cloudstack DOCUMENTATION

Apache CloudStack

Version 4.2.0 Release Notes

cloudstack™
open source cloud computing

Apache CloudStack

Legal Notice

Abstract

Release notes for the Apache CloudStack 4.2.0 release.

Preface

1. Document Conventions

This manual uses several conventions to highlight certain words and phrases and draw attention to specific pieces of information.

In PDF and paper editions, this manual uses typefaces drawn from the Liberation Fonts set. The Liberation Fonts set is also used in HTML editions if the set is installed on your system. If not, alternative but equivalent typefaces are displayed. Note: Red Hat Enterprise Linux 5 and later includes the Liberation Fonts set by default.

1.1. Typographic Conventions

Four typographic conventions are used to call attention to specific words and phrases. These conventions, and the circumstances they apply to, are as follows.

`Mono-spaced Bold`

Used to highlight system input, including shell commands, file names and paths. Also used to highlight keycaps and key combinations. For example:

> To see the contents of the file **`my_next_bestselling_novel`** in your current working directory, enter the **`cat my_next_bestselling_novel`** command at the shell prompt and press **`Enter`** to execute the command.

The above includes a file name, a shell command and a keycap, all presented in mono-spaced bold and all distinguishable thanks to context.

Key combinations can be distinguished from keycaps by the hyphen connecting each part of a key combination. For example:

> Press **`Enter`** to execute the command.
>
> Press **`Ctrl`**+**`Alt`**+**`F2`** to switch to the first virtual terminal. Press **`Ctrl`**+**`Alt`**+**`F1`** to return to your X-Windows session.

The first paragraph highlights the particular keycap to press. The second highlights two key combinations (each a set of three keycaps with each set pressed simultaneously).

If source code is discussed, class names, methods, functions, variable names and returned values mentioned within a paragraph will be presented as above, in **`mono-spaced bold`**. For example:

> File-related classes include **`filesystem`** for file systems, **`file`** for files, and **`dir`** for directories. Each class has its own associated set of permissions.

Proportional Bold

This denotes words or phrases encountered on a system, including application names; dialog box text; labeled buttons; check-box and radio button labels; menu titles and sub-menu titles. For example:

> Choose **System** → **Preferences** → **Mouse** from the main menu bar to launch **Mouse Preferences**. In the **`Buttons`** tab, click the **`Left-handed mouse`** check box and click **`Close`** to switch the primary mouse button from the left to the right (making the mouse suitable for use in the left hand).
>
> To insert a special character into a **gedit** file, choose **Applications** → **Accessories** → **Character Map** from the main menu bar. Next, choose **Search** → **Find…** from the **Character Map** menu bar, type the name of the character in the **`Search`** field and click **`Next`**. The character you sought will be highlighted in the **`Character Table`**. Double-click this highlighted character to place it in the **`Text to copy`** field and then click the **`Copy`** button. Now switch back to your document and choose **Edit** → **Paste** from the **gedit** menu bar.

The above text includes application names; system-wide menu names and items; application-specific menu names; and buttons and text found within a GUI interface, all presented in proportional bold and all distinguishable by context.

`Mono-spaced Bold Italic` or ***`Proportional Bold Italic`***

Whether mono-spaced bold or proportional bold, the addition of italics indicates replaceable or variable text. Italics denotes text you do not input literally or displayed text that changes depending on circumstance. For example:

> To connect to a remote machine using ssh, type **`ssh`** ***`username@domain.name`*** at a shell prompt. If the remote machine is **`example.com`** and your username on that machine is john, type **`ssh john@example.com`**.
>
> The **`mount -o remount`** ***`file-system`*** command remounts the named file system. For example, to

remount the **/home** file system, the command is **mount -o remount /home**.

To see the version of a currently installed package, use the **rpm -q *package*** command. It will return a result as follows: ***package-version-release***.

Note the words in bold italics above — username, domain.name, file-system, package, version and release. Each word is a placeholder, either for text you enter when issuing a command or for text displayed by the system.

Aside from standard usage for presenting the title of a work, italics denotes the first use of a new and important term. For example:

Publican is a *DocBook* publishing system.

1.2. Pull-quote Conventions

Terminal output and source code listings are set off visually from the surrounding text.

Output sent to a terminal is set in **mono-spaced roman** and presented thus:

```
books        Desktop   documentation  drafts  mss    photos   stuff  svn
books_tests  Desktop1  downloads      images  notes  scripts  svgs
```

Source-code listings are also set in **mono-spaced roman** but add syntax highlighting as follows:

```
package org.jboss.book.jca.ex1;

import javax.naming.InitialContext;

public class ExClient
{
   public static void main(String args[])
       throws Exception
   {
      InitialContext iniCtx = new InitialContext();
      Object         ref    = iniCtx.lookup("EchoBean");
      EchoHome       home   = (EchoHome) ref;
      Echo           echo   = home.create();

      System.out.println("Created Echo");

      System.out.println("Echo.echo('Hello') = " + echo.echo("Hello"));
   }
}
```

1.3. Notes and Warnings

Finally, we use three visual styles to draw attention to information that might otherwise be overlooked.

Note

Notes are tips, shortcuts or alternative approaches to the task at hand. Ignoring a note should have no negative consequences, but you might miss out on a trick that makes your life easier.

Important

Important boxes detail things that are easily missed: configuration changes that only apply to the current session, or services that need restarting before an update will apply. Ignoring a box labeled 'Important' will not cause data loss but may cause irritation and frustration.

Warning

Warnings should not be ignored. Ignoring warnings will most likely cause data loss.

2. Feedback

to-do

Chapter 1. Welcome to CloudStack 4.2

Welcome to the 4.2.0 release of CloudStack, the second major release from the Apache CloudStack project since its graduation from the Apache Incubator. CloudStack 4.2 includes more than 70 new features and enhancements. The focus of the release is on three major areas:

- Improved support for both legacy-style and cloud-style workloads
- New third-party plug-in architecture

- New third-party plug-in architecture
- Networking enhancements

In addition to these major new areas of functionality, CloudStack 4.2 provides many additional enhancements in a variety of product areas. All of the new features are summarized later in this Release Note.

This document contains information specific to this release of CloudStack, including upgrade instructions from prior releases, new features added to CloudStack, API changes, and issues fixed in the release. For installation instructions, please see the Installation Guide. For usage and administration instructions, please see the CloudStack Administrator's Guide. Developers and users who wish to work with the API will find instruction in the CloudStack API Developer's Guide

If you find any errors or problems in this guide, please see Section 2, “Feedback”. We hope you enjoy working with CloudStack!

Chapter 2. What's New in 4.2.0

CloudStack 4.2 includes the following new features.

2.1. Features to Support Heterogeneous Workloads

The following new features help CloudStack 4.2 better support both legacy and cloud-era style zones.

2.1.1. Regions

To increase reliability of the cloud, you can optionally group resources into geographic regions. A region is the largest available organizational unit within a cloud deployment. A region is made up of several availability zones, where each zone is equivalent to a datacenter. Each region is controlled by its own cluster of Management Servers, running in one of the zones. The zones in a region are typically located in close geographical proximity. Regions are a useful technique for providing fault tolerance and disaster recovery.

By grouping zones into regions, the cloud can achieve higher availability and scalability. User accounts can span regions, so that users can deploy VMs in multiple, widely-dispersed regions. Even if one of the regions becomes unavailable, the services are still available to the end-user through VMs deployed in another region. And by grouping communities of zones under their own nearby Management Servers, the latency of communications within the cloud is reduced compared to managing widely-dispersed zones from a single central Management Server.

Usage records can also be consolidated and tracked at the region level, creating reports or invoices for each geographic region.

2.1.2. Object Storage Plugin Architecture

Artifacts such as templates, ISOs and snapshots are kept in storage which CloudStack refers to as secondary storage. To improve scalability and performance, as when a number of hosts access secondary storage concurrently, object storage can be used for secondary storage. Object storage can also provide built-in high availability capability. When using object storage, access to secondary storage data can be made available across multiple zones in a region. This is a huge benefit, as it is no longer necessary to copy templates, snapshots etc. across zones as would be needed in an NFS-only environment.

Object storage is provided through third-party software such as Amazon Simple Storage Service (S3) or any other object storage that supports the S3 interface. These third party object storages can be integrated with CloudStack by writing plugin software that uses the object storage plugin capability introduced in CloudStack 4.2. Several new pluggable service interfaces are available so that different storage providers can develop vendor-specific plugins based on the well-defined contracts that can be seamlessly managed by CloudStack.

2.1.3. Zone-Wide Primary Storage

(Supported on KVM and VMware)

In CloudStack 4.2, you can provision primary storage on a per-zone basis. Data volumes in the primary storage can be attached to any VM on any host in the zone.

In previous CloudStack versions, each cluster had its own primary storage. Data in the primary storage was directly available only to VMs within that cluster. If a VM in a different cluster needed some of the data, it must be copied from one cluster to another, using the zone's secondary storage as an intermediate step. This operation was unnecessarily time-consuming.

2.1.4. VMware Datacenter Now Visible As a CloudStack Zone

In order to support zone-wide functions for VMware, changes have been made so that CloudStack is now aware of VMware Datacenters and can map each Datacenter to a CloudStack zone. Previously, CloudStack was only aware of VMware Clusters, a smaller organizational unit than Datacenters. This implies that a single CloudStack zone could possibly contain clusters from different VMware Datacenters. In order for zone-wide functions, such as zone-wide primary storage, to work for VMware hosts, CloudStack has to make sure that a zone contains only a single VMware Datacenter. Therefore, when you are creating a new CloudStack zone, you will now be able to select a VMware Datacenter for the zone. If you are provisioning multiple VMware Datacenters, each one will be set up as a single zone in CloudStack.

Note

If you are upgrading from a previous CloudStack version, and your existing deployment contains a zone with clusters from multiple VMware Datacenters, that zone will not be forcibly migrated to the new model. It will continue to function as before. However, any new zone-wide operations, such as zone-wide primary storage, will not be available in that zone.

2.2. Third-Party UI Plugin Framework

Using the new third-party plugin framework, you can write and install extensions to CloudStack. The installed and enabled plugins will appear in the UI.

The basic procedure for adding a UI plugin is explained in the Developer Guide. In summary, the plugin developer creates the plugin code itself (in Javascript), a thumbnail image, the plugin listing, and a CSS file. The CloudStack administrator adds the folder containing the plugin code under the CloudStack PLUGINS folder and adds the plugin name to a configuration file (plugins.js).

The next time the user refreshes the UI in the browser, the plugin will appear under the Plugins button in the left navigation bar.

2.3. Networking Enhancements

The following new features provide additional networking functionality in CloudStack 4.2.

2.3.1. IPv6

CloudStack 4.2 introduces initial support for IPv6. This feature is provided as a technical preview only. Full support is planned for a future release.

2.3.2. Portable IPs

Portable IPs in CloudStack are elastic IPs that can be transferred across geographically separated zones. As an administrator, you can provision a pool of portable IPs at region level and are available for user consumption. The users can acquire portable IPs if admin has provisioned portable public IPs at the region level they are part of. These IPs can be used for any service within an advanced zone. You can also use portable IPs for EIP service in Basic zones. Additionally, a portable IP can be transferred from one network to another network.

2.3.3. N-Tier Applications

In CloudStack 3.0.6, a functionality was added to allow users to create a multi-tier application connected to a single instance of a Virtual Router that supports inter-VLAN routing. Such a multi-tier application is called a virtual private cloud (VPC). Users were also able to connect their multi-tier applications to a private Gateway or a Site-to-Site VPN tunnel and route certain traffic to those gateways. For CloudStack 4.2, additional features are implemented to enhance VPC applications.

2.3.3.1. Support for KVM

VPC is now supported on KVM hypervisors.

2.3.3.2. Load Balancing Support for VPC

In a VPC, you can configure two types of load balancing—external LB and internal LB. External LB is nothing but a LB rule

created to redirect the traffic received at a public IP of the VPC virtual router. The traffic is load balanced within a tier based on your configuration. Citrix NetScaler and VPC virtual router are supported for external LB. When you use internal LB service, traffic received at a tier is load balanced across different VMs within that tier. For example, traffic reached at Web tier is redirected to another VM in that tier. External load balancing devices are not supported for internal LB. The service is provided by a internal LB VM configured on the target tier.

2.3.3.2.1. Load Balancing Within a Tier (External LB)

A CloudStack user or administrator may create load balancing rules that balance traffic received at a public IP to one or more VMs that belong to a network tier that provides load balancing service in a VPC. A user creates a rule, specifies an algorithm, and assigns the rule to a set of VMs within a tier.

2.3.3.2.2. Load Balancing Across Tiers

CloudStack supports sharing workload across different tiers within your VPC. Assume that multiple tiers are set up in your environment, such as Web tier and Application tier. Traffic to each tier is balanced on the VPC virtual router on the public side. If you want the traffic coming from the Web tier to the Application tier to be balanced, use the internal load balancing feature offered by CloudStack.

2.3.3.2.3. Netscaler Support for VPC

Citrix NetScaler is supported for external LB. Certified version for this feature is NetScaler 10.0 Build 74.4006.e.

2.3.3.3. Enhanced Access Control List

Network Access Control List (ACL) on the VPC virtual router is enhanced. The network ACLs can be created for the tiers only if the NetworkACL service is supported. In CloudStack terminology, Network ACL is a group of Network ACL items. Network ACL items are nothing but numbered rules that are evaluated in order, starting with the lowest numbered rule. These rules determine whether traffic is allowed in or out of any tier associated with the network ACL. You need to add the Network ACL items to the Network ACL, then associate the Network ACL with a tier. Network ACL is associated with a VPC and can be assigned to multiple VPC tiers within a VPC. A Tier is associated with a Network ACL at all the times. Each tier can be associated with only one ACL.

The default Network ACL is used when no ACL is associated. Default behavior is all incoming traffic to guest networks is blocked and all outgoing traffic from guest networks is allowed. Default network ACL cannot be removed or modified.

2.3.3.3.1. ACL on Private Gateway

The traffic on the VPC private gateway is controlled by creating both ingress and egress network ACL rules. The ACLs contains both allow and deny rules. As per the rule, all the ingress traffic to the private gateway interface and all the egress traffic out from the private gateway interface are blocked. You can change this default behaviour while creating a private gateway.

2.3.3.3.2. Allow ACL on All Level 4 Protocols

In addition to the existing protocol support for ICMP, TCP, UDP, support for All Level 4 protocols is added. The protocol numbers from 0 to 255 are supported.

2.3.3.3.3. Support for ACL Deny Rules

In addition to the existing support for ACL Allow rules, support for ACL Deny rules has been added in CloudStack 4.2. As part of this, two operations are supported: Number and Action. You can configure a rule, allow or deny, by using action. Use Number to add a rule number.

2.3.3.4. Deploying VMs to a VPC Tier and Shared Networks

CloudStack allows you to deploy VMs on a VPC tier and one or more shared networks. With this feature, the VMs deployed in a multi-tier application can receive services offered by a service provider over the shared network. One example of such a service is monitoring service.

2.3.3.5. Adding a Private Gateway to a VPC

A private gateway can be added by the root admin only. The VPC private network has 1:1 relationship with the NIC of the physical network. You can configure multiple private gateways to a single VPC. No gateways with duplicated VLAN and IP are allowed in the same data center.

2.3.3.5.1. Source NAT on Private Gateway

You might want to deploy multiple VPCs with the same super CIDR and guest tier CIDR. Therefore, multiple guest VMs from different VPCs can have the same IPs to reach a enterprise data center through the private gateway. In such cases, a NAT service need to be configured on the private gateway. If Source NAT is enabled, the guest VMs in VPC reaches the enterprise network via private gateway IP address by using the NAT service.

