



Red Hat Satellite 6.11

置备主机

在 Red Hat Satellite 服务器上置备物理和虚拟主机的指南。

Red Hat Satellite 6.11 置备主机

在 Red Hat Satellite 服务器上置备物理和虚拟主机的指南。

Red Hat Satellite Documentation Team

satellite-doc-list@redhat.com

法律通告

Copyright © 2024 Red Hat, Inc.

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

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

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

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

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

摘要

Red Hat Satellite 置备指南介绍了置备物理和虚拟主机的说明。这包括设置所需的网络拓扑、配置必要的服务，并提供所有其他配置信息来在您的网络上置备主机。

目录

向红帽文档提供反馈	5
第 1 章 置备简介	6
1.1. 置备概述	6
1.2. 支持的云供应商	6
1.3. 支持的虚拟化基础架构	6
1.4. 网络引导 workflow	6
第 2 章 配置置备资源	10
2.1. 置备上下文	10
2.2. 设置置备上下文	10
2.3. 创建操作系统	10
2.4. 更新多个操作系统的详情	11
2.5. 创建架构	12
2.6. 创建硬件模型	12
2.7. 将同步的 KICKSTART 存储库用于主机的操作系统	13
2.8. 在 SATELLITE 中添加安装介质	13
2.9. 创建分区表	14
2.10. 动态分区示例	15
2.11. 置备模板	16
2.12. 置备模板的类型	17
2.13. 创建置备模板	18
2.14. 克隆置备模板	18
2.15. 创建计算配置集	19
2.16. 为主机设置默认加密根密码	19
2.17. 使用 NOVNC 访问虚拟机	20
第 3 章 配置网络	21
3.1. 事实和 NIC 过滤	21
3.2. 通过从数据库中删除 NIC 来优化性能	21
3.3. 网络资源	22
3.4. SATELLITE 和 DHCP 选项	23
3.5. SATELLITE 中的 DHCP 问题故障排除	24
3.6. 基于镜像置备的先决条件	25
3.7. 配置网络服务	26
3.8. 将域添加到 SATELLITE 服务器	30
3.9. 将子网添加到 SATELLITE 服务器	32
第 4 章 使用 INFOBLOX 作为 DHCP 和 DNS 提供商	35
4.1. INFOBLOX 限制	35
4.2. INFOBLOX 先决条件	35
4.3. 在 CAPSULE 服务器上安装 INFOBLOX CA 证书	35
4.4. 安装 DHCP INFOBLOX 模块	36
4.5. 安装 DNS INFOBLOX 模块	37
第 5 章 配置 IPXE 以减少置备时间	39
5.1. 引导虚拟机	41
5.2. 从 PXELINUX 启动 IPXE	44
第 6 章 使用 PXE 置备主机	48
6.1. 裸机置备的先决条件	48
6.2. 配置安全令牌服务持续时间	49
6.3. 使用未置备创建主机	49

6.4. 使用 PXE-LESS PROVISIONING 创建主机	51
6.5. 使用 UEFI HTTP 引导置备创建主机	55
6.6. 在置备过程中部署 SSH 密钥	58
第 7 章 配置发现服务	60
7.1. 安装发现服务	61
7.2. 置备模板 PXELINUX 发现片断	61
7.3. 发现主机的自动上下文	62
7.4. 发现模板和片段设置	63
7.5. 从发现的主机创建主机	64
7.6. 创建发现规则	67
7.7. 实施 PXE-LESS DISCOVERY	69
7.8. 无人值守使用、自定义和镜像恢复	71
7.9. 扩展发现镜像	73
7.10. 发现故障排除	75
第 8 章 使用红帽镜像构建器镜像置备	77
第 9 章 在 KVM 上置备虚拟机(LIBVIRT)	79
9.1. 为 KVM 连接配置 SATELLITE 服务器	80
9.2. 向 SATELLITE 服务器添加 KVM 连接	81
9.3. 将 KVM 镜像添加到 SATELLITE 服务器中	83
9.4. 在计算配置集中添加 KVM 详情	84
9.5. 在 KVM 上创建主机	85
第 10 章 在 RED HAT VIRTUALIZATION 上置备虚拟机	89
10.1. 将 RED HAT VIRTUALIZATION 连接添加到 SATELLITE 服务器	90
10.2. 在 RED HAT VIRTUALIZATION 中准备 CLOUD-INIT 镜像	92
10.3. 将 RED HAT VIRTUALIZATION 镜像添加到 SATELLITE 服务器中	92
10.4. 准备 CLOUD-INIT 模板	94
10.5. 在 COMPUTE 配置文件中添加 RED HAT VIRTUALIZATION 详情	96
10.6. 在 RED HAT VIRTUALIZATION 上创建主机	98
第 11 章 在 VMWARE VSPHERE 中置备虚拟机	101
11.1. 安装 VMWARE 插件	101
11.2. VMWARE 置备的先决条件	101
11.3. 创建 VMWARE 用户	102
11.4. 向 SATELLITE 服务器添加 VMWARE 连接	103
11.5. 将 VMWARE 镜像添加到 SATELLITE 服务器中	105
11.6. 在计算配置集中添加 VMWARE 详情	107
11.7. 在 VMWARE 上创建主机	109
11.8. 使用 VMWARE CLOUD-INIT 和 USERDATA 模板进行置备	112
11.9. 删除 VMWARE 上的虚拟机	117
11.10. 从 VMWARE 导入虚拟机到 SATELLITE	118
11.11. 计算资源缓存	118
第 12 章 在 OPENSIFT VIRTUALIZATION 中置备虚拟机	120
12.1. 在 SATELLITE 服务器中添加 OPENSIFT VIRTUALIZATION 连接	121
第 13 章 在 RED HAT OPENSTACK PLATFORM 上置备云实例	124
13.1. 将 RED HAT OPENSTACK PLATFORM 连接添加到 SATELLITE 服务器	124
13.2. 将 RED HAT OPENSTACK PLATFORM 镜像添加到 SATELLITE 服务器中	126
13.3. 在 COMPUTE 配置文件中添加 RED HAT OPENSTACK PLATFORM 详情	127
13.4. 在 RED HAT OPENSTACK PLATFORM 上创建基于镜像的主机	128

第 14 章 在 AMAZON EC2 中置备云实例	131
14.1. AMAZON EC2 置备的先决条件	131
14.2. 安装 AMAZON EC2 插件	131
14.3. 向 SATELLITE 服务器添加 AMAZON EC2 连接	131
14.4. 使用带有计算资源的 HTTP 代理	133
14.5. 为 AMAZON EC2 创建镜像	134
14.6. 将 AMAZON EC2 镜像添加到 SATELLITE 服务器中	135
14.7. 在 COMPUTE 配置文件中添加 AMAZON EC2 详情	137
14.8. 在 AMAZON EC2 上创建基于镜像的主机	137
14.9. 使用 SSH 连接到 AMAZON EC2 实例	139
14.10. 为 AMAZON WEB SERVICE EC2 环境配置完成模板	140
14.11. 删除 AMAZON EC2 上的虚拟机	142
14.12. 卸载 AMAZON EC2 插件	142
14.13. 有关 AMAZON WEB 服务和 SATELLITE 的更多信息	143
第 15 章 在 GOOGLE COMPUTE ENGINE 上置备云实例	144
15.1. 安装 GOOGLE GCE 插件	144
15.2. 向 SATELLITE 服务器添加 GOOGLE GCE 连接	145
15.3. 将 GOOGLE COMPUTE ENGINE 镜像添加到 SATELLITE 服务器中	147
15.4. 在 COMPUTE 配置文件中添加 GOOGLE GCE DETAILS	148
15.5. 在 GOOGLE COMPUTE ENGINE 上创建基于镜像的主机	149
15.6. 删除 GOOGLE GCE 上的虚拟机	151
15.7. UNINSTALLING GOOGLE GCE PLUGIN	152
第 16 章 在 MICROSOFT AZURE RESOURCE MANAGER 上置备云实例	153
16.1. 向 SATELLITE 服务器添加 MICROSOFT AZURE RESOURCE MANAGER 连接	154
16.2. 将 MICROSOFT AZURE RESOURCE MANAGER 镜像添加到 SATELLITE 服务器中	155
16.3. 在 COMPUTE 配置集中添加 MICROSOFT AZURE RESOURCE MANAGER 详情	157
16.4. 在 MICROSOFT AZURE RESOURCE MANAGER 上创建基于镜像的主机	160
16.5. 删除 MICROSOFT AZURE 上的虚拟机	162
16.6. 卸载 MICROSOFT AZURE 插件	163
附录 A. 置备示例的初始化脚本	164
附录 B. 置备 FIPS-COMPLIANT 主机	167
B.1. 更改置备密码哈希算法	167
B.2. 设置 FIPS-ENABLED 参数	168
B.3. 验证 FIPS 模式已启用	168
附录 C. 为 RED HAT SATELLITE 构建云镜像	169
C.1. 创建自定义 RED HAT ENTERPRISE LINUX 镜像	169
C.2. 注册中支持的客户端	170
C.3. 配置主机以进行注册	171
C.4. 注册主机	171
C.5. 安装 KATELLO 代理	175
C.6. 在主机上手动安装和配置 PUPPET 代理	176
C.7. 完成 RED HAT ENTERPRISE LINUX 7 镜像	178
C.8. 完成 RED HAT ENTERPRISE LINUX 6 镜像	179
C.9. 后续步骤	181
附录 D. 主机参数层次结构	182

向红帽文档提供反馈

我们感谢您对文档提供反馈信息。请让我们了解如何改进文档。

您可以通过在 Bugzilla 中记录一个 ticket 来提交反馈：

1. 导航到 [Bugzilla](#) 网站。
2. 在 **Component** 字段中，使用 **Documentation**。
3. 在 **Description** 字段中，输入您要改进的建议。包括文档相关部分的链接。
4. 点 **Submit Bug**。

第 1 章 置备简介

1.1. 置备概述

置备是一个以裸机物理或虚拟开始的进程，并以完全配置的、随时使用的操作系统结束。使用 Red Hat Satellite，您可以为大量主机定义和自动化精细配置。

有很多置备方法。例如，您可以使用 Satellite 服务器的集成胶囊或外部 Capsule 服务器使用基于 PXE 和非 PXE 的方法调配裸机主机。您还可以通过 API 从特定供应商置备云实例。这些配置方法是 Red Hat Satellite 应用程序生命周期的一部分，用于创建、管理和更新主机。

Red Hat Satellite 有不同的置备主机的方法：

裸机置备

Satellite 主要通过 PXE 引导和 MAC 地址标识来置备裸机主机。您可以创建主机条目，并指定要置备的物理主机的 MAC 地址。您还可以启动空白主机以使用 Satellite 的发现服务，该服务创建随时置备主机池。

云供应商

Satellite 连接到私有和公有云提供程序，从云环境存储的镜像调配主机实例。这还包括选择要使用的硬件配置文件或类别。

云基础架构

Satellite 连接到虚拟化基础架构服务，如 Red Hat Virtualization 和 VMware，以便从虚拟机镜像模板置备虚拟机，或使用与裸机供应商相同的基于 PXE 的引导方法。

1.2. 支持的云供应商

您可以将以下云供应商作为计算资源连接到 Satellite：

- Red Hat OpenStack Platform
- Amazon EC2
- Google Compute Engine
- Microsoft Azure

1.3. 支持的虚拟化基础架构

您可以将以下虚拟化基础架构作为计算资源连接到 Satellite：

- KVM (libvirt)
- Red Hat Virtualization (已弃用)
- VMware
- OpenShift Virtualization

1.4. 网络引导 workflow

对于物理或虚拟 BIOS 主机：

1. 使用网络将第一次引导设备设置为引导配置。
2. 将第二个引导设备设置为从硬盘引导。Satellite 管理 TFTP 引导配置文件，因此可以通过重新引导来轻松调配主机。

对于物理或虚拟 EFI 主机：

1. 使用网络将第一次引导设备设置为引导配置。
2. 根据 EFI 固件类型和配置，OS 安装程序通常会将 OS 引导装载程序配置为第一个条目。
3. 要再次重启安装程序，请使用 **efibootmgr** 工具切回到从网络引导。

置备过程遵循基本的 PXE 工作流：

1. 您可以创建主机并选择域和子网。Satellite 从 DHCP Capsule 服务器请求可用的 IP 地址，该服务器与 Satellite 中的子网或 PostgreSQL 数据库相关联。Satellite 将此 IP 地址加载到 Create Host 窗口的 IP 地址字段中。完成新主机的所有选项后，提交新主机请求。
2. 根据主机及其域和子网的配置规格，Satellite 会创建以下设置：
 - 与子网关联的 Capsule 服务器上的 DHCP 记录。
 - 与域关联的 Capsule 服务器上的转发 DNS 记录。
 - 与子网关联的 DNS Capsule 服务器上的反向 DNS 记录。
 - pxelinux、Grub、Grub2 和 iPXE 配置文件用于与子网关联的 TFTP Capsule 服务器中的主机。
 - 关联的 Puppet 服务器上的 Puppet 证书。
 - 关联的身份服务器上的域。
3. 主机配置为从网络引导，作为第一个设备，HDD 作为第二个设备。
4. 新主机从 DHCP 服务器请求 DHCP 保留。
5. DHCP 服务器响应保留请求，并返回 TFTP **next-server** 和 **filename** 选项。
6. 主机根据 PXELoader 设置从 TFTP 服务器请求引导装载程序和菜单。
7. 通过 TFTP 返回引导装载程序。
8. 引导装载程序通过其调配接口 MAC 地址获取主机的配置。
9. 引导装载程序获取操作系统安装程序内核、init RAM 磁盘和引导参数。
10. 安装程序从 Satellite 请求置备模板。
11. Satellite 呈现调配模板并将结果返回到主机。
12. 安装程序执行操作系统的安装。
 - 安装程序使用 Subscription Manager 将主机注册到 Satellite。
 - 安装程序会安装管理工具，如 **katello-agent** 和 **puppet**。

- 安装程序会在 **postinstall** 脚本中通知 Satellite 成功构建。
13. PXE 配置文件恢复到本地引导模板。
 14. 主机重启。
 15. 新主机从 DHCP 服务器请求 DHCP 保留。
 16. DHCP 服务器响应保留请求，并返回 TFTP **next-server** 和 **filename** 选项。
 17. 主机根据 PXELoader 设置从 TFTP 服务器请求引导装载程序和菜单。
 18. 通过 TFTP 返回引导装载程序。
 19. 引导装载程序通过其调配接口 MAC 地址获取主机的配置。
 20. 引导装载程序从本地驱动器启动引导。
 21. 如果您将主机配置为使用任何 Puppet 类，则主机将使用模块配置其自身。

完全置备的主机执行以下工作流：

1. 主机配置为从网络引导，作为第一个设备，HDD 作为第二个设备。
2. 新主机从 DHCP 服务器请求 DHCP 保留。
3. DHCP 服务器响应保留请求，并返回 TFTP **next-server** 和 **filename** 选项。
4. 主机根据 PXELoader 设置从 TFTP 服务器请求引导装载程序和菜单。
5. 通过 TFTP 返回引导装载程序。
6. 引导装载程序通过其调配接口 MAC 地址获取主机的配置设置。
7. 对于 BIOS 主机：
 - 引导装载程序返回不可启动的设备，以便 BIOS 跳过到下一个设备（从 HDD 启动）。
8. 对于 EFI 主机：
 - 引导装载程序在 ESP 分区中找到 Grub2，并链它。
9. 如果主机对于 Satellite 未知，则提供了一个默认的引导装载程序配置。启用 Discovery 服务后，它会引导至发现，否则从 HDD 启动。

此工作流因自定义选项而异。例如：

Discovery（发现）

如果您使用发现服务，Satellite 会自动检测新主机的 MAC 地址，并在提交请求后重新启动主机。请注意，TCP 端口 8443 必须可以被附加主机的 Capsule 访问，以供 Satellite 重新启动主机。

无 PXE 置备

提交新主机请求后，您必须使用从 Satellite 下载的引导磁盘引导特定主机，并使用主机的 USB 端口传输。

计算资源

Satellite 创建虚拟机并检索 MAC 地址，并将 MAC 地址存储在 Satellite 中。如果使用基于镜像的置备，主机不会遵循标准 PXE 引导和操作系统安装。计算资源为主机创建要使用的镜像副本。根据

Satellite 中的镜像设置，可以传递看到的数据以进行初始配置，例如使用 **cloud-init**。Satellite 可以通过 SSH 连接到主机并执行模板来完成自定义。



注意

默认情况下，从 Satellite 中删除调配的配置文件主机不会破坏外部计算资源的实际虚拟机。要在删除 Satellite 上的主机条目时销毁虚拟机，请导航到 **Administer > Settings > Provisioning**，并使用 **destroy_vm_on_host_delete** 设置配置此行为。如果您没有销毁关联的虚拟机，并尝试创建具有相同资源名称的新虚拟机，它将会失败，因为该虚拟机名称已存在于外部计算资源中。您仍然可以使用标准主机注册工作流将现有虚拟机注册到 Satellite 中，供您用于任何已调配的主机。

第 2 章 配置置备资源

2.1. 置备上下文

置备上下文是您为 Satellite 组件指定的机构和位置的组合。组件所属的机构和位置设定该组件的所有权和权限。

组织根据所有权、目的、内容、安全级别和其他部门将 Red Hat Satellite 组件分成逻辑组。您可以通过 Red Hat Satellite 创建和管理多个机构，并为各个机构分配组件。这样可确保 Satellite 服务器在特定机构中调配主机，并且仅使用分配给该机构的组件。有关机构的更多信息，[请参阅 内容管理指南中的管理机构](#)。

位置功能类似于组织。区别在于，位置基于物理或地理设置。用户可以嵌套层次结构中的位置。有关位置的更多信息，[请参阅 内容管理指南中的管理位置](#)。

2.2. 设置置备上下文

设置置备上下文时，您可以定义用于置备主机的机构和位置。

机构和位置菜单位于菜单栏中，位于 Satellite Web UI 的左上角。如果您还没有选择要使用的机构和位置，则菜单将显示：**任何机构** 和 **任何位置**。

流程

1. 点 **Any Organization** 并选择机构。
2. 单击 **Any Location**，再选择要使用的位置。

每个用户都可以在其帐户设置中设置其默认置备上下文。单击 Satellite Web UI 右上角的用户名，然后选择 **My account** 以编辑您的用户帐户设置。

CLI 过程

- 使用 CLI 时，将 **--organization** 或 **--organization-label** 和 **--location** 或 **--location-id** 作为一个选项包含。例如：

```
# hammer host list --organization "Default_Organization" --location "Default_Location"
```

此命令输出为 Default_Organization 和 Default_Location 分配的主机。

2.3. 创建操作系统

操作系统是一组资源，用于定义 Satellite 服务器如何在主机上安装基础操作系统。操作系统条目组合了之前定义的资源，如安装介质、分区表、置备模板等。

从红帽的 CDN 导入操作系统会在 **Hosts > Operating Systems** 页面中创建新条目。要从红帽的 CDN 导入操作系统，请启用操作系统的红帽存储库并将存储库同步到 Satellite。如需更多信息，[请参阅 管理内容中的 启用红帽存储库](#) 和同步存储库。

您还可以按照以下流程添加自定义操作系统。要使用 CLI 而不是 Satellite Web UI，[请参阅 CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Hosts > Operating systems**，再点 **New Operating system**。
2. 在 **Name** 字段中输入代表操作系统条目的名称。
3. 在 **Major** 字段中输入与操作系统的主版本对应的数字。
4. 在 **Minor** 字段中输入与操作系统的次要版本对应的数字。
5. 在 **Description** 字段中，输入操作系统的描述。
6. 从 **Family** 列表中，选择操作系统的系列。
7. 从 **Root Password Hash** 列表中，选择 root 密码的编码方法。
8. 从 **Architectures** 列表中，选择操作系统使用的架构。
9. 点击 **分区表** 标签页并选择应用于此操作系统的分区表。
10. 可选：如果您使用非红帽内容，点安装介质标签页并选择应用到此操作系统的安装介质。如需更多信息，请参阅在 [Satellite 中添加安装介质](#)。
11. 单击 **Templates** 选项卡，再选择 **PXELinux 模板**、**Provisioning 模板** 和 **Finish 模板** 供您的操作系统使用。如果您计划使用 **iPXE** 进行调配，您可以选择其他模板，如 **iPXE 模板**。
12. 点 **Submit** 保存您的置备模板。

CLI 过程

- 使用 **hammer os create** 命令创建操作系统：

```
# hammer os create --name "MyOS" \
--description "My_custom_operating_system" \
--major 7 --minor 3 --family "Redhat" --architectures "x86_64" \
--partition-tables "My_Partition" --media "Red_Hat" \
--provisioning-templates "My_Provisioning_Template"
```

2.4. 更新多个操作系统的详情

使用这个流程更新多个操作系统的详情。本例演示了如何为每个操作系统分配一个名为 **Kickstart default** 的分区表、名为 **Kickstart default PXELinux** 的配置模板，以及名为 **Kickstart Default** 的置备模板。

流程

1. 在 Satellite 服务器上，运行以下 Bash 脚本：

```
PARTID=$(hammer --csv partition-table list | grep "Kickstart default," | cut -d, -f1)
PXEID=$(hammer --csv template list --per-page=1000 | grep "Kickstart default PXELinux" |
cut -d, -f1)
SATID=$(hammer --csv template list --per-page=1000 | grep "provision" | grep ",Kickstart
default" | cut -d, -f1)

for i in $(hammer --no-headers --csv os list | awk -F, {'print $1'})
do
    hammer partition-table add-operatingsystem --id="{PARTID}" --operatingsystem-id="{i}"
    hammer template add-operatingsystem --id="{PXEID}" --operatingsystem-id="{i}"
```

```
hammer os set-default-template --id="{i}" --config-template-id=${PXEID}
hammer os add-config-template --id="{i}" --config-template-id=${SATID}
hammer os set-default-template --id="{i}" --config-template-id=${SATID}
done
```

2. 显示更新的操作系统的信息以验证操作系统是否已正确更新：

```
# hammer os info --id 1
```

2.5. 创建架构

Satellite 中的构架代表主机和操作系统的逻辑分组。当主机通过 Puppet 进行检查时，Satellite 会自动创建架构。Satellite 中已预先设置基本的 i386 和 x86_64 架构。

使用这个流程在 Satellite 中创建架构。

支持的构架

只支持使用 PXE、发现和引导磁盘来置备 Intel x86_64 架构。如需更多信息，请参阅红帽知识库解决方案 [支持的架构](#) 和 [Satellite 6 中的置备场景](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Hosts > Architectures**，再点 **Create Architecture**。
2. 在 **Name** 字段中输入架构的名称。
3. 从 **Operating Systems** 列表中，选择一个操作系统。如果没有可用的，您可以在 **Hosts > Operating Systems** 下创建并分配它们。
4. 点 **Submit**。

CLI 过程

- 输入 **hammer architecture create** 命令来创建架构。指定包含此架构的名称和操作系统：

```
# hammer architecture create \
--name "My_Architecture" \
--operatingsystems "My_Operating_System"
```

2.6. 创建硬件模型

使用这个流程在 Satellite 中创建硬件模型，以便您可以指定主机使用的硬件模型。

流程

1. 在 Satellite Web UI 中，导航到 **Hosts > Hardware Models**，再点 **Create Model**。
2. 在 **Name** 字段中输入硬件模型的名称。
3. 另外，在 **Hardware Model** 和 **Vendor Class** 字段中，您可以为您的系统输入对应的信息。
4. 在 **Info** 字段中，输入硬件模型的描述。

5. 点 **Submit** 保存您的硬件模型。

CLI 过程

- 使用 **hammer model create** 命令创建硬件模型。唯一需要的参数是 **--name**。（可选）使用 **--hardware-model** 选项输入硬件模型、带有 **--vendor-class** 选项的厂商类，以及带有 **--info** 选项的描述：

```
# hammer model create \
--hardware-model "My_Hardware_Model" \
--info "My_Description" \
--name "My_Hardware_Model_Name" \
--vendor-class "My_Vendor_Class"
```

2.7. 将同步的 KICKSTART 存储库用于主机的操作系统

Satellite 包含用于安装调配的主机操作系统的一组同步 kickstart 存储库。有关添加存储库的更多信息，请参阅 [内容管理指南中的同步存储库](#)。

使用这个流程设置 kickstart 存储库。

先决条件

您必须在置备前启用 **BaseOS** 和 **Appstream Kickstart**。

流程

1. 添加您要用于现有内容视图的同步 kickstart 存储库，或者创建新的内容视图并添加 kickstart 存储库。
对于 Red Hat Enterprise Linux 8，请确保为 **x86_64 - AppStream Kickstart x86_64 8** 和 **Red Hat Enterprise Linux 8 for x86_64 - BaseOS Kickstart x86_64 8** 软件仓库同时添加了 Red Hat Enterprise Linux 8。

如果您使用断开连接的环境，则必须从 Red Hat Enterprise Linux 二进制 DVD 中导入 Kickstart 软件仓库。如需更多信息，请参阅 [内容管理指南中的导入 Kickstart 存储库](#)。
2. 发布一个新的内容视图版本，其中添加了 kickstart 存储库并将其提升到所需的生命周期阶段。有关更多信息，请参阅 [内容管理指南中的管理内容视图](#)。
3. 当您创建主机时，在 **Operating System** 选项卡中，对于 **Media Selection**，选中 **Synced Content** 复选框。

要查看 kickstart 树，请输入以下命令：

```
# hammer medium list --organization "My_Organization"
```

2.8. 在 SATELLITE 中添加安装介质

安装介质是 Satellite 服务器用来从外部存储库在机器上安装基础操作系统的软件包来源。您可以使用此参数安装第三方内容。红帽内容通过存储库同步提供。

您可以通过进入到 **Hosts > Provisioning Setup > Installation Media** 来查看安装介质。

安装介质必须采用操作系统安装树的格式，且必须通过 HTTP URL 从托管安装程序的机器进行访问。

默认情况下，Satellite 包括一些官方 Linux 发行版的安装介质。请注意，其中一些安装介质针对操作系统的特定版本。例如，CentOS 镜像(7.x) 必须用于 CentOS 7 或更早版本，CentOS 镜像(8.x) 必须用于 CentOS 8 或更高版本。

如果要在多个主机上安装操作系统时提高下载性能，您必须修改安装介质的路径以指向最接近的镜像或本地副本。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 Hosts > Installation Media 并点 Create Medium。
2. 在 Name 字段中输入代表安装介质条目的名称。
3. 在 Path 中输入包含安装树的 URL。您可以使用路径中的以下变量来代表多个不同的系统架构和版本：
 - `$arch` - 系统架构。
 - `$version` - 操作系统版本。
 - `$major` - 操作系统主版本。
 - `$minor` - 操作系统次版本。HTTP 路径示例：

```
http://download.example.com/centos/$version/Server/$arch/os/
```

4. 从 Operating system family 列表中，选择安装介质的发布或系列。例如，CentOS 和 Fedora 位于 Red Hat 系列。
5. 单击 机构和位置 选项卡，以更改调配上下文。Satellite 服务器将安装介质添加到集置备上下文中。
6. 点 Submit 保存您的安装介质。

CLI 过程

- 使用 `hammer medium create` 命令创建安装介质：

```
# hammer medium create \  
--locations "My_Location" \  
--name "My_OS" \  
--organizations "My_Organization" \  
--os-family "Redhat" \  
--path "http://download.example.com/centos/$version/Server/$arch/os/"
```

2.9. 创建分区表

分区表是一种模板，用于定义 Satellite 服务器配置新主机上可用磁盘的方式。分区表使用与置备模板相同的 ERB 语法。Red Hat Satellite 包含一组要使用的默认分区表，包括 Kickstart 默认。您还可以编辑分区表条目来配置首选分区方案，或者创建分区表条目并将其添加到操作系统条目中。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 Hosts > Partition Tables 并点 Create Partition Table。
2. 在 Name 字段中输入分区表的名称。
3. 如果要模板设置为自动与新机构或位置关联，请选择 Default 复选框。
4. 如果要模板标识为其他分区表的可重复使用的代码片段，请选择 Snippet 复选框。
5. 从 Operating System Family 列表中，选择分区布局的发布或系列。例如，Red Hat Enterprise Linux、CentOS 和 Fedora 位于 Red Hat 系列。
6. 在 Template editor 字段中，输入磁盘分区的布局。例如：

```
zerombr
clearpart --all --initlabel
autopart
```

您还可以使用 Template 文件浏览器上传模板文件。

布局的格式必须与预期的操作系统匹配。例如，Red Hat Enterprise Linux 需要与 kickstart 文件匹配的布局。

7. 在 Audit Comment 字段中，添加对分区布局的更改摘要。
8. 点 机构 和 位置 选项卡添加您要与分区表关联的任何其他置备上下文。Satellite 将分区表添加到当前调配上下文中。
9. 点 Submit 保存您的分区表。

CLI 过程

1. 在使用 CLI 创建分区表前，先创建一个包含分区布局的纯文本文件。这个示例使用 `~/my-partition` 文件。
2. 使用 `hammer partition-table create` 命令创建安装介质：

```
# hammer partition-table create \
--file "path/to/my_partition_table" \
--locations "My_Location" \
--name "My Partition Table" \
--organizations "My_Organization" \
--os-family "My_OS_Family" \
--snippet false
```

2.10. 动态分区示例

使用 Anaconda kickstart 模板，以下部分指示 Anaconda 将整个磁盘清除，自动分区，将一个分区增大到最大大小，然后继续执行置备过程中的下一个事件序列：

```
zerombr
clearpart --all --initlabel
autopart <%= host_param('autopart_options') %>
```

动态分区由安装程序执行。因此，您可以编写自己的规则来指定如何根据节点的运行时信息对磁盘进行分区，例如磁盘大小、驱动器数、厂商或制造商。

如果要置备服务器并使用动态分区，请将以下示例添加为模板。当包含 `192.168.1.0/24Dynamic` 条目时，模板的内容会加载到 `%pre shell scriptlet` 中，并创建一个 `/tmp/diskpart.cfg`，然后包含在 Kickstart `partitioning` 部分。

```
#Dynamic (do not remove this line)

MEMORY=$((`grep MemTotal: /proc/meminfo | sed 's/^MemTotal: */|sed 's/ .*//'` / 1024))
if [ "$MEMORY" -lt 2048 ]; then
    SWAP_MEMORY=$((MEMORY * 2))
elif [ "$MEMORY" -lt 8192 ]; then
    SWAP_MEMORY=$MEMORY
elif [ "$MEMORY" -lt 65536 ]; then
    SWAP_MEMORY=$((MEMORY / 2))
else
    SWAP_MEMORY=32768
fi

cat <<EOF > /tmp/diskpart.cfg
zerombr yes
clearpart --all --initlabel
part /boot --fstype ext4 --size 200 --asprimary
part swap --size "$SWAP_MEMORY"
part / --fstype ext4 --size 1024 --grow
EOF
```

2.11. 置备模板

置备模板定义了 Satellite 服务器在主机上安装操作系统的方式。

Red Hat Satellite 包括许多模板示例。在 Satellite Web UI 中，导航到 `Hosts > Provisioning templates` 以查看它们。您可以创建模板或克隆模板并编辑克隆。对于模板的帮助，请导航到 `Hosts > Provisioning templates > Create Template > Help`。

红帽支持的模板由红帽图标表示。

要隐藏不支持的模板，在 Satellite Web UI 中导航至 `Administer > Settings`。在 `Provisioning` 选项卡上，将 `Show unsupported provisioning templates` 的值设置为 `false`，然后单击 `Submit`。您还可以通过进行以下查询 `"supported = true"` 来过滤出支持的模板。

如果您克隆受支持的模板，克隆的模板将不被支持。

模板接受嵌入式 Ruby (ERB) 语法。如需更多信息，请参阅 [管理主机](#) 中的 [编写参考的模板](#)。

您可以下载置备模板。在下载模板前，您必须创建一个 debug 证书。如需更多信息，请参阅 [内容管理指南](#) 中的 [创建机构调试证书](#)。

您可以同步 Satellite 服务器和 Git 存储库或本地目录之间的模板。如需更多信息，请参阅 [管理主机指南](#) 中的 [同步模板存储库](#)。

要查看应用到模板的更改历史记录，请导航到 `Hosts > Provisioning templates`，选择其中一个模板，然后点 `History`。点 `Revert` 使用上一版本覆盖内容。您还可以恢复到早期的更改。点 `Show Diff` 查看特定更改的信息：

- Template Diff 选项卡显示在置备模板的正文中。
- Details 选项卡显示在模板描述中。
- History 选项卡显示更改更改的用户以及更改的日期。

2.12. 置备模板的类型

置备模板有多种类型：

provision

置备过程的主要模板。例如，Kickstart 模板。有关 kickstart 模板语法的更多信息，请参阅 *Red Hat Enterprise Linux 7 安装指南* 中的 [Kickstart 语法参考](#)。

PXELinux, PXEGrub, PXEGrub2

基于 PXE 的模板，部署到与子网关联的模板 Capsule，以确保主机使用带有正确内核选项的安装程序。对于 BIOS 置备，请选择 PXELinux 模板。对于 UEFI 置备，请选择 PXEGrub2。

完成

主置备过程完成后，使用 SSH 连接执行后配置脚本。您只能在不支持 user_data 的物理或虚拟环境中使用 Finish 模板。不要将镜像与 foreman 发现 ISO 混淆，有时被称为 Foreman 发现镜像。此上下文中的镜像是虚拟化环境中的安装镜像，便于部署。

当一个完成脚本成功退出码 0 时，Red Hat Satellite 会将代码视为成功，主机会退出构建模式。

请注意，有几个带有构建模式的完成脚本使用 `调用回` HTTP 调用。这些脚本不用于基于镜像的调配，而是用于配置操作系统安装后，如 Debian、Ubuntu 和 BSD。红帽不支持置备 Red Hat Enterprise Linux 以外的操作系统。

user_data

接受自定义数据的供应商（也称为 seed data）的后配置脚本。您可以使用 user_data 模板仅在云或虚拟化环境中置备虚拟机。此模板不需要 Satellite 能够访问主机；云或虚拟化平台负责向镜像发送数据。

