16 máquinas virtuais em duas máquinas reais - Variação do uso de memória

- Máquina Hospedeira: Educloud05, Educloud10

Ambiente (Máquina Hospedeira):

- 8 Cores de 2.67 GHz;
- 6 GB de Ram;

Ambiente (Máquinas Virtuais):

- 1 Cores de 2.67 GHz;
- 0.625 GB de Ram;

- Variação de memória de 60% a 86% de uso;

- QxP: 2x8;

Resultados:

Cálculo do maior número de operações por segundo:

- Valor de NB ótimo: 160 (82% de uso de memória);
- Número de Operações de Ponto Flutuante por Segundo: 1.31604*10^02 GFlops;

- Gráfico de Tempo vs NB vs Número de Operações:
  https://ncc.unesp.br/its/attachments/293/NBvsGflopsvsTime.png

Estudo da utilização da memória:

- Gráfico do Número de operações Vs Tamanho de problema (NB) para porcentagens de memória de 60% à 86%
  https://ncc.unesp.br/its/attachments/294/NBvsGflops.png

- Gráfico do Tempo Vs Tamanho de problema (NB) para porcentagens de memória de 60% à 86%
  https://ncc.unesp.br/its/attachments/295/NBvsTime.png

- Gráfico da Variação do Número de Operações para cada percentagem de memória.
  https://ncc.unesp.br/its/attachments/296/MemvsGflops.png

- Gráfico da Variação do Tempo para cada percentagem de memória.
  https://ncc.unesp.br/its/attachments/297/MemvsTime.png

Discussão.

- Com a porcentagem de 82% notamos um pico no número de operações por segundos (~ 131 Gflops), com tamanho de bloco 160 para 40 Gflops em tamanho de bloco 216 ;
  - Após essa porcentagem o HPL benchmark começa a ter problemas com alocação de memória:

HPL ERROR from process # 0, on line 240 of function HPL_pdtest:
[1,7] Memory allocation failed for A, x and b. Skip. <<<

  - Um dmesg na máquina virtual obtemos:

[912792.250504] Out of memory: Kill process 27830 (xhpl_hybrid_int) score 750 or sacrifice child
[912792.252876] Killed process 27830 (xhpl_hybrid_int) total-vm:541372kB, anon-rss:480184kB, file-rss:0kB

- Como a máquina virtual foi criada sem swap, as mensagens do kernel indicam que esse foi o momento no qual o processo do hpl começou a utilizar a swap;

- O pico de operações por segundo foi maior do que o dos experimentos com 4 vms e 8 vms distribuídas em duas máquinas reais, com ~ 128 Gflops e ~ 124 Gflops;

Files

NBvsGflopsvsTime.png	7.71 KB	26 May 2013	Raphael Cóbe
NBvsGflops.png	8.1 KB	26 May 2013	Raphael Cóbe
NBvsTime.png	6.39 KB	26 May 2013	Raphael Cóbe
MemvsGflops.png	6.19 KB	26 May 2013	Raphael Cóbe
MemvsTime.png	4.95 KB	26 May 2013	Raphael Cóbe

1 Máquina Real Vs 1 Máquina Virtual

Máquina Real

- Máquina Educloud05:

Ambiente:

- 8 Cores de 2.67 GHz;
- 6 GB de Ram;
- 92% de Uso de memória:
- QxP: 2x4;

Resultados:

- Valor de NB ótimo: 184
- Número de Operações de Ponto Flutuante por Segundo: 7.82992*10^01 GFlops;
- Gráfico de Tempo vs NB vs Número de Operações:
  https://ncc.unesp.br/its/attachments/232/NBvsGflopsvsTime.png

Máquina Virtual

- Máquina Hospedeira: Educloud05:

Ambiente:

- 8 Cores de 2.67 GHz;
- 6 GB de Ram;

Resultados:

- Valor de NB ótimo: 144 (90% de uso de memória);
- Número de Operações de Ponto Flutuante por Segundo: 7.63280*10^01 GFlops;
- Gráfico de Tempo vs NB vs Número de Operações:
  https://ncc.unesp.br/its/attachments/247/NBvsGflopsvsTime.png

Comparação

- **Gráfico Comparativo entre Máquina Virtual e HW Real:**
  https://ncc.unesp.br/its/attachments/254/NBvsGflopsvsTime.png

---

Discussão

- **A diferença entre o HW real e a VM não se mostrou tão grande quanto esperávamos;**

Files

NBvsGflopsvsTime.png	11.7 KB	29 Apr 2013	Raphael Cóbe

2 máquinas virtuais em uma máquina real - Variação do uso de memória

- Máquina Hospedeira: Educloud05

Ambiente (Máquina Hospedeira):

- 8 Cores de 2.67 GHz;
- 6 GB de Ram;

Ambiente (Máquinas Virtuais):

- 4 Cores de 2.67 GHz;
- 2.5 GB de Ram;

- Variação de memória de 70% a 86% de uso;

- QxP: 2x8;

Resultados:

Cálculo do maior número de operações por segundo:

- Valor de NB ótimo: 144 (82% de uso de memória);
- Número de Operações de Ponto Flutuante por Segundo: 7.23732*10^01 GFlops;

- Gráfico de Tempo vs NB vs Número de Operações:
  https://ncc.unesp.br/its/attachments/300/NBvsGflopsvsTime.png

Estudo da utilização da memória:

- Gráfico do Número de operações Vs Tamanho de problema (NB) para porcentagens de memória de 70% à 86%
  https://ncc.unesp.br/its/attachments/301/NBvsGflops.png

- Gráfico do Tempo Vs Tamanho de problema (NB) para porcentagens de memória de 70% à 86%
  https://ncc.unesp.br/its/attachments/302/NBvsTime.png

- Gráfico da Variação do Número de Operações para cada percentagem de memória.
  https://ncc.unesp.br/its/attachments/303/MemvsGflops.png

- Gráfico da Variação do Tempo para cada percentagem de memória.
  https://ncc.unesp.br/its/attachments/304/MemvsTime.png

Discussão.

- Não foi possível encontrar o ponto onde a swap entra em ação e o desempenho começa a cair. A partir de 86% de uso de memória o processo é terminado pelo kernel por falta de memória;

Files

NBvsGflopsvsTime.png	9.61 KB	28 May 2013	Raphael Cóbe
NBvsGflops.png	14.7 KB	28 May 2013	Raphael Cóbe
NBvsTime.png	12 KB	28 May 2013	Raphael Cóbe
MemvsGflops.png	7.21 KB	28 May 2013	Raphael Cóbe
MemvsTime.png	5.37 KB	28 May 2013	Raphael Cóbe

4 Máquinas Reais Vs 4 Máquinas Virtuais

Máquinas Reais

- Máquina Educloud05-08:

Ambiente:

- 8 Cores de 2.67 GHz;
- 6 GB de Ram;
- QxP: 4x8;

Resultados:

- Valor de NB ótimo: 112 (94% de Uso de memória)
- Número de Operações de Ponto Flutuante por Segundo: 2.11695*10^02 GFlops;
- Gráfico de Tempo vs NB vs Número de Operações:
  https://www.ncc.unesp.br/its/attachments/237/NBvsGflopsvsTime.png

Máquinas Virtuais

- Máquina Hospedeira: Educloud05-08:

Ambiente:

- 8 Cores de 2.67 GHz;
- 6 GB de Ram;

Resultados:

- Valor de NB ótimo: 144 (90% de uso de memória);
- Número de Operações de Ponto Flutuante por Segundo: 2.05980*10^02 GFlops;
- Gráfico de Tempo vs NB vs Número de Operações:
  https://www.ncc.unesp.br/its/attachments/255/NBvsGflopsvsTime.png

Comparação

- **Gráfico Comparativo entre Máquina Virtual e HW Real em termos de número de Operações:
  https://www.ncc.unesp.br/its/attachments/263/NBvsGflops.png**

Discussão

- **A diferença entre o HW real e a VM não se mostrou tão grande quanto esperávamos;**

Files

NBvsGflops.png	7.15 KB	03 May 2013	Raphael Cóbe

4 máquinas virtuais em duas máquinas reais - Variação do uso de memória

- Máquina Hospedeira: Educloud05, Educloud06

Ambiente (Máquina Hospedeira):

- 8 Cores de 2.67 GHz;
- 6 GB de Ram;

Ambiente (Máquinas Virtuais):

- 2 Cores de 2.67 GHz;
- 1.25 GB de Ram;

- Variação de memória de 70% a 82% de uso;

- QxP: 2x8;

Resultados:

Cálculo do maior número de operações por segundo:

- Valor de NB ótimo: 128 (82% de uso de memória);
- Número de Operações de Ponto Flutuante por Segundo: 1.28124*10^02;

- Gráfico de Tempo vs NB vs Número de Operações:
  https://ncc.unesp.br/its/attachments/311/NBvsGflopsvsTime.png

Estudo da utilização da memória:

- Gráfico do Número de operações Vs Tamanho de problema (NB) para porcentagens de memória de 70% à 82%
  https://ncc.unesp.br/its/attachments/312/NBvsGflops.png

- Gráfico do Tempo Vs Tamanho de problema (NB) para porcentagens de memória de 70% à 82%
  https://ncc.unesp.br/its/attachments/313/NBvsTime.png

- Gráfico da Variação do Número de Operações para cada percentagem de memória.
  https://ncc.unesp.br/its/attachments/314/MemvsGflops.png

- Gráfico da Variação do Tempo para cada percentagem de memória.
  https://ncc.unesp.br/its/attachments/315/MemvsTime.png

Discussão.

- Não foi possível encontrar o ponto onde a swap entra em ação e o desempenho começa a cair. A partir de 82% de uso de memória o processo é terminado pelo kernel por falta de memória;

Files

NBvsGflopsvsTime.png	9.93 KB	23 May 2013	Raphael Cóbe
NBvsGflops.png	16.2 KB	23 May 2013	Raphael Cóbe
NBvsTime.png	15.5 KB	23 May 2013	Raphael Cóbe
MemvsGflops.png	6.35 KB	23 May 2013	Raphael Cóbe
MemvsTime.png	5.36 KB	23 May 2013	Raphael Cóbe
NBvsGflopsvsTime.png	9.85 KB	28 May 2013	Raphael Cóbe
NBvsGflops.png	13.8 KB	28 May 2013	Raphael Cóbe
NBvsTime.png	12.9 KB	28 May 2013	Raphael Cóbe
MemvsGflops.png	6.74 KB	28 May 2013	Raphael Cóbe
MemvsTime.png	5.41 KB	28 May 2013	Raphael Cóbe

4 máquinas virtuais em uma máquina real - Variação do uso de memória

- Máquina Hospedeira: Educloud06

Ambiente (Máquina Hospedeira):

- 8 Cores de 2.67 GHz;
- 6 GB de Ram;

Ambiente (Máquinas Virtuais):

- 2 Cores de 2.67 GHz;
- 1.25 GB de Ram;

- Variação de memória de 70% a 78% de uso;

- QxP: 2x4;

Resultados:

Cálculo do maior número de operações por segundo:

- Valor de NB ótimo: 152 (74% de uso de memória);
- Número de Operações de Ponto Flutuante por Segundo: 7.05979*10^01 GFlop;

- Gráfico de Tempo vs NB vs Número de Operações:
  https://ncc.unesp.br/its/attachments/310/NBvsGflopsvsTime.png

Estudo da utilização da memória:

- Gráfico do Número de operações Vs Tamanho de problema (NB) para porcentagens de memória de 70% à 76%
  https://ncc.unesp.br/its/attachments/306/NBvsGflops.png

- Gráfico do Tempo Vs Tamanho de problema (NB) para porcentagens de memória de 70% à 76%
  https://ncc.unesp.br/its/attachments/307/NBvsTime.png

- Gráfico da Variação do Número de Operações para cada percentagem de memória.
  https://ncc.unesp.br/its/attachments/308/MemvsGflops.png

- Gráfico da Variação do Tempo para cada percentagem de memória.
  https://ncc.unesp.br/its/attachments/309/MemvsTime.png

Discussão.

- Não foi possível encontrar o ponto onde a swap entra em ação e o desempenho começa a cair. A partir de 86% de uso de memória o processo é terminado pelo kernel por falta de memória;

Files

NBvsGflops.png	8.26 KB	28 May 2013	Raphael Cóbe
NBvsTime.png	6.43 KB	28 May 2013	Raphael Cóbe
MemvsGflops.png	6.69 KB	28 May 2013	Raphael Cóbe
MemvsTime.png	5.31 KB	28 May 2013	Raphael Cóbe
NBvsGflopsvsTime.png	10.1 KB	28 May 2013	Raphael Cóbe

8 máquinas virtuais em duas máquinas reais - Variação do uso de memória

- Máquina Hospedeira: Educloud07, Educloud08

Ambiente (Máquina Hospedeira):

- 8 Cores de 2.67 GHz;
- 6 GB de Ram;

Ambiente (Máquinas Virtuais):

- 2 Cores de 2.67 GHz;
- 1.25 GB de Ram;

- Variação de memória de 60% a 86% de uso;

- QxP: 2x8;

Resultados:

Cálculo do maior número de operações por segundo:

- Valor de NB ótimo: 216 (86% de uso de memória);
- Número de Operações de Ponto Flutuante por Segundo: 1.24232*10^02 GFlops;

- Gráfico de Tempo vs NB vs Número de Operações:
  https://ncc.unesp.br/its/attachments/316/NBvsGflopsvsTime.png

Estudo da utilização da memória:

- Gráfico do Número de operações Vs Tamanho de problema (NB) para porcentagens de memória de 60% à 86%
  https://ncc.unesp.br/its/attachments/317/NBvsGflops.png

- Gráfico do Tempo Vs Tamanho de problema (NB) para porcentagens de memória de 60% à 86%
  https://ncc.unesp.br/its/attachments/318/NBvsTime.png

- Gráfico da Variação do Número de Operações para cada percentagem de memória.
  https://ncc.unesp.br/its/attachments/319/MemvsGflops.png

- Gráfico da Variação do Tempo para cada percentagem de memória.
  https://ncc.unesp.br/its/attachments/320/MemvsTime.png

Discussão.

- Após 86% de uso de memória o HPL benchmark começa a ter problemas com alocação de memória:

  HPL ERROR from process # 0, on line 240 of function HPL_pdtest:
  [1,7] Memory allocation failed for A, x and b. Skip. <<<

  - Um dmesg na máquina virtual obtemos:

  [912792.250504] Out of memory: Kill process 27830 (xhpl_hybrid_int) score 750 or sacrifice child
  [912792.252876] Killed process 27830 (xhpl_hybrid_int) total-vm:541372kB, anon-rss:480184kB, file-rss:0kB

  - Como a máquina virtual foi criada sem swap, as mensagens do kernel indicam que esse foi o momento no qual o processo do hpl começou a utilizar a swap;

Files

NBvsGflopsvsTime.png	9.65 KB	28 May 2013	Raphael Cóbe
NBvsGflops.png	16.2 KB	28 May 2013	Raphael Cóbe
NBvsTime.png	10.3 KB	28 May 2013	Raphael Cóbe
MemvsGflops.png	6.47 KB	28 May 2013	Raphael Cóbe
MemvsTime.png	5.15 KB	28 May 2013	Raphael Cóbe

8 máquinas virtuais em uma máquina real - Variação do uso de memória

- Máquina Hospedeira: Educloud07

Ambiente (Máquina Hospedeira):

- 8 Cores de 2.67 GHz;
- 6 GB de Ram;

Ambiente (Máquinas Virtuais):

- 1 Core de 2.67 GHz;
- 0.625 GB de Ram;

Discussão

Para esse teste, a única porcentagem de memória que conseguimos executar foi 70%. A partir desse ponto o processo era terminado pelo kernel por falta de memória.

Cálculo do maior número de operações por segundo:

- Valor de NB ótimo: 152;
- Número de Operações de Ponto Flutuante por Segundo: 7.21625*10^01 GFlops;

- Gráfico de Tempo vs NB vs Número de Operações:
  https://ncc.unesp.br/its/attachments/299/NBvsGflopsvsTime.png

Files

NBvsGflopsvsTime.png	8.7 KB	28 May 2013	Raphael Cóbe

Abrir Console do Virtualizador

Para executar esta tarefa é necessário realizar os seguintes passos:

- Em termos de arquitetura do sistema, é necessário uma máquina para ter o papel de gerenciador das VMs a serem submetidas; uma máquina que gerencia as imagens das VMs (repositório de imagens) e os nós onde serão submetidas as VMs

- Na máquina com o papel de gerenciador das VMs, será instalado o último release estável do Open Nebula (3.2.1 no momento da elaboração deste) via source. O procedimento para instalação é o descrito a seguir:
  - *Instalar as dependências do open nebula: g++, libxmlrpc-c3-dev, scons, libsqlite3-dev, libmysqlclient-dev, libxml2-dev, libssl-dev, ruby;*
  - *Realizar o download do pacote opennebula-3.2.1.tar.gz;*
  - *Descompactar o pacote opennebula-3.2.1.tar.gz;*
  - *Acessar o diretório previamente descompactado;*
  - *executar o comando: scons sqlite=no mysql=yes parses=yes*
  - *criar o grupo oneadmin: groupadd oneadmin*
  - *criar o usuário oneadmin: useradd -d /var/lib/one -g oneadmin -s /bin/bash oneadmin*
  - *criar o diretório home do usuário oneadmin: mkdir /var/lib/oneadmin*
  - *instalar o opennebula: ./install.sh -u oneadmin -g oneadmin -c -s -o*
  - *instalar as gems do ruby: ./share/install_gems/install_gems*
  - *Instalar o nfs server: apt-get install nfs-kernel-server*
  - *exportar o home do oneadmin para os nós da rede: echo "/var/lib/one 200.145.46.0/24(rw,sync,no_subtree_check,no_root_squash)" > /etc/exports*
  - *instalar os scripts tm_clone.sh, tm_context.sh, tm_delete.sh, tm_ln.sh, tm_mkimage.sh, tm_mkswap.sh, tm_mv.sh no diretório /usr/lib/one/tm_commands/iscsi*
  - *adicionar as seguintes linhas ao arquivo /etc/sudoers:*
    - *oneadmin ALL=(ALL) NOPASSWD: /sbin/lvcreate*
    - *oneadmin ALL=(ALL) NOPASSWD: /sbin/lvremove*
    - *oneadmin ALL=(ALL) NOPASSWD: /sbin/lvs*
    - *oneadmin ALL=(ALL) NOPASSWD: /usr/local/bin/iscsiadd*
    - *oneadmin ALL=(ALL) NOPASSWD: /usr/local/bin/iscsirm*
- Na máquina com o papel de repositório de imagens é necessário os seguintes procedimentos:
  - *criar o usuário oneadmin: useradd -d /var/lib/one -g oneadmin -s /bin/bash oneadmin*
  - *criar o home do usuário oneadmin: mkdir /var/lib/one*
  - *montar o home do oneadmin: mount -t nfs $SERVER:/var/lib/one /var/lib/one*
  - *instalar as seguintes dependências: openssh-server, lvm, iscsitarget*
  - *instalar os scripts iscsiadd e iscsirm (em anexo) no diretório /usr/local/bin*

- *Criar partição lvm: vgcreate osdc1 /dev/sda1*

- Nas máquinas com o papel de nós onde serão submetidas as VMs são necessários os seguintes procedimentos:
  - *Instalar as dependências: openssh, libvirt-bin, ruby*
  - *criar o grupo oneadmin: groupadd oneadmin*
  - *criar o usuário oneadmin: useradd -d /var/lib/one -g oneadmin -s /bin/bash oneadmin*
  - *criar o home do usuário oneadmin: mkdir /var/lib/one*
  - *montar o home do oneadmin: mount -t nfs $SERVER:/var/lib/one /var/lib/one*
  - *instalar os pacotes de instância de iSCSI*
  - *criar bridge LAN para a interface eth1 e configurar regras de roteamento:*
    - *brctl addbr virbr1*
    - *iptables -A INPUT -i virbr1 -p udp -m udp --sport 67:68 --dport 67:68 -j ACCEPT*
    - *iptables -A FORWARD -i virbr1 -o virbr1 -j ACCEPT*
    - *iptables -t nat -A POSTROUTING -s 192.168.123.0/24 -j MASQUERADE*

- Após realizar os procedimentos acima, voltar ao gerenciador e executar os seguintes comandos(com o usuário oneadmin):
  - *onehost create osdc06 im_kvm vmm_kvm tm_iscsi*
  - *onehost create osdc07 im_kvm vmm_kvm tm_iscsi*
  - *onehost create osdc03 im_kvm vmm_kvm tm_iscsi*
  - *sunstone-server -H 0.0.0.0 start*

Voltar

Files

iscsiadd	299 Bytes	21 Mar 2012	Marco Gomes
iscsirm	496 Bytes	21 Mar 2012	Marco Gomes
tm_clone.sh	2.85 KB	21 Mar 2012	Marco Gomes
tm_context.sh	2.23 KB	21 Mar 2012	Marco Gomes
tm_delete.sh	2.21 KB	21 Mar 2012	Marco Gomes
tm_ln.sh	1.54 KB	21 Mar 2012	Marco Gomes
tm_mkimage.sh	1.66 KB	21 Mar 2012	Marco Gomes
tm_mkswap.sh	1.71 KB	21 Mar 2012	Marco Gomes

Ambiente de Desenvolvimento

Lista de Servidores

Educloud00 (LVM) keystone, glance, cinder, nova-*
  Eth0 - 00:15:17:DE:B7:24 - 200.145.46.165/24 - educloud00.ncc.unesp.br CNAME cloud.ncc.unesp.br
  Eth1 - 00:15:17:DE:B7:25

Educloud01 (TPM01)
  Eth0 - 00:1e:67:3c:7c:df - 200.145.46.166/24 - educloud01.local
  Eth1 - 00:1e:67:3c:7c:e0
  Eth2 - 00:1e:67:3c:7c:e1
  Eth3 - 00:1e:67:3c:7c:e2

Educloud02 (TPM02)
  Eth0 - 00:1e:67:3c:89:ce - 200.145.46.167/24 - educloud02.local
  Eth1 - 00:1e:67:3c:89:cf
  Eth2 - 00:1e:67:3c:89:d0
  Eth3 - 00:1e:67:3c:89:d1

Educloud03
  Eth0 - 00:15:17:da:53:92 - 200.145.46.168/24 - educloud03.local
  Eth1 - 00:15:17:da:53:93
  Eth2 - 00:15:17:da:53:94
  Eth3 - 00:15:17:da:53:95
  Eth4 - 00:15:17:da:53:96

Educloud04
  Eth0 - 00:15:17:da:55:62 - 200.145.46.169/24 - educloud04.local
  Eth1 - 00:15:17:da:55:63
  Eth2 - 00:15:17:da:55:64
  Eth3 - 00:15:17:da:55:65
  Eth4 - 00:15:17:da:55:66

Educloud05
  Eth0 - 00:26:18:6d:65:a6 - 200.145.46.170/24 - educloud05.local
  Eth1 - 00:26:18:6d:65:a7

Educloud06
  Eth0 - 00:26:18:6d:61:64 - 200.145.46.171/24 - educloud06.local
  Eth1 - 00:26:18:6d:61:65

Educloud07
  Eth0 - 00:26:18:6d:67:ba - 200.145.46.172/24 - educloud07.local
  Eth1 - 00:26:18:6d:67:bb

Educloud08
  Eth0 - 00:26:18:6d:67:be - 200.145.46.173/24 - educloud08.local
  Eth1 - 00:26:18:6d:67:bf

Educloud09
  Eth0 - 00:26:18:6d:65:22 - 200.145.46.174/24 - educloud09.local
  Eth1 - 00:26:18:6d:65:23

Educloud10
  Eth0 - 00:26:18:6d:65:3e - 200.145.46.175/24 - educloud10.local
  Eth1 - 00:26:18:6d:65:3f

Introduction

This document refers to the installation manual of the main components of OpenStack. The version of each component used can be found in the respective sections of the components.

This document was produced by the Center for Scientific Computing, UNESP, with funding from Intel and UNESP.

OpenStack OpenStack is a global collaboration of developers and cloud computing technologists producing the ubiquitous open source cloud computing platform for public and private clouds. The project aims to deliver solutions for all types of clouds by being simple to implement, massively scalable, and feature rich. The technology consists of a series of interrelated projects delivering various

components for a cloud infrastructure solution.

The OpenStack project is divided into sub projects that have a specific role in the big picture.

The main components are: Cloud Compute (Nova) Cloud Storage (Swift), Image registration and delivery (Glance), Identity (Keystone) and Dashboard (Horizon).

Each component is independent of each other and can be installed on separate machines, or all in one. In the environment of UNESP, OpenStack was installed in seven machines, described in section "Environment used."

This document is modified version of OpenStack Install and Deploy Manual [1] with minor adjustments and changes to resolve problems encountered during installation and adjustment to the environment in which it was installed.

OpenStack Core Components

OpenStack is a collection of open source technology that provides massively scalable open source cloud computing software. Currently OpenStack develops two related projects: OpenStack Compute, which offers computing power through virtual machine and network management, and OpenStack Object Storage which is software for redundant, scalable object storage capacity. Closely related to the OpenStack Compute project is the Image Service project, named Glance. OpenStack can be used by corporations, service providers, VARS, SMBs, researchers, and global data centers looking to deploy large-scale cloud deployments for private or public clouds.

The main components are: Cloud Compute (Nova) Cloud Storage (Swift), Image registration and delivery (Glance), Identity (Keystone) and Dashboard (Horizon).

They are presented here in an order for educational purposes only to a straightforward procedure for installation. But following a brief description of each component.

Keystone

Keystone is an OpenStack project that provides Identity, Token, Catalog and Policy services for use specifically by projects in the OpenStack family. It implements OpenStack's Identity API.

Cloud Compute (Nova)

Nova is the project name for OpenStack Compute, a cloud computing fabric controller, the main part of an IaaS system. Individuals and organizations can use Nova to host and manage their own cloud computing systems. Nova originated as a project out of NASA Ames Research Laboratory.

Object Storage (Swift)

OpenStack Object Storage (Swift) is a system to store objects in a massively scalable large capacity system with built-in redundancy and failover. Object Storage has a variety of applications, such as backing up or archiving data, serving graphics or videos (streaming data to a user's browser), storing secondary or tertiary static data, developing new applications with data storage integration, storing data when predicting storage capacity is difficult, and creating the elasticity and flexibility of cloud-based storage for your web applications.

Image Service (Glance)

OpenStack Image Service (Glance) is a lookup and retrieval system for virtual machine images. It can be configured in three ways: using OpenStack Object Store to store images; using Amazon's Simple Storage Solution (S3) storage directly; or using S3 storage with Object Store as the intermediate for S3 access.

Environment Proposed

This document aims to guide you in setting up a test environment or development. The original documentation suggests many machines. To install OpenStack here,

you will need only one or two machines.

You can install all services on one machine only (all-in-one). Or you can install OpenStack controller (all services except the nova-compute) on machine A and a compute cloud controller on machine B.

For a production environment it is advisable to separate the elements to ensure a high availability and redundancy.

Settings used in this document:
- Cloud Controller:
- Cloud Computer Node:

All machines are installed with Ubuntu Server 12.04 LTS.

Initial configuration

Before starting the installation and configuration of each component must first to make some adjustments.

Network Configuration

For both conserving network resources and ensuring that network administrators understand the needs for networks and public IP addresses for accessing the APIs and VMs as necessary, this section offers recommendations and required minimum sizes. Throughput of at least 1000 Mbps is suggested.

For OpenStack Compute, networking is configured on multi-node installations between the physical machines on a single subnet. For networking between virtual machine instances, three network options are available: flat, DHCP, and VLAN. Two NICs (Network Interface Cards) are recommended on the server running nova-network.

Management Network (RFC1918 IP Range, not publicly routable): This network is utilized for all inter-server communications within the cloud infrastructure. Recommended size: 255 IPs (CIDR /24)

Public Network (Publicly routable IP range): This network is utilized for providing Public IP accessibility to the API endpoints within the cloud infrastructure. Minimum size: 8 IPs (CIDR /29)

VM Network (RFC1918 IP Range, not publicly routable): This network is utilized for providing primary IP addresses to the cloud instances. Recommended size: 1024 IPs (CIDR /22)

Floating IP network (Publicly routable IP Range): This network is utilized for providing Public IP accessibility to selected cloud instances. Minimum size: 16 IPs (CIDR /28)

In this installation we will use eth0 to public interface and eth1 for management interface.

Installing Network Time Protocol (NTP)

To keep all the services in sync, you need to install NTP, and if you do a multi-node configuration you will configure one server to be the reference server.

1. apt-get install -y ntp

Set up the NTP server on your controller node so that it receives data by modifying the ntp.conf file and restarting the service. As root:

1. sed -i 's/server ntp.ubuntu.com/server ntp.ubuntu.com\nserver 127.127.1.0\
   nfudge 127.127.1.0 stratum 10/g' /etc/ntp.conf
2. service ntp restart

Set up the NTP client on your compute node so that the time between controller node and compute node is synchronized. Do the following as a cron job on compute node.

1. ntpdate 'controllernode ip'
2. hwclock -w

Storage

Many cloud computing use cases require persistent remote storage. Storage solutions are often divided into three categories: object storage, block storage, and file storage.

Here we will use a "object storage" and a "block storage".

Object storage (swift)

With object storage, files are exposed through an HTTP interface, typically with a REST API. All client data access is done at the user level: the operating system is unaware of the presence of the remote storage system. In OpenStack, the Object Storage service provides this type of functionality. Users access and modify files by making HTTP requests. Because the data accesss interface provided by an object storage system is at a low level of abstraction, people often build on top of object storage to build file-based applications that provide a higher level of abstraction. For example, the OpenStack Image service can be configured to use the Object Storage service as a backend. Another use for object storage solutions is as a content delivery network (CDN) for hosting static web content (e.g., images, and media files), since object storage already provides an HTTP interface.

Block storage (SAN)

With block storage, files are exposed through a low-level computer bus interface such as SCSI or ATA, that is accessible over the network. Block storage is synonymous with SAN (storage area network). Clients access data through the operating system at the device level: users access the data by mounting the remote device in a similar manner to how they would mount a local, physical disk (e.g., using the "mount" command in Linux). In OpenStack, the nova-volume service that forms part of the Compute service provides this type of functionality, and uses iSCSI to expose remote data as a SCSI disk that is attached to the network.

Because the data is exposed as a physical device, the end-user is responsible for creating partitions and formatting the exposed disk device. In addition, in OpenStack Compute a device can only be attached to one server at a time, so block storage cannot be used for to share data across virtual machine instances concurrently.

Installation Assumptions

OpenStack Compute has a large number of configuration options. To simplify this installation guide, we make a number of assumptions about the target installation:

- You have a collection of compute nodes, each installed with Ubuntu Server 12.04
- You have designated one of the nodes as the Cloud Controller, which will run all of the services (RabbitMQ, MySQL, Identity, Image, nova-api, nova-network, nova-scheduler, nova-volume) except for nova-compute.
- The disk partitions on your cloud controller are being managed by the Logical Volume Manager (LVM)
- Your Cloud Controller has an LVM volume group named "nova-volumes" to provide persistent storage to guest VMs. Either create this during the installation or leave some free space to create it prior to installing nova services.
- Ensure that the server can resolve its own hostname, otherwise you may have problems if you are using RabbitMQ as the messaging backend (RabbitMQ is the default messaging back-end on Ubuntu, on Fedora the default backend is Qpid).
- 192.168.69.1 is the primary IP for our controller host on eth0 and 192.168.69.2 is the primary IP for our computer node on eth0.
- FlatDHCP with a single network interface.
- KVM as the hypervisor.
- Ensure the operating system is up-to-date by running apt-get update and

apt-get upgrade prior to the installation.

- Ensure that all hosts are with ntp and correct time.

This installation process walks through installing a cloud controller node and a compute node using a set of packages that are known to work with each other. The cloud controller node contains all the nova- services including the API server and the database server. The compute node needs to run only the nova-compute service. You only need one nova-network service running in a multi-node install.

Keystone

The OpenStack Identity service manages users, tenants (accounts or projects) and offers a common identity system for all the OpenStack components.

The Identity services has two primary functions:

1. User management: keep track of users and what they are permitted to do
2. Service catalog: Provide a catalog of what services are available and where their API endpoints are located.

User management

The three main concepts of Identity user management are:

- Users
- Tenants
- Roles

A user represents a human user, and has associated information such as username, password and email.

A tenant can be thought of as a project, group, or organization. Whenever you make requests to OpenStack services, you must specify a tenant. For example, if you query the Compute service for a list of running instances, you will receive a list of all of the running instances in the tenant you specified in your query.

A role captures what operations a user is permitted to perform in a given tenant.

Service management

The two main concepts of Identity service management are:

- Services
- Endpoints

The Identity service also maintains a user that corresponds to each service (e.g., a user named nova, for the Compute service) and a special service tenant, which is called service.

Installing Keystone

```
# apt-get install keystone
```

Delete the keystone.db file created in the /var/lib/keystone directory.

```
# rm /var/lib/keystone/keystone.db
```

Install mysql:

```
# apt-get install python-mysqldb mysql-server
```

Use sed to edit /etc/mysql/my.cnf to change bind-address from localhost (127.0.0.1) to any (0.0.0.0) and restart the mysql service, as root:

```
# sed -i 's/127.0.0.1/0.0.0.0/g' /etc/mysql/my.cnf
# service mysql restart
```

To configure the MySQL database, create the keystone database.

```
# mysql -u root -p
mysql> CREATE DATABASE keystone;
mysql> CREATE USER keystone;
mysql> GRANT ALL PRIVILEGES ON keystone.* TO 'keystone'@'%' IDENTIFIED BY 'keystonepass';
mysql> FLUSH PRIVILEGES;
mysql> quit;
```

Edit ***keystone.conf*** to use mysql:

```
[sql]
connection = mysql://keystone:keystonepass@192.168.69.1/keystone
```

Change in ***keystone.conf*** the ADMIN_TOKEN:

```
admin_token = 012345SECRET99TOKEN012345
```

Restart the keystone service and populate with initial data:

```
# service keystone restart
# keystone-manage db_sync
```

Configuring Services to work with Keystone

Create the file */**root**/**openstackrc***, with theses lines

```
export SERVICE_ENDPOINT=http://192.168.69.1:35357/v2.0/
export SERVICE_TOKEN=012345SECRET99TOKEN012345
```

Theses environment variables are useful to avoid execute commands with long args, like --token or --endpoint.

