Oracle

WebLogic Server

12.2.1.4.0 and 14.1.1

Integration Guide

CryptoServer HSM

4.50.0.2

utimaco®

Imprint

Copyright 2026	Utimaco IS GmbH Krefelder Straße 220 52070 Aachen Germany
Phone	AMERICAS +1-844-UTIMACO (+1 844-884-6226) EMEA +49 800-627-3081 APAC +81 800-919-1301
Internet	https://support.hsm.utimaco.com/
e-mail	support@utimaco.com
Document Version	1.0.0
Date	2026-05-26
Status	PUBLISHED
Document No.	IG-2026-0046

Table of Contents

1 Introduction

This guide is part of the information and support provided by Utimaco. Additional documentation produced to support your Utimaco SecurityServer product can be found in the document directory of the Utimaco SecurityServer product bundle. All Utimaco SecurityServer product documentation is available from Utimaco's web site at https://utimaco.com/.

1.1 About This Guide

This guide describes how to integrate an Utimaco SecurityServer Hardware Security Module (HSM) with Oracle WebLogic Server. Utimaco HSM securely stores the private key for SSL and offload the cryptographic operations to the HSM.

1.1.1 Target Audience for This Guide

This guide is intended for Oracle WebLogic Server and Utimaco HSM administrators.

1.1.2 Document Conventions

The following conventions are used in this guide:

Convention	Use	Example
Bold	Items of the Graphical User Interface (GUI), e.g., menu options	Select **Details** and click on **Properties** button
`Monospaced`	Code that is given for explanation or as an example, file paths	`certreq.exe -new request.inf IISCertRequest.csr`
Italic	References and important terms	Operating system listed in *Tested Versions*

Table 1: Document conventions

We use special icons to highlight the most important notes and information.

Here you will find important safety information that should be followed.

Here you will find additional notes or supplementary information.

This message marks the result expected after the successful execution of an instruction.

1.1.3 Abbreviations

The following abbreviations are used in this guide:

Abbreviation	Meaning
CA	Certificate Authority
CD	Compact Disc
CSADM	CryptoServer Command-line Administration Tool
CSR	Certificate Signing Request
CXI	Cryptographic eXtended Services Interface
EC	Elliptic Curve
GUI	Graphical User Interface
HSM	Hardware Security Module

Abbreviation	Meaning
IP	Internet Protocol
JCE	Java Cryptographic Engine Provider
JDK	Java Development Kit
JSPs	JavaServer Pages
LAN	Local Area Network
MBK	Master Backup Key
PCIe	PCI Express Interface
PIN	Personal Identification number
RSA	Rivest-Shamir-Adleman
SSL	Secure Socket Layer
URL	Uniform Resource Locator
VM	Virtual Machine

Table 2: List of abbreviations

2 Overview

2.1 Oracle WebLogic Server

Oracle WebLogic Server is a unified and extensible platform for developing, deploying, and running enterprise applications, such as Java, for on-premises and in the cloud. WebLogic Server offers a robust, mature, and scalable implementation of Java Enterprise Edition (EE) and Jakarta EE.

2.2 Utimaco SecurityServer HSM

SecurityServer is a hardware security module developed by Utimaco IS GmbH. SecurityServer is a physically protected specialized computer unit designed to perform sensitive cryptographic tasks and to securely manage as well as store cryptographic keys and data. It can be used as a universal, independent security component for heterogeneous computer systems.

3 Integration Requirements and Prerequisites

Ensure that the system environment you will be using meets the following hardware and software requirements.

3.1 Tested Versions

The integrations that have been successfully tested with the Utimaco HSM with Oracle WebLogic Server.

Operating System	Oracle WebLogic Server Version	JAVA	Utimaco Security Server Version	Utimaco HSM
Rhel 8	14.1.1	Java 8 Java 11	SecurityServer V4.50.0.2	CryptoServer CSe-Series/Se- Series u.trust Anchor Se*k and u.trust Anchor CSAR
Rhel 8	12.2.1.4.0	Java 8	SecurityServer V4.50.0.2	CryptoServer CSe-Series/Se- Series u.trust Anchor Se*k and u.trust Anchor CSAR

Table 3: List of tested versions

3.2 Software Requirements

Software	Software Requirements
HSM Interfaces	SecurityServer JCE
JDK 8	1.8.0_211
JDK 11	11.0.16
Host OS	Redhat 8
HSM software	Utimaco SecurityServer Software 4.50.0.2
Cxitool	cxitool from product package Utimaco SecurityServer 4.50.0.2
Oracle WebLogic Server	Oracle WebLogic Server version 14.1.1 and 12.2.1.4.0

Table 4: List of software requirements

3.3 Hardware Requirements

Hardware	Hardware Requirements
Utimaco LAN HSM	CryptoServer CSe-Series/Se-Series LAN with firmware SecurityServer 4.50.0.2 or higher u.trust Anchor Se*k and u.trust Anchor CSAR with firmware 4.50 or higher

Hardware	Hardware Requirements
Utimaco PCI-e HSM	CryptoServer CSe-Series/Se-Series PCI-e with firmware SecurityServer 4.50.0.2 or higher u.trust Anchor Se*k and u.trust Anchor CSAR with firmware 4.50 or higher

Table 5: List of hardware requirements

Set up an account on the Utimaco support portal and request download access at the following URL: https://support.hsm.utimaco.com/.

3.4 Prerequisites

Please ensure that:

- The SecurityServer is set up and configured. Refer to the SecurityServer documentations to set up the HSM.
- The SecurityServer Default Admin has been replaced with a new admin user.
- The MBK has been created and stored onto each HSM. Refer to the SecurityServer documentations to set up the MBK.
- The operating system used is listed in Tested Versions.
- The SecurityServer used is listed in Tested Versions.
- You familiarize yourself with the Oracle WebLogic Server documents and setup process.
- The admin user for installing software on Oracle WebLogic Server is set up.
- Ports 7001 and 7002 are allowed through the firewall.

4 Installing and Configuring Utimaco SecurityServer Software

4.1 Download and Install Utimaco Software

If you have not already done so, please create and request an Utimaco Support Portal Account. This will allow you to download the software components needed for this installation.

1. Copy the downloaded software at the appropriate location on the Oracle WebLogic Server.
2. Create `utimaco` folder under `/opt` directory and further create 2 directories; `/opt/utimaco/bin` and `/opt/utimaco/lib`.

›_ Console

```
# mkdir -p /opt/utimaco/bin # mkdir /opt/utimaco/lib
```

3. Copy JCE library file `CryptoServerJCE.jar` from Utimaco SecurityServer software to the `/opt/utimaco/lib` directory.

›_ Console

```
# cp ~/path_to_application_folder/Linux/x86-64/Crypto_APIs/JCE/lib/
CryptoServerJCE.jar /opt/utimaco/lib
```

4. Copy the `csadm`, `ADMIN.key` and `cxitool` files from Utimaco SecurityServer software to

`/opt/utimaco/bin` directory and make both files executable.

›_ Console

```
# cd ~/path_to_application_folder

# cp csadm ADMIN.key cxitool /opt/utimaco/bin

# chmod +x /opt/utimaco/bin/csadm /opt/utimaco/bin/cxitool
```

5. Copy the file `CryptoServerJCE.jar` for Java 8 from Utimaco SecurityServer software to `<java-home>/lib/ext`.

›_ Console

```
cp /opt/utimaco/lib/CryptoServerJCE.jar

/home/oracle/jdk1.8.0_211/jre/lib/ext/
```

4.2 SecurityServer JCE Configuration

1. Locate the Utimaco JCE configuration file in your SecurityServer directory, `Linux/x86-64/Crypto_APIs/JCE/sample/CryptoServer.cfg`.
2. Create a non-root user and set its password.

›_ Console

```
# useradd oracle # passwd oracle
```

3. Log in to Oracle user and copy the Utimaco JCE configuration file `CryptoServer.cfg` into Oracle user's home directory.

›_ Console

```
# cd <installation_directory>/Software/Linux/x86-64/Crypto_APIs/JCE/sample/
CryptoServer.cfg

# cp CryptoServer.cfg $home
```

4. Open `/home/oracle/.bash_profile` and add the following line.

›_ .bash_profile

```
export CRYPTOSERVER_JCE_CONFIG=/home/oracle/CryptoServer.cfg
```

5. Create one Cryptographic user with CXI group.

›_ Console

```
# /opt/utimaco/bin/csadm Dev=3001@127.0.0.1 LogonSign=ADMIN,/opt/utimaco/bin/
ADMIN.key AddUser=<user_name>,00000002{CXI_GROUP=<cxi_group_name>},hmacpwd,<PIN>
```

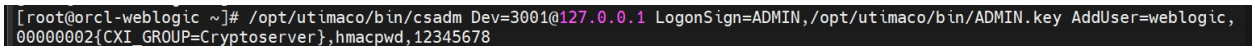

Figure 1 : User creation with csadm

6. Edit the `$home/CryptoServer.cfg` file and make the appropriate changes to the file.

›_CryptoServer.cfg

```
LogFile = /tmp/CryptoServerJCE.log LogLevel = 1

LogSize = 10000 Device = <HSM_IP>

ConnectionTimeout = 3000

Timeout = 30000

KeepSessionAlive = 1

DefaultUser = <Cryptographic_User_Name> KeyGroup = <CXI_Group_Name>

StoreKeysExternal = false
```

For more information regarding the commands and command parameters please check the Utimaco CryptoServer documentation. The device may be a CryptoServer (PCIe or LAN) device. The device line will follow one of these patterns, based on the HSM form-factor:

Device = 288@<HSM IP address> Hardware (LAN) HSM

OR

Device = /dev/cs2.0 Hardware (PCIe) HSM

To make your testing easier, enable the Cryptoserver JCE log file. It can be enabled by editing the Logging Loglevel. Set the LogFile and Logging Loglevel to 1. For testing, you may want to increase it to 4. The added LogFile points to a file. If you encounter problems, check the log file named `CryptoServerJCE.log` in the LogFile defined file. When you are done testing, you should change Logging to 1 or 2. This will limit the logging to only critical and important messages.