确保要置备的镜像有相应的软件来读取安装的数据并在引导过程中启动。例如，cloud-init，它需要 YAML 输入，或 ignition，这需要 JSON 输入。

cloud_init

有些环境（如 VMWare）不支持自定义数据，或者有自己的数据格式来限制自定义期间可以执行的操作。在这种情况下，您可以使用 foreman 插件配置 cloud-init 客户端，这将尝试通过 HTTP 或 HTTPS 直接从 Satellite 下载模板。此技术可以在任何环境中使用，最好使用虚拟化。

确保您满足以下要求以使用 cloud_init 模板：

- 确保要置备的镜像有相应的软件来读取安装的数据并在引导过程中启动。
- 置备的主机可以从与主机调配接口 IP 地址匹配的 IP 地址访问 Satellite。请注意，cloud-init 无法在 NAT 后面工作。

Bootdisk

无 PXE 引导方法的模板。

内核执行(kexec)

无 PXE 引导方法的内核执行模板。



注意

内核执行是一个技术预览功能。技术预览功能不包括在 Red Hat Subscription 服务等级协议(SLA)下被完全支持，且不适用于生产环境。但是，这些功能对即将推出的产品领导提供早期访问，使客户能够在开发过程中测试并提供反馈。

脚本

默认情况下不使用任意脚本，但对自定义任务很有用。

ZTP

Zero Touch Provisioning 模板。

POAP

PowerOn Auto Provisioning 模板。

iPXE

要使用的 iPXE 或 gPXE 环境的模板，而不是 PXELinux。

2.13. 创建置备模板

置备模板定义了 Satellite 服务器在主机上安装操作系统的方式。使用这个流程创建新的置备模板。

流程

1. 在 Satellite Web UI 中，导航到 Hosts > Provisioning Templates，再点 Create Template。
2. 在 Name 字段中输入置备模板的名称。
3. 根据需要填写其余字段。Help 选项卡提供有关模板语法的信息，并详细介绍了可在模板中不同类型的对象上调用的可用功能、变量和方法。

CLI 过程

1. 在使用 CLI 创建模板之前，先创建一个包含模板的纯文本文件。本例使用 ~/my-template 文件。
2. 使用 `hammer template create` 命令创建模板，并使用 `--type` 选项指定类型：

```
# hammer template create --name "My Provisioning Template" \
--file ~/my-template --type provision --organizations "My_Organization" \
--locations "My_Location"
```

2.14. 克隆置备模板

置备模板定义了 Satellite 服务器在主机上安装操作系统的方式。使用此流程克隆模板，并将您的更新添加到克隆中。

流程

1. 在 Satellite Web UI 中，导航到 Hosts > Provisioning Templates 并搜索您要使用的模板。
2. 单击 Clone 以复制模板。
3. 在 Name 字段中输入置备模板的名称。

4. 选中 Default 复选框，以设置模板，以自动与新机构或位置关联。
5. 在 Template editor 字段中，输入置备模板的正文。您还可以使用 Template 文件浏览器上传模板文件。
6. 在 Audit Comment 字段中，输入对置备模板的更改概述以进行审核。
7. 点 Type 选项卡，如果您的模板是片断，请选择 Snippet 复选框。片段不是独立的置备模板，但置备模板的一部分可以插入到其他置备模板中。
8. 从 Type 列表中，选择模板的类型。例如，Provisioning 模板。
9. 单击 关联 选项卡，然后从 Applicable Operating Systems 列表中选择您要与置备模板关联的操作系统的名称。
10. （可选）点 Add combination，然后从 Host Group 列表中选择主机组，或从 Environment 列表选择一个主机组，以将置备模板与主机组和环境关联。
11. 点 机构 和 位置 选项卡，向模板添加任何其他上下文。
12. 点 Submit 保存您的置备模板。

2.15. 创建计算配置集

您可以使用计算配置集预定义虚拟机硬件详情，如 CPU、内存和存储。Red Hat Satellite 的默认安装包包含三个预定义的配置集：

- 1-small
- 2-medium
- 3-large

您可以将计算配置集应用到所有支持的计算资源：

- [第 1.2 节 “支持的云供应商”](#)
- [第 1.3 节 “支持的虚拟化基础架构”](#)

流程

1. 在 Satellite Web UI 中，导航到 Infrastructure > Compute Profiles，再点 Create Compute Profile。
2. 在 Name 字段中输入配置集的名称。
3. 点 Submit。此时会打开一个新窗口，其中包含计算配置集的名称。
4. 在新窗口中，点每个计算资源的名称，再编辑您要为此计算配置集设置的属性。

2.16. 为主机设置默认加密根密码

如果您不想为您置备的主机设置纯文本默认 root 密码，您可以使用默认加密的密码。

流程

1. 生成加密的密码：

```
# python -c 'import crypt,getpass;pw=getpass.getpass(); print(crypt.crypt(pw)) if
(pw==getpass.getpass("Confirm: ")) else exit()'
```

2. 复制密码以供以后使用。
3. 在 Satellite Web UI 中，导航到 Administer > Settings。
4. 在 Settings 页面上，选择 Provisioning 选项卡。
5. 在 Name 列中，导航到 Root 密码，然后单击 点击以编辑。
6. 粘贴加密的密码，然后单击 Save。

2.17. 使用 NOVNC 访问虚拟机

您可以使用浏览器访问 Satellite 创建的虚拟机的 VNC 控制台。

Satellite 支持以下虚拟化平台上使用 noVNC：

- VMware
- libvirt
- Red Hat Virtualization

先决条件

- 您必须具有 Satellite 创建的虚拟机。
- 对于现有虚拟机，请确保 计算资源 设置中的 Display 类型是 VNC。
- 您必须将 Katello root CA 证书导入到 Satellite 服务器中。在浏览器中添加安全例外不足以使用 noVNC。如需更多信息，请参阅 [管理 Red Hat Satellite 指南中的安装 Katello Root CA 证书](#) 部分。

流程

1. 在虚拟机主机系统上，将防火墙配置为允许端口 5900 上的 VNC 服务 5900 到 5930：

- 在 Red Hat Enterprise Linux 6 中：

```
# iptables -A INPUT -p tcp --dport 5900:5930 -j ACCEPT
# service iptables save
```

- 在 Red Hat Enterprise Linux 7 中：

```
# firewall-cmd --add-port=5900-5930/tcp
# firewall-cmd --add-port=5900-5930/tcp --permanent
```

2. 在 Satellite Web UI 中，导航到 Infrastructure > Compute Resources 并选择计算资源的名称。
3. 在 Virtual Machines 选项卡中，选择虚拟机主机名称。确保计算机已开机，然后选择 Console。

第 3 章 配置网络

每个置备类型都需要一些网络配置。使用本章在 Satellite 服务器上的集成胶囊中配置网络服务。

新主机必须有权访问您的 Capsule 服务器。Capsule 服务器可以是您在 Satellite 服务器或外部胶囊式服务器上集成的 Capsule。当主机位于隔离的网络上且无法直接连接到 Satellite 服务器时，或者内容与 Capsule 服务器同步时，您可能希望从外部 Capsule 服务器调配主机。使用外部 Capsule 服务器置备可以保存在网络带宽上。

配置 Capsule 服务器有两个基本要求：

1. 配置网络服务。这包括：
 - 内容交付网络
 - 网络服务(DHCP、DNS 和 TFTP)
 - Puppet 配置
2. 在 Satellite 服务器中定义网络资源数据，以帮助在新主机上配置网络接口。

以下说明类似应用来配置独立 Capsule 管理特定网络。要将 Satellite 配置为使用外部 DHCP、DNS 和 TFTP 服务，请参阅在 [连接的网络环境中安装 Satellite 服务器](#) 中的[配置外部服务](#)。

3.1. 事实和 NIC 过滤

事实描述了主机所报告的所有方面，如内存、操作系统版本或架构。您可以在 Monitor > Facts 中找到事实，并通过事实或模板中使用事实(fact)来搜索主机。

Satellite 从多个来源收集事实：

- 订阅管理器
- ansible
- puppet

Satellite 是主机和网络接口的清单系统。对于虚拟机监控程序或容器主机，每个主机添加数千个接口，每几分钟时间来更新清单。对于每个报告的每个 NIC，Satellite 会创建一个 NIC 条目，这些条目永远不会从数据库中删除。解析所有事实并比较数据库中的所有记录，从而使 Satellite 非常慢且不可用。要优化各种操作的性能，最重要的是事实导入，您可以使用 Administer > Settings 下的 Facts 选项卡中可用的选项。

3.2. 通过从数据库中删除 NIC 来优化性能

使用 Exclude 模式过滤并排除存储在 Satellite 中的事实和带有匹配标识符选项的 Ignore 接口的连接。默认情况下，这些选项设置为最常见的虚拟机监控程序。如果您以不同的方式命名虚拟接口，您可以更新此过滤器根据您的要求使用它。

流程

1. 在 Satellite Web UI 中，导航到 Administer > Settings 并选择 Facts 选项卡。
2. 要过滤掉以特定名称开头的所有接口，如 blu，请将 blubang 添加到带有匹配标识符选项的 Ignore 接口中。

- 要防止数据库存储与特定名称开头的接口相关的事实（例如，`blu`，为 `Satellite` 选项中存储的事实添加 `blu`）模式。
默认情况下，它包含与具有匹配标识符选项的 `Ignore` 接口相同的列表。您可以根据您的要求覆盖它。这会完全过滤掉事实，而不存储它们。
- 要从数据库中删除事实，请输入以下命令：

```
# foreman-rake facts:clean
```

此命令删除与 `Managing > Settings > Facts > Facts > theExclude pattern for facts` 存储在 `Satellite` 选项中的过滤器的所有事实匹配。

- 要从数据库中删除接口，请输入以下命令：

```
# foreman-rake interfaces:clean
```

这个命令会删除与 `Administer > Settings > Facts > Facts > theIgnore` 接口以及匹配的标识符选项中添加的过滤器的所有接口匹配。

3.3. 网络资源

`Satellite` 包含必须设置并配置的网络资源来创建主机。它包括以下网络资源：

域

您必须将由 `Satellite` 管理的每个主机分配到某个域。通过使用域，`Satellite` 可以管理 `A`、`AAAA` 和 `PTR` 记录。即使您不希望 `Satellite` 管理 `DNS` 服务器，您仍必须创建并关联至少一个域。域包含在命名惯例 `Satellite` 主机中，例如，`example.com` 域中名为 `test123` 的主机具有完全限定域名 `test123.example.com`。

subnet

您必须将由 `Satellite` 管理的每个主机分配到子网。使用子网，`Satellite` 可以管理 `IPv4` 保留。如果没有保留集成，您仍必须创建并关联至少一个子网。当您在 `Satellite` 中管理子网时，您无法为该子网创建 `Satellite` 之外的 `DHCP` 记录。在 `Satellite` 中，您可以使用 `IP 地址管理(IPAM)` 使用以下选项之一管理 `IP 地址`：

- DHCP**：DHCP Capsule 通过查找从范围的第一个地址开始的下一个可用 `IP 地址` 来管理 `IP 地址` 的分配，并跳过保留的所有地址。在分配 `IP 地址` 前，Capsule 会发送 `ICMP` 和 `TCP ping` 来检查 `IP 地址` 是否正在使用。请注意，如果主机已关闭，或者将防火墙配置为禁用连接的防火墙，`Satellite` 会导致错误假设 `IP 地址` 可用。此检查不适用于关闭的主机，因此 `DHCP` 选项只能用于 `Satellite` 控制的子网，且没有外部创建的主机。
Capsule DHCP 模块会在短时间内保留提供的 `IP 地址`，以防止并发访问期间冲突，因此 `IP 范围` 内的一些 `IP 地址` 可能会临时使用。
- 内部 DB**：`Satellite` 通过按顺序从 `Satellite` 数据库中排除所有 `IP 地址`，从子网范围中找到下一个可用 `IP 地址`。数据源是数据库，而不是 `DHCP` 保留。当并行创建多个主机时，这个 `IPAM` 并不安全；在这种情况下，使用 `DHCP` 或 `Random DB IPAM`。
- Random DB**：`Satellite` 通过随机从 `Satellite` 数据库中排除所有 `IP 地址`，从子网范围中找到下一个可用 `IP 地址`。数据源是数据库，而不是 `DHCP` 保留。这个 `IPAM` 可以安全地与并发主机创建一起使用，因为 `IP 地址` 会以随机顺序返回，从而减少了冲突的机会。
- EUI-64**：扩展唯一标识符(EUI) 64 位 `IPv6` 地址生成，因为 `RFC2373` 通过 48 位 `MAC 地址` 获得。
- 外部 IPAM**：通过 Capsule 功能将 `IPAM` 应用到外部系统。`Satellite` 目前不附带任何外部 `IPAM` 实现，但一些插件正在开发中。

- **none** : 必须手动输入每个主机的 IP 地址。
选项 DHCP、内部 DB 和 Random DB 可能会导致在子网上与外部创建的记录冲突。这些子网必须采用专用 Satellite 控制。

有关添加子网的详情请参考 [第 3.9 节“将子网添加到 Satellite 服务器”](#)。

DHCP 范围

您可以在 Satellite 服务器中为发现和置备的系统定义相同的 DHCP 范围，但对同一子网中的每个服务都使用单独的范围。

3.4. SATELLITE 和 DHCP 选项

Satellite 通过 DHCP 胶囊管理 DHCP 保留。Satellite 也设置 `next-server` 和 `filename` DHCP 选项。

next-server 选项

`next-server` 选项提供从其引导的 TFTP 服务器的 IP 地址。默认情况下不设置这个选项，必须为每个 TFTP 胶囊设置。您可以使用 `satellite-installer` 命令和 `--foreman-proxy-tftp-servername` 选项在 `/etc/foreman-proxy/settings.d/tftp.yml` 文件中设置 TFTP 服务器：

```
# satellite-installer --foreman-proxy-tftp-servername 1.2.3.4
```

然后，每个 TFTP Capsule 通过 API 报告此设置，Satellite 可以在创建 DHCP 记录时检索配置信息。

当 PXE 加载器设置为 `none` 时，Satellite 不会将下一个服务器选项填充到 DHCP 记录中。

如果下一个服务器选项仍然未定义，Satellite 将使用反向 DNS 搜索来查找要分配的 TFTP 服务器地址，但您可能会遇到以下问题：

- 置备过程中的 DNS 超时
- 查询不正确的 DNS 服务器。例如，权威而不是缓存
- 有关 TFTP 服务器的 IP 地址不正确的错误。例如：PTR 记录无效

如果您遇到这些问题，请检查 Satellite 和 Capsule 上的 DNS 设置，特别是 PTR 记录解析。

filename 选项

`filename` 选项包含在置备过程中下载和执行的文件的完整路径。为主机或主机组选择的 PXE 装载程序定义了要使用的文件名选项。当 PXE 装载器设置为 `none` 时，Satellite 不会将 `filename` 选项填充 DHCP 记录。根据 PXE loader 选项，文件名更改如下：

PXE 加载程序选项	文件名条目	备注
PXELinux BIOS	<code>pxelinux.0</code>	
PXELinux UEFI	<code>pxelinux.efi</code>	
iPXE Chain BIOS	<code>undionly.kpxe</code>	
PXEGrub2 UEFI	<code>grub2/grubx64.efi</code>	X64 可能会因架构而异

PXE 加载程序选项	文件名条目	备注
iPXE UEFI HTTP	<code>\http://capsule.example.com:8000/httpboot/ipxe-x64.efi</code>	需要 httpboot 功能，并将文件名呈现为完整的 URL，其中在 Satellite 中是 Capsule 的已知主机名。
Grub2 UEFI HTTP	<code>\http://capsule.example.com:8000/httpboot/grub2/grubx64.efi</code>	需要 httpboot 功能，并将文件名呈现为完整的 URL，其中在 Satellite 中是 Capsule 的已知主机名。

3.5. SATELLITE 中的 DHCP 问题故障排除

Satellite 可以管理内部或外部 DHCP 胶囊上的 ISC DHCP 服务器。Satellite 可以列出、创建和删除 DHCP 保留和租期。但是，在一些问题中可能会遇到一些问题。

没有同步 DHCP 记录

当在 DHCP 编配过程中发生错误时，Satellite 数据库中的 DHCP 记录可能与 DHCP 服务器不匹配。要解决这个问题，您必须将缺少的 DHCP 记录从 Satellite 数据库添加到 DHCP 服务器中，然后按照以下步骤从 DHCP 服务器中删除不需要的记录：

流程

1. 要预览添加到 DHCP 服务器的 DHCP 记录，请输入以下命令：

```
# foreman-rake orchestration:dhcp:add_missing subnet_name=NAME
```

2. 如果您对上一步中的预览更改满意，请输入上述命令以及 `execute =1` 参数来应用它们：

```
# foreman-rake orchestration:dhcp:add_missing subnet_name=NAME perform=1
```

3. 要在 Satellite 和 DHCP 服务器中同步 DHCP 记录，您可以从 DHCP 服务器中删除不需要的 DHCP 记录。请注意，Satellite 假设所有管理的 DHCP 服务器不包含第三方记录，因此此步骤可能会删除这些意外记录。要预览要从 DHCP 服务器中删除哪些记录，请输入以下命令：

```
# foreman-rake orchestration:dhcp:remove_offending subnet_name=NAME
```

4. 如果您对上一步中的预览更改满意，请输入上述命令以及 `execute =1` 参数来应用它们：

```
# foreman-rake orchestration:dhcp:remove_offending subnet_name=NAME perform=1
```

PXE 加载程序选项更改

当为现有主机更改 PXE loader 选项时，这会导致 DHCP 冲突。唯一的解决方法是覆盖 DHCP 条目。

DHCP 文件的权限不正确

操作系统更新可以更新 `dhcpd` 软件包。这会导致重置重要目录和文件的权限，以便 DHCP 胶囊无法读取所需的信息。

如需更多信息，请参阅 [Satellite 服务器 Error ERF12-6899 ProxyAPI::ProxyException: Unable to set DHCP entry RestClient::ResourceNotFound 404 Resource Not Found on Red Hat Knowledgebase](#)。

更改 DHCP Capsule 条目

Satellite 仅管理分配给设置了 DHCP 胶囊集的子网的 DHCP 记录。如果您创建主机，然后清除或更改 DHCP Capsule，当您尝试删除主机时，操作会失败。

如果您在没有设置 DHCP Capsule 的情况下创建主机，然后尝试设置 DHCP 胶囊，这会导致 DHCP 冲突。

删除 `dhcpd.leases` 文件中的主机条目

对 DHCP 租期的任何更改都会附加到 `dhcpd.leases` 文件的末尾。由于条目附加到文件中，因此同一租期的两个或多个条目可以同时存在于 `dhcpd.leases` 文件中。当同一租期有两个或者多个条目时，文件中的最后一个条目将具有优先权。租期文件中的组、子组和主机声明以同样的方式处理。如果删除了租期，`{ deleted; }` 会附加到声明中。

3.6. 基于镜像置备的先决条件

启动后配置方法

使用完成后引导配置脚本的镜像需要一个受管 DHCP 服务器，如 Satellite 的集成胶囊或外部 Capsule。主机必须使用与 DHCP 胶囊关联的子网创建，并且主机的 IP 地址必须是 DHCP 范围内的有效 IP 地址。

可以使用外部 DHCP 服务，但必须手动输入 IP 地址。必须与镜像中配置与镜像中配置对应的 SSH 凭据必须在 Satellite 中配置，以启用启动后配置。

当虚拟机从依赖于后配置脚本的镜像引导时，检查以下项目：

- 主机在 Satellite 服务器中分配了子网。
- 子网在 Satellite 服务器中分配了 DHCP 胶囊。
- 主机在 Satellite 服务器中分配了有效的 IP 地址。
- 使用 DHCP 的虚拟机获取的 IP 地址与 Satellite 服务器中配置的地址匹配。
- 从镜像创建的虚拟机响应 SSH 请求。
- 从镜像创建的虚拟机通过 SSH 授权用户和密码（与要部署的镜像相关联）。
- Satellite 服务器可以通过 SSH 密钥访问虚拟机。虚拟机需要从 Satellite 服务器接收后配置脚本。

预引导初始化配置方法

使用 `cloud-init` 脚本的镜像需要 DHCP 服务器以避免在镜像中包含 IP 地址。首选使用受管 DHCP Capsule。该镜像必须将 `cloud-init` 服务配置为在系统引导时启动，并获取用于在完成配置中使用的脚本或配置数据。

当从依赖于镜像中包含的初始化脚本的镜像引导的虚拟机引导时，请检查以下项目：

- 子网中有一个 DHCP 服务器。

- 虚拟机已安装并启用 `cloud-init` 服务。

有关虚拟机镜像中完成和 `cloud-init` 脚本的不同支持级别的详情，请查看红帽知识库解决方案在红帽客户门户网站上 [针对完成和 `cloud-init` 脚本支持的计算资源是什么](#)。

3.7. 配置网络服务

某些调配方法使用 `Capsule` 服务器服务。例如，网络可能需要 `Capsule` 服务器充当 `DHCP` 服务器。网络也可以使用 `PXE` 引导服务在新主机上安装操作系统。这需要将 `Capsule` 服务器配置为使用主 `PXE` 引导服务：`DHCP`、`DNS` 和 `TFTP`。

使用 `satellite-installer` 命令和选项在 `Satellite` 服务器上配置这些服务。

要在外部 `Capsule` 服务器上配置这些服务，请运行 `satellite-installer --scenario mailbox`。

`Satellite` 服务器使用 `eth0` 进行外部通信，如连接到红帽的 `CDN`。

流程

1. 输入 `satellite-installer` 命令来配置所需的网络服务：

```
# satellite-installer --foreman-proxy-dhcp true \
--foreman-proxy-dhcp-gateway "192.168.140.1" \
--foreman-proxy-dhcp-interface "eth1" \
--foreman-proxy-dhcp-managed true \
--foreman-proxy-dhcp-nameservers "192.168.140.2" \
--foreman-proxy-dhcp-range "192.168.140.10 192.168.140.110" \
--foreman-proxy-dhcp-server "192.168.140.2" \
--foreman-proxy-dns true \
--foreman-proxy-dns-forwarders "8.8.8.8; 8.8.4.4" \
--foreman-proxy-dns-interface "eth1" \
--foreman-proxy-dns-managed true \
--foreman-proxy-dns-reverse "140.168.192.in-addr.arpa" \
--foreman-proxy-dns-server "127.0.0.1" \
--foreman-proxy-dns-zone "example.com" \
--foreman-proxy-tftp true \
--foreman-proxy-tftp-managed true
```

2. 查找您配置的 `Capsule` 服务器：

```
# hammer proxy list
```

3. 刷新 Capsule 服务器的功能以查看更改：

```
# hammer proxy refresh-features --name "satellite.example.com"
```

4. 验证在 Capsule 服务器上配置的服务：

```
# hammer proxy info --name "satellite.example.com"
```

3.7.1. 使用安装程序的多个子网或域

`satellite-installer` 选项只允许单个 DHCP 子网或 DNS 域。定义多个子网的一种方法是使用自定义配置文件。

对于每个附加子网或域，在 `/etc/foreman-installer/custom-hiera.yaml` 文件中创建一个条目：

```
dhcp::pools:
  isolated.lan:
    network: 192.168.99.0
    mask: 255.255.255.0
    gateway: 192.168.99.1
    range: 192.168.99.5 192.168.99.49

dns::zones:
  # creates @ SOA $::fqdn root.example.com.
  # creates $::fqdn A $::ipaddress
  example.com: {}

  # creates @ SOA test.example.net. hostmaster.example.com.
  # creates test.example.net A 192.0.2.100
  example.net:
    soa: test.example.net
    soaip: 192.0.2.100
    contact: hostmaster.example.com.

  # creates @ SOA $::fqdn root.example.org.
  # does NOT create an A record
  example.org:
    reverse: true

  # creates @ SOA $::fqdn hostmaster.example.com.
  2.0.192.in-addr.arpa:
    reverse: true
    contact: hostmaster.example.com.
```

执行 `satellite-installer` 执行更改，并验证 `/etc/dhcp/dhcpd.conf` 是否包含适当的条目。然后，必须

在 **Satellite** 数据库中定义子网。

3.7.2. 网络配置的 DHCP 选项

--foreman-proxy-dhcp

启用 DHCP 服务。您可以将这个选项设置为 **true** 或 **false**。

--foreman-proxy-dhcp-managed

启用 Foreman 管理 DHCP 服务。您可以将这个选项设置为 **true** 或 **false**。

--foreman-proxy-dhcp-gateway

DHCP 池网关。把它设置为您专用网络上主机的外部网关地址。

--foreman-proxy-dhcp-interface

设置 DHCP 服务的接口以侦听请求。把它设置为 **eth1**。

--foreman-proxy-dhcp-nameservers

设置通过 DHCP 向客户端提供的名称服务器地址。把它设置为 **eth1** 上 **Satellite** 服务器的地址。

--foreman-proxy-dhcp-range

用于发现和非受管服务的空格分隔 DHCP 池范围。

--foreman-proxy-dhcp-server

设置要管理的 DHCP 服务器的地址。

运行 **satellite-installer --help** 以查看与 DHCP 和其他 Capsule 服务相关的更多选项。

3.7.3. 网络配置的 DNS 选项

--foreman-proxy-dns

启用 DNS 功能。您可以将这个选项设置为 **true** 或 **false**。

--foreman-proxy-dns-provider

选择要使用的提供程序。

--foreman-proxy-dns-managed

让安装程序管理 ISC BIND。这只在使用 `nsupdate` 和 `nsupdate_gss` 供应商时才相关。您可以将这个选项设置为 `true` 或 `false`。

`--foreman-proxy-dns-forwarders`

设置 DNS 转发器。仅在安装程序管理 ISC BIND 时使用。把它设置为您的 DNS 递归器。

`--foreman-proxy-dns-interface`

将接口设置为侦听 DNS 请求。仅在安装程序管理 ISC BIND 时使用。把它设置为 `eth1`。

`--foreman-proxy-dns-reverse`

DNS 反向区域名称。仅在安装程序管理 ISC BIND 时使用。

`--foreman-proxy-dns-server`

设置 DNS 服务器的地址。仅供 `nsupdate`、`nsupdate_gss` 和 `infoblox` 供应商使用。

`--foreman-proxy-dns-zone`

设置 DNS 区域名称。仅在安装程序管理 ISC BIND 时使用。

运行 `satellite-installer --help` 以查看与 DNS 和其他 Capsule 服务相关的更多选项。

3.7.4. 网络配置的 TFTP 选项

`--foreman-proxy-tftp`

启用 TFTP 服务。您可以将这个选项设置为 `true` 或 `false`。

`--foreman-proxy-tftp-managed`

启用 Foreman 管理 TFTP 服务。您可以将这个选项设置为 `true` 或 `false`。

`--foreman-proxy-tftp-servername`

设置要使用的 TFTP 服务器。确保您使用 Capsule 的 IP 地址。

运行 `satellite-installer --help` 以查看与 TFTP 和其他 Capsule 服务相关的更多选项。

3.7.5. 使用 TFTP 服务通过 NAT

您可以通过 NAT 使用 Satellite TFTP 服务。要做到这一点，在所有 NAT 路由器或防火墙上，您必须在 UDP 端口 69 上启用 TFTP 服务，并启用 TFTP 状态跟踪功能。如需更多信息，请参阅 NAT 设备的文档。

在 Red Hat Enterprise Linux 7 中使用 NAT：

- 使用以下命令允许 UDP 端口 69 上的 TFTP 服务，载入内核 TFTP 状态跟踪模块，并使更改持久：

```
# firewall-cmd --add-service=tftp && firewall-cmd --runtime-to-permanent
```

在 Red Hat Enterprise Linux 6 中使用 NAT：

1. 将防火墙配置为在端口 69 上允许 TFTP 服务 UDP：

```
# iptables \  
--sport 69 \  
--state ESTABLISHED \  
-A OUTPUT \  
-i eth0 \  
-j ACCEPT \  
-m state \  
-p udp \  
# service iptables save
```

2. 加载 `ip_conntrack_tftp` 内核 TFTP 状态模块。在 `/etc/sysconfig/iptables-config` 文件中，找到 `IPTABLES_MODULES` 并添加 `ip_conntrack_tftp`，如下所示：

```
IPTABLES_MODULES="ip_conntrack_tftp"
```

3.8. 将域添加到 SATELLITE 服务器

Satellite 服务器为网络上的每个主机定义域名。Satellite 服务器必须具有负责域名分配的域和 Capsule 服务器的信息。

检查现有域

Satellite 服务器可能已经创建相关的域作为 Satellite 服务器安装的一部分。将上下文切换到任何机构，以及任何位置，然后检查域列表以查看它是否存在。

DNS 服务器配置注意事项

在创建 DNS 记录期间，Satellite 执行冲突 DNS 查找，以验证主机名不在活跃的使用中。此检查针对以下 DNS 服务器之一运行：

- 如果 `managing > Settings & gt; Query local nameserver` 设置为 `true`，则系统范围的解析器。
- 与主机关联的子网中定义的名称服务器。
- 从与主机关联的域名查询的 authoritative NS-Records。

如果您在 DNS 冲突解析过程中遇到超时，请检查以下设置：

- 子网名称服务器必须可从 Satellite 服务器访问。
- 域名必须有从 Satellite 服务器提供的授权起始(SOA)记录。
- `/etc/resolv.conf` 文件中的系统解析器必须具有有效且正常工作的配置。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Infrastructure > Domains** 并点 **Create Domain**。
2. 在 **DNS Domain** 字段中，输入完整的 DNS 域名。
3. 在 **Fullname** 字段中，输入域的纯文本名称。
4. 单击 **Parameters** 选项卡，再配置任何域级别参数以应用到附加到此域的主机。例如，用户定义的布尔值或字符串参数，用于模板中。

5. 点 **Add Parameter** 并填写 **Name** 和 **Value** 字段。
6. 单击 **Locations** 选项卡，再添加域所在的位置。
7. 单击 **Organizations** 选项卡，再添加域所属的组织。
8. 点 **Submit** 保存更改。

CLI 过程

- 使用 `hammer domain create` 命令创建域：

```
# hammer domain create \  
--description "My_Domain" \  
--dns-id My_DNS_ID \  
--locations "My_Location" \  
--name "my-domain.tld" \  
--organizations "My_Organization"
```

在本例中，`--dns-id` 选项使用 `1`，这是您在 **Satellite** 服务器上集成的 **Capsule** 的 ID。

3.9. 将子网添加到 SATELLITE 服务器

您必须为每个子网添加信息到 **Satellite** 服务器，因为 **Satellite** 为新主机配置接口。要配置接口，**Satellite** 服务器必须具有连接这些接口的网络的所有信息。

要使用 CLI 而不是 **Satellite** Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 **Satellite** Web UI 中，导航到 **Infrastructure > Subnets**，然后在 **Subnets** 窗口中点 **Create Subnet**。
2. 在 **Name** 字段中输入子网的名称。

3. 在 **Description** 字段中，输入子网的描述。
4. 在 **Network address** 字段中，输入子网的网络地址。
5. 在 **Network prefix** 字段中，输入子网的网络前缀。
6. 在 **Network mask** 字段中，输入子网的网络掩码。
7. 在 **Gateway address** 字段中，输入子网的外部网关。
8. 在 **Primary DNS server** 字段中，为子网输入主 DNS。
9. 在二级 DNS 服务器中，为子网输入二级 DNS。
10. 在 IPAM 列表中，选择要用于 IP 地址管理(IPAM)的方法。有关 IPAM 的详情，请参考 [第 3 章 配置网络](#)。
11. 输入您选择的 IPAM 方法的信息。单击 **Remote Execution** 选项卡，再选择控制远程执行的 **Capsule**。
12. 单击 **Domains** 选项卡，再选择应用到此子网的域。
13. 点 **Capsules** 选项卡，再选择适用于子网中各个服务的 **Capsule**，包括 DHCP、TFTP 和反向 DNS 服务。
14. 单击 **Parameters** 选项卡，再配置任何子网级别参数以应用到附加到此子网的主机。例如，用户定义的布尔值或字符串参数，用于模板中。
15. 单击 **Locations** 选项卡，再选择使用此 **Capsule** 的位置。

16. 单击 **Organizations** 选项卡，再选择使用此 **Capsule** 的组织。
17. 点 **Submit** 保存子网信息。

CLI 过程

- 使用以下命令创建子网：

```
# hammer subnet create \  
--boot-mode "DHCP" \  
--description "My_Description" \  
--dhcp-id My_DHCP_ID \  
--dns-id My_DNS_ID \  
--dns-primary "192.168.140.2" \  
--dns-secondary "8.8.8.8" \  
--domains "my-domain.tld" \  
--from "192.168.140.111" \  
--gateway "192.168.140.1" \  
--ipam "DHCP" \  
--locations "_My_Location" \  
--mask "255.255.255.0" \  
--name "My_Network" \  
--network "192.168.140.0" \  
--organizations "My_Organization" \  
--tftp-id My_TFTP_ID \  
--to "192.168.140.250" \  

```



注意

在本例中，`--dhcp-id`、`--dns-id` 和 `--tftp-id` 选项使用 `1`，这是 **Satellite** 服务器中集成的 **Capsule** 的 ID。

第 4 章 使用 INFOBLOX 作为 DHCP 和 DNS 提供商

您可以使用 Capsule 服务器连接到 Infoblox 应用程序，以创建和管理 DHCP 和 DNS 记录，以及保留 IP 地址。

支持的 Infoblox 版本为 NIOS 8.0 或更高版本，Satellite 6.11 或更高版本。

4.1. INFOBLOX 限制

所有 DHCP 和 DNS 记录只能在单一网络或 DNS 视图中管理。在 Capsule 上安装 Infoblox 模块并使用 `satellite-installer` 命令设置视图后，您无法编辑视图。

Capsule 服务器使用标准 HTTPS Web API 与单个 Infoblox 节点通信。如果要配置集群和高可用性，请在 Infoblox 中进行配置。

不支持使用 Infoblox 的 TFTP 功能托管与 PXE 相关的文件。您必须使用 Capsule 作为 TFTP 服务器进行 PXE 置备。更多信息请参阅 [第 3 章 配置网络](#)。

Satellite IPAM 功能无法与 Infoblox 集成。

4.2. INFOBLOX 先决条件

- 您必须具有 Infoblox 帐户凭证来管理 Satellite 中的 DHCP 和 DNS 条目。
- 确保您具有 Infoblox 管理角色，其名称为：DHCP Admin 和 DNS Admin。
- 管理角色必须具有权限或属于 admin 组，允许帐户通过 Infoblox API 执行任务。

4.3. 在 CAPSULE 服务器上安装 INFOBLOX CA 证书

您必须在基本系统上安装 Infoblox HTTPS CA 证书，以用于您要与 Infoblox 应用程序集成的所有 Capsule。

您可以从 **Infoblox web UI** 下载证书，或者您可以使用以下 **OpenSSL** 命令下载证书：

```
# update-ca-trust enable
# openssl s_client -showcerts -connect infoblox.example.com:443 </dev/null | \
openssl x509 -text >/etc/pki/ca-trust/source/anchors/infoblox.crt
# update-ca-trust extract
```

- **infoblox.example.com** 条目必须与 X509 证书中的 Infoblox 应用程序的主机名匹配。

要测试 CA 证书，请使用 **CURL** 查询：

```
# curl -u admin:password https://infoblox.example.com/wapi/v2.0/network
```

正响应示例：

```
[
  {
    "_ref":
    "network/ZG5zLm5ldHdvcmskMTkyLjE2OC4yMDluMC8yNC8w:infoblox.example.com/24/default",
    "network": "192.168.202.0/24",
    "network_view": "default"
  }
]
```

使用以下红帽知识库文章安装证书：[如何在 Red Hat Enterprise Linux 6 / 7 中安装 CA 证书。](#)

4.4. 安装 DHCP INFOBLOX 模块

使用这个流程在 **Capsule** 上安装 **DHCP Infoblox** 模块。请注意，您无法在单独的视图中管理记录。

您还可以通过组合这个过程和 [第 4.5 节“安装 DNS Infoblox 模块”](#) 来同时安装 **DHCP** 和 **DNS Infoblox** 模块。

DHCP Infoblox 记录类型注意事项

仅使用 `--foreman-proxy-plugin-dhcp-infoblox-record-type fixedaddress` 选项来配置 **DHCP** 和 **DNS** 模块。

使用主机记录类型设置配置 DHCP 和 DNS Infoblox 模块会导致 DNS 冲突且不被支持。如果您在 Capsule 服务器上安装 Infoblox 模块，并将 `--foreman-proxy-plugin-dhcp-infoblox-record-type` 选项设置为 `host`，您必须取消设置 DNS Capsule 和 Reverse DNS Capsule 选项，因为 Infoblox 执行 DNS 管理本身。您不能在不创建冲突的情况下使用 `host` 选项，例如无法在 Satellite 中重命名主机。

流程

1.

