Red Hat Enterprise Linux 7

Linux 域身份、身份验证和策略指南

在 Linux 环境中使用红帽身份管理

Last Updated: 2024-07-18

Red Hat Enterprise Linux 7 Linux 域身份、身份验证和策略指南

在 Linux 环境中使用红帽身份管理

Florian Delehaye
Red Hat Customer Content Services
fdelehay@redhat.com

Marc Muehlfeld
Red Hat Customer Content Services

Filip Hanzelka
Red Hat Customer Content Services

Lucie Maňásková
Red Hat Customer Content Services

Aneta Šteflová Petrová
Red Hat Customer Content Services

Tomáš Čapek
Red Hat Customer Content Services

Ella Deon Ballard
Red Hat Customer Content Services

法律通告

Keywords

1. FreeIPA. 2. Identity Management. 3. IdM. 4. IPA.

摘要

用户和机器的身份和策略管理是大部分企业环境的核心功能。身份管理提供了一种创建身份域的方法，允许机器注册域，并立即访问单点登录和身份验证服务所需的身份信息，以及管理授权和访问权限的策略设置。 除了本指南外，您还可以在以下指南中找到有关 Red Hat Enterprise Linux Identity Management 的其他功能和服务的文档： 系统级身份验证指南 记录了用于在本地系统上配置身份验

证的不同应用程序和服务，包括 authconfig 实用程序、系统安全服务守护进程(SSSD)服务、可插拔验证模块(PAM)框架、Kerberos、certmonger 实用程序和单点登录(SSO)以用于应用程序。
Windows 集成指南 文档如何使用身份管理将 Linux 域与 Microsoft Windows Active Directory (AD)集成。除其他主题外，该指南涵盖了直接和间接 AD 集成的各个方面，使用 SSSD 访问通用 Internet 文件系统(CIFS)和域系统。

目录

部分 I. 红帽身份管理概述

这部分解释了 Red Hat **Identity Management** 的目的。它还提供有关 **身份管理域** 的基本信息，包括属于域的客户端和服务器计算机。

第 1 章 红帽身份管理简介

1.1. 红帽身份管理的目标

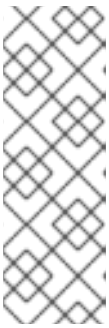
红帽身份管理(IdM)为在基于 Linux 的域中管理身份存储、身份验证、策略和授权策略提供集中和统一的方式。IdM 可显著降低单独管理不同服务以及在不同机器上使用不同工具的管理开销。

IdM 是一个用于中央化身份、策略和授权的软件解决方案，它支持：

- Linux 操作系统环境的高级特性
- 统一大型的 Linux 机器组
- 与 Active Directory 的原生集成

IdM 创建一个基于 Linux 并由 Linux 控制的域：

- IdM 基于现有的原生 Linux 工具和协议构建。它有自己的进程和配置，但其底层的技术已在 Linux 系统中广泛使用，并被 Linux 管理员信任。
- IdM 服务器和客户端是 Red Hat Enterprise Linux 机器。但是，即使 IdM 不支持 Windows 客户端，它也允许与 Active Directory 环境集成。



注意

本指南描述了仅在 Linux 环境中使用 IdM。有关与 Active Directory 集成的更多信息，请参阅《*Windows 集成指南*》。

有关 Samba 套件的详情，允许将 Linux 计算机集成到 Active Directory 环境中，请参阅 *Windows 集成指南*中的 *使用 Samba 进行 Active Directory 集成* 章节。如果您使用 Samba 作为服务器，请注意，将服务器集成到 IdM 域中，并验证连接到 IdM 或可信 Active Directory 域的用户。

