Prerequisite

- JDK 8

Installation

- Download JMeter and extract the downloaded JMeter zip
- Linux
  - Create `setenv.sh` file in `apache-jmeter-5.4.3\bin`
    - Add the following content in `setenv.sh` file

```
JAVA_HOME=/usr/java/jdk1.8.0_281
```

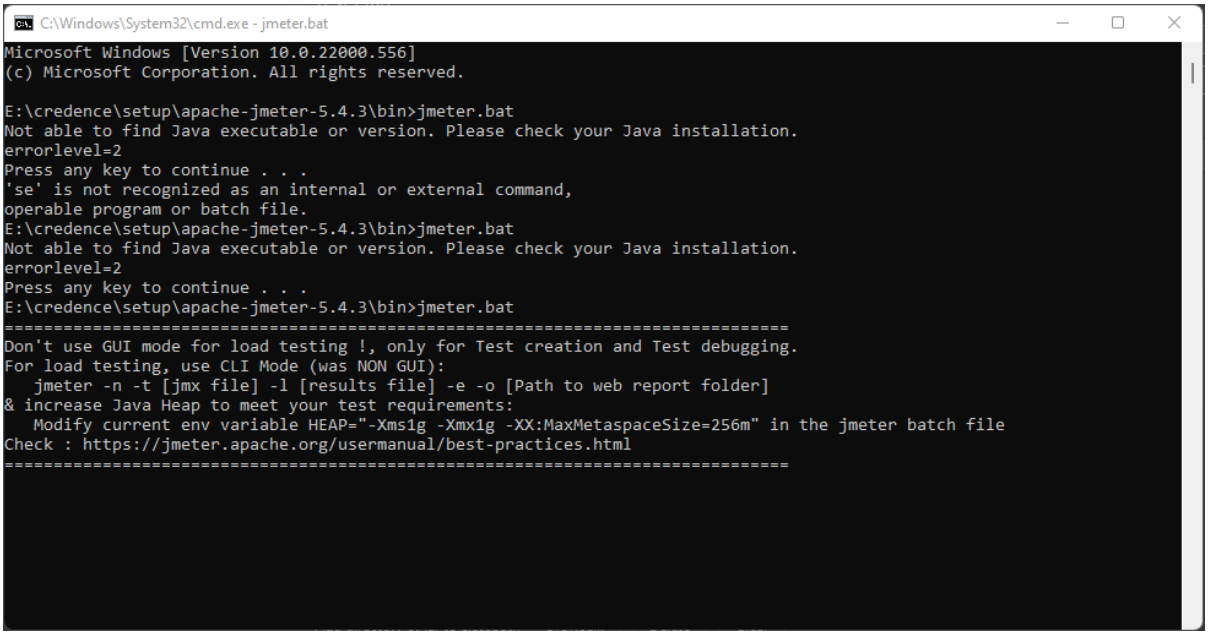
Update the JDK path in JAVA HOME

- Windows
  - Create `setenv.bat` file in `apache-jmeter-5.4.3\bin`
    - Add the following content in `setenv.bat` file

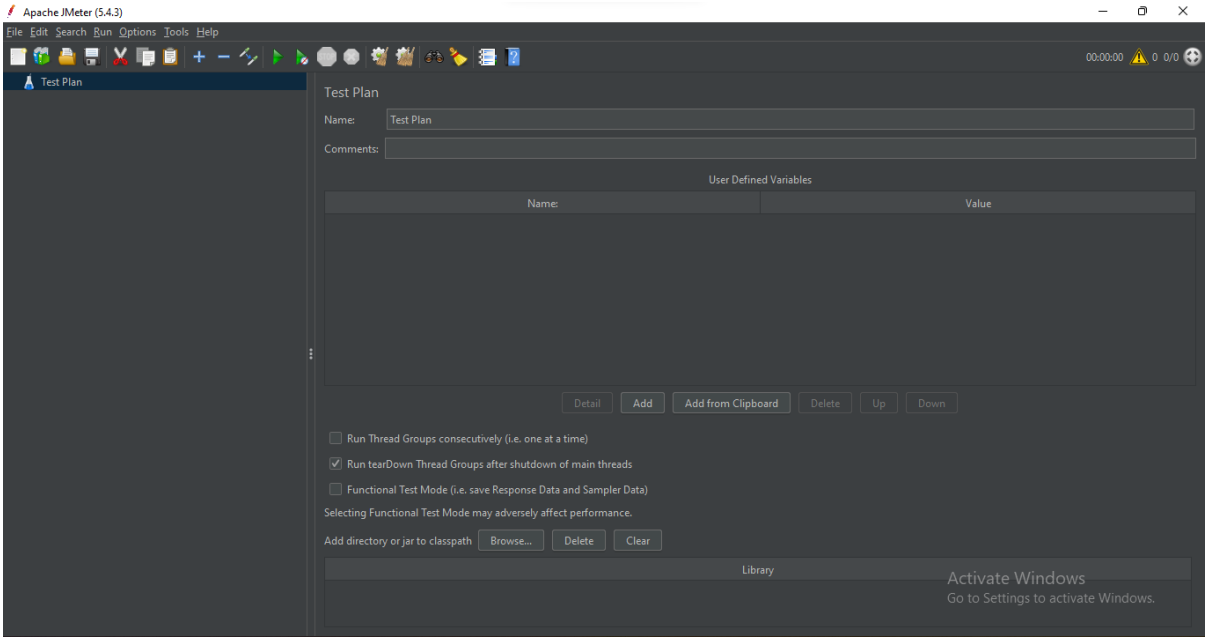
```
JAVA HOME=C:/Program Files/Java/jdk1.8.0_281/bin
```

Update the JDK path in JAVA HOME

  - Open the cmd prompt. Traverse to `apache-jmeter-5.4.3\bin\` and start jmeter.bat