在 Capsule 中输入以下命令：

```
# satellite-installer --enable-foreman-proxy-plugin-dhcp-infoblox \
--foreman-proxy-dhcp true \
--foreman-proxy-dhcp-managed false \
--foreman-proxy-dhcp-provider infoblox \
--foreman-proxy-dhcp-server infoblox.example.com \
--foreman-proxy-plugin-dhcp-infoblox-dns-view default \
--foreman-proxy-plugin-dhcp-infoblox-network-view default \
--foreman-proxy-plugin-dhcp-infoblox-password infoblox \
--foreman-proxy-plugin-dhcp-infoblox-record-type fixedaddress \
--foreman-proxy-plugin-dhcp-infoblox-username admin
```

2.

在 Satellite Web UI 中，导航到 **Infrastructure > Capsules**，再选择带有 Infoblox DHCP 模块的 Capsule，然后单击 **Refresh**。

3.

确保列出了 dhcp 功能。

4.

对于通过 Infoblox 管理的所有域，请确保为该域设置了 DNS 胶囊。要验证，在 Satellite Web UI 中，导航到 **Infrastructure > Domains**，并检查每个域的设置。

5.

对于通过 Infoblox 管理的所有子网，请确保设置了 DHCP Capsule 和 Reverse DNS Capsule。要验证，在 Satellite Web UI 中，导航到 **Infrastructure > Subnets**，并检查每个子网的设置。

4.5. 安装 DNS INFOBLOX 模块

使用这个流程在 Capsule 上安装 DNS Infoblox 模块。您还可以通过组合这个过程和 [第 4.4 节“安装 DHCP Infoblox 模块”](#) 来同时安装 DHCP 和 DNS Infoblox 模块。

DNS 记录只在默认的 DNS 视图中管理，无法指定要使用的 DNS 视图。

流程

1. 在 **Capsule** 上，输入以下命令配置 **Infoblox** 模块：

```
# satellite-installer --enable-foreman-proxy-plugin-dns-infoblox \  
--foreman-proxy-dns true \  
--foreman-proxy-dns-provider infoblox \  
--foreman-proxy-plugin-dns-infoblox-dns-server infoblox.example.com \  
--foreman-proxy-plugin-dns-infoblox-dns-view default \  
--foreman-proxy-plugin-dns-infoblox-password infoblox \  
--foreman-proxy-plugin-dns-infoblox-username admin
```

另外，您可以更改 `--foreman-proxy-plugin-dns-infoblox-dns-view` 选项的值，以指定默认视图以外的 **DNS Infoblox** 视图。

2. 在 **Satellite Web UI** 中，导航到 **Infrastructure > Capsules**，再选择带有 **Infoblox DNS** 模块的 **Capsule**，然后单击 **Refresh**。
3. 确保列出了 **dns** 功能。

第 5 章 配置 IPXE 以减少置备时间

您可以使用 Satellite 将 PXELinux 配置为以 BIOS 模式链 iPXE，并有以下限制来阻止您使用 PXE：

- 具有非受管 DHCP 服务器的网络。
- 在网络上列入黑名单或受防火墙限制的 PXE 服务。
- 一个不可靠的 TFTP UDP 协议，因为一个低带宽网络。

有关 iPXE 支持的更多信息，[请参阅支持的部署架构](#)。

iPXE 概述

iPXE 是开源网络引导固件。它提供增强的额外功能的完整 PXE 实施，包括从 HTTP 服务器引导。如需更多信息，[请参阅 ipxe.org](#)。

在 Red Hat Satellite 中使用 iPXE 有三种方法：

1. 使用 iPXE 作为主固件的虚拟机监控程序启动虚拟机。
2. 通过 TFTP 使用 PXELinux 直接在裸机主机上加载 iPXE。
3. 使用 PXELinux 到 UNDI，它使用 HTTP 传输内核和裸机主机上的初始 RAM 磁盘。

安全信息

Red Hat Enterprise Linux 中的 iPXE 二进制文件是在没有一些安全功能的情况下构建的。因此，您只能使用 HTTP，无法使用 HTTPS。不支持 Red Hat Enterprise Linux 中 iPXE 的所有与安全相关的功能。如需更多信息，[请参阅 iPXE 中的 Red Hat Enterprise Linux HTTPS 支持](#)。

先决条件

- **Red Hat Satellite 中存在主机。**
- **置备接口的 MAC 地址与主机配置匹配。**
- **主机的调配接口具有有效的 DHCP 保留。**
- **NIC 可以进行 PXE 引导。如需更多信息，请参阅 ipxe.org 上支持的硬件，以了解希望与基于 iPXE 的引导磁盘一起使用的硬件驱动程序列表。**
- **NIC 与 iPXE 兼容。**

要准备 iPXE 环境，您必须在所有胶囊上执行此流程。

流程

1.

启用 tftp 和 httpboot 服务：

```
# satellite-installer --foreman-proxy-httpboot true --foreman-proxy-tftp true
```

2.

安装 ipxe-bootimgs 软件包：

```
# yum install ipxe-bootimgs
```

3.

更正 SELinux 文件上下文：

```
# restorecon -RvF /var/lib/tftpboot/
```

•

将带有 Linux 内核标头的 iPXE 固件复制到 TFTP 目录中：

```
# cp /usr/share/ipxe/ipxe.lkern /var/lib/tftpboot/
```

•

将 UNDI iPXE 固件复制到 TFTP 目录中：

```
# cp /usr/share/ipxe/undionly.kpxe /var/lib/tftpboot/undionly-ipxe.0
```

4.

(可选) 配置 Foreman 发现。更多信息请参阅 [第 7 章 配置发现服务](#)。

- 在 Satellite Web UI 中，导航到 **Administer > Settings**，然后点 **Provisioning** 选项卡。
- 在 **Default PXE** 全局模板条目行和 **Value** 列中找到 **Default PXE** 全局模板条目行，将值更改为 **discovery**。

5.1. 引导虚拟机

有些虚拟化虚拟机监控程序使用 iPXE 作为 PXE 引导的主要固件。因此，您可以在没有 TFTP 和 PXELinux 的情况下引导虚拟机。

Chainbooting 虚拟机 workflow

使用虚拟化虚拟机监控程序消除了 TFTP 和 PXELinux 的要求。它有以下 workflow：

1. **虚拟机启动**
2. **iPXE 使用 DHCP 检索网络凭据，包括 HTTP URL**
3. **iPXE 从 Satellite 服务器或 Capsule 加载 iPXE bootstrap 模板**
4. **iPXE 从 Satellite 服务器或 Capsule 加载 MAC 的 iPXE 模板作为 URL 参数**
5. **iPXE 加载安装程序的内核和初始 RAM 磁盘**

先决条件

- 确保要使用的虚拟机监控程序支持 iPXE。以下虚拟化虚拟机监控程序支持 iPXE：
 -

libvirt

- **Red Hat Virtualization**
- **RHEV (已弃用)**
- **如果要使用 Capsule 服务器而不是 Satellite 服务器，请确保已相应地配置了 Capsule 服务器。如需更多信息，请参阅安装 [Capsule 服务器](#) 中的 [为主机注册和置备配置 Capsule](#)。**

将 Satellite 服务器配置为使用 iPXE

您可以使用默认模板为主机配置 iPXE 引导。如果要更改模板中的默认值，请克隆模板并编辑克隆。

流程

1. 在 Satellite Web UI 中，导航到 **Hosts > Provisioning Templates**，输入 **Kickstart default iPXE** 并点 **Search**。
2. 可选：如果要更改模板，点 **Clone**，输入唯一名称，然后点 **Submit**。
3. 点您要使用的模板的名称。
4. 如果克隆模板，您可以在 **Template** 选项卡中进行更改。
5. 单击 **关联** 选项卡，然后选择主机使用的操作系统。
6. 单击 **Locations** 选项卡，再添加主机所在的位置。
7. 单击 **Organizations** 选项卡，再添加主机所属的组织。
8. 点 **Submit** 保存更改。

9. 在 Satellite Web UI 中，导航到 **Hosts > Operating systems** 并选择主机的操作系统。
10. 点 **Templates** 选项卡。
11. 从 **iPXE Template** 列表中，选择您要使用的模板。
12. 点 **Submit** 保存更改。
13. 在 Satellite Web UI 中，导航到 **Hosts > All Hosts**。
14. 在 **Hosts** 页面中，选择您要使用的主机。
15. 选择 **Operating System** 选项卡。
16. 将 **PXE Loader** 设置为 **iPXE 嵌入式**。
17. 选择 **Templates** 选项卡。
18. 从 **iPXE 模板** 列表中，选择 **Review** 以验证 **Kickstart** 默认 **iPXE 模板** 是否正确模板。
19. 设置 **HTTP URL**。

- 如果要使用 **Satellite** 服务器引导，请在 **Satellite** 服务器上运行以下命令：

```
# satellite-installer \  
--foreman-proxy-dhcp-ipxefilename "http://satellite.example.com/unattended/iPXE?  
bootstrap=1"
```

- 如果要使用 **Capsule** 进行引导，在 **Capsule** 上运行以下命令：

```
# satellite-installer \  
--foreman-proxy-dhcp-pxefilename "http://capsule.example.com/unattended/iPXE?  
bootstrap=1"
```

5.2. 从 PXELINUX 启动 IPXE

使用这个流程将 iPXE 设置为将内置驱动程序用于网络通信或 UNDI 接口。要将 HTTP 与 iPXE 搭配使用，请使用带有内置驱动程序(ipxe.lkrn)的 iPXE 构建。通用网络设备接口(UNDI)是一种最小的 UDP/IP 堆栈，它实现了 TFTP 客户端，但不支持 HTTP 等其他协议(undionly-ipxe.0)。根据网络硬件功能和 iPXE 驱动程序的可用性，您可以选择 load ipxe.lkrn 或 undionly-ipxe.0 文件。

直接或带有 UNDI 工作流的Chainbooting iPXE

1. 主机电源
2. PXE 驱动程序使用 DHCP 检索网络凭证
3. PXE 驱动程序使用 TFTP 检索 PXELinux 固件 pxelinux.0
4. pxelinux 搜索 TFTP 服务器上的配置文件
5. PXELinux chainloads iPXE ipxe.lkrn or undionly-ipxe.0
6. iPXE 再次使用 DHCP 检索网络凭证，包括 HTTP URL
7. iPXE 链从模板 Capsule 加载 iPXE 模板
8. iPXE 加载安装程序的内核和初始 RAM 磁盘

前提条件

- 如果要使用 Capsule 服务器而不是 Satellite 服务器，请确保已相应地配置了 Capsule 服务器。如需更多信息，请参阅安装 [Capsule 服务器](#) 中的 [为主机注册和置备配置 Capsule](#)。

将 Satellite 服务器配置为使用 iPXE

您可以使用默认模板为主机配置 iPXE 引导。如果要更改模板中的默认值，请克隆模板并编辑克隆。

流程

1. 在 Satellite Web UI 中，导航到 **Hosts > Provisioning Templates**。
2. 输入 PXELinux 链 iPXE 来使用 `ipxe.lkrn` 或对于 BIOS 系统，输入 PXELinux 链 iPXE UNDI 来使用 `undionly-ipxe.0`，然后单击 **Search**。
3. 可选：如果要更改模板，点 **Clone**，输入唯一名称，然后点 **Submit**。
4. 点您要使用的模板的名称。
5. 如果克隆模板，您可以在 **Template** 选项卡中进行更改。
6. 单击 **关联** 选项卡，然后选择主机使用的操作系统。
7. 单击 **Locations** 选项卡，再添加主机所在的位置。
8. 单击 **Organizations** 选项卡，再添加主机所属的组织。
9. 点 **Submit** 保存更改。
10. 在 **Provisioning Templates** 页面中，在搜索字段中输入 `Kickstart default iPXE`，然后单击 **Search**。
11. 可选：如果要更改模板，点 **Clone**，输入唯一名称，然后点 **Submit**。
12. 点您要使用的模板的名称。

13. 如果克隆模板，您可以在 **Template** 选项卡中进行更改。
14. 单击 **关联** 选项卡，然后将模板与主机使用的操作系统相关联。
15. 单击 **Locations** 选项卡，再添加主机所在的位置。
16. 单击 **Organizations** 选项卡，再添加主机所属的组织。
17. 点 **Submit** 保存更改。
18. 在 **Satellite Web UI** 中，导航到 **Hosts > Operating systems** 并选择主机的操作系统。
19. 点 **Templates** 选项卡。
20. 从 **PXELinux** 模板列表中，选择您要使用的模板。
21. 从 **iPXE** 模板列表中，选择您要使用的模板。
22. 点 **Submit** 保存更改。
23. 在 **Satellite Web UI** 中，导航到 **Configure > Host Groups**，然后选择您要配置的主机组。
24. 选择 **Operating System** 选项卡。
25. 选择 **架构和操作系统**。
26. 将 **PXE Loader** 设置为 **PXELinux BIOS**，以通过 **PXELinux** 的 **chainboot iPXE**，或设置为 **iPXE Chain BIOS** 来直接加载 **undionly-ipxe.0**。

27.

设置 HTTP URL。

- **如果要使用 Satellite 服务器引导，请在 Satellite 服务器上运行以下命令：**

```
# satellite-installer \  
--foreman-proxy-dhcp-ipxefilename "http://satellite.example.com/unattended/iPXE?  
bootstrap=1"
```

- **如果要使用 Capsule 进行引导，在 Capsule 上运行以下命令：**

```
# satellite-installer \  
--foreman-proxy-dhcp-ipxefilename "http://capsule.example.com/unattended/iPXE?  
bootstrap=1"
```

第 6 章 使用 PXE 置备主机

您可以使用以下方法之一使用 Satellite 置备裸机实例：

无人值守置备

新主机由 MAC 地址标识，并且 Satellite 服务器使用 PXE 引导过程调配主机。

使用 Discovery 进行无人值守置备

新主机使用 PXE 引导来加载 Satellite 发现服务。此服务识别主机的硬件信息，并将其列为要置备的可用主机。更多信息请参阅 [第 7 章 配置发现服务](#)。

无 PXE 置备

新主机通过 Satellite 服务器生成的引导磁盘镜像来置备。

BIOS 和 UEFI 支持

使用 Red Hat Satellite，您可以同时执行基于 BIOS 和 UEFI 的 PXE 置备。BIOS 和 UEFI 接口都可以作为计算机操作系统和固件之间的解释器，初始化硬件组件并在引导时启动操作系统。

有关支持的工作流的详情，请参考 [支持的架构和置备场景](#)。

在 Satellite 置备中，PXE 加载程序选项定义在调配期间使用的 DHCP 文件名选项。对于 BIOS 系统，使用 PXELinux BIOS 选项启用置备的节点通过 TFTP 下载 pxelinux.0 文件。对于 UEFI 系统，使用 PXEGrub2 UEFI 选项启用 TFTP 客户端下载 grub2/grubx64.efi 文件，或使用 PXEGrub2 UEFI HTTP 选项启用 UEFI HTTP 客户端从带有 HTTP 引导功能的 Capsule 下载 grubx64.efi。

对于 BIOS 置备，您必须将 PXELinux 模板与操作系统关联。对于 UEFI 置备，您必须将 PXEGrub2 模板与操作系统关联。如果您同时关联 PXELinux 和 PXEGrub2 模板，Satellite 可以在 TFTP 服务器中为两个部署配置文件，以便您可以轻松地在 PXE 加载程序间切换。

6.1. 裸机置备的先决条件

裸机置备的要求包括：

- 管理裸机主机网络的 Capsule 服务器。对于无人值守置备和基于发现的调配，Satellite 服务器需要 PXE 服务器设置。

有关网络要求的详情请参考 [第 3 章 配置网络](#)。

有关 Discovery 服务 [第 7 章 配置发现服务](#) 的更多信息。

- 裸机主机或空白虚拟机。
- 您可以将同步的内容软件仓库用于 Red Hat Enterprise Linux。有关更多信息，[请参阅 内容管理指南中的同步存储库](#)。
- 为主机注册提供激活码。如需更多信息，[请参阅 内容管理指南中的创建激活码](#)。

有关无人值守和无 PXE 置备的安全令牌的详情，请参考 [第 6.2 节 “配置安全令牌服务持续时间”](#)。

6.2. 配置安全令牌服务持续时间

在执行任何类型的置备时，作为安全措施，Satellite 会自动生成唯一令牌，并将此令牌添加到 PXE 配置文件(PXELinux, Grub2)中的 kickstart URL 中。默认情况下，令牌在 360 分钟内有效。当您调配主机时，请确保重启此时间栏中的主机。如果令牌过期，它不再有效，您会收到 404 错误，操作系统安装程序下载会失败。

流程

1. 在 Satellite Web UI 中，导航到 **Administer > Settings**，然后点 **Provisioning** 选项卡。
2. 找到 **Token duration** 选项并点击编辑图标并编辑持续时间，或者输入 0 来禁用令牌生成。如果令牌生成被禁用，攻击者可以欺骗客户端 IP 地址并从 Satellite 服务器下载 kickstart，包括加密的 root 密码。

6.3. 使用未置备创建主机

无人值守调配是最简单的主机调配形式。您可以在 Satellite 服务器上输入主机详情并引导您的主机。Satellite 服务器自动管理 PXE 配置、组织网络服务，并为主机提供操作系统和配置。

这种置备主机的方法在此过程中使用最小的交互。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Hosts > Create Host**。
2. 在 **Name** 字段中输入主机的名称。
3. 点 **Organization and Location** 选项卡，更改上下文以符合您的要求。
4. 从 **Host Group** 列表中，选择要用于填充表单的主机组。
5. 单击 **Interface** 选项卡，然后在主机的接口上点 **Edit**。
6. 验证这些字段是否填充了值。特别是：
 - **Host** 选项卡中的 **Name** 成为 DNS 名称。
 - **Satellite** 服务器自动为新主机分配 IP 地址。
7. 在 **MAC 地址** 字段中，输入主机的 **MAC** 地址。这样可确保在 **PXE** 引导过程中识别主机。
8. 确保 **Satellite** 服务器自动为主机上的第一个接口选择 **Managed**、**Primary** 和 **Provision** 选项。如果没有，请选择它们。
9. 单击 **OK** 保存。要添加另一个接口，请点击 **Add Interface**。您只能为 **Provision** 和 **Primary** 选择一个接口。

10. 点 **Operating System** 选项卡，并验证所有字段都包含值。确认操作系统的每个方面。

11. 可选：点 **Resolve in Provisioning** 模板检查新主机可以识别要使用的正确置备模板。

有关关联置备模板的更多信息，请参阅 [第 2.11 节“置备模板”](#)。

12. 单击 **Parameters** 选项卡，并确保存在一个提供激活码的参数。如果没有，请添加激活码。

13. 点 **Submit** 保存主机详情。

有关网络接口的更多信息，请参阅 [添加网络接口](#)。

这将创建主机条目和相关置备设置。这也包括为 PXE 引导裸机主机创建必要的目录和文件。如果您启动物理主机并将其引导模式设置为 PXE，则主机检测到 Satellite 服务器集成胶囊的 DHCP 服务，接收 Kickstart 树的 HTTP 端点并安装操作系统。

安装完成后，主机还使用激活码注册到 Satellite 服务器，并从 Satellite 客户端 6 存储库安装必要的配置和管理工具。

CLI 过程

1. 使用 `hammer host create` 命令创建主机：

```
# hammer host create --name "My_Unattended_Host" --organization "My_Organization" \
--location "My_Location" --hostgroup "My_Host_Group" --mac "aa:aa:aa:aa:aa:aa" \
--build true --enabled true --managed true
```

2. 使用 `hammer host interface update` 命令确保设置了网络接口选项：

```
# hammer host interface update --host "test1" --managed true \
--primary true --provision true
```

6.4. 使用 PXE-LESS PROVISIONING 创建主机

有些硬件不提供 PXE 引导接口。在 Satellite 中，您可以置备没有 PXE 引导的主机。这也被称为 PXE-less 置备，涉及生成主机可以使用的引导 ISO。使用这个 ISO，主机可以连接到 Satellite 服务器，引导

安装介质并安装操作系统。

Satellite 还提供无 PXE 的发现服务，无需基于 PXE 的服务，如 DHCP 和 TFTP。更多信息请参阅第 7.7 节“[实施 PXE-less Discovery](#)”。

引导 ISO 类型

有以下类型的引导 ISO：

完整主机镜像

包含特定主机的内核和初始 RAM 磁盘镜像的引导 ISO。如果主机无法正确链加载，则此镜像很有用。置备模板仍从 Satellite 服务器下载。

子网镜像

与特定主机没有关联的引导 ISO。ISO 将主机的 MAC 地址发送到 Capsule 服务器，该服务器与主机条目匹配。该镜像不存储 IP 地址详情，需要访问网络上的 DHCP 服务器进行 bootstrap。此镜像对于同一子网上带有置备 NIC 的所有主机都通用。镜像基于 iPXE 引导固件，只支持有限的网卡。



注意

完整主机镜像基于 SYSLINUX 和 Grub，并可处理大多数网卡。使用子网镜像时，请参阅 ipxe.org 上支持的硬件，以了解预期与基于 iPXE 的引导磁盘一起使用的网卡驱动程序列表。

完整主机镜像包含置备令牌，因此生成的镜像具有有限的生命周期。有关配置安全令牌的详情，请参考第 6.2 节“[配置安全令牌服务持续时间](#)”。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 Hosts > Create Host。
2. 在 Name 字段中输入您要成为置备的系统主机名的名称。

3. 点 **Organization and Location** 选项卡，更改上下文以符合您的要求。
4. 从 **Host Group** 列表中，选择要用于填充表单的主机组。
5. 单击 **Interface** 选项卡，然后在主机的接口上点 **Edit**。
6. 验证这些字段是否填充了值。特别是：
 - **Host** 选项卡中的 **Name** 成为 DNS 名称。
 - **Satellite** 服务器自动为新主机分配 IP 地址。
7. 在 **MAC** 地址字段中，输入主机的 **MAC** 地址。
8. 确保 **Satellite** 服务器自动为主机上的第一个接口选择 **Managed**、**Primary** 和 **Provision** 选项。如果没有，请选择它们。
9. 点 **Operating System** 选项卡，并验证所有字段都包含值。确认操作系统的每个方面。
10. 在 **Provisioning Templates** 中点 **Resolve**，以检查新主机可以识别要使用的正确置备模板。

有关关联置备模板的更多信息，请参阅 [第 2.11 节“置备模板”](#)。
11. 单击 **Parameters** 选项卡，并确保存在一个提供激活码的参数。如果没有，请添加激活码。
12. 点 **Submit** 保存主机详情。这会创建一个主机条目，并显示主机详情页面。
13. 从 **Satellite** 服务器下载引导磁盘。
 - 对于 **Full host image**，在主机详细信息页面上，单击垂直弹出，然后选择 **Full host**

'*My_Host_Name*' image。

- 对于子网镜像，导航到 **Infrastructure > Subnets**，点所需子网的 **Actions** 列中的下拉菜单，然后选择 **Subnet generic image**。

14.

如果需要，使用 **dd** 实用程序或 **livecd-tools** 将 ISO 写入 USB 存储设备。

15.

当您启动主机并从 ISO 或 USB 存储设备引导时，主机连接到 **Satellite** 服务器并从其 **kickstart** 树开始安装操作系统。

安装完成后，主机还使用激活码注册到 **Satellite** 服务器，并从 **Satellite** 客户端 6 存储库安装必要的配置和管理工具。

CLI 过程

1.

使用 **hammer host create** 命令创建主机。

```
# hammer host create --name "My_Host_Name" --organization "My_Organization" \
--location "My_Location" --hostgroup "My_Host_Group" --mac "aa:aa:aa:aa:aa:aa" \
--build true --enabled true --managed true
```

2.

确保使用 **hammer host interface update** 命令设置了网络接口选项。

```
# hammer host interface update --host "My_Host_Name" --managed true \
--primary true --provision true
```

3.

使用 **hammer bootdisk** 命令从 **Satellite** 服务器下载引导磁盘：

- 对于完全主机镜像：

```
# hammer bootdisk host --host My_Host_Name.example.com --full true
```

- 对于子网镜像：

```
# hammer bootdisk subnet --subnet My_Subnet_Name
```

这会为您的主机创建一个引导 ISO。

4. 如果需要，使用 `dd` 实用程序或 `livecd-tools` 将 ISO 写入 USB 存储设备。
5. 当您启动物理主机并从 ISO 或 USB 存储设备引导时，主机连接到 Satellite 服务器并从其 kickstart 树开始安装操作系统。

安装完成后，主机还使用激活码注册到 Satellite 服务器，并从 Satellite 客户端 6 存储库安装必要的配置和管理工具。

6.5. 使用 UEFI HTTP 引导置备创建主机

您可以使用 UEFI HTTP 引导从 Satellite 置备主机。这是您可以在 IPv6 网络中置备主机的唯一方法。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

先决条件

- 确保您满足 HTTP 引导的要求。如需更多信息，请参阅规划 *Satellite* 中的 [HTTP 引导要求](#)。

流程

1. 在用于置备的 Capsule 上，将 `grub2-efi` 软件包更新至最新版本：

```
# satellite-maintain packages update grub2-efi
```

2. 启用 `foreman-proxy-http`、`foreman-proxy-httpboot`，和 `foreman-proxy-tftp` 功能。

```
# satellite-installer --scenario satellite \
--foreman-proxy-http true \
--foreman-proxy-httpboot true \
--foreman-proxy-tftp true
```

3. 在 Satellite Web UI 中，导航到 **Hosts > Create Host**。

4. 在 **Name** 字段中输入主机的名称。
5. 点 **Organization and Location** 选项卡，更改上下文以符合您的要求。
6. 从 **Host Group** 列表中，选择要用于填充表单的主机组。
7. 单击 **Interface** 选项卡，然后在主机的接口上点 **Edit**。
8. 验证这些字段是否填充了值。特别是：
 - **Host** 选项卡中的 **Name** 成为 DNS 名称。
 - **Satellite** 服务器自动为新主机分配 IP 地址。
9. 在 **MAC** 地址字段中，输入主机调配接口的 **MAC** 地址。这样可确保在 **PXE** 引导过程中识别主机。
10. 确保 **Satellite** 服务器自动为主机上的第一个接口选择 **Managed**、**Primary** 和 **Provision** 选项。如果没有，请选择它们。
11. 单击 **OK** 保存。要添加另一个接口，请点击 **Add Interface**。您只能为 **Provision** 和 **Primary** 选择一个接口。
12. 点 **Operating System** 选项卡，并验证所有字段都包含值。确认操作系统的每个方面。
13. 从 **PXE Loader** 列表中，选择 **Grub2 UEFI HTTP**。
14. 可选：点 **Resolve in Provisioning** 模板检查新主机可以识别要使用的正确置备模板。

有关关联置备模板的更多信息，请参阅 [第 2.13 节“创建置备模板”](#)。

15. 单击 **Parameters** 选项卡，并确保存在一个提供激活码的参数。如果没有，请添加激活码。
16. 点 **Submit** 保存主机详情。

有关网络接口的更多信息，请参阅 [添加网络接口](#)。
17. 将主机设置为从网络在 **UEFI** 模式中引导。
18. 启动主机。
19. 在引导菜单中选择 **Kickstart 默认 PXEGrub2**。

这将创建主机条目和相关置备设置。这还包括为 **UEFI** 引导裸机主机创建必要的目录和文件。当您启动物理主机并将其引导模式设置为 **UEFI HTTP** 时，主机检测到定义的 **DHCP** 服务，使用 **Kickstart** 树接收 **Capsule** 的 **HTTP** 端点并安装操作系统。

安装完成后，主机还使用激活码注册到 **Satellite** 服务器，并从 **Satellite** 客户端 **6** 存储库安装必要的配置和管理工具。

CLI 过程

1. 在用于置备的 **Capsule** 上，将 **grub2-efi** 软件包更新至最新版本：

```
# satellite-maintain packages update grub2-efi
```
2. 启用 **foreman-proxy-http**、**foreman-proxy-httpboot**，和 **foreman-proxy-tftp true** 功能：

```
# satellite-installer --scenario satellite \  
--foreman-proxy-http true \  
--foreman-proxy-httpboot true \  
--foreman-proxy-tftp true
```
3. 使用 **hammer host create** 命令创建主机。

```
# hammer host create --name "My_Host" \  
--build true \  
--enabled true \  
--hostgroup "My_Host_Group" \  
--location "My_Location" \  
--mac "aa:aa:aa:aa:aa:aa" \  
--managed true \  
--organization "My_Organization" \  
--pxe-loader "Grub2 UEFI HTTP"
```

4.

使用 `hammer host interface update` 命令确保设置了网络接口选项：

```
# hammer host interface update \  
--host "My_Host" \  
--managed true \  
--primary true \  
--provision true
```

5.

将主机设置为从网络在 UEFI 模式中引导。

6.

启动主机。

7.

