**Jconsole Connection Remote Process (windows based)**

***For java EE application:***

1. Open file location: .. // Apache Software Foundation/Tomcat 8.0/bin.

2. Open startup.bat file in editor like notepad++.

3. Find the variable named JAVA\_OPTS. If JAVA\_OPTS is not present in startup.bat then check catalina.bat

4. Set the java system properties for enabling JMX (Java Management Extensions)

* JAVA\_OPTS="%JAVA\_OPTS% -Dcom.sun.management.jmxremote “

# For enabling jmxremote

* JAVA\_OPTS="%JAVA\_OPTS% -Dcom.sun.management.jmxremote.port=9999"

# any free port e.g. 9999

* JAVA\_OPTS="%JAVA\_OPTS% -Dcom.sun.management.jmxremote.authenticate=false"

# authenticate = false for connecting it without user name and password

* JAVA\_OPTS="%JAVA\_OPTS% -Dcom.sun.management.jmxremote.ssl=false"

# SSL is false for connecting without security

#If SSL is true then it need some certificate or some cipher suit.

* **Optional**
* JAVA\_OPTS="%JAVA\_OPTS% -Dcom.sun.management.jmxremote.rmi.port=9999"

# it is optional but if code is using java rmi then it can be set

# Jmxremote.rmi.port may be different from jmxremote.port

5. Restart the tomcat and deploy the application

6. Open Jconsole and choose remote option type => hostip:9999(jmxremote.port)

7. It will open a pop click on insecure connection.

* **If authentication is true then following property need to be set**

JAVA\_OPTS="%JAVA\_OPTS% -Dcom.sun.management.jmxremote.access.file=d:/jmxremote/jmx.access"

JAVA\_OPTS="%JAVA\_OPTS% -Dcom.sun.management.jmxremote.password.file=d:/jmxremote/jmx.password"

* **Example:**

**1. By using user define files**

**# d:/jmxremote/jmx.access**

User read-only

Admin read write

#**d:/jmxremote/jmx.password**

User pass

Admin password

//In windows to resolve the exception related to file protection

//In command prompt go to the folders where the jmx password and access files are kept then

cacls jmx.password /P: R //switch P is used for permission :R is for read only

cacls jmx.access /P: R

**2. By using password file of *JRE\_HOME*/lib/management**

***To set up a Single-User Environment***

You set up the password file in the *JRE\_HOME*/lib/management directory as follows.

1. Copy the password template file, jmxremote.password.template, to jmxremote.password.
2. Set file permissions so that only the owner can read and write the password file.
3. Add passwords for roles such as monitorRole and controlRole.

***To set up a Multiple-User Environment***

You set up the password file in the *JRE\_HOME*/lib/management directory as follows.

1. Copy the password template file, jmxremote.password.template, to your home directory and rename it to jmxremote.password.
2. Set file permissions so that only you can read and write the password file.
3. Add passwords for the roles such as monitorRole and controlRole.
4. Set the following system property when you start the Java VM.
5. com.sun.management.jmxremote.password.file=*pwFilePath*

**Note:** In the above property, *pwFilePath* is the path to the password file.

**For Java Application**

Java -Dcom.sun.management.jmxremote.port=3000

com.sun.management.jmxremote.authenticate=false

com.sun.management.jmxremote.ssl=false

com.cyb.demo.MyFirstApp // fully qualified class name

**References**

[**https://docs.oracle.com/javase/8/docs/technotes/guides/management/agent.html**](https://docs.oracle.com/javase/8/docs/technotes/guides/management/agent.html)

[**http://docs.oracle.com/javase/7/docs/technotes/guides/management/jconsole.html**](http://docs.oracle.com/javase/7/docs/technotes/guides/management/jconsole.html)

[**https://dzone.com/articles/how-enable-jmx-authentication**](https://dzone.com/articles/how-enable-jmx-authentication)