



Red Hat Update Infrastructure 4

配置和管理红帽更新基础架构

设置和重新创建红帽更新基础架构 4

设置和重新创建红帽更新基础架构 4

法律通告

Copyright © 2024 Red Hat, Inc.

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

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

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

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

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

摘要

本指南介绍了如何配置和管理红帽更新基础架构 4(RHUI 4)。

目录

使开源包含更多	4
第 1 章 关于红帽更新基础架构 4	5
1.1. 安装选项	5
1.2. RHUI 4 组件	7
1.3. 内容供应商类型	11
1.4. 组件通信	11
1.5. 更改管理员密码	12
1.6. 其他资源	13
第 2 章 管理存储库	14
2.1. 可用频道	14
2.2. 列出当前由 RHUI 4 管理的软件仓库	14
2.3. 显示仓库的详细信息	15
2.4. 生成存储库状态文件	16
2.5. 添加红帽内容存储库	18
2.6. 使用输入文件添加新红帽内容存储库	19
2.7. 创建新的自定义存储库（仅限RPM 内容）	20
2.8. 从 RHUI 4 中删除存储库	22
2.9. 将内容上传到自定义存储库（仅限RPM 内容）	23
2.10. 从远程网站上传内容（仅限RPM 内容）	24
2.11. 将软件包组元数据导入到自定义存储库	25
2.12. 从自定义存储库中删除内容（仅限自定义 RPM 内容）	25
2.13. 列出存储库中的软件包（仅限RPM 内容）	26
第 3 章 创建授权证书和客户端配置 RPM	28
3.1. 使用 RED HAT UPDATE INFRASTRUCTURE 管理工具创建客户端授权证书	28
3.2. 使用 CLI 创建客户端授权证书	29
3.3. 验证客户端授权证书是否与 FUTURE 加密策略兼容	29
3.4. 使用 CLI 更改客户端配置 RPM 中的存储库 ID 前缀	30
3.5. 使用 RED HAT UPDATE INFRASTRUCTURE 管理工具创建客户端配置 RPM	30
3.6. 使用 CLI 创建客户端配置 RPM	31
第 4 章 管理 RED HAT 权利证书	33
4.1. 红帽更新设备证书	33
4.2. 内容交付服务器证书	33
4.3. 客户端证书	33
第 5 章 检查同步状态和调度	36
5.1. 显示存储库同步概述	36
5.2. 显示正在运行的同步	36
5.3. 查看最后一次存储库同步的详情	37
5.4. 立即同步单个存储库	37
5.5. 取消活跃同步任务	38
5.6. 取消等待同步任务	39
5.7. 查看和更改存储库自动发布状态	40
5.8. 查看和发布存储库 workflow	41
5.9. 将存储库导出到文件系统	41
第 6 章 管理内容交付服务器	43
6.1. 列出由 RHUI 4 管理的所有已知 CDS 实例	43
6.2. 注册新的 CDS	43
6.3. 重新安装并重新应用配置到 CDS	45

6.4. 配置 CDS 以接受旧的 CA	46
6.5. 配置 CDS 以停止接受旧的 CA	46
6.6. 取消注册一个 CDS	47
第 7 章 管理 HAPROXY 负载均衡器实例	48
7.1. 列出由 RHUI 4 管理的所有已知 HAPROXY 负载均衡器实例	48
7.2. 注册新的 HAPROXY 负载均衡器	48
7.3. 重新安装并重新将配置重新应用到 HAPROXY 负载均衡器中	49
7.4. 取消注册 HAPROXY 负载均衡器	50
第 8 章 管理容器	52
8.1. 了解 RED HAT UPDATE INFRASTRUCTURE 中的容器	52
8.2. 在 RED HAT UPDATE INFRASTRUCTURE 中添加容器	52
8.3. 同步容器存储库	54
8.4. 生成容器客户端配置	54
8.5. 在客户端上安装容器配置 RPM	55
8.6. 在客户端上测试 PODMAN PULL 命令	55
第 9 章 配置文件、退出代码和日志文件	57
第 10 章 使用 RHUI 4 命令	60
10.1. 使用 RHUI 4 菜单选项	60
10.2. 使用 RHUI 4 CLI 选项	62
第 11 章 认证云和服务提供商认证 workflow	68
11.1. 其他资源	68
第 12 章 备份和恢复红帽更新基础架构	69
12.1. 备份红帽更新设备	69
12.2. 恢复红帽更新设备	70
12.3. 备份内容交付服务器	71
12.4. 恢复内容交付服务器	72
12.5. 备份 HAPROXY 服务器	72
12.6. 恢复 HAPROXY 服务器	72
第 13 章 解决 RHUI 4 中的常见问题	74

使开源包含更多

红帽致力于替换我们的代码、文档和 Web 属性中存在问题的语言。我们从这四个术语开始：master、slave、黑名单和白名单。由于此项工作十分艰巨，这些更改将在即将推出的几个发行版本中逐步实施。有关更多详情，请参阅[我们的首席技术官 Chris Wright 提供的消息](#)。

第 1 章 关于红帽更新基础架构 4

Red Hat Update Infrastructure 4 (Red Hat Update Infrastructure 4) 是一个高度可扩展的、高度冗余的框架，可让您管理存储库和内容。它还可让云供应商为 Red Hat Enterprise Linux (RHEL) 实例提供内容和更新。根据上游 Pulp 项目，RHUI 允许云提供商在本地镜像红帽托管的存储库内容，创建具有自身内容的自定义软件仓库，并通过负载均衡的内容将这些存储库提供给大型最终用户。

作为系统管理员，您可以通过安装和配置红帽更新设备 (RHUA)、内容交付服务器 (CDS)、存储库、共享存储和负载均衡，准备您的基础架构以参与红帽认证云和服务提供商计划。

配置 RHUI 由以下任务组成：

- 创建并同步红帽软件仓库
- 创建客户端授权证书和密钥配置 RPM
- 为 RHUI 服务器创建客户端配置集

经验丰富的 RHEL 系统管理员是目标读者。具有有限 RHEL 技能的系统管理员应考虑与红帽咨询合作以提供红帽认证云提供商架构服务。

使用以下主题配置、管理和更新 RHUI：

- RHUI 组件
- 内容供应商类型
- 用于管理组件的命令行接口 (CLI)
- 工具命令
- 证书管理
- 内容管理

1.1. 安装选项

下表介绍了各种红帽更新基础架构 4 组件。

表 1.1. Red Hat Update Infrastructure 组件和功能

组件	ancronym	功能	替代方案
Red Hat Update Appliance	RHUA	从红帽内容交付网络下载新软件包，并将新软件包复制到每个 CDS 节点	无
内容交付服务器	CDS	提供客户端连接的 yum 软件仓库用于更新的软件包	无
HAProxy	无	在 CDS 节点间提供负载均衡	现有负载均衡解决方案
共享存储	无	提供共享存储	现有存储解决方案

下表描述了如何执行安装任务。

表 1.2. Red Hat Update Infrastructure 安装任务

安装任务	执行
安装 RHEL 8	RHUA、CDS 和 HAProxy
订阅系统	RHUA、CDS 和 HAProxy
附加 RHUI 订阅	RHUA、CDS 和 HAProxy
应用更新	RHUA、CDS 和 HAProxy
安装 <code>rhui-installer</code>	RHUA
运行 <code>rhui-installer</code>	RHUA

1.1.1. 选项 1：完整安装

- A RHUA
- 使用共享存储的两个或多个 CDS 节点
- 一个或多个 HAProxy 负载均衡器

1.1.2. 选项 2：使用现有存储解决方案安装

- A RHUA
- 有两个或更多带有现有存储解决方案的 CDS 节点
- 一个或多个 HAProxy 负载均衡器

1.1.3. 选项 3：使用现有负载均衡器解决方案进行安装

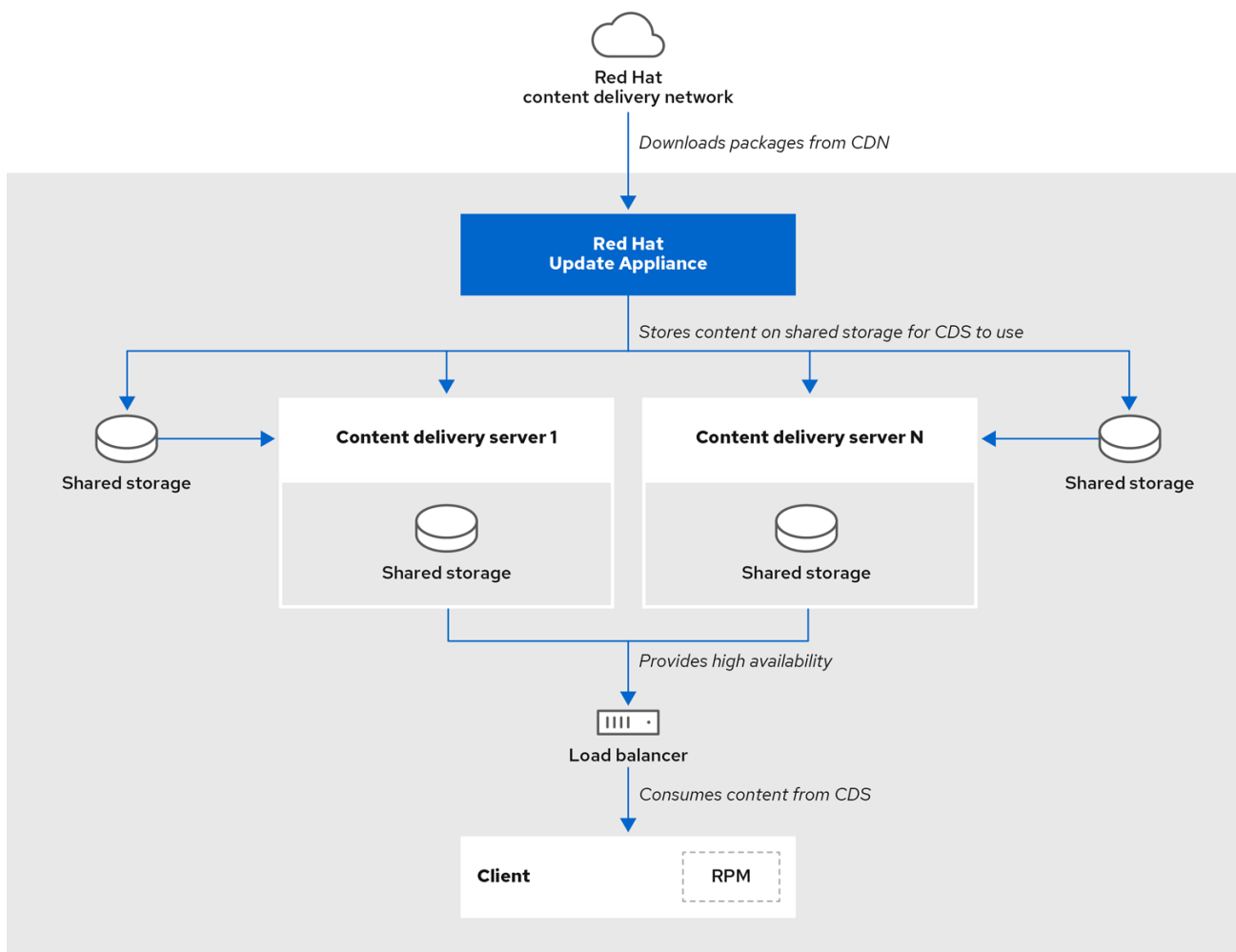
- A RHUA
- 使用共享存储的两个或多个 CDS 节点
- 现有负载均衡器

1.1.4. 选项 4：使用现有存储和负载均衡器解决方案安装

- A RHUA
- 两个或多个带有现有共享存储的 CDS 节点
- 现有负载均衡器

下图展示了红帽更新基础架构 4 组件如何交互的高级别视图。

图 1.1. Red Hat Update Infrastructure 4 概述



172_RHUI_1121



注意

您需要订阅 RHUA 作为 `--type rhui`，并具有 Red Hat Certified Cloud 和 Service Provider 订阅才能安装 RHUI。您还需要适当的内容证书。

在单独的 `x86_64` 服务器（裸机或虚拟机）上安装 RHUA 和 CDS 节点。确定连接到 RHUI 的所有服务器和网络均可访问 Red Hat Subscription Management 服务。

1.2. RHUI 4 组件

了解每个 RHUI 组件如何与其他组件交互，从而使您的作业成为系统管理员的过程比较简单。

1.2.1. Red Hat Update Appliance

每个 RHUI 安装有一个 RHUA，但在许多云环境中，每个地区或数据中心都会有一个 RHUI 安装，例如 Amazon 的 EC2 云包含多个区域。在每个地区，每个地区都有一个带有其自身 RHUA 节点的独立 RHUI。

RHUA 允许您执行以下任务：

- 从红帽内容交付网络(CDN)下载新软件包。RHUA 是连接到红帽的唯一 RHUI 组件，您可以为 RHUA 配置同步计划。

- 将新软件包复制到共享网络存储。
- 验证 RHUI 安装的健康状况，并将结果写入位于 RHUA 的文件。监控解决方案使用此文件来确定 RHUI 安装的健康状态。
- 通过 CLI 工具提供对 RHUI 安装健康状态的人类可读视图。

RHUI 使用两个主要配置文件：`/etc/rhui/rhui-tools.conf` 和 `/etc/rhui/rhui-subscription-sync.conf`。

`/etc/rhui/rhui-tools.conf` 配置文件包含 RHUA 使用的常规选项，如证书的默认文件位置，以及 Red Hat CDN 同步的默认配置参数。通常这个文件不需要编辑。

Red Hat Update Infrastructure Management Tool 根据用户输入的值生成 `/etc/rhui/rhui-subscription-sync.conf` 配置文件。它包含驱动在特定地区运行 RHUA 的所有信息。示例配置包括 RHUA 上用于下载软件包的目的地。

RHUA 使用几个服务来同步、组织和分发内容，以便轻松交付。

RHUA 服务

Pulp

监督支持服务管理的服务，为用户提供与交互的用户接口

PostgreSQL

用于跟踪当前同步的存储库、软件包和其他关键元数据的 PostgreSQL 数据库。

1.2.2. 内容交付服务器

CDS 节点提供客户端连接的仓库以获取更新的内容。可以有多个 CDS。因为 RHUI 提供了一个带有故障切换功能的负载平衡器，因此建议您使用多个 CDS 节点。

CDS 节点托管到最终用户 RHEL 系统的内容。虽然不需要的系统数量，CDS 会以轮循风格的负载均衡方式 (A、B、C、A、B、C) 将内容提供给最终用户系统。CDS 通过基于 httpd 的 **yum** 软件仓库使用 HTTP 托管内容到最终用户系统。

在配置期间，您可以指定同步软件包的 CDS 目录。与 RHUA 类似，唯一的要求是您挂载 CDS 上的目录。最多是云提供商，决定在分配所需设备时的最佳操作方法。Red Hat Update Infrastructure Management Tool 配置 RPM 将 `package` 目录与 NGINX 配置相关联，以便为它提供服务。

