**Performance Testing Activity**

**Table of Contents**

1. JMeter - Distributed Testing Environment……………………………………….2
2. JMeter - Integration with Jenkins…………………………………………………….3
3. JMeter - Database Testing……………………………………………………………….4
4. JMeter - Mobile Performance Testing………………………………………………5
5. JMeter - Email Reader (Gmail)………………………………………………………….7
6. JMeter - HTTP(s) Live Streaming Testing (HLS)………………………………….8
7. JMeter - Generate Report Dashboard…………………………………………….10
8. JMeter - FTP..…………………………………………………………………………………12
9. JMeter - Write Response to File……………………………………………………..13

**1. JMeter-Distributed(Remote) Testing Environment**

**Master & Slave concept:**

**Pr-Requisites :**

1. Make sure all the machines are in the same subnet. If machines are not same subnet then they will not recognize each other in the network.

2. Make sure jmeter can access the server(AUT).

3. Make sure the same version of Jmeter are installed on all the machines(Master & Slaves) to avoid unanticipated issues.

4. The Firewall on all systems are turned off because firewall may be blocking the traffic.

**Terminology :**

**Master** - The machine running JMeter GUI, which controls each slave & the test.

**Slave** - The machine running jmeter-server.bat, which takes commands from the Master Jmeter GUI and send the request to server under test.

**Target** - The web application server under test, get request from slaves.

**Steps** :

1. On the Slave machine, using cmd prompt go to the jmeter/bin folder and enter the command.

"jmeter-server -Djava.rmi.server.hostname=<ip address of the slave machine>"

Initially, slave machine are private. So in order to make it public the above command is used on slave machine using command prompt.

2. On slave machine, go to jmeter/bin folder and the execute the file "jmeter-server.bat" where you can see the "Created remote object: UnicastServerRef.... along with ip address. Note down the ip address.

3. Now go to Master machine to the jmeter/bin folder and open & edit the jmeter.properties file by adding the slave ip address.

Search for the "remote\_hosts" and add the ip address of the each slave with comma delimited then save and close the file.

4. Open Jmeter GUI in the Master machine and open test plan.

5. Click Run on the menu bar of the Jmeter GUI, select Remote Start -> select the ip address of the slave machine from where the load has to be pumped or can select Remote Start All where load will be pumped from all the slave machines on to the AUT.

**Reference:**

<https://www.youtube.com/watch?v=qfPI9vmOBhE>

**2. JMeter Integration with Jenkins**

**JMeter** - Leading open source tool for Performance Testing

**Jenkins** - Leading open source tool for Continuous Integration

Jenkins - A java application and platform independent which can be used on any platform.

Jenkins used for continuous integration and continuous delivery.

**Download process:**

1. Download Jenkins war file from the url : https://jenkins.io

2. Place the war file on any location on your system

3. Go to command prompt->to the folder where the jenkins.war is present.

Enter command "java -jar Jenkins.war" and click enter

4. Go to browser and enter http://localhost:8080 (8080 is the port on which jenkins will be running)

Unlock jenkins by the useername:admin & password which created in the C:\Users\pradeep.savulla\.jenkins\secrets\initialAdminPassword.txt

Inorder to run jmeter scripts from jenkins, you need to deploy the performance plugin. You have two ways to do that.

1. Login to the jenkins, click on the Manage Jenkins, click on manage plugins, goto Available tab and search/check for the Performance plugin, install and restart.

2. Download jenkins performance.hpi, put into Plugins folder and then restart the jenkins.

Goto Jmeter/bin->open user.properties file - add line "jmeter.save.saveservice..output\_format=xml"

**Create JMeter test :**

1. Run the test plan from the command line interface to ensure the test runs in jenkins because the same command has to be used in jenkins

Command to be enter in cmd prompt is given below:

jmeter -jjmeter.save.saveservice.output\_format=xml -n -t E:\Folder\Folder\Testplan.jmx -l E:\Folder\Folder\TestReport.jtl

2. Login to jenkins and add a job in jenkins by clicking on create new jobs.

Enter an Item name say jmeter\_eCommerceDemo -> select Freestyle Project and click on OK and configuration page displays.