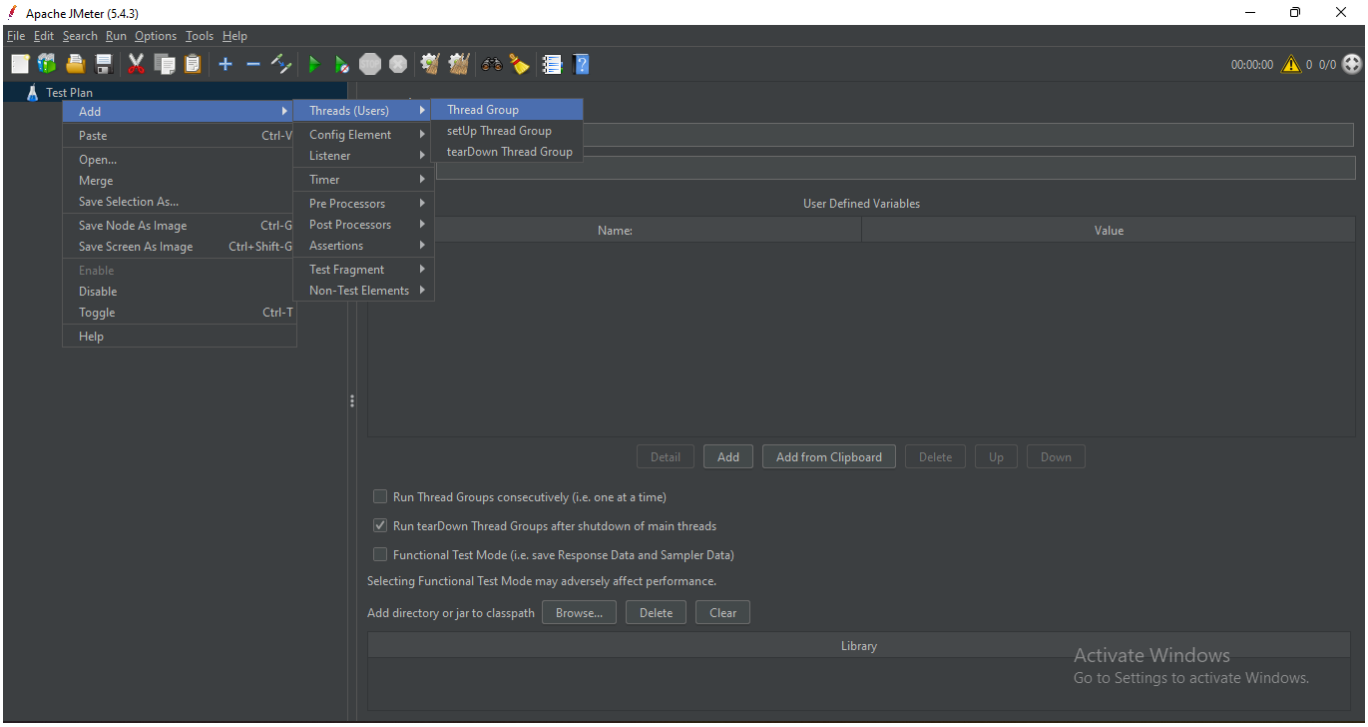


- This will open the JMeter application



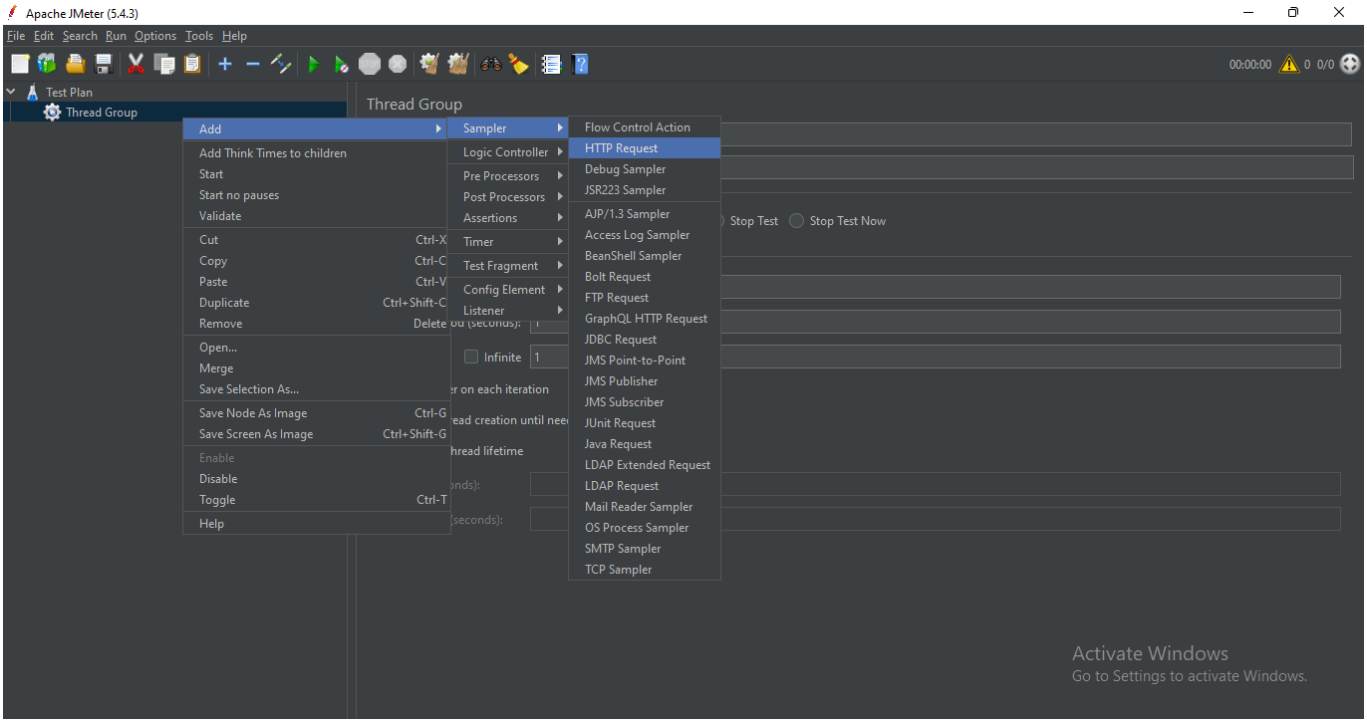
Recording Script

Create `Thread Group`



- Update the parameters
  - `Number of Threads (users)` - Number of users to be tested for
  - `Ramp-up period(seconds)` - Rate at which the users will be onborded to start the transaction. In jMeter, with 10 users if you give a ramp up time of 20, then 1 user will start executing the plan every 2 seconds.
  - `Loop Count`: The number of iterations for each user in the group

