**Security basics:**

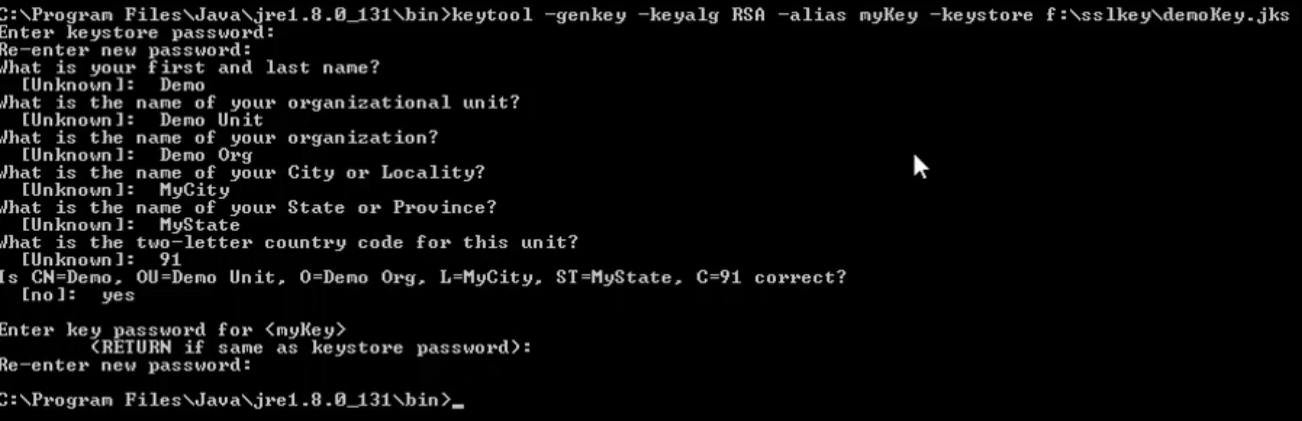
* The credentials we are passing in users.xml to login to manager app is very basic and is not secured.
* Now to make the connection through web secure, we have SSL (Secure Socket Layer).
* The connection established with this will automatically get encrypted. And end user will know how to decrypt it. To do this we need a certificate.

**Enabling SSL:**

* We get java keytool along with java installation. We can find the “keytool.exe” file in below path of windows server.

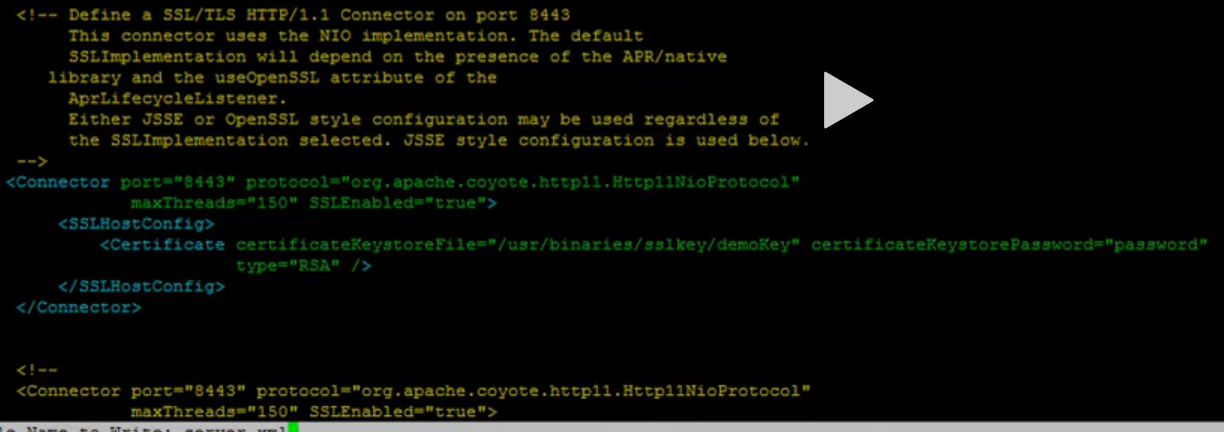
**C:/program Files/java/jre1.8.0\_131/bin**

* Open the command prompt and run the below command to generate the key.

