Red Hat Satellite 6.15

安装 Capsule 服务器

安装和配置 Capsule

Last Updated: 2024-06-21

Red Hat Satellite 6.15 安装 Capsule 服务器

安装和配置 Capsule

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

法律通告

摘要

本指南介绍了如何安装 Red Hat Satellite Capsule 服务器、执行初始配置以及配置外部服务。

目录

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

我们感谢您对我们的文档提供的信息。请让我们了解如何改进文档。

您可以通过在 Bugzilla 中记录一个 ticket 来提交反馈：

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

第 1 章 为安装准备您的环境

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
- 胶囊服务器至少需要 12 GB RAM 才能正常工作。另外，还建议至少 4 GB RAM 交换空间。以 RAM 小于最小值运行的胶囊可能无法正确运行。
- 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 Capsule 服务器前，请确保您的环境满足安装要求。

警告

Capsule 的版本必须与安装的 Satellite 版本匹配。它不应该不同。例如，Capsule 版本 6.15 无法注册到 Satellite 版本 6.14。

胶囊服务器必须安装在全新调配的系统上，而该系统上不再提供其他功能，但运行 Capsule 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 Capsule 服务器创建的本地用户冲突：

- Apache
- foreman-proxy
- postgres
- Pulp

- puppet
- redis

有关扩展胶囊服务器的更多信息，请参阅 Capsule 服务器可扩展性注意事项。

认证的虚拟机监控程序

在物理系统和运行 Red Hat Enterprise Linux 的虚拟机监控程序上运行的虚拟机上完全支持 Capsule 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 Capsule。安装 Capsule 后您无法启用 FIPS 模式。如需更多信息，请参阅 *安全强化* 中的 安装启用了 FIPS 模式的 RHEL 8 系统 。

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。

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. Capsule 服务器安装的存储要求

目录	安装大小	运行时大小
/var/lib/pulp	1 MB	300 GB
/var/lib/pgsql	100 MB	20 GB
/usr	3 GB	不适用
/opt/puppetlabs	500 MB	不适用

Capsule 服务器上 PostgreSQL 数据库的大小可能会显著增加，同时增加从 Satellite 服务器同步的生命周期环境、内容视图或存储库的数量。在最大的 Satellite 环境中，Capsule 服务器上的 **/var/lib/pgsql** 的大小可以增加到 Satellite 服务器上的 **/var/lib/pgsql** 的双倍或三倍。

1.3. 存储指南

安装胶囊服务器以提高效率时应考虑以下准则：

- 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。
- 因为大多数 Capsule 服务器数据都存储在 **/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。

同步 RHEL ISO

如果您计划将 RHEL 内容 ISO 与 Satellite 同步，请注意，Red Hat Enterprise Linux 的所有次要版本也会同步。您必须计划在 Satellite 上有足够的存储来管理这一点。

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 Capsule Server 在最新版本的 Red Hat Enterprise Linux 8 上被支持，该版本在安装 Capsule 服务器时可用。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 Capsule 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 Capsule 服务器，然后再添加任何非红帽软件。

不要将 Capsule 服务器注册到 Red Hat Content Delivery Network (CDN)。

红帽不支持对运行 Capsule 服务器以外的任何系统使用系统。

1.5. 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。

如果在安装启动前没有打开 Satellite 服务器和 Capsule 服务器之间的端口，则胶囊服务器的安装会失败。

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。如需有关 Satellite 拓扑的更多信息，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. 胶囊传入流量

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）	

目的地端口	协议	服务	源	必需 For	描述
443, 80	TCP	HTTPS, HTTP	客户端	内容检索	内容
443, 80	TCP	HTTPS, HTTP	客户端	内容主机注册	胶囊 CA RPM 安装
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	TCP	HTTPS	客户端	内容主机注册	启动 上传事实 发送安装的软件包和追踪
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）
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	HTTP	客户端	PXE 引导	安装
8140	TCP	HTTPS	客户端	Puppet 代理	客户端更新（可选）
8443	TCP	HTTPS	客户端	内容主机注册	弃用且只需要在升级前部署的客户端主机
9090	TCP	HTTPS	Red Hat Satellite	Capsule API	胶囊功能
9090	TCP	HTTPS	客户端	注册端点	带有外部 Capsule 服务器的客户端注册
9090	TCP	HTTPS	客户端	OpenSCAP	配置客户端（如果安装了 OpenSCAP 插件）

目的地端口	协议	服务	源	必需 For	描述
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）

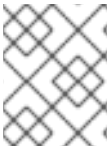
任何直接连接到 Satellite 服务器的主机都是此上下文中的客户端，因为它是集成胶囊的客户端。这包括在其上运行胶囊式服务器的基本操作系统。

DHCP Capsule 将执行 ICMP ping 和 TCP echo 连接，尝试使用 DHCP IPAM 设置的子网中的主机，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。

表 1.4. 胶囊传出流量

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
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）
68	UDP	DHCP	客户端	动态 IP	DHCP （可选）
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	红帽门户	SOS 报告	协助支持问题单（可选）
443	TCP	HTTPS	Satellite	内容	同步
443	TCP	HTTPS	Satellite	客户端通信	将请求从客户端转发到 Satellite

目的地端口	协议	服务	目的地	必需 For	描述
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
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）