1.1.1. IdM 的优势分支示例

使用几个 Linux 服务器管理身份和策略

*没有 IdM:*每个服务器都会单独管理。所有密码都保存在本地机器上。IT 管理员管理每台计算机上的用户，单独设置身份验证和授权策略，并且维护本地密码。

*使用 IdM：*IT 管理员可以：

- 在一个中央位置管理用户的身份：IdM 服务器
- 同时对多个机器统一应用策略
- 使用基于主机的访问控制、委托和其他规则为用户设置不同的访问级别
- 集中管理权限升级规则
- 定义如何挂载主目录

企业单点登录

*没有 IdM:*用户登录系统，每次访问服务或应用时都会提示输入密码。这些密码可能有所不同，用户必须记住使用哪个凭证。

*使用 IdM：*用户登录系统后，他们可以访问多个服务和应用程序，而无需重复请求其凭据。它包括：

- 提高可用性
- 降低以不安全方式写入或保存密码的安全风险
- 提高用户的生产率

管理一个混合了 Linux 和 Windows 的环境

*没有 IdM:*Windows 系统在 Active Directory 林中管理，但开发、生产和其他团队具有许多 Linux 系统。Linux 系统不包含在 Active Directory 环境中。

*使用 IdM：*IT 管理员可以：

- 使用原生 Linux 工具管理 Linux 系统
- 将 Linux 系统与 Windows 系统集成，从而保留集中式用户存储
- 轻松扩展 Linux 基础
- 单独管理 Linux 和 Active Directory 机器，使 Linux 和 Windows 管理员能够直接控制其环境

1.1.2. 将身份管理与标准 LDAP 目录进行比较

标准 LDAP 目录（如红帽目录服务器）是一个通用用途目录：可以自定义，以适应广泛的用例。

- Schema：一种可针对大量条目（如用户、计算机、网络实体、物理设备或设施）自定义的灵活方案。
- 通常用作：用于存储其他应用的数据的后端目录，如在 Internet 上提供服务的业务应用程序。

身份管理(IdM)具有特定目的：管理身份以及与这些身份相关的身份验证和授权策略。

- Schema：定义一组与其目的相关的特定条目的特定架构，如用于用户身份或机器身份的条目。
- 通常，身份和验证服务器用于在企业或项目边界内管理身份。

红帽目录服务器和 IdM 的底层目录服务器技术是相同的。但是，IdM 被优化来管理身份。这限制了其一般的可扩展性，但也带来了一些好处：更简单的配置、更好的资源管理自动化和更高的管理身份的效率。

其它资源

- 红帽企业 *Linux 博客上的身份管理或红帽目录服务器 - 应该使用一个 目录服务器？*

1.2. IDENTITY MANAGEMENT DOMAIN

身份管理(IdM)域由共享相同配置、策略和身份存储的一组计算机组成。共享属性允许域中的计算机互相了解并共同操作。

从 IdM 的角度来看，域包括以下类型的机器：

- 作为域控制器的 IdM 服务器
- IdM 客户端，它们注册到服务器中

IdM 服务器也是注册到其自身的 IdM 客户端：服务器计算机提供与客户端相同的功能。

IdM 支持 Red Hat Enterprise Linux 机器作为 IdM 服务器和客户端。

注意

本指南描述了在 Linux 环境中使用 IdM。有关与 Active Directory 集成的更多信息，请参阅《*Windows 集成指南*》。

1.2.1. 身份管理服务器

IdM 服务器充当身份和策略信息的中央存储库。它们也托管域成员使用的服务。IdM 提供了一组管理工具来集中管理所有 IdM 关联服务：IdM Web UI 和命令行工具。

有关安装 IdM 服务器的详情请参考 第 2 章 *安装和卸载身份管理服务器*。

为了支持冗余和负载平衡，数据和配置可以从 IdM 服务器复制到另一个 IdM 服务器：初始服务器 *的副本*。您可以配置服务器及其副本，以便为客户端提供不同的服务。有关 IdM 副本的详情，请参考 第 4 章 *安装和卸载身份管理副本*。

