



# Red Hat Virtualization 4.1

## RHEVM Shell 指南

为 Red Hat Virtualization 安装和使用 command Shell



## Red Hat Virtualization 4.1 RHEVM Shell 指南

---

为 Red Hat Virtualization 安装和使用 command Shell

Enter your first name here. Enter your surname here.

Enter your organisation's name here. Enter your organisational division here.

Enter your email address here.

## 法律通告

Copyright © 2022 | You need to change the HOLDER entity in the en-US/RHEVM\_Shell\_Guide.ent file |.

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

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

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

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

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

## 摘要

本指南包含用于安装和使用 Red Hat Virtualization Manager 命令行 Shell 的信息。

# 目录

|   |           |
|---|-----------|
| <b>第 1 章 使用 CLI</b> .....                       | <b>4</b>  |
| 1.1. 安装 CLI                                     | 4         |
| 1.2. TLS/SSL 认证                                 | 4         |
| 1.3. .OVIRTSHLLRC CONFIGURATION                 | 5         |
| 1.4. 运行 CLI                                     | 6         |
| 1.5. 与 CLI 交互                                   | 8         |
| 1.6. 集合   | 10        |
| <b>第 2 章 快速入门示例</b> .....                       | <b>12</b> |
| 2.1. 使用 CLI 创建基本虚拟化环境                           | 12        |
| <b>第 3 章 命令</b> .....                           | <b>17</b> |
| 3.1. 连接到 RHEVM                                  | 17        |
| 3.1.1. 连接到 RHEVM (连接)                           | 17        |
| 3.1.2. 从 RHEVM 断开连接(disconnect)                 | 18        |
| 3.2. RESOURCES                                  | 19        |
| 3.2.1. 列出集合中的资源(list)                           | 19        |
| 3.2.2. 显示资源(show)                               | 20        |
| 3.2.3. 添加资源 (添加)                                | 21        |
| 3.2.4. 更新资源 (更新)                                | 21        |
| 3.2.5. 删除资源 (删除)                                | 22        |
| 3.2.6. 对资源执行操作 (操作)                             | 23        |
| 3.2.7. 使用 Sub-Resources (--RESOURCE-identifier) | 24        |
| 3.3. 其他命令                                       | 24        |
| 3.3.1. 结束文件(EOF)                                | 24        |
| 3.3.2. 列出系统功能 (功能)                              | 25        |
| 3.3.3. 清除屏幕 (明确)                                | 25        |
| 3.3.4. 连接到虚拟机 (控制台)                             | 26        |
| 3.3.5. 打印输入(echo)                               | 26        |
| 3.3.6. 从 CLI 退出 (退出)                            | 27        |
| 3.3.7. 运行脚本 (文件)                                | 27        |
| 3.3.8. 显示帮助(help)                               | 27        |
| 3.3.9. 显示用户命令历史记录(history)                      | 28        |
| 3.3.10. 显示 CLI 信息(info)                         | 29        |
| 3.3.11. 测试连接(ping)                              | 29        |
| 3.3.12. 运行 Shell 命令(shell)                      | 30        |
| 3.3.13. 显示 Last Status (status)                 | 30        |
| 3.3.14. 显示系统摘要(summary)                         | 31        |
| <b>第 4 章 资源类型</b> .....                         | <b>32</b> |
| 4.1. BRICK                                      | 32        |
| 4.2. CDROM                                      | 32        |
| 4.3. CLUSTER                                    | 33        |
| 4.4. DATACENTER                                 | 35        |
| 4.5. DISK                                       | 36        |
| 4.6. GLUSTERVOLUME                              | 38        |
| 4.7. GROUP                                      | 41        |
| 4.8. 主机   | 41        |
| 4.9. NETWORK                                    | 44        |
| 4.10. NIC                                       | 46        |
| 4.11. 权限  | 49        |
| 4.12. PERMIT                                    | 51        |

|                           |           |
|---------------------------|-----------|
| 4.13. 配额                  | 51        |
| 4.14. ROLE                | 52        |
| 4.15. SNAPSHOT            | 53        |
| 4.16. 统计                  | 53        |
| 4.17. STORAGECONNECTION   | 54        |
| 4.18. STORAGEDOMAIN       | 56        |
| 4.19. TAG                 | 59        |
| 4.20. 模板                  | 60        |
| 4.21. USER                | 66        |
| 4.22. VM                  | 66        |
| 4.23. VMPOOL              | 74        |
| 4.24. VNICPROFILE         | 75        |
| <b>第 5 章 CLI 查询</b> ..... | <b>77</b> |
| 5.1. 查询语法                 | 77        |
| 5.2. 通配符                  | 77        |



# 第 1 章 使用 CLI

Red Hat Virtualization 套件具有命令行界面(CLI)。此 CLI 为用户提供了在标准 Web 界面外连接到 Red Hat Virtualization Manager 的方法。CLI 还包含脚本系统，它可帮助系统管理员通过客户端计算机在其虚拟化环境中执行定期维护或重复性任务。



## 重要

Red Hat Virtualization Manager 的版本 4 包括 Red Hat Enterprise Linux 7 的 CLI 版本 3.6。此 CLI 版本使用 Python SDK 的版本 3.6，以及 REST API 的版本 3 兼容模式。

## 1.1. 安装 CLI

将 Red Hat Virtualization CLI 安装到客户端机器中：

1. 以 **root** 用户身份登录客户端计算机。
2. 使用 Content Delivery Network 注册您的系统，在提示时输入您的客户门户网站用户名和密码：

```
# subscription-manager register
```

3. 找到 **Red Hat Virtualization** 订阅池并记下池 ID。

```
# subscription-manager list --available
```

4. 使用上一步中的池标识符将 **Red Hat Virtualization** 授权附加到系统：

```
# subscription-manager attach --pool=pool_id
```

5. 启用所需的仓库：

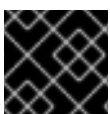
```
# subscription-manager repos --enable=rhel-7-server-rhv-4.1-rpms
```

6. 安装 CLI 软件包和依赖项：

```
# yum install ovirt-engine-cli
```

## 1.2. TLS/SSL 认证

Red Hat Virtualization Manager API 需要 Hypertext Transfer Protocol Secure (HTTPS) <sup>[1]</sup> 用于与客户端软件进行安全交互，如管理器的 SDK 和 CLI 组件。这包括从 Red Hat Virtualization Manager 获取证书并将其导入到客户端的证书存储中的过程。



## 重要

使用安全网络连接从 Red Hat Virtualization Manager 获取您的证书。

### 过程 1.1. 获取证书

您可以从 Red Hat Virtualization Manager 获取证书，并使用以下三种方法之一将其传送到客户端机器：



1. **方法 1** - 使用命令行工具从管理器下载证书。命令行工具示例包括 **cURL** 和 **Wget**，这两个工具均在多个平台上可用。

- a. 如果使用 **cURL**：

```
$ curl -o rhvm.cer http://[manager-fqdn]/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA
```

- b. 如果使用 **Wget**：

```
$ wget -O rhvm.cer http://[manager-fqdn]/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA
```

2. **方法 2** - 使用 Web 浏览器导航到位于以下证书：

```
http://[manager-fqdn]/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA
```

根据所选的浏览器，证书下载或导入到浏览器的密钥存储。

- a. **如果浏览器下载证书**：将文件保存为 **rhvm.cer**。

**如果浏览器导入证书**：将其从浏览器的认证选项导出，并将它保存为 **rhvm.cer**。

3. **方法 3** - 登录到 Manager，从信任存储中导出证书并将其复制到您的客户端机器中。

- a. 以 **root** 用户身份登录 Manager。

- b. 使用 Java **keytool** 管理工具从信任存储中导出证书：

```
$ keytool -exportcert -keystore /etc/pki/ovirt-engine/.truststore -alias cacert -storepass mypass -file rhvm.cer
```

这会创建一个名为 **rhvm.cer** 的证书文件。

- c. 使用 **scp** 命令将证书复制到客户端机器中：

```
$ scp rhvm.cer [username]@[client-machine]:[directory]
```

每个方法都会在客户端计算机上生成名为 **rhvm.cer** 的证书文件。API 用户将此文件导入到客户端的证书存储中。

### 过程 1.2. 将证书导入到客户端

- 将证书导入到客户端依赖于客户端本身存储和解释证书的方式。本指南包含导入证书的一些示例。有关导入证书的更多信息，对于不使用网络安全服务(NSS)或 Java KeyStore (JKS)的客户端，请参阅您的客户端文档。

## 1.3. .OVIRTSHLLRC CONFIGURATION

**.ovirtshellrc** 文件是一个配置文件，当用户首次连接到 **ovirt-shell** 时会自动创建和填充。它允许用户配置用于连接到 Red Hat Virtualization 环境的选项。**.ovirtshellrc** 文件默认位于 **/home/[user name]/.ovirtshellrc** 中。

.ovirtshellrc 文件的配置信息位于两个部分标题 **[cli]** 和 **[ovirt-shell]** 下。需要这些标题才能解析配置文件。

表 1.1. [CLI] 参数

| 名称                 | 类型  | Description   |
|--------------------|-----|---|
| <b>autoconnect</b> | 布尔值 | 切换是否自动连接到 ovirt-shell 会话。状态可以是 <b>True</b> 或 <b>False</b> 。 |
| <b>autopage</b>    | 布尔值 | 切换 shell 中的分页。状态可以是 <b>True</b> 或 <b>False</b> 。            |

表 1.2. [ovirt-shell] 参数

| 名称                              | 类型  | Description   |
|---------------------------------|-----|---|
| <b>用户名</b>                      | 字符串 | 用于登录的用户名。   |
| <b>timeout</b>                  | 整数  | 指定请求的超时。默认值为 -1。  |
| <b>extended_prompt</b>          | 布尔值 | 切换扩展提示选项，该选项在 shell 命令提示符中显示主机名。  |
| <b>url</b>                      | 字符串 | Red Hat Virtualization 环境的地址。   |
| <b>insecure</b>                 | 布尔值 | 切换 CA 证书要求。状态可以是 <b>True</b> 或 <b>False</b> 。                             |
| <b>renew_session</b>            | 布尔值 | 过期后切换会话的自动续订。状态可以是 <b>True</b> 或 <b>False</b> 。                           |
| <b>filter</b>                   | 布尔值 | 切换对象过滤。对象过滤允许用户根据其权限获取对象。只有管理员角色可以关闭过滤。状态可以是 <b>True</b> 或 <b>False</b> 。 |
| <b>session_timeout</b>          | 整数  | 为身份验证会话指定超时（以分钟为单位）。必须是一个正数。  |
| <b>ca_file</b>                  | 字符串 | 指定要使用的服务器 CA 证书。  |
| <b>dont_validate_cert_chain</b> | 布尔值 | 切换服务器 CA 证书的验证。状态可以是 <b>True</b> 或 <b>False</b> 。                         |
| <b>key_file</b>                 | 字符串 | 指定 client PEM key-file。   |
| <b>密码</b>                       | 字符串 | 用于用户名的密码。   |
| <b>cert_file</b>                | 字符串 | 指定 client PEM cert-file。  |

## 1.4. 运行 CLI

使用以下命令启动 CLI 应用程序：

```
# ovirt-shell
```

这个 **ovirt-shell** 应用程序是 Red Hat Virtualization 环境的交互式 shell。

用于连接到 Red Hat Virtualization Manager 的 URL、用户名、证书颁发机构文件和密码可在 `.ovirtshellrc` 文件中配置。`ovirt-shell` 命令使用此文件中的参数连接到 Manager，因此用户每次不需要指定选项。

另外，用户可以使用以下附加选项自动连接到 Red Hat Virtualization Manager。

```
# ovirt-shell -c -l "https://[server]/ovirt-engine/api" -P [port] -u "[user@domain]" -A "[certificate]"
```

确保替换以下值：

- `Server` - Red Hat Virtualization Manager 的主机名或 IP 地址。CLI 通过 REST API 连接到 Red Hat Virtualization Manager。
- `user@domain` - 登录到 Red Hat Virtualization Manager 的用户的用户名和目录服务域。
- `certificate` - 证书颁发机构文件的路径名称。

shell 将提示您输入密码，如果尚未提供，则 Red Hat Virtualization Manager 的用户名和 URL。



### 注意

如果您已在 `.ovirtshellrc` 文件中配置了用户名、密码、URL 和证书颁发机构文件，则不需要指定附加选项。



### 注意

证书是唯一义务性选项，因为本例中使用的其它选项会被 shell 提示。您可以不使用 `'-insecure'` 选项在不进行认证的情况下连接证书，但建议不要这样做，因为它可能会允许中间人(MITM)攻击者欺骗服务器的身份。

## ovirt-shell 的选项

`-h, --help`

显示 `ovirt-shell` 的帮助信息。

`-d, --debug`

启用调试。

`-L URL, --url=URL`

指定 API 入口点 URL。

`-U USERNAME, --username=USERNAME`

以这个用户身份进行连接。

`-K KEY_FILE, --key-file=KEY_FILE`

指定密钥文件。

`-C CERT_FILE, --cert-file=CERT_FILE`

指定证书文件。

`-A CA_FILE, --ca-file=CA_FILE`

指定服务器证书颁发机构文件。

**-I, --insecure**

允许 CLI 在没有认证的情况下通过 SSL 进行连接。请谨慎使用这个选项，因为它允许中间人(MITM)攻击者欺骗服务器的身份。

**-F, --filter**

启用基于用户权限的过滤。

**-P PORT, --port=PORT**

指定端口。

**-T TIMEOUT, --timeout=TIMEOUT**

指定超时。

**-c, --connect**

自动连接。

**-e, --extended-prompt**

为 shell 启用扩展提示选项。这个选项在命令提示符中显示环境的主机名。默认为 'false'。

**-e "命令资源", --execute-command="command resource"**

连接到管理器以"命令资源"的形式执行给定命令；命令资源，并将输出输出到 STDIO。

**-f FILE, --file=FILE**

从 FILE 而不是 stdin 读取命令。

**--kerberos**

使用有效的 Kerberos 票据来验证与 shell 的连接。



**注意**

具有非互动 shell 的用户可以从 shell 中连接到 Red Hat Virtualization Manager，其中可以使用 --password 选项。

## 1.5. 与 CLI 交互

CLI 是一个交互式 shell，用于从命令行控制 Red Hat Virtualization 环境。输入所需的命令和任何其他参数。

### 例 1.1. 输入 shell 命令

```
[RHEVM shell (connected)]# show vm RHEL6-Server
```

为了支持命令和参数组合的结构，CLI 包括通过按 **TAB** 键两次来列出和自动完成命令和参数的功能，类似于 **bash** shell。

-

### 例 1.2. 列出和自动完成命令和参数

在空白提示符处按双 **TAB** 来列出所有可用的命令。

```
[RHEVM shell (connected)]# TAB TAB
EOF      clear      echo      history   remove    summary
action   connect   exit     info     shell     update
add      console   file     list     show
capabilities disconnect help      ping     status
```

选择一个命令并按双 **TAB** 查看 命令的下一个可用参数集合。对于 **add** 命令，这将列出所有资源。

```
[RHEVM shell (connected)]# add TAB TAB
affinitygroup datacenter event      group      nic
quota      label     template  vmpool     cdrom
disk       filter    host      permission  role
storagedomain user      cluster   qos        glustervolume
network    permit    snapshot  tag        vm
```

双重 **TAB** 也完成命令和参数。

```
[RHEVM shell (connected)]# add vm TAB TAB
comment      console-enabled  cpu-architecture  delete_protected
description  disks-clone     display-type      io-threads
memory      name            os-type          rng_device-source
stateless   soundcard_enabled  start_paused     timezone
[RHEVM shell (connected)]# add vm naTAB TAB
[RHEVM shell (connected)]# add vm --name
```