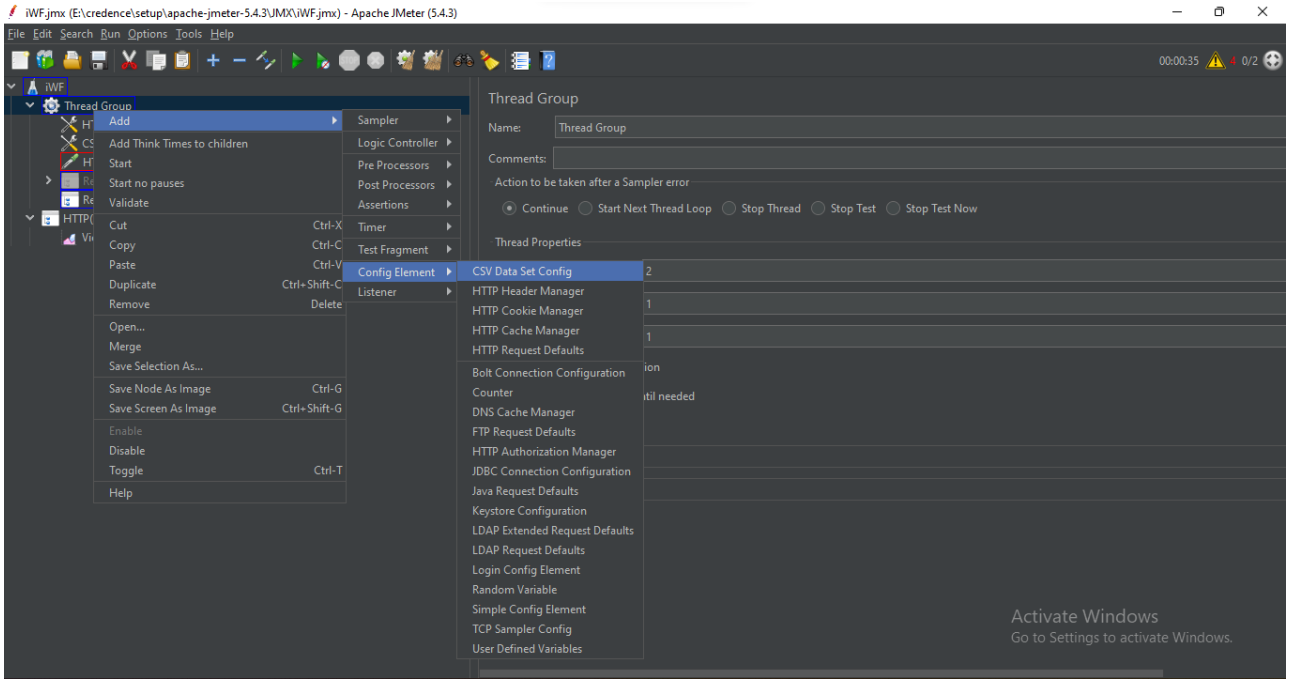
Create HTTP Request



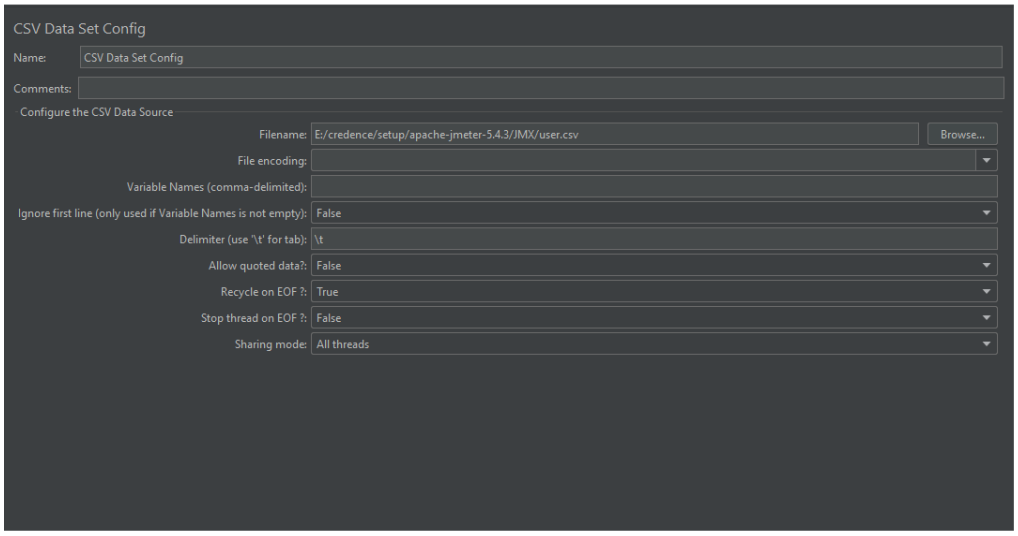
- Update Parameters
  - `Protocol[http]` - Application protocol http/https
  - `Server Name or IP` - Applicaton IP
  - `Port Number` - Application port
  - `Path` - `/`

Create CSV Data Set Config

- This enables using CSV files as an external data source. We will be using to store the list of userid for login into application for performance testing. The userids in csv must be present in the user master table

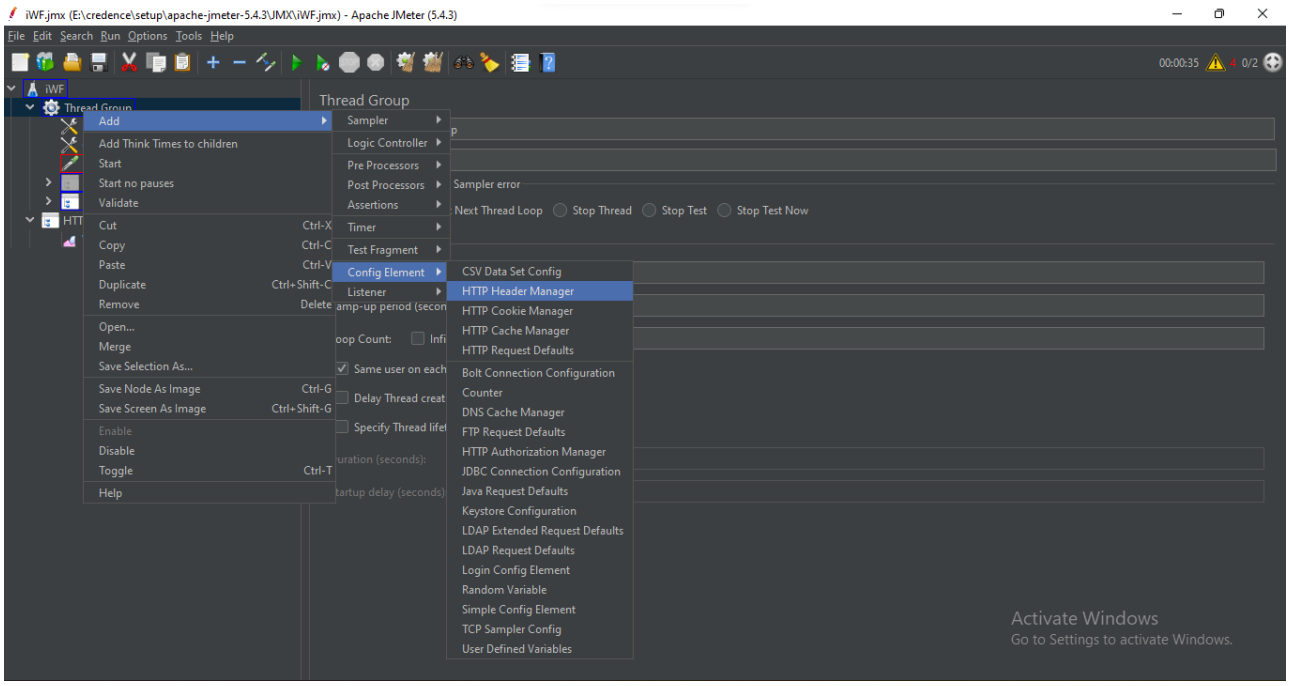


- Create a csv file with header uid. Following the header will be the list of userid from the user master
- Select the csv file and set the delimiter w.r.t to the csv file respectively as shown below



