**Steps to create a self-signed certificate and configure Custom Identity and Custom Trust with Weblogic Server using Keytool...**

August 23, 2021 | 5 minute read

[Puneeth Prakash](https://blogs.oracle.com/authors/puneeth-prakash)

Senior Principal Software Engineer

Original Publish Date : 8/24/2013

**Quick Commands : (Just copy the below 3 commands and run it on your shell/cmd prompt to quickly create an identity and trust Keystore) :**

***keytool -genkeypair -alias mykey -keyalg RSA -sigalg SHA256withRSA -keysize 2048 -validity 365 -dname "cn=localhost, ou=WLS, o=Oracle, c=IN" -storepass password -keypass privatepassword -keystore identity.jks***

***keytool -export -alias mykey -file root.cer -keystore identity.jks -storepass password***

***keytool -import -alias mykey -file root.cer -keystore trust.jks -storepass password -trustcacerts -noprompt***

----------------

**Steps in Detail :**

**Below are the steps to create a self signed certificate :**

Command 1 :

 keytool -genkey -alias mykey -keyalg RSA -keysize 1024 -validity 365 -keypass privatepassword -keystore identity.jks -storepass password

Note :

List of keytool commands which are changed in java 1.6 :

-export, renamed to -exportcert

-genkey, renamed to -genkeypair

-import, renamed to -importcert

All previous commands are still supported in this release ( keytool in java 1.6 ) and will continue to be supported in future releases.

To create a 2048 bit SHA2/SHA256 certificate use the following command :

Command :

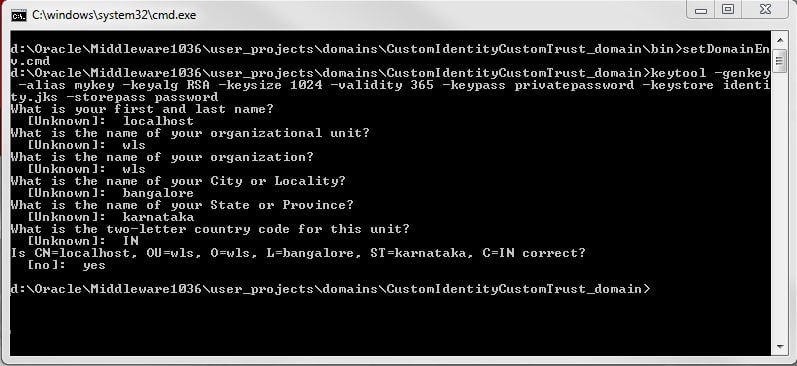
**keytool -genkey -alias mykey -keyalg RSA -keysize 2048 -sigalg SHA256withRSA -validity 365 -keypass privatepassword -keystore identity.jks -storepass password**

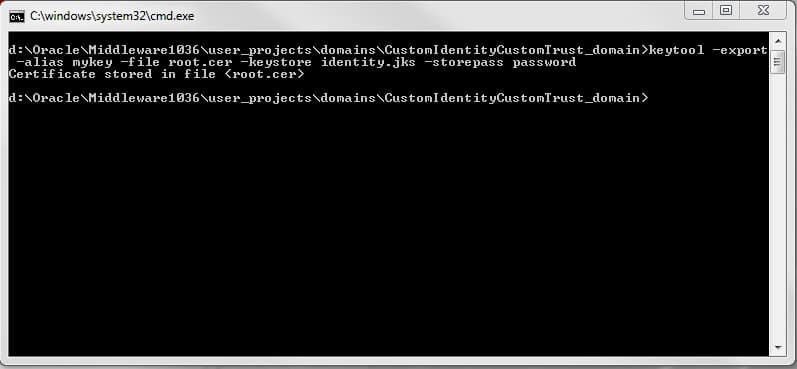
Command 2 :