请注意，双 **TAB** 还自动格式化 **na** 到 **--name** 参数，包括前缀。

如果不完整的参数与多个参数匹配，则双重 **TAB** 列出它们。

```
[RHEVM shell (connected)]# add vTAB TAB
vmpool      vm
```

CLI 提供了使用 **shell** 命令或感叹号(!)字符运行 Linux 命令的功能。

### 例 1.3. 运行 Linux shell 命令

使用 **shell** 命令：

```
[RHEVM shell (connected)]# shell ls -la
```

或者使用 bang (!) 字符：

```
[RHEVM shell (connected)]# !ls -la
```

与 Linux shell 类似，CLI 可以管道数据到其他命令和源。

### 例 1.4. 传送CLI命令

将 CLI 数据传送到 Linux shell 命令：

```
[RHEVM shell (connected)]# list vms --show-all | grep "Example"
name           : Example1
name           : Example2
name           : ExampleEngineering
description    : An Example description
name           : BestExampleVM
```

将 CLI 数据传送到文件：

```
[RHEVM shell (connected)]# list vms --show-all > list vms --show-all > VM_List.txt
```

CLI 还包含通过 `help` 命令提供每个命令的描述和语法的在线 **帮助系统**。

### 例 1.5. 在 `show` 命令中使用在线帮助

```
[RHEVM shell (connected)]# help show
```

您也可以从 Linux shell 连接到 Manager，以“*命令资源*”的形式执行特定的命令，并将其打印到 STDIO

### 例 1.6. 连接到 Manager 以执行特定命令

使用 `--execute` 或 `-E` 参数连接到 Manager 以执行特定的命令。

```
# ovirt-shell -c -I "https://[server]/ovirt-engine/api" -P [port] -u "[user@domain]" -A "[certificate]" -E
"list vms;list hosts"
[RHEVM shell (connected)]# list vms

id       : 9e6977f4-4351-4feb-bba0-dc7c22adec30
name     : desktop-01

id       : 60b12e28-7965-4296-86bf-c991aa32c2d5
name     : server-01

[RHEVM shell (connected)]# list hosts

id       : 3598cdb9-d21b-49bd-9491-59aff89b113
name     : Gluster

id       : a0c384f9-0940-4562-9c42-4ceaadf8f1f1
name     : Host-01

id       : 593ec966-c3ea-4bdc-84ad-5dc3f9fe64c7
name     : Host-03
```

## 1.6. 集合

有些命令参数需要集合。集合是一组子参数数据。集合使用以下语法定义：

```
[RHEVM shell (connected)]# command --param-collection  
{subparam1=value1;subparam2=value2;subparam3=value3;...},  
{subparam1=value1;subparam2=value2;subparam3=value3;...},...
```

集合的子参数列在每个资源页面上的资源参数列表后。

---

[1] [RFC 2818 HTTP Over TLS](#)中描述了 HTTPS。

## 第 2 章 快速入门示例

### 2.1. 使用 CLI 创建基本虚拟化环境

本章提供了一个示例，它演示了 CLI 在基本 Red Hat Virtualization 环境中添加虚拟机的能力。这个示例使用以下先决条件：

- 一个联网并配置了 Red Hat Enterprise Linux 主机，用作 hypervisor ；
- 带有一个两个共享的联网和配置 NFS 存储服务器：
  - /exports/data - 数据存储域；以及
  - /exports/iso - ISO 存储域。
- 已联网和配置 Red Hat Virtualization Manager ；
- 在 Red Hat Virtualization Manager 或客户端机器上安装 CLI ；以及
- 包含要安装的所需虚拟机操作系统的 ISO 文件。本章将 Red Hat Enterprise Linux Server 6 用于我们的安装 ISO 示例。



#### 注意

Red Hat Virtualization Manager 为每个资源生成全局唯一标识符(GUID)。本例中的标识符代码可能与 Red Hat Virtualization 环境中的标识符代码不同。

#### 过程 2.1. 快速入门示例

1. 加载 CLI shell 并连接到您的 Red Hat Virtualization Manager。

```
# ovirt-shell -c --url https://[rhev-host]/ovirt-engine/api --username [user]@[domain] --ca-file certificate/authority/path/name
```

2. 列出环境中的所有数据中心。这个示例使用 **Default** 数据中心。

```
[RHEVM shell (connected)]# list datacenters

id      : 5e3b55d8-c585-11e1-a7df-001a4a400e0d
name    : Default
description: The default Data Center
```

3. 列出所有主机集群并记下相关的集群 ID 或集群名称，在添加主机和创建虚拟机时是必需的。这个示例使用 **Default** 集群对 Red Hat Virtualization 环境中的资源进行分组。

```
[RHEVM shell (connected)]# list clusters

id      : 99408929-82cf-4dc7-a532-9d998063fa95
name    : Default
description: The default server cluster
```

4. 列出所有 CPU 配置文件并记下相关 CPU 配置集 ID，这在创建虚拟机时是必需的。这个示例使用 **Default** CPU 配置集。



```
[RHEVM shell (connected)]# list cpuprofiles

id      : 0000001a-001a-001a-001a-000000000035e
name    : Default
```

5. 使用 **show-all** 选项列出所有逻辑网络，以查看环境中的逻辑网络的详细信息。Red Hat Virtualization Manager 创建一个名为 **ovirtmgmt** 的默认逻辑网络来管理流量。本例使用 **Default** 数据中心的 **ovirtmgmt** 逻辑网络。

```
[RHEVM shell (connected)]# list networks --show-all

id      : 00000000-0000-0000-0000-0000000000009
name    : ovirtmgmt
description : Management Network
data_center-id: 5e3b55d8-c585-11e1-a7df-001a4a400e0d
mtu     : 0
required : True
status-state : operational
stp     : False
usages-usage : VM
```

请注意 **data\_center-id** 值与 **Default** 数据中心的 **id** 匹配。

6. 将 Red Hat Enterprise Linux 主机作为新的 hypervisor 添加到虚拟化环境中。主机被自动激活。

```
[RHEVM shell (connected)]# add host --name MyHost --address host.example.com --cluster-name Default --root_password p@55w0rd!
```

7. 通过创建、附加和激活 NFS 共享，将 NFS 共享添加为数据存储域。NFS 数据存储域是附加到数据中心的导出的 NFS 共享。它为虚拟机提供存储。确保将 **storage-address** 和 **storage-path** 替换为 NFS 服务器的正确值。

- a. 创建数据存储域。

```
[RHEVM shell (connected)]# add storagedomain --host-name MyHost --type data --storage-type nfs --storage_format v3 --storage-address x.x.x.x --storage-path /exports/data --name DataStorage
```

- b. 验证创建的存储域是否可用。创建过程可能需要几分钟时间。当 **status-state** 未连接后，您可以继续下一步。

```
[RHEVM shell (connected)]# show storagedomain DataStorage
id      : xxxx
name    : DataStorage
master  : False
status-state : unattached
...
```

- c.

将数据存储域附加到数据中心。存储域会自动激活。

```
[RHEVM shell (connected)] # add storagedomain --datacenter-identifier Default --name DataStorage
```



### 注意

如果没有激活存储域，使用以下命令手动激活它：

```
[RHEVM shell (connected)]# action storagedomain DataStorage --datacenter-identifier Default activate
```

8.

通过创建、附加和激活 NFS 共享，将 NFS 共享添加为 ISO 存储域。NFS ISO 存储域是附加到数据中心的导出的 NFS 共享。它为 DVD/CD-ROM ISO 和虚拟软盘磁盘(VFD)镜像文件提供存储。确保将 **storage-address** 和 **storage-path** 替换为 NFS 服务器的正确值。

a.

创建 ISO 存储域。

```
[RHEVM shell (connected)]# add storagedomain --host-name MyHost --type iso --storage-type nfs --storage_format v3 --storage-address x.x.x.x --storage-path /exports/iso --name ISOStorage
```

b.

验证创建的存储域是否可用。创建过程可能需要一段时间。当 **status-state** 未连接后，您可以继续下一步。

```
[RHEVM shell (connected)]# show storagedomain --name ISOStorage
id      : xxxx
name    : ISOStorage
master  : False
status-state : unattached
...
```

c.

将 ISO 存储域附加到数据中心。存储域会自动激活。

```
[RHEVM shell (connected)] # add storagedomain --datacenter-identifier Default --name ISOStorage
```

9.

创建新虚拟机。

```
[RHEVM shell (connected)]# add vm --name MyVM --cluster-name Default --template-name Blank --memory 536870912 --os-boot boot.dev=hd --cpu_profile-id 0000001a-001a-001a-001a-000000000035e
```

10.

使用 `add nic` 命令来添加新网络接口。添加 `vm-identifier` 选项，将接口作为 **MyVM** 的子资源附加，以及一个 `network-name` 选项来连接到 `ovirtmgmt` 网络。

```
[RHEVM shell (connected)]# add nic --vm-identifier MyVM --name nic1 --network-name
ovirtmgmt --bootable true
```

11.

使用 `add disk` 命令添加新虚拟硬盘。添加 `vm-identifier` 选项，以将磁盘作为 **MyVM** 的子资源附加。

```
[RHEVM shell (connected)]# add disk --vm-identifier MyVM --provisioned_size 8589934592 -
-interface virtio --format cow --storage_domains-storage_domain
storage_domain.name=DataStorage
```

12.

在 **Manager** 中，将 **ISO** 镜像上传到要使用的虚拟机的 **ISOStorage** 域。**Red Hat Virtualization Manager** 提供了一个 **ISO** 上传程序工具，用于确保镜像以正确的用户权限上传到正确的目录路径中。

```
# engine-iso-uploader --iso-domain=ISOStorage upload rhel-server-6.6-x86_64-dvd.iso
Please provide the REST API password for the admin@internal oVirt Engine user (CTRL+D
to abort):
```

13.

在 **CLI shell** 中，使用 `list files` 命令列出存储域中的可用 **ISO** 文件。

```
[RHEVM shell (connected)]# list files --storagedomain-identifier ISOStorage
```

14.

为您的安装介质添加虚拟 **CD-ROM** 驱动器。添加 `vm-identifier` 选项，以将 **CD-ROM** 连接为 **MyVM** 的子资源。

```
[RHEVM shell (connected)]# add cdrom --vm-identifier MyVM --file-id rhel-server-6.6-
x86_64-dvd.iso
```

15.

启动虚拟机。虚拟环境已完成，虚拟机包含所有必要的组件才能正常工作。

```
[RHEVM shell (connected)]# action vm MyVM start --vm-os-boot boot.dev=cdrom
```

注意 `vm-os-boot` 选项的使用。这会将引导设备更改为 `cdrom`，用于此初始引导会话。安装后，虚拟机重新启动并将引导设备恢复回 `hd`。

16.

使用 **列表事件** 和其他 **查询** 选项来显示特定的事件类型。虚拟机的 **start** 操作会在 **事件** 集合中添加多个条目。

```
[RHEVM shell (connected)]# list events --query "type=153"
```

```
id      : 105
description: MyVM was started by admin (Host: MyHost).
```

**"type=153"** 查询指的是用户启动虚拟机的事件。

17.

使用 **show event** 命令显示事件的综合详情。此命令可用于按类型、名称和 id 显示事件。

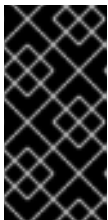
```
[RHEVM shell (connected)]# show event '60'
```

```
id      : 60
description : New Tag foo was created by admin@internal.
code     : 432
correlation_id: 3e4d4350
custom_id  : -1
flood_rate : 30
origin    : oVirt
severity   : normal
time      : 2013-07-03 10:57:43.257000+03:00
user-id   : fdfc627c-d875-11e0-90f0-83df133b58cc
```

18.

使用 **console** 命令访问虚拟机。

```
[RHEVM shell (connected)]# console MyVM
```



### 重要

确保您的客户端计算机安装了控制台应用程序，以匹配虚拟机的 **display-type**。可用的协议包括 **SPICE**（默认）和 **VNC**。

## 第 3 章 命令

## 3.1. 连接到 RHEVM

## 3.1.1. 连接到 RHEVM（连接）

**connect** 命令连接到 Red Hat Virtualization Manager。用于连接到 Red Hat Virtualization Manager 的 URL、用户名、证书颁发机构文件和密码可在 `.ovirtshellrc` 文件中配置。**connect** 命令使用此文件中的参数连接到 Manager，因此用户每次都不需要指定选项。

## 语法

**connect** [*options*]



## 注意

如果您已在 `.ovirtshellrc` 文件中配置了用户名、密码、URL 和证书颁发机构文件，则不需要指定附加选项。

表 3.1. 用于连接的选项

| 选项                 | 描述  | 必填                              |
|--------------------|---|---------------------------------|
| <b>--url</b>       | Red Hat Virtualization Manager 的 REST API 的 URL。它采用 <code>https://[server]/ovirt-engine/api</code> 的形式。 | 是                               |
| <b>--username</b>  | 尝试访问 Red Hat Virtualization Manager 的用户的用户名和目录服务域。这会格式为 <code>[username]@[domain]</code> 。              | 是                               |
| <b>--password</b>  | 尝试访问 Red Hat Virtualization Manager 的用户的密码。   | 是                               |
| <b>--key-file</b>  | 通过 SSL 连接的密钥文件。   | 否                               |
| <b>--cert-file</b> | 通过 SSL 连接的证书文件。   | 否                               |
| <b>--ca-file</b>   | 通过 SSL 进行连接的证书颁发机构文件。   | 是，除非使用了 <code>--insecure</code> |
| <b>--insecure</b>  | 允许 CLI 在没有认证的情况下通过 SSL 进行连接。请谨慎使用这个选项，因为它允许中间人(MITM)攻击者欺骗服务器的身份。  | 是，但只有在不提供证书颁发机构时                |

| 选项                     | 描述   | 必填 |
|------------------------|--|----|
| <code>--filter</code>  | 启用基于用户权限的过滤。                                     | 否  |
| <code>--port</code>    | 如果指定为 <code>--url</code> 的一部分，连接到 REST API 的端口号。 | 否  |
| <code>--timeout</code> | 连接的超时时间。   | 否  |

### 例 3.1. 未配置 `.ovirtshellrc` 时 连接 的示例

```
[RHEVM shell (disconnected)]# connect --url "https://rhev.example.com/ovirt-engine/api" --
username "admin@exampleids.com" --password "p@55w0rd!" --ca-file "/home/user/ca.crt"

=====
>>> connected to RHEVM manager 4.0.0.0 <<<
=====

[RHEVM shell (connected)]#
```



#### 注意

您可以不使用 `'-insecure'` 选项在不进行认证的情况下连接证书，但建议不要这样做，因为它可能会允许中间人(MITM)攻击者欺骗服务器的身份。

### 3.1.2. 从 RHEVM 断开连接(disconnect)

`disconnect` 命令与 Red Hat Virtualization Manager 断开连接。

#### 语法

`disconnect`

### 例 3.2. 断开连接示例

```
[RHEVM shell (connected)]# disconnect

=====
>>> disconnected from RHEVM manager <<<
=====

[RHEVM shell (disconnected)]#
```

## 3.2. RESOURCES

### 3.2.1. 列出集合中的资源(list)

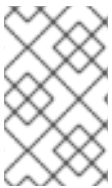
使用 `list` 命令显示特定类型的所有资源。列表还包括用于过滤结果的可选搜索查询。

语法

```
list [collection] [options]
```

表 3.2. 列出 标准选项

| 选项                                       | Description   |
|--|---|
| <code>--show-all</code>                  | 显示每个列出资源的所有非空属性。如果没有这个选项，则仅显示 <b>id</b> 、 <b>名称</b> 和 <b>描述</b> 属性。 |
| <code>--query [QUERY]</code>             | 根据 Red Hat Virtualization Manager 查询语言，使用服务器端查询来过滤列表。               |
| <code>--kwargs [QUERY]</code>            | 使用客户端查询过滤列表。  |
| <code>--case_sensitive true false</code> | 使用大小写敏感度匹配搜索查询。   |
| <code>--max</code>                       | 显示的最大结果数。   |