The Source NAT service on a private gateway can be enabled while adding the private gateway. On deletion of a private gateway, source NAT rules specific to the private gateway are deleted.

2.3.3.5.2. VPN Gateways

Support up to 8 VPN Gateways is added.

2.3.3.5.3. Creating a Static Route

CloudStack enables you to specify routing for the VPN connection you create. You can enter one or CIDR addresses to indicate which traffic is to be routed back to the gateway.

2.3.3.5.4. Blacklisting Routes

CloudStack enables you to block a list of routes so that they are not assigned to any of the VPC private gateways. Specify the list of routes that you want to blacklist in the **`blacklisted.routes`** global parameter. Note that the parameter update affects only new static route creations. If you block an existing static route, it remains intact and continue functioning. You cannot add a static route if the route is blacklisted for the zone.

2.3.4. Assigning VLANs to Isolated Networks

CloudStack provides you the ability to control VLAN assignment to Isolated networks. You can assign a VLAN ID when a network is created, just the way it's done for Shared networks.

The former behaviour also is supported — VLAN is randomly allocated to a network from the VNET range of the physical network when the network turns to Implemented state. The VLAN is released back to the VNET pool when the network shuts down as a part of the Network Garbage Collection. The VLAN can be re-used either by the same network when it is implemented again, or by any other network. On each subsequent implementation of a network, a new VLAN can be assigned.

Note

You cannot change a VLAN once it's assigned to the network. The VLAN remains with the network for its entire life cycle.

2.3.5. Persistent Networks

CloudStack 4.2 supports Persistent Networks. The network that you can provision without having to deploy any VMs on it is called a Persistent Network. A Persistent Network can be part of a VPC or a non-VPC environment. With the addition of this feature, you will have the ability to create a network in CloudStack in which physical devices can be deployed without having to run any VMs. Additionally, you can deploy physical devices on that network. Another advantages is that you can create a VPC with a tier that consists only physical devices. For example, you might create a VPC for a three-tier application, deploy VMs for Web and Application tier, and use physical machines for the Database tier. Another use case is that if you are providing services by using physical hardware, you can define the network as persistent and therefore even if all its VMs are destroyed the services will not be discontinued.

2.3.6. Cisco VNMC Support

Cisco Virtual Network Management Center (VNMC) provides centralized multi-device and policy management for Cisco Network Virtual Services. When Cisco VNMC is integrated with ASA 1000v Cloud Firewall and Cisco Nexus 1000v dvSwitch in CloudStack you will be able to:

- Configure Cisco ASA 1000v Firewalls
- Create and apply security profiles that contain ACL policy sets for both ingress and egress traffic, and NAT policy sets

CloudStack supports Cisco VNMC on Cisco Nexus 1000v dvSwich-enabled VMware hypervisors.

2.3.7. VMware vNetwork Distributed vSwitch

CloudStack supports VMware vSphere Distributed Switch (VDS) for virtual network configuration in a VMware vSphere environment. Each vCenter server instance can support up to 128 VDSs and each VDS can manage up to 500 VMware hosts. CloudStack supports configuring virtual networks in a deployment with a mix of Virtual Distributed Switch, Standard Virtual Switch and Nexus 1000v Virtual Switch.

2.3.8. IP Reservation in Isolated Guest Networks

In Isolated guest networks in CloudStack 4.2, a part of the guest IP address space can be reserved for non-CloudStack VMs or physical servers. To do so, you configure a range of Reserved IP addresses by specifying the CIDR when a guest network is in Implemented state. The advantage of having this feature is that if your customers wish to have non-CloudStack controlled VMs or physical servers on the same network, they can use a part of the IP address space that is primarily provided to the guest network. When IP reservation is configured, the administrator can add additional VMs or physical servers that are not part of CloudStack to the same network and assign them the Reserved IP addresses. CloudStack guest VMs cannot acquire IPs from the Reserved IP Range.

2.3.9. Dedicated Resources: Public IP Addresses and VLANs Per Account

CloudStack provides you the ability to reserve a set of public IP addresses and VLANs exclusively for an account. During zone creation, you can continue to define a set of VLANs and multiple public IP ranges. This feature extends the functionality to enable you to dedicate a fixed set of VLANs and guest IP addresses for a tenant.

This feature provides you the following capabilities:

- Reserve a VLAN range and public IP address range from an Advanced zone and assign it to an account
- Disassociate a VLAN and public IP address range from an account

Note

Ensure that you check whether the required range is available and conforms to account limits. The maximum IPs per account limit cannot be superseded.

2.3.10. Enhanced Juniper SRX Support for Egress Firewall Rules

Egress firewall rules were previously supported on virtual routers, and now they are also supported on Juniper SRX

external networking devices.

Egress traffic originates from a private network to a public network, such as the Internet. By default, the egress traffic is blocked, so no outgoing traffic is allowed from a guest network to the Internet. However, you can control the egress traffic in an Advanced zone by creating egress firewall rules. When an egress firewall rule is applied, the traffic specific to the rule is allowed and the remaining traffic is blocked. When all the firewall rules are removed the default policy, Block, is applied.

Note

Egress firewall rules are not supported on Shared networks. They are supported only on Isolated guest networks.

2.3.11. Configuring the Default Egress Policy

The default egress policy for Isolated guest network can be configured by using Network offering. Use the create network offering option to determine whether the default policy should be block or allow all the traffic to the public network from a guest network. Use this network offering to create the network. If no policy is specified, by default all the traffic is allowed from the guest network that you create by using this network offering.

You have two options: Allow and Deny.

If you select Allow for a network offering, by default egress traffic is allowed. However, when an egress rule is configured for a guest network, rules are applied to block the specified traffic and rest are allowed. If no egress rules are configured for the network, egress traffic is accepted. If you select Deny for a network offering, by default egress traffic for the guest network is blocked. However, when an egress rules is configured for a guest network, rules are applied to allow the specified traffic. While implementing a guest network, CloudStack adds the firewall egress rule specific to the default egress policy for the guest network.

This feature is supported only on virtual router and Juniper SRX.

2.3.12. Non-Contiguous VLAN Ranges

CloudStack provides you with the flexibility to add non contiguous VLAN ranges to your network. The administrator can either update an existing VLAN range or add multiple non contiguous VLAN ranges while creating a zone. You can also use the UpdatephysicalNetwork API to extend the VLAN range.

2.3.13. Isolation in Advanced Zone Using Private VLAN

Isolation of guest traffic in shared networks can be achieved by using Private VLANs (PVLAN). PVLANs provide Layer 2 isolation between ports within the same VLAN. In a PVLAN-enabled shared network, a user VM cannot reach other user VM though they can reach the DHCP server and gateway, this would in turn allow users to control traffic within a network and help them deploy multiple applications without communication between application as well as prevent communication with other users' VMs.

- Isolate VMs in a shared networks by using Private VLANs.
- Supported on KVM, XenServer, and VMware hypervisors.
- PVLAN-enabled shared network can be a part of multiple networks of a guest VM.

For further reading:

- Understanding Private VLANs
- Cisco Systems' Private VLANs: Scalable Security in a Multi-Client Environment
- Private VLAN (PVLAN) on vNetwork Distributed Switch - Concept Overview (1010691)

2.3.14. Configuring Multiple IP Addresses on a Single NIC

(Supported on XenServer, KVM, and VMware hypervisors)

CloudStack now provides you the ability to associate multiple private IP addresses per guest VM NIC. This feature is supported on all the network configurations—Basic, Advanced, and VPC. Security Groups, Static NAT and Port forwarding services are supported on these additional IPs. In addition to the primary IP, you can assign additional IPs to the guest VM NIC. Up to 256 IP addresses are allowed per NIC.

As always, you can specify an IP from the guest subnet; if not specified, an IP is automatically picked up from the guest VM subnet. You can view the IPs associated with for each guest VM NICs on the UI. You can apply NAT on these additional guest IPs by using firewall configuration in the CloudStack UI. You must specify the NIC to which the IP should be associated.

2.3.15. Adding Multiple IP Ranges

(Supported on KVM, xenServer, and VMware hypervisors)

CloudStack 4.2 provides you with the flexibility to add guest IP ranges from different subnets in Basic zones and security groups-enabled Advanced zones. For security groups-enabled Advanced zones, it implies multiple subnets can be added to the same VLAN. With the addition of this feature, you will be able to add IP address ranges from the same subnet or from a different one when IP address are exhausted. This would in turn allows you to employ higher number of subnets and thus reduce the address management overhead.

Ensure that you manually configure the gateway of the new subnet before adding the IP range. Note that CloudStack supports only one gateway for a subnet; overlapping subnets are not currently supported.

You can also delete IP ranges. This operation fails if an IP from the remove range is in use. If the remove range contains

the IP address on which the DHCP server is running, CloudStack acquires a new IP from the same subnet. If no IP is available in the subnet, the remove operation fails.

Note

The feature can only be implemented on IPv4 addresses.

2.3.16. Support for Multiple Networks in VMs

(Supported on XenServer, VMware and KVM hypervisors)

CloudStack 4.2 provides you the ability to add and remove multiple networks to a VM. You can remove a network from a VM and add a new network. You can also change the default network of a VM. With this functionality, hybrid or traditional server loads can be accommodated with ease.

For adding or removing a NIC to work on VMware, ensure that vm-tools are running on guest VMs.

2.3.17. Global Server Load Balancing

CloudStack 4.2 supports Global Server Load Balancing (GSLB) functionalities to provide business continuity by load balancing traffic to an instance on active zones only in case of zone failures . CloudStack achieve this by extending its functionality of integrating with NetScaler Application Delivery Controller (ADC), which also provides various GSLB capabilities, such as disaster recovery and load balancing. The DNS redirection technique is used to achieve GSLB in CloudStack. In order to support this functionality, region level services and service provider are introduced. A new service 'GSLB' is introduced as a region level service. The GSLB service provider is introduced that will provider the GSLB service. Currently, NetScaler is the supported GSLB provider in CloudStack. GSLB functionality works in an Active-Active data center environment.

2.3.18. Enhanced Load Balancing Services Using External Provider on Shared VLANs

Network services like Firewall, Load Balancing, and NAT are now supported in shared networks created in an advanced zone. In effect, the following network services shall be made available to a VM in a shared network: Source NAT, Static NAT, Port Forwarding, Firewall and Load balancing. Subset of these service can be chosen while creating a network offering for shared networks. Services available in a shared network is defined by the network offering and the service chosen in the network offering. For example, if network offering for a shared network has source NAT service enabled, a public IP shall be provisioned and source NAT is configured on the firewall device to provide public access to the VMs on the shared network. Static NAT, Port Forwarding, Load Balancing, and Firewall services shall be available only on the acquired public IPs associated with a shared network.

Additionally, Netscaler and Juniper SRX firewall device can be configured inline or side-by-side mode.

2.3.19. Health Checks for Load Balanced Instances

Note

This feature is supported only on NetScaler version 10.0 and beyond.

(NetScaler load balancer only) A load balancer rule distributes requests among a pool of services (a service in this context means an application running on a virtual machine). When creating a load balancer rule, you can specify a health check which will ensure that the rule forwards requests only to services that are healthy (running and available). When a health check is in effect, the load balancer will stop forwarding requests to any resources that it has found to be unhealthy. If the resource later becomes available again, the periodic health check (periodicity is configurable) will discover it and the resource will once again be made available to the load balancer.

To configure how often the health check is performed by default, use the global configuration setting healthcheck.update.interval. This default applies to all the health check policies in the cloud. You can override this value for an individual health check policy.

2.4. Host and Virtual Machine Enhancements

The following new features expand the ways you can use hosts and virtual machines.

2.4.1. VMware DRS Support

The VMware vSphere Distributed Resources Scheduler (DRS) is supported.

2.4.2. Windows 8 and Windows Server 2012 as VM Guest OS

(Supported on XenServer, VMware, and KVM)

Windows 8 and Windows Server 2012 can now be used as OS types on guest virtual machines. The OS would be made available the same as any other, by uploading an ISO or a template. The instructions for uploading ISOs and templates are given in the Administrator's Guide.

Note

Limitation: When used with VMware hosts, this feature works only for the following versions: vSphere ESXi 5.1 and ESXi 5.0 Patch 4.

2.4.3. Change Account Ownership of Virtual Machines

A root administrator can now change the ownership of any virtual machine from one account to any other account. A domain or sub-domain administrator can do the same for VMs within the domain from one account to any other account in the domain.

2.4.4. Private Pod, Cluster, or Host

Dedicating pod, cluster or host to a specific domain/account means that the domain/account will have sole access to the dedicated pod, cluster or hosts such that scalability, security and manageability within a domain/account can be improved. The resources which belong to that tenant will be placed into that dedicated pod, cluster or host.

2.4.5. Resizing Volumes

CloudStack provides the ability to resize data disks; CloudStack controls volume size by using disk offerings. This provides CloudStack administrators with the flexibility to choose how much space they want to make available to the end users. Volumes within the disk offerings with the same storage tag can be resized. For example, if you only want to offer 10, 50, and 100 GB offerings, the allowed resize should stay within those limits. That implies if you define a 10 GB, a 50 GB and a 100 GB disk offerings, a user can upgrade from 10 GB to 50 GB, or 50 GB to 100 GB. If you create a custom-sized disk offering, then you have the option to resize the volume by specifying a new, larger size. Additionally, using the resizeVolume API, a data volume can be moved from a static disk offering to a custom disk offering with the size specified. This functionality allows those who might be billing by certain volume sizes or disk offerings to stick to that model, while providing the flexibility to migrate to whatever custom size necessary. This feature is supported on KVM, XenServer, and VMware hosts. However, shrinking volumes is not supported on VMware hosts

2.4.6. VMware Volume Snapshot Improved Performance

When you take a snapshot of a data volume on VMware, CloudStack will now use a more efficient storage technique to improve performance.

Previously, every snapshot was immediately exported from vCenter to a mounted NFS share and packaged into an OVA file format. This operation consumed time and resources. Starting from 4.2, the original file formats (e.g., VMDK) provided by vCenter will be retained. An OVA file will only be created as needed, on demand.

The new process applies only to newly created snapshots after upgrade to CloudStack 4.2. Snapshots that have already been taken and stored in OVA format will continue to exist in that format, and will continue to work as expected.

2.4.7. Storage Migration: XenMotion and vMotion

(Supported on XenServer and VMware)

Storage migration allows VMs to be moved from one host to another, where the VMs are not located on storage shared between the two hosts. It provides the option to live migrate a VM's disks along with the VM itself. It is now possible to migrate a VM from one XenServer resource pool / VMware cluster to another, or to migrate a VM whose disks are on local storage, or even to migrate a VM's disks from one storage repository to another, all while the VM is running.

2.4.8. Configuring Usage of Linked Clones on VMware

(For ESX hypervisor in conjunction with vCenter)

In CloudStack 4.2, the creation of VMs as full clones is allowed. In previous versions, only linked clones were possible.

For a full description of clone types, refer to VMware documentation. In summary: A full clone is a copy of an existing virtual machine which, once created, does not depend in any way on the original virtual machine. A linked clone is also a copy of an existing virtual machine, but it has ongoing dependency on the original. A linked clone shares the virtual disk of the original VM, and retains access to all files that were present at the time the clone was created.

A new global configuration setting has been added, vmware.create.full.clone. When the administrator sets this to true, end users can create guest VMs only as full clones. The default value is true for new installations. For customers upgrading from a previous version of CloudStack, the default value of vmware.create.full.clone is false.

2.4.9. VM Deployment Rules

Rules can be set up to ensure that particular VMs are not placed on the same physical host. These "anti-affinity rules" can increase the reliability of applications by ensuring that the failure of a single host can not take down the entire group of VMs supporting a given application. See Affinity Groups in the CloudStack 4.2 Administration Guide.