目前，RHUI 支持以下共享存储解决方案：

NFS

如果使用 NFS，**rhui-installer** 可以配置 RHUA 上的 NFS 共享，以存储内容以及 CDS 节点上的目录以挂载 NFS 共享。以下 **rhui-installer** 选项控制这些设置：

- **--remote-fs-mountpoint** 是应挂载远程文件系统共享的文件系统位置（默认为 `/var/lib/rhui/remote_share`）
- **--remote-fs-server** 是要使用的共享文件系统的远程挂载点，例如 `nfs.example.com:/path/to/share`（默认：`nfs.example.com:/export`）

CephFS

如果使用 CephFS，则必须单独配置 CephFS，然后将它用于 RHUI 作为挂载点。以下 **rhui-installer** 选项控制这些设置：

- **--remote-fs-server** 是要使用的共享文件系统的远程挂载点，例如

`ceph.example.com:/path/to/share` (默认: `ceph.example.com:/export`)



注意

本文档不提供设置或配置 Ceph 共享文件存储的说明。有关任何与 Ceph 相关的任务, 请查阅系统管理员或查看 Ceph 文档。

如果使用这些默认值, 则 RHUA 上的 `/export` 目录以及每个 CDS 上的 `/var/lib/rhui/remote_share` 目录相同。

预期的用法是您在 RHUA 和所有 CDS 节点上使用一个共享的网络文件系统, 例如 NFS。云供应商可能会使用某种形式的共享存储, 其中 RHUA 将软件包写入以及每个 CDS 读取内容。



注意

存储解决方案必须提供 NFS 或 CephFS 端点, 以便在 RHUA 和 CDS 节点上挂载。如果实施了本地存储, 则需要共享存储让集群正常工作。如果要向 RHUA 提供本地存储, 请将 RHUA 配置为 NFS 服务器, 并配置了 `rhua.example.com:/path/to/nfs/share` 端点。

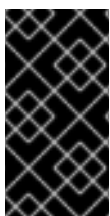
不要在任何 RHUI 节点上设置 Ceph 共享存储。您必须在独立专用机器上配置 CephFS。

每个 CDS 唯一采用的非标准逻辑是权利证书检查。此检查可确保客户在 `yum` 存储库上发出请求, 由云供应商访问这些存储库。检查可确保以下条件:

- 授权证书由云供应商的证书颁发机构(CA)证书签名。CA 证书安装在 CDS 中, 作为其配置的一部分, 以方便这个验证。
- 请求的 URI 与在客户端授权证书中找到的授权匹配。

如果 CA 验证失败, 客户端会看到 SSL 错误。如需更多信息, 请参阅 `/var/log/nginx/` 下的 CDS 节点的 NGINX 日志。

```
[root@cds01 ~]# ls -l /var/log/nginx/
access.log
error.log
unicorn-auth.log
unicorn-content_manager.log
unicorn-mirror.log
ssl-access.log----
```



重要

在 CDS 安装过程中, NGINX 配置通过 `/etc/nginx/conf.d/ssl.conf` 文件进行处理。

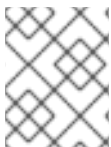
如果多个客户端遇到与库进行更新的问题, 这可能表示 RHUI 存在问题。如需更多详细信息, 请参阅 [Yum 生成 'Errno 14 HTTP Error 401: Authorization Required'](#)。

1.2.3. HAProxy 负载均衡器

如果使用多个 CDS, 则必须存在负载均衡解决方案, 才能在所有服务器间分布客户端 HTTPS 请求。RHUI 附带 HAProxy, 但最好选择在安装过程中要使用的负载均衡解决方案 (例如, 云提供商中的一个)。如果使用 HAProxy, 还必须决定要进行多少个节点。

客户端不配置为直接进入 CDS；其存储库文件配置为指向 HAProxy，RHUI 负载均衡器。HAProxy 是 TCP/HTTP 反向代理，特别是适用于高可用性环境。HAProxy 执行以下任务：

- 根据静态分配的 Cookie 路由 HTTP 请求
- 在通过 HTTP cookies 时将负载分散到多个服务器持久性中
- 交换机在主服务器失败时备份服务器
- 接受与专用于服务监控的特殊端口的连接
- 停止接受连接，而不破坏现有连接
- 添加、修改和删除两个方向中的 HTTP 标头
- 阻塞与特定模式匹配的请求
- 根据应用 Cookie 持久化客户端连接到正确的应用服务器
- 从应用程序截屏的 URI 中报告为经过身份验证的用户的 HTML 页面



注意

如果您使用现有的 load-balancer，请确保在 **cds-lb-hostname** 的负载均衡器中配置了端口 443，并且集群中所有 CDS 都位于负载均衡器的池中。

具体的配置取决于您所使用的特定负载均衡器软件。查看从典型的 HAProxy 设置中获取的以下配置，以了解您应该如何配置您的负载均衡器：

```
[root@rhui4proxy ~]# cat /etc/haproxy/haproxy.cfg
# This file managed by Puppet
global
  chroot /var/lib/haproxy
  daemon
  group haproxy
  log 10.10.153.149 local0
  maxconn 4000
  pidfile /run/haproxy.pid
  stats socket /var/lib/haproxy/stats
  user haproxy

defaults
  log global
  maxconn 8000
  option redispatch
  retries 3
  stats enable
  timeout http-request 10s
  timeout queue 1m
  timeout connect 10s
  timeout client 1m
  timeout server 1m
  timeout check 10s

listen https00
  bind 10.10.153.149:443
```

```
balance roundrobin
option tcplog
option tcp-check
server cds01.example.com cds01.example.com:443 check
server cds02.example.com cds02.example.com:443 check
```

请记住，当客户端无法连接时，务必要查看 `/var/log/nginx/` 下的 CDS 中的 nginx 日志，以确保任何请求到达 CDS。如果请求没有到达 CDS，则 DNS 或一般网络连接等问题可能处于故障。

1.2.4. 仓库和内容

存储库是软件包(RPMs)的存储位置。RHEL 使用 **yum** 命令来搜索存储库、下载、安装和配置 RPM。RPM 包含运行应用程序所需的所有依赖项。RPM 还会下载您的仓库中的软件更新。

RHEL 使用核心技术，如用于资源管理的控制组(cgroups)、用于进程隔离的命名空间以及用于安全性的 SELinux，启用安全多个租期，并减少了安全漏洞的可能性。这些技术可在提高安全性的同时，实现快速部署、更简单的测试、维护和故障排除。

内容与 RHUI 相关，就是您从红帽 CDN 下载的软件（如 RPM），以用于 RHUA 和 CDS 节点。RPM 提供运行特定应用程序和工具所需的文件。客户端是由 rpm 软件包提供的一组 SSL 内容证书和密钥授予访问权限，它还提供一组生成的 **yum** 存储库文件。

其他资源

- [哪些频道可通过红帽认证云和服务提供商\(CCSP\)合作伙伴提供？](#)

1.3. 内容供应商类型

云计算环境有三种类型：

- 公有云
- 私有云
- 混合云

本指南侧重于公共和私有云。我们假定读者理解了使用公共、私有和混合云的影响。

1.4. 组件通信

所有 RHUI 组件都通过端口 443 使用 HTTPS 通信协议。

表 1.3. Red Hat Update Infrastructure 通信协议

源	目的地	协议	用途
Red Hat Update Appliance	Red Hat Content Delivery Network	HTTPS	从红帽下载软件包
load-Balancer	内容交付服务器	HTTPS	转发客户端的 yum
客户端	load-Balancer	HTTPS	客户端上的 yum 用于从 CDS 下载软件包

源	目的地	协议	用途
内容交付服务器	Red Hat Update Appliance	HTTPS	可能会从 Pulp API 请求有关内容的信息

RHUI 节点需要以下网络访问权限相互通信。



注意

确保网络端口已打开，并且网络访问权限仅限于您要使用的那些节点。

表 1.4. Red Hat Update Infrastructure 网络访问权限

节点	端口	权限
RHUA	443	RHUA, CDS01, CDS02, ... CDSn
HAProxy	443	客户端虚拟机

1.5. 更改管理员密码

rhui-installer 设置初始 RHUI 登录密码。它还在 `/etc/rhui/rhui-subscription-sync.conf` 文件中编写。您可以使用 `--rhui-manager-password` 选项覆盖初始密码。

如果稍后要更改初始密码，您可以通过 **rhui-manager** 工具或通过 **rhui-installer** 更改它。运行 `rhui-installer --help` 命令来查看 **rhui-installer** 选项的完整列表。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **u** 键选择 **管理 RHUI 用户**。

3. 在 "用户管理器" 屏幕中，按 **p** 键选择 **更改管理员密码（通过注销跟踪）**：

```
-- User Manager --
```

```
p  change admin's password (followed by logout)
```

```
rhui (users) => p
```

```
Warning: After password change you will be logged out.
```

```
Use ctrl-c to cancel password change.
```

```
New Password:
```

4. 输入您的新密码；重新输入密码来确认更改。

```
New Password:
```



```
Re-enter Password:
```

```
[localhost] env PULP_SETTINGS=/etc/pulp/settings.py /usr/bin/pulpcore-manager reset-admin-password -p *****
```

验证

1. 更改 admin 密码后显示以下消息：

```
Password successfully updated. For security reasons you have been logged out.  
[root@ip-10-141-150-145 ~]#
```

1.6. 其他资源

- [Red Hat Cloud Access 参考指南](#)
- [Red Hat Enterprise Linux 8](#)
- [管理存储设备](#)
- [HAProxy](#)
- [Pulp 项目](#)

第 2 章 管理存储库

2.1. 可用频道

认证云和服务提供商(CCSP)合作伙伴控制通过服务提供哪些频道和包。如需了解针对不同操作系统版本可用的频道的最新信息，请参阅[通过红帽认证的云和服务提供商\(CCSP\)合作伙伴提供的 Red Hat Enterprise Linux 软件仓库](#)。

RHEL 8 可用的软件仓库如下：

- Red Hat Enterprise Linux 8 for x86_64 - AppStream from RHUI(RPMs)
- Red Hat Enterprise Linux 8 for x86_64 - BaseOS from RHUI(RPMs)
- Red Hat Enterprise Linux 8 for x86_64 - High Availability(RPMs)from RHUI
- Red Hat Enterprise Linux 8 for x86_64 - Resilient Storage(RPMs)from RHUI
- Red Hat Enterprise Linux 8 for x86_64 - Supplementary(RPMs)from RHUI
- Red Hat CodeReady Linux Builder for RHEL 8 x86_64(RPMs)从 RHUI

如果缺少所需的频道，请联系您的 CCSP。

其他资源

- [红帽生态系统目录](#)

2.2. 列出当前由 RHUI 4 管理的软件仓库

存储库是包含 Linux 发行版可下载软件的服务器节点。您可以使用 **yum** 搜索、安装和控制仓库中的 RPM 到 RHUA 和 CDS 节点。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **r** 键选择 **管理存储库**。
3. 在 **Repository Management** 屏幕中，按 **l** 键选择当前由 RHUI 管理的仓库：

```
...
Red Hat Enterprise Linux 8 for ARM 64 - AppStream (RPMs) from RHUI (8)
Red Hat Enterprise Linux 8 for ARM 64 - AppStream (RPMs) from RHUI (8.0)
Red Hat Enterprise Linux 8 for ARM 64 - AppStream (RPMs) from RHUI (8.1)
Red Hat Enterprise Linux 8 for ARM 64 - AppStream (RPMs) from RHUI (8.2)
Red Hat Enterprise Linux 8 for ARM 64 - AppStream (RPMs) from RHUI (8.3)
Red Hat Enterprise Linux 8 for ARM 64 - AppStream (RPMs) from RHUI (8.4)
Red Hat Enterprise Linux 8 for ARM 64 - AppStream (Source RPMs) from RHUI (8)
Red Hat Enterprise Linux 8 for ARM 64 - AppStream (Source RPMs) from RHUI (8.0)
Red Hat Enterprise Linux 8 for ARM 64 - AppStream (Source RPMs) from RHUI (8.1)
```

```

Red Hat Enterprise Linux 8 for ARM 64 - AppStream (Source RPMs) from RHUI (8.2)
Red Hat Enterprise Linux 8 for ARM 64 - AppStream (Source RPMs) from RHUI (8.3)
Red Hat Enterprise Linux 8 for ARM 64 - AppStream (Source RPMs) from RHUI (8.4)
Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (Debug RPMs) from RHUI (8)
Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (Debug RPMs) from RHUI (8.0)
Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (Debug RPMs) from RHUI (8.1)
Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (Debug RPMs) from RHUI (8.2)
Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (Debug RPMs) from RHUI (8.3)
Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (Debug RPMs) from RHUI (8.4)

```

...

2.3. 显示仓库的详细信息

您可以使用 **Repository Management** 屏幕显示特定存储库的信息。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **r** 键选择 **管理存储库**。
3. 在 **Repository Management** 屏幕中按 **i**：

```
Enter value (1-1631) to toggle selection, 'c' to confirm selections, or '?' for more commands:
```

4. 通过在存储库名称旁边输入值来选择存储库。在确认您的产品选择前，每次输入一个存储库选择。
5. 按 **c** 键确认：

```

Name:          Red Hat Enterprise Linux 8 for ARM 64 - AppStream (Debug RPMs) from
RHUI (8.4)
ID:            rhel-8-for-aarch64-appstream-debug-rhui-rpms-8.4
Type:         Red Hat
Version:      0
Relative Path: content/dist/rhel8/rhui/8.4/aarch64/appstream/debug
GPG Check:    Yes
Custom GPG Keys: (None)
Red Hat GPG Key: Yes
Content Unit Count:
Last Sync:    2021-11-15 15:56:06
Next Sync:    2021-11-15 22:00:00

```

```

Name:          Red Hat Enterprise Linux 8 for ARM 64 - AppStream (RPMs) from RHUI
(8.4)
ID:            rhel-8-for-aarch64-appstream-rhui-rpms-8.4
Type:         Red Hat
Version:      0
Relative Path: content/dist/rhel8/rhui/8.4/aarch64/appstream/os
GPG Check:    Yes
Custom GPG Keys: (None)

```

```

Red Hat GPG Key:  Yes
Content Unit Count:
Last Sync:      2021-11-15 19:50:20
Next Sync:     2021-11-16 01:55:00

Name:          Red Hat Enterprise Linux 8 for ARM 64 - AppStream (Source RPMs) from
RHUI (8.4)
ID:           rhel-8-for-aarch64-appstream-source-rhui-rpms-8.4
Type:         Red Hat
Version:      0
Relative Path: content/dist/rhel8/rhui/8.4/aarch64/appstream/source/SRPMS
GPG Check:    Yes
Custom GPG Keys:  (None)
Red Hat GPG Key:  Yes
Content Unit Count:
Last Sync:    2021-11-15 15:56:51
Next Sync:    2021-11-15 22:00:00

```

验证

1. 系统将显示类似输出结果供您选择。

2.4. 生成存储库状态文件

您可以生成机器可读 JSON 文件，该文件显示所有 RHUI 存储库的状态，并提供一些附加信息。例如，如果您希望被动监控存储库的状态，这非常有用。

2.4.1. 为 RHUI 软件仓库生成状态文件

您可以使用 **rhui-manager** 命令以机器可读格式获取每个存储库的状态。

流程

- 在 RHUA 节点上运行以下命令。

```
rhui-manager --non-interactive status --repo_json <output file>
```

