CloudForms 1.0

Installation Guide

Installing and Configuring CloudForms

Lana Brindley

Sarah Chua

Daniel Macpherson

Shikha Nansi

CloudForms 1.0 Installation Guide
Installing and Configuring CloudForms
Edition 1

Author	Lana Brindley	*lbrindle@redhat.com*
Author	Sarah Chua	*sachua@redhat.com*
Author	Daniel Macpherson	*dmacpher@redhat.com*
Author	Shikha Nansi	*snansi@redhat.com*

1801 Varsity Drive
Raleigh, NC 27606-2072 USA
Phone: +1 919 754 3700
Phone: 888 733 4281
Fax: +1 919 754 3701

This document describes how to install CloudForms System Engine and Cloud Engine. It also steps through the basic configuration requirements to get CloudForms running in your environment.

Preface

Cloud computing is a model for constant user-friendly access to resources in a shared pool. These resources can include servers, storage, applications and services. Red Hat CloudForms is an application suite to manage many clouds, leverage their resources and manage systems.

This guide provides step-by-step procedures for installing CloudForms applications.

1. Audience

This administration guide is mainly intended for privileged system administrators who generally:

- Are responsible for the configuration, governance and control of private and public cloud resources;
- Have RHCE-level technical expertise (or equivalent) with intermediate to advanced knowledge of Linux systems; and
- Have some familiarity with Virtualisation and Cloud technologies.

Estimated document usage is initial reference for installation of CloudForms applications.

2. 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*[1] 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.

2.1. 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.

[1] https://fedorahosted.org/liberation-fonts/

> 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.

2.2. 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"));
   }
}
```

2.3. 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.

3. Getting Help and Giving Feedback

3.1. Do You Need Help?

If you experience difficulty with a procedure described in this documentation, visit the Red Hat Customer Portal at *http://access.redhat.com*. Through the customer portal, you can:

- search or browse through a knowledgebase of technical support articles about Red Hat products.
- submit a support case to Red Hat Global Support Services (GSS).
- access other product documentation.

Red Hat also hosts a large number of electronic mailing lists for discussion of Red Hat software and technology. You can find a list of publicly available mailing lists at *https://www.redhat.com/mailman/listinfo*. Click on the name of any mailing list to subscribe to that list or to access the list archives.

3.2. We Need Feedback!

If you find a typographical error in this manual, or if you have thought of a way to make this manual better, we would love to hear from you! Please submit a report in Bugzilla: *http://bugzilla.redhat.com/* against the product **CloudForms Common.**

When submitting a bug report, be sure to mention the manual's identifier: *documentation*

If you have a suggestion for improving the documentation, try to be as specific as possible when describing it. If you have found an error, please include the section number and some of the surrounding text so we can find it easily.

Introduction

Installing CloudForms

CloudForms is an infrastructure-as-a-service (IaaS) product, designed to help you create and maintain your virtual images and systems in both private and hybrid clouds.

Infrastructure-as-a-service (IaaS) is used to deliver computing resources like computing power, storage, and networking to users through the internet. Private IaaS clouds allow organizations to have direct control over their infrastructure, although many organizations like to use publicly-available IaaS resources as well, to enable them to be more flexible. A mixture of private and public IaaS resources is known as a hybrid cloud.

CloudForms **Cloud Engine** is used to create and manage cloud computing resources.

CloudForms **System Engine** is used to update and monitor systems within private and hybrid clouds.

CloudForms **Cloud Engine** uses a single web-based user interface, as does CloudForms **System Engine**. Both can be installed from yum repositories or RPM. For CloudForms **System Engine**, you will also need to install **Red Hat Subscription Manager** on your client machines.

1.1. Prerequisites

You must meet the following conditions before installing **CloudForms**:

- A least one networked host with the minimum specifications for **CloudForms**:
  - 64-bit architecture.
  - Red Hat Enterprise Linux 6.2 or newer.
  - At least 4GB and ideally 8GB of memory. It is also recommended to use swap space where possible.
  - Default bridge network of *`virbr0`*, with no active secondary bridge. The default virtualization bridge can be checked by running the **`virsh`** command:

```
# virsh net-info default
```

- A valid Red Hat Network subscription.
- Administrative user (**`root`**) access.

Supported Browsers

Browsers that are supported by **CloudForms** are as follows:

- **Firefox 3.6** and higher
- **Internet Explorer 7** and higher

Important

Some versions of **Internet Explorer** incorrectly render the CloudForms **Cloud Engine** user interface. To resolve rendering issues, select **Internet Options** under the **Tools** menu, click the **Security** tab, then configure **Internet Explorer** with one of the following settings:

- **Add to Trusted Sites** - Select the **Trusted sites** zone and click **Sites**. Ensure the URL for your CloudForms **Cloud Engine** is in the URL field and click **Add**.
- **Display Mixed Content** - Select **Custom Level** for the appropriate security zone. Scroll to **Miscellaneous** and check **Enabled** for **Display Mixed Content**.
- **Turn off Protected Mode** - Uncheck the **Enable Protected Mode** box.

Application Specifications

Specifications for **CloudForms** application installation are as follows:

- CloudForms **System Engine**:
  - Installation on a virtual host. It is recommended to install CloudForms **System Engine** on a virtual machine on your physical host using **Virtual Machine Manager**. Install **Virtual Machine Manager** from the Red Hat Network with the following command:

    ```
    # yum install virt-manager
    ```

    Install CloudForms **System Engine** on a physical host only if the host is separate from the CloudForms **Cloud Engine** host.
- CloudForms **Cloud Engine**:
  - Installation on the physical host. This enables CloudForms **Cloud Engine** to access CPU virtualization extensions directly for the creation of virtual machine images.

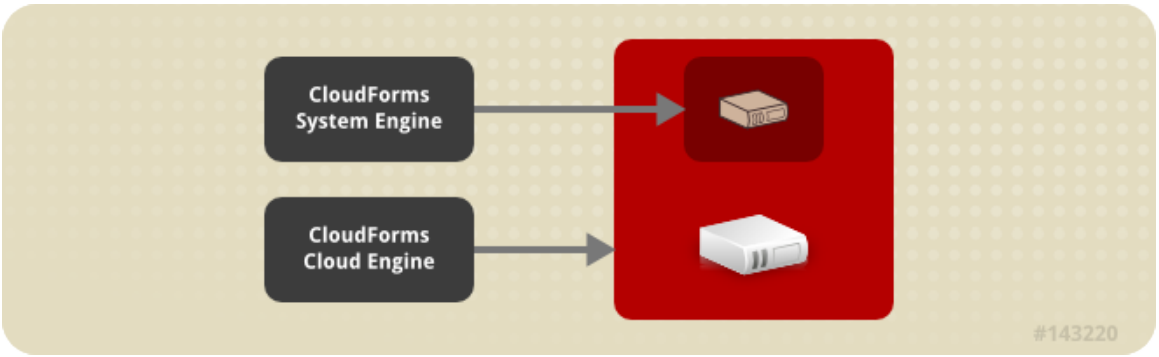

Figure 1.1. Installing CloudForms Cloud Engine to the physical host and CloudForms System Engine to a virtual host within the physical host

System Engine Installation

2.1. Installing from Red Hat Network

About System Engine

CloudForms **System Engine** is an Infrastructure as a Service (IaaS) application that provides tools to update and monitor systems within private and hybrid clouds. **System Engine** can be used to configure new systems, subscribe to updates, and maintain installations in distributed environments.

Installing from the Repository

Task Summary

The following procedure installs CloudForms **System Engine** on a host from the repository.

Task Prerequisites

The following conditions must be met before continuing with this task:

- An installation of the base operating system on a physical or virtual host.
- Ensure port 443 for HTTPS (secure WWW) is open for incoming connections. This can be achieved through the **`system-config-firewall-tui`** tool.
- Ensure Port 5674 is open for SSL communication with managed systems. This can be achieved through the **`system-config-firewall-tui`** tool.
- Check that *`hostname`* and *`localhost`* resolve correctly, using the following commands:

```
ping -c1 localhost
ping -c1 `hostname -s` # my_system
ping -c1 `hostname -f` # my_system.domain.com
```

Procedure 2.1. Installing from the Repository

1. Add the Red Hat Network channel using the following command as the **`root`** user:

```
# rhn-channel --add --channel=rhel-x86_64-server-6-cf-se-1
```

2. Install the **`katello-all`** package using **`yum install`** as the root user:

```
# yum install katello-all
```

Result:

CloudForms **System Engine** is installed on your CloudForms host.

Important

Before CloudForms **System Engine** can be used, it needs to be configured.

Switch to the root user, and run the script.

```
# katello-configure
```

2. When the configuration script has completed successfully, it displays:

```
Starting Katello configuration
The top-level log file is
[/var/log/katello/katello-configure-20111115-170733/main.log]
Creating Katello database user
############################################################ ... OK
Creating Katello database
############################################################ ... OK
Creating Candlepin database user
############################################################ ... OK
Creating Candlepin database
############################################################ ... OK
Candlepin setup
```

```
# katello-configure
Starting Katello configuration
The top-level log file is
[/var/log/katello/katello-configure-20111115-170733/main.log]
err: /Stage[main]/Postgres::Service/Service[postgresql]/ensure: change from
stopped to running failed: Could not start Service[postgresql]: Execution of
'/sbin/service postgresql start' returned 1:  at
```

```
Failed, please check [#{processing_logfile}]
```

3. The **`katello-configure`** command supports the ability to override various default settings, including the administrative username, password and organization name.

```
# katello-configure --user-name=adminuser --user-pass=password --org-name=Example_Org
```

Important

The default username is *`admin`* and the default password is *`admin`*. The default organization name is *`ACME_Corporation`*. It is strongly recommended that you override these default settings prior to your first login. For more information, see *Section 2.3.1.1, "Logging into CloudForms System Engine"*.

Result:

CloudForms **System Engine** is configured on your CloudForms host.

Configuration Using an Answer File

Task Summary

The following task shows how to configure **System Engine** with an Answer File.

Procedure 2.3. Configuration Using an Answer File

For an automated installation with customized options, an answer file can be created and passed to the **`katello-configure`** command.

1. Copy the answer file located at **`/usr/share/katello/install/default-answer-file`** to a location on your local filesystem:

```
# cp  /usr/share/katello/install/default-answer-file /root/my-answer-file
```

2. Open the local copy of the answer file in your preferred text editor and edit the values to suit your environment:

```
# Path of the answer file.
answer_file =

# Katello database name.
# PostgreSQL database name used to store the Katello database
# objects.
db_name = katelloschema