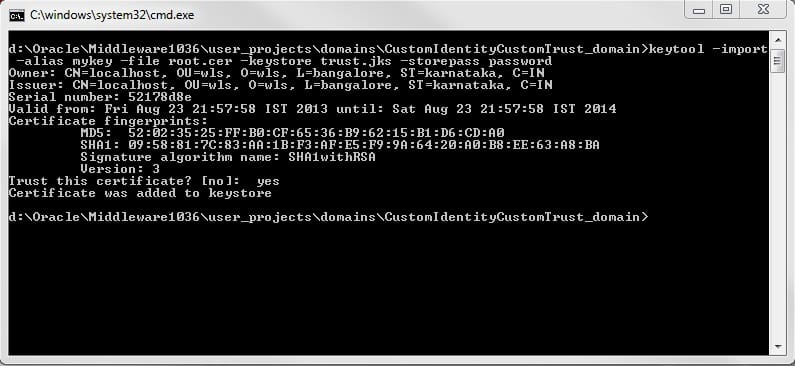
**keytool  -export -alias mykey -file root.cer -keystore identity.jks -storepass password**

Command 3 :

**keytool -import -alias mykey -file root.cer -keystore trust.jks -storepass password**

[](https://https/blogs.oracle.com/content/published/api/v1.1/assets/CONT629D0F0CCCB84427A35162C6AC042139/Medium?cb=_cache_59da&format=jpg&channelToken=e3ef4fd8fc674b8d9e81b98087030487)

">[](https://https/blogs.oracle.com/content/published/api/v1.1/assets/CONT2CDB5DC0DF38471A82AD3F61A85DF0A6/Medium?cb=_cache_59da&format=jpg&channelToken=e3ef4fd8fc674b8d9e81b98087030487)

[](https://https/blogs.oracle.com/content/published/api/v1.1/assets/CONTEA20290315334BACA91DA9529FF32509/Medium?cb=_cache_59da&format=jpg&channelToken=e3ef4fd8fc674b8d9e81b98087030487)

 < Additional Info >

 To see the contents of the keystore use the following command :

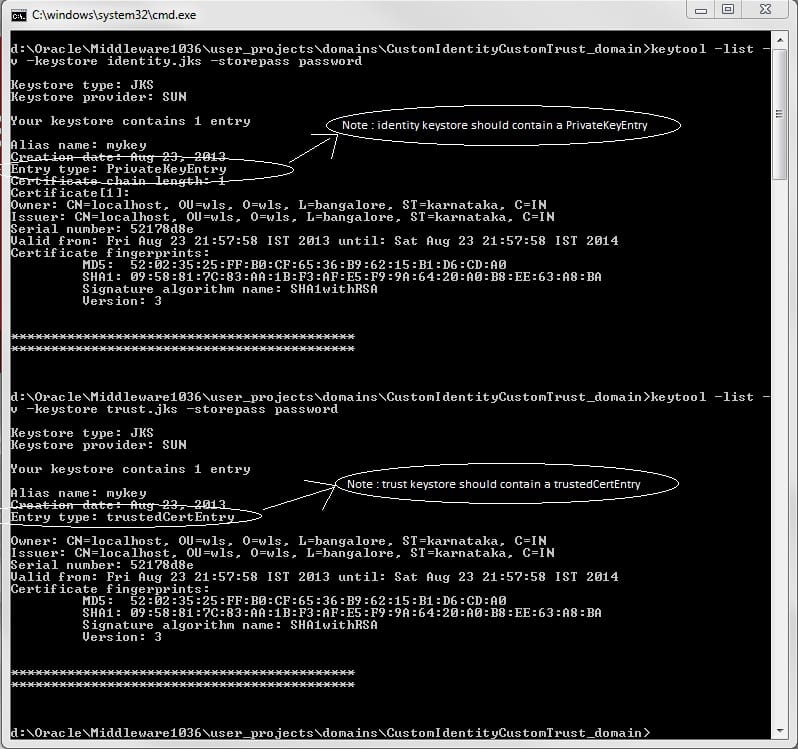
Command :

