



# Red Hat CloudForms

## 4.1

# Installing Red Hat CloudForms on Microsoft Azure

---

How to install and configure Red Hat CloudForms on a Microsoft Azure Cloud environment

Red Hat CloudForms Documentation  
Team



# Red Hat CloudForms 4.1 Installing Red Hat CloudForms on Microsoft Azure

---

How to install and configure Red Hat CloudForms on a Microsoft Azure Cloud environment

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

## Legal Notice

Copyright © 2017 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, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

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

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

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

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

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

The OpenStack ® 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 a Microsoft Azure Cloud environment. 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>1. INSTALLING RED HAT CLOUDFORMS</b> .....	<b>2</b>
1.1. Obtaining the Red Hat CloudForms Virtual Appliance	2
1.2. Uploading the Red Hat CloudForms Virtual Appliance to Microsoft Azure	2
<b>2. ENABLING CLOUDFORMS USER INTERFACE ACCESS</b> .....	<b>10</b>
<b>3. CONFIGURING RED HAT CLOUDFORMS</b> .....	<b>10</b>
3.1. Changing Configuration Settings	11
3.2. Advanced Configuration Settings	11
3.3. Configuring a Database for Red Hat CloudForms	12
3.4. Configuring a Worker Appliance	14
<b>4. LOGGING IN AFTER INSTALLING RED HAT CLOUDFORMS</b> .....	<b>15</b>
4.1. Changing the Default Login Password	15

## 1. INSTALLING RED HAT CLOUDFORMS

Red Hat CloudForms can be installed and ready to configure in a few quick steps. After downloading Red Hat CloudForms as a virtual machine image template from the Red Hat Customer Portal, the following process takes you through the steps of uploading the Red Hat CloudForms appliance to Microsoft Azure.



### Important

After uploading the Red Hat CloudForms appliance, you must configure the database for Red Hat CloudForms; see [Configuring a Database for Red Hat CloudForms](#).

### 1.1. Obtaining the Red Hat CloudForms Virtual 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** → **Download Latest** to access the product download page.
5. From the list of installers and images, select the Red Hat CloudForms appliance specified for Microsoft Azure download link.

### 1.2. Uploading the Red Hat CloudForms Virtual Appliance to Microsoft Azure

You can upload the appliance to an Azure environment using the following two methods.

- ✦ Using Azure PowerShell script
- ✦ Using Azure Command-Line Interface (Azure CLI)

To upload the Red Hat CloudForms appliance file to Microsoft Azure, ensure the following requirements are met:

- ✦ Approximately 2 GB of space for each VHD image; 44+ GB of space, 12 GB RAM, and 4 VCPUs for the Red Hat CloudForms appliance.
- ✦ [Microsoft Azure Account](#).
- ✦ Administrator access to the Azure portal.
- ✦ [Converting and Aligning the Red Hat CloudForms Virtual Appliance Image](#)
- ✦ Depending on your infrastructure, allow time for the upload.

## Important

Azure requires that the uploaded Virtual Hard Disk (VHD) files are in a fixed format. The CloudForms virtual appliance image .vhd file is dynamic by default. Currently, the Azure Powershell script and Azure CLI do not automatically convert the dynamic .vhd file to fixed during upload. To upload using either method, the CloudForms virtual appliance image .vhd file must be first converted from dynamic to fixed, and properly aligned to the nearest 1 MB boundary. Once converted and properly aligned, you can then upload the appliance virtual image .vhd file using either the Azure PowerShell or Azure CLI method.

### 1.2.1. Converting and Aligning the CloudForms Virtual Appliance Image

Complete the following procedure to ensure the CloudForms dynamic .vhd file is properly aligned to the nearest 1 MB boundary, and is in a fixed-size VHD format.