Reload the file:

```
# source /root/openstackrc
```

Create the **tenants**: *openstackDemo* and *service*:

```
# keystone tenant-create --name openstackDemo --description "Default Tenant"  --enabled true
# keystone tenant-create --name service --description "Service Tenant"  --enabled true
```

Create the **user***admin*, replace the tenant_id by openStackDemo tenant_id:

```
# keystone user-create --tenant_id 9c401c9d2b9d41bcabda6919d4aec63a --name admin --pass adminpass --enabled true
```

Create the **roles**: *admin* and *user*:

```
# keystone role-create --name admin
# keystone role-create --name user
```

Grant *admin***role** to **user***admin*:

```
# keystone user-role-add --user [admin_user_id] --tenant_id [openstackDemo_tenant_id] --role [admin_role]
```

Create an user to each service, replace the tenant_id by the service tenant id:

```
# keystone user-create --tenant_id 92920816c7704a8a95c916d34dc7a05c --name glance --pass glancepass --enabled true
# keystone user-create --tenant_id 92920816c7704a8a95c916d34dc7a05c --name nova --pass novapass --enabled true
# keystone user-create --tenant_id 92920816c7704a8a95c916d34dc7a05c --name swift --pass swiftpass --enabled true
```

Add the users of each service to admin **role**:

```
# keystone user-role-add --user [glance_user_id] --tenant_id [service_tenant_id] --role [admin_role]
# keystone user-role-add --user [nova_user_id] --tenant_id [service_tenant_id] --role [admin_role]
```

```
# keystone user-role-add --user [swift_user_id] --tenant_id [service_tenant_id] --role [admin_role]
```

Creating the services:

Until now, we just created: tenants, users and roles. The Keystone manage the services and yours endpoints too. In this way, the OpenStack components query to keystone where each service is located:

Run the follow commands, to create the services:

```
# keystone service-create --name keystone --type identity --description "Keystone Identity Service"
# keystone service-create --name nova --type compute --description "Nova Compute Service"
# keystone service-create --name volume --type volume --description "Nova Volume Service"
# keystone service-create --name glance --type image --description "Glance Image Service"
# keystone service-create --name swift --type object-store --description "Object Storage Service"
```

Run the follow commands, to create the endpoints of each service:

```
# keystone endpoint-create --region RegionOne --service_id [keystone_service_id] --publicurl http://192.168.69.1:5000/v2.0 --internalurl
http://192.168.69.1:5000/v2.0 --adminurl http://192.168.69.1:35357/v2.0
# keystone endpoint-create --region RegionOne --service_id [compute_service_id] --publicurl 'http://192.168.69.1:8774/v2/%(tenant_id)s'
--internalurl 'http://192.168.69.1:8774/v2/%(tenant_id)s' --adminurl 'http://192.168.69.1:8774/v2/%(tenant_id)s'
# keystone endpoint-create --region RegionOne --service_id [volume_service_id] --publicurl 'http://192.168.69.1:8776/v1/%(tenant_id)s'
--internalurl 'http://192.168.69.1:8776/v1/%(tenant_id)s' --adminurl 'http://192.168.69.1:8776/v1/%(tenant_id)s'
# keystone endpoint-create --region RegionOne --service_id [glance_service_id] --publicurl http://192.168.69.1:9292/v1 --internalurl
http://192.168.69.1:9292/v1 --adminurl http://192.168.69.1:9292/v1
# keystone endpoint-create --region RegionOne --service_id [swift_service_id] --publicurl 'http://192.168.69.1:8888/v1/AUTH_%(tenant_id)s'
--adminurl 'http://192.168.69.1:8888/' --internalurl 'http://192.168.69.1:8888/v1/AUTH_%(tenant_id)s'
```

Checking the keystone installation:w

```
# sudo apt-get install curl openssl
```

Do a REST request to keystone-api using curl:

```
# curl -d '{"auth": {"tenantName": "openstackDemo", "passwordCredentials": {"username": "admin", "password": "adminpass"}}}' -H "Content-type:
application/json" http://192.168.69.1:35357/v2.0/tokens  | python -mjson.tool
```

The output should have the services, endpoints and a valid token for 24 hours:


```
{
   "access": {
      "serviceCatalog": [
         {
            "endpoints": [
               {
                  "adminURL": "http://192.168.69.1:8776/v1/9c401c9d2b9d41bcabda6919d4aec63a",
                  "internalURL": "http://192.168.69.1:8776/v1/9c401c9d2b9d41bcabda6919d4aec63a",
                  "publicURL": "http://192.168.69.1:8776/v1/9c401c9d2b9d41bcabda6919d4aec63a",
                  "region": "RegionOne"
               }
            ],
            "endpoints_links": [],
            "name": "volume",
            "type": "volume"
         },
         {
            "endpoints": [
               {
                  "adminURL": "http://192.168.69.1:9292/v1",
                  "internalURL": "http://192.168.69.1:9292/v1",
                  "publicURL": "http://192.168.69.1:9292/v1",
                  "region": "RegionOne"
               }
            ],
            "endpoints_links": [],
            "name": "glance",
            "type": "image"
         },
         {
```

```
            "endpoints": [
               {
                  "adminURL": "http://192.168.69.1:8774/v2/9c401c9d2b9d41bcabda6919d4aec63a",
                  "internalURL": "http://192.168.69.1:8774/v2/9c401c9d2b9d41bcabda6919d4aec63a",
                  "publicURL": "http://192.168.69.1:8774/v2/9c401c9d2b9d41bcabda6919d4aec63a",
                  "region": "RegionOne"
               }
            ],
            "endpoints_links": [],
            "name": "nova",
            "type": "compute"
         },
         {
            "endpoints": [
               {
                  "adminURL": "http://192.168.69.1:35357/v2.0",
                  "internalURL": "http://192.168.69.1:5000/v2.0",
                  "publicURL": "http://192.168.69.1:5000/v2.0",
                  "region": "RegionOne"
               }
            ],
            "endpoints_links": [],
            "name": "keystone",
            "type": "identity"
         },
         {
            "endpoints": [
               {
                  "adminURL": "http://192.168.69.1:8888/",
                  "internalURL": "http://192.168.69.1:8888/v1/AUTH_9c401c9d2b9d41bcabda6919d4aec63a",
                  "publicURL": "http://192.168.69.1:8888/v1/AUTH_9c401c9d2b9d41bcabda6919d4aec63a",
                  "region": "RegionOne"
               }
            ],
            "endpoints_links": [],
            "name": "swift",
            "type": "object-store"
         }
      ],
      "token": {
         "expires": "2012-06-08T15:31:46Z",
         "id": "05b818ff23564d5599c97ea5e1df7bbe",
         "tenant": {
            "description": "Default Tenant",
            "enabled": true,
            "id": "9c401c9d2b9d41bcabda6919d4aec63a",
            "name": "openstackDemo"
         }
      },
      "user": {
         "id": "1fcfbe778fc3428aa30db782056b34d7",
         "name": "admin",
         "roles": [
            {
               "id": "a2d6da77c0a54105aab94484e0d2088b",
               "name": "admin"
            }
         ],
         "roles_links": [],
         "username": "admin"
      }
   }
}
```


You can list users, roles, tenants and services:

```
# keystone user-list
+----------------------------------+---------+-------+--------+
|                id                | enabled | email |  name  |
+----------------------------------+---------+-------+--------+
| 029c5749548946f7b560b2706d31d488 | True    | None  | glance |
| 1fcfbe778fc3428aa30db782056b34d7 | True    | None  | admin  |
| 4b75079aeb1b408ebdb286bfb85e719f | True    | None  | swift  |
```

```
| e25f81d50eb84bdb92907adbd0e41f25 | True    | None  | nova   |
+----------------------------------+---------+-------+--------+
```

```
# keystone role-list
+----------------------------------+-------+
|                id                |  name |
+----------------------------------+-------+
| 845e83e4d9bf4d8d8188b1e261dfe103 | user  |
| a2d6da77c0a54105aab94484e0d2088b | admin |
+----------------------------------+-------+
```

```
# keystone tenant-list
+----------------------------------+--------------+---------+
|                id                |     name     | enabled |
+----------------------------------+--------------+---------+
| 92920816c7704a8a95c916d34dc7a05c | service      | True    |
| 9c401c9d2b9d41bcabda6919d4aec63a | openstackDemo | True    |
+----------------------------------+--------------+---------+
```

```
# keystone service-list
+----------------------------------+----------+---------------+---------------------------+
|                id                |   name   |      type     |        description        |
+----------------------------------+----------+---------------+---------------------------+
| 00486163c7c64c6a86f577627071aafc | nova     | compute       | Nova Compute Service      |
| 69d3a6b21c9348acb1042d7d9386ff81 | volume   | volume        | Nova Volume Service       |
| 7b09782f9faf448781f5a460acabe258 | swift    | object-store  | Object Storage Service    |
| d6b32583a6694e9b828b2a50037623ba | keystone | identity      | Keystone Identity Service |
| f7f964a71e2a45119dcaede68fafbd78 | glance   | image         | Glance Image Service      |
+----------------------------------+----------+---------------+---------------------------+
```

Installing OpenStack Object Storage - Swift

OpenStack Object Storage (Swift) is a system to store objects in a massively scalable large capacity system with built-in redundancy and failover. Object Storage has a variety of applications, such as backing up or archiving data, serving graphics or videos (streaming data to a user's browser), storing secondary or tertiary static data, developing new applications with data storage integration, storing data when predicting storage capacity is difficult, and creating the elasticity and flexibility of cloud-based storage for your web applications.

In this environment we use the same controller node as storage too. So the swift and swift-proxy is installed in the controller-node.

Install the swift and dependencies:

```
# apt-get install swift python-xattr python-memcache rsync memcached python-netifaces
```

Unfortunately at the time of writing this manual swift package not automates some tasks, such as the creation of configuration files. We have to do this manually:

```
# mkdir /etc/swift
```

Create the config file */etc/swift/swift.conf*:

```
[swift-hash]
# random unique string that can never change (DO NOT LOSE)
swift_hash_path_suffix = fLIbertYgibbitZ
```

Install and configure the "object-nodes"

In this case, the object-nodes (computer when data are stored) is the same cloud-controller.

```
# apt-get install swift-account swift-container swift-object xfsprogs
```

Each "Object node" can have many devices Is a requisite use xfs file system.

```
# fdisk /dev/sdb (set up a single partition)
# mkfs.xfs -i size=1024 /dev/sdb1
```

This device should be mount at /srv/node/

```
echo "/dev/sdb1 /srv/node/sdb1 xfs noatime,nodiratime,nobarrier,logbufs=8 0 0" >> /etc/fstab
mkdir -p /srv/node/sdb1
mount /srv/node/sdb1
chown -R swift:swift /srv/node
```

Configure the *rsyncd.conf*

```
uid = swift
gid = swift
log file = /var/log/rsyncd.log
pid file = /var/run/rsyncd.pid
address = 192.168.69.1

[account]
max connections = 2
path = /srv/node/
read only = false
lock file = /var/lock/account.lock

[container]
max connections = 2
path = /srv/node/
read only = false
lock file = /var/lock/container.lock

[object]
max connections = 2
path = /srv/node/
read only = false
lock file = /var/lock/object.lock
```

Edit the */etc/default/rsync*:

```
RSYNC_ENABLE = true
```

Restart the rsync:

```
service rsync restart
```

Create the */etc/swift/account-server.conf*:

```
[DEFAULT]
bind_ip = 192.168.69.1
workers = 2

[pipeline:main]
pipeline = account-server

[app:account-server]
use = egg:swift#account

[account-replicator]

[account-auditor]

[account-reaper]
```

Create the */etc/swift/container-server.conf*:

```
[DEFAULT]
bind_ip = 192.168.69.1
workers = 2

[pipeline:main]
```

```
pipeline = container-server

[app:container-server]
use = egg:swift#container

[container-replicator]

[container-updater]

[container-auditor]
```

Create the */etc/swift/object-server.conf*:

```
[DEFAULT]
bind_ip = 192.168.69.1
workers = 2

[pipeline:main]
pipeline = object-server

[app:object-server]
use = egg:swift#object

[object-replicator]

[object-updater]

[object-auditor]
```

Installing and configuring the swift-proxy

The swift-proxy is the component that acts like a proxy, so you can have multiple object-nodes and one swift-proxy. When an user or openStack component need get a object, its ask for swift-proxy.

Install swift-proxy:

```
# apt-get install swift-proxy memcached
```

Create a self-sign SSL certificate:

```
cd /etc/swift
openssl req -new -x509 -nodes -out cert.crt -keyout cert.key
```

Change the memcached to listen in correct interface, change the /etc/memcached.conf:

```
-l 127.0.0.1
to
-l 192.168.69.1
```

Restart the memcached:

```
service memcached restart
```

Create the file */etc/swift/proxy-server.conf*:

```
[DEFAULT]
bind_port = 8888
user = swift

[pipeline:main]
pipeline = catch_errors healthcheck cache authtoken keystone proxy-server

[app:proxy-server]
use = egg:swift#proxy
account_autocreate = true

[filter:keystone]
paste.filter_factory = keystone.middleware.swift_auth:filter_factory
operator_roles = admin, swiftoperator
```

```
[filter:authtoken]
paste.filter_factory = keystone.middleware.auth_token:filter_factory
# Delaying the auth decision is required to support token-less
# usage for anonymous referrers ('.r:*').
delay_auth_decision = 10
service_port = 5000
service_host = 192.168.69.1
auth_port = 35357
auth_host = 192.168.69.1
auth_token = 012345SECRET99TOKEN012345
admin_token = 012345SECRET99TOKEN012345
auth_protocol = http

[filter:cache]
use = egg:swift#memcache
set log_name = cache

[filter:catch_errors]
use = egg:swift#catch_errors

[filter:healthcheck]
use = egg:swift#healthcheck
```

Create the account, container and object rings. The builder command is basically creating a builder file with a few parameters. The parameter with the value of 18 represents 2 ^ 18th, the value that the partition will be sized to. Set this "partition power" value based on the total amount of storage you expect your entire ring to use. The value of 1 represents the number of replicas of each object, with the last value being the number of hours to restrict moving a partition more than once.

```
cd /etc/swift
swift-ring-builder account.builder create 18 1 1
swift-ring-builder container.builder create 18 1 1
swift-ring-builder object.builder create 18 1 1
```

For every storage device on each node add entries to each ring:

```
swift-ring-builder account.builder add z<ZONE>-<STORAGE_LOCAL_NET_IP>:6002/<DEVICE> 100
swift-ring-builder container.builder add z<ZONE>-<STORAGE_LOCAL_NET_IP_1>:6001/<DEVICE> 100
swift-ring-builder object.builder add z<ZONE>-<STORAGE_LOCAL_NET_IP_1>:6000/<DEVICE> 100
```

In our case, we have only one zone in one storage node with one device (sdb1), so the command should be:

```
swift-ring-builder account.builder add z1-192.168.69.1:6002/sdb1 100
swift-ring-builder container.builder add z1-192.168.69.1:6001/sdb1 100
swift-ring-builder object.builder add z1-192.168.69.1:6000/sdb1 100
```

Verify the ring contents for each ring:

```
swift-ring-builder account.builder
swift-ring-builder container.builder
swift-ring-builder object.builder
```

Rebalance the rings:

```
swift-ring-builder account.builder rebalance
swift-ring-builder container.builder rebalance
swift-ring-builder object.builder rebalance
```

Change the permissions of config files:

```
# chown -R swift:swift /etc/swift/
```

Start the swift-proxy:

```
swift-init proxy start
```

Start all swift services:

```
swift-init main start
swift-init rest start
```

Checking the installation

The file */root/openstackrc* will receive more environment variables:

```
# Used for Keystone
export SERVICE_ENDPOINT=http://192.168.69.1:35357/v2.0/
export SERVICE_TOKEN=012345SECRET99TOKEN012345

# Used for swift
export ST_AUTH=http://192.168.69.1:5000/v2.0
export ST_USER=openstackDemo:admin
export ST_KEY=adminpass
export OS_REGION_NAME=RegionOne
```

Reload the file:

```
# source /root/openstackrc
```

Try to get swift status, with the command:

```
# swift -V 2.0 stat
```

List the containers:

```
# swift -V 2.0 list
```

Do some upload files:

```
# swift -V 2.0 upload myfiles bigfile1.tgz
# swift -V 2.0 upload myfiles bigfile2.tgz
```

Download the container "myfiles":

```
# cd /tmp/
# swift -V 2.0 download myfiles
```

Installing OpenStack Image Service - Glance

OpenStack Image Service (Glance) is a lookup and retrieval system for virtual machine images. It can be configured in three ways: using OpenStack Object Store to store images; using Amazon's Simple Storage Solution (S3) storage directly; or using S3 storage with Object Store as the intermediate for S3 access.

Install the glance on the controller node:

```
# apt-get install glance
```

Remove the sqlite file:

```
# rm /var/lib/glance/glance.sqlite
```

Configuring the Image Service database backend

```
# mysql -u root -p
mysql> CREATE DATABASE glance;
mysql> CREATE USER glance;
mysql> GRANT ALL PRIVILEGES ON glance.* TO 'glance'@'%' IDENTIFIED BY 'glancepass';
mysql> FLUSH PRIVILEGES;
mysql> quit;
```

Copy the config files to */etc/glance/*:

/etc/glance/glance-api.conf:

```
[DEFAULT]
# Show more verbose log output (sets INFO log level output)
verbose = True

# Show debugging output in logs (sets DEBUG log level output)
debug = False

# Which backend store should Glance use by default is not specified
# in a request to add a new image to Glance? Default: 'file'
# Available choices are 'file', 'swift', and 's3'
default_store = swift

# Address to bind the API server
bind_host = 0.0.0.0

# Port the bind the API server to
bind_port = 9292

# Log to this file. Make sure you do not set the same log
# file for both the API and registry servers!
log_file = /var/log/glance/api.log

# Backlog requests when creating socket
backlog = 4096

# Number of Glance API worker processes to start.
# On machines with more than one CPU increasing this value
# may improve performance (especially if using SSL with
# compression turned on). It is typically recommended to set
# this value to the number of CPUs present on your machine.
workers = 0

# Role used to identify an authenticated user as administrator
#admin_role = admin

# ================= Syslog Options ============================

# Send logs to syslog (/dev/log) instead of to file specified
# by `log_file`
use_syslog = False

# Facility to use. If unset defaults to LOG_USER.
# syslog_log_facility = LOG_LOCAL0

# ================= SSL Options ==============================

# Certificate file to use when starting API server securely
# cert_file = /path/to/certfile
```

```
# Private key file to use when starting API server securely
# key_file = /path/to/keyfile

# ================= Security Options ==========================

# AES key for encrypting store 'location' metadata, including
# -- if used -- Swift or S3 credentials
# Should be set to a random string of length 16, 24 or 32 bytes
# metadata_encryption_key = <16, 24 or 32 char registry metadata key>

# ============ Registry Options ===============================

# Address to find the registry server
registry_host = 0.0.0.0

# Port the registry server is listening on
registry_port = 9191

# What protocol to use when connecting to the registry server?
# Set to https for secure HTTP communication
registry_client_protocol = http

# The path to the key file to use in SSL connections to the
# registry server, if any. Alternately, you may set the
# GLANCE_CLIENT_KEY_FILE environ variable to a filepath of the key file
# registry_client_key_file = /path/to/key/file

# The path to the cert file to use in SSL connections to the
# registry server, if any. Alternately, you may set the
# GLANCE_CLIENT_CERT_FILE environ variable to a filepath of the cert file
# registry_client_cert_file = /path/to/cert/file

# The path to the certifying authority cert file to use in SSL connections
# to the registry server, if any. Alternately, you may set the
# GLANCE_CLIENT_CA_FILE environ variable to a filepath of the CA cert file
# registry_client_ca_file = /path/to/ca/file

# ============ Notification System Options =====================

# Notifications can be sent when images are create, updated or deleted.
# There are three methods of sending notifications, logging (via the
# log_file directive), rabbit (via a rabbitmq queue), qpid (via a Qpid
# message queue), or noop (no notifications sent, the default)
notifier_strategy = noop

# Configuration options if sending notifications via rabbitmq (these are
# the defaults)
rabbit_host = localhost
rabbit_port = 5672
rabbit_use_ssl = false
rabbit_userid = guest
rabbit_password = guest
rabbit_virtual_host = /
```

```
rabbit_notification_exchange = glance
rabbit_notification_topic = glance_notifications

# Configuration options if sending notifications via Qpid (these are
# the defaults)
qpid_notification_exchange = glance
qpid_notification_topic = glance_notifications
qpid_host = localhost
qpid_port = 5672
qpid_username =
qpid_password =
qpid_sasl_mechanisms =
qpid_reconnect_timeout = 0
qpid_reconnect_limit = 0
qpid_reconnect_interval_min = 0
qpid_reconnect_interval_max = 0
qpid_reconnect_interval = 0
qpid_heartbeat = 5
# Set to 'ssl' to enable SSL
qpid_protocol = tcp
qpid_tcp_nodelay = True

# ============ Filesystem Store Options ========================

# Directory that the Filesystem backend store
# writes image data to
filesystem_store_datadir = /var/lib/glance/images/

# ============ Swift Store Options =============================

# Version of the authentication service to use
# Valid versions are '2' for keystone and '1' for swauth and rackspace
swift_store_auth_version = 2

# Address where the Swift authentication service lives
# Valid schemes are 'http://' and 'https://'
# If no scheme specified,  default to 'https://'
# For swauth, use something like '127.0.0.1:8080/v1.0/'
swift_store_auth_address = http://192.168.69.1:35357/v2.0/

# User to authenticate against the Swift authentication service
# If you use Swift authentication service, set it to 'account':'user'
# where 'account' is a Swift storage account and 'user'
# is a user in that account
swift_store_user = service:glance

# Auth key for the user authenticating against the
# Swift authentication service
swift_store_key = glancepass

# Container within the account that the account should use
# for storing images in Swift
swift_store_container = glance
```

```
# Do we create the container if it does not exist?
swift_store_create_container_on_put = True

# What size, in MB, should Glance start chunking image files
# and do a large object manifest in Swift? By default, this is
# the maximum object size in Swift, which is 5GB
swift_store_large_object_size = 5120

# When doing a large object manifest, what size, in MB, should
# Glance write chunks to Swift? This amount of data is written
# to a temporary disk buffer during the process of chunking
# the image file, and the default is 200MB
swift_store_large_object_chunk_size = 200

# Whether to use ServiceNET to communicate with the Swift storage servers.
# (If you aren't RACKSPACE, leave this False!)
#
# To use ServiceNET for authentication, prefix hostname of
# `swift_store_auth_address` with 'snet-'.
# Ex. https://example.com/v1.0/ -> https://snet-example.com/v1.0/
swift_enable_snet = False

# ============ S3 Store Options =============================

# Address where the S3 authentication service lives
# Valid schemes are 'http://' and 'https://'
# If no scheme specified,  default to 'http://'
s3_store_host = 127.0.0.1:8080/v1.0/

# User to authenticate against the S3 authentication service
s3_store_access_key = <20-char AWS access key>

# Auth key for the user authenticating against the
# S3 authentication service
s3_store_secret_key = <40-char AWS secret key>

# Container within the account that the account should use
# for storing images in S3. Note that S3 has a flat namespace,
# so you need a unique bucket name for your glance images. An
# easy way to do this is append your AWS access key to "glance".
# S3 buckets in AWS *must* be lowercased, so remember to lowercase
# your AWS access key if you use it in your bucket name below!
s3_store_bucket = <lowercased 20-char aws access key>glance

# Do we create the bucket if it does not exist?
s3_store_create_bucket_on_put = False

# When sending images to S3, the data will first be written to a
# temporary buffer on disk. By default the platform's temporary directory
# will be used. If required, an alternative directory can be specified here.
# s3_store_object_buffer_dir = /path/to/dir
```

```
# ============ RBD Store Options =============================

# Ceph configuration file path
# If using cephx authentication, this file should
# include a reference to the right keyring
# in a client.<USER> section
rbd_store_ceph_conf = /etc/ceph/ceph.conf

# RADOS user to authenticate as (only applicable if using cephx)
rbd_store_user = glance

# RADOS pool in which images are stored
rbd_store_pool = images

# Images will be chunked into objects of this size (in megabytes).
# For best performance, this should be a power of two
rbd_store_chunk_size = 8

# ============ Delayed Delete Options =============================

# Turn on/off delayed delete
delayed_delete = False

# Delayed delete time in seconds
scrub_time = 43200

# Directory that the scrubber will use to remind itself of what to delete
# Make sure this is also set in glance-scrubber.conf
scrubber_datadir = /var/lib/glance/scrubber

# =============== Image Cache Options =============================

# Base directory that the Image Cache uses
image_cache_dir = /var/lib/glance/image-cache/

[paste_deploy]
flavor = keystone
```

/etc/glance/glance-api-paste.ini:

```
# Default minimal pipeline
[pipeline:glance-api]
pipeline = versionnegotiation context apiv1app authtoken

# Use the following pipeline for keystone auth
# i.e. in glance-api.conf:
#   [paste_deploy]
#   flavor = keystone
#
[pipeline:glance-api-keystone]
pipeline = versionnegotiation authtoken context apiv1app
```

```
# Use the following pipeline to enable transparent caching of image files
# i.e. in glance-api.conf:
#   [paste_deploy]
#   flavor = caching
#
[pipeline:glance-api-caching]
pipeline = versionnegotiation context cache apiv1app

# Use the following pipeline for keystone auth with caching
# i.e. in glance-api.conf:
#   [paste_deploy]
#   flavor = keystone+caching
#
[pipeline:glance-api-keystone+caching]
pipeline = versionnegotiation authtoken context cache apiv1app

# Use the following pipeline to enable the Image Cache Management API
# i.e. in glance-api.conf:
#   [paste_deploy]
#   flavor = cachemanagement
#
[pipeline:glance-api-cachemanagement]
pipeline = versionnegotiation context cache cachemanage apiv1app

# Use the following pipeline for keystone auth with cache management
# i.e. in glance-api.conf:
#   [paste_deploy]
#   flavor = keystone+cachemanagement
#
[pipeline:glance-api-keystone+cachemanagement]
pipeline = versionnegotiation authtoken context cache cachemanage apiv1app

[app:apiv1app]
paste.app_factory = glance.common.wsgi:app_factory
glance.app_factory = glance.api.v1.router:API

[filter:versionnegotiation]
paste.filter_factory = glance.common.wsgi:filter_factory
glance.filter_factory = glance.api.middleware.version_negotiation:VersionNegotiationFilter

[filter:cache]
paste.filter_factory = glance.common.wsgi:filter_factory
glance.filter_factory = glance.api.middleware.cache:CacheFilter

[filter:cachemanage]
paste.filter_factory = glance.common.wsgi:filter_factory
glance.filter_factory = glance.api.middleware.cache_manage:CacheManageFilter

[filter:context]
paste.filter_factory = glance.common.wsgi:filter_factory
glance.filter_factory = glance.common.context:ContextMiddleware
```

```
[filter:authtoken]
paste.filter_factory = keystone.middleware.auth_token:filter_factory
service_protocol = http
service_host = 192.168.69.1
service_port = 5000
auth_host = 192.168.69.1
auth_port = 35357
auth_protocol = http
auth_uri = http://192.168.69.1:5000/
admin_tenant_name = service
admin_user = glance
admin_password = glancepass
```

/etc/glance/glance-registry.conf:

```
[DEFAULT]
# Show more verbose log output (sets INFO log level output)
verbose = True

# Show debugging output in logs (sets DEBUG log level output)
debug = True

# Address to bind the registry server
bind_host = 0.0.0.0

# Port the bind the registry server to
bind_port = 9191

# Log to this file. Make sure you do not set the same log
# file for both the API and registry servers!
log_file = /var/log/glance/registry.log

# Backlog requests when creating socket
backlog = 4096

# SQLAlchemy connection string for the reference implementation
# registry server. Any valid SQLAlchemy connection string is fine.
# See: http://www.sqlalchemy.org/docs/05/reference/sqlalchemy/connections.html#sqlalchemy.create_engine
sql_connection = mysql://glance:glancepass@192.168.69.1/glance

# Period in seconds after which SQLAlchemy should reestablish its connection
# to the database.
#
# MySQL uses a default `wait_timeout` of 8 hours, after which it will drop
# idle connections. This can result in 'MySQL Gone Away' exceptions. If you
# notice this, you can lower this value to ensure that SQLAlchemy reconnects
# before MySQL can drop the connection.
sql_idle_timeout = 3600

# Limit the api to return `param_limit_max` items in a call to a container. If
# a larger `limit` query param is provided, it will be reduced to this value.
api_limit_max = 1000
```

```
# If a `limit` query param is not provided in an api request, it will
# default to `limit_param_default`
limit_param_default = 25

# Role used to identify an authenticated user as administrator
#admin_role = admin

# ================= Syslog Options ============================

# Send logs to syslog (/dev/log) instead of to file specified
# by `log_file`
use_syslog = True

# Facility to use. If unset defaults to LOG_USER.
# syslog_log_facility = LOG_LOCAL1

# ================= SSL Options ===============================

# Certificate file to use when starting registry server securely
# cert_file = /path/to/certfile

# Private key file to use when starting registry server securely
# key_file = /path/to/keyfile

[paste_deploy]
flavor = keystone
```

/etc/glance/glance-registry-paste.ini:

```
# Default minimal pipeline
[pipeline:glance-registry]
pipeline = context registryapp authtoken

# Use the following pipeline for keystone auth
# i.e. in glance-registry.conf:
#   [paste_deploy]
#   flavor = keystone
#
[pipeline:glance-registry-keystone]
pipeline = authtoken context registryapp

[app:registryapp]
paste.app_factory = glance.common.wsgi:app_factory
glance.app_factory = glance.registry.api.v1:API

[filter:context]
context_class = glance.registry.context.RequestContext
paste.filter_factory = glance.common.wsgi:filter_factory
glance.filter_factory = glance.common.context:ContextMiddleware

[filter:authtoken]
```

```
paste.filter_factory = keystone.middleware.auth_token:filter_factory
service_protocol = http
service_host = 192.168.69.1
service_port = 5000
auth_host = 192.168.69.1
auth_port = 35357
auth_protocol = http
auth_uri = http://192.168.69.1:5000/
admin_tenant_name = service
admin_user = glance
admin_password = glancepass
```

/etc/glance/glance-scrubber.conf:

```
[DEFAULT]
# Show more verbose log output (sets INFO log level output)
verbose = True

# Show debugging output in logs (sets DEBUG log level output)
debug = False

# Log to this file. Make sure you do not set the same log
# file for both the API and registry servers!
log_file = /var/log/glance/scrubber.log

# Send logs to syslog (/dev/log) instead of to file specified by `log_file`
use_syslog = False

# Should we run our own loop or rely on cron/scheduler to run us
daemon = False

# Loop time between checking for new items to schedule for delete
wakeup_time = 300

sql_connection = mysql://glance:glancepass@192.168.69.1/glance

# Directory that the scrubber will use to remind itself of what to delete
# Make sure this is also set in glance-api.conf
scrubber_datadir = /var/lib/glance/scrubber

# Only one server in your deployment should be designated the cleanup host
cleanup_scrubber = False

# pending_delete items older than this time are candidates for cleanup
cleanup_scrubber_time = 86400

# Address to find the registry server for cleanups
registry_host = 0.0.0.0

# Port the registry server is listening on
registry_port = 9191
```

Synchronizing the database and restarting the services

```
# glance-manage version_control 0
# glance-manage db_sync
# service glance-registry restart
# service glance-api restart
```

Testing the Glance installation:

Put the new env vars to */root/openstackrc*:

```
# Used for Keystone
export SERVICE_ENDPOINT=http://192.168.69.1:35357/v2.0/
export SERVICE_TOKEN=012345SECRET99TOKEN012345

# Used for swift
export ST_AUTH=http://192.168.69.1:5000/v2.0
export ST_USER=openstackDemo:admin
export ST_KEY=adminpass
export OS_REGION_NAME=RegionOne

# Used for glance
export OS_USERNAME=admin
export OS_TENANT_NAME=openstackDemo
export OS_PASSWORD=adminpass
export OS_AUTH_URL=http://192.168.69.1:5000/v2.0/
export OS_REGION_NAME=RegionOne
```

Download a test image:

```
mkdir /tmp/images
cd /tmp/images/
wget http://smoser.brickies.net/ubuntu/ttylinux-uec/ttylinux-uec-amd64-12.1_2.6.35-22_1.tar.gz
tar -zxvf ttylinux-uec-amd64-12.1_2.6.35-22_1.tar.gz
```

Upload the image:

```
# cd /tmp/images/
# glance add name="tty-linux-kernel" disk_format=aki container_format=aki < ttylinux-uec-amd64-12.1_2.6.35-22_1-vmlinuz
# glance add name="tty-linux-ramdisk" disk_format=ari container_format=ari < ttylinux-uec-amd64-12.1_2.6.35-22_1-loader
# glance add name="tty-linux" disk_format=ami container_format=ami kernel_id=[kernel_image_id] ramdisk_id=[kernel_ramdisk_image_id] <
ttylinux-uec-amd64-12.1_2.6.35-22_1.img
```

List the images:

```
# glance index
```

OpenStack Compute service - nova

Nova is the project name for OpenStack Compute, a cloud computing fabric controller, the main part of an IaaS system. Individuals and organizations can use Nova to host and manage their own cloud computing systems. Nova originated as a project out of NASA Ames Research Laboratory.

OpenStack Compute gives you a tool to orchestrate a cloud, including running instances, managing networks, and controlling access to the cloud through users and projects. The underlying open source project's name is Nova, and it provides the software that can control an Infrastructure as a Service (IaaS) cloud computing platform. It is similar in scope to Amazon EC2 and Rackspace Cloud Servers. OpenStack Compute does not include any virtualization software; rather it defines drivers that interact with underlying virtualization mechanisms that run on your host operating system, and exposes functionality over a web-based API.

AMQP and Nova

AMQP is the messaging technology chosen by the OpenStack cloud. The AMQP broker, either RabbitMQ or Qpid, sits between any two Nova components and allows them to communicate in a loosely coupled fashion. More precisely, Nova components (the compute fabric of OpenStack) use Remote Procedure Calls (RPC hereinafter) to communicate to one another.

For more information about AMQP in Nova, please read the AMQP section in nova manual

Hypervisor

For production environments the most tested hypervisors are KVM and Xen-based hypervisors. KVM runs through libvirt, Xen runs best through XenAPI calls. KVM is selected by default and requires the least additional configuration. This guide offers information on specific walkthrough configuration options to KVM.

Configuring the kvm

KVM is configured as the default hypervisor for Compute. To enable KVM explicitly, add the following configuration options /etc/nova/nova.conf:

```
connection_type=libvirt
libvirt_type=kvm
```

The KVM hypervisor supports the following virtual machine image formats:

- Raw
- QEMU Copy-on-write (qcow2)
- VMWare virtual machine disk format (vmdk)

Checking for hardware virtualization support

The processors of your compute host need to support virtualization technology (VT) to use KVM.

