Red Hat Satellite 6.15

在断开连接的网络环境中安装 Satellite 服务器

在没有互联网访问的网络中安装和配置 Satellite 服务器

Last Updated: 2024-10-01

Red Hat Satellite 6.15 在断开连接的网络环境中安装 Satellite 服务器

在没有互联网访问的网络中安装和配置 Satellite 服务器

Red Hat Satellite Documentation Team
satellite-doc-list@redhat.com

法律通告

摘要

本指南论述了如何在断开连接的网络中安装 Red Hat Satellite，执行初始配置和配置外部服务。

目录

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 为安装准备您的环境

在安装 Satellite 前，请确保您的环境满足以下要求。

1.1. 系统要求

以下要求适用于联网的基本操作系统：

- x86_64 架构
- Red Hat Enterprise Linux 8 的最新版本
- 至少 4 核 2.0 GHz CPU
- Satellite 服务器至少需要 20 GB RAM。另外，还建议至少 4 GB RAM 交换空间。以 RAM 小于最小值运行的 Satellite 可能无法正确运行。
- 唯一的主机名，可以包含小写字母、数字、点(.)和连字符(-)
- 当前 Red Hat Satellite 订阅
- 管理用户(root)访问
- 使用完全限定域名进行全正向和反向 DNS 解析

Satellite 只支持 **UTF-8** 编码。如果您的个人是美国的，并且您的语言是英语，请将 **en_US.utf-8** 设置为系统范围的区域设置。有关在 Red Hat Enterprise Linux 中 配置系统区域设置的更多信息，请参阅 配置系统本地指南。

您的 Satellite 必须在您的客户门户网站中有 Red Hat Satellite Infrastructure Subscription 清单。Satellite 必须启用并同步 satellite-capsule-6.x 存储库。要在客户门户网站中创建、管理和导出红帽订阅清单，请参阅在 *Subscription Central* 中为连接的 Satellite 服务器创建和管理清单。

Satellite 服务器和 Capsule 服务器不支持主机名中的短名称。使用自定义证书时，自定义证书的通用名称(CN)必须是完全限定域名(FQDN)，而不是短名称。这不适用于 Satellite 的客户端。

在安装 Satellite 服务器前，请确保您的环境满足安装要求。

必须在全新调配的系统上安装卫星服务器，该系统上不提供其他功能，但运行 Satellite 服务器除外。新置备的系统不能有外部身份提供程序提供的以下用户，以避免与 Satellite 服务器创建的本地用户冲突：

- Apache
- Foreman
- foreman-proxy
- postgres
- Pulp
- puppet
- redis
- tomcat

认证的虚拟机监控程序

在运行 Red Hat Enterprise Linux 的虚拟机监控程序上运行的物理系统和虚拟机上完全支持 Satellite 服务器。有关认证虚拟机监控程序的更多信息，请参阅 Red Hat OpenStack Platform、Red Hat Virtualization、Red Hat OpenShift Virtualization 和带有 KVM 的 Red Hat Enterprise Linux 中的认证的客户机操作系统。

SELinux 模式

SELinux 必须启用，可以是 enforcing 模式或 permissive 模式。不支持在禁用 SELinux 的情况下安装。

FIPS 模式

您可以在以 FIPS 模式运行的 Red Hat Enterprise Linux 系统上安装 Satellite。安装 Satellite 后您无法启用 FIPS 模式。如需更多信息，请参阅 *Red Hat Enterprise Linux 8 安全强化* 中的 将 RHEL 切换到 FIPS 模式。

注意

Satellite 支持 DEFAULT 和 FIPS 加密策略。FUTURE 加密策略不支持 Satellite 和 Capsule 安装 。FUTURE 策略是一种更严格的前进安全级别，用于测试可能的未来策略。如需更多信息，请参阅 Red Hat Enterprise Linux 指南中的 使用系统范围的加密策略。

Satellite 间同步(ISS)

在 air-gapped Satellite 服务器的情况下，所有 Satellite 服务器都必须位于同一 Satellite 版本中，才能使 ISS 导出同步正常工作。ISS 网络同步可用于支持它的所有 Satellite 版本。如需更多信息，请参阅管理 内容 中的 在 Satellite 服务器之间 *同步内容*。

1.2. 存储要求

下表详细介绍了特定目录的存储要求。这些值基于预期的用例场景，并根据各个环境的不同而有所不同。

运行时大小由 Red Hat Enterprise Linux 6、7 和 8 软件仓库同步来测量。

表 1.1. Satellite 服务器安装的存储要求

目录	安装大小	运行时大小
/var/log	10 MB	10 GB
/var/lib/pgsql	100 MB	20 GB
/usr	10 GB	不适用
/opt/puppetlabs	500 MB	不适用
/var/lib/pulp	1 MB	300 GB

对于外部数据库服务器： **/var/lib/pgsql**，安装大小为 100 MB，运行时大小为 20 GB。

有关分区和大小的详细信息，请参阅 *Red Hat Enterprise Linux 8 系统设计指南* 中的分区参考。

1.3. 存储指南

安装 Satellite 服务器以提高效率时，请考虑以下准则。

- 如果将 **/tmp** 目录挂载为单独的文件系统，则必须使用 **/etc/fstab** 文件中的 **exec** 挂载选项。如果 **/tmp** 已经挂载了 **noexec** 选项，您必须将选项更改为 **exec** 并重新挂载文件系统。这是 **puppetserver** 服务正常工作的要求。
- 由于大多数 Satellite 服务器数据存储在 **/var** 目录中，所以在 LVM 存储上挂载 **/var** 可帮助系统扩展。
- 对 **/var/lib/pulp/** 目录使用高带宽、低延迟存储。因为 Red Hat Satellite 有很多 I/O 密集型操作，使用高延迟，低带宽存储会导致性能下降。确保您的安装速度在每秒 60 – 80 MB。

您可以使用 **storage-benchmark** 脚本获取此数据。有关使用 **storage-benchmark** 脚本的更多信息，请参阅 对 Satellite 操作的影响。

文件系统指南

- 不要使用 GFS2 文件系统，因为输入输出延迟太高。

日志文件存储

日志文件被写入 **/var/log/messages/**、**/var/log/httpd/** 和 **/var/lib/foreman-proxy/openscap/content/**。您可以使用 **logrotate** 管理这些文件的大小。如需更多信息，请参阅 如何使用 logrotate 工具来轮转日志文件。

日志消息所需的存储量取决于您的安装和设置。

NFS 挂载的 SELinux 注意事项

当使用 NFS 共享挂载 **/var/lib/pulp** 目录时，SELinux 会阻止同步过程。要避免这种情况，请在文件系统表中指定 **/var/lib/pulp** 目录的 SELinux 上下文，方法是在 **/etc/fstab** 中添加以下行：

```
nfs.example.com:/nfsshare  /var/lib/pulp  nfs  context="system_u:object_r:var_lib_t:s0"  1 2
```

如果 NFS 共享已经挂载，请使用上述配置重新挂载它，并输入以下命令：

```
# restorecon -R /var/lib/pulp
```

重复的软件包

不同存储库中重复的软件包仅在磁盘上存储一次。包含重复软件包的其他软件仓库需要较少的额外存储。批量存储位于 **/var/lib/pulp/** 目录中。这些端点无法手动配置。确保 **/var** 文件系统上可用存储以防止存储问题。

符号链接

您不能对 **/var/lib/pulp/** 使用符号链接。

1.4. 支持的操作系统

您可以使用磁盘、本地 ISO 镜像、Kickstart 或者红帽支持的任何其他方法安装操作系统。Red Hat Satellite Server 在最新版本的 Red Hat Enterprise Linux 8 上被支持，该版本可在 Satellite 服务器安装时可用。以前的 Red Hat Enterprise Linux 版本（包括 EUS 或 z-stream）不被支持。

以下操作系统由安装程序支持，有软件包，并经过测试以部署 Satellite：

表 1.2. satellite-installer 支持的操作系统

操作系统	架构	备注
Red Hat Enterprise Linux 8	仅限 x86_64	

红帽建议使用现有系统，因为 Satellite 安装程序会影响多个组件的配置。Red Hat Satellite Server 需要具有 **@Base** 软件包组的 Red Hat Enterprise Linux 安装，没有其他软件包集修改，而不需要第三方配置或软件直接进行服务器直接操作。这个限制包括强化和其他非红帽安全软件。如果您的基础架构中需要此类软件，请首先安装和验证完整的 Satellite 服务器，然后再添加任何非红帽软件。

红帽不支持对运行 Satellite 服务器以外的任何系统使用系统。

1.5. 支持的浏览器

Satellite 支持最新版本的 Firefox 和 Google Chrome 浏览器。

Satellite Web UI 和命令行界面支持英语、简体中文、日语、法语。

1.6. 端口和防火墙要求

要使 Satellite 架构的组件进行通信，请确保在基础操作系统上开放和释放所需的网络端口。您还必须确保在任何基于网络的防火墙上打开所需的网络端口。

使用这些信息来配置任何基于网络的防火墙。请注意，一些解决方案必须专门配置为允许机器之间的通信，因为它们与基于网络的防火墙类似。如果您使用基于应用程序的防火墙，请确保基于应用程序的防火墙允许表中列出的所有应用程序以及防火墙已知的应用程序。如果可能，禁用应用程序检查并允许基于协议打开的端口通信。

集成胶囊

Satellite 服务器具有集成胶囊，且直接连接到 Satellite 服务器的任何主机都是本节上下文中的 Satellite 客户端。这包括在其上运行胶囊式服务器的基本操作系统。

Capsule 的客户端

是胶囊（除 Satellite 集成胶囊之外的）的客户端不需要访问卫星服务器的主机。有关 Satellite 拓扑和端口连接图的更多信息，请参阅 *概述、概念和部署注意事项* 中的 Capsule 网络。

所需端口可能会根据您的配置而改变。

下表指定目的地端口和网络流量的方向：

表 1.3. Satellite 服务器传入流量

目的地端口	协议	服务	源	必需 For	描述
53	TCP 和 UDP	DNS	DNS 服务器和客户端	名称解析	DNS （可选）
67	UDP	DHCP	客户端	动态 IP	DHCP （可选）

69	UDP	TFTP	客户端	TFTP 服务器（可选）	
443	TCP	HTTPS	Capsule	Red Hat Satellite API	来自 Capsule 的通信
443, 80	TCP	HTTPS, HTTP	客户端	全局注册	将主机注册到 Satellite 注册启动、上传事实和发送已安装的软件包和追踪需要端口 443 端口 80 在注册完成的 **/unattended/built** 端点上通知 Satellite
443	TCP	HTTPS	Red Hat Satellite	内容镜像	管理
443	TCP	HTTPS	Red Hat Satellite	Capsule API	智能代理功能
443, 80	TCP	HTTPS, HTTP	Capsule	内容检索	内容
443, 80	TCP	HTTPS, HTTP	客户端	内容检索	内容
1883	TCP	MQTT	客户端	基于 REX 的拉取（可选）	REX 作业通知的内容主机（可选）
5910 – 5930	TCP	HTTPS	浏览器	计算资源的虚拟控制台	
8000	TCP	HTTP	客户端	置备模板	用于客户端安装程序、iPXE 或 UEFI HTTP 引导的模板检索
8000	TCP	HTTPS	客户端	PXE 引导	安装
8140	TCP	HTTPS	客户端	Puppet 代理	客户端更新（可选）
9090	TCP	HTTPS	Red Hat Satellite	Capsule API	智能代理功能

9090	TCP	HTTPS	客户端	OpenSCAP	配置客户端（如果安装了 OpenSCAP 插件）
9090	TCP	HTTPS	发现的节点	Discovery（发现）	主机发现和置备（如果安装了发现插件）

任何直接连接到 Satellite 服务器的主机都是此上下文中的客户端，因为它是集成胶囊的客户端。这包括在其上运行胶囊式服务器的基本操作系统。

DHCP Capsule 将执行 ICMP ping 或 TCP echo 连接尝试子网中 DHCP IPAM 设置的主机，以找出被视为使用的 IP 地址是空闲的。可以使用 **satellite-installer --foreman-proxy-dhcp-ping-free-ip=false** 关闭此行为。

注意

有些传出流量返回到 Satellite，以启用内部通信和安全操作。

表 1.4. Satellite 服务器传出流量

目的地端口	协议	服务	目的地	必需 For	描述
	ICMP	ping	客户端	DHCP	空闲 IP 检查（可选）
7	TCP	echo	客户端	DHCP	空闲 IP 检查（可选）
22	TCP	SSH	目标主机	远程执行	运行作业
22, 16514	TCP	SSH/TLS	计算资源	Satellite 源自通信，用于 libvirt 中的计算资源	
53	TCP 和 UDP	DNS	互联网上的 DNS 服务器	DNS 服务器	解析 DNS 记录（可选）
53	TCP 和 UDP	DNS	DNS 服务器	胶囊 DNS	验证 DNS 冲突（可选）
53	TCP 和 UDP	DNS	DNS 服务器	编配	验证 DNS 冲突
68	UDP	DHCP	客户端	动态 IP	DHCP （可选）
80	TCP	HTTP	远程存储库	内容同步	远程仓库

目的地端口	协议	服务	目的地	必需 For	描述
389, 636	TCP	LDAP, LDAPS	外部 LDAP 服务器	LDAP	LDAP 身份验证，只有在启用了外部身份验证时才需要。定义 **LDAPAuthSource** 时可以自定义端口
443	TCP	HTTPS	Satellite	Capsule	Capsule 配置管理 模板检索 OpenSCAP 远程执行结果上传
443	TCP	HTTPS	Amazon EC2, Azure, Google GCE	计算资源	虚拟机交互 (query/create/destroy)（可选）
443	TCP	HTTPS	Capsule	内容镜像	启动
443	TCP	HTTPS	Infoblox DHCP Server	DHCP 管理	当使用 Infoblox 进行 DHCP 时，管理 DHCP 租期（可选）
623			客户端	电源管理	BMC On/Off/Cycle/Status
5000	TCP	HTTPS	OpenStack 计算资源	计算资源	虚拟机交互 (query/create/destroy)（可选）
5900 – 5930	TCP	SSL/TLS	虚拟机监控程序	noVNC 控制台	启动 noVNC 控制台

目的地端口	协议	服务	目的地	必需 For	描述
7911	TCP	DHCP、OMAPI	DHCP Server	DHCP	DHCP 目标使用 **--foreman-proxy-dhcp-server** 配置，默认为 localhost ISC 和 **remote_isc** 使用默认为 7911 的可配置端口，并使用 OMAPI
8443	TCP	HTTPS	客户端	Discovery（发现）	Capsule 将 reboot 命令发送到发现的主机（可选）
9090	TCP	HTTPS	Capsule	Capsule API	管理 Capsule

1.7. 启用从客户端到 SATELLITE 服务器的连接

属于 Satellite 服务器内部胶囊的客户端的胶囊和内容主机需要通过 Satellite 的基于主机的防火墙和任何基于网络的防火墙访问。

使用这个流程在安装 Satellite 的系统上配置基于主机的防火墙，从客户端启用进入连接，并在系统重启后保留配置。有关使用 的端口 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 端口和防火墙要求*。

流程

1. 在 Satellite 服务器上为客户端打开端口：

```
# firewall-cmd \
--add-port="8000/tcp" \
--add-port="9090/tcp"
```

2. 允许访问 Satellite 服务器上的服务：

```
# firewall-cmd \
--add-service=dns \
--add-service=dhcp \
--add-service=tftp \
--add-service=http \
--add-service=https \
--add-service=puppetmaster
```

3. 使更改持久：

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

验证

- 输入以下命令：

```
# firewall-cmd --list-all
```

如需更多信息，请参阅 *Red Hat Enterprise Linux 8 保护网络* 中的 使用和配置 firewalld。

1.8. 验证 DNS 解析

使用完全限定域名验证完整正向和反向 DNS 解析，以防止安装 Satellite 时出现问题。

流程

1. 确保主机名和本地主机正确解析：

```
# ping -c1 localhost
# ping -c1 `hostname -f` # my_system.domain.com
```