1. Convert the dynamic .vhd file you downloaded in [Section 1.1, “Obtaining the Red Hat CloudForms Virtual Appliance”](#) to RAW format.

```
$ qemu-img convert -f vpc -O raw <image-name.vhd> <image-name.raw>
```

Example:

```
$ qemu-img convert -f vpc -O raw cfme-azure-5.6.4.2-1.x86_64.vhd cfme-azure-5.6.4.2-1.x86_64.raw
```

2. Copy and paste the script below into a new bash shell script file, for example, **aligned-size.sh**. Change `rawdisk="image-name"` to the image name for your file. This script will calculate the rounded file size to the nearest 1 MB boundary.

```
#!/bin/bash
rawdisk="cfme-azure-5.6.4.2-1.x86_64.raw"
MB=$((1024 * 1024))
size=$(qemu-img info -f raw --output json "$rawdisk" | gawk 'match($0, /"virtual-size": ([0-9]+),/, val) {print val[1]}')
rounded_size=$((($size/$MB + 1) * $MB)
echo "rounded size = $rounded_size"
export rounded_size
```

3. Run the shell script. The file name **aligned-size.sh** is used in this example.

```
$ sh aligned-size.sh

rounded size = 34361835520
```

4. Resize the virtual appliance image using the rounded size.

```
$ qemu-img resize -f raw <image-name.raw> <rounded_size>
```

Example:

```
$ qemu-img resize -f raw cfme-azure-5.6.4.2-1.x86_64.raw 34361835520
```

Image resized.

5. Convert the appliance image to a fixed-size .vhd file.

```
$ qemu-img convert -f raw -o subformat=fixed -O vpc <image-name.raw>
<image-name.vhd>
```

Example:

```
qemu-img convert -f raw -o subformat=fixed -O vpc cfme-azure-5.6.4.2-
1.x86_64.raw cfme-azure-5.6.4.2-1.x86_64.vhd
```

6. Get the virtual size for the .vhd file.

```
$ qemu-img info --output=json -f vpc <path-to-image>
```

Example:

```
$ qemu-img info --output=json -f vpc cfme-azure-5.6.4.2-1.x86_64.vhd
{
  "virtual-size": 34361835520,
  "filename": "cfme-azure-5.6.4.2-1.x86_64.vhd",
  "cluster-size": 2097152,
  "format": "vpc",
  "actual-size": 2133401600,
  "dirty-flag": false
}
```

7. Divide the virtual-size value by 1024, twice. If the result is a whole number, the .vhd file is aligned properly. The example below shows that the file is properly aligned.

```
34361835520 / 1024 / 1024 = 32770
```

### Important

qemu-img version 1.5.3 is used in this procedure. Check the qemu-img version using the command: **yum info qemu-img**. If the version is 2.2.1 or later, add the option *force\_size* in the conversion command, for example, *subformat=fixed,force\_size*.

The Red Hat CloudForms Azure virtual appliance image is ready for uploading and provisioning in Microsoft Azure.

## 1.2.2. Uploading the CloudForms Virtual Appliance Using Azure Powershell Script

Complete the following steps to upload the CloudForms virtual appliance image you converted to a fixed-size VHD format and properly aligned per requirement using the procedure in [Section 1.2.1, "Converting and Aligning the CloudForms Virtual Appliance Image"](#).





## Note

Make sure Azure Resource Manager cmdlets are available. See **To install the cmdlets** section in [Azure Resource Manager Cmdlets](#).

1. Log in to **Azure Resource Manager** using the cmdlet:

```
## Customize for Your Environment
$SubscriptionName = "my subscription"

Login-AzureRmAccount
Select-AzureRmSubscription -SubscriptionName $SubscriptionName
```

When prompted, enter your user name and password for the Azure Portal.

2. Upload the .vhd file to a storage account. As shown in the example script below, you will first create a **Resource Group** through the Portal UI or Powershell. Additionally, create the storage container defined in "BlobDestinationContainer" in advance.