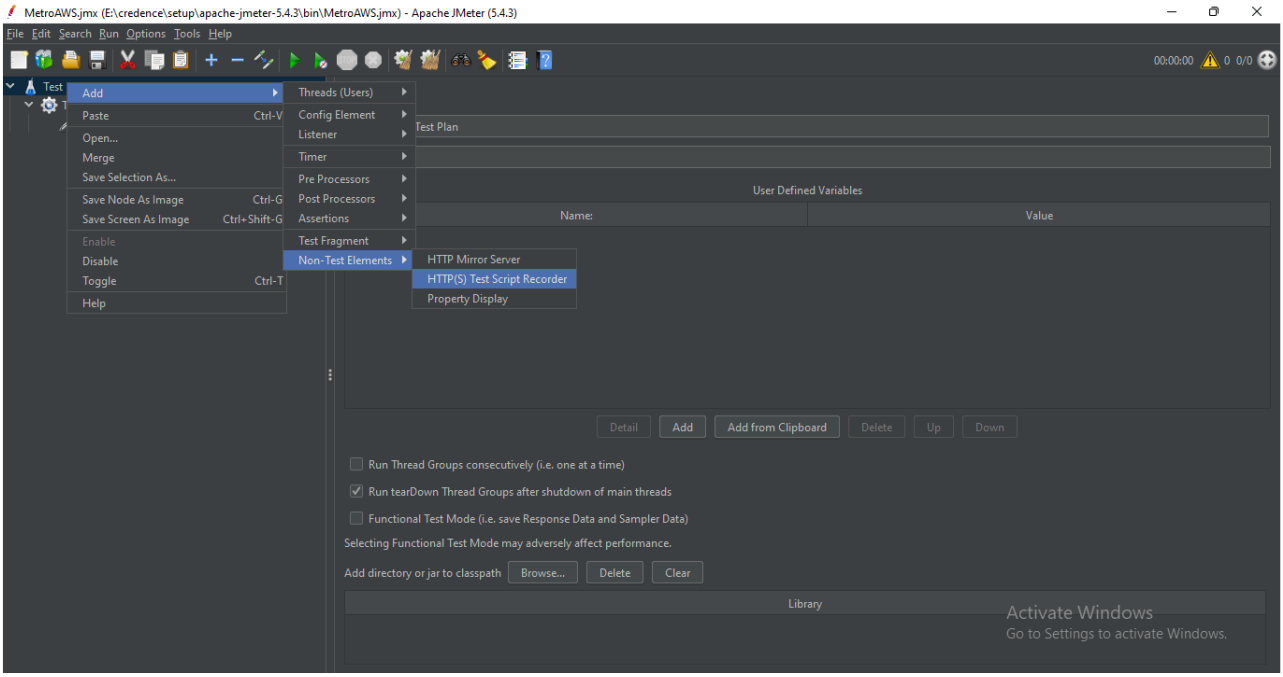
Create Http Header Manager

- To set the sessionid in headers globally. Required for Node Rest Services



Create HTTP Test Script Recorder

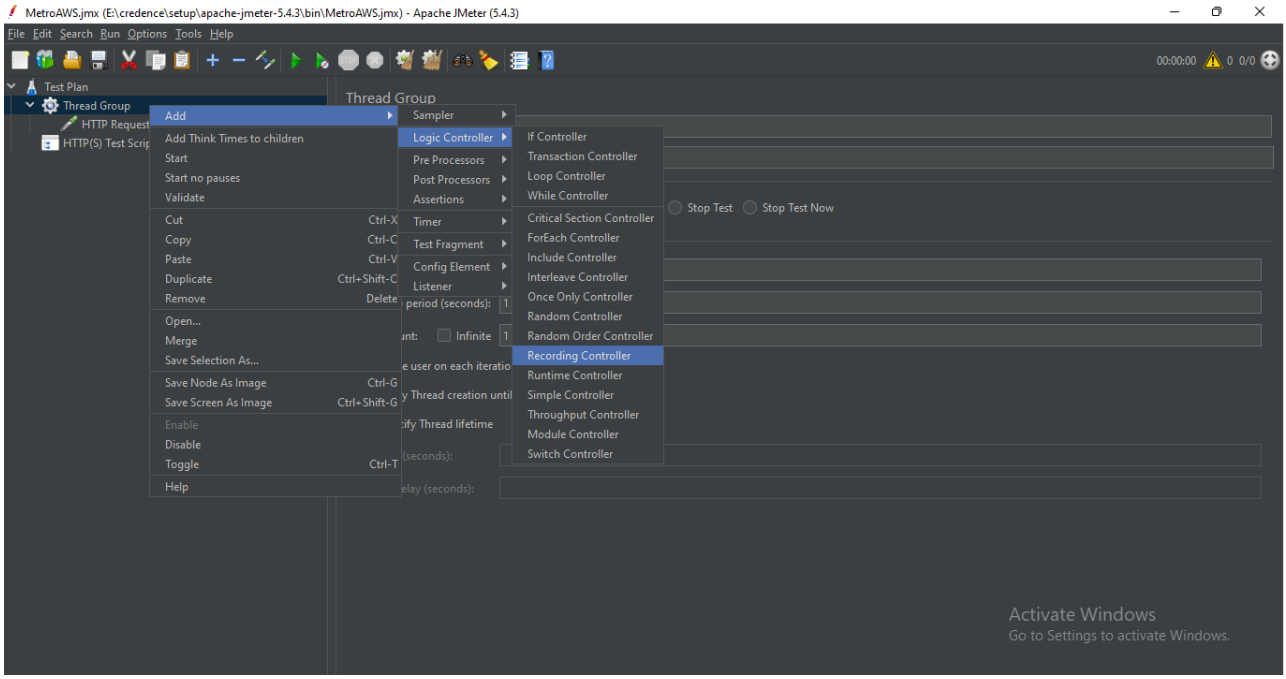
- This helps to record HTTP request. So instead of manually adding our request in an HTTP sampler, we can very quickly do a recording and everything will get added. For recording , jmeter is to be set as proxy so that every request is routed via jmeter and the scripts are recorded. This will explained ahead

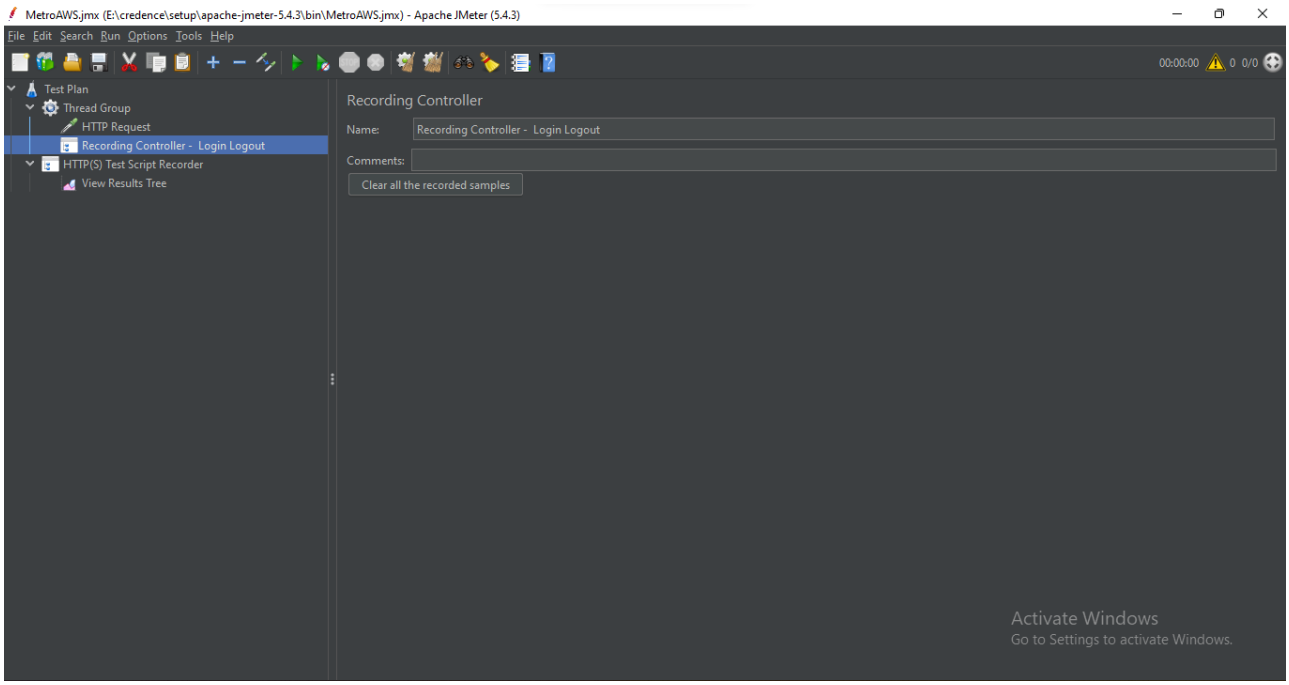


- Following attributes to be changed
    - Port : Any port that is not in use. e.g : 8181
- Run this created `HTTP Test Script Recorder` once to generate `ApacheJMeterTemporaryRootCA.crt` file which will be required later.