3. Now go to Build section and click on Add build step drop down.

4. As you are on windows system, select Execute windows batch command.

Enter the command: cd E:\JMeter\apache-jmeter-3.3\bin\jmeter -jjmeter.save.saveservice.output\_format=xml -n -t E:\JMeter\Scripts\eCommerce\_demo.jmx -l E:\JMeter\Scripts\eCommerce\_demo.jtl

5. Next, go to Add Post-build action.

6. Select Publish Performance test result report configure necessary chart which you want to display during execution and then save it

7. Run

**References:**

<https://www.youtube.com/watch?v=mYGQPj78YhI>

**3. JMeter Database Testing**

**Pre-Requisite:** MySQL database.

This configuration used to hit/query/request the MySQL database and get the response back to jmeter.

**Steps:**

1. Download the MySQL jdbc jar from the below URL.

https://dev.mysql.com/downloads/connector/j/5.0.html

2. Copy the mysql-connector-java-5.1.44-bin.jar(mysqljdbc jar) to the jmeter/lib folder.

3. Start Jmeter GUI from the jmeter/bin folder.

4. To the Test plan add Thread group-> Right click and add JDBC connection configuration element from the config element.

5. At Variable Name Bound to pool section provide the Variable Name which is mandatory.(Ex: mydb)

6. Let the values be default for Connection pool, Validation by pool.

7. At Database Connection Configuration section: Enter the required values.

Ex: For MySQL jdbc connection URL is

Database URL: jdbc: mysql: //localhost:3306/world

JDBC Driver class: com.mysql.jdbc.Driver

Username: root

Password: ....

8. Go to JDBC Request-> Enter same variable name which is specified at JDBC Connection configuration

EX: mydb

9. Select the Query type whether statement is of Select/Update/Callable statement depending on your requirement.

Ex: Select Statement

10. Enter the select query to connect db and which you want to retrieve data.

Ex: select \* from world.countrylanguage;

11. Can also add Response assertion to the jdbc request.

12. Enter the response to check in the patterns to test of Response assertion and apply it to Jmeter variable where you can give the column and row number to validate the response.(Ex: col2\_2 means column no.2 and second row)

13. In the jdbc request, provide the column number to be validated.( Ex: ,col2,,)

14: Add View Results Tree to the Thread group and Run.

References :

<https://www.youtube.com/watch?v=oy53KAKHpts>

<https://www.youtube.com/watch?v=SmSieTKYky8>

<https://www.youtube.com/watch?v=fH03i2FjYa8>

**4. JMeter - Mobile Performance Testing**

**JMeter Script development**

**Pr-Requisites :**

1. Both Desktop/Laptop and Mobile device must be on the same Wlan.

1. JMeter version 3.0, Android phone (versions above Jellybean) or iPhone (version 8.0 onwards)

**Steps:**

1. Start Jmeter on your test machine(desktop/laptop).

2. Create a test plan with Thread Group and necessary elements in jmeter or can create directly using a Recording Template.

3. Goto HTTP(S) Test script recorder on Workbench. At Global settings, set port for recording i.e give any port(Ex:8181). And set url patterns to Exclude.

4. Create Root CA certificate. This is present in the Jmeter bin directory. In case if your recording in jmeter for the first time then Root CA certificate:ApacheJMeterTemporaryRootCA created in JMeter bin directory, when you click on Start button at Http(s) Test Script Recorder.