注意

特定于资源类型的选项列在每种资源类型的定义页面中。

#### 例 3.3. 列表示例

列出虚拟机：

```
[RHEVM shell (connected)]# list vms
```

列出带有列出所有属性的虚拟机：

```
[RHEVM shell (connected)]# list vms --show-all
```

列出状态为"up"的虚拟机：

```
[RHEVM shell (connected)]# list vms --query "status=up"
```

使用通配符列出所有域中与指定用户名匹配的用户：

```
[RHEVM shell (connected)]# list users --query "username=jsmith@*" --case_sensitive false
```

获取列表搜索语法的帮助信息：

```
[RHEVM shell (connected)]# list --help
```

### 3.2.2. 显示资源(show)

使用 **show** 命令显示资源属性。

语法

```
show [resource] [id/name] [options]
```



注意

特定于资源类型的选项列在每种资源类型的定义页面中。

#### 例 3.4. 展示的示例

根据 **id** 显示虚拟机：

```
[RHEVM shell (connected)]# show vm fcadfd5f-9a12-4a1e-bb9b-2b9d5c2e04c3
```

按名称显示虚拟机：

```
[RHEVM shell (connected)]# show vm RHEL6-Server
```

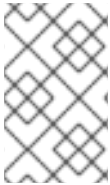


### 3.2.3. 添加资源（添加）

使用 `add` 命令来添加新资源。

语法

添加 *[resource] [options]*



注意

特定于资源类型的选项列在每种资源类型的定义页面中。

#### 例 3.5. add 示例

创建虚拟机：

```
[RHEVM shell (connected)]# add vm [vm-options]
```

创建用户：

```
[RHEVM shell (connected)]# add user [user-options]
```

可以使用 `expect` 选项同步（如果支持） `add` 命令：

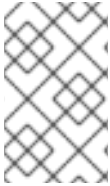
```
[RHEVM shell (connected)]# add vm [vm-options] --expect '201-created'
```

### 3.2.4. 更新资源（更新）

使用 `update` 命令修改现有的资源。

语法

更新 *[resource] [id/name] [options]*

**注意**

特定于资源类型的选项列在每种资源类型的定义页面中。

**例 3.6. 更新示例**

更新虚拟机：

```
[RHEVM shell (connected)]# update vm RHEL6-Server [vm-options]
```

**3.2.5. 删除资源（删除）**

使用 `remove` 命令删除资源。

语法

删除 *[resource] [id/name] [options]*

表 3.3. 删除 标准选项

| 选项                   | Description   |
|----------------------|---|
| <code>--async</code> | 执行异步删除资源。   |
| <code>--force</code> | 对资源执行强制删除。这会删除所有特定资源的数据库条目和关联。此操作仅适用于 <b>数据中心</b> 和 <b>vm</b> 资源。 |

**注意**

特定于资源类型的选项列在每种资源类型的定义页面中。

**例 3.7. 删除示例**

删除虚拟机：

```
[RHEVM shell (connected)]# remove vm RHEL6-Server
```

异步删除虚拟机：

```
[RHEVM shell (connected)]# remove vm RHEL6-Server --async true
```

强制删除虚拟机：

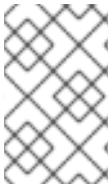
```
[RHEVM shell (connected)]# remove vm RHEL6-Server --force
```

### 3.2.6. 对资源执行操作（操作）

使用 `action` 命令执行与资源类型相关的特殊功能。

语法

```
action [resource] [id/name] [action] [options]
```



注意

特定于资源操作的选项列在每种资源类型的定义页面中。

#### 例 3.8. 操作示例

启动虚拟机

```
[RHEVM shell (connected)]# action vm RHEL6-Server start
```

停止虚拟机：

```
[RHEVM shell (connected)]# action vm RHEL6-Server stop
```

可以使用 `async` 选项同步（如果支持）命令：

```
[RHEVM shell (connected)]# action vm [vm-options] --async false
```

### 3.2.7. 使用 Sub-Resources (--RESOURCE-identifier)

某些资源充当其他资源的子资源。这意味着子资源及其父资源之间存在依赖关系。使用 `-RESOURCE-identifier [name]` 选项，其中 `RESOURCE` 是父资源类型，以作为父资源的子资源部分。

#### 例 3.9. 创建带有添加的子资源的示例

在虚拟机上创建一个 NIC：

```
[RHEVM shell (connected)]# add nic --vm-identifier RHEL6-Server [nic-options]
```

请注意，使用 `--vm-identifier RHEL6-Server` 选项。这会在 `RHEL6-Server` 虚拟机上添加一个 NIC。

在虚拟机上创建存储磁盘：

```
[RHEVM shell (connected)]# add disk --vm-identifier RHEL6-Server [user-options]
```

请注意，使用 `--vm-identifier RHEL6-Server` 选项。这会在 `RHEL6-Server` 虚拟机上添加一个存储磁盘。

## 3.3. 其他命令

### 3.3.1. 结束文件(EOF)

使用 EOF 命令，通过 `Ctrl+D` 序列离开 CLI shell。

语法

```
EOF
```

#### 例 3.10. EOF示例

保留 CLI shell :

```
[RHEVM shell (connected)]# EOF
```

### 3.3.2. 列出系统功能（功能）

使用 `capabilities --features` 命令列出当前版本的所有版本功能和新功能。

语法

`capabilities --features`

#### 例 3.11. 功能示例

列出当前版本的系统功能：

```
[RHEVM shell (connected)]# capabilities --features

name                : Search - Case Sensitivity
description          : Ability to specify whether a search query should ignore case, by
providing a URL parameter
url-parameters_set-parameter-name : case_sensitive
url-parameters_set-parameter-context: matrix
url-parameters_set-parameter-type  : boolean
:
```

### 3.3.3. 清除屏幕（明确）

使用 `clear` 命令来清除 CLI 屏幕。

语法

`clear`

#### 例 3.12. clear 的示例

清除屏幕：

```
[RHEVM shell (connected)]# clear
```

### 3.3.4. 连接到虚拟机（控制台）

使用 `console` 命令打开虚拟机的控制台。这个命令根据虚拟机的 `display-type` 参数打开外部 VNC 或 SPICE 客户端。

语法

```
console [vm-id/vm-name]
```

#### 例 3.13. console 示例

打开虚拟机的图形控制台：

```
[RHEVM shell (connected)]# console RHEL6-Server
```

### 3.3.5. 打印输入(echo)

使用 `echo` 命令将输入打印到屏幕。使用 `$out` 变量打印最后一个 `shell` 命令输出。

语法

```
echo [input]
```

#### 例 3.14. echo 示例

打印输入：

```
[RHEVM shell (connected)]# echo "Example text!"
```

打印最后一个输出：

```
[RHEVM shell (connected)]# echo $out
```

### 3.3.6. 从 CLI 退出（退出）

使用 `exit` 命令来离开 CLI。

语法

`exit`

#### 例 3.15. 例如 `exit`

保留 CLI：

```
[RHEVM shell (connected)]# exit
```

### 3.3.7. 运行脚本（文件）

使用 `file` 命令运行 CLI 脚本文件。脚本是包含要执行的命令列表的纯文本文件。

语法

`file [file-location]`

#### 例 3.16. 文件示例

运行脚本文件：

```
[RHEVM shell (connected)]# file /example/example-script
```

### 3.3.8. 显示帮助(help)

使用 `help` 命令显示 CLI 命令和资源组合的帮助信息。

语法

`help [command] [resource] [options]`

### 例 3.17. 帮助示例

获取 CLI 帮助：

```
[RHEVM shell (connected)]# help
```

获取 **add** 命令的帮助信息：

```
[RHEVM shell (connected)]# help add
```

获取 **vm** 资源类型的 **add** 命令的帮助信息：

```
[RHEVM shell (connected)]# help add vm
```

### 3.3.9. 显示用户命令历史记录(history)

使用 **history** 命令显示 CLI shell 的用户命令的历史记录。

语法

**history**

### 例 3.18. history 示例

显示用户命令历史记录：

```
[RHEVM shell (connected)]# history
```

### 例 3.19. history --first 示例

使用 **--first *n*** 参数显示 **user** 命令历史记录中的第一个指定条目：

```
[RHEVM shell (connected)]# history --first 5
```

### 例 3.20. history --last 的示例



使用 `--last n` 参数显示 `user` 命令历史记录中的最后一个指定条目：

```
[RHEVM shell (connected)]# history --last 5
```

### 3.3.10. 显示 CLI 信息(info)

使用 `info` 命令显示环境连接详细信息和版本信息。

语法

`info`

#### 例 3.21. info 示例

查看 CLI 信息：

```
[RHEVM shell (connected)]# info

backend version: 3.1
sdk version    : 3.1.0.4
cli version    : 3.1.0.6
python version : 2.7.3.final.0

entry point    : https://www.example.com:8443/ovirt-engine/api
```

### 3.3.11. 测试连接(ping)

使用 `ping` 命令测试到 Red Hat Virtualization Manager 的连接。命令检索远程资源，并确保连接的 URL、用户名和密码正确。

语法

`ping`

#### 例 3.22. ping 示例

测试您的连接：

```
[RHEVM shell (connected)]# ping  
success: RHEVM manager could be reached OK.
```

### 3.3.12. 运行 Shell 命令(shell)

使用 `shell` 命令从 Linux shell 运行命令。此命令有助于与 Red Hat Virtualization Manager shell 一起执行文件管理任务。

语法

```
shell [vm-id|vm-name]
```

#### 例 3.23. shell示例

列出当前工作目录中的文件：

```
[RHEVM shell (connected)]# shell ls
```

创建一个文件：

```
[RHEVM shell (connected)]# shell touch example.txt
```

复制文件：

```
[RHEVM shell (connected)]# shell cp example.txt /example-dir/.
```



注意

CLI 使用 **bang (!)** 字符提供了 `shell` 的替代选择。例如：

```
[RHEVM shell (connected)]# !touch example.txt
```

### 3.3.13. 显示 Last Status (status)

使用 **status** 命令显示最近运行的命令状态。

语法

## status

### 例 3.24. status示例

查看最后一个状态：

```
[RHEVM shell (connected)]# status  
last command status: 0 (OK)
```

### 3.3.14. 显示系统摘要(summary)

使用 **summary** 命令显示系统状态的摘要。

语法

## summary

### 例 3.25. summary示例

显示系统状态：

```
[RHEVM shell (connected)]# summary  
  
hosts-active      : 1  
hosts-total      : 2  
storage_domains-active: 2  
storage_domains-total : 3  
users-active     : 1  
users-total      : 1  
vms-active       : 1  
vms-total        : 1
```

## 第 4 章 资源类型

## 4.1. BRICK

**brick** 资源类型将 Red Hat Virtualization 环境中所有 Gluster brick 进行分组。

表 4.1. Gluster brick 参数

| 名称                     | 类型  | Description             | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|------------------------|-----|-------------------------|----|---------------------|-----------------------|
| <b>--server_id</b>     | 字符串 | Gluster 服务器的地址。         | 是  | 是                   | 否                     |
| <b>--brick_dir</b>     | 字符串 | Gluster 服务器上的 brick 目录。 | 是  | 是                   | 否                     |
| <b>--replica_count</b> | 整数  | 定义复制卷的文件复制数。            | 否  | 是                   | 否                     |
| <b>--stripe_count</b>  | 整数  | 定义条状卷的条带数               | 否  | 是                   | 否                     |

下表列出了针对基于资源的命令的其他 **glustervolume** 选项。

表 4.2. 附加命令选项

| 选项                                | Description                                  |
|-----------------------------------|--|
| <b>--cluster-identifier</b>       | 对包含 <b>glustervolume</b> 子资源的 <b>集群</b> 的引用。 |
| <b>--glustervolume-identifier</b> | 将 brick 添加到 <b>glustervolume</b> 作为子资源。      |

## 例 4.1. 创建 brick

```
[RHEVM shell (connected)]# add brick --cluster-identifier Default --glustervolume-identifier
GlusterVol1 --server_id="server1" --brick_dir="/exp1"
```

## 4.2. CDROM

**cdrom** 资源类型对 Red Hat Virtualization 环境中的所有虚拟 CD-ROM 驱动器资源进行分组。

表 4.3. CD-ROM 参数

| 名称                     | 类型  | Description               | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|------------------------|-----|---------------------------|----|---------------------|-----------------------|
| <code>--file-id</code> | 字符串 | 定义驻留在 ISO 存储域中的 ISO 的文件名。 | 是  | 是                   | 是                     |

**例 4.2. 创建新的 CD-ROM**

```
[RHEVM shell (connected)]# add cdrom --vm-identifier MyVM --file-id rhel-server-6.2-x86_64-dvd.iso.iso
```

**例 4.3. 更新 CD-ROM**

```
[RHEVM shell (connected)]# update cdrom --vm-identifier MyVM --file-id rhel-server-6.3-x86_64-dvd.iso.iso
```

**例 4.4. 删除 CD-ROM**

```
[RHEVM shell (connected)]# remove cdrom --vm-identifier MyVM rhel-server-6.3-x86_64-dvd.iso.iso
```

**4.3. CLUSTER**

集群 资源类型对 Red Hat Virtualization 环境中所有主机集群资源进行分组。

表 4.4. 集群参数

| 名称                                 | 类型  | Description                    | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|------------------------------------|-----|--------------------------------|----|---------------------|-----------------------|
| <code>--cpu-id</code>              | 字符串 | 定义所有主机所有主机的服务器 CPU 引用必须在集群中支持。 | 是  | 是                   | 是                     |
| <code>--data_center-id name</code> | 字符串 | 对主机集群的数据中心的引用。                 | 是  | 是                   | 否                     |
| <code>--name</code>                | 字符串 | 主机集群的名称。                       | 是  | 是                   | 是                     |

| 名称   | 类型     | Description   | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|--|--------|---|----|---------------------|-----------------------|
| <b>--version-major</b>                               | int    | 集群的主版本号。例如，对于 Red Hat Virtualization 4.1，主要版本是 4。   | 是  | 是                   | 是                     |
| <b>--version-minor</b>                               | int    | 集群的次版本号。例如，对于 Red Hat Virtualization 4.1，次版本为 0。  | 是  | 是                   | 是                     |
| <b>--description</b>                                 | 字符串    | 主机集群的描述。  | 否  | 是                   | 是                     |
| <b>--error_handling-on_error</b>                     | 字符串    | 定义当集群中的主机无法正常运行时，包括 <b>迁移</b> 、 <b>do_not_migrate</b> 和 <b>migrate_highly_available</b> 。   | 否  | 是                   | 是                     |
| <b>--gluster_service</b>                             | 布尔值    | 状态可以是 <b>true</b> 或 <b>false</b> 。  | 否  | 是                   | 是                     |
| <b>--memory_policy-overcommit-percent</b>            | double | 主机无法再运行任何更多虚拟机前允许使用的主机内存百分比。由于 KSM 下的内存共享，虚拟机可以使用超过可用内存。建议的值包括 <b>100</b> (None), <b>150</b> (Server Load) 和 <b>200</b> (Desktop Load)。 | 否  | 是                   | 是                     |
| <b>--memory_policy-transparent_hugepages-enabled</b> | 布尔值    | 定义 Transparent Hugepages 的启用状态。状态可以是 <b>true</b> 或 <b>false</b> 。   | 否  | 是                   | 是                     |
| <b>--scheduling_policy-policy</b>                    | 字符串    | 集群中的主机的虚拟机调度模式，如 <b>evenly_distributed</b> 、 <b>power_saving</b> 或 <b>blank</b> 。   | 否  | 是                   | 是                     |
| <b>--scheduling_policy-thresholds-duration</b>       | int    | 在调度程序启动并将负载移到另一主机前，主机可以超载的秒数。   | 否  | 是                   | 是                     |
| <b>--scheduling_policy-thresholds-high</b>           | int    | 控制主机在被视为超载前可以具有的最高 CPU 用量百分比。   | 否  | 是                   | 是                     |
| <b>--scheduling_policy-thresholds-low</b>            | int    | 控制主机在被视为使用率不足前可以具有的最小 CPU 用量百分比。  | 否  | 是                   | 是                     |

| 名称                               | 类型            | Description  | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|----------------------------------|---------------|--|----|---------------------|-----------------------|
| <code>-- threads_as_cores</code> | 布尔值           | 主机将线程视为内核，允许主机使用超过主机内核数的处理器内核总数来运行虚拟机。状态可以是 <b>true</b> 或 <b>false</b> 。 | 否  | 否                   | 否                     |
| <code>-- trusted_service</code>  | 布尔值           | 定义是否使用 OpenAttestation 服务器来验证主机。   | 否  | 是                   | 是                     |
| <code>--virt_service</code>      | 布尔值           | 状态可以是 <b>true</b> 或 <b>false</b> 。                                       | 否  | 是                   | 是                     |
| <code>--expect</code>            | '201-created' | 请求会变得异步，直到返回预期的 HTTP 标头。对于长时间运行的任务，在任务完成之前，否则会返回的任务。                     | 否  | 否                   | 否                     |
| <code>-- correlation_id</code>   | 字符串           | 用于跨系统日志记录的操作标记标识符。如果客户端没有定义标识符，则会生成一个。                                   | 否  | 是                   | 否                     |