If you are running on Ubuntu, use the *kvm-ok* command to check if your processor has VT support, it is enabled in the BIOS, and KVM is installed properly, as root:

```
# kvm-ok
INFO: /dev/kvm exists
KVM acceleration can be used
```

On distributions that don't have kvm-ok, you can check if your processor has VT support by looking at the processor flags in the /proc/cpuinfo file. For Intel processors, look for the vmx flag, and for AMD processors, look for the svm flag. A simple way to check is to run the following command and see if there is any output:

```
$ egrep '(vmx|svm)' --color=always /proc/cpuinfo
```

Enabling KVM

KVM requires the kvm and either kvm-intel or kvm-amd modules to be loaded. This may have been configured automatically on your distribution when KVM is installed. You can check that they have been loaded using lsmod, as follows, with expected output for Intel-based processors:

```
$ lsmod | grep kvm
kvm_intel  137721 9
kvm        415459 1 kvm_intel
```

If your compute host is Intel-based, run the following as root to load the kernel modules:

```
# modprobe kvm
# modprobe kvm-intel
```

Add the following lines to /etc/modules so that these modules will load on reboot:

```
kvm
kvm-intel
```

Configuring the Compute Service database backend

To configure the MySQL database, create the nova database on the cloud controller machine:

```
mysql> CREATE DATABASE nova;
mysql> CREATE USER nova;
mysql> GRANT ALL PRIVILEGES ON nova.* TO 'nova'@'%' IDENTIFIED BY 'novapass';
mysql> quit
```

Installing the Cloud Controller

Install the messaging queue server, RabbitMQ. You also have the option of installing

Apache Qpid, refer to the Compute Administration Manual for instructions.

```
sudo apt-get install rabbitmq-server
```

Install the required nova- packages, and dependencies are automatically installed.

```
sudo apt-get install nova-volume nova-compute nova-vncproxy nova-api  nova-ajax-console-proxy nova-cert nova-consoleauth nova-doc
nova-scheduler nova-network
```

```
nova-manage config convert --infile=/etc/nova/nova.conf --outfile=/etc/nova/nova.conf.new
mv /etc/nova/nova.conf.new /etc/nova/nova.conf
```

h3. Configuring nova
The packages automatically do these steps for a user named nova, but if you are installing as another user you should ensure that the nova.conf file should have its owner set to root:nova, and mode set to 0640, since the file contains your MySQL server's username and password. This packaged install ensures that the nova user belongs to the nova group and that the .conf file permissions are set, but here are the manual commands, which should be run as root:

```
# groupadd nova
# usermod -g nova nova
# chown -R root:nova /etc/nova
# chmod 640 /etc/nova/nova.conf
```

Modify your nova.conf, nova-compute and api-paste.ini according to the following examples:

nova-compute.conf

```
[DEFAULT]
libvirt_type=kvm
```

nova.conf

```
[DEFAULT]
# LOGS/STATE
verbose=True
use_syslog=False
debug=False

#logging_default_format_string =  '%(asctime)s %(levelname)s %(name)s [-] %(message)s'
logging_default_format_string = '%(asctime)s %(levelname)s %(name)s [-] %(message)s'
logging_context_format_string = '%(asctime)s %(levelname)s %(name)s [-] %(message)s'

# AUTHENTICATION
auth_strategy=keystone
logdir=/var/log/nova/
state_path=/var/lib/nova
lock_path=/var/run/nova
bindir=/usr/bin/

force_to_raw=False

# SCHEDULER
compute_scheduler_driver=nova.scheduler.filter_scheduler.FilterScheduler

scheduler_default_filters=AllHostsFilter

# VOLUMES
volume_group=nova-volumes
volume_name_template=volume-%08x
iscsi_helper=tgtadm

# DATABASE
sql_connection=mysql://nova:novapass@192.168.69.1/nova

# COMPUTE
libvirt_type=kvm
connection_type=libvirt
```

```
instance_name_template=instance-%08x
api_paste_config=/etc/nova/api-paste.ini
allow_resize_to_same_host=True

# APIS
osapi_compute_extension=nova.api.openstack.compute.contrib.standard_extensions
ec2_dmz_host=192.168.69.1
s3_host=192.168.69.1
# RABBITMQ
rabbit_host=192.168.69.1
# GLANCE
image_service=nova.image.glance.GlanceImageService
glance_api_servers=192.168.69.1:9292
# NETWORK
network_manager=nova.network.manager.FlatDHCPManager
force_dhcp_release=True
dhcpbridge_flagfile=/etc/nova/nova.conf
firewall_driver=nova.virt.libvirt.firewall.IptablesFirewallDriver
my_ip=192.168.69.1
public_interface=br100
vlan_interface=eth0
flat_network_bridge=br100
flat_interface=eth0
fixed_range=192.168.70.0/24
# NOVNC CONSOLE - You should change this IP to public ip of current nova-compute machine
novncproxy_base_url=http://200.145.46.167:6080/vnc_auto.html
vncserver_proxyclient_address=200.145.46.167
vncserver_listen=200.145.46.167
xvpvncproxy_base_url=http://200.145.46.167:6081/console
```

api-paste.ini

```
############
# Metadata #
############
[composite:metadata]
use = egg:Paste#urlmap
/: metaversions
/latest: meta
/1.0: meta
/2007-01-19: meta
/2007-03-01: meta
/2007-08-29: meta
/2007-10-10: meta
/2007-12-15: meta
/2008-02-01: meta
/2008-09-01: meta
/2009-04-04: meta

[pipeline:metaversions]
pipeline = ec2faultwrap logrequest metaverapp

[pipeline:meta]
pipeline = ec2faultwrap logrequest metaapp

[app:metaverapp]
paste.app_factory = nova.api.metadata.handler:Versions.factory

[app:metaapp]
paste.app_factory = nova.api.metadata.handler:MetadataRequestHandler.factory

#######
# EC2 #
#######

[composite:ec2]
use = egg:Paste#urlmap
/services/Cloud: ec2cloud

[composite:ec2cloud]
use = call:nova.api.auth:pipeline_factory
noauth = ec2faultwrap logrequest ec2noauth cloudrequest validator ec2executor
deprecated = ec2faultwrap logrequest authenticate cloudrequest validator ec2executor
```

```
keystone = ec2faultwrap logrequest ec2keystoneauth cloudrequest validator ec2executor

[filter:ec2faultwrap]
paste.filter_factory = nova.api.ec2:FaultWrapper.factory

[filter:logrequest]
paste.filter_factory = nova.api.ec2:RequestLogging.factory

[filter:ec2lockout]
paste.filter_factory = nova.api.ec2:Lockout.factory

[filter:totoken]
paste.filter_factory = nova.api.ec2:EC2Token.factory

[filter:ec2keystoneauth]
paste.filter_factory = nova.api.ec2:EC2KeystoneAuth.factory

[filter:ec2noauth]
paste.filter_factory = nova.api.ec2:NoAuth.factory

[filter:authenticate]
paste.filter_factory = nova.api.ec2:Authenticate.factory

[filter:cloudrequest]
controller = nova.api.ec2.cloud.CloudController
paste.filter_factory = nova.api.ec2:Requestify.factory

[filter:authorizer]
paste.filter_factory = nova.api.ec2:Authorizer.factory

[filter:validator]
paste.filter_factory = nova.api.ec2:Validator.factory

[app:ec2executor]
paste.app_factory = nova.api.ec2:Executor.factory

#############
# Openstack #
#############

[composite:osapi_compute]
use = call:nova.api.openstack.urlmap:urlmap_factory
/: oscomputeversions
/v1.1: openstack_compute_api_v2
/v2: openstack_compute_api_v2

[composite:osapi_volume]
use = call:nova.api.openstack.urlmap:urlmap_factory
/: osvolumeversions
/v1: openstack_volume_api_v1

[composite:openstack_compute_api_v2]
use = call:nova.api.auth:pipeline_factory
noauth = faultwrap noauth ratelimit osapi_compute_app_v2
deprecated = faultwrap auth ratelimit osapi_compute_app_v2
keystone = faultwrap authtoken keystonecontext ratelimit osapi_compute_app_v2
keystone_nolimit = faultwrap authtoken keystonecontext osapi_compute_app_v2

[composite:openstack_volume_api_v1]
use = call:nova.api.auth:pipeline_factory
noauth = faultwrap noauth ratelimit osapi_volume_app_v1
deprecated = faultwrap auth ratelimit osapi_volume_app_v1
keystone = faultwrap authtoken keystonecontext ratelimit osapi_volume_app_v1
keystone_nolimit = faultwrap authtoken keystonecontext osapi_volume_app_v1

[filter:faultwrap]
paste.filter_factory = nova.api.openstack:FaultWrapper.factory

[filter:auth]
paste.filter_factory = nova.api.openstack.auth:AuthMiddleware.factory

[filter:noauth]
paste.filter_factory = nova.api.openstack.auth:NoAuthMiddleware.factory

[filter:ratelimit]
```

```
paste.filter_factory = nova.api.openstack.compute.limits:RateLimitingMiddleware.factory

[app:osapi_compute_app_v2]
paste.app_factory = nova.api.openstack.compute:APIRouter.factory

[pipeline:oscomputeversions]
pipeline = faultwrap oscomputeversionapp

[app:osapi_volume_app_v1]
paste.app_factory = nova.api.openstack.volume:APIRouter.factory

[app:oscomputeversionapp]
paste.app_factory = nova.api.openstack.compute.versions:Versions.factory

[pipeline:osvolumeversions]
pipeline = faultwrap osvolumeversionapp

[app:osvolumeversionapp]
paste.app_factory = nova.api.openstack.volume.versions:Versions.factory

##########
# Shared #
##########

[filter:keystonecontext]
paste.filter_factory = nova.api.auth:NovaKeystoneContext.factory

[filter:authtoken]
paste.filter_factory = keystone.middleware.auth_token:filter_factory
service_protocol = http
service_host = 192.168.69.1
service_port = 5000
auth_host = 192.168.69.1
auth_port = 35357
auth_protocol = http
auth_uri = http://192.168.69.1:5000/
admin_tenant_name = service
admin_user = nova
admin_password = novapass
```

Restart the nova- services prior to running db sync, by running as root:

```
# restart nova-api
# restart nova-compute
# restart nova-network
# restart nova-scheduler
# restart nova-vncproxy
# restart nova-volume
```

Installing additional computer nodes

There are many different ways to perform a multinode install of Compute in order to scale out your deployment and run more. In this case, you can install all the nova- packages and dependencies as you did for the Cloud Controller node, or just install nova-network and nova-compute. Your installation can run any nova- services anywhere, so long as the service can access nova.conf so it knows where the Rabbitmq or Qpid messaging server is installed.

The Compute Node is where you configure the Compute network, the networking between your instances. Because you may need to query the database from the Compute node and learn more information about instances, the nova client and MySQL client or PostgresSQL client packages should be installed on any additional Compute nodes. Copy the nova.conf from your controller node to all additional compute nodes.

If you are using a network bridge configuration and the computer nodes will not have external access, no need to install the nova-network computer nodes.

Configuar e atualizar apt (Verificar se estar usando o mirror ubumirror)

```
# apt-get update
# apt-get upgrade
```

Configure bridge

```
# apt-get install bridge-utils
```

Configurar o network/interfaces

Configurar KVM:

```
# modprobe kvm_intel
echo "kvm_intel" >> /etc/modules
```

Configurar ldap para evitar conflito com usuario "nova" (copiar do /etc/nslcd.conf)

Alterar o /etc/sudoers (copiar da 192.168.1.100)

Instalar nova-compute e nova-network:

```
# apt-get install nova-compute nova-network
# cd /etc/nova/
# mkdir back
# mv * back/
# scp 192.168.1.100:/tmp/nova.conf.tgz .
# tar zxvf nova.conf.tgz
# rm nova.conf.tgz
```

Altear os arquivos de configuracao nova.conf (console de vnc e my_ip)

Colocar o openstack.rc no /root/

Configuring the Database for Compute

Create the tables in your backend data store by running the following command:

```
sudo nova-manage db sync
```

Restart all the services:

```
# restart nova-api
# restart nova-compute
# restart nova-network
# restart nova-scheduler
# restart nova-vncproxy
# restart nova-volume
# restart libvirt-bin
# restart nova-cert
# restart nova-consoleauth
# sudo /etc/init.d/rabbitmq-server restart
```

All nova services are now installed and started. If the "restart" command doesn't work, your services may not be running correctly (or not at all). Review the logs in /var/log/nova.

Creating the Network for Compute VMs

You must run the command that creates the network and the bridge using the br100 specified in the nova.conf file to create the network that the virtual machines use. This example shows the network range using 192.168.70.0/24 as the fixed range for our guest
VMs, but you can substitute the range for the network you have available. We're labeling it with "private" in this case.

```
nova-manage network create private --multi_host=T --fixed_range_v4=192.168.70.0/24 --bridge_interface=br100 --num_networks=1
--network_size=256
```

Verifying the Compute Installation

You can ensure all the Compute services are running by using the nova-manage command:

```
sudo nova-manage service list
```

In return you should see "smiley faces" rather than three X symbols. Here's an example.

```
Binary        Host                    Zone          Status    State Updated_At
nova-compute    educloud01                  nova          disabled   XXX   2012-06-03 14:23:45
```

```
nova-network     educloud01                  nova          enabled    :-)   2012-06-08 12:04:21
nova-scheduler   educloud01                   nova          enabled    :-)   2012-06-08 12:04:14
nova-volume      educloud01                  nova          enabled    :-)   2012-06-08 12:04:21
nova-cert        educloud01                 nova          enabled    :-)   2012-06-08 12:04:21
nova-consoleauth educloud01                    nova          enabled    :-)   2012-06-08 12:04:21
nova-compute     educloud02                   nova          enabled    :-)   2012-06-08 12:04:00
nova-network     educloud02                   nova          enabled    :-)   2012-06-08 12:04:00
```

Creating a multinic instance

In order to create a multinic (multiple network interfaces) the parameter --nic specifying the id of the network created on nova by the nova-manage network create (the net-id parameter).
The user can also specify the ip of the virtual machine (the v4-fixed-ip parameter).
For instance if the user wants to create a virtual machine on two separate network with a specific ips he/she should perform the command

```
# nova boot --flavor 1 --nic net-id=e55b9517-83eb-4377-b7d1-6487529f30f7,v4-fixed-ip=192.168.100.109 --nic
net-id=47a3cef8-b5bf-4880-b621-96d08009d57f,v4-fixed-ip=192.168.200.109
```

Installing and configuring Cinder Volume Manager

Install the packages.

```
$ sudo apt-get install cinder-api cinder-scheduler cinder-volume open-iscsi python-cinderclient tgt
```

Edit /etc/cinder/api-paste.ini (filter authtoken).

```
[filter:authtoken]
paste.filter_factory = keystone.middleware.auth_token:filter_factory
service_protocol = http
service_host = educloud00.local
service_port = 5000
auth_host = educloud00.local
auth_port = 35357
auth_protocol = http
admin_tenant_name = service
admin_user = cinder
admin_password = openstack
```

Edit /etc/cinder/cinder.conf.

```
[DEFAULT]
rootwrap_config=/etc/cinder/rootwrap.conf
sql_connection = mysql://cinder:CINDER_PASS@educloud00.local/cinder
api_paste_config = /etc/cinder/api-paste.ini

iscsi_helper=tgtadm
volume_name_template = volume-%s
volume_group = cinder-volumes
verbose = True
auth_strategy = keystone
#osapi_volume_listen_port=5900

state_path = /var/lib/cinder
lock_path = /var/lock/cinder
volumes_dir = /var/lib/cinder/volumes
```

Verify entries in nova.conf.

```
volume_api_class=nova.volume.cinder.API
enabled_apis=ec2,osapi_compute,metadata
#MAKE SURE NO ENTRY FOR osapi_volume anywhere in nova.conf!!!
#Leaving out enabled_apis altogether is NOT sufficient, as it defaults to include osapi_volume
```

Edit the configuration file: /etc/tgt/targets.conf in order to include the cinder_tgt.conf instead of conf.d/*.conf

```
include /etc/tgt/conf.d/cinder_tgt.conf
```

create the file include /etc/tgt/conf.d/cinder_tgt.conf with the following content:

```
include /var/lib/cinder/volumes/*
```

Restart the tgt service.

```
$ sudo restart tgt
```

Populate the database.

```
$ sudo cinder-manage db sync
```

Restart the services.

```
$ sudo service cinder-volume restart
$ sudo service cinder-api restart
$ sudo service cinder-scheduler restart
```

Create a 1 GB test volume.

```
$ cinder create --display_name test 1
$ cinder list
```

the result should look like this:

```
+--------------------------------------+-----------+--------------+------+-------------+-------------+
|                  ID                  |   Status  | Display Name | Size | Volume Type | Attached to |
+--------------------------------------+-----------+--------------+------+-------------+-------------+
| 5bbad3f9-50ad-42c5-b58c-9b6b63ef3532 | available |     test     |  1   |     None    |             |
+--------------------------------------+-----------+--------------+------+-------------+-------------+
```

Attaching a volume to a virtual machine

First we have to create a virtual machine instance using whatever image and flavor we want:

```
# nova create --image <YOUR IMAGE> --flavor 1 myinstance
```

Then we can use the nova volume-attach tool to attach it to the instance. We need to specify a device identifier within the instance that isn't being used. In

our case we know that /dev/vdb is free, so we will use that:

```
# nova volume-attach myinstance da6ae608-4673-4b24-acd4-75e527b5969a /dev/vdb
```

Booting from a volume

In some cases the users of the virtual machine would like them to be persistent, so changes in kernels, initram would be kept. A solution for these cases is to
use a volume to boot the virtual machine.
After booting from a volume, the user data will be persisted at the file system attached to the virtual machine.

The user can create an image from within a running virtual machine, accessing the device created by the attach command.
Another way to do that is to create a volume from an image. Nova has an utility to do so, the --image-id parameter when using the volume-create utility e.g.:

```
# nova volume-create --image-id <image_id> --display-name my-bootable-vol <size>
```

After creating the volume the user can boot from it using the nova boot command:

```
# nova boot --flavor <favlor_id> --block_device_mapping <dev_name=mapping> boot-from-vol-test
```

where device_mapping} in the format <dev_name$>=<id>:<type>:<size(GB)>:<delete_on_terminate>, with dev_name being the device name where the volume will be attached in the virtual machine at /dev/dev_name.
Most of the times, this value is vda. id is the id of the volume created with the cinder create command.

type is a string that describes the process used to creat the volume, e.g, it can be snap, which means that the volume was created from a snapshot.
This parameter can be left in blank.

The delete_on_terminate specifies wether the volume should be deleted after the virtual machine is terminated.

Installing the OpenStack frontend - Horizon

OpenStack has components that provide a view of the OpenStack installation such as a Django-built website that serves as a dashboard. You can use a dashboard interface with an OpenStack Compute installation with a web-based console provided by the Openstack-Dashboard project. It provides web-based interactions with the OpenStack Compute cloud controller through the OpenStack APIs. For more information about the Openstack-Dashboard project, please visit: https://github.com/openstack/horizon/. These instructions are for an example deployment configured with an Apache web server.

System Requirements for the Dashboard

- You should have a running OpenStack Compute installation with the Identity Service, Keystone, enabled for identity management.
- The dashboard needs to be installed on the node that can contact the Identity Service.
- You should know the URL of your Identity endpoint and the Compute endpoint.
- You must know the credentials of a valid Identity service user.
- Python 2.6 is required;
- It should run on any system with Python 2.6 or 2.7 that is capable of running Django;
- Optional components:
    - *An Image Store (Glance) endpoint.*
    - *An Object Store (Swift) endpoint.*

Install the OpenStack Dashboard, as root:

```
# apt-get install -y memcached libapache2-mod-wsgi openstack-dashboard
```

Next, modify the variable CACHE_BACKEND in /etc/openstack-dashboard/ local_settings.py to match the ones set in /etc/memcached.conf. Open /etc/openstack-dashboard/local_settings.py and look for this line:

```
CACHE_BACKEND = 'memcached://127.0.0.1:11211/'
```

To configure the MySQL database, create the dash database.

```
mysql> CREATE DATABASE dash;
mysql> GRANT ALL ON dash.* TO 'dash'@'%' IDENTIFIED BY 'dashpass';
mysql> quit;
```

After configuring the local_settings.py as shown below, you can run the manage.py syncdb command to populate this newly-created database.

```
import os
from django.utils.translation import ugettext_lazy as _
DEBUG = False
TEMPLATE_DEBUG = DEBUG
PROD = False
USE_SSL = False
SWIFT_ENABLED = True
ENABLE_JUJU_PANEL = True
SECRET_KEY = 'elj1IWiLoWHgcyYxFVLj7cM5rGOOxWl0'
LOCAL_PATH = os.path.dirname(os.path.abspath(__file__))
CACHE_BACKEND = 'memcached://192.168.69.1:11211/'
EMAIL_BACKEND = 'django.core.mail.backends.console.EmailBackend'
OPENSTACK_HOST = "192.168.69.1"
OPENSTACK_KEYSTONE_URL = "http://%s:5000/v2.0" % OPENSTACK_HOST
```

```
OPENSTACK_KEYSTONE_DEFAULT_ROLE = "Member"
OPENSTACK_KEYSTONE_BACKEND = {
   'name': 'native',
   'can_edit_user': True
}
API_RESULT_LIMIT = 1000
LOGGING = {
      'version': 1,
      # When set to True this will disable all logging except
      # for loggers specified in this configuration dictionary. Note that
      # if nothing is specified here and disable_existing_loggers is True,
      # django.db.backends will still log unless it is disabled explicitly.
      'disable_existing_loggers': False,
      'handlers': {
         'null': {
            'level': 'DEBUG',
            'class': 'django.utils.log.NullHandler',
            },
         'console': {
            # Set the level to "DEBUG" for verbose output logging.
            'level': 'INFO',
            'class': 'logging.StreamHandler',
            },
         },
      'loggers': {
         # Logging from django.db.backends is VERY verbose, send to null
         # by default.
         'django.db.backends': {
            'handlers': ['null'],
            'propagate': False,
            },
         'horizon': {
            'handlers': ['console'],
            'propagate': False,
         },
         'novaclient': {
            'handlers': ['console'],
            'propagate': False,
         },
         'keystoneclient': {
            'handlers': ['console'],
            'propagate': False,
         },
         'nose.plugins.manager': {
            'handlers': ['console'],
            'propagate': False,
         }
      }
}
DATABASES = {
'default': {
'ENGINE': 'django.db.backends.mysql',
'NAME': 'dash',
'USER': 'dash',
'PASSWORD': 'dashpass',
'HOST': '192.168.69.1',
'default-character-set': 'utf8'
},
}
```

Run the manage.py syncdb command to initialize the database.

```
$ /usr/share/openstack-dashboard/manage.py syncdb
```

Restart Apache to pick up the default site and symbolic link settings.

```
# /etc/init.d/apache2 restart
```

Restart the nova-api service to ensure the API server can connect to the Dashboard and to avoid an error displayed in the Dashboard

```
sudo restart nova-api
```

Beraldo

16-04-2012

Executou: - Iniciou a atividade: "[[#554]]: Pesquisa investigativa do OpenStack"

Próximos Passos:
- **Continuar na atividade: "[[#554]]: Pesquisa investigativa do OpenStack"**

Blocker/Dependencia:

- Nenhuma

17-04-2012

Executou:
- **Finalizou a atividade: "[[#554]]: Pesquisa investigativa do OpenStack"**

Próximos Passos:
- **Iniciou a atividade: "[[#561]]: Investigação do OpenStack Compute - Nova"**

Blocker/Dependencia:

- Nenhuma

18-04-2012

Executou:
- **Finalizou a atividade: "[[#561]]: Investigação do OpenStack Compute - Nova"**

Próximos Passos:
- **Iniciou a atividade: "[[#562]]: Investigação do OpenStack Object Storage - Swift"**

Blocker/Dependencia:

- Nenhuma

19-04-2012

Executou:
- **Finalizou a atividade: "[[#562]]: Investigação do OpenStack Object Storage - Swift"**

Próximos Passos:
- **Iniciou a atividade: "[[#563]]: Investigação do OpenStack Image Service - Glance"**

Blocker/Dependencia:

- Nenhuma

20-04-2012

Executou:
- **Continuou a atividade: "[[#563]]: Investigação do OpenStack Image Service - Glance"**

Próximos Passos:

- **Continuar a atividade: "[[#563]]: Investigação do OpenStack Image Service - Glance"**

Blocker/Dependencia:

- Nenhuma

23-04-2012

Executou:

- **Finalizou a atividade: "[[#563]]: Investigação do OpenStack Image Service - Glance"**
- **Finalizou a atividade: "[[#566]]: Pesquisar sobre o modo " ip_hash"**
- **Finalizou a atividade: "[[#565]]: Pesquisar sobre o modo "keepalive"**

Próximos Passos:

- **Iniciar a atividade: "[[#555]]: Levantar os Requisitos do Problema"**

Blocker/Dependencia:

- Nenhuma

24-04-2012

Executou:

- **Iniciou e Finalizou a atividade: "[[#555]]: Levantar os Requisitos do Problema"**
- **Iniciou e Finalizou a atividade: "[[#556]]: Pesquisa investigativa na área"**
- **Iniciou a atividade: [[#557]]: Pesquisa investigativa do balanceador HTTP"**

Próximos Passos:

- **Continuar a atividade: [[#557]]: Pesquisa investigativa do balanceador HTTP"**

Blocker/Dependencia:

- Nenhuma

25-04-2012

Executou:

- **Finalizou a atividade: [[#557]]: Pesquisa investigativa do balanceador HTTP"**

Próximos Passos:

- **Iniciar a atividade: [[#564]]: Pesquisar algoritmos disponíveis na ferramenta escolhida"**

Blocker/Dependencia:

- Nenhuma

26-04-2012

Executou:

- **Finalizou a atividade: [[#564]]: Pesquisar algoritmos disponíveis na ferramenta escolhida"**
- **Finalizou a atividade: [[#567]]: Pesquisar sobre análise de desempenho:**

Próximos Passos:

- **Iniciar a atividade: [[#568]]: Pesquisar sobre possíveis modos de degranagem"**

Blocker/Dependencia:

- Nenhuma

27-04-2012

Executou:

- **Finalizou a atividade: [[#568]]: Pesquisar sobre possíveis modos de degranagem"**

Próximos Passos:

- **Preparar o relatório técnico**

Blocker/Dependencia:

- Nenhuma

Beraldo

Voltar

Busca pelo maior número de operações por segundo em 4 máquinas sem HT - Variação do uso de memória

- Variação de memória de 80% a 94% de uso;
- QxP: 2x4;

Ambiente:

- Máquina Educloud05:

- 8 Cores de 2.67 GHz;
- 6 GB de Ram;

- Máquina Educloud06:

- 8 Cores de 2.67 GHz;
- 6 GB de Ram;

- Máquina Educloud07:

- 8 Cores de 2.67 GHz;
- 6 GB de Ram;

- Máquina Educloud08:

- 8 Cores de 2.67 GHz;
- 6 GB de Ram;

Resultados:

Cálculo do maior número de operações por segundo:

- Valor de NB ótimo: 112 (94% de uso de memória);
- Número de Operações de Ponto Flutuante por Segundo: 2.11695e+02 GFlops;
- Gráfico de Tempo vs NB vs Número de Operações:
  https://ncc.unesp.br/its/attachments/237/NBvsGflopsvsTime.png

Estudo da utilização da memória:

- Gráfico do Número de operações Vs Tamanho de problema (NB) para porcentagens de memória de 80 à 94%
  https://ncc.unesp.br/its/attachments/238/NBvsGflops.png

- Gráfico do Tempo Vs Tamanho de problema (NB) para porcentagens de memória de 80 à 94%
  https://ncc.unesp.br/its/attachments/239/NBvsTime.png

- Gráfico da Variação do Tempo para cada percentagem de memória.
  https://ncc.unesp.br/its/attachments/242/MemvsTime.png

- Gráfico da Variação do Número de Operações para cada percentagem de memória.
  https://ncc.unesp.br/its/attachments/243/MemvsGflops.png

Discussão.

- Encontramos o momento, na escala de uso de memória onde a utilização da swap faz com que o desempenho caia: 94%, com tamanho de problemas a partir de NB=160 ;
- Nesse ponto é onde se observa a maior variação do número de operações (de 2.02217e+02 em NB=152 para 1.29049e+02 em NB=160) e no tempo (de 496.41 segundos em NB=152 para 781.38 em NB=160);

Files

NBvsGflopsvsTime.png	8.48 KB	22 Apr 2013	Raphael Cóbe
NBvsGflops.png	8.35 KB	22 Apr 2013	Raphael Cóbe
NBvsTime.png	7.46 KB	22 Apr 2013	Raphael Cóbe
MemvsTime.png	4.37 KB	22 Apr 2013	Raphael Cóbe
MemvsGflops.png	5.33 KB	22 Apr 2013	Raphael Cóbe

Busca pelo maior número de operações por segundo em 4 máquinas virtuais - Variação do uso de memória

- Variação de memória de 80% a 92% de uso;
- QxP: 2x4;

Ambiente:

- Máquina Hospedeiras Educloud05 - Educloud08:

- Máquina Virtual (*4):

- 8 Cores de 2.67 GHz;
- 6 GB de Ram;

Resultados:

Cálculo do maior número de operações por segundo:

- Valor de NB ótimo: 144 (90% de uso de memória);
- Número de Operações de Ponto Flutuante por Segundo: 2.05980*10^02 GFlops;
- Gráfico de Tempo vs NB vs Número de Operações:
  https://ncc.unesp.br/its/attachments/255/NBvsGflopsvsTime.png

Estudo da utilização da memória:

- Gráfico do Número de operações Vs Tamanho de problema (NB) para porcentagens de memória de 80 à 92%
  https://ncc.unesp.br/its/attachments/256/NBvsGflops.png

- Gráfico do Número de operações Vs Tamanho de problema (NB) para porcentagens de memória de 80 à 90%
  https://ncc.unesp.br/its/attachments/260/NBvsGflops.png

- Gráfico do Tempo Vs Tamanho de problema (NB) para porcentagens de memória de 80 à 92%
  https://ncc.unesp.br/its/attachments/257/NBvsTime.png

- Gráfico do Tempo Vs Tamanho de problema (NB) para porcentagens de memória de 80 à 90%
  https://ncc.unesp.br/its/attachments/261/NBvsTime.png

- Gráfico da Variação do Tempo para cada percentagem de memória.
  https://ncc.unesp.br/its/attachments/258/MemvsTime.png

- Gráfico da Variação do Número de Operações para cada percentagem de memória.
  https://ncc.unesp.br/its/attachments/259/MemvsGflops.png

Discussão.

- Encontramos o momento, na escala de uso de memória onde a utilização da swap faz com que o desempenho caia: 92%, com tamanho de problemas a partir de NB=96 ;
- Nesse ponto é onde se observa a maior variação do número de operações (de 1.43483e+02 em NB=256 (90% de uso de memória) para 5.61840e+00 em NB=96 (92% de uso de memória)) e no tempo (de 614.34 segundos em NB=256 (90% de uso de memória) para 16808.45 em NB=96 (92% de uso de memória));

Files

NBvsGflopsvsTime.png	8.94 KB	29 Apr 2013	Raphael Cóbe
NBvsGflops.png	9.53 KB	29 Apr 2013	Raphael Cóbe
NBvsTime.png	8.73 KB	29 Apr 2013	Raphael Cóbe
MemvsTime.png	4.58 KB	29 Apr 2013	Raphael Cóbe
MemvsGflops.png	5.76 KB	29 Apr 2013	Raphael Cóbe
NBvsGflops.png	10.9 KB	29 Apr 2013	Raphael Cóbe
NBvsTime.png	7.33 KB	29 Apr 2013	Raphael Cóbe

Busca pelo maior número de operações por segundo em uma unica máquina sem HT - Variação do uso de memória

- Máquina Educloud05:

Ambiente:

- 8 Processadores de 2.67 GHz;
- 6 GB de Ram;
- Variação de memória de 80% a 94% de uso;
- QxP: 2x4;

Resultados:

Cálculo do maior número de operações por segundo:

- Valor de NB ótimo: 184 (92% de uso de memória);
- Número de Operações de Ponto Flutuante por Segundo: 7.82992e+01 GFlops;
- Gráfico de Tempo vs NB vs Número de Operações:
  https://ncc.unesp.br/its/attachments/232/NBvsGflopsvsTime.png

Estudo da utilização da memória:

- Gráfico do Número de operações Vs Tamanho de problema (NB) para porcentagens de memória de 80 à 94%
  https://ncc.unesp.br/its/attachments/234/NBvsGflops.png

- Gráfico do Tempo Vs Tamanho de problema (NB) para porcentagens de memória de 80 à 94%
  https://ncc.unesp.br/its/attachments/233/NBvsTime.png

- Gráfico da Variação do Tempo para cada percentagem de memória.
  https://ncc.unesp.br/its/attachments/235/MemvsTime.png

- Gráfico da Variação do Número de Operações para cada percentagem de memória.
  https://ncc.unesp.br/its/attachments/236/MemvsGflops.png

Discussão.

- Encontramos o momento, na escala de uso de memória onde a utilização da swap faz com que o desempenho caia: 94%, com tamanho de problemas a partir de 192 ;
- Nesse ponto é onde se observa a maior variação do número de operações (de 7.15501e+01 em NB=192 para 2.97790e+01 em NB=200) e no tempo (de 173.33 segundos em NB=192 para 421.39 em NB=200);

Files

NBvsGflopsvsTime.png	9.58 KB	22 Apr 2013	Raphael Cóbe
NBvsTime.png	8.14 KB	22 Apr 2013	Raphael Cóbe
NBvsGflops.png	8.74 KB	22 Apr 2013	Raphael Cóbe
MemvsTime.png	4.58 KB	22 Apr 2013	Raphael Cóbe
MemvsGflops.png	5.42 KB	22 Apr 2013	Raphael Cóbe

Busca pelo maior número de operações por segundo em uma unica máquina virtual - Variação do uso de memória

- Máquina Hospedeira: Educloud05

Ambiente (Máquina Hospedeira):

- 8 Cores de 2.67 GHz;
- 6 GB de Ram;

Ambiente (Máquina Virtual):

- 8 Cores de 2.67 GHz;
- 6 GB de Ram;

- Variação de memória de 80% a 92% de uso;

- QxP: 2x4;

Resultados:

Cálculo do maior número de operações por segundo:

- Valor de NB ótimo: 144 (90% de uso de memória);
- Número de Operações de Ponto Flutuante por Segundo: 7.63280*10^01 GFlops;
- Gráfico de Tempo vs NB vs Número de Operações:
  https://ncc.unesp.br/its/attachments/247/NBvsGflopsvsTime.png

Estudo da utilização da memória:

- Gráfico do Número de operações Vs Tamanho de problema (NB) para porcentagens de memória de 80 à 92%
  https://ncc.unesp.br/its/attachments/248/NBvsGflops.png

- Gráfico do Número de operações Vs Tamanho de problema (NB) para porcentagens de memória de 80 à 90%
  https://ncc.unesp.br/its/attachments/252/NBvsGflops.png

- Gráfico do Tempo Vs Tamanho de problema (NB) para porcentagens de memória de 80 à 92%
  https://ncc.unesp.br/its/attachments/249/NBvsTime.png

- Gráfico do Tempo Vs Tamanho de problema (NB) para porcentagens de memória de 80 à 90%
  https://ncc.unesp.br/its/attachments/253/NBvsTime.png

- Gráfico da Variação do Número de Operações para cada percentagem de memória.
  https://ncc.unesp.br/its/attachments/250/MemvsGflops.png

- Gráfico da Variação do Tempo para cada percentagem de memória.
  https://ncc.unesp.br/its/attachments/251/MemvsTime.png

Discussão.

- Encontramos o momento, na escala de uso de memória onde a utilização da swap faz com que o desempenho caia: 92%, com tamanho de problemas a partir de 96 ;
- Nesse ponto é onde se observa a maior variação do número de operações (de 7.03238e+01 em NB=256 (90% de memória) para 1.33683e+01 em NB=96 (92% de memória) e no tempo (de 154.34 segundos em NB=256 (90% de uso de memória) para 878.2 em NB=96 (92% de uso de memória));

Files

NBvsGflopsvsTime.png	9.71 KB	29 Apr 2013	Raphael Cóbe
NBvsGflops.png	8.12 KB	29 Apr 2013	Raphael Cóbe
NBvsTime.png	6.86 KB	29 Apr 2013	Raphael Cóbe
MemvsGflops.png	5.17 KB	29 Apr 2013	Raphael Cóbe
MemvsTime.png	4.32 KB	29 Apr 2013	Raphael Cóbe
NBvsGflops.png	11.4 KB	29 Apr 2013	Raphael Cóbe
NBvsTime.png	7.61 KB	29 Apr 2013	Raphael Cóbe

User Stories do Ciclo 1 - Open Nebula Brain Honey e Balanceador de Carga

- Open Nebula
  - [[Abrir Console do Virtualizador]] - Issue #458
  - [[Instanciar uma Máquina Virtual com Acesso à Internet]] - Issue #464
  - [[Liberar uma Máquina Virtual]] - Issue #469
  - [[Pausar uma Máquina Virtual]] - Issue #471
  - [[Resumir uma Máquina Virtual]] - Issue #473

- Imagens Brain Honey
  - [[Instanciar Brain Honey via VMWare]] - Issue #480
  - [[Instanciar Brain Honey via Open Nebula]] - Issue #488
  - [[Escalar Brain Honey Manualmente, via Open Nebula]] - Issue #494

- Balanceador de Carga
  - [[Monitorar Continuamente Duas Métricas de Status dos Servidores]] - Issue #496
  - [[Detectar Excesso de Demanda, e Instanciar VM Webserver]] - Issue #500
  - [[Detectar Baixa Demanda, e Liberar VM Webserver]] - Issue #505

Notas de Scrum do Ciclo 2

[[Marco]]
[[Beraldo]]

Voltar

Ciclo 2 - Arquitetura

[[OpenStack]]
[[Balanceadores]]
[[Hadoop]]
[[OCCI]]
[[Versão Final]]

Notas de Scrum do Ciclo 3

[[ScrumCiclo3Marco|Marco]]
[[ScrumCiclo3Beraldo|Beraldo]]

Voltar

Detectar Baixa Demanda e Liberar VM Webserver

Implantação #506: Definir e Configurar Bottom Treshold

Atualmente para propósitos de demonstração o *educloud-mon* está configurado com duas métricas: **web-cpu-usage** e **web-active-users**.

Estas duas métricas se referem a uma média aritimética do uso de cpu ou usuários ativos do sistema como um todo (todos os hosts web servers).

O parâmetro de configuração do *educloud-mon* é o bottom_threshold de cada métrica. Para saber quais valores estão configurados para bottom_treshold veja o arquivo educloud-mon.conf.