5. Copy the same Root CA certificate to the Mobile Device and install the certificate.

**Configure the Mobile Device:**

6. Make sure Mobile device and Test machine is on the same wlan(Network) or connect to the same Wi-Fi.

7. Note the ip of the test machine. Can use ip-config command in the command prompt to get the ip address.

8. Goto Settings in the Mobile device, click WiFi connections and long press the Wi-Fi you are connected to.

9. Click on Modify Network->on the Menu options you have Show advanced option check that, go down at Proxy select Manual-> give the ip address of the test machine as hostname, enter port number which configured in jmeter Test script recorder Global settings, select DHCP for IPv4 settings and click save. Now our Android mobile is ready with set-up.

10. Start Recording : Actions which you do on mobile device should get recorded in Jmeter. To get this goto JMeter>Http(s) Test script recorder->click on Start. Now on Mobile device, do all the actions on the application which you want to capture the request. As ip address & port no is set as proxy, all the requests gets captured in the Jmeter Controller.

11. After recording is done click stop in Test script recorder. Then add listeners to the Thread group and run the script(Do enhancements to the script if required for the script to pass).

References :

<https://www.youtube.com/watch?v=JFDcgKQ8VFw>

<http://blogs.quovantis.com/mobile-performance-testing-using-jmeter/>

<https://www.youtube.com/watch?v=vfs3sSuFQno>

**5. JMeter - Email Reader (GMail)**

**How to read Emails from the server.**

**Mail Server : Gmail**

1. Download javax.mail.jar from the below url.

https://javaee.github.io/javamail/

2. Copy the jar from the download folder and paste it to the jmeter/lib folder.

3. Create a Test plan -> Thread Group -> Right click select MailReaderSampler.

4. Add required data to Mail reader sampler. Before this, we have to know about the incoming email server, port number and protocol.

5. Enter protocol(Ex: imaps), Server Host(Ex:imap.gmail.com),Server port(Ex:993), Username(Ex:abcd@gmail.com), Password(Ex:\*\*\*\*\*\*\*\*) & FOLDER to read emails from(Ex: INBOX).

6. Also enter Number of message to read. (ALL or specific number to read)

7. Check SSL at security settings.

8. Now, login to the mail -> goto settings -> goto Forwarding and POP/IMAP

At IMAP Access : Enable IMAP

9. Goto My Account setting and Allow less secure apps : ON(To make sure any external clients access the mail(gmail) server.

https://myaccount.google.com/lesssecureapps

1. Goto JMeter>Add View Listener Tree to Thread Group and run the script.

**References :**

<https://www.youtube.com/watch?v=v4d3RtqwyZU>

<https://developers.google.com/gmail/imap/imap-smtp>

<https://support.google.com/mail/answer/7126229?hl=en&visit_id=0-636491794211230232-3278797818&rd=1#ts=1665018>

<https://www.youtube.com/watch?v=-aWjy9E0GUE>

<https://stackoverflow.com/questions/35859794/mail-reader-sampler>

**6. JMeter - HTTP(s) Live Streaming Testing (HLS) Protocol**

This provides a reliable, cost-effective means of delivering continuous and long-form video over the Internet.

It allows a receiver to adapt the bit rate of the media to the current network conditions, in order to maintain uninterrupted playback at the best possible quality.

**In a HTTP Live Streaming process:**

The audio/video to be streamed is reproduced by a media encoder at different quality levels, bit rates and resolutions. Each version is called a variant.

The different variants are split up into smaller Media Segment Files.

The encoder creates a Media Playlist for each variant with the URLs of each Media Segment.

The encoder creates a Master Playlist File with the URLs of each Media Playlist.

To play: The client first downloads the Master Playlist, then the Media Play lists and then they play each Media Segment declared within the chosen Media Playlist. The client can reload the Playlist to discover any added segments. This is needed in cases of live events.

**For example:**

Playlist files must be encoded in UTF-8. The format of the Playlist files is derived from the M3U playlist file format. Each Playlist file must be identifiable by either the path component of its URL or by HTTP Content-Type. In the first case, the path must end with either .m3u8 or .m3u. In the second, the HTTP Content-type must be **"**application/vnd.apple.mpegurl**"** or **"**audio/mpegurl**"**.