7. Obtain the below jurisdiction (unlimited strength) policy files from Oracle for your country and for the correct Java version:
   a. `US_export_policy.jar`.
   b. `local_policy.jar`.

The unlimited policy files are required only for JDK 8 updates earlier than 8u161. On those versions and later, the stronger cryptographic algorithms are available by default.

8. Copy these jurisdiction policy files into the directory `<java-home>/lib/security`.

›_ Console

```
# cp US_export_policy.jar <java_home>/lib/security # cp local_policy.jar
<java_home>/lib/security
```

5 Oracle WebLogic Server Download and Installation

To install Oracle WebLogic server:

1. (Optional) It is recommended to update the system with the latest security patch.

›_ Console

```
# dnf -y update
```

2. Install or open Oracle JDK.

Refer to the Oracle WebLogic support matrix for the version of Java compatible with WebLogic 12c and 14c.

For Java 8:

›_ Console

```
# tar -xvf jdk-8u211-linux-x64.tar.gz
```

For Java 11:

›_ Console

```
C:\> cngtool ListKeys# tar -xvf jdk-11.0.6_linux-x64_bin.tar.gz
```

3. Download Generic Installer for Oracle WebLogic server from the official Oracle site:

https://www.oracle.com/middleware/technologies/weblogic-server-installers-downloads.html.

4. Pre-installation tasks.

›_ Console

```
# groupadd -g 1001 oinstall

# usermod -u 1001 -g oinstall oracle
```

5. Create a directory to install WebLogic software and to set necessary permissions.

›_ Console

```
# mkdir -p /u01/app/oracle/product/<WebLogic_Version> # chown -R
oracle:oinstall /u01/app

# chmod -R 775 /u01
```

6. Log in to oracle user.
7. Edit `.bash_profile` of oracle user in `vim` text editor.

›_ Console

```
# vim .bash_profile
```

Add following environment variables in `/home/oracle/.bash_profile`.

›_ Console

```
export JAVA_HOME=/home/oracle/jdk1.8.0_211 export JRE_HOME=/home/oracle/
jdk1.8.0_211/jre export PATH=$PATH:/home/oracle/jdk1.8.0_211/bin/ export
ORACLE_BASE=/u01/app/oracle

export MW_HOME=$ORACLE_BASE/product/14.1.1 export WLS_HOME=$MW_HOME/wlserver

export WL_HOME=$WLS_HOME

export DOMAIN_BASE=$ORACLE_BASE/config/domain export DOMAIN_HOME=$DOMAIN_BASE/
admin/admindomain

export CLASSPATH=/opt/utimaco/lib/CryptoServerJCE.jar
```

Change the values according to your system configuration.

8. Source the `.bash_profile` script to set environment variables for current Linux shell.

›_ Console

```
$ . ~/.bash_profile
```

9. Execute `unzip` command to extract WebLogic Generic Installer.

›_ Console

```
$ unzip fmw_<version>_wls_lite_Disk1_1of1.zip
```

```
[oracle@orcl-weblogic ~]$ unzip fmw_14.1.1.0.0_wls_lite_Disk1_1of1.zip
Archive:  fmw_14.1.1.0.0_wls_lite_Disk1_1of1.zip
  inflating: fmw_14.1.1.0.0_wls_lite_generic.jar
[oracle@orcl-weblogic ~]$
```

Figure 2 : Unzipping the installer

10. Execute the Generic Installer JAR file by using following java command with oracle user.

›_ Console

```
$ java -jar fmw_<version>_wls_lite_generic.jar
```

11. A window will appear asking for installation related information. Follow the prompt and finish the installation.

12. Open `http://<oracle_weblogic_-server-ip>:7001` in any web browser and verify if Oracle WebLogic page is visible. Login with the Administrative user which was created in the installation and configuration steps.

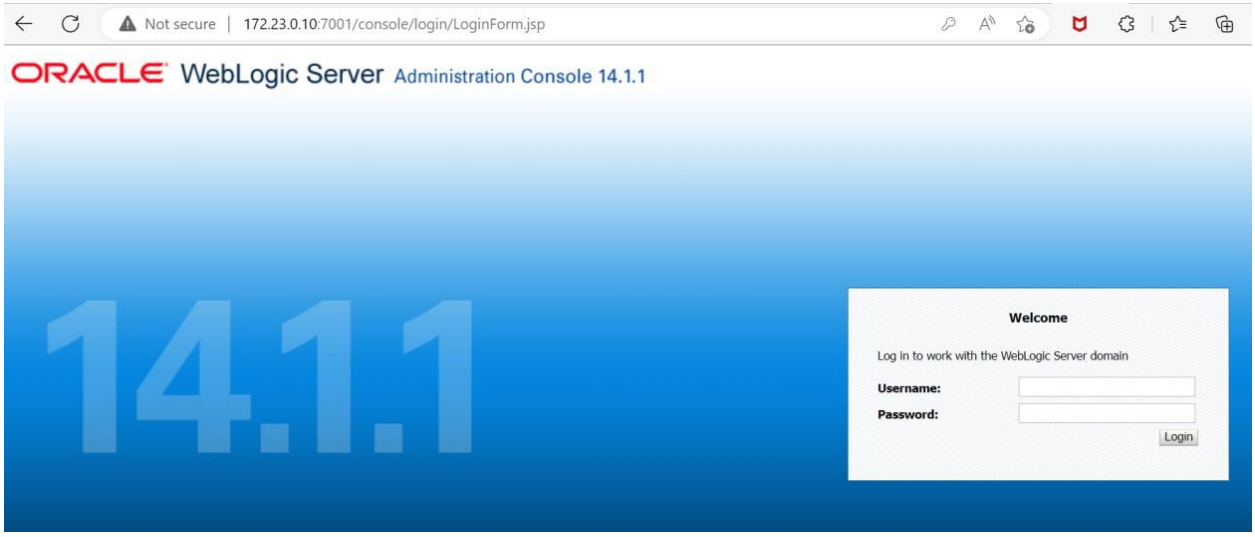

Figure 3 : Browser output over page 7001

6 Java Configuration to use Utimaco HSM

6.1 Update java.security file to use Utimaco HSM for JDK8

1. Go to the `<JDK_Installation_directory>/jre/lib/security` directory.

›_ Console

```
# cd /home/oracle/jdk1.8.0_211/jre/lib/security/
```

2. Edit the `java.security` configuration file to add `provider`, as highlighted below.

›_ Console

```
security.provider.1=sun.security.provider.Sun
security.provider.2=sun.security.rsa.SunRsaSign
security.provider.3=sun.security.ec.SunEC
security.provider.4=com.sun.net.ssl.internal.ssl.Provider
security.provider.5=com.sun.crypto.provider.SunJCE
security.provider.6=sun.security.jgss.SunProvider
security.provider.7=com.sun.security.sasl.Provider
security.provider.8=org.jcp.xml.dsig.internal.dom.XMLDSigRI
security.provider.9=sun.security.smartcardio.SunPCSC
security.provider.10=CryptoServerJCE.CryptoServerProvider

/home/oracle/CryptoServer.cfg
```

Specify the correct provider number and path for CryptoServerJCE Provider.

6.2 Update java.security file to use Utimaco HSM for JDK11

1. Go to the `<JDK_Installation_directory>` `conf/security` directory.

›_ Console

```
# cd /home/oracle/jdk-11.0.6/conf/security/
```

2. Edit the `java.security` configuration file to add CryptoServerJCE provider.

›_ Console

```
security.provider.1=SUN
security.provider.2=SunRsaSign
security.provider.3=SunEC
security.provider.4=SunJSSE
security.provider.5=SunJCE
security.provider.6=SunJGSS
security.provider.7=SunSASL
security.provider.8=XMLDSig
security.provider.9=SunPCSC
security.provider.10=JdkLDAP
security.provider.11=JdkSASL
security.provider.12=SunPKCS11
security.provider.13=CryptoServerJCE.CryptoServerProvider
```

Specify the correct provider number and path for CryptoServerJCE Provider.

7 SSL Setup for Oracle WebLogic Server on Utimaco HSM

7.1 For OpenJDK8 with RSA Key

1. Log in as a root user and generate a keypair on Utimaco HSM.

›_ Console

```
# keytool -genkey -keyalg RSA -keysize 2048 -keystore NONE -storetype
CryptoServer -storepass 12345678 -providername CryptoServer -alias weblogicrsa
```

Provide information when prompted.

Here:

- RSA is the key algorithm.
- 2048 is the key size.
- NONE is the keystore for HSM.
- CryptoServer is the storetype.
- 12345678 is the slot PIN.
- CryptoServer is the provider name.
- `weblogicrsa` is the key name that will be generated on Utimaco HSM.

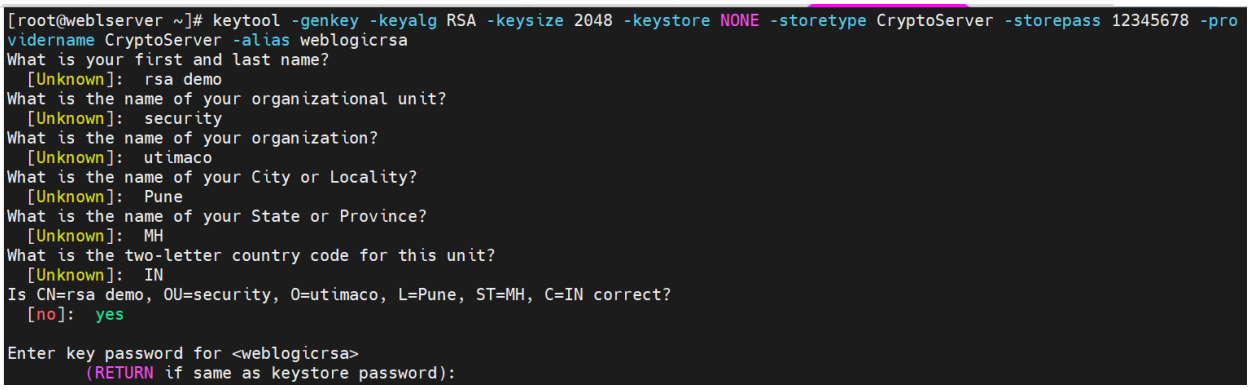

Figure 4 : Key generation using keytool command

Self-signed certificates do not work with the WebLogic server.