Implantação #509: Configurar Comando de Liberação de VM, com Base no Atingimento do Bottom Treshold

O *educloud-mon* libera uma máquina virtual caso nenhuma métrica atingiu o top_thresold e alguma métrica tenha atingido o bottom_thresold.

Ainda existe o parâmetro minimum_vms do arquivo de configuração educloud-mon.conf, caso esta quantidade de máquinas virtuais seja atingido, o daemon, não irá mais liberar máquinas virtuais, mesmo que as métricas retornem valores abaixo do bottom_thresold. Isso evita que a farm de servidores web, fique sem máquinas.

Detectar Excesso de Demanda e Instanciar VM Webserver

Implantação #501: Definir e configurar top treshold

Atualmente para propósitos de demonstração o *educloud-mon* está configurado com duas métricas: **web-cpu-usage** e **web-active-users**.

Estas duas métricas se referem a uma média aritimética do uso de cpu ou usuários ativos do sistema como um todo (todos os hosts web servers).

O parâmetro de configuração do *educloud-mon* é o top_threshold de cada métrica. Para saber quais valores estão configurados para top_treshold veja o arquivo educloud-mon.conf.

Implantação #502: Configurar LVM para Obter Ganho de Performance na Instanciação de VMs

A máquina osdc01 (200.145.46.197) é um repositório de imagens. Nele temos o seguinte layout de lvm:

Volumes físicos (Comando pvdisplay):

```
--- Physical volume ---
PV Name               /dev/sda3
VG Name               osdc1
PV Size               18.18 TiB / not usable 1.00 MiB
Allocatable           yes
PE Size               4.00 MiB
Total PE              4765714
Free PE               254883
Allocated PE          4510831
PV UUID               L4yscI-w5r3-0IVh-FtxE-oGOJ-EtTM-5nBjt2
```

Grupos (Comando vgdisplay):

```
--- Volume group ---
VG Name               osdc1
System ID
Format                lvm2
Metadata Areas        1
Metadata Sequence No  276
VG Access             read/write
VG Status             resizable
MAX LV                0
Cur LV                9
Open LV               7
Max PV                0
Cur PV                1
Act PV                1
VG Size               18.18 TiB
PE Size               4.00 MiB
Total PE              4765714
Alloc PE / Size       4510831 / 17.21 TiB
Free  PE / Size       254883 / 995.64 GiB
VG UUID               3oKaJb-HYqQ-cOZK-duiF-qGVR-5P9H-HcrWaa
```

Volumes Lógicos (Comando lvdisplay):

Cada imagem está em um volume lógico e é exportado para as máquinas nós, via iSCSI. Para isso, elas usam o *transport manager* iscsi desenvolvido pelo Marco.

Para ver a lista de volumes lógicos (São muitos, dependendo das máquinas virtuais que estão em execução), por favor execute:

```
lvdisplay
```

Implementação #503: Configurar Comando de Instanciação de VM, com Base no Atingimento do Top Treshold

O *educloud-mon* inicia uma máquina virtual caso alguma métrica tenha atingido o top_thresold.

Implementação: #504: Simular Excesso de Demanda e Verificar Instanciação Automática de VMs, via Open Nebula

Para simular o excesso de demanda foi utilizado o comando ab do pacote apache-utils.

O ab inicia quantas threads forem necessárias e envia N requisições para um determinado host. Sua utilização é muito simples. Abaixo um exemplo do comando ab:

```
ab -n 2000 -c 50 http://cursos.educacao.sp.gov.br/
```

Abaixo temos a saída do comando acima:

```
This is ApacheBench, Version 2.3 <$Revision: 655654 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking cursos.educacao.sp.gov.br (be patient)
Completed 200 requests
Completed 400 requests
Completed 600 requests
Completed 800 requests
Completed 1000 requests
Completed 1200 requests
Completed 1400 requests
Completed 1600 requests
Completed 1800 requests
Completed 2000 requests
Finished 2000 requests

Server Software:        squid/2.7.STABLE9
Server Hostname:        cursos.educacao.sp.gov.br
Server Port:            80

Document Path:          /
Document Length:        1321 bytes

Concurrency Level:      50
Time taken for tests:   0.641 seconds
Complete requests:      2000
Failed requests:        0
Write errors:           0
Non-2xx responses:      2000
Total transferred:      3312000 bytes
HTML transferred:       2642000 bytes
Requests per second:    3118.24 [#/sec] (mean)
Time per request:       16.035 [ms] (mean)
Time per request:       0.321 [ms] (mean, across all concurrent requests)
Transfer rate:          5042.78 [Kbytes/sec] received

Connection Times (ms)
              min  mean[+/-sd] median   max
Connect:        0    1   0.3      0       5
Processing:     5   15   2.0     15      24
Waiting:        3   13   1.9     13      22
Total:          7   16   1.9     16      25
ERROR: The median and mean for the initial connection time are more than twice the standard
       deviation apart. These results are NOT reliable.

Percentage of the requests served within a certain time (ms)
  50%     16
  66%     16
  75%     16
  80%     17
  90%     18
  95%     18
  98%     23
  99%     24
 100%     25 (longest request)
```

Educloud-monconf

```
# educloud-mon main config file.

# General section
[general]

# This daemon will run in background or in foreground?
background = 0

# Run daemon in debug mode will produce further information in the system
# log.
debug = 1

# Location of the log file
logfile = /var/log/educloud-mon.log

# Run daemon as the follow user
user = "oneadmin"

# Run daemon as the follow group
group = "oneadmin"

# Run in dry-run mode ?
# In this mode virtual machines are not created or deleted.
# Default is false.
dry-run = 0

# How many seconds wait after start a vm ?
# Default is 30 seconds
start_vm_wait = 60

# How many seconds wait to start new cicle of metrics checks ?
# This value determines the seconds between each cicle of N
# metrics. Default is 30 seconds
cicle_wait = 30

# Number maximum of retries to start a vm. After this, if vm state is not
# "running", a notification will be sent
# Default is 3
max_vm_retry = 3

# List of metrics to consider. This wil be converted to a dict in the python
# deamon, so please, use the same python syntax.
# Each metric must have: 'top_thresold', 'bottom_thresold' and 'script'
# Each script MUST return in stdout only the value to consult and compare with
# thresolds defined here.
# TODO: Better documentation
metrics = {
        'cpu-load': {
          'top_thresold': '70',
          'bottom_thresold': '2',
          'script': '/usr/local/bin/educloud-mon/scripts/web-cpu-usage.py',
          'check_interval': '2',
          'average_checks': '5',
        },
        'active-users': {
          'top_thresold': '20',
          'bottom_thresold': '5',
          'script': '/usr/local/bin/educloud-mon/scripts/web-active-users.py',
          'check_interval': '2',
          'average_checks': '5',
        },
    }

# This command will be execute to get number of current http connection
# in a vm. This will be appended th vm id
get_connections_cmd = "ssh osdc07 iis_connected_users.sh"

# nginx web server cluster
[nginx]

# This command will be executed when a new vm is added to the pool
```

```
# This command will be appended with vm_id
nginx_add_cmd = "ssh osdc07 nginx_webservers.sh add"

# This command will be executed when a vm is removed from pool
# This command will be appended with vm_id
nginx_del_cmd = "ssh osdc07 nginx_webservers.sh del"

# This command will be executed after a add or del action in nginx cluster
nginx_restart_cmd = "ssh osdc07 sudo /usr/sbin/service nginx restart"

# Settings for email notification system
[notification]

# If enabled, emails will be sent to administrators when errors occur.
# Default is true.
enable_notifications = 0

# SMTP Server to use
smtp_server = 127.0.0.1

# SMTP Port
smtp_port = 25

# Use SMTP Authentication ?
# Default is true.
smtp_auth = 0

# SMTP User for authentication
smtp_user = user

# SMTP Pass for authentication
smtp_pass = password

# Mail From
mail_from = "educloud-mon <educloud-mon@ncc.unes.pbr"

# Mail to
mail_to = "admin@ncc.unesp.br"

# Options to connect with OpenNebula goes in this section
[opennebula]

# RPC OpenNebula URL. You have to specify a full URL of RCP with port and path.
# Without quotes. Example: http://localhost:2633/RPC2
rpc_url = http://200.145.46.198:2633/RPC2

# User to use in opennebula connection. Without quotes.
user = oneadmin

# Password to use in opennebula connection. Without quotes.
password = !@qazxsw

# Relative path to template to start when metrics get values >= top_thresolds
# This path is relative to the home of the user oneadmin on OpenNebula
# controller. This hardcoded path is necessary because we are using OpenNebula
# 2.2.1 version.
template = "/usr/local/bin/educloud-mon/templates/teste.template"

# Minimum amount of machine in "web cluster"
# If the cluster has this number no more virtual machines will be
# deleted. Default is 1
minimum_vms = 1
```

Escalar Brain Honey Manualmente via Open Nebula

- Para escalar manualmente o brain honey, é necessário realizar o seguinte procedimento:
  - *Submeter uma nova VM webserver:*
    - *onevm submit /path/to/Lena.template*
  - *Após a VM estar up & running, resgatar informação do endereço ip:*
    - *cat /var/lib/misc/dnsmasq.leases | grep $(onevm show $VM_ID | grep MAC= | cut -d "=" -f2 | cut -d "," -f1) | awk '{ print $3 }'*
  - *Adicionar esse ip ao pool de máquinas que recebem direcionamento do nginx*

Voltar

Fase 2

- [[Raphael]]
- [[Thiago]]

Hadoop

Objetivo

O escopo deste documento trata de avaliar um conjunto de soluções de storage aplicado à cloud computing, avaliando a viabilidade de uma delas atender às necessidades do projeto EduCloud, para posterior deployment. Fez parte da investigação descrita por este documento, a verificação dos passos para execução da instalação e configuração da solução de storage.

Projeto Educloud

O projeto Educloud é um projeto para a implementação de uma infraestrutura de cloud computing para a secretaria da educação do estado de São Paulo, com objetivo de fornecer serviços de apoio ao ensino via nuvem, para educadores e estudantes.

Hadoop

Introdução

O Hadoop é uma solução de sistema de arquivos distribuídos desenvolvida para manipular volumes de dados, focando na análise qualitativa dos dados armazenados através de ferramentas de data mining desenvolvidas utilizando o framework fornecido pelo Google, seu fabricante.

Desenvolvido na linguagem Java, o Hadoop oferece um sistema de arquivos paralelo e distribuído com tolerância à falha, e um framework para desenvolver aplicações de análise dos dados armazenados de forma a correlacioná-los.

Alternativas

Algumas tecnologias alternativas ao Hadoop são apresentadas a seguir, porém nem todas são capazes de trabalhar com grandes volumes de dados, ou até mesmo de fazer a correlação de dados.

- dCache - É uma solução de filesystem paralelo para grandes volumes de dados, onde o acesso aos dados é feito através de um conjunto de algoritmos.
- Glusterfs - Diferente do Hadoop, apresenta vários servidores de dados, que são aglutinados no momento em que o cliente realiza a montagem de um diretório.
- Lustre - É uma solução onde o servidor de metadados é centralizado e os servidores de dados são distribuídos para aumentar a velocidade de leitura e gravação de dados.

Todas as soluções acima foram retiradas do seguinte site:
http://en.wikipedia.org/wiki/List_of_file_systems#Distributed_parallel_fault-tolerant_file_systems

Licença

O Hadoop é licenciado sob a Apache v2.0. Nesta versão de licenciamento é possível o uso para qualquer propósito, alterar e redistribuir o código sob outro tipo de licenciamento, desde que seja incluído um changelog indicando as modificações realizadas. O texto da licença se encontra em:
http://www.apache.org/licenses/LICENSE-2.0

Adoção no Mundo Corporativo

O Hadoop tem seu uso bem vasto entre diversas companhias. Algumas delas o utilizam como parte de sua infraestrutura, onde a função do Hadoop é dar suporte ao produto ou serviço comercializado, enquanto em outras o próprio Hadoop, ou uma de suas características, é o serviço oferecido.

Dentre as empresas que usam Hadoop em sua infraestrutura, temos as seguintes:

- Amazon - Intuito de auxiliar no processo de buscas;
- Adobe - Usos diversos dentra companhia;
- AOL - Uso para análises estatísticas;
- eBay - Uso para pesquisa e otimização de buscas;
- Facebook - Uso para análise de logs e aprendizado de máquina;
- FOX - Uso para analises, mineração de dados e aprendizado de máquina;
- Google e IBM - Parceria para estudos sobre desafios em escalabilidade de aplicações em internet;
- LinkedIn - Uso nos algoritmos de buscas de contatos;
- Twitter - Uso para armazenamento de tweets, logs e outros tipos de dados gerados pela companhia;
- Yahoo - Suporte na pesquisa do sistema de propaganda do Yahoo, e em testes de escalabilidade.

Outras empresas que usam Hadoop em sua infraestrutura pode ser listada em: http://wiki.apache.org/hadoop/PoweredBy

A seguir algumas empresas que oferecem o Hadoop como serviço:

- Amazon Web Service - Inserido dentro da infraestrutura EC2 como parte do serviço Amazon Elastic MapReduce;
- Cloudera - Oferece uma plataforma composta por produtos open source (no qual o Hadoop está inserido) para seus clientes;
- IBM - Distribui uma versão derivada do Hadoop para rodar nas JVMs da IBM.

Outras empresas que oferecem Hadoop como um serviço/produto podem ser encontradas no seguinte link:

http://wiki.apache.org/hadoop/Distributions%20and%20Commercial%20Support

Analisando os dados fornecidos acima, podemos observar que grandes empresas utilizam o Hadoop em sua infraestrutura ou como parte de seus serviços/produtos; portanto podemos afirmar que o Hadoop tem uma vasta utilização e por isso o interesse das empresas na continuidade desse projeto é latente.

Observando também o link
https://issues.apache.org/jira/browse/HADOOP#selectedTab=com.atlassian.jira.plugin.system.project:roadmap-panel&allVersions=true , podemos observar a existência de um roadmap para futuras versões do Hadoop. O release atual é o 1.0, e existem trabalhos nas versões 1.1, 2.0 e 3.0. Além disso, é incluído no roadmap melhorias nas versões já lançadas, como as versões 0.21.1, 0.22.1, 0.23.2, 0.23.3 e 0.24.0. Há também planos de criação de uma versão para Windows do Hadoop.

Com essas informações é possível afirmar que o Hadoop, como projeto, não tem uma baixa expectativa de vida, garantindo sua adoção sem preocupações futuras de substituição de tecnologia por obsolescência.

Uso do Hadoop no Projeto Educloud

O projeto EduCloud necessita uma tecnologia que possa armazenar, manipular e gerenciar grandes volumes de dados. As ferramentas que melhor atendem hoje a essa necessidade são conhecidas como sistemas de arquivos distribuídos. Estes sistemas de arquivos potencialmente atendem o requisito do EduCloud, permitindo que este seja utilizado para distribuir de conteúdo para os usuários, e correlacionar dados para detectar semelhanças entre estudos de usuários distintos.

Nesse sentido, o Hadoop atende a necessidade do EduCloud, logo que o objetivo principal do primeiro é oferecer um sistema de arquivos distribuídos, capaz de manter e manipular arquivos muito grandes (várias centenas de terabytes), e em grandes quantidades. Porém, dado que não está claro que o projeto EduCloud irá trabalhar com arquivos grandes, é provável que o Hadoop não seja a solução mais indicada para o projeto, pois todo o trabalho existente no Hadoop envolve a movimentação e disponibilidade de grandes volumes de dados. Segundo artigos, o Hadoop não possui performance ótima mexendo com pequenos volumes de dados. Os links a seguir irão aproveitar e mostrar quais situações nas quais o Hadoop apresenta desvantagem:
http://wiki.apache.org/hadoop/HadoopIsNot
Este link indica uma lista de necessidades para as quais o Hadoop não foi feito; dentre elas:
- Ele não foi feito para substituir um bando de dados.
- O mapreduce nem sempre é o melhor algoritmo de análise.
- O Hadoop não é indicado para pessoas leigas em conhecimento sobre: a linguagem Java; projeto, implantação e manutenção de redes de computadores e administração de sistemas.
- Ele não é o substituto para sistemas de arquivos SAN em alta disponibilidade.
- E o HDFS não está de acordo com o padrão POSIX.

http://www.quantivo.com/blog/top-5-reasons-not-use-hadoop-analytics
Neste link, podemos observar alguns pontos indicados para não usar Hadoop para análises:
- Hadoop é um framework, não é uma solução pronta.
- As ferramentas Hive e Pig não solucionam problemas da arquitetura do Hadoop.
- A instalação é simples, porém manutenção e desenvolvimento é custoso.
- O forte da análise do Hadoop é em sumarização, porém é fraco em análises pontuais.
- O Hadoop será eficaz em suas análises, se os algoritmos de MapReduce também o forem.

http://tech.backtype.com/the-dark-side-of-hadoop
Por fim, este link critica alguns pontos a respeito do Hadoop:
- Má documentação para variáveis de configuração, consideradas críticas.
- Gerenciamento de memória muito ruim.
- Geração de processo zumbis.

Além dos itens já citados acima, a desvantagem do Hadoop é ser um sistema de arquivos baseado em write-once, ou seja, não existe operações de *append* de arquivos, não sendo possível modificações nestes. Outra desvantagem é o map reduce, pois trata-se de uma linguagem para criar jobs de análise de dados. É necessário um conhecimento prévio do que se deseja analisar, pois o Hadoop apenas fornece as ferramentas para análise, não as faz de forma automática. Portanto deve se levar em consideração o tempo de desenvolvimento dos jobs de map reduce ao realizar uma implementação de Hadoop.

Em contrapartida, as vantagens da tecnologia são disponibilizar o dado com redundância, tentar manter o dado próximo ao local onde ele será usado para evitar uso de rede (sobrecarga nos switches centrais) e realizar o correlacionamento de dados semelhantes.

Por outro lado, pensando nas soluções que compõem o projeto EduCloud, podemos observar ao comparar o Hadoop com outras soluções de storages disponibilizadas pelas tecnologias de cloud, que a princípio, o Hadoop oferece um serviço semelhante ao Swift do projeto OpenStack. O Swift é a solução de armazenamento e gerenciamento de dados que compõem o framework fornecido pelo OpenStack, cujo propósito é armazenar os dados que serão utilizados pelas máquinas virtuais. Logo, seu objetivo de disponibilizar o dado para a máquina virtual, vem de encontro com a solução proposta pelo Hadoop.

Considerando que o Projeto EduCloud tenha um deployment de Hadoop em sua composição, os requisitos de hardware são, no mínimo:

- 2 máquinas com a função de master;
- No mínimo, 4 máquinas com a função de slaves, sendo que estas máquinas deverão possuir storage com tamanho considerável, pois serão nelas que os dados serão armazenados.
- Para efeito de análise do algoritmo de localizing, é interessante que essas 6 máquinas sejam ligadas em 2 switches distintos; para que possamos analisar a eficácia do localizing e do algoritmo de replicação.

A documentação descrita no link http://hadoop.apache.org/common/docs/stable é indicada para fazer um deployment de uma versão single node, e outra versão, mais elaborada, de cluster setup. Com esta documentação, parece ser suficiente para realizar o deployment de hadoop em um ambiente para testes.

Além disso, consultando a wiki do projeto Hadoop (http://wiki.apache.org/hadoop/#Setting_up_a_Hadoop_Cluster), existem outros tutoriais de deployment, específicos para determinadas distribuições de linux, e determinados sistemas operacionais.

E os passos necessários para deployment são:

Os pré-requisitos para instalação são os seguintes:

- Java 1.6 ou posterior;
- ssh;
- rsync

Para um modo de operação single node:

Satisfeitos os requisitos acima, é necessário fazer o download e descompactar o conteúdo.

Dentro do diretório descompactado, alterar o arquivo: conf/hadoop-env.sh para setar a variável de ambiente JAVA_HOME de acordo com o java instalado na máquina.

Após isso, basta seguir os seguintes passos:

- criar um diretório;
- copiar os arquivos xml dentro da pasta conf para o diretório criado;
- Executar o seguinte comando:
    - *bin/hadoop jar hadoop-examples-*.jar grep input output 'dfs[a-z.]+'*

Para um modo de operação cluster, além dos passos acima, é necessário executar os seguintes passos:

Definir o seguinte grupo de máquinas:
- NameNode - 1 máquina - Master
- JobTracker - 1 máquina - Master
- DataNode & TaskTracker - Demais máquinas - Slaves

Todas devem conter o mesmo HADOOP_HOME (path para instalação do hadoop nas máquinas)

A configuração de cada grupo de máquinas deve ser feito de acordo com o seguinte link:
http://hadoop.apache.org/common/docs/stable/cluster_setup.html

Configuração de mapreduce e localizing

O map reduce do hadoop é na realidade um framework para programação de algoritmos de análise de grandes quantidades de dados. Uma analogia do map reduce para o mundo dos bancos de dados é que o map reduce se refere à query de busca do banco de dados.
A estrutura do map reduce funciona da seguinte forma:

- no primeiro estágio é feito o mapeamento dos dados através de uma estrutura de par, chave e valor;
- no segundo estágio é feito a redução dessa estrutura, agregando pares que tenham a mesma chave.

Logo, o map reduce atua sobre a análise dos dados, não sobre o dado em si; de forma que o resultado da análise é mais compacto.

Quanto ao localizing, é inerente à arquitetura do hadoop em distribuir os dados de acordo com o fator de replicação estipulado para os dados, e para aproximar o dado dos nós que irão processá-lo.

Instanciar Brain Honey via Open Nebula

- Converter as imagens dos servidores do brain honey, de vmware para kvm:
  - *qemu-img convert /path/to/imagem.vmdk -O raw image.raw*

- Adicionar imagens ao repositorio de imagens LVM:
  - *Criar uma partição lvm para cada imagem das VMs (ir ao repositorio de imagem):*

    - *lvcreate --name $IMG_NAME --size $SIZE osdc1*
    - *dd if=/path/to/image of=/dev/osdc1/$IMG_NAME*

- Criar os arquivos Vaal.template, Atlantico.template e Lena.template (em anexo) para instanciação das máquinas virtuais via Open Nebula.

- Instanciar as máquinas virtuais na seguinte ordem:
  - *onevm submit /path/to/Vaal.template*

  - *onevm submit /path/to/Atlantico.template*

  - *onevm submit /path/to/Lena.template*

Voltar

Files

Vaal.template	552 Bytes	21 Mar 2012	Marco Gomes
Atlantico.template	389 Bytes	21 Mar 2012	Marco Gomes
Lena.template	354 Bytes	21 Mar 2012	Marco Gomes

Instanciar Brain Honey via VMWare

- Esta atividade foi desenvolvida diretamente no repositório de imagens
- Foram realizados os seguintes passos:
- Instalar o vmware workstation
- Importar as máquinas virtuais do Brain Honey: Vaal, Atlantico e Lena;
- Criar os novos adaptadores virtuais de rede, para uma rede interna (sem acesso à internet)
- Adaptar os recursos necessários para cada VM aos recursos disponíveis pela máquina
- Instanciar as máquinas virtuais

Voltar

Instanciar uma Máquina Virtual com Acesso à Internet

- Para realizar esta atividade é necessário:
  - *Criar uma partição lvm com a imagem da VM (ir ao repositorio de imagem):*
    - *lvcreate --name $IMG_NAME --size $SIZE osdc1*
    - *dd if=/path/to/image of=/dev/osdc1/$IMG_NAME*
  - *Submeter o arquivo vm.template para o open nebula com o seguinte comando (ir ao gerenciador do open nebula):*
    - *onevm submit vm.template*

Voltar

Files

vm.template	286 Bytes	21 Mar 2012	Marco Gomes

Procedimentos IPMI

Todos os comandos abaixo são encontrados no pacote **freeipmi-tools**, uma implementação do projeto GNU, para o protocolo IPMI.

Nota: Todos os commandos do **freeipmi-tools** aceita a seguinte syntax para múltiplos hosts: bmc[00-127].

Obtendo um console serial (SOL - Serial over Lan):

```
$ ipmi-console -h bmc01 -u admin -P -W solpayloadsize
```

Nota: Para sair da console digite: &.

Ligando a máquina (Wakeup on Lan)

```
$ ipmipower -h bmc01 -u admin -P --on
```

Desligando a máquina

```
$ ipmipower -h bmc01 -u admin -P --off
```

Reiniciando a máquina:

```
$ ipmipower -h bmc01 -u admin -P --reset
```

Configurar para que o próximo boot seja via PXE:

Nota: Nem todas as máquinas suportam este comando.

```
$ ipmi-chassis-config -h bmc01 -u admin -P  --commit -e "Chassis_Boot_Flags:Boot_Flags_persistent=No"
$ ipmi-chassis-config -h bmc01 -u admin -P  --commit -e "Chassis_Boot_Flags:Boot_Device=PXE"
```

Obter o power status:

```
$ ipmi-chassis -h bmc01 -u admin -P --get-status
```

Obter informações dos sensores:

```
$ ipmi-sensors -h bmc01 -u admin -P --verbose
```

Obter informações da placa mãe/produto:

```
$ ipmi-fru -h bmc01 -u admin -P --verbose
```

Trocar senha

Primeiro precisa-se obter o arquivo base de configuração para ser usado como modelo.

```
$ bmc-config -h bmc01 -u admin -P -o -n bmc01.conf
```

Nesse arquivo, procurar a seção correspondente ao usuário cuja senha precisa ser alterada e substituir o campo

```
## Password
```

por

```
Password                    senha
```

E subir o arquivo com as alterações

```
$ bmc-config -h bmc01 -u admin -P -c -n bmc01.conf
```

Nota_1: Ao subir, tem-se como resposta erros de "Invalid/Unsupported Config". Apenas ignore. A mudança de senha funciona.
Nota_2: Cuidado com a identação.

Configuração da BIOS

Para que a configuração de SOL (Serial over Lan) funcione, Na BIOS, em **Server -> Remote Access Configuration**, deve estar:

```
*********************************************************
* Remote Access                 [Enabled]          *
*                                                  *
* Serial port number            [COM2]             *
*     Base Address, IRQ         [2F8h, 3]          *
* Serial Port Mode              [57600 8,n,1]      *
* Flow Control                  [Hardware]         *
* Redirection After BIOS POST   [Always]           *
* Terminal Type                 [VT-UTF8]          *
*********************************************************
```

Liberar uma Máquina Virtual

- Para liberar uma máquina virtual, basta executar o seguinte procedimento:
  - *Descobrir o ID da VM usando o comando onevm list*
  - *Executar o comando de shutdown da VM:*
    - *onevm shutdown VM_ID*

Voltar

Máquina Real com HT desabilitado

- Máquina Educloud05:

Ambiente:

- 8 Processadores de 2.67 GHz;
- 6 GB de Ram;
- 90% de Uso de memória:
- QxP: 1x8;

Resultados:

- Valor de NB ótimo: 120
- Número de Operações de Ponto Flutuante por Segundo: 7.66826e+01 GFlops;
- Gráfico de Tempo vs NB vs Número de Operações:
  https://ncc.unesp.br/its/attachments/226/NBvsTimevsGflops.png

Máquina Real com HT Vs Máquina Real sem HT

Sem HT

- Máquina Educloud05:

Ambiente:

- 8 Cores de 2.67 GHz;
- 6 GB de Ram;
- 90% de Uso de memória:
- QxP: 1x8;

Resultados:

- Valor de NB ótimo: 120
- Número de Operações de Ponto Flutuante por Segundo: 7.66826e+01 GFlops;
- Gráfico de Tempo vs NB vs Número de Operações:
  https://ncc.unesp.br/its/attachments/226/NBvsTimevsGflops.png

Com HT

- Máquina Educloud06:

Ambiente:

- 16 Cores de 2.67 GHz;
- 6 GB de Ram;
- 90% de Uso de memória:
- QxP: 1x16;

Resultados:

- Valor de NB ótimo: 96
- Número de Operações de Ponto Flutuante por Segundo: 6.71501e+01 GFlops;
- Gráfico de Tempo vs NB vs Número de Operações:
  https://ncc.unesp.br/its/attachments/227/NBvsTimevsGflops.png

Comparação

- **Gráfico Comparativo entre Máquina com HT e sem ht habilitado:**
  https://ncc.unesp.br/its/attachments/228/NBvsTimevsGflops-Comparison.png

---

Discussão

- **O uso do HT claramente piora o resultado;**

16/04/2012

Executou:
- Iniciou Estudos de soluções alternativas ao hadoop.

- Após queda de energia, subiu as máquinas do EduCloud;

- Observou que ocorreram erros nos discos da VM que estavam de pé, que pode ter comprometido o trabalho do pessoal do FDE ao alterar a senha das VMs.

Próximos Passos:
- Tentar recuperar as informações das VMs

- Continuar estudos de soluções alternativas ao hadoop.

Blocker/Dependencia:

- Nenhuma

Voltar

17/04/2012

Executou:
- Tentou recuperar as imagens do OS da Lena e da Vaal, porém não obteve sucesso.

Próximos Passos:
- Entrar em contato com FDE para solicitar novo procedimento.

- Continuar estudos do hadoop

Blocker/Dependencia:

- Nenhuma

Voltar

18/04/2012

Executou:
- Está subindo o ambiente Brain Honey para disponibilizar ao pessoal do FDE e MSTech para que possam alterar novamente as credenciais;

- Finalizou a atividade de Estudar as Tecnologias semelhantes ao Hadoop;

- Finalizou a atividade de verificar a licença do hadoop;

- Executando a atividade referente às vantagens e desvantagens do hadoop para o projeto.

Próximos Passos:
- Entrar em contato com FDE para solicitar alteração das credenciais.

- Continuar estudos do hadoop

Blocker/Dependencia:

- Nenhuma

Voltar

19/04/2012

Executou:
- Subiu o ambiente do Brain Honey para disponibilizar ao pessoal do FDE poder fazer a troca de credenciais;

- Enviou e-mail para o Arnaldo da MSTech com os endereços do hosts solicitando troca de credenciais.

Próximos Passos:
- Continuar estudos do hadoop

Blocker/Dependencia:

- Aguardando resposta do Arnaldo

Voltar

20/04/2012

Executou:
- Finalizou a análise de viabilidade do hadoop (vantagens, desvantagens e se faz sentido para o projeto)

- Iniciou a atividade de verificar os conflitos do hadoop com outras tecnologias envolvidas no projeto

Próximos Passos:

- Continuar estudos do hadoop

Blocker/Dependencia:

- Aguardando resposta do Arnaldo

Voltar

27/04/2012

Executou as seguintes:

- Finalizou a tarefa de prova de conceito do OCCI.
- Gerar relatório

Próximos passos:

- Continuar a geração de relatório

Blocker/Dependencia

- Nenhuma

23/04/2012

Executou:

- Finalizou a análise de viabilidade do hadoop (vantagens, desvantagens e se faz sentido para o projeto)
- Iniciou e finalizou a atividade de estudo da arquitetura da solução (conlitos com outras tecnologias do projeto);
- Iniciou e finalizou a atividade de levantamento de uso da tecnologia (quem a está utilizando e quão larga é a adoção);
- Iniciou e finalizou a atividade de continuidade de uso da tecnologia (estudo de roadmaps futuros);
- Iniciou e finalizou a atividade de estudo da documentação do projeto (existência de documentação para deployment);
- Iniciou a atividade de ler documentação de deployment (listar os passos necessários para um deployment).

Próximos Passos:

- Continuar leitura de documentação de deployment.

Blocker/Dependencia:

- Aguardando resposta do Arnaldo

Voltar

24/04/2012

Executou:

- Finalizou a atividade de ler documentação de deployment (listar os passos necessários para um deployment);
- Iniciou e finalizou a atividade de configuração de map reduce e localizing;
- Iniciou e finalizou a atividade de Levantamento de requisitos do OCCI.
- Iniciou a atividade de verificar se tecnologia atende aos requisitos.
- Reenviou email para o Arnaldo.

Próximos Passos:

- Continuar atividade de verificar se tecnologia atende aos requisitos.

Blocker/Dependencia:

- Aguardando resposta do Arnaldo

Voltar

25/04/2012

Executou:

- Finalizou a atividade de verificar se tecnologia atende aos requisitos.
- Iniciou e finalizou a atividade de verificar tecnologias semelhantes
- Iniciou e finalizou a atividade de análise de viabilidade da tecnologia
- Iniciou e finalizou a atividade de checar licença do projeto
- Iniciou a atividade de verificar de estudo da arquitetura da solução.
- recebeu resposta do arnaldo

Próximos Passos:

- Continuar atividade de estudo da arquitetura da solução

Blocker/Dependencia:

- Aguardando mudança das credenciais.

Voltar

26/04/2012

Executou:

- Finalizou a atividade de verificar de estudo da arquitetura da solução.
- Iniciou e finalizou a atividade de levantamento de uso da solução
- Iniciou e finalizou a atividade de continuidade da solução

Próximos Passos:

- Continuar atividade de continuidade da solução

Blocker/Dependencia:

- Aguardando mudança das credenciais.

Voltar

Marco

Voltar

Matriz QxP quadrada vs Matriz QxP Linear

Com HT:

Matriz QxP Quadrada:

- Máquina Educloud06:

Ambiente:

- 16 Processadores de 2.67 GHz;
- 6 GB de Ram;
- 90% de Uso de memória:
- **QxP: 2x8**;

Resultados:

- Valor de NB ótimo: 104
- Número de Operações de Ponto Flutuante por Segundo: 6.80169e+01 GFlops;
- Gráfico de Tempo vs NB vs Número de Operações:
  https://ncc.unesp.br/its/attachments/229/NBvsTimevsGflops-2x8.png

Matriz QxP Linear

- Máquina Educloud06:

Ambiente:

- 16 Processadores de 2.67 GHz;
- 6 GB de Ram;
- 90% de Uso de memória:
- QxP: 1x16;

Resultados:

- Valor de NB ótimo: 96
- Número de Operações de Ponto Flutuante por Segundo: 6.71501e+01 GFlops;
- Gráfico de Tempo vs NB vs Número de Operações:
  https://ncc.unesp.br/its/attachments/227/NBvsTimevsGflops.png

Comparação

- **Gráfico Comparativo entre Máquina com razão entre QxP mais 'quadrado' e 'linear':
  https://ncc.unesp.br/its/attachments/230/NBvsTimevsGflops-Comparison1x16HTvs2x8HT.png**

Sem HT:

Matriz QxP Quadrada:

- Máquina Educloud05:

Ambiente:

- 8 Processadores de 2.67 GHz;
- 6 GB de Ram;
- 90% de Uso de memória:
- QxP: 2x4;

Resultados:

- Valor de NB ótimo: 136
- Número de Operações de Ponto Flutuante por Segundo: 7.80689e+01 GFlops;
- Gráfico de Tempo vs NB vs Número de Operações:
  https://ncc.unesp.br/its/attachments/231/NBvsGflopsvsTime.png

Matriz QxP Linear:

- Máquina Educloud05:

Ambiente:

- 8 Processadores de 2.67 GHz;
- 6 GB de Ram;
- 90% de Uso de memória:
- QxP: 1x8;

Resultados:

- Valor de NB ótimo: 120
- Número de Operações de Ponto Flutuante por Segundo: 7.66826e+01 GFlops;
- Gráfico de Tempo vs NB vs Número de Operações:
  https://ncc.unesp.br/its/attachments/226/NBvsTimevsGflops.png

Discussão

- **O uso de matrizes QxP mais quadradas causa impacto no resultado:**
  - **Tempos de execução foram menores;**
  - **Quantidade de operações executadas também foram maiores;**

Files

NBvsGflopsvsTime.png	9.37 KB	22 Apr 2013	Raphael Cóbe

Monitorar Continuamente Duas Métricas de Status dos Servidores

Lista de subtasks:

Implantação #497: Investigar Balanceador de Carga

Nesta tarefa, foi feita uma pesquisa sobre os atuais schedulers para o Open Nebula, entre eles, foram pesquisados: SVMSched e o Haizea.

Durante a pesquisada realizada, estes schedulers não implementavam recursos necessários para o projeto EduCloud, como monitoramento constante de métricas, instanciação e liberação de máquinas virtuais, dependendo dos resultados destas métricas.

Foi decidido então, que um "monitor/daemon/balanceador" seria desenvolvido, a este chamamos de **educloud-mon**.