Create Recording Controller

- By using the Recording Controller we can easily record the application navigation script. You can create as many recording controller as per events

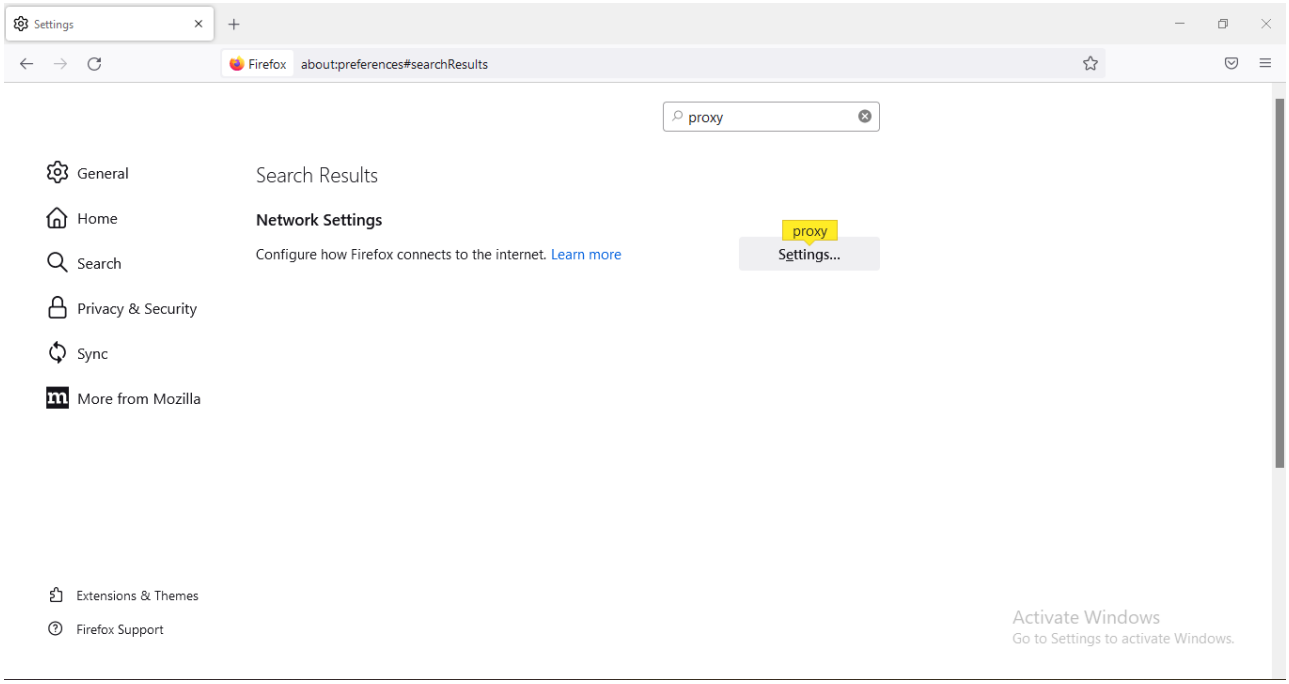


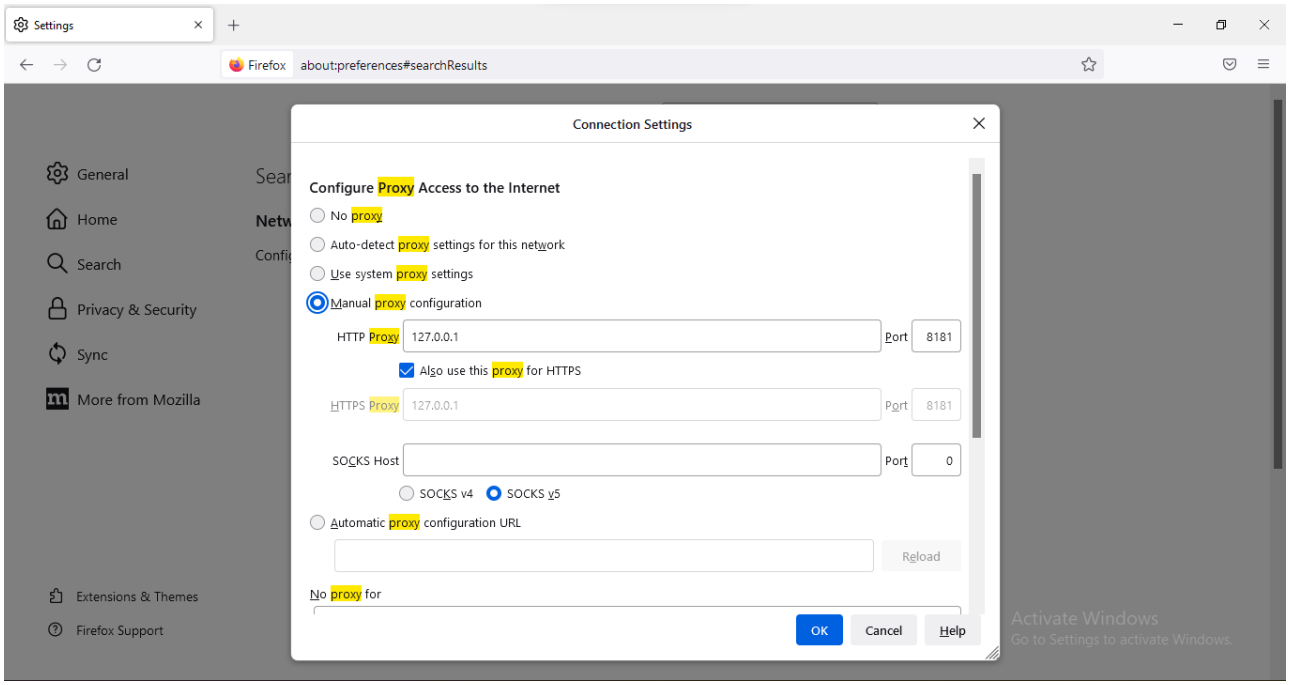


- Add a listener `View Results Tree` to show the results

Setting Up Proxy

- Open Firefox
- Go to Settings and search for proxy

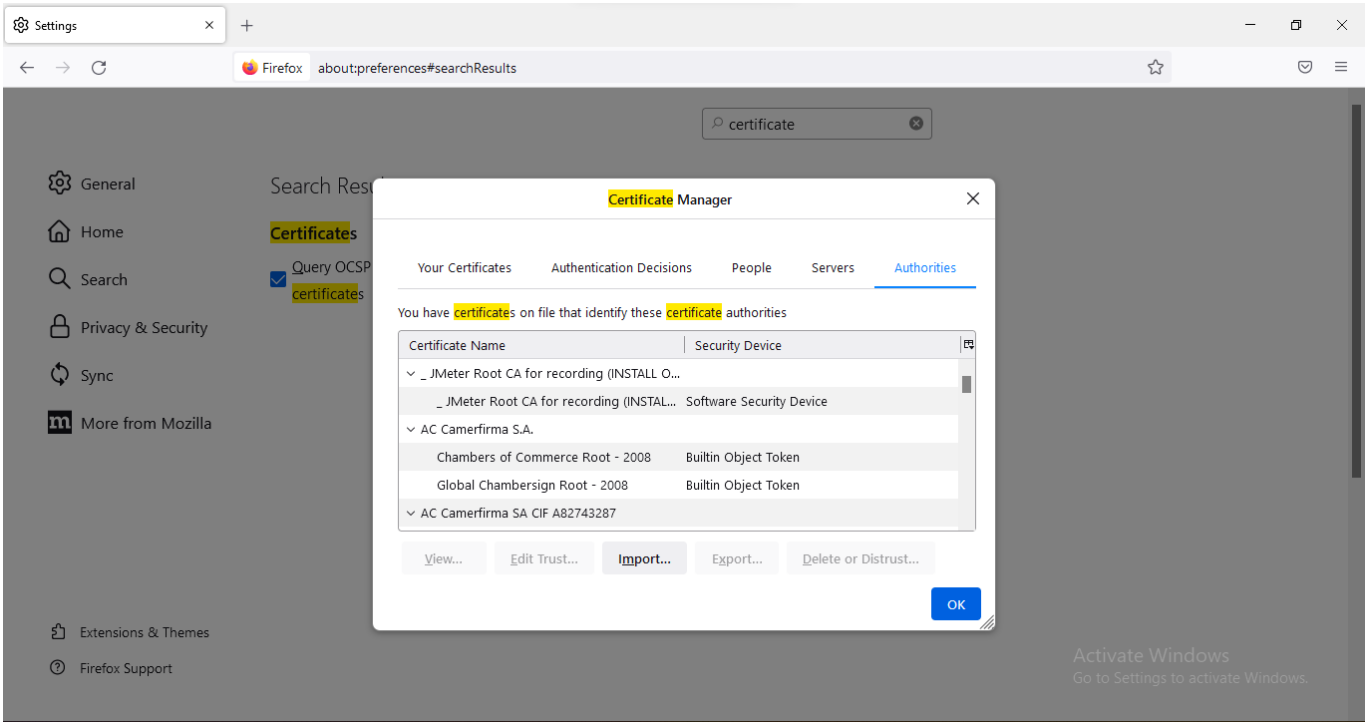




- Select `Manual Proxy Configuration`
- Enter `HTTP Proxy` and `Port`. Port will be set to the one set in Jmeter Test Script Recorder i.e `8181` or any other port that is used
- Click `OK`