Example Script:

```
## Customize for Your Environment
$SubscriptionName = "my subscription"

$ResourceGroupName = "test"
$StorageAccountName = "test"

$BlobNameSource = "cfme-test.vhd"
$BlobSourceContainer = "templates"
$LocalImagePath = "C:\tmp\$BlobNameSource"

##

# Upload VHD to a "templates" directory. You can pass a few arguments,
such as `NumberOfUploaderThreads 8`. The default number of uploader
threads is `8`. See https://msdn.microsoft.com/en-
us/library/mt603554.aspx

Add-AzureRmVhd -ResourceGroupName $ResourceGroupName -Destination
https://$StorageAccountName.blob.core.windows.net/$BlobSourceContainer/$
BlobNameSource -LocalFilePath $LocalImagePath -NumberOfUploaderThreads 8
```

3. Create a virtual machine. Then, define your VM and VHD name, your system/deployment name and size. Next, you will set the appropriate Storage, Network and Configuration options for your environment.

Example Script:

```
## Customize for Your Environment

$BlobNameDest = "cfme-test.vhd"
$BlobDestinationContainer = "vhds"
$VMName = "cfme-test"
$DeploySize= "Standard_A3"
```

```

$vmUserName = "user1"

$InterfaceName = "test-nic"
$VNetName = "test-vnet"
$PublicIPName = "test-public-ip"

$SSHKey = <your ssh public key>

##

$StorageAccount = Get-AzureRmStorageAccount -ResourceGroup
$ResourceGroupName -Name $StorageAccountName

$SourceImageUri =
"https://$StorageAccountName.blob.core.windows.net/templates/$BlobNameSo
urce"
$Location = $StorageAccount.Location
$OSDiskName = $VMName

# Network
$Subnet1Name = "default"
$VNetAddressPrefix = "10.1.0.0/16"
$VNetSubnetAddressPrefix = "10.1.0.0/24"
$PIp = New-AzureRmPublicIpAddress -Name $PublicIPName -ResourceGroupName
$ResourceGroupName -Location $Location -AllocationMethod Dynamic -Force
$SubnetConfig = New-AzureRmVirtualNetworkSubnetConfig -Name $Subnet1Name
-AddressPrefix $VNetSubnetAddressPrefix
$VNet = New-AzureRmVirtualNetwork -Name $VNetName -ResourceGroupName
$ResourceGroupName -Location $Location -AddressPrefix $VNetAddressPrefix
-Subnet $SubnetConfig -Force
$Interface = New-AzureRmNetworkInterface -Name $InterfaceName -
ResourceGroupName $ResourceGroupName -Location $Location -SubnetId
$VNet.Subnets[0].Id -PublicIpAddressId $PIp.Id -Force

# Specify the VM Name and Size
$VirtualMachine = New-AzureRmVMConfig -VMName $VMName -VMSize
$DeploySize

# Add User
$cred = Get-Credential -UserName $VMUserName -Message "Setting user
credential - use blank password"
$VirtualMachine = Set-AzureRmVMOperatingSystem -VM $VirtualMachine -
Linux -ComputerName $VMName -Credential $cred

# Add NIC
$VirtualMachine = Add-AzureRmVMNetworkInterface -VM $VirtualMachine -Id
$Interface.Id

# Add Disk
$OSDiskUri = $StorageAccount.PrimaryEndpoints.Blob.ToString() +
$BlobDestinationContainer + "/" + $BlobNameDest

$VirtualMachine = Set-AzureRmVMOSDisk -VM $VirtualMachine -Name
$OSDiskName -VhdUri $OSDiskUri -CreateOption fromImage -SourceImageUri
$SourceImageUri -Linux

```

```
# Set SSH key
Add-AzureRmVMSShPublicKey -VM $VirtualMachine -Path
"/home/$VMUserName/.ssh/authorized_keys" -KeyData $SSHKey

# Create the VM
New-AzureRmVM -ResourceGroupName $ResourceGroupName -Location $Location
-VM $VirtualMachine
```

### Note

These are the procedural steps as of the time of writing. For more information, see the following Azure documentation.

- » <https://azure.microsoft.com/en-us/documentation/articles/powershell-azure-resource-manager>

The steps covered in the following article are for a Windows machine, however, most of the items are common between Windows and Linux.