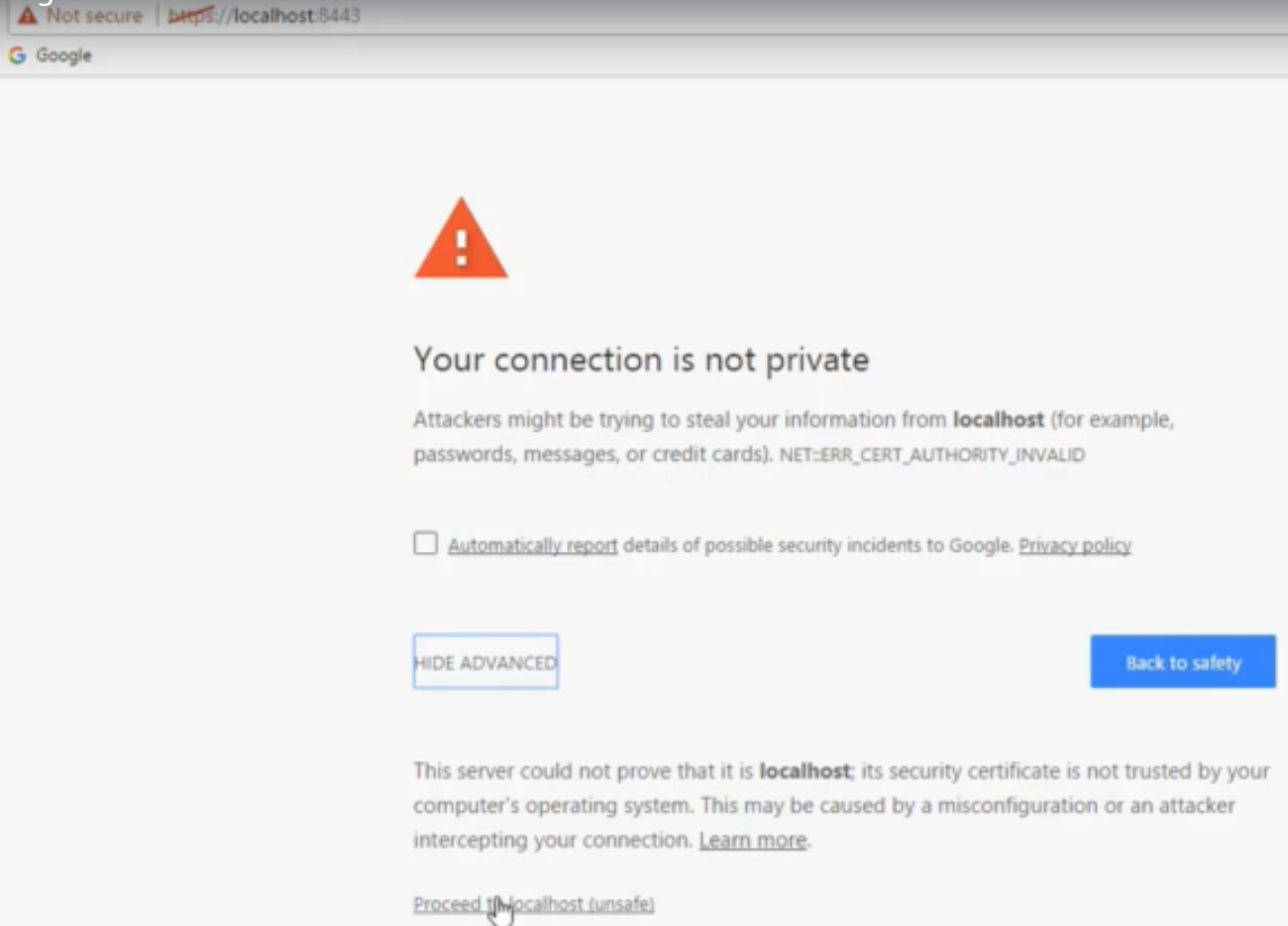


* **“keyalg”** is the algorithm, **“mykey”** is the name as mentioned in command.
* After this, we can find the key generate in the path we have given in command.
* Now, we need to modify the **server.xml** with the path of the key and the password which we have given while creating the key as below.





* Then we need to restart the tomcat.
* Now, when we try to access the tomcat console using 8443 port because that the default port of SSL, it won’t allow to access because the certificate which we created is local one and it’s not accepted because it is not from any authorized one.
* So, the website will be shown as not secure and we need to go to advanced and select to proceed to access the console as below.