在引导菜单中选择 **Kickstart 默认 PXEGrub2**。

这将创建主机条目和相关置备设置。这还包括为 UEFI 引导裸机主机创建必要的目录和文件。当您启动物理主机并将其引导模式设置为 UEFI HTTP 时，主机检测到定义的 DHCP 服务，使用 Kickstart 树接收 Capsule 的 HTTP 端点并安装操作系统。

安装完成后，主机还使用激活码注册到 Satellite 服务器，并从 Satellite 客户端 6 存储库安装必要的配置和管理工具。

6.6. 在置备过程中部署 SSH 密钥

使用此流程在置备过程中部署添加到用户的 SSH 密钥。有关向用户添加 SSH 密钥的详情，请参考 [管理 Red Hat Satellite 指南中的管理用户的 SSH 密钥](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Hosts > Provisioning Templates**。
2. 创建置备模板，或克隆并编辑现有模板。更多信息请参阅 [第 2.13 节“创建置备模板”](#)。
3. 在模板中，点 **Template** 选项卡。
4. 在 **Template editor** 字段中，将 `create_users` 代码片段添加到 `%post` 部分：

```
<%= snippet('create_users') %>
```

5. 选中 **Default** 复选框。
6. 单击 **关联** 选项卡。
7. 在 **Application Operating Systems** 列表中选择操作系统。
8. 点 **Submit** 保存置备模板。
9. 创建与置备模板关联的主机，或使用与修改的模板关联的操作系统来重建主机。如需更多信息，请参阅 [管理主机](#) 指南中的 [创建主机](#)。

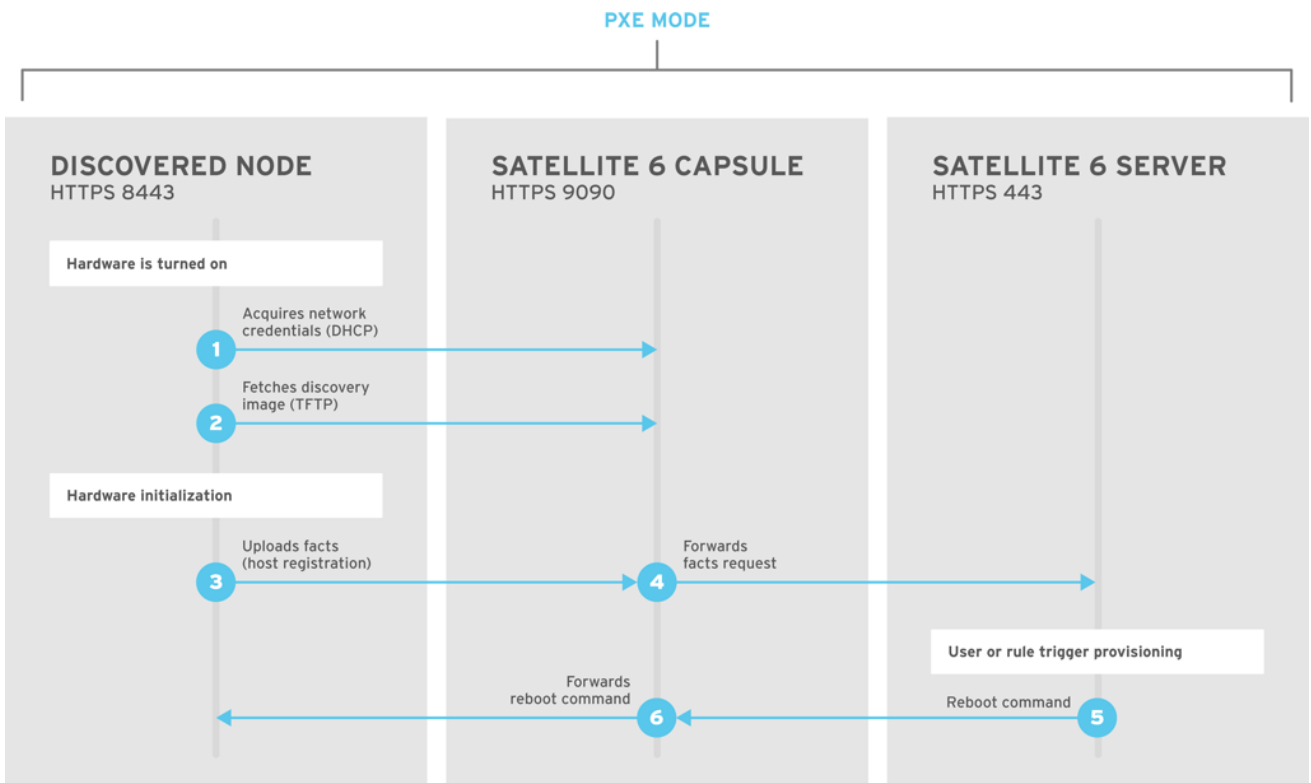
当在置备过程中执行 `create_users` 代码片段时，会自动添加用户拥有的 SSH 密钥。您可以将自己的用户设置为单个用户或用户组。如果将自己的用户设置为用户组，则会自动添加用户组中所有用户的 SSH 密钥。

第 7 章 配置发现服务

Red Hat Satellite 可以检测不在 Satellite 清单中的主机。这些主机引导执行硬件检测的发现镜像，并将此信息转发回 Satellite 服务器。此方法在 Satellite 服务器中创建可随时置备主机的列表，而无需输入每个主机的 MAC 地址。

当您引导空白裸机主机时，引导菜单有两个选项：`local` 和 `discovery`。如果您选择了 `发现` 来引导发现镜像，在几分钟后发现镜像将完成引导，并显示状态屏幕。

在 Satellite 服务器上默认启用 `Discovery` 服务。但是，全局模板的默认设置是从本地硬盘引导。要更改默认设置，在 Satellite Web UI 中导航至 `Administer > Settings`，然后点 `Provisioning` 选项卡。找到 `Default PXE` 全局模板条目行，并在 `Value` 列中输入 `discovery`。



SATELLITE6_376339_1115

确保您要用于发现的所有子网的 DHCP 范围不会与为受管 DHCP 服务配置的 DHCP 租期池干扰。DHCP 范围在 Satellite Web UI 中设置，而租期池范围则使用 `satellite-installer` 命令设置。例如，在 10.1.0.0/16 网络范围内，可以为租期分配 10.1.0.0 到 10.1.127.255，并将 10.1.128.0 分配给 10.1.255.254 可分配给保留。

由于网络接口名称在 Red Hat Enterprise Linux 的主版本之间并不总是相同，因此可以使用不正确的网络配置创建主机或虚拟机。在 Dell 服务器上，可以通过 `biosdevname=1` 内核命令行选项为其他硬件或虚拟机禁用新的命名方案，可以通过 `net.ifnames=1` 完整关闭新的命名方案。

1. 在 **Satellite** 服务器上安装 **Discovery** :

foreman-discovery-image 软件包将发现 ISO 安装到 `/usr/share/foreman-discovery-image/` 目录中。

```
# satellite-installer \  
--enable-foreman-plugin-discovery \  
--enable-foreman-proxy-plugin-discovery
```

2. 安装 **foreman-discovery-image** :

```
satellite-maintain packages install foreman-discovery-image
```

安装完成后，您可以通过导航到 **Hosts > Discovered Hosts** 来查看新的菜单选项。

7.1. 安装发现服务

完成以下步骤，在 **Capsule** 服务器上启用发现服务。

1. 在 **Capsule** 服务器上输入以下命令 :

```
# satellite-installer \  
--enable-foreman-proxy-plugin-discovery
```

2. 安装 **foreman-discovery-image** :

```
# satellite-maintain packages install foreman-discovery-image
```

子网

所有可发现主机的子网都需要选择适当的 **Capsule** 服务器来提供 **Discovery** 服务。

在 **Satellite Web UI** 中，导航到 **Infrastructure > Subnets**，选择一个子网，点 **Capsules** 选项卡，然后选择您要使用的 **Discovery Proxy**。对您要使用的每个子网执行此操作。

7.2. 置备模板 PXELINUX 发现片断

对于 BIOS 置备，Host > Provisioning Templates 窗口中的 PXELinux 全局默认模板 包含代码片段 `pxelinux_discovery`。该片段有以下行：

```

LABEL discovery
  MENU LABEL Foreman Discovery Image
  KERNEL boot/fdi-image/vmlinuz0
  APPEND initrd=boot/fdi-image/initrd0.img rootflags=loop root=live:/fdi.iso rootfstype=auto ro
rd.live.image acpi=force rd.luks=0 rd.md=0 rd.dm=0 rd.lvm=0 rd.bootif=0 rd.neednet=0 nomodeset
proxy.url=<%= foreman_server_url %> proxy.type=foreman
  IPAPPEND 2

```

KERNEL 和 **APPEND** 选项引导发现镜像和 **ramdisk**。**APPEND** 选项包含一个 `proxy.url` 参数，其中 `foreman_server_url` 宏作为其参数。此宏解析为 **Satellite** 服务器的完整 URL。

对于 UEFI 置备，Host > Provisioning Templates 窗口中的 PXEgrub2 全局默认模板 包含片段 `pxegrub2_discovery`：

```

menuentry 'Foreman Discovery Image' --id discovery {
  linuxefi boot/fdi-image/vmlinuz0 rootflags=loop root=live:/fdi.iso rootfstype=auto ro rd.live.image
acpi=force rd.luks=0 rd.md=0 rd.dm=0 rd.lvm=0 rd.bootif=0 rd.neednet=0 nomodeset proxy.url=<%=
foreman_server_url %> proxy.type=foreman BOOTIF=01-$mac
  initrdefi boot/fdi-image/initrd0.img
}

```

要使用 **Capsule** 代理发现步骤，请编辑 `/var/lib/tftpboot/pxelinux.cfg/default` 或 `/var/lib/tftpboot/grub2/grub.cfg`，并将 URL 更改为您要使用的 **Capsule Server FQDN**。

全局模板可用于 **Satellite** 服务器上，以及启用了 TFTP 功能的所有 **Capsule**。

7.3. 发现主机的自动上下文

Satellite 服务器根据以下规则序列为发现的主机分配一个机构和位置：

1. 如果发现的主机使用 **Satellite** 中定义的子网，则主机将使用与子网关联的第一个机构和位置。
2. 如果设置了 `discovery_organization` 或 `discovery_location` 事实值，发现的主机将使用这些事实值作为机构和位置。要设置这些事实值，请导航到 **Administer > Settings >**

Discovered，并将这些事实添加到 **Default 机构** 和 **Default location** 字段中。确保发现的主机的子网也属于事实设置的机构和位置，否则 **Satellite** 会拒绝出于安全原因进行设置。

3.

如果之前没有条件，**Satellite** 会分配第一个机构和位置（按名称排序）。

您可以使用 **Discovered hosts** 页面的批量操作菜单更改机构或位置。选择要修改的发现主机，然后从 **Select Action** 菜单中选择 **Assign Organization** 或 **Assign Location**。

请注意，**foreman_organization** 和 **foreman_location** 事实不再是为发现的主机分配上下文的有效值。您仍然可使用这些事实来配置 **Puppet** 运行的主机。为此，请导航到 **Administer > Settings > Puppet** 部分，并将 **foreman_organization** 和 **foreman_location** 事实添加到 **Default organization** 和 **Default location** 字段。

7.4. 发现模板和片段设置

要使用 **Discovery** 服务，您必须配置置备设置，将 **Discovery** 设置为默认服务，并设置您要使用的模板。

将 **Discovery Service** 设置为默认值

对于 **BIOS** 和 **UEFI**，要将发现服务设置为当前 **Satellite** 清单中不存在的主机引导的默认服务，请完成以下步骤：

1. 在 **Satellite Web UI** 中，导航到 **Administer > Settings** 并点 **Provisioning** 选项卡。
2. 对于 **Default PXE** 全局模板条目，在 **Value** 列中输入 **discovery**。

要使用模板，在 **Satellite Web UI** 中，导航到 **Administer > Settings**，然后点击 **Provisioning** 选项卡并设置您要使用的模板。

自定义模板和片断

模板和片断被锁定以防止更改。如果要编辑模板或片断，请克隆它，使用唯一名称保存它，然后编辑克隆。

当您更改模板或包含的代码片段时，必须将更改传播到 **Satellite** 服务器的默认 **PXE** 模板。

在 Satellite Web UI 中，导航到 Hosts > Provisioning Templates，再点 Build PXE Default。

这会刷新 Satellite 服务器上的默认 PXE 模板。

其他设置

proxy.url 参数

在 Satellite 安装过程中，如果您使用默认选项 `--enable-foreman-plugin-discovery`，您可以编辑模板中的 `proxy.url` 参数，以设置提供发现服务的 Capsule 服务器的 URL。您可以将 `proxy.url` 参数更改为您要使用的另一个置备胶囊的 IP 地址或 FQDN，但请确保附加端口号，如 9090。如果您在 Satellite 安装过程中使用带有 `--foreman-proxy-ssl-port` 选项的替代端口号，您必须添加该端口号。您还可以编辑 `proxy.url` 参数，以使用 Satellite IP 地址或 FQDN，以便发现的主机直接与 Satellite 服务器通信。

proxy.type 参数

如果您将 Capsule Server FQDN 用于 `proxy.url` 参数，请确保将 `proxy.type` 参数设置为 `proxy`。如果使用 Satellite FQDN，请将 `proxy.type` 参数更新为 `foreman`。

```
proxy.url=https://capsule.example.com:9090 proxy.type=proxy
```

渲染 Capsule 的主机名

Satellite 将相同的模板部署到所有 TFTP 胶囊中，且没有可用于呈现胶囊主机名的变量或宏。硬编码的 `proxy.url` 无法用于两个或多个 TFTP Capsule。作为临时解决方案，每次点 Build PXE Defaults 时，使用 SSH 编辑 TFTP 目录中的配置文件，或者将通用 DNS 别名用于适当的子网。

标记的 VLAN 置备

如果要使用标记的 VLAN 置备，并且您希望发现服务发送发现请求，请在发现模板中的 `KERNEL` 选项中添加以下信息：

```
fdi.vlan.primary=example_VLAN_ID
```

7.5. 从发现的主机创建主机

置备发现的主机遵循与 PXE 置备类似的置备过程。主要区别在于，而不是手动输入主机的 MAC 地址，您可以从发现的主机列表中选择要置备的主机。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

先决条件

- 在 Satellite 上配置域和子网。有关网络要求的详情请参考 [第 3 章 配置网络](#)。
- 在 Satellite 上配置发现服务。更多信息请参阅 [第 7.1 节 “安装发现服务”](#)。
- 裸机主机或空白虚拟机。
- 您可以将同步的内容软件仓库用于 Red Hat Enterprise Linux。有关更多信息，请参阅 [内容管理指南中的同步存储库](#)。
- 为主机注册提供激活码。如需更多信息，请参阅 [内容管理指南中的创建激活码](#)。

有关安全令牌的详情，请参考 [第 6.2 节 “配置安全令牌服务持续时间”](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Hosts > Discovered host**。选择要使用的主机，然后单击列表右侧的 **Provision**。
2. 从以下选项之一中选择：
 - 要从主机组置备主机，请选择主机组、机构和位置，然后单击 **Create Host**。
 - 要置备具有进一步自定义的主机，请点 **Customize Host**，并输入您要为新主机指定的附加详情。
3. 验证这些字段是否填充了值。特别是：

- **Host** 选项卡中的 **Name** 成为 DNS 名称。
 - **Satellite** 服务器自动为新主机分配 IP 地址。
 - **Satellite** 服务器从发现结果自动填充 MAC 地址。
4. 确保 **Satellite** 服务器自动为主机上的第一个接口选择 **Managed**、**Primary** 和 **Provision** 选项。如果没有，请选择它们。
 5. 点 **Operating System** 选项卡，并验证所有字段都包含值。确认操作系统的每个方面。
 6. 点 **Provisioning** 模板中的 **Resolve**，以检查新主机可以识别要使用的正确置备模板。主机必须解析为以下置备模板：
 - **kexec Template: Discovery Red Hat kexec**
 - 置备模板：**Satellite Kickstart** 默认有关关联置备模板的更多信息，请参阅 [第 2.11 节“置备模板”](#)。
 7. 点 **Submit** 保存主机详情。

主机置备完成后，发现的主机将变为内容主机。要查看主机，请导航到 **Hosts > Content Hosts**。

CLI 过程

1. 识别用于置备的发现主机：

```
# hammer discovery list
```
2. 选择主机并使用主机组进行置备。使用 `--new-name` 选项设置新主机名：
-

```
# hammer discovery provision \
--build true \
--enabled true \
--hostgroup "My_Host_Group" \
--location "My_Location" \
--managed true \
--name "My_Host_Name" \
--new-name "My_New_Host_Name" \
--organization "My_Organization"
```

这会从发现的主机列表中删除主机，并使用置备设置创建一个主机条目。发现镜像会自动重置主机，以便它可以引导到 PXE。主机检测 Satellite 服务器集成胶囊上的 DHCP 服务，并开始安装操作系统。其余进程与第 6.3 节“使用未置备创建主机”中描述的普通 PXE 工作流相同。

7.6. 创建发现规则

作为为发现的主机自动化置备过程的方法，Satellite 提供了创建发现规则的功能。这些规则根据分配的主机组定义发现的主机如何自动置备其自身。例如，您可以自动置备 CPU 数量高的主机，作为虚拟机监控程序。同样，您可以将大型硬盘的主机置备为存储服务器。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

NIC 注意事项

自动配置目前不允许配置 NIC，所有系统都会使用发现过程中检测到的 NIC 配置来置备。但是，您可以在 kickstart scriptlet 中设置 NIC，使用脚本，或在以后的阶段使用配置管理。

流程

1. 在 Satellite Web UI 中，导航到 **Configure > Discovery rules**，然后选择 **Create Rule**。
2. 在 **Name** 字段中输入规则的名称。
3. 在 **Search** 字段中，输入规则来确定是否置备主机。此字段为您提供您输入的值的建议，并允许多个规则的 **operator**。例如：`cpu_count > 8`。
4. 从 **Host Group** 列表中，选择要用作此主机的模板的主机组。
5. 在 **Hostname** 字段中，输入模式来确定多个主机的主机名。这使用与置备模板使用的相同

ERB 语法。 主机名可以将 `@host` 属性用于特定于主机的值，将 `rand` 宏用于随机数字，或者将 `sequence_hostgroup_param_next` 宏用于递增值。有关置备模板的更多信息，请参阅第 2.11 节“置备模板”和 API 文档。

- `myhost-<%= sequence_hostgroup_param_next("EL7/MyHostgroup", 10, "discovery_host") %>`
- `myhost-<%= rand(99999) %>`
- `abc-<%= @host.facts["bios_vendor"] %>-<%= rand(99999) %>`
- `xyz-<%= @host.hostgroup.name %>`
- `srv-<%= @host.discovery_rule.name %>`
- `server-<%= @host.ip.gsub('.', '-') + '-' + @host.hostgroup.subnet.name %>`

在创建主机名模式时，请确保生成的主机名是唯一的，不要以数字开头，不包含下划线或点。好方法是使用事实器提供的唯一信息，如 MAC 地址、BIOS 或串行 ID。

6. 在 **Hosts limit** 字段中，输入您可以使用该规则置备的最大主机数。为无限输入 0。
7. 在 **Priority** 字段中，输入一个数字来设置规则超过其他规则的优先级。较低值的规则具有更高的优先级。
8. 从 **Enabled** 列表中，选择是否要启用该规则。
9. 要为规则设置不同的置备上下文，请点 **机构** 和 **位置** 选项卡，然后选择您要使用的上下文。
10. 点 **Submit** 保存您的规则。

11.

在 Satellite Web UI 中，导航到 **Hosts > Discovered Host** 并选择以下两个选项之一：

- 从右侧的发现主机列表中，选择 **Auto-Provision** 来自动置备单个主机。
- 在窗口右上角，点 **Auto-Provision All** 来自动置备所有主机。

CLI 过程

1.

使用 **hammer discovery-rule create** 命令创建规则：

```
# hammer discovery-rule create \
--enabled true \
--hostgroup "My_Host_Group" \
--hostname "hypervisor-<%= rand(99999) %>" \
--hosts-limit 5 \
--name "My_Hypervisor" \
--priority 5 \
--search "cpu_count > 8"
```

2.

使用 **hammer discovery auto-provision** 命令自动置备主机：

```
# hammer discovery auto-provision --name "macabcdef123456"
```

7.7. 实施 PXE-LESS DISCOVERY

Satellite 提供了一个 **PXE-less Discovery** 服务，可在不需要基于 **PXE** 的服务(DHCP 和 TFTP)的情况下运行。您使用 **Satellite** 服务器的发现镜像完成此操作。调度发现的节点进行安装后，它会使用 **kexec** 命令重新加载带有 **OS** 安装程序的 **Linux** 内核，而无需重启该节点。

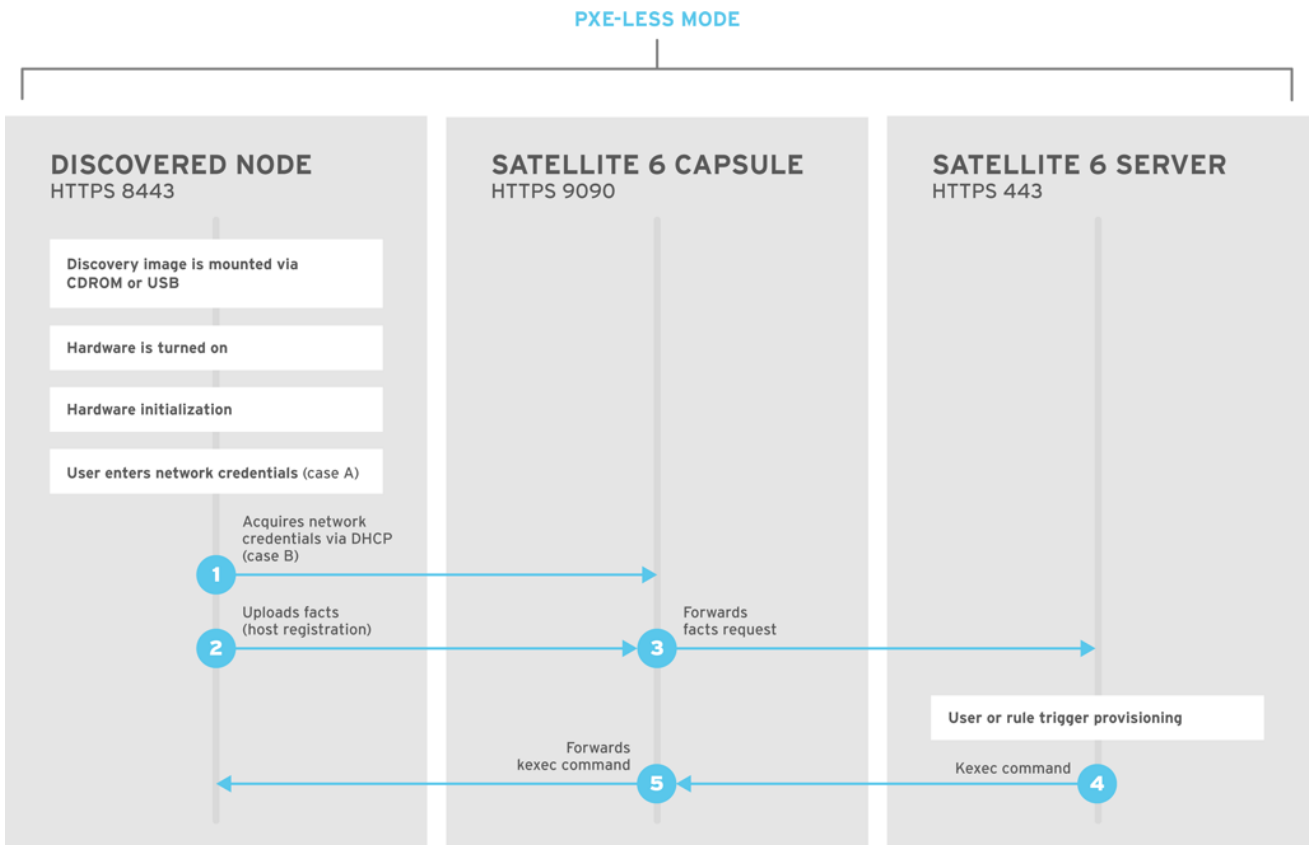
重要

Discovery kexec 只是一个技术预览功能。技术预览功能不受红帽产品服务等级协议 (SLA) 支持，且功能可能并不完整。红帽不推荐在生产环境中使用它们。这些技术预览功能可以使用户提早试用新的功能，并有机会在开发阶段提供反馈意见。有关红帽技术预览功能支持范围的详情，请参考

<https://access.redhat.com/support/offerings/techpreview/>。

已知问题

控制台可能会在此过程中冻结。在某些硬件上，您可能会遇到图形硬件问题。



SATELLITE6_376339_1115

Discovery 服务的 ISO 位于 `/usr/share/foreman-discovery-image/` 中，并使用 `foreman-discovery-image` 软件包安装。

参与使用

1.

将这个介质复制到 CD、DVD 或者 USB 盘中。例如，要复制到位于 `/dev/sdb` 的 USB 盘：

```
# dd bs=4M \
if=/usr/share/foreman-discovery-image/foreman-discovery-image-3.4.4-5.iso \
of=/dev/sdb
```

2.

将 Discovery 引导介质插入到裸机主机中，启动主机并从介质引导。Discovery Image 显示 **Manual 网络设置** 或使用 DHCP 的 Discovery 的选项：

如果选择 **Manual network setup**，则 Discovery 镜像会请求一组网络选项。这包括连接到 Satellite 服务器的主网络接口。此发现镜像还会要求网络接口配置选项，如 IPv4 地址、IPv4 网关和 IPv4 DNS 服务器。

3. 输入这些详细信息后，选择 **Next**。
4. 如果您选择了使用 DHCP 的 **Discovery**，则发现镜像仅请求连接到 **Satellite** 服务器的主网络接口。它尝试使用 DHCP 服务器自动配置网络接口，如 **Capsule** 服务器提供的接口。
5. 主接口配置后，发现镜像会请求服务器 URL，即提供 **Discovery** 服务的 **Satellite** 服务器或 **Capsule** 服务器的 URL。例如，要在 **Satellite** 服务器上使用集成的 **Capsule**，请使用以下 URL：

```
https://satellite.example.com:9090
```
6. 将 **Connection** 类型设置为 **Proxy**，然后选择 **Next**。
7. 可选：**Discovery** 镜像还提供一组字段来为 **Facter** 工具输入自定义事实，以回中继回 **Satellite** 服务器。它们以名称-值格式输入。提供您需要的任何自定义事实，然后选择 **Confirm** 继续。

Satellite 报告与 **Satellite** 服务器的发现服务成功通信。在 **Satellite Web UI** 中，导航到 **Hosts > Discovered Hosts** 并查看新发现的主机。

有关置备发现的主机的更多信息，请参阅 [第 7.5 节“从发现的主机创建主机”](#)。

7.8. 无人值守使用、自定义和镜像恢复

您可以创建自定义 **Discovery ISO**，以便在引导后自动执行镜像配置过程。**Discovery** 镜像将 Linux 内核用于操作系统，它会传递内核参数来配置发现服务。这些内核参数包括以下条目：

proxy.url

提供发现服务的 **Capsule** 服务器或 **Satellite** 服务器的 URL。

proxy.type

代理类型。这通常设置为 **代理** 来连接到 **Capsule** 服务器。此参数还支持传统的 **foreman** 选项，其中通信直接与 **Satellite** 服务器（而非 **Capsule** 服务器）通信。

fdi.pxmac

主接口的 MAC 地址，格式为 AA:BB:CC:DD:EE:FF。这是您用来与 Capsule 服务器通信的接口。在自动模式中，使用第一个 NIC（按字母顺序使用网络标识符），并使用链接。在半自动模式下，会显示屏幕并要求您选择正确的接口。

fdi.pxip, fdi.pxgw, fdi.pxdns

为主网络接口手动配置 IP 地址(fdi.pxip)、网关(fdi.pxgw)和 DNS (fdi.pxdns)。如果省略这些参数，镜像将使用 DHCP 来配置网络接口。

fdi.pxfactname1, fdi.pxfactname2 ... fdi.pxfactnameN

用于指定自定义事实名称。

fdi.pxfactvalue1, fdi.pxfactvalue2 ... fdi.pxfactvalueN

每个自定义事实的值。每个值对应于事实名称。例如，fdi.pxfactvalue1 为使用 fdi.pxfactname1 指定的事实设置值。

fdi.pxauto

设置自动或半自动模式。如果设置为 0，则镜像使用半自动模式，允许您通过一组对话框选项确认您的选择。如果设置为 1，则镜像将使用自动模式，并在不确认的情况下继续。

fdi.initnet

默认情况下，镜像初始化所有网络接口（值 all）。当此设置设置为 bootif 时，只有从中启动的网络接口才会被初始化。

fdi.rootpw

默认情况下，root 帐户被锁定。使用这个选项设定 root 密码。您可以同时输入明文和加密的密码。

fdi.ssh

默认情况下禁用 SSH 服务。把它设置为 1 或 true 以启用 SSH 访问。

fdi.ipv4.method

默认情况下，NetworkManager IPv4 方法设置被设置为 auto。此选项覆盖它，将其设置为 ignore 以禁用 IPv4 堆栈。这个选项只在 DHCP 模式下工作。

fdi.ipv6.method

默认情况下，NetworkManager IPv6 方法设置被设置为 auto。此选项覆盖它，将其设置为 ignore 以禁用 IPv6 堆栈。这个选项只在 DHCP 模式下工作。

fdi.zips

在引导过程中下载和启动扩展的文件名。如需更多信息，请参阅 [第 7.9 节“扩展发现镜像”](#)。

fdi.zipserver

用于从中下载扩展的 TFTP 服务器。如需更多信息，请参阅 [第 7.9 节“扩展发现镜像”](#)。

fdi.countdown

在初始发现尝试后刷新 `text-user` 界面前等待的秒数。这个值默认为 45 秒。如果状态页面将 IP 地址报告为 N/A，则增加这个值。

fdi.dhcp_timeout

NetworkManager DHCP 超时。默认值为 300 秒。

fdi.vlan.primary

为主接口设置的 VLAN 标记 ID。

使用 `discovery-remaster` 工具重新载入操作系统镜像

Satellite 服务器在 `foreman-discovery-image` 软件包中提供 `discovery-remaster` 工具。此工具将镜像重新命名，使其包含这些内核参数。要重新主控机镜像，请运行 `discovery-remaster` 工具。例如：

```
# discovery-remaster ~/iso/foreman-discovery-image-3.4.4-5.iso \
"fdi.pxip=192.168.140.20/24 fdi.pxgw=192.168.140.1 \
fdi.pxdns=192.168.140.2 proxy.url=https://satellite.example.com:9090 \
proxy.type=proxy fdi.pxfactname1=customhostname fdi.pxfactvalue1=myhost
fdi.pxmac=52:54:00:be:8e:8c fdi.pxauto=1"
```

将这个介质复制到 CD、DVD 或者 USB 盘中。例如，要复制到位于 `/dev/sdb` 的 USB 盘：

```
# dd bs=4M \
if=/usr/share/foreman-discovery-image/foreman-discovery-image-3.4.4-5.iso \
of=/dev/sdb
```

将 **Discovery** 引导介质插入到裸机主机中，启动主机并从介质引导。

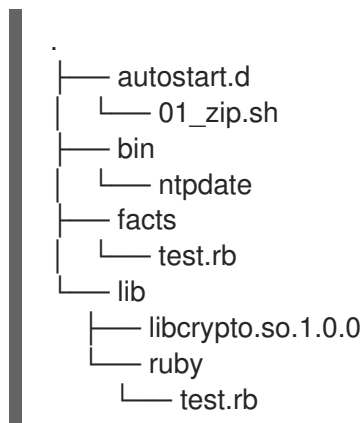
有关置备发现的主机的更多信息，请参阅 [第 7.5 节“从发现的主机创建主机”](#)。

7.9. 扩展发现镜像

您可以使用自定义事实、软件或设备驱动程序来扩展 **Satellite** 发现镜像。您还可以提供一个压缩的存档文件，其中包含要使用的镜像的额外代码。

流程

1. 创建以下目录结构：



- **autostart.d** 目录包含在主机注册到 **Satellite** 之前按 **POSIX** 顺序执行的脚本。
- **bin** 目录添加到 **\$PATH** 变量中；您可以将二进制文件放在此目录中，并将它们放在自动启动脚本中。
- **facts** 目录添加到 **FACTERLIB** 变量中，以便配置自定义事实并发送到 **Satellite**。
- **lib** 目录添加到 **LD_LIBRARY_PATH** 变量中，**lib/ruby** 被添加到 **RUBYLIB** 变量中，以便 **/bin** 中的二进制文件可以正确执行。

2. 创建目录结构后，使用以下命令创建一个 **.zip** 文件归档：

```
# zip -r my_extension.zip .
```

3. 要告知其必须使用的扩展的发现镜像，请将 **zip** 文件放在带有 **Discovery** 镜像的 **TFTP** 服务器上，然后使用 **fdi.zips** 选项更新 **PXELinux** 模板的 **APPEND** 行，其中路径相对于 **TFTP root**。例如，如果您在 **\$TFTP/zip1.zip** 和 **\$TFTP/boot/zip2.zip** 中有两个存档，请使用以下语法：

```
fdi.zips=zip1.zip,boot/zip2.zip
```

您可以将新指令和选项添加到现有环境变量(PATH、LD_LIBRARY_PATH、RUBYLIB 和 FACTERLIB)。如果要在脚本中明确指定路径，请将 .zip 文件内容提取到镜像的 /opt/extension 目录中。

您可以创建多个 .zip 文件，但请注意它们被提取到 Discovery 镜像上的同一位置。如果文件有相同的文件名，则从 .zip 文件中提取的文件会覆盖早期版本。

7.10. 发现故障排除

如果机器没有在 Hosts > Discovered Hosts 中的 Satellite Web UI 中列出，请检查以下配置区域以帮助隔离错误：

- 在 Satellite Web UI 中，导航到 Hosts > Provisioning Templates，并使用 Build PXE Default 按钮重新部署默认的 PXELinux 模板。
- 验证 TFTP Capsule 上的 pxelinux.cfg/default 配置文件。
- 确保主机、Capsule 服务器和 Satellite 服务器之间有足够网络连接。
- 检查使用的 PXELinux 模板，并确定其包含的 PXE 发现片断。代码片段命名如下：`pxelinux_discovery`、`pxegrub_discovery` 或 `pxegrub2_discovery`。验证 PXE 发现片断中的 `proxy.url` 和 `proxy.type` 选项。
- 确保 DNS 已为发现的节点正常工作，或使用您使用的 PXELinux 模板中包含的 PXE 发现片断中的 `proxy.url` 选项中的 IP 地址。
- 确保 DHCP 服务器正确向引导的镜像提供 IP 地址。
- 确保发现的主机或虚拟机至少有 1200 MB 内存。因为镜像被提取在内存中，所以内存减少可能会导致各种随机内核 panic 错误。

要收集重要的系统事实，请使用 `discovery-debug` 命令。它将打印系统日志、网络配置、事实列表以及标准输出的其他信息。典型的用例是重定向此输出，并使用 `scp` 命令复制它以进一步调查。

发现主机上的第一个虚拟控制台是为 **systemd** 日志保留的。特别有用的系统日志被标记为如下：

- **discover-host** - 初始事实上传
- **foreman-discovery** - 事实刷新，重启远程命令
- **nm-prepare** - 预配置 **NetworkManager** 的启动脚本
- **NetworkManager** - 网络信息

使用 **TTY2** 或更高版本登录发现的主机。默认禁用 **root** 帐户和 **SSH** 访问，但您可以使用 **APPEND** 行上的 **Default PXELinux** 模板中以下内核命令行选项启用 **SSH** 并设置 **root** 密码：

```
fdi.ssh=1 fdi.rootpw=My_Password
```


第 8 章 使用红帽镜像构建器镜像置备

在 Satellite 中，您可以与 RHEL web 控制台集成，以执行操作并监控主机。使用 RHEL web 控制台，您可以访问 Red Hat Image Builder 和构建镜像，然后您可以上传到 HTTP 服务器，并使用此镜像来置备主机。当您为镜像置备配置 Satellite 时，Anaconda 安装程序分区磁盘，下载并挂载镜像，并将文件复制到主机。首选镜像类型是 TAR。



注意

构建 TAR 镜像的蓝图必须始终包含一个内核软件包。

有关将 RHEL web 控制台与 Satellite 集成的更多信息，请参阅管理主机指南中的 [使用 RHEL web 控制台进行主机管理和监控](#)。

前提条件

- 使用红帽镜像构建器创建的现有 TAR 镜像。

流程

1. 在 Satellite 上，创建自定义产品，向此产品添加自定义文件存储库，并将镜像上传到存储库。如需更多信息，请参阅 [内容管理指南中的导入单个 ISO 镜像和文件](#)。
2. 在 Satellite Web UI 中，导航到 **Configure > Host Groups**，然后选择您要使用的主机组。
3. 单击 **Parameters** 选项卡，然后单击 **Add Parameter**。
4. 在 **Name** 字段中输入 `kickstart_liveimg`。
5. 从 **Type** 列表中，选择 **字符串**。
6. 在 **Value** 字段中，以 `custom/product/repository/image_name` 格式输入绝对路径或相对路径，指向保存镜像的确切位置。

7.

