



INTENTWISE API TESTING

Performance Test – Cook Book



FEBRUARY 7, 2019

p

Contents

Modifying the Workload Model.....	2
Modifying the Server/End Point	3
Running the test.....	4
Stopping the test.....	4
Kibana Dashboard:.....	5

Modifying the Workload Model

Workload model refers to “Concurrent Load, ramp-up, Iterations/Test Duration”. Below are the steps to modify the concurrent load & Test Duration, etc... for the Group1 & Group2

1. Connect to Server “ec2-34-219-58-127.us-west-2.compute.amazonaws.com”
2. Navigate to path : “/home/ubuntu/ShellScripts”
3. Edit Shell Script : **runGroup1Scripts** (or runGroup2Scripts)

Command: nano runGroup1Scripts

```
#This Script Contains APIs : SP and SB Campaigns - Without Compare, Ad Group - Without Compare, SP KWs - Without Compare, SB KWs - Without Compare, SP $
JMeterPATH="/home/ubuntu/apache-jmeter-5.2.1/bin/"
SCRIPTPATH="/home/ubuntu/scripts/"
SCRIPTNAME="Intentwise_API_Group1_V1.4.jmx"

#Define the load below for each API
SPSBLoad=2
AdGroupLoad=2
SPKWLoad=2
SBKWLoad=2
SPProductsLoad=2

#Define Rampup in Seconds
Rampup=30

#Define Iterations/Loops for each API
Iterations=3

if [ "$1" == "" ]
then
    echo "Command Error : Please Provide the Result Version"
    echo "Usage : ./runGroup1Scripts <Version>"
else
    #Jmeter Command to Run
    echo "Running Test..."
    /$JMeterPATH/jmeter -n -t $SCRIPTPATH$SCRIPTNAME -JVersion=$1 -JRampup=$Rampup -JIterations=$Iterations -JSPSBLoad=$SPSBLoad -JAdGroupLoad=$AdG$
    echo "Test Completed"
fi
```

4. Ramp-up : Refers with in how many seconds , all users should be coming-into application

```
#This Script Contains APIs : SP and SB Campaigns - Without Compare, Ad Group - Without Compare, SP KWs - Without Compare, SB KWs - Without Compare, SP $
JMeterPATH="/home/ubuntu/apache-jmeter-5.2.1/bin/"
SCRIPTPATH="/home/ubuntu/scripts/"
SCRIPTNAME="Intentwise_API_Group1_V1.4.jmx"

#Define the load below for each API
SPSBLoad=2
AdGroupLoad=2
SPKWLoad=2
SBKWLoad=2
SPProductsLoad=2

#Define Rampup in Seconds
Rampup=30

#Define Iterations/Loops for each API
Iterations=3

if [ "$1" == "" ]
then
    echo "Command Error : Please Provide the Result Version"
    echo "Usage : ./runGroup1Scripts <Version>"
else
    #Jmeter Command to Run
    echo "Running Test..."
    /$JMeterPATH/jmeter -n -t $SCRIPTPATH$SCRIPTNAME -JVersion=$1 -JRampup=$Rampup -JIterations=$Iterations -JSPSBLoad=$SPSBLoad -JAdGroupLoad=$AdG$
    echo "Test Completed"
fi
```

5. Iterations :

```
#This Script Contains APIs : SP and SB Campaigns - Without Compare, Ad Group - Without Compare, SP Kws - Without Compare, SB Kws - Without Compare, SP $
JMETERPATH="/home/ubuntu/apache-jmeter-5.2.1/bin/"
SCRIPTPATH="/home/ubuntu/scripts/"
SCRIPTNAME="Intentwise_API_Group1_V1.4.jmx"

#Define the load below for each API
SPSBLoad=2
AdGroupLoad=2
SPKWsLoad=2
SBKWsLoad=2
SPProductsLoad=2

#Define Rampup in Seconds
Rampup=30

#Define Iterations/Loops for each API
Iterations=3 Steady State

if [ "$1" == "" ]
then
    echo "Command Error : Please Provide the Result Version"
    echo "Usage : ./runGroup1Scripts <Version>"
else
    #Jmeter Command to Run
    echo "Running Test..."
    /$JMETERPATH/jmeter -n -t $SCRIPTPATH$SCRIPTNAME -JVersion=$1 -JRampup=$Rampup -JIterations=$Iterations -JSPSBLoad=$SPSBLoad -JAdGroupLoad=$AdG$
    echo "Test Completed"
fi
```