Implantação #498: Desenvolver o Monitor e Balanceador de Carga (EduCloud-Mon)

O *educloud-mon* é atualmente um monitor do OpenNebula, porém com possibilidade para extender-se ao OpenStack, por exemplo. Ele foi feito em Python e utiliza-se da biblioteca: python-oca, para as tarefas de comunicação com o OpenNebula.

O *educloud-mon* precisa ser executado como root, em qualquer máquina na mesma rede dos nós do OpenNebula. Ele foi desenvolvido de modo que permita ao administrador configurar métricas e parâmetros, tornando ele um daemon bem flexível. Seu principal arquivo de configuração é o educloud-mon.conf.

Atualmente, em nosso ambiente de testes, o *educloud-mon* está funcionando na máquina **200.145.146.198**, e encontra-se instalado em /usr/local/bin/educloud-mon/.

Aqui você encontra um exemplo de um arquivo de configuração [[educloud-mon.conf]].

Além do arquivo de configuração você pode passar alguns parâmetro por meio de argumentos ao daemon. Para maiores informações, execute:

```
./educloud-mon.py --help
```

O *educloud-mon*, quando determina que precisa instanciar uma nova máquina virtual, lê o template da máquina virtual, e envia para o OpenNebula o comando para instanciar uma nova máquina virtual. Para isso, os templates das máquinas virtuais (arquivos de configuração), precisa estar na mesma máquina que o *educloud-mon*, e você deve configurar o parâmetro template em seu arquivo de configuração:

```
# Relative path to template to start when metrics get values >= top_thresolds
# This path is relative to the home of the user oneadmin on OpenNebula
# controller. This hardcoded path is necessary because we are using OpenNebula
# 2.2.1 version.
template = "/usr/local/bin/educloud-mon/templates/teste.template"
```

Sistema de envio de alertas

O *educloud-mon* conta também com um sistema simples para envio de alertas por email. Caso alguma ação durante a sua execução falhe, um email de alerta será enviado.

Para isso configure a seção [notification] no educloud-mon.conf:

```
# Settings for email notification system
[notification]

# If enabled, emails will be sent to administrators when errors occur.
# Default is true.
enable_notifications = 0

# SMTP Server to use
smtp_server = 127.0.0.1

# SMTP Port
smtp_port = 25

# Use SMTP Authentication ?
# Default is true.
smtp_auth = 0

# SMTP User for authentication
smtp_user = user

# SMTP Pass for authentication
smtp_pass = password
```

```
# Mail From
mail_from = "educloud-mon <educloud-mon@ncc.unes.pbr"

# Mail to
mail_to = "admin@ncc.unesp.br"
```

Implantação #499: Configurar Balanceador de Carga de Forma que Monitore duas Métricas de VMs Open Nebula

O *educloud-mon*, monitora N métricas passadas por arquivo de configuração (educloud-mon.conf). Até o momento do desenvolvimento, ele executa duas ações: criar uma nova máquina virtual e liberar uma nova máquina virtual, quando o top thresold é atingido e o bottom thresold, respectivamente.

Métricas são passadas via arquivo de configuração bem como seus valores de top e bottom. As métricas ficam em um diretório "scripts/" idependente do *educloud-mon*.

Exemplo de configuração de métricas e seus valores no arquivo de configuração do *educloud-mon*:

```
metrics = {
        'cpu-load': {
          'top_thresold': '70',
          'bottom_thresold': '2',
          'script': '/usr/local/bin/educloud-mon/scripts/web-cpu-usage.py',
          'check_interval': '2',
          'average_checks': '5',
        },
        'active-users': {
          'top_thresold': '20',
          'bottom_thresold': '0',
          'script': '/usr/local/bin/educloud-mon/scripts/web-active-users.py',
          'check_interval': '2',
          'average_checks': '5',
        },
    }
```

No exemplo acima, o *educloud-mon* irá monitorar duas métricas: "cpu-load" e "active-users". O parâmtro "script" de cada métrica, informa qual o script irá ser executado para coletar esta métrica.

Cada métrica tem o seus valores de top_thresold e bottom_thresold. No exemplo acima, caso o script retorne um valor superior a 70, o top_thresold é atingido. Caso ele retorne um valor abaixo de 2, o bottom_thresold é atingido.

A eurística para determinar se uma nova máquina será instanciada ou liberada é a seguinte:

- Caso alguma métrica tenha atingido o top_thresold, uma nova máquina será instanciada;
- Caso nenhuma métrica tenha atingido o top_thresold e nenhuma métrica tenha atingido o bottom_thresold, nada será feito;
- Caso nenhuma métrica atingiu o top_thresold e alguma métrica tenha atingido o bottom_thresold, uma máquina será liberada;

OCCI

Objetivo

O escopo deste documento trata de avaliar o uso de soluções de protocolos de comunicação para interoperabilidade entre clouds, avaliando a viabilidade de uma determinada solução para o contexto do projeto Educloud, e se o projeto dispõe de informações suficientes para realizar tal análise e um possível deployment em nosso ambiente de desenvolvimento.

Projeto EduCloud

O projeto Educloud é um projeto para a implementação de uma infraestrutura de cloud computing para a secretaria da educação do estao de São Paulo.

OCCI

O OCCI é um protocolo para realizar interface entre o usuário e as soluções de cloud de uma forma uniforme. Com ela é possível montar um sistema centralizado de gerenciamento de diferentes soluções de cloud, oferecendo dessa forma interoperabilidade e simplicidade de uso. A especificação do OCCI é desenvolvida e mantida pelo Open Grid Forum e disponibilizada para que seja implementada nas soluções de cloud, assim como em ferramentas para realizar validações da implementação de acordo com a especificação.

O OCCI em específico apresenta-se como uma soluçao de protocolo para interoperabilidade de cloud. Até o presente momento não foram encontradas soluções que servem o mesmo propósito. Existem soluções na mesma linha, mas com propósitos diferentes, e são listadas a seguir:

- Amazon EC2: A função é para administrar a cloud oferecida pela Amazon, mas algumas soluções de cloud oferecem suporte para trabalhar com soluções híbridas (publicas + privada);
- CDMI (Cloud Data Management Interface): É uma interface para trabalhar com manipulação de dados de Cloud;
- Open Data Center Alliance: É um consórcio de empresas com intuito de definir standards para cloud computing.

Ela é uma implementação aberta, e tem seus aspectos legais de uso descritos no link a seguir:

http://occi-wg.org/about/legal

Esta tecnologia é implementada pelas principais soluções de cloud open source existentes e existem ferramentas de validação da implementação de acordo com este site: http://occi-wg.org/community/implementations

A questão da adoção varia da necessidade das empresas de possuírem soluções interoperáveis, e até o presente momento não foi encontrada uma referência que pudesse quantificar tal informação.

Dentro do site do projeto, não há uma referência explicita de roadmap para o OCCI. Porém, eles possuem uma wiki https://forge.ogf.org/sf/projects/occi-wg com informações sobre o processo de desenvolvimento, bugfixes e discussões a respeito do projeto. Acredito que por ser uma especificação, deve haver continuidade na elaboração de novas versões da especificação, porém é apenas uma suposição baseada no desenvolvimento que cloud computing está tendo no mundo.

Uso no Projeto Educloud

Os requisitos para o projeto Educloud no contexto de um protocolo de gerenciamento de cloud são:

- Gerenciamento de VMs (listar, criar, suspender, resumir e deletar)
- Interoperabilidade entre soluções de cloud

Posto isso, o OCCI é um protocolo desenvolvido para ser interoperável, oferecer um conjunto de soluções para gerenciamento de cloud tanto em situações de IaaS, PaaS e SaaS. Além disso, ele oferece uma API com alto nível de extensibilidade.

O uso de OCCI é importante para o projeto pois irá adicionar uma camada de interoperabilidade entre as soluções de cloud existentes e com isso teremos a possiblidade de controlar diferentes tipos de soluções através de uma interface centralizada.
As vantagens para o projeto são as já citadas acima, e além disso, a possibilidade da interface poder ser extendida para adicionar funcionalidades de acordo com nossas necessidades.
Uma desvantagem é a provável variação na implementação da interface nas soluções de cloud existentes, podendo culminar em necessidade de retrabalho para nivelar as soluções de cloud em um mesmo patamar de usabilidade da interface.

O uso do OCCI conflita com as soluções de cloud disponíveis pois ela entra com o papel de substituir as interfaces de gerenciamento das soluções de cloud. Mas isso não é visto como um aspecto negativo, pois a idéia de substituir estas interfaces é justamente sua principal função para ser possível também integrar as soluções e centralizar o comando em uma única interface.

A documentação do OCCI no site do projeto é referente à especificação do mesmo. Para o contexto do projeto, a documentação procurada se refere às implementações de OCCI nas soluções de cloud. Logo, temos as seguintes documentações:

- Open Nebula: http://opennebula.org/documentation:rel3.4:occidd
- Open Stack: http://wiki.openstack.org/occi
- Eucalyptus: http://occi-wg.org/2010/12/14/occi-in-eucalyptus/

Pensando em deployment de OCCI não é efetivamente o termo correto a ser utilizado. Como nosso foco é em interoperabilidade, para esta tecnologia é necessário estudar as implementações de OCCI em cada uma das soluções de cloud que iremos utilizar para poder ser criado uma solução que consiga gerenciar as VMs de forma uniforme entre as clouds.

No opennebula, o OCCI é implementado de forma a ser um protocolo sobre HTTP operando através de operações de GET, POST, PUT e DELETE.

No openstack, o OCCI também é implementado sob HTTP, porém aparenta ser uma implementação mais completa que no opennebula, oferecendo uma gama maior de recursos para gerenciamento da VM.

No eucalyptus não foi encontrado documentação falando a respeito da implementação de OCCI.

OpenFLOW Controllers

Tabela Resumo:

Nome	Linguagem	Licença	Tamanho do Pacote	Features	URL	Versão do OpenFlow
Nox	C++	GPL V3	258 Kb (Fonte)	Provides a C++ OpenFlow 1.0 API Provides fast, asynchronous IO Includes sample components for: Topology discovery Learning switch Network-wide switch	http://www.noxrepo.org/pox/documentation/	1.0, 1.3 via https://github.com/CPqD/nox13oflib
Pox	Python	GPL V3	1.1 Mb (Fonte)	“Pythonic” OpenFlow interfacePOX Performance Graph Reusable sample components for path selection, topology discovery, etc. “Runs anywhere” – Can bundle with install-free PyPy runtime for easy deployment Specifically targets Linux, Mac OS, and Windows Supports the same GUI and visualization tools as NOX Performs well compared to NOX applications written in Python	http://www.noxrepo.org/nox/documentation/	1.0
Beacon	Java	GPL v2	12.2 Mb (Fonte) 17 Mb (Binários + Dependências)	Stable - Beacon has been in development since early 2010, and has been used in several research projects, networking classes, and trial deployments. Beacon currently powers a 100-vswitch, 20-physical switch experimental data center and has run for months without downtime. Cross-platform - Beacon is written in Java and runs on many platforms, from high end multi-core Linux servers to Android phones. Dynamic - Code bundles in Beacon can be started/stopped/ref	*https://openflow.stanford.edu/display/Beacon/Guides*	*1.0*

				reshed/installed at runtime, without interrupting other non-dependent bundles (ie replace your running Learning Switch application without disconnecting switches). Rapid Development - Beacon is easy to get up and running. Java and Eclipse simplify development and debugging of your applications. Fast - Beacon is multithreaded, check out performance benchmarks. Web UI - Beacon optionally embeds the Jetty enterprise web server and a custom extensible UI framework Frameworks - Beacon builds on mature Java frameworks such as Spring and Equinox (OSGi)		
Maestro	Java	LGPL v2.1	117 Kb (Binários) 262 Kb (Fonte)	Developed in Java (both the platform and the components) - Highly portable to various operating systems and architectures. Multi-threaded - Takes full advantage of multi-core processors. Currently provides the control components for realizing either a learning switch network, or a routed network using OpenFlow switches.	http://code.google.com/p/maestro-platform/wiki/MaestroTutorial	1.0
Ryu	Python	Apache V2.0	12.2 Mb (Fonte)	OpenStack supports Ryu. You can create tens of thousands of isolated virtual networks without using VLAN. You manage virtual networks, OpenStack notifies of Ryu and Ryu configures OpenvSwitches properly. You can try Now. Ryu's pre-configured VM	*https://github.com/osrg/ryu/wiki*	*1.0, 1.2, 1.3 e Nicira Extensions*

				image file enables you to easily set up multi-node OpenStack environment. We try Ryu with software OpenFlow Switch (OpenvSwitch) and bunch of hardware OpenFlow Switches.		
Floodlight	Java	Apache V2.0	46 Mb (Fonte)	- Offers a module loading system that make it simple to extend and enhance. - Support for OpenStack cloud orchestration platform - OpenStack Quantum plugin. - It is the core of 'BigSwitch' (BigSwitch open-sourced FloodLight). - Easy to set up with minimal dependencies. - Supports a broad range of virtual- and physical- OpenFlow switches. - Can handle mixed OpenFlow and non-OpenFlow networks – it can manage multiple “islands” of OpenFlow hardware switches. - Designed to be high-performance – is the core of a commercial product from Big Switch Networks. - API REST. - Enterprise class (production). - Floodlight VM*. It includes floodlight running by default and Mininet. - Nos repositórios do Ubuntu. - *Avior (GUI open-source).	http://www.projectfloodlight.org/documentation/	1.0
BigSwitch	java	closed-source	--	- OpenStack - based on Beacon - targets production networks - user-friendly cli	http://www.bigswitch.com/	
MUL	C	GPLv2	16.2 MB (src)	- muli-threaded infrastructure - multi-level north bound interface for	*http://sourceforge.net/projects/mul/*	*1.0, 1.1, 1.2*

				hooking up applications. - designed for performance and reliability (mission-critical networks). - core of a commercial controller from Kulcloud Networks - Openstack Support through Quantum plugins		
Jaxon	java	--	109 KB (src)	- NOX-dependent - provides a thin interface with the NOX - possible to directly access from Java program to C native ones.	http://jaxon.onuos.org/	--
NDDI-OESS	??	Apache V2.0	??	-- set of programs - OESS provides sub-second circuit provisioning - automatic circuit failover - per-interface permissions - automatic per-VLAN statistics - simple and user friendly web-based user interface - web services API	http://code.google.com/p/nddi/wiki/README	1.0
NodeFlow	javascript	MIT-License	?	- provides an asynchronous library over JavaScript for server side programming which is perfect for writing network based applications (ones that don't require an excessive amount of CPU). - Uses on OFLIB-NODE Library.	http://garyberger.net/?p=537	1.0
Helios	C	closed-source	?	- targetting researchers - programmatic shell (integrated experiments)	http://www.nec.com/	1.0
SNAC	Python e C++	GPLv3	?	- targets production networks - based on NOX 0.4 - flexible policy definition language - user-friendly interface (configuration + monitoring)	http://www.openflow.org/wp/snac/	1.0
HP VANSC	?	closed	?	- dinamic control plane - full support for OpenFLOW protocol - Open API	*?*	*?*

				enable third part application		
IBM PNC	?	closed	?	- unlimited virtual machine mobility - enterprise class (production) - automatically and continuous discovers network topology	?	?

Listas

- http://tech.saroscorner.com/2012/12/list-of-openflow-controllers-for-sdn.html

Hypervisor

To a controller: ordinary switch. To a switch: a controller. Must stay on top of switches and under the controller.

FlowVisor

- transparent proxy between switches and controller
- slices of network
- network virtualization platform
- hypervisor
- isolate services at slices (groups?)

Framework

Trema

Trema is a full-stack, easy-to-use framework for developing OpenFlow controllers in Ruby and C.

Floodlight

Sobre o código

Linguagem

Linguagem utilizada para o desenvolvimento.

Licença

Licença em que o código foi liberado. A preferência é por licenças livres, facilitando assim o estudo e desenvolvimento de aplicações ou até mesmo de correções de problemas.

Suporte a versões do OpenFLOW

- Quais são as versões de OpenFlow suportadas pelo controlador / framework.
- Como isso é implementado

Requisitos

- pacotes a serem instalados
- espaço de armazenamento exigido
- outros recursos necessários ao funcionamento do controlador.

Independência de linguagem de programação

Quais linguagens são suportadas na implementação de aplicações e componentes.

Modularidade

Quanto e que tipo de modularidade o controlador oferece.

Repositório

Onde encontrar o código.

Recursos básicos do controlador

Recursos oferecidos pelo controlador sem a necessidade de aplicações, como por exemplo:

- GRE tunneling
- VLAN support

Aplicações

Componentes disponíveis que acrescentam ferramentas ao controlador, como por exemplo:

- topology discovery
- MAC based segregation

Testes

Existência de testes de unidades no codigo (scripts ou códigos), geralmente em uma pasta em separado, que validam e testam o código desenvolvido.

Disponibilidade nas distribuições GNU/Linux

Disponibilidade e facilidade de instação nas distribuição GNU/Linux relevantes atuais.

- disponível nos repositórios da distribuição?
- versão dos pacotes disponíveis
- ênfase nas distros Debian e Ubuntu.

Documentação

Nível de documentação do controlador, incluindo tutoriais e exemplos (tanto de estruturas quanto de aplicações).

Suporte junto ao OpenStack

Como o ambiente estruturado se baseia no OpenStack, é interessante saber qual é o tipo de suporte oferecido a essa ferramenta.

Quem usa

Quem está por trás

Empresas, consórcio, comunidade, universidade, ...

último release / commit

Uma das formas de se ter noção de quanto o desenvolvimento ainda está ativo é procurar pela data da última atualização de código no repositório.

Outras características

Características que não se adequam a nenhuma das anteriores.

- Configure an HP OpenFlow Switch for Floodlight
  http://networkstatic.net/configure-an-hp-openflow-switch-for-floodlight/

NOX

OBSERVAÇÃO

Existem duas versões do NOX. A versão antiga e descontinuada foi renomeada para NOX-Classic. Porém todos os materiais que foram produzidos para essa versão tratam-na com o nome que era corrente, ou seja, apenas por NOX. Isso causa um problema sério: em qualquer conteúdo referente ao NOX não se consegue saber se é sobre a versão antiga ou a nova, a menos que seja dito explicitamente ou que o NOX seja chamado de 'new NOX'.

Sobre o código

Linguagem

C++

Licença

GPLv3

Suporte a versões do OpenFLOW

OpenFlow 1.0

Requisitos

- pacotes a serem instalados
- espaço de armazenamento exigido
- outros recursos necessários ao funcionamento do controlador.

Independência de linguagem de programação

Quais linguagens são suportadas na implementação de aplicações e componentes.

Modularidade

Quanto e que tipo de modularidade o controlador oferece.

Repositório

Onde encontrar o código.

Recursos básicos do controlador

Recursos oferecidos pelo controlador sem a necessidade de aplicações, como por exemplo:

- GRE tunneling
- VLAN support

Aplicações

- jsonmessenger: interact with NOX using JSON.
- messenger: interact with NOX.
- simple_cc_app
- simple_app
- snmp: SNMP get/set support in NOX More...
- Authenticator
- Bindings_Storage: mirrors the Authenticator's internal data structures within the NDB.
- discovery: LLDP discovery application for topology inference.
- hostip: store IP address of hosts.
- hosttracker: Track locations of host (ethernet)Track last n known locations of host attachment.
- trackhost_pktin: Tracks host location using packet in.
- lavi: Components that allows network state to be enquired.
- Monitoring: Collects and maintains switch and port stats for the network.
- datapathmem: Datapath memory to memorize datapath informationAlso join events of the datapath.
- linkload: tracks port stats of links and its loadRefers to links using its source datapath id and port.
- switchrtt: meaure RTT of switch in networkCan configure the interval of probe at runtime using the interval option (giving interval in seconds).
- flowroute_record: record of flow and associated routeDoes not handle detection of flow and route nor its expiration/deletion.
- routeinstaller: install route.
- simplerouting: routing with dependencies on authenticatorCan choose to set flow installed into flowroute_record with arguments
- Routing_module: utility component that can be called to retrieve a shortest path route and/or to set up flow entries in the network with Openflow actions to route a flow.
- switchstats: collects and maintains switch and port stats for the network.
- flowlog: records all flows in networkTo ensure flow removed is sent (in openflow-default.hh).

- rttlog: Logs switch RTTLogs every switchrtt's interval, i.e., probe all switches once.
- tablog: Component that logs tables of informationComponent dumps data into one of the tablog_dump format.
- test1
- test2
- Topology: the current network topology.
- User_Event_Log: The goal of the user_event_log is to let applications easily generate 'network event log' messages that include high-level network names.
- webserver

Testes

Existência de testes de unidades no codigo (scripts ou códigos), geralmente em uma pasta em separado, que validam e testam o código desenvolvido.

Disponibilidade nas distribuições GNU/Linux

Disponibilidade e facilidade de instação nas distribuição GNU/Linux relevantes atuais.

- disponível nos repositórios da distribuição?
- versão dos pacotes disponíveis
- ênfase nas distros Debian e Ubuntu.

Documentação

Nível de documentação do controlador, incluindo tutoriais e exemplos (tanto de estruturas quanto de aplicações).

Suporte junto ao OpenStack

Como o ambiente estruturado se baseia no OpenStack, é interessante saber qual é o tipo de suporte oferecido a essa ferramenta.

Quem usa

Público do controlador

Quem está por trás

Empresas, consórcio, comunidade, universidade, ...

último release / commit

Uma das formas de se ter noção de quanto o desenvolvimento ainda está ativo é procurar pela data da última atualização de código no repositório.

Outras características

Características que não se adequam a nenhuma das anteriores.

Overview

Controlador praticamente desenvolvido em conjunto com OpenFlow.

- Oferece uma API para OpenFlow em C++
- Oferece rápida e assincrona IO (entrada saída)
- Está disponível nas recentes distribuições Ubuntu (mas é possível instalar na Debian e RHEL)
- Inclui componentes exemplos:
  - Topology discovery
  - Learning switch
  - Network-wide switch

Versões

- NOX-Classic: deprecated. Suporte para C++ e Python.

- NOX (aka 'new NOX'): suporte apenas para C++. Código mais limpo. Poucos applicativos.

- NOX com suporte a OpenFlow 1.3 (em desenvolvimento, último commit 07/05/2013)
  https://github.com/CPqD/nox13oflib

NOX API reference

http://www.noxrepo.org/_/nox-classic-doxygen/index.html

(gerado pelo doxygen)

Repositório GitHub

https://github.com/noxrepo/nox

Documentação do NOX-Classic

Dizem que apesar de ultrapassado, a documentação contida no wiki do NOX-Classic ainda pode ajudar, então:
https://github.com/noxrepo/nox-classic/wiki

Tutoriais

- Create OpenFlow network with multiple PCs/NetFPGAs
  http://archive.openflow.org/wp/deploy-labsetup/

- NOX OpenFlow Controller Install on Ubuntu 12.04 (antigo já, talvez esteja no repositório)
  http://mytestbed.net/projects/openflow/wiki/Installing_a_NOX_controller
  http://networkstatic.net/nox-openflow-controller-install-on-ubuntu-12-04-precise-lts/

OpenDaylight

Sobre o código

Linguagem

Linguagem utilizada para o desenvolvimento.

Licença

Licença em que o código foi liberado. A preferência é por licenças livres, facilitando assim o estudo e desenvolvimento de aplicações ou até mesmo de correções de problemas.

Suporte a versões do OpenFLOW

- Quais são as versões de OpenFlow suportadas pelo controlador / framework.
- Como isso é implementado

Requisitos

- pacotes a serem instalados
- espaço de armazenamento exigido
- outros recursos necessários ao funcionamento do controlador.

Independência de linguagem de programação

Quais linguagens são suportadas na implementação de aplicações e componentes.

Modularidade

Quanto e que tipo de modularidade o controlador oferece.

Repositório

Onde encontrar o código.

Recursos básicos do controlador

Recursos oferecidos pelo controlador sem a necessidade de aplicações, como por exemplo:

- GRE tunneling
- VLAN support

Aplicações

Componentes disponíveis que acrescentam ferramentas ao controlador, como por exemplo:

- topology discovery
- MAC based segregation

Testes

Existência de testes de unidades no codigo (scripts ou códigos), geralmente em uma pasta em separado, que validam e testam o código desenvolvido.

Disponibilidade nas distribuições GNU/Linux

Disponibilidade e facilidade de instação nas distribuição GNU/Linux relevantes atuais.

- disponível nos repositórios da distribuição?
- versão dos pacotes disponíveis
- ênfase nas distros Debian e Ubuntu.

Documentação

Nível de documentação do controlador, incluindo tutoriais e exemplos (tanto de estruturas quanto de aplicações).

Suporte junto ao OpenStack

Como o ambiente estruturado se baseia no OpenStack, é interessante saber qual é o tipo de suporte oferecido a essa ferramenta.

Quem usa

Público do controlador

Quem está por trás

Empresas, consórcio, comunidade, universidade, ...

último release / commit

Uma das formas de se ter noção de quanto o desenvolvimento ainda está ativo é procurar pela data da última atualização de código no repositório.

Outras características

Características que não se adequam a nenhuma das anteriores.

Página: http://www.opendaylight.org/

OpenDaylight Developer Wiki
https://wiki.opendaylight.org/view/Main_Page

Tem a Intel como um dos membros, além de outros peixes grandes.

A linguagem utilizada é java.

Acredito q seja baseado no floodlight.

Características dos controladores OpenFlow

Descrição e documentação das características a serem analisadas nos controladores.

Overview

- Descrição geral
- página do projeto

Sobre o código

Linguagem

Linguagem utilizada para o desenvolvimento.

Licença

Licença em que o código foi liberado. A preferência é por licenças livres, facilitando assim o estudo e desenvolvimento de aplicações ou até mesmo de correções de problemas.

Suporte a versões do OpenFLOW

- Quais são as versões de OpenFlow suportadas pelo controlador / framework.
- Como isso é implementado

Requisitos

- pacotes a serem instalados
- espaço de armazenamento exigido
- outros recursos necessários ao funcionamento do controlador.

Independência de linguagem de programação

Quais linguagens são suportadas na implementação de aplicações e componentes.

Modularidade

Quanto e que tipo de modularidade o controlador oferece.

Repositório

Onde encontrar o código.

Recursos básicos do controlador

Recursos oferecidos pelo controlador sem a necessidade de aplicações, como por exemplo:

- GRE tunneling
- VLAN support

Aplicações

Componentes disponíveis que acrescentam ferramentas ao controlador, como por exemplo:

- topology discovery
- MAC based segregation

Testes

Existência de testes de unidades no codigo (scripts ou códigos), geralmente em uma pasta em separado, que validam e testam o código desenvolvido.

Disponibilidade nas distribuições GNU/Linux

Disponibilidade e facilidade de instação nas distribuição GNU/Linux relevantes atuais.

- disponível nos repositórios da distribuição?
- versão dos pacotes disponíveis
- ênfase nas distros Debian e Ubuntu.

Documentação

Nível de documentação do controlador, incluindo tutoriais e exemplos (tanto de estruturas quanto de aplicações).

Suporte junto ao OpenStack

Como o ambiente estruturado se baseia no OpenStack, é interessante saber qual é o tipo de suporte oferecido a essa ferramenta.

Quem usa

Público do controlador

Quem está por trás

Empresas, consórcio, comunidade, universidade, ...

último release / commit

Uma das formas de se ter noção de quanto o desenvolvimento ainda está ativo é procurar pela data da última atualização de código no repositório.

Outras características

Características que não se adequam a nenhuma das anteriores. Como por exemplo:

- VM image file

POX

Overview

Controlador derivado do NOX.

http://www.noxrepo.org/pox/about-pox/

- irmão mais novo do NOX
- plataforma para desenvolvimento rápido e prototipagem de controladores OpenFlow (framework) usando python.
- tem como alvo principal é a pesquisa
- interface pythonica para OpenFlow
- melhor desempenho que o NOX para aplicações escritas em python, porém mais lento que NOX/C++
- especial suporte para Open vSwitch
- SDN debugging
- network virtualization
- projeto de controladores
- modelagem de programação
- desenvolvimento de um moderno controlador SDN

Sobre o código

Linguagem

Python 2.7

Licença

GPLv3

Suporte a versões do OpenFLOW

- suporte para OpenFlow 1.0

Requisitos

- pacotes a serem instalados
- espaço de armazenamento exigido
- outros recursos necessários ao funcionamento do controlador.

Independência de linguagem de programação

Somente python

Modularidade

- componentes reutilizáveis (modularidade) (recursos prontos)

Repositório

https://github.com/noxrepo/pox

Recursos básicos do controlador

???

Aplicações

- py: interface python (debug)
- forwarding.hub: $10 hub
- forwarding.l2_learning: L2 learning switch
- forwarding.l2_pairs: L2 learning switch based on MAC addresses
- forwarding.l3_learning: "L3-lerning-switchy-thing"
- forwarding.l2_multi:
- forwarding.l2_nx
- forwarding.topo_proactive
- openflow.spanning_tree
- web.webcore
- messenger
- openflow.of_01

- openflow.discovery
- openflow.debug
- openflow.keepalive
- proto.pong
- proto.arp_responder
- info.packet_dump
- proto.dns_spy
- proto.dpcp_client
- proto.dhcpd
- misc.of_tutorial
- misc.full_payload
- misc.mac_blocker
- misc.nat
- misc.ip_loadbalancer
- misc.gephi_topo
- log
  - Disabling the Console Log
  - Log Formatting
  - Log Output
- log.color
- log.level
- samples.pretty_log
- tk
- host_tracker

Testes

https://github.com/noxrepo/pox/tree/betta/tests

- controllers (controle distribuído)
- topology
- unit

Disponibilidade nas distribuições GNU/Linux

Precisa ser clonado com git.

último release / commit

Branch 'carp' com atividades recentes.

Documentação

- POX wiki: instalação, componentes, APIs, openflow e FAQ (bem completo)
  https://openflow.stanford.edu/display/ONL/POX+Wiki

- mininet + POX
  http://mininet.org/blog/2013/06/03/automating-controller-startup/

- Configuring an OpenFlow POX SDN Controller and HP Switches
  http://networkstatic.net/configuring-an-openflow-pox-sdn-controller-and-hp-switches/

- openflow tutorial with pox
  http://blog.pythonicneteng.com/2013/02/openflow-tutorial-with-pox.html

Suporte junto ao OpenStack

Nenhum, aparentemente

Quem usa

Voltado para pesquisadores.

Quem está por trás

Nicira

Outras características

- suporta mesmas ferramentas de visualização do NOX

Funcionamento

pox.py recebe uma lista de módulos na linha de comando e chama a função 'launch()' deles caso ela exista e depois muda para estado 'up'.

Os módulos podem estar nos diretórios 'pox' e 'ext', além dos diretórios normalmente vigiados pela instalação do python.

Opções (--) podem ser acrescidas após o nome do módulo e serão passadas como parâmetro para o chamado da função 'launch()'.

Ryu

Página do projeto

www.osrg.net/ryu

- Network opearting system
- OVS, openflow switch

Sobre o código

Linguagem

Python 2.7

Licença

Apache 2.0 license

Suporte a versões do OpenFLOW

- 1.0 with Nicira Extensions
- 1.2
- 1.3

Requisitos

- ambiente Python 2.7
  - python-gevent python-routes python-webob python-paramiko

Independência de linguagem de programação

Comunicação por mensagens, então, pode conversar com qualquer aplicativo que implemente essas mensagens.

Modularidade

???

Repositório

git://github.com/osrg/ryu.git

Recursos básicos do controlador

- GRE tunneling
- VLAN support
- IPv6
- topology discovery
- MAC based segregation

Aplicações

https://github.com/osrg/ryu/tree/master/ryu/app

- cbench.py
- client.py
- conf_switch_key.py
- gre_tunnel.py
- ofctl_rest.py
- quantum_adapter.py
- rest.py: app/rest
- rest_conf_switch.py
- rest_firewall.py
- rest_nw_id.py
- rest_quantum.py
- rest_router.py
- rest_topology.py
- rest_tunnel.py
- simple_isolation.py
- simple_switch.py

- simple_switch_12.py
- simple_switch_13.py
- simple_switch_lacp.py
- simple_vlan.py
- tunnel_port_updater.py
- wsgi.py

Testes

https://github.com/osrg/ryu/tree/master/ryu/tests

- app
- lib
- ofproto
- packe
- sample

Disponibilidade nas distribuições GNU/Linux

Pode ser instalado com o gerenciador 'pip'.

Último release / commit

- Aplicações: "a month ago"
- Testes: "a day ago"
- Core: "a day ago"

Documentação

Documentação oficial

http://ryu.readthedocs.org/en/latest/index.html

Tutoriais

https://github.com/osrg/ryu/wiki/OpenFlow_Tutorial

Suporte junto ao OpenStack

Quantum Ryu plugin. VM pré-configurada para testar OpenStack + Ryu.

https://github.com/osrg/ryu/wiki/OpenStack

- Flat networks regardless of the underlying physical network
- Scalable multi-tenant isolations
  - tunneling based isolations
  - VLAN does not scale larger than 4096
  - do not need high-end switches
- Future items:
  - integration with quantum IPAM and L3 API
  - firewall
  - virtual network <--> physical network
  - distributed controllers (redundancy)

Communication

'nova requests quantum-server for network configuragion'

OVS <--> Ryu: openflow

Ryu agent <--> Ryu: REST API

Quantum server <--> Ryu: REST API

Ryu agent <--> Quantum db: db access

VIF driver <--> quantum: Quantum API
linux_net driver <--> server: Quantum API

Network Creation (network-node = nova-network + linux_net driver + OVS)

1. to nova-network: create netork

2. nova network --> quantum server: create net
3. quantum server --> Ryu: network <uuid>, <tunnel key>
4. quantum server --> nova network: <uuid>
5. nova network --> linux_net driver: plug
6. linux_net driver --> OVS: create gw-xxx
7. linux_net driver --> quantum server: tentant id, network id, dp id, port, mac
8. quantum server --> Ryu: network id, dp id, port, mac
9. ryu --> OVS: set flow entryies

Instance creation (compute-node = nova compute + VIF driver + OVS)

1. to nova compute: create instance
2. nova compute --> quantum server: create port
3. quantum server --> nova compute: uuid
4. nova compute --> VIF driver: plug
5. VIF driver --> OVS: create tap-xxx
6. VIF driver --> quantum server: (tenant_id, dp_id, port, mac)
7. quantum server --> Ryu: (network_id, dp_id, port, mac)
8. Ryu --> OVS: set flow entryies

Node boot up (compute/network-node = OVS + Ryu agent + ryu_v2.ini)

1. ryu_v2.ini --> Ryu agent: get ip address
2. Ryu agent --> quantum db: register (dp_id, ip_address)
3. quantum db --> Ryu agent: get list of (dp_id, ip_address, and polling)
4. Ryu agent --> OVS: create vport_grc to ip address
5. Ryu agent --> Ryu: register (dp_id, port_id, remote dp_id)

Quem usa

???

Quem está por trás

NTT Lab

Outras características

OpenstackCriticalBugs

Aqui uma lista de bugs criticos do OpenStack, que afetam a implantacao na FDE.

Lentidao para subir uma maquina Windows.

Motivos:

- O OpenStack faz uma copia da imagem duas vezes, caso o computer node nao tenha o cache da imagem:
  - Copia do repositorio de imagem para o computer node (_base);
  - Copia do computer node (_base) para a instancia;

- Caso o computer node tenha cache, faz apenas a ultima copia;

- O OpenStack apenas faz uso de LVM + iscsi em se tratando de "volumes";

- Um "volume" nao pode ser attachado a mais de uma instancia
  - Isso impossibilita levantar mais de uma Lena;

Por que nao foi detectado antes?

O time do openstack nao esta fazendo testes pesados com maquinas Windows muito menos imagens grandes.

Segundo a propria documentacao do OpenStack:

" Troubleshooting

Mainly, it wouldn't take more than **5 minutes** in order to go from a "SAVING" to the "ACTIVE" state. If this takes longer than five minutes, here are several hints:

1. The feature doesn't work while you have attached a volume (via nova-volume) to the instance. Thus, you should dettach the volume first, create the image, and re-mount the volume.
2. Make sure the version of qemu you are using is not older than the 0.14 version. That would create "unknown option -s" into nova-compute.log.
3. Look into nova-api.log and nova-compute.log for extra information."