**keytool -list -v -keystore identity.jks -storepass password**

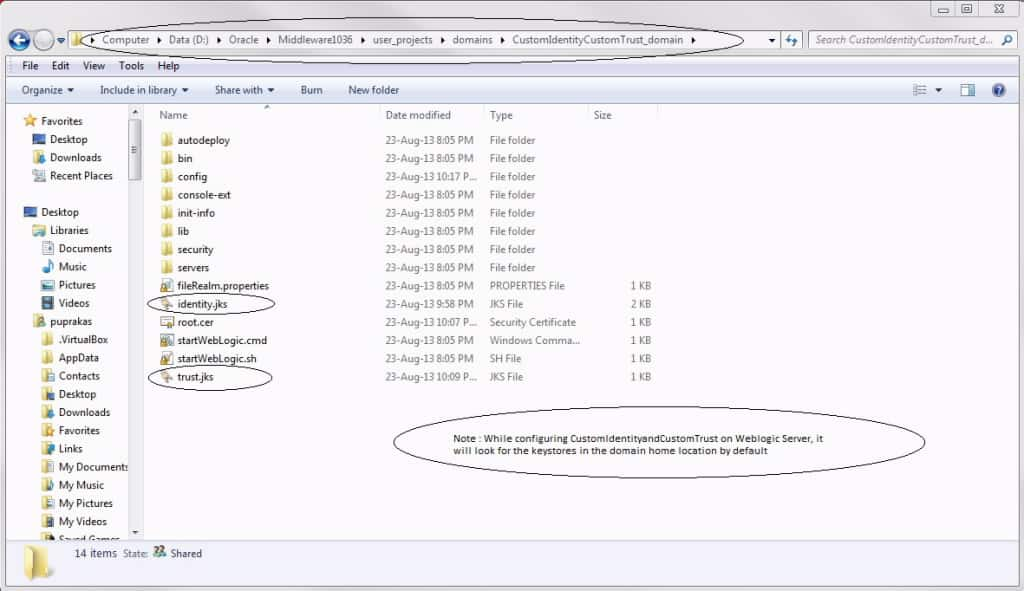
To see the contents of an individual certificate ( like root.cer in our case ).

Command :

**keytool -printcert -file root.cer**

[](https://https/blogs.oracle.com/content/published/api/v1.1/assets/CONTD2863E1340D64B7597EE17215CDA7912/Medium?cb=_cache_59da&format=jpg&channelToken=e3ef4fd8fc674b8d9e81b98087030487)

Copy the keystore files in the domain\_home location :

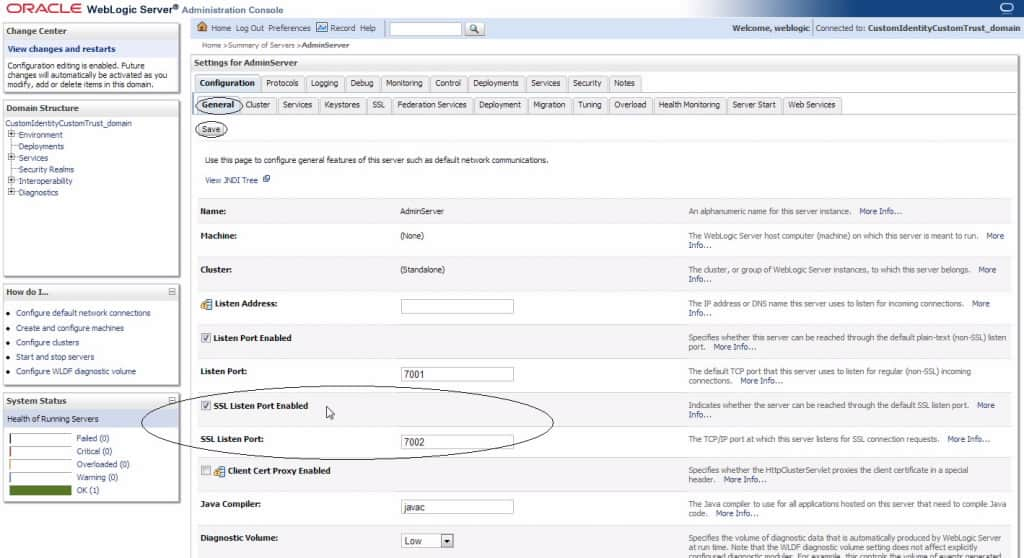
[](https://https/blogs.oracle.com/content/published/api/v1.1/assets/CONT9E4430C06B024E71A733A0D9AE7DB997/Medium?cb=_cache_59da&format=jpg&channelToken=e3ef4fd8fc674b8d9e81b98087030487)

**Below are the steps to configure Custom Identity and Custom Trust with Weblogic Server :**

**Step 1 :**

Login to Weblogic Admin console --> Environment --> Servers --> < server\_name\_where\_ssl\_has\_to\_be\_configured > --> Configuration -> General --> SSL Listen Port Enabled ( Check )

Note : The default SSL Listen Port would be 7002, change it if required.