- » <https://azure.microsoft.com/en-us/documentation/articles/virtual-machines-windows-create-powershell>

### 1.2.3. Uploading and Provisioning the CloudForms Virtual Appliance Using Azure Command-Line Interface

You can upload the appliance to an Azure environment using the Azure Command-Line Interface (Azure CLI) following the process below.

#### Install Azure CLI 2.0:

Complete the steps below to install Azure CLI 2.0 using curl. See <https://docs.microsoft.com/en-us/cli/azure/install-azure-cli> for other installation methods.

1. Make sure Python is updated and install the prerequisite packages.

```
$ sudo yum update
$ sudo yum install -y gcc libffi-devel python-devel openssl-devel
```

2. Install Azure CLI 2.0.

```
$ curl -L https://aka.ms/InstallAzureCli | bash
```

#### Upload and Provision the CloudForms Virtual Appliance Using Azure CLI:

Complete the following steps to upload and provision the CloudForms virtual appliance you converted to a fixed-size VHD format and properly aligned per requirement using the procedure in [Section 1.2.1, “Converting and Aligning the CloudForms Virtual Appliance Image”](#).

1. Upload the image to the storage container. It may take several minutes. Note: Enter **az storage container list** to get the list of storage containers.

```
$ az storage blob upload --account-name <storage-account-name> --
```

```
container-name <container-name> --type page --file <path-to-vhd> --name
<image-name>.vhd
```

Example:

```
$ az storage blob upload --account-name azrhelclistact --container-name
azrhelclistcont --type page --file cfme-azure-5.6.4.2-1.x86_64.vhd --
name cfme-azure-5.6.4.2-1.x86_64.vhd
```

```
Percent complete: %100.0
```

2. Get the URL for the uploaded .vhd file using the following command. You will need to use this URL in the next step.

```
$ az storage blob url -c <container-name> -n <image-name>.vhd
```

Example:

```
$ az storage blob url -c azrhelclistcont -n cfme-azure-5.6.4.2-
1.x86_64.vhd
```

```
"https://azrhelclistact.blob.core.windows.net/azrhelclistcont/cfme-
azure-5.6.4.2-1.x86_64.vhd"
```

3. Create the virtual machine. Note that the following command uses **--generate-ssh-keys**. In this example, the private/public key pair **/home/clouduser/.ssh/id\_rsa** and **/home/clouduser/.ssh/id\_rsa.pub** are created.

```
$ az vm create --resource-group <resource-group> --location <azure-
region> --use-unmanaged-disk --name <vm-name> --storage-account
<storage-account-name> --os-type linux --admin-username <administrator-
name> --generate-ssh-keys --image <URL>
```

Example:

```
az vm create --resource-group azrhelclirgrp --location southcentralus -
-use-unmanaged-disk --name cfme-appliance-1 --storage-account
azrhelclistact --os-type linux --admin-username clouduser --generate-
ssh-keys --image
https://azrhelclistact.blob.core.windows.net/azrhelclistcont/cfme-azure-
5.6.4.2-1.x86_64.vhd
```

```
{
  "fqdns": "",
  "id":
"/subscriptions//resourceGroups/azrhelclirgrp/providers/Microsoft.Compu
te/virtualMachines/cfme-appliance-1",
  "location": "southcentralus",
  "macAddress": "",
  "powerState": "VM running",
  "privateIpAddress": "10.0.0.4",
  "publicIpAddress": "12.84.121.147",
  "resourceGroup": "azrhelclirgrp"
}
```

Make a note of the public IP address. You will need this to log in to the virtual machine in the next step.

4. Start an SSH session and log in to the appliance.

```
$ ssh -i <path-to-ssh-key> <admin-username@public-IP-address>
```

Example:

```
$ ssh -i /home/clouduser/.ssh/id_rsa clouduser@12.84.121.147
The authenticity of host '12.84.121.147' can't be established.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '12.84.121.147' (ECDSA) to the list of known
hosts.
```

```
Welcome to the Appliance Console
```

```
For a menu, please type: appliance_console
```

5. Enter **sudo appliance\_console** at the prompt. The summary screen appears.

You have successfully provisioned a CloudForms virtual appliance in Microsoft Azure.