生成 JSON 文件，其中包含所有自定义和红帽存储库的字典列表。有关可用字典的列表，请参阅 [第 2.4.2 节“存储库状态 JSON 文件中的字典键列表”](#)。

2.4.2. 存储库状态 JSON 文件中的字典键列表

运行命令来获取每个 RHUI 存储库的状态时，会创建一个机器可读的 JSON 文件。JSON 文件包含字典列表，其中包含每个存储库的字典。

自定义存储库的字典键列表

表 2.1. 自定义存储库的字典键列表

键	描述
base_path	存储库的路径。

键	描述
description	存储库的名称。
group	存储库所属的组。它始终设置为字符串 custom 。
id	存储库 ID。
name	存储库的名称。它与存储库 ID 相同。

红帽仓库的字典键列表

表 2.2. 红帽仓库的字典键列表

键	描述
base_path	存储库的路径。
description	存储库的名称。
group	存储库所属的组。它始终设置为字符串 redhat 。
id	存储库 ID。
last_sync_date	存储库最后一次同步的日期和时间。如果存储库没有同步，则该值为 null 。
last_sync_exception	如果存储库未能同步，则引发异常。如果存储库正确同步，则该值为 null 。
last_sync_result	同步任务的结果。 数值是： <ul style="list-style-type: none"> ● 完成：如果存储库正确同步。 ● null：如果仓库没有同步。 ● 失败：如果同步失败。 ● 运行：如果同步任务当前正在运行。
last_sync_traceback	如果存储库未能同步，则日志记录的回溯信息。如果存储库正确同步或未同步，则该值为 null 。
metadata_available	一个布尔值，用于指示元数据是否可用于存储库。
name	存储库的名称。它与存储库 ID 相同。

键	描述
next_sync_date	下一次调度的同步存储库的日期和时间。如果同步任务当前正在运行，则该值将运行。
repo_published	布尔值会取消此存储库是否在 RHUI 中发布。请注意，默认情况下，RHUI 配置为自动发布存储库。

2.5. 添加红帽内容存储库

根据上游 Pulp 项目，RHUI 允许云提供商在本地镜像红帽托管的存储库内容，创建具有自身内容的自定义软件仓库，并通过负载均衡的内容交付系统使这些存储库可供最终用户使用。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **r** 键选择 **管理存储库**。
3. 在 **Repository Management** 屏幕中，按 **a** 键选择**添加新的红帽内容存储库**。
4. 等待 Red Hat Update Infrastructure Management Tool 确定授权的存储库。这可能需要几分钟时间：

```
rhui (repo) => a
Loading latest entitled products from Red Hat...
... listings loaded
Determining undeployed products...
... product list calculated
```

5. Red Hat Update Infrastructure Management Tool 提示选择方法：

```
Import Repositories:
  1 - All in Certificate
  2 - By Product
  3 - By Repository
Enter value (1-3) or 'b' to abort:
```

6. 按 **2** 键选择"产品"方法。
7. 通过在提示符下键入存储库数量，选择要添加的存储库。您还可以通过输入 **1 - 5** 来选择存储库范围，例如：

```
Enter value (1-620) to toggle selection, 'c' to confirm selections, or '?' for more commands:
```

8. 继续操作，直到检查您要添加的所有软件仓库。
9. 选择完仓库后按 **c**。Red Hat Update Infrastructure Management Tool 显示部署的软件仓库并提示确认：

```
The following products will be deployed:
  Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (Debug RPMs) from RHUI
  Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (RPMs) from RHUI
Proceed? (y/n)
```

10. 按 **y** 键继续。一条消息表示每个成功部署：

```
Importing Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (Debug RPMs) from RHUI...
  Importing product repository Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (Debug
  RPMs) from RHUI (8.4)...
  Importing product repository Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (Debug
  RPMs) from RHUI (8.3)...
  Importing product repository Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (Debug
  RPMs) from RHUI (8.2)...
  Importing product repository Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (Debug
  RPMs) from RHUI (8.1)...
  Importing product repository Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (Debug
  RPMs) from RHUI (8.0)...
  Importing product repository Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (Debug
  RPMs) from RHUI (8)...

Importing Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (RPMs) from RHUI...
  Importing product repository Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (RPMs) from
  RHUI (8.4)...
  Importing product repository Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (RPMs) from
  RHUI (8.3)...
  Importing product repository Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (RPMs) from
  RHUI (8.2)...
  Importing product repository Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (RPMs) from
  RHUI (8.1)...
  Importing product repository Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (RPMs) from
  RHUI (8.0)...
  Importing product repository Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (RPMs) from
  RHUI (8)...

Content will not be downloaded to the newly imported repositories
until the next sync is run.
```

验证

1. 在 **Repository Management** 屏幕中，按 **I** 来检查是否已安装了正确的存储库。

其他资源

- [Red Hat Enterprise Linux 软件仓库可通过红帽认证的云和服务提供商\(CCSP\)合作伙伴通过 RHUI 交付。](#)

2.6. 使用输入文件添加新红帽内容存储库

在 Red Hat Update Infrastructure 4.2 及更新的版本中，您可以使用配置的 YAML 输入文件添加自定义软件仓库。您可以在 `/usr/share/rhui-tools/examples/repo_add_by_file.yaml` 目录中找到 RHUA 节点上的 YAML 文件示例。

这个功能只在命令行界面(CLI)中可用。

先决条件

- 确保具有对 RHUA 节点的 root 访问权限。

流程

1. 在 RHUA 节点上，用以下格式创建 YAML 输入文件：

```
# cat /my_home/example.yaml
---
name: Example_YAML_File
repo_ids:
  - rhel-8-for-x86_64-baseos-eus-rhui-rpms-8.1
  - rhel-8-for-x86_64-baseos-eus-rhui-rpms-8.2
  - rhel-8-for-x86_64-baseos-eus-rhui-rpms-8.4
  - rhel-8-for-x86_64-baseos-eus-rhui-rpms-8.6
---
```

2. 使用 `rhui-manager` 实用程序添加输入文件中列出的存储库：

```
# rhui-manager repo add_by_file --file /my_home/example.yaml --sync_now
The name of the repos being added: Example_YAML_File
Loading latest entitled products from Red Hat...
... listings loaded
Successfully added Red Hat Enterprise Linux 8 for x86_64 - BaseOS - Extended Update
Support from RHUI (RPMs) (8.1) (Yum)
Successfully added Red Hat Enterprise Linux 8 for x86_64 - BaseOS - Extended Update
Support from RHUI (RPMs) (8.2) (Yum)
Successfully added Red Hat Enterprise Linux 8 for x86_64 - BaseOS - Extended Update
Support from RHUI (RPMs) (8.4) (Yum)
Successfully added Red Hat Enterprise Linux 8 for x86_64 - BaseOS - Extended Update
Support from RHUI (RPMs) (8.6) (Yum)
... successfully scheduled for the next available timeslot.
... successfully scheduled for the next available timeslot.
... successfully scheduled for the next available timeslot.
... successfully scheduled for the next available timeslot.
```

验证

- 在 CLI 中，使用以下命令列出所有安装的软件仓库，并检查是否已安装了正确的软件仓库：

```
# rhui-manager repo list
```

- 在 RHUI Management Tool 中，在 Repository Management 屏幕上，按 **I** 来列出所有安装的存储库，并检查是否已安装了正确的存储库。

2.7. 创建新的自定义存储库（仅限RPM内容）

您可以创建用来将更新的客户端配置软件包或其他非红帽软件到 RHUI 客户端的自定义软件仓库。对于 64 位 RHUI 服务器（如 **client-rhui-x86_64**）的受保护的软件仓库是将新非红帽软件包（如更新的客户端配置软件包）分发到 RHUI 客户端的首选载体。

与 Red Hat 内容存储库一样，所有这些都受到保护，仅以处理器架构（**i386** 和 **AMD64**）不同的自定义软件仓库合并为授权证书中的单个权利（使用 **\$basearch yum** 变量）。

在出现证书问题时，RHUI 服务器的未受保护的存储库可用作将更新的 RPM 分发到 RHUI 客户端的回退方法。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **r** 键选择 **管理存储库**。
3. 在 **Repository Management** 屏幕中，按 **c** 键选择**创建新自定义存储库（仅限RPM 内容）**。
4. 输入存储库的唯一 ID。只允许字母数字字符、_（下划线）和 - (hyphen)。您不能在唯一 ID 中使用空格。例如，**repo1**、**repo_1** 和 **repo-1** 是有效的条目。

```
Unique ID for the custom repository (alphanumerics, _, and - only):
```

5. 输入存储库的显示名称。此名称用于识别 Red Hat Update Infrastructure 管理工具中的软件仓库。

```
Display name for the custom repository [repo_1]:
```

6. 指定托管存储库的路径。该路径必须在由 RHUI 托管的所有存储库中是唯一的。例如，如果您将路径指定为 **some/unique/name**，则存储库将位于 **//server/pulp/repos/some/unique/name**。

```
Unique path at which the repository will be served [repo_1]:
```

7. 选择 **sha256** 作为要用于存储库元数据的 checksum 类型。



注意

当您为 RHEL 6、RHEL 7 或 RHEL 8 创建自定义软件仓库时，请使用 **sha256**。如果您为 RHEL 5 客户端创建软件仓库，请使用 **sha1**。

```
Algorithm to use when calculating the checksum values for repository metadata:
```

```
1 - sha256 (default)
```

```
2 - sha1 (RHEL 5)
```

```
Enter value (1-2) or 'b' to abort:
```

8. 选择是否保护新存储库。如果不回答此问题，则任何客户端都可以访问该存储库。如果您回答 **yes**，则只有具有适当授权证书的客户端才能访问存储库。



警告

顾名思义，未受保护的存储库中的内容可供请求它的任何系统使用，而无需考虑客户端授权证书。在使用未受保护的存储库分发任何内容时，请小心，特别是更新的客户端配置 RPM 等内容，然后提供对受保护的存储库的访问。

9. 在出现以下情况时，回答"是"或没有以下问题：

Should the repository require clients to perform a GPG check and verify packages are signed by a GPG key? (y/n)

Will the repository be used to host any Red Hat GPG signed content? (y/n)

Will the repository be used to host any custom GPG signed content? (y/n)

Enter the absolute path to the public key of the GPG key pair:

Would you like to enter another public key? (y/n)

Enter the absolute path to the public key of the GPG key pair:

Would you like to enter another public key? (y/n)

10. 此时会显示新存储库的详情。在提示符处按 **y** 确认信息并创建存储库。

验证

1. 在 **Repository Management** 屏幕中，按 **I** 来检查是否已安装了正确的存储库。

其他资源

- [Red Hat Enterprise Linux 软件仓库可通过红帽认证的云和服务提供商\(CCSP\)合作伙伴通过 RHUI 交付。](#)

2.8. 从 RHUI 4 中删除存储库

当 Red Hat Update Infrastructure Management Tool 删除 Red Hat 软件仓库时，它会将软件仓库从 RHUA 以及所有适用的 CDS 节点中删除。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **r** 键选择 **管理存储库**。
3. 在 **Repository Management** 屏幕中，在提示符处按 **d** 键来删除红帽存储库。此时会显示当前由 RHUI 管理的所有仓库的列表。

4. 通过在提示符下键入存储库数量，选择要删除的存储库。键入存储库的数量将在该存储库的名称旁边放置复选标记。您还可以通过输入 **1 - 5** 来选择存储库范围，例如：
5. 继续操作，直到检查您要删除的所有软件仓库。
6. 在提示符下按 **c** 进行确认。



注意

删除存储库后，**yum** 无法使用引用已删除存储库的客户端配置 RPM。

2.9. 将内容上传到自定义存储库（仅限RPM 内容）

您可以上传多个软件包并一次上传到多个仓库。软件包会立即上传到 RHUA，但在下一次 CDS 节点同步为止。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **r** 键选择 **管理存储库**。
3. 在 **Repository Management** 屏幕中按 **u**:

```
Select the repositories to upload the package into:
- 1: test
```

4. 输入值(1-1)来切换选择。
5. 按 **c** 键确认您的选择。
6. 输入要上传的软件包位置。如果位置是 RPM，则将上传该文件。如果位置是一个目录，则目录中的所有 RPM 都将被上传：

```
/home/localuser/bear-4.1-1.noarch.rpm
```

```
The following RPMs will be uploaded:
bear-4.1-1.noarch.rpm
```

7. 按 **y** 来继续或 **n** 取消：

```
Uploaded /home/ec2-user/bear-4.1-1.noarch.rpm
```

验证

1. 如果内容上传成功，则会显示类似的信息：

```
Uploaded /home/ec2-user/bear-4.1-1.noarch.rpm
Total of 1 packages uploaded.
1 RPM(s) associated to repo test.
```

2.10. 从远程网站上传内容（仅限RPM 内容）

您可以上传存储在远程服务器中的软件包，而无需首先手动下载它们。软件包必须可以被 HTTP、HTTPS 或者 FTP 访问。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **r** 键选择 **管理存储库**。
3. 在 **Repository Management** 屏幕中按 **ur**:

```
Select the repositories to upload the package into:
- 1: test
```

4. 输入值(1-1)来切换选择。
5. 按 **c** 键确认您的选择：

```
### WARNING ### WARNING ### WARNING ### WARNING ### WARNING ###
WARNING ###
#                                     #
# Content retrieved from non-Red Hat arbitrary places can contain #
# unsupported or malicious software. Proceed at your own risk. #
#                                     #
#####
```

6. 输入要上传的软件包的远程 URL。如果位置是 RPM，则将上传该文件。如果位置是一个网页，则所有链接到该页面的 RPM 都将被上传：

```
https://www.google.com/url?
q=https://repos.fedorapeople.org/pulp/pulp/demo_repos/zoo/bear-4.1-
1.noarch.rpm&sa=D&source=docs&ust=1637381734614000&usg=AOvVaw2LGPli3y-
CXIVAQJtbtKRw
Retrieving https://www.google.com/url?
q=https://repos.fedorapeople.org/pulp/pulp/demo_repos/zoo/bear-4.1-
1.noarch.rpm&sa=D&source=docs&ust=1637381734614000&usg=AOvVaw2LGPli3y-
CXIVAQJtbtKRw
Found 1 RPMs to download
Retrieving https://repos.fedorapeople.org/pulp/pulp/demo_repos/zoo/bear-4.1-1.noarch.rpm

The following RPMs will be uploaded:
bear-4.1-1.noarch.rpm
```

7. 按 **y** 来继续或 **n** 取消：

```
Uploaded /tmp/rhui.318mfp81.tmp/bear-4.1-1.noarch.rpm
```

验证

1. 如果内容上传成功，则会显示类似的信息：

```
Uploaded /tmp/rhui.318mfp81.tmp/bear-4.1-1.noarch.rpm
Total of 1 packages uploaded.
1 RPM(s) associated to repo test.
```

2.11. 将软件包组元数据导入到自定义存储库

要允许 RHUI 用户查看和安装软件包组或语言包，您可以将 **comps.xml** 或 **comps.xml.gz** 文件导入到自定义存储库。

这个功能只在命令行界面中可用。

先决条件

- 确定您有有效的 **comps.xml** 或 **comps.xml.gz** 文件。
- 确保具有对 RHUA 节点的 root 访问权限。

流程

- 在 RHUA 节点上，使用 **rhui-manager** 实用程序将数据从 **comps** 文件导入到自定义存储库：

```
# rhui-manager repo add_comps --repo_id Example_Custom_Repo --comps
/home/Example-Comps.xml
```