注意

ICMP 到端口 7 UDP 和 TCP 不得被拒绝，但可以丢弃。DHCP Capsule 将 ECHO REQUEST 发送到客户端网络，以验证 IP 地址是否可用。响应可防止分配 IP 地址。

1.6. 启用从 CAPSULE 服务器到 SATELLITE 服务器的连接

在受管主机上，您必须启用从 Capsule 服务器到 Satellite 服务器的传入连接，并使此规则在重启后持久保留。

先决条件

- 确保 Satellite 服务器上的防火墙规则配置为启用客户端与 Satellite 通信的连接，因为胶囊服务器是 Satellite 服务器的客户端。如需更多信息，请参阅在 连接的网络环境中安装 Satellite 服务器中的 启用 *从客户端到 Satellite 服务器的连接*。

流程

1. 在 Satellite 服务器上，输入以下命令打开 Capsule 到 Satellite 通信的端口：

```
# firewall-cmd --add-port="5646/tcp"
```

2. 使更改持久：

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

1.7. 启用从 SATELLITE 服务器和客户端到 CAPSULE 服务器的连接

在您要安装 Capsule 的基本操作系统上，您必须启用从 Satellite 服务器和客户端到 Capsule 服务器的传入连接，并使这些规则在重启后保留。

流程

1. 在 Capsule 服务器上为客户端打开端口：

```
# firewall-cmd \
--add-port="5647/tcp" \
--add-port="8000/tcp" \
--add-port="9090/tcp"
```

2. 允许访问 Capsule 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 安全网络* 中的 使用和配置 firewalld。

第 2 章 安装 CAPSULE 服务器

在安装 Capsule 服务器前，您必须确保您的环境满足安装要求。如需更多信息，请参阅为安装准备您的环境。

2.1. 注册到 SATELLITE 服务器

使用这个流程将要在其上安装 Capsule 服务器的基本操作系统注册到 Satellite 服务器。

Red Hat 订阅清单先决条件

- 在管理门户中，必须安装清单，它必须包含您希望 Capsule 所属的组织的适当存储库。
- 清单必须包含您要在其上安装 Capsule 的基本操作系统以及您要连接到 Capsule 的任何客户端的存储库。
- 必须同步存储库。

如需有关清单和存储库的更多信息， *请参阅管理内容中的红帽订阅。*

代理和网络先决条件

- Satellite 服务器基础操作系统必须能够解析胶囊基本操作系统的主机名，反之亦然。
- 确保在胶囊服务器和 Satellite 服务器之间可能使用客户端证书身份验证进行 HTTPS 连接。不支持 Capsule 服务器和 Satellite 服务器之间的 HTTP 代理。
- 您必须相应地配置主机和基于网络的防火墙。如需更多信息，*请参阅安装 Capsule 服务器 中的 端口和防火墙要求*。您可以使用 Satellite Web UI、Hammer CLI 或 Satellite API 中的主机注册功能在 Satellite 中注册主机。如需更多信息，请参阅管理 主机 中的 *注册主机* 。

流程

1. 在 Satellite Web UI 中，进入到 **Hosts > Register Host**。
2. 从 **Activation Keys** 列表中，选择要分配给您的主机的激活密钥。
3. 点 **Generate** 来创建注册命令。
4. 点 *files* 图标将命令复制到您的剪贴板。
5. 使用 SSH 连接到您的主机，并运行注册命令。
6. 检查 **/etc/yum.repos.d/redhat.repo** 文件，并确保启用了适当的存储库。

CLI 过程

1. 使用 Hammer CLI 生成主机注册命令：

```
# hammer host-registration generate-command \
--activation-keys "My_Activation_Key"
```

如果您的主机不信任 Satellite 服务器的 SSL 证书，您可以通过在注册命令中添加 **--insecure** 标志来禁用 SSL 验证。

```
# hammer host-registration generate-command \
--activation-keys "My_Activation_Key" \
--insecure true
```

2. 使用 SSH 连接到您的主机，并运行注册命令。
3. 检查 **/etc/yum.repos.d/redhat.repo** 文件，并确保启用了适当的存储库。

API 过程

1. 使用 Satellite API 生成主机注册命令：

```
# curl -X POST https://satellite.example.com/api/registration_commands \
--user "My_User_Name" \
-H 'Content-Type: application/json' \
-d '{ "registration_command": { "activation_keys": ["My_Activation_Key_1,
My_Activation_Key_2"] }}'
```

如果您的主机不信任 Satellite 服务器的 SSL 证书，您可以通过在注册命令中添加 **--insecure** 标志来禁用 SSL 验证。

```
# curl -X POST https://satellite.example.com/api/registration_commands \
--user "My_User_Name" \
-H 'Content-Type: application/json' \
-d '{ "registration_command": { "activation_keys": ["My_Activation_Key_1,
My_Activation_Key_2"], "insecure": true }}'
```

使用激活码来简化指定环境。如需更多信息，请参阅管理内容中的管理 *激活码* 。

要将密码作为命令行参数输入密码，请使用 **username:password** 语法。请记住，这会在 shell 历史记录中保存密码。或者，您可以使用临时个人访问令牌而不是密码。要在 Satellite Web UI 中生成令牌，请导航到 **My Account** > **Personal Access Tokens**。