1.2.1.1. IdM 服务器托管的服务

以下大多数服务并没严格要求安装到 IdM 服务器上。例如，可以在 IdM 域外的外部服务器上安装证书认证机构(CA)、DNS 服务器或网络时间协议(NTP)服务器等服务。

Kerberos: krb5kdc 和 kadmin

IdM 使用 Kerberos 协议来支持单点登录。使用 Kerberos，用户只需提供一次正确的用户名和密码，就可以访问 IdM 服务，而系统不需要再次提示输入凭证。

- *Kerberos* 分为两个部分：
  - **krb5kdc** 服务是 Kerberos 身份验证服务和密钥分发中心(KDC)守护进程。
  - **kadmin** 服务是 Kerberos 数据库管理程序。

  有关 Kerberos 的工作原理，请参阅 *系统级身份验证指南中的* 使用 Kerberos。
- 有关如何在 IdM 中使用 Kerberos 进行身份验证的详情请参考 第 5.2 节 “使用 Kerberos 登录 IdM”。
- 有关在 IdM 中管理 Kerberos 的详情请参考 第 29 章 *管理 Kerberos 域*。

LDAP 目录服务器：dirsrv

IdM 内部 *LDAP 目录服务器实例* 存储所有 IdM 信息，如与 Kerberos、用户帐户、主机条目、服务、策略等相关的信息。

LDAP 目录服务器实例基于 与红帽目录服务器 相同的技术。但是，它被调优为特定于 IdM 的任务。

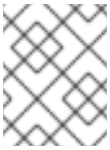
注意

本指南将这个组件称为 Directory Server。

证书颁发机构：pki-tomcatd

集成的 *证书颁发机构(CA)* 基于与 Red Hat Certificate System 相同的技术。**pki** 是用于访问证书系统服务的命令行界面。

- 有关使用不同 CA 配置安装 IdM 服务器的详情，请参考 第 2.3.2 节 "确定要使用的 CA 配置"。



注意

本指南在处理实施过程和证书认证机构时将此组件指代为证书系统，在处理实施提供的服务时作为认证机构。

有关红帽认证系统（独立红帽产品）的信息，请参阅红帽认证系统的 产品文档。

域名系统(DNS)： 命名

IdM 使用 *DNS* 进行动态服务发现。IdM 客户端安装工具可使用 DNS 的信息来自动配置客户端机器。客户端注册到 IdM 域后，它使用 DNS 来定位域中的 IdM 服务器和服务。

Red Hat Enterprise Linux 中的 DNS (**域名系统)协议的 BIND (Berkeley 互联网** 名称域)实现包括 **命名的** DNS 服务器。**named-pkcs11** 是构建了对 PKCS the 加密标准的原生支持的 **BIND** DNS 服务器版本。

- 如需有关服务发现的更多信息，请参阅 *系统级身份验证 指南中的 配置 DNS 服务发现*。
- 有关 DNS 服务器的更多信息，请参阅 *Red Hat Enterprise Linux 网络指南中的* BIND。
- 有关在 IdM 中使用 DNS 和重要先决条件的详情请参考 第 2.1.5 节 "主机名和 DNS 配置"。
- 有关使用或没有集成 DNS 安装 IdM 服务器的详情，请参考 第 2.3.1 节 "确定使用集成 DNS"。

网络时间协议： ntpd

许多服务要求服务器和客户端在特定差异内具有相同的系统时间。例如，Kerberos 票据使用时间戳来确定其有效期并防止重播攻击。如果服务器和客户端之间的时间偏移超出允许范围，Kerberos 票据将无效。

默认情况下，IdM 使用 *网络时间协议(NTP)* 通过 **ntpd** 服务通过网络同步时钟。使用 NTP 时，中央服务器充当权威时钟，客户端会同步时间以匹配服务器时钟。IdM 服务器在服务器安装过程中配置为 IdM 域的 NTP 服务器。