## Note

The exported storage connection string does not persist after a system reboot. If any of the commands in the above steps fail, export the storage connection string again using the following commands:

1. Get the storage account connection string.

```
$ az storage account show-connection-string -n <storage-account-name> -g <resource-group>
```

Example:

```
$ az storage account show-connection-string -n azrhelclistact -g azrhelclirgrp
{
  "connectionString":
  "DefaultEndpointsProtocol=https;EndpointSuffix=core.windows.net;AccountName=azrhelclistact;AccountKey=NreGk...=="
}
```

2. Export the connection string. Copy the connection string and paste it into the following command. This connects your system to the storage account.

```
$ export AZURE_STORAGE_CONNECTION_STRING="<storage-connection-string>"
```

Example:

```
$ export
AZURE_STORAGE_CONNECTION_STRING="DefaultEndpointsProtocol=https;EndpointSuffix=core.windows.net;AccountName=azrhelclistact;AccountKey=NreGk...=="
```

## 2. ENABLING CLOUDFORMS USER INTERFACE ACCESS

To access the Red Hat CloudForms virtual appliance user interface, you need to enable access over ports **80** and **443** to the virtual machine. You can do this using the CLI or from within the Azure portal.

- ✦ To enable a port using the CLI, enter **az vm open-port --port <port-number> --resource-group <resource-group> --name <vm-name>**.
- ✦ To enable a port using the Microsoft Azure portal, open the properties for the resource group where the appliance is located, click on **Network Security Group**, and add HTTP and HTTPS access.

## 3. CONFIGURING RED HAT CLOUDFORMS

Although the Red Hat CloudForms appliance comes preconfigured to be integrated immediately into your environment, you can make some changes to its configuration as needed.

### 3.1. Changing Configuration Settings

1. Log in to the appliance using the SSH key.
2. Enter the **sudo appliance\_console** command. The Red Hat CloudForms appliance summary screen displays.
3. Press **Enter** to manually configure settings.
4. Press the number for the item you want to change, and press **Enter**. The options for your selection are displayed.
5. Follow the prompts to make the changes.
6. Press **Enter** to accept a setting where applicable.



#### Note

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

### 3.2. Advanced Configuration Settings

You can use the following options for advanced configuration of the appliance:

- ✦ Use **Set DHCP Network Configuration** to use DHCP to obtain the IP address and network configuration for your Red Hat CloudForms appliance. The appliance is initially configured as a DHCP client with bridged networking.
- ✦ Use **Set Static Network Configuration** if you have a specific IP address and network setting you need to use for the Red Hat CloudForms appliance.
- ✦ Use **Test Network Configuration** to check that name resolution is working correctly.
- ✦ Use **Set Hostname** to specify a hostname for the Red Hat CloudForms appliance.



#### Important

A valid fully qualified hostname for the Red Hat CloudForms appliance is required for SmartState analysis to work correctly,

- ✦ Use **Set Timezone, Date, and Time** to configure the time zone, date, and time for the Red Hat CloudForms appliance.
- ✦ Use **Restore Database from Backup** to restore the VMDB database from a previous backup.
- ✦ Use **Setup Database Region** to create regions for VMDB replication.
- ✦ Use **Configure Database** to configure the VMDB database. Use this option to configure the database for the appliance after installing and running it for the first time.
- ✦ Use **Extend Temporary Storage** to add temporary storage to the appliance. The appliance formats an unpartitioned disk attached to the appliance host and mounts it at `/var/www/miq_tmp`. The appliance uses this temporary storage directory to perform certain image download functions.

