



**CS 4001/7001 Cloud Computing
Spring 2015**

Lab # 2: Instrumentation and Measurement of a GENI Slice

Dr. Prasad Calyam (Contact: calyamp@missouri.edu)

1. Purpose of the lab

Install and configure a distributed measurement Web App viz., “OnTimeMeasure” within a slice. Configuration involves ‘Node Beacons’ at measurement points, and a ‘Root Beacon’ that acts as a central intelligence module. Schedule measurements from the Root Beacon and query/ visualize performance between two compute nodes that host Node Beacons. Performance data collection comprises of network health metrics such as round trip delay, loss, jitter and throughput.

2. References to guide lab work

- [1] Chapter 3 Distributed and Cloud Computing, Hwang, Fox & Dongarra
- [2] OnTimeMeasure Tutorial: <http://groups.geni.net/geni/wiki/OnTimeMeasure>
- [3] OnTimeControl: <http://groups.geni.net/geni/wiki/OTM-Control>
- [4] OnTimeMeasure Custom Metric Integration: <http://groups.geni.net/geni/wiki/OTM-CustomMetric>
- [5] OnTimeMeasure demo video on Graphite: <http://ontime.rnet.missouri.edu/demo/dashboard.htm>
- [6] Miscellaneous OnTimeMeasure demo videos: <http://ontime.rnet.missouri.edu/demo>

3. Lab Steps and output collection guidelines

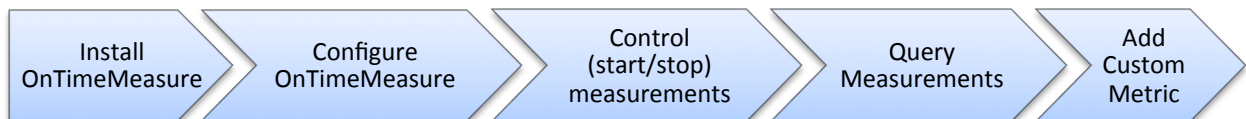


Figure 1: Lab Steps Overview

Figure 1 shows the required steps to successfully complete the lab with OnTimeMeasure in GENI. You will need to install the OnTimeMeasure Node and Root Beacon software, and configure them with ‘network path measurement tasks’ between two compute nodes in your GENI slice. You will control and query active measurements as part of your experiment and monitor the network path status using common network health metrics such as round trip delay, loss, jitter and throughput. You will also experiment with adding a custom metric i.e., CPU usage and schedule ‘host-based measurement tasks’ with OnTimeMeasure and understand programmability and extensibility concepts in cloud monitoring.

The OnTimeMeasure-GENI architecture is shown in Figure 2, for details refer to [2].

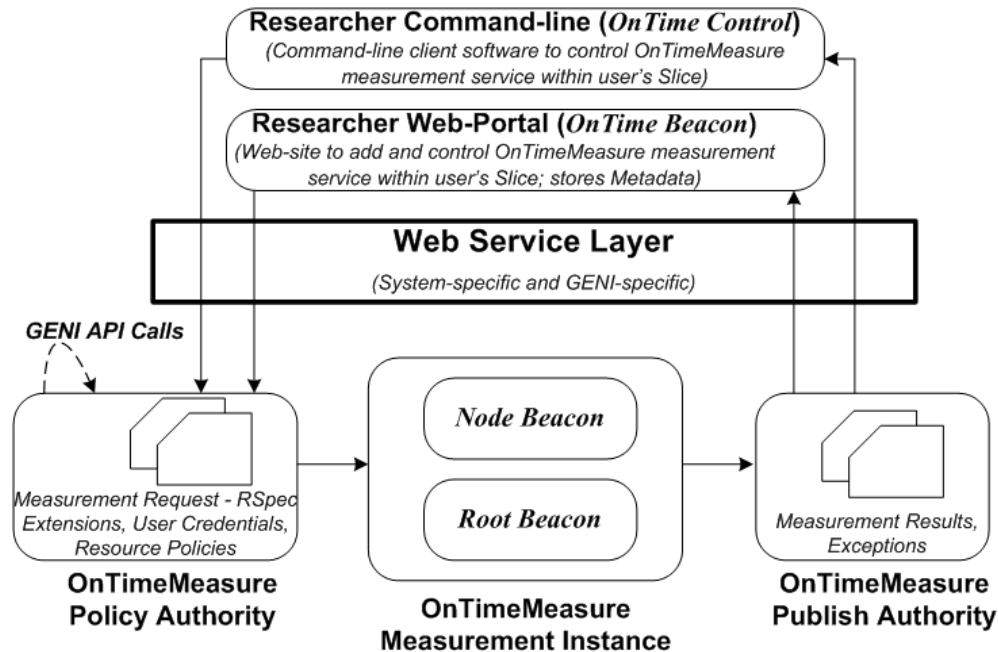
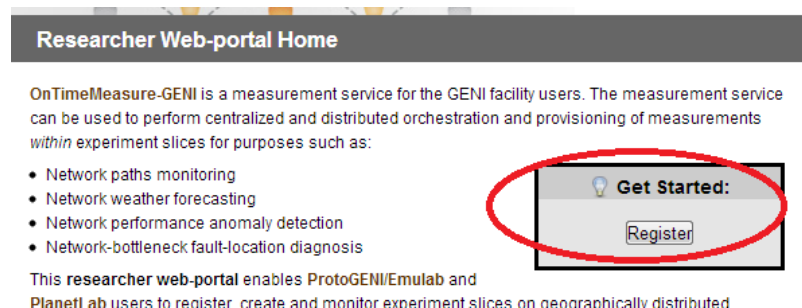