2. Verify that the keys have been generated using `keytool` command.

›_ Console

```
# keytool -list -keystore NONE -storetype CryptoServer -providername
CryptoServer -storepass 12345678 -v
```

Here:

- NONE is the keystore for HSM.
- CryptoServer is the storetype.
- 12345678 is the slot PIN.
- CryptoServer is the provider's name.

```
[root@weblserver ~]# keytool -list -keystore NONE -storetype CryptoServer -providername CryptoServer -storepass 12345678 -v
Keystore type: CRYPTOSERVER
Keystore provider: CryptoServer

Your keystore contains 1 entry

Alias name: weblogicrsa
Creation date: Apr 18, 2023
Entry type: PrivateKeyEntry
Certificate chain length: 1
Certificate[1]:
Owner: CN=rsa demo, OU=security, O=utimaco, L=pune, ST=MH, C=IN
Issuer: CN=rsa demo, OU=security, O=utimaco, L=pune, ST=MH, C=IN
Serial number: 5e89f76
Valid from: Tue Apr 18 12:05:11 UTC 2023 until: Mon Jul 17 12:05:11 UTC 2023
Certificate fingerprints:
         MD5:  81:FE:CE:07:B5:F0:02:1B:AC:41:A4:EE:59:23:CC:F9
         SHA1: 60:7A:A1:FB:6D:E9:AB:62:57:9F:63:C5:68:22:C4:83:93:7C:9C:4F
         SHA256: 83:E3:96:DA:48:18:AE:78:5A:58:1E:FC:86:B0:6B:26:5E:2F:A2:A4:B8:D0:27:19:68:25:4E:82:B7:CE:3A:F1
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 2048-bit RSA key
Version: 3

Extensions:

#1: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: A0 88 AC B2 BA 1E 6D B0   05 CA 32 35 F0 B5 09 B7  ......m...25....
0010: AE 8F D8 28                                        ...(
]
]



*******************************************
*******************************************


[root@weblserver ~]#
```

Figure 5 : Listkeys output

3. List the keys using `cxitool`.

›_ Console

```
# /opt/utimaco/bin/cxitool Dev=3001@127.0.0.1 Logonpass=weblogic,12345678
Group=Cryptoserver Listkeys
```

```
[root@weblserver ~]# /opt/utimaco/bin/cxitool Dev=3001@127.0.0.1 Logonpass=weblogic,12345678 Group=Cryptoserver Listkeys

idx algo  size type     group                  name                           spec
-------------------------------------------------------------------------------------
1   RSA   2048 pub+prv  Cryptoserver           weblogicrsa
```

Figure 6 : List Keys output using cxitool

4. Generate a CSR using `keytool` command.

›_ Console

```
# keytool -certreq -alias weblogicrsa -file rsa.csr -storetype CryptoServer
-keystore NONE -v
```

Provide the keystore password when prompted.

Here:

- NONE is the keystore for HSM.
- CryptoServer is the storetype.
- CryptoServer is the provider name.
- `weblogicrsa` is the key name.
- `rsa.csr` is the CSR file name that will be generated.

5. Get this CSR signed by CA.
6. Copy the signed certificate on the WebLogic server.
7. Import the Root certificate into the HSM keystore.

›_ Console

```
# keytool -importcert -alias RootCA -file /home/LAbCA-Root.crt -storetype
CryptoServer -keystore NONE -providername CryptoServer -storepass 12345678
```

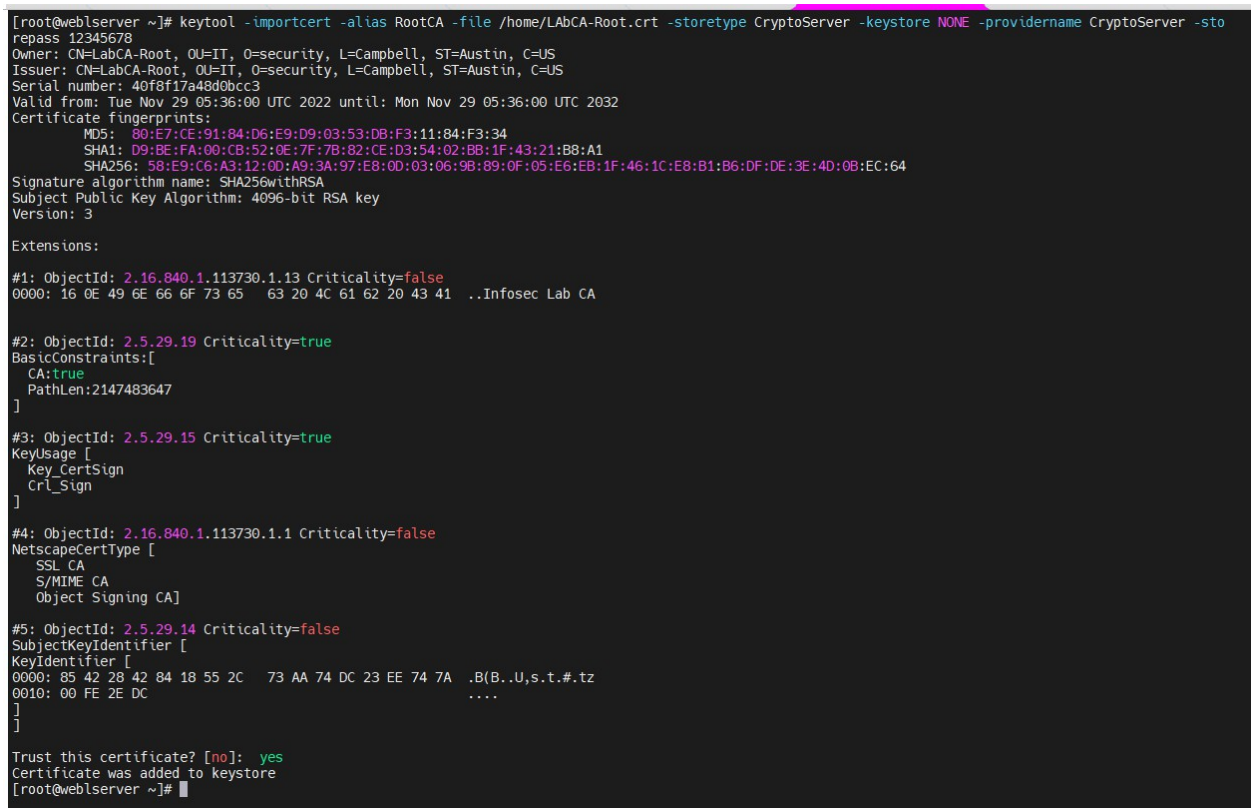

Figure 7 : Importing root certificate

8. Import the signed certificate reply using the command below.

›_ Console

```
# keytool -importcert -alias weblogicrsa -file /home/rsa_demo.pem - storetype
CryptoServer -keystore NONE -providername CryptoServer - storepass 12345678
```

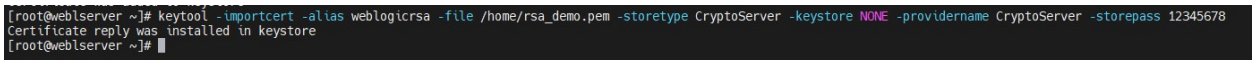

Figure 8 : Import user certificate into keystore

9. List the keystore entries.

›_ Console

```
# keytool -list -keystore NONE -storetype CryptoServer -providername
CryptoServer -storepass 12345678 -v
```

```
[root@weblserver ~]# keytool -list -keystore NONE -storetype CryptoServer -providername CryptoServer -storepass 12345678 -v
Keystore type: CRYPTOSERVER
Keystore provider: CryptoServer

Your keystore contains 2 entries

Alias name: weblogicrsa
Creation date: Apr 18, 2023
Entry type: PrivateKeyEntry
Certificate chain length: 1
Certificate[1]:
Owner: CN=rsa demo, OU=security, O=utimaco, L=pune, ST=MH, C=IN
Issuer: CN=LabCA-Root, OU=IT, O=security, L=Campbell, ST=Austin, C=US
Serial number: 7a02a40a421ae723
Valid from: Tue Apr 18 12:07:00 UTC 2023 until: Thu Apr 18 12:07:00 UTC 2024
Certificate fingerprints:
         MD5:  47:57:35:28:E7:3C:43:04:17:A5:AE:CA:97:64:AB:63
         SHA1: ED:E5:A8:96:C5:81:9D:72:77:97:42:69:B2:43:DF:C6:1B:A5:48:B0
         SHA256: 58:3F:48:C0:A7:DB:31:78:CE:6E:31:B1:81:BF:EF:9E:9E:DB:4D:DC:50:98:19:66:96:39:76:24:E3:CC:10:FC
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 2048-bit RSA key
Version: 3

Extensions:

#1: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: A0 88 AC B2 BA 1E 6D B0   05 CA 32 35 F0 B5 09 B7  ......m...25....
0010: AE 8F D8 28                                        ...(
]
]



*******************************************
*******************************************


Alias name: RootCA
Creation date: Apr 18, 2023
Entry type: trustedCertEntry

Owner: CN=LabCA-Root, OU=IT, O=security, L=Campbell, ST=Austin, C=US
Issuer: CN=LabCA-Root, OU=IT, O=security, L=Campbell, ST=Austin, C=US
Serial number: 40f8f17a48d0bcc3
Valid from: Tue Nov 29 05:36:00 UTC 2022 until: Mon Nov 29 05:36:00 UTC 2032
Certificate fingerprints:
         MD5:  80:E7:CE:91:84:D6:E9:D9:03:53:DB:F3:11:84:F3:34
```

