

# **oVirt Server Suite 0**

## **Installation Guide**

**The comprehensive guide to the oVirt Server Suite installation**

The logo for oVirt, featuring the word "oVirt" in a dark grey, sans-serif font. The letter "i" is lowercase and has a small green dot above it, while the other letters are uppercase.

**Chris Curran**

# **oVirt Server Suite 0 Installation Guide**

## **The comprehensive guide to the oVirt Server Suite installation**

### **Edition 0.95**

Author

Chris Curran

[ccurran@redhat.com](mailto:ccurran@redhat.com)

Copyright © 2008 Red Hat, Inc.

Copyright © 2008 Red Hat, Inc.. This material may only be distributed subject to the terms and conditions set forth in the Open Publication License, V1.0 or later. The latest version of the OPL is presently available at <http://www.opencontent.org/openpub/>.

Red Hat and the Red Hat "Shadow Man" logo are registered trademarks of Red Hat, Inc. in the United States and other countries.

All other trademarks referenced herein are the property of their respective owners.

The GPG fingerprint of the security@redhat.com key is:

CA 20 86 86 2B D6 9D FC 65 F6 EC C4 21 91 80 CD DB 42 A6 0E

1801 Varsity Drive  
Raleigh, NC 27606-2072

USA

Phone: +1 919 754 3700

Phone: 888 733 4281

Fax: +1 919 754 3701

PO Box 13588

Research Triangle Park, NC 27709

USA

The oVirt Server Suite Installation Guide provides detailed instructions for deploying the oVirt Server Suite. The book covers varying hardware and software configurations and installation environments from stand-alone developer installations to enterprise deployments of hundreds of nodes.

---

<b>Preface</b>	<b>v</b>
1. Document Conventions .....	vi
1.1. Typographic Conventions .....	vi
1.2. Pull-quote Conventions .....	viii
1.3. Notes and Warnings .....	viii
2. We Need Feedback! .....	ix
<b>1. Introduction</b>	<b>1</b>
<b>2. System requirements and support</b>	<b>3</b>
2.1. Requirements .....	3
2.2. Requirements for the stand-alone oVirt Appliance .....	3
2.2.1. Preparing to Install the Developer Version .....	3
2.3. Prerequisites .....	4
<b>3. Stand-alone oVirt Appliance installation</b>	<b>5</b>
3.1. Installing oVirt Appliance .....	5
3.1.1. Installing the Appliance from the oVirt website .....	5
3.2. Installing oVirt on a Physical Network .....	6
3.2.1. Setting up the Hardware .....	6
3.3. Installing the Appliance .....	7
3.4. About create-wui-appliance.sh .....	8
<b>4. oVirt Server Suite Production Installation</b>	<b>9</b>
<b>5. oVirt Server Suite advanced installation topics</b>	<b>11</b>
5.1. FreeIPA .....	11
5.2. PostgreSQL .....	11
5.3. Other Relational Database systems .....	13
5.4. Network service configuration - DNS and DHCP .....	13
5.5. PXE .....	13
5.6. Advanced PXE with Cobbler .....	13
5.6.1. Deploying virtual machines with Cobbler .....	13
5.7. Storage topics .....	13
5.7.1. NFS .....	13
5.7.2. iSCSI .....	13
5.7.3. GFS .....	13
5.8. oVirt Server Suite advanced installation .....	13
<b>6. Deploying oVirt Managed Nodes</b>	<b>15</b>
6.1. OEM hardware with pre-installed managed nodes .....	15
6.2. PXE network installation .....	15
6.3. Installing with locally attached portable storage .....	15
6.4. Key Distribution? .....	15
6.5. Multiple node deployment tips and tricks .....	15
<b>7. Upgrading oVirt</b>	<b>17</b>
7.1. Upgrading the oVirt Server .....	17
7.2. Upgrading managed nodes .....	17
<b>A. Revision History</b>	<b>19</b>



---

# Preface

oVirt is an open, cross-platform virtualization management system. oVirt provides a small image that runs on a host and provides virtualization services to virtual machines, and a web-based management interface that lets you manage hosts and storage, install and remove virtual machines, level resources across a large group of machines. oVirt scales from a small group of users with little need for access control and quota management, all the way up to hundreds or even thousands of hosts with robust control over grouping, permissions, and quotas.

