

Red Hat Enterprise Linux 6

V2V Guide

Importing Virtual Machines with virt-v2v



David Jorm

Tim Hildred

Laura Bailey

Red Hat Enterprise Linux 6 V2V Guide

Importing Virtual Machines with virt-v2v

Edition 1

Author	David Jorm	djorm@redhat.com
Author	Tim Hildred	thildred@redhat.com
Author	Laura Bailey	lbailey@redhat.com

Copyright © 2011 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, MetaMatrix, 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.

All other trademarks are the property of their respective owners.

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

This book is a guide to importing virtual machines from foreign hypervisors to Red Hat Enterprise Virtualization and KVM managed by libvirt.

Preface	v
1. About this Guide	v
1.1. Documentation Suite	v
1.2. Audience	vi
2. Document Conventions	vi
2.1. Typographic Conventions	vi
2.2. Pull-quote Conventions	vii
2.3. Notes and Warnings	viii
3. We Need Feedback!	viii
1. Introducing V2V	1
1.1. Installing virt-v2v	1
2. Converting Virtual Machines to Run on KVM Managed by libvirt	3
2.1. Converting a Virtual Machine	3
2.1.1. Preparing to Convert a Virtual Machine	3
2.1.2. Converting Virtual Machines	9
2.1.3. Running Converted Virtual Machines	11
3. Converting Virtual Machines to Run on Red Hat Enterprise Virtualization	13
3.1. Acceptable Converted Storage Output Formats	13
3.2. Attaching an Export Storage Domain	14
3.3. Converting a Virtual Machine	16
3.3.1. Preparing to Convert a Virtual Machine	16
3.3.2. Converting a Virtual Machine	20
3.3.3. Importing and Running the Converted Virtual Machine	23
3.3.4. Scripting the v2v Process	24
3.3.5. Scripted Bulk v2v Process	24
4. Debugging and Troubleshooting	27
4.1. Debugging V2V conversions	27
5. References	29
5.1. virt-v2v Parameters	29
5.2. Configuration Changes	30
5.2.1. Configuration Changes for Linux Virtual Machines	30
5.2.2. Configuration Changes for Windows Virtual Machines	31
A. Revision History	33

Preface

The Red Hat Enterprise Virtualization platform is a richly featured virtualization management solution providing fully integrated management across virtual machines. It is based on the leading open source virtualization platform and provides superior technical capabilities. The platform offers scalability in the management of large numbers of virtual machines.

1. About this Guide

This guide describes how to import virtual machines from foreign hypervisors to Red Hat Enterprise Virtualization and KVM managed by libvirt.

1.1. Documentation Suite

The Red Hat Enterprise Virtualization documentation suite provides information on installation, development of applications, configuration and usage of the Red Hat Enterprise Virtualization platform and its related products.

- *Red Hat Enterprise Virtualization — Administration Guide* describes how to setup, configure and manage Red Hat Enterprise Virtualization. It assumes that you have successfully installed the Red Hat Enterprise Virtualization manager and hosts.
- *Red Hat Enterprise Virtualization — Evaluation Guide* enables prospective customers to evaluate the features of Red Hat Enterprise Virtualization. Use this guide if you have an evaluation license.
- *Red Hat Enterprise Virtualization — Installation Guide* describes the installation prerequisites and procedures. Read this if you need to install Red Hat Enterprise Virtualization. The installation of hosts, manager and storage are covered in this guide. You will need to refer to the *Red Hat Enterprise Virtualization Administration Guide* to configure the system before you can start using the platform.
- *Red Hat Enterprise Virtualization — Manager Release Notes* contain release specific information for Red Hat Enterprise Virtualization Managers.
- *Red Hat Enterprise Virtualization — Power User Portal Guide* describes how power users can create and manage virtual machines from the Red Hat Enterprise Virtualization user portal.
- *Red Hat Enterprise Virtualization — Quick Start Guide* provides quick and simple instructions for first time users to set up a basic Red Hat Enterprise Virtualization environment.
- *Red Hat Enterprise Virtualization — REST API Guide* describes how to use the REST API to set up and manage virtualization tasks. Use this guide if you wish to develop systems which integrate with Red Hat Enterprise Virtualization, using an open and platform independent API.
- *Red Hat Enterprise Virtualization — Technical Reference Guide* describes the technical architecture of Red Hat Enterprise Virtualization and its interactions with existing infrastructure.
- *Red Hat Enterprise Virtualization — User Portal Guide* describes how users of the Red Hat Enterprise Virtualization system can access and use virtual desktops from the user portal.
- *Red Hat Enterprise Linux — Hypervisor Deployment Guide* describes how to deploy and install the hypervisor. Read this guide if you need advanced information about installing and deploying Hypervisors. The basic installation of Hypervisor hosts is also described in the *Red Hat Enterprise Virtualization Installation Guide*.

- *Red Hat Enterprise Linux — V2V Guide* (the book you are reading) describes importing virtual machines from KVM, Xen and VMware ESX to Red Hat Enterprise Virtualization and KVM managed by libvirt.

1.2. Audience

This guide is intended for system administrators who manage a virtualized environment using Red Hat Enterprise Virtualization or Red Hat Enterprise Linux. An advanced level of system administration, preferably including familiarity with virtual machine data center operations, is assumed. This document is not intended for beginners.

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.

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.

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

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. 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 **Red Hat Enterprise Linux**.

When submitting a bug report, be sure to mention the manual's identifier: *doc-V2V_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, include the section number and some of the surrounding text so we can find it easily.

Introducing V2V

V2V is an acronym for virtual to virtual, referring to the process of importing virtual machines from one virtualization platform to another. Red Hat Enterprise Virtualization and Red Hat Enterprise Linux are capable of performing V2V operations using the **virt-v2v** command.

virt-v2v

The **virt-v2v** command converts virtual machines from a foreign hypervisor to run on KVM, managed by Red Hat Enterprise Virtualization or libvirt. **virt-v2v** can currently convert Red Hat Enterprise Linux 4, Red Hat Enterprise Linux 5, Red Hat Enterprise Linux 6, Windows XP, Windows Vista, Windows 7, Windows Server 2003 and Windows Server 2008 virtual machines running on Xen, KVM and VMware ESX. **virt-v2v** enables para-virtualized (**virtio**) drivers in the converted virtual machine if possible.

The following guest operating systems are supported by **virt-v2v**:

- Red Hat Enterprise Linux 4
- Red Hat Enterprise Linux 5
- Red Hat Enterprise Linux 6
- Windows XP
- Windows Vista
- Windows 7
- Windows Server 2003
- Windows Server 2008

The following source hypervisors are supported by **virt-v2v**:

- Xen
- KVM
- VMware ESX

1.1. Installing virt-v2v

virt-v2v is run from a Red Hat Enterprise Linux host. It must be installed on the host.

Procedure 1.1. Installing virt-v2v

1. Subscribe to RHN channel

virt-v2v is available on Red Hat Network (RHN) in the **Red Hat Enterprise Linux Server (v.6 for 64-bit x86_64)** or **Red Hat Enterprise Linux Workstation (v.6 for x86_64)** channel. Ensure the system is subscribed to the appropriate channel before installing **virt-v2v**.

2. Install pre-requisites

If you are converting Windows virtual machines, you must install the *libguestfs-winsupport* and *virtio-win* packages. These packages provide support for NTFS and Windows para-virtualized block and network drivers. If you attempt to convert a virtual machine using NTFS without the *libguestfs-winsupport* package installed, the conversion will fail. If you attempt to convert a virtual machine running Windows without the *virtio-win* package installed, the conversion will fail giving an error message concerning missing files.

The *libguestfs-winsupport* is available for RHEL Server 6 in the **Red Hat Enterprise Linux Server V2V Tools for Windows (v. 6)** channel, while the *virtio-win* package is available in the **Red Hat Enterprise Linux Server Supplementary (v. 6)** channel. To install these packages, ensure that your system has the required permissions to subscribe to both channels and run the following command as root:

```
rhn-channel -a rhel-x86_64-server-supplementary-6 --user USERNAME --password PASSWORD
```

```
rhn-channel -a rhel-x86_64-server-v2vwin-6 --user USERNAME --password PASSWORD
```

```
yum install libguestfs-winsupport virtio-win
```

3. Install **virt-v2v** package

As root, run the command:

```
yum install virt-v2v
```

4. Run **virt-v2v** as the root user from a Linux shell.

Converting Virtual Machines to Run on KVM Managed by libvirt

virt-v2v can convert virtual machines to run on Red Hat Enterprise Linux, using KVM managed by libvirt. Virtual machines can be converted from Xen, KVM and VMware ESX environments.

2.1. Converting a Virtual Machine

virt-v2v converts virtual machines from a foreign hypervisor to run on Red Hat Enterprise Linux, using KVM managed by libvirt. It automatically creates a libvirt domain for the converted virtual machines.

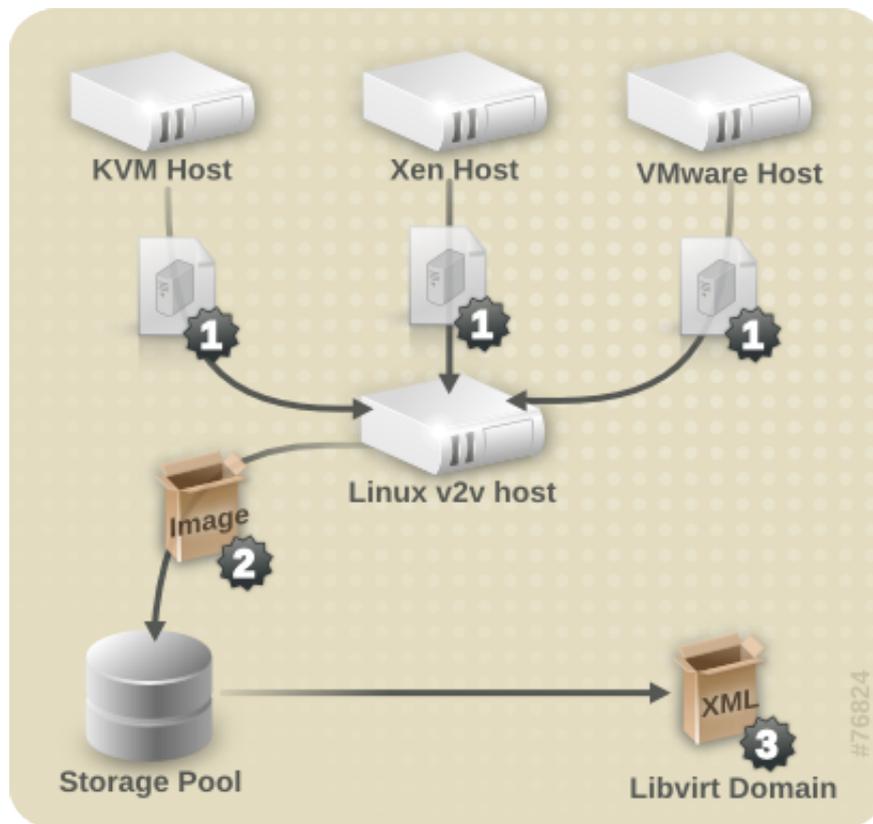


Figure 2.1. Converting a virtual machine

2.1.1. Preparing to Convert a Virtual Machine

Before a virtual machine can be converted, ensure that the following steps are completed.