Figure 9 : Listkeys output

```
         MD5:  80:E7:CE:91:84:D6:E9:D9:03:53:DB:F3:11:84:F3:34
         SHA1: D9:BE:FA:00:CB:52:0E:7F:7B:82:CE:D3:54:02:BB:1F:43:21:B8:A1
         SHA256: 58:E9:C6:A3:12:0D:A9:3A:97:E8:0D:03:06:9B:89:0F:05:E6:EB:1F:46:1C:E8:B1:B6:DF:DE:3E:4D:0B:EC:64
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 4096-bit RSA key
Version: 3

Extensions:

#1: ObjectId: 2.16.840.1.113730.1.13 Criticality=false
0000: 16 0E 49 6E 66 6F 73 65   63 20 4C 61 62 20 43 41  ..Infosec Lab CA


#2: ObjectId: 2.5.29.19 Criticality=true
BasicConstraints:[
  CA:true
  PathLen:2147483647
]

#3: ObjectId: 2.5.29.15 Criticality=true
KeyUsage [
  Key_CertSign
  Crl_Sign
]

#4: ObjectId: 2.16.840.1.113730.1.1 Criticality=false
NetscapeCertType [
   SSL CA
   S/MIME CA
   Object Signing CA]

#5: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 85 42 28 42 84 18 55 2C   73 AA 74 DC 23 EE 74 7A  .B(B..U,s.t.#.tz
0010: 00 FE 2E DC                                        ....
]
]



*******************************************
*******************************************
```

7.2 For OpenJDK8 with EC Key

1. Generate an EC keypair on the Utimaco HSM.

›_ Console

```
# keytool -genkey -keyalg EC -keystore NONE -storetype CryptoServer - storepass
12345678 -providername CryptoServer -alias webleckey
```

Provide information when prompted.

Here:

- EC is the key algorithm.
- NONE is the keystore for HSM.

- CryptoServer is the storetype.
- 12345678 is the slot PIN.
- CryptoServer is the provider name.
- `webleckey` is the key name that will be generated on Utimaco HSM.

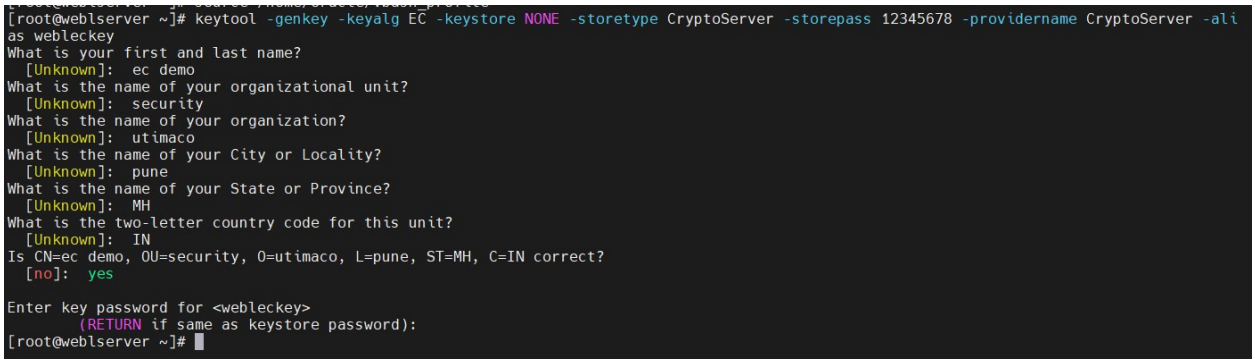

Figure 10 : Key Generation using keytool command output

2. Verify that the keys have been generated.

›_ Console

```
# keytool -list -keystore NONE -storetype CryptoServer -providername
CryptoServer -storepass 12345678 -v
```

Here:

- NONE is the keystore for HSM.
- CryptoServer is the storetype.
- 12345678 is the slot PIN.
- CryptoServer is the provider's name.

```
[root@weblserver ~]# keytool -list -keystore NONE -storetype CryptoServer -providername CryptoServer -storepass 12345678 -v
Keystore type: CRYPTOSERVER
Keystore provider: CryptoServer

Your keystore contains 1 entry

Alias name: webleckey
Creation date: Apr 18, 2023
Entry type: PrivateKeyEntry
Certificate chain length: 1
Certificate[1]:
Owner: CN=ec demo, OU=security, O=utimaco, L=pune, ST=MH, C=IN
Issuer: CN=ec demo, OU=security, O=utimaco, L=pune, ST=MH, C=IN
Serial number: 51b381e5
Valid from: Tue Apr 18 12:19:46 UTC 2023 until: Mon Jul 17 12:19:46 UTC 2023
Certificate fingerprints:
         MD5:  12:C2:78:B7:9B:3B:66:D9:89:B7:46:EB:65:56:FD:C1
         SHA1: A1:B4:E2:7A:9F:EC:76:51:A6:A4:94:26:D2:EE:C9:0F:C7:6C:65:28
         SHA256: 89:43:CB:46:87:9B:85:93:EA:1C:C1:2D:9E:8D:6C:0C:54:3E:B5:1B:CE:49:81:74:81:93:08:CC:BC:8A:F8:3F
Signature algorithm name: SHA256withECDSA
Subject Public Key Algorithm: 256-bit EC key
Version: 3

Extensions:

#1: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 31 1B BE 89 6F 94 62 6F   F8 AF A4 02 83 EF 0E F5  1...o.bo........
0010: 16 87 09 C4                                        ....
]
]



*******************************************
*******************************************


[root@weblserver ~]#
```

Figure 11 : Listkeys output

3. List the keys using `cxitool`.

›_ Console

```
# /opt/utimaco/bin/cxitool Dev=3001@127.0.0.1 Logonpass=weblogic,12345678
Group=Cryptoserver Listkeys
```

```
[root@weblserver ~]# /opt/utimaco/bin/cxitool Dev=3001@127.0.0.1 Logonpass=weblogic,12345678 Group=Cryptoserver Listkeys

idx algo  size type     group                   name                          spec
-------------------------------------------------------------------------------------
1   ECDSA 256  pub+prv  Cryptoserver            webleckey

[root@weblserver ~]#
```

Figure 12 : List keys output using cxitool

4. Generate a CSR using `keytool` command.

›_ Console

```
# keytool -certreq -alias webleckey -file webleckey.csr -storetype CryptoServer
-keystore NONE -v
```

Provide the keystore password when prompted.

Here:

- NONE is the keystore for HSM.
- CryptoServer is the storetype.
- CryptoServer is the provider name.
- `webleckey` is the key name.
- `webleckey.csr` is the CSR file name that will be generated.

5. Get this CSR signed by CA.
6. Copy the signed certificate on the WebLogic server.
7. Import the Root certificate into the HSM keystore.

›_ Console

```
# keytool -importcert -alias RootCA -file /home/rootca.crt -storetype
CryptoServer -keystore NONE -providername CryptoServer -storepass 12345678
```

```
[root@weblserver ~]# keytool -importcert -alias RootCA -file /home/LAbCA-Root.crt -storetype CryptoServer -keystore NONE -providername CryptoServer -storepass 12345678
Owner: CN=LabCA-Root, OU=IT, O=security, L=Campbell, ST=Austin, C=US
Issuer: CN=LabCA-Root, OU=IT, O=security, L=Campbell, ST=Austin, C=US
Serial number: 40f8f17a48d0bcc3
Valid from: Tue Nov 29 05:36:00 UTC 2022 until: Mon Nov 29 05:36:00 UTC 2032
Certificate fingerprints:
         MD5:  80:E7:CE:91:84:D6:E9:D9:03:53:DB:F3:11:84:F3:34
         SHA1: D9:BE:FA:00:CB:52:0E:7F:7B:82:CE:D3:54:02:BB:1F:43:21:B8:A1
         SHA256: 58:E9:C6:A3:12:0D:A9:3A:97:E8:0D:03:06:9B:89:0F:05:E6:EB:1F:46:1C:E8:B1:B6:DF:DE:3E:4D:0B:EC:64
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 4096-bit RSA key
Version: 3

Extensions:

#1: ObjectId: 2.16.840.1.113730.1.13 Criticality=false
0000: 16 0E 49 6E 66 6F 73 65   63 20 4C 61 62 20 43 41  ..Infosec Lab CA


#2: ObjectId: 2.5.29.19 Criticality=true
BasicConstraints:[
  CA:true
  PathLen:2147483647
]

#3: ObjectId: 2.5.29.15 Criticality=true
KeyUsage [
  Key_CertSign
  Crl_Sign
]

#4: ObjectId: 2.16.840.1.113730.1.1 Criticality=false
NetscapeCertType [
   SSL CA
   S/MIME CA
   Object Signing CA]

#5: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 85 42 28 42 84 18 55 2C   73 AA 74 DC 23 EE 74 7A  .B(B..U,s.t.#.tz
0010: 00 FE 2E DC                                        ....
]
]

Trust this certificate? [no]:  yes
Certificate was added to keystore
[root@weblserver ~]#
```

Figure 13 : Import root certificate into keystore

8. Import the signed certificate reply using the command below.

›_ Console

```
# keytool -importcert -alias webleckey -file /home/ec_demo.pem -storetype
CryptoServer -keystore NONE -providername CryptoServer -storepass 12345678
```

```
[root@weblserver ~]# keytool -importcert -alias webleckey -file /home/ec_demo.pem -storetype CryptoServer -keystore NONE -providername CryptoServer -storepass 12345678
Certificate reply was installed in keystore
[root@weblserver ~]#
```

Figure 14 : Import certificate reply into keystore

9. List the keystore entries.

›_ Console

```
# keytool -list -keystore NONE -storetype CryptoServer -providername
CryptoServer -storepass 12345678 -v
```

Here:

- NONE is the keystore for HSM.
- CryptoServer is the storetype.
- 12345678 is the slot PIN.
- CryptoServer is the provider's name.

```
[root@weblserver ~]# keytool -list -keystore NONE -storetype CryptoServer -providername CryptoServer -storepass 12345678 -v
Keystore type: CRYPTOSERVER
Keystore provider: CryptoServer

Your keystore contains 2 entries

Alias name: RootCA
Creation date: Apr 18, 2023
Entry type: trustedCertEntry

Owner: CN=LabCA-Root, OU=IT, O=security, L=Campbell, ST=Austin, C=US
Issuer: CN=LabCA-Root, OU=IT, O=security, L=Campbell, ST=Austin, C=US
Serial number: 40f8f17a48d0bcc3
Valid from: Tue Nov 29 05:36:00 UTC 2022 until: Mon Nov 29 05:36:00 UTC 2032
Certificate fingerprints:
         MD5:  80:E7:CE:91:84:D6:E9:D9:03:53:DB:F3:11:84:F3:34
         SHA1: D9:BE:FA:00:CB:52:0E:7F:7B:82:CE:D3:54:02:BB:1F:43:21:B8:A1
         SHA256: 58:E9:C6:A3:12:0D:A9:3A:97:E8:0D:03:06:9B:89:0F:05:E6:EB:1F:46:1C:E8:B1:B6:DF:DE:3E:4D:0B:EC:64
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 4096-bit RSA key
Version: 3

Extensions:

#1: ObjectId: 2.16.840.1.113730.1.13 Criticality=false
0000: 16 0E 49 6E 66 6F 73 65   63 20 4C 61 62 20 43 41  ..Infosec Lab CA


#2: ObjectId: 2.5.29.19 Criticality=true
BasicConstraints:[
  CA:true
  PathLen:2147483647
]

#3: ObjectId: 2.5.29.15 Criticality=true
KeyUsage [
  Key_CertSign
  Crl_Sign
]

#4: ObjectId: 2.16.840.1.113730.1.1 Criticality=false
NetscapeCertType [
   SSL CA
   S/MIME CA
   Object Signing CA]

#5: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
```

Figure 15 : Keytool list output

```
#5: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 85 42 28 42 84 18 55 2C   73 AA 74 DC 23 EE 74 7A  .B(B..U,s.t.#.tz
0010: 00 FE 2E DC                                        ....
]
]



*******************************************
*******************************************


Alias name: webleckey
Creation date: Apr 18, 2023
Entry type: PrivateKeyEntry
Certificate chain length: 1
Certificate[1]:
Owner: CN=ec demo, OU=security, O=utimaco, L=pune, ST=MH, C=IN
Issuer: CN=LabCA-Root, OU=IT, O=security, L=Campbell, ST=Austin, C=US
Serial number: 12ee47b57f7413fd
Valid from: Tue Apr 18 12:22:00 UTC 2023 until: Thu Apr 18 12:22:00 UTC 2024
Certificate fingerprints:
         MD5:  70:FA:03:D4:AA:D1:7B:A0:F4:63:4C:90:AC:17:90:04
         SHA1: 02:9A:75:5B:09:D5:C7:BC:37:E5:F5:F8:AA:C1:39:E1:07:28:6D:43
         SHA256: DE:8F:96:E3:01:E4:20:11:E1:27:3D:F8:26:AA:AB:D7:4D:C2:7D:05:C6:F6:16:11:53:06:91:AC:F1:A6:8F:05
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 256-bit EC key
Version: 3

Extensions:

#1: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 31 1B BE 89 6F 94 62 6F   F8 AF A4 02 83 EF 0E F5  1...o.bo........
0010: 16 87 09 C4                                        ....
]
]



*******************************************
*******************************************


[root@weblserver ~]#
```

7.3 For OpenJDK11 with EC Key

1. Generate an EC keypair on Utimaco HSM.

›_ Console

```
# keytool -genkeypair -alias weblogiceckey -keyalg EC -keystore NONE - storetype
CryptoServer -storepass 12345678 -providerpath "/opt/utimaco/lib/
CryptoServerJCE.jar" -providerclass CryptoServerJCE.CryptoServerProvider -J-
Djava.library.path=/opt/utimaco/lib/ -J-cp - J/opt/utimaco/lib/
CryptoServerJCE.jar -providername CryptoServer -v
```

Provide information when prompted.

Here:

- EC is the key algorithm.

- NONE is the keystore for HSM.
- CryptoServer is the storetype.
- 12345678 is the slot PIN.
- CryptoServer is the provider name.
- weblogiceckey is the key name that will be generated on Utimaco HSM.

```
[root@orcl-weblogic ~]# keytool -genkeypair -alias weblogiceckey -keyalg EC -keystore NONE -storetype CryptoServer -storepass
12345678 -providerpath "/opt/utimaco/lib/CryptoServerJCE.jar" -providerclass CryptoServerJCE.CryptoServerProvider -J-Djava.lib
rary.path=/opt/utimaco/lib/ -J-cp -J/opt/utimaco/lib/CryptoServerJCE.jar -providername CryptoServer -v
What is your first and last name?
  [Unknown]:  weblogic demo
What is the name of your organizational unit?
  [Unknown]:  security
What is the name of your organization?
  [Unknown]:  utimaco
What is the name of your City or Locality?
  [Unknown]:  pune
What is the name of your State or Province?
  [Unknown]:  MH
What is the two-letter country code for this unit?
  [Unknown]:  IN
Is CN=weblogic demo, OU=security, O=utimaco, L=pune, ST=MH, C=IN correct?
  [no]:  yes

Generating 256 bit EC key pair and self-signed certificate (SHA256withECDSA) with a validity of 90 days
        for: CN=weblogic demo, OU=security, O=utimaco, L=pune, ST=MH, C=IN
Enter key password for <weblogiceckey>
        (RETURN if same as keystore password):
[Storing keystore]
[root@orcl-weblogic ~]#
```

Figure 16 : Key generation using keytool command

For OpenJDK 11 RSA key algorithm is not supported with Utimaco HSM.

2. Verify that the keys have been generated.

›_ Console

```
# keytool -list -keystore NONE -storetype CryptoServer -storepass 12345678
-providerpath "/opt/utimaco/lib/CryptoServerJCE.jar" - providerclass
CryptoServerJCE.CryptoServerProvider -J-Djava.library.path=/opt/utimaco/lib/ -J-
cp - J/opt/utimaco/lib/CryptoServerJCE.jar -providername CryptoServer -v
```

Here:

- NONE is the keystore for HSM.
- CryptoServer is the storetype.

- 12345678 is the slot PIN.
- CryptoServer is the provider's name.

```
[root@orcl-weblogic ~]# keytool -list -keystore NONE -storetype CryptoServer -storepass 12345678 -providerpath "/opt/utimaco/l
ib/CryptoServerJCE.jar" -providerclass CryptoServerJCE.CryptoServerProvider -J-Djava.library.path=/opt/utimaco/lib/ -J-cp -J/o
pt/utimaco/lib/CryptoServerJCE.jar -providername CryptoServer -v
Keystore type: CRYPTOSERVER
Keystore provider: CryptoServer

Your keystore contains 1 entry

Alias name: weblogiceckey
Creation date: Apr 18, 2023
Entry type: PrivateKeyEntry
Certificate chain length: 1
Certificate[1]:
Owner: CN=weblogic demo, OU=security, O=utimaco, L=pune, ST=MH, C=IN
Issuer: CN=weblogic demo, OU=security, O=utimaco, L=pune, ST=MH, C=IN
Serial number: 766af718
Valid from: Tue Apr 18 12:51:04 UTC 2023 until: Mon Jul 17 12:51:04 UTC 2023
Certificate fingerprints:
         SHA1: 6F:69:0C:36:FB:22:86:0A:E1:6D:D6:77:24:F1:6F:57:9D:EF:79:60
         SHA256: 3B:57:D0:C3:F3:0C:49:DC:EF:BC:0F:7D:91:0E:2B:89:CB:24:88:0E:B1:65:A5:55:2F:A9:01:89:FD:CF:94:B7
Signature algorithm name: SHA256withECDSA
Subject Public Key Algorithm: 256-bit EC key
Version: 3

Extensions:

#1: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 34 C5 11 1A 45 E8 95 7E   F7 7D 7D D3 2E 9A CD 1F  4...E...........
0010: BD 3D B2 64                                        .=.d
]
]



*******************************************
*******************************************


[root@orcl-weblogic ~]#
```

Figure 17 : Listkeys output

3. List the keys using `cxitool`.

›_ Console

```
# /opt/utimaco/bin/cxitool Dev=3001@127.0.0.1 Logonpass=weblogic,12345678
Group=Cryptoserver Listkeys
```

```
[root@orcl-weblogic ~]# /opt/utimaco/bin/cxitool Dev=3001@127.0.0.1 Logonpass=weblogic,12345678 Group=Cryptoserver Listkeys

idx algo  size type     group                  name                           spec
-------------------------------------------------------------------------------
1   ECDSA 256  pub+prv  Cryptoserver            weblogiceckey

[root@orcl-weblogic ~]#
```

Figure 18 : List keys output using cxitool

4. Generate a CSR using `keytool` command.

›_ Console

```
# keytool -certreq -alias weblogiceckey -file webec.csr -keystore NONE -
storetype CryptoServer -providerpath "/opt/utimaco/lib/CryptoServerJCE.jar"
-providerclass CryptoServerJCE.CryptoServerProvider -J-Djava.library.path=/opt/
utimaco/lib -J-cp - J/opt/utimaco/lib/CryptoServerJCE.jar/opt/utimaco/lib/
CryptoServerJCE.jar

-providername CryptoServer -v
```

Provide the keystore password when prompted.

Here:

- NONE is the keystore for HSM.
- CryptoServer is the storetype.
- CryptoServer is the provider name.
- weblogiceckey is the key name.
- webec.csr is the CSR file name that will be generated.

5. Get this CSR signed by CA.
6. Copy the signed certificate on the WebLogic server.
7. Import the Root certificate into the HSM keystore.