Figure 2: OnTimeMeasure-GENI Architecture

Pre-work for OnTimeMeasure

Register and create an account on the OnTimeMeasure portal at: <http://ontime.rnet.missouri.edu>
(OnTimeBeacon instance installed at University of Missouri-Columbia)



1. The 'Register' button will take you to the User Registration page. Provide your user information and give the project name as 'Calyam_UMissouri_Fall2013' and click 'Register'.
 2. Once the Instructor approves your registration, you can login to the portal to perform OnTimeMeasure related experiments in GENI.
- Note: You will start using the OnTimeMeasure portal functions from **Step 3.2.2** of this manual.

3.1 Install OnTimeMeasure Software Packages

3.1.1 Create Slice

Create a slice Lab2<Pawprint> similar to **Step 2** of Lab1.



3.1.2 Specify Topology

Download RSpec file from Blackboard (Assignments → GENI Lab-2 RSpec) and make sure it is saved as 'Lab2.rspec'. Upload this RSpec file in Flack using 'Import from file' to specify the slice topology and use the same aggregate manager (a.k.a. Flack nickname) assigned to you previously.

Note: The previous allocation of aggregate managers has changed. Following are the new allocations. Reserve your resources according to the new allocations.

Group Name	Aggregate manager	AM Nickname
A,B,C	instageni.nysernet.org	nysernet-ig
D,E,F,G	instageni.stanford.edu	stanford-ig
H,I,J	instageni.idre.ucla.edu	ucla-ig

This RSpec will reserve 3 compute nodes in your slice with 2 Node Beacons and 1 Root Beacon.

Modify the NODE1 information by clicking on the "i" button" on each of these nodes.

First, select '+ Add Install Service', and Flack will prompt you for an 'Archive URL' and a location for installation. For this experiment, enter the below URL:

```
http://babbage.cs.missouri.edu/~ar442/cc/OTCNodeInstall.tar.gz
```

and ask for this to be installed in:

```
/tmp
```

Similarly, use the '+ Add Execute Service' button to request that the appropriate command:

```
sudo sh /tmp/ontime_software_install.sh
```

be executed using

```
sh
```

Once both the "install" and "execute" services are specified, choose 'Apply'. Then repeat the procedure to add the same "install" and "execute" services on the 'NODE2' node;

NODE1

Manager: instageni.gpolab.bbn.com... Binding: Unbound Sliver Type: emulab-openvz

Disk Image: Select image to override default

Version: If required, give a disk image version

Install: Execute:

Install: http://ontime.oar.net/INSTALL/OTCNodeInstall.tgz in /tmp

Execute: sudo sh /tmp/ontime_software_install.sh using sh



Similarly, modify the 'ROOT' node information by providing the below URL for '+ Add Install Service'

```
http://babbage.cs.missouri.edu/~ar442/cc/OTCRootInstall.tar.gz
```

and ask for this to be installed in:

```
/tmp
```

Similarly, use the '+ Add Execute Service' button to request that the appropriate command:

```
sudo sh /tmp/ontime_software_install.sh otm_slice_password
```

be executed using

```
sh
```

Once both the "install" and "execute" services are specified, choose 'Apply'.