2.4.10. CPU and Memory Scaling for Running VMs

(Supported on VMware and XenServer)

You can now change the CPU and RAM values for a running virtual machine. In previous versions of CloudStack, this could only be done on a stopped VM.

It is not always possible to accurately predict the CPU and RAM requirements when you first deploy a VM. You might need to increase or decrease these resources at any time during the life of a VM. With the new ability to dynamically modify CPU and RAM levels, you can change these resources for a running VM without incurring any downtime.

Dynamic CPU and RAM scaling can be used in the following cases:

- New VMs that are created after the installation of CloudStack 4.2. If you are upgrading from a previous version of CloudStack, your existing VMs created with previous versions will not have the dynamic scaling capability.
- User VMs on hosts running VMware and XenServer.

- System VMs on VMware.
- VM Tools or XenServer Tools must be installed on the virtual machine.
- The new requested CPU and RAM values must be within the constraints allowed by the hypervisor and the VM operating system.

To configure this feature, use the following new global configuration variables:

- enable.dynamic.scale.vm: Set to True to enable the feature. By default, the feature is turned off.
- scale.retry: How many times to attempt the scaling operation. Default = 2.

2.4.11. CPU and Memory Over-Provisioning

(Supported for XenServer, KVM, and VMware)

In CloudStack 4.2, CPU and memory (RAM) over-provisioning factors can be set for each cluster to change the number of VMs that can run on each host in the cluster. This helps optimize the use of resources. By increasing the over-provisioning ratio, more resource capacity will be used. If the ratio is set to 1, no over-provisioning is done.

In previous releases, CloudStack did not perform memory over-provisioning. It performed CPU over-provisioning based on a ratio configured by the administrator in the global configuration setting cpu.overprovisioning.factor. Starting in 4.2, the administrator can specify a memory over-provisioning ratio, and can specify both CPU and memory over-provisioning ratios on a per-cluster basis, rather than only on a global basis.

In any given cloud, the optimum number of VMs for each host is affected by such things as the hypervisor, storage, and hardware configuration. These may be different for each cluster in the same cloud. A single global over-provisioning setting could not provide the best utilization for all the different clusters in the cloud. It had to be set for the lowest common denominator. The new per-cluster setting provides a finer granularity for better utilization of resources, no matter where the CloudStack placement algorithm decides to place a VM.

2.4.12. Kickstart Installation for Bare Metal Provisioning

CloudStack 4.2 supports the kick start installation method for RPM-based Linux operating systems on baremetal hosts in basic zones. Users can provision a baremetal host managed by CloudStack as long as they have the kick start file and corresponding OS installation ISO ready.

Tested on CentOS 5.5, CentOS 6.2, CentOS 6.3, Ubuntu 12.04.

For more information, see the Baremetal Installation Guide.

2.4.13. Enhanced Bare Metal Support on Cisco UCS

You can now more easily provision new Cisco UCS server blades into CloudStack for use as bare metal hosts. The goal is to enable easy expansion of the cloud by leveraging the programmability of the UCS converged infrastructure and CloudStack's knowledge of the cloud architecture and ability to orchestrate. With this new feature, CloudStack can automatically understand the UCS environment, server profiles, etc. to make it easy to deploy a bare metal OS on a Cisco UCS.

2.4.14. Changing a VM's Base Image

Every VM is created from a base image, which is a template or ISO which has been created and stored in CloudStack. Both cloud administrators and end users can create and modify templates, ISOs, and VMs.

In CloudStack 4.2, there is a new way to modify an existing VM. You can change an existing VM from one base image to another. For example, suppose there is a template based on a particular operating system, and the OS vendor releases a software patch. The administrator or user naturally wants to apply the patch and then make sure existing VMs start using it. Whether a software update is involved or not, it's also possible to simply switch a VM from its current template to any other desired template.

2.4.15. Reset VM on Reboot

In CloudStack 4.2, you can specify that you want to discard the root disk and create a new one whenever a given VM is rebooted. This is useful for secure environments that need a fresh start on every boot and for desktops that should not retain state. The IP address of the VM will not change due to this operation.

2.4.16. Virtual Machine Snapshots for VMware

(VMware hosts only) In addition to the existing CloudStack ability to snapshot individual VM volumes, you can now take a VM snapshot to preserve all the VM's data volumes as well as (optionally) its CPU/memory state. This is useful for quick restore of a VM. For example, you can snapshot a VM, then make changes such as software upgrades. If anything goes wrong, simply restore the VM to its previous state using the previously saved VM snapshot.

The snapshot is created using the VMware native snapshot facility. The VM snapshot includes not only the data volumes, but optionally also whether the VM is running or turned off (CPU state) and the memory contents. The snapshot is stored in CloudStack's primary storage.

VM snapshots can have a parent/child relationship. Each successive snapshot of the same VM is the child of the snapshot that came before it. Each time you take an additional snapshot of the same VM, it saves only the differences between the current state of the VM and the state stored in the most recent previous snapshot. The previous snapshot becomes a parent, and the new snapshot is its child. It is possible to create a long chain of these parent/child snapshots, which amount to a "redo" record leading from the current state of the VM back to the original.

2.4.17. Increased Userdata Size When Deploying a VM

You can now specify up to 32KB of userdata when deploying a virtual machine through the CloudStack UI or the deployVirtualMachine API call.

deployVirtualMachine API call.

2.4.18. Set VMware Cluster Size Limit Depending on VMware Version

The maximum number of hosts in a vSphere cluster is determined by the VMware hypervisor software. For VMware versions 4.2, 4.1, 5.0, and 5.1, the limit is 32 hosts.

For CloudStack 4.2, the global configuration setting vmware.percluster.host.max has been removed. The maximum number of hosts in a VMware cluster is now determined by the underlying hypervisor software.

Note

Best Practice: It is advisable for VMware clusters in CloudStack to be smaller than the VMware hypervisor's maximum size. A cluster size of up to 8 hosts has been found optimal for most real-world situations.

2.4.19. Limiting Resource Usage

Previously in CloudStack, resource usage limit was imposed based on the resource count, that is, restrict a user or domain on the basis of the number of VMs, volumes, or snapshots used. In CloudStack 4.2, a new set of resource types has been added to the existing pool of resources (VMs, Volumes, and Snapshots) to support the customization model—need-basis usage, such as large VM or small VM. The new resource types are now broadly classified as CPU, RAM, Primary storage, and Secondary storage. CloudStack 4.2 allows the root administrator to impose resource usage limit by the following resource types for Domain, Project and Accounts.

- CPUs
- Memory (RAM)
- Primary Storage (Volumes)
- Secondary Storage (Snapshots, Templates, ISOs)

2.5. Monitoring, Maintenance, and Operations Enhancements

2.5.1. Deleting and Archiving Events and Alerts

In addition to viewing a list of events and alerts in the UI, the administrator can now delete and archive them. In order to support deleting and archiving alerts, the following global parameters have been added:

- **alert.purge.delay**: The alerts older than specified number of days are purged. Set the value to 0 to never purge alerts automatically.
- **alert.purge.interval**: The interval in seconds to wait before running the alert purge thread. The default is 86400 seconds (one day).

Note

Archived alerts or events cannot be viewed in the UI, or by using the API. They are maintained in the database for auditing or compliance purposes.

2.5.2. Increased Granularity for Configuration Parameters

Some configuration parameters which were previously available only at the global level of the cloud can now be set for smaller components of the cloud, such as at the zone level. To set these parameters, look for the new Settings tab in the UI. You will find it on the detail page for an account, cluster, zone, or primary storage.

The account level parameters are: **`remote.access.vpn.client.iprange`**, **`allow.public.user.templates`**, **`use.system.public.ips`**, and **`use.system.guest.vlans`**

The cluster level parameters are **`cluster.storage.allocated.capacity.notificationthreshold`**, **`cluster.storage.capacity.notificationthreshold`**, **`cluster.cpu.allocated.capacity.notificationthreshold`**, **`cluster.memory.allocated.capacity.notificationthreshold`**, **`cluster.cpu.allocated.capacity.disablethreshold`**, **`cluster.memory.allocated.capacity.disablethreshold`**, **`cpu.overprovisioning.factor`**, **`mem.overprovisioning.factor`**, **`vmware.reserve.cpu`**, and **`vmware.reserve.mem`**.

The zone level parameters are **`pool.storage.allocated.capacity.disablethreshold`**, **`pool.storage.capacity.disablethreshold`**, **`storage.overprovisioning.factor`**, **`network.throttling.rate`**, **`guest.domain.suffix`**, **`router.template.xen`**, **`router.template.kvm`**, **`router.template.vmware`**, **`router.template.hyperv`**, **`router.template.lx`**c, **`enable.dynamic.scale.vm`**, **`use.external.dns`**, and **`blacklisted.routes`**.

2.5.3. API Request Throttling

In CloudStack 4.2, you can limit the rate at which API requests can be placed for each account. This is useful to avoid malicious attacks on the Management Server, prevent performance degradation, and provide fairness to all accounts.

If the number of API calls exceeds the threshold, an error message is returned for any additional API calls. The caller will have to retry these API calls at another time.

To control the API request throttling, use the following new global configuration settings:

- api.throttling.enabled - Enable/Disable API throttling. By default, this setting is false, so API throttling is not enabled.
- api.throttling.interval (in seconds) - Time interval during which the number of API requests is to be counted. When the interval has passed, the API count is reset to 0.
- api.throttling.max - Maximum number of APIs that can be placed within the api.throttling.interval period.
- api.throttling.cachesize - Cache size for storing API counters. Use a value higher than the total number of accounts managed by the cloud. One cache entry is needed for each account, to store the running API total for that account within the current time window.

2.5.4. Sending Alerts to External SNMP and Syslog Managers

In addition to showing administrator alerts on the Dashboard in the CloudStack UI and sending them in email, CloudStack now can also send the same alerts to external SNMP or Syslog management software. This is useful if you prefer to use an SNMP or Syslog manager to monitor your cloud.

The supported protocol is SNMP version 2.

2.5.5. Changing the Default Password Encryption

Passwords are encoded when creating or updating users. The new default preferred encoder, replacing MD5, is SHA256. It is more secure than MD5 hashing. If you take no action to customize password encryption and authentication, SHA256 Salt will be used.

If you prefer a different authentication mechanism, CloudStack 4.2 provides a way for you to determine the default encoding and authentication mechanism for admin and user logins. Two new configurable lists have been introduced: userPasswordEncoders and userAuthenticators. userPasswordEncoders allow you to configure the order of preference for encoding passwords, and userAuthenticator allows you to configure the order in which authentication schemes are invoked to validate user passwords.

The plain text user authenticator has been modified not to convert supplied passwords to their md5 sums before checking them with the database entries. It performs a simple string comparison between retrieved and supplied login passwords instead of comparing the retrieved md5 hash of the stored password against the supplied md5 hash of the password, because clients no longer hash the password.

2.5.6. Log Collection Utility cloud-bugtool

CloudStack provides a command-line utility called cloud-bugtool to make it easier to collect the logs and other diagnostic data required for troubleshooting. This is especially useful when interacting with Citrix Technical Support.

You can use cloud-bugtool to collect the following:

- Basic system and environment information and network configuration including IP addresses, routing, and name resolver settings
- Information about running processes
- Management Server logs
- System logs in /var/log/
- Dump of the cloud database

Warning

cloud-bugtool collects information which might be considered sensitive and confidential. Using the **--nodb** option to avoid the cloud database can reduce this concern, though it is not guaranteed to exclude all sensitive data.

2.5.7. Snaphotting, Backups, Cloning and System VMs for RBD Primary Storage

Note

These new RBD features require at least librbd 0.61.7 (Cuttlefish) and libvirt 0.9.14 on the KVM hypervisors.

This release of CloudStack will leverage the features of RBD format 2. This allows snapshotting and backing up those snapshots.

Backups of snapshots to Secondary Storage are full copies of the RBD snapshot, they are not RBD diffs. This because when restoring a backup of a snapshot it is not mandatory that this backup is deployed on RBD again, it could also be a NFS Primary Storage.

Another key feature of RBD format 2 is cloning. With this release templates will be copied to Primary Storage once and by using the cloning mechanism new disks will be cloned from this parent template. This saves space and decreases deployment time for instances dramatically.

Before this release, a NFS Primary Storage was still required for running the System VMs from. The reason was a so called 'patch disk' that was generated by the hypervisor which contained metadata for the System VM. The scripts generating this disk didn't support RBD and thus System VMs had to be deployed from NFS. With 4.2 instead of the patch disk a VirtIO serial console is used to pass meta information to System VMs. This enabled the deployment of System VMs on RBD Primary Storage.

2.6. Issues Fixed in 4.2.0

Apache CloudStack uses Jira to track its issues. All new features and bugs for 4.2.0 have been tracked in Jira, and have a

Apache CloudStack uses Jira to track its issues. All new features and bugs for 4.2.0 have been tracked in Jira, and have a standard naming convention of "CLOUDSTACK-NNNN" where "NNNN" is the issue number.

For the list of issues fixed, see Issues Fixed in 4.2.

2.7. Known Issues in 4.2.0

This section includes a summary of known issues that were fixed in 4.2.0. For the list of known issues, see Known Issues.

Chapter 3. Upgrade Instructions for 4.2

This section contains upgrade instructions from prior versions of CloudStack to Apache CloudStack 4.2.0. We include instructions on upgrading to Apache CloudStack from pre-Apache versions of Citrix CloudStack (last version prior to Apache is 3.0.2) and from the releases made while CloudStack was in the Apache Incubator.

If you run into any issues during upgrades, please feel free to ask questions on users@cloudstack.apache.org or dev@cloudstack.apache.org.

Overprovisioning cautions

If the CloudStack instance you are upgrading is leveraging overprovisioning you need to read and understand Section 2.4.11, “CPU and Memory Over-Provisioning”. The overprovisioning factors are now cluster specific and should you be overprovisioned, your new system VMs may not start up.

3.1. Upgrade from 4.1.x to 4.2.0

This section will guide you from CloudStack 4.1.x versions to CloudStack 4.2.0.

Any steps that are hypervisor-specific will be called out with a note.

We recommend reading through this section once or twice before beginning your upgrade procedure, and working through it on a test system before working on a production system.

1. Most users of CloudStack manage the installation and upgrades of CloudStack with one of Linux's predominant package systems, RPM or APT. This guide assumes you'll be using RPM and Yum (for Red Hat Enterprise Linux or CentOS), or APT and Debian packages (for Ubuntu).
2. Create RPM or Debian packages (as appropriate) and a repository from the 4.2.0 source, or check the Apache CloudStack downloads page at http://cloudstack.apache.org/downloads.html for package repositories supplied by community members. You will need them for step 7 or step 10.

   Instructions for creating packages from the CloudStack source are in the Installation Guide.
3. Stop your management server or servers. Run this on all management server hosts:

   ```
   # service cloudstack-management stop
   ```
4. If you are running a usage server or usage servers, stop those as well:

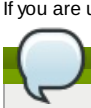
   ```
   # service cloudstack-usage stop
   ```
5. Make a backup of your MySQL database. If you run into any issues or need to roll back the upgrade, this will assist in debugging or restoring your existing environment. You'll be prompted for your password.

   ```
   # mysqldump -u root -p cloud > cloudstack-backup.sql
   ```
6. (KVM Only) If primary storage of type local storage is in use, the path for this storage needs to be verified to ensure it passes new validation. Check local storage by querying the cloud.storage_pool table:

   ```
   # mysql -u cloud -p -e "select id,name,path from cloud.storage_pool where
   pool_type='Filesystem'"
   ```

   If local storage paths are found to have a trailing forward slash, remove it:

   ```
   # mysql -u cloud -p -e 'update cloud.storage_pool set path="/var/lib/libvirt/images"
   where path="/var/lib/libvirt/images/"';
   ```
7. If you are using Ubuntu, follow this procedure to upgrade your packages. If not, skip to step 10.