›_ Console

```
# keytool -importcert -alias RootCA -file /home/LAbCA-Root.crt -storetype
CryptoServer -keystore NONE -providerpath "/opt/utimaco/lib/CryptoServerJCE.jar"
-providerclass CryptoServerJCE.CryptoServerProvider -J-Djava.library.path=/opt/
utimaco/lib -J-cp - J/opt/utimaco/lib/CryptoServerJCE.jar -providername
CryptoServer - storepass 12345678
```

```
[root@orcl-weblogic ~]# keytool -importcert -alias RootCA -file /home/LAbCA-Root.crt -storetype CryptoServer -keystore NONE -providerpath "/opt/utimaco/
lib/CryptoServerJCE.jar" -providerclass CryptoServerJCE.CryptoServerProvider -J-Djava.library.path=/opt/utimaco/lib -J-cp -J/opt/utimaco/lib/CryptoServe
rJCE.jar -providername CryptoServer -storepass 12345678
Owner: CN=LabCA-Root, OU=IT, O=security, L=Campbell, ST=Austin, C=US
Issuer: CN=LabCA-Root, OU=IT, O=security, L=Campbell, ST=Austin, C=US
Serial number: 40f8f17a48d0bcc3
Valid from: Tue Nov 29 05:36:00 UTC 2022 until: Mon Nov 29 05:36:00 UTC 2032
Certificate fingerprints:
         SHA1: D9:BE:FA:00:CB:52:0E:7F:7B:82:CE:D3:54:02:BB:1F:43:21:B8:A1
         SHA256: 58:E9:C6:A3:12:0D:A9:3A:97:E8:0D:03:06:9B:89:0F:05:E6:EB:1F:46:1C:E8:B1:B6:DF:DE:3E:4D:0B:EC:64
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 4096-bit RSA key
Version: 3

Extensions:

#1: ObjectId: 2.16.840.1.113730.1.13 Criticality=false
0000: 16 0E 49 6E 66 6F 73 65   63 20 4C 61 62 20 43 41  ..Infosec Lab CA


#2: ObjectId: 2.5.29.19 Criticality=true
BasicConstraints:[
  CA:true
  PathLen:2147483647
]

#3: ObjectId: 2.5.29.15 Criticality=true
KeyUsage [
  Key_CertSign
  Crl_Sign
]

#4: ObjectId: 2.16.840.1.113730.1.1 Criticality=false
NetscapeCertType [
   SSL CA
   S/MIME CA
   Object Signing CA]

#5: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
```

Figure 19 : Importing root certificate into keystore

```
#5: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 85 42 28 42 84 18 55 2C   73 AA 74 DC 23 EE 74 7A  .B(B..U,s.t.#.tz
0010: 00 FE 2E DC                                        ....
]
]

Trust this certificate? [no]:  yes
Certificate was added to keystore
[root@orcl-weblogic ~]#
```

8. Import the signed certificate reply using the command below.

›_ Console

```
# keytool -importcert -alias weblogiceckey -file /home/weblogic_demo.pem

-storetype CryptoServer -keystore NONE -providerpath "/opt/utimaco/lib/
CryptoServerJCE.jar" -providerclass CryptoServerJCE.CryptoServerProvider -J-
Djava.library.path=/opt/utimaco/lib -J-cp - J/opt/utimaco/lib/
CryptoServerJCE.jar -providername CryptoServer - storepass 12345678
```

```
[root@orcl-weblogic ~]# keytool -importcert -alias weblogiceckey -file /home/weblogic_demo.pem -storetype CryptoServer -keystore NONE -providerpath "/opt/utimaco/lib/CryptoServerJCE.jar" -providerclass CryptoServerJCE.CryptoServerProvider -J-Djava.library.path=/opt/utimaco/lib -J-cp -J/opt/utimaco/lib/CryptoServerJCE.jar -providername CryptoServer -storepass 12345678
Certificate reply was installed in keystore
[root@orcl-weblogic ~]#
```

Figure 20 : Import certificate reply into keystore

9. List the keystore entries.

›_ Console

```
# keytool -list -keystore NONE -storetype CryptoServer -storepass 12345678
-providerpath "/opt/utimaco/lib/CryptoServerJCE.jar" - providerclass
CryptoServerJCE.CryptoServerProvider -J-Djava.library.path=/opt/utimaco/lib/ -J-
cp - J/opt/utimaco/lib/CryptoServerJCE.jar -providername CryptoServer -v
```

```
[root@orcl-weblogic ~]# keytool -list -keystore NONE -storetype CryptoServer -storepass 12345678 -providerpath "/opt/utimaco/lib/CryptoServerJCE.jar" -providerclass CryptoServerJCE.CryptoServerProvider -J-Djava.library.path=/opt/utimaco/lib/ -J-cp -J/opt/utimaco/lib/CryptoServerJCE.jar -providername CryptoServer -v
Keystore type: CRYPTOSERVER
Keystore provider: CryptoServer

Your keystore contains 2 entries

Alias name: weblogiceckey
Creation date: Apr 18, 2023
Entry type: PrivateKeyEntry
Certificate chain length: 1
Certificate[1]:
Owner: CN=weblogic demo, OU=security, O=utimaco, L=pune, ST=MH, C=IN
Issuer: CN=LabCA-Root, OU=IT, O=security, L=Campbell, ST=Austin, C=US
Serial number: b4e961c895a09dd
Valid from: Tue Apr 18 12:53:00 UTC 2023 until: Thu Apr 18 12:53:00 UTC 2024
Certificate fingerprints:
         SHA1: 5E:6F:80:32:BC:7B:AD:A5:59:9B:56:80:CD:B3:ED:F2:3B:30:31:3B
         SHA256: 2C:B9:E4:FC:55:F0:76:A9:04:6D:CA:4D:98:17:04:BC:50:1A:03:D3:9F:F4:2C:A5:AA:DD:A8:06:03:5F:F9:69
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 256-bit EC key
Version: 3

Extensions:

#1: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 34 C5 11 1A 45 E8 95 7E   F7 7D 7D D3 2E 9A CD 1F  4...E...........
0010: BD 3D B2 64                                        .=.d
]
]



*******************************************
*******************************************


Alias name: RootCA
Creation date: Apr 18, 2023
Entry type: trustedCertEntry
```

Figure 21 : Listkeys output

```
Entry type: trustedCertEntry

Owner: CN=LabCA-Root, OU=IT, O=security, L=Campbell, ST=Austin, C=US
Issuer: CN=LabCA-Root, OU=IT, O=security, L=Campbell, ST=Austin, C=US
Serial number: 40f8f17a48d0bcc3
Valid from: Tue Nov 29 05:36:00 UTC 2022 until: Mon Nov 29 05:36:00 UTC 2032
Certificate fingerprints:
         SHA1: D9:BE:FA:00:CB:52:0E:7F:7B:82:CE:D3:54:02:BB:1F:43:21:B8:A1
         SHA256: 58:E9:C6:A3:12:0D:A9:3A:97:E8:0D:03:06:9B:89:0F:05:E6:EB:1F:46:1C:E8:B1:B6:DF:DE:3E:4D:0B:EC:64
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 4096-bit RSA key
Version: 3

Extensions:

#1: ObjectId: 2.16.840.1.113730.1.13 Criticality=false
0000: 16 0E 49 6E 66 6F 73 65   63 20 4C 61 62 20 43 41  ..Infosec Lab CA


#2: ObjectId: 2.5.29.19 Criticality=true
BasicConstraints:[
  CA:true
  PathLen:2147483647
]

#3: ObjectId: 2.5.29.15 Criticality=true
KeyUsage [
  Key_CertSign
  Crl_Sign
]

#4: ObjectId: 2.16.840.1.113730.1.1 Criticality=false
NetscapeCertType [
   SSL CA
   S/MIME CA
   Object Signing CA]

#5: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 85 42 28 42 84 18 55 2C   73 AA 74 DC 23 EE 74 7A  .B(B..U,s.t.#.tz
0010: 00 FE 2E DC                                        ....
]
]



*******************************************
*******************************************
```

7.4 Update Domain Structure in Oracle Web Console to use HSM for SSL

1. Open the **Administration Console** using the link http://hostname:7001/console. Navigate to **Domain Structure** > **Environment**.
2. Click **Servers**.
3. Click **AdminServer** as shown in the figure.