## About oVirt Administration User Interface

oVirt Administration User Interface is a web-based management UI application for oVirt hosts, users, and virtual machines.

## About Using the oVirt Administration User Interface

The guide is provided as online help and as html pages. It describes how to perform a developer install of oVirt and how to use the interface to manage virtual hosts, users and virtual machines.

## The oVirt Documentation Suite

oVirt documentation suite provides information on installation, development of applications, and all the latest changes.

- oVirt Release Notes - this document contains release specific information and known issues with the beta release.
- Online Help - this document describes how to use the oVirt Administration User Interface.
- Developer Installation Guide - this document describes how to install a developer version of oVirt correctly and quickly.
- Production Installation Guide - this document describes how to install oVirt correctly and quickly. A production install of oVirt allows a user to manage physical hosts.
- oVirt Configuration Guide - this document describes how to configure oVirt
- Appliance Development Kit User's Guide - this document describes the oVirt ADK.

## About the Audience

This documentation suite is intended for

- Anyone who wants to manage virtual machines! oVirt is lightweight enough to work for a developer managing, say, a single host with four virtual machines, yet robust enough for a large organization managing tens of thousands of virtual machines. Although oVirt is developed using Fedora, oVirt is not tied to a particular platform. Solaris and Windows users can use the oVirt browser interface, and in the future the oVirt Administration Interface will run across platforms as well.
- Open Source Developers who are prototyping and experimenting with oVirt. Important aspects are an easy installation of the system and the ability to make changes in order to build a community of users and developers.

- Users at the enterprise level: Hardware Administrators, team administrators and virtual machine users at large or small IT facilities.
  - Typically, hardware administrators can use oVirt to manage thousands of Virtual Machines running on thousands of Host systems, often with complex user roles involved.
- Team administrators manage user quotas, permissions, and subgroups for team hardware, Monitors physical hosts, Maintains available team virtual machines, install images/appliances
- oVirt users create, destroy, and generally use virtual machines available to them after connecting to virtual machines from a browser interface.

## 1. Document Conventions

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

In PDF and paper editions, this manual uses typefaces drawn from the *Liberation Fonts*<sup>1</sup> set. The Liberation Fonts set is also used in HTML editions if the set is installed on your system. If not, alternative but equivalent typefaces are displayed. Note: Red Hat Enterprise Linux 5 and later includes the Liberation Fonts set by default.

### 1.1. Typographic Conventions

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

#### **Mono-spaced Bold**

Used to highlight system input, including shell commands, file names and paths. Also used to highlight key caps 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 key cap, all presented in Mono-spaced Bold and all distinguishable thanks to context.

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

Press **Enter** to execute the command.

Press **Ctrl+Alt+F1** to switch to the first virtual terminal. Press **Ctrl+Alt+F7** to return to your X-Windows session.

The first sentence highlights the particular key cap to press. The second highlights two sets of three key caps, 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:

---

<sup>1</sup> <https://fedorahosted.org/liberation-fonts/>

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; dialogue box text; labelled 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.

Note the `>` shorthand used to indicate traversal through a menu and its sub-menus. This is to avoid the difficult-to-follow 'Select **Mouse** from the **Preferences** sub-menu in the **System** menu of the main menu bar' approach.

### *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:

When the Apache HTTP Server accepts requests, it dispatches child processes or threads to handle them. This group of child processes or threads is known as

a *server-pool*. Under Apache HTTP Server 2.0, the responsibility for creating and maintaining these server-pools has been abstracted to a group of modules called *Multi-Processing Modules (MPMs)*. Unlike other modules, only one module from the MPM group can be loaded by the Apache HTTP Server.

## 1.2. Pull-quote Conventions

Two, commonly multi-line, data types 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 are presented and highlighted as follows:

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

import javax.naming.InitialContext;

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

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

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

## 1.3. Notes and Warnings

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



### Note

A note is a tip or shortcut or alternative approach 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 Important boxes won't cause data loss but may cause irritation and frustration.



### Warning

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

## 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/bugzilla/> against the product **oVirt**.

When submitting a bug report, be sure to mention the manual's identifier: *oVirt\_Installation\_Guide*

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

This section will be an introduction.