成功名称解析结果结果类似如下：

```
# ping -c1 localhost
PING localhost (127.0.0.1) 56(84) bytes of data.
64 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64 time=0.043 ms

--- localhost ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.043/0.043/0.043/0.000 ms

# ping -c1 `hostname -f`
PING hostname.gateway (XX.XX.XX.XX) 56(84) bytes of data.
64 bytes from hostname.gateway (XX.XX.XX.XX): icmp_seq=1 ttl=64 time=0.019 ms

--- localhost.gateway ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.019/0.019/0.019/0.000 ms
```

2. 要避免使用静态和临时主机名的差异，请输入以下命令设置系统中的所有主机名：

```
# hostnamectl set-hostname name
```

如需更多信息，请参阅 *Red Hat Enterprise Linux 8 配置和管理网络* 中的 使用 hostnamectl 更改主机名。

警告

名称解析对于 Satellite 的操作至关重要。如果 Satellite 无法正确解析其完全限定域名，则内容管理、订阅管理和置备等任务将失败。

1.9. 使用预定义的配置集调整 SATELLITE 服务器

如果您的 Satellite 部署包含超过 5000 个主机，您可以使用预定义的调优配置文件来提高 Satellite 的性能。

请注意，您不能在 Capsules 上使用调优配置集。

您可以根据 Satellite 管理的主机数量和可用的硬件资源选择其中一个配置集。

调优配置文件位于 **/usr/share/foreman-installer/config/foreman.hiera/tuning/sizes** 目录中。

当您使用 **--tuning** 选项运行 **satellite-installer** 命令时，部署配置设置将按照以下顺序应用到 Satellite：

1. **/usr/share/foreman-installer/config/foreman.hiera/tuning/common.yaml** 文件中定义的默认调优配置文件
2. 要应用到部署的调优配置文件，并在 **/usr/share/foreman-installer/config/foreman.hiera/tuning/sizes/** 目录中定义
3. 可选：如果您配置了 **/etc/foreman-installer/custom-hiera.yaml** 文件，Satellite 会应用这些配置设置。

请注意，**/etc/foreman-installer/custom-hiera.yaml** 文件中定义的配置设置会覆盖在调优配置文件中定义的配置设置。

因此，在应用调优配置文件前，您必须比较 **/usr/share/foreman-installer/config/foreman.hiera/tuning/common.yaml** 中默认调优配置文件中定义的配置设置，并从 **/etc/foreman- installer/custom-hiera.yaml** 文件中删除任何重复的配置。

default

主机数量：0wagon-wagon5000
RAM: 20G

CPU 内核数：4

中

主机数量：5001 iwl-wagon10000
RAM: 32G

CPU 内核数：8

大

主机数量： 10001wagon-wagon20000
RAM: 64G

CPU 内核数：16

extra-large

主机数量： 20001.4-1.-wagon60000
RAM：128G

CPU 内核数：32

extra-extra-large

主机数量：60000+
RAM: 256G

CPU 内核数： 48+

流程

1. 可选： 如果您在 Satellite 服务器上配置了 **custom-hiera.yaml** 文件，请将 **/etc/foreman-installer/custom-hiera.yaml** 文件备份到 **custom-hiera.original**。如果文件损坏，您可以使用备份文件将 **/etc/foreman-installer/custom-hiera.yaml** 文件恢复到其原始状态：

```
# cp /etc/foreman-installer/custom-hiera.yaml \
/etc/foreman-installer/custom-hiera.original
```

2. 可选： 如果您在 Satellite 服务器上配置了 **custom-hiera.yaml** 文件，请查看 **/usr/share/foreman-installer/config/foreman.hiera/tuning/common.yaml** 中默认调优配置文件的定义，以及您要应用到 **/usr/share/foreman-installer/config/foreman.hiera/tuning/sizes/** 中的。将配置条目与您的 **/etc/foreman-installer/custom-hiera.yaml** 文件中的条目进行比较，并删除 **/etc/foreman-installer/custom-hiera.yaml** 文件中的任何重复配置设置。

3. 使用您要应用的配置文件的 **--tuning** 选项输入 **satellite-installer** 命令。例如，要应用中型调优配置文件设置，请输入以下命令：

```
# satellite-installer --tuning medium
```

第 2 章 安装 SATELLITE 服务器

当 Satellite 服务器所需的主机在断开连接的环境中时，您可以使用外部计算机下载软件包的 ISO 镜像，并将软件包复制到您要在其上安装 Satellite 服务器的系统中。对于任何其他情况，不建议使用这个方法，因为 ISO 镜像可能不包含最新的更新、错误修复和功能。

使用以下步骤安装 Satellite 服务器、执行初始配置和导入订阅清单。

继续操作前，请考虑与环境相关的清单。如需有关清单的更多信息，*请参阅管理内容中的红帽订阅*。

注意

您不能将 Satellite 服务器注册到自己。

2.1. 下载二进制 DVD 镜像

使用这个流程下载 Red Hat Enterprise Linux 和 Red Hat Satellite 的 ISO 镜像。

流程

1. 访问红帽客户门户 并登录。
2. **单击下载**。
3. 选择 **Red Hat Enterprise Linux**。
4. 确保具有适合您环境的正确产品和版本。
   - **产品变体** 设置为 **Red Hat Enterprise Linux for x86_64**
   - **Version** 设置为您计划用作基础操作系统的产品的最新次要版本。
   - **架构** 设置为 64 位版本。
5. 在 **Product Software** 选项卡中，为最新的 **Red Hat Enterprise Linux for x86_64**版本下载 Binary DVD 镜像。
6. 单击 **下载** 并选择 **Red Hat Satellite**。
7. 确保具有适合您环境的正确产品和版本。
   - **产品变体** 设置为 **Red Hat Satellite**。
   - **Version** 设置为您计划使用的产品的最新次版本。
8. 在 **Product Software** 选项卡中，下载最新 Red Hat Satellite 版本的 Binary DVD 镜像。
9. 将 ISO 文件复制到 Satellite 基础操作系统上的 **/var/tmp** 或者其它可访问的存储设备中。

```
# scp localfile username@hostname:remotefile
```

2.2. 在 RHEL 8 中使用离线软件仓库配置基础操作系统

使用这个流程为 Red Hat Enterprise Linux 8 和 Red Hat Satellite ISO 镜像配置离线软件仓库。

流程

1. 创建一个目录，以用作与基础操作系统版本对应的 ISO 文件的挂载点。

```
# mkdir /media/rhel8
```

2. 将 Red Hat Enterprise Linux 的 ISO 镜像挂载到挂载点。

```
# mount -o loop rhel8-DVD.iso /media/rhel8
```

3. 要复制 ISO 文件的仓库数据文件并更改权限，请输入：

```
# cp /media/rhel8/media.repo /etc/yum.repos.d/rhel8.repo
# chmod u+w /etc/yum.repos.d/rhel8.repo
```

4. 编辑存储库数据文件并添加 **baseurl** 指令。

```
[RHEL8-BaseOS]
name=Red Hat Enterprise Linux BaseOS
mediaid=None
metadata_expire=-1
gpgcheck=0
cost=500
baseurl=file:///media/rhel8/BaseOS/

[RHEL8-AppStream]
name=Red Hat Enterprise Linux Appstream
mediaid=None
metadata_expire=-1
gpgcheck=0
cost=500
baseurl=file:///media/rhel8/AppStream/
```

5. 验证已配置了存储库。

```
# yum repolist
```

6. 创建一个目录，以用作卫星服务器的 ISO 文件的挂载点。

```
# mkdir /media/sat6
```

7. 将 Satellite 服务器的 ISO 镜像挂载到挂载点。

```
# mount -o loop sat6-DVD.iso /media/sat6
```

2.3. 可选：在 SATELLITE 服务器上使用 FAPOLICYD

通过在 Satellite 服务器上启用 **fapolicyd**，您可以通过监控和控制对文件和目录的访问来提供额外的安全层。fapolicyd 守护进程使用 RPM 数据库作为可信二进制文件和脚本的存储库。

您可以在 Satellite 服务器或 Capsule 服务器上打开或关闭 fapolicyd。

2.3.1. 在 Satellite 服务器上安装 fapolicyd

您可以与 Satellite 服务器一起安装 **fapolicyd**，也可以安装到现有的 Satellite 服务器上。如果您要安装 **fapolicyd** 和新的 Satellite 服务器，安装过程会在 Red Hat Enterprise Linux 主机中检测到 fapolicyd，并自动部署 Satellite 服务器规则。

先决条件

- 确保您的主机可以访问 Red Hat Enterprise Linux 的 BaseOS 软件仓库。

流程

1. 对于新安装，请安装 fapolicyd ：

```
# dnf install fapolicyd
```

2. 对于现有安装，使用 satellite-maintain 软件包安装 fapolicyd：

```
# satellite-maintain packages install fapolicyd
```

3. 启动 **fapolicyd** 服务：

```
# systemctl enable --now fapolicyd
```

验证

- 验证 **fapolicyd** 服务是否正常运行：

```
# systemctl status fapolicyd
```

新的 Satellite 服务器或 Capsule 服务器安装

如果新的 Satellite 服务器或 Capsule 服务器安装，请在 Red Hat Enterprise Linux 主机上安装并启用 fapolicyd 后按照标准安装过程进行操作。

其他资源

有关 fapolicyd 的更多信息，请参阅 *Red Hat Enterprise Linux 8 安全强化* 中的 使用 fapolicyd 阻止和允许应用程序。

2.4. 从离线软件仓库安装 SATELLITE 软件包

使用这个流程从离线软件仓库安装 Satellite 软件包。

流程

1. 确保挂载了 Red Hat Enterprise Linux 服务器和 Red Hat Satellite 的 ISO 镜像：

```
# findmnt -t iso9660
```

2. 导入 Red Hat GPG 密钥：

```
# rpm --import /etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
```

3. 确定基本操作系统使用 Binary DVD 镜像是最新的：

```
# dnf upgrade
```

4. 切换到挂载 Satellite ISO 的目录：

```
# cd /media/sat6/
```

5. 在挂载的目录中运行安装脚本：

```
# ./install_packages
```

注意

脚本包含一个启用 **satellite:el8** 模块的命令。启用模块 **satellite:el8** 会警告与 **postgresql:10** 和 **ruby:2.5** 冲突，因为这些模块被设置为 Red Hat Enterprise Linux 8 上的默认模块版本。模块 **satellite:el8** 具有模块 **postgresql:12** 和 **ruby:2.7** 的依赖项，它将通过 **satellite:el8** 模块启用。这些警告不会导致安装过程失败，因此可以安全地忽略。有关 Red Hat Enterprise Linux 8 模块和生命周期流的更多信息，请参阅 Red Hat Enterprise Linux Application Streams 生命周期 。

如果您成功安装了 Satellite 软件包，则会显示以下信息： **Install is complete。请运行 satellite-installer --scenario satellite**。

2.5. 解决软件包依赖关系错误

如果在安装 Satellite 服务器软件包过程中存在软件包依赖项错误，您可以通过从红帽客户门户网站下载并安装软件包来解决错误。有关解析依赖关系错误的更多信息，请参阅 KCS 解决方案 如何使用 yum 输出来解决 yum 依赖项错误？

如果您已成功安装了 Satellite 软件包，请跳过这个过程。

流程

1. 访问红帽客户门户并登录。
2. 单击 **下载**。
3. 点包含您要下载的软件包的产品。
4. 确保您有适用于您的环境的正确 **产品变体**、 版本和架构。
5. 点 **Packages** 选项卡。
6. 在 **Search** 字段中，输入软件包的名称。
7. 点软件包。
8. 从 **Version** 列表中，选择软件包的版本。
9. 在页面底部，单击 **Download Now**。
10. 将软件包复制到 Satellite 基础操作系统。

11. 在 Satellite 服务器上，切换到软件包所在的目录：

```
# cd /path-to-package/
```

12. 本地安装软件包：

```
# dnf install package_name
```

13. 切换到挂载 Satellite ISO 的目录：

```
# cd /media/sat6/
```

14. 通过安装 Satellite 服务器软件包，验证您已解决了软件包依赖项错误。如果有其他软件包依赖项错误，请重复这个过程。

```
# ./install_packages
```

注意

脚本包含一个启用 **satellite:el8** 模块的命令。启用模块 **satellite:el8** 会警告与 **postgresql:10** 和 **ruby:2.5** 冲突，因为这些模块被设置为 Red Hat Enterprise Linux 8 上的默认模块版本。模块 **satellite:el8** 具有模块 **postgresql:12** 和 **ruby:2.7** 的依赖项，它将通过 **satellite:el8** 模块启用。这些警告不会导致安装过程失败，因此可以安全地忽略。有关 Red Hat Enterprise Linux 8 模块和生命周期流的更多信息，请参阅 Red Hat Enterprise Linux Application Streams 生命周期 。

如果您成功安装了 Satellite 软件包，则会显示以下信息：**Install is complete。请运行 satellite-installer --scenario satellite**。

2.6. 使用 CHRONYD 同步系统时钟

为最大程度降低时间偏移的影响，您必须将系统时钟与您要使用网络时间协议(NTP)服务器安装 Satellite 服务器的基本操作系统同步。如果基本操作系统时钟配置不正确，证书验证可能会失败。

有关 **chrony** 套件的更多信息，请参阅 *Red Hat Enterprise Linux 8 配置基本系统设置* 中的 使用 Chrony 套件配置 NTP。

流程

1. 安装 **chrony** 软件包：

```
# dnf install chrony
```

2. 启动并启用 **chronyd** 服务：

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

2.7. 在基本操作系统中安装 SOS 软件包

在基础操作系统上安装 **sos** 软件包，以便您可以从 Red Hat Enterprise Linux 系统中收集配置和诊断信息。您还可以使用它提供初始系统分析，在打开红帽技术支持的服务请求时需要这样做。有关使用 **sos** 的

更多信息，请参阅 知识库解决方案 什么是 sosreport 以及如何在 Red Hat Enterprise Linux 4.6 及之后的版本中创建？

流程

- 安装 **sos** 软件包：

```
# satellite-maintain packages install sos
```

2.8. 配置 SATELLITE 服务器

使用 **satellite-installer** 安装脚本安装 Satellite 服务器。从以下方法中选择：

- 第 2.8.1 节 “配置 Satellite 安装”.这个方法是通过一个或多个命令选项运行安装脚本来执行。命令选项覆盖对应的默认初始配置选项，并记录在 Satellite 回答文件中。您可以根据需要运行脚本来配置任何必要的选项。

2.8.1. 配置 Satellite 安装

此初始配置流程创建机构、位置、用户名和密码。在初始配置后，如果需要，您可以创建额外的机构和位置。初始配置还会在同一服务器上安装 PostgreSQL 数据库。

安装过程可能需要十分钟才能完成。如果您要远程连接到系统，请使用允许挂起和重新附加通信会话的工具，以便在从远程系统断开连接时检查安装过程。**如果您丢失了与运行安装命令的 shell 的连接，请参阅 /var/log/foreman-installer/satellite.log 的日志，以确定进程是否已成功完成。**

注意事项

- **使用 satellite-installer --scenario satellite --help 命令显示最常用的选项和任何默认值。**
- **使用 satellite-installer --scenario satellite --full-help 命令显示高级选项。**
- 为 选项**指定一个有意义的值： --foreman-initial-organization。这可以是您的公司名称。也会创建与值匹配的内部标签，之后无法更改。如果没有指定值，则会创建一个名为 Default Organization 的组织，其标签为 Default_Organization。您可以重命名机构名称，但不能重命名标签。**
- **默认情况下，由安装程序配置的所有配置文件都被管理。当 satellite-installer 运行时，它会使用预期值覆盖对受管文件的任何手动更改。这意味着，在损坏的系统上运行安装程序应该将其恢复到工作顺序，无论进行了什么更改。有关如何对其他服务应用自定义配置的更多信息，请参阅将自定义 配置应用到 Satellite。**

流程

1. 输入以下命令以及您要使用的任何附加选项：

```
# satellite-installer --scenario satellite \
--foreman-initial-organization "My_Organization" \
--foreman-initial-location "My_Location" \
--foreman-initial-admin-username admin_user_name \
--foreman-initial-admin-password admin_password
```

脚本显示其进度，并将日志写入 /var/log/foreman-installer/satellite.log。

2. 卸载 ISO 镜像：

```
# umount /media/sat6
# umount /media/rhel8
```

2.9. 禁用订阅连接

在断开连接的 Satellite 服务器上禁用订阅连接，以避免连接到红帽门户网站。这也会阻止您刷新清单并更新上游权利。

流程

1. 在 Satellite Web UI 中，进入到 Administer > Settings。
2. 单击 Content 选项卡。
3. 将 Subscription Connection Enabled 值设置为 No。