**The HLS Plugin for JMeter :**

The new HLS plugin for JMeter allows users to simulate a HLS scenario by using only one custom Sampler. Now, you don’t need multiple HTTP Request Samplers, ForEach Controllers or RegEx PostProcessors. This makes the whole process much simpler than before.

Instead, the complete logic is seamlessly encapsulated so you only have to care about the use case: the media type, playback time and network conditions. That’s it! The plugin is brand new and it can be installed via the JMeter Plugins Manager.

**Steps : To create your test**

Install the HLS plugin from the Plugins Manager

1. Create a Thread Group.

2. Add the HLS Sampler i.e Right click Thread Group -> Add -> Sampler -> HLS Sampler.

After that you can add assertions, listeners, etc.

**Video Options :**

3. Set the link to the master playlist file at URL

4. Set the video type corresponding to the playlist i.e VOD,Live Stream & Event Stream

**Play Options :**

5. Set the playback time of the test: Whole video or Video duration(seconds)

**Network options :**

6. Select the protocol of the playlist you want to test. You can identify it in the link to the master playlist file: http or https

7. Select the bandwidth you want to simulate in your test. If there is only one playlist for the selected bandwidth, the plugin will select the playlist based only on this criterion i.e Custom Bandwidth (bits/s), Min bandwidth available or Max bandwidth available

**Resolution :**

8. After selecting the desired bandwidth you can select a resolution to simulate your specific device.

**Results :**

9. You can set listeners to evaluate the results of your tests. The View Results Tree Listener displays the HLS Hierarchy, so you can inspect how the requests and responses worked.

This is the most easiest way to load test HLS using JMeter.

**References :**

Using HLS sampler:

<https://www.blazemeter.com/blog/the-new-hls-plugin-for-jmeter-the-complete-guide>

Using HTTP,RegEx & For-Loop :

<http://www.tothenew.com/blog/load-testing-hls-stream-using-jmeter>

<https://www.youtube.com/watch?v=q_pMq_U0ng8>

<https://www.blazemeter.com/blog/how-load-test-http-live-media-streaming-hls-jmeter>