   Community Packages

This section assumes you're using the community supplied packages for CloudStack. If you've created your own packages and APT repository, substitute your own URL for the ones used in these examples.

a. The first order of business will be to change the sources list for each system with CloudStack packages. This means all management servers, and any hosts that have the KVM agent. (No changes should be necessary for hosts that are running VMware or Xen.)

   Start by opening **`/etc/apt/sources.list.d/cloudstack.list`** on any systems that have CloudStack packages installed.

   This file should have one line, which contains:

   ```
   deb http://cloudstack.apt-get.eu/ubuntu precise 4.0
   ```

   We'll change it to point to the new package repository:

   ```
   deb http://cloudstack.apt-get.eu/ubuntu precise 4.2
   ```

   If you're using your own package repository, change this line to read as appropriate for your 4.2.0 repository.

b. Now update your apt package list:

   ```
   $ sudo apt-get update
   ```

c. Now that you have the repository configured, it's time to install the **`cloudstack-management`** package. This will pull in any other dependencies you need.

   ```
   $ sudo apt-get install cloudstack-management
   ```

d. You will need to manually install the **`cloudstack-agent`** package:

   ```
   $ sudo apt-get install cloudstack-agent
   ```

   During the installation of **`cloudstack-agent`**, APT will copy your **`agent.properties`**, **`log4j-cloud.xml`**, and **`environment.properties`** from **`/etc/cloud/agent`** to **`/etc/cloudstack/agent`**.

   When prompted whether you wish to keep your configuration, say Yes.

e. Verify that the file **`/etc/cloudstack/agent/environment.properties`** has a line that reads:

   ```
   paths.script=/usr/share/cloudstack-common
   ```

   If not, add the line.

f. Restart the agent:

   ```
   service cloudstack-agent stop
   killall jsvc
   service cloudstack-agent start
   ```

8. (VMware only) Additional steps are required for each VMware cluster. These steps will not affect running guests in the cloud. These steps are required only for clouds using VMware clusters:

   a. Stop the Management Server:

      ```
      service cloudstack-management stop
      ```

   b. Generate the encrypted equivalent of your vCenter password:

      ```
      java -classpath /usr/share/cloudstack-common/lib/jasypt-1.9.0.jar
      org.jasypt.intf.cli.JasyptPBEStringEncryptionCLI encrypt.sh
      input="_your_vCenter_password_" password="`cat /etc/cloudstack/management/key`"
      verbose=false
      ```

      Store the output from this step, we need to add this in cluster_details table and vmware_data_center tables in place of the plain text password

   c. Find the ID of the row of cluster_details table that you have to update:

      ```
      mysql -u <username> -p<password>
      ```

      ```
      select * from cloud.cluster_details;
      ```

   d. Update the plain text password with the encrypted one

      ```
      update cloud.cluster_details set value = '_ciphertext_from_step_1_' where id =
      _id_from_step_2_;
      ```

   e. Confirm that the table is updated:

      ```
      select * from cloud.cluster_details;
      ```

   f. Find the ID of the correct row of vmware_data_center that you want to update

      ```
      select * from cloud.vmware_data_center;
      ```

   g. update the plain text password with the encrypted one:

      ```
      update cloud.vmware_data_center set password = '_ciphertext_from_step_1_' where
      id = _id_from_step_5_;
      ```

   h. Confirm that the table is updated:

      ```
      select * from cloud.vmware_data_center;
      ```

   i. Start the CloudStack Management

i. Start the CloudStack Management server

```
service cloudstack-management start
```

9. (KVM only) Additional steps are required for each KVM host. These steps will not affect running guests in the cloud. These steps are required only for clouds using KVM as hosts and only on the KVM hosts.
   a. Configure the CloudStack yum repository as detailed above.
   b. Stop the running agent.

```
# service cloud-agent stop
```

   c. Update the agent software.

```
# yum update cloudstack-agent
```

   d. Start the agent.

```
# service cloudstack-agent start
```

10. If you are using CentOS or RHEL, follow this procedure to upgrade your packages. If not, skip to step 12.

Community Packages

This section assumes you're using the community supplied packages for CloudStack. If you've created your own packages and yum repository, substitute your own URL for the ones used in these examples.

   a. The first order of business will be to change the yum repository for each system with CloudStack packages. This means all management servers, and any hosts that have the KVM agent.
   (No changes should be necessary for hosts that are running VMware or Xen.)
   Start by opening **`/etc/yum.repos.d/cloudstack.repo`** on any systems that have CloudStack packages installed.
   This file should have content similar to the following:

```
[apache-cloudstack]
name=Apache CloudStack
baseurl=http://cloudstack.apt-get.eu/rhel/4.0/
enabled=1
gpgcheck=0
```

   If you are using the community provided package repository, change the base url to http://cloudstack.apt-get.eu/rhel/4.2/
   If you're using your own package repository, change this line to read as appropriate for your 4.2.0 repository.
   b. Now that you have the repository configured, it's time to install the **`cloudstack-management`** package by upgrading the older **`cloudstack-management`** package.

```
$ sudo yum upgrade cloudstack-management
```

   c. For KVM hosts, you will need to upgrade the **`cloud-agent`** package, similarly installing the new version as **`cloudstack-agent`**.

```
$ sudo yum upgrade cloudstack-agent
```

   d. Verify that the file **`/etc/cloudstack/agent/environment.properties`** has a line that reads:

```
paths.script=/usr/share/cloudstack-common
```

   If not, add the line.
   e. Restart the agent:

```
service cloudstack-agent stop
killall jsvc
service cloudstack-agent start
```

11. Now it's time to restart the management server

```
# service cloudstack-management start
```

12. Once you've upgraded the packages on your management servers, you'll need to restart the system VMs. Ensure that the admin port is set to 8096 by using the "integration.api.port" global parameter. This port is used by the cloud-sysvmadm script at the end of the upgrade procedure. For information about how to set this parameter, see "Setting Global Configuration Parameters" in the Installation Guide. Changing this parameter will require management server restart. Also make sure port 8096 is open in your local host firewall to do this.
    There is a script that will do this for you, all you need to do is run the script and supply the IP address for your MySQL instance and your MySQL credentials:

```
# nohup cloudstack-sysvmadm -d IP address -u cloud -p -a > sysvm.log 2>&1 &
```

    You can monitor the log for progress. The process of restarting the system VMs can take an hour or more.

```
# tail -f sysvm.log
```

    The output to **`sysvm.log`** will look something like this:

```
Stopping and starting 1 secondary storage vm(s)...
Done stopping and starting secondary storage vm(s)
Stopping and starting 1 console proxy vm(s)
```

```
Done stopping and starting console proxy vm(s).
Stopping and starting 4 running routing vm(s)...
Done restarting router(s).
```

13. **For Xen Hosts: Copy vhd-utils**

    This step is only for CloudStack installs that are using Xen hosts.

    Copy the file **vhd-utils** to **/usr/share/cloudstack-common/scripts/vm/hypervisor/xenserver**.

3.2. Upgrade from 3.0.x to 4.2.0

This section will guide you from Citrix CloudStack 3.0.x to Apache CloudStack 4.2.0. Sections that are hypervisor-specific will be called out with a note.

1. **Note**

   The following upgrade instructions apply only if you're using VMware hosts. If you're not using VMware hosts, skip this step and move on to 3.

   In each zone that includes VMware hosts, you need to add a new system VM template.

   a. While running the existing 3.0.x system, log in to the UI as root administrator.
   b. In the left navigation bar, click Templates.
   c. In Select view, click Templates.
   d. Click Register template.
      The Register template dialog box is displayed.
   e. In the Register template dialog box, specify the following values (do not change these):

Hypervisor	Description
XenServer	Name: systemvm-xenserver-4.2 Description: systemvm-xenserver-4.2 URL:http://download.cloud.com/templates/4.2/systemvmtemplate-2013-07-12-master-xen.vhd.bz2 Zone: Choose the zone where this hypervisor is used Hypervisor: XenServer Format: VHD OS Type: Debian GNU/Linux 7.0 (32-bit) (or the highest Debian release number available in the dropdown) Extractable: no Password Enabled: no Public: no Featured: no
KVM	Name: systemvm-kvm-4.2 Description: systemvm-kvm-4.2 URL: http://download.cloud.com/templates/4.2/systemvmtemplate-2013-06-12-master-kvm.qcow2.bz2 Zone: Choose the zone where this hypervisor is used Hypervisor: KVM Format: QCOW2 OS Type: Debian GNU/Linux 7.0 (32-bit) (or the highest Debian release number available in the dropdown) Extractable: no Password Enabled: no Public: no Featured: no
VMware	Name: systemvm-vmware-4.2 Description: systemvm-vmware-4.2 URL: http://download.cloud.com/templates/4.2/systemvmtemplate-4.2-vh7.ova Zone: Choose the zone where this hypervisor is used Hypervisor: VMware Format: OVA OS Type: Debian GNU/Linux 7.0 (32-bit) (or the highest Debian release number available in the dropdown) Extractable: no Password Enabled: no

	Public: no Featured: no

f. Watch the screen to be sure that the template downloads successfully and enters the READY state. Do not proceed until this is successful.

2. (KVM on RHEL 6.0/6.1 only) If your existing CloudStack deployment includes one or more clusters of KVM hosts running RHEL 6.0 or RHEL 6.1, perform the following:
   a. Ensure that you upgrade the operating system version on those hosts before upgrading CloudStack
      To do that, change the yum repository for each system with CloudStack packages, that implies that all the Management Servers and any hosts that have the KVM agent.
   b. Open **`/etc/yum.repos.d/cloudstack.repo`** on any systems that have CloudStack packages installed.
   c. Edit as follows:

```
[upgrade]
name=rhel63
baseurl=url-of-your-rhel6.3-repo
enabled=1
gpgcheck=0
[apache CloudStack]
name= Apache CloudStack
baseurl= http://cloudstack.apt-get.eu/rhel/4.0/
enabled=1
gpgcheck=0
```

      If you are using the community provided package repository, change the baseurl to http:// cloudstack.apt-get.eu/rhel/4.2/
      If you are using your own package repository, change this line to read as appropriate for your 4.2.0 repository.
   d. Now that you have the repository configured, upgrade the host operating system from RHEL 6.0 to 6.3:

```
# yum upgrade
```

3. Stop all Usage Servers if running. Run this on all Usage Server hosts.

```
# service cloud-usage stop
```

4. Stop the Management Servers. Run this on all Management Server hosts.

```
# service cloud-management stop
```

5. On the MySQL master, take a backup of the MySQL databases. We recommend performing this step even in test upgrades. If there is an issue, this will assist with debugging.
   In the following commands, it is assumed that you have set the root password on the database, which is a CloudStack recommended best practice. Substitute your own MySQL root password.

```
# mysqldump -u root -pmysql_password cloud > cloud-backup.dmp
# mysqldump -u root -pmysql_password cloud_usage > cloud-usage-backup.dmp
```

6. Either build RPM/DEB packages as detailed in the Installation Guide, or use one of the community provided yum/apt repositories to gain access to the CloudStack binaries.
7. If you are using Ubuntu, follow this procedure to upgrade your packages. If not, skip to step 8.

Community Packages

This section assumes you're using the community supplied packages for CloudStack. If you've created your own packages and APT repository, substitute your own URL for the ones used in these examples.

   a. The first order of business will be to change the sources list for each system with CloudStack packages. This means all management servers, and any hosts that have the KVM agent. (No changes should be necessary for hosts that are running VMware or Xen.)
      Start by opening **`/etc/apt/sources.list.d/cloudstack.list`** on any systems that have CloudStack packages installed.
      This file should have one line, which contains:

```
deb http://cloudstack.apt-get.eu/ubuntu precise 4.0
```

      We'll change it to point to the new package repository:

```
deb http://cloudstack.apt-get.eu/ubuntu precise 4.2
```

      If you're using your own package repository, change this line to read as appropriate for your 4.2.0 repository.
   b. Now update your apt package list:

```
$ sudo apt-get update
```

   c. Now that you have the repository configured, it's time to install the **`cloudstack-management`** package. This will pull in any other dependencies you need.

```
$ sudo apt-get install cloudstack-management
```

   d. You will need to manually install the **`cloudstack-agent`** package:

```
$ sudo apt-get install cloudstack-agent
```

```
$ sudo apt-get install cloudstack-agent
```

During the installation of **cloudstack-agent**, APT will copy your **agent.properties**, **log4j-cloud.xml**, and **environment.properties** from **/etc/cloud/agent** to **/etc/cloudstack/agent**.

When prompted whether you wish to keep your configuration, say Yes.

e. Verify that the file **/etc/cloudstack/agent/environment.properties** has a line that reads:

```
paths.script=/usr/share/cloudstack-common
```

If not, add the line.

f. Restart the agent:

```
service cloud-agent stop
killall jsvc
service cloudstack-agent start
```

g. During the upgrade, **log4j-cloud.xml** was simply copied over, so the logs will continue to be added to **/var/log/cloud/agent/agent.log**. There's nothing *wrong* with this, but if you prefer to be consistent, you can change this by copying over the sample configuration file:

```
cd /etc/cloudstack/agent
mv log4j-cloud.xml.dpkg-dist log4j-cloud.xml
service cloudstack-agent restart
```

h. Once the agent is running, you can uninstall the old cloud-* packages from your system:

```
sudo dpkg --purge cloud-agent
```

8. If you are using CentOS or RHEL, follow this procedure to upgrade your packages. If not, skip to step 9.

Community Packages

This section assumes you're using the community supplied packages for CloudStack. If you've created your own packages and yum repository, substitute your own URL for the ones used in these examples.

a. The first order of business will be to change the yum repository for each system with CloudStack packages. This means all management servers, and any hosts that have the KVM agent. (No changes should be necessary for hosts that are running VMware or Xen.)

Start by opening **/etc/yum.repos.d/cloudstack.repo** on any systems that have CloudStack packages installed.

This file should have content similar to the following:

```
[apache-cloudstack]
name=Apache CloudStack
baseurl=http://cloudstack.apt-get.eu/rhel/4.0/
enabled=1
gpgcheck=0
```

If you are using the community provided package repository, change the baseurl to http://cloudstack.apt-get.eu/rhel/4.2/

If you're using your own package repository, change this line to read as appropriate for your 4.2.0 repository.

b. Now that you have the repository configured, it's time to install the **cloudstack-management** package by upgrading the older **cloud-client** package.

```
$ sudo yum upgrade cloud-client
```

c. For KVM hosts, you will need to upgrade the **cloud-agent** package, similarly installing the new version as **cloudstack-agent**.

```
$ sudo yum upgrade cloud-agent
```

During the installation of **cloudstack-agent**, the RPM will copy your **agent.properties**, **log4j-cloud.xml**, and **environment.properties** from **/etc/cloud/agent** to **/etc/cloudstack/agent**.

d. Verify that the file **/etc/cloudstack/agent/environment.properties** has a line that reads:

```
paths.script=/usr/share/cloudstack-common
```

If not, add the line.

e. Restart the agent:

```
service cloud-agent stop
killall jsvc
service cloudstack-agent start
```

9. If you have made changes to your copy of **/etc/cloud/management/components.xml** the changes will be preserved in the upgrade. However, you need to do the following steps to place these changes in a new version of the file which is compatible with version 4.2.0.

a. Make a backup copy of **/etc/cloud/management/components.xml**. For example:

```
# mv /etc/cloud/management/components.xml /etc/cloud/management/components.xml-backup
```

b. Copy **/etc/cloud/management/components.xml.rpmnew** to create a new **/etc/cloud/management/components.xml**:

```
# cp -ap /etc/cloud/management/components.xml.rpmnew
/etc/cloud/management/components.xml
```

c. Merge your changes from the backup file into the new **components.xml**.

```
# vi /etc/cloudstack/management/components.xml
```

Note

If you have more than one management server node, repeat the upgrade steps on each node.

