Jmeter Performance Testing

Winter 2016

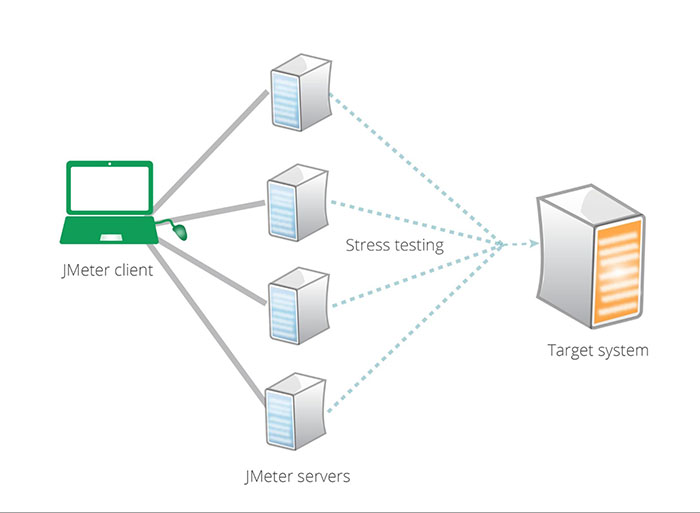
Contents

[1. Week 8: HTTP Simple Table Server, Json Format, Bean Shell Script 4](#_Toc467783108)

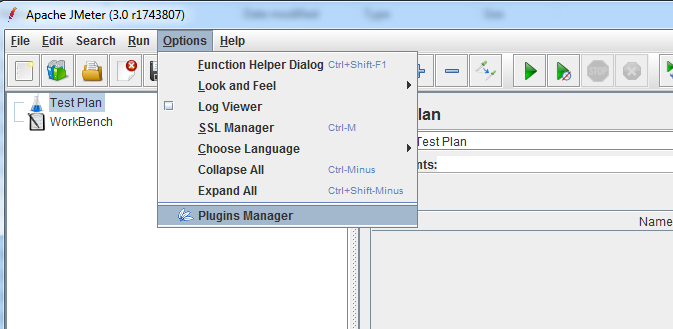
[2. Contact Us 12](#_Toc467783109)

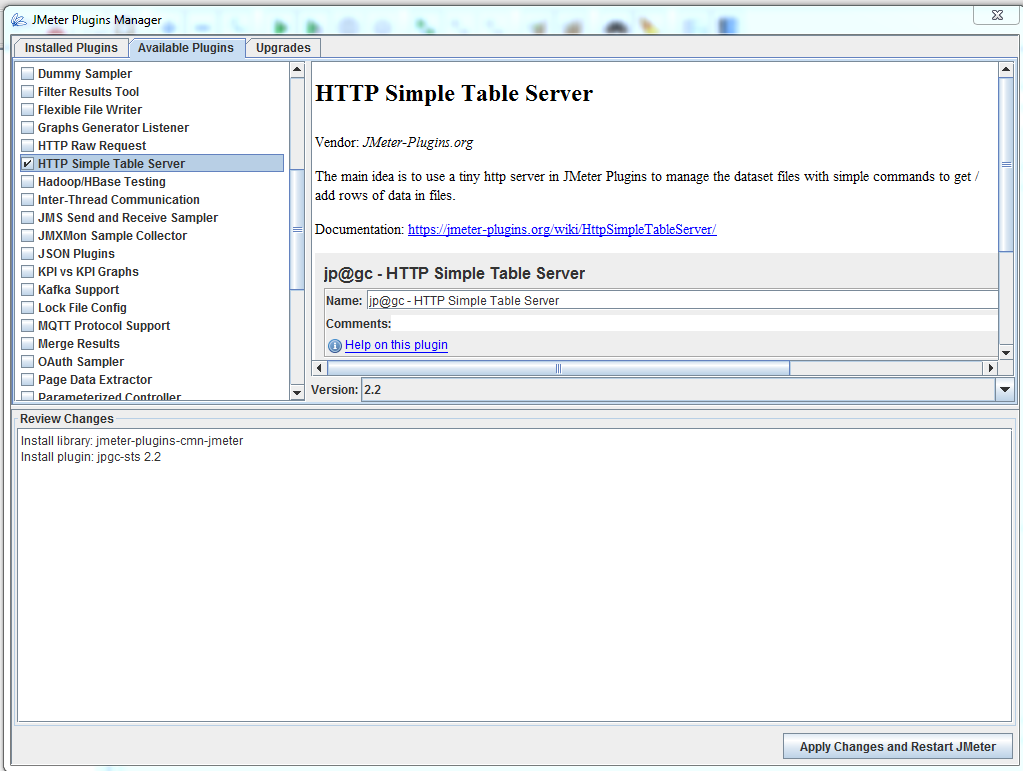
# Week 8: HTTP Simple Table Server, Json Format, Bean Shell Script