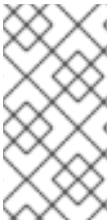
CLI 过程

- 在 Satellite 服务器上输入以下命令：

```
# hammer settings set --name subscription_connection_enabled --value false
```

2.10. 将红帽订阅清单导入到 SATELLITE 服务器中

使用以下步骤将红帽订阅清单导入到 Satellite 服务器中。



注意

在机构上设置简单内容访问(SCA)，而不是清单。导入清单不会更改您机构的简单内容访问状态。

先决条件

- 确保您有一个从红帽客户门户网站导出的红帽订阅清单。如需更多信息，请参阅 *Subscription Central* 中的 对断开连接的 Satellite 服务器使用清单。
- 确保您在 Satellite 服务器上禁用订阅连接。更多信息请参阅 第 2.9 节 “禁用订阅连接”。

流程

1. 在 Satellite Web UI 中，确保将上下文设置为您要使用的组织。
2. 在 Satellite Web UI 中，进入到 Content > Subscriptions 并点 Manage Manifest。
3. 在 Manage Manifest 窗口中，单击 Choose File。
4. 导航到包含红帽订阅清单文件的位置，然后单击 Open。

CLI 过程

1. 将红帽订阅清单文件从本地机器复制到 Satellite 服务器：

```
$ scp ~/manifest_file.zip root@satellite.example.com:~/.
```

2. 以 root 用户身份登录 Satellite 服务器，再导入 Red Hat 订阅清单文件：

```
# hammer subscription upload \
--file ~/manifest_file.zip \
--organization "My_Organization"
```

现在，您可以启用软件仓库并导入红帽内容。如需更多信息，请参阅 *管理内容 中的 导入内容*。

第 3 章 在 SATELLITE 服务器上执行其他配置

3.1. 配置 SATELLITE 服务器以使用自定义 CDN 的内容

如果您在一个可访问的 web 服务器上有一个内部 Content Delivery Network (CDN)或提供内容，您可以将 Satellite 服务器配置为使用来自 CDN 服务器的红帽软件仓库。CDN 服务器可以是在与红帽 CDN 相同的目录结构中镜像软件仓库的任何 Web 服务器。

您可以为每个机构配置内容源。Satellite 可自动识别您机构中订阅清单中的哪些红帽存储库在您的 CDN 服务器上可用。

先决条件

- 您有一个提供红帽内容的 CDN 服务器，可由 Satellite 服务器访问。
- 如果您的 CDN 服务器使用 HTTPS，请确保您已将 SSL 证书上传到 Satellite。如需更多信息，请参阅管理内容中的导入自定义 SSL *证书*。
- 您已将清单上传到您的机构。

流程

1. 在 Satellite Web UI 中，进入到 Content > Subscriptions。
2. 单击 Manage Manifest。
3. 选择 CDN Configuration 选项卡。
4. 选择 Custom CDN 选项卡。
5. 在 URL 字段中，输入您希望 Satellite 服务器使用来自红帽存储库的 CDN 服务器的 URL。
6. 可选：在 SSL CA Content Credential 中，选择 CDN 服务器的 SSL 证书。

7. 点 Update。

8. 现在，您可以启用内部 CDN 服务器消耗的红帽软件仓库。

CLI 过程

1. 使用 SSH 连接到您的 Satellite 服务器。

2. 对自定义 CDN 服务器设置 CDN 配置：

```
# hammer organization configure-cdn --name="My_Organization" \
--type=custom_cdn \
--url https://my-cdn.example.com \
--ssl-ca-credential-id "My_CDN_CA_Cert_ID"
```

其他资源

- *概述、概念和部署注意事项*中的 内容交付网络结构

3.2. 配置 SATELLITE 同步(ISS)

在断开连接的 Satellite 服务器上配置 Inter-Satellite 同步，以在断开连接的网络中提供内容。

3.2.1. Satellite 同步场景

Red Hat Satellite 使用 Inter-Satellite Synchronization (ISS)在两个 Satellite 服务器（包括 air gapped ）之间同步内容。

您可以在以下情况下使用 ISS：

- 如果要将一些内容从 Satellite 服务器复制到其他 Satellite 服务器。例如，您的 IT 部门从 Satellite 服务器使用的内容视图，您希望将这些内容视图从这些内容视图复制到其他 Satellite 服务器。

- 如果要将所有库内容从 Satellite 服务器复制到其他 Satellite 服务器。例如，您的 IT 部门可

从库中 Satellite 服务器使用的产品和存储库，您希望将该机构中的所有产品和存储库复制到其他 Satellite 服务器。

注意

您不能使用 ISS 将内容从 Satellite 服务器同步到胶囊服务器。胶囊服务器原生支持同步。如需更多信息，请参阅 概述 、*概念和部署注意事项* 中的 Capsule 服务器 概述。

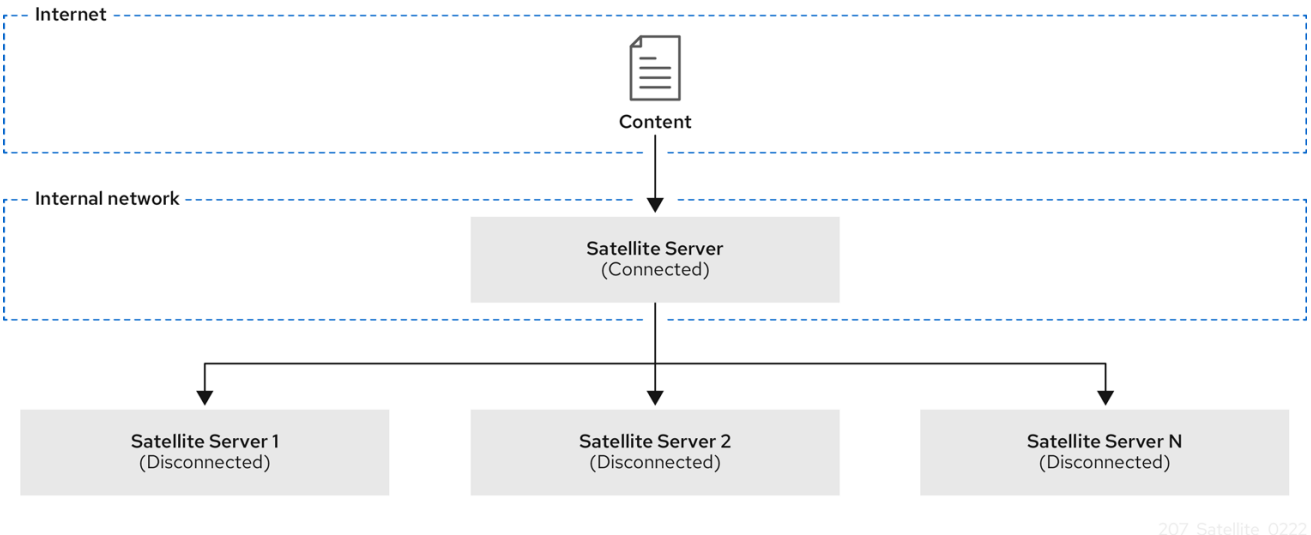
使用 ISS 的方法有多种。您可以使用的方式取决于您的多服务器设置，这些设置可能回退到以下场景之一：

3.2.1.1. 在断开连接的场景中 ISS 网络同步

在断开连接的场景中，有以下设置：

- 上游 Satellite 服务器连接到互联网。这个服务器会消耗来自 Red Hat Content Delivery Network (CDN)或自定义源的内容。
- 下游卫星服务器完全与所有外部网络隔离。
- 下游卫星服务器可以通过内部网络与连接的上游 Satellite 服务器通信。

图 3.1. Satellite ISS 断开连接的场景



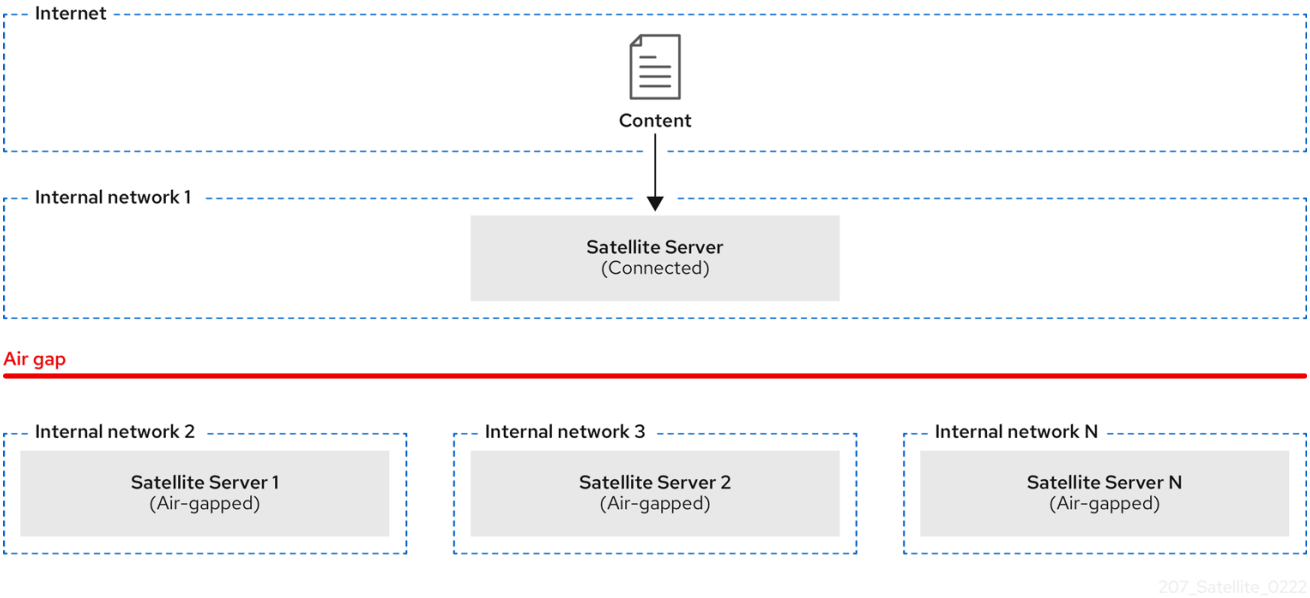
您可以将下游 Satellite 服务器配置为通过网络同步上游 Satellite 服务器的内容。

3.2.1.2. 在 air-gapped 场景中 ISS 导出同步

在 air-gapped 场景中，有以下设置：

- 上游 Satellite 服务器连接到互联网。这个服务器会消耗来自 Red Hat CDN 或自定义源的内容。
- 下游卫星服务器完全与所有外部网络隔离。
- 下游卫星服务器没有连接到上游 Satellite 服务器的网络连接。

图 3.2. Satellite ISS air-gapped 场景



air-gapped downstream Satellite 服务器接收内容更新的唯一方法是从上游 Satellite 服务器导出有效负载，使其物理到下游 Satellite 服务器，并导入有效负载。如需更多信息，请参阅管理 内容 中的 在 Satellite 服务器之间 *同步内容*。

您可以使用导出将下游 Satellite 服务器配置为同步内容。

3.2.2. 使用导出将 Satellite 服务器配置为同步内容

如果您将下游 Satellite 服务器部署为 air gapped，请配置您的 Satellite 服务器，以避免尝试使用来自网络的内容。

流程

1. 在 Satellite Web UI 中，进入到 Content > Subscriptions。
2. 单击 Manage Manifest。
3. 切换到 CDN 配置 选项卡。
4. 选择 Export Sync 选项卡。
5. 点 Update。

CLI 过程

1. 使用 SSH 登录您的 Satellite 服务器。
2. 使用导出将 CDN 配置设置为同步：

```
# hammer organization configure-cdn --name="My_Organization" --type=export_sync
```

其他资源

- 如需更多信息，请参阅 管理内容 中的 使用导出和导入 进行 *内容同步*。

3.2.3. 配置 Satellite 服务器来通过网络同步内容

配置下游卫星服务器，以通过 HTTPS 从连接的上游 Satellite 服务器同步存储库。

先决条件

- 上游 Satellite 服务器和下游 Satellite 服务器之间存在网络连接。
- 您在上游和下游 Satellite 服务器上导入了订阅清单。
- 在上游 Satellite 服务器上，您可以为机构启用了所需的存储库。如需更多信息，*请参阅管理内容中的启用红帽存储库。*
- 上游用户是 admin 或具有以下权限：
  - view_organizations
  - view_products
  - export_content
  - view_lifecycle_environments
  - view_content_views
- 在下游 Satellite 服务器上，您已使用 http://*upstream-satellite.example.com*/pub/katello-server-ca.crt 的内容导入上游 Satellite 服务器的 SSL 证书。如需更多信息，请参阅管理内容中的 导入 SSL *证书*。
- 下游用户是 admin 用户，或具有创建产品存储库和机构的权限。

流程

1. 导航到 Content > Subscriptions。
2. 单击 Manage Manifest。

3. 导航到 **CDN Configuration** 选项卡。

4. 选择 **Network Sync** 选项卡。

5. 在 **URL** 字段中，输入上游 **Satellite** 服务器的地址。

6. 在 **Username** 中，输入上游登录的用户名。

7. 在 **Password** 中，输入您的密码或个人访问令牌以进行上游登录。

8. 在 **Organization label** 字段中，输入上游机构标签。

9. 可选：在 **Lifecycle Environment Label** 字段中输入上游生命周期环境的标签。默认为 **Library**。

10. 可选：在 **Content view** 标签 字段中，输入上游内容视图的标签。默认为 **Default_Organization_View**。

11. 从 **SSL CA Content Credential** 菜单，选择上游 **Satellite** 服务器使用的 **CA** 证书。

12. 点 **Update**。

13. 在 **Satellite Web UI** 中，进入到 **Content > Products**。

14. 选择包含您要同步的存储库的产品。

15. 在 **Select Action** 菜单中，选择 **Sync Now** 以同步产品中的所有存储库。

您还可以创建同步计划以确保定期更新。如需更多信息，*请参阅管理内容中的创建同步计划*。

CLI 过程

1. 使用 SSH 连接到您的下游 Satellite 服务器。

2. 查看上游 CA 证书的信息：

```
# hammer content-credential show \
--name="My_Upstream_CA_Cert" \
--organization="My_Downstream_Organization"
```

记下下一步的 CA 证书的 ID。

3. 对上游 Satellite 服务器设置 CDN 配置：

```
# hammer organization configure-cdn --name="My_Downstream_Organization" \
--type=network_sync \
--url https://upstream-satellite.example.com \
--username upstream_username --password upstream_password \
--ssl-ca-credential-id "My_Upstream_CA_Cert_ID" \ --upstream-organization-
label="_My_Upstream_Organization" \
[--upstream-lifecycle-environment-label="My_Lifecycle_Environment"] \
[--upstream-content-view-label="My_Content_View"]
```

默认生命周期环境标签是 Library。默认内容视图标签为 Default_Organization_View。

3.3. 导入 KICKSTART 软件仓库

Kickstart 软件仓库不是由内容 ISO 镜像提供。要在断开连接的 Satellite 中使用 Kickstart 软件仓库，您必须为您要使用的 Red Hat Enterprise Linux 版本下载二进制 DVD ISO 文件，并将 Kickstart 文件复制到 Satellite。

3.3.1. 为 Red Hat Enterprise Linux 9 导入 Kickstart 软件仓库

使用这个流程为 Red Hat Enterprise Linux 9 导入 Kickstart 软件仓库。

流程

1. 访问红帽客户门户，请访问 access.redhat.com/downloads 并登录。

2. 点 Red Hat Enterprise Linux。

3. 从列表中选择产品变体和产品版本。例如，产品变体 Red Hat Enterprise Linux for x86_64 和产品版本 9.0。

4. 找到完整的安装镜像，例如，Red Hat Enterprise Linux 9.0 Binary DVD，然后点 Download Now。请注意，您无法使用最小 ISO 置备主机。

5. 下载完成后，将 ISO 镜像复制到 Satellite 服务器。

6. 在 Satellite 服务器上，创建一个挂载点，并将 ISO 镜像临时挂载到该位置：

```
# mkdir /mnt/iso
# mount -o loop rhel-binary-dvd.iso /mnt/iso
```

