



Red Hat Satellite 6.15

置备主机

配置调配资源和网络，调配物理计算机，在云提供商或虚拟化基础架构上调配虚拟机，手动创建主机或使用 Discovery 服务创建主机

Red Hat Satellite 6.15 置备主机

配置调配资源和网络，调配物理计算机，在云提供商或虚拟化基础架构上调配虚拟机，手动创建主机或使用 Discovery 服务创建主机

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 Provisioning 指南提供了有关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调配网络上的主机。

目录

使开源包含更多	6
对红帽文档提供反馈	7
第 1 章 置备简介	8
1.1. 在 RED HAT SATELLITE 中置备方法	8
1.2. 支持的云供应商	8
1.3. 支持的虚拟化基础架构	8
1.4. 网络引导置备 workflow	9
1.5. 网络引导所需的引导顺序	11
第 2 章 配置置备资源	12
2.1. 置备上下文	12
2.2. 设置置备上下文	12
2.3. 创建操作系统	12
2.4. 更新多个操作系统的详情	13
2.5. 创建构架	14
2.6. 创建硬件模型	14
2.7. 为主机操作系统使用同步的 KICKSTART 存储库	15
2.8. 在 SATELLITE 中添加安装介质	16
2.9. 创建分区表	17
2.10. 动态分区示例	18
2.11. 置备模板	18
2.12. 置备模板的类型	19
2.13. 创建置备模板	20
2.14. 克隆置备模板	21
2.15. 创建自定义置备片断	21
2.16. RED HAT ENTERPRISE LINUX 的自定义置备片断示例	22
2.17. 将模板与操作系统关联	22
2.18. 创建计算配置集	23
2.19. 为主机设置默认加密的 ROOT 密码	24
2.20. 使用 NOVNC 访问虚拟机	24
第 3 章 配置网络	26
3.1. 事实和 NIC 过滤	26
3.2. 通过从数据库中删除 NIC 来优化性能	26
3.3. 网络资源	27
3.4. SATELLITE 和 DHCP 选项	28
3.5. 对 SATELLITE 中的 DHCP 问题进行故障排除	29
3.6. 基于镜像的置备的先决条件	30
3.7. 配置网络服务	31
3.8. 将域添加到 SATELLITE 服务器	34
3.9. 将子网添加到 SATELLITE 服务器	35
第 4 章 使用 INFOBLOX 作为 DHCP 和 DNS 供应商	37
4.1. INFOBLOX 限制	37
4.2. INFOBLOX 先决条件	37
4.3. 在 CAPSULE 服务器上安装 INFOBLOX CA 证书	37
4.4. 安装 DHCP INFOBLOX 模块	38
4.5. 安装 DNS INFOBLOX 模块	38
第 5 章 使用 PXE 置备主机	40
5.1. 裸机置备的先决条件	40

5.2. 配置安全令牌有效期持续时间	40
5.3. 使用无人值守调配创建主机	41
5.4. 使用无 PXE 的调配创建主机	42
5.5. 使用 UEFI HTTP 引导置备创建主机	44
5.6. 在置备过程中部署 SSH 密钥	47
第 6 章 使用 IPXE 减少置备时间	48
6.1. 使用 IPXE 的先决条件	48
6.2. 配置 IPXE 环境	48
6.3. 引导虚拟机	49
6.4. 来自 PXELINUX 的 CHAINBOOTING IPXE	50
第 7 章 配置发现服务	53
7.1. 安装发现服务	54
7.2. 置备模板 PXELINUX DISCOVERY 片断	54
7.3. 发现主机的自动上下文	55
7.4. 发现模板和片断设置	55
7.5. 从发现的主机创建主机	56
7.6. 创建发现规则	58
7.7. 实施无 PXE 的发现	60
7.8. 无人使用、自定义和镜像重新 MASTERING	62
7.9. 扩展发现镜像	65
7.10. 发现故障排除	66
第 8 章 使用 RED HAT IMAGE BUILDER 镜像进行置备	69
第 9 章 在 KVM (LIBVIRT)上调配虚拟机	71
9.1. 为 KVM 连接配置 SATELLITE 服务器	72
9.2. 向 SATELLITE 服务器添加 KVM 连接	72
9.3. 将 KVM 镜像添加到 SATELLITE 服务器	74
9.4. 在计算配置集中添加 KVM 详情	75
9.5. 在 KVM 上创建主机	76
第 10 章 在 RED HAT VIRTUALIZATION 上置备虚拟机	81
10.1. 将 RED HAT VIRTUALIZATION 连接添加到 SATELLITE 服务器	82
10.2. 在 RED HAT VIRTUALIZATION 中准备 CLOUD-INIT 镜像	84
10.3. 在 SATELLITE 服务器中添加 RED HAT VIRTUALIZATION 镜像	84
10.4. 准备 CLOUD-INIT 模板	86
10.5. 在计算配置集中添加 RED HAT VIRTUALIZATION 详情	88
10.6. 在 RED HAT VIRTUALIZATION 上创建主机	90
第 11 章 在 VMWARE VSPHERE 中置备虚拟机	94
11.1. VMWARE 置备的先决条件	94
11.2. 创建 VMWARE 用户	94
11.3. 将 VMWARE 连接添加到 SATELLITE 服务器	96
11.4. 将 VMWARE 镜像添加到 SATELLITE 服务器	98
11.5. 在计算配置集中添加 VMWARE 详情	99
11.6. 在 VMWARE 上创建主机	101
11.7. 使用 VMWARE CLOUD-INIT 和 USERDATA 模板进行配置	105
11.8. 删除 VMWARE 上的虚拟机	110
11.9. 将虚拟机从 VMWARE 导入到 SATELLITE	111
11.10. 计算资源缓存	111
第 12 章 在 OPENSIFT VIRTUALIZATION 上置备虚拟机	113
12.1. 在 SATELLITE 服务器中添加 OPENSIFT VIRTUALIZATION 连接	114

第 13 章 在 RED HAT OPENSTACK PLATFORM 上置备云实例	116
13.1. 添加 RED HAT OPENSTACK PLATFORM 连接到 SATELLITE 服务器	116
13.2. 将 RED HAT OPENSTACK PLATFORM 镜像添加到 SATELLITE 服务器	118
13.3. 在计算配置集中添加 RED HAT OPENSTACK PLATFORM 详情	119
13.4. 在 RED HAT OPENSTACK PLATFORM 上创建基于镜像的主机	120
第 14 章 在 AMAZON EC2 中置备云实例	124
14.1. AMAZON EC2 置备的先决条件	124
14.2. 安装 AMAZON EC2 插件	124
14.3. 将 AMAZON EC2 连接添加到 SATELLITE 服务器	124
14.4. 将 HTTP 代理与计算资源搭配使用	126
14.5. 为 AMAZON EC2 创建镜像	127
14.6. 将 AMAZON EC2 镜像添加到 SATELLITE 服务器	128
14.7. 在计算配置集中添加 AMAZON EC2 详情	129
14.8. 在 AMAZON EC2 上创建基于镜像的主机	130
14.9. 使用 SSH 连接到 AMAZON EC2 实例	132
14.10. 为 AMAZON WEB SERVICE EC2 环境配置完成模板	134
14.11. 删除 AMAZON EC2 上的虚拟机	136
14.12. 卸载 AMAZON EC2 插件	136
14.13. 有关 AMAZON WEB SERVICES 和 SATELLITE 的更多信息	136
第 15 章 在 GOOGLE COMPUTE ENGINE 上置备云实例	138
15.1. 将 GOOGLE GCE 连接添加到 SATELLITE 服务器	138
15.2. 将 GOOGLE COMPUTE ENGINE 镜像添加到 SATELLITE 服务器	140
15.3. 在计算配置集中添加 GOOGLE GCE 详情	141
15.4. 在 GOOGLE COMPUTE ENGINE 上创建基于镜像的主机	142
15.5. 删除 GOOGLE GCE 上的虚拟机	145
第 16 章 在 MICROSOFT AZURE RESOURCE MANAGER 上置备云实例	146
16.1. 在 SATELLITE 服务器中添加 MICROSOFT AZURE RESOURCE MANAGER 连接	147
16.2. 在 SATELLITE 服务器中添加 MICROSOFT AZURE RESOURCE MANAGER 镜像	148
16.3. 在计算配置集中添加 MICROSOFT AZURE RESOURCE MANAGER 详情	150
16.4. 在 MICROSOFT AZURE RESOURCE MANAGER 上创建基于镜像的主机	153
16.5. 删除 MICROSOFT AZURE 上的虚拟机	156
16.6. 卸载 MICROSOFT AZURE 插件	156
附录 A. 用于置备示例的初始化脚本	157
附录 B. 置备 FIPS 兼容主机	160
B.1. 更改置备密码哈希算法	160
B.2. 设置启用了 FIPS 的参数	161
B.3. 验证启用了 FIPS 模式	161
附录 C. 为 RED HAT SATELLITE 构建一个云镜像	162
C.1. 创建自定义 RED HAT ENTERPRISE LINUX 镜像	162
C.2. 注册中支持的客户端	163
C.3. 配置主机以进行注册	164
C.4. 注册主机	165
C.5. 手动安装和配置 PUPPET 代理	169
C.6. 完成 RED HAT ENTERPRISE LINUX 7 镜像	170
C.7. 完成 RED HAT ENTERPRISE LINUX 6 镜像	172
C.8. 后续步骤	174
附录 D. 主机参数层次结构	175

附录 E. 置备主机所需的权限 176

使开源包含更多

红帽承诺替换我们的代码、文档和网页属性中存在问题的语言。由于这项工作的艰巨性，这些变化正在尽可能地逐步更新。详情请查看 [CTO Chris Wright 的信息](#)。

对红帽文档提供反馈

我们感谢您对我们文档的反馈。让我们了解如何改进它。

使用 Red Hat JIRA 中的 **Create Issue** 表单提供您的反馈。JIRA 问题在 Red Hat Satellite Jira 项目中创建，您可以在其中跟踪其进度。

先决条件

- 确保您已注册了 [红帽帐户](#)。

流程

1. 单击以下链接：[创建问题](#)。如果 Jira 显示登录错误，则登录并在您重定向到表单后继续。
2. 完成 **Summary** 和 **Description** 字段。在 **Description** 字段中，包含文档 URL、章节号以及问题的详细描述。不要修改表单中的任何其他字段。
3. 点 **Create**。

第 1 章 置备简介

置备是一个以裸机或虚拟机开始的进程，并以完全配置的、可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细粒度配置。

1.1. 在 RED HAT SATELLITE 中置备方法

使用 Red Hat Satellite，您可以使用多种方法置备主机。

裸机置备

Satellite 通过使用 PXE 引导和 MAC 地址标识主要置备裸机主机。使用 Satellite 置备裸机主机时，您可以执行以下操作：

- 创建主机条目并指定要置备的物理主机的 MAC 地址。
- 引导空白主机以使用 Satellite 发现服务，该服务会创建一个已准备好调配的主机池。

云供应商

Satellite 连接到私有和公共云提供商，以调配与云环境存储的映像中的主机实例。使用 Satellite 从云置备时，您可以执行以下操作：

- 选择要使用的硬件配置文件或类别。
- 使用其 API 从特定提供程序置备云实例。

虚拟化基础架构

Satellite 连接到虚拟化基础架构服务，如 Red Hat Virtualization 和 VMware。使用 Satellite 置备虚拟机时，您可以执行以下操作：

- 从虚拟映像模板调配虚拟机。
- 使用与裸机提供程序相同的基于 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. 网络引导置备 workflow

置备过程遵循基本的 PXE workflow :

1. 您可以创建一个主机并选择域和子网。Satellite 从 DHCP 胶囊服务器请求一个可用 IP 地址，该地址与子网或 Satellite 中的 PostgreSQL 数据库关联。Satellite 将此 IP 地址加载到 Create Host 窗口中的 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：
 - 与子网关联的胶囊服务器上的 DHCP 记录。
 - 与域关联的胶囊服务器上的转发 DNS 记录。
 - 与子网关联的 DNS 胶囊服务器上的反向 DNS 记录。
 - pxelinux、Grub、Grub2 和 iPXE 配置文件，用于与子网关联的 TFTP 胶囊服务器中的主机。
 - 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. 引导加载程序通过其置备接口 MAC 地址获取主机的配置。
9. 引导加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 provision 模板，并将结果返回给主机。
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。
 - 安装程序会通知 Satellite 在安装后脚本中 **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. 引导加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，则主机将使用模块来配置其自身。

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. 引导加载程序通过其置备接口 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引导的设备，因此 BIOS 跳到下一个设备（从 HDD 引导）。
8. 对于 EFI 主机：
 - 引导加载程序在 ESP 分区上查找 Grub2，并串联它。
9. 如果主机对 Satellite 未知，则会提供默认的引导装载程序配置。启用 Discovery 服务时，它会引导到发现中，否则它会从 HDD 引导。

此工作流因自定义选项而异。例如：

Discovery（发现）

如果使用发现服务，Satellite 会自动检测新主机的 MAC 地址，并在提交请求后重启主机。请注意，TCP 端口 8443 必须能够被主机附加到的胶囊访问，以便 Satellite 重新启动主机。

无 PXE 的置备

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 地址并在 Satellite 中存储 MAC 地址。如果您使用基于镜像的置备，则主机不会遵循标准 PXE 引导和操作系统安装。计算资源创建要使用的镜像副本。根据 Satellite 中的镜像设置，可以在中传递查找的数据以进行初始配置，例如使用 **cloud-init**。Satellite 可以使用 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 删除调配的配置集主机不会销毁外部计算资源上的实际虚拟机。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。如果您不销毁关联的虚拟机，并尝试在以后创建具有相同资源名称的新虚拟机，它将会失败，因为虚拟机名称已存在于外部计算资源中。您仍然可以用于任何已调配的主机的标准主机注册 workflow 将现有虚拟机注册到 Satellite。

1.5. 网络引导所需的引导顺序

对于物理或虚拟 BIOS 主机

1. 使用 network 将第一个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驱动器引导。Satellite 管理 TFTP 启动配置文件，因此主机可以通过重启来轻松调配。

对于物理或虚拟 EFI 主机

1. 使用 network 将第一个引导设备设置为引导配置。
2. 根据 EFI 固件类型和配置，操作系统安装程序通常将 OS 引导装载程序配置为第一个条目。
3. 要再次重启到安装程序，请使用 `efibootmgr` 工具切回到从网络引导。

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，设置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 组件划分为逻辑组。您可以通过 Red Hat Satellite 创建和管理多个机构，并为每个机构分配组件。这样可确保 Satellite 服务器置备特定机构中的主机，且只使用分配给该机构的组件。有关机构的更多信息，[请参阅管理 Red Hat Satellite 中的管理机构](#)。

与机构类似的位置功能。区别在于，位置基于物理或地理位置设置。用户可以在层次结构中嵌套位置。有关位置的更多信息，[请参阅管理 Red Hat Satellite 中的管理位置](#)。

2.2. 设置置备上下文

当您设置置备上下文时，您可以定义要用于置备主机的机构和位置。

组织和位置菜单位于菜单栏中，位于 Satellite Web UI 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：**Any Organization** 和 **Any Location**。

流程

1. 单击 **Any Organization**，再选择 组织。
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调配上下文。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，包含 **--organization** 或 **--organization-label** 和 **--location** 或 **--location-id** 作为选项。例如：

```
# hammer host list --organization "My_Organization" --location "My_Location"
```

此命令输出分配给 **My_Organization** 和 **My_Location** 的主机。

2.3. 创建操作系统

操作系统是资源的集合，用于定义卫星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 CDN 导入操作系统会在 **Hosts > Provisioning Setup > 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，进入到 **Hosts > Operating systems** 并点 **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构架。
9. 点 **分区表** 选项卡并选择适用于此操作系统的可能分区表。
10. 可选：如果您使用非红帽内容，点 **Installation Media** 选项卡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**、配置模板，以及要使用的操作系统的 **Finish 模板**。如果您计划使用 **iPXE** 进行置备，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 \
--architectures "x86_64" \
--description "My_Operating_System" \
--family "Redhat" \
--major 8 \
--media "Red Hat" \
--minor 8 \
--name "Red Hat Enterprise Linux" \
--partition-tables "My_Partition_Table" \
--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。这个示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. 显示有关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. 创建构架

Satellite 中的构架代表主机和操作系统的逻辑分组。在主机通过 Puppet 检查时，Satellite 会自动创建架构。在 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

仅支持使用 PXE、Discovery 和引导磁盘进行 Intel x86_64 架构。如需更多信息，请参阅 [Satellite 6 中的红帽知识库解决方案支持的架构和置备场景](#)。

流程

1. 在 Satellite Web UI 中，进入到 **Hosts > Provisioning Setup > Architectures**。
2. 单击 **Create Architecture**。
3. 在 **Name** 字段中输入架构的名称。
4. 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 > Provisioning Setup > Operating Systems** 下创建并分配它们。
5. 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，进入到 **Hosts > Provisioning Setup > Hardware Models**。
2. 点 **Create Model**。
3. 在 **Name** 字段中输入硬件模型的名称。
4. 另外，在 **Hardware Model** 和 **Vendor Class** 字段中，您可以为您的系统输入相应的信息。
5. 在 **Info** 字段中，输入硬件模型的描述。
6. 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。唯一必需的参数是 **--name**。另外，使用 **--hardware-model** 选项输入硬件模型、带有 **--vendor-class** 选项的 vendor 类，以及 **--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. 为主机操作系统使用同步的 KICKSTART 存储库

Satellite 包含一组同步的 Kickstart 存储库，用于安装置备的主机的操作系统。有关添加存储库的更多信息，请参阅 [管理内容中的 同步存储库](#)。

使用这个流程设置 Kickstart 存储库。

先决条件

在置备前，您必须同时启用 **BaseOS** 和 **Appstream Kickstart**。

流程

1. 将要使用的同步 Kickstart 存储库添加到现有内容视图或创建新内容视图并添加 kickstart 存储库。
对于 Red Hat Enterprise Linux 8，请确保同时添加了 **Red Hat Enterprise Linux 8 for 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。

如果使用断开连接的环境，您必须从 Red Hat Enterprise Linux 二进制 DVD 导入 Kickstart 软件仓库。如需更多信息，请参阅管理内容中的 [导入 Kickstart 存储库](#)。

2. 发布添加 Kickstart 存储库的内容视图的新版本，并将其提升到所需的生命周期环境。如需更多信息，请参阅[管理内容中的 管理内容视图](#)。
3. 当您创建主机时，在 **Operating System** 选项卡中为 **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 包含某些官方 Linux 发行版的安装介质。请注意，其中一些安装介质是针对操作系统的特定版本的目标。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，进入到 **Hosts > Provisioning Setup > Installation Media**。
2. 点 **Create Medium**。
3. 在 **Name** 字段中输入代表安装介质条目的名称。
4. 在 **Path** 中输入包含安装树的 URL。您可以在路径中使用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/
```

5. 在 **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。
6. 单击 **Organizations** 和 **Locations** 选项卡，以更改调配上下文。Satellite 服务器将安装介质添加到设置置备上下文中。
7. 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_Operating_System" \  

```

```
--organizations "My_Organization" \
--os-family "Redhat" \
--path "http://download.example.com/centos/$version/Server/$arch/os/"
```

2.9. 创建分区表

分区表是定义卫星服务器配置新主机上可用磁盘的方式的模板类型。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，进入到 Hosts > Templates > Partition Tables。
2. 点 Create Partition Table。
3. 在 Name 字段中输入分区表的名称。
4. 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。
5. 如果您要将模板识别为其他分区表的可重复使用的片断，请选择 Snippet 复选框。
6. 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。
7. 在 Template 编辑器 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操作系统匹配。例如，Red Hat Enterprise Linux 需要一个与 Kickstart 文件匹配的布局。

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

CLI 过程

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

```
# hammer partition-table create \
--file "~/My_Partition_Table" \
--locations "My_Location" \
```

```
--name "My_Partition_Table" \  
--organizations "My_Organization" \  
--os-family "Redhat" \  
--snippet false
```