- ✦ Use **Configure External Authentication (httpd)** to configure authentication through an IPA server.
- ✦ Use **Generate Custom Encryption Key** to regenerate the encryption key used to encode plain text password.
- ✦ Use **Harden Appliance Using SCAP Configuration** to apply Security Content Automation Protocol (SCAP) standards to the appliance. You can view these SCAP rules in the `/var/www/miq/lib/appliance_console/config/scap_rules.yml` file.
- ✦ Use **Stop Server Processes** to stop all server processes. You may need to do this to perform maintenance.
- ✦ Use **Start Server Processes** to start the server. You may need to do this after performing maintenance.
- ✦ Use **Restart Appliance** to restart the Red Hat CloudForms appliance. You can either restart the appliance and clear the logs or just restart the appliance.
- ✦ Use **Shut Down Appliance** to power down the appliance and exit all processes.
- ✦ Use **Summary Information** to go back to the network summary screen for the Red Hat CloudForms appliance.
- ✦ Use **Quit** to leave the Red Hat CloudForms appliance console.

### 3.3. Configuring a Database for Red Hat CloudForms

Red Hat CloudForms uses a database to store information about the cloud environment it manages. You must configure a database for the appliance before you can use it to administer your cloud infrastructure. Red Hat CloudForms provides the following two options for database configuration:

- ✦ Configuring an internal PostgreSQL database
- ✦ Configuring an external PostgreSQL database

#### 3.3.1. Configuring an Internal Database



##### Important

Before installing an internal database, add a disk to the infrastructure hosting your appliance. See the storage documentation specific to your infrastructure for instructions on how to add 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. The installation will fail if the disks are not blank.

1. Start the appliance and open a terminal console.
2. Log in to the appliance using the SSH key.
3. Enter the **sudo appliance\_console** command. The Red Hat CloudForms appliance summary screen displays.
4. Press **Enter** to manually configure settings.
5. Select **8) 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 Red Hat CloudForms appliance. All Red Hat CloudForms appliances in a multi-region deployment must use the same key.

7. Choose **1) Internal** for the database location.

8. Choose a disk for the database. For example:

```
1) /dev/vdb: 20480
```

Choose disk:

Enter **1** to choose **/dev/vdb** for the database location.

9. When prompted, enter a unique three digit region ID to create a new region. As your deployment grows, you can add more regions in the future to manage multiple appliances.



### Important

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

10. Confirm the configuration when prompted.

Red Hat CloudForms configures the internal database.

### 3.3.2. 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 where each zone provides specific functionality such as, Database, User Interface, Reporting among others. The appliances in these zones must be configured to use an external database.

Note that 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 9.4.
- ✦ 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.

1. Start the appliance and open a terminal console.
2. Log in to the appliance using the SSH key.
3. Enter the **sudo appliance\_console** command. The Red Hat CloudForms appliance summary screen displays.
4. Press **Enter** to manually configure settings.
5. Select **8) Configure Database** from the menu.
6. You are prompted to create or fetch a security key.
  - ✦ If this is the first Red Hat CloudForms appliance, select the option to create a key.
  - ✦ If this is not the first Red Hat CloudForms appliance, select the option to fetch the key from the first Red Hat CloudForms appliance. All Red Hat CloudForms appliances in a multi-region deployment must use the same key.
7. Choose **2) External** 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 (**vldb\_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.

### 3.4. 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 up the appliance and open a terminal console.
2. Log in to the appliance using the SSH key.
3. Enter the **sudo appliance\_console** command. The Red Hat CloudForms appliance summary screen displays.
4. Press **Enter** to manually configure settings.
5. Select **8) Configure Database** from the menu.
6. You are prompted to create or fetch a security key. Select the option to fetch the key from the first Red Hat CloudForms appliance. All Red Hat CloudForms appliances in a multi-region deployment must use the same key.
7. Choose **2) External** 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 password.
12. Confirm the configuration if prompted.

## 4. LOGGING IN AFTER INSTALLING RED HAT CLOUDFORMS

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**.

### 4.1. Changing the Default Login Password

Change your password to ensure more private and secure access to Red Hat CloudForms.

1. Navigate to the URL for the login screen. (<https://xx.xx.xx.xx> on the virtual machine instance)
2. Click **Update Password** beneath the **Username** and **Password** text fields.
3. Enter your current **Username** and **Password** in the text fields.
4. Input a new password in the **New Password** field.
5. Repeat your new password in the **Verify Password** field.
6. Click **Login**.