点 **Submit** 保存您的更改。

您可以使用此镜像通过计算资源进行裸机置备和置备。有关裸机置备的详情，请参考 [第 6 章 使用 PXE 置备主机](#)。有关使用不同计算资源置备的更多信息，请参阅您要使用的计算资源的相关章节。

第 9 章 在 KVM 上置备虚拟机(LIBVIRT)

基于内核的虚拟机(KVM)使用开源虚拟化守护进程和名为 libvirt 在 Red Hat Enterprise Linux 上运行的 API。Red Hat Satellite 可以连接到 KVM 服务器上的 libvirt API，调配虚拟机监控程序上的主机，并控制某些虚拟化功能。

只有通过 Satellite 创建的虚拟机才能进行管理。不支持具有目录存储池类型以外的虚拟机。

您可以使用 KVM 置备通过网络连接或现有镜像创建主机。

先决条件

- 您可以将同步的内容软件仓库用于 Red Hat Enterprise Linux。有关更多信息，[请参阅 内容管理指南中的同步存储库](#)。
- 为主机注册提供激活码。如需更多信息，[请参阅 内容管理指南中的创建激活码](#)。
- 在 KVM 服务器上管理网络的 Capsule 服务器。确保没有在此网络上运行其他 DHCP 服务，以避免与 Capsule 服务器冲突。有关 Capsule 服务器的网络服务配置的更多信息，[请参阅置备指南中的配置网络](#)。
- 运行 KVM 虚拟化工具(libvirt 守护进程)的 Red Hat Enterprise Linux 服务器。如需更多信息，[请参阅 Red Hat Enterprise Linux 7 虚拟化入门指南](#)。
- 如果要使用基于镜像的置备，现有的虚拟机镜像。确保此镜像存在于 KVM 主机上的存储池中。默认存储池通常位于 /var/lib/libvirt/images 中。只有目录池存储类型可以通过 Satellite 管理。
- 可选：这些步骤中的示例将 root 用户用于 KVM。如果要在 KVM 服务器上使用非 root 用户，您必须将该用户添加到 KVM 服务器上的 libvirt 组中：

```
# usermod -a -G libvirt non_root_user
```

- 具有以下角色的 Satellite 用户帐户：

- 编辑主机
- 查看主机

如需更多信息，[请参阅](#) *管理 Red Hat Satellite 指南*中的将角色分配给用户。

- 在 Satellite 中具有以下权限的自定义角色：

- view_compute_resources
- destroy_compute_resources_vms
- power_compute_resources_vms
- create_compute_resources_vms
- view_compute_resources_vms
- view_locations
- view_subnets

有关创建角色的更多信息，[请参阅](#)*管理 Red Hat Satellite 指南*中的创建角色。有关向角色添加权限的更多信息，[请参阅](#) *管理 Red Hat Satellite 指南*中的向角色添加权限。

9.1. 为 KVM 连接配置 SATELLITE 服务器

在添加 KVM 连接前，请为 foreman 用户创建一个 SSH 密钥对，以确保 Satellite 服务器和 KVM 之间的安全连接。

流程

1. 在 **Satellite** 服务器上，切换到 **foreman** 用户：

```
# su foreman -s /bin/bash
```

2. 生成密钥对：

```
$ ssh-keygen
```

3. 将公钥复制到 **KVM** 服务器中：

```
$ ssh-copy-id root@kvm.example.com
```

4. 退出 **foreman** 用户的 **bash shell**：

```
$ exit
```

5. 安装 **libvirt-client** 软件包：

```
# satellite-maintain packages install libvirt-client
```

6. 使用以下命令测试与 **KVM** 服务器的连接：

```
# su foreman -s /bin/bash -c 'virsh -c qemu+ssh://root@kvm.example.com/system list'
```

9.2. 向 **SATELLITE** 服务器添加 **KVM** 连接

使用这个流程在 **Satellite** 中将 **KVM** 添加为计算资源。要使用 **CLI** 而不是 **Satellite Web UI**，请参阅 [CLI 过程](#)。

流程

1. 在 **Satellite Web UI** 中，导航到 **Infrastructure > Compute Resources**，再点 **Create Compute Resource**。
2. 在 **Name** 字段中输入新计算资源的名称。

3. 从 **Provider** 列表中，选择 **Libvirt**。
4. 在 **Description** 字段中，输入计算资源的描述。
5. 在 **URL** 字段中输入到 **KVM** 服务器的连接 **URL**。例如：

```
qemu+ssh://root@kvm.example.com/system
```

6. 从 **Display type** 列表中，选择 **VNC** 或 **Tailoring**。

7. 可选：要保护带有随机生成的密码的新主机的控制台访问，请在显示连接框中选择 **Set a random generated password**。您可以检索 **VNC** 控制台的密码，以便从 **KVM** 服务器上执行以下命令的输出中访问客户机虚拟机控制台：

```
# virsh edit your_VM_name
<graphics type='vnc' port='-1' autoport='yes' listen='0.0.0.0'
passwd='your_randomly_generated_password'>
```

每次打开虚拟机的控制台时都会随机生成密码，例如，使用 **virt-manager**。

8. 单击 **Test Connection**，以确保 **Satellite** 服务器在没有故障的情况下连接到 **KVM** 服务器。
9. 验证 **位置** 和 **机构** 选项卡是否自动设置为您当前的上下文。如果需要，在这些标签页中添加额外的上下文。
10. 点 **Submit** 保存 **KVM** 连接。

CLI 过程

- 要创建计算资源，请输入 **hammer compute-resource create** 命令：

```
# hammer compute-resource create --name "My_KVM_Server" \
--provider "Libvirt" --description "KVM server at kvm.example.com" \
--url "qemu+ssh://root@kvm.example.com/system" --locations "New York" \
--organizations "My_Organization"
```

9.3. 将 KVM 镜像添加到 SATELLITE 服务器中

要使用基于镜像的调配创建主机，您必须将有关镜像的信息（如访问详情和镜像位置）添加到 Satellite 服务器中。

请注意，您可以通过 Satellite 只管理目录池存储类型。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Infrastructure > Compute Resources**，然后单击 **KVM 连接** 的名称。
2. 单击 **Create Image**。
3. 在 **Name** 字段中输入镜像的名称。
4. 从 **Operating System** 列表中，选择镜像的基础操作系统。
5. 从 **Architecture** 列表中，选择操作系统架构。
6. 在 **Username** 字段中，为镜像访问输入 **SSH 用户名**。这通常是 **root 用户**。
7. 在 **Password** 字段中，输入 **SSH 密码**以进行镜像访问。
8. 在 **Image path** 字段中，输入指向 KVM 服务器上镜像的完整路径。例如：

```
/var/lib/libvirt/images/TestImage.qcow2
```
9. 可选：如果镜像支持用户数据输入，如 **cloud-init 数据**，请选择 **User Data** 复选框。

10. 点 **Submit** 保存镜像详情。

CLI 过程

- 使用 `hammer compute-resource image create` 命令创建镜像。使用 `--uuid` 字段存储 KVM 服务器上镜像位置的完整路径。

```
# hammer compute-resource image create \  
--name "KVM Image" \  
--compute-resource "My_KVM_Server" \  
--operatingsystem "RedHat version" \  
--architecture "x86_64" \  
--username root \  
--user-data false \  
--uuid "/var/lib/libvirt/images/KVMimage.qcow2" \  

```

9.4. 在计算配置集中添加 KVM 详情

使用这个流程在计算配置集中添加 KVM 硬件设置。当您使用这个计算配置集在 KVM 上创建主机时，这些设置会自动填充。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Infrastructure > Compute Profiles**。
2. 在 **Compute Profiles** 窗口中，点现有计算配置集的名称，或者单击 **Create Compute Profile**，输入 **Name**，然后单击 **Submit**。
3. 点 KVM 计算资源的名称。
4. 在 **CPU** 字段中，输入要分配给新主机的 CPU 数量。
5. 在 **Memory** 字段中，输入分配给新主机的内存量。

6. 从 **Image** 列表中，选择在执行基于镜像的置备时要使用的镜像。
7. 从 **Network Interfaces** 列表中，为主机网络接口选择网络参数。您可以创建多个网络接口。但是，至少有一个接口必须指向 **Capsule** 管理的网络。
8. 在 **Storage** 区域，输入主机的存储参数。您可以为主机创建多个卷。
9. 点 **Submit** 将设置保存到 **compute** 配置集。

CLI 过程

1. 运行以下命令来创建计算配置集：

```
# hammer compute-profile create --name "Libvirt CP"
```

2. 要为 **compute** 配置集添加值，请输入以下命令：

```
# hammer compute-profile values create --compute-profile "Libvirt CP" \
--compute-resource "My_KVM_Server" \
--interface
"compute_type=network,compute_model=virtio,compute_network=examplenetwork" \
--volume "pool_name=default,capacity=20G,format_type=qcow2" \
--compute-attributes "cpus=1,memory=1073741824"
```

9.5. 在 KVM 上创建主机

在 **Satellite** 中，您可以使用 **KVM** 置备通过网络连接或从现有镜像创建主机：

- 如果要通过网络连接创建主机，新主机必须能够访问 **Satellite** 服务器的集成胶囊或 **KVM** 虚拟网络上的外部 **Capsule** 服务器，以便主机能够访问 **PXE** 置备服务。此新主机条目会触发 **KVM** 服务器来创建和启动虚拟机。如果虚拟机通过虚拟网络检测到定义的 **Capsule** 服务器，虚拟机将启动至 **PXE**，并开始安装所选操作系统。
- 如果要使用现有镜像创建主机，新的主机条目会触发 **KVM** 服务器使用预先存在的镜像作为新卷的基础来创建虚拟机。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

DHCP Conflicts

对于基于网络的置备，如果您在 KVM 服务器上使用虚拟网络进行调配，请选择不提供 DHCP 分配的网络。这会在引导新主机时导致 DHCP 与 Satellite 服务器冲突。

流程

1. 在 Satellite Web UI 中，导航到 Hosts > Create Host。
2. 在 Name 字段中输入主机的名称。
3. 点 Organization and Location 选项卡，以确保置备上下文自动设置为当前上下文。
4. 从 Host Group 列表中，选择您要用来填充表单的主机组。
5. 从 Deploy on 列表中，选择 KVM 连接。
6. 从 Compute Profile 列表中，选择要用于自动填充虚拟机设置的配置集。
7. 点 Interface 选项卡，点主机接口上的 Edit。
8. 验证这些字段是否已自动填充，特别是以下项目：
 - Host 选项卡中的 Name 成为 DNS 名称。
 - Satellite 服务器自动为新主机分配 IP 地址。
 - MAC 地址 字段为空。KVM 服务器为主机分配一个 MAC 地址。

- 为主机上的第一个接口自动选择 **Managed**、**Primary** 和 **Provision** 选项。如果没有，请选择它们。
 - 特定于 KVM 的字段填充您的计算配置文件中的设置。如果需要，修改这些设置。
9. 单击 **Operating System** 选项卡，并确认所有字段都自动包含值。
 10. 选择要使用的 **Provisioning** 方法：
 - 对于基于网络的置备，请单击 **Network Based**。
 - 对于基于镜像的调配，请点击 **Image Based**。
 11. 点 **Provisioning** 模板中的 **Resolve**，以检查新主机可以识别要使用的正确置备模板。
 12. 单击 **Virtual Machine** 选项卡，并确认这些设置填充了主机组和计算配置集的详情。修改这些设置以符合您的需要。
 13. 单击 **Parameters** 选项卡，并确保存在一个提供激活码的参数。如果没有，请添加激活码。
 14. 点 **Submit** 保存主机条目。

CLI 过程

- 要使用基于网络的调配，请使用 **hammer host create** 命令创建主机，并包含 **--provision-method build**。将以下示例中的值替换为适合您的环境的值。

```
# hammer host create \
--name "kvm-host1" \
--organization "My_Organization" \
--location "New York" \
--hostgroup "Base" \
--compute-resource "My_KVM_Server" \
--provision-method build \
--build true \
```

```
--enabled true \  
--managed true \  
--interface  
"managed=true,primary=true,provision=true,compute_type=network,compute_network=exam  
plenetwork" \  
--compute-attributes="cpus=1,memory=1073741824" \  
--volume="pool_name=default,capacity=20G,format_type=qcow2" \  
--root-password "password"
```

- 要使用基于镜像的调配，请使用 **hammer host create** 命令创建主机，并包含 **--provision-method** 镜像。将以下示例中的值替换为适合您的环境的值。

```
# hammer host create \  
--name "kvm-host2" \  
--organization "My_Organization" \  
--location "New York" \  
--hostgroup "Base" \  
--compute-resource "My_KVM_Server" \  
--provision-method image \  
--image "KVM Image" \  
--enabled true \  
--managed true \  
--interface  
"managed=true,primary=true,provision=true,compute_type=network,compute_network=exampl  
enetwork" \  
--compute-attributes="cpus=1,memory=1073741824" \  
--volume="pool_name=default,capacity=20G,format_type=qcow2"
```

有关此计算资源的其他主机创建参数的更多信息，请输入 **hammer host create --help** 命令。

第 10 章 在 RED HAT VIRTUALIZATION 上置备虚拟机

Red Hat Virtualization 是一个企业级服务器和桌面虚拟化平台。在 Red Hat Satellite 中，您可以通过 Red Hat Virtualization 的 REST API 管理虚拟化功能。这包括创建虚拟机并控制其电源状态。

您可以使用 Red Hat Virtualization 置备通过网络连接或从现有镜像创建虚拟机。

您可以使用 cloud-init 来配置您调配的虚拟机。使用 cloud-init 避免网络上任何特殊配置（如受管 DHCP 和 TFTP）完成虚拟机的安装。此方法不需要 Satellite 使用 SSH 连接到调配的虚拟机来运行完成脚本。

先决条件

- 您可以将同步的内容软件仓库用于 Red Hat Enterprise Linux。有关更多信息，[请参阅 内容管理指南中的同步存储库](#)。
- 为主机注册提供激活码。如需更多信息，[请参阅 内容管理指南中的创建激活码](#)。
- 在 Red Hat Virtualization 环境中管理逻辑网络的 Capsule 服务器。确保没有在此网络上运行其他 DHCP 服务，以避免与 Capsule 服务器冲突。如需更多信息，[请参阅置备指南中的配置网络](#)。
- 如果要使用基于镜像的置备，现有的模板（除空白模板之外）。有关为虚拟机创建模板的更多信息，[请参阅 Red Hat Virtualization 虚拟机管理指南中的模板](#)。
- Red Hat Virtualization 上的类似管理的用户，用于与 Satellite 服务器通信。不要将 admin@internal 用户用于此通信。反之，使用以下权限创建新的 Red Hat Virtualization 用户：
 - system > Configure System > Login Permissions
 - Network > Configure vNIC Profile > Create
 - Network > Configure vNIC Profile > Edit Properties

- **Network > Configure vNIC Profile > Delete**
- **Network > Configure vNIC Profile > Assign vNIC Profile to VM**
- **Network > Configure vNIC Profile > Assign vNIC Profile to Template**
- **Template > Provisioning Operations > Import/Export**
- **VM > Provisioning Operations > Create**
- **VM > Provisioning Operations > Delete**
- **VM > Provisioning Operations > Import/Export**
- **VM > Provisioning Operations > Edit Storage**
- **Disk > Provisioning Operations > Create**
- **Disk > Disk Profile > Attach Disk Profile**

有关如何在 Red Hat Virtualization 中创建用户和添加权限的更多信息，请参阅 *Red Hat Virtualization 管理指南中的从管理门户管理用户任务*。

10.1. 将 RED HAT VIRTUALIZATION 连接添加到 SATELLITE 服务器

使用这个流程在 Satellite 中将 Red Hat Virtualization 添加为计算资源。要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Infrastructure > Compute Resources**，再点 **Create Compute Resource**。

2. 在 **Name** 字段中输入新计算资源的名称。
3. 从 **Provider** 列表中，选择 **RHV**。
4. 在 **Description** 字段中，输入计算资源的描述。
5. 在 **URL** 字段中输入 **Red Hat Virtualization Manager API** 的连接 **URL**，格式为：
https://rhv.example.com/ovirt-engine/api/v4。
6. 在 **User** 字段中，输入具有访问 **Red Hat Virtualization Manager** 资源权限的用户名称。
7. 在 **Password** 字段中，输入用户的密码。
8. 点 **Load Datacenters**，使用 **Red Hat Virtualization** 环境中的数据中心填充电信列表。
9. 从 **Datacenter** 列表选择一个数据中心。
10. 从 **Quota ID** 列表中，选择一个配额来限制 **Satellite** 可用资源。
11. 在 **X509 认证颁发机构** 字段中，为 **SSL/TLS** 访问输入证书颁发机构。或者，如果您将字段留空，服务器会在第一个 **API** 请求上生成一个自签名证书。
12. 单击 **Locations** 选项卡，再选择您要使用的位置。
13. 单击 **Organizations** 选项卡，再选择您要使用的组织。
14. 点 **Submit** 保存计算资源。

- 为 `--provider` 输入 `hammer compute-resource create` 命令，以及您要用于 `--datacenter` 的数据中心的名称。

```
# hammer compute-resource create \  
--name "My_RHV" --provider "Ovirt" \  
--description "RHV server at rhv.example.com" \  
--url "https://rhv.example.com/ovirt-engine/api" \  
--user "Satellite_User" --password "My_Password" \  
--locations "New York" --organizations "My_Organization" \  
--datacenter "My_Datacenter"
```

10.2. 在 RED HAT VIRTUALIZATION 中准备 CLOUD-INIT 镜像

要在置备过程中使用 `cloud-init`，您必须准备 Red Hat Virtualization 中安装的 `cloud-init` 镜像，然后将镜像导入到 `Satellite` 以用于置备。

流程

1. 在 Red Hat Virtualization 中，创建一个虚拟机，以便在 `Satellite` 中进行基于镜像的调配。
2. 在虚拟机上安装 `cloud-init`：

```
yum install cloud-init
```

3. 在 `/etc/cloud/cloud.cfg` 文件中，添加以下信息：

```
datasource_list: ["NoCloud", "ConfigDrive"]
```

4. 在 Red Hat Virtualization 中，从此虚拟机创建镜像。

将此镜像添加到 `Satellite` 时，请确保选中 `User Data` 复选框。

10.3. 将 RED HAT VIRTUALIZATION 镜像添加到 SATELLITE 服务器中

要使用基于镜像的调配创建主机，您必须将有关镜像的信息（如访问详情和镜像位置）添加到 `Satellite` 服务器中。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Infrastructure > Compute Resources**，然后点击 **Red Hat Virtualization** 连接的名称。
2. 点 **Create Image**。
3. 在 **Name** 字段中输入镜像的名称。
4. 从 **Operating System** 列表中，选择镜像的基础操作系统。
5. 从 **Architecture** 列表中，选择操作系统架构。
6. 在 **Username** 字段中，为镜像访问输入 **SSH** 用户名。这通常是 **root** 用户。
7. 在 **Password** 字段中，输入 **SSH** 密码以进行镜像访问。
8. 从 **Image** 列表中选择 **Red Hat Virtualization** 计算资源的镜像。
9. 可选：如果镜像支持用户数据输入，如 **cloud-init** 数据，请选择 **User Data** 复选框。
10. 点 **Submit** 保存镜像详情。

CLI 过程

- 使用 `hammer compute-resource image create` 命令创建镜像。使用 `--uuid` 选项，将模板 **UUID** 存储在 **Red Hat Virtualization** 服务器上。

```
# hammer compute-resource image create \  
--name "RHV_Image" \  

```

```
--compute-resource "My_RHV"
--operatingsystem "RedHat version" \
--architecture "x86_64" \
--username root \
--uuid "9788910c-4030-4ae0-bad7-603375dd72b1" \
```

10.4. 准备 CLOUD-INIT 模板

流程

1. 在 **Satellite Web UI** 中，导航到 **Hosts > Provisioning Templates**，然后点 **Create Template**。
2. 在 **Name** 字段中输入模板的名称。
3. 在 **Editor** 字段中输入以下模板详情：

```
<%#
kind: user_data
name: Cloud-init
-%>
#cloud-config
hostname: <%= @host.shortname %>

<%# Allow user to specify additional SSH key as host parameter -%>
<% if @host.params['sshkey'].present? ||
@host.params['remote_execution_ssh_keys'].present? -%>
ssh_authorized_keys:
<% if @host.params['sshkey'].present? -%>
  - <%= @host.params['sshkey'] %>
<% end -%>
<% if @host.params['remote_execution_ssh_keys'].present? -%>
<% @host.params['remote_execution_ssh_keys'].each do |key| -%>
  - <%= key %>
<% end -%>
<% end -%>
<% end -%>
<% end -%>
runcmd:
  - |
    #!/bin/bash
<%= indent 4 do
  snippet 'subscription_manager_registration'
end %>
<% if @host.info['parameters']['realm'] && @host.realm && @host.realm.realm_type == 'Red
Hat Identity Management' -%>
  <%= indent 4 do
    snippet 'freeipa_register'
  end %>
<% end -%>
```

```

<% unless @host.operatingsystem.atomic? -%>
  # update all the base packages from the updates repository
  yum -t -y -e 0 update
<% end -%>
<%
  # safemode renderer does not support unary negation
  non_atomic = @host.operatingsystem.atomic? ? false : true
  pm_set = @host.puppetmaster.empty? ? false : true
  puppet_enabled = non_atomic && (pm_set || @host.params['force-puppet'])
%>
<% if puppet_enabled %>
  yum install -y puppet
  cat > /etc/puppet/puppet.conf << EOF
  <%= indent 4 do
    snippet 'puppet.conf'
  end %>
  EOF
  # Setup puppet to run on system reboot
  /sbin/chkconfig --level 345 puppet on

  /usr/bin/puppet agent --config /etc/puppet/puppet.conf --onetime --tags no_such_tag <%=
  @host.puppetmaster.blank? ? " : "--server #{@host.puppetmaster}" %> --no-daemonize
  /sbin/service puppet start
<% end -%>
phone_home:
url: <%= foreman_url('built') %>
post: []
tries: 10pp

```

4. 点 **Type** 选项卡，从 **Type** 列表中选择 **User data template**。
5. 单击 **关联** 选项卡，然后从 **Applicable Operating Systems** 列表中选择您要与模板关联的操作系统。
6. 单击 **Locations** 选项卡，然后从位置列表中选择您要与模板关联的位置。
7. 点 **Organizations** 选项卡，从 **Organization** 列表中选择您要与模板关联的机构。
8. 点 **Submit**。
9. 在 **Satellite Web UI** 中，导航到 **Hosts > Operating Systems**，然后选择您要与模板关联的操作系统。

10. 单击 **Templates** 选项卡，然后从 **User data template** 列表中选择新模板的名称。
11. 点 **Submit**。

10.5. 在 COMPUTE 配置文件中添加 RED HAT VIRTUALIZATION 详情

使用这个流程将 Red Hat Virtualization 硬件设置添加到计算配置集中。当您使用这个计算配置集在 KVM 上创建主机时，这些设置会自动填充。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Infrastructure > Compute Profiles**。
2. 在 **Compute Profiles** 窗口中，点现有计算配置集的名称，或者单击 **Create Compute Profile**，输入 **Name**，然后单击 **Submit**。
3. 点 **Red Hat Virtualization** 计算资源的名称。
4. 在 **Cluster** 列表中选择 **Red Hat Virtualization** 环境中的目标主机集群。
5. 从 **Template** 列表中，选择用于 **Cores** 和 **Memory** 设置的 **RHV** 模板。
6. 在 **Cores** 字段中，输入要分配给新主机的 **CPU** 内核数。
7. 在 **Memory** 字段中，输入分配给新主机的内存量。
8. 从 **Image** 列表中，选择要用于基于镜像的置备的镜像。
9. 在 **网络接口** 区域中，为主机网络接口输入网络参数。您可以创建多个网络接口。但是，至少

有一个接口必须指向 Capsule 管理的网络。对于每个网络接口，请输入以下详情：

- a. 在 **Name** 字段中输入网络接口的名称。
- b. 在 **Network** 列表中选择您要使用的逻辑网络。

10.

在 **Storage** 区域，输入主机的存储参数。您可以为主机创建多个卷。对于每个卷，请输入以下详情：

- a. 在 **Size (GB)** 中，输入新卷的大小（以 GB 为单位）。
- b. 从 **存储域** 列表中，选择卷的存储域。
- c. 在 **Preallocate disk** 中，选择 **thin provisioning** 或 **preallocation of the full disk**。
- d. 在 **Bootable** 列表中，选择是否希望可引导的卷或不可启动的卷。

11.

点 **Submit** 保存计算配置集。

CLI 过程

1.

运行以下命令来创建计算配置集：

```
# hammer compute-profile create --name "Red Hat Virtualization CP"
```

2.

要为 **compute** 配置集设置值，请输入以下命令：

```
# hammer compute-profile values create --compute-profile "Red Hat Virtualization CP" \
--compute-resource "My_RHV" \
--interface
"compute_interface=Interface_Type,compute_name=eth0,compute_network=satnetwork" \
--volume "size_gb=20G,storage_domain=Data,bootable=true" \
--compute-attributes "cluster=Default,cores=1,memory=1073741824,start=true"
```

10.6. 在 RED HAT VIRTUALIZATION 上创建主机

在 Satellite 中，您可以使用 Red Hat Virtualization 置备通过网络连接或现有镜像创建主机：

- 如果要通过网络连接创建主机，新主机必须能够访问 Satellite 服务器的集成胶囊或 Red Hat Virtualization 虚拟网络上的外部 Capsule 服务器，以便主机能够访问 PXE 置备服务。这个新主机条目会触发 Red Hat Virtualization 服务器来创建和启动虚拟机。如果虚拟机通过虚拟网络检测到定义的 Capsule 服务器，虚拟机将启动至 PXE，并开始安装所选操作系统。
- 如果要使用现有镜像创建主机，新的主机条目会触发 Red Hat Virtualization 服务器使用预先存在的镜像作为新卷的基础来创建虚拟机。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

DHCP Conflicts

对于基于网络的置备，如果您在 Red Hat Virtualization 服务器上使用虚拟网络进行置备，请选择不提供 DHCP 分配的网络。这会在引导新主机时导致 DHCP 与 Satellite 服务器冲突。

流程

1. 在 Satellite Web UI 中，导航到 Hosts > Create Host。
2. 在 Name 字段中输入主机的名称。
3. 点 Organization and Location 选项卡，以确保置备上下文自动设置为当前上下文。
4. 从 Host Group 列表中，选择您要用来填充表单的主机组。
5. 从 Deploy on 列表中，选择 Red Hat Virtualization 连接。
6. 从 Compute Profile 列表中，选择要用于自动填充虚拟机设置的配置集。

7. 点 **Interface** 选项卡，点主机接口上的 **Edit**。
8. 验证这些字段是否已自动填充，特别是以下项目：
 - **Host** 选项卡中的 **Name** 成为 DNS 名称。
 - **Satellite** 服务器自动为新主机分配 IP 地址。
 - **MAC 地址** 字段为空。**Red Hat Virtualization** 服务器为主机分配一个 **MAC** 地址。
 - 为主机上的第一个接口自动选择 **Managed**、**Primary** 和 **Provision** 选项。如果没有，请选择它们。
 - **Red Hat Virtualization** 特定字段使用计算配置文件中的设置填充。如果需要，修改这些设置。
9. 单击 **Operating System** 选项卡，并确认所有字段都自动包含值。
10. 选择要使用的 **Provisioning** 方法：
 - 对于基于网络的置备，请单击 **Network Based**。
 - 对于基于镜像的调配，请点击 **Image Based**。
11. 点 **Provisioning** 模板中的 **Resolve**，以检查新主机可以识别要使用的正确置备模板。
12. 单击 **Virtual Machine** 选项卡，并确认这些设置填充了主机组和计算配置集的详情。修改这些设置以符合您的需要。

13. 单击 **Parameters** 选项卡，并确保存在一个提供激活码的参数。如果没有，请添加激活码。
14. 点 **Submit** 保存主机条目。

CLI 过程

- 要使用基于网络的调配，请使用 **hammer host create** 命令创建主机，并包含 **--provision-method build**。将以下示例中的值替换为适合您的环境的值。

```
# hammer host create \  
--name "RHV-vm1" \  
--organization "My_Organization" \  
--location "New York" \  
--hostgroup "Base" \  
--compute-resource "My_RHV" \  
--provision-method build \  
--build true \  
--enabled true \  
--managed true \  
--interface  
"managed=true,primary=true,provision=true,compute_name=eth0,compute_network=satnetwor  
k" \  
--compute-attributes="cluster=Default,cores=1,memory=1073741824,start=true" \  
--volume="size_gb=20G,storage_domain=Data,bootable=true"
```

- 要使用基于镜像的调配，请使用 **hammer host create** 命令创建主机，并包含 **--provision-method 镜像**。将以下示例中的值替换为适合您的环境的值。

```
# hammer host create \  
--name "RHV-vm2" \  
--organization "My_Organization" \  
--location "New York" \  
--hostgroup "Base" \  
--compute-resource "My_RHV" \  
--provision-method image \  
--image "RHV_Image" \  
--enabled true \  
--managed true \  
--interface  
"managed=true,primary=true,provision=true,compute_name=eth0,compute_network=satnetwor  
k" \  
--compute-attributes="cluster=Default,cores=1,memory=1073741824,start=true" \  
--volume="size_gb=20G,storage_domain=Data,bootable=true"
```

有关此计算资源的其他主机创建参数的更多信息，请输入 **hammer host create --help** 命令。

第 11 章 在 VMWARE VSPHERE 中置备虚拟机

VMware vSphere 是 VMware 的一个企业级虚拟化平台。Red Hat Satellite 可以与 vSphere 平台交互，包括创建新虚拟机并控制其电源管理状态。

11.1. 安装 VMWARE 插件

安装 VMware 插件，将 VMware 计算资源提供程序附加到 Satellite。这可让您管理和将主机部署到 VMware。

流程

1. 在 Satellite 服务器上安装 VMware 计算资源供应商：

```
# satellite-installer --enable-foreman-compute-vmware
```

2. 可选：在 Satellite Web UI 中，导航到 **Administer > About** 并选择 **compute resources** 选项卡来验证 VMware 插件的安装。

11.2. VMWARE 置备的先决条件

VMware vSphere 置备的要求包括：

- 支持的 VMware vCenter Server 版本。以下版本已使用 Satellite 进行了全面测试：
 - vCenter Server 7.0
 - vCenter Server 6.7 (EOL)
 - vCenter Server 6.5 (EOL)
- 在 vSphere 环境中管理网络的 Capsule 服务器。确保没有在此网络上运行其他 DHCP 服务，以避免与 Capsule 服务器冲突。更多信息请参阅 [第 3 章 配置网络](#)。

- 如果要使用基于镜像的置备，现有的 VMware 模板。
- 您可以将同步的内容软件仓库用于 Red Hat Enterprise Linux。有关更多信息，[请参阅 内容管理指南中的同步存储库](#)。
- 为主机注册提供激活码。如需更多信息，[请参阅 内容管理指南中的创建激活码](#)。

11.3. 创建 VMWARE 用户

VMware vSphere 服务器需要类似管理的用户进行 Satellite 服务器通信。为安全起见，请不要将管理员用户用于此类通信。相反，请创建具有以下权限的用户：

对于 VMware vCenter Server 版本 6.7，请设置以下权限：

- All Privileges → Datastore → Allocate Space, Browse datastore, Update Virtual Machine files, Low level file operations
- All Privileges → Network → Assign Network
- All Privileges → Resource → Assign virtual machine to resource pool
- All Privileges → Virtual Machine → Change Config (All)
- All Privileges → Virtual Machine → Interaction (All)
- All Privileges → Virtual Machine → Edit Inventory (All)
- All Privileges → Virtual Machine → Provisioning (All)
- 所有 Privileges → Virtual Machine → Guest Operations (All)

请注意，相同的步骤也适用于 VMware vCenter Server 版本 7.0。

对于 VMware vCenter Server 版本 6.5，请设置以下权限：

- All Privileges → Datastore → Allocate Space, Browse datastore, Update Virtual Machine files, Low level file operations
- All Privileges → Network → Assign Network
- All Privileges → Resource → Assign virtual machine to resource pool
- All Privileges → Virtual Machine → Configuration (All)
- All Privileges → Virtual Machine → Interaction (All)
- All Privileges → Virtual Machine → Inventory (All)
- All Privileges → Virtual Machine → Provisioning (All)
- 所有 Privileges → Virtual Machine → Guest Operations (All)

11.4. 向 SATELLITE 服务器添加 VMWARE 连接

使用这个流程在 Satellite 服务器的计算资源中添加 VMware vSphere 连接。要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

先决条件

- 确保将主机和基于网络的防火墙配置为允许从 Satellite 服务器到 TCP 端口 443 上的 vCenter 的通信。

- 验证 **Satellite** 服务器和 **vCenter** 是否可以解析彼此的主机名。

流程

1. 在 **Satellite Web UI** 中，导航到 **Infrastructure > Compute Resources**，然后在 **Compute Resources** 窗口中点 **Create Compute Resource**。
2. 在 **Name** 字段中输入资源名称。
3. 从 **Provider** 列表中，选择 **VMware**。
4. 在 **Description** 字段中，输入资源的描述。
5. 在 **VCenter/Server** 字段中输入 **vCenter** 服务器的 IP 地址或主机名。
6. 在 **User** 字段中，输入具有访问 **vCenter** 资源权限的用户名称。
7. 在 **Password** 字段中，输入用户的密码。
8. 点 **Load Datacenters**，从 **VMware vSphere** 环境中填充数据中心列表。
9. 从 **Datacenter** 列表中，选择要从此列表管理的特定数据中心。
10. 在 **Fingerprint** 字段中，确保使用数据中心的指纹填充此字段。
11. 从 **Display Type** 列表中，选择一个控制台类型，例如 **VNC** 或 **VMRC**。请注意，**VNC** 控制台 在 **VMware ESXi 6.5** 及更新的版本上不被支持。
12. 可选：在 **VNC Console Passwords** 字段中，选择 **Set a random generated password on the display connection** 复选框，为具有随机生成的密码的新主机保护控制台访问。您可以检索 **VNC** 控制台的密码，以从以下命令的输出中从 **libvirtd** 主机访问客户机虚拟机控制台：

```
# virsh edit your_VM_name
<graphics type='vnc' port='-1' autoport='yes' listen='0.0.0.0'
passwd='your_randomly_generated_password>
```

每次打开虚拟机的控制台时都会随机生成密码，例如使用 `virt-manager`。

13.