2. 使用 SSH 连接到您的主机，并运行注册命令。
3. 检查 **/etc/yum.repos.d/redhat.repo** 文件，并确保启用了适当的存储库。

2.2. 附加 SATELLITE 基础架构订阅

注意

如果您在 Satellite 上启用了 SCA，请跳过这一步。不需要使用 subscription-manager 将 Red Hat Satellite Infrastructure 订阅附加到 Capsule 服务器。有关 SCA 的更多信息，请参阅简单内容访问。

在注册了 Capsule 服务器后，您必须识别您的订阅池 ID 并附加可用订阅。Red Hat Satellite Infrastructure 订阅提供对 Red Hat Satellite 和 Red Hat Enterprise Linux 内容的访问。

Red Hat Satellite Infrastructure 包括在所有包括 Satellite 的订阅中，以前称为 智能管理。如需更多信息，请参阅 *红帽知识库* 中的 Satellite 基础架构订阅 MCT3718 MCT3719 。

如果订阅尚未附加到系统，则订阅被归类为可用。如果您无法找到可用的 Satellite 订阅，请查看红帽知识库解决方案 如何找出在 Red Hat Subscription Manager 下注册的客户端使用了哪个订阅？ 运行脚本以查看另一个系统是否消耗了您的订阅。

流程

1. 确定 Satellite 基础架构订阅的池 ID：

```
# subscription-manager list --all --available --matches 'Red Hat Satellite Infrastructure
Subscription'
```

该命令显示类似如下的输出：

```
Subscription Name:   Red Hat Satellite Infrastructure Subscription
Provides:            Red Hat Satellite
                     Red Hat Software Collections (for RHEL Server)
                     Red Hat CodeReady Linux Builder for x86_64
                     Red Hat Satellite Capsule
                     Red Hat Ansible Engine
                     Red Hat Satellite with Embedded Oracle
                     Red Hat Satellite 5 Managed DB
                     Red Hat Enterprise Linux Load Balancer (for RHEL Server)
                     Red Hat Beta
                     Red Hat Software Collections Beta (for RHEL Server)
                     Red Hat Enterprise Linux Server
                     Red Hat Enterprise Linux for x86_64
                     Red Hat Satellite Proxy
                     Red Hat Enterprise Linux High Availability for x86_64
                     Red Hat Discovery
SKU:                 MCT3718
Contract:
Pool ID:             8aca43dd771bf31101771c0231f906a5
Provides Management: Yes
Available:           10
Suggested:           1
Service Type:        L1-L3
Roles:
Service Level:       Premium
Usage:
Add-ons:
Subscription Type:   Standard
Starts:              11/11/2020
Ends:                11/11/2023
Entitlement Type:    Physical
```

2. 记录订阅池 ID。您的订阅池 ID 与提供的示例不同。

3. 将 Satellite Infrastructure 订阅附加到您的胶囊服务器在其上运行的基本操作系统。如果 Satellite 服务器上启用了 SCA，您可以跳过此步骤：

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

该命令显示类似如下的输出：

```
Successfully attached a subscription for: Red Hat Satellite Infrastructure Subscription
```

4. 可选：验证是否已附加 Satellite Infrastructure 订阅：

```
# subscription-manager list --consumed
```

2.3. 配置软件仓库

使用这个流程启用安装 Capsule 服务器所需的存储库。

1. 禁用所有软件仓库：

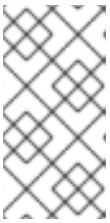
```
# subscription-manager repos --disable "*"
```

2. 启用以下软件仓库：

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

3. 启用模块：

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



注意

如果在启用 **satellite-capsule:el8** 模块时与 Ruby 或 PostgreSQL 冲突有任何警告，请参阅对 DNF 模块进行故障排除。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。

注意

如果您要将 Capsule 服务器作为托管在 Red Hat Virtualization 上的虚拟机，还必须启用 **Red Hat Common** 软件仓库，然后安装 Red Hat Virtualization 客户机代理和驱动程序。如需更多信息，请参阅 *虚拟机管理指南* 中的 在 Red Hat Enterprise Linux 上安装客户机代理和驱动程序。

4. 可选：验证是否启用了所需的存储库：

```
# dnf repolist enabled
```

2.4. 可选：在 CAPSULE 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。

2.4.1. 在 Capsule 服务器上安装 fapolicyd

您可以安装 **fapolicyd** 和 Capsule 服务器，也可以安装到现有的 Capsule 服务器上。如果您要安装 **fapolicyd** 以及新的 Capsule 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 Capsule 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. 安装 fapolicyd ：

```
# dnf install fapolicyd
```

2. 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。

2.5. 安装 CAPSULE 服务器软件包

在安装胶囊式服务器软件包前，您必须更新安装在基本操作系统上的所有软件包。

流程

要安装 Capsule 服务器，请完成以下步骤：

1. 更新所有软件包：

```
# dnf update
```

2. 安装 Satellite 服务器软件包：

```
# dnf install satellite-capsule
```

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)服务器安装胶囊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. 使用 SSL 证书配置 CAPSULE 服务器

Red Hat Satellite 使用 SSL 证书启用 Satellite 服务器、外部胶囊服务器和所有主机之间的加密通信。根据您的机构要求，您必须使用默认或自定义证书配置 Capsule 服务器。

