Crash Course in Open Source Cloud Computing

Build and manage clouds with free and open source tools

Mark R. Hinkle
VP of Community
Cloud.com
mrhinkle@cloud.com
mrhinkle@gmail.com
Twitter: @mrhinkle
%\texttt{whoami}

- Responsible for Driving Adoption of CloudStack Open Source Cloud Computing Software
- Former manager of Zenoss Open Source project 100,000 users, 1.5 million downloads
- Former Linux Desktop Advocate (Zealot?)
- Former LinuxWorld Magazine Editor-in-Chief
- Open Management Consortium Conspirator
- Open Desktop Consortium Instigator
- Author - “Windows to Linux Business Desktop Migration” - Thomson
- NetDirector Project - Open Source Configuration Management Project
- Sometimes Author and Blogger at SocializedSoftware.com/NetworkWorld
- Start-up junkie, Glutton for punishment
Cloud Computing Adoption

- 365,000 web sites are running on Amazon EC2 - Netcraft May 2010
- £76 billion spent on Cloud Computing in the U.K. in 2010 - IT Candor
- Cloud Computing will be a $126 billion market by 2012 - IBM
- 20% of Businesses won’t have IT Assets by 2012 - Gartner

Bottom Line: A large portion of our infrastructure no longer lives in our data center... but we gotta manage it.
Open Source Cloud Adoption

- 41 of users prefer to deploy servers virtually, 29.3% use virtualization whenever possible

- Xen is 2nd most popular virtualization technology after VMware followed by Linux Kernel-Based Virtual Machines (KVM) was fourth with 21.3%

- 49.2% of respondents indicated they planned to deploy hosted Linux servers in 2010 while 32.6% indicated that they would be deploying hosted Microsoft Windows.

- 50.8% indicated they used no specific management tools for cloud computing, 33.3% indicated using tools provided by their hosting provider

Five Characteristics of Clouds

- On-demand self-service
- Broad network access
- Resource pooling
- Rapid Elasticity
- Measured Service
Three Types of “Clouds”

**Application Cloud a.k.a. Software-as-a-Service**

Single application, multi-tenancy, network-based, one-to-many delivery of applications, all users have same access to features.

*Examples: Salesforce.com, Google Docs, Red Hat Network/RHEL*

**Compute Cloud a.k.a. Platform-as-a-Service**

Application developer model, Application deployed to an elastic service that autoscales, low administrative overhead. No concept of virtual machines or operating system. Code it and deploy it.

*Examples: Google AppEngine, Windows Azure, Rackspace Site, Red Hat Makara*

**Compute Cloud a.k.a Infrastructure-as-a-Service**

Servers and storage are made available in a Scalable way over a network.

*Examples: EC2, Rackspace CloudFiles, OpenStack, CloudStack, Eucalyptus, Ubuntu Enterprise Cloud, OpenNebula*
Use Cases for Cloud Computing

- Offer Software-as-a-Service
- Development/Test Labs
- Resell Hosting Services
- Mimic Public Cloud Computing Behind a Firewall
- High Performance Computing (HPC)
Anatomy of the Cloud

- Cloud Application (SaaS)
- Cloud Software Environment (PaaS)
- Cloud Infrastructure (IaaS)

- Virtualized Resources
  - Compute
  - Networking
  - Storage
  - Etc.

- Virtualized Images
  - Image metadata

- Software Kernel (OS, VM Manager)
- Firmware, Hardware

- Security

- Management
  - Reporting
  - SLA Mgmt
  - Capacity Planning
  - Billing

- Metering
- Provisioning
- Billing
Hypervisors

Open Source

- Xen Cloud Platform (XCP)
- KVM – Kernel-based Virtualization
- VirtualBox* - Oracle supported VMs
- OpenVZ* - Container-based, Similar to Solaris Containers or BSD Zones
- LXC – User Space, chrooted installs