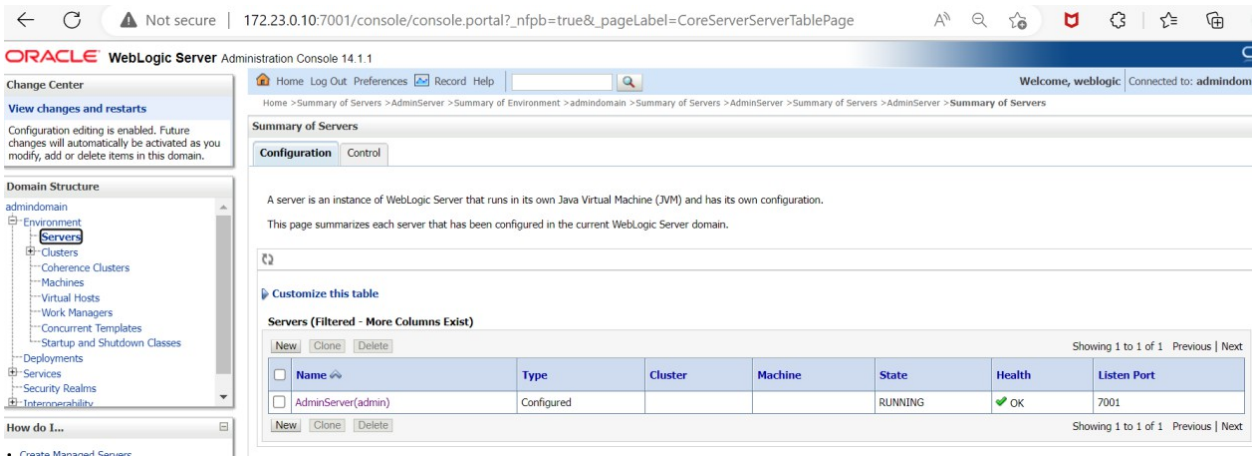

Figure 22 : Updating AdminServer

4. Select the **SSL Listen Port Enabled** checkbox and click **Save**.

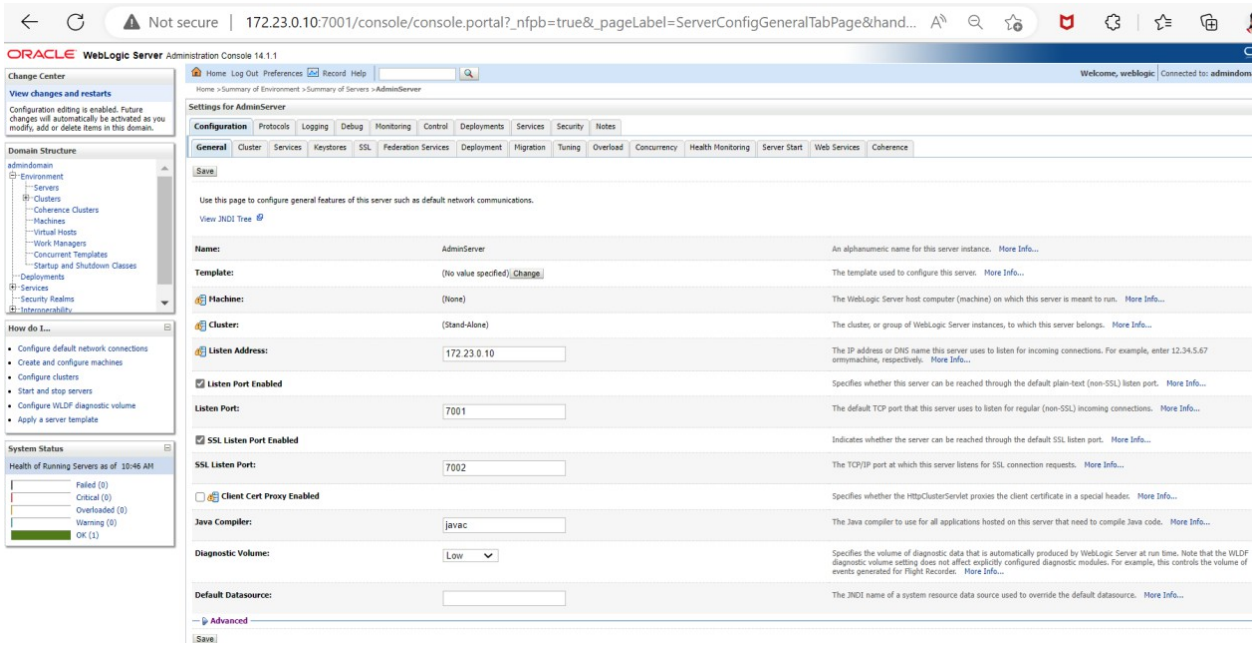

Figure 23 : Enabling SSL configuration

5. Open the **Keystores** tab and then complete the following steps:

   a. Click on the **Change** button.

   b. From the drop-down menu, select **Custom Identity** and **Custom Trust** and click **Save**.

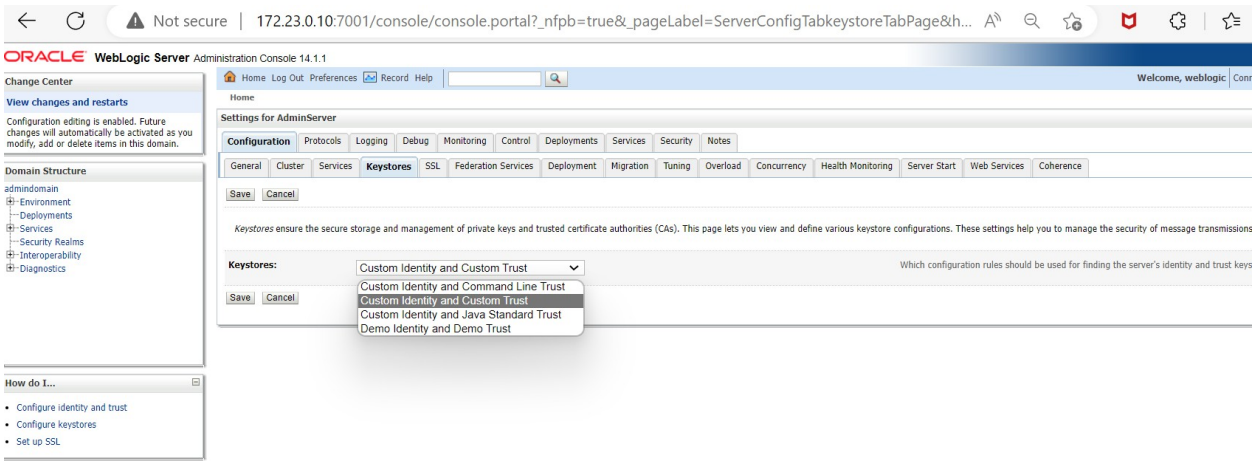

Figure 24 : Updating the keystore configuration

c. In the **Custom Identity Keystore** field, enter CryptoServer.

d. In the **Custom Identity Keystore Type** field, enter CryptoServer.

e. In the **Custom Identity Keystore Passphrase** and **Confirm Custom Identity Keystore Passphrase** fields, enter the password for the HSM.

f. In the **Custom Trust Keystore** field, enter CryptoServer.

g. In the **Custom Trust Keystore Type** field, enter CryptoServer.

h. In the **Custom Trust Keystore Passphrase** and **Confirm Custom Trust Keystore Passphrase** fields, enter the password for the HSM and click **Save**, as shown below.

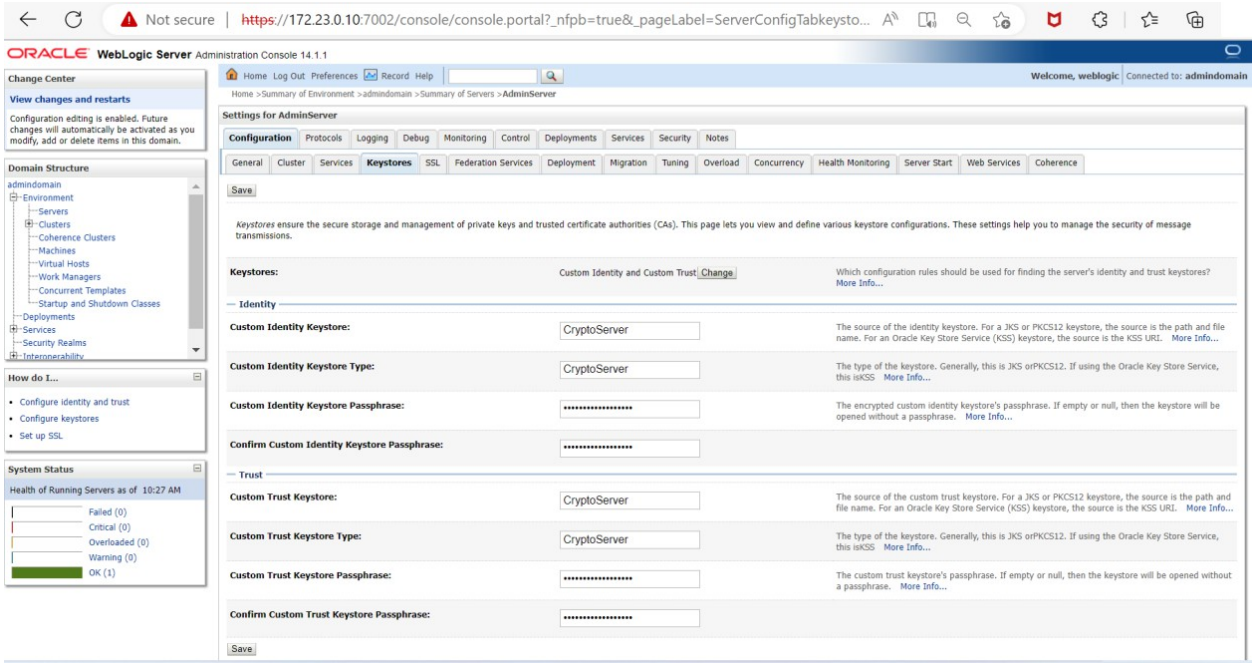

Figure 25 : Updating the keystore configuration

6. Open the SSL tab.

   a. In the **Private Key Alias** field, enter the name of the SSL key generated on the HSM (e.g., `weblogiceckey`.

   b. In the **Private Key Passphrase** field, leave the field empty and click on **Save**.

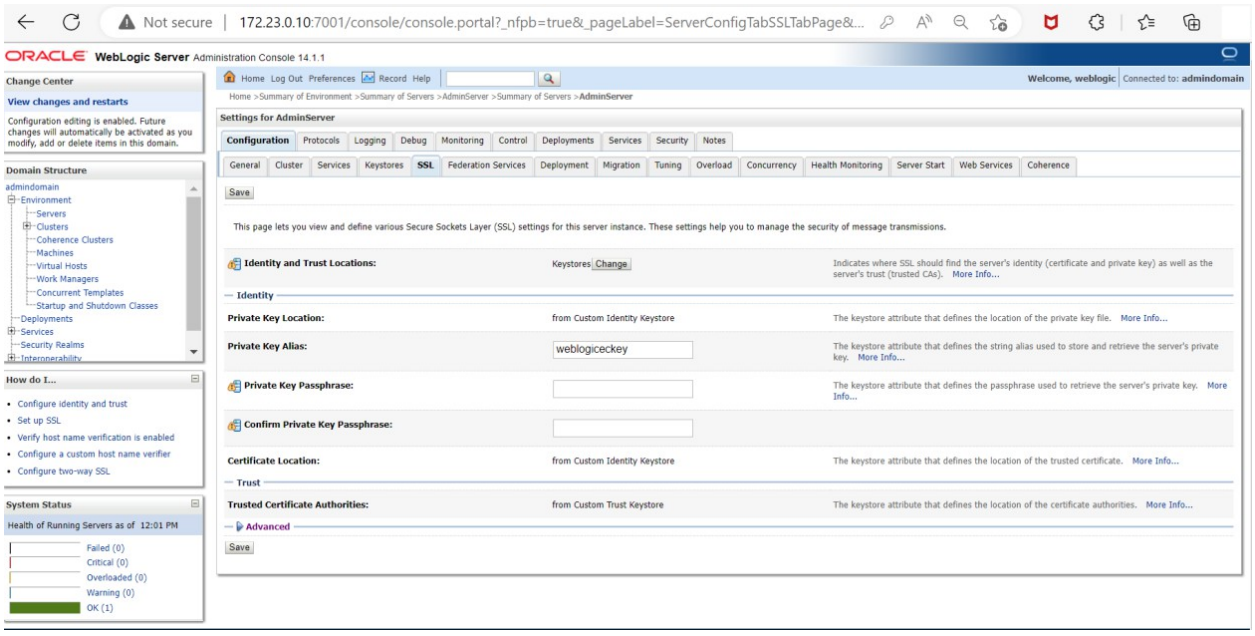

Figure 26 : Updating SSL private key

7. Restart the WebLogic server by clicking on **Environment** > **Servers**. Click on **AdminServer**, then click on **control tab**, and lastly click on **restart the SSL**.