验证

- 在使用自定义存储库的客户端系统中：
 - a. 刷新存储库数据：

```
# yum clean metadata
```

- b. 列出存储库数据，并验证 **comps** 文件是否已更新：

```
# yum grouplist
```

2.12. 从自定义存储库中删除内容（仅限自定义 RPM 内容）

您可以使用 RHUI 的文本用户界面(TUI)从自定义存储库中删除软件包。

有关命令行界面(CLI)命令，请参阅 [第 10.2 节“使用 RHUI 4 CLI 选项”](#)。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 输入 **r** 以选择 **管理存储库**。

3. 在 **Repository Management** 屏幕中，输入 **r** 以选择要从存储库中删除的软件包（仅限自定义 RPM 内容）：

```

-= Repository Management =-

l list repositories currently managed by the RHUI
i display detailed information on a repository
a add a new Red Hat content repository
ac add a new Red Hat container
c create a new custom repository (RPM content only)
d delete a repository from the RHUI
u upload content to a custom repository (RPM content only)
ur upload content from a remote web site (RPM content only)
p list packages in a repository (RPM content only)
r select packages to remove from a repository (Custom RPM content only)

```

4. 输入值来选择存储库：

```

Choose a repository to delete packages from:
 1 - Test-RPM-1
 2 - Test-RPM-2

```

5. 输入值来选择要删除的软件包。

```

Select the packages to remove:
- 1: example-package-1.noarch.rpm
- 2: example-package-2.noarch.rpm

```

6. 输入 **c** 以确认选择。

```

The following packages will be removed:
example-package-1.noarch.rpm

```

7. 输入 **y** 来继续或 **n** 取消：

```

Removed example-package-1.noarch.rpm

```

2.13. 列出存储库中的软件包（仅限RPM 内容）

当列出 Red Hat Update Infrastructure Management Tool 中的软件仓库时，只有少于 100 个软件包的软件仓库会显示其内容。超过 100 个软件包的结果仅显示软件包数。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **r** 键选择 **管理存储库**。
3. 在 **Repository Management** 屏幕中按 **p**。

4. 选择您要查看的存储库数量。Red Hat Update Infrastructure Management Tool 会询问您是否要过滤结果。将行留空以查看没有过滤器的结果。

```
Enter value (1-1631) or 'b' to abort: 1
```

```
Enter the first few characters (case insensitive) of an RPM to filter the results  
(blank line for no filter):
```

```
Only filtered results that contain less than 100 packages will have their  
contents displayed. Results with more than 100 packages will display  
a package count only.
```

```
Packages:
```

```
  bear-4.1-1.noarch.rpm
```

验证

1. 显示三种类型的信息之一：

```
Packages:
```

```
  bear-4.1-1.noarch.rpm
```

```
Package Count: 8001
```

```
No packages in the repository.
```

第 3 章 创建授权证书和客户端配置 RPM

RHUI 使用授权证书来确保在存储库上发出请求的客户端由云供应商访问这些存储库。授权证书必须由云供应商的证书颁发机构(CA)证书签名。CA 证书安装在 CDS 中，作为其配置的一部分。

3.1. 使用 RED HAT UPDATE INFRASTRUCTURE 管理工具创建客户端授权证书

当红帽发布原始授权证书时，它会授予您所请求的存储库的访问权限。当您创建客户端授权证书时，您决定如何分配客户端并为每一证书创建一个单独的证书。然后，每个证书都可用于创建单个 RPM。

先决条件

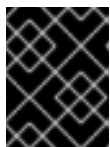
- 授权证书必须由云供应商的 CA 证书签名。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **e** 键选择 **创建授权证书和客户端配置 RPM**。
3. 按 **e** 键选择 **生成授权证书**。
4. 通过在提示符下键入存储库数量，选择在授权证书中包含哪些软件仓库。键入存储库的数量将 x 放置到该存储库的名称旁边。继续操作，直到检查您要添加的所有软件仓库。



重要

在一个权利中只包括单个 RHEL 版本的软件仓库。为多个 RHEL 版本添加软件仓库会导致 **yum** 配置文件不可用。

5. 在提示符下按 **c** 进行确认。
6. 输入证书的名称。此名称有助于识别 Red Hat Update Infrastructure Management Tool 中的证书，并生成证书和密钥文件的名称。

Name of the certificate. This will be used as the name of the certificate file (name.crt) and its associated private key (name.key). Choose something that will help identify the products contained with it.

7. 输入保存证书的路径。将字段留空以将其保存到当前工作目录中。
8. 输入证书应有效的天数。将 365 天字段留空。证书显示中要包含的软件仓库的详情。

Repositories to be included in the entitlement certificate:

```
Red Hat Repositories
Red Hat Enterprise Linux 8 for ARM 64 - AppStream (Debug RPMs) from RHUI
Red Hat Enterprise Linux 8 for ARM 64 - AppStream (RPMs) from RHUI
```



```
Red Hat Enterprise Linux 8 for ARM 64 - AppStream (Source RPMs) from RHUI
```

```
Proceed? (y/n)
```

- 在提示符处按 **y** 确认信息并创建授权证书。

验证

- 如果创建了授权证书，您会看到类似的消息：

```
.....+++++
...+++++
Entitlement certificate created at ./rhel8-for-rhui4.crt
-----
```

3.2. 使用 CLI 创建客户端授权证书

当红帽发布原始授权证书时，它会授予您所请求的存储库的访问权限。当您创建客户端授权证书时，您决定如何分配客户端并为每一证书创建一个单独的证书。然后，每个证书都可用于创建单个 RPM。

先决条件

- 授权证书必须由云供应商的 CA 证书签名。

流程

- 使用以下命令从 RHUI CLI 创建授权证书：

```
# rhui-manager client cert --repo_label rhel-8-for-x86_64-appstream-eus-rhui-source-rpms --
name rhuiclientexample --days 365 --dir /root/clientcert
.....+++++
.....+++++
Entitlement certificate created at /root/clientcert/rhuiclientexample.crt
```



注意

使用红帽存储库标签，而不是 ID。要获取所有标签的列表，请运行 **rhui-manager 客户端标签** 命令。如果您在证书中包含受保护的自定义存储库，请使用存储库的 ID。

验证

- 如果您成功创建并授权证书，则会显示类似的信息：

```
Entitlement certificate created at /root/clientcert/rhuiclientexample.crt
```

3.3. 验证客户端授权证书是否与 FUTURE 加密策略兼容

您可以通过检查客户端授权证书来验证您的 RHUI 实例是否合规：

- 由 RHUI 版本 3.1 到 4.0 生成的证书与 **FIPS** 和 **DEFAULT** 加密策略兼容。

- 由 RHUI 版本 4.1 及更新版本生成的证书与 **FIPS**、**DEFAULT** 和 **FUTURE** 加密策略兼容。

先决条件

- 确保您知道客户端授权证书的位置。
默认位置为 `/etc/pki/rhui/product/content.crt`。

流程

1. 在客户端 RPM 中，或者安装 RPM 的机器上，运行以下命令指定存储了客户端授权证书的路径：

```
# openssl x509 -noout -text -in /etc/pki/rhui/product/content.crt | grep bit
```

2. 检查 RSA 密钥长度：

- 如果长度为 2048 位，则客户端授权证书与 **FUTURE** 策略不兼容。
- 如果长度为 4096 位，则客户端授权证书与 **FUTURE** 策略兼容。

其他资源

- [使用 Red Hat Update Infrastructure Management Tool 创建客户端授权证书](#)
- [使用 CLI 创建客户端授权证书](#)

3.4. 使用 CLI 更改客户端配置 RPM 中的存储库 ID 前缀

在创建 RPM 时，您可以设置自定义存储库 ID 前缀或完全删除它。默认情况下，前缀为 **rhui-**。

流程

- 在 RHUA 节点上，使用 RHUI 安装程序命令设置或删除前缀：
 - 设置自定义前缀：

```
rhui-installer --rerun --client-repo-prefix CUSTOM_PREFIX
```

- 使用两个引号而不是前缀来完全删除前缀。

```
rhui-installer --rerun --client-repo-prefix ""
```

3.5. 使用 RED HAT UPDATE INFRASTRUCTURE 管理工具创建客户端配置 RPM

当红帽发布原始授权证书时，它会授予您所请求的存储库的访问权限。当您创建客户端授权证书时，您需要决定如何划分客户端并为每一证书创建一个单独的证书。然后，您可以使用每个证书在相应的客户机镜像上创建单独的 RPM 进行安装。

使用这个流程使用 RHUI 管理工具创建 RPM。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

- 按 **e** 键选择 **创建授权证书和客户端配置 RPM**。
- 在 **Client Entitlement Management** 屏幕中，按 **c** 从授权证书中选择**客户端配置 RPM**。
- 输入本地目录的完整路径来保存配置文件：

```
Full path to local directory in which the client configuration files generated by this tool
should be stored (if this directory does not exist, it will be created):
```

- 输入 RPM 的名称。
- 输入配置 RPM 的版本。默认版本为 2.0。
- 输入配置 RPM 的发行。默认版本为 1。
- 输入授权证书的完整路径来授权客户端访问特定频道。
- 输入到授权证书的私钥的完整路径。
- 选择任何未受保护的自定义存储库，以包含在客户端配置中。
- 按 **c** 键确认选择 或 **?** 是否有更多命令。

验证

- 如果成功创建了 RPM，则会显示类似的信息：

```
Successfully created client configuration RPM.
Location: /tmp/clientrpmtest-2.0/build/RPMS/noarch/clientrpmtest-2.0-1.noarch.rpm
```

3.6. 使用 CLI 创建客户端配置 RPM

当红帽发布原始授权证书时，它会授予您所请求的存储库的访问权限。当您创建客户端授权证书时，您需要决定如何划分客户端并为每一证书创建一个单独的证书。然后，您可以使用每个证书在相应的客户镜像上创建单独的 RPM 进行安装。

使用这个流程通过 CLI 创建 RPM。

流程

- 使用以下命令创建带有 RHUI CLI 的 RPM：

```
# rhui-manager client rpm --entitlement_cert /root/clientcert/rhuiclientexample.crt --
private_key /root/clientcert/rhuiclientexample.key --rpm_name clientrpmtest --dir /tmp --
unprotected_repos unprotected_repo1
Successfully created client configuration RPM.
Location: /tmp/clientrpmtest-2.0/build/RPMS/noarch/clientrpmtest-2.0-1.noarch.rpm
```



注意

使用 CLI 时，您还可以指定用于 RHUI 存储库的代理服务器的 URL，或使用 `_none_`（包括下划线）来覆盖客户端机器上的任何全局 `yum` 设置。要指定代理，请使用 `--proxy` 参数。

验证

1. 如果您成功创建了客户端配置 RPM，会显示类似的信息：

```
Successfully created client configuration RPM.  
Location: /tmp/clientrpmtest-2.0/build/RPMS/noarch/clientrpmtest-2.0-1.noarch.rpm
```

第 4 章 管理 RED HAT 权利证书

4.1. 红帽更新设备证书

RHUI 中的 RHUA 使用以下证书和密钥：

- 内容证书和私钥
- 授权证书和私钥
- SSL 证书和私钥
- 云供应商的 CA 证书

RHUA 使用内容证书和授权证书进行配置。RHUA 使用内容证书连接到红帽 CDN。它还使用红帽 CA 证书来验证与 Red Hat CDN 的连接。因为 RHUA 是连接到 Red Hat CDN 的唯一组件，它是部署此证书的 RHUI 组件。请注意，多个 RHUI 安装可以使用相同的内容证书。例如，Amazon EC2 云运行多个 RHUI 安装（每个区域一个），但每个 RHUI 安装都会使用相同的内容证书。

客户端使用授权证书允许访问 RHUI 中的软件包。要执行环境健康检查，RHUA 会尝试对每个 CDS 尝试 `yum` 请求。要成功，`yum` 请求必须指定有效的授权证书。

4.2. 内容交付服务器证书

RHUI 中的每个 CD 节点都使用以下证书和密钥：

- SSL 证书和私钥
- 云供应商的 CA 证书

CDS 的唯一证书是 SSL 证书，允许客户端与 CDS 间的 HTTPS 通信。SSL 证书限定于特定主机名，因此每个 CD 节点都需要一个唯一的 SSL 证书。如果连接到 CDS 时发生 SSL 错误，请验证证书的通用名称是否已设置为安装它的 CDS 的完全限定域名(FQDN)。

CA 证书用于验证客户端作为 `yum` 请求发送的授权证书是否已由云供应商签名。这可以防止恶意实例生成自己的授权证书，以便在 RHUI 中使用。

4.3. 客户端证书

RHUI 中的每个客户端都使用权利证书和密钥以及云供应商的 CA 证书。

授权证书及其私钥启用了从 CDS 返回到客户端的信息加密。连接到 CDS 时，每个客户端都使用授权证书来证明其有下载其软件包的权限。所有客户端都使用单个授权证书。

云供应商的 CA 证书用于在连接到 CDS 时验证 CDS 的 SSL 证书。这可确保恶意实例不会模拟 CDS，并将潜在的恶意软件包引入到客户端中。

CA 证书验证 SSL 证书，而不是授权证书。CDS 节点的反向情况是 `true`。SSL 证书和私钥用于加密从客户端到 CDS 的数据。CDS 中存在的 CA 证书验证 CDS 节点是否应该信任客户端发送的权利证书。

4.3.1. 列出证书的授权产品

Entitlements Manager 屏幕用于列出当前红帽内容证书中的授权产品并上传新证书。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **n** 键选择管理红帽授权证书。
3. 在 **Entitlements Manager** 屏幕中，按 **l** 列出有关当前内容证书的数据：

```
rhui (entitlements) => l

Red Hat Enterprise Linux 8 for ARM 64 - AppStream (Debug RPMs) from RHUI
Expiration: 02-27-2022  Certificate: c885597492374720bb5d398c3f65d1ed.pem

Red Hat Enterprise Linux 8 for ARM 64 - AppStream (RPMs) from RHUI
Expiration: 02-27-2022  Certificate: c885597492374720bb5d398c3f65d1ed.pem

Red Hat Enterprise Linux 8 for ARM 64 - AppStream (Source RPMs) from RHUI
Expiration: 02-27-2022  Certificate: c885597492374720bb5d398c3f65d1ed.pem

Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (Debug RPMs) from RHUI
Expiration: 02-27-2022  Certificate: c885597492374720bb5d398c3f65d1ed.pem

Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (RPMs) from RHUI
Expiration: 02-27-2022  Certificate: c885597492374720bb5d398c3f65d1ed.pem

Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (Source RPMs) from RHUI
Expiration: 02-27-2022  Certificate: c885597492374720bb5d398c3f65d1ed.pem
```