[](https://https/blogs.oracle.com/content/published/api/v1.1/assets/CONTAADC313397294855850AFBD1180C921F/Medium?cb=_cache_59da&format=jpg&channelToken=e3ef4fd8fc674b8d9e81b98087030487)

**Step 2 :**

Click on Keystores tab under " Configuration " tab :

**Step 2a :**

Click on the drop down menu next to Keystores and sleect " Custom Identity and Custom Trust "

**Step 2b :**

Now fill in the following information :

---Identity---

Custom Identity Keystore : < location\_of\_identity\_keystore\_that\_you\_have\_created>

NOTE : By default WLS will look for this keystore file in domain\_home location.

 Custom Identity Keystore Type : jks

 Custom Identity Keystore Passphrase: < This\_would\_be\_your\_storepass >

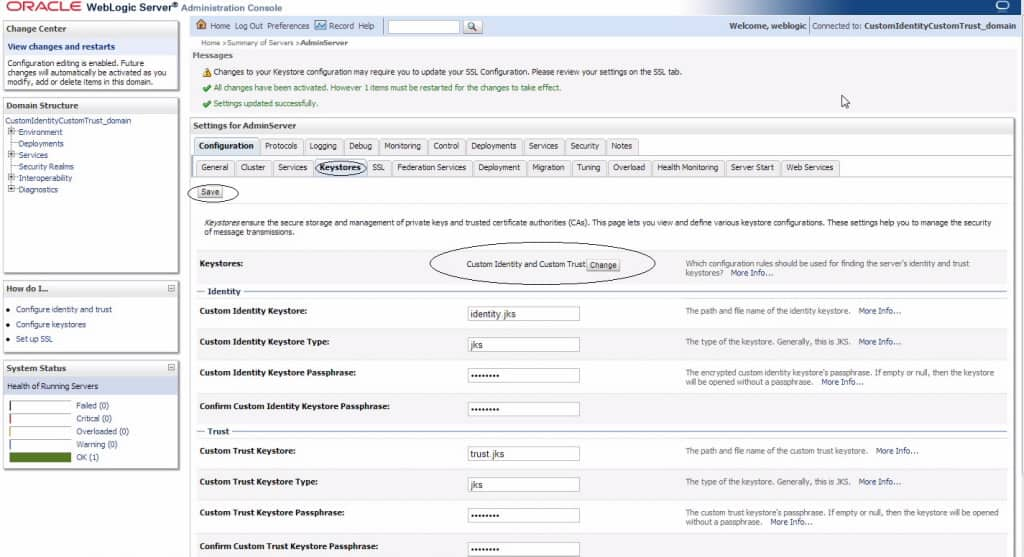
 ---Trust---

 Custom Trust Keystore : < location\_of\_trust\_keystore\_that\_you\_have\_created>

NOTE : By default WLS will look for this keystore file in domain\_home location.

 Custom Trust Keystore Type : jks

 Custom Trust Keystore Passphrase: < This\_would\_be\_your\_storepass >

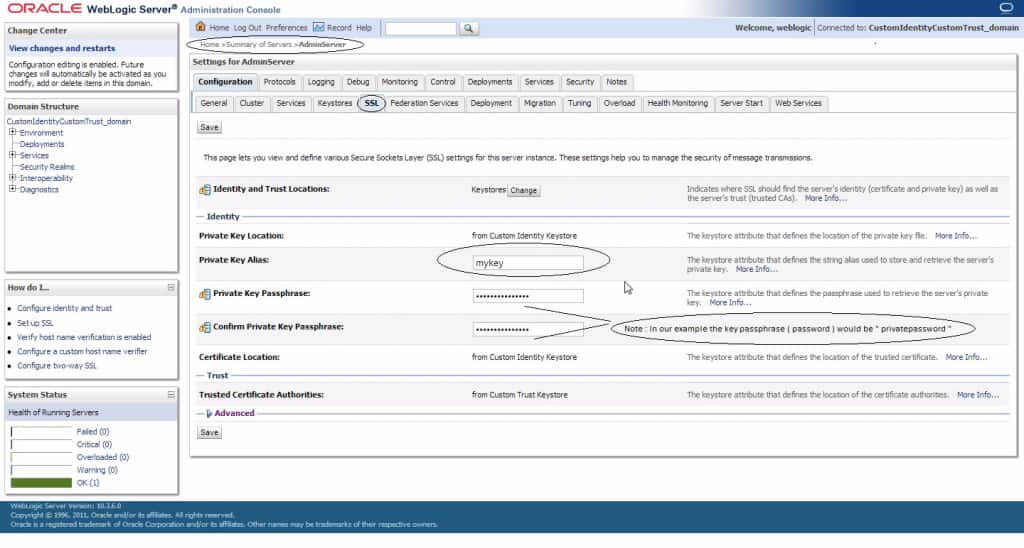
[](https://https/blogs.oracle.com/content/published/api/v1.1/assets/CONTF3DE0B17E7BA4C808947490F94D5AAB8/Medium?cb=_cache_59da&format=jpg&channelToken=e3ef4fd8fc674b8d9e81b98087030487)