# Katello database user.
db_user = katellouser

# Katello database password.
db_password = katellopw
```

   Save your answer file once you have finished editing it.

3. Pass the answer file to the **`katello-configure`** command:

```
# katello-configure --answer-file=/root/my-answer-file

Starting Katello configuration
The top-level log file is
[/var/log/katello/katello-configure-20111115-170733/main.log]
Creating Katello database user
############################################################ ... OK
Creating Katello database
############################################################ ... OK
Creating Candlepin database user
############################################################ ... OK
Creating Candlepin database
############################################################ ... OK
Candlepin setup
############################################################ ... OK
Populating Katello database schema
############################################################ ... OK
```

Result:

System Engine is configured on your CloudForms host.

2.2.1. Configuration Options

Additional options can be passed to the **katello-configure** command, to adjust various aspects of the configuration. View a complete list of options using the command:

```
# katello-configure --help
```

Table 2.1. Configuration Options

Parameter	Description
answer-file=*ANSWER_FILE*	Path to an answer file to be used for configuration
user-name=*USER_NAME*	The administrative username (default value: *admin*)
user-pass=*USER_PASS*	The administrative user's password (default value: *admin*)
user-email=*USER_EMAIL*	Katello user's email (default value: *root@localhost*)
org-name=*ORG_NAME*	Initial organization (default value: *ACME_Corporation*)
proxy-url=*PROXY_URL*	HTTP proxy URL (for example: http://172.31.1.1)
proxy-port=*PROXY_PORT*	HTTP Proxy port (default value: *3128*)
proxy-user=*PROXY_USER*	HTTP proxy username, if authentication is required
proxy-pass=*PROXY_PASS*	HTTP proxy password, if authentication is required
db-name=*DB_NAME*	Katello database name
db-user=*DB_USER*	Katello database username
db-password=*DB_PASSWORD*	Katello database password
deployment=*DEPLOYMENT*	The deployment type to use
non-interactive=*NON_INTERACTIVE*	Non-interactive installer mode
skip-ssl-ca-generation=*SKIP_SSL_CA_GENERATION*	skip SSL CA generation
ssl-ca-password=*SSL_CA_PASSWORD*	SSL CA password
ssl-ca-country=*SSL_CA_COUNTRY*	SSL CA country
ssl-ca-state=*SSL_CA_STATE*	SSL CA state
ssl-ca-city=*SSL_CA_CITY*	SSL CA city
ssl-ca-org=*SSL_CA_ORG*	SSL CA organization
ssl-ca-org-unit=*SSL_CA_ORG_UNIT*	SSL CA organization unit
ssl-ca-cn=*SSL_CA_CN*	SSL CA common name
ssl-ca-email=*SSL_CA_EMAIL*	SSL CA e-mail address
ssl-cert-expiration=*SSL_CERT_EXPIRATION*	SSL certificate expiration (in days)

Parameter	Description
ssl-ca-password-file=*SSL_CA_PASSWORD_FILE*	SSL CA password file path
keystore-password-file=*KEYSTORE_PASSWORD_FILE*	Keystore password file path
nss-db-password-file=*NSS_DB_PASSWORD_FILE*	NSS DB password file path
only-show-config	Print details of the configuration and exit without making any changes
help, **h**	Show this short summary

2.3. Booting CloudForms System Engine for the first time

2.3.1. Accessing CloudForms System Engine

CloudForms **System Engine** is accessed using a web interface. In order to use the web interface, you will require a username and password. You can create your username and password during the initial configuration of **System Engine**.

Important

By default, the initial username and password are both set to *admin*. If you have used the default values during the configuration process, change them immediately after logging in for the first time. For more information, see *Section 2.3.1.1, "Logging into CloudForms System Engine"*.

2.3.1.1. Logging into CloudForms System Engine

CloudForms **System Engine** Login

These steps show how to log into CloudForms **System Engine**.

Task Prerequisites

You must meet the following conditions before continuing with this task:

- A configured CloudForms **System Engine**.

Procedure 2.4. Logging into CloudForms **System Engine**

1. Access CloudForms **System Engine** using a web browser at the address **`https://[HOSTNAME or IP ADDRESS]/katello`**.