验证

1. 您将看到当前红帽内容证书中的授权产品列表。

4.3.2. 列出自定义存储库权限

您可以使用 **Entitlements Manager** 屏幕列出自定义存储库权利。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **n** 键选择管理红帽授权证书。
3. 在 **Entitlements Manager** 屏幕中，按 **c** 列出自定义存储库权利的数据：

```
rhui (entitlements) => c

Custom Repository Entitlements
For each entitlement URL listed, the corresponding repositories that are
configured with that entitlement are listed.
```

/protected/\$basearch/os

Name: Repo 1

URL: protected/i386/os

Name: Repo 2

URL: protected/x86_64/os

第 5 章 检查同步状态和调度

存储库是软件包(RPMs)的存储位置。RHEL 使用 **yum** 命令来搜索存储库、下载、安装和配置 RPM。RPM 包含运行应用程序所需的所有依赖项。RPM 还会下载您的仓库中的软件更新。

红帽内容初始同步的长度可能会有所不同。如果您选择尽快同步存储库，您可以在 CLI 中运行 **rhui-manager repo sync_all** 来同步 Red Hat Update Infrastructure 4 中的所有软件仓库。

5.1. 显示存储库同步概述

您可以使用 **Synchronization Status** 屏幕来显示特定存储库的信息。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **s** 键，以选择 **同步状态和调度**。
3. 在 **Synchronization Status** 屏幕中，按 **dr**：

```
-- Repository Summary Synchronization Status --

Last Refreshed: 02:01:22
(updated every 5 seconds, ctrl+c to exit)

Last Sync          Last Result
-----
Red Hat Enterprise Linux 8 for ARM 64 - BaseOS (Debug RPMs) from RHUI (8)
  Never              None
....
....
Red Hat Enterprise Linux 8 for x86_64 - AppStream from RHUI (Debug RPMs) (8.2)
  2021-07-29 17:45:41      Running
  Associating Content: 11001 (97%)
  Downloading Artifacts: 7376
```

5.2. 显示正在运行的同步

您可以使用 **Synchronization Status** 屏幕检查正在运行的同步任务的状态。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **s** 键，以选择 **同步状态和调度**。
3. 在 **Synchronization Status** 屏幕中，按 **rr**：

```
Last Refreshed: 02:06:46
```



```
(updated every 5 seconds, ctrl+c to exit)
```

```
Current Sync          Result
-----
Red Hat Enterprise Linux 8 for x86_64 - AppStream from RHUI (Debug RPMs) (8.2)
2021-07-29 17:45:41    Running
Associating Content: 11001 (97%)
Downloading Artifacts: 7376
```

5.3. 查看最后一次存储库同步的详情

您可以使用 **Synchronization Status** 屏幕查看最后一个存储库同步的详情。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **s** 键，以选择 **同步状态和调度**。
3. 在 **Synchronization Status** 屏幕中按 **vr**。
4. 输入您要查看详情的仓库号：

```
Enter value (1-66) or 'b' to abort:
```

验证

1. 如果所选存储库没有同步，则会显示类似的信息：

```
Repo: Red Hat Enterprise Linux 8 for x86_64 - AppStream from RHUI (Debug RPMs) (8.2)
No syncs have been completed for this repository.
```

5.4. 立即同步单个存储库

初始同步内容可能需要一段时间，通常为 10 到 20 分钟。如果您选择尽快同步存储库，您可以在 CLI 中运行 **rhui-manager repo sync_all** 来同步 Red Hat Update Infrastructure 4 中的所有软件仓库。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **s** 键，以选择 **同步状态和调度**。
3. 在 **Synchronization Status** 屏幕中按 **sr**：

```
Select one or more repositories to schedule to be synchronized before its scheduled time.
The sync will happen as soon as possible depending on other tasks that may be executing
in the RHUI. Sync requests for repositories with tasks in running
```

or pending state will be ignored.

```
Last Result  Next Sync      Repository
-----
```

- 通过在存储库名称旁边输入值来选择存储库。在确认您的产品选择前，每次输入一个仓库选择：

```
x 714: Error    2021-11-17 20:30:00  Red Hat Enterprise Linux 8 for ARM 64 - AppStream
(RPMs) from RHUI (8.4)
```

- 按 **c** 键确认：

```
The following repositories will be scheduled for synchronization:
Red Hat Enterprise Linux 8 for ARM 64 - AppStream (RPMs) from RHUI (8.4)
Proceed? (y/n) y
```

- 按 **y** 继续：

```
Scheduling sync for Red Hat Enterprise Linux 8 for ARM 64 - AppStream (RPMs) from RHUI
(8.4)...
... successfully scheduled for the next available timeslot.
```



注意

如果所选存储库的任务正在运行，这个消息会显示。**忽略 Red Hat Enterprise Linux 8 for x86_64 的同步请求 - AppStream from RHUI(Debug RPMs)(8.2), 因为正在运行的任务当前保留了存储库。**

5.5. 取消活跃同步任务

大多数环境会按照计划的基础同步存储库。您可能会遇到需要取消活跃同步任务的情况。

先决条件

- 存在现有的软件仓库。
- 有活跃的同步任务。

流程

- 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

- 按 **s** 键，以选择 **同步状态和调度**。
- 在 **Synchronization Status** 屏幕中，按 **ca** 键选择 **cancel active sync tasks**。
- 为要取消的任务或任务输入值：

```
Select one or more repositories for which you want to cancel their active tasks.
- 1: Red Hat Enterprise Linux 8 for x86_64 - AppStream from RHUI (Debug RPMs) (8.2)
Enter value (1-1) to toggle selection, 'c' to confirm selections, or '?' for more commands:
```

- 按 **c** 键确认您的选择。
- 按 **y** 取消同步任务或任务：

```
The active tasks will be canceled for the following repositories:
  Red Hat Enterprise Linux 8 for x86_64 - AppStream from RHUI (Debug RPMs) (8.2)
Proceed? (y/n)
```

验证

- 如果您取消活跃同步任务，则会显示类似的消息：

```
Canceling active task for repo Red Hat Enterprise Linux 8 for x86_64 - AppStream from
RHUI (Debug RPMs) (8.2) ...
... done
```

5.6. 取消等待同步任务

大多数环境会按照计划的基础同步存储库。您可能会遇到需要取消待处理的同步任务的情况。

先决条件

- 存在现有的软件仓库。
- 有调度的同步任务。

流程

- 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

- 按 **s** 键，以选择 **同步状态和调度**。
- 在 **Synchronization Status** 屏幕中，按 **cw** 键以选择 **取消等待同步任务**。
- 为要取消的任务或任务输入值：

```
Select one or more repositories for which you want to cancel their pending tasks.
- 1: Single Sign-On 7.4 for RHEL 8 x86_64 (Source RPMs) from RHUI
Enter value (1-1) to toggle selection, 'c' to confirm selections, or '?' for more commands: 1
```

- 按 **c** 键确认您的选择：

```
Select one or more repositories for which you want to cancel their pending tasks.
x 1: Single Sign-On 7.4 for RHEL 8 x86_64 (Source RPMs) from RHUI
Enter value (1-1) to toggle selection, 'c' to confirm selections, or '?' for more commands: c
```

- 按 **y** 继续：

```
The pending tasks will be canceled for the following repositories:
  Single Sign-On 7.4 for RHEL 8 x86_64 (Source RPMs) from RHUI
Proceed? (y/n) y
```

验证

1. 如果取消成功，则会显示类似的信息：

```
Canceling pending task for repo Single Sign-On 7.4 for RHEL 8 x86_64 (Source RPMs) from RHUI ...
... done
```

2. 如果没有待处理的同步任务，则会显示以下消息：

```
There are no repositories with pending sync related tasks.
```

5.7. 查看和更改存储库自动发布状态

您可以使用 **Synchronization Status** 屏幕来查看和修改存储库的自动发布状态。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **s** 键，以选择 **同步状态和调度**。

3. 在 **Synchronization Status** 屏幕中按 **ap**：

```
rhui (sync) => ap
```

```
Select one or more repositories to toggle the auto-publish status.
The operation will be executed as soon as possible depending on other tasks
that may be executing in the RHUI.
```

```
      Status | Repository
-----|-----
```

```
Select one or more repositories:
```

```
Custom Repositories
```

```
Red Hat Repositories: yum
```

```
- 713:  AUTO Red Hat Enterprise Linux 8 for ARM 64 - AppStream (RPMs) from RHUI
(8.3)
- 714:  AUTO Red Hat Enterprise Linux 8 for ARM 64 - AppStream (RPMs) from RHUI
(8.4)
- 719:  AUTO Red Hat Enterprise Linux 8 for ARM 64 - AppStream (Source RPMs)
from RHUI (8.3)
- 720:  AUTO Red Hat Enterprise Linux 8 for ARM 64 - AppStream (Source RPMs)
from RHUI (8.4)
```

4. 输入一个值(1-1631)，切换选择，**c** 确认选择或 **?** 有关更多命令：

```
The following repositories will have their auto-publish status changed:
```

```

Red Hat Repositories
yum
  Red Hat Enterprise Linux 8 for ARM 64 - AppStream (RPMs) from RHUI (8)

```

- 按 **c** 键确认您的选择。
- 按 **y** 键继续。

验证

- 在制作并确认一个选择时会显示类似的信息：

```

Scheduling a task to turn off auto-publish status of repository Red Hat Enterprise Linux 8 for
ARM 64 - AppStream (RPMs) from RHUI (8)

```

其他资源

- Red Hat Enterprise Linux 软件仓库可通过红帽认证的云和服务提供商(CCSP)合作伙伴通过 RHUI 交付。

5.8. 查看和发布存储库工作流

您可以使用 **Synchronization Status** 屏幕来查看并更改存储库的工作流。

流程

- 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

- 按 **s** 键，以选择 **同步状态和调度**。
- 在 **Synchronization Status** 屏幕中，按 **wf**。
- 输入一个值(1-1631)，切换选择，**c** 确认选择或 **?** 有关更多命令：

```

The following repositories will be scheduled for workflow push:
Red Hat Repositories
yum
  Red Hat Enterprise Linux 8 for ARM 64 - AppStream (RPMs) from RHUI (8.4)

```

- 按 **y** 继续：

验证

- 如果调度成功，则会显示类似的信息：

```

Scheduling a task for generating metadata version 0 for repo Red Hat Enterprise Linux 8 for
ARM 64 - AppStream (RPMs) from RHUI (8.4) ...
... task scheduled.

```

5.9. 将存储库导出到文件系统

您可以使用 **Synchronization Status** 屏幕将存储库导出到文件系统。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **s** 键，以选择 **同步状态和调度**。
3. 在 **Synchronization Status** 屏幕中按 **ex**。
4. 输入一个值来切换选择。
5. 按 **c** 键确认选择：

```
The following repositories will be exported:
Red Hat Repositories
  yum
  Red Hat Enterprise Linux 8 for ARM 64 - AppStream (Source RPMs) from RHUI (8)
```

6. 按 **y** 键继续。

验证

1. 如果存储库导出到文件系统，则会显示类似的信息：

```
[1/1] Exporting version 1 of the repo Red Hat Enterprise Linux 8 for ARM 64 - AppStream
(Source RPMs) from RHUI (8).
```

第 6 章 管理内容交付服务器

CDS 节点是内容交付网络(CDN)的主要组件，为客户端提供高可用性。以地理分散的方式运行服务器也可以提高响应时间。

您可以使用 **Content Delivery Server(CDS)Management** 屏幕列出、添加、删除和重新安装 CDS 节点。

最多是云提供商，决定在分配所需设备时的最佳操作方法。Red Hat Update Infrastructure Management Tool 配置 RPM 将软件包目录与 NGINX 配置链接到一起，以便为它提供服务。



重要

在 CDS 安装过程中，NGINX 配置通过 `/etc/nginx/conf.d/ssl.conf` 文件进行处理。

如果多个客户端遇到与库进行更新的问题，这可能表示 RHUI 存在问题。

6.1. 列出由 RHUI 4 管理的所有已知 CDS 实例

您可以使用 **Content Delivery Server(CDS)Management** 屏幕列出由 Red Hat Update Infrastructure 4 管理的所有 CDS 节点。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **c** 键选择 **管理内容交付服务器(CDS)**：
3. 在 **Content Delivery Server(CDS)Management** 屏幕中，按 **l** 列出 Red Hat Update Infrastructure 4 管理的所有已知的 CDS 节点：

```
Hostname:      <cds1.example.com>
SSH Username:  <cloud-user>
SSH Private Key:  /<cloud-user>/.ssh/id_rsa_rhua
```

6.2. 注册新的 CDS

Red Hat Update Infrastructure Management Tool 提供了几个选项，用于在 RHUI 中配置 CDS。

先决条件

- 确保 **sshd** 在 CDS 节点上运行，且 **端口 443** 已被打开。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **c** 键选择 **管理内容交付服务器(CDS)**。

3. 在 **Content Delivery Server(CDS)Management** 屏幕中，按 **a** 来添加新的 CDS 实例。

4. 输入要添加的 CDS 主机名：

```
Hostname of the CDS instance to register:
cds1.example.com
```

5. 输入对 CDS 具有 SSH 访问权限的用户名，并具有 sudo 权限。

```
Username with SSH access to <cds1.example.com> and sudo privileges:
<cloud-user>
```

6. 输入到 SSH 私钥的绝对路径，以登录到 CDS 并按 **Enter** 键。

```
Absolute path to an SSH private key to log into <cds1.example.com> as <cloud-user>:
/home/<cloud-user>/.ssh/id_rsa_rhua
```

7. **可选**：如果要使用自定义 SSL 证书，请输入自定义 SSL 证书、SSL 密钥和 SSL crt 文件的绝对路径。



注意

如果没有提供 SSL 证书，它将自动生成。

```
Optional absolute path to user supplied SSL key file:
/home/<cloud-user>/custom_ssl.key
```

```
Optional absolute path to user supplied SSL crt file:
/home/<cloud-user>/custom_ssl.crt
```

```
.....
The following CDS has been successfully added:
```

```
Hostname:      <cds1.example.com>
SSH Username:  <cloud-user>
SSH Private Key: /home/<cloud-user>/.ssh/id_rsa_rhua
```

```
The CDS will now be configured:
.....
```

```
The CDS was successfully configured.
```

8. 如果添加内容交付服务器失败，请检查防火墙规则是否在 RHUA 和 CDS 之间允许访问。

9. 运行 **mount** 命令，查看共享存储是否挂载为读写。

```
[root@rhua ~]# mount | grep rhui
```

```
nfs.example.com:/export on /var/lib/rhui/remote_share type nfs4
(rw,relatime,vers=4.2,rsize=1048576,wsiz=1048576,namlen=255,hard,proto=tcp,timeo=600,re
trans=2,sec=sys,clientaddr=10.8.41.163,local_lock=none,addr=10.8.41.163)
```

10. 成功配置后，对所有剩余的 CDS 节点重复这些步骤。

6.3. 重新安装并重新应用配置到 CDS

您可能会遇到需要重新安装并重新应用 CDS 配置的情况。红帽更新基础架构管理工具提供了一种简单的方式来完成此任务。

先决条件

- 至少一个安装的 CDS

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **c** 键选择 **管理内容交付服务器(CDS)**。
3. 在 **Content Delivery Server(CDS)Management** 屏幕中，按 **r** 键选择**重新安装并重新应用配置到现有的 CDS 实例**。Red Hat Update Infrastructure Management Tool 会自动执行所有重新安装和重新配置任务。
4. 选择要重新安装的 CDS：

```
1 -
Hostname:      <cds1.example.com>
SSH Username:  <cloud-user>
SSH Private Key:  /<cloud-user>/.ssh/id_rsa_rhua
```