将 *rhel-binary-dvd.iso* 替换为 ISO 镜像的名称。

7. 为 Red Hat Enterprise Linux 9 AppStream 和 BaseOS Kickstart 软件仓库创建目录：

```
# mkdir --parents /var/www/html/pub/satellite-
import/content/dist/rhel9/9.0/x86_64/appstream/kickstart
# mkdir --parents /var/www/html/pub/satellite-
import/content/dist/rhel9/9.0/x86_64/baseos/kickstart
```

8. 从 ISO 镜像复制 kickstart 文件：

```
# cp -a /mnt/iso/AppStream/* /var/www/html/pub/satellite-
import/content/dist/rhel9/9.0/x86_64/appstream/kickstart
# cp -a /mnt/iso/BaseOS/* /mnt/iso/images/ /var/www/html/pub/satellite-
import/content/dist/rhel9/9.0/x86_64/baseos/kickstart
```

请注意，对于 BaseOS，还必须复制 /mnt/iso/images/ 目录的内容。

9. 在列表文件中添加以下条目：

在 /var/www/html/pub/satellite-import/content/dist/rhel9/9.0/x86_64/appstream/listing

文件中，附加带有新行的 kickstart。

在 /var/www/html/pub/satellite-import/content/dist/rhel9/9.0/x86_64/baseos/listing 文件中，附加带有新行的 kickstart。

在 /var/www/html/pub/satellite-import/content/dist/rhel8/listing 文件中，附加带有新行的版本号。例如，对于 Red Hat Enterprise Linux 9.0 二进制 ISO，请附加 9.0。

10. 从 ISO 镜像复制 .treeinfo 文件：

```
# cp /mnt/iso/.treeinfo /var/www/html/pub/satellite-
import/content/dist/rhel9/9.0/x86_64/appstream/kickstart/treeinfo
# cp /mnt/iso/.treeinfo /var/www/html/pub/satellite-
import/content/dist/rhel9/9.0/x86_64/baseos/kickstart/treeinfo
```

11. 打开 /var/www/html/pub/satellite-import/content/dist/rhel9/9.0/x86_64/baseos/kickstart/treeinfo 文件进行编辑。

12. 在 [general] 部分中，进行以下更改：

- 将 packagedir = AppStream/Packages 改为 packagedir = Packages
- 将 repository = AppStream 更改为 repository = .
- 将 variant = AppStream 改为 variant = BaseOS
- 将 变体 = AppStream,BaseOS 改为 variants = BaseOS

13. 在 [tree] 部分中，将 variants = AppStream,BaseOS 更改为 variants = BaseOS。

14. 在 [variant-BaseOS] 部分进行以下更改：

- 将软件包 = BaseOS/Packages 更改为 packages= Packages
- 将 repository = BaseOS 更改为 repository = .

15. 删除 [media] 和 [variant-AppStream] 部分。

16. 保存并关闭该文件。

17. 验证 /var/www/html/pub/satellite-import/content/dist/rhel9/9.0/x86_64/baseos/kickstart/treeinfo 文件是否具有以下格式：

```
[checksums]
images/efiboot.img =
sha256:c01c18acc6778d6e66c8d0872bac59bfd7219ccf3cfa70a5c605c0fb37f33a83
images/install.img =
sha256:ddd08e5a5d92edee150f91ff4f12f39253eae72ff496465cf1b2766fe4a4df49
images/pxeboot/initrd.img =
sha256:a09a8ec89d485d71ed1bdad83584d6d816e67448221172d9aad97886cd70adca
images/pxeboot/vmlinuz =
sha256:6e523d7c3266e26c695923ab12b2873b16b0c61fb2e48ade608ad8998821584b

[general]
; WARNING.0 = This section provides compatibility with pre-productmd treeinfos.
; WARNING.1 = Read productmd documentation for details about new format.
arch = x86_64
family = Red Hat Enterprise Linux
name = Red Hat Enterprise Linux 9.0.0
packagedir = Packages
platforms = x86_64,xen
repository = .
timestamp = 1571146127
variant = BaseOS
variants = BaseOS
version = 9.0.0

[header]
type = productmd.treeinfo
version = 1.2

[images-x86_64]
efiboot.img = images/efiboot.img
initrd = images/pxeboot/initrd.img
kernel = images/pxeboot/vmlinuz

[images-xen]
initrd = images/pxeboot/initrd.img
kernel = images/pxeboot/vmlinuz
```

```
[release]
name = Red Hat Enterprise Linux
short = RHEL
version = 9.0.0

[stage2]
mainimage = images/install.img

[tree]
arch = x86_64
build_timestamp = 1571146127
platforms = x86_64,xen
variants = BaseOS

[variant-BaseOS]
id = BaseOS
name = BaseOS
packages = Packages
repository = .
type = variant
uid = BaseOS
```

18. **打开 /var/www/html/pub/satellite-import/content/dist/rhel9/9.0/x86_64/appstream/kickstart/treeinfo 文件进行编辑。**

19. **在 [general] 部分中，进行以下更改：**

    - **将 packagedir = AppStream/Packages 改为 packagedir = Packages**
    - **将 repository = AppStream 更改为 repository = .**
    - **更改 变体 = AppStream,BaseOS to variants = AppStream**

20. **在 [tree] 部分中，将 variants = AppStream,BaseOS 改为 variants = AppStream**

21. **在 [variant-AppStream] 部分中进行以下更改：**

    - **将软件包 = AppStream/Packages 改为 packages = Packages**

- 将 **repository = AppStream** 更改为 **repository = .**

22. 删除文件中的以下部分：**[checksums] , [images-x86_64], [images-xen], [media], [stage2], [variant-BaseOS]**。

23. 保存并关闭该文件。

24. 验证 **/var/www/html/pub/satellite-import/content/dist/rhel9/9.0/x86_64/appstream/kickstart/treeinfo** 文件是否具有以下格式：

```
[general]
; WARNING.0 = This section provides compatibility with pre-productmd treeinfos.
; WARNING.1 = Read productmd documentation for details about new format.
arch = x86_64
family = Red Hat Enterprise Linux
name = Red Hat Enterprise Linux 9.0.0
packagedir = Packages
platforms = x86_64,xen
repository = .
timestamp = 1571146127
variant = AppStream
variants = AppStream
version = 9.0.0

[header]
type = productmd.treeinfo
version = 1.2

[release]
name = Red Hat Enterprise Linux
short = RHEL
version = 9.0.0

[tree]
arch = x86_64
build_timestamp = 1571146127
platforms = x86_64,xen
variants = AppStream

[variant-AppStream]
id = AppStream
name = AppStream
packages = Packages
repository = .
type = variant
uid = AppStream
```

25. 如果您不打算使用挂载的二进制 DVD ISO 镜像，卸载并删除该目录：

```
# umount /mnt/iso
# rmdir /mnt/iso
```

26. 在 Satellite Web UI 中，启用 Kickstart 存储库。

3.3.2. 为 Red Hat Enterprise Linux 8 导入 Kickstart 软件仓库

使用这个流程为 Red Hat Enterprise Linux 8 导入 Kickstart 软件仓库。

流程

1. 访问红帽客户门户，请访问 access.redhat.com/downloads 并登录。
2. 点 Red Hat Enterprise Linux。
3. 从列表中选择产品变体和产品版本。例如，产品变体 Red Hat Enterprise Linux for x86_64 和产品版本 8.1。
4. 找到完整的安装镜像，例如： Red Hat Enterprise Linux 8.1 Binary DVD，然后点 Download Now。
5. 下载完成后，将 ISO 镜像复制到 Satellite 服务器。
6. 在 Satellite 服务器上，创建一个挂载点，并将 ISO 镜像临时挂载到该位置：

```
# mkdir /mnt/iso
# mount -o loop rhel-binary-dvd.iso /mnt/iso
```

将 *rhel-binary-dvd.iso* 替换为 ISO 镜像的名称。

7. 为 Red Hat Enterprise Linux 8 AppStream 和 BaseOS Kickstart 软件仓库创建目录：

```
# mkdir --parents /var/www/html/pub/satellite-
import/content/dist/rhel8/8.1/x86_64/appstream/kickstart
# mkdir --parents /var/www/html/pub/satellite-
import/content/dist/rhel8/8.1/x86_64/baseos/kickstart
```

8. **从 ISO 镜像复制 kickstart 文件：**

```
# cp -a /mnt/iso/AppStream/* /var/www/html/pub/satellite-
import/content/dist/rhel8/8.1/x86_64/appstream/kickstart
# cp -a /mnt/iso/BaseOS/* /mnt/iso/images/ /var/www/html/pub/satellite-
import/content/dist/rhel8/8.1/x86_64/baseos/kickstart
```

请注意，对于 BaseOS，还必须复制 /mnt/iso/images/ 目录的内容。

9. **在列表文件中添加以下条目：**

在 /var/www/html/pub/satellite-import/content/dist/rhel8/8.1/x86_64/appstream/listing 文件中，附加带有新行的 kickstart。

在 /var/www/html/pub/satellite-import/content/dist/rhel8/8.1/x86_64/baseos/listing 文件中，附加带有新行的 kickstart。

在 /var/www/html/pub/satellite-import/content/dist/rhel8/listing 文件中，附加带有新行的版本号。例如，对于 Red Hat Enterprise Linux 8.1 二进制 ISO，请附加 8.1。

10. **从 ISO 镜像复制 .treeinfo 文件：**

```
# cp /mnt/iso/.treeinfo /var/www/html/pub/satellite-
import/content/dist/rhel8/8.1/x86_64/appstream/kickstart/treeinfo
# cp /mnt/iso/.treeinfo /var/www/html/pub/satellite-
import/content/dist/rhel8/8.1/x86_64/baseos/kickstart/treeinfo
```

11. **打开 /var/www/html/pub/satellite-import/content/dist/rhel8/8.1/x86_64/baseos/kickstart/treeinfo 文件进行编辑。**

12. **在 [general] 部分中，进行以下更改：**

- 将 packagedir = AppStream/Packages 改为 packagedir = Packages
- 将 repository = AppStream 更改为 repository = .
- 将 variant = AppStream 改为 variant = BaseOS
- 将 变体 = AppStream,BaseOS 改为 variants = BaseOS

13. 在 [tree] 部分中，将 variants = AppStream,BaseOS 更改为 variants = BaseOS。

14. 在 [variant-BaseOS] 部分进行以下更改：

- 将软件包 = BaseOS/Packages 更改为 packages= Packages
- 将 repository = BaseOS 更改为 repository = .

15. 删除 [media] 和 [variant-AppStream] 部分。

16. 保存并关闭该文件。

17. 验证 /var/www/html/pub/satellite-import/content/dist/rhel8/8.1/x86_64/baseos/kickstart/treeinfo 文件是否具有以下格式：

```
[checksums]
images/efiboot.img =
sha256:c01c18acc6778d6e66c8d0872bac59bfd7219ccf3cfa70a5c605c0fb37f33a83
images/install.img =
sha256:ddd08e5a5d92edee150f91ff4f12f39253eae72ff496465cf1b2766fe4a4df49
images/pxeboot/initrd.img =
sha256:a09a8ec89d485d71ed1bdad83584d6d816e67448221172d9aad97886cd70adca
images/pxeboot/vmlinuz =
sha256:6e523d7c3266e26c695923ab12b2873b16b0c61fb2e48ade608ad8998821584b

[general]
; WARNING.0 = This section provides compatibility with pre-productmd treeinfos.
```

```
; WARNING.1 = Read productmd documentation for details about new format.
arch = x86_64
family = Red Hat Enterprise Linux
name = Red Hat Enterprise Linux 8.1.0
packagedir = Packages
platforms = x86_64,xen
repository = .
timestamp = 1571146127
variant = BaseOS
variants = BaseOS
version = 8.1.0

[header]
type = productmd.treeinfo
version = 1.2

[images-x86_64]
efiboot.img = images/efiboot.img
initrd = images/pxeboot/initrd.img
kernel = images/pxeboot/vmlinuz

[images-xen]
initrd = images/pxeboot/initrd.img
kernel = images/pxeboot/vmlinuz

[release]
name = Red Hat Enterprise Linux
short = RHEL
version = 8.1.0

[stage2]
mainimage = images/install.img

[tree]
arch = x86_64
build_timestamp = 1571146127
platforms = x86_64,xen
variants = BaseOS

[variant-BaseOS]
id = BaseOS
name = BaseOS
packages = Packages
repository = .
type = variant
uid = BaseOS
```

18. **打开 /var/www/html/pub/satellite-import/content/dist/rhel8/8.1/x86_64/appstream/kickstart/treeinfo 文件进行编辑。**

19. **在 [general] 部分中，进行以下更改：**

- 将 packagedir = AppStream/Packages 改为 packagedir = Packages
- 将 repository = AppStream 更改为 repository = .
- 更改 变体 = AppStream,BaseOS to variants = AppStream

20. 在 [tree] 部分中，将 variants = AppStream,BaseOS 改为 variants = AppStream

21. 在 [variant-AppStream] 部分中进行以下更改：

- 将软件包 = AppStream/Packages 改为 packages = Packages
- 将 repository = AppStream 更改为 repository = .

22. 删除文件中的以下部分：[checksums] , [images-x86_64], [images-xen], [media], [stage2], [variant-BaseOS]。

23. 保存并关闭该文件。

24. 验证 /var/www/html/pub/satellite-import/content/dist/rhel8/8.1/x86_64/appstream/kickstart/treeinfo 文件是否具有以下格式：

```
[general]
; WARNING.0 = This section provides compatibility with pre-productmd treeinfos.
; WARNING.1 = Read productmd documentation for details about new format.
arch = x86_64
family = Red Hat Enterprise Linux
name = Red Hat Enterprise Linux 8.1.0
packagedir = Packages
platforms = x86_64,xen
repository = .
timestamp = 1571146127
variant = AppStream
variants = AppStream
version = 8.1.0

[header]
```

```
type = productmd.treeinfo
version = 1.2

[release]
name = Red Hat Enterprise Linux
short = RHEL
version = 8.1.0

[tree]
arch = x86_64
build_timestamp = 1571146127
platforms = x86_64,xen
variants = AppStream

[variant-AppStream]
id = AppStream
name = AppStream
packages = Packages
repository = .
type = variant
uid = AppStream
```

25. **如果您不打算使用挂载的二进制 DVD ISO 镜像，卸载并删除该目录：**

```
# umount /mnt/iso
# rmdir /mnt/iso
```

26. **在 Satellite Web UI 中，启用 Kickstart 存储库。**

3.3.3. 为 Red Hat Enterprise Linux 7 导入 Kickstart 软件仓库

使用这个流程为 Red Hat Enterprise Linux 7 导入 Kickstart 软件仓库。

流程

1. 访问红帽客户门户，请访问 **access.redhat.com/downloads 并登录。**
2. **点 Red Hat Enterprise Linux。**
3. **点 Product Variant 列表上面的 Switch to version 7 and below。**
4. **从列表中选择产品变体和产品版本。例如，产品变体 Red Hat Enterprise Linux for x86_64**

和产品版本 7.9。

5. 找到完整的安装镜像，例如： Red Hat Enterprise Linux 7.9 Binary DVD，然后点 Download Now。

6. 下载完成后，将 ISO 镜像复制到 Satellite 服务器。

7. 在 Satellite 服务器上，创建一个挂载点，并将 ISO 镜像临时挂载到该位置：

```
# mkdir /mnt/iso
# mount -o loop rhel-binary-dvd.iso /mnt/iso
```

将 ***rhel-binary-dvd.iso*** 替换为 ISO 镜像的名称。

8. 创建 Kickstart 目录：

```
# mkdir --parents /var/www/html/pub/satellite-
import/content/dist/rhel/server/7/7.9/x86_64/kickstart/
```

9. 从 ISO 镜像复制 kickstart 文件：

```
# cp -a /mnt/iso/* /var/www/html/pub/satellite-
import/content/dist/rhel/server/7/7.9/x86_64/kickstart/
```

10. 在列表文件中添加以下条目：

在 /var/www/html/pub/satellite-import/content/dist/rhel/server/7/listing 文件中，附加带有新行的版本号。例如，对于 Red Hat Enterprise Linux 7.9 ISO，请附加 7.9。