在 **Enable Cache** 列表中，您可以选择是否启用计算资源的缓存。更多信息请参阅 [第 11.11 节“计算资源缓存”](#)。

14.

单击 **位置和机构** 选项卡，并验证这些值是否自动设置为您当前的上下文。您还可以添加额外的上下文。

15.

点 **Submit** 保存连接。

CLI 过程

-

使用 `hammer compute-resource create` 命令创建连接。选择 **Vmware** 作为 `--provider`，并将数据中心的实例 **UUID** 设置为 `--uuid`：

```
# hammer compute-resource create \
--datacenter "My_Datacenter" \
--description "vSphere server at vsphere.example.com" \
--locations "My_Location" \
--name "My_vSphere" \
--organizations "My_Organization" \
--password "My_Password" \
--provider "Vmware" \
--server "vsphere.example.com" \
--user "My_User"
```

11.5. 将 VMWARE 镜像添加到 SATELLITE 服务器中

VMware vSphere 使用模板作为创建新虚拟机的镜像。如果使用基于镜像的置备来创建新主机，则需要 **Satellite 服务器** 中添加 **VMware 模板** 详情。这包括访问详情和模板名称。

要使用 **CLI** 而不是 **Satellite Web UI**，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Infrastructure > Compute Resources**。
2. 选择 **Vmware** 计算资源。
3. 点 **Create Image**。
4. 在 **Name** 字段中输入镜像的名称。
5. 从 **Operating System** 列表中，选择镜像的基础操作系统。
6. 从 **Architecture** 列表中，选择操作系统架构。
7. 在 **Username** 字段中，为镜像访问输入 **SSH** 用户名。默认情况下，这设置为 **root**。
8. 如果您的镜像支持用户数据输入，如 **cloud-init** 数据，点 **User data** 复选框。
9. 可选：在 **Password** 字段中输入 **SSH** 密码来访问镜像。
10. 从 **Image** 列表中，从 **VMware** 选择一个镜像。
11. 点 **Submit** 保存镜像详情。

CLI 过程

- 使用 **hammer compute-resource image create** 命令创建镜像。使用 **--uuid** 字段将相对模板路径存储在 **vSphere** 环境中：

```
# hammer compute-resource image create \  
--architecture "My_Architecture" \  
--compute-resource "My_VMware" \  
--name "My_Image" \  
--uuid "My_UUID"
```

```
--operatingsystem "My_Operating_System" \  
--username root \  
--uuid "My_UUID"
```

11.6. 在计算配置集中添加 VMWARE 详情

您可以为 VMware vSphere 上的虚拟机预定义某些硬件设置。您可以通过在计算配置集中添加这些硬件设置来实现此目的。要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 Infrastructure > Compute Profiles。
2. 选择一个计算配置集。
3. 选择 VMware 计算资源。
4. 在 CPU 字段中，输入要分配给主机的 CPU 数量。
5. 在 Cores per socket 字段中，输入分配给每个 CPU 的内核数。
6. 在 Memory 字段中，输入要分配给主机的 MiB 的内存量。
7. 在 Firmware 复选框中，选择 BIOS 或 UEFI 作为主机的固件。默认情况下，它被设置为 *automatic*。
8. 在 Cluster 列表中，在 VMware 环境中选择目标主机集群名称。
9. 从 Resource pool 列表中，为主机选择一个可用的资源分配。
10. 在 Folder 列表中，选择要组织主机的文件夹。

11. 从 **Guest OS** 列表中，选择在 **VMware vSphere** 中使用的操作系统。
12. 从 **Virtual H/W version** 列表中，选择用于虚拟机的底层 **VMware** 硬件抽象。
13. 如果要在虚拟机开机时添加更多内存，请选择 **Memory hot add** 复选框。
14. 如果要在虚拟机开机时添加更多 CPU，请选择 **CPU 热添加** 复选框。
15. 如果要添加 **CD-ROM** 驱动器，请选择 **CD-ROM 驱动器** 复选框。
16. 从 **Boot order** 列表中，定义虚拟机尝试启动的顺序。
17. 可选：在 **注解备注** 字段中，输入任意描述。
18. 如果使用基于镜像的置备，请从 **Image** 列表中选择镜像。
19. 从 **SCSI 控制器** 列表中，选择主机的磁盘访问方法。
20. 如果要使用 **eager zero thick provisioning**，请选中 **Eager 零** 复选框。默认情况下，磁盘使用 **lazy 零** 密集置备。
21. 从 **Network Interfaces** 列表中，为主机网络接口选择网络参数。至少一个接口必须指向 **Capsule** 管理的网络。
22. 可选：点 **Add Interface** 创建另一个网络接口。
23. 点 **Submit** 保存计算配置集。

CLI 过程

1. 创建计算配置集：

```
# hammer compute-profile create --name "My_Compute_Profile"
```

2. 将 VMware 详情设置为计算配置集：

```
# hammer compute-profile values create \
--compute-attributes
"cpus=1,corespersocket=2,memory_mb=1024,cluster=MyCluster,path=MyVMs,start=true" \
--compute-profile "My_Compute_Profile" \
--compute-resource "My_VMware" \
--interface "compute_type=VirtualE1000,compute_network=mynetwork \
--volume "size_gb=20G,datastore=Data,name=myharddisk,thin=true"
```

11.7. 在 VMWARE 上创建主机

VMware vSphere 置备过程提供了通过网络连接或使用现有镜像创建主机的选项。

对于基于网络的置备，您必须创建一个主机来访问 Satellite 服务器的集成胶囊或 VMware vSphere 虚拟网络上的外部 Capsule 服务器，以便主机能够访问 PXE 置备服务。新主机条目会触发 VMware vSphere 服务器来创建虚拟机。如果虚拟机通过虚拟网络检测到定义的 Capsule 服务器，虚拟机将启动至 PXE，并开始安装所选操作系统。

DHCP Conflicts

如果您使用 VMware vSphere 服务器上的虚拟网络进行置备，请确保选择一个不提供 DHCP 分配的虚拟网络。这会在引导新主机时导致 DHCP 与 Satellite 服务器冲突。

对于基于镜像的置备，请使用预先存在的镜像作为新卷的基础。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 Hosts > Create Host。
2. 在 Name 字段中输入主机名。

3. 点 **Organization and Location** 选项卡，以确保置备上下文自动设置为当前上下文。
4. 从 **Host Group** 列表中，选择您要用来填充表单的主机组。
5. 从 **Deploy on** 列表中，选择 **VMware vSphere** 连接。
6. 从 **Compute Profile** 列表中，选择要用于自动填充基于虚拟机的设置的配置集。
7. 点 **Interface** 选项卡，点主机接口上的 **Edit**。
8. 验证这些字段是否自动填充值。特别是：
 - **Host** 选项卡中的 **Name** 成为 **DNS** 名称。
 - **Satellite** 服务器自动为新主机分配 **IP** 地址。
9. 确保 **MAC 地址** 字段为空。**VMware vSphere** 服务器为主机分配一个。
10. 验证是否为主机上的第一个接口自动选择 **Managed**、**Primary** 和 **Provision** 选项。如果没有，请选择它们。
11. 在接口窗口中，查看使用计算配置集中的设置填充的 **VMware vSphere** 特定字段。修改这些设置以符合您的需要。
12. 单击 **Operating System** 选项卡，并确认所有字段都自动包含值。
13. 选择您要的 **Provisioning** 方法：
 - 对于基于网络的置备，请单击 **Network Based**。

- 对于基于镜像的调配，请点击 **Image Based**。
 - 对于引导磁盘置备，请点击 **基于引导磁盘的 Boot disk**。
14. 点 **Provisioning** 模板中的 **Resolve**，以检查新主机可以识别要使用的正确置备模板。
 15. 单击 **Virtual Machine** 选项卡，并确认这些设置填充了主机组和计算配置集的详情。修改这些设置以符合您的要求。
 16. 单击 **Parameters** 选项卡，并确保存在一个提供激活码的参数。如果参数不存在，请单击 **+** 添加参数。在 **Name** 字段中，输入 **kt_activation_keys**。在 **Value** 字段中，输入用于注册 **Content Hosts** 的激活码的名称。
 17. 点 **Submit** 在 **VMware** 上置备您的主机。

CLI 过程

- 使用 **hammer host create** 命令从网络创建主机，并包含 **--provision-method build** 来使用基于网络的置备。

```
# hammer host create \
--build true \
--compute-
attributes="cpus=1,corespersocket=2,memory_mb=1024,cluster=MyCluster,path=MyVMs,start=
true" \
--compute-resource "My_VMware" \
--enabled true \
--hostgroup "My_Host_Group" \
--interface
"managed=true,primary=true,provision=true,compute_type=VirtualE1000,compute_network=my
network" \
--location "My_Location" \
--managed true \
--name "My_Host" \
--organization "My_Organization" \
--provision-method build \
--volume="size_gb=20G,datastore=Data,name=myharddisk,thin=true"
```

- 使用 **hammer host create** 命令从镜像创建主机，并包含 **--provision-method 镜像** 以使用基于镜像的置备。

```
# hammer host create \
--compute-
attributes="cpus=1,corespersocket=2,memory_mb=1024,cluster=MyCluster,path=MyVMs,start=
true" \
--compute-resource "My_VMware" \
--enabled true \
--hostgroup "My_Host_Group" \
--image "Test VMware Image" \
--interface
"managed=true,primary=true,provision=true,compute_type=VirtualE1000,compute_network=my
network" \
--location "My_Location" \
--managed true \
--name "My_Host" \
--organization "My_Organization" \
--provision-method image \
--volume="size_gb=20G,datastore=Data,name=myharddisk,thin=true"
```

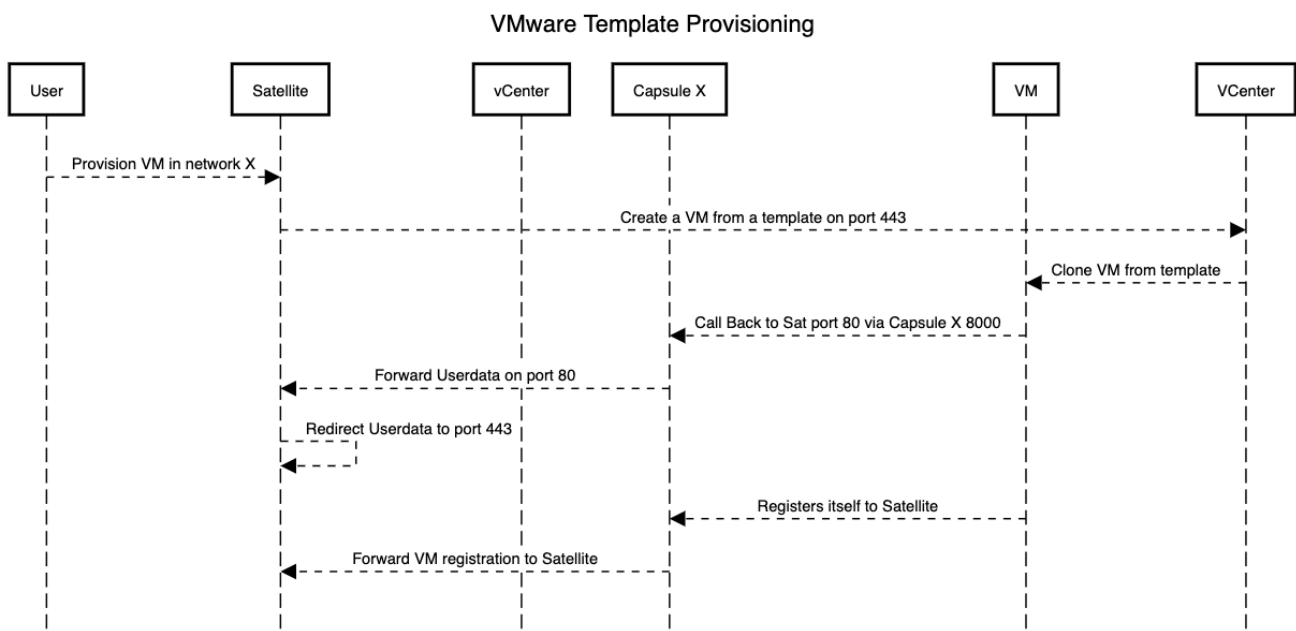
有关此计算资源的其他主机创建参数的更多信息，请输入 **hammer host create --help** 命令。

11.8. 使用 VMWARE CLOUD-INIT 和 USERDATA 模板进行置备

您可以将 VMware 与 Cloud-init 和 Userdata 模板一起使用，将用户数据插入到新虚拟机中，进行进一步的 VMware 自定义，并使 VMware 托管的虚拟机调用给 Satellite。

您可以使用相同的步骤在 Satellite 中设置 VMware 计算资源，并对工作流程进行一些修改。

图 11.1. VMware cloud-init 置备概述



当您在 Satellite 中为 VMware 置备设置计算资源和镜像时，会发生以下置备事件序列：

- 用户使用 Satellite Web UI、API 或 hammer 置备一个或多个虚拟机
- Satellite 调用 VMware vCenter 来克隆虚拟机模板
- Satellite 用户数据 置备模板添加自定义身份信息
- 置备完成后，Cloud-init 置备模板指示虚拟机在 cloud-init 运行时调用 Capsule
- VMware vCenter 将模板克隆到虚拟机
- VMware vCenter 对虚拟机的身份应用自定义，包括主机名、IP 和 DNS
- 虚拟机构建 cloud-init 被调用，并在端口 80 上调用后端 Satellite，然后重定向到 443

先决条件

- 配置端口和防火墙设置，以打开任何必要的连接。由于 cloud-init 服务，虚拟机始终调用回 Satellite，即使您将虚拟机注册到 Capsule。有关 [端口和防火墙要求](#) 的更多信息，请参阅安装 *Capsule 服务器* 中的 [在连接的网络环境中安装 Satellite 服务器](#) 中的 [端口和防火墙要求](#)。
- 如果要使用 Capsule 服务器而不是 Satellite 服务器，请确保已相应地配置了 Capsule 服务器。如需更多信息，请参阅安装 [Capsule 服务器](#) 中的 [为主机注册和置备配置 Capsule](#)。

将 Userdata 和 Cloud-init 模板与操作系统关联

1. 在 Satellite Web UI 中，进入到 Hosts > Templates > Provisioning Templates。
2. 搜索 CloudInit 默认模板，再单击其名称。

3. 单击 **关联** 选项卡。
4. 选择模板应用到的所有操作系统，然后单击 **Submit**。
5. 对 **UserData open-vm-tools** 模板重复上述步骤。
6. 进入 **Hosts > Provisioning Setup > Operating Systems**。
7. 选择您要用于置备的操作系统。
8. 点 **Templates** 选项卡。
9. 从 **Cloud-init** 模板列表中，选择 **CloudInit default**。
10. 从 **User data template** 列表中，选择 **UserData open-vm-tools**。
11. 点 **Submit** 保存更改。

准备镜像以使用 cloud-init 模板

要准备镜像，您必须首先配置虚拟机上所需的设置，然后可以保存为要在 **Satellite** 中使用的镜像。

要使用 **cloud-init** 模板进行调配，您必须配置虚拟机，以便安装、启用并配置为返回到 **Satellite** 服务器。

为了安全起见，您必须安装 **CA** 证书以使用 **HTTPS** 进行所有通信。此流程包括清理虚拟机的步骤，以便不需要的信息传送到用于置备的镜像中。

如果您使用 **cloud-init** 镜像，您仍然必须按照以下步骤启用 **cloud-init** 以便与 **Satellite** 通信，因为禁用了 **cloud-init**。

流程

1. 在用于创建镜像的虚拟机上，安装 **cloud-init**、**open-vm-tools** 和 **perl** 软件包：

```
# yum -y install cloud-init open-vm-tools perl
```

2. 通过 **cloud-init** 禁用网络配置：

```
# cat << EOM > /etc/cloud/cloud.cfg.d/01_network.cfg
network:
  config: disabled
EOM
```

3. 配置 **cloud-init** 以从 **Satellite** 获取数据：

```
# cat << EOM > /etc/cloud/cloud.cfg.d/10_datasource.cfg
datasource_list: [NoCloud]
datasource:
  NoCloud:
    seedfrom: https://satellite.example.com/userdata/
EOM
```

如果要通过 **Capsule** 服务器置备，请在 **seedfrom** 选项中使用 **Capsule** 服务器的 URL，如 **https:// Capsule.example.com:9090/userdata/**。

4. 配置要在 **cloud-init** 中使用的模块：

```
# cat << EOM > /etc/cloud/cloud.cfg
cloud_init_modules:
  - bootcmd
  - ssh

cloud_config_modules:
  - runcmd

cloud_final_modules:
  - scripts-per-once
  - scripts-per-boot
  - scripts-per-instance
  - scripts-user
  - phone-home

system_info:
  distro: rhel
  paths:
```

```
cloud_dir: /var/lib/cloud
templates_dir: /etc/cloud/templates
ssh_svcname: sshd
EOM
```

5. 为镜像启用 CA 证书：

```
# update-ca-trust enable
```

6. 从 Satellite 服务器下载 `katello-server-ca.crt` 文件：

```
# wget -O /etc/pki/ca-trust/source/anchors/cloud-init-ca.crt
https://satellite.example.com/pub/katello-server-ca.crt
```

如果要通过胶囊服务器调配，请从 Capsule 服务器下载文件，如 `https://Capsule.example.com/pub/katello-server-ca.crt`。

7. 更新证书的记录：

```
# update-ca-trust extract
```

8. 清理镜像：

```
# systemctl stop rsyslog
# systemctl stop auditd
# package-cleanup --oldkernels --count=1
# yum clean all
```

9. 减少日志空间、删除旧的日志和截断日志：

```
# logrotate -f /etc/logrotate.conf
# rm -f /var/log/*-???????? /var/log/*.gz
# rm -f /var/log/dmesg.old
# rm -rf /var/log/anaconda
# cat /dev/null > /var/log/audit/audit.log
# cat /dev/null > /var/log/wtmp
# cat /dev/null > /var/log/lastlog
# cat /dev/null > /var/log/grubby
```

10. 删除 udev 硬件规则：


```
# rm -f /etc/udev/rules.d/70*
```

11. 删除与现有网络配置相关的 `ifcfg` 脚本：

```
# rm -f /etc/sysconfig/network-scripts/ifcfg-ens*  
# rm -f /etc/sysconfig/network-scripts/ifcfg-eth*
```

12. 删除 SSH 主机密钥：

```
# rm -f /etc/ssh/ssh_host_*
```

13. 删除 root 用户的 SSH 历史记录：

```
# rm -rf ~root/.ssh/known_hosts
```

14. 删除 root 用户的 shell 历史记录：

```
# rm -f ~root/.bash_history  
# unset HISTFILE
```

15. 从此虚拟机创建镜像。

16. 将您的镜像添加到 [Satellite](#)。

11.9. 删除 VMWARE 上的虚拟机

您可以从 [Satellite](#) 中删除 [VMware](#) 上运行的虚拟机。

流程

1. 在 [Satellite Web UI](#) 中，导航到 [Infrastructure > Compute Resources](#)。
2. 选择 [VMware](#) 供应商。
- 3.

在 **Virtual Machines** 选项卡中，从 **Actions** 菜单中点 **Delete**。这会从 VMware 计算资源中删除虚拟机，同时在 **Satellite** 中保留任何关联的主机。如果要删除孤立的主机，请导航到 **Hosts > All Hosts** 并手动删除主机。

11.10. 从 VMWARE 导入虚拟机到 SATELLITE

您可以将 VMware 上运行的现有虚拟机导入到 Satellite 中。

流程

1. 在 **Satellite Web UI** 中，导航到 **Infrastructure > Compute Resources**。
2. 选择 **VMware** 计算资源。
3. 在 **Virtual Machines** 选项卡中，从 **Actions** 菜单中点 **Import as managed Host** 或 **Import as unmanaged Host**。以下页面类似于创建已选中计算资源的主机。如需更多信息，请参阅 [管理主机](#) 中的 [在 Satellite 中创建 主机](#)。
4. 单击 **Submit**，将虚拟机导入到 **Satellite**。

11.11. 计算资源缓存

计算资源的缓存加快 VMware 信息的呈现速度。

11.11.1. 启用计算资源的缓存

启用或禁用计算资源的缓存：

流程

1. 在 **Satellite Web UI** 中，导航到 **Infrastructure > Compute Resources**。
2. 点击您要更新的 **VMware** 服务器右侧的 **Edit** 按钮。

3. 选中 启用缓存 复选框。

11.11.2. 刷新计算资源缓存

刷新计算资源的缓存，以更新计算资源信息。

流程

1. 在 Satellite Web UI 中，导航到 Infrastructure > Compute Resources。
2. 选择您要刷新计算资源缓存的 VMware 服务器，然后点击 Refresh Cache 按钮。

CLI 过程

- 使用此 API 调用来刷新计算资源缓存：

```
# curl -H "Accept:application/json" \  
-H "Content-Type:application/json" -X PUT \  
-u username:password -k \  
https://satellite.example.com/api/compute_resources/compute_resource_id/refresh_cache
```

使用 hammer 计算资源 列表来确定您要刷新计算资源缓存的 VMware 服务器的 ID。

第 12 章 在 OPENSIFT VIRTUALIZATION 中置备虚拟机

OpenShift Virtualization 满足已采用或希望采用 Kubernetes 但现有虚拟机(VM)的工作负载无法轻松容器化的开发团队的需求。此技术提供了一个统一的开发平台，开发人员可以在共享环境中的应用程序容器和虚拟机构建、修改和部署应用程序。这些功能支持跨开放混合云快速应用程序现代化。

使用 Red Hat Satellite，您可以为 OpenShift Virtualization 创建计算资源，以便您可以使用 Satellite 来置备和管理 Kubernetes 虚拟机。

请注意，这个版本不支持模板置备。



重要

OpenShift Virtualization 计算资源只是一个技术预览功能。技术预览功能不受红帽产品服务等级协议 (SLA) 支持，且功能可能并不完整。红帽不推荐在生产环境中使用它们。这些技术预览功能可以使用户提早试用新的功能，并有机会在开发阶段提供反馈意见。有关红帽技术预览功能支持范围的详情，请参考

<https://access.redhat.com/support/offerings/techpreview/>。

先决条件

- 您可以将同步的内容软件仓库用于 Red Hat Enterprise Linux。有关更多信息，请参阅 [内容管理指南中的同步存储库](#)。
- 为主机注册提供激活码。如需更多信息，请参阅 [内容管理指南中的创建激活码](#)。
- 具有 OpenShift Container Platform 虚拟集群的 cluster-admin 权限的 OpenShift Virtualization 用户。如需更多信息，请参阅 OpenShift Container Platform 文档中的 [身份验证指南中的使用 RBAC 定义和应用权限](#)。
- 在 OpenShift Virtualization 服务器上管理网络的胶囊服务器。确保没有在此网络上运行其他 DHCP 服务，以避免与 Capsule 服务器冲突。有关 Capsule 服务器的网络服务配置的更多信息，请参阅 [置备指南中的配置网络](#)。
- 具有以下角色的 Satellite 用户帐户：

- 编辑主机
- 查看主机

如需更多信息，[请参阅](#) *管理 Red Hat Satellite 指南*中的将角色分配给用户。

- 在 Satellite 中具有以下权限的自定义角色：

- view_compute_resources
- destroy_compute_resources_vms
- power_compute_resources_vms
- create_compute_resources_vms
- view_compute_resources_vms
- view_locations
- view_subnets

有关创建角色的更多信息，[请参阅](#)*管理 Red Hat Satellite 指南*中的创建角色。有关向角色添加权限的更多信息，[请参阅](#) *管理 Red Hat Satellite 指南*中的向角色添加权限。

12.1. 在 SATELLITE 服务器中添加 OPENSIFT VIRTUALIZATION 连接

使用这个流程在 Satellite 中添加 OpenShift Virtualization 作为计算资源。

流程

1. 输入以下 `satellite-installer` 命令，为 **Satellite** 启用 **OpenShift Virtualization** 插件：

```
# satellite-installer --enable-foreman-plugin-kubevirt
```

2. 生成用于 **HTTP** 和 **HTTP** 身份验证的 **bearer** 令牌。在 **OpenShift Virtualization** 服务器上，列出包含令牌的 **secret**：

```
# kubectl get secrets
```

3. 列出 **secret** 的令牌：

```
# kubectl get secrets YOUR_SECRET -o jsonpath='{.data.token}' | base64 -d | xargs
```

记录此令牌以便在此流程中稍后使用。

4. 在 **Satellite Web UI** 中，导航到 **Infrastructure > Compute Resources**，再点 **Create Compute Resource**。

5. 在 **Name** 字段中输入新计算资源的名称。

6. 从 **Provider** 列表中，选择 **OpenShift Virtualization**。

7. 在 **Description** 字段中，输入计算资源的描述。

8. 在 **Hostname** 字段中，输入您要使用的 **OpenShift Virtualization** 服务器的 **FQDN**、主机名或 **IP 地址**。

9. 在 **API Port** 字段中，输入您要用于从 **Satellite** 到 **OpenShift Virtualization** 置备请求的端口号。

10. 在 **Namespace** 字段中，输入您要使用的 **OpenShift Virtualization** 虚拟主机的用户名。

11. 在 **Token** 字段中，为 **HTTP** 和 **HTTPS** 身份验证输入 **bearer** 令牌。
12. 可选：在 **X509 认证颁发机构** 字段中，输入证书来为 **API 服务器调用启用客户端证书身份验证**。

第 13 章 在 RED HAT OPENSTACK PLATFORM 上置备云实例

Red Hat OpenStack Platform 为构建私有或公共基础架构即服务(DSL)云提供构建的基础。它为启用云的工作负载提供了一个高度可扩展的、容错的平台。在 Red Hat Satellite 中，您可以与红帽 OpenStack Platform REST API 交互，以创建云实例并控制其电源管理状态。

先决条件

- 您可以将同步的内容软件仓库用于 Red Hat Enterprise Linux。有关更多信息，[请参阅 内容管理指南中的同步存储库](#)。
- 为主机注册提供激活码。如需更多信息，[请参阅 内容管理指南中的创建激活码](#)。
- 在 OpenStack 环境中管理网络的 Capsule 服务器。如需更多信息，[请参阅置备指南中的配置网络](#)。
- 添加到 OpenStack Image Storage (glance)服务的镜像，以进行基于镜像的调配。有关更多信息，[请参阅 Red Hat OpenStack Platform 实例和镜像指南](#)。

13.1. 将 RED HAT OPENSTACK PLATFORM 连接添加到 SATELLITE 服务器

您可以将 Red Hat OpenStack Platform 添加为 Satellite 中的计算资源。要使用 CLI 而不是 Satellite Web UI，[请参阅 CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 Infrastructure > Compute Resources。
2. 单击 Create Compute Resource。
3. 在 Name 字段中输入新计算资源的名称。
4. 从 Provider 列表中，选择 RHEL OpenStack Platform。

5. 可选：在 **Description** 字段中输入计算资源的描述。
6. 在 **URL** 字段中，在令牌资源（如 `http://openstack.example.com:5000/v2.0/tokens` 或 `http://openstack.example.com:5000/v3/auth/tokens`）输入 OpenStack Authentication keystone 服务的 API 的 URL。
7. 在 **Username** 和 **Password** 字段中，输入 Satellite 的用户身份验证来访问环境。
8. 可选：在 **Project (Tenant)名称** 字段中，输入要管理的 **Satellite Server** 的租户名称 (v2)或项目(v3)。
9. 在 **User domain** 字段中，输入 v3 身份验证的用户域。
10. 在 **Project domain name** 字段中，输入 v3 身份验证的项目域名。
11. 在 **Project domain ID** 字段中，输入 v3 身份验证的项目域 ID。
12. 可选：选择 **Allow external network as main network** 来使用外部网络作为主机的主要网络。
13. 可选：点 **Test Connection** 来验证 **Satellite** 是否可以连接到您的计算资源。
14. 单击**位置**和**机构**选项卡，并验证您要使用的位置和机构是否已设置为您当前的上下文。在这些标签页中添加您想要的额外上下文。
15. 点 **Submit** 保存 Red Hat OpenStack Platform 连接。

CLI 过程

- 要创建计算资源，请输入 `hammer compute-resource create` 命令：

```
# hammer compute-resource create --name "My_OpenStack" \  
--provider "OpenStack" \  

```

```
--description "My OpenStack environment at openstack.example.com" \  
--url "http://openstack.example.com:5000/v3/auth/tokens" \  
--user "My_Username" --password "My_Password" \  
--tenant "My_Openstack" --domain "My_User_Domain" \  
--project-domain-id "My_Project_Domain_ID" \  
--project-domain-name "My_Project_Domain_Name" \  
--locations "New York" --organizations "My_Organization"
```

13.2. 将 RED HAT OPENSTACK PLATFORM 镜像添加到 SATELLITE 服务器中

要使用基于镜像的调配创建主机，您必须将有关镜像的信息（如访问详情和镜像位置）添加到 Satellite 服务器中。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Infrastructure > Compute Resources**，再点 **Red Hat OpenStack Platform** 连接的名称。
2. 点 **Create Image**。
3. 在 **Name** 字段中输入镜像的名称。
4. 从 **Operating System** 列表中，选择镜像的基础操作系统。
5. 从 **Architecture** 列表中，选择操作系统架构。
6. 在 **Username** 字段中，为镜像访问输入 **SSH** 用户名。这通常是 **root** 用户。
7. 在 **Password** 字段中，输入 **SSH** 密码以进行镜像访问。
8. 从 **Image** 列表中选择 **Red Hat OpenStack Platform** 计算资源的镜像。

9. 可选：如果镜像支持用户数据输入，如 cloud-init 数据，请选择 **User Data** 复选框。
10. 点 **Submit** 保存镜像详情。

CLI 过程

- 使用 `hammer compute-resource image create` 命令创建镜像。使用 `--uuid` 字段将镜像位置的完整路径存储在 Red Hat OpenStack Platform 服务器上。

```
# hammer compute-resource image create \
--name "OpenStack Image" \
--compute-resource "My_OpenStack_Platform" \
--operatingsystem "RedHat version" \
--architecture "x86_64" \
--username root \
--user-data true \
--uuid "/path/to/OpenstackImage.qcow2"
```

13.3. 在 COMPUTE 配置文件中添加 RED HAT OPENSTACK PLATFORM 详情

使用这个流程将 Red Hat OpenStack Platform 硬件设置添加到计算配置集中。当您使用此计算配置集在 Red Hat OpenStack Platform 上创建主机时，这些设置会自动填充。

流程

1. 在 Satellite Web UI 中，导航到 **Infrastructure > Compute Profiles**。
2. 在 **Compute Profiles** 窗口中，点现有计算配置集的名称，或者单击 **Create Compute Profile**，输入 **Name**，然后单击 **Submit**。
3. 点 **Red Hat OpenStack Platform** 计算资源的名称。
4. 从 **Flavor** 列表中，选择用于主机的 Red Hat OpenStack Platform 上的硬件配置文件。
5. 从 **Availability zone** 列表中，选择 Red Hat OpenStack Platform 环境中使用的目标集群。

6. 从 **Image** 列表中，选择要用于基于镜像的置备的镜像。
7. 从 **Tenant** 列表中，为 **Red Hat OpenStack Platform** 实例选择租户或项目。
8. 从 **Security Group** 列表中，为端口和 IP 地址选择基于云的访问规则。
9. 从 **内部网络** 中，选择要加入主机的专用网络。
10. 从 **浮动 IP 网络** 中，选择要加入并分配浮动 IP 地址的主机的外部网络。
11. 从 **从卷启动**，选择是否从镜像创建卷。如果没有选择，则实例会直接引导镜像。
12. 在 **New boot volume size (GB)** 字段中，输入新引导卷的大小（以 **GB** 为单位）。
13. 点 **Submit** 保存计算配置集。

CLI 过程

计算配置文件 CLI 命令尚未在 Red Hat Satellite 中实施。另外，您可以在主机创建过程中直接包含相同的设置。

13.4. 在 RED HAT OPENSTACK PLATFORM 上创建基于镜像的主机

在 Satellite 中，您可以使用 Red Hat OpenStack Platform 置备从现有镜像创建主机。新主机条目会触发 Red Hat OpenStack Platform 服务器，以使用预先存在的镜像作为新卷的基础来创建实例。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Hosts > Create Host**。