注意

在虚拟机上安装的 IdM 服务器中运行 NTP 服务器可能会导致某些环境中的时间同步不准确。为避免潜在的问题，不要在虚拟机上安装的 IdM 服务器中运行 NTP。有关虚拟机上 NTP 服务器可靠性的更多信息，请参阅 此知识库解决方案。

Apache HTTP 服务器： httpd

Apache HTTP Web 服务器 提供了 IdM Web UI，还管理证书颁发机构和其他 IdM 服务之间的通信。

- 如需更多信息，请参阅 *系统管理员指南中的* Apache HTTP 服务器。

Samba/ Winbind： smb、winbind

Samba 在红帽企业 Linux 中实施服务器消息块(SMB)协议，也称为通用 Internet 文件系统(CIFS)协议。通过 **smb** 服务，SMB 协议可让您访问服务器上的资源，如文件共享和共享打印机。如果您使用 Active Directory(AD)环境配置了信任，*Winbind* 服务将管理 IdM 服务器和 AD 服务器之间的通信。

- 如需更多信息，请参阅 *系统管理员指南中的* Samba。
- 如需更多信息，请参阅 *系统级身份验证指南中的* Winbind

一次性密码(OTP)身份验证： ipa-otpd

一次性密码(OTP) 是仅由一个会话的身份验证令牌生成的密码，作为双因素身份验证的一部分。OTP 身份验证在 Red Hat Enterprise Linux 中是通过 **ipa-otpd** 服务实现的。