10. After upgrading to 4.2, API clients are expected to send plain text passwords for login and user creation, instead of MD5 hash. Incase, api client changes are not acceptable, following changes are to be made for backward compatibility:

    Modify componentsContext.xml, and make PlainTextUserAuthenticator as the default authenticator (1st entry in the userAuthenticators adapter list is default)

```
<!-- Security adapters -->
<bean id="userAuthenticators" class="com.cloud.utils.component.AdapterList">
  <property name="Adapters">
    <list>
      <ref bean="PlainTextUserAuthenticator"/>
      <ref bean="MD5UserAuthenticator"/>
      <ref bean="LDAPUserAuthenticator"/>
    </list>
  </property>
</bean>
```

    PlainTextUserAuthenticator works the same way MD5UserAuthenticator worked prior to 4.2.

11. Start the first Management Server. Do not start any other Management Server nodes yet.

```
# service cloudstack-management start
```

    Wait until the databases are upgraded. Ensure that the database upgrade is complete. After confirmation, start the other Management Servers one at a time by running the same command on each node.

Note

Failing to restart the Management Server indicates a problem in the upgrade. Having the Management Server restarted without any issues indicates that the upgrade is successfully completed.

12. Start all Usage Servers (if they were running on your previous version). Perform this on each Usage Server host.

    # service cloudstack-usage start

13. Additional steps are required for each KVM host. These steps will not affect running guests in the cloud. These steps are required only for clouds using KVM as hosts and only on the KVM hosts.

    a. Configure a yum or apt repository containing the CloudStack packages as outlined in the Installation Guide.

    b. Stop the running agent.

    # service cloud-agent stop

    c. Update the agent software with one of the following command sets as appropriate for your environment.

    # yum update cloud-*

    # apt-get update

    # apt-get upgrade cloud-*

    d. Edit **/etc/cloudstack/agent/agent.properties** to change the resource parameter from "com.cloud.agent.resource.computing.LibvirtComputingResource" to "com.cloud.hypervisor.kvm.resource.LibvirtComputingResource".

    e. Upgrade all the existing bridge names to new bridge names by running this script:

```
# cloudstack-agent-upgrade
```

    f. Install a libvirt hook with the following commands:

```
# mkdir /etc/libvirt/hooks
# cp /usr/share/cloudstack-agent/lib/libvirtqemuhook /etc/libvirt/hooks/qemu
# chmod +x /etc/libvirt/hooks/qemu
```

    g. Restart libvirtd.

```
# service libvirtd restart
```

    h. Start the agent.

```
# service cloudstack-agent start
```

    i. When the Management Server is up and running, log in to the CloudStack UI and restart the virtual router for proper functioning of all the features.

14. Log in to the CloudStack UI as administrator, and check the status of the hosts. All hosts should come to Up state (except those that you know to be offline). You may need to wait 20 or 30 minutes, depending on the number of hosts.

Note

Troubleshooting: If login fails, clear your browser cache and reload the page.

Do not proceed to the next step until the hosts show in Up state.

15. If you are upgrading from 3.0.x, perform the following:
    a. Ensure that the admin port is set to 8096 by using the "integration.api.port" global parameter.

       This port is used by the cloud-sysvmadm script at the end of the upgrade procedure. For information about how to set this parameter, see "Setting Global Configuration Parameters" in the Installation Guide.

    b. Restart the Management Server.

       Note

       If you don't want the admin port to remain open, you can set it to null after the upgrade is done and restart the management server.

16. Run the **cloudstack-sysvmadm** script to stop, then start, all Secondary Storage VMs, Console Proxy VMs, and virtual routers. Run the script once on each management server. Substitute your own IP address of the MySQL instance, the MySQL user to connect as, and the password to use for that user. In addition to those parameters, provide the **-c** and **-r** arguments. For example:

    ```
    # nohup cloudstack-sysvmadm -d 192.168.1.5 -u cloud -p password -c -r > sysvm.log 2>&1 &
    # tail -f sysvm.log
    ```

    This might take up to an hour or more to run, depending on the number of accounts in the system.

17. If needed, upgrade all Citrix XenServer hypervisor hosts in your cloud to a version supported by CloudStack 4.2.0. The supported versions are XenServer 5.6 SP2 and 6.0.2. Instructions for upgrade can be found in the CloudStack 4.2.0 Installation Guide under "Upgrading XenServer Versions."
18. Now apply the XenServer hotfix XS602E003 (and any other needed hotfixes) to XenServer v6.0.2 hypervisor hosts.
    a. Disconnect the XenServer cluster from CloudStack.

       In the left navigation bar of the CloudStack UI, select Infrastructure. Under Clusters, click View All. Select the XenServer cluster and click Actions - Unmanage.

       This may fail if there are hosts not in one of the states Up, Down, Disconnected, or Alert. You may need to fix that before unmanaging this cluster.

       Wait until the status of the cluster has reached Unmanaged. Use the CloudStack UI to check on the status. When the cluster is in the unmanaged state, there is no connection to the hosts in the cluster.

    b. To clean up the VLAN, log in to one XenServer host and run:

       ```
       /opt/xensource/bin/cloud-clean-vlan.sh
       ```

    c. Now prepare the upgrade by running the following on one XenServer host:

       ```
       /opt/xensource/bin/cloud-prepare-upgrade.sh
       ```

       If you see a message like "can't eject CD", log in to the VM and unmount the CD, then run this script again.

    d. Upload the hotfix to the XenServer hosts. Always start with the Xen pool master, then the slaves. Using your favorite file copy utility (e.g. WinSCP), copy the hotfixes to the host. Place them in a temporary folder such as /tmp.

       On the Xen pool master, upload the hotfix with this command:

       ```
       xe patch-upload file-name=XS602E003.xsupdate
       ```

       Make a note of the output from this command, which is a UUID for the hotfix file. You'll need it in another step later.

       Note

       (Optional) If you are applying other hotfixes as well, you can repeat the commands in this section with the appropriate hotfix number. For example, XS602E004.xsupdate.

    e. Manually live migrate all VMs on this host to another host. First, get a list of the VMs on this host:

       ```
       # xe vm-list
       ```

       Then use this command to migrate each VM. Replace the example host name and VM name with your own:

       ```
       # xe vm-migrate live=true host=host-name vm=VM-name
       ```

       Troubleshooting

       If you see a message like "You attempted an operation on a VM which requires PV drivers to be installed but the drivers were not detected," run:
       /opt/xensource/bin/make_migratable.sh b6cf79c8-02ee-050b-922f-49583d9f1a14.

    f. Apply the hotfix. First, get the UUID of this host:

       ```
       # xe host-list
       ```

       Then use the following command to apply the hotfix. Replace the example host UUID with the current host ID, and replace the hotfix UUID with the output from the patch-upload command you ran on this machine earlier. You can also get the hotfix UUID by running xe patch-list.

       ```
       xe patch-apply host-uuid=host-uuid uuid=hotfix-uuid
       ```

    g. Copy the following files from the CloudStack Management Server to the host

g. Copy the following files from the CloudStack Management Server to the host.

Copy from here...	...to here
/usr/lib64/cloud/common/scripts/vm/hypervisor/xenserver/xenserver60/NFSSR.py	/opt/xensource/sm/NFSSR.py
/usr/lib64/cloud/common/scripts/vm/hypervisor/xenserver/setupxenserver.sh	/opt/xensource/bin/setupxenserver.sh
/usr/lib64/cloud/common/scripts/vm/hypervisor/xenserver/make_migratable.sh	/opt/xensource/bin/make_migratable.sh

h. (Only for hotfixes XS602E005 and XS602E007) You need to apply a new Cloud Support Pack.

- Download the CSP software onto the XenServer host from one of the following links:

  For hotfix XS602E005: http://coltrane.eng.hq.xensource.com/release/XenServer-6.x/XS-6.0.2/hotfixes/XS602E005/56710/xe-phase-2/xenserver-cloud-supp.tgz

  For hotfix XS602E007: http://coltrane.eng.hq.xensource.com/release/XenServer-6.x/XS-6.0.2/hotfixes/XS602E007/57824/xe-phase-2/xenserver-cloud-supp.tgz
- Extract the file:

```
# tar xf xenserver-cloud-supp.tgz
```

- Run the following script:

```
# xe-install-supplemental-pack xenserver-cloud-supp.iso
```

- If the XenServer host is part of a zone that uses basic networking, disable Open vSwitch (OVS):

```
# xe-switch-network-backend  bridge
```

i. Reboot this XenServer host.

j. Run the following:

```
/opt/xensource/bin/setupxenserver.sh
```

Note

If the message "mv: cannot stat `/etc/cron.daily/logrotate': No such file or directory" appears, you can safely ignore it.

k. Run the following:

```
for pbd in `xe pbd-list currently-attached=false| grep ^uuid | awk '{print
$NF}'`; do xe pbd-plug uuid=$pbd ;
```

l. On each slave host in the Xen pool, repeat these steps, starting from "manually live migrate VMs."

Troubleshooting Tip

If passwords which you know to be valid appear not to work after upgrade, or other UI issues are seen, try clearing your browser cache and reloading the UI page.

3.3. Upgrade from 2.2.14 to 4.2.0

1. Ensure that you query your IPaddress usage records and process them; for example, issue invoices for any usage that you have not yet billed users for.

   Starting in 3.0.2, the usage record format for IP addresses is the same as the rest of the usage types. Instead of a single record with the assignment and release dates, separate records are generated per aggregation period with start and end dates. After upgrading to 4.2.0, any existing IP address usage records in the old format will no longer be available.
2. If you are using version 2.2.0 - 2.2.13, first upgrade to 2.2.14 by using the instructions in the 2.2.14 Release Notes.

   KVM Hosts

   If KVM hypervisor is used in your cloud, be sure you completed the step to insert a valid username and password into the host_details table on each KVM node as described in the 2.2.14 Release Notes. This step is critical, as the database will be encrypted after the upgrade to 4.2.0.
3. While running the 2.2.14 system, log in to the UI as root administrator.
4. Using the UI, add a new System VM template for each hypervisor type that is used in your cloud. In each zone, add a system VM template for each hypervisor used in that zone
   a. In the left navigation bar, click Templates.
   b. In Select view, click Templates.
   c. Click Register template.

      The Register template dialog box is displayed.
   d. In the Register template dialog box, specify the following values depending on the hypervisor type (do not change these):

Hypervisor	Description
XenServer	Name: systemvm-xenserver-4.2 Description: systemvm-xenserver-4.2 URL:http://download.cloud.com/templates/4.2/systemvmtemplate-

	2013-07-12-master-xen.vhd.bz2 Zone: Choose the zone where this hypervisor is used Hypervisor: XenServer Format: VHD OS Type: Debian GNU/Linux 7.0 (32-bit) (or the highest Debian release number available in the dropdown) Extractable: no Password Enabled: no Public: no Featured: no
KVM	Name: systemvm-kvm-4.2 Description: systemvm-kvm-4.2 URL: http://download.cloud.com/templates/4.2/systemvmtemplate-2013-06-12-master-kvm.qcow2.bz2 Zone: Choose the zone where this hypervisor is used Hypervisor: KVM Format: QCOW2 OS Type: Debian GNU/Linux 7.0 (32-bit) (or the highest Debian release number available in the dropdown) Extractable: no Password Enabled: no Public: no Featured: no
VMware	Name: systemvm-vmware-4.2 Description: systemvm-vmware-4.2 URL: http://download.cloud.com/templates/4.2/systemvmtemplate-4.2-vh7.ova Zone: Choose the zone where this hypervisor is used Hypervisor: VMware Format: OVA OS Type: Debian GNU/Linux 7.0 (32-bit) (or the highest Debian release number available in the dropdown) Extractable: no Password Enabled: no Public: no Featured: no

5. Watch the screen to be sure that the template downloads successfully and enters the READY state. Do not proceed until this is successful
6. **WARNING**: If you use more than one type of hypervisor in your cloud, be sure you have repeated these steps to download the system VM template for each hypervisor type. Otherwise, the upgrade will fail.
7. (KVM on RHEL 6.0/6.1 only) If your existing CloudStack deployment includes one or more clusters of KVM hosts running RHEL 6.0 or RHEL 6.1, perform the following:
    a. Ensure that you upgrade the operating system version on those hosts before upgrading CloudStack

       To do that, change the yum repository for each system with CloudStack packages, that implies that all the Management Servers and any hosts that have the KVM agent.
    b. Open **`/etc/yum.repos.d/cloudstack.repo`** on any systems that have CloudStack packages installed.
    c. Edit as follows:

       ```
       [upgrade]
       name=rhel63
       baseurl=url-of-your-rhel6.3-repo
       enabled=1
       gpgcheck=0
       [apache CloudStack]
       name= Apache CloudStack
       baseurl= http://cloudstack.apt-get.eu/rhel/4.0/
       enabled=1
       gpgcheck=0
       ```

       If you are using the community provided package repository, change the baseurl to http:// cloudstack.apt-get.eu/rhel/4.2/

       If you are using your own package repository, change this line to read as appropriate for your 4.2.0 repository.
    d. Now that you have the repository configured, upgrade the host operating system from RHEL 6.0 to 6.3:

       ```
       # yum upgrade
       ```
8. Stop all Usage Servers if running. Run this on all Usage Server hosts.

   ```
   # service cloud-usage stop
   ```
9. Stop the Management Servers. Run this on all Management Server hosts.

   ```
   # service cloud-management stop
   ```
10. On the MySQL master, take a backup of the MySQL databases. We recommend performing this step even in test upgrades. If there is an issue, this will assist with debugging.

    In the following commands, it is assumed that you have set the root password on the database, which is a CloudStack recommended best practice. Substitute your own MySQL root password.

CloudStack recommended best practice. Substitute your own MySQL root password.

```
# mysqldump -u root -pmysql_password cloud > cloud-backup.dmp
# mysqldump -u root -pmysql_password cloud_usage > cloud-usage-backup.dmp
```

11. Either build RPM/DEB packages as detailed in the Installation Guide, or use one of the community provided yum/apt repositories to gain access to the CloudStack binaries.
12. If you are using Ubuntu, follow this procedure to upgrade your packages. If not, skip to step 13.

Community Packages

This section assumes you're using the community supplied packages for CloudStack. If you've created your own packages and APT repository, substitute your own URL for the ones used in these examples.

    a. The first order of business will be to change the sources list for each system with CloudStack packages. This means all management servers, and any hosts that have the KVM agent. (No changes should be necessary for hosts that are running VMware or Xen.)

    Start by opening **/etc/apt/sources.list.d/cloudstack.list** on any systems that have CloudStack packages installed.

    This file should have one line, which contains:

    ```
    deb http://cloudstack.apt-get.eu/ubuntu precise 4.0
    ```

    We'll change it to point to the new package repository:

    ```
    deb http://cloudstack.apt-get.eu/ubuntu precise 4.2
    ```

    If you're using your own package repository, change this line to read as appropriate for your 4.2.0 repository.

    b. Now update your apt package list:

    ```
    $ sudo apt-get update
    ```

    c. Now that you have the repository configured, it's time to install the **cloudstack-management** package. This will pull in any other dependencies you need.

    ```
    $ sudo apt-get install cloudstack-management
    ```

    d. On KVM hosts, you will need to manually install the **cloudstack-agent** package:

    ```
    $ sudo apt-get install cloudstack-agent
    ```

    During the installation of **cloudstack-agent**, APT will copy your **agent.properties**, **log4j-cloud.xml**, and **environment.properties** from **/etc/cloud/agent** to **/etc/cloudstack/agent**.