Modifying the Server/End Point

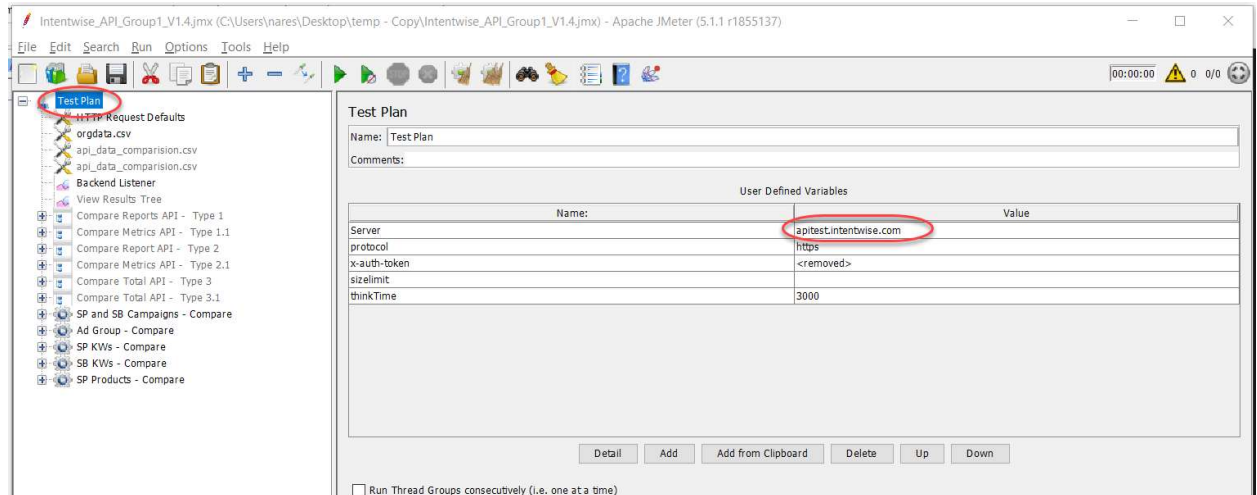
Changing on Linux Server:

1. Navigate to path `"/home/ubuntu/scripts"`
2. Edit the Script `"Intentwise_API_Group1_V1.4.jmx"` or `"Intentwise_API_Group2_V1.6.jmx"`
3. Search for `"apitest.intentwise.com"`
4. Change the server as per your need.

```
<?xml version="1.0" encoding="UTF-8"?>
<jmeterTestPlan version="1.2" properties="5.0" jmeter="5.1.1 r1855137">
  <hashTree>
    <TestPlan guiclass="TestPlanGui" testclass="TestPlan" testname="Test Plan" enabled="true">
      <stringProp name="TestPlan.comments"></stringProp>
      <boolProp name="TestPlan.functional_mode">false</boolProp>
      <boolProp name="TestPlan.tearDown_on_shutdown">true</boolProp>
      <boolProp name="TestPlan.serialize_threadgroups">false</boolProp>
      <elementProp name="TestPlan.user_defined_variables" elementType="Arguments" guiclass="ArgumentsPanel" testclass="Arguments" testname="User Define$
        <collectionProp name="Arguments.arguments">
          <elementProp name="Server" elementType="Argument">
            <stringProp name="Argument.name">Server</stringProp>
            <stringProp name="Argument.value">apitest.intentwise.com</stringProp>
            <stringProp name="Argument.metadata">=</stringProp>
          </elementProp>
          <elementProp name="protocol" elementType="Argument">
            <stringProp name="Argument.name">protocol</stringProp>
            <stringProp name="Argument.value">https</stringProp>
            <stringProp name="Argument.metadata">=</stringProp>
          </elementProp>
          <elementProp name="x-auth-token" elementType="Argument">
            <stringProp name="Argument.name">x-auth-token</stringProp>
            <stringProp name="Argument.value">a86c1b84-5d9c-40ff-8b03-dd457d0b753d</stringProp>
            <stringProp name="Argument.metadata">=</stringProp>
          </elementProp>
          <elementProp name="sizelimit" elementType="Argument">
            <stringProp name="Argument.name">sizelimit</stringProp>
            <stringProp name="Argument.value"></stringProp>
            <stringProp name="Argument.metadata">=</stringProp>
          </elementProp>
          <elementProp name="thinkTime" elementType="Argument">
            <stringProp name="Argument.name">thinkTime</stringProp>
            <stringProp name="Argument.value"></stringProp>
            <stringProp name="Argument.metadata">=</stringProp>
          </elementProp>
        </collectionProp>
      </TestPlan>
  </hashTree>
</jmeterTestPlan>
```

Changing using Jmeter GUI:

- Navigate to <Jmeter folder>/bin directory on your local.
- Open Jmeter.bat or ApacheJMeter.jar
- Now Navigate to Menu, File->Open and then select the respective script on your local machine.
- Now navigate to component "Test Plan"



”, change the server as below

Running the test

- Navigate to `/home/ubuntu/ShellScripts`
- Run the Shell scripts based on your requirement , i.e., `runGroup1Scripts` or `runGroup2Scripts` as below
 - Command Syntax : `runGroup1Scripts <Version>`
 - Example : `runGroup1Scripts V.1.1`
 - Use `&` at the end to run in back ground mode, so that you can safe stop whenever you need using Stop test command.
 - Example : `runGroup1Scripts V.1.1 &`

Stopping the test

Force Stop/Immediate Stop:

- Navigate to directory : `/home/ubuntu/ShellScripts`
- To Stop the test interactive mode press “`Ctrl+C`” to terminate the test
- To Stop the test which is running in background mode:
 - `./stopTestForcefully`

Graceful Stop:

- Navigate to directory : `/home/ubuntu/ShellScripts`

- Top Stop the test which is running in back ground mode run the below shell script
 - `./stopTestGracefully`

Kibana Dashboard:

- Kibana URL : <http://ec2-54-202-191-186.us-west-2.compute.amazonaws.com:5601/>
 - Admin Credentials : **elastic/Intentwise14**
 - Dashboard Users : **girish/Intentwise14**
- Elastic Search URL : <http://ec2-54-202-191-186.us-west-2.compute.amazonaws.com:9200/>
 - Admin Credentials : **elastic/Intentwise14**