#### 例 4.5. 创建新集群

```
[RHEVM shell (connected)]# add cluster --name Engineering --cpu-id "Intel Penryn Family" --
datacenter-name Default --version-major 3 --version-minor 2
```

#### 例 4.6. 更新集群

```
[RHEVM shell (connected)]# update cluster Engineering --name Finance
```

#### 例 4.7. 删除集群

```
[RHEVM shell (connected)]# remove cluster Engineering
```

## 4.4. DATACENTER

数据中心 资源类型对 Red Hat Virtualization 环境中的所有数据中心资源进行分组。

表 4.5. 数据中心参数

| 名称                      | 类型            | Description   | 必填 | 用户<br>Crea-<br>table | User<br>Upda<br>table |
|-------------------------|---------------|---|----|----------------------|-----------------------|
| <b>--name</b>           | 字符串           | 数据中心的名称。  | 是  | 是                    | 是                     |
| <b>--storage-type</b>   | 字符串           | 数据中心的存储类型，包括 <b>iscsi</b> 、 <b>fcp</b> 、 <b>nfs</b> 、 <b>localfs</b> 或 <b>posixfs</b> 。 | 是  | 是                    | 是                     |
| <b>--version-major</b>  | int           | 数据中心的主版本号。例如，对于 Red Hat Virtualization 4.1，主要版本是 4。                                     | 是  | 是                    | 是                     |
| <b>--version-minor</b>  | int           | 数据中心的次要版本号。例如，对于 Red Hat Virtualization 4.1，次版本为 0。                                     | 是  | 是                    | 是                     |
| <b>--description</b>    | 字符串           | 数据中心的描述。  | 否  | 是                    | 是                     |
| <b>--storage-format</b> | 字符串           | 数据中心的元数据格式，包括 <b>v1</b> 、 <b>v 2</b> 或 <b>v3</b> 。                                      | 否  | 是                    | 是                     |
| <b>--expect</b>         | '201-created' | 请求会变得异步，直到返回预期的 HTTP 标头。对于长时间运行的任务，在任务完成之前，否则会返回的任务。                                    | 否  | 否                    | 否                     |
| <b>--correlation-id</b> | 字符串           | 用于跨系统日志记录的操作标记标识符。如果客户端没有定义标识符，则会生成一个。  | 否  | 是                    | 否                     |

#### 例 4.8. 创建新数据中心

```
[RHEVM shell (connected)]# add datacenter --name Boston --storage-type nfs --version-major 3 --version-minor 2
```

#### 例 4.9. 更新数据中心

```
[RHEVM shell (connected)]# update datacenter Boston --name India
```

#### 例 4.10. 删除数据中心

```
[RHEVM shell (connected)]# remove datacenter Boston
```

## 4.5. DISK



磁盘 资源类型对 Red Hat Virtualization 环境中所有虚拟硬盘资源进行分组。

表 4.6. 磁盘参数

| 名称  | 类型  | Description   | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|---|-----|---|----|---------------------|-----------------------|
| <code>--provisioned_size</code>               | int | 为磁盘保留的存储空间。这个空间是预先分配的，即使 <b>磁盘大小</b> 小于 <b>provisioned_size</b>                 | 是  | 是                   | 是                     |
| <code>--interface</code>                      | 字符串 | 磁盘的接口类型。 <b>ide</b> 或 <b>virtio</b> 。   | 是  | 是                   | 是                     |
| <code>--format</code>                         | 字符串 | 底层存储格式。Copy On Write ( <b>cow</b> )允许快照，且性能较小的开销。原始( <b>原始</b> )不允许快照，但提供更好的性能。 | 是  | 是                   | 是                     |
| <code>--size</code>                           | int | 磁盘的实际大小。  | 否  | 是                   | 是                     |
| <code>--sparse</code>                         | 布尔值 | 如果磁盘的物理存储不应预分配，则为 <b>true</b> 。   | 否  | 是                   | 是                     |
| <code>--bootable</code>                       | 布尔值 | 如果此磁盘被标记为可引导，则为 <b>true</b> 。   | 否  | 是                   | 是                     |
| <code>--shareable</code>                      | 布尔值 | 如果此磁盘可以共享，则为 <b>true</b> 。  | 否  | 是                   | 是                     |
| <code>--allow_snapshots</code>                | 布尔值 | 如果此磁盘允许快照，则为 <b>true</b> 。  | 否  | 是                   | 是                     |
| <code>--propagate_errors</code>               | 布尔值 | 如果磁盘错误不应导致虚拟机暂停，且应该向客户端操作系统传播磁盘错误，则为 <b>true</b> 。                              | 否  | 是                   | 是                     |
| <code>--wipe_after_delete</code>              | 布尔值 | 如果删除磁盘时，磁盘的底层物理存储应清零为 <b>true</b> 。这会提高安全性，但是一个更密集的操作，可能会延长删除时间。                | 否  | 是                   | 是                     |
| <code>--storage_domains-storage_domain</code> | 集合  | 为磁盘定义特定的存储域。  | 否  | 是                   | 否                     |

`--storage_domains-storage_domain` 参数是下表中使用子参数的集合。

表 4.7. `--storage_domains-storage_domain` parameters

| 名称                                       | 类型  | Description |
|--|-----|-------------|
| <code>storage_domain.i<br/>d name</code> | 字符串 | 对磁盘存储域的引用。  |

下表列出了基于资源命令的其他磁盘选项。

表 4.8. 附加命令选项

| 选项                           | Description  |
|------------------------------|--|
| <code>--vm-identifier</code> | 将磁盘作为子资源添加到 <b>vm</b> 中。   |
| <code>--alias</code>         | 使用 <b>show</b> 命令时标识磁盘名称。使用 <code>--alias</code> 而不是 <code>--name</code> 参数进行磁盘特定查询。 |

#### 例 4.11. 创建新磁盘

```
[RHEVM shell (connected)]# add disk --name MyDisk --provisioned_size 8589934592 --interface virtio --format cow
```

#### 例 4.12. 更新存储域

```
[RHEVM shell (connected)]# update disk MyDisk --shareable true
```

#### 例 4.13. 删除存储域

```
[RHEVM shell (connected)]# remove disk MyDisk
```

下表列出了虚拟磁盘资源的操作。

表 4.9. 虚拟磁盘操作

| 操作                      | Description  |
|-------------------------|--------------|
| 激活                      | 在虚拟机上激活磁盘。   |
| <code>deactivate</code> | 取消激活虚拟机上的磁盘。 |

## 4.6. GLUSTERVOLUME

**glustervolume** 资源类型对 Red Hat Virtualization 环境中的所有 Gluster 存储卷资源进行分组。

表 4.10. Gluster 卷参数

| 名称                       | 类型  | Description   | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|--------------------------|-----|---|----|---------------------|-----------------------|
| <b>--name</b>            | 字符串 | Gluster 卷的名称。   | 是  | 是                   | 否                     |
| <b>--volume_type</b>     | 字符串 | 定义 Gluster 卷类型。从 <b>DISTRIBUTE、REPLICATE、DISTRIBUTED_REPLICATE、STRIPE</b> 或 <b>DISTRIBUTED_STRIPE</b> 中选择。                        | 是  | 是                   | 否                     |
| <b>--bricks-brick</b>    | 集合  | 新的 Gluster 卷需要一组 Gluster brick 来添加和管理。此参数引用 brick 详细信息的集合。至少指定一个 brick，但为多个 <b>brick</b> 指定多个 <b>brick-brick</b> 参数。有关集合详情，请参见以下。 | 是  | 是                   | 否                     |
| <b>--transport_types</b> | 集合  | 对 Gluster 卷可用传输方法的引用。有关集合详情，请参见以下。  | 否  | 是                   | 否                     |
| <b>--replica_count</b>   | 整数  | 定义复制卷的文件复制数。  | 否  | 是                   | 否                     |
| <b>--stripe_count</b>    | 整数  | 定义条状卷的条带数   | 否  | 是                   | 否                     |
| <b>--options-option</b>  | 集合  | 对 Gluster 卷选项的引用。有关集合详情，请参见以下。  | 否  | 是                   | 否                     |

**--bricks-brick** 参数是一个集合，其使用下表中的子参数。

表 4.11. bricks-brick 参数

| 名称                     | 类型  | Description             |
|------------------------|-----|-------------------------|
| <b>brick.server_id</b> | 字符串 | Gluster 服务器的地址。         |
| <b>brick.brick_dir</b> | 字符串 | Gluster 服务器上的 brick 目录。 |

**--transport\_types** 参数是使用下表中子参数的集合。

表 4.12. transport\_types parameters

| 名称                    | 类型  | Description   |
|-----------------------|-----|---|
| <b>transport_type</b> | 字符串 | 定义要使用的传输类型。为多个类型指定多个 <b>transport_type</b> 参数。从 <b>TCP</b> 和 <b>RDMA</b> 中选择。 |

**--options-option** 参数是下表中使用子参数的集合。

表 4.13. options-option 参数

| 名称                  | 类型  | Description   |
|---------------------|-----|---------------|
| <b>option.name</b>  | 字符串 | Gluster 选项名称。 |
| <b>option.value</b> | 字符串 | Gluster 选项值。  |

下表列出了针对基于资源的命令的其他 **glustervolume** 选项。

表 4.14. 附加命令选项

| 选项                          | Description                       |
|-----------------------------|-----------------------------------|
| <b>--cluster-identifier</b> | 将 Gluster 卷 <b>添加到集群中</b> ，作为子资源。 |

#### 例 4.14. 创建具有两个 brick 的 Gluster 卷

```
[RHEVM shell (connected)]# add glustervolume --cluster-identifier Default --name GlusterVol1 --
volume-type DISTRIBUTE --bricks-brick "brick.server_id=UUID,brick.brick_dir=filepath"--bricks-
brick "brick.server_id=UUID,brick.brick_dir=filepath"
```

#### 例 4.15. 删除 Gluster 卷

```
[RHEVM shell (connected)]# remove glustervolume --cluster-identifier Default --name GlusterVol1
```

下表列出了 **Gluster** 卷资源的操作。

表 4.15. Gluster 卷操作

| 操作           | Description      |
|--------------|------------------|
| <b>start</b> | 使 Gluster 卷可供使用。 |

| 操作                     | Description            |
|------------------------|------------------------|
| <b>stop</b>            | 停用 Gluster 卷。          |
| <b>setOption</b>       | 设置 Gluster 卷选项。        |
| <b>resetOption</b>     | 将 Gluster 卷选项重置为默认值。   |
| <b>resetAllOptions</b> | 将所有 Gluster 卷选项重置为默认值。 |

## 4.7. GROUP

**group** 资源类型定义 Red Hat Virtualization 环境的所有身份服务组。

表 4.16. 组参数

| 名称            | 类型  | Description            | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|---------------|-----|------------------------|----|---------------------|-----------------------|
| <b>--name</b> | 字符串 | 组名称，通常是身份目录服务中的完整组群路径。 | 否  | 否                   | 否                     |

### 例 4.16. 创建组

```
[RHEVM shell (connected)]# add group --name www.example.com/accounts/groups/mygroup --
domain-name example.com
```

## 4.8. 主机

**主机** 资源类型对 Red Hat Virtualization 环境中所有主机资源进行分组。

表 4.17. 主机参数

| 名称               | 类型  | Description    | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|------------------|-----|----------------|----|---------------------|-----------------------|
| <b>--name</b>    | 字符串 | 主机的名称。         | 是  | 是                   | 是                     |
| <b>--address</b> | 字符串 | 主机的 IP 地址或主机名。 | 是  | 是                   | 是                     |

| 名称                                       | 类型  | Description                            | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|--|-----|--|----|---------------------|-----------------------|
| <b>--root_password</b>                   | 字符串 | 主机 <b>root</b> 用户的密码。                  | 是  | 是                   | 是                     |
| <b>--cluster-id name</b>                 | 字符串 | 定义包含主机的集群。                             | 是  | 是                   | 是                     |
| <b>--port</b>                            | int | 与主机上运行的 VDSM 守护进程通信的端口。                | 否  | 是                   | 是                     |
| <b>--storage_manager-priority</b>        | int | 设置存储池管理器(SPM)的主机顺序的优先级。                | 否  | 是                   | 是                     |
| <b>--power_management-type</b>           | 字符串 | 主机中的电源管理设备类型。                          | 否  | 是                   | 是                     |
| <b>--power_management-enabled</b>        | 布尔值 | 指明是否启用或禁用电源管理配置。                       | 否  | 是                   | 是                     |
| <b>--power_management-address</b>        | 字符串 | 电源管理设备的主机名或 IP 地址。                     | 否  | 是                   | 是                     |
| <b>--power_management-user_name</b>      | 字符串 | 电源管理的有效用户名。                            | 否  | 是                   | 是                     |
| <b>--power_management-password</b>       | 字符串 | 用于电源管理的有效、强大的密码。                       | 否  | 是                   | 是                     |
| <b>--power_management-options-option</b> | 集合  | 所选 <b>power_management-type</b> 的隔离选项。 | 否  | 是                   | 是                     |
| <b>--reboot_after_installation</b>       | 布尔值 | 定义主机在 VDSM 安装后是否重新启动。                  | 否  | 是                   | 否                     |

**--power\_management-options-option** 参数是下表中使用子参数的集合。

表 4.18. --power\_management-options-option parameters

| 名称                  | 类型  | Description |
|---------------------|-----|-------------|
| <b>option.name</b>  | 字符串 | 电源管理选项名称。   |
| <b>option.value</b> | 字符串 | 电源管理选项值。    |

**例 4.17. 创建新主机**

```
[RHEVM shell (connected)]# add host --name Host1 --address host1.example.com --
root_password p@55w0rd! --cluster-name Default
```

**例 4.18. 更新主机**

```
[RHEVM shell (connected)]# update host Host1 --name Host2
```

**例 4.19. 删除主机**

```
[RHEVM shell (connected)]# remove host Host1
```

下表列出了主机资源的操作。

表 4.19. 主机操作

| 操作                     | Description      |
|------------------------|------------------|
| <b>激活</b>              | 激活主机。            |
| <b>批准</b>              | 批准主机。            |
| <b>commitnetconfig</b> | 保存网络配置。          |
| <b>deactivate</b>      | 取消激活一个主机。        |
| <b>fence</b>           | 隔离主机。            |
| <b>forceselectspm</b>  | 选择要成为存储池管理程序的主机。 |
| <b>install</b>         | 在主机上安装 VDSM。     |
| <b>iscsidiscover</b>   | 执行 iSCSI 发现命令。   |

| 操作                | Description    |
|-------------------|----------------|
| <b>iscsilogin</b> | 执行 iSCSI 登录命令。 |