- 如果使用默认 SSL 证书，还必须为每个外部 Capsule 服务器配置不同的默认 SSL 证书。更多信息请参阅 第 2.7.1 节 “使用默认 SSL 证书配置 Capsule 服务器”。
- 如果使用自定义 SSL 证书，还必须为每个外部 Capsule 服务器配置不同的自定义 SSL 证书。更多信息请参阅 第 2.7.2 节 “使用自定义 SSL 证书配置 Capsule 服务器”。

2.7.1. 使用默认 SSL 证书配置 Capsule 服务器

使用本节为胶囊服务器配置由 Satellite 服务器默认证书颁发机构(CA)签名的 SSL 证书。

先决条件

- 胶囊服务器已注册到 Satellite 服务器。如需更多信息，请参阅 注册到 Satellite 服务器。
- 安装了胶囊服务器软件包。如需更多信息，请参阅安装 Capsule 服务器软件包。

流程

1. 在 Satellite 服务器上，要为您的胶囊服务器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，如 **/root/capsule_cert** ：

```
# mkdir /root/capsule_cert
```

2. 在 Satellite 服务器上，为您的 Capsule 服务器生成 **/root/*capsule_cert*/ *Capsule.example.com*-certs.tar** 证书存档：

```
# capsule-certs-generate \
--foreman-proxy-fqdn capsule.example.com \
--certs-tar /root/capsule_cert/capsule.example.com-certs.tar
```

保留 Capsule- **certs-generate 命令返回的 satellite- installer** 命令的副本，以将证书部署到您的胶囊服务器。

Capsule -certs-generate 的输出示例

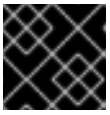
```
output omitted
satellite-installer --scenario capsule \
--certs-tar-file "/root/capsule_cert/capsule.example.com-certs.tar" \
```

```
--foreman-proxy-register-in-foreman "true" \
--foreman-proxy-foreman-base-url "https://satellite.example.com" \
--foreman-proxy-trusted-hosts "satellite.example.com" \
--foreman-proxy-trusted-hosts "capsule.example.com" \
--foreman-proxy-oauth-consumer-key "s97QxvUAgFNAQZNGg4F9zLq2biDsxM7f" \
--foreman-proxy-oauth-consumer-secret "6bpzAdMpRAfYaVZtaepYetomgBVQ6ehY"
```

3. 在 Satellite 服务器上，将证书归档文件复制到您的 Capsule 服务器中：

```
# scp /root/capsule_cert/capsule.example.com-certs.tar \
root@capsule.example.com:/root/capsule.example.com-certs.tar
```

4. 在 Capsule 服务器上，若要部署证书，请输入 Capsule- **certs-generate 命令返回的 satellite-installer** 命令。
当到 Satellite 的网络连接或端口尚未打开时，您可以将 **--foreman-proxy-register-in-foreman** 选项设置为 **false**，以防止 Capsule 尝试连接到 Satellite 并报告错误。当网络和防火墙正确配置时，再次运行安装程序，并将此选项设置为 **true**。



重要

部署证书后不要删除证书存档文件。例如，在升级胶囊服务器时是必需的。

2.7.2. 使用自定义 SSL 证书配置 Capsule 服务器

如果将 Satellite 服务器配置为使用自定义 SSL 证书，还必须将每个外部 Capsule 服务器配置为使用不同的自定义 SSL 证书。

要使用自定义证书配置您的 Capsule 服务器，在每个 Capsule 服务器上完成以下步骤：

1. 第 2.7.2.1 节 “为 Capsule 服务器创建自定义 SSL 证书”
2. 第 2.7.2.2 节 “将自定义 SSL 证书部署到 Capsule 服务器”
3. 第 2.7.2.3 节 “将自定义 SSL 证书部署到主机”

2.7.2.1. 为 Capsule 服务器创建自定义 SSL 证书

在受管主机上，为您的胶囊服务器创建一个自定义证书。如果您已经为 Capsule 服务器有自定义 SSL 证书，请跳过这个过程。

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/capsule_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。

   如果您已有此胶囊服务器的私钥，请跳过这一步。

```
# openssl genrsa -out /root/capsule_cert/capsule_cert_key.pem 4096
```

3. 为 CSR 创建 **/root/capsule_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 = capsule.example.com

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

[ alt_names ]
DNS.1 = capsule.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 = capsule.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/capsule_cert/capsule_cert_key.pem \ (1)
-config /root/capsule_cert/openssl.cnf \ (2)
-out /root/capsule_cert/capsule_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。

2.7.2.2. 将自定义 SSL 证书部署到 Capsule 服务器

使用这个流程，使用证书颁发机构签名的自定义 SSL 证书配置您的 Capsule 服务器。**satellite-installer** 命令返回 Capsule **-certs-generate** 命令对于每个胶囊服务器都是唯一的。不要在多个胶囊服务器上使用相同的命令。

先决条件

- Satellite 服务器配置了自定义证书。如需更多信息，请参阅 在 连接的网络环境中安装 Satellite 服务器 中的 配置带有自定义 SSL *证书的 Satellite 服务器*。
- 胶囊服务器已注册到 Satellite 服务器。如需更多信息，请参阅 注册到 Satellite 服务器。
- 安装了胶囊服务器软件包。如需更多信息，请参阅安装 Capsule 服务器软件包。