    When prompted whether you wish to keep your configuration, say Yes.

    e. Verify that the file **/etc/cloudstack/agent/environment.properties** has a line that reads:

    ```
    paths.script=/usr/share/cloudstack-common
    ```

    If not, add the line.

    f. Restart the agent:

    ```
    service cloud-agent stop
    killall jsvc
    service cloudstack-agent start
    ```

    g. During the upgrade, **log4j-cloud.xml** was simply copied over, so the logs will continue to be added to **/var/log/cloud/agent/agent.log**. There's nothing *wrong* with this, but if you prefer to be consistent, you can change this by copying over the sample configuration file:

    ```
    cd /etc/cloudstack/agent
    mv log4j-cloud.xml.dpkg-dist log4j-cloud.xml
    service cloudstack-agent restart
    ```

    h. Once the agent is running, you can uninstall the old cloud-* packages from your system:

    ```
    sudo dpkg --purge cloud-agent
    ```

13. If you are using CentOS or RHEL, follow this procedure to upgrade your packages. If not, skip to step 14.

Community Packages

This section assumes you're using the community supplied packages for CloudStack. If you've created your own packages and yum repository, substitute your own URL for the ones used in these examples.

    a. The first order of business will be to change the yum repository for each system with CloudStack packages. This means all management servers, and any hosts that have the KVM agent. (No changes should be necessary for hosts that are running VMware or Xen.)

    Start by opening **/etc/yum.repos.d/cloudstack.repo** on any systems that have CloudStack packages installed.

    This file should have content similar to the following:

    ```
    [apache-cloudstack]
    ```

```
name=Apache CloudStack
baseurl=http://cloudstack.apt-get.eu/rhel/4.0/
enabled=1
gpgcheck=0
```

If you are using the community provided package repository, change the baseurl to http://cloudstack.apt-get.eu/rhel/4.2/

If you're using your own package repository, change this line to read as appropriate for your 4.2.0 repository.

b. Now that you have the repository configured, it's time to install the **cloudstack-management** package by upgrading the older **cloud-client** package.

```
$ sudo yum upgrade cloud-client
```

c. For KVM hosts, you will need to upgrade the **cloud-agent** package, similarly installing the new version as **cloudstack-agent**.

```
$ sudo yum upgrade cloud-agent
```

During the installation of **cloudstack-agent**, the RPM will copy your **agent.properties**, **log4j-cloud.xml**, and **environment.properties** from **/etc/cloud/agent** to **/etc/cloudstack/agent**.

d. Verify that the file **/etc/cloudstack/agent/environment.properties** has a line that reads:

```
paths.script=/usr/share/cloudstack-common
```

If not, add the line.

e. Restart the agent:

```
service cloud-agent stop
killall jsvc
service cloudstack-agent start
```

14. If you have made changes to your existing copy of the file components.xml in your previous-version CloudStack installation, the changes will be preserved in the upgrade. However, you need to do the following steps to place these changes in a new version of the file which is compatible with version 4.0.0-incubating.

Note

How will you know whether you need to do this? If the upgrade output in the previous step included a message like the following, then some custom content was found in your old components.xml, and you need to merge the two files:

```
warning: /etc/cloud/management/components.xml created as
/etc/cloud/management/components.xml.rpmnew
```

a. Make a backup copy of your **/etc/cloud/management/components.xml** file. For example:

```
# mv /etc/cloud/management/components.xml
/etc/cloud/management/components.xml-backup
```

b. Copy **/etc/cloud/management/components.xml.rpmnew** to create a new **/etc/cloud/management/components.xml**:

```
# cp -ap /etc/cloud/management/components.xml.rpmnew
/etc/cloud/management/components.xml
```

c. Merge your changes from the backup file into the new components.xml file.

```
# vi /etc/cloudstack/management/components.xml
```

15. After upgrading to 4.2, API clients are expected to send plain text passwords for login and user creation, instead of MD5 hash. If API client changes are not acceptable, following changes are to be made for backward compatibility:

Modify componentsContext.xml, and make PlainTextUserAuthenticator as the default authenticator (1st entry in the userAuthenticators adapter list is default)

```
<!-- Security adapters -->
<bean id="userAuthenticators" class="com.cloud.utils.component.AdapterList">
  <property name="Adapters">
    <list>
      <ref bean="PlainTextUserAuthenticator"/>
      <ref bean="MD5UserAuthenticator"/>
      <ref bean="LDAPUserAuthenticator"/>
    </list>
  </property>
</bean>
```

PlainTextUserAuthenticator works the same way MD5UserAuthenticator worked prior to 4.2.

16. If you have made changes to your existing copy of the **/etc/cloud/management/db.properties** file in your previous-version CloudStack installation, the changes will be preserved in the upgrade. However, you need to do the following steps to place these changes in a new version of the file which is compatible with version 4.0.0-incubating.

a. Make a backup copy of your file **/etc/cloud/management/db.properties**. For example:

```
# mv /etc/cloud/management/db.properties /etc/cloud/management/db.properties-
backup
```

b. Copy **/etc/cloud/management/db.properties.rpmnew** to create a new

/etc/cloud/management/db.properties:

```
# cp -ap /etc/cloud/management/db.properties.rpmnew
etc/cloud/management/db.properties
```

c. Merge your changes from the backup file into the new db.properties file.

```
# vi /etc/cloudstack/management/db.properties
```

17. On the management server node, run the following command. It is recommended that you use the command-line flags to provide your own encryption keys. See Password and Key Encryption in the Installation Guide.

```
# cloudstack-setup-encryption -e encryption_type -m management_server_key -k
database_key
```

When used without arguments, as in the following example, the default encryption type and keys will be used:

- (Optional) For encryption_type, use file or web to indicate the technique used to pass in the database encryption password. Default: file.
- (Optional) For management_server_key, substitute the default key that is used to encrypt confidential parameters in the properties file. Default: password. It is highly recommended that you replace this with a more secure value
- (Optional) For database_key, substitute the default key that is used to encrypt confidential parameters in the CloudStack database. Default: password. It is highly recommended that you replace this with a more secure value.

18. Repeat steps 10 - 14 on every management server node. If you provided your own encryption key in step 14, use the same key on all other management servers.
19. Start the first Management Server. Do not start any other Management Server nodes yet.

```
# service cloudstack-management start
```

Wait until the databases are upgraded. Ensure that the database upgrade is complete. You should see a message like "Complete! Done." After confirmation, start the other Management Servers one at a time by running the same command on each node.

20. Start all Usage Servers (if they were running on your previous version). Perform this on each Usage Server host.

```
# service cloudstack-usage start
```

21. (KVM only) Perform the following additional steps on each KVM host.

These steps will not affect running guests in the cloud. These steps are required only for clouds using KVM as hosts and only on the KVM hosts.

a. Configure your CloudStack package repositories as outlined in the Installation Guide

b. Stop the running agent.

```
# service cloud-agent stop
```

c. Update the agent software with one of the following command sets as appropriate.

```
# yum update cloud-*
```

```
# apt-get update
# apt-get upgrade cloud-*
```

d. Copy the contents of the **agent.properties** file to the new **agent.properties** file by using the following command

```
sed -i
's/com.cloud.agent.resource.computing.LibvirtComputingResource/com.cloud.hyperv
isor.kvm.resource.LibvirtComputingResource/g'
/etc/cloudstack/agent/agent.properties
```

e. Upgrade all the existing bridge names to new bridge names by running this script:

```
# cloudstack-agent-upgrade
```

f. Install a libvirt hook with the following commands:

```
# mkdir /etc/libvirt/hooks
# cp /usr/share/cloudstack-agent/lib/libvirtqemuhook /etc/libvirt/hooks/qemu
# chmod +x /etc/libvirt/hooks/qemu
```

g. Restart libvirtd.

```
# service libvirtd restart
```

h. Start the agent.

```
# service cloudstack-agent start
```

i. When the Management Server is up and running, log in to the CloudStack UI and restart the virtual router for proper functioning of all the features.

22. Log in to the CloudStack UI as admin, and check the status of the hosts. All hosts should come to Up state (except those that you know to be offline). You may need to wait 20 or 30 minutes, depending on the number of hosts.

Do not proceed to the next step until the hosts show in the Up state. If the hosts do not come to the Up state, contact support.

23. Run the following script to stop, then start, all Secondary Storage VMs, Console Proxy VMs, and virtual routers.

a. Run the command once on one management server. Substitute your own IP address of the MySQL

a. Run the command once on one management server. Substitute your own IP address of the MySQL instance, the MySQL user to connect as, and the password to use for that user. In addition to those parameters, provide the "-c" and "-r" arguments. For example:

```
# nohup cloudstack-sysvmadm -d 192.168.1.5 -u cloud -p password -c -r >
sysvm.log 2>&1 &
# tail -f sysvm.log
```

This might take up to an hour or more to run, depending on the number of accounts in the system.

b. After the script terminates, check the log to verify correct execution:

```
# tail -f sysvm.log
```

The content should be like the following:

```
                                Stopping and starting 1 secondary storage
vm(s)...
                                Done stopping and starting secondary storage
vm(s)
                                Stopping and starting 1 console proxy vm(s)...
                                Done stopping and starting console proxy vm(s).
                                Stopping and starting 4 running routing
vm(s)...
                                Done restarting router(s).
```

24. If you would like additional confirmation that the new system VM templates were correctly applied when these system VMs were rebooted, SSH into the System VM and check the version.

    Use one of the following techniques, depending on the hypervisor.

    XenServer or KVM:

    SSH in by using the link local IP address of the system VM. For example, in the command below, substitute your own path to the private key used to log in to the system VM and your own link local IP.

    Run the following commands on the XenServer or KVM host on which the system VM is present:

    ```
    # ssh -i private-key-path link-local-ip -p 3922
                            # cat /etc/cloudstack-release
    ```

    The output should be like the following:

    ```
    Cloudstack Release 4.0.0-incubating Mon Oct 9 15:10:04 PST 2012
    ```

    ESXi

    SSH in using the private IP address of the system VM. For example, in the command below, substitute your own path to the private key used to log in to the system VM and your own private IP.

    Run the following commands on the Management Server:

    ```
    # ssh -i private-key-path private-ip -p 3922
                            # cat /etc/cloudstack-release
    ```

    The output should be like the following:

    ```
    Cloudstack Release 4.0.0-incubating Mon Oct 9 15:10:04 PST 2012
    ```

25. If needed, upgrade all Citrix XenServer hypervisor hosts in your cloud to a version supported by CloudStack 4.0.0-incubating. The supported versions are XenServer 5.6 SP2 and 6.0.2. Instructions for upgrade can be found in the CloudStack 4.0.0-incubating Installation Guide.
26. Apply the XenServer hotfix XS602E003 (and any other needed hotfixes) to XenServer v6.0.2 hypervisor hosts.
    a. Disconnect the XenServer cluster from CloudStack.

       In the left navigation bar of the CloudStack UI, select Infrastructure. Under Clusters, click View All. Select the XenServer cluster and click Actions - Unmanage.

       This may fail if there are hosts not in one of the states Up, Down, Disconnected, or Alert. You may need to fix that before unmanaging this cluster.

       Wait until the status of the cluster has reached Unmanaged. Use the CloudStack UI to check on the status. When the cluster is in the unmanaged state, there is no connection to the hosts in the cluster.

    b. To clean up the VLAN, log in to one XenServer host and run:

       ```
       /opt/xensource/bin/cloud-clean-vlan.sh
       ```

    c. Prepare the upgrade by running the following on one XenServer host:

       ```
       /opt/xensource/bin/cloud-prepare-upgrade.sh
       ```

       If you see a message like "can't eject CD", log in to the VM and umount the CD, then run this script again.

    d. Upload the hotfix to the XenServer hosts. Always start with the Xen pool master, then the slaves. Using your favorite file copy utility (e.g. WinSCP), copy the hotfixes to the host. Place them in a temporary folder such as /root or /tmp.

       On the Xen pool master, upload the hotfix with this command:

       ```
       xe patch-upload file-name=XS602E003.xsupdate
       ```

       Make a note of the output from this command, which is a UUID for the hotfix file. You'll need it in another step later.

       
       Note

Note

(Optional) If you are applying other hotfixes as well, you can repeat the commands in this section with the appropriate hotfix number. For example, XS602E004.xsupdate.

e. Manually live migrate all VMs on this host to another host. First, get a list of the VMs on this host:

```
# xe vm-list
```

Then use this command to migrate each VM. Replace the example host name and VM name with your own:

```
# xe vm-migrate live=true host=host-name vm=VM-name
```

Troubleshooting

If you see a message like "You attempted an operation on a VM which requires PV drivers to be installed but the drivers were not detected," run:
/opt/xensource/bin/make_migratable.sh b6cf79c8-02ee-050b-922f-49583d9f1a14.

f. Apply the hotfix. First, get the UUID of this host:

 # xe host-list

Then use the following command to apply the hotfix. Replace the example host UUID with the current host ID, and replace the hotfix UUID with the output from the patch-upload command you ran on this machine earlier. You can also get the hotfix UUID by running xe patch-list.

xe patch-apply host-uuid=*host-uuid* uuid=*hotfix-uuid*

g. Copy the following files from the CloudStack Management Server to the host.

Copy from here...	...to here
/usr/share/cloudstack-common/scripts/vm/hypervisor/xenserver/xenserver60/NFSSR.py	**/opt/xensource/sm/NFSSR.py**
/usr/share/cloudstack-common/scripts/vm/hypervisor/xenserver/setupxenserver.sh	**/opt/xensource/bin/setupxenserver.sh**
/usr/lib64/cloudstack-common/scripts/vm/hypervisor/xenserver/make_migratable.sh	**/opt/xensource/bin/make_migratable.sh**

h. (Only for hotfixes XS602E005 and XS602E007) You need to apply a new Cloud Support Pack.

- Download the CSP software onto the XenServer host from one of the following links:

  For hotfix XS602E005: http://coltrane.eng.hq.xensource.com/release/XenServer-6.x/XS-6.0.2/hotfixes/XS602E005/56710/xe-phase-2/xenserver-cloud-supp.tgz

  For hotfix XS602E007: http://coltrane.eng.hq.xensource.com/release/XenServer-6.x/XS-6.0.2/hotfixes/XS602E007/57824/xe-phase-2/xenserver-cloud-supp.tgz
- Extract the file:

  # tar xf xenserver-cloud-supp.tgz
- Run the following script:

  # xe-install-supplemental-pack xenserver-cloud-supp.iso
- If the XenServer host is part of a zone that uses basic networking, disable Open vSwitch (OVS):

  # xe-switch-network-backend bridge

i. Reboot this XenServer host.

j. Run the following:

/opt/xensource/bin/setupxenserver.sh

Note

If the message "mv: cannot stat `/etc/cron.daily/logrotate': No such file or directory" appears, you can safely ignore it.

k. Run the following:

```
for pbd in `xe pbd-list currently-attached=false| grep ^uuid | awk '{print $NF}'`;
do xe pbd-plug uuid=$pbd ;
```

l. On each slave host in the Xen pool, repeat these steps, starting from "manually live migrate VMs."

Chapter 4. API Changes in 4.2

4.1. Added API Commands in 4.2

4.1.1. Secondary Storage

- addImageStore (Adds all types of secondary storage providers, S3/Swift/NFS)
- createSecondaryStagingStore (Adds a staging secondary storage in each zone)
- listImageStores (Lists all secondary storages, S3/Swift/NFS)
- listSecondaryStagingStores (Lists all staging secondary storages)
- addS3 (Adds a Amazon Simple Storage Service instance.) It is recommended to use addImageStore instead.
- listS3s (Lists all the Amazon Simple Storage Service instances.) It is recommended to use listImageStores instead.