2.10. 动态分区示例

使用 Anaconda kickstart 模板，以下部分指示 Anaconda 清除整个磁盘，自动分区，放大一个分区到最大大小，然后继续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，以指定如何根据节点的运行时信息对磁盘进行分区，如磁盘大小、驱动器数量、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 `#Dynamic` 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 分区部分。

```
#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

调配模板定义了卫星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，进入到 `Hosts > Templates > Provisioning Templates` 查看它们。您可以创建模板或克隆模板并编辑克隆。如需模板的帮助，请导航到 `Hosts > Templates > Provisioning Templates > Create Template > Help`。

红帽支持的模板由 Red Hat 图标表示。

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

如果您克隆了一个支持的模板，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模板编写 参考](#)。

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，[请参阅管理 Red Hat Satellite 中的创建机构调试证书](#)。

您可以在 Satellite 服务器和 Git 存储库或本地目录之间同步模板。如需更多信息，[请参阅管理主机中的同步模板 存储库](#)。

要查看应用到模板的更改历史记录，请导航到 Hosts > Templates > Provisioning Templates，选择一个模板，然后点击 History。单击 Revert，以通过之前的版本覆盖内容。您还可以恢复到之前的更改。点击 Show Diff 查看特定更改的信息：

- Template Diff 选项卡显示置备模板的正文中的更改。
- Details 选项卡显示模板描述中的更改。
- History 选项卡显示更改更改模板和日期的用户。

2.12. 置备模板的类型

有不同的置备模板：

置备

调配过程的主要模板。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，该模板部署到与子网关联的模板胶囊，以确保主机使用带有正确内核选项的安装程序。对于 BIOS 调配，请选择 PXELinux 模板。对于 UEFI 置备，请选择 PXEGrub2。

finish

当主置备过程完成后，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，这有时被称为 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，用于轻松部署。

当完成脚本成功退出时，使用返回代码 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，而是用于操作系统安装的后配置，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

用于接受自定义数据的供应商后配置脚本，也称为 seed 数据。您可以使用 user_data 模板仅在云或虚拟环境中调配虚拟机。此模板不需要 Satellite 可以访问主机；云或虚拟化平台负责将数据传送到镜像。

确保要置备的镜像有软件来读取安装的数据，并在引导时设置为启动数据。例如，需要 YAML 输入的 cloud-init 或 ignition，这需要 JSON 输入。

cloud_init

某些环境（如 VMWare）不支持自定义数据，或者有自己的数据格式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，该插件会尝试通过 HTTP 或 HTTPS 直接从 Satellite 下载模板。这种技术可以在任何环境中使用，最好是虚拟化。

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软件来读取安装的数据，并在引导时设置为启动数据。
- 调配的主机可以从与主机的调配接口 IP 相匹配的 IP 地址访问 Satellite。请注意，cloud-init 无法在 NAT 后面工作。

Bootdisk

用于无 PXE 引导方法的模板。

内核执行(kexec)

用于无 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。技术预览功能不受红帽订阅服务级别协议(SLA)完全支持，且功能可能并不完善，且不适用于生产环境。但是，这些功能提供对即将推出的产品创新的早期访问，使客户可以在开发过程中测试并提供反馈。

脚本

默认情况下不使用的任意脚本，但对自定义任务很有用。

ZTP

Zero Touch Provisioning 模板。

POAP

PowerOn 自动配置模板。

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

调配模板定义了卫星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置备模板。

流程

1. 在 Satellite Web UI 中，进入到 Hosts > Templates > Provisioning Templates 并点 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的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 创建模板前，创建一个包含模板的纯文本文件。这个示例使用 ~/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 \
--file ~/my-template \
--locations "My_Location" \
```



```
--name "My_Provisioning_Template" \
--organizations "My_Organization" \
--type provision
```

2.14. 克隆置备模板

调配模板定义了卫星服务器在主机上安装操作系统的方式。使用这个流程克隆模板，并将更新添加到克隆中。

流程

1. 在 Satellite Web UI 中，进入到 Hosts > Templates > Provisioning Templates。
2. 查找您要使用的模板。
3. 单击 Clone 以复制该模板。
4. 在 Name 字段中输入置备模板的名称。
5. 选中 Default 复选框，将模板设置为自动与新机构或位置关联。
6. 在 Template 编辑器 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。
7. 在 Audit Comment 字段中，输入对置备模板的更改摘要，以满足审计目的。
8. 点 Type 选项卡，如果模板是一个片断，请选择 Snippet 复选框。片段不是独立置备模板，而是可插入到其他置备模板中的置备模板的一部分。
9. 从 Type 列表中，选择模板的类型。例如，配置模板。
10. 点 关联 选项卡并从 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。
11. （可选）点 Add combination，从 Host Group 列表选择一个主机组，或者从 Environment 列表选择一个环境，将置备模板与主机组和环境关联。
12. 单击 Organizations 和 Locations 选项卡，向模板添加任何其他上下文。
13. 点 Submit 保存您的置备模板。

2.15. 创建自定义置备片断

自定义置备片段允许您在主机置备过程中执行自定义代码。您可以在置备过程前和/或之后运行代码。

先决条件

- 根据您的置备模板，存在多个自定义片断 hook，您可以使用它们包含自定义置备片断。确保首先检查您的置备模板，以验证您可以使用的自定义片断。

流程

1. 在 Satellite Web UI 中，进入到 Hosts > Templates > Provisioning Templates 并点 Create Template。

2. 在 Name 字段中输入自定义置备片断的名称。名称必须以支持包括自定义置备片断的置备模板的名称开始：
 - 将 'custom pre' 附加到置备模板的名称，以便在置备主机前运行代码。
 - 将 'custom post' 附加到置备模板的名称，以便在置备主机后运行代码。
3. 在 Type 选项卡中，选择 Snippet。
4. 点 Submit 以创建自定义置备片断。

CLI 过程

1. 在使用 CLI 创建模板前，创建一个包含自定义片断的纯文本文件。
2. 使用 **hammer** 创建模板：

```
# hammer template create \
--file "/path/to/My_Snippet" \
--locations "My_Location" \
--name "My_Template_Name_custom_pre" \ --organizations "_My_Organization" \
--type snippet
```

2.16. RED HAT ENTERPRISE LINUX 的自定义置备片断示例

在置备主机后，您可以使用自定义 **Post** 片断直接从置备模板内调用外部 API。

Kickstart 默认完成 Red Hat Enterprise Linux 的自定义后 示例

```
echo "Calling API to report successful host deployment"

yum install -y curl ca-certificates

curl -X POST \
-H "Content-Type: application/json" \
-d '{"name": "<%= @host.name %>", "operating_system": "<%= @host.operatingsystem.name %>",
"status": "provisioned,"}' \
"https://api.example.com/"
```

2.17. 将模板与操作系统关联

您可以将模板与 Satellite 中的操作系统关联。以下示例将置备模板添加到操作系统条目中。

流程

1. 在 Satellite Web UI 中，进入到 Hosts > Templates > Provisioning Templates。
2. 选择置备模板。
3. 在 关联 选项卡上，选择所有适用的操作系统。
4. 点 Submit 保存您的更改。

CLI 过程

1. 可选：查看所有模板：

```
# hammer template list
```

2. 可选：查看所有操作系统：

```
# hammer os list
```

3. 将模板与操作系统关联：

```
# hammer template add-operatingsystem \
--id My_Template_ID \
--operatingsystem-id My_Operating_System_ID
```

2.18. 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。

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

Red Hat Satellite 的默认安装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，进入到 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。此时将打开一个新窗口，其中包含 compute 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。

CLI 过程

1. 创建新的计算配置集：

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

2. 为 compute 配置集设置属性：

```
# hammer compute-profile values create \
```

```
--compute-attributes "flavor=m1.small,cpus=2,memory=4GB,cpu_mode=default \  
--compute-resource "My_Compute_Resource" \  
--compute-profile "My_Compute_Profile" \  
--volume size=40GB
```

3. 可选：要更新 compute 配置集的属性，请指定您要更改的属性。例如，要更改 CPU 和内存大小：

```
# hammer compute-profile values update \  
--compute-resource "My_Compute_Resource" \  
--compute-profile "My_Compute_Profile" \  
--attributes "cpus=2,memory=4GB" \  
--interface "type=network,bridge=br1,index=1" \  
--volume "size=40GB"
```

4. 可选：要更改 compute 配置集的名称，请使用 `--new-name` 属性：

```
# hammer compute-profile update \  
--name "My_Compute_Profile" \  
--new-name "My_New_Compute_Profile"
```

其他资源

- 有关使用 Hammer 创建计算配置集的更多信息，请输入 `hammer compute-profile --help`。

2.19. 为主机设置默认加密的 ROOT 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密码。

默认的 root 密码可由主机组继承，并由该组中的主机排序。

如果您更改密码并重新置备组中继承密码的主机，则密码将在主机上覆盖。

流程

1. 生成加密的密码：

```
$ python3 -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，进入到 Administer > Settings。
4. 在 Settings 页面中，选择 Provisioning 选项卡。
5. 在 Name 列中，导航到 Root 密码，然后单击以编辑。
6. 粘贴加密的密码，然后单击保存。

2.20. 使用 NOVNC 访问虚拟机

您可以使用浏览器访问 Satellite 所创建的虚拟机的 VNC 控制台。

Satellite 支持在以下虚拟平台上使用 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 类型为 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并不足以用于 noVNC。如需更多信息，请参阅管理 *Red Hat Satellite* 中的 [安装 Katello Root CA 证书](#)。

流程

1. 在受管主机上，将防火墙配置为允许端口 5900 到 5930 的 VNC 服务。

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

2. 在 Satellite Web UI 中，进入到 Infrastructure > Compute Resources 并选择计算资源的名称。
3. 在 Virtual Machines 选项卡中，选择虚拟机的名称。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胶囊服务器。胶囊服务器可以是卫星服务器上的集成胶囊，也可以是外部胶囊服务器。当主机位于隔离的网络上且无法直接连接到 Satellite 服务器，或者内容与胶囊服务器同步时，您可能希望从外部胶囊服务器调配主机。使用外部胶囊服务器进行调配可以在网络带宽上保存。

配置 Capsule 服务器有两个基本要求：

1. 配置网络服务,这包括：
 - 内容交付服务
 - 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。

以下说明具有类似的应用，用于配置管理特定网络的独立胶囊。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

事实描述主机报告的内存、操作系统版本或架构等方面。您可以在 Monitor > Facts 和 search host by factss 中找到事实，或者在模板中使用事实。

Satellite 从多个源收集事实：

- 订阅管理器
- ansible
- puppet

Satellite 是主机和网络接口的清单系统。对于 hypervisor 或容器主机，每个主机添加数千个接口，并每几分钟更新清单一次。对于报告的每个单独 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会导致 Satellite 缓慢且不可用。要优化各种操作的性能，最重要的是导入，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 模式为存储在 Satellite 和 Ignore 接口及匹配标识符选项的事实过滤和排除连接。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以根据您的要求使用它。

流程

1. 在 Satellite Web UI 中，进入到 Administer > Settings 并选择 Facts 选项卡。
2. 要过滤以特定名称开始的所有接口，例如 blu，请在带有匹配标识符选项的 Ignore 接口中添加 blu*。
3. 要防止数据库存储与特定名称开始的接口相关的事实，例如 blu，将 blu* 添加到 Satellite 选项中存储的事实的 Exclude 模式。

默认情况下，它包含与带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。这会完全过滤事实而不存储它们。

4. 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 **Administer > Settings > Facts > theExclude pattern for fact in Satellite** 选项中添加的过滤器匹配的所有事实。

5. 要从数据库中删除接口，请输入以下命令：

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

这个命令删除了与 **Administer > Settings > Facts > theIgnore interfaces with match identifier** 选项中添加的过滤器匹配的接口。

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给域。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，您仍必须创建和关联至少一个域。域包含在命名约定 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**。

子网

您必须将 Satellite 管理的每个主机分配到子网中。然后，Satellite 可以管理 IPv4 保留。如果没有保留集成，您仍然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 Satellite 之外的子网创建 DHCP 记录。在 Satellite 中，您可以使用 IP 地址管理(IPAM)通过以下选项之一管理 IP 地址：

- **DHCP:** DHCP Capsule 通过从范围的第一个地址开始查找下一个可用 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，或者配置了用于禁用连接的防火墙，Satellite 假定 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，并且没有外部创建的主机。Capsule DHCP 模块在短时间内保留提供的 IP 地址，以防止并发访问过程中发生冲突，因此 IP 范围内的一些 IP 地址可能会保持临时未使用。
- **内部 DB：**通过按顺序排除 Satellite 数据库的所有 IP 地址，从子网范围内查找下一个可用 IP 地址。数据的主要来源是数据库，而不是 DHCP 保留。当多个主机并行创建时，这个 IPAM 不安全；在这种情况下，使用 DHCP 或 Random DB IPAM。
- **Random DB：**通过随机排除 Satellite 数据库中的所有 IP 地址，从子网范围内查找下一个可用 IP 地址。数据的主要来源是数据库，而不是 DHCP 保留。此 IPAM 可以安全地与并发主机创建一起使用，因为 IP 地址以随机顺序返回，从而尽量减少冲突的可能性。
- **EUI-64：**扩展唯一标识符(EUI) 64 位 IPv6 地址生成，具体根据 RFC2373 获取，通过 48 位 MAC 地址获得。
- **外部 IPAM：**通过胶囊功能将 IPAM 提供给外部系统。Satellite 目前不附带任何外部 IPAM 实施，但几个插件正在开发中。
- **none：**必须手动输入每个主机的 IP 地址。

选项 DHCP、内部数据库和 Random DB 可能会导致外部创建的记录上的 DHCP 冲突。这些子网必须处于专用 Satellite 控制下。

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 服务器中为发现和调配的系统定义相同的 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 还会设置 `next-server` 和 `filename` DHCP 选项。

next-server 选项

`next-server` 选项提供 TFTP 服务器的 IP 地址，以便从中引导。默认情况下不设置此选项，必须为每个 TFTP 胶囊设置。您可以将 `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 加载程序设置为 `none` 时，Satellite 不会将 `next-server` 选项填充到 DHCP 记录中。

如果 `next-server` 选项未定义，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` 选项包含文件的完整路径，该文件在调配期间下载并执行。为主机或主机组选择的 PXE 加载程序定义要使用的文件名选项。当 PXE 加载程序设置为 `none` 时，Satellite 不会将 `filename` 选项填充到 DHCP 记录中。根据 PXE loader 选项，文件名更改如下：

PXE loader 选项	文件名条目	备注
PXELinux BIOS	pxelinux.0	
PXELinux UEFI	pxelinux.efi	
iPXE 链 BIOS	undionly.kpxe	
PXEGrub2 UEFI	grub2/grubx64.efi	X64 因架构而异

PXE loader 选项	文件名条目	备注
iPXE UEFI HTTP	http://capsule.example.com:8000/httpboot/ipxe-x64.efi	需要 httpboot 功能，并将文件名呈现为完整 URL，其中 Capsule.example.com 是 Satellite 中已知胶囊的主机名。
Grub2 UEFI HTTP	http://capsule.example.com:8000/httpboot/grub2/grubx64.efi	需要 httpboot 功能，并将文件名呈现为完整 URL，其中 Capsule.example.com 是 Satellite 中已知胶囊的主机名。

3.5. 对 SATELLITE 中的 DHCP 问题进行故障排除

Satellite 可以在内部或外部 DHCP 胶囊上管理 ISC DHCP 服务器。Satellite 可以列出、创建和删除 DHCP 保留和租用。但是，您可能会遇到很多问题。

没有同步 DHCP 记录

DHCP 编排期间发生错误时，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 Satellite 数据库中缺少的 DHCP 记录添加到 DHCP 服务器，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. 要预览要添加到 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，请使用 **perform=1** 参数输入以上命令来应用它们：

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

3. 要在 Satellite 和 DHCP 服务器中保持 DHCP 记录同步，您可以从 DHCP 服务器中删除不需要的 DHCP 记录。请注意，Satellite 假定所有管理的 DHCP 服务器都不包含第三方记录，因此此步骤可能会删除这些意外的记录。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，请使用 **perform=1** 参数输入以上命令来应用它们：

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

PXE loader 选项更改

当对现有主机更改 PXE loader 选项时，这会导致 DHCP 冲突。唯一的临时解决方案是覆盖 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 Capsule 无法读取所需的信息。

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

更改 DHCP Capsule 条目

Satellite 仅对分配给设置了 DHCP 胶囊的子网的主机管理 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。

如果您在不设置 DHCP Capsule 的情况下创建主机，并尝试设置 DHCP Capsule，这会导致 DHCP 冲突。

已删除 `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更多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，文件中的最后条目将具有优先权。租期文件中的组、子组和主机声明的处理方式相同。如果删除了租期，`{ deleted; }` 会附加到声明中。

3.6. 基于镜像的置备的先决条件

启动后配置方法

使用完成启动后配置脚本的镜像需要托管的 DHCP 服务器，如 Satellite 的集成胶囊或外部胶囊。必须使用与 DHCP Capsule 关联的子网创建主机，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。必须在 Satellite 中配置与镜像中配置对应的 SSH 凭据，以启用启动后配置。

在对使用启动后配置脚本的镜像引导的虚拟机进行故障排除时检查以下项目：

- 主机在 Satellite 服务器中分配有子网。
- 该子网在 Satellite 服务器中分配有 DHCP Capsule。
- 主机具有在 Satellite 服务器中分配的有效 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，与要部署的镜像关联。
- Satellite 服务器可以通过 SSH 密钥访问虚拟机。虚拟机需要它才能从 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。首选受管 DHCP Capsule。镜像必须将 `cloud-init` 服务配置为在系统启动时启动，并获取用于完成配置脚本或配置数据。

在对从镜像引导的虚拟机进行故障排除时检查以下项目，具体取决于镜像中包含的初始化脚本：

- 子网上有一个 DHCP 服务器。

- 虚拟机安装了并启用了 `cloud-init` 服务。

有关 `virtual-machine` 镜像中完成和 `cloud-init` 脚本不同支持级别的详情，请查看红帽知识库解决方案中支持的计算资源是什么，以及红帽客户门户网站中的 `cloud-init` 脚本。

3.7. 配置网络服务

有些调配方法使用 Capsule 服务器服务。例如，网络可能需要胶囊服务器来充当 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，以在同一个服务器上配置这些服务。

要在外部胶囊服务器上配置这些服务，请运行 `satellite-installer`。

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-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-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. 找到您配置的 Capsule 服务器：

```
# hammer capsule list
```

3. 刷新 Capsule 服务器的功能以查看更改：

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

4. 验证 Capsule 服务器上配置的服务：

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

3.7.1. 使用安装程序的多个子网或域

`satellite-installer` 选项仅允许单个 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。

对于每个额外子网或域，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** 来执行更改，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。把它设置为专用网络上主机的外部网关地址。

--foreman-proxy-dhcp-interface

设置用于侦听请求的 DHCP 服务的接口。将它设置为 **eth1**。

--foreman-proxy-dhcp-nameservers

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

--foreman-proxy-dhcp-range

用于 **Discovered** 和 **Unmanaged** 服务的以空格分隔的 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** 来查看与 DHCP 和其他胶囊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 ISC BIND 由安装程序管理时使用。把它设置为您的 DNS 递归器。

`--foreman-proxy-dns-interface`

设置要侦听 DNS 请求的接口。仅在 ISC BIND 由安装程序管理时使用。将它设置为 `eth1`。

`--foreman-proxy-dns-reverse`

DNS 反向区域名称。仅在 ISC BIND 由安装程序管理时使用。

`--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 ISC BIND 由安装程序管理时使用。

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

3.7.4. 用于网络配置的 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。确保使用胶囊的 IP 地址。

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

3.7.5. 通过 NAT 使用 TFTP 服务

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 设备文档。

在 Red Hat Enterprise Linux 7 中使用 NAT:

1. 在防火墙配置中允许 TFTP 服务：