ROOT

Manager: instageni.gpolab.bbn.com...

Binding: Unbound

Silver Type: emulab-openvz

Disk Image: Select image to override default

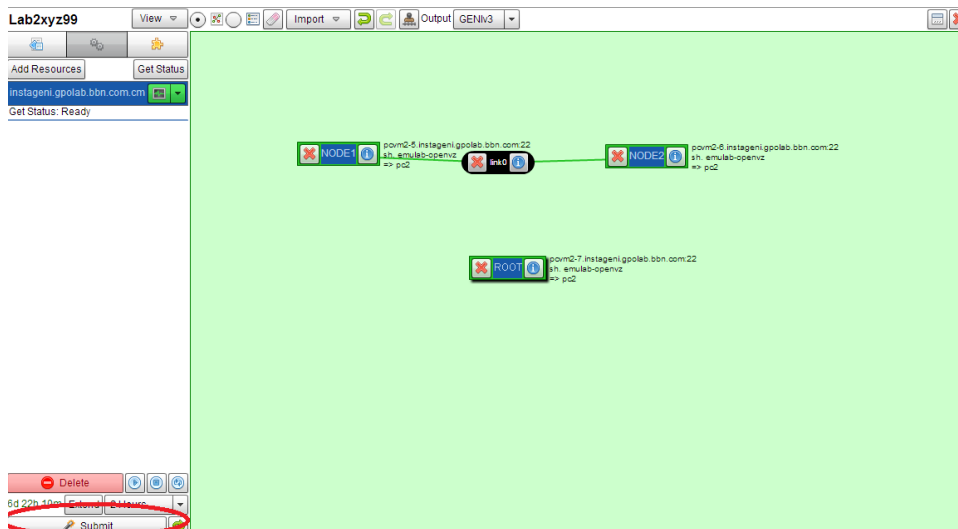
Version: If required, give a disk image version.

Install: http://ontime.oar.net/INSTALL/OTCRootInstall.tar.gz

Execute: sudo sh /tmp/ontime_software_install.sh otm_slice_password using sh

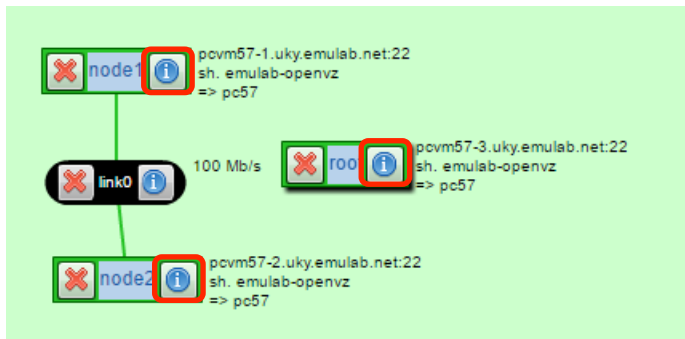
3.1.3 Create and Access your Sliver

Select the 'Submit' button in the left hand corner to create your sliver. **Wait for the canvas to turn green which confirms the reservation of resources.** Use "Get Status" to refresh asynchronously to see if the reservation is complete.





Once the nodes are reserved successfully, in the GENI portal the canvas turns green and each of the nodes have an address associated with them. Click on “i” symbol beside each of the nodes



Copy the address of each of the nodes as highlighted and save them where they are accessible.

Please note that the login information can be different for every student (and not necessarily the same as shown in the screenshot)

3.1.4 OnTimeMeasure Installation Confirmation

Confirm the installation of Root Beacon:

Note: Installation of the Root Beacon and Node Beacon can take more than 5 minutes.