---

# System requirements and support

This is a test paragraph

## 2.1. Requirements

Test of a section

## 2.2. Requirements for the stand-alone oVirt Appliance

The developer version has the following prerequisites:

- A machine that supports hardware virtualization. This machine is called a host machine throughout the rest of the document.
- Fedora 9 must be installed on the host machine with the latest versions of the following:
  - kernel version  $\geq$  2.6.24
  - kvm version  $\geq$  60
  - libvirt  $\geq$  0.4.1-3
  - virt-manager
  - virt-viewer
- The Fedora 9 repository install tree or boot CD or DVD.

### 2.2.1. Preparing to Install the Developer Version

The following steps prepare your host machine for a successful installation of the oVirt Administration User Interface.

- Check if the host machine supports full hardware virtualization. If you have an Intel processor use this command:

```
# cat /proc/cpuinfo | grep vmx
```

or if you have an AMD processor use this command:

```
# cat /proc/cpuinfo | grep svm
```

- Check if the correct bare metal kernel is available.

```
# uname -r  
2.6.24.14107.fc9
```

- Check if the `kvm`, `libvirt` and `virt-manager` packages are installed on the host machine.

```
# rpm -q libvirt kvm virt-manager
```

- Check if the host system has `kvm`, and `kvm_intel` or `kvm_amd` modules are loaded. For example,

```
# lsmod | grep kvm
kvm_intel 29677 0
kvm      88609 1 kvm_intel
```

### 2.3. Prerequisites

There are a number of prerequisites for the bundled install:

- The management host machine must support hardware virtualization and have at least two network interfaces.
- The managed nodes are one or more additional machines that support hardware virtualization. These machines are the physical machines on which virtual machines run.
- Fedora 9 must be installed on the host machine with the following:
  - kernel version 2.6.24 newer
  - kvm version 60 or newer
  - libvirt 0.4.1-3 or newer
  - virt-manager
  - virt-viewer
- The Fedora 9 repository install tree or boot CD or DVD.

# Stand-alone oVirt Appliance installation

This chapter covers the steps required to get the oVirt Server Suite Appliance installed and running on a stand-alone machine.

There are two types of oVirt Server Suite installation:

- Stand-alone oVirt Appliance. The stand-alone oVirt Appliance runs from a single machine, it can be expanded later and is suitable for testing and development.
- oVirt Appliance with Managed Nodes. The oVirt Appliance with Managed Nodes is made up of the oVirt Appliance and dedicated, Managed Nodes. This installation setup fully utilizes the potential of oVirt in providing manageability and scalability. Managed Nodes require additional hardware. For information on installing the oVirt Appliance with Managed Nodes refer to [Chapter 5, oVirt Server Suite advanced installation topics](#).

The advanced installation topics chapter([Chapter 5, oVirt Server Suite advanced installation topics](#)) also covers how to integrate individual oVirt components.

## 3.1. Installing oVirt Appliance

This section describes how to install the stand-alone version of the oVirt Appliance. Installing the developer version is the fastest way to get oVirt up and running because it is intended to help you to start using oVirt and become familiar with the user interface. The developer version of the oVirt appliance cannot be used to manage physical hosts. To manage physical hosts, you must install the bundled version of the appliance, using the instructions in [Section 3.2, “Installing oVirt on a Physical Network”](#).

### 3.1.1. Installing the Appliance from the oVirt website

This section describes how to install the developer version using the pre-built appliance provided on the oVirt website download page.

1. Download the [appliance](#)<sup>1</sup> script onto the host machine.
2. Extract the image to `/var/lib/libvirt/images/`:

```
# cd /var/lib/libvirt/images; tar jxf ovirt-developer-  
appliance-0.91-1-*.tar;  
# mv ovirt-developer-appliance-*/developer.img /var/lib/libvirt/images
```

3. Run `create-wui-appliance.sh` to create the virtual machine.

```
# bash ovirt-developer-appliance-*/create-wui-appliance.sh -v
```

4. Remove the list of known hosts:

```
#rm -Rf ~/.ssh/known_hosts
```

5. Start the virtual machine named `developer`.

```
virsh start developer
```

6. Start up some fake managed nodes. These are additional KVM guests on the same host machine, but they act as if they were separate physical nodes. You can use up to 3 fake managed nodes, named node3, node4, and node5. To start the nodes:

```
# virsh start node3
# virsh start node4
```

7. Invoke the main oVirt page by running Firefox on the guest.

```
# ssh -Y root@192.168.50.2 firefox
```

Enter the root password: ovirt

A browser window opens displaying the oVirt User Administration Interface.