Proprietary

- VMware
- Citrix Xenserver
- Microsoft Hyper-V
# Virtual Machine Formats

<table>
<thead>
<tr>
<th>VM Format</th>
<th>Abbreviation</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Open Virtualization Format</strong></td>
<td>OVF</td>
<td>Evolving Standard, meta</td>
</tr>
<tr>
<td><strong>Virtual Machine Disk</strong></td>
<td>VMDK</td>
<td>Encodes a single disk, specific to VMware</td>
</tr>
<tr>
<td><strong>QEMU Copy-on-Write</strong></td>
<td>QCOW2</td>
<td>Allows you to do RAW disk with snapshotting and provisioning utilities used by KVM</td>
</tr>
<tr>
<td><strong>Virtual Hard Disk</strong></td>
<td>VHD</td>
<td>Virtual Hard disk format was pioneered by Microsoft but adopted by many virtualization including Xen and Virtual Box</td>
</tr>
<tr>
<td><strong>Amazon Machine Image</strong></td>
<td>AMI</td>
<td>Amazon’s virtual machine format</td>
</tr>
</tbody>
</table>

*QEMU is a processor virtualization technology that also has a utility to convert VM formats. (qemu-img)*
Public Cloud Services & Private Clouds

Popular Public Cloud Services

Open Source Cloud Computing Software
## Open Source Cloud Computing

<table>
<thead>
<tr>
<th>Product</th>
<th>Year Started</th>
<th>License</th>
<th>Hypervisors Supported</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloudstack</td>
<td>2010</td>
<td>GPL</td>
<td>Xenserver, VMware, KVM, Hyper-V</td>
</tr>
<tr>
<td>Eucalyptus</td>
<td>2008</td>
<td>GPL</td>
<td>Xen, KVM, VMware</td>
</tr>
<tr>
<td>OpenStack</td>
<td>2010</td>
<td>Apache</td>
<td>Xen, KVM, VMware, LXC, Hyper-V</td>
</tr>
<tr>
<td>Ubuntu Enterprise Server (UEC)</td>
<td>2009 (Karmic Koala)</td>
<td>GPL</td>
<td>Xen,KVM,</td>
</tr>
<tr>
<td>Abiquo</td>
<td>2009 (Development 2006)</td>
<td></td>
<td>VMware, Hyper-V Citrix XenServer Virtual Box Xen KVM</td>
</tr>
</tbody>
</table>
Here an API, there an API, Everywhere an API

Every cloud has its own API, so how do you avoid lock-in and migrate between clouds, libraries to write to and translate to different cloud APIs?

- **jclouds** – Java, active development
- **libcloud** – Started by CloudKick, now an Apache incubator project
- **deltacloud** – Started by Red Hat, now an Apache project
# Open Source Cloud Storage

<table>
<thead>
<tr>
<th></th>
<th>Year Started</th>
<th>License</th>
</tr>
</thead>
<tbody>
<tr>
<td>OpenStack Storage (Swift)</td>
<td>2010</td>
<td>GPL</td>
</tr>
<tr>
<td>Gluster</td>
<td>2008</td>
<td>GPL</td>
</tr>
<tr>
<td>Sheepdog (KVM and BTRFS only)</td>
<td>2009</td>
<td>GPL</td>
</tr>
</tbody>
</table>
I JUST LAUNCHED 1,000 INSTANCES IN THE CLOUD....NOW WHAT DO I DO?
Cloud Computing Changes Everything

- MeatCloud, Can’t Keep up with Cloud Computing
- Devops & Agile IT Philosophy
- Script Repetitive Tasks
- Automate, Automate, Automate
This Guy doesn’t Automate
DEVOPS
Cultural Changes to Help Cloud Adoption

Agile IT and DevOps movements mean fast not loose

- Break down Silos within IT - Operations and Developers should collaborate with each other to deliver excellent products
- Systems Administrators need to become Systems Engineers building automated, fault tolerant systems not just maintaining infrastructure
- More frequent changes, more outages (albeit short) to rapidly improve IT products and services
- Process, version control, and automation are important
4 Core Types of Server Management Tools

Provisioning
Installation of operating systems and other software

Configuration Management
Sets the parameters for servers, can specify installation parameters

Orchestration/Automation
Automate tasks across systems

Monitoring
Records errors and health of IT infrastructure
Open Source Management Tools Adoption

- 98% of enterprises use open source systems management tools
- 76% indicate they prefer to use open source whenever possible
- Compelling factors for using open source is flexibility followed by cost savings
- 50% are already using some form of cloud technology including but not limited to hosted applications, Amazon Web services and/or hosted storage
- Top IT management priorities for 2010: monitoring, configuration management, patching and provisioning and security
## Comparison of Provisioning Tools