在 /var/www/html/pub/satellite-import/content/dist/rhel/server/7/7.9/listing 文件中，附加带有新行的架构。例如，x86_64。

在 /var/www/html/pub/satellite-import/content/dist/rhel/server/7/7.9/x86_64/listing 文件中，附加带有新行的 kickstart。

11. 从 ISO 镜像复制 .treeinfo 文件：

```
# cp /mnt/iso/.treeinfo /var/www/html/pub/satellite-
import/content/dist/rhel/server/7/7.9/x86_64/kickstart/treeinfo
```

12. 如果您不打算使用挂载的二进制 DVD ISO 镜像，卸载并删除该目录：

```
# umount /mnt/iso
# rmdir /mnt/iso
```

13. 在 Satellite Web UI 中，启用 Kickstart 存储库。

3.4. 启用和同步 RED HAT SATELLITE CLIENT 6 软件仓库

Red Hat Satellite Client 6 存储库为注册到 Satellite 的主机提供 katello-host-tools 和 puppet 软件包。您必须定期将软件仓库从 Red Hat Content Delivery Network (CDN)同步到 Satellite 服务器，并在主机上启用存储库。

3.4.1. 为 Red Hat Enterprise Linux 9 和 Red Hat Enterprise Linux 8 同步 Red Hat Satellite Client 6 软件仓库

要使用 CLI 而不是 Satellite Web UI，请查看与 Red Hat Enterprise Linux 版本相关的流程：

- Red Hat Enterprise Linux 9 的 CLI 步骤
- Red Hat Enterprise Linux 8 的 CLI 步骤

流程

1. 在 Satellite Web UI 中，进入到 Content > Sync Status。
2. 单击 Red Hat Enterprise Linux for x86_64 产品旁边的箭头，以查看可用的内容。
3. 选择 Red Hat Satellite Client 6 for RHEL 9 x86_64 RPMs 或 Red Hat Satellite Client 6 for RHEL 8 x86_64 RPMs。

4. 点 **Synchronize Now**。

Red Hat Enterprise Linux 9 的 CLI 步骤

- 同步 **Red Hat Satellite Client 6** 存储库：

```
# hammer repository synchronize \
--name "Red Hat Satellite Client 6 for RHEL 9 x86_64 RPMs" \
--organization "My_Organization" \
--product "Red Hat Enterprise Linux for x86_64"
```

Red Hat Enterprise Linux 8 的 CLI 步骤

- 同步 **Red Hat Satellite Client 6** 存储库：

```
# hammer repository synchronize \
--name "Red Hat Satellite Client 6 for RHEL 8 x86_64 RPMs" \
--organization "My_Organization" \
--product "Red Hat Enterprise Linux for x86_64"
```

其他资源

- 有关 **hammer 存储库 synchronize** 命令的详情，请输入 **hammer 存储库 synchronize --help**。

3.4.2. 同步 Red Hat Enterprise Linux 7 和 Red Hat Enterprise Linux 6 的 Red Hat Satellite Client 6 软件仓库

注意

您需要 **Red Hat Enterprise Linux** 延长生命周期支持(ELS)附加服务 来同步 **Red Hat Enterprise Linux 6** 的软件仓库。如需更多信息，请参阅 **Red Hat Enterprise Linux Extended Lifecycle Support (ELS)附加服务 指南**。

要使用 **CLI** 而不是 **Satellite Web UI**，请查看与 **Red Hat Enterprise Linux** 版本相关的流程：

- Red Hat Enterprise Linux 7 的 CLI 步骤

- Red Hat Enterprise Linux 6 的 CLI 步骤

流程

1. 在 Satellite Web UI 中，进入到 Content > Sync Status。
2. 单击 Red Hat Enterprise Linux Server 或 Red Hat Enterprise Linux Server - Extended Lifecycle Support 旁的箭头。
3. 选择 Red Hat Satellite Client 6 (for RHEL 7 Server) RPMs x86_64 或 Red Hat Satellite Client 6 for RHEL 6 Server - 基于您的操作系统版本的 ELS RPMs x86_64。
4. 点 Synchronize Now。

Red Hat Enterprise Linux 7 的 CLI 步骤

- 同步 Red Hat Satellite Client 6 存储库：

```
# hammer repository synchronize \
--async \
--name "Red Hat Satellite Client 6 for RHEL 7 Server RPMs x86_64" \
--organization "My_Organization" \
--product "Red Hat Enterprise Linux Server"
```

Red Hat Enterprise Linux 6 的 CLI 步骤

- 同步 Red Hat Satellite Client 6 存储库：

```
# hammer repository synchronize \
--async \
--name "Red Hat Satellite Client 6 for RHEL 6 Server - ELS RPMs x86_64" \
--organization "My_Organization" \
--product "Red Hat Enterprise Linux Server - Extended Lifecycle Support"
```

其他资源

- 有关 hammer 存储库 synchronize 命令的详情，请输入 hammer 存储库 synchronize --help。

3.4.3. 为 Red Hat Enterprise Linux 9 和 Red Hat Enterprise Linux 8 启用 Red Hat Satellite Client 6 软件仓库

要使用 CLI 而不是 Satellite Web UI，请查看与 Red Hat Enterprise Linux 版本相关的流程：

- Red Hat Enterprise Linux 9 的 CLI 步骤
- Red Hat Enterprise Linux 8 的 CLI 步骤

先决条件

- 确保您将需要的所有内容 ISO 镜像导入到 Satellite 服务器中。

流程

1. 在 Satellite Web UI 中，进入到 Content > Red Hat Repositories。
2. 在 Available Repositories 窗格中，启用 recommended Repositories 来获取存储库列表。
3. 点 Red Hat Satellite Client 6 for RHEL 9 x86_64 (RPMs) 或 Red Hat Satellite Client 6 for RHEL 8 x86_64 (RPMs) 以展开存储库集。
4. 对于 x86_64 架构，请单击 + 图标以启用该存储库。

如果 Red Hat Satellite Client 6 项不可见，这可能是因为它们没有包含在从客户门户网站获取的红帽订阅清单中。要更正这一点，请登录到客户门户网站，添加这些软件仓库，下载红帽订阅清单并将其导入到 Satellite。如需更多信息，*请参阅管理内容中的管理红帽订阅*。

为主机上运行的每个受支持的 Red Hat Enterprise Linux 主版本启用 Red Hat Satellite Client 6 软件仓库。启用红帽软件仓库后，会自动创建此软件仓库的产品。

Red Hat Enterprise Linux 9 的 CLI 步骤

- 启用 **Red Hat Satellite Client 6** 存储库：

```
# hammer repository-set enable \
--basearch="x86_64" \
--name "Red Hat Satellite Client 6 for RHEL 9 x86_64 (RPMs)" \
--organization "My_Organization" \
--product "Red Hat Enterprise Linux for x86_64"
```

Red Hat Enterprise Linux 8 的 CLI 步骤

- 启用 **Red Hat Satellite Client 6** 存储库：

```
# hammer repository-set enable \
--basearch="x86_64" \
--name "Red Hat Satellite Client 6 for RHEL 8 x86_64 (RPMs)" \
--organization "My_Organization" \
--product "Red Hat Enterprise Linux for x86_64"
```

其他资源

- **有关 hammer repository-set enable 命令的详情，请输入 hammer repository-set enable --help。**

3.4.4. 为 Red Hat Enterprise Linux 7 和 Red Hat Enterprise Linux 6 启用 Red Hat Satellite Client 6 软件仓库

注意

您需要 Red Hat Enterprise Linux 延长生命周期支持(ELS)附加服务 才能启用 Red Hat Enterprise Linux 6 的软件仓库。如需更多信息，请参阅 Red Hat Enterprise Linux Extended Lifecycle Support (ELS)附加服务 指南。

要使用 CLI 而不是 Satellite Web UI，请查看与 Red Hat Enterprise Linux 版本相关的流程：

- **Red Hat Enterprise Linux 7 的 CLI 步骤**
- **Red Hat Enterprise Linux 6 的 CLI 步骤**

先决条件

- 确保您将需要的所有内容 ISO 镜像导入到 Satellite 服务器。提供

1. 在 Satellite Web UI 中，进入到 Content > Red Hat Repositories。

2. 在 Available Repositories 窗格中，启用 recommended Repositories 来获取存储库列表。

3. 在 Available Repositories 窗格中，点 Red Hat Satellite Client 6 (for RHEL 7 Server) (RPMs) 或 Red Hat Satellite Client 6 (for RHEL 6 Server - ELS) (RPMs) 来扩展存储库集。

   如果 Red Hat Satellite Client 6 项不可见，这可能是因为它们没有包含在从客户门户网站获取的红帽订阅清单中。要更正这一点，请登录到客户门户网站，添加这些软件仓库，下载红帽订阅清单并将其导入到 Satellite。如需更多信息，*请参阅管理内容中的管理红帽订阅*。

4. 对于 x86_64 架构，请单击 + 图标以启用该存储库。为主机上运行的每个受支持的 Red Hat Enterprise Linux 主版本启用 Red Hat Satellite Client 6 软件仓库。启用红帽软件仓库后，会自动创建此软件仓库的产品。

Red Hat Enterprise Linux 7 的 CLI 步骤

- 启用 Red Hat Satellite Client 6 存储库：

```
# hammer repository-set enable \
--basearch="x86_64" \
--name "Red Hat Satellite Client 6 (for RHEL 7 Server) (RPMs)" \
--organization "My_Organization" \
--product "Red Hat Enterprise Linux Server"
```

Red Hat Enterprise Linux 6 的 CLI 步骤

- 启用 Red Hat Satellite Client 6 存储库：

```
# hammer repository-set enable \
--basearch="x86_64" \
--name "Red Hat Satellite Client 6 (for RHEL 6 Server - ELS) (RPMs)" \
--organization "My_Organization" \
--product "Red Hat Enterprise Linux Server - Extended Lifecycle Support"
```

其他资源

- 有关 **hammer repository-set enable** 命令的详情，请输入 **hammer repository-set enable --help**。

3.5. 为远程执行配置基于拉取的传输

默认情况下，远程执行使用基于推送的 SSH 作为脚本供应商的传输机制。如果您的基础架构禁止从 Satellite Server 到主机的传出连接，您可以使用带有基于 pull 的传输的远程执行，因为主机启动与 Satellite 服务器的连接。使用基于拉取的传输不会限制这些基础架构。

基于拉取的传输在 Capsule 上包括 pull-mqtt 模式，以及主机上运行的拉取客户端。

注意

pull-mqtt 模式仅适用于 Script 提供程序。Ansible 和其他提供程序将继续使用其默认传输设置。

流程

1. 在 Satellite 服务器上启用基于拉取的传输：

```
# satellite-installer --foreman-proxy-plugin-remote-execution-script-mode=pull-mqtt
```

2. 配置防火墙以允许端口 1883 上的 MQTT 服务：

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

3. 使更改持久：

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

4. 在 pull-mqtt 模式中，主机向 Satellite 服务器或他们注册的任何胶囊服务器订阅作业通知。确保 Satellite 服务器将远程执行作业发送到相同的 Satellite 服务器或 Capsule 服务器：

   a. 在 Satellite Web UI 中，进入到 **Administer > Settings**。

b. 在 Content 选项卡上，将 Prefer registered through Capsule for remote execution 的值设为 Yes 。

后续步骤

- 为基于拉取的传输配置主机。如需更多信息， *请参阅管理主机中的远程执行的传输模式*。

3.6. 在主机上启用电源管理

要使用智能平台管理接口(IPMI)或类似协议在主机上执行电源管理任务，您必须在 Satellite 服务器上启用基板管理控制器(BMC)模块。

先决条件

- 所有主机都必须具有 BMC 类型的网络接口。Satellite 服务器使用此 NIC 将适当的凭据传递给主机。如需更多信息，请参阅管理主机中的 添加基板管理控制器(BMC) *接口*。

流程

- 要启用 BMC，请输入以下命令：

```
# satellite-installer \
--foreman-proxy-bmc "true" \
--foreman-proxy-bmc-default-provider "freeipmi"
```

3.7. 配置 DNS、DHCP 和 TFTP

您可以在 Satellite 环境中集中管理 DNS、DHCP 和 TFTP，也可以在禁用其 Satellite 维护后独立管理它们。您还可以在 Satellite 环境外部运行 DNS、DHCP 和 TFTP。

3.7.1. 在控制台中配置 DNS、DHCP 和 TFTP

要在 Satellite 服务器上配置 DNS、DHCP 和 TFTP 服务，请使用 satellite-installer 命令及适合您环境的选项。

对设置的任何更改都需要再次输入 satellite-installer 命令。您可以多次输入命令，每次使用更改后的值更新所有配置文件。

先决条件

- 确保以下信息可供您使用：
  - DHCP IP 地址范围
  - DHCP 网关 IP 地址
  - DHCP 名称服务器 IP 地址
  - DNS 信息
  - TFTP 服务器名称
- 在网络更改时，请使用 FQDN 而不是 IP 地址。
- 请联系您的网络管理员，以确保您有正确的设置。

流程

- 输入 **satellite-installer** 命令以及适合您的环境的选项。以下示例显示了配置完整置备服务：

```
# satellite-installer \
--foreman-proxy-dns true \
--foreman-proxy-dns-managed true \
--foreman-proxy-dns-zone example.com \
--foreman-proxy-dns-reverse 2.0.192.in-addr.arpa \
--foreman-proxy-dhcp true \
--foreman-proxy-dhcp-managed true \
--foreman-proxy-dhcp-range "192.0.2.100 192.0.2.150" \
--foreman-proxy-dhcp-gateway 192.0.2.1 \
--foreman-proxy-dhcp-nameservers 192.0.2.2 \
--foreman-proxy-tftp true \
--foreman-proxy-tftp-managed true \
--foreman-proxy-tftp-servername 192.0.2.3
```

您可以监控提示符中显示的 **satellite-installer** 命令的进度。您可以查看 **/var/log/foreman-**

installer/satellite.log 中的日志。

其他资源

- 如需有关 satellite-installer 命令的更多信息，请输入 satellite-installer --help。

3.7.2. 为非受管网络禁用 DNS、DHCP 和 TFTP

如果要手动管理 TFTP、DHCP 和 DNS 服务，您必须防止 Satellite 在操作系统上维护这些服务，并禁用编配以避免 DHCP 和 DNS 验证错误。但是，Satellite 不会删除操作系统中的后端服务。

流程

1. 在 Satellite 服务器上输入以下命令：

```
# satellite-installer --foreman-proxy-dhcp false \
--foreman-proxy-dns false \
--foreman-proxy-tftp false
```

2. 在 Satellite Web UI 中，进入到 Infrastructure > Subnets 并选择子网。

3. 单击 Capsules 选项卡，再清除 DHCP Capsule、TFTP Capsule 和 反向 DNS Capsule 字段。

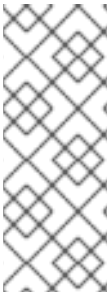
4. 在 Satellite Web UI 中，进入到 Infrastructure > Domains 并选择域。

5. 清除 DNS Capsule 字段。

6. 可选：如果您使用由第三方提供的 DHCP 服务，请将 DHCP 服务器配置为传递以下选项：

```
Option 66: IP address of Satellite or Capsule
Option 67: /pxelinux.0
```

有关 DHCP 选项的更多信息，请参阅 RFC 2132。



注意

当没有为给定子网和域设置胶囊时，Satellite 不会执行编排。在启用或禁用 Capsule 关联时，如果预期的记录和配置文件不存在，现有主机的编配命令可能会失败。当关联一个 Capsule 以进行打开电源编配时，请确保现有 Satellite 主机所需的 DHCP 和 DNS 记录以及 TFTP 文件，以防止主机删除失败。

3.7.3. 其他资源

- 有关外部配置 DNS、DHCP 和 TFTP 的详情，请参考 *第 4 章 使用外部服务配置 Satellite 服务器*。
- 有关配置 DHCP、DNS 和 TFTP 服务的更多信息，请参阅 *置备主机 中的 配置网络服务*。

3.8. 为出站电子邮件配置 SATELLITE 服务器

要从 Satellite 服务器发送电子邮件消息，您可以使用 SMTP 服务器或 sendmail 命令。

前提条件

- 已知的某些具有反垃圾邮件保护或问候功能的 SMTP 服务器会导致问题。要设置具有此类服务的传出电子邮件，可以安装和配置卫星服务器上的 vanilla SMTP 服务进行转发，或者使用 sendmail 命令。

流程

1. 在 Satellite Web UI 中，进入到 Administer > Settings。
2. 单击 Email 选项卡，并将配置选项设置为与您首选的发送方法匹配。更改会立即生效。
   a. 以下示例显示了使用 SMTP 服务器的配置选项：

表 3.1. 使用 SMTP 服务器作为发送方法

Name	示例值
交付方法	SMTP

Name	示例值
SMTP 地址	*smtp.example.com*
SMTP 身份验证	login
SMTP HELO/EHLO 域	*example.com*
SMTP 密码	*password*
SMTP 端口	25
SMTP 用户名	*user@example.com*

SMTP 用户名和 SMTP 密码 指定 SMTP 服务器的登录凭据。

b.