```
# firewall-cmd --add-service=tftp
```

2. 使更改持久：

```
# firewall-cmd --runtime-to-permanent
```

在 Red Hat Enterprise Linux 6 中使用 NAT:

1. 配置防火墙以允许端口 69 上的 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胶囊服务器的信息。

检查现有域

Satellite 服务器可能已在 Satellite 服务器安装中创建相关域。将上下文切换到 **Any Organization** 和 **Any Location**, 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 会执行冲突 DNS 查找来验证主机名是否不在活跃使用中。这个检查针对以下 DNS 服务器之一运行 :

- 如果 **Administer > Settings > Query local nameserver** 设为 `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权威 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有效的配置。

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

流程

1. 在 Satellite Web UI 中, 进入到 **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. 点 Parameters 选项卡，并将任何域级别参数配置为应用到附加到此域的主机。例如，用户定义了要在模板中使用的布尔值或字符串参数。
5. 单击 Add Parameter，再填写 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 以保存更改。

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 Satellite 服务器上集成胶囊的 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，进入到 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，输入子网的主 DNS。
9. 在二级 DNS 服务器中，输入子网的辅助 DNS。
10. 从 IPAM 列表中，选择您要用于 IP 地址管理(IPAM)的方法。有关 IPAM 的更多信息，请参阅 [第 3 章 配置网络](#)。

11. 输入您选择的 IPAM 方法的信息。点 Remote Execution 选项卡，然后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. 单击 Capsules 选项卡，再选择应用到子网中每个服务的胶囊，包括 DHCP、TFTP 和反向 DNS 服务。
14. 点 Parameters 选项卡，并将任何子网级别参数配置为应用到附加到此子网的主机。例如，用户定义了要在模板中使用的布尔值或字符串参数。
15. 单击位置选项卡，再选择使用此胶囊的位置。
16. 单击 Organizations 选项卡，再选择使用此胶囊的组织。
17. 单击 Submit 以保存子网信息。

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，这是卫星服务器中集成胶囊的 ID。

第 4 章 使用 INFOBLOX 作为 DHCP 和 DNS 供应商

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

支持的 Infoblox 版本为 NIOS 8.0 或更高版本。

4.1. INFOBLOX 限制

所有 DHCP 和 DNS 记录都只能在单个网络或 DNS 视图进行管理。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。
- 确保具有名称为 **DHCP Admin** 和 **DNS Admin** 的 Infoblox 管理角色。
- 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，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"
  }
]

```

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

如果要同时使用 DHCP 和 DNS Infoblox 模块，请只使用 `fixedaddress` 记录类型配置 DHCP Infoblox 模块。主机记录类型会导致 DNS 冲突且不被支持。

如果您使用主机记录类型配置 DHCP Infoblox 模块，则必须同时取消设置 Infoblox 管理的子网上的 DNS Capsule 和 Reverse DNS Capsule 选项，因为 Infoblox 会自行管理 DNS。使用主机记录类型会导致创建冲突，且无法重命名 Satellite 中的主机。

流程

1. 在 Capsule 上输入以下命令：

```

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

```

2. 可选：在 Satellite Web UI 中，进入到 Infrastructure > Capsules，选择带有 DHCP Infoblox 模块的 Capsule，并确保列出了 dhcp 功能。
3. 在 Satellite Web UI 中，进入到 Infrastructure > Subnets。
4. 对于通过 Infoblox 管理的所有子网，请确保子网的 IP 地址管理(IPAM)方法设置为 DHCP。

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。

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-username admin \  
--foreman-proxy-plugin-dns-infoblox-password infoblox \  
--foreman-proxy-plugin-dns-infoblox-dns-view default
```

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 Infoblox DNS 视图。

2. 可选：在 Satellite Web UI 中，进入到 Infrastructure > Capsules，选择带有 Infoblox DNS 模块的 Capsule，并确保列出了 dns 功能。
3. 在 Satellite Web UI 中，进入到 Infrastructure > Domains。
4. 对于通过 Infoblox 管理的所有域，请确保为这些域设置了 DNS 代理。
5. 在 Satellite Web UI 中，进入到 Infrastructure > Subnets。
6. 对于通过 Infoblox 管理的所有子网，请确保为这些子网设置了 DNS Capsule 和 Reverse DNS Capsule。

第 5 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 地址和 Satellite 服务器使用 PXE 引导过程来置备主机。

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 的置备

使用 Satellite 服务器生成的引导磁盘镜像调配新主机。

BIOS 和 UEFI 支持

使用 Red Hat Satellite，您可以执行基于 BIOS 和 UEFI 的 PXE 置备。BIOS 和 UEFI 接口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 调配中，PXE 加载程序选项定义调配期间要使用的 DHCP 文件名选项。对于 BIOS 系统，使用 PXELinux BIOS 选项启用调配的节点，以通过 TFTP 下载 pxelinux.0 文件。对于 UEFI 系统，使用 PXEGrub2 UEFI 选项使 TFTP 客户端能够下载 grub2/grubx64.efi 文件，或使用 PXEGrub2 UEFI HTTP 选项启用 UEFI HTTP 客户端使用 HTTP 引导功能从 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 服务器上为这两者部署配置文件，以便您可以轻松地在 PXE 加载程序间切换。

5.1. 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胶囊服务器。对于无人值守配置和基于发现的调配，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](#)。

有关发现服务的更多信息，[第 7 章 配置发现服务](#)。
- 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。如需更多信息，请参阅 [管理内容中的同步存储库](#)。
- 为主机注册提供激活码。如需更多信息，[请参阅管理内容中的创建激活码](#)。

有关无人值守和无 PXE 置备的安全令牌详情，请参考 [第 5.2 节 “配置安全令牌有效期持续时间”](#)。

5.2. 配置安全令牌有效期持续时间

在执行任何类型的调配时，Satellite 会自动生成唯一的令牌，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，令牌对 360 分钟有效。当您调配主机时，请确保在此时间范围内重新引导主机。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失败。

流程

1. 在 Satellite Web UI 中，进入到 Administer > Settings，然后点 Provisioning 选项卡。
2. 找到 Token duration 选项并点编辑图标并编辑持续时间，或者输入 0 来禁用令牌生成。如果禁用令牌生成，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。

5.3. 使用无人值守调配创建主机

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过程中使用最少的交互。

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

流程

1. 在 Satellite Web UI 中，进入到 Hosts > Create Host。
2. 在 Name 字段中输入主机的名称。
3. 可选：点 Organization 选项卡并更改机构上下文以匹配您的要求。
4. 可选：点 Location 选项卡并更改位置上下文以匹配您的要求。
5. 从 Host Group 列表中，选择要为其分配主机的主机组。该主机组将填充表单。
6. 单击 Interfaces 选项卡，然后在主机的接口上点 Edit。
7. 验证字段是否填充了值。特别是：
 - Satellite 会自动为新主机分配 IP 地址。
 - 在 MAC 地址字段中，输入主机的置备接口的 MAC 地址。这样可确保在 PXE 引导过程中识别主机。
 - Host 选项卡中的 Name 变为 DNS 名称。
 - 确保 Satellite 会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。
8. 单击 OK 以保存。要添加另一个接口，请单击 Add Interface。您只能为 Provision 和 Primary 选择一个接口。
9. 单击 Operating System 选项卡，然后验证所有字段是否包含值。确认操作系统的每个方面。
10. 可选：点 Provisioning 模板中的 Resolve 检查新主机是否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，并确保存在提供激活密钥的参数。如果没有，添加激活码。
12. 点 Submit 以保存主机详情。
有关网络接口的更多信息，请参阅 [管理主机中的添加网络接口](#)。
https://access.redhat.com/documentation/zh-cn/red_hat_satellite/6.15/html-single/managing_hosts/index#Adding_Network_Interfaces_managing-hosts

这将创建主机条目和相关调配设置。这还包括创建用于 PXE 引导裸机主机所需的目录和文件。如果您启动物理主机并将其引导模式设置为 PXE，则主机会检测到 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，主机还会使用激活码注册到 Satellite 服务器，并从 Satellite Client 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 \
--build true \
--enabled true \
--hostgroup "My_Host_Group" \
--location "My_Location" \
--mac "My_MAC_Address" \
--managed true \
--name "My_Host_Name" \
--organization "My_Organization"
```

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
```

5.4. 使用无 PXE 的调配创建主机

有些硬件不提供 PXE 引导接口。在 Satellite 中，您可以在不进行 PXE 引导的情况下置备主机。这也称为无 PXE 的调配，并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，可在没有基于 PXE 的服务（如 DHCP 和 TFTP）的情况下运行。如需更多信息，请参阅 [第 7.7 节“实施无 PXE 的发现”](#)。

引导 ISO 类型

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，此镜像很有用。自定义模板仍然从 Satellite 服务器下载。

子网镜像

与特定主机关联的引导 ISO。ISO 将主机的 MAC 地址发送到 Capsule 服务器，这与主机条目匹配。该镜像不存储 IP 地址详情，且需要访问网络上的 DHCP 服务器来引导。此镜像对于在同一子网上具有置备 NIC 的所有主机都通用。该镜像基于 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，可用于大多数网卡。在使用子网镜像时，请参阅 ipxe.org 上支持的硬件，以了解应该使用基于 iPXE 的引导磁盘的网卡驱动程序列表。

完整主机镜像包含 调配令牌，因此生成的镜像的生命周期有限。有关配置安全令牌的详情，请参考第 5.2 节“配置安全令牌有效期持续时间”。

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

流程

1. 在 Satellite Web UI 中，进入到 Hosts > Create Host。
2. 在 Name 字段中输入主机的名称。
3. 可选：点 Organization 选项卡并更改机构上下文以匹配您的要求。
4. 可选：点 Location 选项卡并更改位置上下文以匹配您的要求。
5. 从 Host Group 列表中，选择要为其分配主机的主机组。该主机组将填充表单。
6. 单击 Interfaces 选项卡，然后在主机的接口上点 Edit。
7. 验证字段是否填充了值。特别是：
 - Satellite 会自动为新主机分配 IP 地址。
 - 在 MAC 地址字段中，输入主机的置备接口的 MAC 地址。这样可确保在 PXE 引导过程中识别主机。
 - Host 选项卡中的 Name 变为 DNS 名称。
 - 确保 Satellite 会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。
8. 单击 OK 以保存。要添加另一个接口，请单击 Add Interface。您只能为 Provision 和 Primary 选择一个接口。
9. 单击 Operating System 选项卡，然后验证所有字段是否包含值。确认操作系统的每个方面。
10. 点 Provisioning Templates 中的 Resolve 检查新主机是否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，并确保存在提供激活密钥的参数。如果没有，添加激活码。
12. 点 Submit 以保存主机详情。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，在主机详情页面中，点垂直 ellipsis 并选择 Full host 'My_Host_Name' image。
 - 对于子网镜像，请导航到 Infrastructure > Subnets，单击所需子网的 Actions 列中的下拉菜单，然后选择子网通用镜像。
14. 如果需要，使用 dd 工具或 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，主机还会使用激活码注册到 Satellite 服务器，并从 Satellite Client 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 \
--build true \
--enabled true \
--hostgroup "My_Host_Group" \
--location "My_Location" \
--mac "My_MAC_Address" \
--managed true \
--name "My_Host_Name" \
--organization "My_Organization"
```

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：

- 对于完整主机镜像：

```
# hammer bootdisk host \
--full true \
--host My_Host_Name
```

- 对于子网镜像：

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

这会创建一个引导 ISO 供您的主机使用。

4. 如果需要，使用 **dd** 工具或 **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，主机还会使用激活码注册到 Satellite 服务器，并从 Satellite Client 6 存储库安装必要的配置和管理工具。

5.5. 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 [Overview](#)、[概念和部署注意事项](#) 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 \
--foreman-proxy-http true \
--foreman-proxy-httpboot true \
--foreman-proxy-tftp true
```

3. 确保 Capsule 具有 TFTP 和 HTTPBoot 功能。在 Satellite Web UI 中，进入到 Infrastructure > Capsules 并点 Capsule 来查看可识别的功能列表。如果缺少任何功能，请单击 Refresh Features。
4. 确保胶囊与 provisioning 子网关联。在 Satellite Web UI 中，进入到 Infrastructure > Subnets > Edit Subnet > Capsules，并为 TFTP 和 HTTPBoot 选项选择 Capsule。
5. 单击 OK 以保存。
6. 在 Satellite Web UI 中，进入到 Hosts > Create Host。
7. 在 Name 字段中输入主机的名称。
8. 可选：点 Organization 选项卡并更改机构上下文以匹配您的要求。
9. 可选：点 Location 选项卡并更改位置上下文以匹配您的要求。
10. 从 Host Group 列表中，选择要为其分配主机的主机组。该主机组将填充表单。
11. 单击 Interfaces 选项卡，然后在主机的接口上点 Edit。
12. 验证字段是否填充了值。特别是：
 - Satellite 会自动为新主机分配 IP 地址。
 - 在 MAC 地址字段中，输入主机的置备接口的 MAC 地址。这样可确保在 PXE 引导过程中识别主机。
 - Host 选项卡中的 Name 变为 DNS 名称。
 - 确保 Satellite 会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。
13. 单击 OK 以保存。要添加另一个接口，请单击 Add Interface。您只能为 Provision 和 Primary 选择一个接口。
14. 单击 Operating System 选项卡，然后验证所有字段是否包含值。确认操作系统的每个方面。
15. 从 PXE Loader 列表中，选择 Grub2 UEFI HTTP。
16. 可选：点 Provisioning 模板中的 Resolve 检查新主机是否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”。
17. 单击 Parameters 选项卡，并确保存在提供激活密钥的参数。如果没有，添加激活码。

18. 点 **Submit** 以保存主机详情。
有关网络接口的更多信息，请参阅管理主机中的 **添加网络接口**。
https://access.redhat.com/documentation/zh-cn/red_hat_satellite/6.15/html-single/managing_hosts/index#Adding_Network_Interfaces_managing-hosts
19. 将主机设置为从网络在 UEFI 模式中引导。
20. 启动主机。
21. 从引导菜单中，选择 **Kickstart default PXEGrub2**。

这将创建主机条目和相关调配设置。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，接收带有 Kickstart 树的 Capsule 的 HTTP 端点并安装操作系统。

安装完成后，主机还会使用激活码注册到 Satellite 服务器，并从 Satellite Client 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 \
--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 \
--build true \
--enabled true \
--hostgroup "My_Host_Group" \
--location "My_Location" \
--mac "My_MAC_Address" \
--managed true \
--name "My_Host_Name" \
--organization "My_Organization" \
--pxe-loader "Grub2 UEFI HTTP"
```

4. 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
```

5. 将主机设置为从网络在 UEFI 模式中引导。
6. 启动主机。

7. 从引导菜单中，选择 Kickstart default PXEGrub2。

这将创建主机条目和相关调配设置。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，接收带有 Kickstart 树的 Capsule 的 HTTP 端点并安装操作系统。

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

5.6. 在置备过程中部署 SSH 密钥

使用这个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 中的管理用户 SSH 密钥](#)。

流程

1. 在 Satellite Web UI 中，进入到 Hosts > Templates > 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，单击 Template 选项卡。
4. 在 Template editor 字段中，将 `create_users` 片断添加到 `%post` 部分：

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

5. 选中 Default 复选框。
6. 点 关联 选项卡。
7. 从 Application Operating Systems 列表中，选择一个操作系统。
8. 点 Submit 以保存置备模板。
9. 创建与置备模板关联的主机，或使用与修改模板关联的操作系统重建主机。如需更多信息，请参阅管理主机中的 [创建主机](#)。

在置备过程中执行 `create_users` 代码片段时，用户拥有的 SSH 密钥会自动添加。您可以将拥有的设置为个人用户或用户组。如果将 Owned 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 6 章 使用 IPXE 减少置备时间

iPXE 是一个开源网络引导固件。它提供了一个功能增强的完整 PXE 实现，比如从 HTTP 服务器引导。有关 iPXE 的更多信息，请参阅 [iPXE 网站](#)。

如果以下限制会阻止使用 PXE，您可以使用 iPXE：

- 具有非受管 DHCP 服务器的网络。
- 因防火墙限制而无法访问的 PXE 服务。
- 由于一个不可靠的基于 TFTP UDP 的协议，例如低带宽网络。

6.1. 使用 IPXE 的先决条件

在以下情况下，您可以使用 iPXE 引导虚拟机：

- 您的虚拟机运行在使用 iPXE 作为主固件的虚拟机监控程序上。
- 您的虚拟机处于 BIOS 模式。在这种情况下，您可以将 PXELinux 配置为串联 iPXE 并使用 HTTP 协议引导。

对于使用 HTTP 在 UEFI 模式中引导虚拟机，您可以遵循 [第 5.5 节“使用 UEFI HTTP 引导置备创建主机”](#)。

支持性

红帽官方不支持 Red Hat Satellite 中的 iPXE。如需更多信息，请参阅 [红帽知识库中的 Satellite 6 中支持的架构和 kickstart 场景](#)。

主机要求

- 置备接口的 MAC 地址与主机配置匹配。
- 主机 的置备接口具有有效的 DHCP 保留。
- NIC 能够进行 PXE 引导。如需更多信息，请参阅 [ipxe.org 上支持的硬件](#) 硬件，以了解应该使用基于 iPXE 的引导磁盘的硬件驱动程序列表。
- NIC 与 iPXE 兼容。

6.2. 配置 IPXE 环境

在您要用于 iPXE 置备的所有胶囊上配置 iPXE 环境。



重要

在 Red Hat Enterprise Linux 中，不支持 iPXE 的安全相关功能，并且 iPXE 二进制文件在没有安全功能的情况下构建。因此，您只能使用 HTTP 而不是 HTTPS。如需更多信息，请参阅 [iPXE 中的 Red Hat Enterprise Linux HTTPS 支持](#)。

先决条件

- 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](#)。

流程

1. 在 Capsule 上启用 TFTP 和 HTTPboot 服务：

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

2. 在 Capsule 上安装 ipxe-bootimgs 软件包：

```
# satellite-maintain packages install ipxe-bootimgs
```

3. 将 iPXE 固件复制到 TFTP 目录中。

- 使用 Linux 内核标头复制 iPXE 固件：

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

- 复制 UNDI iPXE 固件：

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

4. 修正 SELinux 文件上下文：

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

5. 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 服务器引导，请在 Capsule 服务器上运行以下命令：