Possiveis Solucoes:

Criacao do snapshot manual:

1. Criar um volume na mao via snapshot de um volume ja existente;
2. Registar este volume no nova-volume;
3. Instanciar uma VM com este novo volume atachado;

Este papel seria do balanceador de vm;

Problemas:

1. O nova-volume nao da pra "registrar um volume ja criado", apenas se cria nele um volume vazio de tamanho N;

Criar snapshot de uma imagem e bootar por ele:

1. Criar um volume com o comando nova volume-snapshot-create, que cria um snapshot a partir de uma imagem;
2. Instanciar uma VM com este novo "snapshot" attachado;

Problemas:

1. O nova so attacha "volumes" a instancias, "snapshots", nao;

Bugs relacionados:

- https://bugs.launchpad.net/nova/+bug/1008622
- https://bugs.launchpad.net/nova/+bug/1005956
- https://bugs.launchpad.net/nova/+bug/962615
- https://bugs.launchpad.net/nova/+bug/1006950
- https://bugs.launchpad.net/nova/+bug/889299

OpenStack Networking - Netron

XLCloud-Networking_Needs_Quantum.pdf: continuar da página 9

Guias de Integração com Openflow:

Floodlight

Trema

Ryu

RestProxy Plugin

OpenVSwitch

Neutron

Serviço OpenStack que lida com rede (OpenStack Networking)

- é uma API disponível para os tenants
- oferece recursos dinamicamente
- criar e configurar redes virtuais

- Neutron é apenas uma camada de abstração que unifica as chamadas de configuração. Quem realmente faz o trabalho é o Neutron Plugin.
- Neutron controla a virtualização de rede (network services) da mesma forma que Nova controla a virtualização de servidores e instâncias.
- Neutron fornece uma API para gerenciamento de rede pelos projetos (tenants)
- Neutron pode definir topologias de rede complexas
- Neutron consegue lidar com diferentes tecnologias de virtualização de rede.

Implementação de referência

Neutron-PhysNet-Diagram.png
Retirada do OpenStack Admin Guide

- cloud controller host (Cloud Controller Node): quantum-server, nova-api, nova-scheduler, keystone, RabbitMQ, MySQL.
- network gateway host (Network Node): quantum-metadata-agent, quantum-dhcp-agent, quantum-l3-agent, quantum-*-plugin-agent
- conjunto de hypervisors para rodar VMs. Compute node: nova-compute, quantum-*-plugin-agent

cloud controller host e network gateway host podem ser combinados em uma máquina, porém é recomendado que exista um network gateway host separado para evitar restrições de CPU entre os encaminhamentos realizados pelo quantum-l3-agent e outros serviços OS.

- DataNetwork: comunicação de dados dentro da implementação de nuvem.
- Management Network: comunicações entre componentes OS.
- External Network: oferece acesso à internet à implementação OS.
- API Network: expõe todas as APIs do OS, incluindo a Netron API para os tenants. Pode ser a própria rede externa utilizando o recurso de subredes para separação.

Controller host config (OVS)

under-the-hood-scenario-1-ovs-network.png
Retirada do OpenStack Guide

Compute host config (OVS)

under-the-hood-scenario-2-ovs-compute.png
Retirada do OpenStack Guide

Networking API

Networking oferece uma API para definir interface entre os serviços OpenStack (e.g.: vNICs de máquinas virtuais com Nova). Os recursos alocados possuem dois modos: controle e operação
Abstrações que descrevem os recursos:
- Network: segmento L2 isolado (análogo a VLAN)
- Subnet: faixa de endereços IP (v4 ou v6) e estado de configuração associado.
- Port: ponto de conexão para um dispositivo. Também descreve configurações associadas da rede (como MAC e endereço IP usado na porta em questão).

Modo de operação:
- Rede: List, get, add, update, remove
- Porta: List, get, add, update, remove
- Periférico: Get, plug, unplug

Agentes

Agentes são softwares que devem rodar em cada nó que lida com pacotes de dados.

- plug-in agent (neutron-*-agent): roda em cada hypervisor para configuração local do vswitch.
- dhcp agent (neutron-dhcp-agent): provê serviço DHCP para as redes dos tenants.
- l3 agent (neutron-l3-agent): provê acesso externo a VMs de dentro das redes dos tentants. Porém, alguns plugins tem sua própria implementação desse recurso.
- l3 metering agent (neutron-metering-agent): provê medidas de tráfico das redes dos tenants.

Neutron Manager

- Gerenciador do Nova para o Neutron. Redireciona requisições relacionadas.
- Oferece outros serviços de rede (gerenciamento de IP, DHCP, NAT, ...)

Camadas do Neutron

1. presentation: configuration managers, operational APIs, ...
2. virtual network application: L2 switch, firewall, load balancer, router, ...
3. controller: floodlight, cisco overdrive, nicira OVP, ...
4. management: VxLan, STT, OpenFlow, SNMP, nvGRE, GRE, VLANs, NetConf, ...
5. devices: Nexus 5k, UCS, Open vSwitch, OF-enabled switches, Linux Bridge, ...

Neutron Plugin

Neutron Plugin são pedaços de código que implementam chamadas da API Neutron.

- composto por um controle centralizado e data model.
- Implementa todas as operações inclusas na "Network API v2.0".
- Traduz as operações da "Network API v2.0" em chamadas de reconfiguração de switches reais e virtuais.
- Pode ser compatível com mais de um tipo de switch (um plugin pode ter múltiplos drivers).
- Define se as mensagens devem ser síncronas ou assíncronas
- implementa ou faz interface com qualquer tipo de SDN (OpenFlow, Orchestration, Tunneling, Network Interface, virtualization, ‘legacy’, ...)

- Apenas um plugin pode rodar de cada vez.
- não existe uma interface padrão para os drivers.
- "meta-plugin' que chama código de dois diferentes plugins. (proposto -- ver l2 modular plugin)
- plugins e drivers devem ser escritos como biblioteca para permitir o reuso.

Componentes:

Um plugin geralmente é formado por códigos para:

- armazenar informação sobre a configução lógica atual da rede (e.g., SQL database)
- determina e armazena informações sobre o 'mapeamento' entre modelo lógico e rede física (e.g. escolhendo uma VLAN para representar uma rede virtual)..
- comunica com um ou mais tipos de switches para configurá-los de acordo com o mapeamento. Isso pode ser na forma de um agente rodando no *hypervisor* ou de código que remotamente se conecta a um switch e o reconfigura.

Plugins disponíveis:

- Big Switch Plug-in
- Brocade Plug-in
- Cisco
- Cloudbase Hyper-V Plug-in
- Linux Bridge Plug-in
- Mellanox Plug-in
- Midonet Plug-in
- ML2 (Modular Layer 2) Plug-in
- NEC OpenFlow Plug-in
- Nicira NVP Plug-in
- Open vSwitch Plug-in
- PLUMgrid
- Ryu Plug-in

Observações

- Cria-se 'netork' e a associa auma 'sub-net'
- vif-plugging + switch
- vif-plugging deve ser mantido tão simples quanto o possível para evitar complicações do nova com detalhes da rede. (e.g., não se deve passar VLAN ID para o nova; deve-se apenas deixar que o nova crie o dispositivo básico e fazer com que o agente do plugin do neutron cuide de configurar a conexão para aquela vlan.

- data-forward node tipicamente tem uma interface de rede com um endereço IP para "management network" e outro para "data network".

Esquemas e relações

- nova-compute <> VIF driver <> AMPQ <> nova-network <> Neutron manager <> API client <> Neutron API <> Neutron plugin
- Neutron common:
  1. API e API Extension
  2. AuthN e AuthZ
  3. Backend
  4. Hypervisor(n)
     - VM (n) + eth(n) <> hypervisor's Network stack
     - Quantum plugin agent <> hypervisor's Network stack
- Neutron Plugin <> Neutron common
- Neutron Plugin <> Hypervisor(n) (quantum plugin agent)
- Neutron Plugin <> Network switch (quantum plugin agent)

Quantum Plugin BaseClass

- get_all_networks(self, tenant_id, **kwargs): returns a dictionary containing all <network_uuid, network_name> for the specified tenant.
- create_network(self, tenant_id, net_name, **kwargs): creates a new Virtual Network, and assigns it a symbolic name.
- delete_network(self, tenant_id, net_id): deletes the network with the specified network identifier belonging to the specified tenant.
- get_network_details(self, tenant_id, net_id): retrieves a list of all the remote vifs that are attached to the network.
- update_network(self, tenant_id, net_id, **kwargs): updates the attributes of a particular Virtual Network.
- get_all_ports(self, tenant_id, net_id, **kwargs): retrieves all port identifiers belonging to the specified Virtual Network.
- create_port(self, tenant_id, net_id, port_state=None, **kwargs): creates a port on the specified Virtual Network.
- update_port(self, tenant_id, net_id, port_id, **kwargs): updates the attributes of a specific port on the specified Virtual Network.
- delete_port(self, tenant_id, net_id, port_id): deletes a port on a specified Virtual Network, if the port contains a remote interface attachment, the remote interface is first un-plugged and then the port is deleted
- get_port_details(self, tenant_id, net_id, port_id): this method allows the user to retrieve a remote interface that is attached to this particular port
- plug_interface(self, tenant_id, net_id, port_id, remote_interface_id): attaches a remote interface to the specified port on the specified Virtual Network.
- unplug_interface(self, tenant_id, net_id, port_id): detaches a remote interface from the specified port on the specified Virtual Network.
- *subclasshook*(cls, klass): the *subclasshook* method is a class method that will be called (Already Implemented) everytime a class is tested using issubclass(klass, Plugin). In that case, it will check that every method marked with the abstractmethod decorator is provided by the plugin class

Dúvidas ou perguntas para orientação

- O que é e o que faz o Neutron Backend?
- Como é feita a autenticação?
- Quais são as funções da API?
- Como definir quais máquinas físicas fazem parte da núvem?

Definições

- AuthN: authentication
- AuthZ: authorization
- Backend: tecnologia que está por trás do plugin.
- Quantum API core: conjunto mínimo de operações
- driver: código que é capaz de conversar com um modelo particular ou uma família de switches.

Referências

Visitadas

- http://docs.openstack.org/api/openstack-network/2.0/
- http://www.slideshare.net/danwent/openstack-quantum-intro-os-meetup-32612
- OpenStack Quantum Networking update
- https://wiki.openstack.org/wiki/NeutronDevelopment
- http://docs.openstack.org/havana/config-reference/content/
- Networking plug-ins
- OpenStack Admin Guide - Networking
- OpenStack Networking API v2.0 Reference

Não visitadas

- http://www.ovirt.org/images/b/b3/Quantum_Ovirt_discussion-20111103.pdf
- http://events.csdn.net/OpenStack/Takashi Torii-OpenStack and OpenFlowSDN.pdf
- http://www.openstack.org/assets/presentation-media/HK-Openstack-Namespaces1-.pdf
- http://www.iaas.uni-stuttgart.de/lehre/vorlesung/aktuell/vorlesungen/smcc/materialien/SoftwareDefinedNetwork_20131024_v04.pdf
- http://www.cisco.com/en/US/prod/collateral/switches/ps9441/ps11541/data_sheet_c78-727737.pdf
- http://www.linux-kvm.org/wiki/images/7/7b/Kvm-forum-2013-openstack.pdf
- https://wiki.openstack.org/wiki/NeutronDevstack
- http://www.openstack.org/assets/presentation-media/HK-Openstack-Namespaces1-.pdf

- http://www.iaas.uni-stuttgart.de/lehre/vorlesung/aktuell/vorlesungen/smcc/materialien/SoftwareDefinedNetwork_20131024_v04.pdf
- http://www.cisco.com/en/US/prod/collateral/switches/ps9441/ps11541/data_sheet_c78-727737.pdf
- Quantum Plugins and Networking

Planejamento para o terceiro entregável

plan.png

Neutron e SDN (Em esboço ainda)

- Neutron:
  1. presentation: operational APIs
  2. virtual network application: L2 switch
  3. controller; Neutron pode fazer interface ou implementar essa camada.
  4. management: Neutron pode fazer interface ou implementar essa camada.
  5. devices: Neutron pode fazer interface ou implementar essa camada.

Neutron Server

quantum-server: processo principal do Netron. Expõe Netron API e encaminha as requisições de usuário para o plugin.

Agentes

Agentes enviam mensagens para o Quantum-Server via RPC (rabbitmq ou qpid) ou Netron API padrão (REST).

- plugin agent (quantum-*-agent): roda em cada hypervisor para realizar configuração local do vswitch. Dependente do plugin escolhido.
- dhcp agent (quantum-dhcp-agent). serviço de DHCP para redes dos tentents. Usado quando as VMs devem receber IP de forma automatizada. Endereços via DHCP. Requer no mínimo dnsmasq 2.59. Configurado especificamente para o plugin utilizado.
- L3 agent (quantum-l3-agent). Encaminhamento L3/NAT para oferecer acessos as redes externas. Roteador que conecta as redes L2. Fica no Network Node. Usa: Linux IP Stack e IP tables.
  - Deve-se garantir que existam rotas entre o serviço de metadata do host e cada rede privada.
  - Nós que rodem quantum-l3-agent não devem ter endereços IP especificados manualmente para a NIC conectada a rede externa e sim uma faixa de IPs que poderá ser usada para roteamento.

```
$ sudo ovs-vsctrl add-br br-ext
$ sudo ovs-vsctrl add-port br-ext eth1
```

- não é possível executar ping para IPs fixos. Ao invés disso, usar: router network namespace (qrouter-<UUID>)
  ```
  $ ip netns exec qrouter-<UUID> ping <fixed ip>
  ```

Netron API

Serviço de redes virtual. Define conexões e endereços para os outros serviços.

- Network: segmento L2 isolado (análogo a uma VLAN).
- Subnet: bloco de endereços de IP (V4 ou V6) e estados de configuração associados.
- Port: ponto para conexção de dispositivos.

```
network <-- port
  |       |
   ---------- subnet
```

> *fixed_ips é atributo do port*
> *port.fixed_ips = {<IP1>:<subnet1>, <IP2>:<subnet2>, ...}*

Netron Plugins

Projeto original assume um modelo muito básico para realizar isolamento através de Linux VLANs e tabelas IP. Netron introduz o conceito de plugin, que é a
implementação de requisições lógicas de API em forma de um back-end acoplável. Padrão: Open vSwitch e Linux Bridge. Plugin deve estar em cada nó que usa
pacotes de dados.

Os plugins devem processar chamadas de API e gerenciar vSwitches.

OpenVSwitch: open vswitch, quantum-plugin-openvswitch-agent

```
$ sudo ovs-vsctrl add-br br-int
```

- br-int: integration bridge, nome padrão e fixo.

Netron Command Line Application

Requer: python-pyparsing, python-cliff, python-quantumclient

OpenStack Compute

Interagem com Netron através da API. Comunica com Netron API para connectar cada vNIC da VM em uma rede em particular.

Neutron Client

agent-*

dhcp-agent-*

ext-* (extensões)

floatingip-*

l3-agent-*

lb-* (loading balance)
  lb-pool-*
  lb-healthmonitor-*
  lb-member-*
  lb-vip-*

net-*

queue-*

router-*

security-*

subnet-*

quota-*

security-group-*

Exemplos - Netron API

```
$ quantum net-create net1
$ quantum net-create net2 --provider:network_type local
$ quantum net-create --tenant-id=<id> net1

$ quantum subnet-create net1 192.168.0.2/22 --name subnet1

$ quantum port-create net1 --fixed-ip ip_addr=192.168.2.40
$ quantum port-create net1
$ quantum port-list --fixed-ips
$ quantum port-list --tenant-id=<tenant id>
$ quantum port-list --device-id=<vm id>
```

- as redes já devem estar criadas para serem adicionadas na instanciação das VMs

```
$ nova boot --image=<img id> --flavor=<fl id> --nic net_id=<net id> <vm name>
```

- para ip fixo (port deve ser criado antes da instanciação)

```
$ nova boot --image=<img id> --flavor=<fl id> --nic port_id=<net id> <vm name>
```

Físico -> Virtual

1) eth0 (VM)
2) TAP VNET0 (hypervisor network interface card, VIF, vNIC)
3) LINUX BRIDGE qbrXXX (hub, multi NIC, virtual ou físico)
4) veth pair {qcbXXX, qvoXXX} (entra de um lado e sai do outro)
5) open switch bridge br-int (virtual switch, integration bridge,todos as VM estão conectadas a ele)
6) veth pair (int-br-eth1, phy-br-eth1)
7) physical network interface card eth1

Ryu

```
$ sudo pip install ryu
$ sudo apt-get install quantum-plugin-ryu-agent openvswitch-switch python-openvswitch openvswitch-datapath-dkms
```

- ovs_ryu_plugin.ini e quantum.conf devem ser copiados em cada nó que tenha agente ryu

Floodlight

Floodlight pode ser utilizado como o backend para o OpenStack Network utilizando um Plug-in para Neutron. O Neutron expõe um modelo de networking-as-a-service via API REST implementada no Floodlight. Dois componentes principais são utilizados para essa integração: um aplicativo Virtual Switch do Floodlight que implementa a Neutron API e o o RestProxy plugin que conecta o Neutron com o Floodlight.

Conexão entre FloodLight e Neutron em dois passos:

1. Rodar o REST Proxy Plug-in no neutron-server. Esse plugin foi projetado para rodar como parte do Neutro service.
2. Habilitar o Virtual Switch no Floodlight (antigo Virtual Network Filter). Esse é o módulo que implementa um isolamento de nível 2 (baseado em MAC) em redes openflow. Esse módulo vem com o Floodlight por padrão e não depende do Neutron ou do Openstack para ser executado.

Not to do

- Proxies ainda não implementados então não se deve executar:
  - nova-manage
  - nova para administrar recursos de rede

Files

Neutron-PhysNet-Diagram.png	76 KB	22 Jan 2014	Thiago Paiva
under-the-hood-scenario-2-ovs-compute.png	199 KB	22 Jan 2014	Thiago Paiva
under-the-hood-scenario-1-ovs-network.png	162 KB	22 Jan 2014	Thiago Paiva
plan.png	7.12 KB	23 Jan 2014	Thiago Paiva

Pausar uma Máquina Virtual

- Para pausar uma máquina virtual, basta executar o seguinte procedimento:
  - *Descobrir o ID da VM usando o comando onevm list*
  - *Executar o comando de pausa da VM:*
    - *onevm suspend VM_ID*

Voltar

Raphael

28-06-2013

Executou:

- Finalização da Escrita do relatório;

Próximas Ações:

- Migração de ambiente de testes para Quantum;

27-06-2013

Executou:

- Escrita do Relatório;

Próximas Ações:

- Finalização a Escrita do Relatório;

26-06-2013

Executou:

- Atualização do Wiki com informações sobre arquitetura Quantum;

Próximas Ações:

- Escrita de Relatório;

25-06-2013

Executou:

- Leitura sobre OpenVSwitch;
- Leituras sobre plugins e extensões da API do quantum;

Próximas Ações:

- Criação de Wiki com informações sobre arquitetura Quantum;

24-06-2013

Executou:

- Assistindo vídeos do OpenStack Summit 2013:
  - Quantum Release;
  - Quantum Architecture;

Próximas Ações:

- Pesquisa sobre diferenças entre plugins e extensões;

21-06-2013

Executou:

- Leitura sobre topologias de deploy utilizando quantum;

Próximas Ações:

- Realizar instalação quantum

20-06-2013

Executou:

- Leitura de Documentação sobre instalação do quantum;

Próximas Ações:

- Realizar Instalação do quantum;

19-06-2013

Executou:

- Estudo sobre Arquitetura do Quantum;

Próximas Ações:

- Ler sobre integração Quantum/OpenFlow
- Pesquisar sobre Plugins Quantum;

18-06-2013

Executou:

- Início da leitura da documentação do Quantum (Network Administrator Guide)

Próximas Ações:

- Pesquisar componentes do Quantum

17-06-2013

Executou:

- Leitura da Especificação 1.3 do Openflow

Próximas Ações:

- Terminar leitura da especificação 1.3 do Openflow

14-06-2013

Executou:

- Início da leitura da especificação 1.3

Próximas Ações:

- Ler especificação 1.3 do Openflow

13-06-2013

Executou:

- Leitura da especificação 1.0

Próximas Ações:

- Ler especificação 1.3 do Openflow

12-06-2013

Executou:

- Tentativas de dar boot na VM via volumes;
  - Criação de snapshots;
  - Tentativa com outras imagens;

Próximas Ações:

- Ler especificação 1.3 do Openflow

11-06-2013

Executou:

- Testes com o Trema;
- Escrita de controlador "hello world";
- Estudo da api do trema, principalmente classe:
  - Tremma::PacketIn: http://rubydoc.info/github/trema/trema/master/Trema/PacketIn

Próximas Ações:

- Ler especificação 1.3 do Openflow

10-06-2013

Executou:

- Testes com o ryu;
- Tentativas de colocar uma vm com estado persistente e dar boot a partir dela.

Próximas Ações:

- Testar Trema;

07-06-2013

Executou:

- Leitura Especificação 1.0 do Openflow

Próximas Ações:

- Colocar Controladores pra rodar

06-06-2013

Executou:

- Leitura Whitepaper OpenFlow:
  - http://www.openflow.org/documents/openflow-wp-latest.pdf

Próximas Ações:

- Ler Especificação 1.0 do OpenFlow;

05-06-2013

Executou:

- Leitura Whitepaper OpenFlow:
  - http://www.openflow.org/documents/openflow-wp-latest.pdf

Próximas Ações:

- Ler Especificação 1.0 do OpenFlow;

04-06-2013

Executou:

- Levantamente de features de controladores;
- Criação da tabela no wiki com comparativo de controladores

Próximas Ações:

- Continuar estudo do OpenFlow;

03-06-2013

Executou:

- Leitura sobre Mininet (walkthrough) e testes:
  - http://mininet.org/walkthrough/
- Leituras Introdutórias OpenFlow:**

Próximas Ações:

- Estudo Controladores OpenFlow;

28-05-2013

Executou:

- Relatório

Próximas Ações:

- Estudo sobre OpenFlow;

27-05-2013

Executou:

- Elaboração de Relatório

Próximas Ações:

- Elaboração de Relatório Continuação;

24-05-2013

Executou:

- Coleta de dados e construção de gráficos de perfomance;
- Criação de wikis com dados coletados;

Próximas Ações:

- Elaboração de Relatório;

23-05-2013

Executou:

- Modificações ao script de coleta de dados e geração de gráficos:
  - Ver log do git para detalhes;
- Leituras sobre gerência de memória no KVM:
  - http://www.linux-kvm.org/page/Memory
  - FAQ do KVM (http://www.linux-kvm.org/page/FAQ);
- Acompanhamento das threads:
  - https://www.redhat.com/archives/rhelv6-list/2011-February/msg00011.html
- Overcommiting de CPUs e Memória:
  -

http://www.linuxtopia.org/online_books/rhel6/rhel_6_virtualization/rhel_6_virtualization_sect-Virtualization-Tips_and_tricks-Overcommitting_with_KVM.html

Próximas Ações:

- Coletar dados;

22-05-2013

Executou:

- Resolvido problemas com mpirun;
  - Os testes deveriam iniciar consumindo menos memória;

- Máquinas criada com o nova não estavam recebendo hostnames;
  - Hostnames definidos na mão;
  - /etc/hosts alterado na mão;

- Iniciados Testes;

Próximas Ações:

- Coletar dados;

21-05-2013

Executou:

- Detectados problemas na execução dos teste:
  - mpirun alegando problemas de comunicação;
  - leituras de referências para resolução do problema;

Próximas Ações:

- Coletar dados;

20-05-2013

Executou:

- Criação de ambiente de testes:
  - 8 Vms em 2 HW (vm8hw2) - Hospedeiras: educloud05 e educloud06 2 cores/vm, 1.25 GB/vm
  - 16 Vms em 2 HW (vm16hw2) - Hespedeiras: educloud07 e educloud08 1 core/vm, 0.625 GB/vm

- Inicio dos testes;

Próximas Ações:

- Coletar dados;

17-05-2013

Executou:

- Coleta de dados:
  - 4 Vms em 2 HW (vm4hw2) - Vms: 192.168.100.18 ~ 21 (Hospedeiras: educloud08 e educloud10)4 cores/vm, 2.5 GB/vm

- Criação de Páginas com resultados;

Próximas Ações:

- Criação de ambiente de testes para 8 vms em 2 hw e 16 vm em 2 hw;

16-05-2013

Executou:

- Coleta de dados:
  - 2 Vms em um HW (vm2hw1) - Vms: 192.168.100.3 e 4) (Hospedeira: educloud05) 4 cores/vm, 2.5 GB/vm
  - 4 Vms em um HW (vm4hw1) - Vms: 192.168.100.5,7,8,9 (Hospedeira: educloud06) 2 cores/vm, 1.25 GB/vm
  - 8 Vms em um HW (vm8hw1) - Vms: 192.168.100.10 ~ 17 (Hospedeira: educloud07) 1 cores/vm, 0.625 GB/vm

- Criação de Páginas com resultados;

Próximas Ações:

- Verificar resultados para 4 Vms em 2 HW (vm4hw2)

15-05-2013

Executou:

- Criação de ambiente (VMs) e execução de testes:
  - 2 Vms em um HW (vm2hw1) - Vms: 192.168.100.3 e 4) (Hospedeira: educloud05) 4 cores/vm, 2.5 GB/vm

- 4 Vms em um HW (vm4hw1) - Vms: 192.168.100.5,7,8,9 (Hospedeira: educloud06) 2 cores/vm, 1.25 GB/vm
- 8 Vms em um HW (vm8hw1) - Vms: 192.168.100.10 ~ 17 (Hospedeira: educloud07) 1 cores/vm, 0.625 GB/vm
- 4 Vms em 2 HW (vm4hw2) - Vms: 192.168.100.18 ~ 21 (Hospedeiras: educloud08 e educloud10)4 cores/vm, 2.5 GB/vm

Próximas Ações:

- Verificar resultados e plotar gráficos

14-05-2013

Executou:

- Coleta de dados para 8 vms falha pois o SO hospedeiro está ficando sem memória para executar nada.
- Desempenho com 36% de uso de memória: 8 Gflops;

Próximas Ações:

- Verificar possibilidade de criar VMs com menos memória

13-05-2013

Executou:

- Implementação do script perl para conter a lógica da calculadora (terminado);
  - Atualizado script no git;
- Testes executando para 8vms em uma máquina física (uma por core) de 30% a 60% de uso de memória;

Próximas Ações:

- Verificar resultados e plotar gráficos

10-05-2013

Executou:

- Implementação do script perl para conter a lógica da calculadora;

Próximas Ações:

- Terminar a implementação do script;
- Executar testes com as novas entradas geradas;

09-05-2013

Executou:

- Execução de testes demorando muito - tamanho N utilizado incompatível com a RAM disponível;
- Testes para descobrir possível valor de porcentagem de memória aceitável para rodar benchmark;
- Máquina roda benchmark com até 55% de uso de memória para tamanho de bloco 248;

Próximas Ações:

- Estudo de cálculo utilizado pelo hpl-calculator;
- Implementação no script perl gerador de entradas a lógica utilizada pela calculadora;

08-05-2013

Executou:

- Execução de testes;
- Resolução de problemas de comunicação do impi: máquinas não reconheciam seus próprios hostnames;

Próximas Ações:

- Execução de testes;

07-05-2013

Executou:

- Configuração de ambiente para novos testes:
  - Pesquisa sobre como instanciar uma vm numa máquina específica;
  - Criação de 8 VMS em educloud05
  - Criação de flavor para teste: 1 processador, 750mb de ram;

Próximas Ações:

- Iniciar testes;

06-05-2013

Executou:

- Definição de calendário de atividades para novo mês;
- Leituras de referências sobre comparativos entre vms e hws reais;
  - http://www.loria.fr/~lnussbau/files/linux-virtualization-mls09-slides.pdf
  - https://indico.fnal.gov/getFile.py/access?contribId=55&sessionId=80&resId=0&materialId=slides&confId=2012

Próximas Ações:

- Subir ambiente para novos testes;

03-05-2013

Executou:

- Atualização do Wiki com comparativo:
  - [[4 Máquinas Reais Vs 4 Máquinas Virtuais]]

Próximas Ações:

- Estudar parâmetros para otimização do kvm e tentar obter melhores resultados;

02-05-2013

Executou:

- Confecção do Relatório;

Próximas Ações:

- Atualizar Wiki com comparativo:
  - 4vm vs 4hw

30-04-2013

Executou:

- Confecção do Relatório;

Próximas Ações:

- Atualizar Wiki com comparativo:
  - 4vm vs 4hw

29-04-2013

Executou:

- Atualização da Página do wiki com dados do benchmark para 1vm;
- Atualização de Página wiki com comparativo entre os benchmarks:
  - 1vm vs 1hw
- Início do Relatório;

Próximas Ações:

- Relatório;
- Atualizar Wiki com comparativo:

- 4vm vs 4hw

26-04-2013

Executou:

- Coleta de dados Inicial para 4 vms;
  - Resultado inicial em Número de Operações: 2.05980*10^02 GFlops

Próximas Ações:

- Atualizar Página com comparativos:
  - 1vm vs 1hw
  - 4vm vs 4hw

25-04-2013

Executou:

- Coleta de dados Inicial para 1 vm;
  - Resultado inicial em Número de Operações: 7.63280*10^01 GFlops
- Script de geração de estatísticas comitado para git;

Próximas Ações:

- Coletar dados para 4 vms;

24-04-2013

Executou:

- Instalação do nova-compute em máquinas;
- Teste Benchmark com uma VM;
- Teste de Benchmark com 4 VMs

Próximas Ações:

- Coletar dados;

23-04-2013

Executou:

- Criação da Imagem para rodar benchmarks

Próximas Ações:

- Instalação do nova-compute em máquinas;
- Rodar teste com uma VM única e compararar com Hw real;

22-04-2013

Executou:

- Coleta de dados e filtragem de dados no experimento de 4 máquinas (32 processadores);
- Atualização e reorganização de Página no Wiki com comparativos;

Próximas Ações:

- Instalação do nova-compute em máquinas;
- Rodar teste com uma VM única e compararar com Hw real;

19-04-2013

Executou:

- Modificação do script para coleta de dados de forma que fique genérico para criação de gráficos

Próximas Ações:

- Coletar resultados para 4 máquinas (32 processadores);

18-04-2013

Executou:

- Criação de entradas para benchmark;
- Modificação do script run-multi.sh para rodar com todas as possíveis entradas;

Próximas Ações:

- Coletar resultados e plotar gráficos;

17-04-2013

Executou:

- Script para coleta de dados da execução em uma máquina sem HT;
- Criação de gráficos para apresentar variação do tempo em cada porcentagem de memória;
- Criação de gráficos para apresentar variação do Número de operações em cada porcentagem de memória;
- Criação de gráficos com Número de operações e Tempo para cada porcentagem de uso de memória;
- Descoberta de melhor valor de uso de memória e tamanho de problema:
  - 92% de uso de memória;
  - Tamanho de problema (NB) em 184;
  - Número de Operações: 7.82992*10^01 GFlops;
- Setup do ambiente para execução de testes em mais de uma máquina (ssh sem senha, testes com o mpirun);

Próximas Ações:

- Executar testes para 4 máquinas;

16-04-2013

Executou:

- Execução de experimentos com processador utilizando HT - 90% de uso de memória;
- Criação de gráfico que compara a execução do benchmark com HT e sem.
- Criação de página no wiki para guardar resultados de benchmarks;

Próximas Ações:

- Encontrar valor otimizado de memória e tamanho de problema para uma única máquina sem HT.

Bloqueio:

- Aguardando benchmark terminar de executar;

15-04-2013

Executou:

- Script para busca de valores otimizados para tamanho do problema, NB e melhor uso de memória;

Próximas Ações:

- Plotar Gráfico de resultados;
- Criar página no wiki com resultados;

12-04-2013

Executou:

- Valor otimizado para uso do hpl em nb = 120 (com 90% de uso de memória: 7.66826e+01 GFlops);
- Gráfico de estudo do tempo vs nbs vs gflops
  https://ncc.unesp.br/its/attachments/225/NBvsTimevsGflops.png

Próximas Ações:

- instalar nova-compute em máquinas;

11-04-2013

Executou:

- Codificação de script para geração de arquivos .dat de entrada;
- Geração de entradas;
- Script para execução do linpack, jpgando saídas em output/
- Ticket 779 fechado (https://ncc.unesp.br/its/issues/779);

Próximas Ações:

- Verificar valores otimizados para testes;
- Plotar gráfico com dados de saída dos testes;

10-04-2013

Executou:

- Execução do Procedimento UNESP nas máquinas 04-08;
- Modificação do nfs para rw;
- Teste do Linpack utilizando nfs;
- Desabilitação do ht na educloud05;

Próximas Ações:

- Rodar conjuntos de testes para achar valores otimizados de Ns e NBs

09-04-2013

Executou:

- Tentativa de compilação do Linpack;
  - Utilizando openmpi;
- Testes de execução do linpack com Intel® Math Kernel Library;
- Screen sobre a utilização do Linpack utilizando Intel mpi;
- Leituras sobre o uso da calculadora para HPL.dat (http://hpl-calculator.sourceforge.net/):
  - http://ufpr.dl.sourceforge.net/project/hpl-calculator/HPL%20Calculator/HPL-HowTo.pdf

Próximas Ações:

- Completar instalação do ambiente;
- Testar execução do linpack otimizado para intel (fornecido pelo ncc) - em /opt;

08-04-2013

Executou:

- Leitura sobre Top500
  - http://www.top500.org/project/introduction/
  - http://www.top500.org/project/linpack/
- Leitura sobre Linpack
  - http://www.netlib.org/utk/people/JackDongarra/PAPERS/The-LINPACK-Benchmark-An-Explanation.pdf
  - Sobre HPL: http://www.netlib.org/benchmark/hpl/
  - http://support.clearspeed.com/resources/documentation/LINPACK_Getting_Started_Manual_Rev1.B.pdf
    - Guia de uso do Linpack

- Sobre a diferença entre LAPACK e LINPACK:
  Até onde entendi o LAPACK é uma biblioteca para algebra linear e pode (deve) ser utilizado em um futuro para a substituição de algumas rotinas do problema solucionado pelo LINPACK. A idéia é que o LINPACK foi concebido para rodar nos computadores que eram modernos na época, os Vector CPUS (http://en.wikipedia.org/wiki/Vector_processor). Ao se substituir as rotinas do problema solucionado pelo LINPACK pelo LAPACK o benchmark (ainda o LINPACK) vai poder explorar melhor as arquiteturas modernas baseadas em cache.

- LAPACK vs LINPACK vs EISPACK: http://www.netlib.org/lapack/lug/node10.html.

03-04-2013

Executou:

- Definição cronograma próxima fase;
- Criação de imagem do ubuntu e upload para Glance;
- Instalação do Cinder;

- Atualização da página do [[Ambiente]] com manual de instalação do Cinder;

Próximas ações:

- Começar estudo Top500.org;

02-04-2013

Executou:

- Escrita do Relatório

Próximas ações:

- Instalação Cinder;

01-04-2013

Executou:

- Leituras sobre criação de imagens personalizadas:
  - Utilizando o proprio nova: http://docs.openstack.org/trunk/openstack-compute/admin/content/creating-images-from-running-instances.html
  - Utilizando o VMBuilder:
    - http://wiki.debian.org/VMBuilder - Criação de imagens debian;
    - Ubuntu Juice (JeOS): https://help.ubuntu.com/12.04/serverguide/jeos-and-vmbuilder.html
    - Usando debootstrap: http://www.diogogomes.com/2012/07/13/debootstrap-kvm-image
- Leitura de QEMU vs KVM:
  - http://blog.vmsplice.net/2011/03/should-i-use-qemu-or-kvm.html

Próximas ações:

- Criação de imagem personalizada

28-03-2013

Executou:

- Instalação do nova;
- Problemas com a documentação http://docs.openstack.org/trunk/openstack-compute/install/apt/content/osfolubuntu-computesevice.html
  - O serviço não inicializa, esperando o componente nova-conductor que não está na lista de dependência dos pacotes do nova;
  - Leitura sobre o nova-conductor http://russellbryantnet.wordpress.com/2012/11/19/a-new-nova-service-nova-conductor
    - Basicamente nova-conductor é o componente responsável por funcionar como um proxy para o banco de dados, evitando assim que em cenários com múltiplos nova-compute todos façam acesso ao banco de dados. Nós com nova-compute não são considerados seguros.