- 有关 OTP 身份验证的详情请参考 第 22.3 节 “一次性密码”。

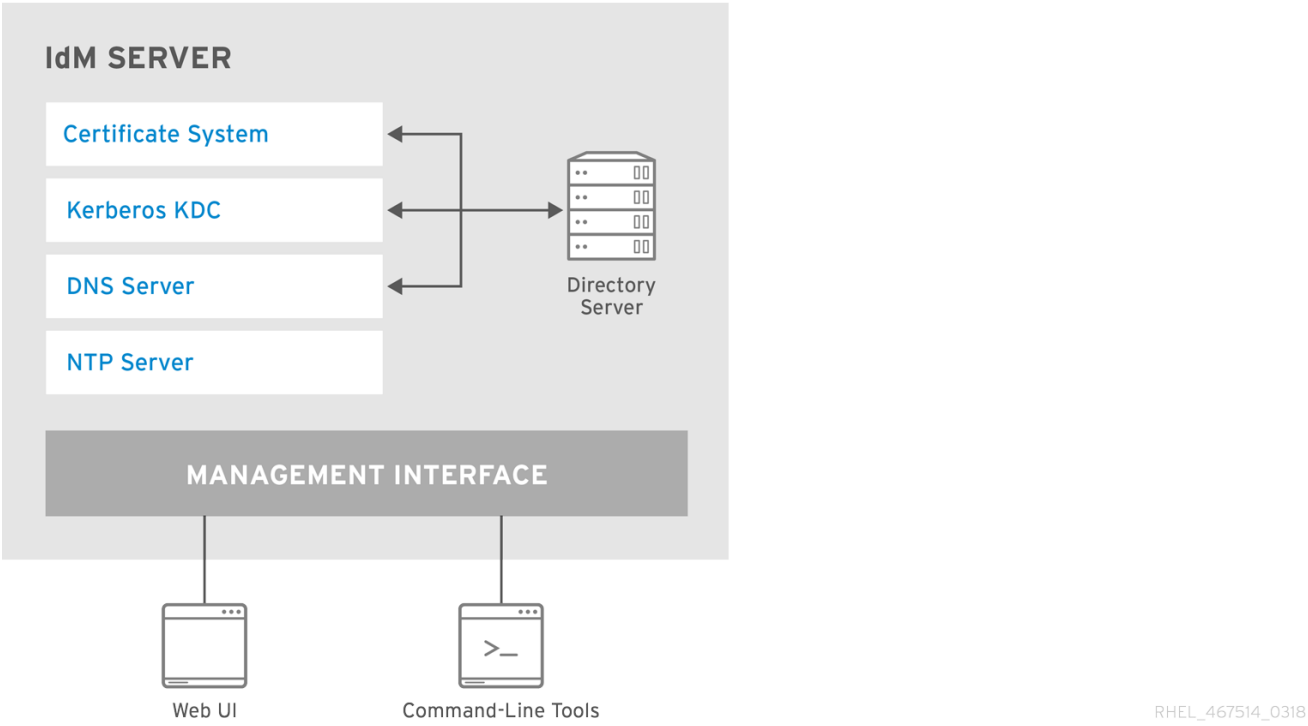
custodia: ipa-custodia

custodia 是 Secrets 服务提供商，它存储和共享对密码、密钥、令牌和证书等机密资料的访问。

OpenDNSSEC: ipa-dnskeysyncd

OpenDNSSEC 是一个 DNS 管理器，自动化了跟踪 DNS 安全扩展(DNSSEC)密钥和区域签名的过程。**ipa-dnskeysyncd** servuce 管理 IdM 目录服务器和 OpenDNSSEC 之间的同步。

图 1.1. **身份管理服务器：统一服务**



1.2.2. 身份管理客户端

IdM 客户端是配置为在 IdM 域中运行的机器。它们与 IdM 服务器交互以访问域资源。例如，它们属于服务器上配置的 Kerberos 域，接收服务器发布的证书和票据，并使用其他集中式服务进行身份验证和授权。

IdM 客户端不需要专用的客户端软件作为域的一部分进行交互。它只需要正确配置某些服务和库，如 Kerberos 或 DNS。此配置指示客户端机器使用 IdM 服务。

有关安装 IdM 客户端的详情请参考 第 3 章 *安装和卸载身份管理客户端*。

1.2.2.1. IdM 客户端托管的服务

系统安全服务守护进程： sssd

系统安全服务守护进程(SSSD) 是管理用户身份验证和缓存凭证的客户端应用。

缓存可让本地系统在 IdM 服务器不可用或客户端离线时能够继续正常的身份验证操作。

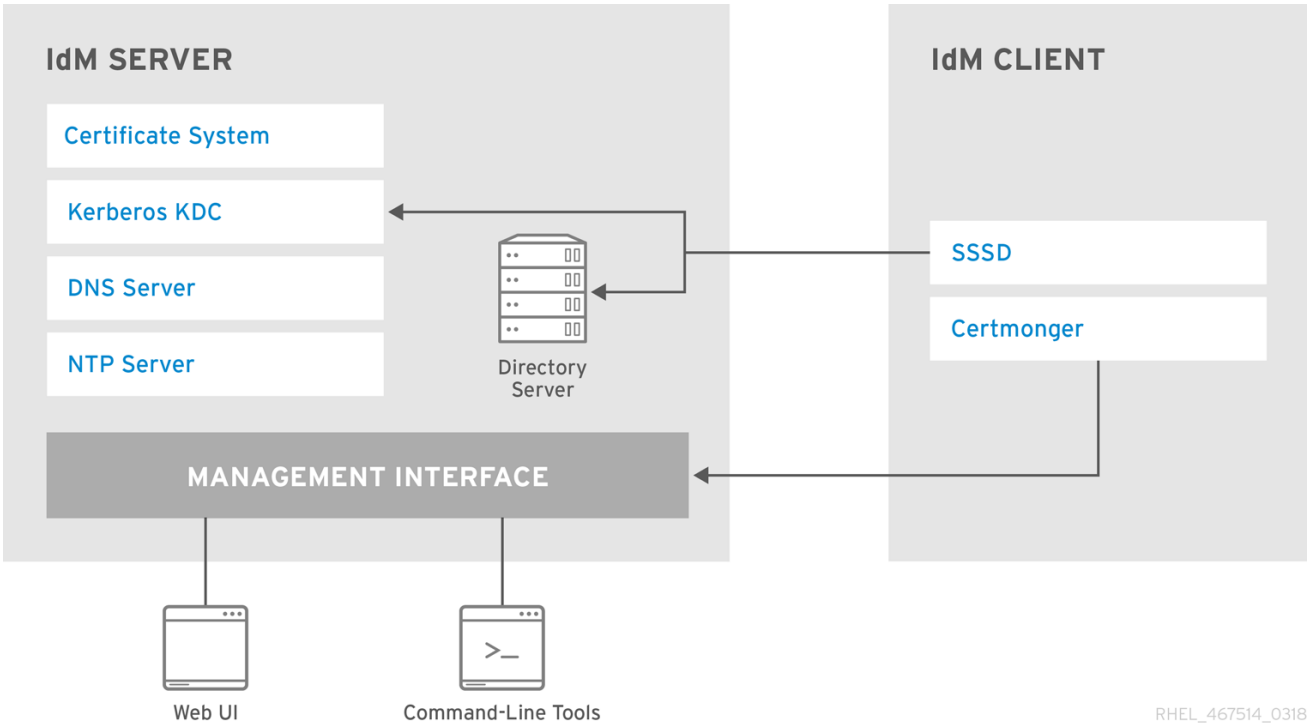
如需更多信息，请参阅 *系统级身份验证指南中的配置 SSSD*。SSSD 还支持 Windows Active Directory(AD)。有关在 AD 中使用 SSSD 的更多信息，请参阅 *Windows 集成指南中的* 将 Active Directory 用作 SSSD 的身份提供程序。