8. To connect to the console of a virtual machine running on a managed node:

```
# ssh -Y root@192.168.50.2 virt-viewer -c qemu+tcp://
node3.priv.ovirt.org/system VM_NAME
```



### Note

The kickstart can be copied to a local webserver and modified so that the repos referenced in the kickstart point to local mirrors. In this case, the command is:

```
# create-wui-appliance.sh -t http://foo/f9-x86_64 -k http://
foo/wui-rel-x86_64.ks -v
```

## 3.2. Installing oVirt on a Physical Network

Set up the bundled oVirt appliance if you wish to manage physical hosts. This version allows you to manage physical hardware. It requires a minimum of 2 machines and a separately cabled network.



### Note

You must be the root user to perform the installation.

### 3.2.1. Setting up the Hardware

This section describes how to set up the hardware for a simple bundled oVirt installation.

1. Set up the management node to enable it to run the management console as a KVM guest, and take over the second network card (eth1) for a private oVirt network.

2. Set up the managed nodes. Connect the managed nodes to the same network as the second network interface on the management host.
3. Set up the private network. This step is fully automated by `create-wui-appliance.sh`.
4. Set up storage.

### 3.3. Installing the Appliance

This section describes how to install the pre-built appliance from the oVirt website download page to manage physical hardware.

1. Download the [appliance<sup>2</sup>](#) script onto the host machine.
2. Extract the image to `/var/lib/libvirt/images/`:

```
# cd /var/lib/libvirt/images; tar jxf ovirt-developer-
appliance-0.91-1-*.tar; \
mv ovirt-developer-appliance-*/developer.img /var/lib/libvirt/images
```

3. Run `create-wui-appliance.sh` to create the virtual machine.

```
# bash ovirt-developer-appliance-*/create-wui-appliance.sh -b
```

4. Remove the list of known hosts:

```
#rm -Rf ~/.ssh/known_hosts
```

5. Start the virtual machine named **developer**.

```
virsh start developer
```

6. Start up some managed nodes.

```
# virsh start node3
# virsh start node4
```

7. Invoke the main oVirt page by running Firefox on the guest.

```
# ssh -Y root@192.168.50.2 firefox
```

Enter the root password: `ovirt`

A browser window opens displaying the oVirt User Administration Interface.

8. To connect to the console of a virtual machine running on a managed node:

```
# ssh -Y root@192.168.50.2 virt-viewer -c qemu+tcp://
node3.priv.ovirt.org/system VM_NAME
```



**Note**

The kickstart can be copied to a local webserver and modified so that the repos referenced in the kickstart point to local mirrors. In this case, the command is:

```
# create-wui-appliance.sh -t http://foo/f9-x86_64 -k http://foo/wui-rel-x86_64.ks -b
```

### 3.4. About create-wui-appliance.sh

create-wui-appliance is the pre-built appliance that is available from the oVirt website download page. Use this appliance to install either version of oVirt.

If you have not already done so, you can download the [appliance](#)<sup>3</sup> onto the host machine, and follow the instructions in either [Section 3.1, “Installing oVirt Appliance”](#) or [Section 3.2, “Installing oVirt on a Physical Network”](#) to install the appliance.

Usage	
<code>create-wui-appliance.sh -i install_iso   -t install_tree [-d image_dir] [-a x86_64 i686]</code>	
Parameter	Description
-i	Location of installation ISO (required if -t not present)
-t	Location of installation tree (required if -i not present)
-k	URL of kickstart file for use with installation tree
-d	Directory to place virtual disk (default: /var/lib/libvirt/images)
-a	Architecture for the virtual machine (default: x86_64)
-v	Install in developer mode
-b	Install in bundled mode
-h	Display this help and exit

Table 3.1. create-wui-appliance.sh Parameters

<sup>3</sup> [http://ovirt.org/download/ovirt-developer-appliance-0.91-1-x86\\_64.tar](http://ovirt.org/download/ovirt-developer-appliance-0.91-1-x86_64.tar)

# oVirt Server Suite Production Installation

This chapter covers the step by step installation of the default oVirt Server Suite with managed physical nodes.