流程

1. 在 Satellite 服务器上生成证书捆绑包：

```
# capsule-certs-generate \
--foreman-proxy-fqdn capsule.example.com \
--certs-tar ~/capsule.example.com-certs.tar \
--server-cert /root/capsule_cert/capsule_cert.pem \ 1
--server-key /root/capsule_cert/capsule_cert_key.pem \ 2
--server-ca-cert "/root/capsule_cert/ca_cert_bundle.pem \ 3
--certs-update-server
```

1. 由证书颁发机构签名的 Capsule 服务器证书文件的路径。
2. 用于为胶囊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。

2. 保留 Capsule- **certs-generate 命令返回的 satellite- installer** 命令的副本，以将证书部署到您的胶囊服务器。

Capsule -certs-generate 的输出示例

```
output omitted
satellite-installer --scenario capsule \
--certs-tar-file "/root/capsule.example.com-certs.tar" \
--foreman-proxy-register-in-foreman "true" \
--foreman-proxy-foreman-base-url "https://satellite.example.com" \
--foreman-proxy-trusted-hosts "satellite.example.com" \
--foreman-proxy-trusted-hosts "capsule.example.com" \
--foreman-proxy-oauth-consumer-key "My_OAuth_Consumer_Key" \
--foreman-proxy-oauth-consumer-secret "My_OAuth_Consumer_Secret"
```

3. 在 Satellite 服务器上，将证书归档文件复制到您的 Capsule 服务器中：

```
# scp ~/capsule.example.com-certs.tar \
root@capsule.example.com:/root/capsule.example.com-certs.tar
```

4. 在 Capsule 服务器上，若要部署证书，请输入 Capsule- **certs-generate 命令返回的 satellite-installer** 命令。
如果 Satellite 的网络连接或端口尚未打开，您可以将 **--foreman-proxy-register-in-foreman** 选项设置为 **false**，以防止 Capsule 尝试连接到 Satellite 并报告错误。当网络和防火墙正确配置时，再次运行安装程序，并将此选项设置为 **true**。

重要

部署证书后不要删除证书存档文件。例如，在升级胶囊服务器时是必需的。

2.7.2.3. 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。

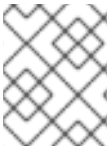
流程

- 更新每个主机上的 SSL 证书：

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

2.8. 在 SATELLITE WEB UI 中为胶囊服务器分配正确的机构和位置

安装 Capsule 服务器软件包后，如果有多个机构或位置，您必须将正确的机构和位置分配给 Capsule，以便在 Satellite Web UI 中可见 Capsule。



注意

将 Capsule 分配给与带有嵌入式 Capsule 的 Satellite 服务器相同的位置，可防止 Red Hat Insights 上传 Insights 清单。若要启用清单上传，请为这两个胶囊同步 SSH 密钥。

流程

1. 登录 Satellite Web UI。
2. 从屏幕左上角的 **组织** 列表中，选择 **Any Organization**。
3. 从屏幕左上角的 **Location** 列表中，选择 **Any Location**。
4. 在 Satellite Web UI 中，进入到 **Hosts** > **All Hosts** 并选择 Capsule Server。
5. 在 **Select Actions** 列表中，选择 **Assign Organization**。
6. 在 **Organization** 列表中，选择要分配此 Capsule 的组织。
7. 单击 **Mismatch 上的 Fix Organization**。
8. 点 **Submit**。
9. 选择胶囊式服务器。在 **Select Actions** 列表中，选择 **Assign Location**。
10. 从 **Location** 列表中，选择要分配此胶囊的位置。

11. 单击 **Mismatch 上的 Fix Location**。
12. 点 **Submit**。
13. 在 Satellite Web UI 中，进入到 **Administer** > **Organizations** 并点您分配胶囊的机构。
14. 单击 **Capsules** 选项卡，并确保 Capsule Server 在 **Selected items** 列表下列出，然后单击 **Submit**。
15. 在 Satellite Web UI 中，进入到 **Administer** > **Locations** 并点您分配 Capsule 的位置。
16. 单击 **Capsules** 选项卡，并确保 Capsule Server 在 **Selected items** 列表下列出，然后单击 **Submit**。

验证

另外，您还可以验证 Capsule 服务器是否在 Satellite Web UI 中正确列出。

1. 从 **Organization** 列表中选择机构。
2. 从 **Location** 列表中选择位置。
3. 在 Satellite Web UI 中，进入到 **Hosts** > **All Hosts**。
4. 在 Satellite Web UI 中，进入到 **Infrastructure** > **Capsules**。

第 3 章 在 CAPSULE 服务器上执行其他配置

使用本章在胶囊服务器上配置其他设置。

3.1. 配置 CAPSULE 以进行主机注册和置备

使用这个流程配置 Capsule，以便您可以使用 Capsule 服务器而不是 Satellite 服务器注册和置备主机。

流程

- 在命令行中，将胶囊添加到可信代理列表中。
  Satellite 需要此项来识别通过 Capsule 设定的 **X-Forwarded-For** HTTP 标头转发的主机 IP 地址。为安全起见，Satellite 默认仅识别来自 localhost 的这个 HTTP 标头。您可以输入可信代理作为有效的 IPv4 或 Capsules 的 IPv6 地址或网络范围。