1. Create a local storage domain for transferred storage

virt-v2v copies the guest storage to a locally defined libvirt storage pool during import. This pool can be defined using any libvirt tool, and can be of any type. The simplest way to create a new pool is with **virt-manager**. Select your host, right click and select details.

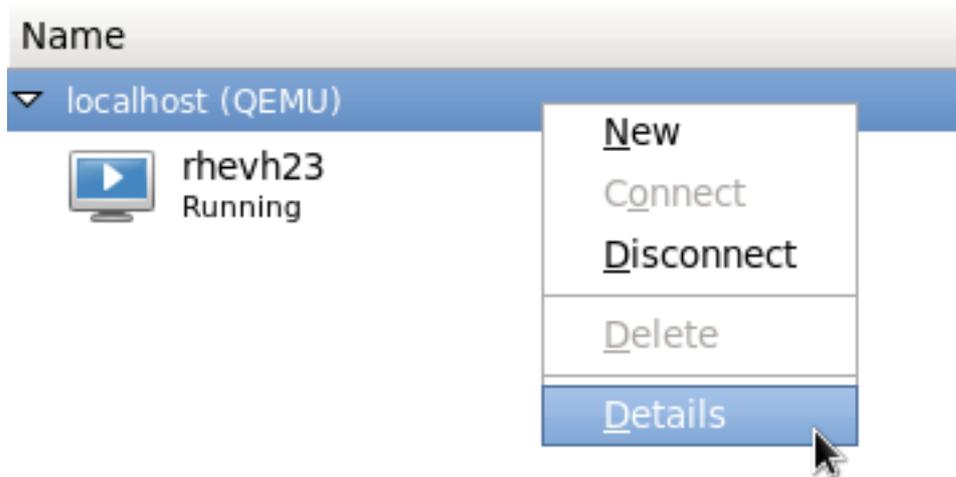


Figure 2.2. Select host details

Select the **Storage** tab.

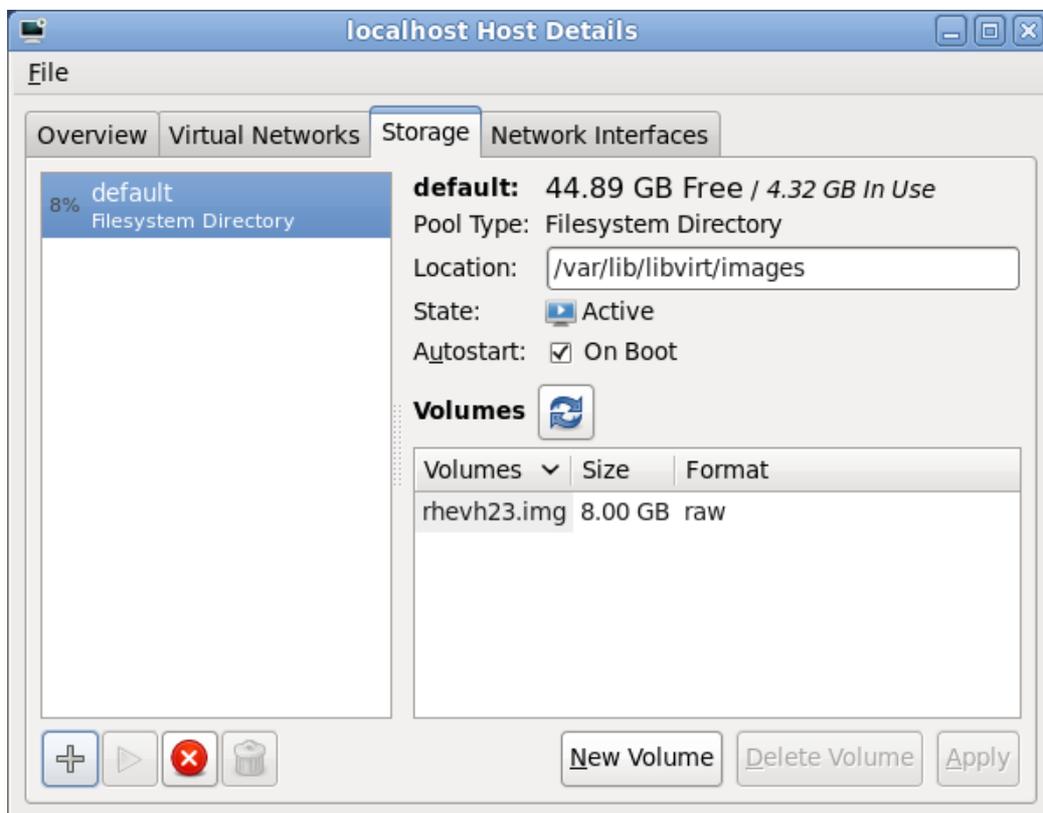


Figure 2.3. The storage tab

Click the plus sign (+) button to add a new storage pool.

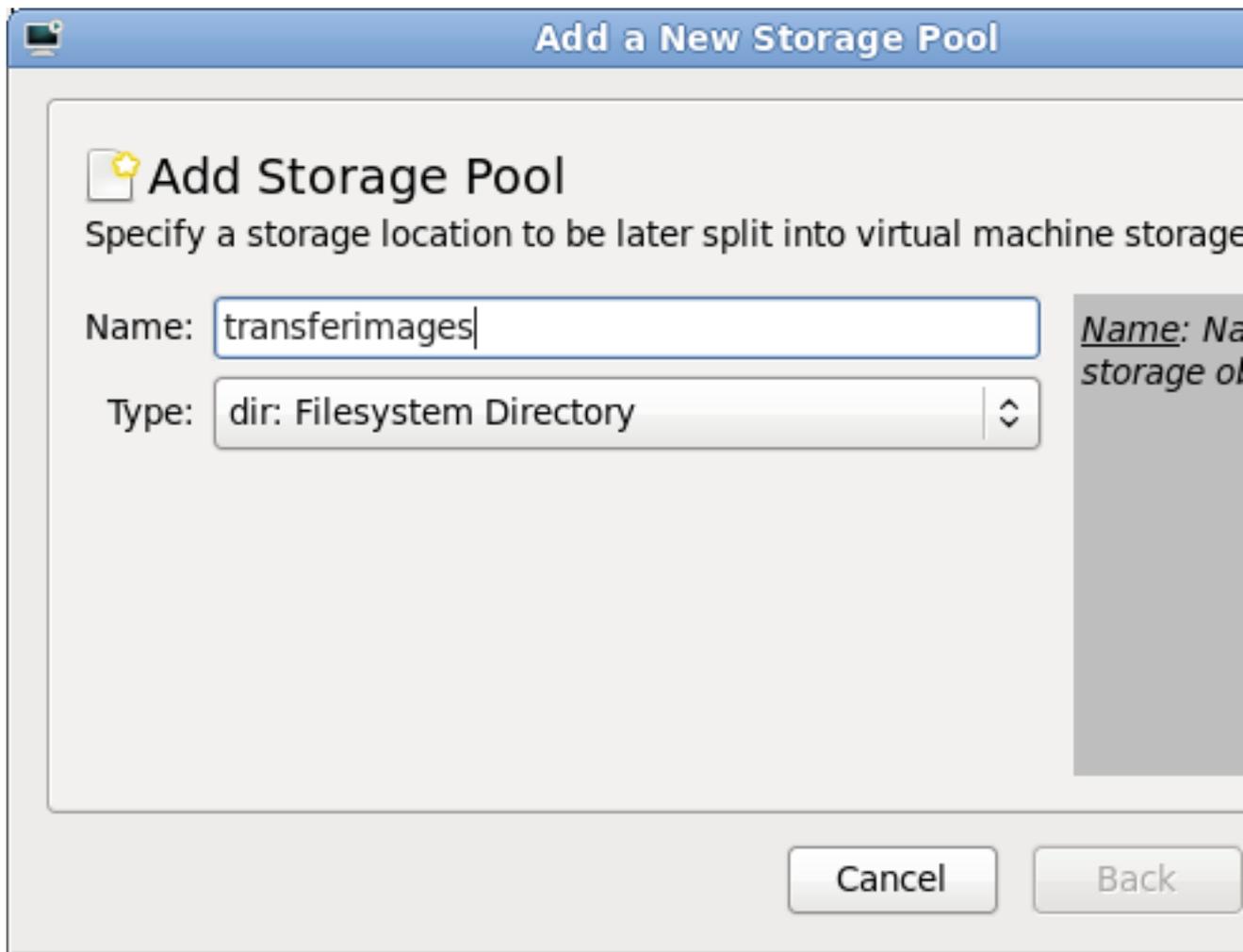


Figure 2.4. Adding a storage pool - step 1

Enter the name for the new storage pool, and select the type of storage target to use.