For example, from the login information, if your ROOT login is xyz99@pcvm2-7.instageni.gpolab.bbn.com, open a browser and type, <http://pcvm2-7.instageni.gpolab.bbn.com:17291>). You should see your supervisor running as below -

State	Description	Name	Action
running	pid 1507, uptime 0:31:34	ActionScript	Restart Stop Clear Log Tail -f
running	pid 1506, uptime 0:31:34	Collector	Restart Stop Clear Log Tail -f
running	pid 1505, uptime 0:31:34	File_server	Restart Stop Clear Log Tail -f
running	pid 1503, uptime 0:31:34	Graphite	Restart Stop Clear Log Tail -f
running	pid 1504, uptime 0:31:34	RootService	Restart Stop Clear Log Tail -f



Similarly, confirm the installation of Node Beacons in the web-browser URL of each Node Beacon as shown below:

As an example, <http://pcvm2-5.instageni.gpolab.bbn.com:17291> and <http://pcvm2-6.instageni.gpolab.bbn.com:17291>

The login information can be different for every student. Please change the URL accordingly and point the port to 17291.

The screenshot shows a web browser window with the title 'Supervisor Status'. The address bar shows the URL 'pcvm2-6.instageni.gpolab.bbn.com:17291'. The page has a yellow header with the 'Supervisor status' logo. Below the header are three buttons: 'REFRESH', 'RESTART ALL', and 'STOP ALL'. A table lists the status of three services: 'File_server', 'Node Scheduler', and 'Node Service', all of which are 'running'. Each row in the table includes links for 'Restart', 'Stop', 'Clear Log', and 'Tail -f'.

State	Description	Name	Action
running	pid 928, uptime 0:36:30	File_server	Restart Stop Clear Log Tail -f
running	pid 929, uptime 0:36:30	Node Scheduler	Restart Stop Clear Log Tail -f
running	pid 927, uptime 0:36:30	Node Service	Restart Stop Clear Log Tail -f

3.2 Configure OnTimeMeasure

3.2.1 Identify the IP addresses of all 3 nodes

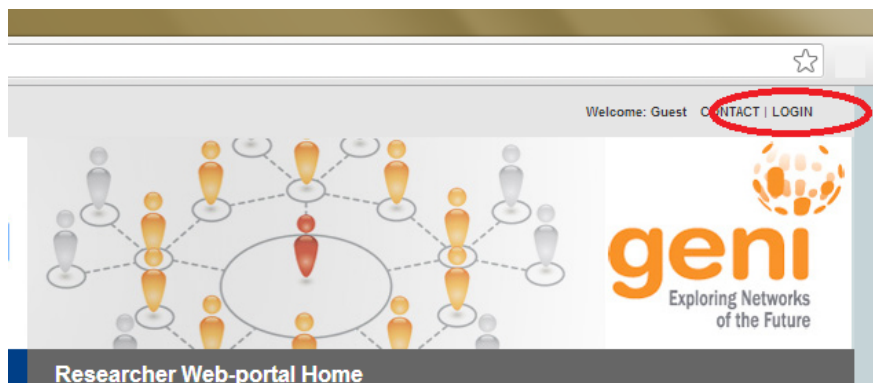
Open a terminal, Ping all the three nodes to see if they are accessible, and note down the IP addresses in the Ping tool output.

If your login for the Root and Node Beacons are pcvm2-7.instageni.gpolab.bbn.com, pcvm2-5.instageni.gpolab.bbn.com and pcvm2-6.instageni.gpolab.bbn.com, type for e.g.:

```
$: ping pcvm2-7.instageni.gpolab.bbn.com
```

3.2.2 Resource Setup

Login to the portal at: <http://ontime.rnet.missouri.edu>



Once you login, click on the 'Setup resources' in your home page



NAVIGATION

- HOME
- MEASUREMENT REQUEST
 - RESOURCE SETUP
 - REQUEST SPECIFICATION
 - REQUEST SUBMISSION
- SERVICE CONTROL
- MEASUREMENT QUERY

Researcher Web-portal Home

OnTimeMeasure-GENI is a measurement service for the GENI facility users. The measurement service can be used to perform centralized and distributed orchestration and provisioning of measurements *within* experiment slices for purposes such as:

- Network paths monitoring
- Network weather forecasting
- Network performance anomaly detection
- Network-bottleneck fault-location diagnosis

This **researcher web-portal** enables **ProtoGENI/Emulab** and **PlanetLab** users to register, create and monitor experiment slices on geographically distributed resources in their facility.