下表列出了 **fence** 操作的附加选项。

表 4.20. 隔离选项

| 选项             | Description  |
|----------------|--|
| <b>manual</b>  | 手动隔离主机。使用此操作确认主机变得不响应并手动重启了管理器。                    |
| <b>restart</b> | 重新启动主机，作为 stop, wait, status, start, wait, status. |
| <b>start</b>   | 打开主机。  |
| <b>stop</b>    | 关闭主机。  |
| <b>status</b>  | 检查主机的运行状态。   |

#### 例 4.20. 确认主机已重启

```
[RHEVM shell (connected)]# action host Host1 fence --fence_type manual
```

## 4.9. NETWORK

网络资源类型对 Red Hat Virtualization 环境中的所有逻辑网络资源进行分组。

表 4.21. 逻辑网络参数

| 名称                           | 类型  | Description    | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|------------------------------|-----|----------------|----|---------------------|-----------------------|
| <b>--data_center-id name</b> | 字符串 | 对逻辑网络的数据中心的引用。 | 是  | 是                   | 否                     |
| <b>--name</b>                | 字符串 | 逻辑网络的纯文本名称。    | 是  | 是                   | 否                     |
| <b>--description</b>         | 字符串 | 逻辑网络的描述。       | 否  | 是                   | 是                     |



| 名称                    | 类型  | Description   | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|-----------------------|-----|---|----|---------------------|-----------------------|
| <b>--vlan-id</b>      | 字符串 | VLAN 标签。  | 否  | 是                   | 是                     |
| <b>--ip-address</b>   | 字符串 | 逻辑网络的网桥的 IP 地址。                                     | 否  | 是                   | 是                     |
| <b>--ip-gateway</b>   | 字符串 | 逻辑网络的网桥的网关。   | 否  | 是                   | 是                     |
| <b>--ip-netmask</b>   | 字符串 | 逻辑网络的网桥子网掩码。  | 否  | 是                   | 是                     |
| <b>--display</b>      | 布尔值 | 表示是否逻辑网络用于显示通信使用情况。设置为 <b>true</b> 或 <b>false</b> 。 | 否  | 是                   | 是                     |
| <b>--stp</b>          | 布尔值 | 如果在此网络上启用了生成树协议，则设置为 <b>true</b> 。                  | 否  | 是                   | 是                     |
| <b>--mtu</b>          | int | 为逻辑网络的最大传输单元设置用户定义的值。                               | 否  | 是                   | 是                     |
| <b>--usages-usage</b> | 集合  | 定义逻辑网络的使用参数。  | 否  | 否                   | 是                     |

**--usages-usage** 参数是下表中使用子参数的集合。

表 4.22. usages-usage 参数

| 名称           | 类型  | Description                                 |
|--------------|-----|---|
| <b>usage</b> | 字符串 | 网络的使用情况类型。选项包括 <b>VM</b> 和 <b>DISPLAY</b> 。 |

下表列出了基于资源命令的其他 网络选项。

表 4.23. 附加命令选项

| 选项                          | Description      |
|-----------------------------|------------------|
| <b>--cluster-identifier</b> | 将网络作为子资源 添加到集群中。 |

#### 例 4.21. 创建新网络

```
[RHEVM shell (connected)]# add network --name WebNetwork --datacenter-name Default
```

**例 4.22. 将现有网络附加到集群**

```
[RHEVM shell (connected)]# add network --name WebNetwork --datacenter-name Default --
cluster-identifier Default
```

**例 4.23. 更新网络**

```
[RHEVM shell (connected)]# update network WebNetwork --name DataNetwork
```

**例 4.24. 删除网络**

```
[RHEVM shell (connected)]# remove network WebNetwork
```

**4.10. NIC**

**nic** 资源类型在 Red Hat Virtualization 环境中对网络接口资源进行分组。这些资源充当主机和 vm 资源的子资源，但为每个资源定义不同。本节包含两个带有每个参数的表。

**表 4.24. 主机网络接口参数**

| 名称                               | 类型  | Description  | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|----------------------------------|-----|--|----|---------------------|-----------------------|
| <b>--network-id name</b>         | 字符串 | 对网络的引用（如果有的话），该接口被附加。                              | 是  | 是                   | 是                     |
| <b>--name</b>                    | 字符串 | 主机网络接口的名称，如 <b>eth0</b> 。                          | 是  | 是                   | 是                     |
| <b>--bonding-slaves-host_nic</b> | 集合  | 组成绑定接口的从属网络接口集合。                                   | 否  | 是                   | 是                     |
| <b>--bonding-options-option</b>  | 集合  | 绑定接口的选项列表。每个选项都包含属性 <b>name</b> 和 <b>value</b> 属性。 | 否  | 是                   | 是                     |
| <b>--ip-gateway</b>              | 字符串 | 网络网关的 IP 地址。                                       | 否  | 是                   | 是                     |
| <b>--boot_protocol</b>           | 字符串 | 主机引导时 IP 地址分配的协议，如 <b>dhcp</b> 或 <b>static</b> 。   | 否  | 是                   | 是                     |

| 名称                  | 类型  | Description     | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|---------------------|-----|-----------------|----|---------------------|-----------------------|
| <b>--mac</b>        | 字符串 | 接口的 MAC 地址。     | 否  | 是                   | 是                     |
| <b>--ip-address</b> | 字符串 | 接口的 IP 地址。      | 否  | 是                   | 是                     |
| <b>--ip-netmask</b> | 字符串 | 接口的 IP 地址的子网掩码。 | 否  | 是                   | 是                     |
| <b>--ip-mtu</b>     | int | 接口的最大传输单元。      | 否  | 否                   | 是                     |

表 4.25. 虚拟机网络接口参数

| 名称                                       | 类型  | Description  | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|--|-----|--|----|---------------------|-----------------------|
| <b>--network-id name</b>                 | 字符串 | 对网络的引用（如果有的话），该接口被附加。  | 是  | 是                   | 是                     |
| <b>--name</b>                            | 字符串 | 接口的名称，如 <b>eth0</b> 。  | 是  | 是                   | 是                     |
| <b>--mac-address</b>                     | 字符串 | 接口的 MAC 地址。  | 否  | 是                   | 是                     |
| <b>--interface</b>                       | 字符串 | 定义接口类型，如 <b>e1000</b> 、 <b>virtio</b> 、 <b>rtl8139</b> 和 <b>rtl8139_virtio</b> 。 | 否  | 是                   | 是                     |
| <b>--port_mirroring-networks-network</b> | 集合  | 定义一组要复制(mirror)网络数据的网络数据。  | 否  | 是                   | 是                     |

**--bonding-slaves-host\_nic** 参数是使用下表中子参数的集合。

表 4.26. --bonding-slaves-host\_nic

| 名称                      | 类型  | Description     |
|-------------------------|-----|-----------------|
| <b>host_nic.id name</b> | 字符串 | 对另一个主机 NIC 的引用。 |

**--bonding-options-option** 参数是下表中使用子参数的集合。

表 4.27. --bonding-options-option

| 名称                        | 类型  | Description |
|---------------------------|-----|-------------|
| <code>option.name</code>  | 字符串 | 绑定选项名称。     |
| <code>option.value</code> | 字符串 | 绑定选项值。      |
| <code>type</code>         | 字符串 | 绑定选项类型。     |

`--port_mirroring-networks-network` 参数是下表中使用子参数的集合。

表 4.28. --port\_mirroring-networks-network

| 名称                      | 类型  | Description |
|-------------------------|-----|-------------|
| <code>network.id</code> | 字符串 | 对要镜像的网络的引用。 |

下表列出了基于资源命令的额外 NIC 选项。

表 4.29. 附加命令选项

| 选项                             | Description                |
|--------------------------------|----------------------------|
| <code>--host-identifier</code> | 将 NIC 添加到 <b>主机</b> 作为子资源。 |
| <code>--vm-identifier</code>   | 将 NIC 添加到 <b>vm</b> 作为子资源。 |

#### 例 4.25. 在虚拟机上创建新网络接口

```
[RHEVM shell (connected)]# add nic --vm-identifier MyVM1 --name eth0 --network-name MyNetwork
```

#### 例 4.26. 更新虚拟机上的网络接口

```
[RHEVM shell (connected)]# update nic eth0 --vm-identifier MyVM1 --ip-address 10.5.68.123
```

#### 例 4.27. 删除虚拟机上的网络接口

```
[RHEVM shell (connected)]# remove nic eth0 --vm-identifier MyVM1
```

**例 4.28. 在主机上配置网络绑定**

```
[RHEVM shell (connected)]# add nic --host-identifier MyHost1 --name bond1 --network-name
MyNetwork --bonding-slaves-host_nic host_nic.name=eth0 --bonding-slaves-host_nic
host_nic.name=eth1
```

**例 4.29. 将逻辑网络分配给主机网络接口**

```
[RHEVM shell (connected)]# action nic eth0 attach --host-identifier MyHost1 --network-name
MyNetwork
```

下表列出了主机 NIC 资源的操作。

表 4.30. 主机 NIC 操作

| 操作            | Description  |
|---------------|--------------|
| <b>attach</b> | 将 NIC 附加到主机。 |
| <b>detach</b> | 从主机中分离 NIC。  |

下表列出了虚拟机 NIC 资源的操作。

表 4.31. 虚拟机 NIC 操作

| 操作                | Description    |
|-------------------|----------------|
| <b>激活</b>         | 在虚拟机上激活 NIC。   |
| <b>deactivate</b> | 在虚拟机上取消激活 NIC。 |

#### 4.11. 权限

权限 资源类型对 Red Hat Virtualization 环境中的所有权限资源进行分组。

表 4.32. 权限参数

| 名称                           | 类型            | Description  | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|------------------------------|---------------|--|----|---------------------|-----------------------|
| <b>--user-id, --group-id</b> | 字符串           | 使用权限对用户或组的引用。  | 是  | 是                   | 否                     |
| <b>--role-id</b>             | 字符串           | 对为权限分配的角色引用。   | 是  | 是                   | 否                     |
| <b>--expect</b>              | '201-created' | 请求会变得异步，直到返回预期的 HTTP 标头。对于长时间运行的任务，在任务完成之前，否则会返回的任务。 | 否  | 否                   | 否                     |

下表列出了基于资源命令的额外权限选项。

表 4.33. 附加命令选项

| 选项                                | Description      |
|-----------------------------------|------------------|
| <b>--cluster-identifier</b>       | 在集群中添加权限。        |
| <b>--correlation-id</b>           | 用于跨系统日志记录的标记标识符。 |
| <b>--cpuprofile-identifier</b>    | 为 CPU 配置集添加权限。   |
| <b>--datacenter-identifier</b>    | 为数据中心添加权限。       |
| <b>--disk-identifier</b>          | 为磁盘添加权限。         |
| <b>--diskprofile-identifier</b>   | 为磁盘配置文件添加权限。     |
| <b>--host-identifier</b>          | 为主机添加权限。         |
| <b>--iscsibond-identifier</b>     | 为 iSCSI 绑定添加权限。  |
| <b>--network-identifier</b>       | 向网络添加权限。         |
| <b>--storagedomain-identifier</b> | 为存储域添加权限。        |
| <b>--template-identifier</b>      | 为模板添加权限。         |
| <b>--vm-identifier</b>            | 为虚拟机添加权限。        |
| <b>--vmprofile-identifier</b>     | 为虚拟机池添加权限。       |
| <b>--vnicprofile-identifier</b>   | 为 VNIC 配置集添加权限。  |

**例 4.30. 创建新权限**

```
[RHEVM shell (connected)]# add permission --role-id 00000000-0000-0000-0000-000000000001
--user-id 8b9456ae-e2c8-426e-922d-b01bb8a805fb
```

**4.12. PERMIT**

**permit** 资源类型对 Red Hat Virtualization 环境中角色的所有允许进行分组。

表 4.34. 权限参数

| 名称          | 类型  | Description | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|-------------|-----|-------------|----|---------------------|-----------------------|
| <b>--id</b> | 字符串 | 对允许 添加的引用。  | 是  | 是                   | 否                     |

下表列出了其他 允许基于 资源的命令的选项。

表 4.35. 附加命令选项

| 选项                       | Description     |
|--------------------------|-----------------|
| <b>--role-identifier</b> | 将 permit 添加到角色。 |

**例 4.31. 创建新权限**

```
[RHEVM shell (connected)]# add permit --role-identifier MyRole --id 1
```

**4.13. 配额**

**配额** 资源类型对 Red Hat Virtualization 环境中所有数据中心配额进行分组。

表 4.36. 配额参数

| 名称 | 类型 | Description | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|----|----|-------------|----|---------------------|-----------------------|
|----|----|-------------|----|---------------------|-----------------------|

| 名称                   | 类型  | Description | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|----------------------|-----|-------------|----|---------------------|-----------------------|
| <b>--name</b>        | 字符串 | 配额的名称。      | 是  | 是                   | 是                     |
| <b>--description</b> | 字符串 | 配额的描述。      | 是  | 是                   | 是                     |

#### 4.14. ROLE

角色 资源类型对 Red Hat Virtualization 环境中所有单个角色进行分组。

表 4.37. 角色参数

| 名称                      | 类型  | Description                 | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|-------------------------|-----|-----------------------------|----|---------------------|-----------------------|
| <b>--name</b>           | 字符串 | 角色的名称。                      | 是  | 是                   | 是                     |
| <b>--permits-permit</b> | 集合  | 列表允许角色的初始包含。允许资源类型包含的额外允许。  | 是  | 是                   | 否                     |
| <b>--description</b>    | 字符串 | 角色的描述。                      | 否  | 是                   | 是                     |
| <b>--administrative</b> | 布尔值 | 如果这是管理角色，则设置为 <b>true</b> 。 | 否  | 是                   | 是                     |

**--permits-permit** 参数是下表中使用子参数的集合。

表 4.38. --permits-permit 参数

| 名称               | 类型  | Description     |
|------------------|-----|-----------------|
| <b>permit.id</b> | 字符串 | 对允许添加到角色的允许的引用。 |

#### 例 4.32. 创建新角色

```
[RHEVM shell (connected)]# add role --name MyRole --permits-permit {permit.id: 1;},{permit.id: 2;}
```



## 4.15. SNAPSHOT

快照 资源类型对 Red Hat Virtualization 环境中的所有虚拟机快照资源进行分组。

表 4.39. 快照参数

| 名称                         | 类型  | Description | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|----------------------------|-----|-------------|----|---------------------|-----------------------|
| <code>--description</code> | 字符串 | 快照的描述。      | 是  | 是                   | 否                     |

下表列出了基于资源命令的额外快照选项。

表 4.40. 附加命令选项

| 选项                           | Description              |
|------------------------------|--------------------------|
| <code>--vm-identifier</code> | 将磁盘作为子资源添加到 <b>vm</b> 中。 |

### 例 4.33. 创建新快照

```
[RHEVM shell (connected)]# add snapshot --vm-identifier MyVM --description 'My Snapshot'
```

### 例 4.34. 删除存储域

```
[RHEVM shell (connected)]# remove snapshot [snapshot_id]
```

下表列出了虚拟机快照资源的操作。

表 4.41. 虚拟机快照操作

| 操作                   | Description |
|----------------------|-------------|
| <code>restore</code> | 恢复快照。       |