JMeter Certificate Import

- The JMeter certificate in folder `apache-jmeter-5.4.3\bin\ApacheJMeterTemporaryRootCA.crt` is to be imported.
- Go to Firefox Settings -> Search Certificate -> View Certificates.. -> Authorities
- Click on import and select the certificate to be imported
- Once the certificate is imported click on `OK`

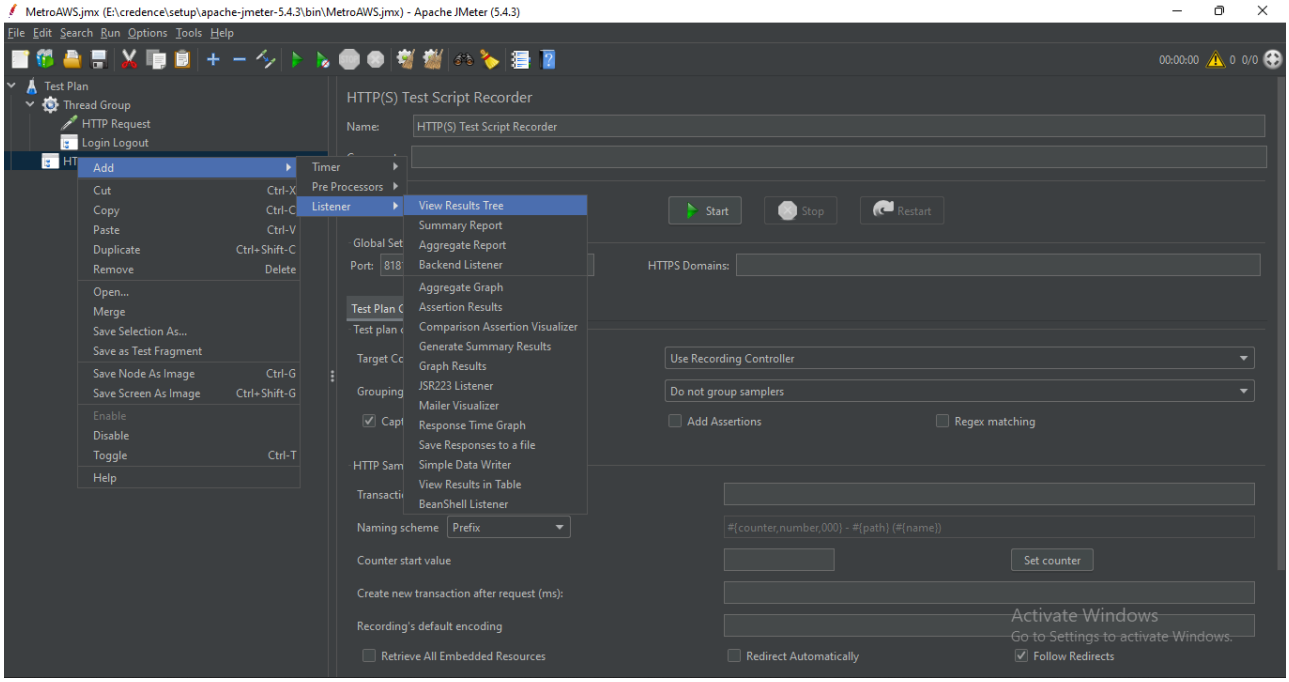


Firefox Config to be disabled

- Follow this step if you face the error `Peer's certificate has an invalid signature.`
- Go to Firefox Browser -> New Tab -> about:config -> Search `network.captive-portal-service.enabled` - > Disable the parameter