2. 在 **Name** 字段中输入主机的名称。
3. 点 **Organization and Location** 选项卡，以确保置备上下文自动设置为当前上下文。
4. 从 **Host Group** 列表中，选择您要用来填充表单的主机组。
5. 从 **Deploy on** 列表中，选择 **Red Hat OpenStack Platform** 连接。
6. 从 **Compute Profile** 列表中，选择要用于自动填充虚拟机设置的配置集。
7. 从 **Lifecycle Environment** 列表中，选择环境。
8. 单击 **Interfaces** 选项卡，再单击主机接口上的 **Edit**。
9. 验证这些字段是否已自动填充，特别是以下项目：
 - **Host** 选项卡中的 **Name** 成为 DNS 名称。
 - **MAC 地址** 字段为空。**Red Hat OpenStack Platform** 在调配过程中为主机分配一个 **MAC 地址**。
 - **Satellite** 服务器自动为新主机分配 IP 地址。
 - 为主机上的第一个接口自动选择 **Managed**、**Primary** 和 **Provision** 选项。如果没有，请选择它们。
10. 单击 **Operating System** 选项卡，并确认所有字段都自动包含值。
11. 如果要更改从计算配置集自动填充的镜像，请从 **Images** 列表中选择一个新的镜像来基础新主机的根卷。

12. 点 **Provisioning** 模板中的 **Resolve**，以检查新主机可以识别要使用的正确置备模板。
13. 单击 **Virtual Machine** 选项卡，并确认这些设置填充了主机组和计算配置集的详情。修改这些设置以符合您的需要。
14. 单击 **Parameters** 选项卡，并确保存在一个提供激活码的参数。如果没有，请添加激活码。
15. 点 **Submit** 保存主机条目。

CLI 过程

- 使用 **hammer host create** 命令创建主机，并包含 **--provision-method** 镜像。将以下示例中的值替换为适合您的环境的值。

```
# hammer host create \  
--compute-  
attributes="flavor_ref=m1.small,tenant_id=openstack,security_groups=default,network=mynetwork" \  
--compute-resource "My_OpenStack_Platform" \  
--enabled true \  
--hostgroup "Base" \  
--image "OpenStack Image" \  
--interface "managed=true,primary=true,provision=true" \  
--location "New York" \  
--managed true \  
--name "openstack-host1" \  
--organization "My_Organization" \  
--provision-method image
```

有关此计算资源的其他主机创建参数的更多信息，请输入 **hammer host create --help** 命令。

第 14 章 在 AMAZON EC2 中置备云实例

Amazon Elastic Compute Cloud (Amazon EC2)是一个提供公有云计算资源的 Web 服务。使用 Satellite，您可以与 Amazon EC2 的公共 API 交互，以创建云实例并控制其电源管理状态。使用本章中的步骤添加与 Amazon EC2 帐户的连接并调配云实例。

14.1. AMAZON EC2 置备的先决条件

Amazon EC2 置备的要求包括：

- 在 EC2 环境中管理网络的 Capsule 服务器。使用虚拟私有云(VPC)确保主机和 Capsule 服务器之间的安全网络。
- 用于基于镜像的置备的 Amazon Machine Image (AMI)。
- 您可以将同步的内容软件仓库用于 Red Hat Enterprise Linux。有关更多信息，[请参阅 内容管理指南中的同步存储库](#)。
- 为主机注册提供激活码。如需更多信息，[请参阅 内容管理指南中的创建激活码](#)。

14.2. 安装 AMAZON EC2 插件

安装 Amazon EC2 插件，以将 EC2 计算资源提供程序附加到 Satellite。这可让您管理和将主机部署到 EC2。

流程

1. 在 Satellite 服务器上安装 EC2 计算资源供应商：

```
# satellite-installer --enable-foreman-compute-ec2
```

2. 可选：在 Satellite Web UI 中，导航到 **Administer > About** 并选择 **compute resources** 选项卡来验证 Amazon EC2 插件的安装。

14.3. 向 SATELLITE 服务器添加 AMAZON EC2 连接

使用这个流程在 **Satellite** 服务器的计算资源中添加 **Amazon EC2** 连接。要使用 CLI 而不是 **Satellite Web UI**，请参阅 [CLI 过程](#)。

前提条件

- 执行此流程的 **AWS EC2** 用户需要 **AmazonEC2FullAccess** 权限。您可以从 **AWS** 附加这些权限。

时间设置和 Amazon Web Services

Amazon Web Services 使用时间设置作为身份验证过程的一部分。确保 **Satellite** 服务器的时间正确同步。确保 **NTP** 服务（如 **ntpd** 或 **chronyd**）在 **Satellite** 服务器上正常运行。无法为 **Amazon Web Services** 提供正确时间可能会导致身份验证失败。

有关在 **Satellite** 中同步时间的更多信息，请参阅在 [连接的网络环境中安装 Satellite 服务器](#) 中的同步时间。

流程

1. 在 **Satellite Web UI** 中，导航到 **Infrastructure > Compute Resources**，并在 **Compute Resources** 窗口中点 **Create Compute Resource**。
2. 在 **Name** 字段中输入用于标识 **Amazon EC2** 计算资源的名称。
3. 从 **Provider** 列表中，选择 **EC2**。
4. 在 **Description** 字段中，输入有助于区分资源以备将来使用的信息。
5. 可选：从 **HTTP 代理** 列表中，选择一个 **HTTP** 代理来连接到外部 **API** 服务。您必须将 **HTTP** 代理添加到 **Satellite**，然后才能从此列表中选择代理。更多信息请参阅 [第 14.4 节“使用带有计算资源的 HTTP 代理”](#)。
6. 在 **Access Key** 和 **Secret Key** 字段中，输入 **Amazon EC2** 帐户的访问密钥。如需更多信息，请参阅在 [Amazon 文档网站上为您的 AWS 帐户管理访问密钥](#)。

7. 可选：点 **Load Regions** 按钮填充 **Regions** 列表。
8. 从 **Region** 列表中，选择要使用的 **Amazon EC2** 区域或数据中心。
9. 单击 **Locations** 选项卡，并确保选中您要使用的位置，或添加不同的位置。
10. 单击 **Organizations** 选项卡，并确保选择要使用的机构，或添加不同的组织。
11. 点 **Submit** 保存 **Amazon EC2** 连接。
12. 选择新的计算资源，然后单击 **SSH 密钥** 选项卡，然后单击 **Download** 以保存用于 **SSH** 身份验证的 **SSH** 密钥副本。在 [BZ1793138](#) 解决前，您只能在创建 **Amazon EC2** 计算资源后立即下载 **SSH** 密钥的副本。如果您在后续阶段需要 **SSH** 密钥，请按照 [第 14.9 节](#) “使用 **SSH** 连接到 **Amazon EC2** 实例” 中的步骤操作。

CLI 过程

- 使用 `hammer compute-resource create` 命令创建连接。使用 `--user` 和 `--password` 选项分别添加访问密钥和密钥。

```
# hammer compute-resource create \
--description "Amazon EC2 Public Cloud" \
--locations "My_Location" \
--name "My_EC2_Compute_Resource" \
--organizations "My_Organization" \
--password "My_Secret_Key" \
--provider "EC2" \
--region "My_Region" \
--user "My_User_Name"
```

14.4. 使用带有计算资源的 HTTP 代理

在某些情况下，您使用的 **EC2** 计算资源可能需要特定的 **HTTP** 代理与 **Satellite** 通信。在 **Satellite** 中，您可以创建一个 **HTTP** 代理，然后将 **HTTP** 代理分配给 **EC2** 计算资源。

但是，如果您在 **Administer > Settings** 中为 **Satellite** 配置 **HTTP** 代理，并为计算资源添加另一个 **HTTP** 代理，则您在 **Administer > Settings** 中定义的 **HTTP** 代理将具有优先权。

流程

1. 在 Satellite Web UI 中，导航到 **Infrastructure > HTTP Proxies**，然后选择 **New HTTP Proxy**。
2. 在 **Name** 字段中输入 HTTP 代理的名称。
3. 在 **URL** 字段中输入 HTTP 代理的 URL，包括端口号。
4. 可选：如果您的 HTTP 代理需要身份验证，请输入用户名和密码来向 HTTP 代理进行身份验证。
5. 单击 **Test Connection**，以确保您可以从 Satellite 连接到 HTTP 代理。
6. 单击 **Locations** 选项卡，再添加一个位置。
7. 点 **Organization** 选项卡并添加机构。
8. 点 **Submit**。

14.5. 为 AMAZON EC2 创建镜像

您可以从 Satellite 中为 Amazon EC2 创建镜像。

流程

1. 在 Satellite Web UI 中，导航到 **Infrastructure > Compute Resources**。
2. 选择 **Amazon EC2** 供应商。
3. 点 **Create Image**。

- 在 **Name** 字段中输入 EC2 镜像的有意义的且唯一的名称。
- 从 **Operating System** 列表中，选择要与镜像关联的操作系统。
- 从 **Architecture** 列表中，选择要与镜像关联的架构。
- 在 **Username** 字段中输入 SSH 到机器所需的用户名。
- 在 **Image ID** 字段中，输入 Amazon 提供的镜像 ID 或操作系统厂商。
- 可选：选择 **User Data** 复选框来启用对用户数据输入的支持。
- 可选：设置在创建此镜像时要使用的 *Fog* 角色的 *iam* 角色。
- 单击 **Submit**，将您的更改保存到 **Satellite**。

14.6. 将 AMAZON EC2 镜像添加到 SATELLITE 服务器中

Amazon EC2 使用基于镜像的置备来创建主机。您必须在 **Satellite** 服务器中添加镜像详情。这包括访问详情和镜像位置。

要使用 CLI 而不是 **Satellite Web UI**，请参阅 [CLI 过程](#)。

流程

1. 在 **Satellite Web UI** 中，导航到 **Infrastructure > Compute Resources** 并选择 **Amazon EC2** 连接。
2. 单击 **Images** 选项卡，然后单击 **Create Image**。

3. 在 **Name** 字段中输入名称来标识镜像以供以后使用。
4. 从 **Operating System** 列表中，选择与您要添加的镜像对应的操作系统。
5. 从 **Architecture** 列表中，选择操作系统的架构。
6. 在 **Username** 字段中，为镜像访问输入 **SSH** 用户名。这通常是 **root** 用户。
7. 在 **Password** 字段中，输入 **SSH** 密码以进行镜像访问。
8. 在 **Image ID** 字段中，为镜像输入 **Amazon Machine Image (AMI) ID**。这通常采用以下格式：**ami-xxxxxxx**。
9. 可选：如果镜像支持用户数据输入，如 **cloud-init** 数据，请选择 **User Data** 复选框。如果启用用户数据，则自动禁用 **Finish** 脚本。这也适用于反向：如果您启用 **Finish** 脚本，这将禁用用户数据。
10. 可选：在 **IAM 角色** 字段中，输入用于创建镜像的 **Amazon** 安全角色。
11. 点 **Submit** 保存镜像详情。

CLI 过程

- 使用 **hammer compute-resource image create** 命令创建镜像。使用 **--uuid** 字段将镜像位置的完整路径存储在 **Amazon EC2** 服务器上。

```
# hammer compute-resource image create \  
--architecture "My_Architecture" \  
--compute-resource "My_EC2_Compute_Resource" \  
--name "My_Amazon_EC2_Image" \  
--operatingsystem "My_Operating_System" \  
--user-data true \  
--username root \  
--uuid "ami-My_AMI_ID"
```

14.7. 在 COMPUTE 配置文件中添加 AMAZON EC2 详情

您可以将 Amazon EC2 上实例的硬件设置添加到计算配置集。

流程

1. 在 Satellite Web UI 中，导航到 **Infrastructure > Compute Profiles**，再点击您的配置集的名称，然后点 **EC2 连接**。
2. 从 **Flavor** 列表中，选择用于主机的 EC2 上的硬件配置文件。
3. 从 **Image** 列表中，选择要用于基于镜像的置备的镜像。
4. 从 **Availability zone** 列表中，选择要在所选 EC2 区域中使用的目标集群。
5. 从 **子网** 列表中，为 EC2 实例添加子网。如果您有 VPC 来置备新主机，请使用其子网。
6. 从 **Security Groups** 列表中，为要应用到主机的端口和 IP 地址选择基于云的访问规则。
7. 从 **Managed IP** 列表中，选择 **公共 IP** 或 **私有 IP**。
8. 点 **Submit** 保存计算配置集。

CLI 过程

计算配置文件 CLI 命令尚未在 Red Hat Satellite 中实施。另外，您可以在主机创建过程中直接包含相同的设置。

14.8. 在 AMAZON EC2 上创建基于镜像的主机

Amazon EC2 置备过程从 Amazon EC2 服务器上的现有镜像创建主机。要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Hosts > Create Host**。
2. 在 **Name** 字段中输入主机的名称。
3. 从 **Host Group** 列表中，您可以选择一个主机组来填充大多数新主机的字段。
4. 从 **Deploy on** 列表中，选择 **EC2** 连接。
5. 从 **Compute Profile** 列表中，选择要用于自动填充基于虚拟机的设置的配置集。
6. 点 **Interface** 选项卡，然后点主机接口上的 **Edit**，并验证该字段是否填充了值。将 **Mac Address** 字段留空。Satellite 服务器为主机上的第一个接口自动选择和 **IP 地址** 和 **Managed**、**Primary** 和 **Provision** 选项。
7. 点 **Operating System** 选项卡，并确认所有字段都填充了值。
8. 点 **Virtual Machine** 选项卡，并确认所有字段都填充了值。
9. 单击 **Parameters** 选项卡，并确保存在一个提供激活码的参数。如果没有，请添加激活码。
10. 点 **Submit** 保存您的更改。

此新主机条目会触发 Amazon EC2 服务器来创建实例，使用预先存在的镜像作为新卷的基础。

CLI 过程

- 使用 `hammer host create` 命令创建主机，并包含 `--provision-method` 镜像 以使用基于镜像的置备。

```
# hammer host create \  
--compute-attributes="flavor_id=m1.small,image_id=TestImage,availability_zones=us-east-
```

```

1a,security_group_ids=Default,managed_ip=Public" \
--compute-resource "My_EC2_Compute_Resource" \
--enabled true \
--hostgroup "My_Host_Group" \
--image "My_Amazon_EC2_Image" \
--interface "managed=true,primary=true,provision=true,subnet_id=EC2" \
--location "My_Location" \
--managed true \
--name "My_Amazon_EC2_Host_" \
--organization "My_Organization" \
--provision-method image

```

有关此计算资源的其他主机创建参数的更多信息，请输入 `hammer host create --help` 命令。

14.9. 使用 SSH 连接到 AMAZON EC2 实例

您可以使用 SSH 从 Satellite 服务器远程连接到 Amazon EC2 实例。但是，要连接到您通过 Red Hat Satellite 置备的任何 Amazon Web Services EC2 实例，您必须首先访问与 Foreman 数据库中计算资源关联的私钥，并使用此密钥进行身份验证。

流程

1. 要找到计算资源列表，请在 Satellite 服务器基础系统上输入以下命令，并记录您要使用的计算资源的 ID：

```
# hammer compute-resource list
```

2. 将用户切换到 postgres 用户：

```
# su - postgres
```

3. 启动 postgres shell：

```
$ psql
```

4. 以 postgres 用户身份连接到 Foreman 数据库：

```
# postgres=# \c foreman
```

5. 从 `compute_resource_id = 3` 的 `key_pairs` 中选择 `secret` :

```
# select secret from key_pairs where compute_resource_id = 3; secret
```

6. 在 `-----BEGIN RSA PRIVATE KEY-----` 后复制密钥，直到 `-----END RSA PRIVATE KEY-----`。

7. 创建一个 `.pem` 文件，并将您的密钥粘贴到文件中：

```
# vim Keyname.pem
```

8. 确保限制对 `.pem` 文件的访问：

```
# chmod 600 Keyname.pem
```

9. 要连接到 Amazon EC2 实例，请输入以下命令：

```
ssh -i Keyname.pem ec2-user@example.aws.com
```

14.10. 为 AMAZON WEB SERVICE EC2 环境配置完成模板

您可以在 Amazon EC2 环境中置备 Red Hat Enterprise Linux 实例的过程中使用 Red Hat Satellite 完成模板。

如果要使用 Finish 模板与 SSH 搭配使用，Satellite 必须驻留在 EC2 环境和正确安全组中。Satellite 目前直接执行 SSH 完成置备，而不使用 Capsule 服务器。如果 Satellite 服务器不在 EC2 中，EC2 虚拟机将报告内部 IP，而不是访问它所需的外部 IP。

流程

1. 在 Satellite Web UI 中，导航到 `Hosts > Provisioning Templates`。
2. 在 `Provisioning Templates` 页面中，在搜索字段中输入 `Kickstart 默认完成`，然后单击 `Search`。

3. 在 Kickstart 默认完成 模板中，选择 Clone。
4. 在 Name 字段中输入模板的唯一名称。
5. 在模板中，使用 `sudo` 为需要 root 权限的每个命令添加前缀，但 `subscription-manager register` 和 `yum` 命令除外，或者添加以下行以 `sudo` 用户身份运行整个模板：

```
sudo -s << EOS
__Template__Body__
EOS
```

6. 点 关联 选项卡，并将模板与您要使用的 Red Hat Enterprise Linux 操作系统关联。
7. 单击 Locations 选项卡，再添加主机所在的位置。
8. 单击 Organizations 选项卡，再添加主机所属的组织。
9. 进行任何所需的其他自定义或更改，然后单击 Submit 以保存模板。
10. 在 Satellite Web UI 中，导航到 Hosts > Operating systems，再选择您要用于主机的操作系统。
11. 点 Templates 选项卡，从 Finish Template 列表中选择您的完成模板。
12. 在 Satellite Web UI 中，导航到 Hosts > Create Host，并输入您要创建的主机的信息。
13. 单击 Parameters 选项卡，再导航到 Host parameters。
14. 在主机 参数 中，单击 Add Parameter 按钮三次，以添加三个新参数字段。添加以下三个参数：
 - a. 在 Name 字段中输入 `remote_execution_ssh_keys`。在对应的 Value 字段中，输入

`cat /usr/share/foreman-proxy/.ssh/id_rsa_foreman_proxy.pub` 的输出。

- b. 在 Name 字段中输入 `remote_execution_ssh_user`。在对应的 Value 字段中，输入 `ec2-user`。
 - c. 在 Name 字段中输入 `activation_keys`。在对应的 Value 字段中，输入您的激活码。
15. 点 **Submit** 保存更改。

14.11. 删除 AMAZON EC2 上的虚拟机

您可以从 Satellite 中删除在 Amazon EC2 上运行的虚拟机。

流程

1. 在 Satellite Web UI 中，导航到 **Infrastructure > Compute Resources**。
2. 选择 **Amazon EC2 供应商**。
3. 在 **Virtual Machines** 选项卡中，从 **Actions** 菜单中点 **Delete**。这会从 Amazon EC2 计算资源中删除虚拟机，同时保留 Satellite 中的任何关联的主机。如果要删除孤立的主机，请导航到 **Hosts > All Hosts** 并手动删除主机。

14.12. 卸载 AMAZON EC2 插件

如果您之前已安装了 Amazon EC2 插件，但没有使用它来管理和部署主机到 EC2，您可以从 Satellite 服务器中卸载它。

流程

1. 从 Satellite 服务器卸载 EC2 计算资源供应商：

```
# yum remove -y foreman-ec2
# satellite-installer --no-enable-foreman-compute-ec2
```

2.

可选：在 Satellite Web UI 中，导航到 **Administer > About** 并选择 **Available Providers** 选项卡来验证 Amazon EC2 插件的删除。

14.13. 有关 AMAZON WEB 服务和 SATELLITE 的更多信息

有关如何在 Amazon Web Services EC2 上找到红帽黄金镜像的详情，请参考 [如何在 AWS EC2 上撤销红帽云访问黄金镜像](#)。

有关如何在 Linux 上安装和使用 Amazon Web Service 客户端的详情，请参考 Amazon Web Services 文档中的 [在 Linux 上安装 AWS 命令行界面](#)。

有关在 Amazon Web Services 中导入和导出虚拟机的详情，请参考 Amazon Web Services 文档中的 [VM Import/Export](#)。

第 15 章 在 GOOGLE COMPUTE ENGINE 上置备云实例

Red Hat Satellite 可以与 Google Compute Engine (GCE)交互，包括创建新虚拟机并控制其电源管理状态。您只能使用带有 Satellite 支持的金级镜像来创建 GCE 主机。

先决条件

- 您可以将同步的内容软件仓库用于 Red Hat Enterprise Linux。有关更多信息，[请参阅 内容管理指南中的同步存储库](#)。
- 为主机注册提供激活码。如需更多信息，[请参阅 内容管理指南中的创建激活码](#)。
- 在 GCE 项目中，使用必要的 IAM Compute 角色配置服务帐户。如需更多信息，[请参阅 GCE 文档中的 Compute Engine IAM 角色](#)。
- 在 GCE 项目中，将 enable-oslogin 设置为 FALSE。如需更多信息，[请参阅 GCE 文档中的启用或禁用操作系统登录](#)。
- 可选：如果要与 Puppet 与 GCE 主机搭配使用，请导航到 **Administer > Settings > Puppet**，并为证书设置启用 Use UUID，将 Puppet 配置为使用一致的 Puppet 证书 ID。
- 根据您的需要，将 `完成` 或 `user_data` 置备模板与您要使用的操作系统相关联。如需有关置备模板的更多信息，[请参阅 置备主机 中的 置备 模板](#)。

15.1. 安装 GOOGLE GCE 插件

安装 Google GCE 插件，将 GCE 计算资源供应商附加到 Satellite。这可让您管理和将主机部署到 GCE。

流程

1. 在 Satellite 服务器上安装 Google GCE 计算资源供应商：

```
# satellite-installer --enable-foreman-compute-gce
```

- 2.

可选：在 **Satellite Web UI** 中，导航到 **Administer > About** 并选择 **Compute Resources** 选项卡来验证 **Google GCE** 插件的安装。

15.2. 向 SATELLITE 服务器添加 GOOGLE GCE 连接

在 **Satellite** 中将 **Google Compute Engine (GCE)** 添加为计算资源。要使用 **CLI** 而不是 **Satellite Web UI**，请参阅 [CLI 过程](#)。

流程

1. 在 **Google GCE** 中，以 **JSON** 格式生成服务帐户密钥。

2. 将该文件从本地机器复制到 **Satellite** 服务器：

```
# scp gce_key.json root@satellite.example.com:/usr/share/foreman/gce_key.json
```

3. 在 **Satellite** 服务器上，将服务帐户密钥的所有者改为 **foreman** 用户：

```
# chown foreman /usr/share/foreman/gce_key.json
```

4. 在 **Satellite** 服务器上，为您的服务帐户密钥配置权限，以确保该文件可读：

```
# chmod 0600 /usr/share/foreman/gce_key.json
```

5. 在 **Satellite** 服务器上，为您的服务帐户密钥恢复 **SELinux** 上下文：

```
# restorecon -vv /usr/share/foreman/gce_key.json
```

6. 在 **Satellite Web UI** 中，导航到 **Infrastructure > Compute Resources**，再点 **Create Compute Resource**。

7. 在 **Name** 字段中输入计算资源的名称。

8. 从 **Provider** 列表中，选择 **Google**。

9. 可选：在 **Description** 字段中输入资源的描述。
10. 在 **Google Project ID** 字段中，输入项目 ID。
11. 在 **Client Email** 字段中，输入客户端电子邮件。
12. 在 **Certificate Path** 字段中输入服务帐户密钥的路径。例如：
`/usr/share/foreman/gce_key.json`。
13. 点 **Load Zones**，从 GCE 环境中填充区域列表。
14. 从 **Zone** 列表中，选择要使用的 GCE 区域。
15. 点 **Submit**。

CLI 过程

1. 在 Google GCE 中，以 JSON 格式生成服务帐户密钥。
2. 将该文件从本地机器复制到 Satellite 服务器：

```
# scp gce_key.json root@satellite.example.com:/usr/share/foreman/gce_key.json
```

3. 在 Satellite 服务器上，将服务帐户密钥的所有者改为 **foreman** 用户：

```
# chown foreman /usr/share/foreman/gce_key.json
```

4. 在 Satellite 服务器上，为您的服务帐户密钥配置权限，以确保该文件可读：

```
# chmod 0600 /usr/share/foreman/gce_key.json
```

5. 在 **Satellite** 服务器上，为您的服务帐户密钥恢复 **SELinux** 上下文：

```
# restorecon -vv /usr/share/foreman/gce_key.json
```

6. 使用 **hammer compute-resource create** 命令将 **GCE** 计算资源添加到 **Satellite** 中：

```
# hammer compute-resource create \  
--email "My_GCE_Email" \  
--key-path "Path_To_My_GCE_Key.json" \  
--name "My_GCE_Compute_Resource" \  
--project "My_GCE_Project_ID" \  
--provider "gce" \  
--zone "My_Zone"
```

15.3. 将 GOOGLE COMPUTE ENGINE 镜像添加到 SATELLITE 服务器中

要使用基于镜像的调配创建主机，您必须将有关镜像的信息（如访问详情和镜像位置）添加到 **Satellite** 服务器中。

要使用 **CLI** 而不是 **Satellite Web UI**，请参阅 [CLI 过程](#)。

流程

1. 在 **Satellite Web UI** 中，导航到 **Infrastructure > Compute Resources**，然后单击 **Google Compute Engine** 连接的名称。
2. 点 **Create Image**。
3. 在 **Name** 字段中输入镜像的名称。
4. 从 **Operating System** 列表中，选择镜像的基础操作系统。
5. 从 **Architecture** 列表中，选择操作系统架构。
6. 在 **Username** 字段中，为镜像访问输入 **SSH** 用户名。指定 **root** 以外的用户，因为 **root** 用户无法使用 **SSH** 密钥连接到 **GCE** 实例。用户名必须以字母开头，由小写字母和数字组成。

7. 从 Image 列表中，从 Google Compute Engine 计算资源中选择一个镜像。
8. 可选：如果镜像支持用户数据输入，如 cloud-init 数据，请选择 **User Data** 复选框。
9. 点 **Submit** 保存镜像详情。

CLI 过程

- 使用 `hammer compute-resource image create` 命令创建镜像。使用 `--username` 选项，指定 `root` 以外的用户，因为 `root` 用户无法使用 SSH 密钥连接到 GCE 实例。用户名必须以字母开头，由小写字母和数字组成。

```
# hammer compute-resource image create \
--name 'gce_image_name' \
--compute-resource 'gce_cr' \
--operatingsystem-id 1 \
--architecture-id 1 \
--uuid '3780108136525169178' \
--username 'admin'
```

15.4. 在 COMPUTE 配置文件中添加 GOOGLE GCE DETAILS

使用这个流程将 Google GCE 硬件设置添加到计算配置集中。当您使用这个计算配置集在 Google GCE 上创建主机时，这些设置会自动填充。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Infrastructure > Compute Profiles**。
2. 在 **Compute Profiles** 窗口中，点现有计算配置集的名称，或者单击 **Create Compute Profile**，输入 **Name**，然后单击 **Submit**。
3. 点 **GCE** 计算资源的名称。

4. 从 **Machine Type** 列表中，选择用于置备的机器类型。
5. 从 **Image** 列表中，选择要用于置备的镜像。
6. 从 **Network** 列表中，选择要用于置备的 **Google GCE** 网络。
7. 可选：选择 **关联临时外部 IP** 复选框，以分配 **Satellite** 用于与主机通信的动态临时 **IP** 地址。重启主机时，此公共 **IP** 地址会改变。如果您需要永久 **IP** 地址，请在 **Google GCE** 上保留静态公共 **IP** 地址，并将其附加到主机。
8. 在 **Size (GB)** 字段中，输入要在主机上创建的存储大小。
9. 点 **Submit** 保存计算配置集。

CLI 过程

1. 创建用于 **Google GCE** 计算资源的计算配置集：

```
# hammer compute-profile create --name My_GCE_Compute_Profile
```

2. 在 **compute** 配置集中添加 **GCE** 详情：

```
# hammer compute-profile values create \  
--compute-attributes "machine_type=f1-micro,associate_external_ip=true,network=default" \  
--compute-profile "My_GCE_Compute_Profile" \  
--compute-resource "My_GCE_Compute_Resource" \  
--volume "size_gb=20"
```

15.5. 在 GOOGLE COMPUTE ENGINE 上创建基于镜像的主机

在 **Satellite** 中，您可以使用 **Google Compute Engine** 置备从现有镜像创建主机。新主机条目会触发 **Google Compute Engine** 服务器，以使用预先存在的镜像作为新卷的基础来创建实例。

要使用 **CLI** 而不是 **Satellite Web UI**，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Hosts > Create Host**。
2. 在 **Name** 字段中输入主机的名称。
3. 点 **Organization and Location** 选项卡，以确保置备上下文自动设置为当前上下文。
4. 从 **Host Group** 列表中，选择您要用来填充表单的主机组。
5. 从 **Deploy on** 列表中，选择 **Google Compute Engine** 连接。
6. 从 **Compute Profile** 列表中，选择要用于自动填充虚拟机设置的配置集。
7. 从 **Lifecycle Environment** 列表中，选择环境。
8. 单击 **Interfaces** 选项卡，再单击主机接口上的 **Edit**。
9. 验证这些字段是否已自动填充，特别是以下项目：
 - **Host** 选项卡中的 **Name** 成为 DNS 名称。
 - **MAC 地址** 字段为空。Google Compute Engine 在置备过程中为主机分配一个 MAC 地址。
 - **Satellite** 服务器自动为新主机分配 IP 地址。
 - **Domain** 字段填充所需的域。
 - 为主机上的第一个接口自动选择 **Managed**、**Primary** 和 **Provision** 选项。如果没有，

请选择它们。

10. 单击 **Operating System** 选项卡，并确认所有字段都自动包含值。
11. 点 **Provisioning** 模板中的 **Resolve**，以检查新主机可以识别要使用的正确置备模板。
12. 单击 **Virtual Machine** 选项卡，并确认这些设置填充了主机组和计算配置集的详情。修改这些设置以符合您的需要。
13. 单击 **Parameters** 选项卡，并确保存在一个提供激活码的参数。如果没有，请添加激活码。
14. 点 **Submit** 保存主机条目。

CLI 过程

- 使用 `hammer host create` 命令创建主机，并包含 `--provision-method` 镜像。将以下示例中的值替换为适合您的环境的值。

```
# hammer host create \
--architecture x86_64 \
--compute-profile "gce_profile_name" \
--compute-resource "My_GCE_Compute_Resource" \
--image "My_GCE_Image" \
--interface "type=interface,domain_id=1,managed=true,primary=true,provision=true" \
--location "My_Location" \
--name "GCE_VM" \
--operatingsystem "My_Operating_System" \
--organization "My_Organization" \
--provision-method 'image' \
--puppet-ca-proxy-id 1 \
--puppet-environment-id 1 \
--puppet-proxy-id 1 \
--root-password "My_Root_Password"
```

有关此计算资源的其他主机创建参数的更多信息，请输入 `hammer host create --help` 命令。

15.6. 删除 GOOGLE GCE 上的虚拟机

您可以删除在 Satellite Server 上的 Google GCE 上运行的虚拟机。

流程

1. 在 Satellite Web UI 中，导航到 **Infrastructure > Compute Resources**。
2. 选择 **Google GCE 供应商**。
3. 在 **Virtual Machines** 选项卡中，从 **Actions** 菜单中点 **Delete**。这会从 Google GCE 计算资源中删除虚拟机，同时保留 Satellite 中的任何关联的主机。如果要删除孤立的主机，请导航到 **Hosts > All Hosts** 并手动删除主机。

15.7. UNINSTALLING GOOGLE GCE PLUGIN

如果您之前已安装了 Google GCE 插件，但没有再使用它来将主机部署到 GCE，您可以从 Satellite 服务器中卸载它。

流程

1. 从 Satellite 服务器卸载 GCE 计算资源供应商：

```
# yum remove -y foreman-gce
# satellite-installer --no-enable-foreman-compute-gce
```

2. 可选：在 Satellite Web UI 中，导航到 **Administer > About** 并选择 **Available Providers** 选项卡来验证 Google GCE 插件的删除。

第 16 章 在 MICROSOFT AZURE RESOURCE MANAGER 上置备云实例

Red Hat Satellite 可以与 Microsoft Azure Resource Manager 交互，包括创建新虚拟机并控制其电源管理状态。创建 Azure 主机只支持基于镜像的置备。这包括使用 Marketplace 镜像、自定义镜像和共享镜像 gallery 置备。

如需有关 Azure Resource Manager 概念的更多信息，请参阅 [Azure Resource Manager 文档](#)。

先决条件

- 您可以将同步的内容软件仓库用于 Red Hat Enterprise Linux。有关更多信息，请参阅 [内容管理指南中的同步存储库](#)。
- 为主机注册提供激活码。如需更多信息，请参阅 [内容管理指南中的创建激活码](#)。
- 确保具有创建 Azure Active Directory 应用程序的正确权限。如需更多信息，请参阅 [Microsoft 身份平台\(Azure Active Directory for developers\)文档中的检查 Azure AD 权限](#)。
- 您必须创建和配置 Azure Active Directory 应用程序和服务原则，以获取应用程序或 客户端 ID、目录 或租户 ID 和客户端 Secret。如需更多信息，请参阅使用 [门户创建 Azure AD 应用程序和服务主体](#)，该应用程序和服务主体可以访问 [Microsoft 身份平台\(Azure Active Directory for developers\) 文档中的资源](#)。
- 可选：如果要与 Puppet 与 Azure 主机搭配使用，请导航到 **Administer > Settings > Puppet**，并为证书设置启用 Use UUID，将 Puppet 配置为使用一致的 Puppet 证书 ID。
- 根据您的需要，将 `完成` 或 `user_data` 置备模板与您要使用的操作系统相关联。如需有关置备模板的更多信息，请参阅 [置备模板](#)。
- 可选：如果您希望虚拟机使用静态专用 IP 地址，请在 Satellite 中创建一个与 Azure 子网地址匹配的 Network Address 字段的子网。
- 在创建 RHEL BYOS 镜像前，您必须接受 Azure CLI 或门户中的镜像条款，以便镜像可用于创建和管理订阅的虚拟机。