  警告

  不要使用过大的网络范围，因为这可能会造成潜在的安全风险。

  输入以下命令。请注意，命令会覆盖当前存储在 Satellite 中的列表。因此，如果您之前设置了任何可信代理，则必须将它们包含在命令中：

  ```
  # satellite-installer \
  --foreman-trusted-proxies "127.0.0.1/8" \
  --foreman-trusted-proxies "::1" \
  --foreman-trusted-proxies "My_IP_address" \
  --foreman-trusted-proxies "My_IP_range"
  ```

  localhost 条目是必需的，不要省略它们。

验证

1. 使用 Satellite 安装程序的完整帮助列出当前可信代理：

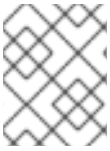
   ```
   # satellite-installer --full-help | grep -A 2 "trusted-proxies"
   ```

2. 当前列表包含您需要的所有可信代理。

3.2. 为拉取客户端配置远程执行

默认情况下，远程执行使用 SSH 作为 Script 供应商的传输机制。但是，远程执行还提供基于拉取的传输，如果您的基础架构禁止从 Capsule 到主机的传出连接，则可以使用它。

这由 Capsule 上的 **pull-mqtt** 模式组成，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。

模式为每个胶囊配置。有些胶囊可以配置为使用 **pull-mqtt** 模式，而其他胶囊则使用 SSH。如果出现这种情况，则给定主机上的一个远程作业可能会使用拉取客户端，同一主机上的下一个作业将使用 SSH。如果要避免这种情况，请将所有 Capsules 配置为使用相同的模式。

流程

1. 在每个相关 Capsule 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 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，主机向注册它们的胶囊订阅作业通知。因此，建议确保 Satellite 服务器向同一胶囊发送远程执行作业。要做到这一点，在 Satellite Web UI 中进入到 **Administer** > **Settings**。在 **Content** 选项卡上，将 **Prefer registered through Capsule for remote execution** 的值设为 **Yes** 。

5. 在 Capsule 上设置基于拉取的传输后，您还必须在每个主机上进行配置。如需更多信息，*请参阅管理主机* 中的 远程执行的传输模式。

3.3. 在 CAPSULE 服务器上启用 OPENSCAP

在 Satellite Server 和 Satellite 服务器的集成胶囊上，OpenSCAP 会被默认启用。要在外部胶囊上使用 OpenSCAP 插件和内容，您必须在每个胶囊上启用 OpenSCAP。

流程

- 要启用 OpenSCAP，请输入以下命令：

```
# satellite-installer --scenario capsule \
--enable-foreman-proxy-plugin-openscap \
--foreman-proxy-plugin-openscap-ansible-module true \
--foreman-proxy-plugin-openscap-puppet-module true
```

如果要使用 Puppet 部署合规策略，您必须首先启用它。有关更多信息，*请参阅使用 Puppet 集成管理配置*。

3.4. 在 CAPSULE 服务器中添加生命周期环境

如果您的胶囊服务器启用了内容功能，您必须添加一个环境，以便 Capsule 可以从 Satellite 服务器同步内容，并将内容提供给主机系统。

不要将 *库* 生命周期环境分配给您的胶囊服务器，因为它在每次 CDN 更新存储库时触发自动胶囊同步。这可能会消耗 Capsules 上的多个系统资源、Satellite 和 Capsule 之间的网络带宽，以及 Capsules 上的可用磁盘空间。

您可以在 Satellite 服务器或 Satellite Web UI 上使用 Hammer CLI。

流程

1. 在 Satellite Web UI 中，进入到 **Infrastructure** > **Capsules**，然后选择您要向其添加生命周期的 Capsule。
2. 单击 **Edit**，再单击 **Lifecycle Environments** 选项卡。
3. 在左侧菜单中选择您要添加到胶囊的生命周期环境，然后点 **Submit**。
4. 要同步胶囊中的内容，请单击 **Overview** 选项卡，再单击 **Synchronize**。
5. 选择 **Optimized Sync** 或 **Complete Sync**。
   有关每种同步类型的定义，请参阅恢复存储库。

CLI 过程

1. 要显示所有 Capsule 服务器的列表，请在 Satellite 服务器上输入以下命令：

```
# hammer capsule list
```

   记录下您要将生命周期添加到的胶囊 ID。

2. 使用 ID 验证 Capsule 的详情：

```
# hammer capsule info \
--id My_capsule_ID
```

3. 要查看 Capsule 服务器可用的生命周期环境，请输入以下命令并记录 ID 和机构名称：

```
# hammer capsule content available-lifecycle-environments \
--id My_capsule_ID
```

4. 将生命周期环境添加到 Capsule 服务器中：

```
# hammer capsule content add-lifecycle-environment \
--id My_capsule_ID \
--lifecycle-environment-id My_Lifecycle_Environment_ID
--organization "My_Organization"
```

   对您要添加到 Capsule 服务器的每个生命周期环境重复此操作。

5. 将 Satellite 的内容同步到 Capsule。
   - 要将 Satellite 服务器环境中的所有内容同步到 Capsule 服务器，请输入以下命令：

```
# hammer capsule content synchronize \
--id My_capsule_ID
```

- 要将 Satellite 服务器的特定生命周期环境同步到 Capsule 服务器，请输入以下命令：

```
# hammer capsule content synchronize \
--id My_capsule_ID
--lifecycle-environment-id My_Lifecycle_Environment_ID
```