5. 输入值或 **b** 以中止：1.

```
Checking that the RHUA services are reachable from the instance...
Done.
```

```
Installing and configuring the CDS...
```

```
PLAY [Registering a CDS instance] *****
```

```
PLAY RECAP *****
```

```
cloud-user@cds1.example.com : ok=24  changed=10  unreachable=0  failed=0
skipped=2  rescued=0  ignored=0
```

```
Done.
```

验证

1. 通过查看代码输出，检查是否成功重新安装并重新配置了 CDS：

```
Ensuring that instance ports are reachable ...
Done.
```

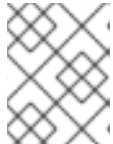
6.4. 配置 CDS 以接受旧的 CA

默认情况下，内容交付服务器(CDS)节点只接受由您的 RHUI 系统上当前配置的证书颁发机构(CA)签名的权利证书。但是，您可能希望接受之前创建的 CA，以便在更改主 CA 或 CA 证书过期时，客户端可以继续工作。

此流程通过在 CDS 节点上安装 CA 证书，在 RHUI 上支持旧的 CA 的说明。

先决条件

- 请确定您正在运行 RHUI 的最新版本。



注意

如果您已安装旧版本的 RHUI，则必须在 **rhui-manager** 中重新安装您的 CDS 节点。

流程

1. 在 CDS 节点上，创建 **/etc/pki/rhui/legacy** 目录（如果它尚不存在）：

```
# mkdir /etc/pki/rhui/legacy
```

2. 将旧的 CA 证书保存到目录中。

验证

- 将 CA 证书存储在目录中后，CDS 节点就会立即开始接受旧的 CA。

6.5. 配置 CDS 以停止接受旧的 CA

要限制您的内容交付服务器(CDS)节点接受传统的证书颁发机构(CA)，请删除对应的 CA 证书。

先决条件

- 客户端不再使用 CA。

流程

1. 在 CDS 节点上，导航到 **/etc/pki/rhui/legacy/** 目录：

```
# cd /etc/pki/rhui/legacy/
```

2. 可选：备份现有的 CA 证书：
3. 删除与您要限制的 CA 证书对应的 CA 证书：

```
# rm example-legacy.crt
```

验证

- 删除 CA 证书后，CDS 节点会立即停止接受旧的 CA。

6.6. 取消注册一个 CDS

您可以取消注册（删除）不使用的 CDS 实例。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **c** 键选择 **管理内容交付服务器(CDS)**。
3. 在 **Content Delivery Server(CDS)Management** 屏幕中，按 **d** 键删除 CDS 实例。
4. 输入要删除的 CDS 的主机名：

```
Hostname of the CDS instance to unregister:  
cds1.example.com
```

第 7 章 管理 HAPROXY 负载均衡器实例

如果有多个 CDS，则必须存在负载均衡解决方案，才能在所有服务器中分散客户端 HTTPS 请求。Red Hat Update Infrastructure 4 附带 HAProxy，但您可以选择在安装过程中要使用的负载均衡解决方案（例如，云提供商中的一个）。如果使用 HAProxy，还必须决定要进行多少个节点。

7.1. 列出由 RHUI 4 管理的所有已知 HAPROXY 负载均衡器实例

您可以使用 **Load-balancer(HAProxy)** 管理屏幕显示 RHUI 4 管理的所有已知的 HAProxy 负载均衡器实例。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **I** 选择 **管理 HAProxy 负载均衡器实例**。
3. 在 **Load-balancer(HAProxy)**管理 屏幕中，按 **I** 列出 RHUI 管理的负载均衡器实例：

```
Hostname:      <haproxy1.example.com>
SSH Username:  <cloud-user>
SSH Private Key:  /<cloud-user>/.ssh/id_rsa_rhua
```

7.2. 注册新的 HAPROXY 负载均衡器

Red Hat Update Infrastructure 4 使用 DNS 访问 CDN。在大多数情况下，您的实例应预先配置为与作为云基础架构一部分托管的正确 DNS 服务器进行通信。如果您运行自己的 DNS 服务器或更新客户端 DNS 配置，您很可能会看到与 **yum Could 没有联系任何 CDS 负载均衡器的** 错误。在这些情况下，请检查您的 DNS 服务器是否为请求转发到云的 DNS 服务器，或者您的 DNS 客户端被配置为回退到云 DNS 服务器以进行名称解析。

当 rhui-installer 运行时，当 rhui-installer 运行时，使用多个 HAProxy 节点需要一个循环 DNS 条目，用作 **--cds-lb-hostname** 参数的值，当 **rhui-installer** 运行时(**cds.example.com**)解析为所有 HAProxy 节点的 IP 地址。[如何配置 DNS Round Robin](#) 展示了配置循环 DNS 的方法。在 Red Hat Update Infrastructure 4 中，这些将是 HAProxy 节点的 IP 地址，它们映射到在调用 **rhui-installer** 时，被映射到指定为 **--cds-lb-hostname** 的主机名。

先决条件

1. 确保 **sshd** 在 HAProxy 负载均衡器节点上运行，并打开了 **端口 443**。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **I** 选择 **管理 HAProxy 负载均衡器实例**。
3. 在 **Load-balancer(HAProxy)**管理 屏幕中，按 **a** 来添加新的负载均衡器实例。

4. 输入新负载均衡器的主机名：

```
Hostname of the HAProxy Load-balancer instance to register:
<haproxy1.example.com>
```

5. 输入有对负载均衡器的 SSH 访问的用户名，并具有 sudo 权限：

```
Username with SSH access to cds.example.com and sudo privileges:
<cloud-user>
```

6. 输入到负载均衡器实例的 SSH 私钥的绝对路径，然后按 **Enter**：

```
Absolute path to an SSH private key to log into cds.example.com as <cloud-user>:
/<cloud-user>/ssh/id_rsa_rhua
```

7. **可选**：输入用户提供的 HAProxy 配置文件的可选绝对路径，然后按 **Enter** 键。如果您没有指定自定义配置文件的路径，则使用默认的文件 `/usr/share/rhui-tools/templates/haproxy.cfg`。

```
Optional absolute path to user supplied HAProxy config file:
```

```
.....
The following load-balancer has been successfully added:
```

```
Hostname:      <haproxy1.example.com>
SSH Username:  <cloud-user>
SSH Private Key: /<cloud-user>/ssh/id_rsa_rhua
```

```
The load-balancer will now be configured:
```

8. 如果负载均衡器无法添加，请检查防火墙规则是否允许 RHUA 和负载均衡器间的访问。
9. 成功配置后，对所有剩余的负载均衡器实例重复这些步骤。

验证

- 这时将显示以下信息：

```
The HAProxy Load-balancer was successfully configured.
```

其他资源

- [HAProxy 配置](#)

7.3. 重新安装并重新将配置重新应用到 HAPROXY 负载均衡器中

您可能会遇到一个需要重新安装并重新应用 HAProxy 负载均衡器的配置的情况。红帽更新基础架构管理工具提供了一种简单的方式来完成此任务。

先决条件

- 确保 `sshd` 在 HAProxy 负载均衡器节点上运行，并打开了 **端口 443**。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

2. 按 **I** 选择 **管理 HAProxy 负载均衡器实例**。
3. 在 **Load-balancer(HAProxy)管理** 屏幕中，按 **r** 来重新安装配置并重新应用配置到负载均衡器实例。



重要

关键在于恢复中包含的文件保留其当前属性。

Red Hat Update Infrastructure Management Tool 会自动执行所有重新安装和重新配置任务。

1. 选择要重新安装的负载均衡器：

```
1 -
  Hostname:      <haproxy1.example.com>
  SSH Username:  <cloud-user>
  SSH Private Key:  /<cloud-user>/.ssh/id_rsa_rhua
```

2. 输入值或 **b** 以中止：1.

```
Installing and configuring the HAProxy Load-balancer...

PLAY [Registering a load balancer instance] *****

PLAY RECAP *****
cloud-user@haproxy1.example.com : ok=8  changed=3  unreachable=0  failed=0
skipped=0  rescued=0  ignored=0

Done.
```

验证

1. 通过查看代码输出，检查是否成功重新安装并重新配置了负载均衡器：

```
Ensuring that HAProxy is available...
Done.
```

7.4. 取消注册 HAProxy 负载均衡器

您可以取消注册（删除）不会用到的 HAProxy 负载均衡器实例。

先决条件

- 确保 **sshd** 在 HAProxy 负载均衡器节点上运行，并打开了 **端口 443**。

流程

1. 进入 Red Hat Update Infrastructure Management Tool 主页：

```
[root@rhua ~]# rhui-manager
```

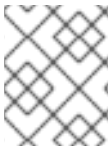
2. 按 **l** 选择 **管理 HAProxy 负载均衡器实例**。
3. 在 **Load-balancer(HAProxy)管理** 屏幕中，按 **d** 键删除负载均衡器的实例。
4. 输入要删除的负载均衡器的主机名：

```
Hostname of the load-balancer instance to unregister:  
<haproxy1.example.com>
```

第 8 章 管理容器

您可以使用 RHUI 在 Linux 容器中自动部署应用程序。使用容器具有以下优点：

- 需要比虚拟机少存储和内存空间：因为容器只保存运行应用程序所需的内容，因此与容器相比，与包括整个操作系统的虚拟机相比，节省和共享效率更高。
- 提高性能：因为您没有运行一个完全独立的操作系统，因此容器通常比执行新虚拟机开销的应用程序要快。
- 安全：由于容器通常具有自己的网络接口、文件系统和内存，因此在该容器中运行的应用可以与主机计算机上的其他活动隔离和保护。
- 灵活：由于应用在容器中包含应用程序的运行时要求，容器可以在多个环境中运行。



注意

默认情况下不启用对容器的支持。您必须通过设置 `--container-support-enabled` 标志来手动启用支持。

8.1. 了解 RED HAT UPDATE INFRASTRUCTURE 中的容器

容器是一个应用程序沙盒。每个容器基于包含所需配置数据的镜像。从镜像启动容器时，该镜像之上添加一个可写层。每次提交容器时，都会添加一个新镜像层来存储您的更改。

镜像是永远不会修改的只读层。所有更改都在顶层进行，更改只能通过创建新镜像来保存更改。每个镜像依赖于一个或多个父镜像。

平台镜像是没有父平台的镜像。平台镜像定义运行容器化应用程序所需的运行时环境、软件包和实用程序。平台镜像是只读的，因此任何更改都会反映在其上堆栈的复制镜像中。

8.2. 在 RED HAT UPDATE INFRASTRUCTURE 中添加容器

您可以使用 `rhui-manager` 工具使用 Repository Management 部分添加容器。

流程

1. 在 RHUA 节点上，启用对容器的支持，并将其应用到所有 CDS 节点上：

```
# rhui-installer --rerun --container-support-enabled True
# rhui-manager --noninteractive cds reinstall --all
```

2. 可选：编辑 `/etc/rhui/rhui-tools.conf` 文件，并通过删除 `[container]` 部分中的以下行来设置容器 registry 凭证。

如果您有一个 RHUI 4.1.1 或更高版本的清洁安装，则最后几行包含包含特定于 podman 的选项和手写的 `[container]` 部分。如果您从早期版本的 RHUI 更新，则该部分位于 `etc/rhui/rhui-tools.conf.rpmnew` 文件的末尾，您可以将其复制到 `rhui-tools.conf` 文件。

```
[container]
...
registry_username: your_RH_login
registry_password: your_RH_password
```




注意

如果您通常与 registry. **redhat.io** 不同的 registry 同步，还要相应地更改 registry_url 和 registry_auth 选项的值。

- 在 RHUA 节点上，运行 **rhui-manager**：

```
# rhui-manager
```

- 按 **r** 访问 Repository Management 屏幕。

```
== Red Hat Update Infrastructure Management Tool ==
```

```
== Repository Management ==
```

```
l list repositories currently managed by the RHUI
i display detailed information on a repository
a add a new Red Hat content repository
ac add a new Red Hat container
c create a new custom repository (RPM content only)
d delete a repository from the RHUI
u upload content to a custom repository (RPM content only)
ur upload content from a remote web site (RPM content only)
p list packages in a repository (RPM content only)
```

```
Connected: rhua.example.com
```

- 按 **ac** 来添加新的红帽容器。

```
rhui (repo) => ac Specify URL of registry [https://registry.redhat.io]:
```

- 如果要添加的容器存在于非默认 registry 中，请输入 registry URL。按 **Enter** 键而不输入任何操作来使用默认的 registry。
- 在 registry 中输入容器的名称：

```
jboss-eap-6/eap64-openshift
```

- 为容器输入唯一 ID。
RHUI-manager 将容器的名称从 registry 转换为 Pulp 中可用的格式，方法是将斜杠和点替换为下划线。您可以通过按 **Enter** 或输入您选择的名称来使用这样的转换名称。
- 输入容器的显示名称。

```
jboss-eap-6_eap64-openshift
```

- 可选：收到提示，在 RHUI 配置中设置您的登录和密码。
- 验证显示的摘要。

```
The following container will be added:
Registry URL: http://registry.redhat.io
Container Id: jboss-eap-6_eap64-openshift
```

```

Display Name: jboss-eap-6_eap64-openshift
Upstream Container Name: jboss-eap-6/eap64-openshift
Proceed? (y/n)

```

- 按 **y** 继续并添加容器。

```

y
Successfully added container jboss-eap-6_eap64-openshift

```

8.3. 同步容器存储库

将容器添加到红帽更新基础架构后，您可以使用 **rhui-manager** 工具同步容器。

流程

- 在 RHUA 节点上，运行 **rhui-manager**：

```

# rhui-manager

```

- 按 **s** 来访问 **同步状态和调度** 屏幕。
- 按 **sr** 立即同步单个存储库。
- 输入您要同步的存储库数量。
- 按 **c** 确认选择。
- 验证存储库，然后按 **y** 来同步或 **n** 取消。

```

The following repositories will be scheduled for synchronization: jboss-eap-6_eap64-
openshift
Proceed? (y/n) y
Scheduling sync for jboss-eap-6_eap64-openshift...
... successfully scheduled for the next available timeslot.