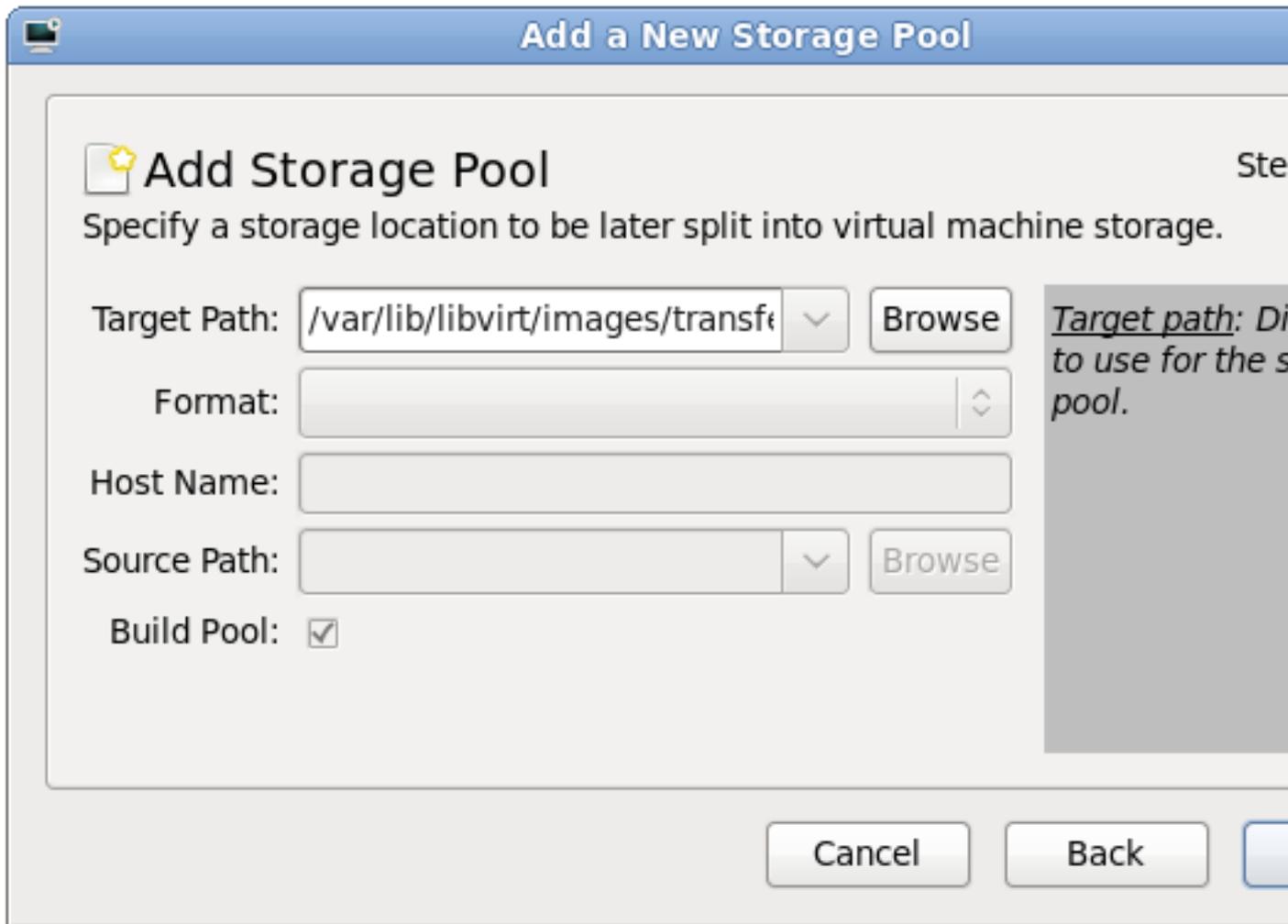


Figure 2.5. Adding a storage pool - step 2

Specify the path to the storage target, along with any type-specific details.

2. Create local network interfaces.

The local machine must have an appropriate network to which the converted virtual machine can connect. This is likely to be a bridge interface. A bridge interface can be created using standard tools on the host. **virt-manager** can also create and manage bridges. For more information on bridged networking with libvirt, see the *Red Hat Enterprise Linux Virtualization Guide*.

3. Specify network mappings in **virt-v2v.conf**. This step is *optional*, and is not required for most use cases.

If your virtual machine has multiple network interfaces, **/etc/virt-v2v.conf** must be edited to specify the network mapping for all interfaces. You can specify an alternative **virt-v2v.conf** file with the **-f** parameter.

If your virtual machine only has a single network interface, it is simpler to use the **--network** or **--bridge** parameters, rather than modifying **virt-v2v.conf**.

4. Create a profile for the conversion in **virt-v2v.conf**. This step is *optional*. Profiles specify a conversion method, storage location, output format and allocation policy. When a profile is defined, it can be called using **--profile** rather than individually providing the **-o**, **-os**, **-of** and **-oa** parameters. See **virt-v2v.conf(5)** for details.

2.1.1.1. Preparing to convert a virtual machine running Linux

Before a virtual machine running Linux can be converted, ensure that the following steps are completed.

1. Obtain the software

As part of the conversion process, **virt-v2v** may install a new kernel and drivers on the virtual machine. If the virtual machine being converted is registered to Red Hat Network (RHN), the required packages will be automatically downloaded. If the virtual machine is not registered to RHN, the **virt-v2v.db** file ships with a list of RPMs used for this purpose. The RPMs relevant to your virtual machine must be downloaded manually from RHN and made available on the host running **virt-v2v**. The RPMs should be saved in the directory specified by the *path-root* configuration element, which by default is `/var/lib/virt-v2v/software/`. An error similar to [Example 2.1, “Missing Package error”](#) will be displayed by **virt-v2v** if software it depends upon for a particular conversion is not available.

Example 2.1. Missing Package error

```
virt-v2v: Installation failed because the following files referenced in the
configuration file are required, but missing:
rhel/6/kernel-2.6.32-128.el6.x86_64.rpm
rhel/6/ecryptfs-utils-82-6.el6.x86_64.rpm
rhel/6/ecryptfs-utils-82-6.el6.i686.rpm
```

To obtain the relevant RPMs for your environment, repeat these steps for each missing package:

1. Login to [Red Hat Network](#)¹
2. Select the **Package Search** tab.
3. In the **Search For** field, type the package name exactly matching the one shown in the error message. For the example shown in [Example 2.1, “Missing Package error”](#), the first package is **kernel-2.6.32-128.el6.x86_64**
4. In the **Where to search** field, select **In the following architectures** and tick the **x86_64** checkbox. Click **Search**.
5. A list of packages displays. Click the package name identical to the one in the error message.
6. You will be directed to the Details page, containing detailed descriptions of the package. Select **Download Package** at the bottom of the page
7. Save the downloaded package to the appropriate directory in `/var/lib/virt-v2v/software`. For Red Hat Enterprise Linux 6, the directory is `/var/lib/virt-v2v/software/rhel/6`

2.1.1.2. Preparing to convert a virtual machine running Windows

Before a virtual machine running Windows can be converted, ensure that the following steps are completed.

1. Install the *libguestfs-winsupport* package on the host running **virt-v2v**. This package provides support for NTFS, which is used by many Windows systems. The *libguestfs-winsupport* package is provided by the **RHEL V2VWIN (v. 6 for 64-bit x86_64)** channel. Ensure your system is subscribed to this channel, then run the following command as root:

```
yum install libguestfs-winsupport
```

If you attempt to convert a virtual machine using NTFS without the *libguestfs-winsupport* package installed, the conversion will fail. An error message similar to [Example 2.2, “Error message when converting a Windows virtual machine without libguestfs-winsupport installed”](#) will be shown.

Example 2.2. Error message when converting a Windows virtual machine without libguestfs-winsupport installed

```
No operating system could be detected inside this disk image.
```

```
This may be because the file is not a disk image, or is not a virtual machine image, or because the OS type is not understood by virt-inspector.
```

```
If you feel this is an error, please file a bug report including as much information about the disk image as possible.
```

2. Install the *virtio-win* package on the host running **virt-v2v**. This package provides para-virtualized block and network drivers for Windows guests. The *virtio-win* package is provided by the **RHEL Server Supplementary (v. 6 64-bit x86_64)** channel. Ensure your system is subscribed to this channel, then run the following command as root:

```
yum install virtio-win
```

If you attempt to convert a virtual machine running Windows without the *virtio-win* package installed, the conversion will fail. An error message similar to [Example 2.3, “Error message when converting a Windows virtual machine without virtio-win installed”](#) will be shown.

Example 2.3. Error message when converting a Windows virtual machine without virtio-win installed

```
virt-v2v: Installation failed because the following files referenced in the configuration file are required, but missing: /usr/share/virtio-win/drivers/i386/Win2008
```



Post-processing for Windows virtual machines

When virtual machines running Windows are converted for output to Red Hat Enterprise Virtualization, post-processing of the virtual machine image will be performed by the Red Hat Enterprise Virtualization Manager to install updated drivers. See [Section 5.2.2, “Configuration Changes for Windows Virtual Machines”](#) for details of the process. This step will be omitted when virtual machines running Windows are converted for output to libvirt.

2.1.1.3. Preparing to convert a local Xen virtual machine

The following is required when converting virtual machine on a host which used to run Xen, but has been updated to run KVM. It is not required when converting a Xen guest imported directly from a running libvirt/Xen instance.

1. Obtain the XML for the virtual machine

virt-v2v uses a libvirt domain description to determine the current configuration of the virtual machine, including the location of its storage. Before starting the conversion, obtain this from the host running the virtual machine with the following command:

```
virsh dumpxml vm-name > vm-name.xml
```

This will require booting into a Xen kernel to obtain the XML, as libvirt needs to connect to a running Xen hypervisor to obtain its metadata. The conversion process is optimized for KVM, so obtaining domain data while running a Xen kernel, then performing the conversion using a KVM kernel will be more efficient than running the conversion on a Xen kernel.

2.1.2. Converting Virtual Machines

Once you have prepared to convert the virtual machines, use **virt-v2v** to perform the actual conversions. This section provides the steps to convert the virtual machines, and command syntax for **virt-v2v**. Note that conversions are resource intensive processes, involving copying the whole disk image for a virtual machine. In typical environments, converting a single virtual machine takes approximately 5-10 minutes.

2.1.2.1. virt-v2v

virt-v2v converts guests from a foreign hypervisor to run on KVM, managed by libvirt. The general command syntax for converting machines to run on KVM managed by libvirt is:

```
virt-v2v -i libvirtxml -os pool --bridge brname vm-name.xml
virt-v2v -ic xen+ssh://root@vmhost.example.com -os pool --bridge brname vm-name
virt-v2v -ic esx://esx.example.com/?no_verify=1 -os pool --bridge brname vm-name
```

A full specification of the parameters which can be used with **virt-v2v** is available in [Section 5.1, "virt-v2v Parameters"](#).