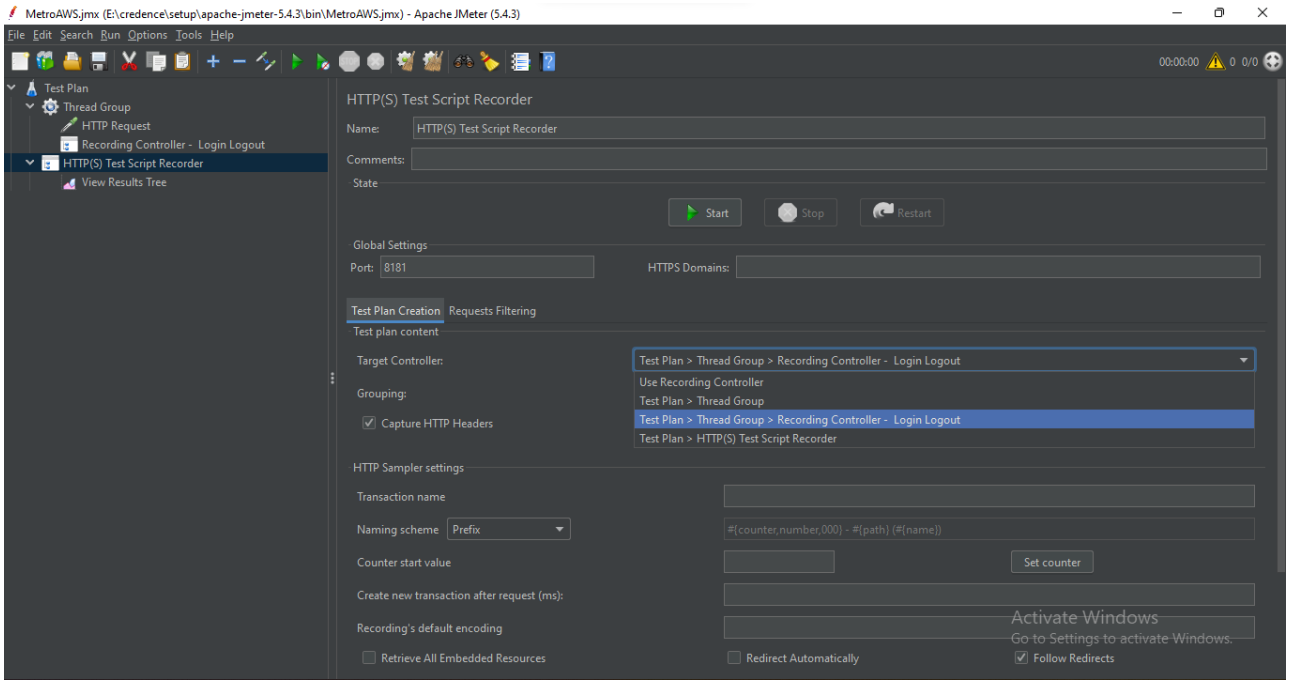
Create Tree Listener

- Shows the recorded results of `HTTP Test Script Recorder` in the form of tree

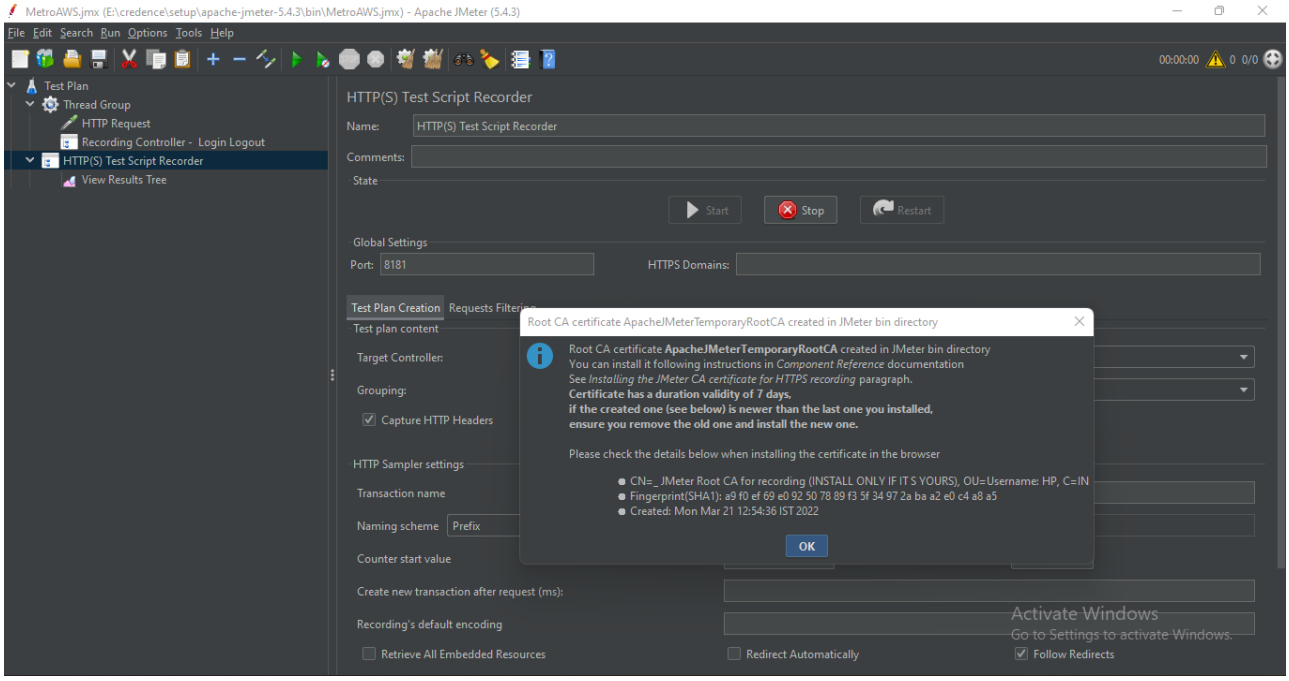


Start Recording

- Go to HTTP Test Script Recorder
- Before proceeding , select the target controller where recording is to be stored. Assuming we are recording LoginLogout,set the Target Controller to `Login Logout`
- Also the application should be running

