The School of Computing Cloud Testbed

1. Introduction

The Cloud Computing Testbed is aimed at supporting both teaching and applications research within the School of Computing.

The testbed has currently 14 (new) physical machines. The Virtual Infrastructure Management is supported by OpenNebula [1]. OpenNebula controls / monitors physical machines and executes Virtual Machines (VM) via Secure Shell.

The images necessary to start VMs are shared via a repository on the Network File System.

Examples of existing testbeds include Emulab [2] and PlanetLab [3].

Current research efforts on the cloud testbed, both internal to the University and external, include the following research areas: distributed systems, virtual machines, contextualization, data-mining, monitoring, scheduling, and operating systems.

2. Access

You can access the testbed through the gateway **csgate1.leeds.ac.uk** using your School of Computing/University username and password:

ssh username@csgate1.leeds.ac.uk

Note that the machines on the testbed are labelled cscloud1nX.cloud.comp.leeds.ac.uk, X = 1..14

You need to **email support** so that you are granted access to the gateway.

3. Cloud Operations Center

The OpenNebula cloud operation centre is available on:

https://csgate1.leeds.ac.uk:8443/

You need to login using your University username and password.

OpenNebula Sunstone is the OpenNebula Cloud Operations Center, a Graphical User Interface (GUI) intended for regular users and administrators that simplifies the typical management operations in private and hybrid cloud infrastructures. OpenNebula Sunstone allows to easily manage all OpenNebula resources and perform typical operations on them.

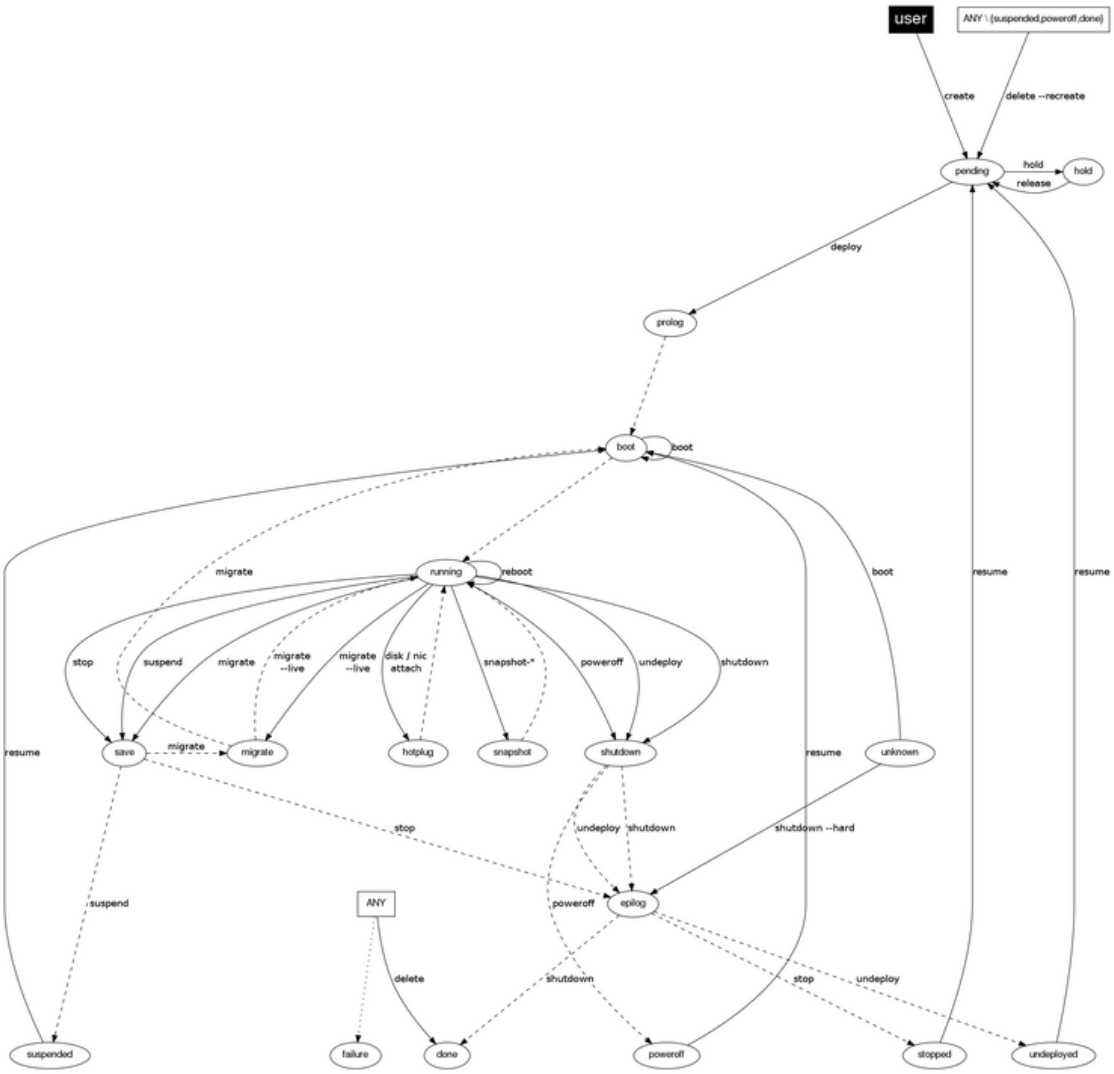
The menu on the top left of the page provides information on:

- Dashboard
- Virtual Resources
- Infrastructure
- Market Place

4. VM Life-Cycle

The life-cycle of a Virtual Machine (VM) within OpenNebula includes a number of stages, e.g. boot, running, stopped etc.

The life cycle is represented in the following diagram.



For details see the document page [4].

5. VM Creation

Virtual Machines run on top of the physical infrastructure and are created from templates. These templates make use of images that represent disk storage inside the VM.

Basic Non-Persistent VMs

To instantiate a VM select **Virtual Resources -> Virtual Machines** from the menu and press the add VM button.

A list of templates and images are available for use, will be provided e.g. Debian Squeeze x86_64 (VNC, DHCP). VMs can be instantiated by simply choosing a template and clicking on “Instantiate”. The status of the VM can be checked accordingly, e.g. host on which it is running. VM Logs and Monitoring Information are also available.

You can log into the VM as **root**. The password is … **password**

Should you require specific VM images please **email support**.

Caveat: VMs instantiated in this way will however not be persistent and once closed down all changes made to them will be lost. This can be changed by making the VM’s disk image persistent.

Persistent VMs

To make a VM persist a few additional steps are required before booting a VM.

1. The first step is to clone the template. Click **Virtual Resources -> Templates**, select a template and then click clone.
2. The template will have an image attached to it, make a note of which image is in use.