   To identify your hostname, use the **`hostname`** command at the prompt:

   ```
   $ hostname
   ```

   To identify your IP address, use the **`ip add show`** command at the prompt and locate the *`inet`* address for your primary connection:

   ```
   $ ip add show
   ```

2. Enter the username and password that you created during the configuration process.

   If you forget your password, click on **Forgotten username or password** and an email with instructions on resetting your password with be sent to you.

Result

When you have successfully logged in, you are taken to the CloudForms **System Engine** dashboard, which displays critical information about your subscriptions and lists other important information.

2.3.1.2. Using the CloudForms System Engine Dashboard

CloudForms **System Engine** Dashboard

The dashboard is the first screen seen after logging in to CloudForms **System Engine**. It provides a status overview of the subscriptions and systems currently registered, an overview of promotions and synchronization as well as a list of the latest notifications.

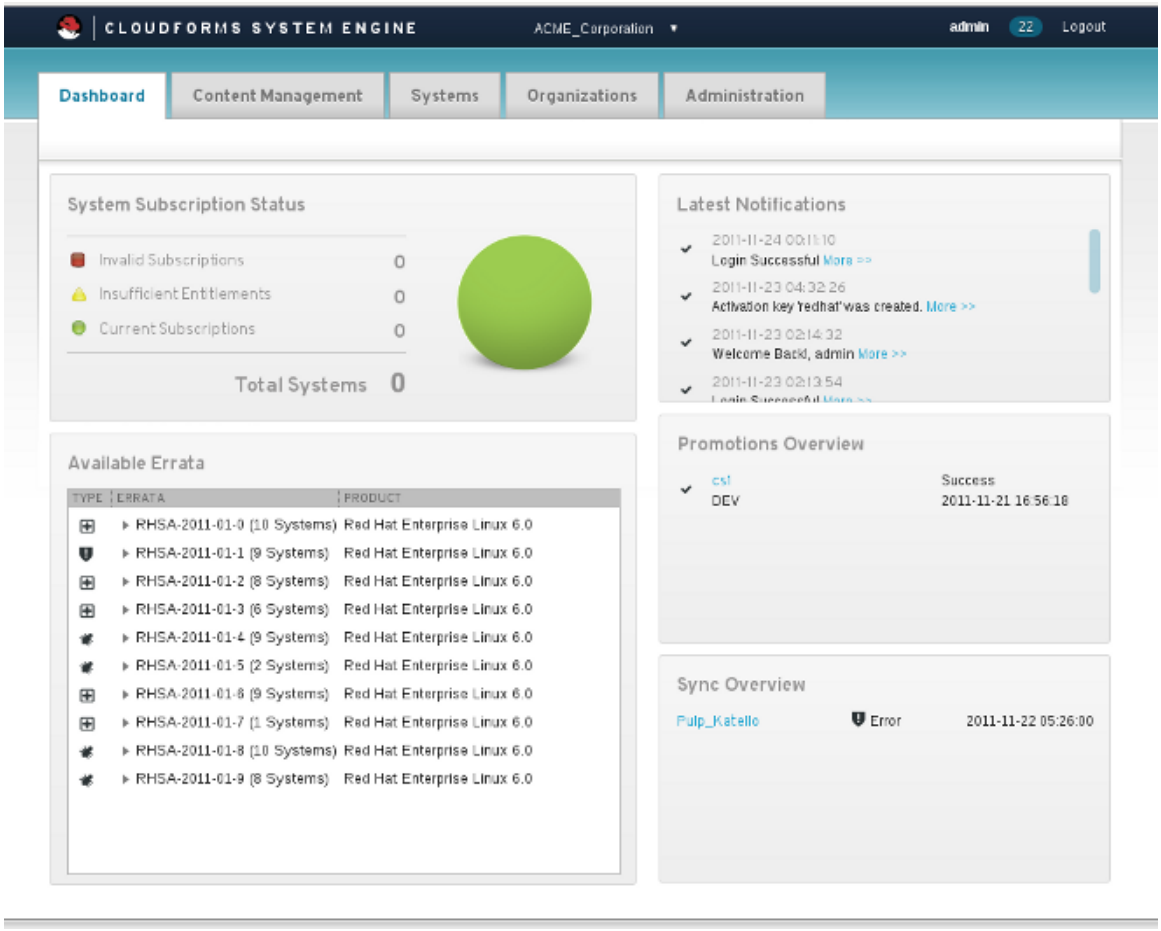

Figure 2.1. The CloudForms System Engine Dashboard

CloudForms **System Engine** is used to manage *entitlements* for client machines. Each entitlement provides access to a specified number of *certificates*. Each certificate grants the right for the client machine to download, update, and receive support for a product.

System Subscription Status

The **System Subscription Status** gives an overview of the status of the subscriptions currently being managed by CloudForms **System Engine**. A subscription is a purchased certificate that unlocks access to software, upgrades and security fixes for systems.

Table 2.2. System Subscription States

State	Description	Icon
Invalid Subscriptions	These are systems that have products installed, but have not consumed a subscription. These systems need attention immediately.	
Insufficient Entitlements	These are systems that have consumed a subscription and have a valid entitlement, but that are not consuming their full entitlements. These systems should be monitored to ensure they are configured as expected.	
Current Subscriptions	These are systems that have a valid entitlement and are consuming their full entitlements.	

Available Errata

A list of errata that are currently available for installed systems. Click the drop-down arrow to the left of the erratum title to see more information.

Latest Notifications

All messages produced by the system are listed in the **Latest Notifications** section. This includes administration information, product and subscription changes, and any errors.

This section should be monitored for global notifications sent to all users as well as to pick up any unusual activity or errors.

Promotions Overview

All changesets that have been promoted or are being promoted are listed in the **Promotions Overview** section. Click on the name of a changeset to view the changeset history.

Synchronization Overview

All products that have been synchronized are listed in the **Sync Overview** section. Click on a product name to view the synchronization status.

Cloud Engine Installation

3.1. Installing from Red Hat Network

About CloudForms **Cloud Engine**

CloudForms **Cloud Engine** is a management application for cloud resources. It provides the following functions:

- Tools to abstract and utilize resources from multiple public and private cloud resource providers.
- Creation of components used for producing virtual machine instances on cloud resource providers.
- Deployment and runtime configuration of virtual machine instances.
- Definition of deployment quotas and permissions for users.

Installing Cloud Engine

Task Summary

The following procedure installs CloudForms **Cloud Engine** on a host.

Task Prerequisites

The following conditions must be met before continuing with this task:

- An installation of the base operating system on a physical host.
- Ensure port 443 for HTTPS (secure WWW) is open for incoming connections. This can be achieved through the **`system-config-firewall-tui`** tool.

Procedure 3.1. Installing Cloud Engine

1. Add the Red Hat Network channel using the following command as the **`root`** user:

```
# rhn-channel --add --channel=rhel-x86_64-server-6-cf-ce-1
```

2. Run the following command as the **`root`** user:

```
# yum install aeolus-all
```

Wait for **yum** to finish installing **Cloud Engine** and all dependencies.

Result:

Cloud Engine is installed on your host.

3.2. Configuring CloudForms Cloud Engine

Important

Configuration of private cloud resource providers, such as Red Hat Enterprise Virtualization and VMware vSphere, require additional configuration options. Please view *Appendix A, Private Cloud Configuration* for information about these configuration options.

Configuring CloudForms **Cloud Engine**

Task Summary

This task configures CloudForms **Cloud Engine** after installation.

Task Prerequisites

You must meet the following conditions before continuing with this task:

- Configuration requires CloudForms **Cloud Engine** installation on a Red Hat Enterprise Linux host.

Procedure 3.2. Configuring Cloud Engine

1. **Configuring CloudForms Cloud Engine with the Configuration Script**
   Run the configuration script

   a. Run the following command as the **`root`** user:

   ```
   # aeolus-configure [optional-arguments].
   ```

   b. Wait for the configuration script to complete the initial configuration of CloudForms **Cloud Engine**

2. The **`aeolus-configure`** allows a series of optional arguments to modify your installation.

   Optional arguments

   -h, --help
   Show the syntax format for **`aeolus-configure`**.

   -d, --debug
   Use debug mode when logging.

   -v, --verbose
   Display configuration messages in greater detail.

   -i, --interactive
   Use the interactive configuration method.

   -p *`[profile]`*, --profile *`[profile]`*
   Use a predefined profile for configuration. Choose from **`ec2`**, **`rhevm`** or **`vsphere`**.

Result:

The CloudForms **Cloud Engine** installation is configured.

3.2.1. Setting up LDAP for Cloud Engine

CloudForms **Cloud Engine** includes Lightweight Directory Access Protocol (LDAP) authentication for users. Setup a connection to your LDAP server with the **`settings.yml`** located on your CloudForms server. Open a **terminal** session and enter the following command:

```
# vi /usr/share/aeolus-conductor/config/settings.yml
```

Edit the **`:auth:`** settings in the **`settings.yml`** using the following properties:

Table 3.1. Cloud Engine LDAP Settings

Property	Description
`:strategy:`	The authentication method. The default setting is **`database`**, which authenticates users against the CloudForms database. Change this setting to **`ldap`** to enable LDAP authentication.
`:host:`	The hostname or IP address for the LDAP server.
`:username_dn:`	The distinguished name of the user to authenticate. For example: `uid=%s,ou=CloudForms Users,dc=cfserver,dc=com` The **`%s`** variable is required to substitute for a Cloud Engine user login.
`:port:`	The LDAP server's port. The default is 389.

3.3. Booting CloudForms Cloud Engine for the first time

3.3.1. Accessing CloudForms Cloud Engine

Accessing CloudForms **Cloud Engine**

Task Summary

Access CloudForms **Cloud Engine** with a web browser. Login to CloudForms **Cloud Engine** with your username and password

Task Prerequisites

You must meet the following conditions before continuing with this task:

- This task requires a configured CloudForms **Cloud Engine** installation.

Procedure 3.3. Accessing CloudForms **Cloud Engine**

1. Use your preferred internet browser to navigate to `https://[CloudEngineHost]/conductor`
2. Enter your **username** and **password** and click **Login**.

Result

A successful login takes you to the CloudForms **Cloud Engine Monitor** Dashboard, which shows critical information about your cloud environments.

Important

The default password for the **admin** user is **password**. It is recommended to change this password after your first login.

3.4. Installing Audrey

Audrey

Audrey is a set of tools for run-time configuration on cloud instances. Audrey communicates using two components:

- **Audrey Configuration Server** - This component acts as a proxy for CloudForms **Cloud Engine** to communicate with Audrey-enabled instances.
- **Audrey Agent** - This component is installed on an instance to communicate with the **Audrey Configuration Server**. Users should include this component in an image template as an additional package to install in order to enable Audrey on an instance. The Audrey Agent communicates with the Audrey Configuration Server when the instance launches. This enables a CloudForms **Cloud Engine** user to perform further configuration using a customizable script.

Installation and configuration of Audrey is optional. A user only installs Audrey if virtual machine instances require runtime configuration.

Audrey Configuration Server requires the following specifications:

- A physical or virtual host with a 64-bit CPU. Ensure this host is a seperate host from both your CloudForms **System Engine** and CloudForms **Cloud Engine** hosts.
- At least 2GB of RAM.
- A minimum of 10GB hard disk space. Provide your **Audrey Configuration Servers** with additional durability through the use of an elastic storage solution. Mount your elastic storage at:

  ```
  /var/lib/aeolus-configserver/configs
  ```

  The **aeolus** user (**uid/gid: 180**) requires write permissions to this directory.
- An exposed domain name or IP address. This ensures instances on a cloud resource providers have access to the configuration server for runtime configuration.

3.4.1. Installing Audrey

Installing **Audrey Configuration Server**

Task Summary

The following procedure installs the **Audrey Configuration Server** on a host.

Task Prerequisites

You must meet the following conditions before continuing with this task:

- Ensure cloud resource providers can access the host for Audrey either via a fully qualified hostname or IP address.

Procedure 3.4. Installing **Audrey Configuration Server**

1. Add the Red Hat Network channel using the following command as the **`root`** user:

```
# rhn-channel --add --channel=rhel-x86_64-server-6-cf-ce-1
```

2. Run the following command as the **`root`** user:

```
# yum install aeolus-configserver
```

   Wait for **yum** to finish installing **Audrey Configuration Server** and its dependencies.

Result:

Audrey Configuration Server is installed on your host.

3.4.2. Configuring Audrey

Configuring Audrey

Task Summary

This task configures the **Audrey Configuration Server** for CloudForms **Cloud Engine**.

Task Prerequisites

You must meet the following conditions before continuing with this task:

- The configuration requires **Audrey Configuration Server** installation on a host.

Procedure 3.5. Proxy configuration

1. Run the following command as the **`root`** user:

```
# aeolus-configserver-setup
```

2. The following message displays:

```
This script will help you configure Apache as a proxy for a Config Server.
Typically this is only useful if you are not familiar with Apache
configurations and modules, specifically with mod_proxy, mod_auth_basic, and
mod_ssl.

Also, this configuration tool assumes that you are not currently running Apache
for any purposes on this server.  This configuration tool will create a Named
Virtual Host for *:443.  If this server is currently using Apache to serve
secure pages on port 443, then this tool should not be used.

Do you wish to continue [y/N]:
```

   Type **`Y`** and press **`Enter`**.

3. The script asks the configuration server's URL:

```
Enter the application URL [http://[hostname]:4567/]:
```

   Press **`Enter`** for the default URL and port number. Otherwise, input the fully qualified domain name and port number of your configuration server.

4. Authentication details for your configuration server display:

```
App URL: [configuration server url]
Conductor Auth Key: [configuration server key]
Conductor Auth Secret: [configuration server password]
```

   Note these authentication details for configuration with a resource provider account.

Result:

Audrey Configuration Server is configured for **Cloud Engine** use.

Appendix A. Private Cloud Configuration

A.1. Red Hat Enterprise Virtualization

Edit the configuration file located at **`/etc/aeolus-configure/nodes/rhevm_configure`**.

Table A.1. Configuration options for Red Hat Enterprise Virtualization providers

Option	Description
`nfs_server`	The fully qualified domain name or IP address of the Network File System (NFS) server for Red Hat Enterprise Virtualization environment.
`nfs_export`	The export domain on the NFS server.
`nfs_mount_point`	The mount point to the export domain on the NFS server.
`deltacloud_username`	Username for the chosen Red Hat Enterprise Virtualization account.
`deltacloud_password`	The password for the chosen Red Hat Enterprise Virtualization account.
`deltacloud_api`	The URL of the Red Hat Enterprise Virtualization REST API. This URL takes the form of `https://[RHEVM-HOST]:8443/api`.
`deltacloud_data_center`	The UUID for the chosen data center.
`push_timeout`	The timeout value for pushing images to a Red Hat Enterprise Virtualization provider.

Export Domain

Cloud Engine uses the **`nfs_export`** value to find the necessary export domain on your Red Hat Enterprise Virtualization provider. Ensure the value for **`nfs_export`** is the same listed with your provider. Ensure also your export domain has a consistent hostname or IP address.

ISO Domain

Red Hat Enterprise Virtualization Manager's default boot protocol is set to boot from CD/DVD before booting from the hard disk. If no ISO domain exists, the instance fails to launch. Please ensure your Red Hat Enterprise Virtualization environment contains an ISO Domain.

Querying Data Center UUIDs

The **`deltacloud_data_center`** option refers to the UUID for the chosen data center in your Red Hat Enterprise Virtualization environment. View a list of data center UUIDs using the following command on your Red Hat Enterprise Virtualization Manager's REST API:

```
# curl https://[RHEVM-HOST]:8443/api/datacenters --user [USER]@[DOMAIN]:[PASSWORD]
```

To search for a data center with a specific name, use the following command:

```
# curl https://[RHEVM-HOST]:8443/api/datacenters?search=name%3D[DATACENTER-NAME] --
user [USER]@[DOMAIN]:[PASSWORD]
```

The API returns an XML representation of each data center and its UUID:

```
<datacenters>
  <datacenter id="[UUID]" href="/api/datacenters/[UUID]">
    <name>Data_Center_01</name>
    ...
  </datacenter>
  <datacenter id="[UUID]" href="/api/datacenters/[UUID]">
    <name>Data_Center_02</name>
    ...
  </datacenter>
  ...
</datacenters>
```

Use this UUID to identify your chosen data center for the **deltacloud_data_center** option when configuring Cloud Engine. For more information, see the *Red Hat Enterprise Virtualization REST API Guide*.

Using Audrey with Red Hat Enterprise Virtualization

Each hypervisor in a Red Hat Enterprise Virtualization environment requires the **floppyinject** VDSM hook to accept user data. Ensure to install on each hypervisor the **vdsm-hook-floppyinject** package from the **rhel-x86_64-server-6-cf-tools-1** Red Hat Network channel.

Procedure A.1. Installing **floppyinject** on your hypervisor

1. Login to your hypervisor as the **root** user.
2. Add the CloudForms tools channel. For Red Hat Enterprise Linux 6-based hypervisors:

   ```
   # rhn-channel --add --channel=rhel-x86_64-server-6-cf-tools-1
   ```

3. Install the **floppyinject** VDSM hook:

   ```
   # yum install vdsm-hook-floppyinject
   ```

   yum installs and configures the **floppyinject** VDSM hook on your hypervisor.

A.2. VMware vSphere

Edit the configuration file located at **/etc/aeolus-configure/nodes/vsphere_configure**

Table A.2. Configuration options for VMware vSphere providers

Option	Description
deltacloud_provider	The URL of the vSphere provider.
username	Username for the chosen vSphere account.
password	The password for the chosen vSphere account.
datastore	The name of the chosen vSphere datastore.
network_name	The name of the chosen vSphere network.

[1] For more information, see *http://www.vmware.com/support/developer/vc-sdk/wssdk_5_0_releasenotes.html#knownissues*.

vSphere 5.0 Server WSDL

vSphere 5.0 Server requires additional files from the VMware vSphere Web Services SDK due to a known issue [1] . These files are **`reflect-messagestypes.xsd`** and **`reflect-types.xsd`**. Use the following steps to rectify this issue:

1. Download a copy of the VMware vSphere Web Services SDK to your vSphere 5.0 Server. Find this file in the Community Downloads section of the VMware website.
2. Extract the Web Services SDK to a temporary location on your vSphere 5.0 Server.
3. Copy the files from the Web Services SDK to your vSphere installation:

```
C:\> copy [extract-directory]\SDK\vsphere-ws\wsdl\vim25\reflect-messagestypes.xsd
      C:\ProgramData\VMware\VMware VirtualCenter\docRoot\sdk\

C:\> copy [extract-directory]\SDK\vsphere-ws\wsdl\vim25\reflect-types.xsd
      C:\ProgramData\VMware\VMware VirtualCenter\docRoot\sdk\
```

Important

Cloud Engine fails to push images to your vSphere provider if these files are missing. Please ensure to copy these files from the Web Services SDK to your vSphere environment.

Appendix B. Elastic Storage for Image Warehouse

This appendix provides instructions for configuring an elastic storage solution for CloudForms **Cloud Engine's** Image Warehouse service (**iwhd**), which stores virtual machine images. It is recommended to configure CloudForms **Cloud Engine** to use an elastic file system to expand storage for Image Warehouse (**iwhd**). Without an elastic file system, Image Warehouse might exceed the allocated storage space.

Use one of the following options to help configure your storage for Image Warehouse.

Mount storage at the default Image Warehouse location

The default Image Warehouse location is `/var/lib/iwhd`. Move the data in the Image Warehouse directory to a temporary location:

```
# mv /var/lib/iwhd/* [TEMPDIR]
```

Configure the mounted storage in the CloudForms **Cloud Engine** **`/etc/fstab`** file:

```
[STORAGE]  /var/lib/iwhd  [TYPE]  [OPTIONS]  0  0
```

Run the following command to mount the storage:

```
# mount -a
```

Move the original contents back to `/var/lib/iwhd`:

```
# mv [TEMPDIR]/* /var/lib/iwhd/
```

Reboot the CloudForms **Cloud Engine** host for the changes to take effect.

Mount storage at a user-defined location

CloudForms **Cloud Engine** provides a configuration file to set a user-defined location for the Image Warehouse data. Create the user-defined location on your CloudForms **Cloud Engine** host:

```
# mkdir /my/location
```

Configure the mounted storage in the CloudForms **Cloud Engine** **`/etc/fstab`** file:

```
[STORAGE]  /my/location  [TYPE]  [OPTIONS]  0  0
```

Run the following command to mount the storage:

```
# mount -a
```

Move the contents of `/var/lib/iwhd` to your new location:

```
# mv /var/lib/iwhd/* /my/location/
```

Change the Image Warehouse **`path`** in **`/etc/iwhd/conf.js`**:

```
"path": "/my/location"
```

Reboot the CloudForms **Cloud Engine** host for the changes to take effect.

Appendix C. Uninstall Procedure

The following appendix provides instructions for uninstalling CloudForms applications.

C.1. Uninstalling System Engine

The following procedure uninstalls CloudForms **System Engine**.

Procedure C.1. Uninstalling System Engine

1. Shutdown all CloudForms **System Engine** services:

```
# service mongod stop; service pulp-server stop; service tomcat6 stop; service katello
 stop; service katello-jobs stop;
# kill -9 `ps -aef | grep katello | grep -v grep | awk '{print $2}'`
# kill -9 `ps -aef | grep delayed_job | grep -v grep | awk '{print $2}'`
```

2. Uninstall CloudForms **System Engine** packages:

```
# yum erase -y `rpm -qa | grep candlepin` `rpm -qa | grep katello` `rpm -qa | grep
 ^pulp` `rpm -qa | grep mongo` `rpm -qa | grep postgre` `rpm -qa | grep httpd` `rpm -qa
 | grep ^mod_` puppet tomcat6 `rpm -qa | grep ^rubygem` ruby rubygems elasticsearch
```

3. Delete configuration files:

```
# rm -rf /etc/pulp/ /etc/candlepin/ /etc/katello/ /usr/share/katello/ /var/lib/puppet/ /
var/lib/pgsql/ /var/lib/mongodb/ /var/lib/katello/ /var/lib/pulp/ /etc/httpd/ /etc/
tomcat6/ /etc/elasticsearch /var/lib/elasticsearch /usr/share/pulp
```

4. Delete log files:

```
# rm -rf /var/log/katello/ /var/log/tomcat6/ /var/log/pulp/ /var/log/candlepin/ /var/
log/httpd/ /var/log/mongodb/
```

5. Delete CloudForms **System Engine** certificate files:

```
# rm -rf /etc/pki/pulp/ /etc/pki/content/* /etc/pki/katello /root/ssl-build
```

C.2. Uninstalling Cloud Engine

To remove the configuration data from an Aeolus server, run the aeolus-cleanup command as the **`root`** user:

```
# aeolus-cleanup
```

This command clears the CloudForms **Cloud Engine** database and permanently deletes all-related data on the local Aeolus server.

Note

The **`aeolus-cleanup`** command does not remove any data from cloud resource providers. Ensure to manually remove any information in configured cloud resource providers.

Appendix D. Revision History

Revision 0-53 Mon Jun 25 2012 **Daniel Macpherson** *dmacpher@redhat.com*

Reverting revision from 0-52

Revision 0-52 Tue Jun 19 2012 **Daniel Macpherson** *dmacpher@redhat.com*

Minor revision for 1.0.1

Revision 0-51 Tue Apr 24 2012 **Daniel Macpherson** *dmacpher@redhat.com*

Removing references to beta
BZ#815329 - Removing reference **vdsm-hook-floppyinject** for RHEL5 hosts

Revision 0-48 Tue Apr 10 2012 **Daniel Macpherson** *dmacpher@redhat.com*

Dropping FINAL documentation to translation

Revision 0-47 Thu Apr 5 2012 **Daniel Macpherson** *dmacpher@redhat.com*

Final minor revisions

Revision 0-46 Tue Apr 4 2012 **Daniel Macpherson** *dmacpher@redhat.com*

BZ#807467, BZ#807469 - Added admonition to supported browsers regarding configuration of IE8

Revision 0-45 Tue Apr 3 2012 **Daniel Macpherson** *dmacpher@redhat.com*

BZ#795906 - Added Elastic Storage Solution for iwhd as Appendix

Revision 0-44 Tues Apr 02 2012 **Sarah Chua** *sachua@redhat.com*

BZ#798378 - Add Supported Browsers

Revision 0-43 Thur Mar 22 2012 **Daniel Macpherson** *dmacpher@redhat.com*

BZ#797273 - Resolution to System Engine configuration issue

Revision 0-36 Thur Mar 8 2012 **Daniel Macpherson** *dmacpher@redhat.com*

BZ#800108 - Corrected vSphere parameters
BZ#800109 - Corrected RHEVM parameters

Revision 0-35 Tues Feb 28 2012 **Daniel Macpherson** *dmacpher@redhat.com*

Additions to Preface and Beta Subtitle.

Revision 0-34 Tues Feb 28 2012 **Daniel Macpherson** *dmacpher@redhat.com*

Additions to Preface and Beta Subtitle.

Revision 0-31 Wed Feb 22 2012 **Daniel Macpherson** ***dmacpher@redhat.com***

Revising installation instructions for Beta 6 release.

Revision 0-24 Wed Feb 8 2012 **Daniel Macpherson** ***dmacpher@redhat.com***

BZ#786590 - Added uninstallation instructions.

Revision 0-18 Wed Feb 1 2012 **Daniel Macpherson** ***dmacpher@redhat.com***

BZ#783880, BZ#783882, BZ#783883 - Changes for Documentation QE Review.

Revision 0-17 Tue Jan 31 2012 **Daniel Macpherson** ***dmacpher@redhat.com***

BZ#772304 - Added basic instructions for patching vSphere 5.0 Server environment for image pushing.
BZ#784032 - Included ISO Domain requirement.
BZ#783560 - Added basic instructions regarding **`floppyinject`** hook requirement for RHEV Hypervisors.

Revision 0-16 Mon Jan 30 2012 **Daniel Macpherson** ***dmacpher@redhat.com***

BZ#757684 - Added note in Appendix regarding RHEV export domain requirements for Cloud Engine.

Revision 0-13 Fri Jan 13 2012 **Daniel Macpherson** ***dmacpher@redhat.com***

Added additional configuration options

Revision 0-10 Thu Jan 12 2012 **Lana Brindley** ***lbrindle@redhat.com***

Changes for technical review by Chris Alfonso
Changes for technical review by Cliff Perry

Revision 0-9 Wed Jan 11 2012 **Lana Brindley** ***lbrindle@redhat.com***

Changes for technical review by Lukas Zapletal
Changes for technical review by James Laska

Revision 0-8 Fri Dec 09 2011 **Daniel Macpherson** ***dmacpher@redhat.com***

Building book for technical review

Revision 0-7 Wed Dec 07 2011 **Sarah Chua** ***sachua@redhat.com***

Added New Topics

Revision 0-2 Fri Dec 02 2011 **Sarah Chua** ***sachua@redhat.com***

Added New Topics

Revision 0-1 **Tue Nov 25 2011** **Shikha Nansi** *snansi@redhat.com*

Initial creation of book by publican

Revision 0-0 **Tue Nov 22 2011** **Dan Mcpherson** *dmacpher@redhat.com*

Initial creation of book by publican