以下示例使用 gmail.com 作为 SMTP 服务器：

表 3.2. 使用 gmail.com 作为 SMTP 服务器

Name	示例值
交付方法	SMTP
SMTP 地址	smtp.gmail.com
SMTP 身份验证	plain
SMTP HELO/EHLO 域	smtp.gmail.com
SMTP 启用 StartTLS 自动	是
SMTP 密码	*password*
SMTP 端口	587
SMTP 用户名	*user*@gmail.com

c.

以下示例使用 sendmail 命令作为发送方法：

表 3.3. 使用 sendmail 作为发送方法

Name	示例值
交付方法	sendmail
Sendmail 位置	/usr/sbin/sendmail
Sendmail 参数	-i

出于安全考虑，Sendmail 位置和 Sendmail 参数设置都是只读的，只能在 /etc/foreman/settings.yaml 中设置。目前无法通过 satellite-installer 设置这两个设置。如需更多信息，请参阅 sendmail 1 man page。

3. 如果您决定使用 TLS 验证的 SMTP 服务器发送电子邮件，请执行以下步骤之一：

- 将 SMTP 服务器的 CA 证书标记为可信。要做到这一点，请在 Satellite 服务器上执行以下命令：

```
# cp mailca.crt /etc/pki/ca-trust/source/anchors/
# update-ca-trust enable
# update-ca-trust
```

其中 *mailca.crt* 是 SMTP 服务器的 CA 证书。

- 或者，在 Satellite Web UI 中，将 SMTP enable StartTLS auto 选项设置为 No。

4. 单击 Test email，将测试消息发送到用户的电子邮件地址，以确认配置是否正常工作。如果消息无法发送，Satellite Web UI 会显示错误。详情请查看 /var/log/foreman/production.log 的日志。

其他资源

- 有关为单个用户或用户组配置电子邮件通知的详情，请参考 管理 *Red Hat Satellite 中的 配置电子邮件通知* 首选项。

3.9. 使用自定义 SSL 证书配置 SATELLITE 服务器

默认情况下，Red Hat Satellite 使用自签名 SSL 证书来启用 Satellite 服务器、外部胶囊服务器和所有主机之间的加密通信。如果无法使用 Satellite 自签名证书，您可以将 Satellite 服务器配置为使用由外部

证书颁发机构(CA)签名的 SSL 证书。

当使用自定义 SSL 证书配置 Red Hat Satellite 时，您必须满足以下要求：

- 您必须对 SSL 证书使用隐私增强型邮件(PEM)编码。
- 您不能将相同的 SSL 证书用于 Satellite 服务器和 Capsule 服务器。
- 同一 CA 必须为 Satellite 服务器和 Capsule 服务器签名证书。
- SSL 证书也不能是 CA 证书。
- SSL 证书必须包含与通用名称(CN)匹配的主题 alt name (SAN)条目。
- 在使用密钥用法扩展时，必须允许 SSL 证书。
- SSL 证书不能有短名称作为 CN。
- 您不能为私钥设置密码短语。

要使用自定义证书配置 Satellite 服务器，请完成以下步骤：

1. 第 3.9.1 节 “为 Satellite 服务器创建自定义 SSL 证书”
2. 第 3.9.2 节 “将自定义 SSL 证书部署到 Satellite 服务器”
3. 第 3.9.3 节 “将自定义 SSL 证书部署到主机”
4. 如果您的外部胶囊服务器注册到 Satellite 服务器，请使用自定义 SSL 证书进行配置。如需更

多信息，请参阅安装 Capsule 服务器 中的 使用自定义 SSL 证书配置 *Capsule 服务器*。

3.9.1. 为 Satellite 服务器创建自定义 SSL 证书

使用这个流程为 Satellite 服务器创建自定义 SSL 证书。如果您已有 Satellite 服务器的自定义 SSL 证书，请跳过此步骤。

流程

1. 要存储所有源证书文件，请创建一个只能被 root 用户访问的目录：

```
# mkdir /root/satellite_cert
```

2. 创建为证书签名请求(CSR)签名的私钥。

请注意，私钥必须未加密。如果您使用密码保护的私钥，请删除私钥密码。

如果您已有此 Satellite 服务器的私钥，请跳过这一步。

```
# openssl genrsa -out /root/satellite_cert/satellite_cert_key.pem 4096
```

3. 为 CSR 创建 /root/satellite_cert/openssl.cnf 配置文件并包含以下内容：

```
[ req ]
req_extensions = v3_req
distinguished_name = req_distinguished_name
prompt = no

[ req_distinguished_name ]
commonName = satellite.example.com

[ v3_req ]
basicConstraints = CA:FALSE
keyUsage = digitalSignature, nonRepudiation, keyEncipherment, dataEncipherment
extendedKeyUsage = serverAuth, clientAuth, codeSigning, emailProtection
subjectAltName = @alt_names

[ alt_names ]
DNS.1 = satellite.example.com
```

4. 可选： 如果要向 CSR 添加可辨识名称(DN)详情，请在 [req_distinguished_name] 部分添

加以下信息：

```
[req_distinguished_name]
CN = satellite.example.com
countryName =My_Country_Name (1)
stateOrProvinceName = My_State_Or_Province_Name (2)
localityName = My_Locality_Name (3)
organizationName = My_Organization_Or_Company_Name
organizationalUnitName = My_Organizational_Unit_Name (4)
```

(1) **两个字母代码**

(2) **全名**

(3) **全名（例如：New York）**

(4) **负责证书的部门（示例：IT 部门）**

5. **生成 CSR：**

```
# openssl req -new \
-key /root/satellite_cert/satellite_cert_key.pem \ (1)
-config /root/satellite_cert/openssl.cnf \ (2)
-out /root/satellite_cert/satellite_cert_csr.pem (3)
```

(1) **私钥的路径**

(2) **配置文件的路径**

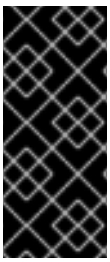
(3) **要生成的 CSR 的路径**

6. 将证书签名请求发送到证书颁发机构(CA)。同一 CA 必须为 Satellite 服务器和 Capsule 服务器签名证书。

提交请求时，指定证书的寿命。发送证书请求的方法会有所不同，因此请查阅 CA 查看首选方法。为了响应请求，您可以在单独的文件中接收 CA 捆绑包和签名证书。

3.9.2. 将自定义 SSL 证书部署到 Satellite 服务器

使用这个流程将 Satellite 服务器配置为使用证书颁发机构签名的自定义 SSL 证书。katello-certs-check 命令验证输入证书文件，并返回将自定义 SSL 证书部署到 Satellite 服务器所需的命令。



重要

不要将 SSL 证书或 .tar 捆绑包存储在 /tmp 或 /var/tmp 目录中。操作系统定期从这些目录中删除文件。因此，satellite-installer 在启用功能或升级 Satellite 服务器时无法执行。

流程

1. 验证自定义 SSL 证书输入文件。请注意，对于 katello-certs-check 命令正常工作，证书中的通用名称(CN)必须与 Satellite 服务器的 FQDN 匹配。

```
# katello-certs-check \
-c /root/satellite_cert/satellite_cert.pem \ 1
-k /root/satellite_cert/satellite_cert_key.pem \ 2
-b /root/satellite_cert/ca_cert_bundle.pem 3
```

1. 由证书颁发机构签名的 Satellite 服务器证书文件的路径。
2. 用于为 Satellite 服务器证书签名的私钥的路径。
3. 证书颁发机构捆绑包的路径。

如果命令成功，它会返回两个 satellite-installer 命令，其中之一必须用于部署证书到 Satellite 服务器。

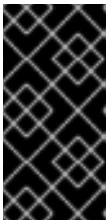
katello-certs-check的输出示例

```
Validation succeeded.

To install the Red Hat Satellite Server with the custom certificates, run:

  satellite-installer --scenario satellite \
    --certs-server-cert "/root/satellite_cert/satellite_cert.pem" \
    --certs-server-key "/root/satellite_cert/satellite_cert_key.pem" \
    --certs-server-ca-cert "/root/satellite_cert/ca_cert_bundle.pem"

To update the certificates on a currently running Red Hat Satellite installation, run:

  satellite-installer --scenario satellite \
    --certs-server-cert "/root/satellite_cert/satellite_cert.pem" \
    --certs-server-key "/root/satellite_cert/satellite_cert_key.pem" \
    --certs-server-ca-cert "/root/satellite_cert/ca_cert_bundle.pem" \
    --certs-update-server --certs-update-server-ca
```

请注意，不得访问或修改 /root/ssl-build。

2. 根据您的要求，在 katello-certs-check 命令的输出中，输入 satellite-installer 命令，该命令使用自定义 SSL 证书或更新当前运行的 Satellite 上的证书。

如果您不确定要运行的命令，您可以通过检查文件 /etc/foreman-installer/scenarios.d/.installed 来验证是否安装了 Satellite。如果文件存在，请运行第二个 satellite-installer 命令来更新证书。



重要

在部署证书后，Satellite-installer 需要证书存档文件。不要修改或删除它。例如，升级 Satellite 服务器时需要这样做。

3. 在可访问 Satellite 服务器的计算机上，导航到以下 URL ：

4. 在您的浏览器中，查看证书详情以验证部署的证书。

3.9.3. 将自定义 SSL 证书部署到主机

将 Satellite 配置为使用自定义 SSL 证书后，您必须将证书部署到注册到 Satellite 的主机。

流程

- 更新每个主机上的 SSL 证书：

```
# dnf install http://satellite.example.com/pub/katello-ca-consumer-latest.noarch.rpm
```

3.10. 在 SATELLITE 中使用外部数据库

作为 Red Hat Satellite 的安装过程的一部分，satellite-installer 命令会在与 Satellite 相同的服务器上安装 PostgreSQL 数据库。在某些 Satellite 部署中，使用外部数据库而不是默认的本地数据库可帮助进行服务器负载。

红帽不提供对外部数据库维护的支持或工具。这包括备份、升级和数据库调优。您必须具有自己的数据库管理员才能支持和维护外部数据库。

要为 Satellite 创建和使用外部数据库，您必须完成以下步骤：

1. 第 3.10.2 节 “为外部数据库准备主机”.准备托管外部数据库的 Red Hat Enterprise Linux 8 服务器。
2. 第 3.10.3 节 “安装 PostgreSQL”.使用 Satellite 的数据库准备 PostgreSQL，使用拥有他们的专用用户 Candlepin 和 Pulp。
3. 第 3.10.4 节 “将 Satellite 服务器配置为使用外部数据库”.编辑 satellite-installer 的参数以指向新数据库，并运行 satellite-installer。

3.10.1. PostgreSQL 作为外部数据库注意事项

Foreman、Katello 和 Candlepin 使用 PostgreSQL 数据库。如果要将 PostgreSQL 用作外部数据

库，则以下信息可帮助您确定此选项是否适合您的 Satellite 配置。Satellite 支持 PostgreSQL 版本 12。

外部 PostgreSQL 的优点

- 在 Satellite 上增加可用内存和可用 CPU
- 在 PostgreSQL 数据库上设置 shared_buffers 的灵活性，使其没有与 Satellite 上的其他服务干扰的风险
- 在不影响 Satellite 操作的情况下灵活地调整 PostgreSQL 服务器系统

外部 PostgreSQL 的缺点

- 增加部署复杂性，使故障排除变得更加困难
- 外部 PostgreSQL 服务器是一个额外的系统来修补和维护
- 如果 Satellite 或 PostgreSQL 数据库服务器都存在硬件或存储故障，则 Satellite 无法正常工作
- 如果 Satellite 服务器和数据库服务器之间有延迟，则性能可能会下降

如果您怀疑 Satellite 上的 PostgreSQL 数据库导致性能问题，请在 Satellite 6 中使用信息：如何启用 postgres 查询日志记录来检测运行较慢的查询，以确定您是否有缓慢的查询。超过一秒的查询通常是由于大型安装出现性能问题导致的，而迁移到外部数据库可能并不有所帮助。如果您的查询缓慢，请联系红帽支持团队。

3.10.2. 为外部数据库准备主机

使用最新的 Red Hat Enterprise Linux 8 安装一个全新的置备系统，以托管外部数据库。

Red Hat Enterprise Linux 的订阅不提供将 Satellite 与外部数据库的正确服务级别协议。您还必须将 Satellite 订阅附加到要用于外部数据库的基本操作系统。

先决条件

- 准备的主机必须满足 Satellite 的存储要求。

流程

1. 使用附加 Satellite 基础架构订阅 中的说明，将 Satellite 订阅附加到您的服务器。
2. 禁用所有软件仓库并只启用以下软件仓库：

```
# subscription-manager repos --disable '*'
# subscription-manager repos \
--enable=satellite-6.15-for-rhel-8-x86_64-rpms \
--enable=satellite-maintenance-6.15-for-rhel-8-x86_64-rpms \
--enable=rhel-8-for-x86_64-baseos-rpms \
--enable=rhel-8-for-x86_64-appstream-rpms
```

3. 启用以下模块：

```
# dnf module enable satellite:el8
```

注意

启用模块 satellite:el8 会警告与 postgresql:10 和 ruby:2.5 冲突，因为这些模块被设置为 Red Hat Enterprise Linux 8 上的默认模块版本。模块 satellite:el8 具有模块 postgresql:12 和 ruby:2.7 的依赖项，它将通过 satellite:el8 模块启用。这些警告不会导致安装过程失败，因此可以安全地忽略。有关 Red Hat Enterprise Linux 8 模块和生命周期流的更多信息，请参阅 Red Hat Enterprise Linux Application Streams 生命周期。

3.10.3. 安装 PostgreSQL

您只能在内部数据库安装过程中安装 satellite-installer 工具安装的相同版本的 PostgreSQL。Satellite 支持 PostgreSQL 版本 12。

流程

1. 要安装 PostgreSQL，请输入以下命令：

```
# dnf install postgresql-server postgresql-evr postgresql-contrib
```

2. 要初始化 PostgreSQL，请输入以下命令：

```
# postgresql-setup initdb
```

3. 编辑 /var/lib/pgsql/data/postgresql.conf 文件：

```
# vi /var/lib/pgsql/data/postgresql.conf
```

请注意，需要调整外部 PostgreSQL 的默认配置才能使用 Satellite。基础推荐的外部数据库配置调整如下：

- checkpoint_completion_target: 0.9
- max_connections: 500
- shared_buffers: 512MB
- work_mem: 4MB

4. 删除 # 并编辑以侦听入站连接：

```
listen_addresses = '*'
```

5. 编辑 /var/lib/pgsql/data/pg_hba.conf 文件：

```
# vi /var/lib/pgsql/data/pg_hba.conf
```

6. 在文件中添加以下行：

```
host  all  all  Satellite_ip/32  md5
```

7. 要启动并启用 PostgreSQL 服务，请输入以下命令：

```
# systemctl enable --now postgresql
```

8. **在外部 PostgreSQL 服务器上打开 postgresql 端口：**

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

9. **使更改持久：**

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

10. **切换到 postgres 用户并启动 PostgreSQL 客户端：**

```
$ su - postgres -c psql
```

11. **创建三个数据库和专用角色：一个用于 Satellite，一个用于 Candlepin，另一个用于 Pulp：**

```
CREATE USER "foreman" WITH PASSWORD 'Foreman_Password';
CREATE USER "candlepin" WITH PASSWORD 'Candlepin_Password';
CREATE USER "pulp" WITH PASSWORD 'Pulpcore_Password';
CREATE DATABASE foreman OWNER foreman;
CREATE DATABASE candlepin OWNER candlepin;
CREATE DATABASE pulpcore OWNER pulp;
```