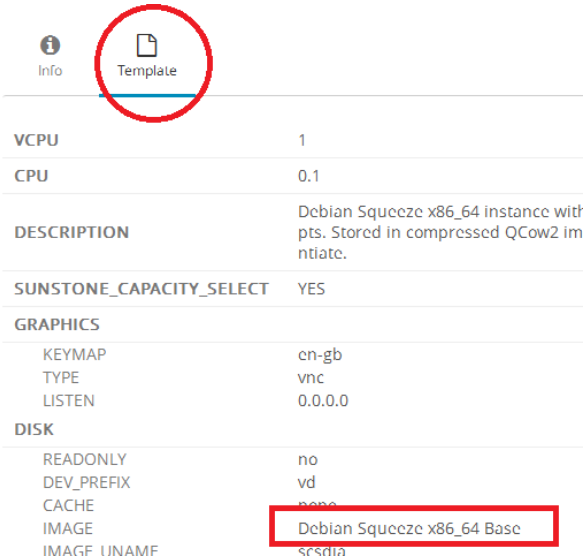


Figure 1 - Finding a VMs Disk Image

3. Now clone this disk image (see: **Virtual Resources -> Images)**.

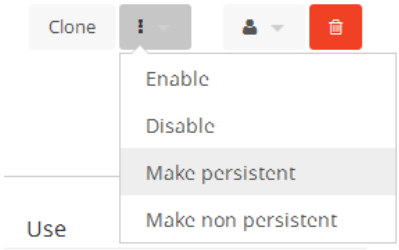


Figure 2 - Making Disk Images Persistent

4. Now make the disk image persistent. This will mean that data written to this image will survive past the shutdown of the VM and it also means only one VM can use this disk image at once.
5. The template that you cloned will now need updating to use the new disk image. Select the template and click **update**.
6. Now select **Storage** and then select the persistent disk image that you created (see Figure 3.

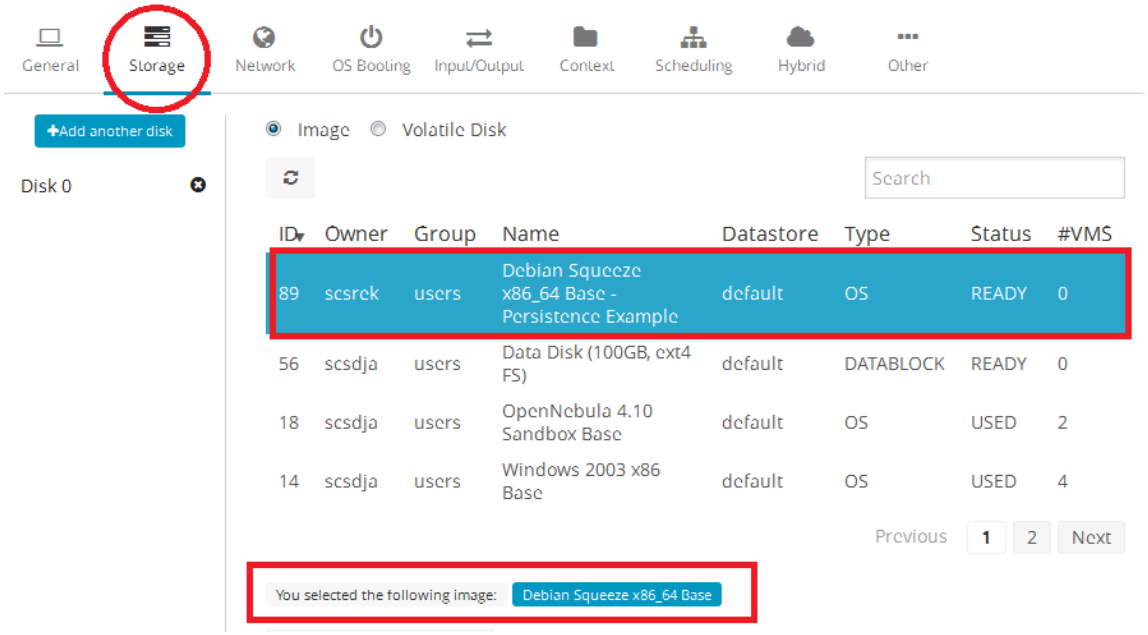


Figure 3- Changing a templates disk image

7. Now click **Update**.
8. A persistent VM can now be created by utilising the template that has just been created, by clicking **Virtual Resources -> Virtual Machines** and then using the add new VM button.

6. OpenVPN Access

These instructions show how to access the cloud testbed from Windows. It requires access to the University VPN first, then to the testbed through OpenVPN [5].

- Install OpenVPN on your Windows machine
- After installing OpenVPN, extract the zip file cscloud-vpn.zip [6] into C:\Program Files\OpenVPN\config
- Change "password" to your password in C:\Program Files\OpenVPN\config\cscloud\login.conf
- Copy "OpenVPN\OpenVPN GUI" shortcut to the "startup" folder in Windows Start Menu
- Edit properties on new shortcut to "Run this program as Administrator" in the Compatibility tab (Might have to enter Admin credentials here)
- Reboot and the OpenVPN GUI icon should be available in the taskbar.
- Select cscloud after connecting to the University VPN through Pulse.
- The cloud testbed can then be accessed on **http://10.0.0.100:9869/login**

7. Using Zabbix on the Testbed

Zabbix [7] is a monitoring tool that allows us to monitor the physical hosts and on the School's Cloud computing testbed. In addition it can see physical hosts and VMs on the Distributed Systems and Services older Cloud Computing Testbed (Noting: that physical hosts on this cloud names are in the series TestnodeX where X = 1..8.)