## oVirt Server Suite production installation procedure

There are several RPMs that need to be installed on the host machine that are provided in a yum repository hosted on the oVirt website. Because this process updates libvirt and you'll need to restart libvirt at the end, shut down any existing virtual machines before proceeding. To get your host set up properly, run the following commands: Install the ovirt-release RPM to get yum configured:

```
# rpm -ivh http://ovirt.org/repos/ovirt/9/ovirt-release-LATEST.fc9.noarch.rpm
```

Update any existing RPMs with newer RPMs available in the oVirt repository:

```
# yum update --enablerepo=ovirt
```

Install additional RPMs from the oVirt repository:

```
# yum install --enablerepo=ovirt ovirt-appliance ovirt-docs ovirt-node virt-viewer-plugin
```

NOTE: The ovirt-appliance RPM is 424MB and will take some time to download. 4. Restart libvirtd to pick up the newer version:

```
# service libvirtd restart
```

After installation has finished, the guest should be rebooted to perform first-run setup. At the end of the boot process, you will see a message saying "Starting ovirt-dev-wui-first-run:". This command will take a while to finish, and is completely normal.

At this point, your appliance should be ready. Now, you can bring up the main oVirt screen by running firefox on the guest (here you'll need the root password, which is ovirt):

```
# ssh -fY root@192.168.50.2 firefox -no-remote
```

To connect to the console of a virtual machine running on a managed node use:

```
# ssh -fY root@192.168.50.2 virt-viewer -c qemu+tcp://node3.priv.ovirt.org/system VM_NAME
```



# oVirt Server Suite advanced installation topics

This chapter covers the step by step installation procedure for installing the oVirt Appliance for production environments with separate services for databases, PXE booting and networking. This type of installation allows administrators to build more robust oVirt Server Suite installations. Installation procedure for installing the oVirt Appliance for production environments with separate services for databases, PXE booting and networking. This type of installation allows administrators to build more robust and flexible oVirt Server Suite deployments. Configuring the individual services and applications required by oVirt is described in [Chapter 5, oVirt Server Suite advanced installation topics](#). Links to particular sections will be provided throughout this chapter. description - what it is and why how to set up the component - could be just links to other documentation or ports of existing documentation what oVirt needs from the component - if the component is set up what does oVirt need from it

## 5.1. FreeIPA

This section covers integrating oVirt with FreeIPA.

## 5.2. PostgreSQL

This section covers integrating oVirt with the **PostgreSQL** Relational Database. PostgreSQL is the default Relational Database used with the oVirt Server Suite. PostgreSQL is a powerful, widely used and tested, open source Relational Database.

The oVirt Server Suite only supports PostgreSQL as the database at this time.

[Preparing PostgreSQL for oVirt](#) details preparing a postgres database for the oVirt Appliance. Complete this procedure *before* installing oVirt if you require a network connected database on a separate server. The oVirt Server Suite requires PostgreSQL version 8.2 or higher. Version 8.2 is recommended.

### Procedure 5.1. Preparing PostgreSQL for oVirt

1. Install PostgreSQL and the ruby connector and server packages:

```
$ sudo yum install ruby-postgres postgresql-server
```

2. The database must be initialized before you can create tables. If the Postgres database is already initialized skip this step.

```
$ sudo /sbin/service postgresql initdb
```

3. Start the service.

```
$ sudo /sbin/service postgresql start
```

- It is recommended to add the database to the boot sequence as well. Use the `chkconfig` command to add **postgres** to the **initscripts**. `35` in the example specifies which run levels should start the application, modify this number to set postgres to start on your servers default run level.

```
sudo /sbin/chkconfig --level 35 postgres on
```

- Change to the `postgres` user to add the oVirt tables.

```
$ sudo su - postgres
```

- In the postgres shell, create a new database for the oVirt Server Suite and open that database.

```
$ createdb ovirt
$ psql ovirt
```