12. **连接到 Pulp 数据库：**

```
postgres=# \c pulpcore
You are now connected to database "pulpcore" as user "postgres".
```

13. **创建 hstore 扩展：**

```
pulpcore=# CREATE EXTENSION IF NOT EXISTS "hstore";
CREATE EXTENSION
```

14. **退出 postgres 用户：**

```
# \q
```

15. 从 Satellite 服务器中，测试您可以访问数据库。如果连接成功，命令会返回 1。

```
# PGPASSWORD='Foreman_Password' psql -h postgres.example.com  -p 5432 -U foreman
-d foreman -c "SELECT 1 as ping"
# PGPASSWORD='Candlepin_Password' psql -h postgres.example.com -p 5432 -U
candlepin -d candlepin -c "SELECT 1 as ping"
# PGPASSWORD='Pulpcore_Password' psql -h postgres.example.com -p 5432 -U pulp -d
pulpcore -c "SELECT 1 as ping"
```

3.10.4. 将 Satellite 服务器配置为使用外部数据库

使用 satellite-installer 命令，将 Satellite 配置为连接到外部 PostgreSQL 数据库。

前提条件

- 您已在 Red Hat Enterprise Linux 服务器中安装并配置 PostgreSQL 数据库。

流程

1. 要为 Satellite 配置外部数据库，请输入以下命令：

```
# satellite-installer \
--foreman-db-database foreman \
--foreman-db-host postgres.example.com \
--foreman-db-manage false \
--foreman-db-password Foreman_Password \
--foreman-proxy-content-pulpcore-manage-postgresql false \
--foreman-proxy-content-pulpcore-postgresql-db-name pulpcore \
--foreman-proxy-content-pulpcore-postgresql-host postgres.example.com \
--foreman-proxy-content-pulpcore-postgresql-password Pulpcore_Password \
--foreman-proxy-content-pulpcore-postgresql-user pulp \
--katello-candlepin-db-host postgres.example.com \
--katello-candlepin-db-name candlepin \
--katello-candlepin-db-password Candlepin_Password \
--katello-candlepin-manage-db false
```

要为这些外部数据库启用安全套接字层(SSL)协议，请添加以下选项：

```
--foreman-db-root-cert <path_to_CA>
--foreman-db-sslmode verify-full
--foreman-proxy-content-pulpcore-postgresql-ssl true
--foreman-proxy-content-pulpcore-postgresql-ssl-root-ca <path_to_CA>
```

```
--katello-candlepin-db-ssl true
--katello-candlepin-db-ssl-ca <path_to_CA>
--katello-candlepin-db-ssl-verify true
```

第 4 章 使用外部服务配置 SATELLITE 服务器

如果您不想在 Satellite 服务器上配置 DNS、DHCP 和 TFTP 服务，请使用本节将 Satellite 服务器配置为处理外部 DNS、DHCP 和 TFTP 服务。

4.1. 使用外部 DNS 配置 SATELLITE 服务器

您可以使用外部 DNS 配置 Satellite 服务器。Satellite 服务器使用 nsupdate 实用程序更新远程服务器上的 DNS 记录。

要使任何更改持久，您必须使用适合您的环境的选项输入 satellite-installer 命令。

先决条件

- 您必须已配置了外部 DNS 服务器。
- 本指南假设您有现有的安装。

流程

1. 将 /etc/rndc.key 文件从外部 DNS 服务器复制到 Satellite 服务器：

```
# scp root@dns.example.com:/etc/rndc.key /etc/foreman-proxy/rndc.key
```

2. 配置所有权、权限和 SELinux 上下文：

```
# restorecon -v /etc/foreman-proxy/rndc.key
# chown -v root:foreman-proxy /etc/foreman-proxy/rndc.key
# chmod -v 640 /etc/foreman-proxy/rndc.key
```

3. 要测试 nsupdate 工具，请远程添加主机：

```
# echo -e "server DNS_IP_Address\n \
update add aaa.example.com 3600 IN A Host_IP_Address\n \
send\n" | nsupdate -k /etc/foreman-proxy/rndc.key
# nslookup aaa.example.com DNS_IP_Address
```

```
# echo -e "server DNS_IP_Address\n \
update delete aaa.example.com 3600 IN A Host_IP_Address\n \
send\n" | nsupdate -k /etc/foreman-proxy/rndc.key
```

4. **输入 satellite-installer 命令，对 /etc/foreman-proxy/settings.d/dns.yml 文件进行以下更改：**

```
# satellite-installer --foreman-proxy-dns=true \
--foreman-proxy-dns-managed=false \
--foreman-proxy-dns-provider=nsupdate \
--foreman-proxy-dns-server="DNS_IP_Address" \
--foreman-proxy-keyfile=/etc/foreman-proxy/rndc.key
```

5. **在 Satellite Web UI 中，进入到 Infrastructure > Capsules。**

6. **找到 Satellite Server，然后从 Actions 列中的列表中选择 Refresh。**

7. **将 DNS 服务与适当的子网和域关联。**

4.2. 使用外部 DHCP 配置 SATELLITE 服务器

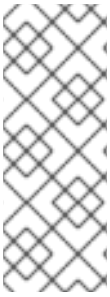
要使用外部 DHCP 配置 Satellite 服务器，您必须完成以下步骤：

1. **第 4.2.1 节 “配置外部 DHCP 服务器以用于 Satellite 服务器”**

2. **第 4.2.2 节 “使用外部 DHCP 服务器配置 Satellite 服务器”**

4.2.1. 配置外部 DHCP 服务器以用于 Satellite 服务器

要将运行 Red Hat Enterprise Linux 的外部 DHCP 服务器配置为与 Satellite 服务器搭配使用，您必须安装 ISC DHCP Service 和 Berkeley Internet Name Domain (BIND)工具软件包。您还必须与 Satellite 服务器共享 DHCP 配置和租用文件。此流程中的示例使用分布式网络文件系统(NFS)协议共享 DHCP 配置和租期文件。



注意

如果您使用 dnsmasq 作为外部 DHCP 服务器，请启用 dhcp-no-override 设置。这是必要的，因为 Satellite 在 TFTP 服务器上创建 grub2/ 子目录下的配置文件。如果禁用 dhcp-no-override 设置，主机会从根目录获取引导装载程序及其配置，这可能会导致错误。

流程

1. **在 Red Hat Enterprise Linux 主机上，安装 ISC DHCP 服务和 Berkeley Internet Name Domain (BIND)工具软件包：**

```
# dnf install dhcp-server bind-utils
```

2. **生成安全令牌：**

```
# tsig-keygen -a hmac-md5 omapi_key
```

3. **编辑所有子网的 dhcpd 配置文件并添加 tsig-keygen 生成的密钥。以下是一个示例：**

```
# cat /etc/dhcp/dhcpd.conf
default-lease-time 604800;
max-lease-time 2592000;
log-facility local7;

subnet 192.168.38.0 netmask 255.255.255.0 {
 range 192.168.38.10 192.168.38.100;
 option routers 192.168.38.1;
 option subnet-mask 255.255.255.0;
 option domain-search "virtual.lan";
 option domain-name "virtual.lan";
 option domain-name-servers 8.8.8.8;
}

omapi-port 7911;
key omapi_key {
 algorithm hmac-md5;
 secret "My_Secret";
};
omapi-key omapi_key;
```

请注意，选项 router 值是您要与外部 DHCP 服务一起使用的 Satellite 服务器或 Capsule 服务器的 IP 地址。

4. 在管理门户中，定义每个子网。不要为定义的子网设置 DHCP Capsule。

   要防止冲突，请单独设置租期和保留范围。例如，如果租期范围是 192.168.38.10 到 192.168.38.100，在 Satellite Web UI 中将保留范围定义为 192.168.38.101 to 192.168.38.250。

5. 配置防火墙以从外部访问 DHCP 服务器：

```
# firewall-cmd --add-service dhcp
```

6. 使更改持久：

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

7. 在 Satellite 服务器上，确定 foreman 用户的 UID 和 GID：

```
# id -u foreman
993
# id -g foreman
990
```

8. 在 DHCP 服务器上，创建 foreman 用户和组，其 ID 与上一步中确定的 ID 相同：

```
# groupadd -g 990 foreman
# useradd -u 993 -g 990 -s /sbin/nologin foreman
```

9. 要确保配置文件可以访问，请恢复读取和执行标记：

```
# chmod o+rx /etc/dhcp/
# chmod o+r /etc/dhcp/dhcpd.conf
# chattr +i /etc/dhcp/ /etc/dhcp/dhcpd.conf
```

10. 启用并启动 DHCP 服务：

```
# systemctl enable --now dhcpd
```

11. 使用 NFS 导出 DHCP 配置和租期文件：

```
# dnf install nfs-utils
# systemctl enable --now nfs-server
```

12. 为您要使用 NFS 导出的 DHCP 配置和租期文件创建目录：

```
# mkdir -p /exports/var/lib/dhcpd /exports/etc/dhcp
```

13. 要为创建的目录创建挂载点，请在 /etc/fstab 文件中添加以下行：

```
/var/lib/dhcpd /exports/var/lib/dhcpd none bind,auto 0 0
/etc/dhcp /exports/etc/dhcp none bind,auto 0 0
```

14. 在 /etc/fstab 中挂载文件系统：

```
# mount -a
```

15. 确保 /etc/exports 中存在以下行：

```
/exports 192.168.38.1(rw,async,no_root_squash,fsid=0,no_subtree_check)

/exports/etc/dhcp 192.168.38.1(ro,async,no_root_squash,no_subtree_check,nohide)

/exports/var/lib/dhcpd 192.168.38.1(ro,async,no_root_squash,no_subtree_check,nohide)
```

请注意，您输入的 IP 地址是您要与外部 DHCP 服务一起使用的 Satellite 或 Capsule IP 地址。

16. 重新载入 NFS 服务器：

```
# exportfs -rva
```

17. 为 DHCP omapi 端口 7911 配置防火墙：

```
# firewall-cmd --add-port=7911/tcp
```

18. 可选：配置防火墙以从外部访问 NFS。客户端使用 NFSv3 配置。

```
# firewall-cmd \
--add-service mountd \
--add-service nfs \
--add-service rpc-bind \
--zone public
```

19. **使更改持久：**

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

4.2.2. 使用外部 DHCP 服务器配置 Satellite 服务器

您可以使用外部 DHCP 服务器配置 Satellite 服务器。

先决条件

- **确保您已配置了外部 DHCP 服务器，并且您已与 Satellite 服务器共享 DHCP 配置和租用文件。更多信息请参阅 第 4.2.1 节 “配置外部 DHCP 服务器以用于 Satellite 服务器”。**

流程

1. **安装 nfs-utils 软件包：**

```
# satellite-maintain packages install nfs-utils
```

2. 为 **NFS 创建 DHCP 目录：**

```
# mkdir -p /mnt/nfs/etc/dhcp /mnt/nfs/var/lib/dhcpd
```

3. **更改文件所有者：**

```
# chown -R foreman-proxy /mnt/nfs
```

4. 验证与 **NFS 服务器和远程过程调用(RPC)通信路径的通信：**

```
# showmount -e DHCP_Server_FQDN
# rpcinfo -p DHCP_Server_FQDN
```

5. 在 /etc/fstab 文件中添加以下行：

```
DHCP_Server_FQDN:/exports/etc/dhcp /mnt/nfs/etc/dhcp nfs
ro,vers=3,auto,nosharecache,context="system_u:object_r:dhcp_etc_t:s0" 0 0

DHCP_Server_FQDN:/exports/var/lib/dhcpd /mnt/nfs/var/lib/dhcpd nfs
ro,vers=3,auto,nosharecache,context="system_u:object_r:dhcpd_state_t:s0" 0 0
```

6. 在 /etc/fstab 中挂载文件系统：

```
# mount -a
```

7. 要验证 foreman-proxy 用户可以访问通过网络共享的文件，请显示 DHCP 配置和租期文件：

```
# su foreman-proxy -s /bin/bash
$ cat /mnt/nfs/etc/dhcp/dhcpd.conf
$ cat /mnt/nfs/var/lib/dhcpd/dhcpd.leases
$ exit
```

8. 输入 satellite-installer 命令，对 /etc/foreman-proxy/settings.d/dhcp.yml 文件进行以下更改：

```
# satellite-installer \
--enable-foreman-proxy-plugin-dhcp-remote-isc \
--foreman-proxy-dhcp-provider=remote_isc \
--foreman-proxy-dhcp-server=My_DHCP_Server_FQDN \
--foreman-proxy-dhcp=true \
--foreman-proxy-plugin-dhcp-remote-isc-dhcp-config /mnt/nfs/etc/dhcp/dhcpd.conf \
--foreman-proxy-plugin-dhcp-remote-isc-dhcp-leases /mnt/nfs/var/lib/dhcpd/dhcpd.leases \
--foreman-proxy-plugin-dhcp-remote-isc-key-name=omapi_key \
--foreman-proxy-plugin-dhcp-remote-isc-key-secret=My_Secret \
--foreman-proxy-plugin-dhcp-remote-isc-omapi-port=7911
```

9. 将 DHCP 服务与适当的子网和域关联。

4.3. 使用外部 TFTP 配置 SATELLITE 服务器

您可以使用外部 TFTP 服务配置 Satellite 服务器。

流程

1. 为 **NFS** 创建 **TFTP** 目录：

```
# mkdir -p /mnt/nfs/var/lib/tftpboot
```

2. 在 **/etc/fstab** 文件中，添加以下行：

```
TFTP_Server_IP_Address:/exports/var/lib/tftpboot /mnt/nfs/var/lib/tftpboot nfs
rw,vers=3,auto,nosharecache,context="system_u:object_r:tftpdir_rw_t:s0" 0 0
```

3. 在 **/etc/fstab** 中挂载文件系统：

```
# mount -a
```

4. 输入 **satellite-installer** 命令，对 **/etc/foreman-proxy/settings.d/tftp.yml** 文件进行以下更改：

```
# satellite-installer \
--foreman-proxy-tftp-root /mnt/nfs/var/lib/tftpboot \
--foreman-proxy-tftp=true
```

5. 如果 **TFTP** 服务在与 **DHCP** 服务不同的服务器上运行，请使用 **TFTP** 服务运行的服务器的 **FQDN** 或 **IP** 地址更新 **tftp_servername** 设置：

```
# satellite-installer --foreman-proxy-tftp-servername=TFTP_Server_FQDN
```

6. 在 **Satellite Web UI** 中，进入到 **Infrastructure > Capsules**。

7. 找到 **Satellite Server**，然后从 **Actions** 列中的列表中选择 **Refresh**。

8. 将 **TFTP** 服务与适当的子网和域关联。

4.4. 使用外部 IDM DNS 配置 SATELLITE 服务器

当 **Satellite** 服务器为主机添加 **DNS** 记录时，它会首先确定哪个胶囊为该域提供 **DNS**。然后，它与配置为为您的部署提供 **DNS** 服务的 **Capsule** 通信并添加记录。主机不涉及此过程。因此，您必须在当前配

置为使用 IdM 服务器管理的域提供 DNS 服务的 Satellite 或 Capsule 上安装和配置 IdM 客户端。

Satellite 服务器可以配置为使用红帽身份管理(IdM)服务器来提供 DNS 服务。有关红帽身份管理的更多信息，请参阅 Linux 域身份、身份验证和策略指南。

要将 Satellite 服务器配置为使用 Red Hat Identity Management (IdM)服务器来提供 DNS 服务，请使用以下流程之一：

- 第 4.4.1 节 “使用 GSS-TSIG 身份验证配置动态 DNS 更新”
- 第 4.4.2 节 “使用 TSIG 身份验证配置动态 DNS 更新”

要恢复到内部 DNS 服务，请使用以下流程：

- 第 4.4.3 节 “恢复到内部 DNS 服务”

注意

您不需要使用 Satellite 服务器来管理 DNS。当您使用 Satellite 的域注册功能时，调配的主机会自动注册到 IdM 时，ipa-client-install 脚本会为客户端创建 DNS 记录。使用外部 IdM DNS 和域注册配置 Satellite 服务器是互斥的。有关配置域注册的更多信息，请参阅 在 *连接的网络环境中安装 Satellite 服务器中的置备主机的外部身份验证*。