```
# satellite-installer --foreman-proxy-dhcp-ipxe-bootstrap true
```

6.3. 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。如果使用这样的虚拟机监控程序，您可以在没有 TFTP 和 PXELinux 的情况下引导虚拟机。

引导虚拟机有以下 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 Capsule 加载 iPXE bootstrap 模板。
4. iPXE 从 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

- 您的管理程序必须支持 iPXE。以下虚拟化虚拟机监控程序支持 iPXE：
 - libvirt
 - Red Hat Virtualization（已弃用）
- 您已配置了 iPXE 环境。如需更多信息，请参阅 [第 6.2 节“配置 iPXE 环境”](#)。



注意

您可以使用 Satellite 中提供的原始模板，如下所述。如果您需要修改原始模板，请克隆模板，编辑克隆，并关联克隆而不是原始模板。如需更多信息，请参阅 [第 2.14 节“克隆置备模板”](#)。

流程

1. 在 Satellite Web UI 中，进入到 Hosts > Templates > Provisioning Templates。
2. 搜索 Kickstart 默认 iPXE 模板。
3. 点模板的名称。
4. 点 关联 选项卡，然后选择主机使用的操作系统。
5. 单击 Locations 选项卡，再添加主机所在的位置。
6. 单击 Organizations 选项卡，再添加主机所属的组织。
7. 点 Submit 以保存更改。
8. 在 Satellite Web UI 中，进入到 Hosts > Operating systems 并选择主机的操作系统。
9. 点 Templates 选项卡。
10. 从 iPXE 模板列表中，选择 Kickstart 默认 iPXE 模板。
11. 点 Submit 以保存更改。
12. 在 Satellite Web UI 中，进入到 Hosts > All Hosts。
13. 在 Hosts 页面中，选择要使用的主机。
14. 选择 Operating System 选项卡。
15. 将 PXE Loader 设置为 iPXE 嵌入式。
16. 选择 Templates 选项卡。
17. 在 Provisioning Templates 中，单击 Resolve，再验证 iPXE 模板 是否解析为所需的模板。
18. 点 Submit 以保存主机设置。

6.4. 来自 PXELINUX 的 CHAINBOOTING IPXE

您可以将 iPXE 设置为使用内置驱动程序进行网络通信(ipxe.lkrn)或通用网络设备接口(UNDI) (undionly-ipxe.0)。您可以选择根据网络硬件功能和 iPXE 驱动程序可用性加载任一文件。

UNDI 是一个最小的 UDP/IP 堆栈，可实施 TFTP 客户端。但是，UNDI 无法支持 HTTP 等其他协议。要将 HTTP 与 iPXE 搭配使用，请使用带有内置驱动程序的 iPXE 构建(ipxe.lkrn)。

Chainbooting iPXE 有以下工作流：

1. 主机开机。
2. PXE 驱动程序使用 DHCP 检索网络凭据。
3. PXE 驱动程序使用 TFTP 检索 PXELinux 固件 pxelinux.0。
4. pxelinux 在 TFTP 服务器上搜索配置文件。
5. pxelinux chainloads iPXE ipxe.lkrn 或 undionly-ipxe.0。
6. iPXE 再次使用 DHCP 检索网络凭据，包括 HTTP URL。
7. iPXE 链从模板胶囊加载 iPXE 模板。
8. iPXE 加载安装程序的内核和初始 RAM 磁盘。

先决条件

- 您已配置了 iPXE 环境。如需更多信息，请参阅 [第 6.2 节“配置 iPXE 环境”](#)。



注意

您可以使用 Satellite 中提供的原始模板，如下所述。如果您需要修改原始模板，请克隆模板，编辑克隆，并关联克隆而不是原始模板。如需更多信息，请参阅 [第 2.14 节“克隆置备模板”](#)。

流程

1. 在 Satellite Web UI 中，进入到 Hosts > Templates > Provisioning Templates。
2. 搜索所需的 PXELinux 模板：
 - pxelinux 链 iPXE 使用 ipxe.lkrn
 - pxelinux 链 iPXE UNDI 以使用 undionly-ipxe.0
3. 点您要使用的模板的名称。
4. 点 关联 选项卡，然后选择主机使用的操作系统。
5. 单击 Locations 选项卡，再添加主机所在的位置。
6. 单击 Organizations 选项卡，再添加主机所属的组织。
7. 点 Submit 以保存更改。
8. 在 Provisioning Templates 页面中，搜索 Kickstart 默认 iPXE 模板。
9. 点模板的名称。

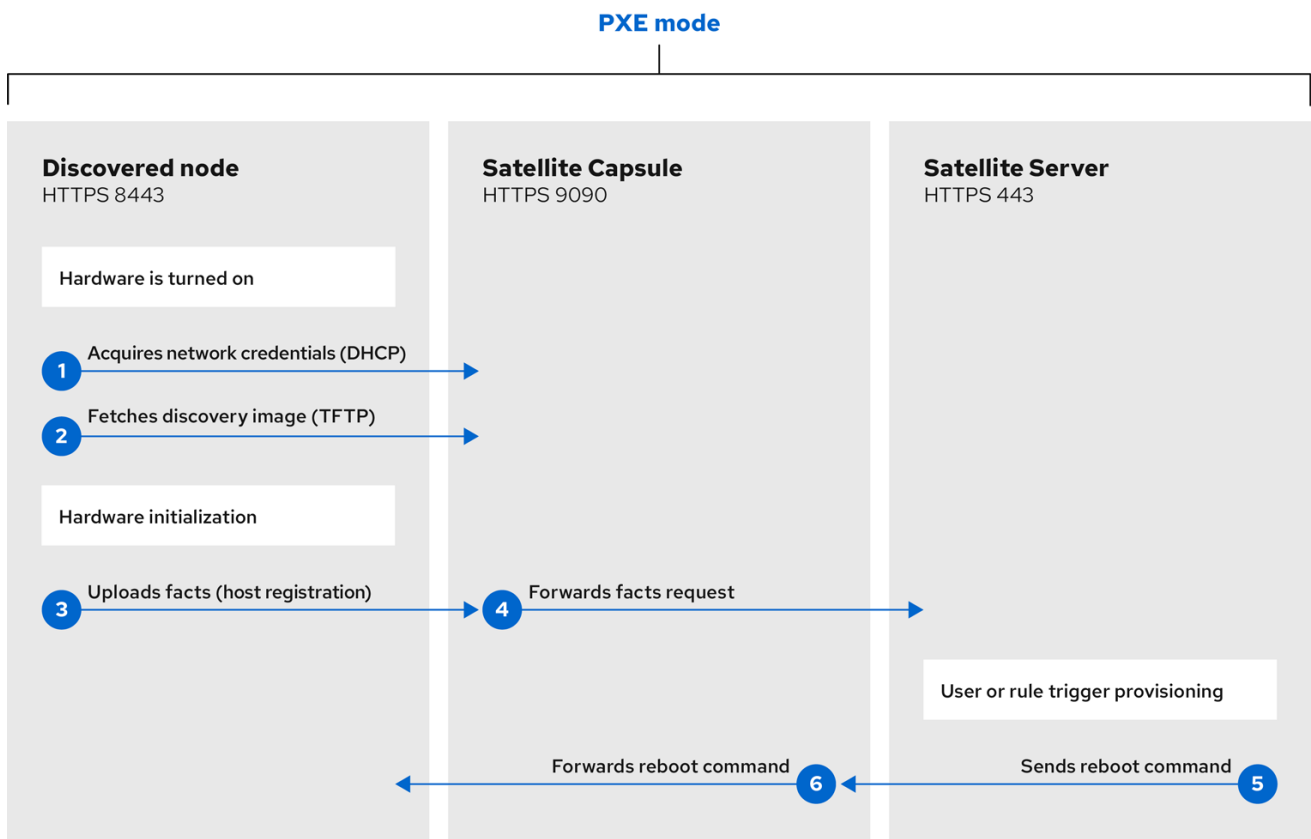
10. 点 **关联** 选项卡，并将模板与您的主机使用的操作系统相关联。
11. 单击 **Locations** 选项卡，再添加主机所在的位置。
12. 单击 **Organizations** 选项卡，再添加主机所属的组织。
13. 点 **Submit** 以保存更改。
14. 在 Satellite Web UI 中，进入到 **Hosts > Operating systems** 并选择主机的操作系统。
15. 点 **Templates** 选项卡。
16. 从 **PXELinux** 模板列表中，选择要使用的模板。
17. 从 **iPXE** 模板列表中，选择 **Kickstart** 默认 **iPXE** 模板。
18. 点 **Submit** 以保存更改。
19. 在 Satellite Web UI 中，进入到 **Configure > Host Groups**，再选择您要配置的主机组。
20. 选择 **Operating System** 选项卡。
21. 选择 **架构和操作系统**。
22. 设置 **PXE Loader** :
 - 从 **PXELinuxLinux** 选择 **PXELinux BIOS** 以串联 **iPXE (ipxe.lkrn)**。
 - 选择 **iPXE Chain BIOS** 以直接加载 **undionly-ipxe.0**。

第 7 章 配置发现服务

Red Hat Satellite 可以检测不在 Satellite 清单中的网络上的主机。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**。如果您选择发现来引导发现镜像，几分钟后 Discovery 镜像完成引导，并显示状态屏幕。

Discovery 服务在 Satellite 服务器上默认启用。但是，全局模板的默认设置是从本地硬盘驱动器引导的。要更改默认设置，在 Satellite Web UI 中进入到 **Administer > Settings**，然后点击 **Provisioning** 选项卡。找到 **Default PXE global template** 条目行，然后在 **Value** 列中输入 **discovery**。然后构建 PXE 默认配置。进入 **Hosts > Templates > Provisioning Templates** 并点 **Build PXE Default**。Satellite 将 PXE 配置分发到任何 TFTP 胶囊。



278_Satellite_0922

确保您计划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 租期池进行干扰。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 10.1.0.0 到 10.1.127.255 中，可以为租期分配 10.1.128.0 到 10.1.255.254 以进行保留。

因为网络接口名称不一定在 Red Hat Enterprise Linux 的主版本之间始终相同，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（对于其他硬件或虚拟机）禁用新的命名方案，可以通过 `net.ifnames=1` 完全关闭新的命名方案。



注意

Satellite 提供的 Foreman 发现镜像基于 Red Hat Enterprise Linux 8。

流程

1. 在 Satellite 服务器上安装发现：

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

2. 安装 **foreman-discovery-image**：

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

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

安装完成后，您可以通过进入到 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。

流程

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

具有可发现主机的所有子网都需要选择适当的胶囊服务器来提供发现服务。

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

7.2. 置备模板 PXELINUX DISCOVERY 片断

对于 BIOS 置备，Host > Templates & gt; Provisioning Templates 窗口中的 PXELinux 全局默认模板包含片断 `pxelinux_discovery`。这个片段有以下行：

```
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 Discovery 镜像和 ramdisk。APPEND 选项包含一个 `proxy.url` 参数，其参数 `foreman_server_url` 宏。此宏解析为 Satellite 服务器的完整 URL。

对于 UEFI 置备，Host > Templates > Provisioning Templates 窗口中的 PXEgrub2 全局默认模板包含 snippet `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 服务器 FQDN。

全局模板可在 Satellite 服务器上提供，并且启用了 TFTP 功能的所有胶囊。

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机构和位置：

1. 如果发现的主机使用 Satellite 中定义的子网，主机将使用与子网关联的第一个机构和位置。
2. 如果在全局设置中配置了默认位置和机构，则发现的主机将放置在这个机构和位置。要配置这些设置，请导航到 Administer > Settings > Discovery，并为 Discovery 机构和 Discovery 位置设置选择值。确保发现的主机的子网也属于所选组织和位置，否则 Satellite 会拒绝出于安全原因设置它。
3. 如果满足上述任何条件，Satellite 会按照名称分配第一个机构和位置。

您可以通过在 Discovered Hosts 页面上使用批量操作来手动更改机构或位置。选择要修改的发现的宿主，并从 Select Action 菜单中选择 Assign Organization 或 Assign Location。

7.4. 发现模板和片断设置

要使用发现服务，您必须配置置备设置，将 Discovery 设置为默认服务，并设置要使用的模板。

将 Discovery 服务设置为默认值

对于 BIOS 和 UEFI，要将 Discovery 服务设置为默认服务，该服务为当前 Satellite 清单中不存在的宿主引导，请完成以下步骤：

1. 在 Satellite Web UI 中，进入到 Administer > Settings 并点 Provisioning 选项卡。
2. 对于 Default PXE global template 条目，在 Value 列中输入 `discovery`。

要使用模板，在 Satellite Web UI 中进入到 Administer > Settings 并点 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。

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，进入到 Hosts > Templates > Provisioning Templates，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其他置备胶囊的 IP 地址或 FQDN，但请确保附加端口号，如 9090。如果在 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，并且没有可用的变量或宏来呈现胶囊的主机名。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，或者对适当的子网使用通用 DNS 别名。

标记的 VLAN 置备

如果要使用标记的 VLAN 置备，并且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 调配类似的置备过程。主要区别在于，除了手动输入主机的 MAC 地址外，您可以从发现的主机列表中选择要调配的主机。

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。如需更多信息，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。如需更多信息，请参阅 [管理内容中的 同步存储库](#)。
- 为主机注册提供激活码。如需更多信息，请参阅 [管理内容中的 创建激活码](#)。

有关安全令牌的详情，请参考 [第 5.2 节 “配置安全令牌有效期持续时间”](#)。

流程

1. 在 Satellite Web UI 中，进入到 Hosts > Discovered host。选择要使用的主机，再单击列表右侧的 Provision to。
2. 从以下两个选项之一中选择：
 - 要从主机组调配主机，请选择主机组、组织和位置，然后单击 Create Host。
 - 若要置备带有进一步自定义的主机，请单击 Customize Host，再输入您要为新主机指定的附加详情。
3. 验证字段是否填充了值。特别是：
 - Host 选项卡中的 Name 变为 DNS 名称。
 - Satellite 服务器会自动为新主机分配 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. 单击 Operating System 选项卡，然后验证所有字段是否包含值。确认操作系统的每个方面。
6. 点 Provisioning 模板中的 Resolve 来检查新主机是否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
 - Provision Template: Satellite Kickstart Default
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 以保存主机详情。

主机调配完成后，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. 识别要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调配。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 provisioning 设置创建主机条目。发现镜像会自动重置主机，使其可以引导到 PXE。主机在 Satellite 服务器集成的胶囊上检测 DHCP 服务，并开始

安装操作系统。其余过程与 [第 5.3 节 “使用无人值守调配创建主机”](#) 中描述的普通 PXE 工作流程相同。

7.6. 创建发现规则

作为为发现的主机自动执行调配流程的方法，Satellite 提供了创建发现规则的功能。这些规则定义了发现的主机如何根据分配的机组自动调配自身。例如，您可以自动将 CPU 数高的主机调配为虚拟机监控程序。同样，您可以将具有大型硬盘的主机调配为存储服务器。

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；所有系统都使用在发现过程中检测到的 NIC 配置来置备。但是，您可以使用脚本或在稍后阶段使用配置管理在 kickstart scriptlet 中设置 NIC。

流程

1. 在 Satellite Web UI 中，进入到 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要置备主机的规则。此字段提供您输入的值的建议，并允许操作者用于多个规则。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，以确定多个主机的主机名。这使用了与置备模板相同的 ERB 语法。主机名可以将 `@host` 属性用于特定于主机的值，以及 `rand` 宏作为随机数字，或 `sequence_hostgroup_param_next` 宏来递增值。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，且不包含下划线或点。良好的方法是使用 **Factor** 提供的唯一信息，如 **MAC 地址**、**BIOS** 或 **串行 ID**。

6. 在 **Hosts limit** 字段中，输入您可以使用规则置备的主机的最大数量。输入 **0** 代表无限。
7. 在 **Priority** 字段中，输入数字来设置规则的优先级高于其他规则的优先级。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调配上下文，请单击 **组织和位置** 选项卡，再选择要使用的上下文。
10. 点 **Submit** 保存您的规则。
11. 在 **Satellite Web UI** 中，进入到 **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** 以自动调配单个主机。
 - 在窗口右上角，单击 **Auto-Provision All** 以自动调配所有主机。

CLI 过程

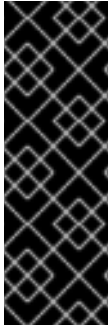
1. 使用 **hammer discovery-rule create** 命令创建规则：


```
# 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. 实施无 PXE 的发现

Satellite 提供了一个无 PXE 的发现服务，可在不需要基于 PXE 的服务(DHCP 和 TFTP)的情况下运行。您可以使用 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` 命令重新加载带有操作系统安装程序的 Linux 内核，而无需重新引导节点。



重要

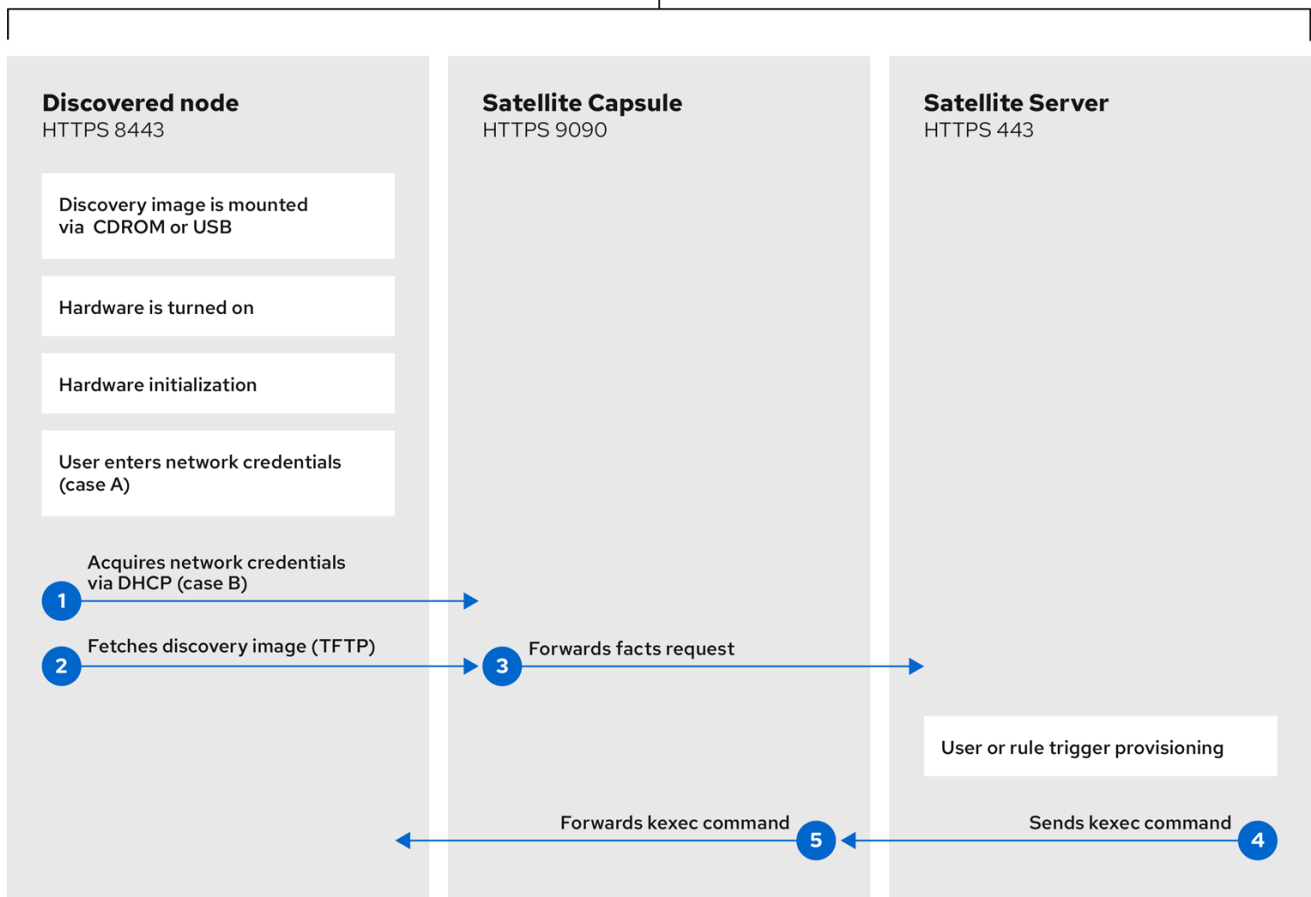
Discovery kexec 只是一个技术预览功能。技术预览功能不受红帽产品服务等级协议 (SLA) 支持，且功能可能并不完整。红帽不推荐在生产环境中使用它们。这些技术预览功能可以使用户提早试用新的功能，并有机会在开发阶段提供反馈意见。有关红帽技术预览功能支持范围的详情，请参考

<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。

PXE-less mode



278_Satellite_0922

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

已参加的使用

1. 将此介质复制到 CD、DVD 或者 USB 盘。例如，要复制到 `/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，启动主机并从介质引导。Discovery Image 显示 **Manual network setup** 或 **Discovery with DHCP** 的选项：

如果您选择 **Manual network setup**，Discovery 镜像会请求一组网络选项。这包括连接到 Satellite 服务器的主要网络接口。此发现镜像还要求提供网络接口配置选项，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 下一步。
4. 如果您选择 **Discovery with DHCP**，则 **Discovery** 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，例如胶囊服务器提供的网络接口。
5. 在主接口配置后，发现镜像会请求 **服务器 URL**，这是提供 **Discovery** 服务的 **Satellite** 服务器或胶囊服务器的 **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，以转发回 **Satellite** 服务器。它们以 **name-value** 格式输入。提供您需要的任何自定义事实，然后选择 **Confirm to continue**。

Satellite 报告了与卫星服务器的发现服务成功通信。在 **Satellite Web UI** 中，进入到 **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. 无人使用、自定义和镜像重新 MASTERING

您可以创建自定义发现 **ISO**，以便在引导后自动执行镜像配置过程。发现镜像为操作系统使用 **Linux** 内核，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：

fdi.cachefacts

在没有缓存的情况下上传的事实数。默认情况下，**Satellite** 不会缓存任何上传的事实。

fdi.countdown

在初始发现尝试后刷新文本用户界面前等待的秒数。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.dns_nameserver

用于 DNS SRV 记录的名称服务器。

fdi.dns_ndots

用于 DNS SRV 记录的 ndots 选项。

fdi.dns_search

用于 DNS SRV 记录的搜索域。

fdi.initnet

默认情况下，镜像初始化所有网络接口（值为 all）。当此设置设为 bootif 时，只会初始化它从中启动的网络接口。

fdi.ipv4.method

默认情况下，NetworkManager IPv4 方法设置被设置为 auto。此选项将覆盖它，将其设置为 ignore 以禁用 IPv4 堆栈。这个选项只适用于 DHCP 模式。

fdi.ipv6.method

默认情况下，NetworkManager IPv6 方法设置被设置为 auto。这个选项会覆盖它，将其设置为 ignore 以禁用 IPv6 堆栈。这个选项只适用于 DHCP 模式。

fdi.ipwait

等待 IP 在 HTTP 代理 SSL 证书启动时可用的时间（以秒为单位）。默认情况下，Satellite 等待 120 秒。

fdi.nmwait

NetworkManager 的 nmcli -wait 选项。默认情况下，nmcli 会等待 120 秒。

fdi.proxy_cert_days

自签名 HTTPS 证书有效的天数。默认情况下，证书在 999 天内有效。

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，该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，则镜像使用自动模式，并在不确认的情况下继续。

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

使用 指定自定义事实名称。

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

每个自定义事实的值。每个值都对应一个事实名称。例如，`fdi.pxfactvalue1` 为名为 `fdi.pxfactname1` 的事实设置值。

fdi.pxip, fdi.pxgw, fdi.pxdns

手动配置主网络接口的 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。您可以在逗号分隔中添加多个 DNS 条目 ^[1] list，如 `fdi.pxdns=192.168.1.1,192.168.200.1`。

fdi.pxmac

主接口的 MAC 地址，格式为 `AA:BB:CC:DD:EE:FF`。这是您旨在与胶囊服务器通信的接口。在自动模式中，使用第一个带有链接的 NIC（按字母顺序使用网络标识符）。在半自动化模式中，会出现一个屏幕，并要求您选择正确的接口。

fdi.rootpw

默认情况下，`root` 帐户被锁定。使用这个选项设置 `root` 密码。您可以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.uploadsleep

事实程序运行之间的持续时间（以秒为单位）。默认情况下，`facter` 每 30 秒运行一次。

fdi.vlan.primary

为主接口设置的 VLAN 标记 ID。

fdi.zips

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

fdi.zipserver

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

proxy.type

代理类型。这通常设置为 `proxy` 以连接到胶囊服务器。此参数还支持传统的 `foreman` 选项，其中通信直接发送到 `Satellite` 服务器，而不是胶囊服务器。

proxy.url

提供发现服务的胶囊服务器或 Satellite 服务器的 URL。

使用 discovery-remaster 工具重新 master OS 镜像

Satellite 服务器在 foreman-discovery-image 软件包中提供 discovery-remaster 工具。这个工具会重新管理镜像使其包含这些内核参数。要重新 master 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"
```