**Step 2c :**

Now save the changes and click on " SSL " tab :

Private Key Alias: < This\_would\_be\_your\_certificate\_alias >

Private Key Passphrase: < This\_would\_be\_your\_keypass >

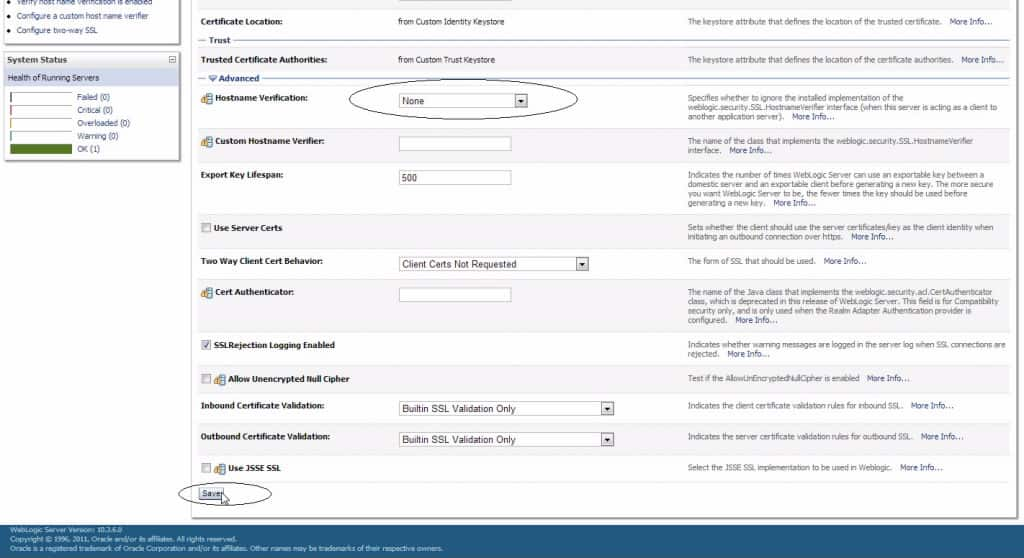
[](https://https/blogs.oracle.com/content/published/api/v1.1/assets/CONT1B43800C5B10489FAB3138F7E401C74F/Medium?cb=_cache_59da&format=jpg&channelToken=e3ef4fd8fc674b8d9e81b98087030487)

**Step 3 :**

Save the changes and click on the " >Advanced " field under the " SSL " tab :