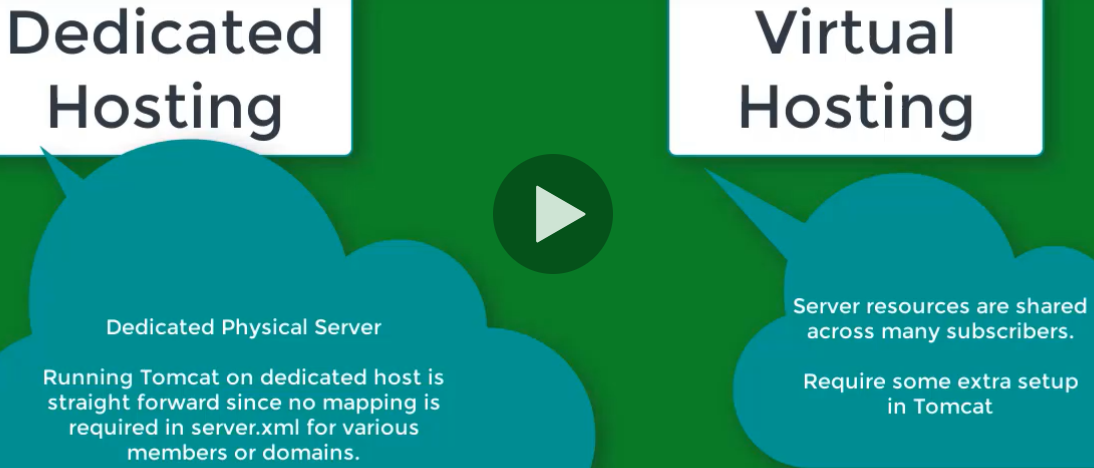


* We can see the application is opening in HTTPS port.
* Suppose, whenever user is trying to access the application on HTTP, we can always redirect it to HTTPS in web.xml file inside the webapps directory.

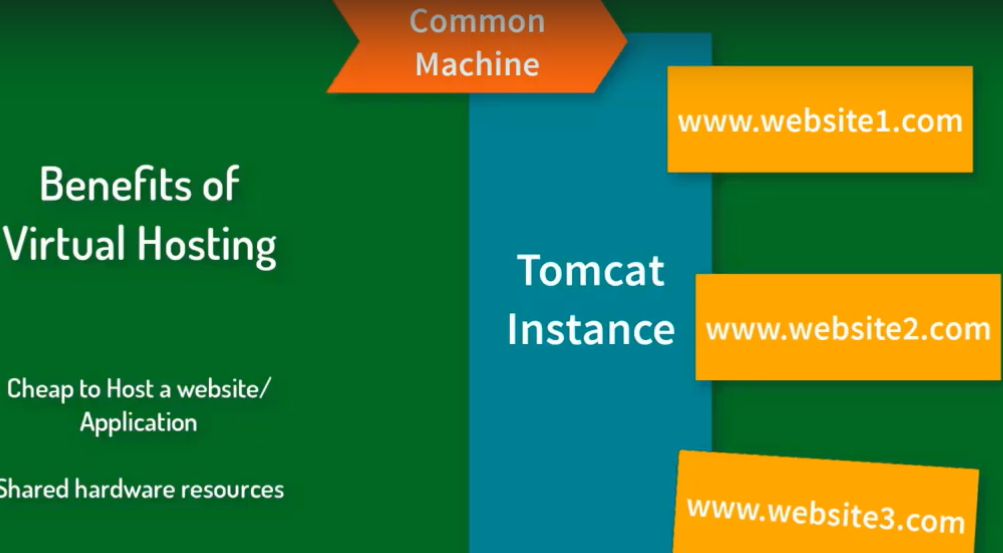
**Running tomcat on HTTPS:**

* This involves two steps. We need to get the key from certified authority like GeoTrust
* And place the key somewhere and let the tomcat know about it by changing the configuration file with the key location.
* We can see the trusted certification in manage certificates option in chrome browser settings.

**Dedicated hosting vs virtual hosting:**



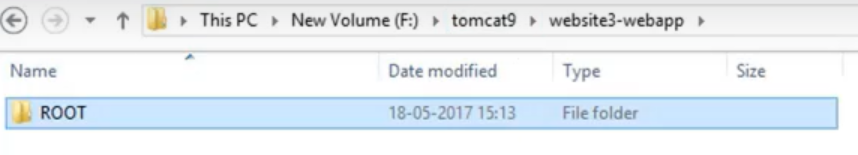
* Virtual hosting means where the multiple websites resides on same physical machine.
* We can configure multiple domains on same tomcat instance. So, every domain shares the same physical server.



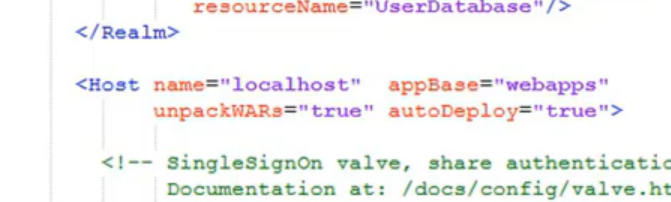
* Let’s say we want to host 3 domains, so we need to create three folders in tomcat as below.



* Now, go to each folder and create a folder called “ROOT” which to keep the relevant files as below



* Now, open conf/server.xml file. We can find the below tag



* Copy paste that tag downside of the file and modify the host name and appbase as below.



**Load balancing and clustering:**