[Measurement Requests](#)

Get Started:

[Setup resources](#)

This will take you to another page, 'Measurement Request Creation' page, please read the steps involved for measurement request creation, and then select the 'Setup Resources' button.

Measurement Request Creation

To submit a measurement request to the OnTimeMeasure-GENI measurement service, please follow a 3-step process:

STEP 1

Resource Setup: You first need to setup a ProtoGENI experiment slice along with Node and Root Beacons

STEP 2

Request Specification: You next need to specify measurement tasks to be executed between Node Beacons

STEP 3

Request Submission: You lastly need to review and submit your measurement request to initialize the measurement service

Proceed to Step 1: [Setup Resources](#)

The 'Measurement Request Setup' page comes up. Retain the desired topology as 'Centralized'. Enter the Root and Node Beacon information and IP addresses noted down in the previous step. Also enter the database username and password as collector and otm_slice_password.



1 2 3 Measurement Request Setup

Please select the desired measurement topology architecture:

☒ Centralized ☐ Distributed

Please input the nodes information:

Slice name:

Root Beacon:

Machine name:

IP address:

Database user name:

Database password:

Node Beacon: delete ↓

Machine name:

IP address:

Node Beacon: delete ↓

Machine name:

IP address:

Proceed to Step-2: [Specify Request](#)

Once the information is entered, select the 'Specify Request' button. You should observe the status changing to 'Measurement Request Specification' as shown below:

1 2 3 Measurement Request Setup

Please select the desired measurement topology architecture:

☒ Centralized ☐ Distributed

Please input the nodes information:

Slice name:

Root Beacon:

Machine name:

IP address:

Database user name:

Database password:

Node Beacon: delete ↓

Machine name:

IP address:

Node Beacon: delete ↓

Machine name:

IP address:

Status: Your resource has been updated.
Going to Measurement Request Specification in 1 second. [Cancel](#)



3.3: Control (start/stop) measurement

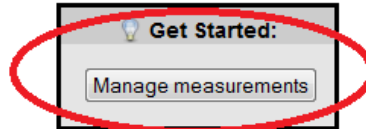
You should be transferred to the 'Measurement Request Specification' page. In case you do not see the metric options in the drop down menu of Metric field, logout of the portal and login again and click the 'Manage Measurements' button in the home page.

Researcher Web-portal Home

OnTimeMeasure-GENI is a measurement service for the GENI facility users. The measurement service can be used to perform centralized and distributed orchestration and provisioning of measurements *within* experiment slices for purposes such as:

- Network paths monitoring
- Network weather forecasting
- Network performance anomaly detection
- Network-bottleneck fault-location diagnosis

This **researcher web-portal** enables **ProtoGENI/Emulab** and **PlanetLab** users to register, create and monitor experiment slices on geographically distributed resources in their facility.



Add the following tasks for network path measurement metrics: Round-trip Delay, Jitter, Loss and Throughput from Node2 to Node1 by selecting the metrics from the drop down menu of Metric field and click 'AddTask' button. Also retain the pattern as periodic. Do the same again with source as Node1 and Destination as Node 2.

Also, do 'Add Task' for the Memory Free metric on both Node1 and Node2 as shown below.

1 2 3 Measurement Request Specification

Specify a measurement task using the following options:

Path based measurement:

Source: node1(155.98.38.99) Metric: Loss

Destination: node2(155.98.38.97) Pattern: Periodic

Add Task Clear All

Host based measurement:

Host: node2(155.98.38.97) Metric: Memory Free

Pattern: Periodic

Add Task Clear All

The tasks added to the measurement service are as follows:

Source	Destination	Metric	Pattern	Action
node2(155.98.38.97)	node1(155.98.38.99)	Loss	Periodic	Remove
node2(155.98.38.97)	node1(155.98.38.99)	Jitter	Periodic	Remove
node2(155.98.38.97)	node1(155.98.38.99)	Round-trip Delay	Periodic	Remove
node2(155.98.38.97)	node1(155.98.38.99)	Throughput	Periodic	Remove
node1(155.98.38.99)	node2(155.98.38.97)	Throughput	Periodic	Remove
node1(155.98.38.99)	node2(155.98.38.97)	Round-trip Delay	Periodic	Remove
node1(155.98.38.99)	node2(155.98.38.97)	Jitter	Periodic	Remove
node1(155.98.38.99)	node2(155.98.38.97)	Loss	Periodic	Remove

Host	Metric	Pattern	Action
node1(155.98.38.99)	Memory Free	Periodic	Remove
node2(155.98.38.97)	Memory Free	Periodic	Remove

Proceed to Step-3: **Review Request**



Once the above steps are complete, select the Step3 'Review Request' button.