将此介质复制到 CD、DVD 或者 USB 盘。例如，要复制到 /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，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
│   └── 01_zip.sh
├── bin
│   └── ntpdate
├── facts
│   └── test.rb
```

```
├── lib
│   ├── libcrypto.so.1.0.0
│   ├── ruby
│   └── test.rb
```

- **autostart.d** 目录包含镜像启动时按 **POSIX** 顺序执行的脚本，但在主机注册到 **Satellite** 之前。
 - **bin** 目录添加到 **\$PATH** 变量中；您可以将二进制文件放在这个目录中，并将它们用于自动启动脚本。
 - **事实** 目录添加到 **FACTERLIB** 变量中，以便可以配置自定义事实并将其发送到 **Satellite**。
 - **lib** 目录添加到 **LD_LIBRARY_PATH** 变量中，**lib/ruby** 被添加到 **RUBYLIB** 变量中，以便正确执行 **/bin** 中的二进制文件。
2. 创建目录结构后，使用以下命令创建一个 **.zip** 文件存档：

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

3. 要告知发现镜像必须使用的扩展，请将 **zip** 文件放在您的 **TFTP** 服务器上带有 **Discovery** 镜像，然后使用 **fdi.zips** 选项更新 **PXE Linux** 模板 **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** 有两个存档，请使用以下语法：

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

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

您可以创建多个 **.zip** 文件，但请注意，它们将提取到 **Discovery** 镜像上的同一位置。如果文件具有相同的文件名，则后续 **.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** 的 **Satellite web UI** 中列出，请检查以下配置区域来帮助隔离错误：

- 在 Satellite Web UI 中, 进入到 Hosts > Templates > Provisioning Templates.
 1. 单击 **Build PXE Default** 以重新部署默认的 PXELinux 模板。
- 验证 TFTP 胶囊上的 `pxelinux.cfg/default` 配置文件。
- 确保主机、胶囊服务器和 Satellite 服务器之间进行适当的网络连接。
- 检查使用的 PXELinux 模板, 并确定其包含的 PXE 发现片断。代码片段被命名如下: `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 在发现的节点正常工作, 或使用您正在使用的 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。较少的内存可能会导致各种随机内核 panic 错误, 因为镜像被提取在内存中。

要收集重要的系统事实, 请使用 `discovery-debug` 命令。它打印出系统日志、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` 命令进行进一步调查。

发现主机上的第一个虚拟控制台为 `systemd` 日志保留。特别有用的系统日志标记为如下:

- `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 的引导脚本

- **NetworkManager - 网络信息**

使用 TTY2 或更高版本登录到发现的主机。root 帐户和 SSH 访问默认为禁用，但您可以使用 APPEND 行上的 Default PXELinux 模板中的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
```

[1]

NetworkManager 期望 ; 作为列表分隔符，但当前也接受 ,。如需更多信息，请参阅 [GRUB 中的 man nm-settings-keyfile](#) 和类似 [Shell](#) 的脚本

第 8 章 使用 RED HAT IMAGE BUILDER 镜像进行置备

在 Satellite 中，您可以启用与 RHEL web 控制台集成，以执行操作并监控您的主机。使用 RHEL web 控制台，您可以访问 Red Hat Image Builder 并构建镜像，然后上传到 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 安装程序对磁盘进行分区，将镜像下载并挂载到主机中。首选镜像类型是 TAR。

确保构建 TAR 镜像的蓝图包含内核软件包。

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

先决条件

- 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，添加自定义文件存储库到此产品，并将镜像上传到存储库。如需更多信息，请参阅管理内容中的 [导入单个 ISO 镜像和文件](#)。
2. 在 Satellite Web UI 中，进入到 **Configure > Host Groups**，然后选择要使用的主机组。
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，以以下格式输入绝对路径或相对路径，指向存储镜像的确切位置。
7. 点 **Submit** 保存您的更改。

您可以使用计算资源将此镜像用于裸机置备和置备。有关裸机置备的更多信息，请参阅 [第 5 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 (LIBVIRT)上调配虚拟机.

基于内核的虚拟机(KVM)使用运行在 Red Hat Enterprise Linux 上的开源虚拟化守护进程和名为 libvirt 的 API。Satellite 可以连接到 KVM 服务器上的 libvirt API，在虚拟机监控程序上配置主机，并控制某些虚拟化功能。

只有通过 Satellite 创建的虚拟机才能管理。不支持具有目录存储池类型的虚拟机。

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。如需更多信息，请参阅 [管理内容中的同步存储库](#)。
- 为主机注册提供激活码。如需更多信息，请参阅[管理内容中的创建激活码](#)。
- 在 KVM 服务器上管理网络的胶囊服务器。确保此网络上没有运行其他 DHCP 服务，以避免与胶囊服务器冲突。有关 Capsule 服务器的网络服务配置的更多信息，请参阅 [置备主机中的配置网络](#)。
- 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 8 配置和管理虚拟化](#)。
- 如果要使用基于镜像的置备，现有虚拟机镜像。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 进行管理。
- 可选：这些步骤中的示例为 KVM 使用 root 用户。如果要在 KVM 服务器上使用非 root 用户，则必须将用户添加到 KVM 服务器上的 libvirt 组中：

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

其他资源

- 有关非管理员用户需要置备主机的权限列表，请参阅 [附录 E, 置备主机所需的权限](#)。

9.1. 为 KVM 连接配置 SATELLITE 服务器

在添加 KVM 连接前，为 foreman 用户创建一个 SSH 密钥对，以确保卫星服务器和 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

在 **Satellite** 中使用此流程将 **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, 进入到 **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 **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 **Spice**。

7. 可选 : 要使用随机生成的密码保护新主机的控制台访问, 请在 **显示连接** 复选框上选择设置随机生成的密码。您可以从 **KVM** 服务器上执行以下命令的输出中检索 **VNC** 控制台的密码, 以访问客户虚拟机控制台 :

```
# 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, 密码都会随机生成, 例如使用 **virt-manager**。

8. 单击 **Test Connection**, 以确保 **Satellite** 服务器在没有出错的情况下连接到 **KVM** 服务器。

9. 验证 **Locations** 和 **Organizations** 选项卡是否已自动设置为您当前的上下文。如果需要, 请在这些标签页中添加其他上下文。

10. 点 **Submit** 以保存 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

要使用基于镜像的调配来创建主机，您必须将有关镜像的信息（如访问详细信息和镜像位置）添加到 Satellite 服务器。

请注意，您只能通过 **Satellite 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，进入到 **Infrastructure > Compute Resources** 并点击 **KVM 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，选择镜像的基本操作系统。
5. 从 **Architecture** 列表中，选择操作系统架构。

6. 在 **Username** 字段中, 输入用于镜像访问的 **SSH 用户名**。这通常是 **root 用户**。
7. 在 **Password** 字段中, 输入用于镜像访问的 **SSH 密码**。
8. 在 **Image path** 字段中, 输入指向 **KVM 服务器上的镜像的完整路径**。例如 :

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

CLI 过程

- 使用 **hammer compute-resource image create** 命令创建镜像。使用 **--uuid** 字段在 **KVM 服务器上存储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**。当使用此计算配置集在 **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, 进入到 **Infrastructure > Compute Profiles**。

2. 在 **Compute Profiles** 窗口中，单击现有计算配置文件的名称，或者单击 **Create Compute Profile**，输入 **Name**，然后单击 **Submit**。
3. 单击 **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，输入要分配给新主机的内存量。
6. 从 **Image** 列表中，如果执行基于镜像的调配，请选择要使用的镜像。
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，至少一个接口必须指向 **Capsule** 管理网络。
8. 在 **Storage** 区域中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. 要添加 **compute** 配置集的值，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现有镜像创建主机：

- 如果要通过网络连接创建主机，则新主机必须能够访问 Satellite 服务器的集成胶囊或 KVM 虚拟网络上的外部胶囊服务器，以便主机能够访问 PXE 置备服务。这个新主机条目会触发 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胶囊服务器，虚拟机将引导至 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，新的主机条目将触发 KVM 服务器，以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 冲突

对于基于网络的调配，如果您使用 KVM 服务器上的虚拟网络进行调配，请选择不提供 DHCP 分配的网络。这会导致在启动新主机时 DHCP 与 Satellite 服务器冲突。

流程

1. 在 Satellite Web UI 中，进入到 Hosts > Create Host。
2. 在 Name 字段中输入主机的名称。
3. 可选：点 Organization 选项卡并更改机构上下文以匹配您的要求。
4. 可选：点 Location 选项卡并更改位置上下文以匹配您的要求。
5. 从 Host Group 列表中，选择要为其分配主机的主机组。该主机组将填充表单。
6. 在 Deploy on 列表中，选择 KVM 连接。
7. 从 Compute Profile 列表中，选择一个用于自动填充虚拟机设置的配置集。KVM 特定字段填充了您的计算配置文件中的设置。如果需要，修改这些设置。

8. 单击 **Interfaces** 选项卡，然后在主机的接口上点 **Edit**。
9. 验证字段是否填充了值。特别是：
 - **Satellite** 会自动为新主机分配 IP 地址。
 - 确保 **MAC address** 字段为空。KVM 在置备过程中为主机分配 MAC 地址。
 - **Host** 选项卡中的 **Name** 变为 DNS 名称。
 - 确保 **Satellite** 会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。
10. 单击 **OK** 以保存。要添加另一个接口，请单击 **Add Interface**。您只能为 **Provision** 和 **Primary** 选择一个接口。
11. 单击 **Operating System** 选项卡，并确认所有字段都自动包含值。
12. 选择要使用的 **Provisioning Method**：
 - 对于基于网络的调配，请单击 **Network Based**。
 - 对于基于镜像的调配，请单击 **基于镜像的调配**。
13. 点 **Provisioning templates** 中的 **Resolve** 检查新主机是否可以识别要使用的正确置备模板。
14. 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文件的详细信息。修改这些设置以满足您的需要。

15. 单击 **Parameters** 选项卡, 并确保存在提供激活密钥的参数。如果没有, 添加激活码。
16. 单击 **Submit** 以保存主机条目。

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` 构建。将以下示例中的值替换为适合您的环境的值。

```
# hammer host create \
--build true \
--compute-attributes="cpus=1,memory=1073741824" \
--compute-resource "My_KVM_Server" \
--enabled true \
--hostgroup "My_Host_Group" \
--interface
"managed=true,primary=true,provision=true,compute_type=network,compute_network=exam
plenetwork" \
--location "My_Location" \
--managed true \
--name "My_Host_Name" \
--organization "My_Organization" \
--provision-method "build" \
--root-password "My_Password" \
--volume="pool_name=default,capacity=20G,format_type=qcow2"
```

- 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 \
--compute-attributes="cpus=1,memory=1073741824" \
--compute-resource "My_KVM_Server" \
--enabled true \
--hostgroup "My_Host_Group" \
--image "My_KVM_Image" \
--interface
"managed=true,primary=true,provision=true,compute_type=network,compute_network=exampl
enetwork" \
--location "My_Location" \
--managed true \
--name "My_Host_Name" \
--organization "My_Organization" \
--provision-method "image" \
--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 是一个企业级的服务器和客户端虚拟化平台。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 可避免网络上的任何特殊配置，如受管 DHCP 和 TFTP，以完成虚拟机的安装。此方法不需要 Satellite 使用 SSH 连接到调配的虚拟机，以运行完成脚本。

先决条件

- **您可以将同步的内容软件仓库用于 Red Hat Enterprise Linux。如需更多信息，请参阅 [管理内容中的同步存储库](#)。**
- **为主机注册提供激活码。如需更多信息，请参阅[管理内容中的创建激活码](#)。**
- **在 Red Hat Virtualization 环境中管理逻辑网络的胶囊服务器。确保此网络上没有运行其他 DHCP 服务，以避免与胶囊服务器冲突。如需更多信息，请参阅 [置备主机中的配置网络](#)。**
- **如果要使用基于镜像的置备，现有模板（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，以便与 Satellite 服务器进行通信。不要将 admin@internal 用户用于此通信。反之，创建一个具有以下权限的新 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 管理指南中的管理用户任务。https://access.redhat.com/documentation/zh-cn/red_hat_virtualization/4.3/html/administration_guide/sect-red_hat_enterprise_virtualization_manager_user_tasks

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

在 Satellite 中使用此流程将 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, 进入到 **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, 以以下格式输入 **Red Hat Virtualization Manager API** 的连接 URL : `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 **Datacenter** 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, 输入用于 **SSL/TLS** 访问的证书颁发机构。或者, 如果您将字段留空, 则会在服务器的第一个 **API** 请求中生成一个自签名证书。
12. 单击 **位置** 选项卡, 再选择要使用的位置。
13. 单击 **Organizations** 选项卡, 再选择要使用的组织。

14.

单击 **Submit** 以保存计算资源。

CLI 过程

- 输入带有 **Ovirt for --provider** 的 **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**，您必须准备一个在 **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，用于在 **Satellite** 中用于基于镜像的调配。

2. 在虚拟机上安装 **cloud-init**：

```
# dnf 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. 在 SATELLITE 服务器中添加 RED HAT VIRTUALIZATION 镜像

要使用基于镜像的调配来创建主机，您必须将有关镜像的信息（如访问详细信息和镜像位置）添加到 Satellite 服务器。

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

流程

1. 在 Satellite Web UI 中，进入到 **Infrastructure > Compute Resources**，点 **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，选择镜像的基本操作系统。
5. 从 **Architecture** 列表中，选择操作系统架构。
6. 在 **Username** 字段中，输入用于镜像访问的 SSH 用户名。这通常是 **root** 用户。
7. 在 **Password** 字段中，输入用于镜像访问的 SSH 密码。
8. 从 **Image** 列表中，从 **Red Hat Virtualization** 计算资源选择一个镜像。
9. 可选：如果镜像支持用户数据输入，请选择 **User Data** 复选框，如 **cloud-init** 数据。
10. 点 **Submit** 以保存镜像详情。

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，进入到 Hosts > Templates > Provisioning Templates，然后点击 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 -%>
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. 单击 **Type** 选项卡，然后从 **Type** 列表中选择 **User data template**。
5. 点 **关联** 选项卡，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 **Locations** 列表中选择您要与模板关联的位置。
7. 单击 **Organizations** 选项卡，然后在 **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，进入到 Hosts > Provisioning Setup > Operating Systems。**
10. **选择您要与模板关联的操作系统。**
11. **单击 Templates 选项卡，然后从 User data template 列表中选择新模板的名称。**
12. **点 Submit。**

10.5. 在计算配置集中添加 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。当使用此计算配置集在 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，进入到 Infrastructure > Compute Profiles。**
2. **在 Compute Profiles 窗口中，单击现有计算配置文件的名称，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，选择要用于 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，输入要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调配的镜像。
9. 在 **Network Interfaces** 区域中，输入主机网络接口的网络参数。您可以创建多个网络接口。但是，至少一个接口必须指向 **Capsule** 管理网络。对于每个网络接口，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，选择要使用的逻辑网络。
10. 在 **Storage** 区域中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，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 **Storage domain** 列表中选择卷的存储域。
 - c. 从 **Preallocate** 磁盘中，选择完整磁盘的精简配置或预分配。
 - d. 在 **Bootable** 列表中，选择是否需要可引导卷还是不可启动的卷。
11. 单击 **Submit** 以保存计算配置文件。

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，则新主机必须能够访问 Red Hat Virtualization 虚拟网络上的 Satellite 服务器集成胶囊或外部 Capsule 服务器，以便主机可以访问 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胶囊服务器，虚拟机将引导至 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，以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 冲突

对于基于网络的调配，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。这会导致在启动新主机时 DHCP 与 Satellite 服务器冲突。

流程

1. 在 Satellite Web UI 中，进入到 Hosts > Create Host。
2. 在 Name 字段中输入主机的名称。
3. 可选：点 Organization 选项卡并更改机构上下文以匹配您的要求。
4. 可选：点 Location 选项卡并更改位置上下文以匹配您的要求。

5. 从 **Host Group** 列表中，选择要为其分配主机的主机组。该主机组将填充表单。
6. 在 **Deploy on** 列表中，选择 **Red Hat Virtualization** 连接。
7. 从 **Compute Profile** 列表中，选择一个用于自动填充虚拟机设置的配置集。Red Hat Virtualization 特定字段使用计算配置集中的设置填充。如果需要，修改这些设置。
8. 单击 **Interfaces** 选项卡，然后在主机的接口上点 **Edit**。
9. 验证字段是否填充了值。特别是：
 - **Satellite** 会自动为新主机分配 IP 地址。
 - 确保 **MAC address** 字段为空。Red Hat Virtualization 在置备过程中为主机分配 MAC 地址。
 - **Host** 选项卡中的 **Name** 变为 DNS 名称。
 - 确保 **Satellite** 会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。
10. 单击 **OK** 以保存。要添加另一个接口，请单击 **Add Interface**。您只能为 **Provision** 和 **Primary** 选择一个接口。
11. 单击 **Operating System** 选项卡，并确认所有字段都自动包含值。
12. 选择要使用的 **Provisioning Method**：
 - 对于基于网络的调配，请单击 **Network Based**。