- 在不检查元数据的情况下，将 Satellite 服务器中的所有内容同步到您的 Capsule 服务器：

```
# hammer capsule content synchronize \
--id My_capsule_ID \
--skip-metadata-check true
```

这等同于在 Satellite Web UI 中选择 **Complete Sync**。

3.5. 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 Capsule 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。胶囊服务器使用此 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 --scenario capsule \
--foreman-proxy-bmc "true" \
--foreman-proxy-bmc-default-provider "freeipmi"
```

3.6. 在 CAPSULE 服务器上配置 DNS、DHCP 和 TFTP

要在 Capsule 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

- 对于 DNS 服务器，您必须有正确的网络名称(**dns-interface**)。
- 对于 DHCP 服务器，您必须具有正确的接口名称(**dhcp-interface**)。
- 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

- 有关 **satellite-installer --scenario satellite** 命令的更多信息，请输入 **satellite-installer --scenario satellite --help**。
- 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 Capsule 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* 。

第 4 章 使用外部服务配置 CAPSULE 服务器

如果您不想在 Capsule 服务器上配置 DNS、DHCP 和 TFTP 服务，请使用本节将 Capsule 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 CAPSULE 服务器

您可以使用外部 DNS 配置 Capsule 服务器。胶囊服务器使用 **nsupdate** 工具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 Capsule 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 Capsule 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. 配置带有外部 DHCP 的 CAPSULE 服务器

要使用外部 DHCP 配置 Capsule 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 Capsule 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 Capsule 服务器

要将运行 Red Hat Enterprise Linux 的外部 DHCP 服务器配置为与 Capsule 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。您还必须与胶囊服务器共享 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：

```
# dnssec-keygen -a HMAC-MD5 -b 512 -n HOST omapi_key
```

因此，当前目录中创建由两个文件组成的密钥对。

3. 从密钥复制 secret 哈希：

```
# grep ^Key Komapi_key.+*.private | cut -d ' ' -f2
```

4. 编辑所有子网的 **dhcpd** 配置文件并添加密钥。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。

5. 从在其中创建文件的目录中删除这两个密钥文件。
6. 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。
7. 配置防火墙以从外部访问 DHCP 服务器：

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

8. 使更改持久：

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

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

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

10. 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
```

11. 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
```

12. 启用并启动 DHCP 服务：

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

13. 使用 NFS 导出 DHCP 配置和租期文件：

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

14. 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
```

15. 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
```

16. 在 **/etc/fstab** 中挂载文件系统：

```
# mount -a
```

17. 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。

18. 重新载入 NFS 服务器：

```
# exportfs -rva
```

19. 为 DHCP omapi 端口 7911 配置防火墙：

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

20. 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
```

21. 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 Capsule 服务器。

先决条件

- 确保您已配置了外部 DHCP 服务器，并且您已将 DHCP 配置和租用文件与胶囊服务器共享。更多信息请参阅 第 4.2.1 节 “配置外部 DHCP 服务器以用于 Capsule 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. 配置带有外部 TFTP 的 CAPSULE 服务器

您可以使用外部 TFTP 服务配置 Capsule 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 Capsule 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 CAPSULE 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。

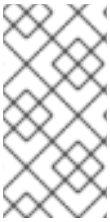
胶囊式服务器可以配置为使用红帽身份管理(IdM)服务器来提供 DNS 服务。有关红帽身份管理的更多信息，请参阅 Linux 域身份、身份验证和策略指南。

要将 Capsule 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 Capsule 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 和域注册配置胶囊服务器是互斥的。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 Capsule 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 Capsule 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.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. 使用 **satellite-installer** 命令配置管理域的 DNS 服务的 Satellite 或 Capsule：

- 在 Satellite 上输入以下命令：

```
# satellite-installer --scenario satellite \
--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
```

- 在 Capsule 上输入以下命令：

```
# satellite-installer --scenario capsule \
--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
```

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**，找到 Capsule Server，从 **Actions** 列中的列表中，选择 **Refresh**。

2. 配置域：

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

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

3. 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 **None**。

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

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

   e. 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 --scenario satellite \
--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。有关应答文件的更多信息，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。

第 5 章 使用 CAPSULE 管理 DHCP

Satellite 可以使用您的胶囊与 DHCP 服务集成。Capsule 具有多个 DHCP 提供程序，可用于将 Satellite 与现有 DHCP 基础架构集成或部署新的基础架构。您可以使用 Capsule 的 DHCP 模块来查询可用的 IP 地址，添加新 IP 地址并删除现有的保留。请注意，您的胶囊无法管理子网声明。

可用的 DHCP 提供商

- **dhcp_infoblox** - 如需更多信息，请参阅 *置备主机* 中的 使用 Infoblox 作为 DHCP 和 DNS 提供商。
- **dhcp_isc** - OMAPI 上的 ISC DHCP 服务器.如需更多信息，请参阅安装 Capsule 服务器 中的 在 Capsule 服务器上配置 DNS、DHCP 和 TFTP。
- ***dhcp_remote_isc*** *- 通过 OMAPI 进行 ISC DHCP 服务器，并通过网络挂载租期。如需更多信息，请参阅安装 Capsule 服务器 中的 配置外部 DHCP 服务器以与 Capsule 服务器一起使用。*