4.1.2. VM Snapshot

- createVMSnapshot (Creates a virtual machine snapshot; see Section 2.4.16, "Virtual Machine Snapshots for VMware")
- deleteVMSnapshot (Deletes a virtual machine snapshot)
- listVMSnapshot (Shows a virtual machine snapshot)
- revertToVMSnapshot (Returns a virtual machine to the state and data saved in a given snapshot)

4.1.3. Load Balancer Health Check

- createLBHealthCheckPolicy (Creates a new health check policy for a load balancer rule; see Section 2.3.19, "Health Checks for Load Balanced Instances")
- deleteLBHealthCheckPolicy (Deletes an existing health check policy from a load balancer rule)
- listLBHealthCheckPolicies (Displays the health check policy for a load balancer rule)

4.1.4. Egress Firewall Rules

- createEgressFirewallRules (Creates an egress firewall rule on the guest network; see Section 2.3.10, "Enhanced Juniper SRX Support for Egress Firewall Rules")
- deleteEgressFirewallRules (Deletes a egress firewall rule on the guest network.)
- listEgressFirewallRules (Lists the egress firewall rules configured for a guest network.)

4.1.5. SSH Key

- resetSSHKeyForVirtualMachine (Resets the SSHkey for virtual machine.)

4.1.6. Bare Metal

- addBaremetalHost (Adds a new host. Technically, this API command was present in v3.0.6, but its functionality was disabled. See Section 2.4.12, "Kickstart Installation for Bare Metal Provisioning")
- addBaremetalDhcp (Adds a DHCP server for bare metal hosts)
- addBaremetalPxePingServer (Adds a PXE PING server for bare metal hosts)
- addBaremetalPxeKickStartServer (Adds a PXE server for bare metal hosts)
- listBaremetalDhcp (Shows the DHCP servers currently defined for bare metal hosts)
- listBaremetalPxePingServer (Shows the PXE PING servers currently defined for bare metal hosts)

4.1.7. NIC

- addNicToVirtualMachine (Adds a new NIC to the specified VM on a selected network; see Section 2.3.14, "Configuring Multiple IP Addresses on a Single NIC")
- removeNicFromVirtualMachine (Removes the specified NIC from a selected VM.)
- updateDefaultNicForVirtualMachine (Updates the specified NIC to be the default one for a selected VM.)
- addIpToNic (Assigns secondary IP to a NIC.)
- removeIpFromNic (Assigns secondary IP to a NIC.)
- listNics (Lists the NICs associated with a VM.)

4.1.8. Regions

- addRegion (Registers a Region into another Region; see Section 2.1.1, "Regions")
- updateRegion (Updates Region details: ID, Name, Endpoint, User API Key, and User Secret Key.)
- removeRegion (Removes a Region from current Region.)
- listRegions (Get all the Regions. They can be filtered by using the ID or Name.)

4.1.9. User

- getUser (This API can only be used by the Admin. Get user account details by using the API Key.)

4.1.10. API Throttling

- getApiLimit (Show number of remaining APIs for the invoking user in current window)

- getApiLimit (Show number of remaining APIs for the invoking user in current window)
- resetApiLimit (For root admin, if accountId parameter is passed, it will reset count for that particular account, otherwise it will reset all counters)
- resetApiLimit (Reset the API count.)

4.1.11. Locking

- lockAccount (Locks an account)
- lockUser (Locks a user account)

4.1.12. VM Scaling

- scaleVirtualMachine (Scales the virtual machine to a new service offering.)

4.1.13. Migrate Volume

- migrateVirtualMachineWithVolume (Attempts migrating VM with its volumes to a different host.)
- listStorageProviders (Lists storage providers.)
- findStoragePoolsForMigration (Lists storage pools available for migrating a volume.)

4.1.14. Dedicated IP and VLAN

- dedicatePublicIpRange (Dedicates a Public IP range to an account.)
- releasePublicIpRange (Releases a Public IP range back to the system pool.)
- dedicateGuestVlanRange (Dedicates a guest VLAN range to an account.)
- releaseDedicatedGuestVlanRange (Releases a dedicated guest VLAN range to the system.)
- listDedicatedGuestVlanRanges (Lists dedicated guest VLAN ranges.)

4.1.15. Port Forwarding

- updatePortForwardingRule (Updates a port forwarding rule. Only the private port and the VM can be updated.)

4.1.16. Scale System VM

- scaleSystemVm (Scale the service offering for a systemVM, console proxy, or secondary storage.)

4.1.17. Deployment Planner

- listDeploymentPlanners (Lists all the deployment planners available.)

4.1.18. Archive and Delete Events and Alerts

- archiveEvents (Archive one or more events.)
- deleteEvents (Delete one or more events.)
- archiveAlerts (Archive one or more alerts.)
- deleteAlerts (Delete one or more alerts.)

4.1.19. Host Reservation

- releaseHostReservation (Releases host reservation.)

4.1.20. Resize Volume

- resizeVolume (Resizes a volume.)
- updateVolume (Updates the volume.)

4.1.21. Egress Firewall Rules

- createEgressFirewallRule (Creates a egress firewall rule for a given network.)
- deleteEgressFirewallRule (Deletes an egress firewall rule.)
- listEgressFirewallRules (Lists all egress firewall rules for network.)

4.1.22. Network ACL

- updateNetworkACLItem (Updates ACL item with specified ID.)
- createNetworkACLList (Creates a Network ACL for the given VPC.)
- deleteNetworkACLList (Deletes a Network ACL.)
- replaceNetworkACLList (Replaces ACL associated with a Network or private gateway.)
- listNetworkACLLists (Lists all network ACLs.)

4.1.23. Resource Detail

- addResourceDetail (Adds detail for the Resource.)
- removeResourceDetail (Removes detail for the Resource.)
- listResourceDetails (List resource details.)

4.1.24. Nicira Integration

- addNiciraNvpDevice (Adds a Nicira NVP device.)
- deleteNiciraNvpDevice (Deletes a Nicira NVP device.)
- listNiciraNvpDevices (Lists Nicira NVP devices.)
- listNiciraNvpDeviceNetworks (Lists network that are using a Nicira NVP device.)

4.1.25. BigSwitch VNS

- addBigSwitchVnsDevice (Adds a BigSwitch VNS device.)
- deleteBigSwitchVnsDevice (Deletes a BigSwitch VNS device.)
- listBigSwitchVnsDevices (Lists BigSwitch VNS devices.)

4.1.26. Simulator

- configureSimulator (Configures a simulator.)

4.1.27. API Discovery

- listApis (Lists all the available APIs on the server, provided by the API Discovery plugin.)

4.1.28. Global Load Balancer

- createGlobalLoadBalancerRule (Creates a global load balancer rule.)
- deleteGlobalLoadBalancerRule (Deletes a global load balancer rule.)
- updateGlobalLoadBalancerRule (update global load balancer rules.)
- listGlobalLoadBalancerRules (Lists load balancer rules.)
- assignToGlobalLoadBalancerRule (Assign load balancer rule or list of load balancer rules to a global load balancer rules.)
- removeFromGlobalLoadBalancerRule (Removes a load balancer rule association with global load balancer rule)

4.1.29. Load Balancer

- createLoadBalancer (Creates a Load Balancer)
- listLoadBalancers (Lists Load Balancers)
- deleteLoadBalancer (Deletes a load balancer)
- configureInternalLoadBalancerElement (Configures an Internal Load Balancer element.)
- createInternalLoadBalancerElement (Create an Internal Load Balancer element.)
- listInternalLoadBalancerElements (Lists all available Internal Load Balancer elements.)

4.1.30. Affinity Group

- createAffinityGroup (Creates an affinity or anti-affinity group.)
- deleteAffinityGroup (Deletes an affinity group.)
- listAffinityGroups (Lists all the affinity groups.)
- updateVMAffinityGroup (Updates the affinity or anti-affinity group associations of a VM. The VM has to be stopped and restarted for the new properties to take effect.)
- listAffinityGroupTypes (Lists affinity group types available.)

4.1.31. Portable IP

- createPortableIpRange (Adds a range of portable portable IPs to a Region.)
- deletePortableIpRange (Deletes a range of portable portable IPs associated with a Region.)
- listPortableIpRanges (Lists portable IP ranges.)

4.1.32. Internal Load Balancer VM

- stopInternalLoadBalancerVM (Stops an Internal LB VM.)
- startInternalLoadBalancerVM (Starts an existing Internal LB VM.)
- listInternalLoadBalancerVMs (List internal LB VMs.)

4.1.33. Network Isolation

- listNetworkIsolationMethods (Lists supported methods of network isolation.)

4.1.34. Dedicated Resources

- dedicateZone (Dedicates a zone.)
- dedicatePod (Dedicates a pod.)
- dedicateCluster (Dedicate an existing cluster.)
- dedicateHost (Dedicates a host.)
- releaseDedicatedZone (Release dedication of zone.)
- releaseDedicatedPod (Release dedication for the pod.)
- releaseDedicatedCluster (Release dedication for cluster.)
- releaseDedicatedHost (Release dedication for host.)
- listDedicatedZones (List dedicated zones.)
- listDedicatedPods (Lists dedicated pods.)
- listDedicatedClusters (Lists dedicated clusters.)
- listDedicatedHosts (Lists dedicated hosts.)

4.2. Changed API Commands in 4.2

API Commands	Description
listNetworkACLs	The following new request parameters are added: aclid (optional), action (optional), protocol (optional) The following new response parameters are added: aclid, action, number
copyTemplate	The following new response parameters are added: isdynamicallyscalable, sshkeyenabled
listRouters	The following new response parameters are added: ip6dns1, ip6dns2, role
updateConfiguration	The following new request parameters are added: accountid (optional), clusterid (optional), storageid (optional), zoneid (optional) The following new response parameters are added: id, scope
listVolumes	The following request parameter is removed: details The following new response parameter is added: displayvolume
suspendProject	The following new response parameters are added: cpuavailable, cpulimit, cputotal, ipavailable, iplimit, iptotal, memoryavailable, memorylimit, memorytotal, networkavailable, networklimit, networktotal, primarystorageavailable, primarystoragelimit, primarystoragetotal, secondarystorageavailable, secondarystoragelimit, secondarystoragetotal, snapshotavailable, snapshotlimit, snapshottotal, templateavailable, templatelimit, templatetotal, vmavailable, vmlimit, vmrunning, vmstopped, vmtotal, volumeavailable, volumelimit, volumetotal, vpcavailable, vpclimit, vpctotal
listRemoteAccessVpns	The following new response parameters are added: id
registerTemplate	The following new request parameters are added: imagestoreuuid (optional), isdynamicallyscalable (optional), isrouting (optional) The following new response parameters are added: isdynamicallyscalable, sshkeyenabled
addTrafficMonitor	The following response parameters are removed: privateinterface, privatezone, publicinterface, publiczone, usageinterface, username
createTemplate	The following response parameters are removed: clusterid, clustername, disksizeallocated, disksizetotal, disksizeused, ipaddress, path, podid, podname, state, tags, type The following new response parameters are added: account, accountid, bootable, checksum, crossZones, details, displaytext, domain, domainid, format, hostid, hostname, hypervisor, isdynamicallyscalable, isextractable, isfeatured, ispublic, isready, ostypeid, ostypename, passwordenabled, project, projectid, removed, size, sourcetemplateid, sshkeyenabled, status, templatetag, templatetype, tags
listLoadBalancerRuleInstances	The following new response parameters are added: diskioread, diskiowrite, diskkbsread, diskkbswrite, displayvm, isdynamicallyscalable, affinitygroup
migrateVolume	The following new request parameters is added: livemigrate (optional)

	livemigrate (optional) The following new response parameters is added: displayvolume
createAccount	The following new request parameters are added: accountid (optional), userid (optional) The following new response parameters are added: accountdetails, cpuavailable, cpulimit, cputotal, defaultzoneid, ipavailable, iplimit, iptotal, iscleanuprequired, isdefault, memoryavailable, memorylimit, memorytotal, name, networkavailable, networkdomain, networklimit, networktotal, primarystorageavailable, primarystoragelimit, primarystoragetotal, projectavailable, projectlimit, projecttotal, receivedbytes, secondarystorageavailable, secondarystoragelimit, secondarystoragetotal, sentbytes, snapshotavailable, snapshotlimit, snapshottotal, templateavailable, templatelimit, templatetotal, vmavailable, vmlimit, vmrunning, vmstopped, vmtotal, volumeavailable, volumelimit, volumetotal, vpcavailable, vpclimit, vpctotal, user The following parameters are removed: account, accountid, apikey, created, email, firstname, lastname, secretkey, timezone, username
updatePhysicalNetwork	The following new request parameters is added: removevlan (optional)
listTrafficMonitors	The following response parameters are removed: privateinterface, privatezone, publicinterface, publiczone, usageinterface, username
attachIso	The following new response parameters are added: diskioread, diskiowrite, diskkbsread, diskkbswrite, displayvm, isdynamicallyscalable, affinitygroup
listProjects	The following new request parameters are added: cpuavailable, cpulimit, cputotal, ipavailable, iplimit, iptotal, memoryavailable, memorylimit, memorytotal, networkavailable, networklimit, networktotal, primarystorageavailable, primarystoragelimit, primarystoragetotal, secondarystorageavailable, secondarystoragelimit, secondarystoragetotal, snapshotavailable, snapshotlimit, snapshottotal, templateavailable, templatelimit, templatetotal, vmavailable, vmlimit, vmrunning, vmstopped, vmtotal, volumeavailable, volumelimit, volumetotal, vpcavailable, vpclimit, vpctotal
enableAccount	The following new response parameters are added: cpuavailable, cpulimit, cputotal, isdefault, memoryavailable, memorylimit, memorytotal, primarystorageavailable, primarystoragelimit, primarystoragetotal, secondarystorageavailable, secondarystoragelimit, secondarystoragetotal
listPublicIpAddresses	The following new response parameters are added: isportable, vmipaddress
enableStorageMaintenance	The following new response parameters are added: hypervisor, scope, suitableformigration
listLoadBalancerRules	The following new request parameters is added: networkid (optional) The following new response parameters is added: networkid
stopRouter	The following new response parameters are added: ip6dns1, ip6dns2, role
listClusters	The following new response parameters are added: cpuovercommitratio, memoryovercommitratio
attachVolume	The following new response parameter is added: displayvolume
updateVPCOffering	The following request parameters is made mandatory: id
resetSSHKeyForVirtualMachine	The following new request parameter is added: keypair (required) The following parameter is removed: name The following new response parameters are added: diskioread, diskiowrite, diskkbsread, diskkbswrite, displayvm, isdynamicallyscalable, affinitygroup
updateCluster	The following request parameters are removed: cpuovercommitratio, memoryovercommitratio (optional)
listPrivateGateways	The following new response parameters are added: aclid, sourcenatsupported
ldapConfig	The following new request parameters are added: listall (optional)