Próximas ações:

- instalação cinder;
- geração de imagem debian personalizada;

Bloqueio:

- configuração da rede na instalação do nova;

27-03-2013

Executou:

- Resolução de problema de autenticação no glance + keystone;
  - Problemas com a documentação http://docs.openstack.org/trunk/openstack-compute/install/apt/content/osfolubuntu-imageservice.html:
    - Usuário admin não é adicionad a tenant service;
    - Trocar o usuário admin em *-paste.ini de admin para glance;

Próximas ações:

- Instalação nova;
- Instalação cinder;

26-03-2013

Executou:

- Implementação das configurações de servidor na máquina 00.
- Inicio da implementação do Ambiente na máquina 00.
  - instalação do keystone
  - instalação do glance

Próximas ações:

- finalizar instalação do glance
- instalação do cinder

Bloqueio:

- glance não faz autenticação no keystone
  - Rever http://docs.openstack.org/developer/glance/authentication.html;
  - Possivelmente relacionado:
    - https://lists.launchpad.net/openstack/msg21700.html

25-03-2013

Executou

- Leituras sobre Quantum:
  - Instalação e configuração do Quantum-Server;
  - Setup de RPC;
  - Lista de plugins;
  - Comparação Quantum vs nova-network: http://www.gossamer-threads.com/lists/openstack/dev/18373
    - Recomendação de continuar com nova-network para instalação Flat DHCP;
  - Integração com Keystone;
    - Criação de Serviço e endpoint;

Próximos passos:

- **Definição de arquitetura de rede a ser utilizada pelas máquinas de nova-compute**

21-03-2013

Executou

- Leitura OPS - Guia de operação do openstack;
  - Capítulo 1;
  - Diferença entre back-end de armazenamento em blocos e em objetos (cinder vs swift);
- Estudo de diferenças entre Cinder e nova-volumes;
  - http://docs.openstack.org/trunk/openstack-compute/admin/content/cinder-vs-nova-volumes.html

Próximos Passos:

- **Instalaçõa de Máquinas;**

20-03-2013 (presencialmente na Unesp)

Executou

- **Configuração BMC da máquina 06;**
- **Atualização da Bios;**
- **Atualização Firmware do BMC;**

Próximos Passos:

- **Estudo Cinder;**
- **Estudo Quantum**

19-03-2013

Executou

- Leitura Release Notes Openstack Folsom;

- https://wiki.openstack.org/wiki/ReleaseNotes/Folsom;
  *Tradução de texto com perguntas sobre TPM/TXT;

Próximos Passos:
- **Estudo Cinder;**
- **Estudo Quantum**

18-03-2013

Executou
- **Screen de instalação do nova + nós;**

Próximos Passos:
- **Screen instalação Horizon**

15-03-2013

Executou
- **Screen de instalação do glance;**

Próximos Passos:
- **Instalação do nova;**

14-03-2013

Executou:
- **Screen de instalação do Keystone**

Próximos Passos:
- **Instalação do Glance;**

13-03-2013

Executou:
- **Instalação de SO - 3 Máquinas:**
  - **Sistema Operacional;**
  - **Procedimento UNESP;**

Próximos Passos:
- **Leitura do documento produzido no Educloud com o procedimento de instalação - [[Ambiente]];**

12-03-2013

Executou:
- **Leitura sobre Glance:**
  - **http://docs.openstack.org/developer/glance/installing.html - Instalação;**
  - **http://docs.openstack.org/developer/glance/configuring.html - Configuração (Leitura rápida pra familiarização com Parâmetros);**
  - **http://docs.openstack.org/developer/glance/authentication.html - Integração com o Keystone;**

Próximos Passos:
- **Leitura do documento produzido no Educloud com o procedimento de instalação - [[Ambiente]];**
- **Instalação de SO em máquinas - Quarta pela manhã.**

11-03-2013

Executou:

- **Leitura de artigo sobre UEFI vs TPM (em especial sobre o secure boot e implementação no Suse.:**
  - **https://www.suse.com/blogs/uefi-secure-boot-overview/**
  - **https://www.suse.com/blogs/uefi-secure-boot-plan/**
- **Leitura sobre openstack: Keystone:**
  - **http://docs.openstack.org/developer/keystone/architecture.html - Arquitetura;**
  - **http://docs.openstack.org/developer/keystone/installing.html - Instalação;**
- **Leitura básica sobre devstack**
  - **http://devstack.org/**

Próximos Passos:

- **Continuar leituras sobre openstack;**

Files

NBvsTimevsGflops.png	8.06 KB	12 Apr 2013	Raphael Cóbe

Resultados Benchmark

Introdução

Com a aplicação da virtualização de hardware, mostrou-se necessário haver alguma forma de comparação entre o desempenho dessas máquinas com aquelas que não utilizam esse recurso. Isso é importante pois dessa forma pode-se mensurar qual é a sobrecarga que o sistema de virtualização provoca em condições correlatas e também para mensurar o desempenho de máquinas que só se tornaram possíveis com a virtualização.

Existem tipos diferentes de benchmark como o usado no TOP500 (top500.org) e no Graph500 (graph500.org). Cada um mensura características diferentes do computador.

Benchmark

Benchmark é a normalização de determinados procedimentos de forma a tornar possível a comparação entre eles.

LINPACK Benchmark

LINPACK Benchmark é um sistema de avaliação do poder computacional de computadores, utilizando ponto flutuante. Para isso, mede-se a velocidade de resolução de um sistema linear de equações n x n do tipo "Ax = b".

Será utilzado o HPLinpack que possibilita o teste da paralelização de computadores, em que o tamanho do problema pode ser definido de acordo com a disponibilidade de recursos de forma a fazer uso da plena capacidade das máquinas.

Os seguintes resultados são mostrados para cada teste:

R_max: a performance em Gflops/s para o maior problema executado
N_max: o tamanho do maior problema executado
N_{1/2}: o tamanho do problema para quando metade da **R_max** foi atingida.
R_peak: o pico teórico de performance.

Parâmetros utilizados

Os parâmetros utilizados foram obtidos a partir do software [hpl-calculator | http://hpl-calculator.sourceforge.net/] e seguem as seguintes especificações:

- Número de nós equivale ao número de máquinas a serem utilizadas: 1 ou 4.
- Número de cores é o número de núcleos de processamento em cada um dos nós: 8 (sem hyper-thread) ou 16 (com hyper-thread).
- Velocidade por core: 2.67.
- Quantidade de memória por nó: 6GB.
- Operações por ciclo: 4.

Das indicações feitas pelo hpl-calculator, escolhemos:

- P x Q aproximadamente iguais
- NBs entre 96 e 256.
- N entre 80 e 94%

Dos resultados:

Após realizar os testes com os valores escolhidos anteriormente, obtém-se:

- valor ótimo de N e NB a partir do pico real de performance.
- comparação direta entre os desempenhos das máquinas em condições correlacionadas:
  - influência do recurso de Hyper-Threading
  - influência da escolha da Matriz P x Q (quadrada e linear)
  - máquina virtual x máquina real

Nomenclatura

Cada teste é armazenado na máquina 00 em /opt/hpl/ utilizando um diretório cujo nome indica a qual teste pertence. Esse nome é composto por abreviações seguidas por valores:

- hw: número de máquinas reais
- vm: número de máquinas virtuais
- ht: hyper-thread ativo ou não.

Exemplos

- hw4ht0: teste com quatro máquinas reais sem hyper-threading
- hw1ht1: teste com uma máquina real com hyper-threading
- mv1ht0: teste com uma máquina virtual sem hyper-threading

Instruções para teste de "single"

- criar diretório e acessá-lo para receber informações do teste. O nome do diretório deve obedecer a nomenclatura estabelecida.

~$ mkdir <nome>
~$ cd <nome>

- copiar os arquivos necessários para o diretório recém criado

  ~$ cp -r /opt/hpl/* .

- modificar o arquivo "HPL.template" nos quesitos P e Q

  ~$ vim hpl-files/HPL.template

- editar o script "generateFile.pl" e alterar os parâmetros Ns e NBs

  ~$ vim hpl-files/generateFile.pl

- executar script "generateFile.pl" para gerar arquivos de entrada

  $ perl hpl-files/genereteFile.pl

- modificar "runme-single.sh" no quesito número de cores (NUMBER_CORES)

  ~$ vim hpl/runme-single.sh

- executar runme-single.sh

  ~$ hpl/runme-single.sh

Testes:

[[4 Máquinas Reais Vs 4 Máquinas Virtuais]]

[[1 Máquina Real Vs 1 Máquina Virtual]]

[[Máquina Real com HT Vs Máquina Real sem HT]]

[[Matriz QxP quadrada vs Matriz QxP Linear]]

[[Busca pelo maior número de operações por segundo em uma unica máquina sem HT - Variação do uso de memória]]

[[Busca pelo maior número de operações por segundo em uma unica máquina virtual - Variação do uso de memória]]

[[Busca pelo maior número de operações por segundo em 4 máquinas sem HT - Variação do uso de memória]]

[[Busca pelo maior número de operações por segundo em 4 máquinas virtuais - Variação do uso de memória]]

Testes Fase 2:

[[2 máquinas virtuais em uma máquina real - Variação do uso de memória]]
[[4 máquinas virtuais em uma máquina real - Variação do uso de memória]]
[[8 máquinas virtuais em uma máquina real - Variação do uso de memória]]

[[4 máquinas virtuais em duas máquinas reais - Variação do uso de memória]]

[[8 máquinas virtuais em duas máquinas reais - Variação do uso de memória]]

[[16 máquinas virtuais em duas máquinas reais - Variação do uso de memória]]

Files

NBvsTimevsGflops.png	8.09 KB	16 Apr 2013	Raphael Cóbe
NBvsTimevsGflops.png	7.1 KB	16 Apr 2013	Raphael Cóbe
NBvsTimevsGflops-Comparison.png	7.43 KB	16 Apr 2013	Raphael Cóbe
NBvsTimevsGflops-2x8.png	6.67 KB	16 Apr 2013	Raphael Cóbe
NBvsTimevsGflops-Comparison1x16HTvs2x8HT.png	8.3 KB	16 Apr 2013	Raphael Cóbe

Resumir uma Máquina Virtual

- Para resumir uma máquina virtual, basta executar o seguinte procedimento:
  - *Descobrir o ID da VM usando o comando onevm list*
  - *Executar o comando de resumo da VM:*
    - *onevm resume VM_ID*

Voltar

Scrum

[[Ciclo 2]]
[[Ciclo 3]]
[[Fase 2]]
[[Fase 3]]

Voltar

Fase 3

Lista de atividades da Fase 3 do projeto Educloud.

Beraldo

17-05-2012

Executou: - Iniciou e finalizou a atividade: [[#627]]: Formatar, instalar e configurar máquina de backup.

Próximos Passos:

- Iniciar a atividade: [[#627]]: Formatar as máquinas a serem utilizadas no Ciclo 3, instalar e configurar o sistema operacional em cada uma delas

---

Blocker/Dependencia:

- Esperando o backup ser feito para formatar as máquinas que serão utilizadas no ciclo 3

18-05-2012

Executou:

- Ficou em block

Próximos Passos:

- Iniciar a atividade: [[#627]]: Formatar as máquinas a serem utilizadas no Ciclo 3, instalar e configurar o sistema operacional em cada uma delas

---

Blocker/Dependencia:

- Esperando o backup ser feito para formatar as máquinas que serão utilizadas no ciclo 3

21-05-2012

Executou:

- Iniciar e finalizou a atividade: [[#627]]: Formatar as máquinas a serem utilizadas no Ciclo 3, instalar e configurar o sistema operacional em cada uma delas

- Iniciou a atividade: [[#629]]: Estudar a instalação e configuração do Keystone (autenticação)

Próximos Passos:

- Finalizar a atividade: [[#629]]: Estudar a instalação e configuração do Keystone (autenticação).

- Iniciar e finalizar a atividade: [[#630]]: Instalar e configurar o Keystone (autenticação).

Blocker/Dependencia:

- Nenhuma

22-05-2012

Executou:

- Finalizou a atividade: [[#629]]: Estudar a instalação e configuração do Keystone (autenticação)

- Finalizou a atividade: [[#630]]: Instalar e configurar o Keystone (autenticação).

Próximos Passos:

- Iniciar e finalizar a atividade: [[#631]]: Estudar a instalação e configuração do Glance (imaging service).

- Iniciar a atividade: [[#632]]: Instalar e configurar o Glance (imaging service).

Blocker/Dependencia:

- Nenhuma

23-05-2012

Executou:

- **Finalizou a atividade: [[#631]]: Estudar a instalação e configuração do Glance (imaging service).**

Próximos Passos:

- **Iniciar a atividade: [[#632]]: Instalar e configurar o Glance (imaging service).**

Blocker/Dependencia:

- Nenhuma

24-05-2012

Executou:

- **Iniciou a atividade: [[#632]]: Instalar e configurar o Glance (imaging service).**

Próximos Passos:

- **Finalizar a atividade: [[#632]]: Instalar e configurar o Glance (imaging service).**

Blocker/Dependencia:

- Nenhuma

25-05-2012

Executou:

- **Finalizou a atividade: [[#632]]: Instalar e configurar o Glance (imaging service).**
- **Iniciou e Finalizou a atividade: [[#633]]: Adicionar, listar, recuperar e remover imagens de máquinas virtuais.**
- **Iniciou e Finalizou a atividade: [[#634]]: Estudar a instalação e configuração do Nova Common/Scheduler (virtualização).**

Próximos Passos:

- **Iniciar a atividade: [[#636]]: Instalar e configurar o Nova Common/Scheduler (virtualização).**

Blocker/Dependencia:

- Resolver conflito LDAP + APT
- Nenhuma

28-05-2012

Executou:

- **Ficou em blocker por causa da mudanca dos servidores;**

Próximos Passos:

- **Iniciar a atividade: [[#636]]: Instalar e configurar o Nova Common/Scheduler (virtualização).**

Blocker/Dependencia:

- Resolver conflito LDAP + APT

29-05-2012

Executou:

- **Iniciou a atividade: [[#636]]: Instalar e configurar o Nova Common/Scheduler (virtualização).**
- **Iniciou a atividade: [[#635]]: Estudar a instalação e configuração do Nova Network (rede virtual).**
- **Iniciou a atividade: [[#637]]: Instalar e configurar o Nova Network (rede virtual).**

Próximos Passos:

- Finalizar a atividade: [[#636]]: Instalar e configurar o Nova Common/Scheduler (virtualização).
- Finalizar a atividade: [[#635]]: Estudar a instalação e configuração do Nova Network (rede virtual).
- Finalizar a atividade: [[#637]]: Instalar e configurar o Nova Network (rede virtual).

Blocker/Dependencia:

- Nenhuma

30-05-2012

Executou:

- **Executou a atividade: [[#636]]: Instalar e configurar o Nova Common/Scheduler (virtualização).**

Próximos Passos:

- Finalizar a atividade: [[#636]]: Instalar e configurar o Nova Common/Scheduler (virtualização).
- Finalizar a atividade: [[#635]]: Estudar a instalação e configuração do Nova Network (rede virtual).
- Finalizar a atividade: [[#637]]: Instalar e configurar o Nova Network (rede virtual).

Blocker/Dependencia:

- Nenhuma

31-05-2012

Executou:

- **Iniciou a atividade: [[#636]]: Instalar e configurar o Nova Common/Scheduler (virtualização).**
- **Iniciou a atividade: [[#650]]: Estudar a instalação e configuração do Nova Volume (disco virtual).**
- **Iniciou a atividade: [[#644]]: Instalar e configurar o Swift (storage).**
- **Iniciou a atividade: [[#643]]: Estudar a instalação e configuração do Swift (storage).**

Próximos Passos:

- Finalizar a atividade: [[#636]]: Instalar e configurar o Nova Common/Scheduler (virtualização).
- Finalizar a atividade: [[#635]]: Estudar a instalação e configuração do Nova Network (rede virtual).
- Finalizar a atividade: [[#637]]: Instalar e configurar o Nova Network (rede virtual).
- Finalizar a atividade: [[#644]]: Instalar e configurar o Swift (storage).

Blocker/Dependencia:

- Nenhuma

01-06-2012

Executou:

Próximos Passos:

- Finalizar a atividade: [[#636]]: Instalar e configurar o Nova Common/Scheduler (virtualização).
- Finalizar a atividade: [[#635]]: Estudar a instalação e configuração do Nova Network (rede virtual).
- Finalizar a atividade: [[#637]]: Instalar e configurar o Nova Network (rede virtual).
- Finalizar a atividade: [[#644]]: Instalar e configurar o Swift (storage).

Blocker/Dependencia:

- Nenhuma

04-06-2012

Executou:

- Finalizou a atividade: [[#643]]: Estudar a instalação e configuração do Swift (storage).
- Finalizou a atividade: [[#644]]: Instalar e configurar o Swift (storage).
- Finalizou a atividade: [[#648]]: Configurar permissões de acesso por usuário/grupo (Keystone, Swift)
- Finalizou a atividade: [[#645]]: Instalar e configurar o Swift Proxy.
- Finalizou a atividade: [[#644]]: Instalar e configurar o Swift (storage).
- Finalizou a atividade: [[#646]]: Fazer upload de um conjunto de livros digitais para a nuvem via Swift (storage).

Próximos Passos:

- Finalizar a atividade: [[#636]]: Instalar e configurar o Nova Common/Scheduler (virtualização).
- Finalizar a atividade: [[#635]]: Estudar a instalação e configuração do Nova Network (rede virtual).
- Finalizar a atividade: [[#637]]: Instalar e configurar o Nova Network (rede virtual).

Blocker/Dependencia:

- Nenhuma

05-06-2012

Executou:

- Executou a atividade: [[#651]]: Estabelecer a comunicação entre Swift e Glance, para que o primeiro possa manter as imagens manipuladas pelo segundo.
- Executou a atividade: [[#647]]: Configurar limite de quota por usuário/grupo (Keystone, Swift).

Próximos Passos:

- Finalizar a atividade: [[#651]]: Estabelecer a comunicação entre Swift e Glance, para que o primeiro possa manter as imagens manipuladas pelo segundo.
- Finalizar a atividade: [[#647]]: Configurar limite de quota por usuário/grupo (Keystone, Swift).
- Finalizar a atividade: [[#636]]: Instalar e configurar o Nova Common/Scheduler (virtualização).
- Finalizar a atividade: [[#635]]: Estudar a instalação e configuração do Nova Network (rede virtual).
- Finalizar a atividade: [[#637]]: Instalar e configurar o Nova Network (rede virtual).

Blocker/Dependencia:

- Nenhuma

06-06-2012

Executou:

- Finalizou a atividade: [[#651]]: Estabelecer a comunicação entre Swift e Glance, para que o primeiro possa manter as imagens manipuladas pelo segundo.
- Finalizou a atividade: [[#647]]: Configurar limite de quota por usuário/grupo (Keystone, Swift).
- Finalizou a atividade: [[#653]]: Estudar a instalação e configuração do Horizon (web frontend).
- Executou a atividade: [[#636]]: Instalar e configurar o Nova Common/Scheduler (virtualização).

Próximos Passos:

- Finalizar a atividade: [[#636]]: Instalar e configurar o Nova Common/Scheduler (virtualização).
- Finalizar a atividade: [[#635]]: Estudar a instalação e configuração do Nova Network (rede virtual).
- Finalizar a atividade: [[#637]]: Instalar e configurar o Nova Network (rede virtual).

Blocker/Dependencia:

- Nenhuma

11-06-2012

Executou:

- Finalizou a atividade: [[#636]]: Instalar e configurar o Nova Common/Scheduler (virtualização).
- Finalizou a atividade: [[#635]]: Estudar a instalação e configuração do Nova Network (rede virtual).
- Finalizou a atividade: [[#637]]: Instalar e configurar o Nova Network (rede virtual).
- Finalizou a atividade: [[#654]]: Instalar e configurar o Horizon (web frontend).
- Finalizou a atividade: [[#652]]: Instalar e configurar o Nova Volume (disco virtual).

- Finalizou a atividade: [[#638]]: Instanciar uma máquina virtual via Horizon (web frontend).
- Finalizou a atividade: [[#639]]: Remover uma máquina virtual via Horizon (web frontend).
- Finalizou a atividade: [[#640]]: Pausar uma máquina virtual via Horizon (web frontend).
- Finalizou a atividade: [[#641]]: Resumir uma máquina virtual via Horizon (web frontend).
- Finalizou a atividade: [[#649]]: Fazer download do conteúdo através da interface web (Horizon).

Próximos Passos:

Blocker/Dependencia:

- Nenhuma

12-06-2012

Executou:

- Iniciou a atividade: [[#675]]: Investigar problemas relacionadas ao Openstack x Maquinas Windows

Próximos Passos:

- Finalizar a atividade: [[#675]]: Investigar problemas relacionadas ao Openstack x Maquinas Windows

Blocker/Dependencia:

- Nenhuma

13-06-2012

Executou:

- Executou a atividade: [[#675]]: Investigar problemas relacionadas ao Openstack x Maquinas Windows

Próximos Passos:

Blocker/Dependencia:

- Nenhuma

14-06-2012

Executou:

- Nenhuma

Próximos Passos:

- Preparar codigo do educloud-mon para receber api de comunicacao com o openstack

Blocker/Dependencia:

- Nenhuma

15-06-2012

Executou:

- Iniciou e finalizou a configuracao do servidor GIT;
- Publicou o codigo do educloud-mon no servidor GIT:
- Iniciou a refatoracao do codigo do educloud-mon para receber api de comunicacao com o openstack

Próximos Passos:

- Finalizar a refatoracao do codigo do educloud-mon para receber api de comunicacao com o openstack

Blocker/Dependencia:

- Nenhuma

18-06-2012

Executou:

- Continuou a refatoracao do codigo do educloud-mon para receber api de comunicacao com o openstack

Próximos Passos:

- Finalizar a refatoracao do codigo do educloud-mon para receber api de comunicacao com o openstack

Blocker/Dependencia:

- Nenhuma

19-06-2012

Executou:

- Finalizou a refatoracao do codigo do educloud-mon para receber api de comunicacao com o openstack

Próximos Passos:

- Iniciar a tarefa: [[#659]]: Instalar e configurar o HAProxy.

Blocker/Dependencia:

- Nenhuma

20-06-2012

Executou:

- Iniciou a tarefa: [[#659]]: Instalar e configurar o HAProxy.

Próximos Passos:

- Finalizar a tarefa: [[#659]]: Instalar e configurar o HAProxy.

Blocker/Dependencia:

- Nenhuma

21-06-2012

Executou:

- Finalizou a tarefa: [[#659]]: Instalar e configurar o HAProxy.
- Finalizou a atividade: [[#655]]: Adicionar máquinas virtuais ao pool de maneira automatizada (sem a necessidade de executar manualmente um script, como no Ciclo 1).
- Finalizou a atividade: [[#656]]: Remover máquinas virtuais ao pool de maneira automatizada (sem a necessidade de executar manualmente um script, como no Ciclo 1).
- Finalizou a tarefa: [[#678]]: Investigar problemas relacionados ao acesso a certas vms
- Iniciou a tarefa: [[#660]]: Configurar o modo drenagem de conexões.

Próximos Passos:

- Finalizar a tarefa: [[#660]]: Configurar o modo drenagem de conexões.

Blocker/Dependencia:

- Nenhuma

22-06-2012

Executou:

- Finalizou a tarefa: [[#660]]: Configurar o modo drenagem de conexões.
- Finalizou a tarefa: [[#665]]: Executar testes de forte flutuação de carga.

Próximos Passos:

- Ajudar o Marco com a atividade [[#663]]: Estender balanceador atual para comunicar-se com OpenStack.

Blocker/Dependencia:

- Nenhuma

25-06-2012

Executou:

- Executou a atividade: [[#675]]: Investigar problemas relacionadas ao Openstack x Maquinas Windows

Próximos Passos:

- Executou a atividade: [[#675]]: Investigar problemas relacionadas ao Openstack x Maquinas Windows

Blocker/Dependencia:

- Nenhuma

26-06-2012

Executou:

- Executou a atividade: [[#675]]: Investigar problemas relacionadas ao Openstack x Maquinas Windows

Próximos Passos:

- Executar a atividade: [[#675]]: Investigar problemas relacionadas ao Openstack x Maquinas Windows

Blocker/Dependencia:

- Nenhuma

27-06-2012

Executou:

- Executou a atividade: [[#675]]: Investigar problemas relacionadas ao Openstack x Maquinas Windows

Próximos Passos:

- Executar a atividade: [[#675]]: Investigar problemas relacionadas ao Openstack x Maquinas Windows

Blocker/Dependencia:

- Nenhuma

17/05/2012

Executou:
- Não houve execução de tarefas

Próximos Passos:
- Cadastrar atividades no ITS

- Efetuar a atividade de Criar cópia de Segurança das máquinas do ciclo 1

Blocker/Dependencia:

- Nenhuma

Voltar

18/05/2012

Executou:
- Realizou cadastro das atividades no ITS

- Iniciou atividade de Criar Cópia de Segurança das máquinas do Ciclo 1

Próximos Passos:

- Continuar a atividade de Criar cópia de Segurança das máquinas do ciclo 1

Blocker/Dependencia:

- Nenhuma

Voltar

21/05/2012

Executou:
- Finalizou a atividade de Criar Cópia de Segurança das máquinas do Ciclo 1

Próximos Passos:

- Iniciar atividade de Estudar instalação e configuração do HAProxy.

Blocker/Dependencia:

- Nenhuma

Voltar

22/05/2012

Executou:
- Iniciou a atividade de estudo do HAProxy

Próximos Passos:

- Continuar atividade de estudo do HAProxy

Blocker/Dependencia:

- Nenhuma

Voltar

23/05/2012

Executou:
- Continuou a atividade de estudo do HAProxy

Próximos Passos:

- Continuar atividade de estudo do HAProxy

Blocker/Dependencia:

- Nenhuma

Voltar

24/05/2012

Executou:
- Nenhuma atividade

Próximos Passos:

- Continuar atividade de estudo do HAProxy

Blocker/Dependencia:

- Nenhuma

Voltar

25/05/2012

Executou:
- Finalizou a atividade de estudo do HAProxy
- Iniciou a atividade de instalação do HAProxy

Próximos Passos:

- Continuar atividade de instalação do HAProxy

Blocker/Dependencia:

- Nenhuma

Voltar

28/05/2012

Executou:
- Continuou a atividade de instalação do HAProxy

Próximos Passos:

- Continuar atividade de instalação do HAProxy

Blocker/Dependencia:

- Nenhuma

Voltar

04/06/2012

Executou:
- Iniciar atividade de estudo do Nova Volume

Próximos Passos:

- Continuar atividade de estudo do Nova Volume

- Finalizar atividade de instalação e configuração do HAProxy

Blocker/Dependencia:

- Nenhuma

Voltar

05/06/2012

Executou:
- Continuar atividade de estudo do Nova Volume

Próximos Passos:

- Continuar atividade de estudo do Nova Volume

- Finalizar atividade de instalação e configuração do HAProxy

Blocker/Dependencia:

- Nenhuma

Voltar

06/06/2012

Executou:

- Finalizou atividade de estudo do Nova Volume
- Iniciou atividade de instalação e configuração do Nova Volume

Próximos Passos:

- Finalizar atividade de instalação e configuração do Nova Volume
- Finalizar atividade de instalação e configuração do HAProxy

Blocker/Dependencia:

- Nenhuma

Voltar

11/06/2012

Executou:

- Continuou atividade de instalação e configuração do HAProxy

Próximos Passos:

- Finalizar atividade de instalação e configuração do HAProxy

Blocker/Dependencia:

- Nenhuma

Voltar

12/06/2012

Executou:

- Finalizou atividade de instalação e configuração do HAProxy
- Iniciou atividade de configurar modo de drenagem

Próximos Passos:

- Continuar atividade de configurar modo de drenagem

Blocker/Dependencia:

- Nenhuma

Voltar

13/06/2012

Executou:

- Iniciou a atividade de estudo da Interface de comunicação do OpenStack (API)

Próximos Passos:

- Continuar a atividade de estudo da Interface de comunicação do OpenStack (API)
- Continuar atividade de configurar modo de drenagem

Blocker/Dependencia:

- Nenhuma

Voltar

14/06/2012

Executou:

- Continuou a atividade de estudo da Interface de comunicação do OpenStack (API)

Próximos Passos:

- Continuar a atividade de estudo da Interface de comunicação do OpenStack (API)
- Continuar atividade de configurar modo de drenagem

Blocker/Dependencia:

- Nenhuma

Voltar

15/06/2012

Executou:

- Continuou a atividade de estudo da Interface de comunicação do OpenStack (API)

Próximos Passos:

- Continuar a atividade de estudo da Interface de comunicação do OpenStack (API)
- Continuar atividade de configurar modo de drenagem

Blocker/Dependencia:

- Nenhuma

Voltar

18/06/2012

Executou:

- Finalizou a atividade de estudo da Interface de comunicação do OpenStack (API)
- Iniciou a atividade de extensão do balanceador para comunicação com OpenStack

Próximos Passos:

- Continuar a atividade de extensão do balanceador para comunicação com OpenStack
- Continuar atividade de configurar modo de drenagem

Blocker/Dependencia:

- Nenhuma

Voltar

20/06/2012

Executou:

- Continuou a atividade de extensão do balanceador para comunicação com OpenStack

Próximos Passos:

- Continuar a atividade de extensão do balanceador para comunicação com OpenStack

Blocker/Dependencia:

- Nenhuma

Voltar

21/06/2012

Executou:

- Continuou a atividade de extensão do balanceador para comunicação com OpenStack

Próximos Passos:

- Continuar a atividade de extensão do balanceador para comunicação com OpenStack

Blocker/Dependencia:

- Nenhuma

Voltar

26/06/2012

Executou:

- Finalizou a atividade de extensão do balanceador para comunicação com OpenStack

Próximos Passos:

- Iniciou a atividade de investigar tempo de boot de VMs Windows no OpenStack.

Blocker/Dependencia:

- Nenhuma

Voltar

27/06/2012

Executou:

- Continuou a atividade de investigar tempo de boot de VMs Windows no OpenStack.

Próximos Passos:

- Continuar a atividade de investigar tempo de boot de VMs Windows no OpenStack.

Blocker/Dependencia:

- Nenhuma

Voltar

Thiago

28-06-2013 (sexta-feira)

Executou

- Leitura sobre extensões da API do OSNeutron
- Início do relatório mensal.

27-06-2013 (quinta-feira)

Executou

- Estudo sobre estrutura interna do plugin para OSNeutron.
- Leitura sobre extensões da API do OSNeutron

26-06-2013 (quarta-feira)

Executou

- Consolidação do estudo sobre OSNeutron
- página no wiki sobre OpenStack Networking

25-06-2013 (terça-feira)

Executou

- Leitura sobre OSNeutron.

24-06-2013 (segunda-feira)

Executou

- reunião presencial
- Leitura sobre OSNeutron.

21-06-2013 (sexta-feira)

Executou

- Leitura do OS Networking admin guide

20-06-2013 (quinta-feira)

Executou

- Leitura do OS Networking admin guide

19-06-2013 (quarta-feira)

Executou

- Reunião presencial*

14-06-2013 (sexta-feira)

Executou

- Estudo da especificação 1.3 do OpenFLOW

13-06-2013 (quinta-feira)

Executou

- Estudo da especificação 1.3 do OpenFLOW

12-06-2013 (quarta-feira)

Executou

- Reunião presencial

11-06-2013 (terça-feira)

Executou

- Estudo da especificação 1.0 do OpenFLOW

10-06-2013 (segunda-feira)

Executou

- Estudo da especificação 1.0 do OpenFLOW

07-06-2013 (sexta-feira)

Executou

- Consolidação do estudo sobre OpenFLOW Controller Ryu.

06-06-2013 (quinta-feira)

Executou

- Estudo sobre OpenFLOW Controller Ryu.

05-06-2013 (quarta-feira)

Executou

- Reunião presencial
  - Levantamento de dúvidas sobre os OpenFLOW Controllers.
  - Especulações a cerca das possíveis direções de evolução do trabalho.

04-06-2013 (terça-feira)

Executou

- Estudo / pesquisa sobre OpenFLOW e OpenFLOW Controllers. (continuação)

03-06-2013 (segunda-feira)

Executou

- Leitura sobre OpenFLOW e OpenFLOW Controllers.

31-05-2013 (sexta-feira)

Recesso

30-05-2013 (quinta-feira)

Feriado

29-05-2013 (quarta-feira)

Executou

- Relatório mensal (finalização)

28-05-2013 (terça-feira)

Executou

- Relatório mensal (continuação)

27-05-2013 (segunda-feira)

Executou

- Relatório mensal (continuação)

26-05-2013 (domingo)

Executou

- Início de estruturação do relatório mensal

24-05-2013 (sexta-feira)

Executou

- Reunião de planejamento

23-05-2013 (quinta-feira)

Executou

- Implementação das retras de monitoramento no Zabbix

22-05-2013 (quarta-feira)

Executou

- Finalização do processo de instalação do Zabbix e estudo e testes em sua interface web.

Próximos passos:

- Implementação das regras de monitoramento das máquinas

21-05-2013 (terça-feira)

Executou

- Início do processo de instalação do Zabbix

Próximos passos:

- Término do processo de instalação do Zabbix

20-05-2013 (segunda-feira)

Executou

- Término do processo de instalação do SNMP e implementação de script de monitoramento.

Próximos passos:

- Instalação do Zabbix

19-05-2013 (domingo)

Executou

- Início de instalação do SNMP

Próximos passos:

- Completar uma instalação

17-05-2013 (sexta-feira)

Executou

- Pesquisa sobre quantidade de memória RAM destinada a máquina hospedeira em relação à quantidade de instâncias.

Próximos passos:

- Instalação do snmp

16-05-2013 (quinta-feira)

Executou

- revisão openstack (boot, flavor, ...)
- scheduler filters / availability-zone
- resolução de problemas com PAM na educloud02 (sudoers + nslcd)
- instalação de nova-compute na educloud02

07-05-2013 (segunda-feira)

Executou

- reunião presencial de planejamento

Próximos passos

- atualizar issues com pendências para este mês
- reinstalação da máquina educloud01 (ou 02) que está com debian antigo e fora do padrão.

03-05-2013 (quinta-feira)

Executou

- construção de gráficos via LaTex como opção para o relatório

02-05-2013 (quinta-feira)

Executou

- iniciou verificação sobre obtenção de consele de uma máquina virtual a partir de uma seção virsh na máquina hospedeira.
- alterações no relatório mensal.

01-05-2013 (quarta-feira)

Feriado

30-04-2013 (terça-feira)

Executou

- Confecção do relatório mensal.

29-04-2013 (segunda-feira)

Executou

- Confecção do relatório mensal.

26-04-2013 (sexta-feira)

Executou

- Estudo sobre virsh
- Implementação de máquina virtual na educloud09

Próximos passos

- Confecção do relatório.

25-04-2013 (quinta-feira)

Executou

- Verificação e fechamento dos Issues já completados
- Correção da página "Procedimentos IPMI" (educloud -> bmc)
- Complementação da página "Resultados Benchmark"

Próximos passos

- estudo do script bash que realiza os testes

24-04-2013 (quarta-feira)

Executou

- Instalação, configuração e teste de implementação de máquina virtual no openstack tendo como hyper-visor educloud05, ou seja, ambiente externo a educloud00.