Zabbix may be accessed at the following address:

http://10.10.0.1/dashboard.php

You will be logged in as a guest, which will allow you access to view what is happening on the testbed. The most useful tab for you to use is the graphs tab. It can be seen under **monitoring -> Graphs** and is shown below.

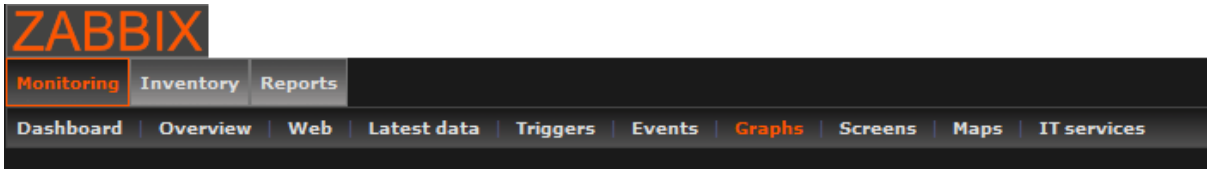


Figure 4 - Graphs in Zabbix

There is a drop down box at the right hand side that allows you to select groups of machines.

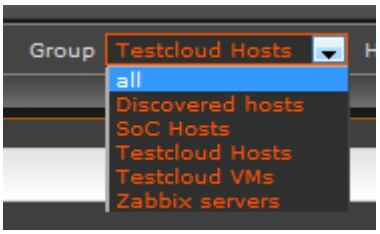


Figure 5 - Groups in Zabbix

The options are:

- Discovered hosts – which are VMs detected on the cloud testbed
- SoC Hosts – that are the physical hosts on the school's cloud computing testbed
- Testcloud hosts – that are physical hosts on the DSS cloud testbed
- Testcloud VMs – which are VMs on the DSS cloud testbed
- Zabbix Servers – Which is testnode1, which is where the Zabbix server resides (i.e. 10.10.0.1 that you typed in earlier)

The next drop down box provides you the host to monitor and the final column allows you to select the graph to display.

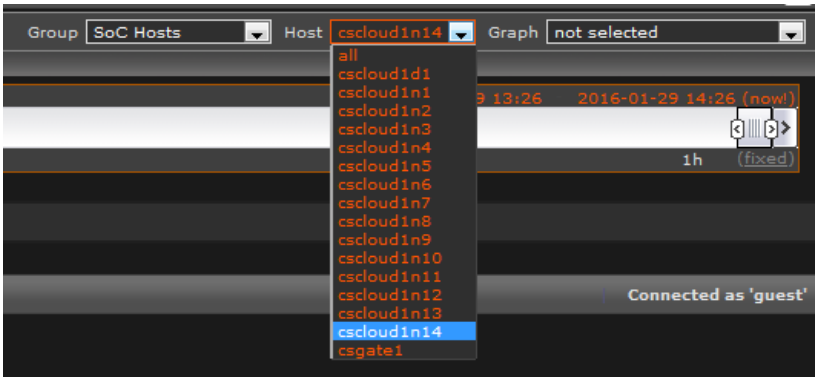


Figure 6 - Hosts and their Graphs in Zabbix

In addition to the graphs page, it is possible to see a list of current metrics gathered from the testbed by clicking on **monitoring -> Latest data**. This can be seen in Figure 7.