	(optional) The following parameters has been made optional: searchbase, hostname, queryfilter The following new response parameter is added: ssl
listTemplates	The following new response parameters are added: isdynamicallyscalable, sshkeyenabled
listNetworks	The following new response parameters are added: aclid, displaynetwork, ip6cidr, ip6gateway, ispersistent, networkcidr, reservediprange
restartNetwork	The following new response parameters are added: isportable, vmipaddress
prepareTemplate	The following new response parameters are added: isdynamicallyscalable, sshkeyenabled
rebootVirtualMachine	The following new response parameters are added: diskioread, diskiowrite, diskkbsread, diskkbswrite, displayvm, isdynamicallyscalable, affinitygroup
changeServiceForRouter	The following new request parameters are added: aclid (optional), action (optional), protocol (optional) The following new response parameters are added: id, scope
updateZone	The following new request parameters are added: ip6dns1 (optional), ip6dns2 (optional) The following new response parameters are added: ip6dns1, ip6dns2
ldapRemove	The following new response parameters are added: ssl
updateServiceOffering	The following new response parameters are added: deploymentplanner, isvolatile
updateStoragePool	The following new response parameters are added: hypervisor, scope, suitableformigration
listFirewallRules	The following request parameter is removed: traffictype The following new response parameters are added: networkid
updateUser	The following new response parameters are added: iscallerchilddomain, isdefault
updateProject	The following new response parameters are added: cpuavailable, cpulimit, cputotal, ipavailable, iplimit, iptotal, memoryavailable, memorylimit, memorytotal, networkavailable, networklimit, networktotal, primarystorageavailable, primarystoragelimit, primarystoragetotal, secondarystorageavailable, secondarystoragelimit, secondarystoragetotal, snapshotavailable, snapshotlimit, snapshottotal, templateavailable, templatelimit, templatetotal, vmavailable, vmlimit, vmrunning, vmstopped, vmtotal, volumeavailable, volumelimit, volumetotal, vpcavailable, vpclimit, vpctotal
updateTemplate	The following new request parameters are added: isdynamicallyscalable (optional), isrouting (optional) The following new response parameters are added: isdynamicallyscalable, sshkeyenabled
disableUser	The following new response parameters are added: iscallerchilddomain, isdefault
activateProject	The following new response parameters are added: cpuavailable, cpulimit, cputotal, ipavailable, iplimit, iptotal, memoryavailable, memorylimit, memorytotal, networkavailable, networklimit, networktotal, primarystorageavailable, primarystoragelimit, primarystoragetotal, secondarystorageavailable, secondarystoragelimit, secondarystoragetotal, snapshotavailable, snapshotlimit, snapshottotal, templateavailable, templatelimit, templatetotal, vmavailable, vmlimit, vmrunning, vmstopped, vmtotal, volumeavailable, volumelimit, volumetotal, vpcavailable, vpclimit, vpctotal
createNetworkACL	The following new request parameters are added: aclid (optional), action (optional), number (optional) The following request parameter is now optional: networkid The following new response parameters are added: aclid, action, number
enableStaticNat	The following new request parameters are added: vmguestip (optional)
registerIso	The following new request parameters are added: imagestoreuuid (optional), isdynamicallyscalable

	imagestoreuuid (optional), isdynamicallyscalable (optional) The following new response parameters are added: isdynamicallyscalable, sshkeyenabled
createIpForwardingRule	The following new response parameter is added: vmguestip
resetPasswordForVirtualMachine	The following new response parameters are added: diskioread, diskiowrite, diskkbsread, diskkbswrite, displayvm, isdynamicallyscalable, affinitygroup
createVolume	The following new request parameter is added: displayvolume (optional) The following new response parameter is added: displayvolume
startRouter	The following new response parameters are added: ip6dns1, ip6dns2, role
listCapabilities	The following new response parameters are added: apilimitinterval and apilimitmax. See Section 2.5.3, "API Request Throttling".
createServiceOffering	The following new request parameters are added: deploymentplanner (optional), isvolatile (optional), serviceofferingdetails (optional). isvolatie indicates whether the service offering includes Volatile VM capability, which will discard the VM's root disk and create a new one on reboot. See Section 2.4.15, "Reset VM on Reboot". The following new response parameters are added: deploymentplanner, isvolatile
restoreVirtualMachine	The following request parameter is added: templateID (optional). This is used to point to the new template ID when the base image is updated. The parameter templateID can be an ISO ID in case of restore vm deployed using ISO. See Section 2.4.14, "Changing a VM's Base Image". The following response parameters are added: diskioread, diskiowrite, diskkbsread, diskkbswrite, displayvm, isdynamicallyscalable, affinitygroup
createNetwork	The following new request parameters are added: aclid (optional), displaynetwork (optional), endipv6 (optional), ip6cidr (optional), ip6gateway (optional), isolatedpvlan (optional), startipv6 (optional) The following new response parameters are added: aclid, displaynetwork, ip6cidr, ip6gateway, ispersistent, networkcidr, reservediprange
createVlanIpRange	The following new request parameters are added: startipv6, endipv6, ip6gateway, ip6cidr Changed parameters: startip (is now optional) The following new response parameters are added: startipv6, endipv6, ip6gateway, ip6cidr
CreateZone	The following new request parameters are added: ip6dns1, ip6dns2 The following new response parameters are added: ip6dns1, ip6dns2
deployVirtualMachine	The following request parameters are added: affinitygroupids (optional), affinitygroupnames (optional), displayvm (optional), ip6address (optional) The following request parameter is modified: iptonetworklist has a new possible value, ipv6 The following new response parameters are added: diskioread, diskiowrite, diskkbsread, diskkbswrite, displayvm, isdynamicallyscalable, affinitygroup
createNetworkOffering	The following request parameters are added: details (optional), egressdefaultpolicy (optional), ispersistent (optional) ispersistent determines if the network or network offering created or listed by using this offering are persistent or not. The following response parameters are added: details, egressdefaultpolicy, ispersistent
listNetworks	The following request parameters is added: isPersistent,

	This parameter determines if the network or network offering created or listed by using this offering are persistent or not.
listNetworkOfferings	The following request parameters is added: isPersistent. This parameter determines if the network or network offering created or listed by using this offering are persistent or not. For listNetworkOfferings, the following response parameter has been added: details, egressdefaultpolicy, ispersistent
addF5LoadBalancer configureNetscalerLoadBalancer addNetscalerLoadBalancer listF5LoadBalancers configureF5LoadBalancer listNetscalerLoadBalancers	The following response parameter is removed: inline.
listRouters	For nic responses, the following fields have been added. • ip6address • ip6gateway • ip6cidr
listVirtualMachines	The following request parameters are added: affinitygroupid (optional), vpcid (optional) The following response parameters are added: diskioread, diskiowrite, diskkbsread, diskkbswrite, displayvm, isdynamicallyscalable, affinitygroup
listRouters listZones	For DomainRouter and DataCenter response, the following fields have been added. • ip6dns1 • ip6dns2 For listZones, the following optional request parameters are added: name, networktype
listFirewallRules createFirewallRule	The following request parameter is added: traffictype (optional). The following response parameter is added: networkid
listUsageRecords	The following response parameter is added: virtualsize.
deleteIso	The following request parameter is removed: forced
addCluster	The following request parameters are added: guestvswitchtype (optional), guestvswitchtype (optional), publicvswitchtype (optional), publicvswitchtype (optional) See Section 2.4.11, “CPU and Memory Over-Provisioning”. The following request parameters are removed: cpuovercommitratio, memoryovercommitratio
updateCluster	The following request parameters are added: cpuovercommitratio, ramovercommitratio See Section 2.4.11, “CPU and Memory Over-Provisioning”.
createStoragePool	The following request parameters are added: hypervisor (optional), provider (optional), scope (optional) The following request parameters have been made mandatory: podid, clusterid See Section 2.1.3, “Zone-Wide Primary Storage”. The following response parameter has been added: hypervisor, scope, suitableformigration
listStoragePools	The following request parameter is added: scope (optional) See Section 2.1.3, “Zone-Wide Primary Storage”. The following response parameters are added: hypervisor, scope, suitableformigration
updateDiskOffering	The following response parameter is added: displayoffering
changeServiceForVirtualMachine	The following response parameter are added: diskioread,

	diskiowrite, diskkbsread, diskkbswrite, displayvm, isdynamicallyscalable, affinitygroup
recoverVirtualMachine	The following response parameters are added: diskioread, diskiowrite, diskkbsread, diskkbswrite, displayvm, isdynamicallyscalable, affinitygroup
listCapabilities	The following response parameters are added: apilimitinterval, apilimitmax
createRemoteAccessVpn	The following response parameters are added: id
startVirtualMachine	The following response parameters are added: diskioread, diskiowrite, diskkbsread, diskkbswrite, displayvm, isdynamicallyscalable, affinitygroup
detachIso	The following response parameters are added: diskioread, diskiowrite, diskkbsread, diskkbswrite, displayvm, isdynamicallyscalable, affinitygroup
updateVPC	The following request parameters has been made mandatory: id, name
associateIpAddress	The following request parameters are added: isportable (optional), regionid (optional) The following response parameters are added: isportable, vmipaddress
listProjectAccounts	The following response parameters are added: cpuavailable, cpulimit, cputotal, ipavailable, iplimit, iptotal, memoryavailable, memorylimit, memorytotal, networkavailable, networklimit, networktotal, primarystorageavailable, primarystoragelimit, primarystoragetotal, secondarystorageavailable, secondarystoragelimit, secondarystoragetotal, snapshotavailable, snapshotlimit, snapshottotal, templateavailable, templatelimit, templatetotal, vmavailable, vmlimit, vmrunning, vmstopped, vmtotal, volumeavailable, volumelimit, volumetotal, vpcavailable, vpclimit, vpctotal
disableAccount	The following response parameters are added: cpuavailable, cpulimit, cputotal, isdefault, memoryavailable, memorylimit, memorytotal, primarystorageavailable, primarystoragelimit, primarystoragetotal, secondarystorageavailable, secondarystoragelimit, secondarystoragetotal
listPortForwardingRules	The following response parameters are added: vmguestip
migrateVirtualMachine	The following response parameters are added: diskioread, diskiowrite, diskkbsread, diskkbswrite, displayvm, isdynamicallyscalable, affinitygroup
cancelStorageMaintenance	The following response parameters are added: hypervisor, scope, suitableformigration
createPortForwardingRule	The following request parameter is added: vmguestip (optional) The following response parameter is added: vmguestip
addVpnUser	The following response parameter is added: state
createVPCOffering	The following request parameter is added: serviceproviderlist (optional)
assignVirtualMachine	The following response parameters are added: diskioread, diskiowrite, diskkbsread, diskkbswrite, displayvm, isdynamicallyscalable, affinitygroup
listConditions	The following response parameters are added: account, counter, domain, domainid, project, projectid, relationaloperator, threshold Removed response parameters: name, source, value
createPrivateGateway	The following request parameters are added: aclid (optional), sourcenatsupported (optional) The following response parameters are added: aclid, sourcenatsupported
updateVirtualMachine	The following request parameters are added: displayvm (optional), isdynamicallyscalable (optional) The following response parameters are added: diskioread, diskiowrite, diskkbsread, diskkbswrite, displayvm, isdynamicallyscalable, affinitygroup
destroyRouter	The following response parameters are added: ip6dns1, ip6dns2, role
listServiceOfferings	The following response parameters are added: deploymentplanner, isvolatile
listUsageRecords	The following response parameters are removed: virtualsize
createProject	The following response parameters are added: cpuavailable, cpulimit, cputotal, ipavailable, iplimit, iptotal, memoryavailable, memorylimit, memorytotal

	memoryavailable, memorylimit, memorytotal, networkavailable, networklimit, networktotal, primarystorageavailable, primarystoragelimit, primarystoragetotal, secondarystorageavailable, secondarystoragelimit, secondarystoragetotal, snapshotavailable, snapshotlimit, snapshottotal, templateavailable, templatelimit, templatetotal, vmavailable, vmlimit, vmrunning, vmstopped, vmtotal, volumeavailable, volumelimit, volumetotal, vpcavailable, vpclimit, vpctotal
enableUser	The following response parameters are added: iscallerchilddomain, isdefault
createLoadBalancerRule	The following response parameter is added: networkid
updateAccount	The following response parameters are added: cpuavailable, cpulimit, cputotal, isdefault, memoryavailable, memorylimit, memorytotal, primarystorageavailable, primarystoragelimit, primarystoragetotal, secondarystorageavailable, secondarystoragelimit, secondarystoragetotal
copyIso	The following response parameters are added: isdynamicallyscalable, sshkeyenabled
uploadVolume	The following request parameters are added: imagestoreuuid (optional), projectid (optional The following response parameters are added: displayvolume
createDomain	The following request parameter is added: domainid (optional)
stopVirtualMachine	The following response parameters are added: diskioread, diskiowrite, diskkbsread, diskkbswrite, displayvm, isdynamicallyscalable, affinitygroup
listAccounts	The following response parameters are added: cpuavailable, cpulimit, cputotal, isdefault, memoryavailable, memorylimit, memorytotal, primarystorageavailable, primarystoragelimit, primarystoragetotal, secondarystorageavailable, secondarystoragelimit, secondarystoragetotal
createSnapshot	The following response parameter is added: zoneid
updateIso	The following request parameters are added: isdynamicallyscalable (optional), isrouting (optional) The following response parameters are added: isdynamicallyscalable, sshkeyenabled
listIpForwardingRules	The following response parameter is added: vmguestip
updateNetwork	The following request parameters are added: displaynetwork (optional), guestvmcidr (optional) The following response parameters are added: aclid, displaynetwork, ip6cidr, ip6gateway, ispersistent, networkcidr, reservediprange
destroyVirtualMachine	The following response parameters are added: diskioread, diskiowrite, diskkbsread, diskkbswrite, displayvm, isdynamicallyscalable, affinitygroup
createDiskOffering	The following request parameter is added: displayoffering (optional) The following response parameter is added: displayoffering
rebootRouter	The following response parameters are added: ip6dns1, ip6dns2, role
listConfigurations	The following request parameters are added: accountid (optional), clusterid (optional), storageid (optional), zoneid (optional) The following response parameters are added: id, scope
createUser	The following request parameter is added: userid (optional) The following response parameters are added: iscallerchilddomain, isdefault
listDiskOfferings	The following response parameter is added: displayoffering
detachVolume	The following response parameter is added: displayvolume
deleteUser	The following response parameters are added: displaytext, success Removed parameters: id, account, accountid, accounttype, apikey, created, domain, domainid, email, firstname, lastname, secretkey, state, timezone, username

listSnapshots	The following request parameter is added: zoneid (optional) The following response parameter is added: zoneid
markDefaultZoneForAccount	The following response parameters are added: cpuavailable, cpulimit, cputotal, isdefault, memoryavailable, memorylimit, memorytotal, primarystorageavailable, primarystoragelimit, primarystoragetotal, secondarystorageavailable, secondarystoragelimit, secondarystoragetotal
restartVPC	The following request parameters are made mandatory: id
updateHypervisorCapabilities	The following response parameters are added: hypervisor, hypervisorversion, maxdatavolumeslimit, maxguestslimit, maxhostspercluster, securitygroupenabled, storagemotionenabled Removed parameters: cpunumber, cpuspeed, created, defaultuse, displaytext, domain, domainid, hosttags, issystem, limitcpuuse, memory, name, networkrate, offerha, storagetype, systemvmtype, tags
updateLoadBalancerRule	The following response parameter is added: networkid
listVlanIpRanges	The following response parameters are added: endipv6, ip6cidr, ip6gateway, startipv6
listHypervisorCapabilities	The following response parameters are added: maxdatavolumeslimit, maxhostspercluster, storagemotionenabled
updateNetworkOffering	The following response parameters are added: details, egressdefaultpolicy, ispersistent
createVirtualRouterElement	The following request parameters are added: providertype (optional)
listVpnUsers	The following response parameter is added: state
listUsers	The following response parameters are added: iscallerchilddomain, isdefault
listSupportedNetworkServices	The following response parameter is added: provider
listIsos	The following response parameters are added: isdynamicallyscalable, sshkeyenabled

4.3. Deprecated APIs

- addExternalLoadBalancer (Adds F5 external load balancer appliance.)
- deleteExternalLoadBalancer (Deletes a F5 external load balancer appliance added in a zone.)
- listExternalLoadBalancers (Lists F5 external load balancer appliances added in a zone.)