Iteration 2

PerfMon Records

References:
1. https://jmeter-plugins.org/wiki/PluginsManagerAutomated/
2. https://jmeter-plugins.org/wiki/PerfMon/

Installation of PerfMon

On JMeter Node

First Try (test failed -> not sure if this works):
1. root@ip-172-31-15-56:/home/ubuntu# wget https://jmeter-plugins.org/files/packages/jpgc-perfmon-2.1.zip -O perfmon.zip --2025-03-13 05:31:50-- https://jmeter-plugins.org/files/packages/jpgc-perfmon-2.1.zip Resolving jmeter-plugins.org (jmeter-plugins.org)... 104.251.213.72 Connecting to jmeter-plugins.org (jmeter-plugins.org)|104.251.213.72|:443... connected. HTTP request sent, awaiting response... 200 OK Length: 1071847 (1.0M) [application/zip] Saving to: 'perfmon.zip' perfmon.zip 100%[===>] 1.02M 767KB/s in 1.4s 2025-03-13 05:31:52 (767 KB/s) - 'perfmon.zip' saved [1071847/1071847]

2. root@ip-172-31-15-56:/home/ubuntu# unzip perfmon.zip -d /opt/jmeter Archive: perfmon.zip inflating: /opt/jmeter/lib/perfmon-2.2.2.jar inflating: /opt/jmeter/lib/jmeter-plugins-cmn-jmeter-0.4.jar inflating: /opt/jmeter/lib/ext/jmeter-plugins-perfmon-2.1.jar inflating: /opt/jmeter/lib/ext/jmeter-plugins-manager-0.20.jar

Second Try (Refer to official document):
1. cd /opt/jmeter/lib
2. wget http://search.maven.org/remotecontent?filepath=kg/apc/cmdrunner/2.0/cmdrunner-2.0.jar -O cmdrunner-2.0.jar

3. Since the `/opt/jmeter/bin/` directory does not contain `PluginsManagerCMD.sh` or `PluginsManagerCMD.bat`, execute the following command:
   a. java -cp /opt/jmeter/lib/ext/jmeter-plugins-manager-0.20.jar org.jmeterplugins.repository.PluginManagerCMDInstaller
      - After execution:
        root@ip-172-31-15-56:/opt/jmeter/lib/ext# ls /opt/jmeter/bin/
        ApacheJMeter.jar create-rmi-keystore.bat jmeter jmeter.bat
        mirror-server.cmd stoptest.cmd user.properties
        BeanShellAssertion.bshrc create-rmi-keystore.sh jmeter-n-r.cmd
        jmeter.properties mirror-server.sh stoptest.sh utility.groovy
        BeanShellFunction.bshrc examples jmeter-n.cmd jmeter.sh
        report-template system.properties BeanShellListeners.bshrc
        hc.parameters jmeter-server jmeterw.cmd
        reportgenerator.properties templates BeanShellSampler.bshrc
        heapdump.cmd jmeter-server.bat krb5.conf saveservice.properties
        threaddump.cmd PluginsManagerCMD.bat heapdump.sh
        jmeter-t.cmd log4j2.xml shutdown.cmd threaddump.sh
        PluginsManagerCMD.sh jaas.conf jmeter.backup mirror-server
        shutdown.sh upgrade.properties

4. Install perfmon: /opt/jmeter/bin/PluginsManagerCMD.sh install jpgc-perfmon
   root@ip-172-31-15-56:/opt/jmeter/bin#
   /opt/jmeter/bin/PluginsManagerCMD.sh install jpgc-perfmon 06:52:51.583
   [main] INFO org.jmeterplugins.repository.PluginManager - Plugins Status:
   [jpgc-perfmon=2.1, jpgc-plugins-manager=0.20, jmeter-core=5.6.3,
   jmeter-ftp=5.6.3, jmeter-http=5.6.3, jmeter-jdbc=5.6.3, jmeter-jms=5.6.3,
   jmeter-junit=5.6.3, jmeter-java=5.6.3, jmeter-ldap=5.6.3,
   jmeter-mail=5.6.3, jmeter-mongodb=5.6.3, jmeter-native=5.6.3,
   jmeter-tcp=5.6.3, jmeter-components=5.6.3] 2025-03-13 06:52:51,583
   INFO o.j.r.PluginManager: Plugins Status: [jpgc-perfmon=2.1,
   jpgc-plugins-manager=0.20, jmeter-core=5.6.3, jmeter-ftp=5.6.3,
   jmeter-http=5.6.3, jmeter-jdbc=5.6.3, jmeter-jms=5.6.3, jmeter-junit=5.6.3,
   jmeter-java=5.6.3, jmeter-ldap=5.6.3, jmeter-mail=5.6.3,
   jmeter-mongodb=5.6.3, jmeter-native=5.6.3, jmeter-tcp=5.6.3,
   jmeter-components=5.6.3] 06:52:51.618 [main] INFO
   org.jmeterplugins.repository.PluginManager - Restarting JMeter...
   2025-03-13 06:52:51,618 INFO o.j.r.PluginManager: Restarting JMeter...

