



## Red Hat CloudForms 4.7

# Installing Red Hat CloudForms on Amazon Elastic Compute Cloud (Amazon EC2)

How to install and configure Red Hat CloudForms on Amazon Elastic Compute Cloud (Amazon EC2)



# Red Hat CloudForms 4.7 Installing Red Hat CloudForms on Amazon Elastic Compute Cloud (Amazon EC2)

---

How to install and configure Red Hat CloudForms on Amazon Elastic Compute Cloud (Amazon EC2)

Red Hat CloudForms Documentation Team  
cloudforms-docs@redhat.com

## Legal Notice

Copyright © 2019 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

XFS<sup>®</sup> is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack<sup>®</sup> Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## Abstract

This guide provides instructions on how to install and configure Red Hat CloudForms on Amazon EC2. If you have a suggestion for improving this guide or have found an error, please submit a Bugzilla report at <http://bugzilla.redhat.com> against Red Hat CloudForms Management Engine for the Documentation component. Please provide specific details, such as the section number, guide name, and CloudForms version so we can easily locate the content.

---

## Table of Contents

<b>CHAPTER 1. INSTALLING RED HAT CLOUDFORMS</b> .....	<b>3</b>
1.1. OBTAINING THE APPLIANCE	3
1.2. REQUIREMENTS	3
1.2.1. CloudForms Requirements	3
1.2.2. Amazon EC2 Requirements	3
1.3. UPLOADING THE APPLIANCE TO AN AMAZON S3 BUCKET	3
1.4. CONFIGURING AMAZON EC2 TO IMPORT THE APPLIANCE	4
1.5. IMPORTING THE APPLIANCE TO AMAZON ELASTIC COMPUTE CLOUD (AMAZON EC2)	5
<b>CHAPTER 2. CONFIGURING RED HAT CLOUDFORMS</b> .....	<b>7</b>
2.1. ACCESSING THE APPLIANCE CONSOLE	7
2.2. CONFIGURING A DATABASE	7
2.2.1. Configuring an Internal Database	7
2.2.2. Configuring an External Database	9
2.3. CONFIGURING A WORKER APPLIANCE	10
<b>APPENDIX A. APPENDIX</b> .....	<b>12</b>
A.1. APPLIANCE CONSOLE COMMAND-LINE INTERFACE (CLI)	12



# CHAPTER 1. INSTALLING RED HAT CLOUDFORMS

Installing Red Hat CloudForms consists of the following steps:

1. Downloading the appliance for your environment as a virtual machine image template.
2. Setting up a virtual machine based on the appliance.
3. Configuring the CloudForms appliance.

After you have completed all the procedures in this guide, you will have a working environment on which additional customizations and configurations can be performed.

## 1.1. OBTAINING THE APPLIANCE

1. Go to [access.redhat.com](https://access.redhat.com) and log in to the Red Hat Customer Portal using your customer account details.
2. Click **Downloads** in the menu bar.
3. Click **A-Z** to sort the product downloads alphabetically.
4. Click **Red Hat CloudForms** to access the product download page.
5. From the list of installers and images, click the **Download Now** link for **CFME EC2 Virtual Appliance**.

## 1.2. REQUIREMENTS

Below are the two sets of requirements for installing Red Hat CloudForms on Amazon EC2.

### 1.2.1. CloudForms Requirements

- 44 GB of space on the chosen datastore.
- 12 GB RAM.
- 4 vCPUs.

### 1.2.2. Amazon EC2 Requirements

- An Amazon S3 bucket to store disk images.
- A VM import service role (IAM role) named **vmimport**.

For information on creating an Amazon S3 bucket and a VM Import Service Role, see the [Amazon EC2 documentation](#).

## 1.3. UPLOADING THE APPLIANCE TO AN AMAZON S3 BUCKET

From your local file system, you can now upload the CloudForms appliance VHD image obtained in [Section 1.1, "Obtaining the Appliance"](#) to the Amazon S3 bucket, using your choice of tool.