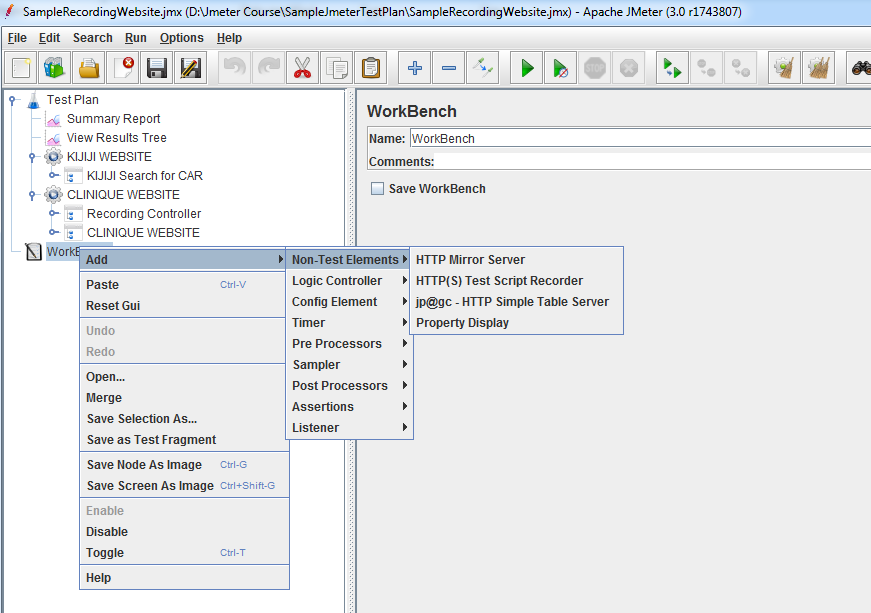
* What Is Http Simple Table Server?

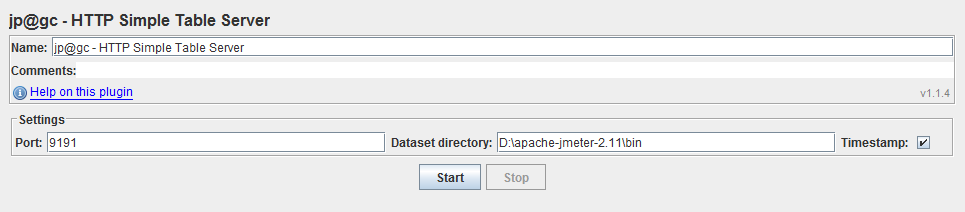
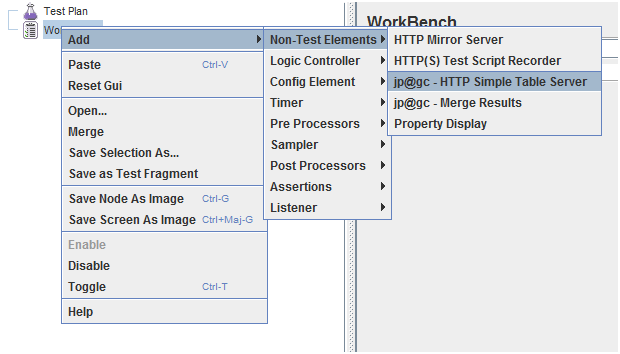


* Performance testing with JMeter can be done with several JMeter injectors (on a remote host) and one JMeter controller (with GUI, on your local host). Scripts are sent to JMeter injectors using RMI protocol. Results are brought back periodically to the JMeter controller. Unfortunately the dataset and csv files aren't transferred from the controller to injectors.
* The main idea is to use a tiny http server in Jmeter Plugins to manage the dataset files with simple commands to get / add rows of data in files. When running Jmeter test plan using multiple remote servers, Jmeter scripts are sent to Jmeter servers using RMI protocol, test results are brought back to Jmeter client. Unfortunately, dataset in csv files aren't transferred from Jmeter client to servers. Therefore, HTTP Simple Table is developed to manage the dataset files with a simple command to get/add rows of data in files.
* Install Http Simple Table Server Plugin:
  + Option 1: Download: <https://jmeter-plugins.org/?search=jpgc-sts>
  + Option 2: Go to Plugin Manager under Options in Jmeter. Select Available Plugin Tab, then select HTTP Simple Table Server. Then click on Apply Changes and Restart Jmeter.