This will take you to the Step 3 'Measurement Request Submission' page. Review the tasks information and select the 'Submit Request' button:

1

2

3

Measurement Request Submission

Please review and submit your measurement request to the OnTimeMeasure-GENI measure service:

Resource Setup
Status: Complete

Modify

Measurement topology architecture selected is: Centralized

Measurement topology is as follows:

Slice name: Lab2xyz

Root Beacon: root(155.98.38.94)

Node Beacon 1: node2(155.98.38.97)

Node Beacon 2: node1(155.98.38.99)

Request Specification
Status: Complete

Modify

The tasks added to the measurement service are as follows:

Source	Destination	Metric	Pattern
node2(155.98.38.97)	node1(155.98.38.99)	Loss	Periodic
node2(155.98.38.97)	node1(155.98.38.99)	Jitter	Periodic
node2(155.98.38.97)	node1(155.98.38.99)	Round-trip Delay	Periodic
node2(155.98.38.97)	node1(155.98.38.99)	Throughput	Periodic
node1(155.98.38.99)	node2(155.98.38.97)	Throughput	Periodic
node1(155.98.38.99)	node2(155.98.38.97)	Round-trip Delay	Periodic
node1(155.98.38.99)	node2(155.98.38.97)	Jitter	Periodic
node1(155.98.38.99)	node2(155.98.38.97)	Loss	Periodic

Host	Metric	Pattern
node1(155.98.38.99)	Memory Free	Periodic
node2(155.98.38.97)	Memory Free	Periodic

Request Submission
Submit the request to initialize the measurement service.

Submit Request

Once, you select the 'Submit Request', the 'Service Control' page comes up, initiate the measurement collection by selecting the 'Start' button. The status changes to 'Running'.



Service Control

Start

Initiates communications between Root Beacons and/or Node Beacons to start the active measurements data collection

Stop

Terminates communications between Root Beacons and/or Node Beacons to stop the active measurements data collection

Status: ▶ Running: Measurements are being collected in the experiment slice.

Update

Refreshes the service status notification; can be used to verify whether or not any of the service components are functioning as expected

The status of the service components are as follows:

Component	Status
Slice Accessibility	✔ OK
Root Beacon Scheduler	✔ OK
Node and Root Beacon Communications & Data Collector	✔ OK
Publish Authority	✔ OK
Measurement Data Visualization	✔ OK

➡ Proceed to query measurements data collected:

Query Data

3.4: Query Measurements

Select the 'Query Data' button to visualize the results. This will take you to the 'Measurement Query' page. Allow some time for the results to be generated (**at least 30 minutes**). Scroll down to the end of page and select the 'View Dashboard' button. You can also download the measurement data by selecting the 'Submit Query' button and clicking 'View Result' as shown below:



Measurement Query

Please select from the following query options:

User: xyz99

Metric: Memory Free

Start time(UTC): 2013-09-06 16:39:41

Host: node2(192.1.242.194)

End time(UTC): 2013-09-13 17:39:41

Results type:

☐ Raw Files

☐ Time Series with Anomalies

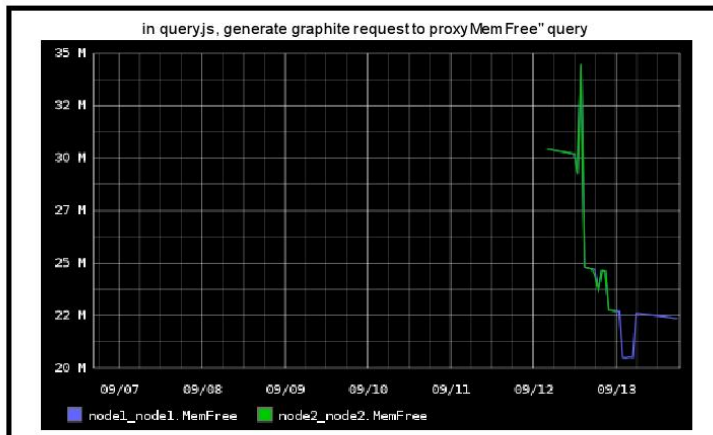
☒ Time Series

☐ Time Series with Forecasts

Submit Query

Status: Measurement query was successful. [View Result](#)