## 1.4. CONFIGURING AMAZON EC2 TO IMPORT THE APPLIANCE



### IMPORTANT

These are the procedural steps as of the time of writing. For the latest information on importing a virtual machine as an image, see the [Amazon EC2 documentation](#).

1. Install the AWS client on the computer you want to interact with the AWS API from.

```
$ pip install awscli
```

2. Configure and download your AWS secret/access key by following the steps in the [Managing Access Keys for Your AWS Account](#) documentation.
3. Configure the AWS client with your access/secret key. For example:

```
$ aws configure
AWS Access Key ID [*****]: ACCESS_KEY
AWS Secret Access Key [*****]: SECRET_KEY
Default region name [None]:
Default output format [None]:
```

4. Create the **trust-policy.json** file for the **vmimport** role. For example:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": { "Service": "vmie.amazonaws.com" },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "sts:Externalid": "vmimport"
        }
      }
    }
  ]
}
```

5. Create the **vmimport** role using the **trust-policy.json** file that you just created.

```
$ aws iam create-role --role-name vmimport --assume-role-policy-document file://trust-policy.json
```



### NOTE

This user must have permissions to create and modify IAM roles.

6. Create the **role-policy.json** file. Be sure to use the exact image name that you uploaded to the S3 bucket. For example:



```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:ListBucket",
        "s3:GetBucketLocation",
        "s3:FullAccess"
      ],
      "Resource": [
        "arn:aws:s3:::NAME_OF_IMAGE_IN_S3_BUCKET"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetObject"
      ],
      "Resource": [
        "arn:aws:s3:::NAME_OF_IMAGE_IN_S3_BUCKET/*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:ModifySnapshotAttribute",
        "ec2:CopySnapshot",
        "ec2:RegisterImage",
        "ec2:Describe*",
        "ec2:FullAccess"
      ],
      "Resource": "*"
    }
  ]
}

```

7. Apply the **vmimport** role to the CloudForms appliance image you uploaded to the S3 bucket.

```

$ aws iam put-role-policy --role-name vmimport --policy-name vmimport --policy-document
file://role-policy.json

```

## 1.5. IMPORTING THE APPLIANCE TO AMAZON ELASTIC COMPUTE CLOUD (AMAZON EC2)

To import the appliance:

1. Create a `containers.json` file:

```

[
  {
    "Description": "NAME OF INSTANCE",
    "Format": "vhd",
    "UserBucket": {

```

```
"S3Bucket": "BUCKET WITH UPLOADED .VHD IMAGE",  
"S3Key": "NAME OF .VHD IMAGE"  
}  
}  
]
```

2. Use AWS-CLI tools to import the image. Run the following command:

```
$ aws ec2 import-image --disk-containers file://containers.json --region BUCKET_REGION
```

3. Check the progress of your imported image by running the following command:

```
$ aws ec2 describe-import-image-tasks --region BUCKET_REGION --import-task-ids  
import-ami-AMI_ID_GOT_FROM_RESPONSE
```

## CHAPTER 2. CONFIGURING RED HAT CLOUDFORMS

After installing CloudForms and running it for the first time, you must perform some basic configuration. To configure CloudForms, you must at a minimum:

1. Add a disk to the infrastructure hosting your appliance.
2. Configure the database.

Configure the CloudForms appliance using the internal appliance console.

### 2.1. ACCESSING THE APPLIANCE CONSOLE

1. Start the appliance and open a terminal console.
2. After starting the appliance, log in with a user name of **root** and the default password of **smartvm**. This displays the Bash prompt for the **root** user.
3. Enter the **appliance\_console** command. The Red Hat CloudForms appliance summary screen displays.
4. Press **Enter** to manually configure settings.
5. Press the number for the item you want to change, and press **Enter**. The options for your selection are displayed.
6. Follow the prompts to make the changes.
7. Press **Enter** to accept a setting where applicable.



#### NOTE

The CloudForms appliance console automatically logs out after five minutes of inactivity.

### 2.2. CONFIGURING A DATABASE

CloudForms uses a database to store information about the environment. Before using CloudForms, configure the database options for it; CloudForms provides the following two options for database configuration:

- Install an internal PostgreSQL database to the appliance
- Configure the appliance to use an external PostgreSQL database

#### 2.2.1. Configuring an Internal Database



#### IMPORTANT

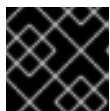
Before installing an internal database, add a disk to the infrastructure hosting your appliance. See the documentation specific to your infrastructure for instructions for adding a disk. As a storage disk usually cannot be added while a virtual machine is running, Red Hat recommends adding the disk before starting the appliance. Red Hat CloudForms only supports installing of an internal VMDB on blank disks; installation will fail if the disks are not blank.

1. Start the appliance and open a terminal console.
2. After starting the appliance, log in with a user name of **root** and the default password of **smartvm**. This displays the Bash prompt for the **root** user.
3. Enter the **appliance\_console** command. The Red Hat CloudForms appliance summary screen displays.
4. Press **Enter** to manually configure settings.
5. Select **5) Configure Database** from the menu.
6. You are prompted to create or fetch an encryption key.
  - If this is the first Red Hat CloudForms appliance, choose **1) Create key**.
  - If this is not the first Red Hat CloudForms appliance, choose **2) Fetch key from remote machine** to fetch the key from the first appliance. For worker and multi-region setups, use this option to copy key from another appliance.

**NOTE**

All CloudForms appliances in a multi-region deployment must use the same key.

7. Choose **1) Create Internal Database** for the database location.
8. Choose a disk for the database. This can be either a disk you attached previously, or a partition on the current disk.

**IMPORTANT**

Red Hat recommends using a separate disk for the database.

If there is an unpartitioned disk attached to the virtual machine, the dialog will show options similar to the following:

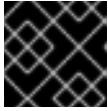
- ```