* How To Use Http Simple Table Server To Get Data, Add Data And Save Data
  + Configuration: In jmeter.properties file:
    - jmeterPlugin.sts.port=9191
    - jmeterPlugin.sts.addTimestamp=true
    - jmeterPlugin.sts.datasetDirectory=D:/apache-jmeter-3.0/bin/dataset
    - jmeterPlugin.sts.loadAndRunOnStartup=false
    - sampleresult.default.encoding=UTF-8
  + Start HTTP Simple Table Server from Jmeter GUI:
  + Start HTTP Simple Table Server with simple-table-server.cmd (.sh for UNIX) script located in your JMETER\_HOME/bin directory.
  + Start HTTP Simple Table Server on JMeter start-up with following properties in user.properties file:
    - jmeterPlugin.sts.loadAndRunOnStartup=true
    - jmeterPlugin.sts.port=9191
    - beanshell.init.file=startup-simple-table-server.bsh
  + Start HTTP Simple Table Server with BeanShell script
  + Initial file: Load file in memory. Lines are stored in a linked list, 1 line = 1 element
    - http://hostname:port/sts/INITFILE?FILENAME=logins.csv
  + Read: Get one line from list:
    - http://hostname:port/sts/READ?READ\_MODE={FIRST, LAST, RANDOM}&KEEP={TRUE, FALSE}&FILENAME=logins.csv

Available options:

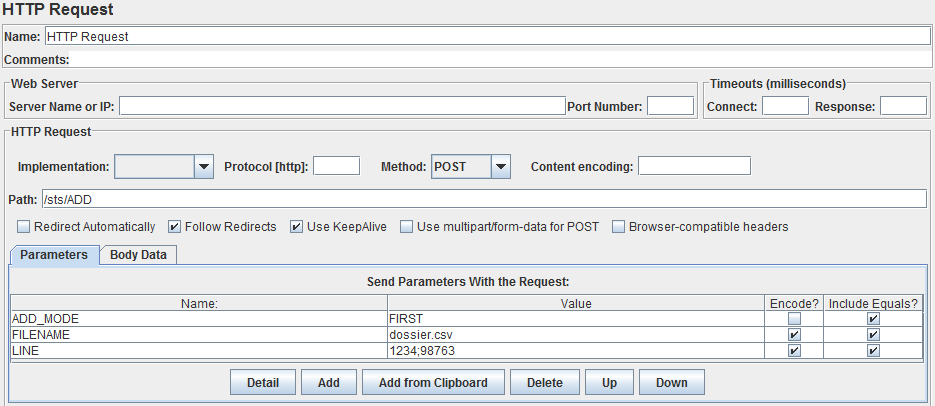
* READ\_MODE=FIRST => login1;password1
* READ\_MODE=LAST => login5;password5
* READ\_MODE=RANDOM => login?;password?
* KEEP=TRUE => the data is kept and put to the end of list
* KEEP=FALSE => the data is removed

KEEP=TRUE, READ\_MODE=FIRST => login1; password1

* + ADD: Add a line into a file: (GET OR POST HTTP protocol)
    - http://hostname:port/sts/ADD?FILENAME=dossier.csv&LINE=D0001123&ADD\_MODE={FIRST, LAST}

Available options:

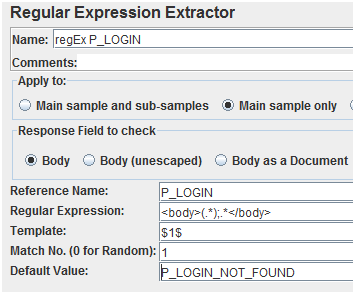
* + - ADD\_MODE=FIRST => add to the top
    - ADD\_MODE=LAST => add to the end
    - FILENAME=dossier.csv => if doesn't already exist it creates a LinkList in memory
    - LINE=1234;98763 => the line to add



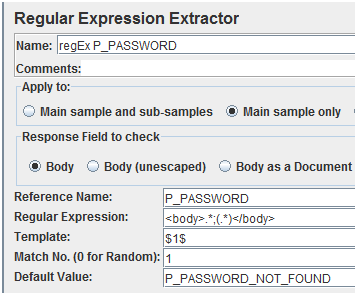
* + LENGTH: Return the number of remaining lines of a linked list
    - http://hostname:port/sts/LENGTH?FILENAME=logins.csv
  + STATUS: Display the list of loaded files and the number of remaining lines for each linked list
    - http://hostname:port/sts/STATUS
  + SAVE: Save the specified linked list in a file to the specified location
    - <http://hostname:port/sts/SAVE?FILENAME=logins.csv>

(If jmeterPlugin.sts.addTimestamp is set to true then a timestamp will be add to the filename, the file is stored in the custom directory specified by editing the jmeterPlugin.sts.datasetDirectory property or in JMETER\_HOME/bin directory by default)