2.1.2.2. Converting a local Xen virtual machine

Ensure that the virtual machine's XML is available locally, and that the storage referred to in the XML is available locally at the same paths.

To convert the virtual machine from an XML file, run:

```
virt-v2v -i libvirtxml -os pool --bridge brname vm-name.xml
```

Where **pool** is the local storage pool to hold the image, **brname** is the name of a local network bridge to connect the converted virtual machine's network to, and **vm-name.xml** is the path to the virtual machine's exported XML. You may also use the `--network` parameter to connect to a locally managed network, or specify multiple mappings in `/etc/virt-v2v.conf`.

If your guest uses a Xen para-virtualized kernel (it would be called something like kernel-xen or kernel-xenU), **virt-v2v** will attempt to install a new kernel during the conversion process. You can avoid this requirement by installing a regular kernel, which won't reference a hypervisor in its name, alongside the Xen kernel prior to conversion. You should not make this newly installed kernel your default kernel, because Xen will not boot it. **virt-v2v** will make it the default during conversion.

2.1.2.3. Converting a remote Xen virtual machine



Setup SSH Keys Prior to Converting a Remote VM with Multiple

Because each disk transfer requires a new SSH session, it is recommended that SSH keys be set up prior to the conversion for authentication. This is especially important for large disks. Otherwise, a user will be required to manually enter SSH credentials for each disk being transferred. Failure to do so before the SSH negotiation times out will cause virt-v2v to fail.

Xen virtual machines can be converted remotely via SSH. Ensure that the host running the virtual machine is accessible via SSH.

To convert the virtual machine, run:

```
virt-v2v -ic xen+ssh://root@vmhost.example.com -os pool --bridge brname vm-name
```

Where **vmhost.example.com** is the host running the virtual machine, **pool** is the local storage pool to hold the image, **brname** is the name of a local network bridge to connect the converted virtual machine's network to, and **vm-name** is the domain of the Xen virtual machine. You may also use the `--network` parameter to connect to a locally managed network, or specify multiple mappings in `/etc/virt-v2v.conf`. You will be prompted to enter the password for the user specified in the connection string, which is **root** in the example above

If your guest uses a Xen para-virtualized kernel (it would be called something like kernel-xen or kernel-xenU), **virt-v2v** will attempt to install a new kernel during the conversion process. You can avoid this requirement by installing a regular kernel, which won't reference a hypervisor in its name, alongside the Xen kernel prior to conversion. You should not make this newly installed kernel your default kernel, because Xen will not boot it. **virt-v2v** will make it the default during conversion.

2.1.2.4. Converting a VMware ESX virtual machine



Uninstall VMware Tools prior to conversion

When converting a Windows virtual machine from VMware ESX, ensure that VMware Tools is not installed on the virtual machine. The VMware Tools must be uninstalled prior to the conversion process. If a virtual machine is converted with the VMware Tools installed, it will not function correctly.

Ensure that the virtual machine is stopped prior to running the v2v process.

To convert the virtual machine, run:

```
virt-v2v -ic esx://esx.example.com/ -os pool --bridge brname vm-name
```

Where **esx.example.com** is the VMware ESX server, **pool** is the local storage pool to hold the image, **brname** is the name of a local network bridge to connect the converted virtual machine's

network to, and **vm-name** is the name of the virtual machine. You may also use the `--network` parameter to connect to a locally managed network, or specify multiple mappings in `/etc/virt-v2v.conf`. The progress of the conversion process will be displayed in percent as it runs.

Authenticating to the ESX server

Connecting to the ESX server will require authentication. **virt-v2v** supports password authentication when connecting to ESX. It reads passwords from `$HOME/.netrc`. The format of this file is described in the `netrc(5)` man page. An example entry is:

```
machine esx.example.com login root password s3cr3t
```



.netrc permissions

The `.netrc` file must have a permission mask of `0600` to be read correctly by **virt-v2v**

Connecting to an ESX server with an invalid certificate

In non-production environments, the ESX server may have a non-valid certificate, for example a self-signed certificate. In this case, certificate checking can be explicitly disabled by adding `'?no_verify=1'` to the connection URI as shown below:

```
... -ic esx://esx.example.com/?no_verify=1 ...
```

2.1.3. Running Converted Virtual Machines

On successful completion, **virt-v2v** will create a new libvirt domain for the converted virtual machine with the same name as the original virtual machine. It can be started as usual using libvirt tools, for example **virt-manager**.

Guest network configuration

virt-v2v cannot currently reconfigure a guest's network configuration. If the converted guest is not connected to the same subnet as the source, its network configuration may have to be updated.

Converting Virtual Machines to Run on Red Hat Enterprise Virtualization

virt-v2v can convert virtual machines to run on Red Hat Enterprise Virtualization. Virtual machines can be converted from Xen, KVM and VMware ESX environments. Before converting virtual machines to run on Red Hat Enterprise Virtualization, you must attach an export storage domain to the Red Hat Enterprise Virtualization data center being used. [Section 3.2, “Attaching an Export Storage Domain”](#) explains the process of attaching an export storage domain. For more information on export storage domains, see the *Red Hat Enterprise Virtualization Administration Guide*.

3.1. Acceptable Converted Storage Output Formats

It is important to note that when converting a guest to run on Red Hat Enterprise Virtualization, not all combinations of storage format and allocation policy are supported. The supported combinations differ according to whether the Red Hat Enterprise Virtualization data center the guest will be imported into uses block (FC or iSCSI) or file (NFS) for its *data* storage domain. Note that virt-v2v writes to an *export* storage domain, and this is always required to be NFS. The important element for a successful VM import into Red Hat Enterprise Virtualization is the type of the *data* domain. virt-v2v is unable to detect the data center type, so this check must be applied manually by the user.

Table 3.1. Allocation Policy: Preallocated.

Data Domain Type	Storage Format	Supported
NFS	raw	Yes
	qcow2	No
FC/iSCSI	raw	Yes
	qcow2	No

Table 3.2. Allocation Policy: Sparse.

Data Domain Type	Storage Format	Supported
NFS	raw	Yes
	qcow2	Yes
FC/iSCSI	raw	No
	qcow2	Yes

Data format and allocation policy of the virtual machine being converted by virt-v2v will be preserved unless the output data format and allocation policy are specified using the **-of** and **-oa** parameters respectively. To import virtual machines using sparse allocation into an FC or iSCSI data center, the storage format must be converted to qcow2. This is achieved by passing the parameters **-of qcow2 -oa sparse** to virt-v2v. Note that converting between raw and qcow2 formats is a resource intensive operation, and roughly doubles the length of time taken for the conversion process.



Note

Preallocated qcow2 storage is never supported in Red Hat Enterprise Virtualization, although virt-v2v is able to write it. Import to Red Hat Enterprise Virtualization will fail.

3.2. Attaching an Export Storage Domain

An export domain can be attached to a data center to enable the import or export of virtual machines from one data center to another. An export domain can also be used to backup virtual machines and templates.



Note

At a given time, an export data domain can only be attached to a single data center.

To attach an export storage domain:

1. Login to the Red Hat Enterprise Virtualization administration portal. Click the **Data Centers** tab.
Select the data center to which the export storage domain is to be attached.
2. The Details pane displays. Select the **Storage** tab.

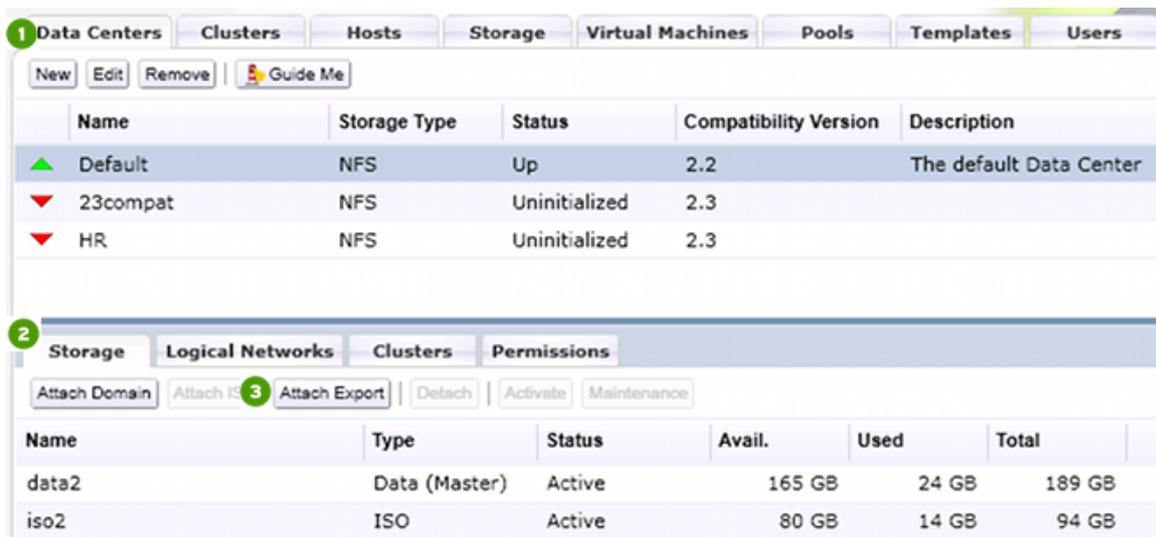


Figure 3.1. Attaching an Export Domain

3. Click the **Attach Export** button to add the storage location where the images are stored.
4. The **Attach Export Domain** dialog box displays, if there are export domains available.