1) /dev/vdb: 20480
2) Don't partition the disk

```

- Enter **1** to choose **/dev/vdb** for the database location. This option creates a logical volume using this device and mounts the volume to the appliance in a location appropriate for storing the database. The default location is **/var/opt/rh/rh-postgresql95/lib/pgsql**, which can be found in the environment variable **\$APPLIANCE\_PG\_MOUNT\_POINT**.
  - Enter **2** to continue without partitioning the disk. A second prompt will confirm this choice. Selecting this option results in using the root filesystem for the data directory (not advised in most cases).
9. Enter **Y** or **N** for **Should this appliance run as a standalone database server?**
    - Select **Y** to configure the appliance as a database-only appliance. As a result, the appliance is configured as a basic PostgreSQL server, without a user interface.
    - Select **N** to configure the appliance with the full administrative user interface.

- When prompted, enter a unique number to create a new region.



### IMPORTANT

Creating a new region destroys any existing data on the chosen database.

- Create and confirm a password for the database.

Red Hat CloudForms then configures the internal database. This takes a few minutes. After the database is created and initialized, you can log in to CloudForms.

## 2.2.2. Configuring an External Database

Based on your setup, you will choose to configure the appliance to use an external PostgreSQL database. For example, we can only have one database in a single region. However, a region can be segmented into multiple zones, such as database zone, user interface zone, and reporting zone, where each zone provides a specific function. The appliances in these zones must be configured to use an external database.

The **postgresql.conf** file used with Red Hat CloudForms databases requires specific settings for correct operation. For example, it must correctly reclaim table space, control session timeouts, and format the PostgreSQL server log for improved system support. Due to these requirements, Red Hat recommends that external Red Hat CloudForms databases use a **postgresql.conf** file based on the standard file used by the Red Hat CloudForms appliance.

Ensure you configure the settings in the **postgresql.conf** to suit your system. For example, customize the **shared\_buffers** setting according to the amount of real storage available in the external system hosting the PostgreSQL instance. In addition, depending on the aggregate number of appliances expected to connect to the PostgreSQL instance, it may be necessary to alter the **max\_connections** setting.



### NOTE

- Red Hat CloudForms 4.x requires PostgreSQL version 9.5.
- Because the **postgresql.conf** file controls the operation of all databases managed by a single instance of PostgreSQL, do not mix Red Hat CloudForms databases with other types of databases in a single PostgreSQL instance.

- Start the appliance and open a terminal console.
- After starting the appliance, log in with a user name of **root** and the default password of **smartvm**. This displays the Bash prompt for the **root** user.
- Enter the **appliance\_console** command. The Red Hat CloudForms appliance summary screen displays.
- Press **Enter** to manually configure settings.
- Select **5) Configure Database** from the menu.
- You are prompted to create or fetch a security key.
  - If this is the first Red Hat CloudForms appliance, choose **1) Create key**.

- If this is not the first Red Hat CloudForms appliance, choose **2) Fetch key from remote machine** to fetch the key from the first appliance.



#### NOTE

All CloudForms appliances in a multi-region deployment must use the same key.

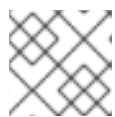
7. Choose **2) Create Region in External Database** for the database location.
8. Enter the database hostname or IP address when prompted.
9. Enter the database name or leave blank for the default (**vmdb\_production**).
10. Enter the database username or leave blank for the default (**root**).
11. Enter the chosen database user's password.
12. Confirm the configuration if prompted.

Red Hat CloudForms will then configure the external database.

## 2.3. CONFIGURING A WORKER APPLIANCE

You can use multiple appliances to facilitate horizontal scaling, as well as for dividing up work by roles. Accordingly, configure an appliance to handle work for one or many roles, with workers within the appliance carrying out the duties for which they are configured. You can configure a worker appliance through the terminal. The following steps demonstrate how to join a worker appliance to an appliance that already has a region configured with a database.

1. Start the appliance and open a terminal console.
2. After starting the appliance, log in with a user name of **root** and the default password of **smartvm**. This displays the Bash prompt for the **root** user.
3. Enter the **appliance\_console** command. The Red Hat CloudForms appliance summary screen displays.
4. Press **Enter** to manually configure settings.
5. Select **5) Configure Database** from the menu.
6. You are prompted to create or fetch a security key. Since this is not the first Red Hat CloudForms appliance, choose **2) Fetch key from remote machine**. For worker and multi-region setups, use this option to copy the security key from another appliance.



#### NOTE

All CloudForms appliances in a multi-region deployment must use the same key.

7. Choose **3) Join Region in External Database** for the database location.
8. Enter the database hostname or IP address when prompted.
9. Enter the port number or leave blank for the default (**5432**).

10. Enter the database name or leave blank for the default (**vmdb\_production**).
11. Enter the database username or leave blank for the default (**root**).
12. Enter the chosen database user's password.
13. Confirm the configuration if prompted.

Once Red Hat CloudForms is installed, you can log in and perform administration tasks.

Log in to Red Hat CloudForms for the first time after installing by:

1. Navigate to the URL for the login screen. (<https://xx.xx.xx.xx> on the virtual machine instance)
2. Enter the default credentials (Username: **admin** | Password: **smartvm**) for the initial login.
3. Click **Login**.

## APPENDIX A. APPENDIX

### A.1. APPLIANCE CONSOLE COMMAND-LINE INTERFACE (CLI)

Currently, the **appliance\_console\_cli** feature is a subset of the full functionality of the **appliance\_console** itself, and covers functions most likely to be scripted using the command-line interface (CLI).