- 对于基于镜像的调配，请单击 [基于镜像的调配](#)。
13. 点 **Provisioning templates** 中的 **Resolve** 检查新主机是否可以识别要使用的正确置备模板。
 14. 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文件的详细信息。修改这些设置以满足您的需要。
 15. 单击 **Parameters** 选项卡，并确保存在提供激活密钥的参数。如果没有，添加激活码。
 16. 单击 **Submit** 以保存主机条目。

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` 构建。将以下示例中的值替换为适合您的环境的值。

```
# hammer host create \
--build true \
--compute-attributes="cluster=Default,cores=1,memory=1073741824,start=true" \
--compute-resource "MyRHV_" \
--enabled true \
--hostgroup "My_Host_Group" \
--interface
"managed=true,primary=true,provision=true,compute_name=eth0,compute_network=satnetwork" \
--location "My_Location" \
--managed true \
--name "My_Host_Name" \
--organization "My_Organization" \
--provision-method build \
--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 \
--compute-attributes="cluster=Default,cores=1,memory=1073741824,start=true" \
--compute-resource "MyRHV_" \
--enabled true \
--hostgroup "My_Host_Group" \
--image "MyRHV_Image_" \
--interface
```



```
"managed=true,primary=true,provision=true,compute_name=eth0,compute_network=satnetwork" \  
--location "My_Location" \  
--managed true \  
--name "My_Host_Name" \  
--organization "My_Organization" \  
--provision-method "image" \  
--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. 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 环境中管理网络的胶囊服务器。确保此网络上没有运行其他 DHCP 服务，以避免与胶囊服务器冲突。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 VMware 模板。
- 您可以将同步的内容软件仓库用于 Red Hat Enterprise Linux。如需更多信息，请参阅 [管理内容中的同步存储库](#)。
- 为主机注册提供激活码。如需更多信息，请参阅 [管理内容中的创建激活码](#)。

11.2. 创建 VMWARE 用户

VMware vSphere 服务器需要类似管理的用户用于 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。反之，创建一个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***

- ***所有 Privileges → Virtual Machine → Change Config (All)***

- ***所有 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***

- **所有 Privileges → Virtual Machine → Configuration (All)**
- **所有 Privileges → Virtual Machine → Interaction (All)**
- **所有 Privileges → Virtual Machine → Inventory (All)**
- **All Privileges → Virtual Machine → Provisioning (All)**
- **所有 Privileges → Virtual Machine → Guest Operations (All)**

11.3. 将 VMWARE 连接添加到 SATELLITE 服务器

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 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, 进入到 **Infrastructure > Compute Resources**, 并在 **Compute Resources** 窗口中点击 **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，选择要从此列表中管理的特定数据中心。
10. 在 **Fingerprint** 字段中，确保此字段使用数据中心的指纹填充。
11. 在 **Display Type** 列表中，选择一个控制台类型，如 VNC 或 VMRC。请注意，VMware ESXi 6.5 及更新的版本不支持 VNC 控制台。
12. 可选：在 **VNC Console Passwords** 字段中，选择在显示连接复选框上设置随机生成的密码，以使用随机生成的密码保护新主机的控制台访问。您可以从以下命令的输出结果中检索 VNC 控制台的密码来访问 libvirtd 主机中的客户虚拟机控制台：

```
# 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，密码都会随机生成，例如使用 virt-manager。
13. 从 **Enable Caching** 列表中，您可以选择是否启用计算资源缓存。如需更多信息，请参阅第 11.10 节“计算资源缓存”。
14. 单击 **Locations** 和 **Organizations** 选项卡，并验证这些值是否已自动设置为当前上下文。您还可以添加额外的上下文。
15. 点 **Submit** 以保存连接。

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"
```

11.4. 将 VMWARE 镜像添加到 SATELLITE 服务器

VMware vSphere 使用模板作为镜像来创建新虚拟机。如果使用基于镜像的调用来创建新主机，则需要在 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，进入到 **Infrastructure > Compute Resources**。
2. 选择您的 **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，选择镜像的基本操作系统。
6. 从 **Architecture** 列表中，选择操作系统架构。

7. 在 **Username** 字段中，输入用于镜像访问的 **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** 以保存镜像详情。

CLI 过程

- 使用 **hammer compute-resource image create** 命令创建镜像。使用 **--uuid** 字段在 **vSphere** 环境中存储相对模板路径：

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

11.5. 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，进入到 **Infrastructure > Compute Profiles**。
2. 选择计算配置文件。
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，输入要分配给每个 CPU 的内核数。**
6. **在 Memory 字段中，输入要分配给主机的内存量（以 MiB 为单位）。**
7. **在固件复选框中，选择 BIOS 或 UEFI 作为主机的固件。默认情况下，这设置为自动。**
8. **在 Cluster 列表中，选择 VMware 环境中的目标主机集群的名称。**
9. **从 Resource pool 列表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. **在 Guest OS 列表中，选择要在 VMware vSphere 中使用的操作系统。**
12. **从 Virtual H/W 版本 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. **在 Boot order 列表中，定义虚拟机尝试引导的顺序。**
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，为主机选择磁盘访问方法。
20. 如果要使用 **eager** 零厚置备，请选择 **Eager** 零复选框。默认情况下，磁盘使用 **lazy** 零厚置备。
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. 单击 **Submit** 以保存计算配置文件。

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"
```

11.6. 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。

对于基于网络的调配，您必须创建一个主机来访问 **VMware vSphere** 虚拟网络上的 **Satellite** 服务器集成胶囊或外部胶囊服务器，以便主机能够访问 **PXE** 调配服务。新主机条目触发 **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胶囊服务器，虚拟机将引导至 PXE，并开始安装所选操作系统。

DHCP 冲突

如果您在 VMware vSphere 服务器上使用虚拟网络进行置备，请确保选择一个不提供 DHCP 分配的虚拟网络。这会导致在启动新主机时 DHCP 与 Satellite 服务器冲突。

对于基于镜像的调配，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，进入到 Hosts > Create Host。
2. 在 Name 字段中输入主机的名称。
3. 可选：点 Organization 选项卡并更改机构上下文以匹配您的要求。
4. 可选：点 Location 选项卡并更改位置上下文以匹配您的要求。
5. 从 Host Group 列表中，选择要为其分配主机的主机组。该主机组将填充表单。
6. 从 Deploy on 列表中，选择 VMware vSphere 连接。
7. 从 Compute Profile 列表中，选择一个用于自动填充基于虚拟机的设置的配置集。
8. 单击 Interfaces 选项卡，然后在主机的接口上点 Edit。
9. 验证字段是否填充了值。特别是：

- **Satellite 会自动为新主机分配 IP 地址。**
 - **确保 MAC address 字段为空。VMware 在置备过程中为主机分配 MAC 地址。**
 - **Host 选项卡中的 Name 变为 DNS 名称。**
 - **确保 Satellite 会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。**
10. **在接口窗口中，查看使用计算配置集中的设置填充的 VMware 特定字段。修改这些设置以满足您的需要。**
 11. **单击 OK 以保存。要添加另一个接口，请单击 Add Interface。您只能为 Provision 和 Primary 选择一个接口。**
 12. **单击 Operating System 选项卡，并确认所有字段都自动包含值。**
 13. **选择您想要的 Provisioning Method :**
 - **对于基于网络的调配，请单击 Network Based。**
 - **对于基于镜像的调配，请单击 基于镜像的调配。**
 - **对于引导磁盘调配，请单击 基于 Boot disk。**
 14. **点 Provisioning templates 中的 Resolve 检查新主机是否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文件的详细信息。修改这些设置以满足您的要求。**
 - 16.

单击 **Parameters** 选项卡，并确保存在提供激活密钥的参数。如果参数不存在，请单击 **+ Add Parameter**。在 **Name** 字段中，输入 `kt_activation_keys`。在字段 **Value** 中，输入用于注册内容主机的激活密钥的名称。

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` 构建 以使用基于网络的调配：

```
# 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 "My_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"

```

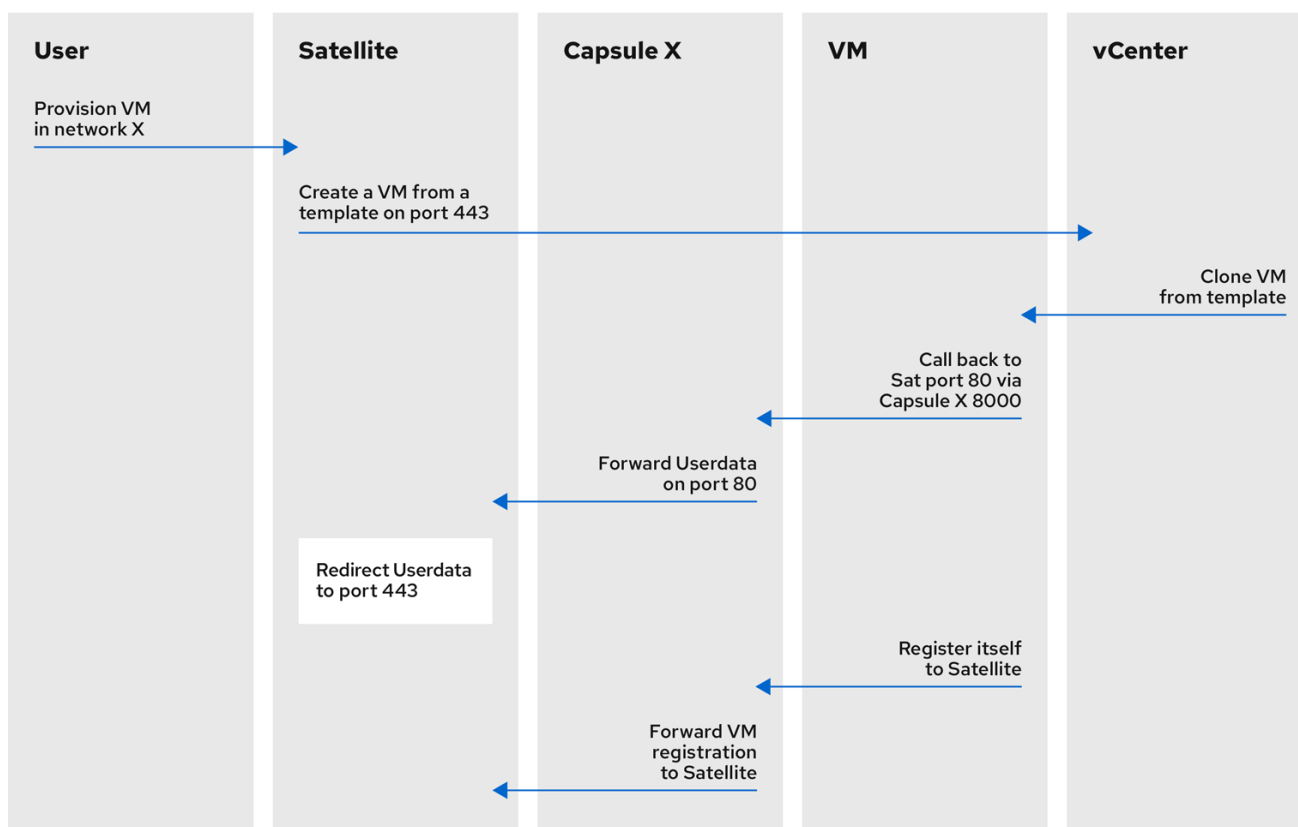
有关此计算资源的其他主机创建参数的更多信息，请输入 `hammer host create --help` 命令。

11.7. 使用 VMWARE CLOUD-INIT 和 USERDATA 模板进行配置

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，以进行进一步的 VMware 自定义，并使 VMware 托管的虚拟机能够重新调用 Satellite。

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

图 11.1. VMware cloud-init 置备概述



278_Satellite_0922

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 userdata 置备模板添加了自定义身份信息**
- **在置备完成后, Cloud-init 置备模板指示虚拟机在 cloud-init 运行时重新调用 Capsule**
- **VMware vCenter 将模板克隆到虚拟机**
- **VMware vCenter 为虚拟机的身份应用自定义, 包括主机名、IP 和 DNS**
- **虚拟机构建, 调用 cloud-init, 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, 即使将虚拟机注册到胶囊, 虚拟机也会始终调用回卫星。如需更多信息, 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。 https://access.redhat.com/documentation/zh-cn/red_hat_satellite/6.15/html-single/installing_capsule_server/index#capsule-port-and-firewall-requirements_capsule**
- **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。**
- **备份以下配置文件 :**
 - **`/etc/cloud/cloud.cfg.d/01_network.cfg`**
 - **`/etc/cloud/cloud.cfg.d/10_datasource.cfg`**
 - **`/etc/cloud/cloud.cfg`**

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. 点 **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** 以保存更改。

准备镜像以使用 cloud-init 模板

要准备镜像, 您必须首先在虚拟机上配置所需的设置, 然后可以保存为 **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、启用并配置为重新调用 **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

便不需要的信息传输到您用于置备的镜像。

如果您使用 `cloud-init` 的镜像，仍必须按照以下步骤启用 `cloud-init` 与 `Satellite` 通信，因为 `cloud-init` 默认禁用。

流程

1. 在用于创建镜像的虚拟机中，安装所需的软件包：

```
# dnf install cloud-init open-vm-tools perl-interpreter perl-File-Temp
```

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` 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
# dnf 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](#)。

11.8. 删除 VMWARE 上的虚拟机

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

流程

1. 在 **Satellite Web UI** 中，进入到 **Infrastructure > Compute Resources**。
2. 选择 **VMware 供应商**。
3. 在 **Virtual Machines** 选项卡上，从 **Actions** 菜单中选择 **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。

11.9. 将虚拟机从 VMWARE 导入到 SATELLITE

您可以将 **VMware** 上运行的现有虚拟机导入到 **Satellite**。

流程

1. 在 **Satellite Web UI** 中，进入到 **Infrastructure > Compute Resources**。
2. 选择 **VMware 计算资源**。
3. 在 **Virtual Machines** 选项卡上，从 **Actions** 菜单中选择 **Import as managed Host** 或 **Import as unmanaged Host**。以下页面看起来与创建已选定计算资源的主机相同。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** 以将虚拟机导入到 **Satellite**。

11.10. 计算资源缓存

计算资源缓存可加快 **VMware** 信息的渲染。

11.10.1. 启用计算资源缓存

启用或禁用计算资源的缓存：

流程

1. 在 **Satellite Web UI** 中，进入到 **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。

11.10.2. 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，进入到 **Infrastructure > Compute Resources**。
2. 选择您要为刷新计算资源缓存的 **VMware** 服务器，然后单击 **Refresh Cache**。

CLI 过程

- 使用此 **API** 调用刷新计算资源缓存：

```
# 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 compute-resource** 列表 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 上置备虚拟机

OpenShift Virtualization 满足已采用或希望采用 Red Hat OpenShift Container Platform 但现有虚拟机(VM)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构建、修改和部署应用程序。这些功能支持跨开放混合云的快速应用程序现代化。

使用 Satellite，您可以为 OpenShift Virtualization 创建计算资源，以便您可以使用 Satellite 在 OpenShift Container Platform 中置备和管理虚拟机。

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。如需更多信息，请参阅 [管理内容](#) 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 [管理内容](#) 中的 [创建激活码](#)。
- OpenShift Container Platform 集群必须具有 cluster-admin 权限。
- 在 OpenShift Container Platform 集群中管理网络的 Capsule 服务器。确保此网络上没有运行其他 DHCP 服务，以避免与胶囊服务器冲突。有关 Capsule 服务器的网络服务配置的更多信息，请参阅 [置备主机](#) 中的 [配置网络](#)。

其他资源

- 有关非管理员用户需要置备主机的权限列表，请参阅 [附录 E](#), [置备主机所需的权限](#)。

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. 获取用于 HTTP 和 HTTP 验证的令牌：

- a. 登录到 OpenShift Container Platform 集群并列出包含令牌的 secret：

```
$ oc get secrets
```

- b. 获取 secret 的令牌：

```
$ oc get secrets MY_SECRET -o jsonpath='{.data.token}' | base64 -d | xargs
```

- c. 记录令牌以便稍后在此流程中使用。
3. 在 Satellite Web UI 中，进入到 Infrastructure > Compute Resources，再点 Create Compute Resource。
 4. 在 Name 字段中输入新计算资源的名称。
 5. 从 Provider 列表中，选择 OpenShift Virtualization。
 6. 在 Description 字段中，输入计算资源的描述。
 7. 在 Hostname 字段中，输入 OpenShift Container Platform 集群的 FQDN、主机名或 IP 地址。

8. 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。
9. 在 **Namespace** 字段中输入 **OpenShift Container Platform** 集群的用户名。
10. 在 **Token** 字段中, 为 **HTTP** 和 **HTTPs** 身份验证输入 **bearer** 令牌。
11. 可选: 在 **X509 Certification authority** 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(IaaS)云提供了基础。它为支持云的工作负载开发提供了高度可扩展的容错平台。在 **Satellite** 中，您可以与红帽 OpenStack 平台 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。如需更多信息，请参阅 [管理内容中的同步存储库](#)。
- 为主机注册提供激活码。如需更多信息，请参阅[管理内容中的创建激活码](#)。
- 在 OpenStack 环境中管理网络的胶囊服务器。如需更多信息，请参阅 [置备主机中的配置网络](#)。
- 为基于镜像的调配添加到 OpenStack Image Storage (glance)服务中的镜像。如需更多信息，请参阅 [Red Hat OpenStack Platform 实例和镜像指南](#)。

13.1. 添加 RED HAT OPENSTACK PLATFORM 连接到 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，进入到 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 **Locations** 和 **Organizations** 选项卡，并验证您要使用的位置和组织是否已设置为当前的上下文。添加您想要在这些选项卡的任何其他上下文。
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

要使用基于镜像的调用来创建主机，您必须将有关镜像的信息（如访问详细信息和镜像位置）添加到 Satellite 服务器。

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

流程

1. 在 Satellite Web UI 中，进入到 **Infrastructure > Compute Resources**，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，选择镜像的基本操作系统。
5. 从 **Architecture** 列表中，选择操作系统架构。
6. 在 **Username** 字段中，输入用于镜像访问的 **SSH** 用户名。这通常是 **root** 用户。
7. 在 **Password** 字段中，输入用于镜像访问的 **SSH** 密码。
8. 从 **Image** 列表中，从 **Red Hat OpenStack Platform** 计算资源选择一个镜像。

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

CLI 过程

- 使用 **hammer compute-resource image create** 命令创建镜像。使用 **--uuid** 字段在 **Red Hat OpenStack Platform** 服务器上存储镜像位置的完整路径。

```
# 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. 在计算配置集中添加 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。当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，进入到 **Infrastructure > Compute Profiles**。
2. 在 **Compute Profiles** 窗口中，单击现有计算配置文件的名称，或者单击 **Create Compute Profile**，输入 **Name**，然后单击 **Submit**。
3. 单击 **Red Hat OpenStack Platform** 计算资源的名称。
4. 从 **Flavor** 列表中，选择 **Red Hat OpenStack Platform** 上的硬件配置文件，以用于主机。
5. 从 **Availability zone** 列表中，选择要在 **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调配的镜像。
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 **Internal** 网络，选择要加入的主机的专用网络。
10. 从 **浮动 IP** 网络，选择要加入并分配浮动 IP 地址的外部网络。
11. 从卷引导中，选择是否从镜像创建了一个卷。如果没有选择，实例将直接引导镜像。
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. 单击 **Submit** 以保存计算配置文件。

CLI 过程

-

将 **Red Hat OpenStack Platform** 详情设置为计算配置集：

```
# hammer compute-profile values create
--compute-resource "My_Laptop" \
--compute-profile "My_Compute_Profile" \
--compute-attributes
"availability_zone=My_Zone,image_ref=My_Image,flavor_ref=m1.small,tenant_id=openstack,security_groups=default,network=My_Network,boot_from_volume=false"
```