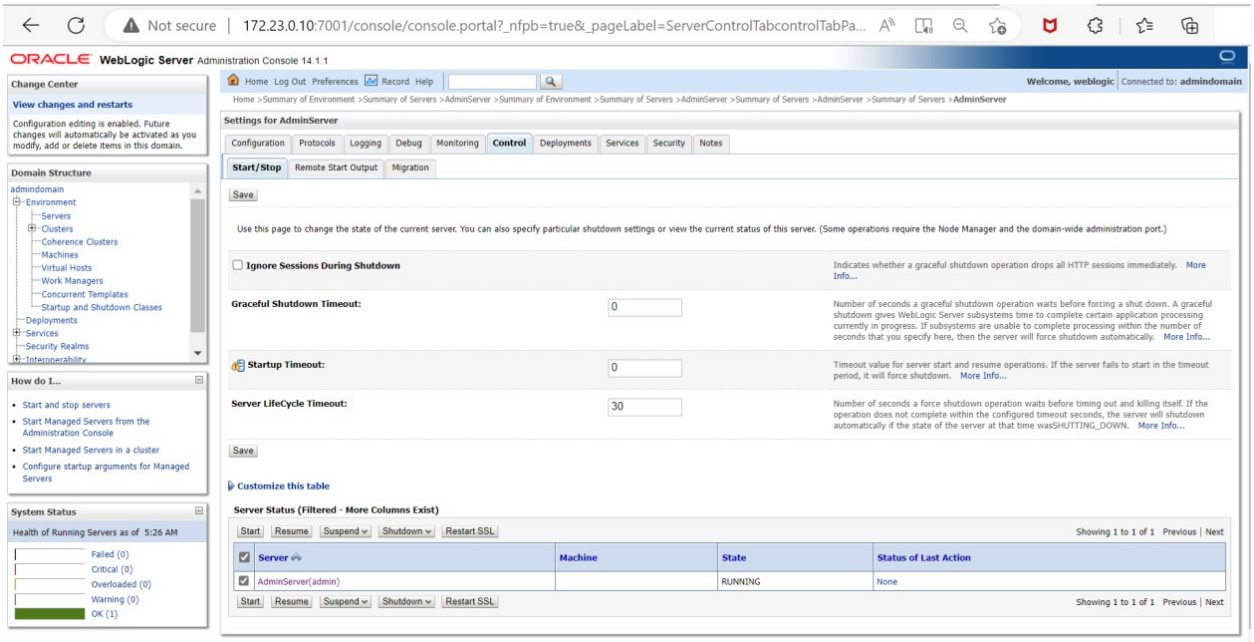

Figure 27 : Restarting SSL

8. After restarting the WebLogic server, access the **Administration** console over https using https://hostname:7002/console.

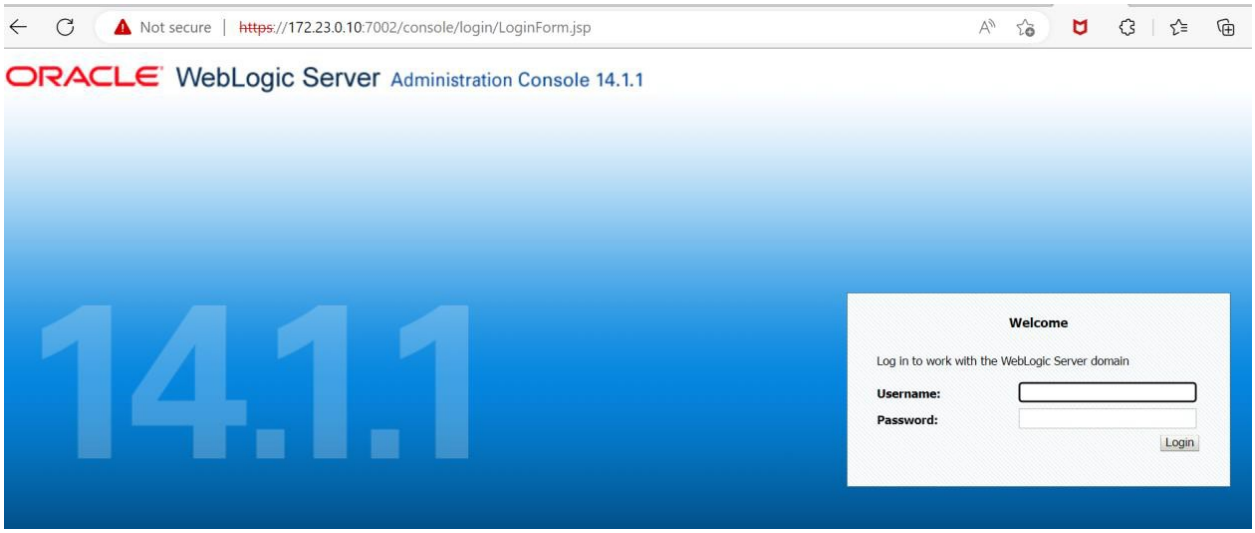

Figure 28 : WebLogic service status over https

This completes the integration of Oracle WebLogic Server with Utimaco SecurityServer.

8 Troubleshooting

Error	Diagnosis
<BEA-090166> <Failed to load identity keystore of type CryptoServer from file /u01/app/oracle/config/domain/admindomain/ CryptoServer on server AdminServer	Need to export the variable by running export `CLASSPATH=` `/u01/app/oracle/product/14.1.1/ wlserver/serv er/lib/ CryptoServerJCE.jar`.
<Error> <WebLogicServer> <BEA-000297> <Incons istent security configuration, weblogic.management.configuration.Configura tionException: Failed to load identity keystore of type CryptoServer from file /u01/app/oracle/config/domains/admindomai n/ CryptoServer on server AdminServer>	CryptoServer i.e `Cryptoserver.cfg` file should exist at location `/u01/app/oracle/config/domains/ admindomai n/CryptoServer` and having adequate permissions on that file.
<BEA-090172> <No trusted certificates have been loaded. Server will not trust to any certificate it receives.>	Need to import Root CA certificate as well into the HSM keystore.
Each time when WebLogic service is started, and the page is loaded over SSL a new key is generated with random alias name on the HSM slot.	You can delete this key with random alias name with the help of `p11tool2` command.

Table 6: List of errors and their diagnoses

9 Further Information

This document forms a part of the information and support which is provided by the Utimaco IS GmbH. Additional documentation can be found on the product CD in the Documentation directory.

All CryptoServer product documentation is also available at the Utimaco IS GmbH website: https://utimaco.com/.

10 References

Reference	Title/Company	Document No.
[CSADMIN]	CryptoServer – csadm Manual/Utimaco IS GmbH	2009-0003
[CSTrSh]	CryptoServer Troubleshooting/Utimaco IS GmbH	M011-0008-en
[CSADMIN2]	CryptoServer_csadm_Manual_Systemadministrators.pdf	2009-0003
[CSLAN5]	CryptoServerLAN_Manual_Systemadministrators.pdf	2018-000

Table 7: References

11 Contact and Support Information

You can reach us from Monday to Friday, 09.00 a.m. to 05.00 p.m., Central European Time (CET).

Utimaco IS GmbH
Krefelder Str. 220
52070 Aachen
Germany

RMA Query

If you need to send the device back to Utimaco IS GmbH, please open a new RMA case (Return Merchandise Authorization). We request that you use the following web address. RMA cases cannot be opened by email or phone.

https://support.hsm.utimaco.com/support/rma/new

Other Support Queries

- Mail (preferred contact method)
  support@utimaco.com
  Attach the diagnostic information to your email.
- Web portal
  https://support.hsm.utimaco.com/support/cases/new/
  The diagnostic information will be requested in our response if necessary.
- By phone
  AMERICAS +1-844-UTIMACO (+1 844-884-6226)
  EMEA +49 800-627-3081
  APAC +81 800-919-1301
  The diagnostic information will be requested in our response if necessary.

You can reach us from Monday to Friday, 09.00 a.m. to 05.00 p.m., Central European Time (CET).

Utimaco IS GmbH
Krefelder Str. 220
52070 Aachen
Germany

RMA Query

If you need to send the device back to Utimaco IS GmbH, please open a new RMA case (Return Merchandise Authorization). We request that you use the following web address. RMA cases cannot be opened by email or phone.

https://support.hsm.utimaco.com/support/rma/new

Other Support Queries

- Mail (preferred contact method)
  support@utimaco.com
  Attach the diagnostic information to your email.
- Web portal
  https://support.hsm.utimaco.com/support/cases/new/
  The diagnostic information will be requested in our response if necessary.
- By phone
  AMERICAS +1-844-UTIMACO (+1 844-884-6226)
  EMEA +49 800-627-3081
  APAC +81 800-919-1301
  The diagnostic information will be requested in our response if necessary.