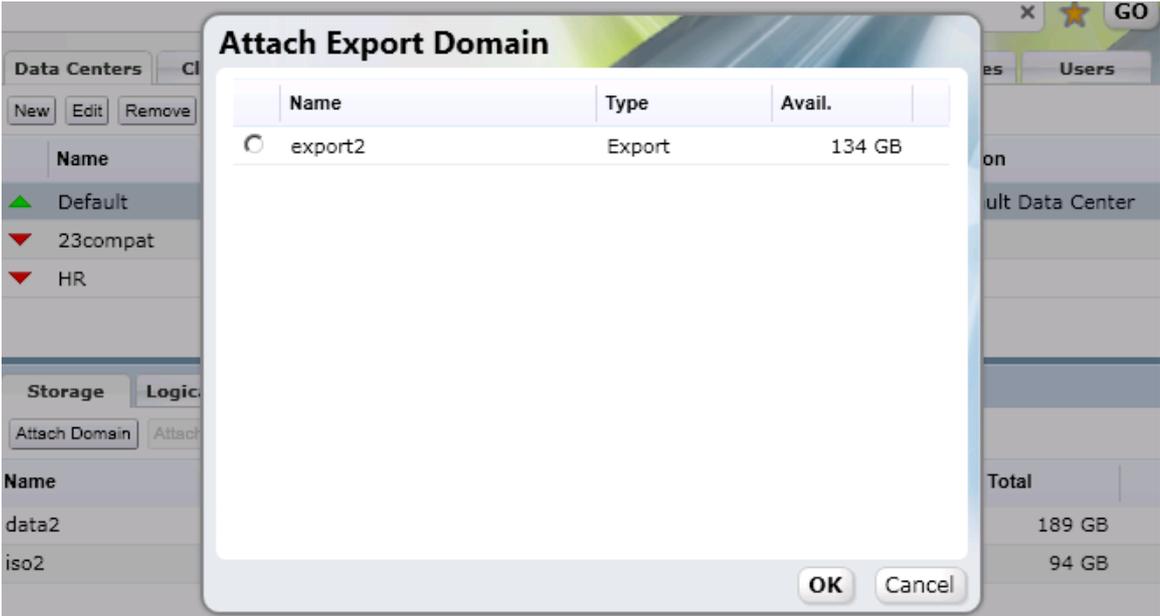


Figure 3.2. Attach Export Domain Dialog

- 5. Select the export domain from the list.
- 6. Click the **OK** button. The new export storage domain displays on the Storage tab of the Details pane, with a status of *Locked*, followed by *Inactive*.

Name	Type	Status	Avail.	Used	Total
data2	Data (Master)	Active	165 GB	24 GB	189 GB
iso2	ISO	Active	80 GB	14 GB	94 GB
export2	Export	Inactive	134 GB	7 GB	141 GB

Figure 3.3. The Inactive Export Domain

- 7. Select the new export storage domain on the Storage tab of the Details pane, and click the **Activate** button.
- 8. The export domain will be activated in a few moments and display an *Active* status.

Name	Type	Status	Avail.	Used	Total
data2	Data (Master)	Active	165 GB	24 GB	189 GB
iso2	ISO	Active	80 GB	14 GB	94 GB
export2	Export	Active	134 GB	7 GB	141 GB

Figure 3.4. Activated Export Domain

3.3. Converting a Virtual Machine

virt-v2v converts virtual machines from a foreign hypervisor to run on Red Hat Enterprise Virtualization. It automatically packages the virtual machine images and metadata, then uploads them to a Red Hat Enterprise Virtualization export storage domain. For more information on export storage domains, see [Section 3.2, “Attaching an Export Storage Domain”](#). **virt-v2v** always makes a copy of storage before conversion.

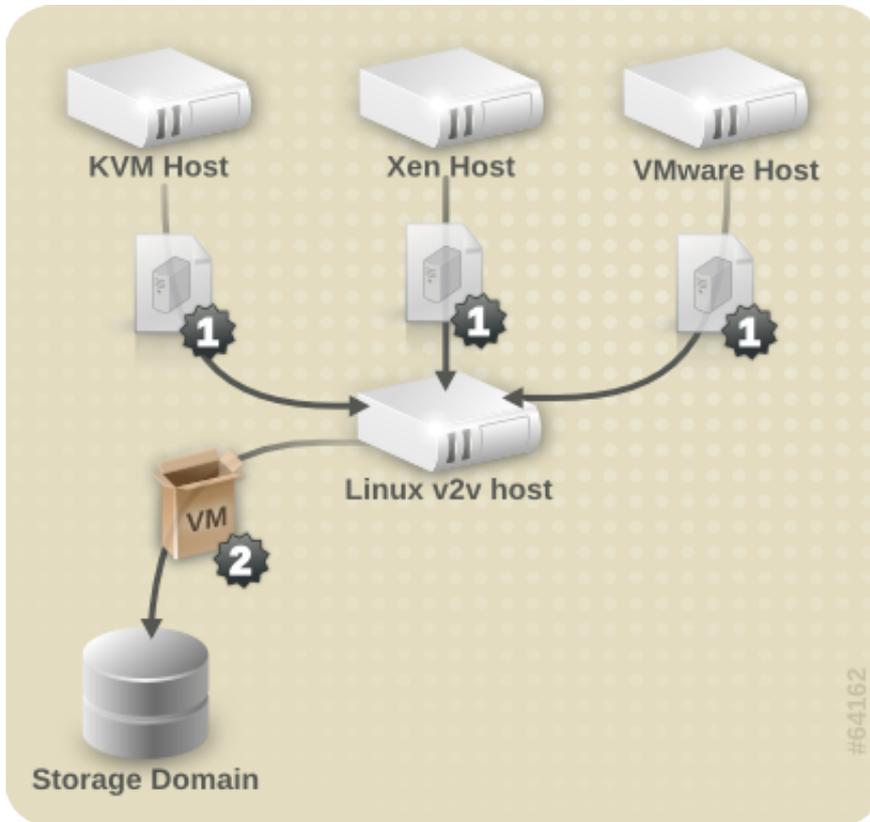


Figure 3.5. Converting a virtual machine

From the export storage domain, the virtual machine images can be imported into Red Hat Enterprise Virtualization using the administration portal.



Figure 3.6. Importing a virtual machine

3.3.1. Preparing to Convert a Virtual Machine

Before a virtual machine can be converted, ensure that the following steps are completed.

1. Create an NFS export domain. **virt-v2v** can transfer the converted VM directly to an NFS export storage domain. From the export storage domain, the VM can be imported into a Red Hat Enterprise Virtualization Data Center. The storage domain must be mountable by the machine running **virt-v2v**. When exporting to a Red Hat Enterprise Virtualization export domain, **virt-v2v** must run as root.



Configure rpcbind and nfslock on NFSv2 and NFSv3 clients

The export storage domain is accessed as an NFS share. By default, Red Hat Enterprise Linux 6 uses NFSv4, which does not require further configuration. However, for NFSv2 and NFSv3 clients, the **rpcbind** and **nfslock** services must be running on the host used to run **virt-v2v**. The network must also be configured to allow NFS access to the storage server. For more details refer to the *Red Hat Enterprise Linux Storage Administration Guide*.

2. Specify network mappings in **virt-v2v.conf**. This step is *optional*, and is not required for most use cases.

If your virtual machine has multiple network interfaces, **/etc/virt-v2v.conf** must be edited to specify the network mapping for all interfaces. You can specify an alternative **virt-v2v.conf** file with the **-f** parameter. If you are converting a virtual machine for output to both libvirt and Red Hat Enterprise Virtualization, separate **virt-v2v.conf** files should be used for each conversion. This is because the destination network bridge corresponding to the same source network bridge is usually different for libvirt and Red Hat Enterprise Virtualization output.

If your virtual machine only has a single network interface, it is simpler to use the **--network** or **--bridge** parameters, rather than modifying **virt-v2v.conf**.

3. Create a profile for the conversion in **virt-v2v.conf**. This step is *optional*. Profiles specify a conversion method, storage location, output format and allocation policy. When a profile is defined, it can be called using **--profile** rather than individually providing the **-o**, **-os**, **-of** and **-oa** parameters. See **virt-v2v.conf(5)** for details.

3.3.1.1. Preparing to convert a virtual machine running Linux

The following is required when converting virtual machines which run Linux, regardless of which hypervisor they are being converted from.

1. Obtain the software

As part of the conversion process, **virt-v2v** may install a new kernel and drivers on the virtual machine. If the virtual machine being converted is registered to Red Hat Network (RHN), the required packages will be automatically downloaded. If the virtual machine is not registered to RHN, the **virt-v2v.db** file ships with a list of RPMs used for this purpose. The RPMs relevant to your virtual machine must be downloaded manually from RHN and made available on the host running **virt-v2v**. The RPMs should be saved in the directory specified by the **path-root** configuration element, which by default is **/var/lib/virt-v2v/software/**. **virt-v2v** will display an error similar to [Example 2.1, “Missing Package error”](#) if software it depends upon for a particular conversion is not available.

Example 3.1. Missing Package error

```
virt-v2v: Installation failed because the following files referenced in the
configuration file are required, but missing:
rhel/6/kernel-2.6.32-128.el6.x86_64.rpm
rhel/6/ecryptfs-utils-82-6.el6.x86_64.rpm
rhel/6/ecryptfs-utils-82-6.el6.i686.rpm
```

To obtain the relevant RPMs for your environment, repeat these steps for each missing package:

1. Login to [Red Hat Network](#)¹
2. Select the **Package Search** tab.
3. In the **Search For** field, type the package name exactly matching the one shown in the error message. For the example shown in [Example 2.1, “Missing Package error”](#), the first package is **kernel-2.6.32-128.el6.x86_64**
4. In the **Where to search** field, select **In the following architectures** and tick the **x86_64** checkbox. Click **Search**.
5. A list of packages displays. Click the package name identical to the one in the error message.
6. You will be directed to the Details page, containing detailed descriptions of the package. Select **Download Package** at the bottom of the page
7. Save the downloaded package to the appropriate directory in **/var/lib/virt-v2v/software**. For Red Hat Enterprise Linux 6, the directory is **/var/lib/virt-v2v/software/rhel/6**

3.3.1.2. Preparing to convert a virtual machine running Windows



Converting a Windows virtual machine with multiple drives

When converting a virtual machine running Windows with multiple drives, for output to Red Hat Enterprise Virtualization, it is possible in certain circumstances that additional drives will not be displayed by default. Red Hat Enterprise Virtualization will always add a CD-ROM device to a converted virtual machine. If the virtual machine did not have a CD-ROM device before conversion, the new CD-ROM device may be assigned a drive letter which clashes with an existing drive on the virtual machine. This will render the existing device inaccessible. The occluded disk device can still be accessed by manually assigning it a new drive letter. It is also possible to maintain previous drive letter assignment by manually changing the drive letter assigned to the new CD-ROM device, then rebooting the virtual machine.

The following is required when converting virtual machines running Windows, regardless of which hypervisor they are being converted from. The conversion procedure depends on post-processing by the Red Hat Enterprise Virtualization Manager for completion. See [Section 5.2.2, “Configuration Changes for Windows Virtual Machines”](#) for details of the process.