1. After starting the Red Hat CloudForms appliance, log in with a user name of **root** and the default password of **smartvm**. This displays the Bash prompt for the root user.
2. Enter the **appliance\_console\_cli** or **appliance\_console\_cli --help** command to see a list of options available with the command, or simply enter **appliance\_console\_cli --option <argument>** directly to use a specific option.

Table A.1. Database Configuration Options

| Option          | Description                                                                                |
|-----------------|--------------------------------------------------------------------------------------------|
| --region (-r)   | region number (create a new region in the database - requires database credentials passed) |
| --internal (-i) | internal database (create a database on the current appliance)                             |
| --dbdisk        | database disk device path (for configuring an internal database)                           |
| --hostname (-h) | database hostname                                                                          |
| --port          | database port (defaults to <b>5432</b> )                                                   |
| --username (-U) | database username (defaults to <b>root</b> )                                               |
| --password (-p) | database password                                                                          |
| --dbname (-d)   | database name (defaults to <b>vmdb_production</b> )                                        |

Table A.2. v2\_key Options

| Option           | Description                                |
|------------------|--------------------------------------------|
| --key (-k)       | create a new v2_key                        |
| --fetch-key (-K) | fetch the v2_key from the given host       |
| --force-key (-f) | create or fetch the key even if one exists |



| Option        | Description                                                     |
|---------------|-----------------------------------------------------------------|
| --sshlogin    | ssh username for fetching the v2_key (defaults to <b>root</b> ) |
| --sshpassword | ssh password for fetching the v2_key                            |

Table A.3. IPA Server Options

| Option               | Description                                                                                      |
|----------------------|--------------------------------------------------------------------------------------------------|
| --host (-H)          | set the appliance hostname to the given name                                                     |
| --ipaserver (-e)     | IPA server FQDN                                                                                  |
| --ipaprincipal (-n)  | IPA server principal (default: <b>admin</b> )                                                    |
| --ipapassword (-w)   | IPA server password                                                                              |
| --ipadomain (-o)     | IPA server domain (optional). Will be based on the appliance domain name if not specified.       |
| --iparealm (-l)      | IPA server realm (optional). Will be based on the domain name of the ipaserver if not specified. |
| --uninstall-ipa (-u) | uninstall IPA client                                                                             |

**NOTE**

- In order to configure authentication through an IPA server, in addition to using **Configure External Authentication (httpd)** in the **appliance\_console**, external authentication can be optionally configured via the **appliance\_console\_cli** (command-line interface).
- Specifying **--host** will update the hostname of the appliance. If this step was already performed via the **appliance\_console** and the necessary updates made to **/etc/hosts** if DNS is not properly configured, the **--host** option can be omitted.

Table A.4. Certificate Options

| Option                      | Description                                        |
|-----------------------------|----------------------------------------------------|
| --ca (-c)                   | CA name used for certmonger (default: <b>ipa</b> ) |
| --postgres-client-cert (-g) | install certs for postgres client                  |
| --postgres-server-cert      | install certs for postgres server                  |

| Option                           | Description                                                                    |
|----------------------------------|--------------------------------------------------------------------------------|
| <code>--http-cert</code>         | install certs for http server (to create certs/httpd* values for a unique key) |
| <code>--extauth-opts (-x)</code> | external authentication options                                                |

**NOTE**

The certificate options augment the functionality of the **certmonger** tool and enable creating a certificate signing request (CSR), and specifying **certmonger** the directories to store the keys.

**Table A.5. Other Options**

| Option                      | Description                                                                               |
|-----------------------------|-------------------------------------------------------------------------------------------|
| <code>--logdisk (-l)</code> | log disk path                                                                             |
| <code>--tmpdisk</code>      | initialize the given device for temp storage (volume mounted at <b>/var/www/miq_tmp</b> ) |
| <code>--verbose (-v)</code> | print more debugging info                                                                 |

**Example Usage**

```
$ ssh root@appliance.test.company.com
```

To create a new database locally on the server using **/dev/sdb**:

```
# appliance_console_cli --internal --dbdisk /dev/sdb --region 0 --password smartvm
```

To copy the `v2_key` from a host *some.example.com* to local machine:

```
# appliance_console_cli --fetch-key some.example.com --sshlogin root --sshpassword smartvm
```

You could combine the two to join a region where *db.example.com* is the appliance hosting the database:

```
# appliance_console_cli --fetch-key db.example.com --sshlogin root --sshpassword smartvm --hostname db.example.com --password mydatabasepassword
```

To configure external authentication:

```
# appliance_console_cli --host appliance.test.company.com
    --ipaserver ipaserver.test.company.com
    --ipadomain test.company.com
```

```
--iprealm TEST.COMPANY.COM  
--ipprincipal admin  
--ipapassword smartvm1
```

To uninstall external authentication:

```
# appliance_console_cli --uninstall-ipa
```