintain multiple domains/hostnames on
your
# machine you can setup VirtualHost containers for them. Most
configurations
# use only name-based virtual hosts so the server doesn't need to worry
about
# IP addresses. This is indicated by the asterisks in the directives
below.
#
# Please see the documentation at
# <URL:http://httpd.apache.org/docs/2.2/vhosts/>
# for further details before you try to setup virtual hosts.
#
# You may use the command line option '-S' to verify your virtual host
# configuration.

#
# Use name-based virtual hosting.
#
#NameVirtualHost *

#
# VirtualHost example:
# Almost any Apache directive may go into a VirtualHost container.
# The first VirtualHost section is used for requests without a known
# server name.
#
#<VirtualHost *>
#    ServerAdmin webmaster@dummy-host.example.com
#    DocumentRoot /www/docs/dummy-host.example.com
#    ServerName dummy-host.example.com
#    ErrorLog logs/dummy-host.example.com-error_log
#    CustomLog logs/dummy-host.example.com-access_log common
#</VirtualHost>

# Example SSL configuration which supports SSLv3 and TLSv1
# To enable this support:
#   1) Create a key database with ikeyman
#   2) Update the KeyFile directive below to point to that key database
#   3) Uncomment the directives up through the end of the example
#
#LoadModule ibm_ssl_module modules/mod_ibm_ssl.so
#Listen 443
#<VirtualHost *:443>
#SSLEnable
#SSLProtocolDisable SSLv2
```



```
#</VirtualHost>
#KeyFile /opt/IBM/HTTPServer/ihsserverkey.kdb
#SSLDisable
# End of example SSL configuration


#
# Enable IBM HTTP Server diagnostic features.
#
# CoreDumpDirectory directory: Sets the location where the server will
# attempt to put a core dump.  The child processes running as 'User'
# (see User config directive above) must have permission to write to
# this directive.  The filesystem will have to be large enough to hold
# potentially large core files.
#
# The /tmp directory is often sufficient.
#
#CoreDumpDirectory /tmp

# mod_mpmstats logs statistics about server activity to the main
# error log.  No records are written while the server is idle.
LoadModule mpmstats_module modules/debug/mod_mpmstats.so
<IfModule mod_mpmstats.c>
# Write a record every 10 minutes (if server isn't idle).
# Recommendation: Lower this interval to 60 seconds, which will
# result in the error log growing faster but with more accurate
# information about server load.
ReportInterval 600
# Include details of active module in the statistics.
TrackModules On
</IfModule>

# EnableExceptionHook allows modules such as mod_backtrace and
# mod_whatkilledus to run after a crash and gather additional
# diagnostic information.
# EnableExceptionHook must be "on" in order to use mod_backtrace or
# mod_whatkilledus.
EnableExceptionHook On

# mod_backtrace will record a backtrace of the crashing thread to the
# error log at the time of a crash.  This is important information for
# diagnosing the cause of the crash.
LoadModule backtrace_module modules/debug/mod_backtrace.so

# mod_whatkilledus will record information about the current request
# and connection to the error log at the time of a crash.  This is
# important information for diagnosing the cause of the crash.
LoadModule whatkilledus_module modules/debug/mod_whatkilledus.so
```

```
# mod_net_trace will record actual data sent/received from the client
# and on proxy connections, even for SSL connections.  Unlike an IP
# trace, interaction with the platform network APIs can be seen.
# The following example configuration can be activated by uncommenting
# the LoadModule directive.
#LoadModule net_trace_module modules/debug/mod_net_trace.so
<IfModule mod_net_trace.c>
NetTraceFile /tmp/nettrace
NetTrace client * dest file event senddata=65535 event recvdata=65535
</IfModule>

LoadModule was_ap22_module
/opt/IBM/HTTPServer/Plugins/bin/mod_was_ap22_http.so
WebSpherePluginConfig
/opt/IBM/HTTPServer/Plugins/config/webserver1/plugin-cfg.xml
#
# Dynamic Shared Object (DSO) Support
#
# To be able to use the functionality of a module which was built as a
DSO you
# have to place corresponding `LoadModule' lines at this location so
the
# directives contained in it are actually available _before_ they are
used.
# Statically compiled modules (those listed by `httpd -l') do not need
# to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
# Alias: Maps web paths into filesystem paths and is used to
# access content that does not live under the DocumentRoot.
# Example:
# Alias /webpath /full/filesystem/path
#
# AddHandler allows you to map certain file extensions to "handlers":
# actions unrelated to filetype. These can be either built into the
server
# or added with the Action directive (see below)
#
# To use CGI scripts outside of ScriptAliased directories:
# (You will also need to add "ExecCGI" to the "Options" directive.)
#
#AddHandler cgi-script .cgi

RewriteEngine On
RewriteCond %{REQUEST_URI} /(.*)/ibm_security_logout(.*)
```



```
RewriteCond %{QUERY_STRING}
!=logoutExitPage=https://connections.example.com/homepage
RewriteRule /(.*)/ibm_security_logout(.*)
/homepage/web/ibm_security_logout?logoutExitPage=https://connections.e
xample.com/homepage [noescape,L,R]