**7. JMeter - Generate Report Dashboard**

**Configurations to be done to generate Report Dashboard in HTML format.**

1. Open reportgenerator.properties from the jmeter/bin directory.

2. Copy all the Report Configuration elements.

3. Then open user.properties file from the jmeter/bin directory.

4. Paste Report configuration elements into user.properties file then save it.

5. Open saveservice properties file from the jmeter/bin folder.

6. Copy the Save Service configuration elements.

7. Paste Save Service configuration elements into user.properties file and save it.

8. Also uncomment required Results file configuration elements in the jmeter.properties file from the jmeter/bin folder.

#---------------------------------------------------------------------------

# **Results file configuration**

#---------------------------------------------------------------------------

# This section helps determine how result data will be saved.

# The commented out values are the defaults.

# legitimate values: xml, csv, db. Only xml and csv are currently supported.

jmeter.save.saveservice.output\_format=csv

# true when field should be saved; false otherwise

# assertion\_results\_failure\_message only affects CSV output

#jmeter.save.saveservice.assertion\_results\_failure\_message=true

# legitimate values: none, first, all

jmeter.save.saveservice.assertion\_results=none

jmeter.save.saveservice.output\_format=xml

jmeter.save.saveservice.data\_type=true

jmeter.save.saveservice.label=true

jmeter.save.saveservice.response\_code=true

# response\_data is not currently supported for CSV output

jmeter.save.saveservice.response\_data=true

# Save ResponseData for failed samples

jmeter.save.saveservice.response\_data.on\_error=false

jmeter.save.saveservice.response\_message=true

jmeter.save.saveservice.successful=true

jmeter.save.saveservice.thread\_name=true

jmeter.save.saveservice.time=true

jmeter.save.saveservice.subresults=true

jmeter.save.saveservice.assertions=true

jmeter.save.saveservice.latency=true

# Only available with HttpClient4

jmeter.save.saveservice.connect\_time=true

jmeter.save.saveservice.samplerData=false

jmeter.save.saveservice.responseHeaders=false

jmeter.save.saveservice.requestHeaders=false

jmeter.save.saveservice.encoding=false

jmeter.save.saveservice.bytes=true

# Only available with HttpClient4

#jmeter.save.saveservice.sent\_bytes=true

#jmeter.save.saveservice.url=false

#jmeter.save.saveservice.filename=false

jmeter.save.saveservice.hostname=false

jmeter.save.saveservice.thread\_counts=true

jmeter.save.saveservice.sample\_count=false

#jmeter.save.saveservice.idle\_time=true

9. Restart the Jmeter.

10. Develop any test plan.

11. Add simple data writer/summary listener to Thread Group and create the test results in csv format.

12. Before Running your test plan we should configure below fields in Simple Data Writer window panel,those are

timeStamp, elapsed, label, responseCode, responseMessage, threadName, dataType, success, failureMessage, bytes, grpThreads, allThreads, Latency, IdleTime

12. Open Cmd prompt -> goto the jmeter/bin folder and enter the below command which saves all the results in HTML in a graphical format.

jmeter -g <Location of test results csv file> -o <Location of the HTML to save>

**References:**

<https://learn-jmeter.blogspot.in/2016/10/how-to-generate-jmeter-report-dashboard.html>

<https://www.youtube.com/watch?v=YbQHL8lTnZ4>

<http://jmeter.apache.org/usermanual/generating-dashboard.html>

**8. JMeter - FTP**

**To test FTP Upload & Download using JMeter**

**FTP Url:** <http://www.swfwmd.state.fl.us/data/ftp/>

**Server**: ftp.swfwmd.state.fl.us

**User Name**: Anonymous

**Password** : any valid email address

**Steps:**

1. Create a Test plan add Thread Group and to it , add FTP request sampler.

2. Enter the FTP connection parameters. Server details:ftp.swfwmd.state.fl.us in the FTP request Server Name or IP.

3. Enter username : Anonymous & Password : any valid email address.

4. Inorder to get the file or download to the local system then check GET(RETR) & if you want the local file to upload to the FTP server, then check PUT(STOR).

5. Check Save File in Response : To check the file saved in the response of the Jmeter i.e View Results Tree (file downloaded or uploaded)

5. Goto the FTP location to check what files are present and based on that you can Get(download) or Put(upload).

6. At Remote file, enter the path of the remote FTP server file location. (Ex: /README.txt)

7. At Local File, enter the path of the local system where you want the file to be downloaded.(Ex:E://Jmeter\_Scripts/FTP)

8. Add view results tree to the Thread Group and run the script. You may also add the FTP Request Defaults, so that there is no need of adding the server details in every request.

**References:**

<https://www.youtube.com/watch?v=tLtHW6edVlo>

<http://www.swfwmd.state.fl.us/data/ftp/>

**9. JMeter - Write Responses to File**

**How to export/save test results to csv/jtl/xml file in Jmeter.**

**Steps:**

1. Develop the jmeter script with all the enhancements required for a successful script.

2. Add the Simple Data Writer to the Thread group.

3. Thread Group -> Right click Add -> goto Listener -> select Simple Data Writer.

4. Simple Data Writer writes the test results to the csv/jtl/txt/xml file.

Here in this example, using csv as it is easier to generate dashboard report in JMeter.

5. click on SimpleDataWriter, give/browse the path where the csv file has to be saved in the Filename field.

6. End of the path, give the filename where all the results will be save to it.

7. Save the testplan and run the jmeter script which will save all test results into file using Simple Data Writer Listener.

**Reference:**

<https://www.youtube.com/watch?v=AnpX7xGImNk>