5. Check whether it is installed: /opt/jmeter/bin/PluginsManagerCMD.sh status

On Nginx Node

Working directory path: ubuntu@ip-172-31-1-46:~$ (= /home/ubuntu)

1. wget <https://github.com/undera/perfmon-agent/releases/download/2.2.3/ServerAgent-2.2.3.zip>

2. sudo apt install unzip

3. unzip ServerAgent-2.2.3.zip

4. cd ServerAgent-2.2.3/

5. sudo apt update

6. sudo apt install default-jdk -y

7. java --version
   ubuntu@ip-172-31-1-46:~/ServerAgent-2.2.3$ java --version openjdk 21.0.6 2025-01-21 OpenJDK Runtime Environment (build 21.0.6+7-Ubuntu-124.04.1) OpenJDK 64-Bit Server VM (build 21.0.6+7-Ubuntu-124.04.1, mixed mode, sharing)

8. ubuntu@ip-172-31-1-46:~/ServerAgent-2.2.3$ sudo ufw allow 4444/tcp
   Rule added
   Rule added (v6)
   ubuntu@ip-172-31-1-46:~/ServerAgent-2.2.3$ sudo ufw allow 4444/udp
   Rule added
   Rule added (v6)

9. Configure AWS instance inbound rule to allow port 4444

10. ./startAgent.sh
    ubuntu@ip-172-31-1-46:~/ServerAgent-2.2.3$ ./startAgent.sh INFO 2025-03-13 06:08:12.469 [kg.apc.p] (): Binding UDP to 4444 INFO 2025-03-13 06:08:12.572 [kg.apc.p] (): Binding TCP to 4444 INFO 2025-03-13 06:08:12.579 [kg.apc.p] (): JP@GC Agent v2.2.3 started

11. Test connection on JMeter - Success
    Note: The startAgent.sh should always be running during connections

ubuntu@ip-172-31-15-56:~/0312$ nc -zv 172.31.1.46 4444 Connection to 172.31.1.46 4444 port [tcp/*] succeeded!

Test Using PerfMon (Not Working Well)

On the JMeter node, I configured a test plan with PerfMon fields and then ran the test to check whether it works properly and whether the results.jtl file contains metric information.

1. **First try:**
   a. Configuration:

```
<jmeterTestPlan version="1.2" properties="5.0" jmeter="5.5">
  <hashTree>
    <TestPlan testclass="TestPlan" testname="Simple Test Plan"
enabled="true">
    <boolProp name="TestPlan.functional_mode">false</boolProp>
    </TestPlan>
    <hashTree>
    <ThreadGroup testclass="ThreadGroup" testname="Thread Group"
enabled="true">
    <stringProp name="ThreadGroup.num_threads">10</stringProp>
    <stringProp name="ThreadGroup.ramp_time">1</stringProp>
    </ThreadGroup>
    <hashTree>
    <HTTPSamplerProxy testclass="HTTPSamplerProxy"
testname="HTTP Request" enabled="true">
    <stringProp name="HTTPSampler.domain">localhost</stringProp>
    <stringProp name="HTTPSampler.port">8080</stringProp>
    <stringProp name="HTTPSampler.path">/test</stringProp>
    <stringProp name="HTTPSampler.method">GET</stringProp>
    </HTTPSamplerProxy>
    <hashTree/>
    </hashTree>
    <hashTree>
    <!-- PerfMon Metrics Collector -->
    <PerfMonMetricsCollector testclass="PerfMonMetricsCollector"
testname="PerfMon" enabled="true">
    <stringProp name="server">127.0.0.1</stringProp>
    <stringProp name="port">4444</stringProp>
    <stringProp name="metrics">cpu, memory, disk</stringProp>
```

                </hashTree>
                </hashTree>
        </hashTree>
     </jmeterTestPlan>

b. Execute the test plan, but got error:

> An error occurred: Error in NonGUIDriver Problem loading XML from:'/home/ubuntu/0312/test.jmx'. Cause: CannotResolveClassException: PerfMonMetricsCollector Detail:com.thoughtworks.xstream.converters.ConversionException: ---- Debugging information ---- cause-exception : com.thoughtworks.xstream.converters.ConversionException cause-message : first-jmeter-class : org.apache.jmeter.save.converters.HashTreeConverter.unmarshal(HashTreeConverter.java:66) class : org.apache.jmeter.save.ScriptWrapper required-type : org.apache.jmeter.save.ScriptWrapper converter-type : org.apache.jmeter.save.ScriptWrapperConverter path : /jmeterTestPlan/hashTree/hashTree/hashTree/PerfMonMetricsColle ctor line number : 38 version : 5.6.3

c. Cause Analysis: *PerfMonMetricsCollector* is probably an invalid field

2. Second try:

a. Cgonfiuration (.jmx file):

```
<?xml version="1.0" encoding="UTF-8"?> <jmeterTestPlan version="1.2"
properties="5.0" jmeter="5.5"> <hashTree> <TestPlan guiclass="TestPlanGui"
testclass="TestPlan" testname="Simple Test Plan" enabled="true"> <stringProp
name="TestPlan.comments"></stringProp> <boolProp
name="TestPlan.functional_mode">false</boolProp> <boolProp
name="TestPlan.serialize_threadgroups">false</boolProp> <elementProp
name="TestPlan.user_defined_variables" elementType="Arguments"
guiclass="ArgumentsPanel" testclass="Arguments" testname="User Defined Variables"
enabled="true"/> <stringProp name="TestPlan.user_define_classpath"></stringProp>
</TestPlan> <hashTree> <ThreadGroup guiclass="ThreadGroupGui"
testclass="ThreadGroup" testname="Thread Group" enabled="true"> <stringProp
name="ThreadGroup.on_sample_error">continue</stringProp> <elementProp
name="ThreadGroup.main_controller" elementType="LoopController"
```

```
guiclass="LoopControlPanel" testclass="LoopController" testname="Loop Controller"
enabled="true"> <boolProp name="LoopController.continue_forever">false</boolProp>
<stringProp name="LoopController.loops">1</stringProp> </elementProp> <stringProp
name="ThreadGroup.num_threads">100</stringProp> <stringProp
name="ThreadGroup.ramp_time">1</stringProp> <longProp
name="ThreadGroup.start_time">1659101628000</longProp> <longProp
name="ThreadGroup.end_time">1659101628000</longProp> <boolProp
name="ThreadGroup.scheduler">false</boolProp> <stringProp
name="ThreadGroup.duration"></stringProp> <stringProp
name="ThreadGroup.delay"></stringProp> </ThreadGroup> <hashTree> <!-- HTTP
Request Sampler --> <HTTPSamplerProxy guiclass="HttpTestSampleGui"
testclass="HTTPSamplerProxy" testname="HTTP Request" enabled="true">
<stringProp name="HTTPSampler.domain">172.31.1.46</stringProp> <stringProp
name="HTTPSampler.port">80</stringProp> <stringProp
name="HTTPSampler.protocol">http</stringProp> <stringProp
name="HTTPSampler.path">/</stringProp> <stringProp
name="HTTPSampler.method">GET</stringProp> </HTTPSamplerProxy> <hashTree/>
<!-- PerfMon Metrics Collector --> <PerfMonMetricsCollector
guiclass="PerfMonMetricsCollectorGui" testclass="PerfMonMetricsCollector"
testname="PerfMon Metrics" enabled="true"> <stringProp
name="server.host">172.31.1.46</stringProp> <!-- Target server IP --> <stringProp
name="server.port">4444</stringProp> <!-- PerfMon server port --> <stringProp
name="metrics">cpu,memory,load</stringProp> <!-- Metrics to monitor --> <stringProp
name="interval">10</stringProp> <!-- Interval in seconds for data collection -->
<stringProp name="loglevel">info</stringProp> <!-- Log level -->
</PerfMonMetricsCollector> <hashTree/> </hashTree> </hashTree> </hashTree>
</jmeterTestPlan>
```

    b. Execute the test plan: There is no error, but the results.jtl file does not contain CPU or memory-related information as expected

    c. Analysis: The cause is uncertain, but it could be due to a configuration mistake. We need more time to learn about PerfMon. Since it requires further investigation and the testing time is limited, we have decided to use *top/htop* to observe the results at this stage as well.