RewriteCond %{REQUEST_URI} !^/blogs/roller-ui/rendering/(.*)
RewriteRule ^/blogs/(.*)/api/(.*)
/blogs/roller-ui/rendering/api/$1/api/$2 [R,L]
RewriteCond %{REQUEST_URI} !^/blogs/roller-ui/rendering/(.*)
RewriteRule ^/blogs/(.*)/feed/tags/atom(.*)
/blogs/roller-ui/rendering/feed/$1/tags/atom/ [R,L]
RewriteCond %{REQUEST_URI} !^/blogs/roller-ui/rendering/(.*)
RewriteRule ^/blogs/(.*)/feed/entries/atom(.*)
/blogs/roller-ui/rendering/feed/$1/entries/atom/ [R,L]
RewriteCond %{REQUEST_URI} !^/blogs/roller-ui/rendering/(.*)
RewriteRule ^/blogs/(.*)/feed/comments/atom(.*)
/blogs/roller-ui/rendering/feed/$1/comments/atom/ [R,L]
RewriteCond %{REQUEST_URI} !^/blogs/roller-ui/rendering/(.*)
RewriteRule ^/blogs/(.*)/feed/blogs/atom(.*)
/blogs/roller-ui/rendering/feed/$1/blogs/atom/ [R,L]

LoadModule ibm_ssl_module modules/mod_ibm_ssl.so
<IfModule mod_ibm_ssl.c>
Listen 0.0.0.0:443
<VirtualHost *:443>
ServerName connections.example.com
SSLEnable
AllowEncodedSlashes On

RewriteEngine On
RewriteCond %{REQUEST_URI} /(.*)/ibm_security_logout(.*)
RewriteCond %{QUERY_STRING}
!=logoutExitPage=https://connections.example.com/homepage
RewriteRule /(.*)/ibm_security_logout(.*)
/homepage/web/ibm_security_logout?logoutExitPage=https://connections.e
xample.com/homepage [noescape,L,R]

RewriteCond %{REQUEST_URI} !^/blogs/roller-ui/rendering/(.*)
RewriteRule ^/blogs/(.*)/api/(.*)
/blogs/roller-ui/rendering/api/$1/api/$2 [R,L]
RewriteCond %{REQUEST_URI} !^/blogs/roller-ui/rendering/(.*)
RewriteRule ^/blogs/(.*)/feed/tags/atom(.*)
/blogs/roller-ui/rendering/feed/$1/tags/atom/ [R,L]
RewriteCond %{REQUEST_URI} !^/blogs/roller-ui/rendering/(.*)
RewriteRule ^/blogs/(.*)/feed/entries/atom(.*)
/blogs/roller-ui/rendering/feed/$1/entries/atom/ [R,L]
RewriteCond %{REQUEST_URI} !^/blogs/roller-ui/rendering/(.*)
```

```
RewriteRule ^/blogs/(.*)/feed/comments/atom(.*)
/blogs/roller-ui/rendering/feed/$1/comments/atom/ [R,L]
RewriteCond %{REQUEST_URI} !^/blogs/roller-ui/rendering/(.*)
RewriteRule ^/blogs/(.*)/feed/blogs/atom(.*)
/blogs/roller-ui/rendering/feed/$1/blogs/atom/ [R,L]

Alias /downloadfiles /opt/IBM/LC_Share/files/upload/
Alias /downloadwikis /opt/IBM/LC_Share/wikis/upload/

<Directory /opt/IBM/LC_Share/files/upload/>
Order Deny,Allow
Deny from all
Allow from env=REDIRECT_FILES_CONTENT
</Directory>

<Directory /opt/IBM/LC_Share/wikis/upload/>
 Order Deny,Allow
Deny from all
 Allow from env=REDIRECT_WIKIS_CONTENT
</Directory>

<Location /files>
 IBMLocalRedirect On
 IBMLocalRedirectKeepHeaders
X-LConn-Auth,Cache-Control,Content-Type,Content-Disposition,Last-Modif
ied,ETag,Content-Language,Set-Cookie
 SetEnv FILES_CONTENT true
</Location>

<Location /wikis>
 IBMLocalRedirect On
 IBMLocalRedirectKeepHeadErs
X-LConn-Auth,Cache-Control,Content-Type,Content-Disposition,Last-Modif
ied,ETag,Content-Language,Set-Cookie
 SetEnv WIKIS_CONTENT true
</Location>


</VirtualHost>
</IfModule>
SSLDisable

Keyfile "/opt/IBM/Keyfiles/webserver-key.kdb"
SSLStashFile "/opt/IBM/Keyfiles/webserver-key.sth"


Listen 444
<VirtualHost *:444>
```



```
    ServerName connections.example.com
    SSLEnable
    Keyfile "/opt/IBM/Keyfiles/webserver-key.kdb"
SSLStashFile "/opt/IBM/Keyfiles/webserver-key.sth"
</VirtualHost>
```

WebAgent.conf

```
# WebAgent.conf: configuration file for SiteMinder Web Agent
# Web Agent Version = 6QMR6, Build = 667, Update = 0

#agentname="<AgentName>, <IPAddress>"
HostConfigFile="/opt/netegrity/webagent/config/SmHost.conf"
AgentConfigObject="dslvm767_wa_conf"

EnableWebAgent="YES"

ServerPath="/opt/IBM/HTTPServer/conf"
localconfigfile="/opt/IBM/HTTPServer/conf/LocalConfig.conf"
LoadPlugin="/opt/netegrity/webagent/bin/libHttpPlugin.so"
#LoadPlugin="/opt/netegrity/webagent/bin/libAffiliate10Plugin.so"
#LoadPlugin="/opt/netegrity/webagent/bin/libSAMLAffiliatePlugin.so"
#LoadPlugin="/opt/netegrity/webagent/bin/libeTSSOPlugin.so"
#LoadPlugin="/opt/netegrity/webagent/bin/libIntroscopePlugin.so"
```