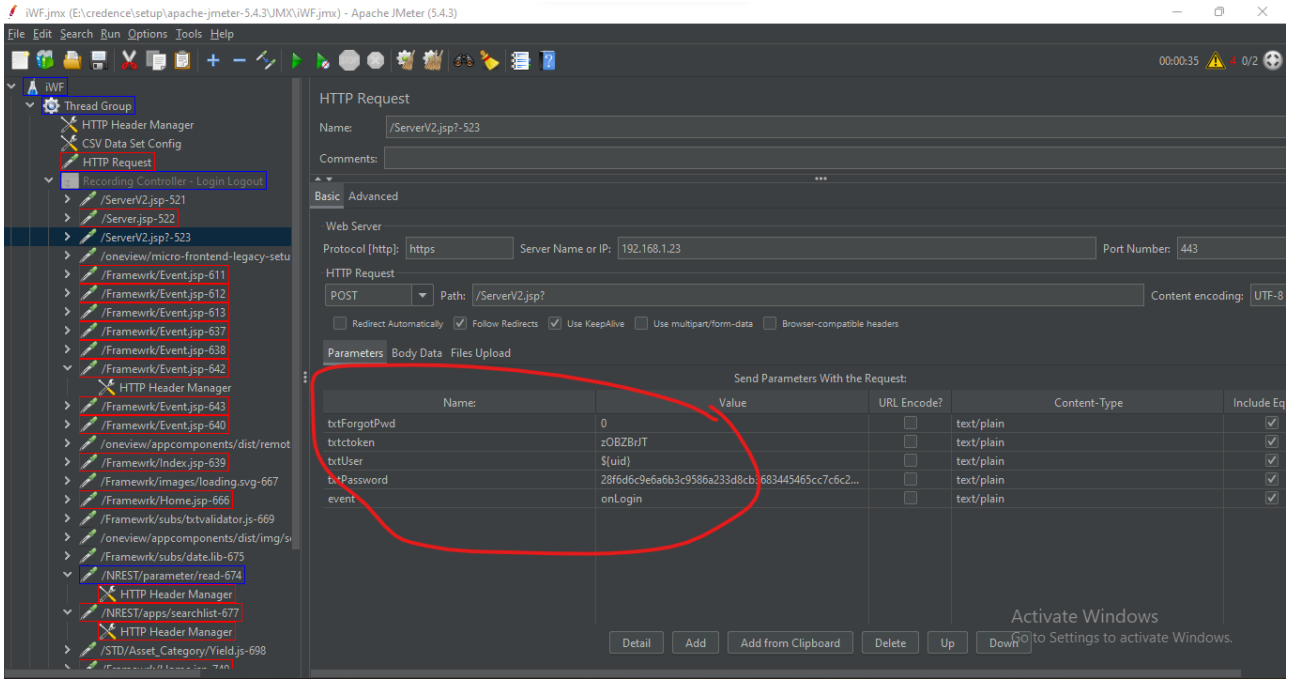


- Start Recording. Certificate prompt will appear and click on `OK`

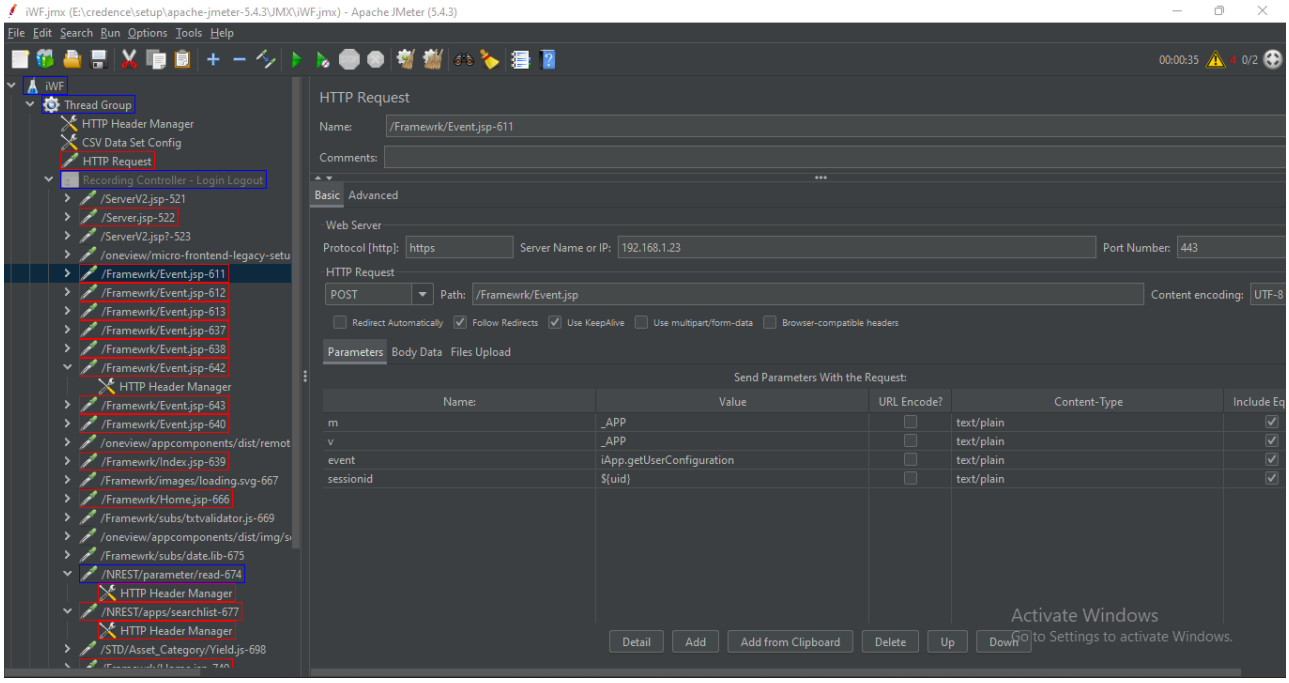


- Go to firefox browser. Record an event. For e.g Login into application and then Logout.
- After recording, check `HTTP Request Defaults -> View Results Tree` for any errors. In case of any error , the event is to be recorded again. Clean the previous recording
- Once the recording is done successfully, revert the proxy settings in Firefox.

- The userid captured in Server2.jsp is to be set to ${uid}



- The sessionid captured in every request of the recording controller, is to be set to ${uid}

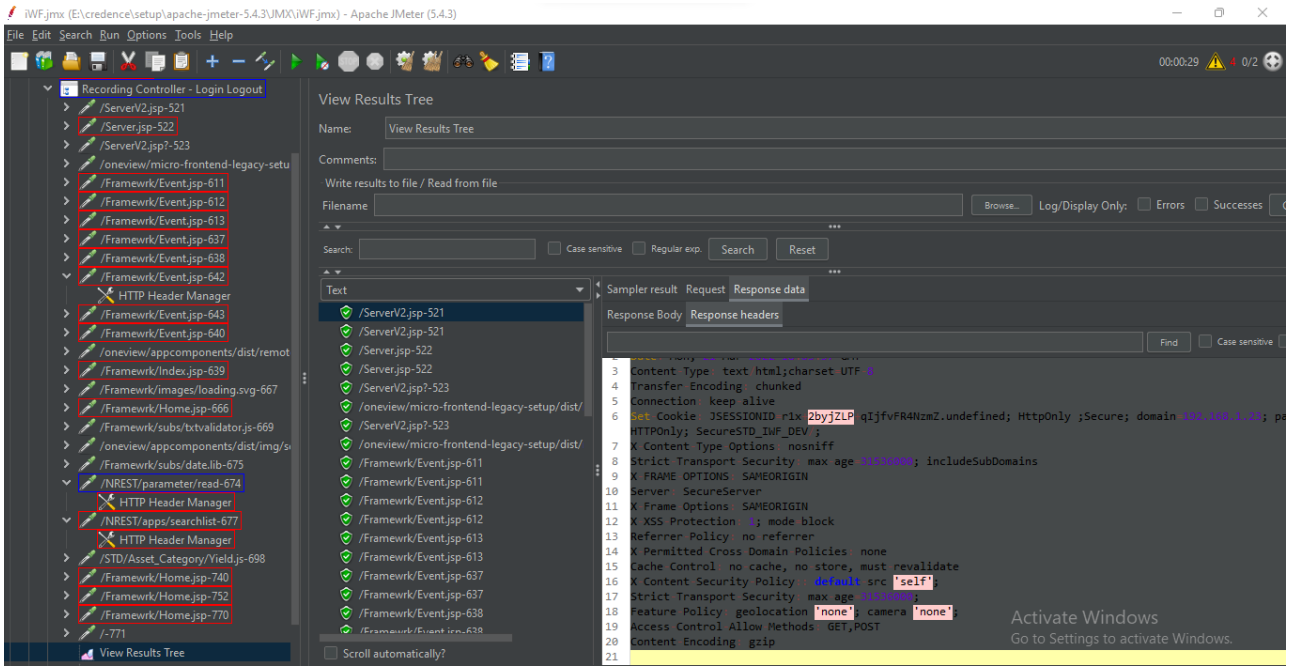


iWF Changes

- Compare and take the changes of main.jds

Run the recorded scripts

- Click on the Start button to run the scripts
- Change the Thread Group parameter as required and run the scripts.
- View the tree results to check if the scripts are executed successfully



Run in Non GUI Mode

- Traverse to `apache-jmeter-5.4.3\bin\` and run the following command to run .jmx file in non-GUI mode.

```
./jmeter -n -t <.jmx file path> -l <path where csv report will be created> -
e -o <path where html report will be created>
```