23-04-2013 (terça-feira)

Executou

- estudo das CLI do openstack (revisão: keystone, glance e nova)
- alteração da imagem de boot do ubuntu server acrescentando
  - configuração ssh (acho q faltou instalar o ssh)
  - instalação de cliente nfs
  - configração do acesso a educloud00:/opt/ via nfs
- criação de servidor virtual com a imagem modificada
- obtenção de acesso por novnc (browser)

Próximos passos:

- verificar se na imagem criada existe servidor ssh
- eliminar e recriar servidor virtual tomando cuidado de vinculá-lo ao kernel (não foi feito)

22-04-2013 (segunda-feira)

Executou

- atualização do wiki de testes

19-04-2013 (sexta-feira)

Executou

- pesquisa sobre método de alteração de senha de ipmi
- documentação do método
- alteração das senhas de ipmi das máquinas bmc05 a bmc10

15-04-2013 (segunda-feira)

Executou

- reorganização da estrutura de trabalho com HPL
- confecção dos arquivos de configuração a partir dos scripts preparados
- modificação do arquivo README com instruções atualizadas para testes SINGLE
- teste "hw1ht1" na educloud06
- leitura de:
  - http://gec.di.uminho.pt/DISCIP/MInf/cpd0708/PAC/rocksHpCaseStudy.pdf
  - http://www.netlib.org/benchmark/hpl/tuning.html
  - http://www.advancedclustering.com/faq/how-do-i-tune-my-hpldat-file.html
    - outra ferramenta para encontrar parâmetros do HPL.dat

Próximos passos

- leitura de:

- http://www.clustermonkey.net/Projects/life-the-universe-and-your-cluster-a-study-in-cluster-optimization.html

12-04-2013 (sexta-feira)

Executou

- estudo da estrutura do HPL montada para os testes

11-04-2013 (quinta-feira)

Executou

- verificação das regras de servidor e da montagem por nfs do educloud00:/opt/ nas máquinas da rede local.
- estudo da estrutura do HPL montada para os testes

10-04-2013 (quarta-feira)

Executou:

- Reunião presencial.
- Configuração das regras de servidores para as máquinas da rede local (educloud04, ..., educloud10)

09-04-2013 (terça-feira)

Executou:

- aprofundamento do estudo sobre os Benchmarks do LINPACK
- screen sobre LINPACK

09-04-2013 (terça-feira)

05-04-2013 (sexta-feira)

Executou:

- Continuação da leitura sobre LINPACK
  - http://www.top500.org
  - http://www.top500.org/project/introduction/
  - http://www.top500.org/project/linpack/
  - http://en.wikipedia.org/wiki/LINPACK_benchmarks
  - http://www.netlib.org/utk/people/JackDongarra/PAPERS/The-LINPACK-Benchmark-An-Explanation.pdf
  - http://www.spec.org/workshops/2007/austin/slides/Keynote_Jack_Dongarra.pdf
    - apresentação (overview - HPL)
  - http://www.netlib.org/benchmark/hpl/ (script)
  - http://support.clearspeed.com/resources/documentation/LINPACK_Getting_Started_Manual_Rev1.B.pdf
    - linpack getting start guide

- LAPACK (alternativa mais atual ao LINPACK?)

LINPACK was designed to run on the then-modern vector computers with shared memory. LAPACK, in contrast, was designed to effectively exploit the caches on modern cache-based architectures, and thus can run orders of magnitude faster than LINPACK on such machines, given a well-tuned BLAS implementation.

- http://en.wikipedia.org/wiki/LAPACK
- http://www.cs.utexas.edu/~plapack/
- http://en.wikipedia.org/wiki/ScaLAPACK
- http://www.netlib.org/lapack/lug/node1.html
- http://www.physics.oregonstate.edu/~rubin/nacphy/lapack/cprogp.html

04-04-2013 (quinta-feira)

Executou:

- Pesquisa por material sobre LINPACK
- Inicio de leitura do material pesquisado

Próximos passos:

- Continuação da leitura sobre LINPACK

03-04-2013 (quarta-feira)

Executou:

- reunião presencial
  - planejamento das atividades do mês
- teste da resolução de domínios
  - download da imagem do Ubuntu 13.04 para nuvem

Próximos passos:

- Estudo sobre Linpack

02-04-2013 (terça-feira)

Executou:

- busca por imagens para nuvens
- releitura do capítulo 7. Image Management do OPS
- tentativa de instalação na máquina 00
  - http://uec-images.ubuntu.com/raring/current/
  - falha no resolv para uec-ubuntu
- análise da imagem oferecida no próprio computador

Próximos passos:

- subir a imagem do ubuntu raring no glance 00

01-04-2013 (segunda-feira)

Executou:

- Estudo sobre instalação de Debian para nuvens
  - http://cssoss.wordpress.com/2011/11/28/bundling-debian-image-for-openstack/
  - http://cykod.com/blog/post/2010-03-working-with-uec--eucalyptus
  - http://www.logilab.org/blogentry/115219
  - http://wiki.debian.org/Cloud/AmazonEC2Image
  - http://dachary.org/?p=1379

- Estudo sobre instalação de Ubuntu para nuvens
  - https://help.ubuntu.com/community/UEC/Images
  - http://cloud-images.ubuntu.com/

- confecção do terceiro relatório

Próximos passos:

- instalação do cinder
- seleção e instalação de uma imagem do ubuntu na educloud.

28-03-2013

Executou:

- Instalação do nova

Bloqueio:

- conflito das vlans internas (automaticamente gerenciadas) com as externas (forçadas)

27-03-2013

Executou:

- Resolução de problema de autenticação no glance + keystone;
  - Problemas com a documentação http://docs.openstack.org/trunk/openstack-compute/install/apt/content/osfolubuntu-imageservice.html
    - usuários e serviços não criados e/ou desvinculados

Próximos passos:

- Instalação do Nova

26-03-2013

Executou:

- implementação das configurações de servidor na máquina 00.
- inicio da implementação do Ambiente na máquina 00.
  - instalação do keystone
  - instalação do glance

Próximas ações:

- finalizar instalação do glance
- instalação do cinder

Bloqueio:

- glance não faz autenticação no keystone

25-03-2013

Executou:

- acompanhamento do irc #openstack
- leitura de http://docs.openstack.org/ops/

Proximas ações:

- instalação do Ambiente na máquina 00.

22-03-2013

Executou:

- acompanhamento do irc #openstack
- leituras diversas:
  - changing log: Essex -> folsom
    - http://my1.fr/blog/whats-new-in-openstack-folsom/
  - https://www.ibm.com/developerworks/mydeveloperworks/blogs/e93514d3-c4f0-4aa0-8844-497f370090f5/entry/quantum_folsom12?lang=en

Próximas ações:

- assistir vídeo OpenStack Quantum Networking update
- leitura de http://docs.openstack.org/ops/

21-03-2013

Executou:

- leituras diversas:
  - changing log: Essex -> folsom
  - leitura do livro OPS
  - acompanhamento do irc #openstack

Próximas ações:

- assistir vídeo OpenStack Quantum Networking update
  - https://www.youtube.com/watch?v=_I1SrO2QBpg
- leitura:
  - http://my1.fr/blog/whats-new-in-openstack-folsom/
  - http://docs.openstack.org/ops/

20-03-2013

Executou:

- reunião presencial
- estudo da nova estrutura de cloud
- instalação do Ubuntu Server 13.04

- estudo do ipmi para controle das máquinas por BMC.

Próximas ações:

- changing log: Essex -> folsom
- assistir vídeo OpenStack Quantum Networking update
  - https://www.youtube.com/watch?v=_I1SrO2QBpg
- leitura do livro OPS
  - http://docs.openstack.org/ops/

Bloqueio:

- redirecionamento do console via BMC - SOL

19-03-2013

Executou:

- acompanhamento do irc #openstack
- issue 764 (trusted pools)
- issue 763 (delete command)

Próximas ações:

- changing log: Essex -> folsom
- assistir vídeo OpenStack Quantum Networking update
  - https://www.youtube.com/watch?v=_I1SrO2QBpg

18-03-2013

Executou:

- implementação das regras de configuração de servidores para máquina 00
- pesquisar sobre bug de remoção de máquinas instanciadas com alguns status de erros
  - http://markmail.org/message/gygnjvdy3njwfrtg
    - https://code.seas.harvard.edu/openstack/stackutil
  - http://www.sebastien-han.fr/blog/2012/07/10/delete-a-vm-in-an-error-state/
  - http://www.gossamer-threads.com/lists/openstack/dev/17120
    - http://docs.openstack.org/essex/openstack-compute/admin/content/reset-state.html
- pesquisar sobre trusted pools e OpenStack
  - https://wiki.openstack.org/wiki/TrustedComputingPools
  - https://blueprints.launchpad.net/nova/+spec/trusted-computing-pools
- acompanhamento do irc #openstack
- acompanhamento remoto da instanciação de uma máquina virtual com compute-node e network-node.

Próximas ações:

- issue 764

15-03-2013

Executou:

- atualização dos ip's das máquinas na wiki (ambiente)
- implementação das regras de configuração de servidores para máquina 09
- leitura sobre OpenStack (docs.openstack.org)
  - interessante: http://docs.openstack.org/trunk/openstack-compute/admin/content/logical-architecture.html
- acompanhamento remoto da instalação do glance, cinder e compute

Próximas ações:

- pesquisar sobre trusted pools e OpenStack
- pesquisar sobre bug de remoção de máquinas instanciadas com alguns status de erros
- atualizar wiki Ambiente com passos não documentados
- implementação das regras de configuração de servidores para máquina 00

14-03-2013

Executou:

- acompanhamento da instalação do Keystone do OpenStack

Próximos passos:

- finalização da instalação do OpenStack
  - glance (imagens)
  - cinder (volumes)
  - compute (nova)

13-03-2013

Executou:

- acompanhamento de Raphael na instalação do sistema operacional de uma máquina do data center
- reconfiguração das interfaces de redes das 11 máquinas
- implementação das regras de configuração de servidores para máquina 10
  - https://www.ncc.unesp.br/twiki/bin/view/Technical/NormasParaServidores?skin=clean.nat%2cnat#Ubuntu

Próximos passos

- implementação das regras de configuração de servidores para máquina 09
- finalização da leitura de Leitura parcial de: https://ncc.unesp.br/its/projects/educloud/wiki/Ambiente
- configuração do OpenStack

12-03-2013

Executou:

- Leitura de artigo sobre UEFI vs TPM (em especial sobre o secure boot e implementação no Suse.:
  - https://www.suse.com/blogs/uefi-secure-boot-overview/
  - https://www.suse.com/blogs/uefi-secure-boot-plan/
- Finalização da instalação local de uma máquina virtual para execução de serviços google (scrum).
  - testes de funcionamento local
  - testes de funcionamento via internet
- Documentou o passo-a-passo de instalação do sistema operacional das máquinas
- Leitura parcial de: https://ncc.unesp.br/its/projects/educloud/wiki/Ambiente

Próximos Passos:

- finalização da leitura de Leitura parcial de: https://ncc.unesp.br/its/projects/educloud/wiki/Ambiente

11-03-2013

Executou:

- Leitura dos emails de referência enviados por Beraldo
- Início da instalação local de uma máquina virtual para execução de serviços google (scrum).
  - primeira tentativa com qemu (falhou)
  - segunda tentativa com virtualbox (não terminada)

Próximos Passos:

- Finalização da instalação da máquina virtual

Ciclo 2 - Relatório final

Resumo

O conceito de computação em nuvem (em inglês, *cloud computing*) refere-se à utilização da memória, capacidade de armazenamento e poder computacional dos computadores interligados por meio da *Internet*, seguindo o princípio da computação em grade. Com a subutilização destes recursos computacionais em equipamentos modernos, surgiu a necessidade de compartilhar recursos de forma isolada, tornando assim uma máquina física em várias máquinas virtuais, cada uma servindo para um determinado propósito e com seus recursos bem definidos, tentando obter o máximo de aproveitamento dos recursos computacionais.

Prover uma infraestrutura de máquinas virtuais para clientes de forma escalável é hoje uma necessidade mundial e um desafio tecnológico. Este relatório tem por objetivo exibir as características e requisitos necessários para instalar e configurar uma infraestrutura de Cloud para prover máquinas virtuais aos usuários do projeto **EduCloud**.

Introdução

O projeto **EduCloud** é uma iniciativa da **Universidade Estadual Paulista - UNESP**, em parceria com a **Secretaria de Educação do estado de São Paulo**, **Intel** e **ACCEPT**. Esta iniciativa visa fornecer uma infraestrutura de computação em nuvem para hospedar sistemas de apoio ao ensino para as escolas da rede pública do estado de São Paulo.

Uma das grandes vantagens da computação em nuvem é a possibilidade de utilizar *softwares* remotamente, oferecidos na forma de serviços, mais comumente chamado de: *Software as a Service (SaaS)*, sem a necessidade de que esses estejam instalados no disco local do computador do usuário. Todavia o leque de vantagens trazidos pelo *SaaS* é mais amplo, não limitando-se unicamente à economia do recurso de disco do usuário[1]:

- Independência de plataforma, ou seja, é possível acessar a nuvem virtualmente a partir de qualquer computador e sistema operacional;
- As atualizações dos *softwares* são feitas de forma transparente ao usuário, sem necessidade que este intervenha;
- O trabalho colaborativo torna-se mais dinâmico e simples, uma vez que os arquivos criados pelo grupo são editados na nuvem, onde existe um controle de acessos simultâneos para garantir consistência das informações, e instantaneamente disponibilizados para todos os membros da equipe;
- Os *softwares* e os dados podem ser acessados em qualquer lugar, bastando que haja acesso à *Internet*, não estando mais restritos ao ambiente local de computação, nem dependendo da sincronização de mídias removíveis;
- O usuário tem um melhor controle de gastos ao usar aplicativos, pois a maioria dos sistemas de computação em nuvem fornece aplicações gratuitamente, e mesmo quando estas não são gratuitas, são pagas somente pelo tempo de utilização dos recursos. Não é necessário pagar por uma licença integral de uso de *software*;
- Diminuição dos gastos com manutenção da infraestrutura física de redes locais dos usuários, bem como da instalação dos *softwares* nos computadores corporativos, pois esta fica a cargo do provedor da estrutura de nuvem, bastando que os computadores clientes tenham acesso à *Internet*;
- A infraestrutura necessária para uma solução de cloud computing é mais enxuta do que uma solução tradicional de hosting ou collocation[17], consumindo menos energia, refrigeração e espaço físico e consequentemente contribuindo para preservação e uso racional dos recursos naturais.

Para implementar uma infraestrutura de nuvem, o **Núcleo de Computação Ciêntifica - NCC**, da **UNESP** pesquisou as principais ferramentas livres[2] do mercado para que juntas, executem esta tarefa.

Para prover uma infraestrutura de nuvem, faz-se atender as seguintes necessidades:

1. Disponibilizar máquinas virtuais de acordo com a demanda de forma automática e rápida, com diferentes configurações de memória, disco e *cpu*;
2. Entregar unidades persistentes de armazenamento de dados, juntamente às máquinas virtuais disponibilizadas;
3. Suportar controle de acesso de usuários e grupos que podem utilizar os recursos da nuvem, bem como um mecanismo de quota de recursos por usuário/grupo;
4. Armazenar as máquinas virtuais e os dados de forma descentralizada, possibilitando também que a arquitetura seja escalável facilmente.

Alguns *softwares* existem para suprir estas necessidades. No capítulo 1 falaremos sobre o projeto *OpenStack*[3], responsável por disponibilizar uma infraestrutura de máquinas virtuais e unidades de armazenamento para as máquinas virtuais. No capítulo 2 falaremos sobre o projeto *Hadoop*[4], responsável por gerenciamento e manipulação de grandes volumes de dados em ambientes largamente distribuídos. No capítulo 5, falaremos sobre os balanceadores de conexões e de máquinas virtuais, fundamental para balancear a carga da nuvem entre os nós *"workers"*, no capítulo 6 falaremos sobre o *OCCI*[5], uma interface de comunicação entre os serviços de cloud.

OpenStack

O OpenStack consiste em uma coleção de projetos open source criada para fornecer uma infraestrutura de computação em nuvem e armazenamento distribuído. A empresa Rackspace[6] e a NASA[7] são os principais contribuidores do conjunto de projetos. Hoje o consórcio possui mais de 100 membros, dentre as principais temos: Rackspace, AT&T, NASA, DreamHost, Safew Labs, Citrix, Dell, AMD, Intel, Canonical, Yahoo e HP. O projeto mantém uma página[8] com uma lista dos principais contribuidores do projeto.

No OpenStack, existem 3 principais componentes:

- Nova - Compute Infrastructure

Nova é um controlador de poder computacional para o OpenStack. Todas as atividades necessárias durante o ciclo de vida de uma instância (uma máquina virtual sendo executada) com o OpenStack são gerenciadas pelo Nova. Isto faz dele uma plataforma de gerenciamento que manipula recursos computacionais (e.g. rede), além de manter o controle de acesso sobre os recursos, e zelar pela escabilidade da infraestrutura de máquinas virtuais. O Nova não provê nenhuma funcionalidade de virtualização por si só; ele usa a interface libvirt[9] para interagir com os hypervisors[10] suportados.

- Swift - Storage Infrastructure

O Swift provê um armazenamento para os objetos (dados) de forma distribuída. Ele é análogo ao serviço da Amazon: Simple Storage Service, ou S3[13]. O Swift é capaz de armazenar bilhões de objetos distribuindos entre os nós. Ele possui um mecanismo de redundância e *failover* e é capaz de armazenar e transmitir os dados. Além disso o Swift é capaz de escalar arquivos considerados muito grandes (petabytes), mesmo que estes sejam em grande quantidade (número de objetos).

- Glance - Imaging Service

O Glance é um sistema de busca e recuperação de imagens de máquinas virtuais. Ele é responsável por gerenciar as imagens das máquinas virtuais, armazenando-as no sistema de arquivos local ou via o OpenStack Object Store - Swift.

Outros componentes, como um front end web (Horizon) e um serviço de autenticação (keystone), também fazem parte do projeto. O OpenStack está disponível no site do projeto[10] e está dispobilizado sobre a licença Apache[11].

O OpenStack atende as necessidades do projeto EduCloud por se tratar de um conjunto de ferramentas que permitem montar uma infra estrutura de cloud utilizando ferramentas livres. Cada subprojeto do OpenStack tem suas próprias características, mas de modo geral as principais vantagens do OpenStack são:

- Escalabilidade.
- Robustez.
- VNC proxy via web browser.
- API com suporte a REST nativo.
- Projeto com apoio de grande empresas.
- Larga adoção na iniciativa privada e comunidade do Software Livre.
- API com *"rate limite"*, ou seja, pode-se estabelecer quotas máximas de requisições na API.
- Comunidade ativa.
- Licença Livre.

As principais desvantagens são:

- Documentação dispersa.
- Muitos subprojetos, o que potencialmente dificulta a instalação e configuração.

Requisitos de Hardware

O OpenStack é uma infraestrutura complexa que exige um conjunto mínimo de algumas máquinas. Embora uma única máquina pode ser utilizada para fins de desenvolvimento, os requisitos mínimos para um ambiente de produção exige uma quantidade maior de máquinas.

A arquitetura mínima é suficiente para testes de staging. Uma vez que o EduCloud tenha passado por essa fase, faz-se necessário a adoção do cenário médio para que sejam tratadas as requisições do treinamento remoto oferecido pela FDE hoje. O cenário mínimo é utilizado apenas fins de desenvolvimento. O cenário médio é robusto o suficiente para substituir a atual arquitetura sobre o qual roda o treinamento hoje. Por fim, quando chegar o momento de se oferecer serviços adicionais através da infraestrutura do projeto EduCloud, tais como ferramentas de ensino à distância, será necessário expandir o cenário médio até que atinja-se o porte do cenário grande. Dessa forma a infraestrutura será suficiente para, além de oferecer o treinamento remoto já disponibilizado, suportar novos serviços educacionais a serem oferecidos a educadores e estudantes da rede pública de ensino do estado de São Paulo.

As recomendações de hardware foram retiradas do site do projeto OpenStack[11]. Abaixo temos 3 propostas de arquiteturas para o projeto EduCloud, com suas respectivas configurações:

Arquitetura proposta - mínima

https://ncc.unesp.br/its/attachments/189/openstack_small.png

Nesta arquitetura proposta, temos as seguintes máquinas:

- 05 Computer Nodes;
- 03 Storage Nodes;
- 04 Standard Servers (Storage Proxy, Nova Server, API Server e Network Controller)
- 04 Switches 10 GBoE;

Com as seguintes configurações:

- Computer Nodes: 2 x 64 bits QuadCore x86 CPU with HT support | 16 GB RAM | 2 x 2TB SATA | 2 x 10GBoE NICs

- Storage Nodes: 2 x 64 bits QuadCore x86 CPU with HT support | 16 GB RAM | 24 2T SATA | 2 x 10GBoE NICs
- Standard Servers: 2 x 64 bits QuadCore x86 CPU with HT support | 16 GB RAM | 2 x 2TB SATA | 2 x 10GBoE NICs

Arquitetura proposta - caso médio

https://ncc.unesp.br/its/attachments/190/openstack_medium.png

Nesta arquitetura proposta, temos as seguintes máquinas:

- 10 Computer Nodes;
- 05 Storage Nodes;
- 08 Standard Servers (Storage Proxy, Nova Server, API Server, Network Controller, VPN, etc..)
- 04 Switches 10 GBoE;

Com as seguintes configurações:

- Computer Nodes: 2 x 64 bits QuadCore x86 CPU with HT support | 32 GB RAM | 2 x 2TB SATA | 2 x 10GBoE NICs
- Storage Nodes: 2 x 64 bits QuadCore x86 CPU with HT support | 32 GB RAM | 24 2T SATA | 2 x 10GBoE NICs
- Standard Servers: 2 x 64 bits QuadCore x86 CPU with HT support | 32 GB RAM | 2 x 2TB SATA | 2 x 10GBoE NICs

Arquitetura proposta - melhor caso

https://ncc.unesp.br/its/attachments/191/openstack_big.png

Nesta arquitetura proposta, temos as seguintes máquinas:

- 20 Computer Nodes;
- 10 Storage Nodes;
- 13 Standard Servers (Storage Proxy, Nova Server, API Server, Network Controller, VPN, etc..)
- 08 Switches 10 GBoE;

Com as seguintes configurações:

- Computer Nodes: 2 x 64 bits QuadCore x86 CPU with HT support | 32 GB RAM | 2 x 2TB SATA | 2 x 10GBoE NICs
- Storage Nodes: 2 x 64 bits QuadCore x86 CPU with HT support | 32 GB RAM | 24 2T SATA | 2 x 10GBoE NICs
- Standard Servers: 2 x 64 bits QuadCore x86 CPU with HT support | 32 GB RAM | 2 x 2TB SATA | 2 x 10GBoE NICs

Tecnologias alternativas

Várias tecnologias podem ser utilizadas para provêr uma infraestrutura de máquinas virtuais. Algumas delas na forma de solução livre, podendo ser instalada em infraestrutura própria, e outras na forma de serviço web, sendo executada na infraestrutura da empresa que a provê.

Entre as tecnologias livres mais conhecidas, temos:

1. Abiquo (na versão mantida pela comunidade).
2. Nimbus.
3. OpenStack.
4. Open Nebula.
5. Eucalyptus.

Entre as tecnologias funcionando como serviço web, temos:

1. Amazon Web Service.
2. Linode.
3. Rackspace.
4. Gogrid.

Estas são apenas as principais alternativas que atendem os requisitos do projeto, levando em consideração popularidade, quantidade de downloads/mês e nível de maturidade do projeto.

Balanceador de conexões HTTP

Um balanceador de conexões é o elemento responsável por redirecionar o fluxo de pacotes de redes vindo da Internet para os *workernodes* na rede interna, dividindo as conexões de forma mais justa possível para evitar sobre carga em um nó específico. Desta forma, se temos 10 servidores web, o balanceador recebe as conexões e encaminha para os 10 balanceadores dependendo do algoritmo por ele suportado.

Como as conexões HTTP são *stateless*, é fundamental que o balanceador tenha a capacidade de enviar sempre para o mesmo servidor conexões vindas de um mesmo cliente. O balanceador também é responsável por adicionar ou remover *workernodes* da fila de máquinas sempre que necessário.

Nginx vs HAProxy

Durante esta etapa foi feita uma pesquisa inicial sobre os projetos Nginx[14] e HAProxy[15]. Todas as duas tecnologias se encaixam, com vantagens e desvantagens no projeto EduCloud.

O Nginx (pronuncia-se em inglês: Engine X) é um servidor HTTP + proxy reverso e está disponibilizado pela licença BSD. Ele funciona no modo de *load balance* via um módulo externo.

Já o HAProxy nao é um servidor HTTP (ele não serve páginas estáticas), foi feito exclusivamente para ser um proxy e *loadbalance* não só HTTP mas para qualquer protocolo TCP. Seu código está licenciado sob a GPL.

Nenhuma das tecnologias conflita com demais componentes do projeto EduCloud, porém o Nginx possui algumas limitações.

Nginx é um servidor e proxy reverso HTTP de alta performance, gratuito e livre, bem como um servidor proxy para IMAP/POP3.

Foi criado pelo programador russo Igor Sysoev em 2005. Segundo pesquisa da Alexa, reportada pela W3Techs. Em presquisa realizada pela Netcraft em Julho de 2010[16], o Nginx é usado em 5.21% dos domínios, tornando-se o quarto servidor web mais popular do mundo.

E funciona em UNIX, Linux, BSD, Mac OS, Solaris, e Microsoft Windows.

Todas as duas tecnologias possui um desenvolvimento ativo.

O HAProxy é usado por empresas como Twitter, Farmville, Fedora, github, Reddit, Stack Overflow, Tumblr, entre outras empresas.

Todos os dois projetos existem uma boa documentação, embora elas estejam espalhadas.

Desvantagens do Nginx

- Possui um algoritmo simples de *round-robin*. Se as requisições são pequenas e retornam rapidamente, o ngnix atende perfeitamente. Porém se a página demora para ser atendida, o Ngnix enviará novas conexões para máquinas que ainda estão atendendo requisições, fazendo com que algumas máquinas fiquem sobrecarregadas e outras não.
- Existem módulos de terceiros para o Ngnix que torce esse balanceamento mais justo, mas ainda não está maduro.
- O Ngnix apenas reconhece um host como *down*, quando o servidor retorna um erro com código 500.
- Apenas atende HTTP.

Desvantagens do HAProxy

Durante a pesquisa inicial a única desvantagem encontrada no HAProxy é que nativamente ele não suporta HTTPS. Porém alguns módulos extras permitem adicionar esta funcionalidade.

Referências

1. O que é computação em nuvem?
2. Definição de software livre
3. http://www.gnu.org/philosophy/free-sw.html
4. Projeto Hadoop
5. Open Cloud Compute Interface - OCCI
6. Rackspace
7. Nasa
8. OpenStack - Empresas Participantes
9. The virtualization API
10. Hypervisor
11. Projeto OpenStack
12. Apache License
13. Amazon Simple Storage Service
14. nginx [engine x] an HTTP and reverse proxy server
15. HAProxy, The Reliable, High Performance TCP/HTTP Load Balancer
16. Netcraft, web server survey
17. Cloud Colocation and data center real estate analysis

Files

openstack_small.png	65.7 KB	30 Apr 2012	Beraldo Leal
openstack_medium.png	95 KB	30 Apr 2012	Beraldo Leal
openstack_big.png	157 KB	30 Apr 2012	Beraldo Leal

Documentação do projeto EduCloud

- [[Ciclo 1 - Open Nebula, Brain Honey e Balanceador de Carga]]
- [[Ciclo 2 - Relatório final]]
- [[Ciclo 3 - OpenStack + SDN; DeltaCloud; Puppet]]

- [[OpenstackCriticalBugs| Openstack - Critical Bugs]]
- [[Ambiente|Ambiente de Desenvolvimento]]
- [[ipmi|Procedimentos IPMI]]
- [[Scrum]]
- [[Resultados Benchmark]]

Ciclo 3 - OpenStack + SDN, DeltaCloud e Puppet

- [[SDN]]
- [[DeltaCloud]]
- [[Puppet]]

- [[Referências_Keystone_+_LDAP|Referências Keystone + LDAP]]
- [[OpenStack Networking|OpenStack Networking -- Neutron]]

OpenFlow Documentation

Specific Protocol between the controller and the OF switches. And a set of operations/messages sent to the switches.

- Original Paper: OpenFlow: Enabling Innovation in Campus Networks
- Original site: www.opennetworking.org
- Old site: www.openflow.org
- Specifications: 1.4.0, 1.3.2, 1.3.1, 1.3.0, 1.2.0, 1.1.0, 1.0.0

Novos materiais (por tecepe)

hcprobe

- hcprobe
- Advanced Study of SDN/OpenFlow Controllers

Artigos interessantes

- Evaluating OpenFlow Controller Paradigms
- Parâmetros balizadores para experimentos com comutadores OpenFlow : avaliação experimental baseada em medições de alta precisão
- The Beacon OpenFlow Controller
- POX CONTROLLER PERFORMANCE FOR OPENFLOW NETWORKS
- Monitoring latency with OpenFlow

Artigo de possível interesse:

- Design and Implementation of an Efficient and Programmable Future Internet Testbed in Taiwan

Artigos que não tive acesso e podem ser interessantes

- Simulative Comparison of Multiprotocol Label Switching and OpenFlow Network Technologies for Transmission Operations
- Formal Modeling and Verification of SDN-OpenFlow
- Fast, accurate simulation for SDN prototyping
- Modeling and performance evaluation of an OpenFlow architecture
- Metrics and Measurement Tools in OpenFlow and the OFELIA Testbed
- OpenFlow: Why latency does matter

OpenFlow Basics and Overview

- What OpenFlow is and more importantly, what it's not
- Origins and Evolution of OpenFlow/SDN
- OpenFlow components
- Notes on OpenFlow
- OpenFlow is like IPv6
- OpenFlow, Controllers – Whats missing in Routing Protocols today?
- OpenFlow protocol guide: SDN controllers and apps

OpenFlow Scalability and Performance

- On data center scale, OpenFlow, and SDN
- The Scaling Implications of SDN
- On Controller Performance in Software-Defined Networks
- Performance Analysis of OpenFlow Hardware erro 404
- OpenFlow Switching Performance
- Performance Evaluation of OpenFlow Switches
- Why OpenFlow Hardware Control Matters in the Data Center?
- CAP for Networks
- Controller Performance Comparisons
- Evaluation of OF Controllers
- OFLOPS: An Open Framework for OpenFlow Switch Evaluation
- Beacon Benchmarking Guidelines

OpenFlow Programming Languages

- Languages for Software-Defined Networks
- Frenetic: A Network Programming Language
- Composing Software-Defined Networks
- Participatory Networking: An API for Application Control of SDNs
- A Compile and Run-time System for Network Programming Languages
- Machine-Verified Network Controllers

- Nettle: Functional Reactive Programming of OpenFlow Networks
- Procera: A Language for High-Level Reactive Network Control
- FatTire: Declarative Fault Tolerance for Software-defined Networks
- A Balance of Power: Expressive, Analyzable Controller Programming

OpenFlow Federation and Isolation

- Hierarchical Policies for Software Defined Networks
- Splendid Isolation: A Slice Abstraction for Software-Defined Networks

OpenFlow Testing, Verification, Evaluation, Debugging and Tracing

- ONF Testing and Interoperability Working Group
- OFRewind: Enabling Record and Replay Troubleshooting for Networks
- Where Is the Debugger for My Software-Defined Network?
- Logically Centralized? State Distribution Trade-offs in Software Defined Networks
- Veriflow: Verifying Network-Wide Invariants in Real Time
- Header Space Analysis: Static Checking for Networks
- Real Time Network Policy Checking using Header Space Analysis
- A NICE Way to Test OpenFlow Applications
- A SOFT Way for OpenFlow Interoperability Testing
- FlowChecker: Configuration, Analysis, and Verification of Federated OpenFlow Infrastructures

OpenFlow Emulation and Simulation

- A Network in a Laptop: Rapid Prototyping for Software-Defined Networks
- Fast, Accurate Simulation for SDN Prototyping
- High-Fidelity Switch Models for Software-Defined Network Emulation

OpenFlow HOWTOs and Tutorials

- Original OpenFlow Tutorial
- OpenFlow - SDN for engenniers
- OpenFlow/SDN Beginner's Tutorial
- OpenFlow 1.3 Tutorial by CPqD

OpenFlow Software

OpenFlow Controllers

Discussão sobre características a serem avaliadas

[[OpenFlowControllers_Features|Características e funcionalidades]]

Análise dedicada

- [[Ryu]]
- [[NOX]]
- [[POX]]
- [[Trema]]
- [[Floodlight]]
- [[OpenDaylight]]

Resumo (desatualizado)

[[OpenFLOW Controllers]]

OpenFlow Controllers Frameworks

OpenFlow Hypervisors

OpenFlow Test Tools

OpenFlow Switches

- OpenFlow 1.3 Software Switch

Trema

Sobre o código

Linguagem

Linguagem utilizada para o desenvolvimento.

Licença

Licença em que o código foi liberado. A preferência é por licenças livres, facilitando assim o estudo e desenvolvimento de aplicações ou até mesmo de correções de problemas.

Suporte a versões do OpenFLOW

- Quais são as versões de OpenFlow suportadas pelo controlador / framework.
- Como isso é implementado

Requisitos

- pacotes a serem instalados
- espaço de armazenamento exigido
- outros recursos necessários ao funcionamento do controlador.

Independência de linguagem de programação

Quais linguagens são suportadas na implementação de aplicações e componentes.

Modularidade

Quanto e que tipo de modularidade o controlador oferece.

Repositório

Onde encontrar o código.

Recursos básicos do controlador

Recursos oferecidos pelo controlador sem a necessidade de aplicações, como por exemplo:

- GRE tunneling
- VLAN support

Aplicações

Componentes disponíveis que acrescentam ferramentas ao controlador, como por exemplo:

- topology discovery
- MAC based segregation

Testes

Existência de testes de unidades no codigo (scripts ou códigos), geralmente em uma pasta em separado, que validam e testam o código desenvolvido.

Disponibilidade nas distribuições GNU/Linux

Disponibilidade e facilidade de instação nas distribuição GNU/Linux relevantes atuais.

- disponível nos repositórios da distribuição?
- versão dos pacotes disponíveis
- ênfase nas distros Debian e Ubuntu.

Documentação

Nível de documentação do controlador, incluindo tutoriais e exemplos (tanto de estruturas quanto de aplicações).

Suporte junto ao OpenStack

Como o ambiente estruturado se baseia no OpenStack, é interessante saber qual é o tipo de suporte oferecido a essa ferramenta.

Quem usa

Público do controlador

Quem está por trás

Empresas, consórcio, comunidade, universidade, ...

último release / commit

Uma das formas de se ter noção de quanto o desenvolvimento ainda está ativo é procurar pela data da última atualização de código no repositório.

Outras características

Características que não se adequam a nenhuma das anteriores.

SDN

Referências sobre SDN

What Is SDN All About, Then?
http://www.noxrepo.org/2012/03/sdn/

OpenFlow

- [[Referências sobre OpenFlow|OpenFlow]]

Referências Keystone + LDAP

Algumas referências sobre integração LDAP + Keystone

Material da Folsom
http://docs.openstack.org/folsom/openstack-compute/admin/content/configuring-keystone-for-ldap-backend.html

Keystone com open LDAP e horizon (grizzly)
https://ask.openstack.org/en/question/1350/how-to-configure-keystone-with-open-ldap-horizon-on-grizzly/

Change I672b90e6: Read-only default domain for LDAP (bug 1168726)
https://review.openstack.org/#/c/28197

Openstack Keystone LDAP Redux
http://adam.younglogic.com/2012/02/openstack-keystone-ldap-redux/

Re: New schema for LDAP + Keystone Grizzly?
https://lists.launchpad.net/openstack/msg22556.html

How to configure Keystone with open LDAP + horizon on grizzly
https://lists.launchpad.net/openstack/msg23814.html

Enable the Keystone LDAP back end in OpenStack
http://www.ibm.com/developerworks/cloud/library/cl-ldap-keystone/

Referências sobre SDN

Lista de Referências sobre SDN, Openflow e Controladores

What Is SDN All About, Then?
http://www.noxrepo.org/2012/03/sdn/