## 4.16. 统计

统计 资源类型对 Red Hat Virtualization 环境中资源的统计信息进行分组。资源统计数据根据其资源标

标识符列出。

表 4.42. 统计资源标识符

| 选项                                | Description                |
|-----------------------------------|----------------------------|
| <b>--brick-identifier</b>         | 用于查看指定 brick 统计信息的资源标识符。   |
| <b>--cluster-identifier</b>       | 查看指定集群统计信息的资源标识符。          |
| <b>--datacenter-identifier</b>    | 查看指定数据中心统计信息的资源标识符。        |
| <b>--disk-identifier</b>          | 查看指定虚拟磁盘的统计信息的资源标识符。       |
| <b>--glustervolume-identifier</b> | 查看指定 gluster 卷的统计信息的资源标识符。 |
| <b>--host-identifier</b>          | 查看指定主机统计信息的资源标识符。          |
| <b>--job-identifier</b>           | 查看指定作业统计信息的资源标识符。          |
| <b>--nic-identifier</b>           | 查看指定 NIC 统计信息的资源标识符。       |
| <b>--numanode-identifier</b>      | 查看指定 NUMA 节点统计信息的资源标识符。    |
| <b>--step-identifier</b>          | 查看指定步骤统计信息的资源标识符。          |
| <b>--storagedomain-identifier</b> | 查看指定存储域统计信息的资源标识符。         |
| <b>--vm-identifier</b>            | 查看指定虚拟机统计信息的资源标识符。         |

使用 `list` 命令和相关资源标识符，查看每个资源的统计信息集合。以下示例提供了指定主机的可用统计列表：

```
[RHEVM shell (connected)]# list statistics --host-identifier Host_name/id
```

提供的统计数据 的名称或 id 可与 `show` 命令和资源标识符一起使用，以查看指定统计的更多信息。以下示例显示了主机指定统计的详情：

```
[RHEVM shell (connected)]# show statistic statistic_name/id --host-identifier Host name/id
```

#### 4.17. STORAGECONNECTION

**storageconnection** 资源类型允许您添加、编辑和删除存储连接。

表 4.43. 存储连接参数

| 名称                      | 类型            | Description  | 必填   | 用户<br>Crea<br>table | User<br>Upda<br>table |
|-------------------------|---------------|--|--|---------------------|-----------------------|
| <b>--address</b>        | 字符串           | 存储域的主机名或 IP 地址。                                      | 是<br>(仅<br>NFS<br>和<br>iSCSI<br>)                      | 是                   | 是                     |
| <b>--correlation_id</b> | 字符串           | 存储连接的标记标识符。  | 否  | 否                   | 是                     |
| <b>--expect</b>         | '201-created' | 请求会变得异步，直到返回预期的 HTTP 标头。对于长时间运行的任务，在任务完成之前，否则会返回的任务。 | 否  | 否                   | 否                     |
| <b>--iqn</b>            | 字符串           | 存储设备的目标 IQN。   | 是<br>(仅<br>限<br>iSCSI<br>)                             | 是                   | 是                     |
| <b>--mount_options</b>  | 字符串           | 挂载 PosixFS 共享的选项。                                    | 否  | 是                   | 是                     |
| <b>--nfs_retrans</b>    | 整数            | NFS 客户端重新传输的请求数量将尝试完成请求。                             | 否  | 是                   | 是                     |
| <b>--nfs_timeo</b>      | 整数            | NFS 客户端将等待请求完成的时间（以秒为单位）。                            | 否  | 是                   | 是                     |
| <b>--nfs_version</b>    | 字符串           | 使用的 NFS 版本。  | 否  | 是                   | 是                     |
| <b>--password</b>       | 字符串           | 用于登录到 iSCSI 存储域目标的 CHAP 密码。                          | 否  | 是                   | 是                     |
| <b>--path</b>           | 字符串           | 存储域挂载的文件路径。该路径不能更新为存储连接已使用的路径。                       | 是<br>(仅<br>限<br>NFS<br>、<br>local<br>和<br>Posix<br>FS) | 是                   | 是                     |

| 名称   | 类型      | Description                  | 必填                           | 用户<br>Crea<br>table | User<br>Upda<br>table |
|--|---------|------------------------------|------------------------------|---------------------|-----------------------|
| <b>--port</b>                                    | 整数      | 用于 iSCSI 存储域的 TCP 端口。        | 是<br>(仅<br>限<br>iSCSI<br>)   | 是                   | 是                     |
| <b>--<br/>storage<br/>domain-<br/>identifier</b> | 字符<br>串 | 对磁盘存储域的引用。                   | 否                            | 否                   | 否                     |
| <b>--type</b>                                    | 字符<br>串 | 存储域的类型。                      | 是                            | 是                   | 否                     |
| <b>--username</b>                                | 字符<br>串 | 用于登录到 iSCSI 存储域目标的 CHAP 用户名。 | 否                            | 是                   | 是                     |
| <b>--vfs_type</b>                                | 字符<br>串 | Linux 支持的 PosixFS 共享的文件系统类型。 | 是<br>(仅<br>限<br>Posix<br>FS) | 是                   | 是                     |

#### 例 4.35. 创建新存储连接

```
[RHEVM shell (connected)]# add storageconnection --address storage.example.com --path /storage/nfs --type nfs
```

## 4.18. STORAGEDOMAIN

**storagedomain** 资源类型对 Red Hat Virtualization 环境中的所有存储域资源进行分组。

表 4.44. 存储域参数

| 名称              | 类型      | Description  | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|-----------------|---------|--|----|---------------------|-----------------------|
| <b>--name</b>   | 字符<br>串 | 存储域的名称。  | 否  | 是                   | 是                     |
| <b>--format</b> | 布尔<br>值 | 数据中心的元数据格式，包括 <b>v1</b> 、 <b>v 2</b> 或 <b>v3</b> 。 | 是  | 是                   | 否                     |

| 名称                             | 类型  | Description  | 必填    | 用户<br>Crea<br>table | User<br>Upda<br>table |
|--------------------------------|-----|--|-------|---------------------|-----------------------|
| <b>--host-id name</b>          | 字符串 | 对应从中初始化此存储域的主机的引用。此主机的唯一限制是它应当有权访问指定的物理存储。   | 是     | 是                   | 否                     |
| <b>--storage-address</b>       | 字符串 | 存储设备的 IP 地址或主机名。   | 是     | 是                   | 否                     |
| <b>--storage-logical_unit</b>  | 集合  | 存储设备的逻辑单元信息。   | 请参见以下 | 是                   | 否                     |
| <b>--storage-mount_options</b> | 字符串 | 挂载存储域的选项。  | 请参见以下 | 是                   | 否                     |
| <b>--storage-override_luns</b> | 布尔值 | 定义是否覆盖逻辑单元号。状态可以是 <b>true</b> 或 <b>false</b> 。   | 请参见以下 | 是                   | 否                     |
| <b>--storage-path</b>          | 字符串 | 用于存储域的存储设备的路径。   | 请参见以下 | 是                   | 否                     |
| <b>--storage-type</b>          | 字符串 | 数据中心的存储类型，包括 <b>iscsi</b> 、 <b>fcp</b> 、 <b>nfs</b> 、 <b>glusterfs</b> 、 <b>localfs</b> 或 <b>posixfs</b> 。 | 是     | 是                   | 否                     |
| <b>--storage-ufs_type</b>      | 字符串 | 定义存储域的文件系统类型。  | 请参见以下 | 是                   | 否                     |
| <b>--type</b>                  | 字符串 | 存储域的类型，包括 <b>数据</b> 、 <b>iso</b> 和 <b>export</b> 。   | 是     | 是                   | 否                     |

**--storage-logical\_unit** 参数是需要下表中所有子参数的集合。

表 4.45. storage-logical\_unit parameters

| 名称                           | 类型  | Description        |
|------------------------------|-----|--------------------|
| <b>logical_unit.address</b>  | 字符串 | 包含存储设备的服务器地址。      |
| <b>logical_unit.port</b>     | 整数  | 服务器的端口号。           |
| <b>logical_unit.target</b>   | 字符串 | 存储设备的目标 IQN。       |
| <b>logical_unit.username</b> | 字符串 | 用于登录到目标的 CHAP 用户名。 |

| 名称                                    | 类型  | Description       |
|---------------------------------------|-----|-------------------|
| <code>logical_unit.password</code>    | 字符串 | 用于登录到目标的 CHAP 密码。 |
| <code>logical_unit.serial</code>      | 字符串 | 目标的串行 ID。         |
| <code>logical_unit.vendor_id</code>   | 字符串 | 目标的供应商名称。         |
| <code>logical_unit.product_id</code>  | 字符串 | 目标的产品代码。          |
| <code>logical_unit.lun_mapping</code> | 整数  | 目标的逻辑单元号设备映射。     |
| <code>logical_unit.portal</code>      | 字符串 | 逻辑单元门户。           |
| <code>logical_unit.paths</code>       | 整数  | 逻辑单元路径。           |
| <code>logical_unit.id</code>          | 字符串 | 逻辑单元 ID 的引用。      |

根据 存储类型，使用以下参数：

表 4.46. 存储类型参数

| 类型          | 参数  |
|-------------|---|
| nfs         | <code>--storage-address, --storage-path</code>  |
| iSCSI 或 fcp | <code>--storage-address, --storage-logical_unit, --storage-override_luns</code>             |
| glusterfs   | <code>--storage-address, --storage-path, --storage-vfs_type</code>                          |
| local       | <code>--storage-path</code>   |
| posixfs     | <code>--storage-path, --storage-vfs_type, --storage-address, --storage-mount_options</code> |

下表列出了基于资源命令的额外 `storagedomain` 选项。

表 4.47. 附加命令选项

| 选项                                   | Description       |
|--------------------------------------|-------------------|
| <code>--datacenter-identifier</code> | 将存储域作为子资源添加到数据中心。 |

**例 4.36. 创建新存储域**

```
[RHEVM shell (connected)]# add storagedomain --name DataStorage --datacenter-name Default
-type data
```

**例 4.37. 添加 gluster 存储域**

```
[RHEVM shell (connected)]# add storagedomain --type data --storage-type glusterfs --name
RHS_01 --storage-address 192.0.2.0 --storage-path Vol_ONE --storage-vfs_type glusterfs
```

**例 4.38. 更新存储域**

```
[RHEVM shell (connected)]# update storagedomain DataStorage --name DataStorageOld
```

**例 4.39. 删除存储域**

```
[RHEVM shell (connected)]# remove storagedomain DataStorage
```

下表列出了存储域资源的操作。

表 4.48. 存储域操作

| 操作         | Description    |
|------------|----------------|
| 激活         | 在数据中心上激活存储域。   |
| deactivate | 取消激活数据中心上的存储域。 |

**4.19. TAG**

标签 资源类型对 Red Hat Virtualization 环境中的所有标签进行分组。

表 4.49. 标签参数

| 名称     | 类型  | Description | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|--------|-----|-------------|----|---------------------|-----------------------|
| --name | 字符串 | 标签的名称。      | 是  | 是                   | 是                     |

| 名称                   | 类型  | Description   | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|----------------------|-----|---------------|----|---------------------|-----------------------|
| <b>--description</b> | 字符串 | 字符串的描述。       | 是  | 是                   | 是                     |
| <b>--parent-name</b> | 字符串 | 对标签附加的父标签的引用。 | 是  | 是                   | 是                     |

#### 例 4.40. 创建新标签

```
[RHEVM shell (connected)]# add tag --name MyTag --description "A virtual machine tag" --parent MyParentTag
```

## 4.20. 模板

模板 资源类型对 Red Hat Virtualization 环境中的所有虚拟机模板进行分组。只有 **--vm-id|name** 和 **--name** 是必需的参数。如果没有指定可选参数，则模板将从用于制作模板的虚拟机中继承设置。

表 4.50. 模板参数

| 名称                                 | 类型   | Description  | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|------------------------------------|------|--|----|---------------------|-----------------------|
| <b>--vm-id name</b>                | 字符串  | 对用作模板基础的虚拟机的引用。  | 是  | 是                   | 否                     |
| <b>--name</b>                      | 字符串  | 虚拟机模板的名称。  | 是  | 是                   | 是                     |
| <b>--memory</b>                    | long | 虚拟机模板的内存量，以字节为单位。  | 否  | 是                   | 是                     |
| <b>--cpu-topology-cores</b>        | int  | 虚拟机模板可用的 CPU 内核数。  | 否  | 是                   | 是                     |
| <b>--high-availability-enabled</b> | 布尔值  | 设置为 <b>true</b> ，以便为虚拟机模板启用高可用性。                               | 否  | 是                   | 是                     |
| <b>--os-cmdline</b>                | 字符串  | 与定义的内核一起使用的内核命令行参数字符串。这个选项支持直接引导 Linux 内核，而不是通过 BIOS 引导装载程序引导。 | 否  | 是                   | 是                     |
| <b>--origin</b>                    | 字符串  | 虚拟机模板的源。指定 <b>rhev</b> 、 <b>VMware</b> 或 <b>xen</b> 。          | 否  | 是                   | 是                     |



| 名称                                  | 类型  | Description  | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|-------------------------------------|-----|--|----|---------------------|-----------------------|
| <b>--high_availability-priority</b> | int | 设置每个虚拟机模板高可用性的优先级值（如引导顺序）。   | 否  | 是                   | 是                     |
| <b>--timezone</b>                   | 字符串 | Windows 虚拟机模板的 Sysprep 时区设置。指定格式，如 <b>GMT+00:00</b> 。                              | 否  | 是                   | 是                     |
| <b>--domain-name</b>                | 字符串 | 虚拟机模板的域名。  | 否  | 是                   | 是                     |
| <b>--type</b>                       | 字符串 | 定义虚拟机类型。指定 <b>desktop</b> 或 <b>server</b> 。  | 否  | 是                   | 是                     |
| <b>--stateless</b>                  | 布尔值 | 如果生成的虚拟机是无状态的，则设置为 <b>true</b> 。无状态虚拟机包含引导时获取的磁盘镜像的快照，并在关机时删除。这意味着状态更改在重启后不会保留。    | 否  | 是                   | 是                     |
| <b>--delete_protected</b>           | 布尔值 | 设置为 <b>true</b> 以便无法删除从此模板创建的虚拟机。  | 否  | 是                   | 是                     |
| <b>--sso-methods-method</b>         | 集合  | 定义所用的单点登录方法。例如， <b>--sso-methods-method method.id=GUEST_AGENT</b> 。                | 否  | 是                   | 是                     |
| <b>--rng_device-rate-bytes</b>      | int | 指定每个期间内允许消耗的字节数。   | 否  | 是                   | 是                     |
| <b>--rng_device-rate-period</b>     | int | 以毫秒为单位指定期间的持续时间。如果指定，则必须指定 <b>--rng_device-rate-bytes</b> 。                        | 否  | 是                   | 是                     |
| <b>--rng_device-source</b>          | 字符串 | 随机数字生成器的来源。指定 <b>random</b> 或 <b>hwrng</b> 。                                       | 否  | 是                   | 是                     |
| <b>--console-enabled</b>            | 布尔值 | 设置为 <b>true</b> 以启用 VirtIO 控制台设备功能。  | 否  | 是                   | 是                     |
| <b>--placement_policy-affinity</b>  | 字符串 | 从模板创建的每个虚拟机的迁移关联性。指定可 <b>migratable</b> 、 <b>user_migratable</b> 或 <b>pinned</b> 。 | 否  | 是                   | 是                     |
| <b>--description</b>                | 字符串 | 虚拟机模板的描述。  | 否  | 是                   | 是                     |
| <b>--comment</b>                    | 字符串 | 虚拟机模板的评论。  | 否  | 是                   | 是                     |