Measurement graph:



View real-time graphs of measurement data: [View Dashboard](#)

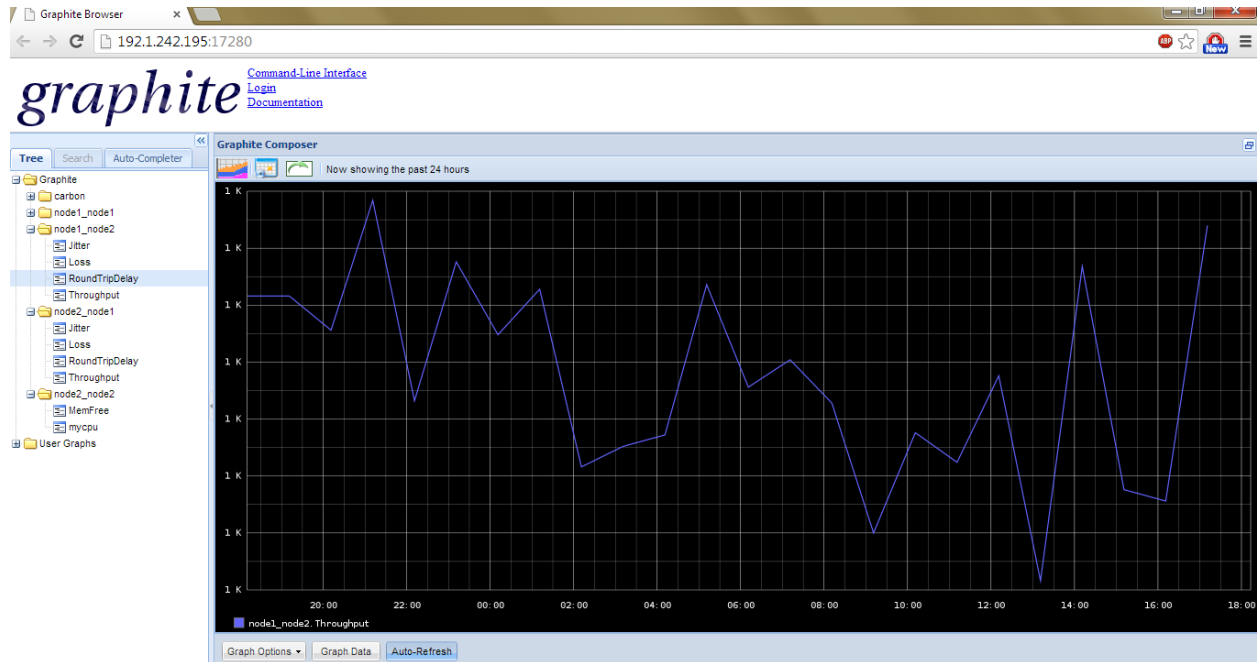
Download entire slice measurement archive: [Query Archive](#)

Proceed to control measurement service status: [Service Control](#)

The 'View Dashboard' button will take you to a new page called the Graphite Browser. You can visualize all the measurement metrics data by navigating the left hand folder structure. At any moment, in order to visualize a new metric, de-select the current metric and highlight the desired metric to view the results at any given time. You can also change the measurement data period by clicking third icon  in the top left corner of the graph.

For more details, see the OnTimeMeasure demo video on Graphite:

<http://ontime.rnet.missouri.edu/demo/dashboard.htm>



Note: Measurement graphs in Graphite can vary for every student.