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, 进入到 **Hosts > Create Host**.
2. 在 **Name** 字段中输入主机的名称。
3. 可选: 点 **Organization** 选项卡并更改机构上下文以匹配您的要求。
4. 可选: 点 **Location** 选项卡并更改位置上下文以匹配您的要求。
5. 从 **Host Group** 列表中, 选择要为其分配主机的主机组。该主机组将填充表单。
6. 从 **Deploy on** 列表中, 选择 **Red Hat OpenStack Platform** 连接。
7. 从 **Compute Profile** 列表中, 选择一个用于自动填充虚拟机设置的配置集。
8. 从 **Lifecycle Environment** 列表中, 选择环境。
9. 单击 **Interfaces** 选项卡, 然后在主机的接口上点 **Edit**。
10. 验证字段是否填充了值。特别是:
 - **Satellite** 会自动为新主机分配 IP 地址。
 - 确保 **MAC address** 字段为空。Red Hat OpenStack Platform 在置备过程中为主机分配 MAC 地址。
 - **Host** 选项卡中的 **Name** 变为 DNS 名称。
 - 确保 **Satellite** 会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** 以保存。要添加另一个接口，请单击 **Add Interface**。您只能为 **Provision** 和 **Primary** 选择一个接口。
12. 单击 **Operating System** 选项卡，并确认所有字段都自动包含值。
13. 如果要更改从计算配置集自动填充的镜像，请从 **Images** 列表中选择不同的镜像，以新主机的 **root** 卷为基础。
14. 点 **Provisioning templates** 中的 **Resolve** 检查新主机是否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文件的详细信息。修改这些设置以满足您的需要。
16. 单击 **Parameters** 选项卡，并确保存在提供激活密钥的参数。如果没有，添加激活码。
17. 单击 **Submit** 以保存主机条目。

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=mynetw  
ork" \  
--compute-resource "My_OpenStack_Platform" \  
--enabled true \  
--hostgroup "My_Host_Group" \  
--image "My_OpenStack_Image" \  
--interface "managed=true,primary=true,provision=true" \  
--location "My_Location" \  
--managed true \  
--name "My_Host_Name" \  
--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) 是一个提供公共云计算资源的 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 环境中管理网络的胶囊服务器。使用虚拟私有云(VPC)来确保主机和胶囊服务器之间的安全网络。
- 用于基于镜像的调配的 Amazon Machine Image (AMI)。
- 您可以将同步的内容软件仓库用于 Red Hat Enterprise Linux。如需更多信息，请参阅 [管理内容中的同步存储库](#)。
- 为主机注册提供激活码。如需更多信息，请参阅 [管理内容中的创建激活码](#)。

14.2. 安装 AMAZON EC2 插件

安装 Amazon EC2 插件，将 EC2 计算资源提供程序附加到 Satellite。这样，您可以管理和部署主机到 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，进入到 **Administer > About** 并选择 **compute resources** 选项卡来验证 Amazon EC2 插件的安装。

14.3. 将 AMAZON EC2 连接添加到 SATELLITE 服务器

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](#)。

先决条件

- 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 服务器的时间已正确同步。确保 NTP 服务（如 `ntpd` 或 `chronyd`）已在 Satellite 服务器上正确运行。无法为 Amazon Web Services 提供正确的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。https://access.redhat.com/documentation/zh-cn/red_hat_satellite/6.15/html-single/installing_satellite_server_in_a_connected_network_environment/index#synchronizing-the-system-clock-with-chronyd_satellite

流程

1. 在 Satellite Web UI 中, 进入到 **Infrastructure > Compute Resources**, 并在 **Compute Resources** 窗口中点击 **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: 在 **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。如需更多信息, 请参阅 [第 14.4 节“将 HTTP 代理与计算资源搭配使用”](#)。
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 [Amazon 文档网站上的 AWS 帐户管理访问密钥](#)。

7. 可选：点击 **Load Regions** 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. 单击位置选项卡，并确保选中了要使用的位置或添加其他位置。
10. 点 **Organizations** 选项卡，并确保选择了您要使用的机构，或添加不同的机构。
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** 密钥的副本。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。如果您在稍后阶段需要 **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. 将 HTTP 代理与计算资源搭配使用

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，然后为计算资源添加另一个 **HTTP** 代理，您在 **Administer > Settings** 中定义的 **HTTP** 代理具有优先权。

流程

1. 在 **Satellite Web UI** 中, 进入到 **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 **HTTP 代理** 需要身份验证, 请输入用户名和密码来对 **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. 点 **Locations** 标签页并添加一个位置。
7. 点 **Organization** 标签页并添加一个机构。
8. 点 **Submit**。

14.5. 为 AMAZON EC2 创建镜像

您可以从 **Satellite** 内部为 **Amazon EC2** 创建镜像。

流程

1. 在 **Satellite Web UI** 中, 进入到 **Infrastructure > Compute Resources**。
2. 选择您的 **Amazon EC2** 供应商。
3. 点 **Create Image**。
 - 在 **Name** 字段中输入您的 **EC2** 镜像的有意义的和唯一名称。

在 **Operating System** 列表中选择要关联的操作系统。

- 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 或操作系统厂商提供的镜像 ID。
- 可选：选择 **User Data** 复选框来启用对用户数据输入的支持。
- 可选：在创建此镜像时为 Fog 设置 **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

Amazon EC2 使用基于镜像的调配来创建主机。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，进入到 **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** 字段中输入用于标识镜像供以后使用的名称。

4. 在 **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，输入用于镜像访问的 **SSH 用户名**。这通常是 **root 用户**。
7. 在 **Password** 字段中，输入用于镜像访问的 **SSH 密码**。
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-xxxxxxxx**。
9. 可选：如果镜像支持用户数据输入，请选择 **User Data** 复选框，如 **cloud-init 数据**。如果启用用户数据，则会自动禁用 **Finish** 脚本。这也适用于反向：如果您启用了 **Finish** 脚本，这会禁用用户数据。
10. 可选：在 **IAM role** 字段中输入用于创建镜像的 **Amazon 安全角色**。
11. 点 **Submit** 以保存镜像详情。

CLI 过程

- 使用 **hammer compute-resource image create** 命令创建镜像。使用 **--uuid** 字段在 **Amazon EC2 服务器** 上存储镜像位置的完整路径。

```
# 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. 在计算配置集中添加 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，进入到 Infrastructure > Compute Profiles 并点您的配置集名称，然后点 EC2 连接。
2. 从 Flavor 列表中，选择 EC2 上要用于主机的硬件配置文件。
3. 从 Image 列表中，选择要用于基于镜像的调配的镜像。
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 Subnet 列表中，添加 EC2 实例的子网。如果您有 VPC 用于置备新主机，请使用其子网。
6. 从 Security Groups 列表中，选择要应用到主机的端口和 IP 地址的基于云的访问规则。
7. 从 Managed IP 列表中，选择公共 IP 或私有 IP。
8. 单击 Submit 以保存计算配置文件。

CLI 过程

- 将 Amazon EC2 详情设置为 compute 配置集：

```
# hammer compute-profile values create
--compute-resource "My_Laptop" \
--compute-profile "My_Compute_Profile" \
--compute-attributes "flavor_id=1,availability_zone=
My_Zone,subnet_id=1,security_group_ids=1,managed_ip=public_ip"
```

14.8. 在 AMAZON EC2 上创建基于镜像的主机

Amazon EC2 调配流程从 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, 进入到 **Hosts > Create Host**。
2. 在 **Name** 字段中输入主机的名称。
3. 可选: 点 **Organization** 选项卡并更改机构上下文以匹配您的要求。
4. 可选: 点 **Location** 选项卡并更改位置上下文以匹配您的要求。
5. 从 **Host Group** 列表中, 选择要为其分配主机的主机组。该主机组将填充表单。
6. 在 **Deploy on** 列表中, 选择 **EC2** 连接。
7. 从 **Compute Profile** 列表中, 选择一个用于自动填充基于虚拟机的设置的配置集。
8. 单击 **Interfaces** 选项卡, 然后在主机的接口上点 **Edit**。
9. 验证字段是否填充了值。特别是:
 - **Satellite** 会自动为新主机分配 IP 地址。
 - 确保 **MAC address** 字段为空。EC2 在置备过程中为主机分配 MAC 地址。
 - **Host** 选项卡中的 **Name** 变为 DNS 名称。
 - 确保 **Satellite** 会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。

10. 单击 **OK** 以保存。要添加另一个接口，请单击 **Add Interface**。您只能为 **Provision** 和 **Primary** 选择一个接口。
11. 单击 **Operating System** 选项卡，并确认所有字段都填充了值。
12. 点 **Virtual Machine** 选项卡，并确认所有字段都填充了值。
13. 单击 **Parameters** 选项卡，并确保存在提供激活密钥的参数。如果没有，添加激活码。
14. 点 **Submit** 保存您的更改。

此新主机条目触发 **Amazon EC2** 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。

CLI 过程

- 使用 **hammer host create** 命令创建主机，并包含 **--provision-method** 镜像 以使用基于镜像的调配。

```
# 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_Host_Name_" \
--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 服务器基本系统** 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. 以用户 **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. 从 **-----BEGIN RSA PRIVATE KEY-----** 后复制密钥，直到 **-----END RSA PRIVATE KEY-----** 为止。

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 实例时使用 Red Hat Satellite 完成模板。

如果要使用带有 SSH 的 Finish 模板，Satellite 必须驻留在 EC2 环境以及正确安全组中。Satellite 目前直接执行 SSH 完成调配，而不使用胶囊服务器。如果 Satellite 服务器不在 EC2 中，EC2 虚拟机会报告内部 IP，而不是可以访问它的必要外部 IP。

流程

1. 在 Satellite Web UI 中，进入到 Hosts > Templates > Provisioning Templates。
2. 在 Provisioning Templates 页面中，在搜索字段中输入 Kickstart default finish，然后单击 Search。
3. 在 Kickstart 默认完成 模板上，选择 Clone。
4. 在 Name 字段中输入模板的唯一名称。
5. 在模板中，为每个命令添加 sudo 需要 root 权限的命令，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 操作系统相关联。
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. 进行所需的任何其他自定义或更改，然后点 **Submit** 以保存模板。
10. 在 **Satellite Web UI** 中，进入到 **Hosts > Operating systems** 并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 **finish** 模板。
12. 在 **Satellite Web UI** 中，进入到 **Hosts > Create Host**。
13. 在 **Name** 字段中输入主机的名称。
14. 可选：点 **Organization** 选项卡并更改机构上下文以匹配您的要求。
15. 可选：点 **Location** 选项卡并更改位置上下文以匹配您的要求。
16. 从 **Host Group** 列表中，选择要为其分配主机的主机组。该主机组将填充表单。
17. 单击 **Parameters** 选项卡，再导航到 **Host** 参数。
18. 在 **Host** 参数中，单击 **Add Parameter** 两次，以添加两个新参数字段。添加以下参数：
 - a. 在 **Name** 字段中输入 **activation_keys**。在对应的 **Value** 字段中输入您的激活码。
 - b. 在 **Name** 字段中输入 **remote_execution_ssh_user**。在对应的 **Value** 字段中，输入 **ec2-user**。
19. 点 **Submit** 以保存更改。

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，进入到 **Infrastructure > Compute Resources**。
2. 选择您的 **Amazon EC2 供应商**。
3. 在 **Virtual Machines** 选项卡上，从 **Actions** 菜单中选择 **Delete**。这会从 Amazon EC2 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。

14.12. 卸载 AMAZON EC2 插件

如果您之前已安装了 Amazon EC2 插件，但不再使用它来管理和部署主机到 EC2，您可以从 Satellite 服务器卸载它。

流程

1. 从 Satellite 服务器卸载 EC2 计算资源供应商：

```
# satellite-maintain packages remove foreman-ec2
# satellite-installer --no-enable-foreman-compute-ec2
```
2. 可选：在 Satellite Web UI 中，进入到 **Administer > About** 并选择 **Available Providers** 选项卡来验证删除 Amazon EC2 插件。

14.13. 有关 AMAZON WEB SERVICES 和 SATELLITE 的更多信息

有关如何在 Amazon Web Services EC2 上查找红帽黄金镜像的详情，请参考 [如何在 AWS EC2 上查找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](#) 部分。

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](#) 部分。

第 15 章 在 GOOGLE COMPUTE ENGINE 上置备云实例

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。

您只能在 Satellite 中使用红帽支持的金级镜像来创建 GCE 主机。

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](#)。
- 您可以将同步的内容软件仓库用于 Red Hat Enterprise Linux。如需更多信息，请参阅 [管理内容](#) 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 [管理内容中的创建激活码](#)。
- 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 project-wise 元数据中，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](#) 部分。
- 可选：如果要使用 Puppet 与 GCE 主机搭配使用，请导航到 `Administer > Settings > Puppet`，并启用 `Use UUID for certificate` 设置，将 Puppet 配置为使用一致的 Puppet 证书 ID。
- 根据您的需要，将 `finish` 或 `user_data` 置备模板与您要使用的操作系统关联。如需更多信息，请参阅 [置备主机](#) 中的 [置备模板](#)。

15.1. 将 GOOGLE GCE 连接添加到 SATELLITE 服务器

在 Satellite 中，使用此流程将 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。如需更多信息，请参阅 GCE 文档中的 [创建和管理服务帐户密钥](#)。
2. 在 Satellite Web UI 中，进入到 **Infrastructure > Compute Resources**，再点 **Create Compute Resource**。
3. 在 **Name** 字段中输入计算资源的名称。
4. 从 **Provider** 列表中，选择 **Google**。
5. 可选：在 **Description** 字段中输入资源的描述。
6. 在 **JSON key** 字段中，单击 **Choose File** 并找到您的服务帐户密钥，以便从本地机器上传。
7. 点 **Load Zones** 从 GCE 环境中填充区列表。
8. 从 **Zone** 列表中，选择要使用的 GCE 区域。
9. 点 **Submit**。

CLI 过程

1. 在 Google GCE 中，以 JSON 格式生成服务帐户密钥。如需更多信息，请参阅 GCE 文档中的 [创建和管理服务帐户密钥](#)。
2. 将文件从本地机器复制到 Satellite 服务器：

```
# scp My_GCE_Key.json root@satellite.example.com:/etc/foreman/My_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 root:foreman /etc/foreman/My_GCE_Key.json
```

4. 在管理门户中，为您的服务帐户密钥配置权限，以确保该文件可读：

```
# chmod 0640 /etc/foreman/My_GCE_Key.json
```

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

```
# restorecon -vv /etc/foreman/My_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 \  
--key-path "/etc/foreman/My_GCE_Key.json" \  
--name "My_GCE_Compute_Resource" \  
--provider "gce" \  
--zone "My_Zone"
```

15.2. 将 GOOGLE COMPUTE ENGINE 镜像添加到 SATELLITE 服务器

要使用基于镜像的调配来创建主机，您必须将有关镜像的信息（如访问详细信息和镜像位置）添加到 Satellite 服务器。

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

流程

1. 在 Satellite Web UI 中，进入到 **Infrastructure > Compute Resources**，点 **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，选择镜像的基本操作系统。

5. 从 **Architecture** 列表中，选择操作系统架构。
6. 在 **Username** 字段中，输入用于镜像访问的 **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，并包含小写字母和数字。
7. 从 **Image** 列表中，从 **Google Compute Engine** 计算资源选择一个镜像。
8. 可选：如果镜像支持用户数据输入，请选择 **User Data** 复选框，如 **cloud-init** 数据。
9. 点 **Submit** 以保存镜像详情。

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，并包含小写字母和数字。

```
# hammer compute-resource image create \
--name 'gce_image_name' \
--compute-resource 'gce_cr' \
--operatingsystem-id 1 \
--architecture-id 1 \
--uuid '3780108136525169178' \
--username 'admin'
```

15.3. 在计算配置集中添加 GOOGLE GCE 详情

使用这个流程将 **Google GCE** 硬件设置添加到计算配置集。当使用此计算配置集在 **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，进入到 **Infrastructure > Compute Profiles**。

- 2.

在 **Compute Profiles** 窗口中，单击现有计算配置文件的名称，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，选择要用于置备的机器类型。
5. 从 **Image** 列表中，选择要用于调配的镜像。
6. 从 **Network** 列表中，选择用于置备的 **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，这个公共 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的大小。
9. 单击 **Submit** 以保存计算配置文件。

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"
```

15.4. 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，进入到 **Hosts > Create Host**。
2. 在 **Name** 字段中输入主机的名称。
3. 可选：点 **Organization** 选项卡并更改机构上下文以匹配您的要求。
4. 可选：点 **Location** 选项卡并更改位置上下文以匹配您的要求。
5. 从 **Host Group** 列表中，选择要为其分配主机的主机组。该主机组将填充表单。
6. 从 **Deploy on** 列表中，选择 **Google Compute Engine** 连接。
7. 从 **Compute Profile** 列表中，选择一个用于自动填充虚拟机设置的配置集。
8. 从 **Lifecycle Environment** 列表中，选择环境。
9. 单击 **Interfaces** 选项卡，然后在主机的接口上点 **Edit**。
10. 验证字段是否填充了值。特别是：
 - **Satellite** 会自动为新主机分配 **IP** 地址。
 - 确保 **MAC address** 字段为空。**Google Compute Engine** 在置备过程中为主机分配 **MAC** 地址。

- **Host** 选项卡中的 **Name** 变为 **DNS** 名称。
 - **Domain** 字段填充所需的域。
 - 确保 **Satellite** 会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** 以保存。要添加另一个接口，请单击 **Add Interface**。您只能为 **Provision** 和 **Primary** 选择一个接口。
 12. 单击 **Operating System** 选项卡，并确认所有字段都自动包含值。
 13. 点 **Provisioning templates** 中的 **Resolve** 检查新主机是否可以识别要使用的正确置备模板。
 14. 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文件的详细信息。修改这些设置以满足您的需要。
 15. 单击 **Parameters** 选项卡，并确保存在提供激活密钥的参数。如果没有，添加激活码。
 16. 单击 **Submit** 以保存主机条目。

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_Compute_Resource" \
--image "My_GCE_Image" \
--interface "type=interface,domain_id=1,managed=true,primary=true,provision=true" \
--location "My_Location" \
--name "My_Host_Name" \
```

```
--operatingsystem "My_Operating_System" \  
--organization "My_Organization" \  
--provision-method "image" \  
--puppet-ca-proxy-id My_Puppet_CA_Proxy_ID \  
--puppet-environment-id My_Puppet_Environment_ID \  
--puppet-proxy-id My_Puppet_Proxy_ID \  
--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。

15.5. 删除 GOOGLE GCE 上的虚拟机

您可以删除 Satellite 服务器上的 Google GCE 上运行的虚拟机。

流程

1. 在 Satellite Web UI 中，进入到 **Infrastructure > Compute Resources**。
2. 选择 **Google GCE 供应商**。
3. 在 **Virtual Machines** 选项卡上，从 **Actions** 菜单中选择 **Delete**。这会从 Google GCE 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。

第 16 章 在 MICROSOFT AZURE RESOURCE MANAGER 上置备云实例

Satellite 可以与 **Microsoft Azure** 资源管理器交互，包括创建新虚拟机并控制其电源管理状态。仅支持基于镜像的调配，才能创建 **Azure** 主机。这包括使用 **Marketplace** 镜像、自定义镜像和共享镜像 **gallery** 进行置备。

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**。如需更多信息，请参阅 [管理内容](#) 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 [管理内容](#) 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 developer](#))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 Client 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](#)，该主体可以访问 **Microsoft 身份平台**([Azure Active Directory for 开发人员](#)) 文档中的资源。
- 可选：如果要 **Puppet** 与 **Azure** 主机搭配使用，请导航到 **Administer > Settings > Puppet**，并启用 **Use UUID for certificate** 设置，将 **Puppet** 配置为使用一致的 **Puppet** 证书 ID。
- 根据您的需要，将 **finish** 或 **user_data** 置备模板与您要使用的操作系统关联。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私有 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** 或门户网站中的镜像术语，以便镜像可用于创建和管理您的订阅的虚拟机。

16.1. 在 SATELLITE 服务器中添加 MICROSOFT AZURE RESOURCE MANAGER 连接

在 Satellite 中使用此流程将 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，进入到 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，云 设置为 Public/Standard。Azure Government Cloud 支持以下区域：
 - 美国政府
 - 中国
 - 德国
6. 在 Client ID 字段中，输入您的 Application 或 client 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。

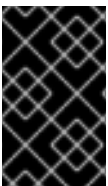
9. 在 **Tenant ID** 字段中, 输入您的 **Directory** 或 **租户 ID**。
10. 单击 **Load Regions**。这会测试您与 **Azure Resource Manager** 的连接是否成功, 并加载您的订阅中可用区域。
11. 从 **Azure Region** 列表中, 选择要使用的 **Azure** 区域。
12. 点 **Submit**。

CLI 过程

- 使用 **hammer compute-resource create** 向 **Satellite** 添加 **Azure** 计算资源。