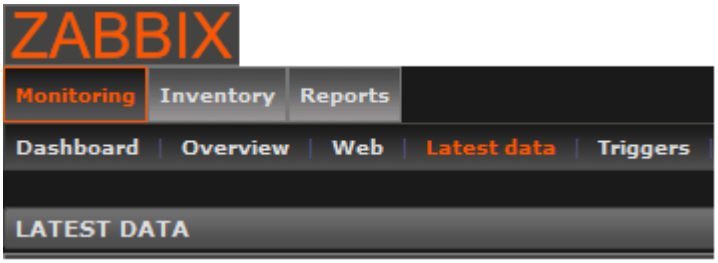


Figure 7 - Latest Data

Running the Zabbix Data Logger Tool on the Testbed

An additional tool for monitoring the testbed or cloud is provided with this guide. This is called the Zabbix Data logger. It allows you to have the values that are recorded in Zabbix written to disk in a live fashion as soon as the values arrive.

It is also available at: **https://github.com/RichardKav/zabbix-data-logger**

1. Extract the Zip file: Zabbix Data Logger.zip. The following command will achieve this.

   unzip "Zabbix Data Logger.zip" -d .

2. Open NetBeans IDE, by typing netbeans on the console or "Main Menu -> Programming -> NetBeans IDE 7.2" on a cslin workstation.

3. Compile the application.

4. In the folder "Zabbix Data Logger/dist" the compiled binary Zabbix_Data_Logger.jar should have been created.

5. This jar can be run with the command:

   java -jar Zabbix_Data_Logger.jar testnode1

6. Zabbix lists all hosts that it can monitor at the following url: **http://10.10.0.1/charts.php** visit this page and choose another host to monitor. i.e. testnode2 or VM10-10-1-5 etc. Ensure that if you chose a VM to monitor, that it marked as available. The general way for executing this tool is:

   java -jar Zabbix_Data_Logger.jar <hostname>

   Once running the tool can be closed by writing the word quit. It will output a file containing all the metric data gathered for the host/VM that you named.

Offline Gathering of Data from Zabbix

The Zabbix-data logger tool gathers the current data regarding hosts in Zabbix. If you are interested in historical values then Zabbix data collector script is an alternative.

It is also available at: **https://github.com/RichardKav/zabbix-data-collector**

The Zabbix data collector is designed to collect data directly from the Zabbix database.

At the start of the script the database connection information for connecting to Zabbix is held.

username=zabbixinfo
database=zabbix
password=readonly
hostname=10.10.0.1

These should be set to the following values if they are not already.

1. Download a copy and place it in a directory within a computer connected to VPN with mysql client installed.
2. Make the script executable. i.e. chmod u+x zabbix_data_collector.sh
3. It is now executable and its usage is as follows:

   ./zabbix_data_collector.sh <Hostname> <Metric> <Start_time_unix_time> <end_time_unix_time>

An example of this is: ./zabbix_data_collector.sh testnode1 power 1439337600 1439373722

If you need to convert times to Unix time an online converter as indicated below will be suitable:

http://www.onlineconversion.com/unix_time.htm

Advanced Capabilities

Energy Measurements

In order to aid research there is a Watt Meter attached to cscloud1n14.cloud.comp.leeds.ac.uk. This allows for additional measurements of power consumption and overall energy used to be gained. Additional Watt Meters are accessible on the DSS cloud and their output can be seen in Zabbix.

Graphics Card

On hosts cscloud1n1 and cscloud1n2 there are graphics cards installed, for the purpose of executing graphic card accelerated parallel workloads, for more details on their usage please contact technical support.

References

1. OpenNebula. http://opennebula.org/
2. Emulab. http://emulab.net/
3. Planetlab. http://www.planet-lab.org/
4. OpenNebula. Managing Virtual Machines. http://docs.opennebula.org/4.4/user/virtual_resource_management/vm_guide_2.html#virtual-machine-life-cycle
5. OpenVPN. https://openvpn.net/
6. School of Computing. /usr/not-backed-up/karim/research/cloud/cscloud-vpn.zip on cslin136.leeds.ac.uk
7. Zabbix. http://www.zabbix.com/