1. Install the *libguestfs-winsupport* package on the host running **virt-v2v**. This package provides support for NTFS, which is used by many Windows systems. The *libguestfs-winsupport* package is provided by the **RHEL V2VWIN (v. 6 for 64-bit x86_64)** channel. Ensure your system is subscribed to this channel, then run the following command as root:

```
yum install libguestfs-winsupport
```

If you attempt to convert a virtual machine using NTFS without the *libguestfs-winsupport* package installed, the conversion will fail. An error message similar to [Example 2.2, “Error message when converting a Windows virtual machine without libguestfs-winsupport installed”](#) will be shown.

Example 3.2. Error message when converting a Windows virtual machine without libguestfs-winsupport installed

```
No operating system could be detected inside this disk image.

This may be because the file is not a disk image, or is not a virtual machine
image, or because the OS type is not understood by virt-inspector.

If you feel this is an error, please file a bug report including as much
information about the disk image as possible.
```

2. Install the *virtio-win* package on the host running **virt-v2v**. This package provides para-virtualized block and network drivers for Windows guests. The *virtio-win* package is provided by the **RHEL Server Supplementary (v. 6 64-bit x86_64)** channel. Ensure your system is subscribed to this channel, then run the following command as root:

```
yum install virtio-win
```

If you attempt to convert a virtual machine running Windows without the *virtio-win* package installed, the conversion will fail. An error message similar to [Example 2.3, “Error message when converting a Windows virtual machine without virtio-win installed”](#) will be shown.

Example 3.3. Error message when converting a Windows virtual machine without virtio-win installed

```
virt-v2v: Installation failed because the following files referenced in the
configuration file are required, but missing: /usr/share/virtio-win/drivers/i386/
Win2008
```

3. Upload the Guest Tools ISO to the Red Hat Enterprise Virtualization Manager

Note that the Guest Tools ISO is not required for the conversion process to succeed. However, it is recommended for all virtual machines running on Red Hat Enterprise Virtualization. The Manager installs drivers on the guest using the Guest Tools ISO after the virtual machines have been converted. See [Section 5.2.2, “Configuration Changes for Windows Virtual Machines”](#) for details.

The Guest Tools ISO is obtained as follows:

1. From the Manager, login to [Red Hat Network](#)².
2. Click on the **Download Software** tab.
3. Select the **Red Hat Enterprise Virtualization (x86-64)** channel.
4. Select the **Red Hat Enterprise Virt Manager for Desktops (v.2 x86)** or **Red Hat Enterprise Virt Manager for Servers (v.2 x86)** channel, as appropriate for your subscription.
5. Download **Guest Tools ISO for 2.2** and save it locally.
6. Upload the Guest Tools ISO using the ISO Uploader. See the *Red Hat Enterprise Virtualization Administration Guide* for instructions.

3.3.1.3. Preparing to convert a local Xen virtual machine

The following is required when converting virtual machines on a host which used to run Xen, but has been updated to run KVM. It is not required when converting a Xen virtual machine imported directly from a running libvirt/Xen instance.

1. Obtain the XML for the virtual machine

virt-v2v uses a libvirt domain description to determine the current configuration of the virtual machine, including the location of its storage. Before starting the conversion, obtain this from the host running the virtual machine with the following command:

```
virsh dumpxml vm-name > vm-name.xml
```

This will require booting into a Xen kernel to obtain the XML, as libvirt needs to connect to a running Xen hypervisor to obtain its metadata. The conversion process is optimized for KVM, so obtaining domain data while running a Xen kernel, then performing the conversion using a KVM kernel will be more efficient than running the conversion on a Xen kernel.

3.3.2. Converting a Virtual Machine

Once you have prepared to convert the virtual machines, use **virt-v2v** to perform the actual conversions. This section provides the steps to convert the virtual machines, and command syntax for **virt-v2v**. Note that conversions are resource intensive processes, involving copying the whole disk image for a virtual machine over the network. In typical environments, converting a single virtual machine takes approximately 5-10 minutes.

3.3.2.1. virt-v2v

virt-v2v converts virtual machines from a foreign hypervisor to run on Red Hat Enterprise Virtualization. The general command syntax for converting machines to run on Red Hat Enterprise Virtualization is:

```
virt-v2v -i libvirtxml -o rhev -os storage.example.com:/exportdomain --network rhev vm-name.xml
virt-v2v -o rhev -os storage.example.com:/exportdomain --network rhev vm-name
virt-v2v -ic esx://esx.example.com/?no_verify=1 -o rhev -os storage.example.com:/exportdomain --network rhev vm-name
```

A full specification of the parameters which can be used with **virt-v2v** is available in [Section 5.1, “virt-v2v Parameters”](#).

3.3.2.2. Converting a local Xen Virtual Machine

Ensure that the virtual machine's XML is available locally, and that the storage referred to in the XML is available locally at the same paths.

To convert the virtual machine from an XML file, run:

```
virt-v2v -i libvirtxml -o rhev -os storage.example.com:/exportdomain --network rhev vm-name.xml
```

Where **storage.example.com:/exportdomain** is the export storage domain, **rhev** is the locally managed network to connect the converted virtual machine's network to, and **vm-name.xml** is the path to the virtual machine's exported xml. You may also use the **--bridge** parameter to connect to a local network bridge, or specify multiple mappings in **/etc/virt-v2v.conf**.

To convert the virtual machine from a running Xen hypervisor, run:

```
virt-v2v -ic xen:/// -o rhev -os storage.example.com:/exportdomain --network rhevm vm-name
```

Where **storage.example.com:/exportdomain** is the export storage domain, **rhevm** is the locally managed network to connect the converted virtual machine's network to, and **vm-name** is the domain of the Xen virtual machine. You may also use the *--bridge* parameter to connect to a local network bridge, or specify multiple mappings in **/etc/virt-v2v.conf**.

If your guest uses a Xen para-virtualized kernel (it would be called something like kernel-xen or kernel-xenU), **virt-v2v** will attempt to install a new kernel during the conversion process. You can avoid this requirement by installing a regular kernel, which won't reference a hypervisor in its name, alongside the Xen kernel prior to conversion. You should not make this newly installed kernel your default kernel, because Xen will not boot it. **virt-v2v** will make it the default during conversion.

3.3.2.3. Converting a remote Xen Virtual Machine



Setup SSH Keys Prior to Converting a Remote VM with Multiple

Because each disk transfer requires a new SSH session, it is recommended that SSH keys be set up prior to the conversion for authentication. This is especially important for large disks. Otherwise, a user will be required to manually enter SSH credentials for each disk being transferred. Failure to do so before the SSH negotiation times out will cause **virt-v2v** to fail.

Xen virtual machines can be converted remotely via SSH. Ensure that the host running the virtual machine is accessible via SSH.

To convert the virtual machine, run:

```
virt-v2v -o rhev -ic xen+ssh://root@vmhost.example.com -os storage.example.com:/exportdomain --network rhevm vm-name
```

Where **vmhost.example.com** is the host running the virtual machine, **storage.example.com:/exportdomain** is the export storage domain, **rhevm** is the locally managed network to connect the converted virtual machine's network to, and **vm-name** is the domain of the Xen virtual machine. You may also use the *--bridge* parameter to connect to a local network bridge, or specify multiple mappings in **/etc/virt-v2v.conf**.

If your guest uses a Xen para-virtualized kernel (it would be called something like kernel-xen or kernel-xenU), **virt-v2v** will attempt to install a new kernel during the conversion process. You can avoid this requirement by installing a regular kernel, which won't reference a hypervisor in its name, alongside the Xen kernel prior to conversion. You should not make this newly installed kernel your default kernel, because Xen will not boot it. **virt-v2v** will make it the default during conversion.

3.3.2.4. Converting a local KVM Virtual Machine

Ensure that the virtual machine is stopped prior to running the v2v process.

To convert the virtual machine, run:

```
virt-v2v -o rhev -os storage.example.com:/exportdomain --network rhevm vm-name
```

Where **storage.example.com:/exportdomain** is the export storage domain, **rhevm** is the locally managed network to connect the converted virtual machine's network to, and **vm-name** is the domain of the KVM virtual machine. You may also use the `--bridge` parameter to connect to a local network bridge, or specify multiple mappings in `/etc/virt-v2v.conf`.

3.3.2.5. Converting a remote KVM Virtual Machine



Setup SSH Keys Prior to Converting a Remote VM with Multiple

Disks

Because each disk transfer requires a new SSH session, it is recommended that SSH keys be set up prior to the conversion for authentication. This is especially important for large disks. Otherwise, a user will be required to manually enter SSH credentials for each disk being transferred. Failure to do so before the SSH negotiation times out will cause `virt-v2v` to fail.

KVM virtual machines can be converted remotely via SSH. Ensure that the host running the virtual machine is accessible via SSH, and that the virtual machine is stopped prior to running the `v2v` process.

To convert the virtual machine, run:

```
virt-v2v -ic qemu+ssh://root@kvmhost.example.com/system -o rhev -os storage.example.com:/exportdomain --network rhevm vm-name
```

Where **kvmhost.example.com** is the host running the virtual machine, **storage.example.com:/exportdomain** is the export storage domain, **rhevm** is the locally managed network to connect the converted virtual machine's network to, and **vm-name** is the domain of the KVM virtual machine. You may also use the `--bridge` parameter to connect to a local network bridge, or specify multiple mappings in `/etc/virt-v2v.conf`.

3.3.2.6. Converting a VMware ESX Virtual Machine



Uninstall VMware Tools prior to conversion

When converting a Windows virtual machine from VMware ESX, ensure that VMware Tools is not installed on the virtual machine. The VMware Tools must be uninstalled prior to the conversion process. If a virtual machine is converted with the VMware Tools installed, it will not function correctly.

Ensure that the virtual machine is stopped prior to running the `v2v` process.

To convert the virtual machine, run:

```
virt-v2v -ic esx://esx.example.com/ -o rhev -os storage.example.com:/exportdomain --network
rhevm vm-name
```

Where **storage.example.com:/exportdomain** is the export storage domain, **rhevm** is the locally managed network to connect the converted virtual machine's network to, and **vm-name** is the name of the virtual machine. You may also use the `--bridge` parameter to connect to a local network bridge, or specify multiple mappings in `/etc/virt-v2v.conf`.