<table>
<thead>
<tr>
<th>Tool</th>
<th>Year Started</th>
<th>Language</th>
<th>License</th>
<th>Installation Targets</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cobbler</td>
<td>2007</td>
<td>Python</td>
<td>GPL</td>
<td>Red Hat, OpenSUSE Fedora, Debian, Ubuntu</td>
</tr>
<tr>
<td>Fully Automatic Installation (FAI)</td>
<td>2000</td>
<td>Perl</td>
<td>GPL</td>
<td>Debian</td>
</tr>
<tr>
<td>Kickstart</td>
<td>?</td>
<td>Python</td>
<td>GPL</td>
<td>Most .deb and RPM based Linux distros</td>
</tr>
<tr>
<td>Spacewalk</td>
<td>2008</td>
<td>Perl, Python, Java</td>
<td>GPL</td>
<td>Fedora, Centos</td>
</tr>
<tr>
<td>Viper</td>
<td>2008</td>
<td>Perl</td>
<td>GPL</td>
<td>Debian</td>
</tr>
</tbody>
</table>
## Comparison of Configuration Management Tools

<table>
<thead>
<tr>
<th>Tool</th>
<th>Year Started</th>
<th>Language</th>
<th>License</th>
<th>Client/Server</th>
</tr>
</thead>
<tbody>
<tr>
<td>bcfg2</td>
<td>2003</td>
<td>Python</td>
<td>BSD</td>
<td>Yes</td>
</tr>
<tr>
<td>Cfengine</td>
<td>1993</td>
<td>C</td>
<td>Apache</td>
<td>Yes</td>
</tr>
<tr>
<td>Chef</td>
<td>2009</td>
<td>Ruby</td>
<td>Apache</td>
<td>Chef Solo – No Chef Server - Yes</td>
</tr>
<tr>
<td>Puppet</td>
<td>2004</td>
<td>Ruby</td>
<td>GPL</td>
<td>yes</td>
</tr>
</tbody>
</table>
# Comparison of Open Source Monitoring Tools

<table>
<thead>
<tr>
<th>Year Started</th>
<th>License</th>
<th>Language</th>
<th>Type of Monitoring</th>
<th>Collection Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cacti</td>
<td>2001</td>
<td>GPL</td>
<td>PHP</td>
<td>Performance</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SNMP, syslog</td>
</tr>
<tr>
<td>Nagios</td>
<td>1999</td>
<td>GPL</td>
<td>C/PHP</td>
<td>Availability</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SNMP, TCP, ICMP, IPMI, syslog</td>
</tr>
<tr>
<td>Zabbix</td>
<td>2001</td>
<td>GPL</td>
<td>C/PHP</td>
<td>Availability/ Performance and more</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SNMP, TCP/ ICMP, IPMI, Synthetic Transactions</td>
</tr>
<tr>
<td>Zenoss</td>
<td>2005</td>
<td>GPL</td>
<td>Python</td>
<td>Availability, Performance, Event Management</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SNMP, ICMP, SSH, syslog, WMI</td>
</tr>
</tbody>
</table>
## Comparison of Open Source Automation/Orchestration Tools

<table>
<thead>
<tr>
<th>Tool</th>
<th>Year Started</th>
<th>Language</th>
<th>License</th>
<th>Client/Server</th>
<th>Support Organization</th>
</tr>
</thead>
<tbody>
<tr>
<td>AutomateIT</td>
<td>2009</td>
<td>Ruby</td>
<td>GPL</td>
<td>No</td>
<td>None</td>
</tr>
<tr>
<td>Capistrano</td>
<td>2006</td>
<td>Ruby</td>
<td>MIT</td>
<td>Yes</td>
<td>None</td>
</tr>
<tr>
<td>Control Tier</td>
<td>2005</td>
<td>Java</td>
<td>Apache</td>
<td>Yes</td>
<td>DTO Solutions</td>
</tr>
<tr>
<td>Func</td>
<td>2007</td>
<td>Python</td>
<td>GPL</td>
<td>Yes</td>
<td>Fedora Project</td>
</tr>
<tr>
<td>RunDeck</td>
<td>2010</td>
<td>Java</td>
<td>Apache</td>
<td>Yes</td>
<td>DTO Solutions</td>
</tr>
</tbody>
</table>