```

8.4. 生成容器客户端配置

RHUI 客户端可以使用客户端配置从 RHUI 中拉取容器。RPM 包含负载均衡器的证书，您可以使用它来将负载均衡器添加到容器 registry 中，并修改容器配置。

流程

- 在 RHUA 节点上，运行 **rhui-manager**：

```

# rhui-manager

```

- 按 **e** 访问 **授权证书和客户端配置 RPM** 屏幕。
- 按 **d** 创建容器客户端配置 RPM。
- 输入要保存配置文件的本地目录的完整路径。

```

/root/

```

-
- 5. 输入 RPM 的名称。

```
containertest
```

- 6. 输入配置 RPM 的版本号。默认值为 **2.0**。
- 7. 输入配置 RPM 的发行号。默认值为 **1**。
- 8. 输入证书应有效的天数。默认值为 **365**。

```
Successfully created client configuration RPM.  
Location: /root/containertest-2.0/build/RPMS/noarch/containertest-2.0-1.noarch.rpm
```

8.5. 在客户端上安装容器配置 RPM

生成容器配置 RPM 后，您可以通过将其导入到本地机器来在客户端上安装它。

流程

1. 从 RHUA 节点检索 RPM 到您的本地机器：

```
# root@rhua.example.com:/root/containertest-2.0/build/RPMS/noarch/containertest-2.0-1.noarch.rpm .
```

2. 将 RPM 从本地计算机传输到客户端。

```
# scp containertest-2.0-1.noarch.rpm root@cli01.example.com:.
```

3. 切换到客户端并安装 RPM：

```
[root@cli01 ~]# yum install containertest-2.0-1.noarch.rpm
```

8.6. 在客户端上测试 PODMAN PULL 命令

您可以使用 `podman pull` 命令验证容器中的内容。

流程

1. 运行 `podman pull` 命令。

```
[root@cli01 ~]# podman pull jboss-eap-6_eap64-openshift  
  
Resolving "jboss-eap-6_eap64-openshift" using unqualified-search registries  
(/etc/containers/registries.conf)  
Trying to pull cds.example.com/jboss-eap-6_eap64-openshift:latest...  
Getting image source signatures  
Copying blob b0e0b761a531 done  
Copying blob aa23ac04e287 done  
Copying blob 0d30ea1353f9 done  
Copying config 3d0728c907 done
```

Writing manifest to image destination

Storing signatures

3d0728c907d55d9faedc4d19de003f21e2a1ebdf3533b3d670a4e2f77c6b35d2

2. 如果 **podman pull** 命令失败，请检查 **rhui-manager** 状态。同步可能尚未执行，您必须等待同步。

Resolving "jboss-eap-6_eap64-openshift" using unqualified-search registries

(/etc/containers/registries.conf)

Trying to pull cds.example.com/jboss-eap-6_eap64-openshift:latest...

Error: initializing source docker://cds.example.com/jboss-eap-6_eap64-openshift:latest: reading manifest latest in cds.example.com/jboss-eap-6_eap64-openshift: manifest unknown: Manifest not found.

第 9 章 配置文件、退出代码和日志文件

以下配置文件 RHUI 管理器退出代码和日志文件在 Red Hat Update Infrastructure 4 中使用。

配置文件

表 9.1. 配置文件

组件	文件或目录	使用
Red Hat Update Appliance	/etc/pulp/*	Pulp 配置文件
	/etc/rhui/rhui-tools.conf	RHUI-manager 配置文件
	/etc/pki/rhui/*	Red Hat Update Infrastructure 的证书
	/root/.rhui/answers.yaml	用于设置 RHUA
	/etc/rhui/rhui-subscription-sync.conf	配置订阅同步脚本
内容交付服务器	/etc/pki/rhui/certs/	CDS 的证书
HAProxy	/etc/haproxy/haproxy.cfg	HAProxy 配置文件

RHUI Manager Exit Codes

RHUI Manager 使用以下代码来指示运行 **rhui-manager status** 命令并运行 **rhui-manager** CLI 命令的结果。

表 9.2. RHUI Manager Exit Codes

状态代码	描述
0	成功
1	常规错误或存储库同步错误
2	CDS 上的 SSL 证书错误
32	权利 CA 或 SSL 证书过期警告
64	权利 CA 或 SSL 证书过期错误
128	一个或多个 RHUI 服务不在 RHUA、CDS 或 HAProxy 节点上运行
239	无法删除存储库，因为它不存在。

状态代码	描述
240	所需资源存在问题。例如，无法构建客户端配置 RPM，因为没有找到有效的存储库。
241	无法调度同步任务，因为指定了未知存储库。 故障排除 * 检查 spelling * Add the repository first * Check logs for Pulp 问题
242	由于 Pulp 问题，无法创建自定义存储库。检查消息和日志以了解详细信息。
243	无法添加红帽软件仓库，因为它们已在 RHUI 中存在，并且其中某些软件仓库在权利中没有。
244	无法创建自定义存储库，因为它已存在于 RHUI 中。
245	无法添加红帽软件仓库，因为它已存在于 RHUI 中。
246	无法添加红帽软件仓库，因为它在权利中没有。使用 rm -f /var/cache/rhui tracing ，检查拼写或删除存储库映射缓存，然后重试。
247	由于 Pulp 问题，无法添加红帽存储库。检查消息和日志以了解详细信息。
248	从 RHUI 3 迁移到 RHUI 4 已停止，因为 RHUI 4 中已存在一个或多个红帽存储库。您必须删除软件仓库或使用 --force 标志。
249	RHUI 配置 /etc/rhui/rhui-tools.conf 无效。检查消息了解详情。
250	授权证书不可写入。
251	授权证书已过期。
252	授权证书无效，因为它不包含 RHUI 软件仓库。
253	授权证书文件不是有效的证书。
254	命令行错误：RHUI CLI 无法因为网络问题而运行。
255	参数错误：未提供所需的参数。

日志文件

表 9.3. 日志文件

组件	文件或目录	使用
Red Hat Update Appliance	/root/.rhui/rhui.log* 假定以 root 用户身份运行 rhui-manager	Red Hat Update Infrastructure Management Tool 日志
	/var/log/messages	除当前写入的文件外，最新的四个版本还会保留。
	/var/log/nginx/*	NGINX 日志
	/var/log/messages	Qpid 日志
	/var/log/rhui-subscription-sync.log	订阅同步日志
内容交付服务器	/var/log/nginx/*	NGINX 日志
	/nginx/ssl-access.log	Logging - Successful yum repolist : CDS 节点记录此客户端活动
客户端	RHEL 7 和更早版本的 /var/log/yum.log	yum 命令日志
客户端	RHEL 8 及更新的版本的 /var/log/dnf.log	dnf 命令日志
	/var/log/messages	客户端 syslog
	yum/dnf 命令的输出	

第 10 章 使用 RHUI 4 命令

Red Hat Update Infrastructure Management Tool 中的屏幕从 RHUpdate 访问，提供菜单选项，供您配置和更新各种 Red Hat Update Infrastructure 4 组件。

10.1. 使用 RHUI 4 菜单选项

Red Hat Update Infrastructure Management Tool 提供了用于配置和管理 RHUI 4 的交互式命令。

表 10.1. Red Hat Update Infrastructure Management Tool 菜单和命令

screen	屏幕命令	菜单选项	菜单选项命令
管理软件仓库	r		
仓库管理		列出当前由 RHUI 管理的仓库	l
		显示仓库的详细信息	i
		添加一个新的红帽内容存储库	a
		创建新的自定义存储库 (仅限RPM 内容)	c
		从 RHUI 中删除存储库	d
		将内容上传到自定义存储库 (仅限RPM 内容)	u
		从远程网站上传内容 (仅限RPM 内容)	ur
		列出存储库中的软件包 (仅限RPM 内容)	p
管理内容交付服务器 (CDS)	c		
内容交付服务器(CDS)管理		列出由 RHUI 管理的所有已知 CDS 实例	l
		注册 (添加) 一个新的 CDS 实例	a
		重新安装并重新应用配置到现有的 CDS 实例	r

screen	屏幕命令	菜单选项	菜单选项命令
		从 RHUI 取消注册（删除）CDS 实例	d
管理 HAProxy 负载均衡器实例	l		
load-balancer(HAProxy) 管理		列出由 RHUI 管理的所有已知 HAProxy 负载均衡器实例	l
		注册（添加）一个新的 HAProxy Load-balancer 实例	a
		将配置重新安装并重新应用到现有的 HAProxy 负载均衡器实例中	r
		从 RHUI 取消注册（删除）HAProxy Load-balancer 实例	d
同步状态和调度	s		
同步状态		显示存储库同步概述	dr
		显示正在运行的同步	rr
		查看最后一次存储库同步的详情	vr
		立即同步单个存储库	sr
		取消活跃的同步任务	ca
		取消等待同步任务	cw
		查看/更改存储库自动发布状态	ap
		查看/高级存储库工作流	wf
		将仓库导出到文件系统	ex
创建权利证书和密钥配置 RPM	e		

screen	屏幕命令	菜单选项	菜单选项命令
客户端权利管理		生成授权证书	e
		从授权证书创建客户端配置 RPM	c
管理 Red Hat 授权证书	n		
授权管理器		列出红帽内容证书授权	l
		列出自定义存储库权利	c
		上传新的或更新的红帽内容证书	u
管理 RHUI 用户	u		
用户管理器		更改管理员密码（由注销跟踪）	p

10.2. 使用 RHUI 4 CLI 选项

Red Hat Update Infrastructure 4 的大多数管理任务都是其安装中。安装后，它会自行运行，定期从红帽 CDN 获取更新的软件包，并自动将那些软件包提供给客户端。

名为 Red Hat Update Infrastructure Management Tool 的命令行界面（使用 **rhui-manager** 运行）有助于安装。此工具提供了每个 RHUI 组件所需的配置元素的交互式提示：RHUA、CDS 和 load-balancer。该工具还提供了用来获取红帽提供的内容证书，以便在连接到 Red Hat CDN 时使用并生成内部云相关证书，以供客户端用于连接到 RHUI。Red Hat Update Infrastructure Management Tool 允许云供应商生成客户端配置捆绑包，以便在客户端 RHEL 实例中安装。此捆绑包允许客户端从 RHUI 安装中获取更新。

Red Hat Update Infrastructure Management Tool 使用互动 shell；一些功能也可以在 shell 提示符中运行。Red Hat Update Infrastructure Management Tool 使用七大命令。对于每个命令的子命令，如果子命令需要一个或多个选项 **-h** 和 **--help**，则提供一个选项列表。

查看所有选项和命令。

```
[root@ip-10-141-150-145 ~]# rhui-manager --help
Usage: rhui-manager [options]
```

OPTIONS

```
-h/--help show this help message and exit
--debug enables debug logging
--noninteractive prevents console input, used for scripting
--config absolute path to the configuration file; defaults to /etc/rhui/rhui-tools.conf
--server location of the RHUA server (overrides the config file)
--username if specified, previously saved authentication credentials are ignored and this username is used to login
--password used in conjunction with --username
--logout logout from the active session
```

COMMANDS

cert : Red Hat content certificate management
 packages : package manipulation on repositories
 repo : repository listing and manipulation
 cds : CDS listing and manipulation
 migrate : Migrate from {RHUI3}
 haproxy : Load balancer listing and manipulation
 status : RHUI status and health information
 client : Red Hat client management

10.2.1. cert

Red Hat content certificate management

upload : uploads a new content certificate
 info : display information about the current content certificate

rhui-manager cert upload

upload: uploads a new content certificate
 --cert - full path to the new content certificate (required)
 --key - full path to the new content certificate's key

10.2.2. 软件包

package manipulation on repositories

list : lists all packages in a repository
 remove : removes a package from a custom repository
 upload : uploads a package or directory of packages to a custom repository
 remote : uploads RPM content from a remote URL to a custom repository

list: lists all packages in a repository
 --repo_id - id of the repository to list packages for (required)

remove: removes a package from a custom repository
 --repo_id - id of the custom repository to remove a package from (required)
 --package - name of the package to be removed (required)
 --vr - if specified, only the supplied version-release of the package will be removed
 --force - don't ask for confirmation

upload: uploads a package or directory of packages to a custom repository
 --repo_id - id of the custom repository where the packages will be uploaded (required)
 --packages - path to an .rpm file or directory of RPMs that will be uploaded (required)

remote: uploads RPM content from a remote URL to a custom repository
 --repo_id - id of the custom repository where the packages will be uploaded (required)
 --url - remote URL of the package or a web page that will be scraped for RPM content (required)

10.2.3. repo

repository listing and manipulation

list : lists all repositories in the RHUI
 info : displays information on an individual repo
 add : add a Red Hat repository to the RHUA
 add_by_repo: add Red Hat repositories to the RHUA via repo ID
 add_by_file: add Red Hat repositories to the RHUA using an input file
 add_errata: associate errata metadata with a repository
 delete : delete a repository
 sync : sync a repository
 export : export a repository to the filesystem
 enable_sync: enable scheduled synchronization of a repository
 disable_sync: disable scheduled synchronization of a repository
 sync_all : sync all repositories
 metadata : ensure metadata is generated for the latest version of repositories
 enable_autopublish: enable automatic publishing of a new repository version
 disable_autopublish: disable automatic publishing of a new repository version
 create_custom: create a custom repository
 unused : list of products available but not synced to the RHUA

info: displays information on an individual repo
 --repo_id - identifies the repository to display (required)

add: add a Red Hat repository to the RHUA
 --product_name - product to add the RHUA (required)

add_by_repo: add Red Hat repositories to the RHUA via repo ID
 --repo_ids - repo IDs to add, comma-separated (required)
 --sync-now - Use to sync any repos that are added (optional)

add_by_file: add Red Hat repositories to the RHUA using an input file
 --file - file containing repo IDs to add, one per line (required)
 --sync_now - Use to sync any repos that are added (optional)

add_errata: associate errata metadata with a repository
 --repo_id - repo ID to associate the metadata with (required)
 --updateinfo - updateinfo file to be applied (required)

delete: delete a repository
 --repo_id - identifies the repository to delete (required)

sync: sync a repository
 --repo_id - identifies the repository to sync (required)

export: export a repository to the filesystem
 --repo_id - identifies the repository to export (required)

metadata : ensure metadata is generated for the latest version of repositories
 --repo_id - explicit repo ID to generate metadata for

enable_sync: enable scheduled synchronization of a repository
 --repo_id - identifies the repository to enable scheduled synchronization for (required)
 --verbose - if present, info on last/next synchronization tasks will be displayed

`disable_sync`: disable scheduled synchronization of a repository
 --repo_id - identifies the repository to disable scheduled synchronization for (required)
 --verbose - if present, info on last/next synchronization tasks will be displayed

`enable_autopublish`: enable automatic publishing of a new repository version
 --repo_id - identifies the repository to enable automatic publishing for (required)

`disable_autopublish`: disable automatic publishing of a new repository version
 --repo_id - identifies the repository to disable automatic publishing for (required)

`create_custom`: create a custom repository
 --repo_id - identifies the repository to add (required)
 --path - path to the content being served by CDS; defaults to repo_id
 --display_name - display name for the custom repository
 --entitlement - path used in the entitlement certificate; may use yum variable substitutions
 --legacy_md - if present, the repo will use SHA1, otherwise default value is used (SHA256)
 --redhat_content - repository will host Red Hat GPG signed content
 --protected - make the content protected by entitlement certificate
 --gpg_public_keys - comma separated list of public keys used to sign the served content; the filenames must not contain comma

`unused`: list all unused Red Hat repositories
 Loading latest entitled products from Red Hat...
 ... listings loaded
 Available Repositories

10.2.4. cds

CDS listing and manipulation
 list : lists all cds instances in the RHUI
 add : register a cds instance to the RHUI
 reinstall : reinstalls an already registered cds instance
 delete : unregisters cds instances from the RHUI

`add`: register a cds instance to the RHUI
 --hostname - The hostname of the instance to add. (required)
 --ssh_user - Username with SSH access to the instance and sudo privileges. (required)
 --keyfile_path - Absolute path to an SSH private key to use with the given user. (required)
 --hostfile - Absolute path to a known_hosts file to use to determine the identity of the instance; if this is not provided and the instance hostkey is not in the system-wide known_hosts file, this command will fail.
 --user_supplied_ssl_key - Optional absolute path to the user supplied SSL key file.
 --user_supplied_ssl_cert - Optional absolute path to the user supplied SSL crt file.
 --force - Add the system even if the hostname is already registered.
 --unsafe - Proceed even if the instance host key is not in the known_hosts file. This is not secure!