Authenticating to the ESX server

Connecting to the ESX server will require authentication. **virt-v2v** supports password authentication when connecting to ESX. It reads passwords from `$HOME/.netrc`. The format of this file is described in `netrc(5)`. An example entry is:

```
machine esx.example.com login root password s3cr3t
```



.netrc permissions

The `.netrc` file must have a permission mask of `0600` to be read correctly by **virt-v2v**

Connecting to an ESX server with an invalid certificate

In non-production environments, the ESX server may have a non-valid certificate, for example a self-signed certificate. In this case, certificate checking can be explicitly disabled by adding `'?no_verify=1'` to the connection URI as shown below:

```
... -ic esx://esx.example.com/?no_verify=1 ...
```

3.3.3. Importing and Running the Converted Virtual Machine

On successful completion, **virt-v2v** will upload the exported virtual machine to the specified export domain. To import and run the converted virtual machine:

1. In the Red Hat Enterprise Virtualization administration portal, select the export domain from the **Storage** tab.
2. Open the **VM Import** tab, select the appropriate virtual machine and click **Import**.
3. The **Import Virtual Machine(s)** dialog will display. Select the appropriate Destination Cluster and Destination Storage, then click **OK**. The import process will run in the background and may take several minutes. While it is running, the imported virtual machine will appear in the **Virtual Machines** tab with a status of **Image Locked**. You can monitor the status of the import operation from the **Events** tab.
4. When the import completes, the status will move to Down and the VM can be manually started.

For more information on importing virtual machines, see the *Red Hat Enterprise Virtualization Administration Guide*.

Network Configuration

virt-v2v cannot currently reconfigure a virtual machine's network configuration. If the converted virtual machine is not connected to the same subnet as the source, its network configuration may have to be updated.

3.3.4. Scripting the v2v Process

The entire v2v process can be scripted, enabling the automated batch processing of a large number of virtual machines. The process is broken up into two steps, which must be run on separate hosts.

1. Use **virt-v2v** to convert the virtual machines and copy them to the export storage domain. This step must be run on a Linux host. The process is detailed in [Section 3.3.2, “Converting a Virtual Machine”](#).
2. Once the conversion is complete, use the Red Hat Enterprise Virtualization Powershell API to import the virtual machines from the export storage domain. This step must be run on the Red Hat Enterprise Virtualization Manager server. The **Import-Vm** command performs the import process, and must be run once per virtual machine.

Example 3.4. Importing all VMs from the export storage domain to the DataStore storage domain on the Default Data Center

```
$exportdomain = Get-StorageDomain | ? {$_.Name -eq "export"}
$datadomain = Get-StorageDomain | ? {$_.Name -eq "DataStore"}
$dc = Select-DataCenter Name=Default
$cluster = Select-Cluster Name=Default
$candidates = Get-VmImportCandidates -StorageDomainId $exportdomain.StorageDomainId -
DataCenterId $dc.DataCenterId
foreach ($candidate in $candidates)
{
    Import-Vm -DataCenterId $dc.DataCenterId -SourceDomainId
    $exportdomain.StorageDomainId -DestDomainId $datadomain.StorageDomainId -ClusterId
    $cluster.ClusterId -VmId $candidate.VmId
}
```

Detailed documentation for the PowerShell API is available in the *Red Hat Enterprise Virtualization PowerShell API Guide*.

3.3.5. Scripted Bulk v2v Process

For bulk import scenarios, it is advantageous to be able to perform the scripted v2v process from a single host. Remote procedure calls to the Red Hat Enterprise Virtualization Manager can be made using the REST API. This enables a single script running on a single Linux host to perform both steps of the v2v process. [Figure 3.7, “Scripted bulk v2v process”](#) illustrates the steps performed by the script.

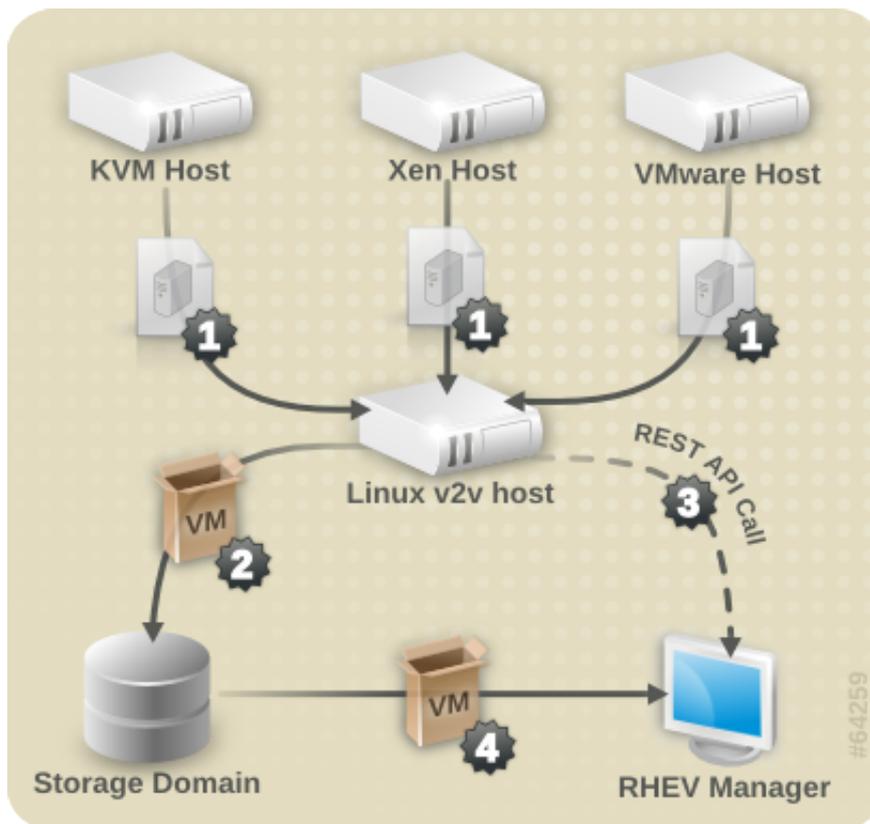


Figure 3.7. Scripted bulk v2v process

The scripted bulk v2v process involves the following steps, as shown in [Figure 3.7, “Scripted bulk v2v process”](#):

1. The virtual machine image is retrieved from the source hypervisor.
2. The virtual machine image is packaged and copied to the export storage domain.
3. A remote procedure call is made to the Red Hat Enterprise Virtualization Manager, telling it to import the virtual machine.
4. The Manager imports the virtual machine from the import storage domain.

To configure and run the scripted bulk v2v process:

1. Ensure the REST API is enabled on the Red Hat Enterprise Virtualization Manager, and it is accessible from the Linux host running the v2v process. For more information about the REST API, see the *Red Hat Enterprise Virtualization REST API Guide*.
2. On the Linux host, create the file `v2v.sh` with the following contents. Ensure you edit the script to contain appropriate values for your environment.

Example 3.5. Single host v2v script

```
#!/bin/sh
# Declare all VMs to import
XENDOMAINS=("rhelxen" "rhel2")
KVMDOMAINS=("rhelkvm")
VMWAREVMS=("rhel154vmware")

# Iterate through each Xen domain, performing the conversion
for domain in ${XENDOMAINS[@]}
```

```

do
    virt-v2v -ic xen:///localhost -o rhev -os storage.example.com:/exportdomain --
network rhevm $domain
done

# Iterate through each KVM domain, performing the conversion
for domain in ${KVMDOMAINS[@]}
do
    virt-v2v -o rhev -os storage.example.com:/exportdomain --network rhevm $domain
done

# Iterate through each VMware VM, performing the conversion
for vm in ${VMWAREVMS[@]}
do
    virt-v2v -ic esx://esx.example.com/?no_verify=1 -o rhev -os
storage.example.com:/exportdomain --network rhevm $vm
done

# Call the import VM procedure remotely on the RHEV Manager

# Set API variables
export BASE_URL='http://rhevm.example.com'
export HTTP_USER='rhevadmin@domain.example.com'
export HTTP_PASSWORD='password123'

# Get the storage domains
wget --auth-no-challenge --http-user=${HTTP_USER} --http-password=${HTTP_PASSWORD}
--header="Accept: application/xml" ${BASE_URL}/rhevm-api/storagedomains?search=name
%3D23compat -O exportdomain
EXPORT_DOMAIN=`xpath exportdomain '/storage_domains/storage_domain/@id' | sed -e 's/
id=/' | sed -e 's/"//g'`

# Get the datacenter
wget --auth-no-challenge --http-user=${HTTP_USER} --http-password=${HTTP_PASSWORD}
--header="Accept: application/xml" ${BASE_URL}/rhevm-api/datacenters?search=name
%3D23compat -O dc
DC=`xpath dc '/data_centers/data_center/@id' | sed -e 's/ id=/' | sed -e 's/"//g'`

# Get the cluster
wget --auth-no-challenge --http-user=${HTTP_USER} --http-password=${HTTP_PASSWORD} --
header="Accept: application/xml" ${BASE_URL}/rhevm-api/clusters?search=name%3DDefault
-O cluster
CLUSTER_ELEMENT=`grep "cluster id" cluster`
echo ${CLUSTER_ELEMENT}

# List contents of export storage domain
wget --auth-no-challenge --http-user=${HTTP_USER} --http-password=${HTTP_PASSWORD}
--header="Accept: application/xml" ${BASE_URL}/rhevm-api/datacenters/${DC}/
storagedomains/${EXPORT_DOMAIN}/vms -O vms

# For each vm, export
VMS=`xpath vms '/vms/vm/actions/link[@rel="import"]/@href' | sed -e 's/ href="//' |
sed -e 's/"//'`
echo '<action><cluster><name>23compat</name></cluster><storage_domain><name>data2</
name></storage_domain></action>' > importaction
wget --auth-no-challenge --http-user=${HTTP_USER} --http-password=${HTTP_PASSWORD}
--header="Accept: application/xml" --header="Content-Type: application/xml" --post-
file=importaction ${BASE_URL}$VMS

```

3. Run the **v2v.sh** script. It can take several hours to convert and import a large number of virtual machines.

Debugging and Troubleshooting

4.1. Debugging V2V conversions

Problems encountered when attempting a V2V conversion can be more easily explained to engineers or support services if debugging messages are enabled when V2V is run.