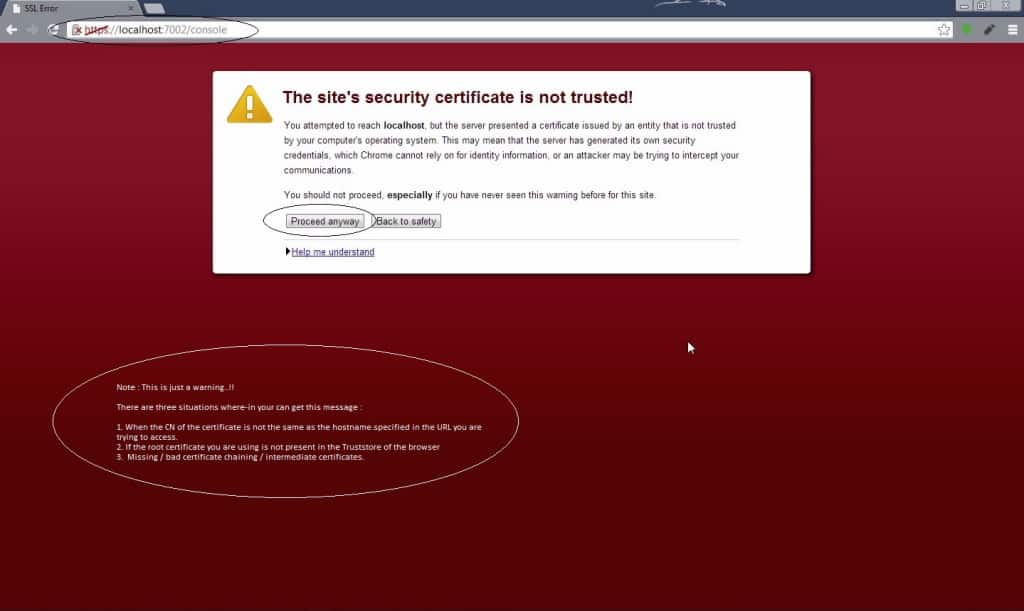
Set the " Hostname Verification: " to None ( from the drop down menu ).

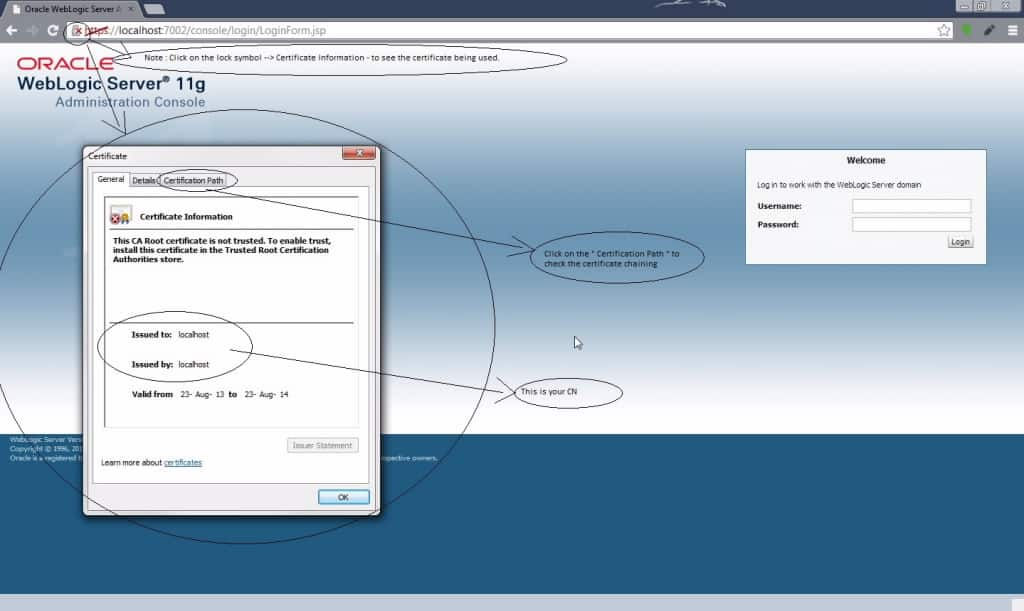
Note : We need to select the hostname verification as none if the CN of the certificate is not the same as the hostname of the machine where WLS is installed.

[](https://https/blogs.oracle.com/content/published/api/v1.1/assets/CONTA040824D11E144AAB7190E770DE3B064/Medium?cb=_cache_59da&format=jpg&channelToken=e3ef4fd8fc674b8d9e81b98087030487)

 Now access your Weblogic Admin console over https URL :

 " https://localhost:7002/console "

[](https://https/blogs.oracle.com/content/published/api/v1.1/assets/CONT409E8F2C8B00409DB2CB105A5CDD2895/Medium?cb=_cache_59da&format=jpg&channelToken=e3ef4fd8fc674b8d9e81b98087030487)

[](https://https/blogs.oracle.com/content/published/api/v1.1/assets/CONTD8AEB10618594FC9A705FD33510FB47E/Medium?cb=_cache_59da&format=jpg&channelToken=e3ef4fd8fc674b8d9e81b98087030487)



**Puneeth Prakash**

Senior Principal Software Engineer