`reinstall`: reinstalls an already registered cds instance
 --hostname - The hostname of the instance to reinstall on; this instance must be registered already.
 --all - Reinstall all the registered instances.

delete: unregisters cds instances from the RHUI
 --force - Delete the system, even if it is the last of its kind.
 --hostnames - Comma-separated list of hostnames to delete (unregister) from RHUI. (required)

10.2.5. migrate

Migrate from {RHUI3}
 --hostname : the {RHUI3} hostname
 --password : <your_password>
 --keyfile_path : ~/.ssh/id_rsa_rhua*

10.2.6. hapoxy

Load balancer listing and manipulation
 list : lists all haproxy instances in the RHUI
 add : register a haproxy instance to the RHUI
 reinstall : reinstalls an already registered haproxy instance
 delete : unregisters haproxy instances from the RHUI

add: register a haproxy instance to the RHUI
 --hostname - The hostname of the instance to add. (required)
 --ssh_user - Username with SSH access to the instance and sudo privileges. (required)
 --keyfile_path - Absolute path to an SSH private key to use with the given user. (required)
 --hostfile - Absolute path to a known_hosts file to use to determine the identity of the instance; if this is not provided and the instance hostkey is not in the system-wide known_hosts file, this command will fail.
 --config - Optional absolute path to a user supplied HAProxy config file.
 --force - Add the system even if the hostname is already registered.
 --unsafe - Proceed even if the instance host key is not in the known_hosts file. This is not secure!

reinstall: reinstalls an already registered haproxy instance
 --hostname - The hostname of the instance to reinstall on; this instance must be registered already.
 --all - Reinstall all the registered instances.

delete: unregisters haproxy instances from the RHUI
 --force - Delete the system, even if it is the last of its kind.
 --hostnames - Comma-separated list of hostnames to delete (unregister) from RHUI. (required)

10.2.7. status

status: RHUI status and health information
 --code - if specified, only a numeric code for the result will be displayed
 --repo_json - Name of the JSON file for a repo status

10.2.8. 客户端

Red Hat client management
 labels : list the labels required for client certificate creation
 cert : create a content certificate for a rhui client

rpm : create a client config rpm
 content_source: create an alternate source config rpm
 acs_config: output a JSON representation of the alternate source config

cert: create a content certificate for a rhui client
 --repo_label - identifies the repositories to add. Comma delimited string of repo labels (required)
 --name - identifies the certificate name (required)
 --days - number of days cert will be valid (required)
 --dir - directory where the certificate will be stored (required)

rpm: create a client config rpm
 --private_key - entitlement private key
 --entitlement_cert - entitlement certificate
 --rpm_version - version number of the client config rpm
 --rpm_release - release of rpm package. Default is 1
 --rpm_name - name of the client config rpm (required)
 --dir - directory where the rpm will be created (required)
 --unprotected_repos - comma-separated list of unprotected repos to include
 --cert - generate certificate also before building client config rpm if given
 --ca_cert - full path to the certificate authority of CDS servers
 --repo_label - identifies the repositories to add. Comma delimited string of repo labels
 --name - identifies the certificate name if it is different from rpm name
 --days - number of days cert will be valid
 --proxy - url/string in case proxy option is necessary in yum repo file

content_source: create an alternate source config rpm
 --private_key - entitlement private key
 --entitlement_cert - entitlement certificate
 --rpm_version - version number of the client config rpm
 --rpm_name - name of the client config rpm (required)
 --dir - directory where the rpm will be created (required)
 --unprotected_repos - comma-separated list of unprotected repos to include
 --cert - generate certificate also before building client config rpm if given
 --ca_cert - full path to the certificate authority of CDS servers
 --repo_label - identifies the repositories to add. Comma delimited string of repo labels
 --name - identifies the certificate name if it is different from rpm name
 --days - number of days cert will be valid

acs_config: output a JSON representation of the alternate source config
 --dir - directory where the JSON representation will be stored (required)
 --private_key - entitlement private key
 --entitlement_cert - entitlement certificate
 --cert - generate certificate based on the the repos supplied via --repo_label
 --ssl_ca_cert - full path to the certificate authority of CDS servers (defaults to ssl_ca_cert specified in rhui-tools.conf)
 --repo_label - identifies the repositories to add. Comma delimited string of repo labels
 --days - number of days cert will be valid if new cert is generated

第 11 章 认证云和服务提供商认证 workflow

认证云供应商协议要求红帽认证用于创建租户实例的镜像(templates)以确保最终客户完全支持的配置。

可以通过两种方法为 Red Hat Enterprise Linux 认证镜像。首选方法是使用认证云和服务提供商(CCSP)镜像认证 workflow。

在经过红帽审核后，将安排通过/失败并向 [红帽生态系统目录](#) 发布通过认证网站。

11.1. 其他资源

- [Red Hat Certified Cloud and Service Provider Certification Workflow Guide](#)
- [Red Hat Certified Cloud and Service Provider Certification 7.34 产品文档](#)

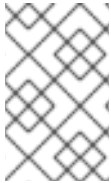
第 12 章 备份和恢复红帽更新基础架构

安装和配置 Red Hat Update Infrastructure (RHUI) 服务器后，您可能需要备份它们。如果您遇到任何问题或没有正确配置 RHUI，则备份 RHUI 非常有用。在这种情况下，您可以通过恢复 RHUI 来返回到以前的工作配置。

要成功备份 RHUI，您必须备份所有 RHUA、CDS 和 HAProxy 节点。

12.1. 备份红帽更新设备

要备份红帽更新设备，您必须备份所有相关文件和存储。



注意

要备份 RHUA，您必须停止相关的服务。但是，停止服务不会禁用任何客户端实例来更新或安装软件包，因为客户端仅连接到内容交付网络(CDS)。因此，如果您有一个自动监控解决方案，您的监控在备份过程中可能会失败。

流程

1. 停止 `pulp-server` 服务：

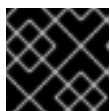
```
# systemctl stop pulpcore-api pulpcore-content pulpcore-worker*
```

2. 验证服务是否已停止：

```
# systemctl status pulpcore-api pulpcore-content pulpcore-worker*
```

3. 备份以下文件：

```
# cp -a <source_files_path> <destination_files_path>
```



重要

确保文件在备份时保留其当前属性。

文件列表：

- `/etc/pki/rhui/*`
- `/etc/pulp/*`
- `/etc/rhui/*`
- `/etc/rhui/rhui-tools.conf`
- `/etc/nginx/*`
- `/root/.rhui/*`
- `/var/log/rhui/*`
- `/var/log/rhui-subscription-sync.log*`

- 可选：/var/lib/rhui.0/24



注意

备份此目录备份所有下载的内容，该内容可能是大量数据。

4. 备份任何生成的客户端授权证书和客户端配置 RPM。
5. 重启 RHUI 服务

```
# rhui-services-restart
```

12.2. 恢复红帽更新设备

要恢复 RHUA，您必须创建新的 RHUA 节点，并将关联的文件替换为备份的版本。

流程

1. 创建新的 RHUA 节点。如需更多信息，[请参阅设置 RHUA 节点。](#)
2. 停止 **pulp-server** 服务：

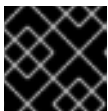
```
# systemctl stop pulpcore-api pulpcore-content pulpcore-worker*
```

3. 验证服务是否已停止：

```
# systemctl status pulpcore-api pulpcore-content pulpcore-worker*
```

4. 恢复以下文件：

```
# cp -a <source_files_path> <destination_files_path>
```



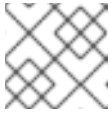
重要

确保文件在恢复时保留其当前属性。

文件列表：

- /etc/pki/rhui/*
- /etc/pulp/*
- /etc/rhui/*
- /etc/rhui/rhui-tools.conf
- /etc/nginx/*
- /root/.rhui/*
- /var/log/rhui/*
- /var/log/rhui-subscription-sync.log*

- 可选：/var/lib/rhui.0/24



注意

恢复此目录恢复所有下载的内容，该内容可能是大量数据。

5. 恢复任何生成的客户端授权证书和客户端配置 RPM。
6. 重启 RHUI 服务

```
# rhui-services-restart
```

12.3. 备份内容交付服务器

要备份 CDS，您必须备份所有相关文件和存储。



注意

为了避免服务完全丢失，请一次备份单个 CDS 节点。客户端将自动切换到其他正在运行的 CDS 节点。

流程

1. 停止 **nginx** 服务：

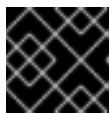
```
# systemctl stop nginx
```

2. 验证 **nginx** 服务是否已停止：

```
# systemctl status nginx
```

3. 备份以下文件：

```
# cp -a <source_files_path> <destination_files_path>
```



重要

确保文件在备份时保留其当前属性。

文件列表：

- /etc/nginx/*
 - /var/log/nginx/*
 - /etc/pki/rhui/*
4. 重启 RHUI 服务。

```
# rhui-services-restart
```

12.4. 恢复内容交付服务器

要恢复内容交付服务器，您必须创建新的 CDS 节点，并将关联的文件替换为备份的版本。

流程

1. 创建新的 CDS 节点。如需更多信息，[请参阅设置 CDS 节点](#)。

2. 停止 `nginx` 服务：

```
# systemctl stop nginx
```

3. 验证 `nginx` 服务是否已停止：

```
# systemctl status nginx
```

4. 恢复以下文件：

```
# cp -a <source_files_path> <destination_files_path>
```



重要

确保文件在恢复时保留其当前属性。

文件列表：

- `/etc/nginx/*`
 - `/var/log/nginx/*`
 - `/etc/pki/rhui/*`
5. 重启 RHUI 服务。

```
# rhui-services-restart
```

12.5. 备份 HAPROXY 服务器

要备份 HAProxy 服务器，您必须备份所有关联的文件和存储。

流程

1. 备份 `/etc/haproxy/haproxy.cfg` 文件。

```
# cp -a <source_files_path> <destination_files_path>
```



重要

确保文件在备份时保留其当前属性。

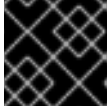
12.6. 恢复 HAPROXY 服务器

要恢复 HAProxy 服务器，您必须创建新的 HAProxy 节点，并将关联的文件替换为备份的版本。

流程

1. 创建新的 HAProxy 节点。如需更多信息，[请参阅设置 HAProxy 节点](#)。
2. 恢复 `/etc/haproxy/haproxy.cfg` 文件。

```
# cp -a <source_files_path> <destination_files_path>
```



重要

确保文件在恢复时保留其当前属性。

第 13 章 解决 RHUI 4 中的常见问题

下表列出了 Red Hat Update Infrastructure 中已知的问题。如果您遇到任何问题，请通过 Bugzilla 报告问题。

表 13.1. Red Hat Update Infrastructure 中的常见问题

事件	已知问题的描述	建议
安装和配置	您体验 RHUA 和 CDS 之间的通信问题。	<p>验证为 RHUA 和 CDS 设置完全限定域名(FQDN)，并可解析。</p> <p>正确配置 HTTP 代理，如 Bug 726420 - 在代理 URL 上快速备注</p>
同步	您无法将软件仓库与红帽同步。	<p>验证 RHUI SKUs 存在于您的帐户中。</p> <p>验证正确的内容证书已加载到 RHUA。</p> <p>查找临时 CDN 问题。</p> <p>在环境中查找任何 HTTP 代理，并确保您没有将 While syncing repository to RHUA 达到 RHUA，它会失败并显示 "RepoError: Cannot retrieve repository metadata(repomd.xml)for repository metadata(repomd.xml)。请验证其路径并在代理使用 错误时重试"。</p> <p>RHUA 无法与 CDS 同步，通常是因为过期的 qpid 证书而失败：请参阅 CDS 同步失败，并显示错误 "ssl3 警报证书过期"，因为 RHUI 上的 qpid CA 证书已到期。</p>

事件	已知问题的描述	建议
<p>Red Hat Update Appliance/Content Delivery Network Communication</p>	<p>Red Hat Update Appliance 没有与内容交付网络通信。</p>	<p>使用 /etc/pki/rhui/redhat (.pem 文件) 中的内容证书测试 RHUA 和 CDN 之间的连接和访问。</p> <p>evince CD /etc/pki/rhui/redhatwget --certificate=8a85f98146a087b80146afacb3362499.pem --ca-certificate=/etc/rhsm/ca/redhat-uep.pem</p> <p>https://cdn.redhat.com/content/dist/rhel/rhui/server/6/6Server/x86_64/os/repodata/repomd.xml</p> <p>curl(1)man page 中的备注：如果 NSS PEM PKCS#11 模块(lib-nsspem.so)可用，则可以加载 PEM 文件。如果您要使用当前目录中的文件，请在它前面带有 ./ 前缀，以避免与 nickname 混淆。</p> <p>在每个 CDS 中，/etc/pki/pulp/content 中的授权证书可以用来使用 # curl --cert ./rhui-ec2-20120619.pem 来测试 RHUA 内容的可用性。</p> <p>在 RHUA 上托管的存储库的 URL 始终以 https://fqdn/pulp/content 开头。您可以通过以下方式减少剩余的 URL：-Looking at RHUA 中的 /var/lib/rhui/remote_share/sym links/pulp/content/ - 直接使用 openssl 命令扩展内容证书，因为 OID 以 1.6 结尾</p>

事件	已知问题的描述	建议
客户端/内容交付服务器通信	Curl 也可用于验证与内容交付服务器节点的客户端通信。	<pre># curl --cert /etc/pki/entitlement/product/content.crt --key /etc/pki/entitlement/key.pem https://ip-10-4-58-34.ec2.internal/pulp/repos/content/dist/rhel/rhui/server/6/6Server/x86_64 /rhui/2.1/os/repodata/repomd.xml -k <?xml version="1.0" encoding="UTF-8"?> <repomd xmlns="http://linux.duke.edu/metadata/repo" xmlns :RPM="http://linux.duke.edu/metadata/rpm"> <revision>1339940325</revision> <data type="other_db"> <location href="repodata/4f86b0ae203bba90d22a8363120c66ed6f37da81- other.sqlite.bz2"/> <checksum type="sha" ">4f86b0ae203bba90d22a8363120c66ed6f37da81</checksum> <timestamp>1339940328.43</timestamp></pre>
客户端/HAProxy 通信	所有 HAProxy 节点都关闭。客户端无法访问 RHUI 软件仓库。	添加并配置至少一个新 HAProxy 节点。如果出于某种原因而无法这样做，请临时更改 DNS 配置，以便该指南中的主负载均衡器主机名 (cds.example.com) 解析到其中一个 CDS 节点的 IP 地址。这将允许客户端避免不可用的 HAProxy 节点，并直接与 CDS 进行通信。