* + RESET: Remove all of the elements from the specified list
    - http://hostname:port/sts/RESET?FILENAME=logins.csv
  + STOP: Shutdown the Simple Table Server
    - http://hostname:port/sts/STOP
  + Using STS in a Test Plan
    - Initialize file using a "setUp Thread Group" by calling URL with one or more HTTP Request Sampler
* Reading a row of data is done by calling READ method at each iteration by a HTTP Request Sampler. Then you can use a Regular Expression Extractor to parse the response data.
* Reading login:



* Reading Password:



* + For more information about HTTP Simple Table Server: https://jmeter-plugins.org/wiki/HttpSimpleTableServer/
* What Is Json Request And Response in REST API?
  + REST API: When Web services use REST architecture, they are called REST full APIs (Application Programming Interfaces) or REST APIs. REST architecture involves reading a designated Web page that contains a Json request and Json response.
  + Json format: JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate. It is based on a subset of the JavaScript Programming Language, Standard ECMA-262 3rd Edition - December 1999.
  + Json format sample:

{"menu":

{

"id": "file",

"value": "File",

"popup":

{

"menuitem": [

{"value": "New", "onclick": "CreateNewDoc()"},

{"value": "Open", "onclick": "OpenDoc()"},

{"value": "Close", "onclick": "CloseDoc()"}

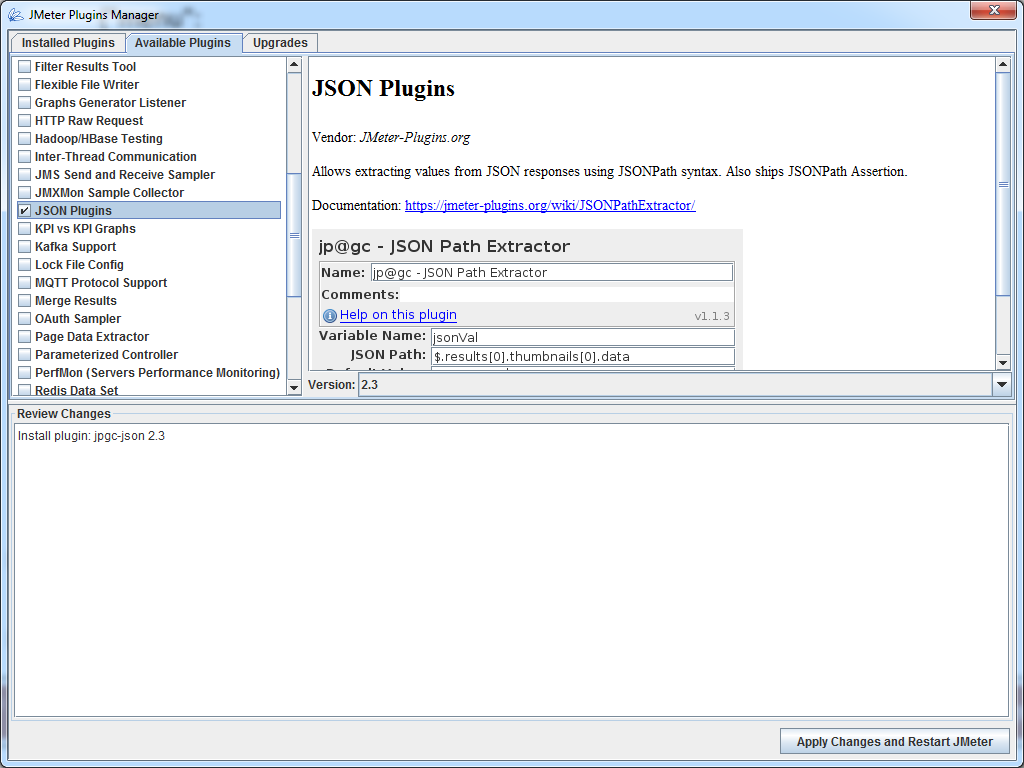
]

}

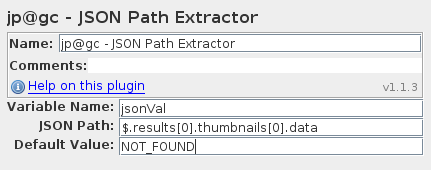
}

}

* Install Json Plugin:



* JSON Path Extractor: Allows extracting values from JSON responses using JSONPath syntax.
* It will try to find the JSON path provided and extract corresponding value. If the path is not found, it will use default value. For the null value the "null" string will be used.
* If result is array, it will additionally set loop variables. This is much like RegExp Extractor or XPath Extractor do to support ForEach Controller input:
  + varName\_1
  + varName\_2
  + varName\_3



* How To Test Json Format
* How To Use Bean Shell Script
* Some Sample Of Bean Shell Script