- Create a user, `ovirt`, replacing `password` with a strong password. Then create tables for `ovirt` and `ovirt_test` with privileges for the user `ovirt`. Exit after creating the tables.

```
$-# CREATE USER ovirt WITH PASSWORD 'password';
$-# CREATE DATABASE ovirt;
$-# GRANT ALL PRIVILEGES ON DATABASE ovirt to ovirt;
$-# CREATE DATABASE ovirt_test;
$-# GRANT ALL PRIVILEGES ON DATABASE ovirt_test to ovirt;
$-# \q
$ exit
```

- Enable client authentication for networked connections. Open `/var/lib/pgsql/data/pg_hba.conf` and append the line in the example, replacing the IP addresses with valid IP addresses for systems which will have the oVirt Server Suite installed.

```
host all all 10.0.0.1 trust
```

- Enable TCP/IP connections to your postgres database by editing the PostgreSQL configuration file `/var/lib/pgsql/data/postgresql.conf` on the system containing the oVirt database. Locate the line `"listen_addresses='localhost'"`. Replace the `localhost` address with a list of valid addresses of oVirt servers. For example if oVirt Server Suite was installed on `10.0.0.1` and `10.0.0.2` then the modified entry would be `listen_addresses='10.0.0.1 10.0.0.2'`. Specifying which IP address clients can connect to the database is considered a good security precaution.

- Verify the connection is open to external Postgres connections on the specified addresses.
- Restart the PostgreSQL service and continue with other advanced installation topic or commence the installation.

```
$ sudo /sbin/service postgres restart
Stopping postgresql service:          [ OK ]
Starting postgresql service:         [ OK ]
```

## Configuring the oVirt Appliance the PostgreSQL database

TBD

## 5.3. Other Relational Database systems

This section covers integrating oVirt with other relational databases.

## 5.4. Network service configuration - DNS and DHCP

This section covers integrating oVirt with existing DNS and DHCP services.

## 5.5. PXE

The Preboot Execution Environment(PXE) is an environment for booting computers using only a network interface independent of local storage.

## 5.6. Advanced PXE with Cobbler

This section covers integrating oVirt with **Cobbler** and `tftpd` for advanced network management and deployment of managed nodes.

### 5.6.1. Deploying virtual machines with Cobbler

This section covers deploying virtual machines with **Cobbler** to oVirt Nodes. Notes: The cobbler server is on the appliance.

## 5.7. Storage topics

para

### 5.7.1. NFS

para

### 5.7.2. iSCSI

covers iSCSI, fake and real.

### 5.7.3. GFS

GFS for guests or shared storage.

## 5.8. oVirt Server Suite advanced installation

After all the services are configured, install. This covers installing oVirt after all the additional services are configured.



# Deploying oVirt Managed Nodes

This chapter covers deploying oVirt managed nodes.

## 6.1. OEM hardware with pre-installed managed nodes

This section covers OEM hardware with pre-installed managed nodes.

## 6.2. PXE network installation

This section covers how to configure PXE boot servers and clients as a method for deploying managed nodes.

### Configuring the PXE Boot Server(Cobbler)

This section describes using a Cobbler based PXE boot server in your oVirt Server Suite environment.

### Configuring client nodes for PXE booting

This section covers how to configure nodes for PXE booting.

## 6.3. Installing with locally attached portable storage

(USB drives and CD-ROMs)

## 6.4. Key Distribution?

Using a secure network with autoget key via wget

Using a USB key

## 6.5. Multiple node deployment tips and tricks

This section covers tips and tricks for deploying large numbers of managed nodes.



# Upgrading oVirt

The oVirt project is still in active development. Features and bug fixes will be released in new versions and your organization may decided to upgrade to a newer version of oVirt. This chapter covers the procedures for upgrading oVirt servers and managed nodes.

## 7.1. Upgrading the oVirt Server

This section covers the procedures for upgrading oVirt servers.

## 7.2. Upgrading managed nodes

This section covers the procedures for upgrading oVirt managed nodes.



---

# Appendix A. Revision History

Revision History  
Revision 0.1  
First version

ChrisCurran [ccurran@redhat.com](mailto:ccurran@redhat.com)