| 名称   | 类型  | Description  | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|--|-----|--|----|---------------------|-----------------------|
| <code>--custom-properties-custom_property</code> | 集合  | 一组用户定义的环境变量，作为参数传递给自定义脚本。  | 否  | 是                   | 是                     |
| <code>--os-type</code>                           | 字符串 | 虚拟机模板的操作系统类型。  | 否  | 是                   | 是                     |
| <code>--os-boot</code>                           | 集合  | 虚拟机模板的引导设备。指定 <code>cdrom</code> 、 <code>hd</code> 或 <code>网络</code> 。<br>例如： <code>--os-boot boot.dev=hd</code> 。 | 否  | 是                   | 是                     |
| <code>--cpu-topology-sockets</code>              | int | 虚拟机模板可用的 CPU 插槽数量。   | 否  | 是                   | 是                     |
| <code>--cpu_shares</code>                        | int | 虚拟机相对于其他虚拟机可能需要的 CPU 资源级别。<br>例如， <b>512</b> 代表低优先级虚拟机， <b>1024</b> 代表中等优先级虚拟机， <b>2048</b> 个高优先级虚拟机。              | 否  | 是                   | 是                     |
| <code>--cpu-architecture</code>                  | 字符串 | 定义 CPU 架构。指定 <code>x86_64</code> 、 <code>ppc64</code> 或未定义。  | 否  | 是                   | 是                     |
| <code>--os-kernel</code>                         | 字符串 | 所生成的虚拟机配置为引导的内核镜像的路径。这个选项支持直接引导 Linux 内核，而不是通过 BIOS 引导装载程序引导。  | 否  | 是                   | 是                     |
| <code>--display-type</code>                      | 字符串 | 定义显示类型。指定 <code>spice</code> 或 <code>vnc</code> 。  | 否  | 是                   | 是                     |
| <code>--display-monitors</code>                  | int | 定义可用显示的数量。   | 否  | 是                   | 是                     |
| <code>--display-single_qxl_pci</code>            | 布尔值 | 设置为 <code>true</code> ，以使用单个虚拟 PCI 设备驱动多个监视器。  | 否  | 是                   | 是                     |
| <code>--display-allow_override</code>            | 布尔值 | 设置为 <code>true</code> 以允许覆盖模板控制台设置。  | 否  | 是                   | 是                     |
| <code>--display-smartcard_enabled</code>         | 布尔值 | 设置为 <code>true</code> ，为虚拟机启用智能卡功能。  | 否  | 是                   | 是                     |
| <code>--display-file_transfer_enabled</code>     | 布尔值 | 设置为 <code>true</code> 以启用 SPICE 文件传输。  | 否  | 是                   | 是                     |

| 名称  | 类型  | Description   | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|---|-----|---|----|---------------------|-----------------------|
| <code>--display-copy_paste_enabled</code> | 布尔值 | 设置为 <b>true</b> 以启用 SPICE 剪贴板复制和粘贴。   | 否  | 是                   | 是                     |
| <code>--display-keyboard_layout</code>    | 字符串 | 定义虚拟机的键盘布局。这个选项只在使用 VNC 协议时可用。指定格式，如 <b>en-US</b> 。   | 否  | 是                   | 是                     |
| <code>--os-initRd</code>                  | 字符串 | 与指定内核一起使用的 <code>initrd</code> 镜像的路径。这个选项支持直接引导 Linux 内核，而不是通过 BIOS 引导装载程序引导。   | 否  | 是                   | 是                     |
| <code>--usb-enabled</code>                | 布尔值 | 设置为 <b>true</b> 以在虚拟机上启用 USB 支持。此选项仅适用于使用 SPICE 协议的虚拟机。   | 否  | 是                   | 是                     |
| <code>--usb-type</code>                   | 字符串 | 如果启用 USB 支持，请定义 USB 类型。指定 <b>Legacy</b> 或 <b>Native</b> 。<br><br> <b>重要</b><br><br>Legacy USB 选项已弃用，并将在 Red Hat Virtualization 4.1 中删除。 | 否  | 是                   | 是                     |
| <code>--tunnel_migration</code>           | 布尔值 | 设置为 <b>true</b> ，以通过 <code>libvirt</code> 守护进程启用数据传输。隧道的传输使用更强大的加密算法，但会增加传输过程中的数据负载。  | 否  | 是                   | 是                     |
| <code>--migration_downtime</code>         | int | 定义虚拟机在实时迁移期间可以停机的最大毫秒数。   | 否  | 是                   | 是                     |
| <code>--virtio_scsi-enabled</code>        | 布尔值 | 设置为 <b>true</b> 以允许将 VirtIO 控制台设备附加到虚拟机。  | 否  | 是                   | 是                     |
| <code>--soundcard_enabled:</code>         | 布尔值 | 设置为 <b>true</b> 以启用声音卡。   | 否  | 是                   | 是                     |
| <code>--vm-disks-disk</code>              | 集合  | 对附加到模板的磁盘的引用。   | 否  | 是                   | 否                     |
| <code>--id</code>                         | 字符串 | 虚拟机模板的 ID。  | 否  | 是                   | 是                     |
| <code>--permissions-clone</code>          | 布尔值 | 设置为 <b>true</b> ，以将源虚拟机的权限复制到模板。  | 否  | 是                   | 是                     |

| 名称                                      | 类型            | Description   | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|---|---------------|---|----|---------------------|-----------------------|
| <code>--version-version_name</code>     | 字符串           | 与 <code>--version-base_template-id</code> 参数一起使用。定义子模板的名称。  | 否  | 是                   | 是                     |
| <code>--version-base_template-id</code> | 字符串           | 定义要用作 root 模板的模板 ID。如果您要将此模板创建为根模板的子模板，则使用。   | 否  | 是                   | 是                     |
| <code>--cpu-cpu_tune-vcpu_pin</code>    | 集合            | 定义要固定到主机的物理 CPU 的虚拟机 CPU。   | 否  | 是                   | 是                     |
| <code>--serial_number-policy</code>     | 字符串           | 定义虚拟机模板的序列号策略。指定 <b>主机</b> 、 <b>vm</b> 或 <b>自定义</b> 。如果使用 <b>custom</b> ，也使用 <code>--serial_number-value</code> 定义序列号值。 | 否  | 是                   | 是                     |
| <code>--serial_number-value</code>      | 字符串           | 定义虚拟机模板的序列号。  | 否  | 是                   | 是                     |
| <code>--bios-boot_menu-enabled</code>   | 布尔值           | 设置为 <b>true</b> 以启用引导菜单。  | 否  | 是                   | 是                     |
| <code>--cluster-id</code>               | 字符串           | 通过指定集群 ID 来定义要使用的集群。  | 否  | 是                   | 是                     |
| <code>--cluster-name</code>             | 字符串           | 通过指定集群名称来定义要使用的集群。  | 否  | 是                   | 是                     |
| <code>--cpu_profile-id</code>           | 字符串           | 定义要使用的 CPU 配置集。使用 <code>list cpuprofiles</code> 命令检索 CPU 配置集 ID 的完整列表。  | 否  | 是                   | 是                     |
| <code>--expect</code>                   | '201-created' | 请求会变得异步，直到返回预期的 HTTP 标头。对于长时间运行的任务，在任务完成之前，否则会返回的任务。  | 否  | 是                   | 是                     |
| <code>--correlation_id</code>           | 字符串           | 用于跨系统日志记录的操作标记标识符。如果客户端没有定义标识符，则会生成一个。  | 否  | 是                   | 是                     |

`--sso-methods-method` 参数是下表中使用子参数的集合。

表 4.51. `--SSO-methods-method` 参数

| 名称                     | 类型  | Description                     |
|------------------------|-----|---------------------------------|
| <code>method.id</code> | 字符串 | 使用的单点登录方法： <b>GUEST_AGENT</b> 。 |

`--custom_properties-custom_property` 参数是下表中使用子参数的集合。

表 4.52. `--custom_properties-custom_property` parameters

| 名称                                 | 类型  | Description |
|------------------------------------|-----|-------------|
| <code>custom_property.name</code>  | 字符串 | 自定义属性名称。    |
| <code>custom_property.value</code> | 字符串 | 自定义属性值。     |

`--os-boot` 参数是一个集合，其使用下表中的子参数。

表 4.53. `--os-boot` 参数

| 名称                    | 类型  | Description  |
|-----------------------|-----|--|
| <code>boot.dev</code> | 字符串 | 虚拟机模板的引导设备。指定 <code>cdrom</code> 、 <code>hd</code> 或 <code>网络</code> 。 |

`--vm-disks-disk` 参数是下表中使用子参数的集合。

表 4.54. `--vm-disks-disk` parameters

| 名称  | 类型  | Description     |
|---|-----|-----------------|
| <code>disk.id</code>                        | 字符串 | 对虚拟磁盘的引用。       |
| <code>storage_domains.storage_domain</code> | 集合  | 为磁盘的存储域定义一组子参数。 |

`--cpu-cpu_tune-vcpu_pin` 参数是下表中使用子参数的集合。

表 4.55. `--cpu-cpu_tune-vcpu_pin` parameters

| 名称                            | 类型  | Description |
|-------------------------------|-----|-------------|
| <code>vcpu_pin.vcpu</code>    | int | 要分配的虚拟 CPU。 |
| <code>vcpu_pin.cpu_set</code> | 字符串 | 主机上的物理 CPU。 |

#### 例 4.41. 创建新模板

-

```
[RHEVM shell (connected)]# add template --name MyTemplate1 --vm-name MyVM1
```

#### 例 4.42. 更新模板

```
[RHEVM shell (connected)]# update template MyTemplate1 --memory 1073741824
```

#### 例 4.43. 删除模板

```
[RHEVM shell (connected)]# remove template MyTemplate1
```

下表列出了虚拟机模板资源的操作。

表 4.56. 虚拟机模板操作

| 操作            | Description  |
|---------------|--------------|
| <b>export</b> | 将模板导出到导出存储域。 |

## 4.21. USER

**user** 资源类型对 Red Hat Virtualization 环境中的所有用户进行分组。

表 4.57. 用户参数

| 名称                      | 类型  | Description | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|-------------------------|-----|-------------|----|---------------------|-----------------------|
| <b>--user_name</b>      | 字符串 | 来自目录服务的用户名。 | 是  | 是                   | 否                     |
| <b>--domain-id name</b> | 字符串 | 对目录服务域的引用。  | 是  | 是                   | 否                     |

#### 例 4.44. 创建新用户

```
[RHEVM shell (connected)]# add user --user_name jsmith --domain-name example.com
```

## 4.22. VM

vm 资源类型对 Red Hat Virtualization 环境中的所有虚拟机资源进行分组。

表 4.58. 虚拟机参数

| 名称   | 类型  | Description  | 必填 | 用户<br>Crea<br>table | User<br>Upd<br>atabl<br>e |
|--|-----|--|----|---------------------|---------------------------|
| <code>--name</code>                              | 字符串 | 虚拟机的名称   | 是  | 是                   | 是                         |
| <code>--template-id name</code>                  | 字符串 | 对用作虚拟机基础的模板的引用。  | 是  | 是                   | 否                         |
| <code>--cluster-id name</code>                   | 字符串 | 对包含此虚拟机的集群的引用。   | 是  | 是                   | 是                         |
| <code>--instance_type-id name</code>             | 字符串 | 定义实例类型。指定自定义、大、medium、small、tiny 或 xlarge。                       | 否  | 是                   | 是                         |
| <code>--quota-id</code>                          | 字符串 | 对虚拟机配额使用量的引用。  | 否  | 是                   | 否                         |
| <code>--timezone</code>                          | 字符串 | Windows 虚拟机的 Sysprep 时区设置。                                       | 否  | 是                   | 是                         |
| <code>--os-boot</code>                           | 集合  | 虚拟机的引导设备。指定 <b>cdrom</b> 、 <b>hd</b> 或 <b>网络</b> 。               | 否  | 是                   | 是                         |
| <code>--custom_properties-custom_property</code> | 集合  | 一组用户定义的环境变量，作为参数传递给自定义脚本。  | 否  | 是                   | 是                         |
| <code>--os-type</code>                           | 字符串 | 此虚拟机的操作系统类型。   | 否  | 是                   | 是                         |
| <code>--usb-enabled</code>                       | 布尔值 | 定义虚拟机的 USB 策略。设置为 <b>true</b> 以在虚拟机上启用 USB。                      | 否  | 是                   | 是                         |
| <code>--usb-type</code>                          | 字符串 | 定义 USB 类型（如果启用）。   | 否  | 是                   | 是                         |
| <code>--type</code>                              | 字符串 | 定义虚拟机类型。指定 <b>desktop</b> 或 <b>server</b> 。                      | 否  | 是                   | 是                         |
| <code>--os-initRd</code>                         | 字符串 | 与指定内核一起使用的 initrd 镜像的路径。这个选项支持直接引导 Linux 内核，而不是通过 BIOS 引导装载程序引导。 | 否  | 是                   | 是                         |
| <code>--display-monitors</code>                  | int | 定义可用显示的数量。   | 否  | 是                   | 是                         |
| <code>--display-single_qxl_pci</code>            | 布尔值 | 设置为 <b>true</b> ，以使用单个虚拟 PCI 设备驱动多个监视器。                          | 否  | 是                   | 是                         |

| 名称                                     | 类型   | Description  | 必填 | 用户<br>Crea<br>table | User<br>Upd<br>atabl<br>e |
|--|------|--|----|---------------------|---------------------------|
| <b>--display-type</b>                  | 字符串  | 定义显示类型。指定 <b>spice</b> 或 <b>vnc</b> 。                          | 否  | 是                   | 是                         |
| <b>--display-allow_override</b>        | 布尔值  | 设置为 <b>true</b> 以允许覆盖虚拟机控制台设置。                                 | 否  | 是                   | 是                         |
| <b>--display-smartcard_enabled</b>     | 布尔值  | 设置为 <b>true</b> 以启用智能卡功能。                                      | 否  | 是                   | 是                         |
| <b>--display-file_transfer_enabled</b> | 布尔值  | 设置为 <b>true</b> 以启用 SPICE 文件传输。                                | 否  | 是                   | 是                         |
| <b>--display-copy_paste_enabled</b>    | 布尔值  | 设置为 <b>true</b> 以启用 SPICE 剪贴板复制和粘贴。                            | 否  | 是                   | 是                         |
| <b>--display-keyboard_layout</b>       | 字符串  | 定义虚拟机的键盘布局。这个选项只在使用 VNC 协议时可用。指定格式，如 <b>en-US</b> 。            | 否  | 是                   | 是                         |
| <b>--os-cmdline</b>                    | 字符串  | 与定义的内核一起使用的内核命令行参数字符串。这个选项支持直接引导 Linux 内核，而不是通过 BIOS 引导装载程序引导。 | 否  | 是                   | 是                         |
| <b>--cpu-topology-cores</b>            | int  | 虚拟机可用的 CPU 内核数。  | 否  | 是                   | 是                         |
| <b>--cpu-architecture</b>              | 字符串  | 定义 CPU 架构。指定 <b>x86_64</b> 、 <b>ppc64</b> 或未定义。                | 否  | 是                   | 是                         |
| <b>--memory</b>                        | long | 虚拟机的内存量，以字节为单位。  | 否  | 是                   | 是                         |
| <b>--memory_policy-guaranteed</b>      | long | 在主机上保证最小内存量（以字节为单位），以便虚拟机运行。                                   | 否  | 是                   | 是                         |
| <b>--memory_policy-ballooning</b>      | 布尔值  | 设置为 <b>true</b> 以启用内存气球设备。                                     | 否  | 是                   | 是                         |
| <b>--high_availability-priority</b>    | int  | 使用高可用性设置每个虚拟机的优先级值（迁移和重启顺序）。                                   | 否  | 是                   | 是                         |