4.4.1. 使用 GSS-TSIG 身份验证配置动态 DNS 更新

您可以将 IdM 服务器配置为对 RFC3645 中定义的 secret 密钥事务(GSS-TSIG)技术使用通用安全服务算法。要将 IdM 服务器配置为使用 GSS-TSIG 技术，您必须在 Satellite 服务器基本操作系统上安装 IdM 客户端。

先决条件

- 您必须确保 IdM 服务器已部署，并且基于主机的防火墙已正确配置。如需更多信息，请参阅 *安装身份管理指南* 中的 IdM 的端口要求。
- 您必须联系 IdM 服务器管理员，以确保在 IdM 服务器上获取具有在 IdM 服务器上创建区域

权限的 IdM 服务器上的帐户。

- 您应创建应答文件的备份。如果应答文件损坏，您可以使用备份将应答文件恢复到其原始状态。如需更多信息，请参阅配置 Satellite 服务器。

流程

要使用 GSS-TSIG 身份验证配置动态 DNS 更新，请完成以下步骤：

在 IdM 服务器中创建 Kerberos 主体

1. 为从 IdM 管理员获取的帐户获取 Kerberos 票据：

```
# kinit idm_user
```

2. 为 Satellite 服务器创建一个新的 Kerberos 主体，用于在 IdM 服务器上进行身份验证：

```
# ipa service-add capsule/satellite.example.com
```

安装和配置 idM 客户端

1. 在为部署管理 DNS 服务的 Satellite 或 Capsule 的基本操作系统中，安装 ipa-client 软件包：

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

2. 运行安装脚本并根据屏幕提示配置 IdM 客户端：

```
# ipa-client-install
```

3. 获取Kerberos ticket：

```
# kinit admin
```

4. 删除任何已存在的 keytab ：

```
# rm /etc/foreman-proxy/dns.keytab
```

5. 获取这个系统的 **keytab** ：

```
# ipa-getkeytab -p capsule/satellite.example.com@EXAMPLE.COM \
-s idm1.example.com -k /etc/foreman-proxy/dns.keytab
```

注意

将 **keytab** 添加到与服务中原始系统相同的备用系统时，添加 **r** 选项以防止生成新凭证并在原始系统上渲染凭证无效。

6. 对于 **dns.keytab** 文件，将 **group** 和 **owner** 设置为 **foreman-proxy** ：

```
# chown foreman-proxy:foreman-proxy /etc/foreman-proxy/dns.keytab
```

7. 可选： 要验证 **keytab** 文件是否有效，请输入以下命令：

```
# kinit -kt /etc/foreman-proxy/dns.keytab \
capsule/satellite.example.com@EXAMPLE.COM
```

在 IdM Web UI 中配置 DNS 区域

1. 创建并配置您要管理的区域：

   a. 导航到 **Network Services > DNS > DNS Zones**。

   b. 选择 **Add** 并输入区域名称。例如： **example.com**。

   c. 点 **Add and Edit**。

   d. 点 **Settings** 选项卡并在 **BIND** 更新策略 框中，将以下内容添加到分号分隔列表中：

```
grant capsule\047satellite.example.com@EXAMPLE.COM wildcard * ANY;
```

e. **将 Dynamic update 设置为 True。**

f. **启用 Allow PTR 同步。**

g. **点 Save 保存更改。**

2. **创建并配置反向区：**

a. **导航到 Network Services > DNS > DNS Zones。**

b. **点击 Add。**

c. **选择 Reverse zone IP 网络，并以 CIDR 格式添加网络地址以启用反向查找。**

d. **点 Add and Edit。**

e. **点 Settings 选项卡并在 BIND 更新策略 框中，将以下内容添加到分号分隔列表中：**

```
grant capsule\047satellite.example.com@EXAMPLE.COM wildcard * ANY;
```

f. **将 Dynamic update 设置为 True。**

g. **点 Save 保存更改。**

配置管理域的 DNS 服务的 Satellite 或 Capsule 服务器

1. **配置 Satellite 服务器或 Capsule 服务器以连接到您的 DNS 服务：**

```
# satellite-installer \
--foreman-proxy-dns-managed=false \
--foreman-proxy-dns-provider=nsupdate_gss \
--foreman-proxy-dns-server="idm1.example.com" \
```

```
--foreman-proxy-dns-tsig-keytab=/etc/foreman-proxy/dns.keytab \
--foreman-proxy-dns-tsig-principal="capsule/satellite.example.com@EXAMPLE.COM" \
--foreman-proxy-dns=true
```

2. 对于每个受影响的 Capsule，在 Satellite Web UI 中更新该 Capsule 的配置：

a. 在 Satellite Web UI 中，进入到 Infrastructure > Capsules，找到 Satellite 服务器，从 Actions 列中的列表中，选择 Refresh。

b. 配置域：

i. 在 Satellite Web UI 中，进入到 Infrastructure > Domains 并选择域名。

ii. 在 Domain 选项卡中，确保 DNS Capsule 设置为连接子网的胶囊。

c. 配置子网：

i. 在 Satellite Web UI 中，进入到 Infrastructure > Subnets 并选择子网名称。

ii. 在 Subnet 选项卡中，将 IPAM 设置为 None。

iii. 在 Domains 选项卡中，选择您要使用 IdM 服务器管理的域。

iv. 在 Capsules 选项卡中，确保 Reverse DNS Capsule 设置为连接子网的 Capsule。

v. 点 Submit 以保存更改。

4.4.2. 使用 TSIG 身份验证配置动态 DNS 更新

您可以将 IdM 服务器配置为对 DNS (TSIG)技术使用 rndc.key 密钥文件进行身份验证的 secret 密钥事务身份验证。TSIG 协议在 RFC2845 中定义。

先决条件

- 您必须确保 IdM 服务器已部署，并且基于主机的防火墙已正确配置。如需更多信息，请参阅 *Linux 域身份、身份验证和策略指南* 中的 端口要求。
- 您必须在 IdM 服务器上获取 root 用户访问权限。
- 您必须确认 Satellite 服务器或 Capsule 服务器是否已配置为为您的部署提供 DNS 服务。
- 您必须在为部署管理 DNS 服务的 Satellite 或 Capsule 的基本操作系统上配置 DNS、DHCP 和 TFTP 服务。
- 您必须创建应答文件的备份。如果应答文件损坏，您可以使用备份将应答文件恢复到其原始状态。如需更多信息，请参阅配置 Satellite 服务器。

流程

要使用 TSIG 身份验证配置动态 DNS 更新，请完成以下步骤：

在 IdM 服务器中启用对 DNS 区的外部更新

1. 在 IdM 服务器上，将以下内容添加到 /etc/named.conf 文件的顶部：

```
###############################################################################

include "/etc/rndc.key";
controls  {
inet _IdM_Server_IP_Address_ port 953 allow { _Satellite_IP_Address_; } keys { "rndc-key";
};
};
###############################################################################
```

2. 重新载入 named 服务以使更改生效：

```
# systemctl reload named
```

3. 在 IdM Web UI 中，进入到 Network Services & gt; DNS > DNS Zones 并点区的名称。在 Settings 选项卡中，应用以下更改：

a. **在 BIND 更新策略框中添加以下内容 ：**

```
grant "rndc-key" zonesub ANY;
```

b. **将 Dynamic update 设置为 True。**

c. **点 Update 保存更改。**

4. **将 /etc/rndc.key 文件从 IdM 服务器复制到 Satellite 服务器的基本操作系统。使用以下命令：**

```
# scp /etc/rndc.key root@satellite.example.com:/etc/rndc.key
```

5. **要为 rndc.key 文件设置正确的所有权、权限和 SELinux 上下文，请输入以下命令：**

```
# restorecon -v /etc/rndc.key
# chown -v root:named /etc/rndc.key
# chmod -v 640 /etc/rndc.key
```

6. **手动将 foreman-proxy 用户分配给 named 组。通常，satellite-installer 确保 foreman-proxy 用户属于 named UNIX 组，但是在这种情况下，Satellite 不管理用户和组，因此您需要手动将 foreman-proxy 用户分配给 named 组。**

```
# usermod -a -G named foreman-proxy
```

7. **在 Satellite 服务器上，输入以下 satellite-installer 命令，将 Satellite 配置为使用外部 DNS 服务器：**

```
# satellite-installer \
--foreman-proxy-dns-managed=false \
--foreman-proxy-dns-provider=nsupdate \
--foreman-proxy-dns-server="IdM_Server_IP_Address" \
--foreman-proxy-dns-ttl=86400 \
--foreman-proxy-dns=true \
--foreman-proxy-keyfile=/etc/rndc.key
```

测试 IdM 服务器中的 DNS 区的外部更新

1. 确保 Satellite 服务器上的 /etc/rndc.key 文件中的密钥与 IdM 服务器上使用的密钥相同：

```
key "rndc-key" {
        algorithm hmac-md5;
        secret "secret-key==";
};
```

2. 在受管主机上，为主机创建测试 DNS 条目。例如，主机 *test.example.com* 在 IdM 服务器上 A 记录为 192.168.25.20，地址为 192.168.25.1。

```
# echo -e "server 192.168.25.1\n \
update add test.example.com 3600 IN A 192.168.25.20\n \
send\n" | nsupdate -k /etc/rndc.key
```

3. 在 Satellite 服务器上，测试 DNS 条目：

```
# nslookup test.example.com 192.168.25.1
Server:  192.168.25.1
Address: 192.168.25.1#53

Name: test.example.com
Address: 192.168.25.20
```

4. 要在 IdM web UI 中查看条目，请进入 Network Services > DNS > DNS Zones。单击区域的名称，再按名称搜索主机。

5. 如果成功解析，请删除测试 DNS 条目：

```
# echo -e "server 192.168.25.1\n \
update delete test.example.com 3600 IN A 192.168.25.20\n \
send\n" | nsupdate -k /etc/rndc.key
```

6. 确认 DNS 条目已被删除：

```
# nslookup test.example.com 192.168.25.1
```

以上 nslookup 命令失败，如果记录被成功删除，则返回 SERVFAIL 错误消息。

4.4.3. 恢复到内部 DNS 服务

您可以恢复到使用 Satellite 服务器和 Capsule 服务器作为 DNS 提供程序。您可以使用配置外部 DNS 之前创建的应答文件备份，或者您可以创建应答文件的备份。有关回答文件的更多信息，请参阅配置 Satellite 服务器。

流程

在您要配置为管理域的 DNS 服务的 Satellite 或 Capsule 服务器上，完成以下步骤：

将 Satellite 或 Capsule 配置为 DNS 服务器

- 如果您在配置外部 DNS 前创建了回答文件备份，请恢复应答文件，然后输入 satellite-installer 命令：

```
# satellite-installer
```

- 如果您没有应答文件的合适的备份，请立即创建应答文件的备份。要在不使用应答文件的情况下将 Satellite 或 Capsule 配置为 DNS 服务器，请在 Satellite 或 Capsule 上输入以下 satellite-installer 命令：

```
# satellite-installer \
--foreman-proxy-dns-managed=true \
--foreman-proxy-dns-provider=nsupdate \
--foreman-proxy-dns-server="127.0.0.1" \
--foreman-proxy-dns=true
```

如需更多信息，请参阅 在胶囊服务器上配置 DNS、DHCP 和 TFTP。

运行 satellite-installer 命令并对 Capsule 配置进行任何更改后，您必须更新 Satellite Web UI 中每个受影响的胶囊的配置。

在 Satellite Web UI 中更新配置

1. 在 Satellite Web UI 中，进入到 Infrastructure > Capsules。
2. 对于您要更新的每个胶囊，从 Actions 列表中选择 Refresh。
3. 配置域：

a. 在 Satellite Web UI 中，进入到 Infrastructure > Domains，然后点击您要配置的域名。

b. 在 Domain 选项卡中，将 DNS Capsule 设置为连接子网的胶囊。

4. 配置子网：

a. 在 Satellite Web UI 中，进入到 Infrastructure > Subnets 并选择子网名称。

b. 在 Subnet 选项卡中，将 IPAM 设置为 DHCP 或 Internal DB。

c. 在 Domains 选项卡中，选择您要使用 Satellite 或 Capsule 管理的域。

d. 在 Capsules 选项卡中，将 Reverse DNS Capsule 设置为连接子网的胶囊。

e. 点 Submit 以保存更改。

附录 A. 将自定义配置应用到 RED HAT SATELLITE

当您使用 satellite-installer 首次安装和配置 Satellite 时，您可以指定 DNS 和 DHCP 配置文件不由 Puppet 使用安装程序标志 --foreman-proxy-dns-managed=false 和 --foreman-proxy-dhcp-managed=false 管理。如果在初始安装程序运行期间没有指定这些标志，则重新运行安装程序会覆盖所有手动更改，例如，用于升级目的。如果更改被覆盖，您必须运行恢复过程来恢复手动更改。如需更多信息，请参阅恢复由 Puppet 运行编写的手动更改。

要查看所有可用于自定义配置的安装程序标志，请运行 satellite-installer --scenario satellite --full-help。有些 Puppet 类不公开给卫星安装程序。要手动管理它们并防止安装程序覆盖其值，请通过向配置文件 /etc/foreman-installer/custom-hiera.yaml 添加条目来指定配置值。此配置文件采用 YAML 格式，每行包含一个条目，格式为 < puppet class>::<parameter name>: <value>。此文件中指定的配置值会在安装程序重新运行后保留。

常见示例包括：

- 对于 Apache，将 ServerTokens 指令设置为仅返回产品名称：

```
apache::server_tokens: Prod
```

- 完全关闭 Apache 服务器签名：

```
apache::server_signature: Off
```

Satellite 安装程序的 Puppet 模块存储在 /usr/share/foreman-installer/modules 下。检查 .pp 文件（例如： *moduleName*/manifests/*example*.pp）来查找类、参数和值。或者，使用 grep 命令进行关键字搜索。

设置一些值可能会对 Red Hat Satellite 的性能或功能造成意外的后果。在应用之前，请考虑更改的影响，然后首先测试非生产环境中的更改。如果您没有非生产环境的 Satellite 环境，请使用 --noop 和 --verbose 选项运行 Satellite 安装程序。如果更改造成问题，请从 custom-hiera.yaml 中删除关闭行，然后重新运行 Satellite 安装程序。如果您对特定值是否安全改变，请联络红帽支持。

附录 B. 恢复 PUPPET 运行覆盖的手动更改

如果您的手动配置已被 Puppet 运行覆盖，您可以将文件恢复到之前的状态。下例演示了如何恢复由 Puppet 运行覆盖的 DHCP 配置文件。

流程

1. 复制您要恢复的文件。这可让您比较文件来检查升级所需的任何强制更改。对于 DNS 或 DHCP 服务，这并不常见。

```
# cp /etc/dhcp/dhcpd.conf /etc/dhcp/dhcpd.backup
```

2. 检查日志文件，以记下覆盖文件的 md5sum。例如：

```
# journalctl -xe
...
/Stage[main]/Dhcp/File[/etc/dhcp/dhcpd.conf]: Filebucketed /etc/dhcp/dhcpd.conf to puppet
with sum 622d9820b8e764ab124367c68f5fa3a1
...
```

3. 恢复覆盖的文件：

```
# puppet filebucket restore --local --bucket \
/var/lib/puppet/clientbucket /etc/dhcp/dhcpd.conf \ 622d9820b8e764ab124367c68f5fa3a1
```

4. 比较备份文件和恢复的文件，并编辑恢复的文件，使其包含升级所需的任何强制更改。

附录 C. 恢复 SATELLITE 服务器以从 RED HAT CDN 下载内容

如果您的环境从断开连接变为已连接，您可以重新配置断开连接的 Satellite 服务器来直接从红帽 CDN 下载内容。

流程

1. 在 Satellite Web UI 中，进入到 Content > Subscriptions。
2. 单击 Manage Manifest。
3. 切换到 CDN 配置 选项卡。
4. 选择 Red Hat CDN。
5. 编辑 URL 字段以指向 Red Hat CDN URL：

   https://cdn.redhat.com

6. 点 Update。

现在，Satellite 服务器被配置为在下次同步存储库时从 Red Hat CDN 下载内容。

CLI 过程

1. 使用 SSH 登录 Satellite 服务器。
2. 使用 Hammer 重新配置 CDN：

```
# hammer organization configure-cdn --name="My_Organization" --type=redhat_cdn
```