16.1. 向 SATELLITE 服务器添加 MICROSOFT AZURE RESOURCE MANAGER 连接

使用这个流程在 Satellite 中将 Microsoft Azure 添加为计算资源。请注意，您必须为您要使用的每个 Microsoft Azure 区域添加单独的计算资源。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Infrastructure > Compute Resources**，再点 **Create Compute Resource**。
2. 在 **Name** 字段中输入计算资源的名称。
3. 从 **Provider** 列表中，选择 **Azure Resource Manager**。
4. 可选：在 **Description** 字段中输入资源的描述。
5. 默认情况下，**Cloud** 设置为 **Public/Standard**。Azure Government Cloud 支持以下区域：
 - 美国政府
 - 中国
 - 法国
6. 在 **Client ID** 字段中输入应用程序或 *客户端 ID*。
7. 在 **Client Secret** 字段中，输入您的客户端 **secret**。
8. 在 **Subscription ID** 字段中输入您的订阅 **ID**。

9. 在 **Tenant ID** 字段中，输入您的 **Directory** 或 **租户 ID**。
10. 单击 **Load Regions**。此测试您是否与 **Azure Resource Manager** 的连接成功，并载入您的订阅中可用的区域。
11. 从 **Azure Region** 列表中，选择要使用的 **Azure** 区域。
12. 点 **Submit**。

CLI 过程

- 使用 **hammer compute-resource create** 将 **Azure** 计算资源添加到 **Satellite**。

```
# hammer compute-resource create \
--app-ident My_Client_ID \
--name My_Compute_Resource_Name \
--provider azurearm \
--region "My_Region" \
--secret-key My_Client_Secret \
--sub-id My_Subscription_ID \
--tenant My_Tenant_ID
```

请注意，**-- region** 选项的值必须采用小写，且不得包含特殊字符。



重要

如果使用 **Azure Government Cloud**，则必须传递 **--cloud** 参数。cloud 参数的值有：

Azure Government Cloud 的名称	hammer --cloud 的值
美国政府	azureusgovernment
中国	azurechina
法国	azuregermancloud

16.2. 将 MICROSOFT AZURE RESOURCE MANAGER 镜像添加到 SATELLITE 服务器中

要使用基于镜像的调配创建主机，您必须将有关镜像的信息（如访问详情和镜像位置）添加到 Satellite 服务器中。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Infrastructure > Compute Resources**，然后单击 **Microsoft Azure Resource Manager** 连接的名称。
2. 点 **Create Image**。
3. 在 **Name** 字段中输入镜像的名称。
4. 从 **Operating System** 列表中，选择镜像的基础操作系统。
5. 从 **Architecture** 列表中，选择操作系统架构。
6. 在 **Username** 字段中，为镜像访问输入 SSH 用户名。您不能使用 **root** 用户。
7. 可选：在 **Password** 字段中，输入要进行身份验证的密码。
8. 在 **Azure Image Name** 字段中，输入格式为 *prefix://UUID* 的镜像名称。
 - 对于自定义镜像，请使用前缀 **custom**。例如：**custom://image-name**。
 - 对于共享 **gallery** 镜像，请使用前缀 **gallery**。例如，**gallery://image-name**。
 - 对于公共和 RHEL 自带您自己的订阅(BYOS)镜像，请使用前缀 **marketplace**。例如，**marketplace://OpenLogicCentOS:7.5:latest**。

如需更多信息，请参阅使用 [Azure CLI 在 Azure Marketplace 中查找 Linux 虚拟机镜像](#)。

9. 可选：如果镜像支持用户数据输入，如 cloud-init 数据，请选择 **User Data** 复选框。
10. 点 **Submit** 保存镜像详情。

CLI 过程

-

使用 `hammer compute-resource image create` 命令创建镜像。请注意，您为镜像输入的用户名必须与使用此镜像创建主机时使用的相同。在创建镜像时，`--password` 选项是可选的。您不能使用 `root` 用户。

```
# hammer compute-resource image create \
--name Azure_image_name \
--compute-resource azure_cr_name \
--uuid 'marketplace://RedHat:RHEL:7-RAW:latest' \
--username 'azure_username' \
--user-data no
```

16.3. 在 COMPUTE 配置集中添加 MICROSOFT AZURE RESOURCE MANAGER 详情

使用这个流程将 Microsoft Azure 硬件设置添加到计算配置集中。当您使用这个计算配置集在 Microsoft Azure 上创建主机时，这些设置会自动填充。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Infrastructure > Compute Profiles**。
2. 在 **Compute Profiles** 窗口中，点现有计算配置集的名称，或者单击 **Create Compute Profile**，输入 **Name**，然后单击 **Submit**。
3. 点 **Azure** 计算资源的名称。

4. 从 **Resource group** 列表中，选择要置备的资源组。
5. 从 **VM Size** 列表中选择要置备的虚拟机的大小。
6. 从 **Platform** 列表中选择 **Linux**。
7. 在 **Username** 字段中，输入要进行身份验证的用户名。请注意，您为计算配置集输入的用户名必须与创建镜像时使用的相同。
8. 要验证用户，请使用以下选项之一：
 - 要使用密码进行身份验证，请在 **Password** 字段中输入密码。
 - 要使用 **SSH** 密钥进行身份验证，请在 **SSH Key** 字段中输入 **SSH** 密钥。
9. 可选：如果您希望虚拟机使用高级虚拟机磁盘，请选择 **Premium OS Disk** 复选框。
10. 从 **OS Disk 缓存** 列表中，选择磁盘缓存设置。
11. 可选：在 **Custom Script Command** 字段中输入在置备虚拟机时要在虚拟机上执行的命令。
12. 可选：如果您要在置备完成时运行自定义脚本，请在 **Comma** 分隔的文件 **URIs** 字段中输入要使用的、以逗号分隔的脚本文件 **URI**。脚本开始时必须包含 **sudo**，因为 **Red Hat Satellite** 下载文件到主机上的 **/var/lib/waagent/custom-script/download/0/** 目录，并且脚本需要执行 **sudo** 权限。
13. 可选：您可以选择 **NVIDIA 驱动程序 / CUDA** 复选框来添加 **NVIDIA 驱动程序**。如需更多信息，请参阅以下 **Microsoft Azure** 文档：
 - [Linux 的 NVIDIA GPU 驱动程序扩展](#)

- **Windows 的 NVIDIA GPU 驱动程序扩展**

14.

可选：如果要在虚拟机上创建额外的卷，点 **Add Volume** 按钮，以 GB 为单位输入 **Size**，然后选择 **Data Disk** 缓存方法。

-

请注意，这些磁盘的最大数量取决于所选的 VM Size。有关 Microsoft Azure 虚拟机存储要求的更多信息，请参阅 [Microsoft Azure 文档](#)。

15.

点 **Add Interface**。



重要

接口的最大数量取决于所选的 VM Size。如需更多信息，请参阅上述 [Microsoft Azure 文档链接](#)。

16.

从 **Azure Subnet** 列表中，选择要置备的 **Azure** 子网。

17.

从 **公共 IP** 列表中，选择 **公共 IP** 设置。

18.

可选：如果您希望虚拟机使用静态专用 IP，请选择 **静态私有 IP** 复选框。

19.

点 **Submit**。

CLI 过程

1.

创建用于 **Azure Resource Manager** 计算资源的计算配置集：

```
# hammer compute-profile create --name compute_profile_name
```

2.

在 **compute** 配置集中添加 **Azure** 详情。使用 **username** 设置，输入镜像访问的 **SSH** 用户名。请注意，您为计算配置集输入的用户名必须与创建镜像时使用的相同。

```
# hammer compute-profile values create \
```

```

--compute-
attributes="resource_group=resource_group,vm_size=Standard_B1s,username=azure_user,
password=azure_password,platform=Linux,script_command=touch /var/tmp/text.txt" \
--compute-profile "compute_profile_name" \
--compute-resource azure_cr_name \
--
interface="compute_public_ip=Dynamic,compute_network=mysubnetID,compute_private_ip=false" \
--volume="disk_size_gb=5,data_disk_caching=None"

```

可选：如果要在置备后在虚拟机上运行脚本，请指定以下设置：

- 要直接输入脚本，请使用 `script_command` 设置，请输入要在调配的虚拟机上执行的命令。
- 要从 URI 运行脚本，请使用 `script_uris` 设置，请输入要使用的脚本的逗号分隔列表 URI。脚本开始时必须包含 `sudo`，因为 Red Hat Satellite 下载文件到主机上的 `/var/lib/waagent/custom-script/download/0/` 目录，因此脚本需要执行 `sudo` 特权。

16.4. 在 MICROSOFT AZURE RESOURCE MANAGER 上创建基于镜像的主机

在 Satellite 中，您可以使用 Microsoft Azure Resource Manager 置备从现有镜像创建主机。新主机条目会触发 Microsoft Azure Resource Manager 服务器，以使用预先存在的镜像作为新卷的基础来创建实例。

要使用 CLI 而不是 Satellite Web UI，请参阅 [CLI 过程](#)。

流程

1. 在 Satellite Web UI 中，导航到 **Hosts > Create Host**。
2. 在 **Name** 字段中输入主机的名称。
3. 点 **Organization and Location** 选项卡，以确保置备上下文自动设置为当前上下文。
4. 从 **Host Group** 列表中，选择您要用来填充表单的主机组。

5. 从 **Deploy on** 列表中，选择 **Microsoft Azure Resource Manager** 连接。
6. 从 **Compute Profile** 列表中，选择要用于自动填充虚拟机设置的配置集。
7. 从 **Lifecycle Environment** 列表中，选择环境。
8. 单击 **Interfaces** 选项卡，再单击主机接口上的 **Edit**。
9. 验证这些字段是否已自动填充，特别是以下项目：
 - **Host** 选项卡中的 **Name** 成为 **DNS** 名称。
 - **MAC 地址** 字段为空。**Microsoft Azure Resource Manager** 在置备过程中为主机分配一个 **MAC** 地址。
 - **Azure Subnet** 字段填充所需的 **Azure** 子网。
 - 为主机上的第一个接口自动选择 **Managed**、**Primary** 和 **Provision** 选项。如果没有，请选择它们。
10. 可选：如果要使用静态专用 IP 地址，请从 **IPv4 Subnet** 列表中选择与 **Azure** 子网地址匹配的 **Network Address** 字段的 **Satellite** 子网。在 **IPv4 Address** 字段中，在 **Azure** 子网范围内输入 **IPv4** 地址。
11. 单击 **Operating System** 选项卡，并确认所有字段都自动包含值。
12. 对于 **Provisioning** 方法，请确保选择了 **Image Based**。
13. 从 **Image** 列表中，选择您要用于置备的 **Azure Resource Manager** 镜像。

14. 在 **Root Password** 字段中，输入要进行身份验证的根密码。
15. 点 **Provisioning** 模板中的 **Resolve**，以检查新主机可以识别要使用的正确置备模板。
16. 单击 **Virtual Machine** 选项卡，并确认这些设置填充了主机组和计算配置集的详情。修改这些设置以符合您的需要。
17. 单击 **Parameters** 选项卡，并确保存在一个提供激活码的参数。如果没有，请添加激活码。
18. 点 **Submit** 保存主机条目。

CLI 过程

- 使用 `hammer host create` 命令创建主机，并包含 `--provision-method` 镜像。将以下示例中的值替换为适合您的环境的值。

```
# hammer host create \  
--architecture x86_64 \  
--compute-profile "My_Compute_Profile" \  
--compute-resource "My_Azure_Compute_Resource" \  
--domain "My_Domain" \  
--image "My_Azure_Image" \  
--location "My_Location" \  
--name="Azure_VM" \  
--operatingsystem "My_Operating_System" \  
--organization "My_Organization" \  
--provision-method "image"
```

有关此计算资源的其他主机创建参数的更多信息，请输入 `hammer host create --help` 命令。

16.5. 删除 MICROSOFT AZURE 上的虚拟机

您可以从 **Satellite** 中删除在 **Microsoft Azure** 上运行的虚拟机。

流程

1. 在 **Satellite Web UI** 中，导航到 **Infrastructure > Compute Resources**。

2. 选择 **Microsoft Azure** 供应商。
3. 在 **Virtual Machines** 选项卡中，从 **Actions** 菜单中点 **Delete**。这会从 **Microsoft Azure** 计算资源中删除虚拟机，同时保留 **Satellite** 中的任何关联的主机。如果要删除孤立的主机，请导航到 **Hosts > All Hosts** 并手动删除主机。

16.6. 卸载 MICROSOFT AZURE 插件

如果您之前已安装了 **Microsoft Azure** 插件，但没有再使用它来将主机部署到 **Azure**，您可以从 **Satellite** 服务器中卸载它。

流程

1. 从 **Satellite** 服务器卸载 **Azure** 计算资源供应商：
 - **Red Hat Enterprise Linux 8 :**

```
# satellite-maintain packages remove rubygem-foreman_azure_rm rubygem-  
ms_rest_azure  
# satellite-installer --no-enable-foreman-plugin-azure
```
 - **Red Hat Enterprise Linux 7 :**

```
# satellite-maintain packages remove -y tfm-rubygem-foreman_azure_rm tfm-rubygem-  
ms_rest_azure  
# satellite-installer --no-enable-foreman-plugin-azure
```
2. 可选：在 **Satellite Web UI** 中，导航到 **Administer > About** 并选择 **Available Providers** 选项卡来验证 **Microsoft Azure** 插件的删除。

附录 A. 置备示例的初始化脚本

如果您没有遵循 *管理内容 中的示例*，您可以使用以下初始化脚本创建用于置备示例的环境。

流程

1.

创建一个脚本文件(`content-init.sh`)并包含以下内容：

```
#!/bin/bash

MANIFEST=$1

# Import the content from Red Hat CDN
hammer organization create \
--name "ACME" \
--label "ACME" \
--description "Our example organization for managing content."

hammer subscription upload \
--file ~/$MANIFEST \
--organization "ACME"

hammer repository-set enable \
--name "Red Hat Enterprise Linux 7 Server (RPMs)" \
--releasever "7Server" \
--basearch "x86_64" \
--product "Red Hat Enterprise Linux Server" \
--organization "ACME"

hammer repository-set enable \
--name "Red Hat Enterprise Linux 7 Server (Kickstart)" \
--releasever "7Server" \
--basearch "x86_64" \
--product "Red Hat Enterprise Linux Server" \
--organization "ACME"

hammer repository-set enable \
--name "Red Hat Satellite Client 6 (for RHEL 7 Server) (RPMs)" \
--basearch "x86_64" \
--product "Red Hat Enterprise Linux Server" \
--organization "ACME"

hammer product synchronize --name "Red Hat Enterprise Linux Server" \
--organization "ACME"

# Create our application life cycle
hammer lifecycle-environment create \
--name "Development" \
--description "Environment for ACME's Development Team" \
--prior "Library" \
--organization "ACME"
```



```

hammer lifecycle-environment create \
--name "Testing" \
--description "Environment for ACME's Quality Engineering Team" \
--prior "Development" \
--organization "ACME"

hammer lifecycle-environment create \
--name "Production" \
--description "Environment for ACME's Product Releases" \
--prior "Testing" \
--organization "ACME"

# Create and publish our Content View
hammer content-view create \
--name "Base" \
--description "Base operating system" \
--repositories "Red Hat Enterprise Linux 7 Server RPMs x86_64 7Server,Red Hat Satellite
Client 6 for RHEL 7 Server RPMs x86_64" \
--organization "ACME"

hammer content-view publish \
--name "Base" \
--description "Initial Content View for our operating system" \
--organization "ACME"

hammer content-view version promote \
--content-view "Base" \
--version 1 \
--to-lifecycle-environment "Development" \
--organization "ACME"

hammer content-view version promote \
--content-view "Base" \
--version 1 \
--to-lifecycle-environment "Testing" \
--organization "ACME"

hammer content-view version promote \
--content-view "Base" \
--version 1 \
--to-lifecycle-environment "Production" \
--organization "ACME"

```

2.

在脚本中设置可执行权限：

```
# chmod +x content-init.sh
```

3.

从红帽客户门户网站下载您的红帽订阅清单副本，并在清单上运行该脚本：

```
# ./content-init.sh manifest_98f4290e-6c0b-4f37-ba79-3a3ec6e405ba.zip
```

这会导入本指南中置备示例所需的红帽内容。

附录 B. 置备 FIPS-COMPLIANT 主机

Satellite 支持置备符合 **Cryptographic 模块** 标准的国家标准与技术安全要求的主机，参考号 FIPS 140-2，这里称为 FIPS。

要启用 FIPS 兼容的主机置备，请完成以下任务：

- 更改操作系统的置备密码哈希算法
- 创建主机组并设置主机组参数以启用 FIPS

如需更多信息，请参阅 *管理主机* 指南中的 [创建主机组](#)。

置备的主机应用了 FIPS 兼容设置。要确认启用了这些设置，请完成 [第 B.3 节“验证 FIPS 模式已启用”](#) 中的步骤。

B.1. 更改置备密码哈希算法

要置备 FIPS 兼容主机，您必须首先将置备中使用的密码哈希算法设置为 SHA256。这个配置设置必须针对您要部署为 FIPS 兼容的每个操作系统应用。

流程

1. 确定操作系统 ID：

```
# hammer os list
```

2. 更新每个操作系统的密码哈希。

```
# hammer os update \  
--password-hash SHA256  
--title "My_Operating_System"
```

请注意，您不能使用以逗号分隔的值列表。

B.2. 设置 FIPS-ENABLED 参数

要置备 FIPS 兼容主机，您必须创建一个主机组，并将主机组参数 `fips_enabled` 设置为 `true`。如果它没有设置为 `true`，或者不存在，则特定于 FIPS 的更改不会应用到该系统。当您置备主机或主机组时，您可以设置此参数。

要在置备主机时设置此参数，请将 `--parameters fips_enabled=true` 附加到 `hammer` 命令中。

```
# hammer hostgroup set-parameter \  
--hostgroup "My_Host_Group" \  
--name fips_enabled \  
--value "true"
```

如需更多信息，请参阅命令 `hammer hostgroup set-parameter --help` 的输出。

B.3. 验证 FIPS 模式已启用

要验证这些 FIPS 合规更改是否成功，您必须置备主机并检查其配置。

流程

1. 以 `root` 或具有管理员级别帐户登录到主机。
2. 输入以下命令：

```
$ cat /proc/sys/crypto/fips_enabled
```

`1` 表示启用了 FIPS 模式。

附录 C. 为 RED HAT SATELLITE 构建云镜像

使用本节在 Red Hat Satellite 中构建并注册镜像。

您可以使用预配置的 Red Hat Enterprise Linux KVM 客户机 QCOW2 镜像：

- [最新的 RHEL 8 KVM 客户机镜像](#)
- [最新的 RHEL 7 KVM 客户机镜像](#)

这些镜像包含 cloud-init。要正常工作，它们必须使用 ec2 兼容元数据服务来置备 SSH 密钥。



注意

对于 KVM 客户机镜像：

- 镜像中的 root 帐户被禁用，但 sudo 访问权限被授予一个名为 cloud-user 的特殊用户。
- 没有为此镜像设置 root 密码。通过将 ! 放在第二个字段中，root 密码在 /etc/shadow 中锁定。

如果要创建自定义 Red Hat Enterprise Linux 镜像，请参阅为 [Red Hat Enterprise Linux 7 编写自定义 Red Hat Enterprise Linux 8 镜像或镜像构建器指南](#)

C.1. 创建自定义 RED HAT ENTERPRISE LINUX 镜像

先决条件

- 使用 Linux 主机计算机创建镜像。在这个示例中，我们使用 Red Hat Enterprise Linux 7 Workstation。
- 在您的工作站上使用 virt-manager 完成此流程。如果您在远程服务器上创建镜像，请使用

virt-manager 从工作站连接到服务器。

- **Red Hat Enterprise Linux 7 或 6 个 ISO 文件**（请参阅 [Red Hat Enterprise Linux 7.4 Binary DVD](#) 或 [Red Hat Enterprise Linux 6.9 Binary DVD](#)）。

有关安装 Red Hat Enterprise Linux 工作站的详情，请参考 [Red Hat Enterprise Linux 7 安装指南](#)。

在创建自定义镜像前，请安装以下软件包：

- **安装 libvirt、qemu-kvm 和图形工具：**

```
# yum install virt-manager virt-viewer libvirt qemu-kvm
```

- **安装以下命令行工具：**

```
# yum install virt-install libguestfs-tools-c
```



注意

在以下步骤中，在托管 libvirt 环境的工作站上输入带有 [root@host] 114 提示符的所有命令。

C.2. 注册中支持的客户端

Satellite 支持以下用于注册的操作系统和架构：

支持的主机操作系统

主机可以使用以下操作系统：

- **Red Hat Enterprise Linux 9、8、7**
- 带有 [ELS 附加组件](#) 的 **Red Hat Enterprise Linux 6**

支持的主机架构

主机可以使用以下架构：

- **i386**
- **x86_64**
- **s390x**
- **ppc_64**

C.3. 配置主机以进行注册

配置您的主机以注册到 **Satellite 服务器** 或 **Capsule 服务器**。

先决条件

- 主机必须使用受支持的操作系统。如需更多信息，请参阅 [第 C.2 节“注册中支持的客户端”](#)。

流程

- 确保启用了 **time-synchronization** 工具并在主机上运行。

C.4. 注册主机

您可以使用注册模板注册主机，并在注册过程中设置各种集成功能和主机工具。

先决条件

- 您的用户帐户分配了 **create_hosts** 权限的角色。

- 您必须在要注册的主机上具有 root 权限。
- **Satellite 服务器、任何 Capsule 服务器和所有主机都必须与同一 NTP 服务器同步，并且启用了和运行时间同步工具。**
- 主机必须有一个激活码。如需更多信息，[请参阅管理内容中的管理激活码](#)。
- 如果要使用 Capsule 服务器而不是 Satellite 服务器，请确保已相应地配置了 Capsule 服务器。如需更多信息，[请参阅安装 Capsule 服务器中的为主机注册和置备配置 Capsule](#)。
- 如果您的 Satellite 服务器或 Capsule 服务器位于 HTTP 代理后面，请将主机上的订阅管理器配置为使用 HTTP 代理进行连接。如需更多信息，[请参阅红帽知识库中的如何通过防火墙或代理访问 Red Hat Subscription Manager \(RHSM\)](#)。

流程

1. 在 Satellite Web UI 中，导航到 Hosts > Register Host.
2. 可选：选择不同的机构。
3. 可选：选择不同的位置。
4. 可选：从 Host Group 列表中，选择要将主机关联的主机组。从主机组继承值的字段：操作系统、激活码和生命周期环境。
5. 可选：从 Operating system 列表中选择您要注册的主机操作系统。
6. 可选：从 Capsule 列表中，选择要通过的 Capsule 注册主机的 Capsule。
7. 可选：如果您要使第一个调用不安全，请选择 Insecure 选项。在此第一次调用期间，主机从 Satellite 下载 CA 文件。主机将使用此 CA 文件与以后的所有调用连接 Satellite，使其安全。

红帽建议您避免不安全的调用。

如果攻击者位于 **Satellite** 和主机之间的网络中，从第一个不安全的调用中获取 CA 文件，攻击者可以访问 API 调用的内容，并从注册的主机和 JSON Web 令牌(JWT)获取 API 调用的内容。因此，如果您选择在注册过程中部署 SSH 密钥，攻击者可以使用 SSH 密钥访问主机。

相反，您可以在注册主机前在每个主机上手动复制并安装 CA 文件。

要做到这一点，找到 **Satellite** 通过导航到 **Administer > Settings > Authentication** 并查找 **SSL CA** 文件设置的值来存储 CA 文件的位置。

将 CA 文件复制到主机上的 `/etc/pki/ca-trust/source/anchors/` 目录中，并输入以下命令：

```
# update-ca-trust enable
# update-ca-trust
```

然后，使用安全 `curl` 命令注册主机，例如：

```
# curl -sS https://satellite.example.com/register ...
```

以下是带有 `--insecure` 选项的 `curl` 命令示例：

```
# curl -sS --insecure https://satellite.example.com/register ...
```

8.

选择 **Advanced** 选项卡。

9.

从 **Setup REX** 列表中，选择是否要将 **Satellite SSH 密钥** 部署到主机。

如果设置为 **Yes**，则会在注册的主机上安装公共 SSH 密钥。继承的值基于 `host_registration_remote_execution` 参数。它可以继承，例如从主机组、操作系统或机构中继承。覆盖后，所选值将存储在 `host` 参数级别上。

10.

从 **Setup Insights** 列表中，选择是否要安装 `insights-client` 并将主机注册到 **Insights**。

Insights 工具仅适用于 Red Hat Enterprise Linux。它对其他操作系统没有影响。

您必须在注册的机器中启用以下软件仓库：

- **RHEL 6: rhel-6-server-rpms**
- **RHEL 7: rhel-7-server-rpms**
- **RHEL 8: rhel-8-for-x86_64-appstream-rpms**

在 RHEL 8 上默认安装 insights-client 软件包，但使用 "Minimal Install" 选项部署 RHEL 8 的环境中。

11. **可选：在 Install packages 字段中，列出您要在注册后在主机上安装的软件包（以空格分开）。这可以通过 host_packages 参数设置。**
12. **可选：选择 Update packages 选项，以在注册后更新主机上的所有软件包。这可以通过 host_update_packages 参数设置。**
13. **可选：在 Repository 字段中，输入在执行注册前要添加的存储库。例如，使 subscription-manager 软件包可用于注册目的非常有用。对于 Red Hat 系列发行版本，请输入存储库的 URL，例如 http://rpm.example.com/。**
14. **可选：在 Repository GPG 密钥 URL 字段中，指定公钥来验证 GPG 签名软件包的签名。它需要在 ASCII 表单中使用 GPG 公钥标头指定。**
15. **可选：在 Token 生命周期（小时）字段中，更改 Satellite 用于身份验证的 JSON Web Token (JWT) 的有效性持续时间。此令牌的持续时间定义了生成的 curl 命令的工作方式。您可以将持续时间设置为 039) -114999 999 小时或无限。**

请注意，Satellite 应用生成 curl 命令的用户的权限到主机授权。如果用户丢失或获得额外权限，则 JWT 更改的权限也会有变化。因此，不要在令牌期间删除、阻止或更改用户的权限。

JWT 范围仅限于注册端点，无法在其它任何位置使用。

16. **可选：在 Remote Execution Interface 字段中输入主机必须用于 SSH 连接的网络接口标识符。如果将此字段留空，Satellite 将使用默认网络接口。**
17. **在 Activation Keys 字段中输入要分配给主机的一个或多个激活码。**
18. **可选：选择 生命周期阶段。**
19. **可选：如果要忽略订阅管理器 错误，请选择 Ignore error 选项。**
20. **可选：如果您要在注册前删除任何 katello-ca-consumer rpms，并使用 --force 参数运行 subscription-manager，请选择 Force 选项。**
21. **点 Generate 按钮。**
22. **复制生成的 curl 命令。**
23. **在您要注册的主机上，以 root 用户身份运行 curl 命令。**

C.5. 安装 KATELLO 代理

您可以安装 Katello 代理来远程更新 Satellite 客户端。



注意

Katello 代理已弃用，并将在以后的 Satellite 版本中删除。迁移进程以使用远程执行功能远程更新客户端。如需更多信息，请参阅管理主机中的将主机从 Katello Agent 迁移到远程执行。

katello-agent 软件包依赖于提供 goferd 服务的 gofer 软件包。

先决条件

- 您已在 **Satellite 服务器** 上启用了 **Satellite 客户端 6 存储库**。如需更多信息，请参阅 [在连接的网络环境中安装 Satellite 服务器 中的 启用 Satellite 客户端 6 存储库](#)。
- 您已在 **Satellite 服务器** 上同步了 **Satellite 客户端 6 存储库**。如需更多信息，请参阅 [在连接的网络环境中安装 Satellite 服务器 中的 同步 Satellite 客户端 6 存储库](#)。
- 您已在客户端上启用了 **Satellite 客户端 6 存储库**。

流程

1. 安装 **katello-agent** 软件包：

```
# yum install katello-agent
```

2. 启动 **goferd** 服务：

```
# systemctl start goferd
```

C.6. 在主机上手动安装和配置 PUPPET 代理

在主机上手动安装和配置 Puppet 代理。

先决条件

- 主机必须分配有 Puppet 环境。
- 必须启用 **Satellite Client 6 存储库** 并同步到 **Satellite 服务器**，并在主机上启用。如需更多信息，请参阅管理内容中的 [导入内容](#)。

流程

1. 以 **root 用户** 身份登录主机。

2.

安装 Puppet 代理软件包。

-

在运行 Red Hat Enterprise Linux 8 及更高版本的主机上：

```
# dnf install puppet-agent
```

-

在运行 Red Hat Enterprise Linux 7 及更早版本的主机上：

```
# yum install puppet-agent
```

3.

使用以下脚本，将 Puppet 代理添加到当前 shell 中的 PATH 中：

```
./etc/profile.d/puppet-agent.sh
```

4.

配置 Puppet 代理。将 environment 参数设置为主机所属的 Puppet 环境的名称：

```
# puppet config set server satellite.example.com --section agent  
# puppet config set environment My_Puppet_Environment --section agent
```

5.

启动 Puppet 代理服务：

```
# puppet resource service puppet ensure=running enable=true
```

6.

为主机创建证书：

```
# puppet ssl bootstrap
```

7.

在 Satellite Web UI 中，导航到 Infrastructure > Capsules。

8.

从所需 Capsule 服务器的 Actions 列中的列表中，选择 Certificates。

9.

单击所需主机右侧的 Sign，以签署 Puppet 代理的 SSL 证书。

10.

在主机上再次运行 Puppet 代理：

```
# puppet ssl bootstrap
```

其它资源

•

有关 Puppet 的更多信息，请参阅在 [Red Hat Satellite 中使用 Puppet 集成管理配置](#)。

C.7. 完成 RED HAT ENTERPRISE LINUX 7 镜像

流程

1.

更新系统：

```
# {package-update}
```

2.

安装 `cloud-init` 软件包：

```
# yum install cloud-utils-growpart cloud-init
```

3.

打开 `/etc/cloud/cloud.cfg` 配置文件：

```
# vi /etc/cloud/cloud.cfg
```

4.

在标题 `cloud_init_modules` 下添加：

```
- resolv-conf
```

当实例第一次引导时，`resolv-conf` 选项会自动配置 `resolv.conf`。此文件包含与实例相关的信息，如名称服务器、域和其他选项。

5.

打开 `/etc/sysconfig/network` 文件：

```
# vi /etc/sysconfig/network
```

6. **添加以下行以避免访问 EC2 元数据服务的问题：**

```
NOZEROCONF=yes
```

7. **取消注册虚拟机，以便生成的镜像不包含基于它的每个实例的相同订阅详情：**

```
# subscription-manager repos --disable=*  
# subscription-manager unregister
```

8. **关闭实例：**

```
# poweroff
```

9. **在 Red Hat Enterprise Linux Workstation 中，以 root 用户身份连接到终端，并进入 /var/lib/libvirt/images/ 目录：**

```
# cd /var/lib/libvirt/images/
```

10. **使用 virt-sysprep 命令重置并清理镜像，以便它可用于在没有问题的情况下创建实例：**

```
# virt-sysprep -d rhel7
```

11. **使用 virt-sparsify 命令减少镜像大小。这个命令将磁盘镜像中的任何可用空间转换为主机内的可用空间：**

```
# virt-sparsify --compress rhel7.qcow2 rhel7-cloud.qcow2
```

这会在输入命令的位置创建一个新的 rhel7-cloud.qcow2 文件。

C.8. 完成 RED HAT ENTERPRISE LINUX 6 镜像

流程

1. **更新系统：**

```
# {package-update}
```

2.

安装 cloud-init 软件包：

```
# yum install cloud-utils-growpart cloud-init
```

3.

编辑 /etc/cloud/cloud.cfg 配置文件并在 cloud_init_modules 下添加：

```
- resolv-conf
```

当实例第一次引导时，**resolv-conf** 选项会自动配置 **resolv.conf** 配置文件。此文件包含与实例相关的信息，如名称服务器、域和其他选项。

4.

要防止网络问题，请创建 /etc/udev/rules.d/75-persistent-net-generator.rules 文件，如下所示：

```
# echo "#" > /etc/udev/rules.d/75-persistent-net-generator.rules
```

这可防止创建 **/etc/udev/rules.d/70-persistent-net.rules** 文件。如果创建了 **/etc/udev/rules.d/70-persistent-net.rules**，则从快照引导时联网可能无法正常工作（网络接口是以“eth1”而不是“eth0”分配的 IP 地址）。

5.

在 /etc/sysconfig/network 中添加以下行以避免访问 EC2 元数据服务出现问题：

```
NOZEROCONF=yes
```

6.

取消注册虚拟机，以便生成的镜像不包含基于它的每个实例的相同订阅详情：

```
# subscription-manager repos --disable=*  
# subscription-manager unregister  
# yum clean all
```

7.

关闭实例：

```
# poweroff
```

8.

在 Red Hat Enterprise Linux Workstation 中，以 root 身份登录并使用 virt-sysprep 命令重置并清理镜像，以便它可用于在没有问题的情况下创建实例：


```
# virt-sysprep -d rhel6
```

9.

使用 **virt-sparsify** 命令减少镜像大小。这个命令将磁盘镜像中的任何可用空间转换为主机内的可用空间：

```
# virt-sparsify --compress rhel6.qcow2 rhel6-cloud.qcow2
```

这会在输入命令的位置创建一个新的 **rhel6-cloud.qcow2** 文件。



注意

您必须根据应用到实例的类别中的磁盘空间来手动调整实例分区大小。

C.8.1. 后续步骤

- 对您要使用 **Satellite** 置备的每个镜像重复这个过程。
- 将镜像移动到您要存储的位置，以供以后使用。

C.9. 后续步骤

- 对您要使用 **Satellite** 置备的每个镜像重复这个过程。
- 将镜像移动到您要存储的位置，以供以后使用。

附录 D. 主机参数层次结构

您可以在置备主机时访问主机参数。主机从以下位置继承其参数，以增加优先级：

参数级别	在 Satellite Web UI 中设置
全局定义的参数	配置 > Global 参数
机构级别参数	管理 > 机构
位置级别参数	Management > Locations
域级别参数	infrastructure > Domains
子网级别参数	infrastructure > Subnets
操作系统级别参数	hosts > Operating Systems
主机组级别参数	配置 > 主机组
主机参数	hosts > All Hosts