1. Before running a V2V conversion, enter the following in the terminal:

```
export LIBGUESTFS_TRACE=1
export LIBGUESTFS_DEBUG=1
```

2. The above exports increase the verbosity of the V2V process, causing virt-v2v to print out messages as it runs. These messages will be displayed in the terminal from which virt-v2v is run.
3. Simple redirection can be used to print virt-v2v debug messages to a file. Instead of running the conversion normally like this:

```
virt-v2v -i libvirtxml -os pool --bridge brname vm-name.xml
```

virt-v2v can be run like this:

```
LIBGUESTFS_TRACE=1 LIBGUESTFS_DEBUG=1 virt-v2v -i libvirtxml -os pool --bridge brname
vm-name.xml ... 2>&1 | tee virt-v2v.log
```


References

This chapter contains reference information for **virt-v2v**.

5.1. virt-v2v Parameters

The following parameters can be used with **virt-v2v**:

-i input	<p>Specifies the input method to obtain the guest for conversion. The default is libvirt. Supported options are:</p> <ul style="list-style-type: none"> • libvirt Guest argument is the name of a libvirt domain. • libvirtxml Guest argument is the path to an XML file containing a libvirt domain.
-ic URI	<p>Specifies the connection to use when using the libvirt input method. If omitted, this defaults to <code>qemu:///system</code>.</p> <p>virt-v2v can currently automatically obtain guest storage from local libvirt connections, ESX connections, and connections over SSH. Other types of connection are not supported.</p>
-o method	<p>Specifies the output method. If no output method is specified, the default is libvirt. Supported output methods are:</p> <ul style="list-style-type: none"> • libvirt, create a libvirt guest. See the <code>-oc</code> and <code>-os</code> options. <code>-os</code> must be specified for the libvirt output method. • rhev, create a guest on a Red Hat Enterprise Virtualization Export storage domain, which can later be imported using the manager. The export storage domain must be specified using <code>-os</code> for the <code>rhev</code> output method.
-oc URI	<p>Specifies the libvirt connection to use to create the converted guest. If omitted, this defaults to <code>qemu:///system</code>. Note that <code>virt-v2v</code> must be able to write directly to storage described by this libvirt connection. This makes writing to a remote connection impractical at present.</p>
-os storage	<p>Specifies the location where new storage will be created for the converted guest. This is dependent on the output method, specified by the <code>-o</code> parameter.</p> <p>For the libvirt output method, this must be the name of a storage pool. For the <code>rhev</code> output method, this specifies the NFS path to a Red Hat Enterprise Virtualization export storage domain. Note that the storage domain must have been previously initialized by the Red Hat Enterprise Virtualization Manager. The domain must be in the format <code><host>:<path></code>. For example:</p> <p>rhev-storage.example.com:/rhev/export</p> <p>The NFS export must be mountable and writable by the host running <code>virt-v2v</code>.</p>
-op pool	<p><i>DEPRECATED</i> Use <code>-os</code> instead. This parameter is still supported, but is deprecated in favor of <code>-os</code>.</p>
-osd domain	<p><i>DEPRECATED</i> Use <code>-os</code> instead. This parameter is still supported, but is deprecated in favor of <code>-os</code>.</p>

-of format	Specifies the on-disk format which will be used for the converted guest. Currently supported options are raw and qcow2 . The output format does not need to be the same as the source format - virt-v2v can convert from raw to qcow2 and vice versa. If not specified, the converted guest will use the same format as the source guest.
-oa allocation	Specifies whether the converted guest should use sparse or preallocated storage. The allocation scheme does not need to be the same as the source scheme - virt-v2v can convert from sparse to preallocated and vice versa. If not specified, the converted guest will use the same allocation scheme as the source.
-f file --config file	Load the virt-v2v configuration from file. Defaults to /etc/virt-v2v.conf if it exists.
-n network --network network	Map all guest bridges or networks which don't have a mapping in the configuration file to the specified network. This option cannot be used in conjunction with --bridge.
-b bridge --bridge bridge	Map all guest bridges or networks which don't have a mapping in the configuration file to the specified bridge. This option cannot be used in conjunction with --network.
--help	Display brief help.
--version	Display version number and exit.

5.2. Configuration Changes

As well as configuring libvirt appropriately, **virt-v2v** will make certain changes to a guest to enable it to run on a KVM hypervisor either with or without virtio drivers. These changes are specific to the guest operating system. The details specified here pertain to supported Red Hat based Linux distributions and Windows.

5.2.1. Configuration Changes for Linux Virtual Machines

Table 5.1. virt-v2v changes to Linux virtual machines

Change	Description
Kernel	Unbootable kernels (i.e. Xen para-virtualized kernels) will be uninstalled. No new kernel will be installed if there is a remaining kernel which supports virtio. If no remaining kernel supports virtio and the configuration file specifies a new kernel it will be installed and configured as the default.
X reconfiguration	If the guest has X configured, its display driver will be updated. See Table 5.2, “Configured drivers in a Linux Guest” for which driver will be used.
Rename block devices	If changes have caused block devices to change name, these changes will be reflected in /etc/fstab
Configure device drivers	Whether virtio or non-virtio drivers are configured, virt-v2v will ensure that the

Change	Description
	correct network and block drivers are specified in the modprobe configuration.
initrd	virt-v2v will ensure that the initrd for the default kernel supports booting the root device, whether it is using virtio or not.
SELinux	virt-v2v will initiate a relabel of the guest on the next boot. This ensures that any changes it has made are correctly labeled according to the guest's local policy.

virt-v2v will configure the following drivers in a Linux guest:

Table 5.2. Configured drivers in a Linux Guest

Para-virtualized driver type	Driver module
Display	cirrus
Storage	virtio_blk
Network	virtio_net
In addition, initrd will preload the virtio_pci driver	
Other drivers	
Display	cirrus
Block	Virtualized IDE
Network	Virtualized e1000

5.2.2. Configuration Changes for Windows Virtual Machines



Install *libguestfs-winsupport* and *virtio-win* packages

Before converting Windows virtual machines, ensure that the *libguestfs-winsupport* and *virtio-win* packages are installed on the host running **virt-v2v**. These packages provide support for NTFS and Windows para-virtualized block and network drivers. If you attempt to convert a virtual machine using NTFS without the *libguestfs-winsupport* package installed, the conversion will fail. If you attempt to convert a virtual machine running Windows without the *virtio-win* package installed, the conversion will fail giving an error message concerning missing files. See [Section 2.1.1.2, “Preparing to convert a virtual machine running Windows”](#) for details.

virt-v2v can convert virtual machines running Windows XP, Windows Vista, Windows 7, Windows Server 2003 and Windows Server 2008. The conversion process for virtual machines running Windows is slightly different to the process for virtual machines running Linux. Windows virtual machine images are converted as follows:

1. **virt-v2v** installs virtio block drivers.
2. **virt-v2v** installs the CDUpgrader utility.
3. **virt-v2v** copies virtio block and network drivers to **%SystemRoot%\Drivers\VirtIO**. The *virtio-win* package does not include network drivers for Windows 7 and Windows XP. For those

operating systems, the `rtl8139` network drivers are used. `rtl8139` support must be already available in the guest.

4. `virt-v2v` adds `%SystemRoot%\Drivers\VirIO` to `DevicePath`, meaning this directory is automatically searched for drivers when a new device is detected.
5. `virt-v2v` makes registry changes to include the virtio block drivers in the `CriticalDeviceDatabase` section of the registry, and ensure the `CDUpgrader` service is started at the next boot.

At this point, `virt-v2v` has completed the conversion. The converted virtual machine is now fully functional, and the conversion is complete for output to KVM managed by `libvirt`. If the virtual machine is being converted for output to Red Hat Enterprise Virtualization, the Red Hat Enterprise Virtualization Manager will perform additional steps to complete the conversion:

1. The virtual machine is imported and run on the Manager. See the *Red Hat Enterprise Virtualization Administration Guide* for details.



Do not interrupt the virtual machine import process

The first boot stage can take several minutes to run, and must not be interrupted. It will run automatically without any administrator intervention other than starting the virtual machine. To ensure the process is not interrupted, no user should login to the virtual machine until it has quiesced. You can check for this in the Manager GUI.

2. If the Guest Tools ISO has been uploaded to the Manager, as detailed in [Section 2.1.1.2, “Preparing to convert a virtual machine running Windows”](#), the Manager attaches the Guest Tools CD to the virtual machine.
3. `CDUpgrader` detects the Guest Tools ISO and installs all the virtio drivers from it, including additional tools that are not included in `virtio-win`. The virtio drivers are re-installed if the drivers in the Guest Tools ISO are newer than the ones previously installed from `virtio-win`. This ensures that the tools are kept up to date.

Appendix A. Revision History

Revision 7-0 **Friday December 02 2011** **Laura Bailey** lbailey@redhat.com

Release for GA of Red Hat Enterprise Linux 6.2.

Revision 6-0 **Friday July 22 2011** **Tim Hildred** timhildred@redhat.com

Promoted and emphasised section detailing acceptable storage format/allocation policy combinations
Changed warnings to importants

Revision 5-0 **Friday June 17 2011** **Tim Hildred** timhildred@redhat.com

Fixed the following bugs:
BZ#712320 -Added warnings about transferring multiple HDD images over ssh.
BZ#696050 -Can't import qcow2/preallocated guest to NFS data domain when convert a qcow2/sparse guest with -oa preallocated parameter. .
BZ#710161 -Section detailing the installation of the pre-reqs for virt-v2v is not detailed enough.

Revision 4-0 **Monday April 11 2011** **Cheryn Tan** cheryntan@redhat.com

Fixed the following bugs:
BZ#694773 - It is not required for upload Guest Tools ISO for guest to install driver
BZ#694778 - It's better to add converting a remote KVM virtual machine to RHEV to guide
BZ#694775 - The virt-v2v convert command syntax "virt-v2v -os pool --network netname vm-name" need update

Revision 3-0 **Friday April 8 2011** **Cheryn Tan** cheryntan@redhat.com

Fixed the following bugs:
BZ#694437 - The time stamp of copyright should be updated
BZ#694441 - Can't obtain virt-v2v relevant software with the V2V Guide steps
BZ#694442 - The doc should update about portmap service, as portmap is replaced by rpcbind in RHEL6

Revision 2-0 **Monday November 29 2010** **David Jorm** djorm@redhat.com

Complete draft except for example 3.3

Revision 1-0 **Monday October 25 2010** **David Jorm** djorm@redhat.com

Initial build