5.1. 保护 DHCPD API

*Capsule 使用 dhcpd API 与 DHCP 守护进程交互，以管理 DHCP。默认情况下，dhcpd API 侦听没有访问控制的任何主机。您可以添加 **omapi_key** 以提供基本安全性。*

流程

1. *在 Capsule 上，安装所需的软件包：*

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

2. *生成密钥：*

```
# dnssec-keygen -r /dev/urandom -a HMAC-MD5 -b 512 -n HOST omapi_key
# cat Komapi_key.+*.private | grep ^Key|cut -d ' ' -f2-
```

3. *使用 **satellite-installer** 来保护 dhcpd API：*

```
# satellite-installer \
--foreman-proxy-dhcp-key-name "My_Name" \
--foreman-proxy-dhcp-key-secret "My_Secret"
```

第 6 章 使用 CAPSULE 管理 DNS

Satellite 可以使用您的胶囊管理 DNS 记录。DNS 管理包含从现有 DNS 区域更新和删除 DNS 记录。Capsule 具有多个 DNS 提供程序，可用于将 Satellite 与现有 DNS 基础架构集成或部署新的基础架构。

启用 DNS 后，您的胶囊可以使用 **dns_nsupdate** 供应商处理符合 RFC 2136 的任何 DNS 服务器。其他提供商提供更多直接集成，如 Infoblox 的 **dns_infoblox**。

可用的 DNS 供应商

- **dns_infoblox** - 如需更多信息，请参阅 置备主机 中的 使用 Infoblox 作为 DHCP 和 DNS 提供商。
- **dns_nsupdate** - 使用 nsupdate 进行动态 DNS 更新。如需更多信息，请参阅 置备主机 中的 使用 Infoblox 作为 DHCP 和 DNS 提供程序。
- **dns_nsupdate_gss** - 使用 GSS-TSIG 进行动态 DNS 更新。如需更多信息，请参阅 第 4.4.1 节 “使用 GSS-TSIG 身份验证配置动态 DNS 更新”。

附录 A. 胶囊式服务器可扩展性注意事项

Satellite 服务器可以支持的最大胶囊服务器数量没有固定限制。它已被测试，Satellite 服务器可以支持具有 2 个 vCPU 的 17 个胶囊服务器。但是，可扩展性是高度变量，特别是在管理 Puppet 客户端时。

管理 Puppet 客户端时的胶囊服务器可扩展性取决于 CPU 数量、run-interval 分发和 Puppet 受管资源的数量。胶囊服务器具有 100 个并发 Puppet 代理限制，在任何时间点上运行。运行超过 100 个并发 Puppet 代理会导致 503 HTTP 错误。

例如，假设 Puppet 代理运行均匀分布在 run-interval 中的任何单一点上运行的 100 个并发 Puppet 代理，则具有 4 个 CPU 的 Capsule 服务器最多有 1250categories-categories1600 个 Puppet 客户端，其中有 10 个 Puppet 类中分配有 10 个 Puppet 类。根据所需的 Puppet 客户端数量，卫星安装可以横向扩展胶囊服务器的数量来支持它们。

如果要在管理 Puppet 客户端时扩展 Capsule 服务器，则进行以下假设：

- 没有外部 Puppet 客户端直接报告到卫星集成胶囊。
- 所有其他 Puppet 客户端直接报告外部胶囊。
- 所有 Puppet 代理都有一个平均分布式 run-interval。

注意

从甚至分发中分离会增加 Satellite 服务器过载的风险。应用 100 个并发请求的限值。

下表描述了推荐的 4 个 CPU 的可扩展性限制。

表 A.1. 使用 4 个 CPU 的 Puppet 可扩展性

每个主机 Puppet 管理的资源	run-Interval Distribution
1	3000 – 2500
10	2400 – 2000
20	1700 – 1400

下表描述了最小 2 个 CPU 的可扩展性限制。

表 A.2. 使用 2 个 CPU 的 Puppet 可扩展性

每个主机 Puppet 管理的资源	run-Interval Distribution
1	1700 – 1450
10	1500 – 1250
20	850 – 700

附录 B. DNF 模块故障排除

如果 DNF 模块无法启用，这可能代表启用了不正确的模块。在这种情况下，您必须手动解析依赖项，如下所示。列出启用的模块：

```
# dnf module list --enabled
```

B.1. RUBY

如果 Ruby 模块无法启用，这可能代表启用了不正确的模块。在这种情况下，您必须手动解析依赖项，如下所示：

列出启用的模块：

```
# dnf module list --enabled
```

如果启用了 Ruby 2.5 模块，请执行模块重置：

```
# dnf module reset ruby
```

B.2. POSTGRESQL

如果 PostgreSQL 模块无法启用，这可能意味着启用了不正确的模块。在这种情况下，您必须手动解析依赖项，如下所示：

列出启用的模块：

```
# dnf module list --enabled
```

如果启用了 PostgreSQL 10 模块，请执行模块重置：

```
# dnf module reset postgresql
```

如果数据库之前使用 PostgreSQL 10 创建，请执行升级：

1. 启用 DNF 模块：

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

2. 安装 PostgreSQL 升级软件包：

   ```
   # dnf install postgresql-upgrade
   ```

3. 执行升级：

   ```
   # postgresql-setup --upgrade
   ```