3.5 Add Custom Metric

3.5.1 Configure IP address and database password

Login to your ROOT node using the ssh login information in the 'Details' page of GENI portal as shown previously.

```
$: ssh xyz99@pcvm2-35.instageni.gpolab.bbn.com
```

Enter OnTimeControl folder by entering the following command -

```
$: cd /opt/OnTimeMeasure/OnTimeControl
```

Copy the given config_example.yaml under the directory to config.yaml

```
$: sudo cp config_example.yaml config.yaml
```

Open the config.yaml file and edit the IP addresses of 'root_beacon', 'node1' and 'node2', change the 'root_hostname' to localhost and also change the 'db_pwd' to otm_slice_password and change <http://ontime.oar.net/xmlrpc.php> to <http://ontime.rnet.missouri.edu/xmlrpc.php>. Your config.yaml should look similar to below.

Note: Please make sure you do not give extra spaces or unnecessary quotes anywhere in the file.

```
$: sudo vi config.yaml
```



```
#####  
connection: direct  
  
#####  
# Setup for direct connection  
#####  
root_beacon: 192.1.242.205  
root_hostname: localhost  
db_user: collector  
db_pwd: otm_slice_password  
  
nodes:  
- name: node1  
  IP: 192.1.242.206  
  
- name: node2  
  IP: 192.1.242.207  
  
#####  
# Setup for portal connection  
#####  
portal: http://ontime.rnet.missouri.edu/xmlrpc.php  
web_user: demo  
web_pwd: password
```



3.5.2 Install custom metric CPU on to the OnTimeMeasure Framework

Run the following commands to download a custom CPU metric which gives CPU utilization of the nodes.

```
$: sudo wget http://ontime.rnet.missouri.edu/INSTALL/metric/CPU.tgz  
$: sudo tar -xzf CPU.tgz
```

You will get custom metric specification file CPUSpecs.yaml and parser file CPUParser.pm

Add custom metric specifications file and parser file into your OnTimeMeasure framework instance.

```
$: python add_metric.py -c CPUSpecs.yaml -p CPUParser.pm
```

Edit the measurement configuration file (measurement.yaml) to add the new metric on both nodes

```
$: sudo vi measurement.yaml
```

Configure the two nodes CPU metric at the end of the file as shown below:

```
- source: node1  
  metric: [MemFree, Mycpu]  
- source: node2  
  metric: [MemFree, Mycpu]
```

3.5.3 Restart the measurement service

Run command "task_manage.py" to manage the measurement tasks with custom metric (i.e., requests), which are configured in the measurement.yaml file and restart the measurements

```
$: python task_manage.py -c measurement.yaml  
$: python service_control.py restart
```

Allow sometime for results to be generated (**at least 30 minutes**).

You can go back to your graphite page to visualize the new metric or download the results from the OnTimeBeacon portal page as shown previously. You can also download the results from command line. For example to download the data for the Throughput metric for a certain start and end time, run:

```
python query.py -f "2013-08-24 18:00:00" -t "2012-08-27 20:00:00" -s  
node2 -d node1 -m Throughput -r TimeSeries
```

You can change the -m, -f, -t and -r parameter accordingly.



4. What to turn in for Grading?

1. Provide screenshots of your OnTimeMeasure instance' Graphite page for 'RoundTripDelay' of node2 -> node1, node1-> node2 , 'MemFree' and MyCPU of node1 and node2.
2. Explain the role and functions of "Instrumentation and Measurement Tools" such as OnTimeMeasure (http://groups.geni.net/geni/attachment/wiki/OnTimeMeasure/OnTimeMeasure_Tutorial.pdf?format=raw), and GEMINI Tool Set (<http://groups.geni.net/geni/attachment/wiki/GEMINI/gemini-gec13.pptx?format=raw>) in GENI infrastructure.
3. Briefly explain in your own words the architecture of the 'OnTimeControl' framework (see Reference [3]).
4. Describe the workflow that was involved when you added the custom metric feature to your OnTimeMeasure framework instance in the last step of your GENI experiment.

NOTE:

Release your resources after finishing your lab report by following below steps

The command to use is:

```
omni.py -a AM_NICKNAME deletesliver SLICENAME
```

where AM_NICKNAME is the aggregate manager nickname and SLICENAME is the name of your slice. A minute or so later, Omni should respond with:

```
INFO:omni: Completed deletesliver:
```

Or you can delete the resources by using flack by clicking delete button located bottom left of the canvas.