| 名称                                       | 类型  | Description  | 必填 | 用户<br>Crea<br>table | User<br>Upd<br>atabl<br>e |
|--|-----|--|----|---------------------|---------------------------|
| <code>--high_availability-enabled</code> | 布尔值 | 定义是否为虚拟机启用高可用性。  | 否  | 是                   | 是                         |
| <code>--domain-name</code>               | 字符串 | 虚拟机的域名。  | 否  | 是                   | 是                         |
| <code>--description</code>               | 字符串 | 虚拟机的描述。  | 否  | 是                   | 是                         |
| <code>--comment</code>                   | 字符串 | 虚拟机的评论。  | 否  | 是                   | 是                         |
| <code>--stateless</code>                 | 布尔值 | 如果虚拟机无状态，则设置为 <b>true</b> 。无状态虚拟机包含引导时获取的磁盘镜像的快照，并在关机时删除。这意味着状态更改在重启后不会保留。 | 否  | 是                   | 是                         |
| <code>--permissions-clone</code>         | 布尔值 | 设置为 <b>true</b> ，以将源虚拟机的权限复制到模板。   | 否  | 是                   | 是                         |
| <code>--delete-protected</code>          | 布尔值 | 设置为 <b>true</b> 以便无法删除从此模板创建的虚拟机。  | 否  | 是                   | 是                         |
| <code>--sso-methods-method</code>        | 集合  | 定义所用的单点登录方法。例如， <code>--sso-methods-method method.id=GUEST_AGENT</code> 。  | 否  | 是                   | 是                         |
| <code>--rng_device-rate-bytes</code>     | int | 指定每个期间内允许消耗的字节数。   | 否  | 是                   | 是                         |
| <code>--rng_device-rate-period</code>    | int | 以毫秒为单位指定期间的持续时间。如果指定，则必须指定 <code>--rng_device-rate-bytes</code> 。          | 否  | 是                   | 是                         |
| <code>--rng_device-source</code>         | 字符串 | 随机数字生成器的来源。指定 <b>random</b> 或 <b>hwrng</b> 。                               | 否  | 是                   | 是                         |
| <code>--console-enabled</code>           | 布尔值 | 设置为 <b>true</b> 以启用 VirtIO 控制台设备功能。  | 否  | 是                   | 是                         |
| <code>--cpu-mode</code>                  | 字符串 | 定义 CPU 模式。指定自定义、 <code>host_model</code> 或 <code>host_passthrough</code> 。 | 否  | 是                   | 是                         |

| 名称                                     | 类型  | Description   | 必填 | 用户<br>Crea<br>table | User<br>Upd<br>atabl<br>e |
|--|-----|---|----|---------------------|---------------------------|
| <b>--cpu-topology-sockets</b>          | int | 虚拟机可用的 CPU 插槽数量。  | 否  | 是                   | 是                         |
| <b>--cpu-shares</b>                    | int | 虚拟机相对于其他虚拟机可能需要的 CPU 资源级别。例如， <b>512</b> 代表低优先级虚拟机， <b>1024</b> 代表中等优先级虚拟机， <b>2048</b> 个高优先级虚拟机。 | 否  | 是                   | 是                         |
| <b>--placement-policy-affinity</b>     | 字符串 | 每个虚拟机的迁移关联性。指定可 <b>migratable</b> 、 <b>user_migratable</b> 或 <b>pinned</b> 。                      | 否  | 是                   | 是                         |
| <b>--placement-policy-host-id name</b> | 字符串 | 对迁移关联性的首选主机的引用。   | 否  | 是                   | 是                         |
| <b>--origin</b>                        | 字符串 | 虚拟机的源。指定 <b>rhev</b> 、 <b>VMware</b> 或 <b>xen</b> 。   | 否  | 是                   | 是                         |
| <b>--os-kernel</b>                     | 字符串 | 为虚拟机配置的内核镜像的路径。这个选项支持直接引导 Linux 内核，而不是通过 BIOS 引导装载程序引导。   | 否  | 是                   | 是                         |
| <b>--disks-clone</b>                   | 布尔值 | 定义是否从定义的 <b>模板</b> 克隆磁盘。  | 否  | 是                   | 否                         |
| <b>--disks-disk</b>                    | 集合  | 对附加到虚拟机的磁盘引用。   | 否  | 是                   | 是                         |
| <b>--tunnel-migration</b>              | 布尔值 | 设置为 <b>true</b> ，以通过 <b>libvirt</b> 守护进程启用数据传输。隧道的传输使用更强大的加密算法，但会增加传输过程中的数据负载。                    | 否  | 是                   | 是                         |
| <b>--migration-downtime</b>            | int | 定义虚拟机在实时迁移期间可以停机的最大毫秒数。   | 否  | 是                   | 是                         |
| <b>--virtio-scsi-enabled</b>           | 布尔值 | 设置为 <b>true</b> 以允许将 VirtIO 控制台设备附加到虚拟机。  | 否  | 是                   | 是                         |
| <b>--soundcard-enabled:</b>            | 布尔值 | 设置为 <b>true</b> 以启用声音卡。   | 否  | 是                   | 是                         |

| 名称   | 类型            | Description   | 必填 | 用户<br>Crea<br>table | User<br>Upd<br>atabl<br>e |
|--|---------------|---|----|---------------------|---------------------------|
| <b>--payloads-payload</b>                  | 集合            | 定义在启动时要发送到虚拟机的内容。   | 否  | 是                   | 是                         |
| <b>--initialization-configuration-type</b> | 字符串           | 定义虚拟机格式。仅接受 <b>ovf</b> 。  | 否  | 是                   | 是                         |
| <b>--initialization-configuration-data</b> | 字符串           | 这个参数必须与 <b>--initialization-configuration-type</b> 参数匹配。仅接受 <b>ovf</b> 。  | 否  | 是                   | 是                         |
| <b>--cpu-cpu_tune-vcpu_pin</b>             | 集合            | 定义要固定到主机的物理 CPU 的虚拟机 CPU。   | 否  | 是                   | 是                         |
| <b>--serial_number-policy</b>              | 字符串           | 定义虚拟机模板的序列号策略。指定 <b>主机</b> 、 <b>vm</b> 或 <b>自定义</b> 。如果使用 <b>custom</b> ，也使用 <b>--serial_number-value</b> 定义序列号值。 | 否  | 是                   | 是                         |
| <b>--serial_number-value</b>               | 字符串           | 定义虚拟机模板的序列号。  | 否  | 是                   | 是                         |
| <b>--bios-boot_menu-enabled</b>            | 布尔值           | 设置为 <b>true</b> 以启用引导菜单。  | 否  | 是                   | 是                         |
| <b>--numa_tune_mode</b>                    | 字符串           | 定义如何为 NUMA 主机上的域进程分配内存。指定 <b>interleave</b> 、 <b>严格</b> 或 <b>首选</b> 。如果未指定值，则参数默认为 <b>strict</b> 。                | 否  | 是                   | 是                         |
| <b>--cpu_profile-id</b>                    | 字符串           | 定义要使用的 CPU 配置集。使用 <b>list cpuprofiles</b> 命令检索 CPU 配置集 ID 的完整列表。  | 否  | 是                   | 是                         |
| <b>--expect</b>                            | '201-created' | 请求会变得异步，直到返回预期的 HTTP 标头。对于长时间运行的任务，在任务完成之前，否则会返回的任务。  | 否  | 否                   | 否                         |
| <b>--correlation_id</b>                    | 字符串           | 用于跨系统日志记录的操作标记标识符。如果客户端没有定义标识符，则会生成一个。  | 否  | 是                   | 否                         |

**--os-boot** 参数是一个集合，其使用下表中的子参数。

表 4.59. --os-boot 参数

| 名称              | 类型  | Description  |
|-----------------|-----|--|
| <b>boot.dev</b> | 字符串 | 虚拟机模板的引导设备。指定 <b>cdrom</b> 、 <b>hd</b> 或 <b>网络</b> 。 |

**--custom\_properties-custom\_property** 参数是下表中使用子参数的集合。

表 4.60. --custom\_properties-custom\_property parameters

| 名称                           | 类型  | Description |
|------------------------------|-----|-------------|
| <b>custom_property.name</b>  | 字符串 | 自定义属性名称。    |
| <b>custom_property.value</b> | 字符串 | 自定义属性值。     |

**--sso-methods-method** 参数是下表中使用子参数的集合。

表 4.61. --SSO-methods-method 参数

| 名称               | 类型  | Description                     |
|------------------|-----|---------------------------------|
| <b>method.id</b> | 字符串 | 使用的单点登录方法： <b>GUEST_AGENT</b> 。 |

**--disks-disk** 参数是下表中使用子参数的集合。

表 4.62. --disks-disk 参数

| 名称                                    | 类型  | Description     |
|---------------------------------------|-----|-----------------|
| <b>disk.id</b>                        | 字符串 | 对虚拟磁盘的引用。       |
| <b>storage_domains.storage_domain</b> | 集合  | 为磁盘的存储域定义一组子参数。 |

**--payloads-payload** 参数是下表中使用子参数的集合。

表 4.63. --payloads-payload parameters

| 名称                          | 类型  | Description                                |
|-----------------------------|-----|--|
| <b>payload.type</b>         | 字符串 | 有效负载交付类型。指定 <b>cdrom</b> 或 <b>floppy</b> 。 |
| <b>payload.file.name</b>    | 字符串 | 有效负载文件名和虚拟机根文件系统中的位置。                      |
| <b>payload.file.content</b> | 字符串 | 要提供给该文件的内容。                                |

**--cpu-cpu\_tune-vcpu\_pin** 参数是下表中使用子参数的集合。

表 4.64. **--cpu-cpu\_tune-vcpu\_pin**

| 名称                      | 类型  | Description |
|-------------------------|-----|-------------|
| <b>vcpu_pin.vcpu</b>    | int | 要分配的虚拟 CPU。 |
| <b>vcpu_pin.cpu_set</b> | 字符串 | 主机上的物理 CPU。 |

#### 例 4.45. 创建新虚拟机

```
[RHEVM shell (connected)]# add vm --name MyVM --template-name Blank --cluster-name Default --memory 536870912
```

#### 例 4.46. 更新虚拟机

```
[RHEVM shell (connected)]# update vm MyVM --memory 1073741824
```

#### 例 4.47. 删除虚拟机

```
[RHEVM shell (connected)]# remove vm MyVM
```

下表列出了虚拟机资源的操作。

表 4.65. 虚拟机操作

| 操作           | Description |
|--------------|-------------|
| <b>start</b> | 启动虚拟机。      |
| <b>stop</b>  | 停止虚拟机。      |

| 操作              | Description               |
|-----------------|---------------------------|
| <b>shutdown</b> | 关闭虚拟机。                    |
| <b>suspend</b>  | 暂停虚拟机。                    |
| <b>detach</b>   | 从池中分离虚拟机。                 |
| <b>migrate</b>  | 迁移虚拟机到另一主机上。              |
| <b>取消迁移</b>     | 停止正在进行的迁移。                |
| <b>export</b>   | 将虚拟机导出到导出存储域。             |
| <b>Move</b>     | 将虚拟磁盘移动到另一个存储域。           |
| <b>ticket</b>   | 为控制台访问创建一个 ticket。        |
| <b>logon</b>    | 启用用户登录，以使用第三方应用程序进行控制台访问。 |

#### 4.23. VMPOOL

**vmpool** 资源类型对 Red Hat Virtualization 环境中所有虚拟机池资源进行分组。

表 4.66. 虚拟机池参数

| 名称                        | 类型  | Description  | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|---------------------------|-----|--------------|----|---------------------|-----------------------|
| <b>--cluster-id name</b>  | 字符串 | 对虚拟机池的集群的引用。 | 是  | 是                   | 是                     |
| <b>--template-id name</b> | 字符串 | 对虚拟机池的模板的引用。 | 是  | 是                   | 是                     |
| <b>--name</b>             | 字符串 | 虚拟机池的名称。     | 是  | 是                   | 是                     |
| <b>--size</b>             | 整数  | 池中的虚拟机数量。    | 是  | 是                   | 是                     |

#### 例 4.48. 创建新虚拟机池

```
[RHEVM shell (connected)]# add vmpool --cluster-name MyCluster --template-name MyTemplate
--name MyPool --size 3
```

**例 4.49. 更新虚拟机池**

```
[RHEVM shell (connected)]# update vmpool MyPool --size 4
```

**例 4.50. 删除虚拟机池**

```
[RHEVM shell (connected)]# remove vmpool MyPool
```

**4.24. VNICPROFILE**

`vnicprofile` 资源类型对所有 **VNIC**（虚拟网络接口控制器）配置集进行分组，也称为 **Red Hat Virtualization** 环境中的虚拟机（虚拟机）接口配置集。

**表 4.67. 虚拟网络接口控制器配置集参数**

| 名称   | 类型            | Description   | 必填 | 用户<br>Crea<br>table | User<br>Upda<br>table |
|--|---------------|---|----|---------------------|-----------------------|
| <code>--name</code>                              | 字符串           | VNIC 配置集的名称。  | 是  | 是                   | 是                     |
| <code>--network-id</code>                        | 字符串           | 对要应用配置文件的逻辑网络的引用。   | 是  | 否                   | 否                     |
| <code>--correlation_id</code>                    | 字符串           | 用于跨系统日志记录的操作标记标识符。如果客户端没有定义标识符，则会生成一个。                            | 否  | 是                   | 否                     |
| <code>--description</code>                       | 字符串           | 配置文件的描述。  | 否  | 是                   | 是                     |
| <code>--expect</code>                            | '201-created' | 请求会变得异步，直到返回预期的 HTTP 标头。对于长时间运行的任务，在任务完成之前，否则会返回的任务。              | 否  | 否                   | 否                     |
| <code>--custom_properties-custom_property</code> | 集合            | 一组用户定义的环境变量，作为参数传递给自定义脚本。   | 否  | 是                   | 是                     |
| <code>--port_mirroring</code>                    | 布尔值           | 切换是否将端口镜像用于配置集。状态可以是 <b>True</b> 或 <b>False</b> 。默认为 <b>False</b> | 否  | 否                   | 否                     |

`--custom_properties-custom_property` 参数是下表中使用子参数的集合。

表 4.68. --custom\_properties-custom\_property parameters

| 名称                           | 类型  | Description |
|------------------------------|-----|-------------|
| <b>custom_property.name</b>  | 字符串 | 自定义属性名称。    |
| <b>custom_property.value</b> | 字符串 | 自定义属性值。     |

**例 4.51. 创建新的 vnic 配置集**

```
[RHEVM shell (connected)]# add vnicprofile --name Gold --network-id 08305a2f-6952-4999-9646-c16137dc6d42
```

**例 4.52. 更新 vnic 配置集**

```
[RHEVM shell (connected)]# update vnicprofile Gold --port_mirroring true
```

**例 4.53. 删除 vnic 配置集**

```
[RHEVM shell (connected)]# remove vnicprofile Gold
```



## 第 5 章 CLI 查询

## 5.1. 查询语法

CLI `list` 命令使用 `--query` 属性来执行服务器端查询，其格式与 Red Hat Virtualization Manager 搜索查询语言相同：

表 5.1. 搜索查询示例

| 集合     | 标准   | 结果  |
|--------|--|---|
| 主机     | <code>vms.status=up</code>                       | 显示所有运行状态为 <b>up</b> 的虚拟机的主机的列表。                               |
| vms    | <code>domain=qa.company.com</code>               | 显示在指定域中运行的所有虚拟机的列表。   |
| vms    | <code>users.name=mary</code>                     | 显示属于用户名为 <b>mary</b> 的所有虚拟机的列表。                               |
| events | <code>severity&gt; 常规排序时间</code>                 | 显示所有严重性高于 <b>normal</b> 的事件的列表，并根据 <b>time</b> element 值进行排序。 |
| events | <code>severity&gt;normal sortby time desc</code> | 显示所有严重性高于 <b>normal</b> 的事件列表，并按 <b>时间</b> 元素值降序排列。           |

## 5.2. 通配符

搜索查询将值的一部分替换为星号作为通配符。

例 5.1. `name=vm*` 的通配符搜索查询

```
[RHEVM shell (connected)]# list vms --query "name=vm*"
```

此查询将生成名称以 `vm` 开头的所有虚拟机，如 `vm 1`、`vm2`、`vm a` 或 `vm-webserver`。

例 5.2. `name=v*1` 的通配符搜索查询

```
[RHEVM shell (connected)]# list vms --query "name=v*1"
```

此查询将生成名称以 `v` 开头的虚拟机，并以 `1` 结尾，如 `vm1`、`vr1` 或 `virtualmachine1`。