```
# 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** 选项的值必须为小写, 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
德国	azuregermancloud

16.2. 在 SATELLITE 服务器中添加 MICROSOFT AZURE RESOURCE MANAGER 镜像

要使用基于镜像的调配来创建主机，您必须将有关镜像的信息（如访问详细信息和镜像位置）添加到 **Satellite 服务器**。

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

流程

1. 在 Satellite Web UI 中，进入到 **Infrastructure > Compute Resources**，点 **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，选择镜像的基本操作系统。
5. 从 **Architecture** 列表中，选择操作系统架构。
6. 在 **Username** 字段中，输入用于镜像访问的 **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** 字段中输入前缀：**//UUID** 格式的镜像名称。
 - 对于自定义镜像，请使用前缀 **custom**。例如，**custom://image-name**。
 - 对于共享的 **gallery** 镜像，请使用前缀 **gallery**。例如，**gallery://image-name**。
 - 对于公共和 **RHEL Bring Your Own Subscription (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，请选择 **User Data** 复选框，如 **cloud-init** 数据。
10. 点 **Submit** 以保存镜像详情。

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. 在计算配置集中添加 MICROSOFT AZURE RESOURCE MANAGER 详情

使用这个流程在计算配置集中添加 Microsoft Azure 硬件设置。当使用此计算配置集在 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，进入到 **Infrastructure > Compute Profiles**。
2. 在 **Compute Profiles** 窗口中，单击现有计算配置文件的名称，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。

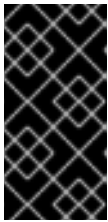
4. 从 **Resource group** 列表中，选择要置备的资源组。
5. 从 **VM Size** 列表中，选择要置备的虚拟机的大小。
6. 从 **Platform** 列表中，选择 **Linux**。
7. 在 **Username** 字段中输入要进行身份验证的用户名。请注意，您在为 **compute** 配置集输入的用户名必须与创建镜像时所用的相同。
8. 要验证用户，请使用以下选项之一：
 - 要使用密码进行身份验证，请在 **Password** 字段中输入密码。
 - 要使用 **SSH** 密钥进行身份验证，请在 **SSH Key** 字段中输入 **SSH** 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 Caching** 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在虚拟机上执行的命令。
12. 可选：如果要在置备完成后运行自定义脚本，在 **Comma separated file URIs** 字段中输入要使用的脚本的逗号分隔列表。脚本在启动时必须包含 **sudo**，因为 **Red Hat Satellite** 会将文件下载到主机和脚本上的 `/var/lib/waagent/custom-script/download/0/` 目录中。
13. 可选：您可以通过选择 **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，输入 **Size in GB** 并选择 **Data Disk Cache** 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 VM 存储要求的更多信息，请参阅 [Microsoft Azure 文档](#)。

15. 点 **Add Interface**。

**重要**

最大接口数量取决于所选的 **VM Size**。如需更多信息，请参阅上面的 [Microsoft Azure 文档](#) 链接。

16. 从 **Azure Subnet** 列表中，选择要置备的 **Azure** 子网。

17. 从 **Public 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私有 **IP**，请选择 **Static Private 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，输入用于镜像访问的 **SSH** 用户名。请注意，您在为 **compute** 配置集输入的用户名必须与创建镜像时所用的相同。

```
# 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=mynetworkID,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（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 使用 `script_uris` 设置运行脚本，请输入要使用的脚本以逗号分隔的文件 URI。脚本在启动时必须包含 `sudo`，因为 Red Hat Satellite 会将文件下载到主机上的 `/var/lib/foreman-proxy/custom-script/download/0/` 目录中，因此脚本需要执行 `sudo` 权限。

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，进入到 Hosts > Create Host。
2. 在 Name 字段中输入主机的名称。
3. 可选：点 Organization 选项卡并更改机构上下文以匹配您的要求。
4. 可选：点 Location 选项卡并更改位置上下文以匹配您的要求。

5. 从 **Host Group** 列表中，选择要为其分配主机的主机组。该主机组将填充表单。
6. 在 **Deploy on** 列表中，选择 **Microsoft Azure Resource Manager** 连接。
7. 从 **Compute Profile** 列表中，选择一个用于自动填充虚拟机设置的配置集。
8. 从 **Lifecycle Environment** 列表中，选择环境。
9. 单击 **Interfaces** 选项卡，然后在主机的接口上点 **Edit**。
10. 验证字段是否填充了值。特别是：
 - **Satellite** 会自动为新主机分配 IP 地址。
 - 确保 **MAC address** 字段为空。**Microsoft Azure Resource Manager** 在置备过程中为主机分配 MAC 地址。
 - **Host** 选项卡中的 **Name** 变为 DNS 名称。
 - **Azure Subnet** 字段填充所需的 Azure 子网。
 - 可选：如果要使用静态私有 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。
 - 确保 **Satellite** 会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** 以保存。要添加另一个接口，请单击 **Add Interface**。您只能为 **Provision** 和 **Primary** 选择一个接口。

12. 单击 **Operating System** 选项卡，并确认所有字段都自动包含值。
13. 对于 **Provisioning Method**，请确保选择了基于 镜像。
14. 在 **Image** 列表中，选择您要用于置备的 **Azure Resource Manager** 镜像。
15. 在 **Root Password** 字段中，输入要进行身份验证的 **root** 密码。
16. 点 **Provisioning templates** 中的 **Resolve** 检查新主机是否可以识别要使用的正确置备模板。
17. 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文件的详细信息。修改这些设置以满足您的需要。
18. 单击 **Parameters** 选项卡，并确保存在提供激活密钥的参数。如果没有，添加激活码。
19. 单击 **Submit** 以保存主机条目。

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_Compute_Resource" \
--domain "My_Domain" \
--image "My_Azure_Image" \
--location "My_Location" \
--name "My_Host_Name" \
--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，进入到 **Infrastructure > Compute Resources**。
2. 选择您的 **Microsoft Azure** 供应商。
3. 在 **Virtual Machines** 选项卡上，从 **Actions** 菜单中选择 **Delete**。这会从 **Microsoft Azure** 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。

16.6. 卸载 MICROSOFT AZURE 插件

如果您之前已安装了 **Microsoft Azure** 插件，但不再使用它来管理主机并将其部署到 **Azure**，您可以从 **Satellite** 服务器卸载它。

流程

1. 从 **Satellite** 服务器卸载 **Azure** 计算资源供应商：

```
# satellite-maintain packages remove rubygem-foreman_azure_rm rubygem-ms_rest_azure
# satellite-installer --no-enable-foreman-plugin-azure
```
2. 可选：在 **Satellite Web UI** 中，进入到 **Administer > About** 并选择 **Available Providers** 选项卡来验证删除 **Microsoft Azure** 插件。

附录 A. 用于置备示例的初始化脚本

如果您还没有遵循 管理内容 中的示例，您可以使用以下初始化脚本为置备示例创建一个环境。

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 "My 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 your application lifecycle
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 your 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 "My initial content view for my 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.

对脚本设置可执行权限：

```
# chmod +x content-init.sh
```

3.

从红帽客户门户网站下载您的 Red Hat Subscription Manifest 的副本，并在清单上运行该脚本：

```
#!/content-init.sh manifest_98f4290e-6c0b-4f37-ba79-3a3ec6e405ba.zip
```

这为本指南中的置备示例导入必要的红帽内容。

附录 B. 置备 FIPS 兼容主机

Satellite 支持置备符合 **Cryptographic 模块** 标准国家标准与技术安全要求的主机，参考编号 **FIPS 140-2**，这里称为 **FIPS**。

要启用 **FIPS 兼容主机**的置备，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“验证启用了 FIPS 模式”** 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与 **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值。**

```
# hammer os update \  
--password-hash SHA256 \  
--title "My_Operating_System"
```

请注意，您无法使用以逗号分隔的值列表。

B.2. 设置启用了 FIPS 的参数

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`。如果这没有设为 `true`，或者不存在，则特定于 FIPS 的更改不会应用到系统。您可在置备主机或主机组时设置此参数。

要在调配主机时设置此参数，请将 `--parameters fips_enabled=true` 附加到 Hammer 命令。

```
# hammer hostgroup set-parameter \  
--hostgroup "My_Host_Group" \  
--name fips_enabled \  
--value "true"
```

如需更多信息，请参阅 `hammer hostgroup set-parameter --help` 命令的输出。

B.3. 验证启用了 FIPS 模式

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 `admin` 级别帐户登录主机。
2. 使用以下命令：

```
$ cat /proc/sys/crypto/fips_enabled
```

值 `1` 确认启用了 FIPS 模式。

附录 C. 为 RED HAT SATELLITE 构建一个云镜像

使用本节在 Red Hat Satellite 中构建和注册镜像。

您可以使用预配置的 Red Hat Enterprise Linux KVM 客户机 QCOW2 镜像：

- [最新 RHEL 9 KVM 客户机镜像](#)
- [最新 RHEL 8 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。
- 此镜像没有设置 root 密码。通过将 !! 放置到第二个字段中，将 root 密码锁定在 /etc/shadow 中。

如果要创建自定义 Red Hat Enterprise Linux 镜像，请参阅[生成自定义的 Red Hat Enterprise Linux 9 镜像](#)，或生成[自定义的 Red Hat Enterprise Linux 8 镜像](#)。

C.1. 创建自定义 RED HAT ENTERPRISE LINUX 镜像

先决条件

- 使用 Linux 主机机器来创建镜像。在这个示例中，我们使用 Red Hat Enterprise Linux 7 Workstation。
- 在工作站上使用 virt-manager 来完成此流程。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 ISO 文件**（请参阅 [Red Hat Enterprise Linux 7.4 Binary DVD](#) 或者 [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 环境的工作站中输入带有 `[root@host]#` 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。

- 对于 **Red Hat Enterprise Linux 7 及更新版本**：

```
# systemctl enable --now chronyd
```

- **Red Hat Enterprise Linux 6**：

```
# chkconfig --add ntpd  
# chkconfig ntpd on  
# service ntpd start
```


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、任何胶囊服务器和所有主机都必须与相同的 NTP 服务器同步，并且启用和运行时间同步工具。**
- 主机必须使用激活码。如需更多信息，[请参阅管理内容中的管理 激活码](#)。
- 可选：如果要注册主机到 Red Hat Insights，您必须同步 `rhel-8-for-x86_64-baseos-rpms` 和 `rhel-8-for-x86_64-appstream-rpms` 存储库，并在您使用的激活码中提供它们。这是在主机上安装 `insights-client` 软件包所必需的。
- 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，进入到 `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：**在 **Host Group** 列表中选择要关联主机的主机组。从 **Host group** 继承值的字段：**操作系统、激活码和生命周期环境**。
5. **可选：**在 **Operating system** 列表中，选择您要注册的主机的操作系统。
6. **可选：**在 **Capsule** 列表中，选择 **Capsule** 来注册主机。

**注意**

负载均衡器后面的胶囊优先于 Satellite Web UI 中选择的胶囊作为主机的内容源。

7. 在 **Activation Keys** 字段中，输入一个或多个要分配给主机的激活码。
8. **可选：**如果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** 文件与以后进行安全的所有调用连接到 **Satellite**。

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，攻击者可以从注册的主机和 **JSON Web 令牌(JWT)** 访问 **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手动复制并安装 **CA** 文件。

要做到这一点，找到 **Satellite** 在哪里通过进入到 **Administer > Settings > Authentication** 并查找 **SSL CA file** 设置的值来保存 **CA** 文件。

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 ...
```

9.

选择 **Advanced** 选项卡。

10.

可选：在 **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**，则将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、操作系统或组织继承。覆盖后，所选值将存储在主机参数级别上。

11.

可选：在 **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上启用以下软件仓库：

- **Red Hat Enterprise Linux 6: rhel-6-server-rpms**
- **Red Hat Enterprise Linux 7: rhel-7-server-rpms**
- **Red Hat Enterprise Linux 8: rhel-8-for-x86_64-appstream-rpms**

`insights-client` 软件包默认安装在 **Red Hat Enterprise Linux 8** 上，除非使用 **"Minimal Install"** 选项部署 **Red Hat Enterprise Linux 8** 的环境中。

12.

可选：在 **Install packages** 字段中，列出您要在注册时在主机上安装的软件包（与空格分开）。这可以通过 `host_packages` 参数设置。

13. **可选：**选择 **Update packages** 选项，在注册时更新主机上的所有软件包。这可以通过 **host_update_packages** 参数设置。
14. **可选：**在 **Repository** 字段中，输入要在执行注册前添加的存储库。例如，使 **subscription-manager** 软件包可供注册使用。对于红帽系列分发，请输入存储库的 URL，例如 **http://rpm.example.com/**。
15. **可选：**在 **Repository GPG key URL** 字段中，指定验证 GPG 签名软件包签名的公钥。它需要在 ASCII 表单中使用 GPG 公钥标头指定。
16. **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生成的 **curl** 命令的工作方式。您可以将持续时间设置为 **0–999 999** 小时，或没有限制。

请注意，Satellite 应用生成 curl 命令授权的用户权限。如果用户丢失或获得额外权限，则 JWT 的权限也会改变。因此，不要在令牌期间删除、阻止或更改用户的权限。

JWT 的范围仅限于注册端点，不能在任何其他地方使用。

17. **可选：**在 **Remote Execution Interface** 字段中输入主机必须用于 SSH 连接的网络接口的标识符。如果将此字段留空，**Satellite** 将使用默认网络接口。
18. **可选：**从 **REX pull 模式** 列表中，选择是否要部署 **Satellite** 远程执行 pull 客户端。

如果设置为 Yes，则将在注册的主机上安装远程执行 pull 客户端。继承的值基于 host_registration_remote_execution_pull 参数。它可以继承，例如从主机组、操作系统或组织继承。覆盖后，所选值存储在主机参数级别上。

注册的主机必须有权访问 Red Hat Satellite Client 6 软件仓库。

有关 pull 模式的更多信息，请参阅管理主机 中的 [远程执行的传输模式](#)。
19. **可选：**如果要忽略订阅管理器 错误，请选择 **Ignore** 错误。

20. 可选：如果要在注册前删除任何 `katello-ca-consumer` 软件包，并选择 **Force** 选项，并使用 `--force` 参数运行 `subscription-manager`。
21. 点 **Generate**。
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. 手动安装和配置 PUPPET 代理

您可以在主机上手动安装和配置 Puppet 代理。主机上需要配置了 Puppet 代理，以便 Puppet 与您的 Satellite 集成。有关 Puppet 的更多信息，请参阅[使用 Puppet 集成管理配置](#)。

先决条件

- Puppet 必须在您的 Satellite 中启用。有关更多信息，请参阅[使用 Puppet 集成管理配置中的启用 Puppet 与 Satellite 集成](#)。
- 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，进入到 **Infrastructure > Capsules**。

8. 从所需胶囊服务器的 **Actions** 列中的列表中，选择 **Certificates**。

9. 单击所需主机右侧的 **Sign**，为 **Puppet 代理** 签署 **SSL 证书**。

10. 在主机上再次运行 **Puppet 代理**：

```
# puppet ssl bootstrap
```

C.6. 完成 RED HAT ENTERPRISE LINUX 7 镜像

流程

1. **更新系统：**

```
# yum 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 cloud_init_modules 标题下，添加：**

```
- resolv-conf
```

resolv-conf 选项会在实例第一次引导时自动配置 **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 元数据服务出现问题：**

```
NOZEROCONF=yes
```

7. **取消注册虚拟机，以便生成的镜像不包含基于它克隆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 上，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` 命令减小镜像大小。这个命令将磁盘镜像中的任何可用空间转换为主机中的可用空间：

```
# 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.7. 完成 RED HAT ENTERPRISE LINUX 6 镜像

流程

1. 更新系统：

```
# yum update
```

2. 安装 `cloud-init` 软件包：

```
# yum install cloud-utils-growpart cloud-init
```

3. 编辑 `/etc/cloud/cloud.cfg` 配置文件以及 `cloud_init_modules add` 下：

```
- resolv-conf
```

`resolv-conf` 选项会在实例第一次引导时自动配置 `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`，则网络在从快照引导时可能无法正常工作（网络接口创建为 "eth1" 而不是 "eth0"，且不会分配 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克隆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` 命令重置和清理镜像，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` 命令减小镜像大小。这个命令将磁盘镜像中的任何可用空间转换为主机中的可用空间：

```
# 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手动调整实例的分区大小。

C.7.1. 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，以备将来使用。**

C.8. 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，以备将来使用。**

附录 D. 主机参数层次结构

您可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	Manage > Organizations
位置级参数	Manage > Locations
域级参数	Infrastructure > Domains
子网级参数	infrastructure > Subnets
操作系统级别参数	Host > Provisioning Setup > Operating Systems
主机组级参数	配置 > Host Groups
主机参数	hosts > All Hosts

附录 E. 置备主机所需的权限

以下列表概述了非 `admin` 用户调配主机所需的权限。

资源名称	权限	其它详情
激活码	view_activation_keys	
Ansible 角色	view_ansible_roles	如果使用 Ansible，则需要此项。
架构	view_architectures	
Compute 配置集	view_compute_profiles	
计算资源	view_compute_resources, create_compute_resources, destroy_compute_resources, power_compute_resources	需要此项以调配裸机主机。
	view_compute_resources_vms, create_compute_resources_vms, destroy_compute_resources_vms, power_compute_resources_vms	需要此项以调配虚拟机。
内容视图	view_content_views	
域	view_domains	
环境	view_environments	
主机	view_hosts, create_hosts, edit_hosts, destroy_hosts, build_hosts, power_hosts, play_roles_on_host	
	view_discovered_hosts, submit_discovered_hosts, auto_provision_discovered_hosts, provision_discovered_hosts, edit_discovered_hosts, destroy_discovered_hosts	如果启用了 Discovery 服务，则需要此项。
hostgroup	view_hostgroups, create_hostgroups, edit_hostgroups, play_roles_on_hostgroup	
Image	view_images	

资源名称	权限	其它详情
生命周期环境	view_lifecycle_environments	
位置	view_locations	
Medium	view_media	
OperatingSystem	view_operatingsystems	
机构 (Organization)	view_organizations	
参数	view_params, create_params, edit_params, destroy_params	
产品和存储库	view_products	
置备模板	view_provisioning_templates	
ptable	view_ptables	
Capsule	view_smart_proxies, view_smart_proxies_puppetca	
	view_openscap_proxies	如果启用了 OpenSCAP 插件，则需要此项。
子网	view_subnets	

其他资源

- [在管理 Red Hat Satellite 中创建角色](#)
- [管理 Red Hat Satellite 中的角色 添加权限](#)
- [管理 Red Hat Satellite 中的用户分配角色 \[https://access.redhat.com/documentation/zh-cn/red_hat_satellite/6.15/html-single/administering_red_hat_satellite/index#Assigning_Roles_to_a_User_admin\]\(https://access.redhat.com/documentation/zh-cn/red_hat_satellite/6.15/html-single/administering_red_hat_satellite/index#Assigning_Roles_to_a_User_admin\)](https://access.redhat.com/documentation/zh-cn/red_hat_satellite/6.15/html-single/administering_red_hat_satellite/index#Assigning_Roles_to_a_User_admin)