

## **Tomcat Real Time issue and Solutions**

### **Issues1:**

#### **Tomcat wrong Datasource Definition:**

Javax.SQLNet exception, when your application running on the Tomcat sever and trying to connect with the database. If you give wrong IP/Hostname of the database in Datasources, then your application will not able to connect with the database through the tomcat.

### **Issue2:**

#### **Someone changed the database username and password in the database:**

Yesterday your application in the tomcat is running fine, and today it's not working fine. Check in the Tomcat logs. If you find, Login Denied, Invalid Username/Password, you need to check the username or password in the database. If someone changed the database username or password, your tomcat will not able to connect with the database.

### **Issue3:**

#### **Someone changed the database port number:**

Your application is trying to connect with the database on 1521 (In case of Oracle Database), if someone changed the port number of the database to 1522, then your tomcat will not connect with the database.

### **Issue4:**

#### **Port Conflict exception:**

You have multiple tomcats running on the same server (One of them running on 8080). If you want to start another tomcat instance to be run on the same port number. You will see the BindException in logs.

### **Issue5:**

**When your tomcat server is talking to another application which is running on the different server.** The remote application port should be open for your Tomcat server. Otherwise, you will get an error.

telnet <remotehost IP/Hostname> <port Number>  
it should not give "Connection Refused".

### **Issue6:**

#### **Deployment in Tomcat servers failed:**

This is because, your tomcat server should run with the dedicated user. If you try to run the tomcat with different user (root) and trying to deploy in tomcat server, it will not deploy.

### **Issue6:**

#### **OutOfMemory Issue:**

When your tomcat server needs 3 GB of RAM and you have allocated only 2 GB of RAM, you will get Tomcat OutOfMemory error in Tomcat server logs.

### **Issue7:**

#### **Application Slowness**

### **Issue8:**

#### **SSL certificates error:**

You have imported SSL certificates successfully into your Tomcat server. After few days, certificates will get expire, if you don't renew the SSL certificates, you will see SSL Handshake exception, through which you can't provide security to your websites.

### **Issue9:**

Tomcat integration with Apache Web Server. Your Tomcat instances are running on different machines and you are trying to integrate them with Apache Web Server which is running on another machine. In this situation, you need to have proper connection parameters and required ports should be allowed.

#### **Issue10:**

##### **User authentication failed:**

When your user tomcat server is connecting with the external LDAP and if your password expired after 90 days (Password expire policy), then you will need to investigate the log files for errors.

here are some issues in Tomcat

sometimes we will get 400 series and 500 series errors. we can troubleshoot those errors by seeing log files only

Error troubleshooting depends on the error that we are seeing in the log files.

#### **Issue11:**

##### **When we deploy the new version of application, still the end user getting the old output.**

Solution: Remove temp and work directories under \$TOMCAT\_HOME or \$CATALINA\_HOME and restart the Tomcat

work and temp refers the Tomcat server level cache

#### **Issue12:**

When we are upgrading the Tomcat version from Tomcat7 to Tomcat8, we may miss some pre-requisites ... In this case, our server migration may not be proper.

Server migration depends on lot of parameters like OS infrastructure, tomcat heap memory, disk space

#### **Issue13:**

when someone change server.xml (with improper settings), and your tomcat instance is not coming up (Not starting properly)..... then you have to validate the server.xml...

In this case, you just run configtest.sh under \$TOMCAT\_HOME/bin or \$CATALINA\_HOME/bin this script validates your server.xml and Java path settings etc.

#### **Tomcat customization:**

##### **Log level:**

we can change the log levels in tomcat by editing logging.properties file under conf directory

log file location:

you need to edit server.xml file

```
<Valve className="org.apache.catalina.valves.AccessLogValve" directory="logs"
      prefix="localhost_access_log" suffix=".txt"
```

webapps location:

you have to edit server.xml file appBase="webapps"

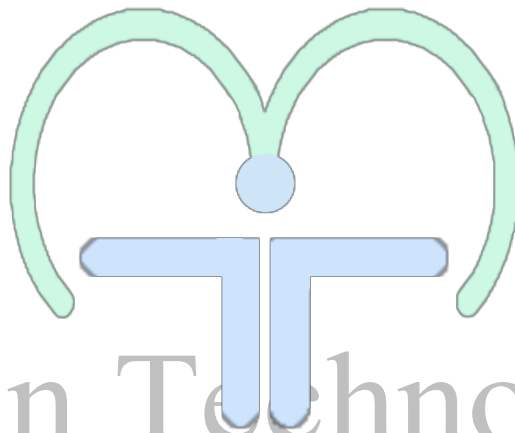
instead of webapps, we can give any custom location, From that location, our war files will pick-up

unpackWARs="true"

this means, if you place your war file inside webapps directory it will extract automatically.

autoDeploy="true"

once you start your tomcat, automatically your applications inside tomcat will start automatically



Mithun Technologies  
Mithun  
Technologies