certmonger

certmonger 服务监控并更新客户端上的证书。它可以为系统上的服务请求新的证书。

如需更多信息，请参阅 *系统级身份验证指南中的* 使用证书。

图 1.2. IdM 服务间的交互



部分 II. 安装身份管理

本节解释了如何规划 **身份管理** 部署以及如何安装 **身份管理服务器**、客户端和副本。

第 2 章 安装和卸载身份管理服务器

身份管理 (IdM) *服务器* 是一个域控制器：它定义和管理 IdM 域。要设置 IdM 服务器，您必须：

1. 安装所需的软件包
2. 使用设置脚本配置机器

红帽强烈建议在您的域中设置多个域控制器，以实现负载平衡和冗余。这些额外的服务器是初始 master IdM 服务器 *的副本*。

本章论述了安装第一个初始 IdM 服务器。有关从初始服务器安装副本的详情请参考 第 4 章 *安装和卸载身份管理副本*。

2.1. 安装服务器的先决条件

2.1.1. 最低硬件要求

要运行身份管理(IdM)，服务器至少需要以下硬件配置：

- 1（虚拟）CPU 内核
- 2 GB RAM

  即使您可以安装较少的 RAM 的 IdM，某些操作（如更新 IdM）至少需要 4 GB RAM。
- 10 GB 硬盘

重要

根据数据库中存储的数据量，IdM 需要更多资源，特别是更多 RAM。详情请查看 第 2.1.2 节 “硬件建议”。所需的硬件资源还依赖于其他因素，如服务器的生产工作负载或者配置了与 Active Directory 的信任。

2.1.2. 硬件建议

对于性能调整，RAM 是最重要的硬件。要确定您需要的 RAM 量，请考虑以下建议：

- 对于 10,000 用户和 100 组：至少 3 GB RAM 和 1 GB 交换空间
- 对于 100,000 个用户和 50,000 个组：至少 16 GB RAM 和 4 GB swap 空间

注意

基本用户条目或具有证书的简单主机条目的大小约为 5 - 10 KiB。

对于较大的部署，增加 RAM 比增加磁盘空间更为有效，因为许多数据都存储在缓存中。

要提高性能，您可以调优底层目录服务器以提高性能。详情请查看 *红帽目录服务器性能调优指南*。

2.1.3. 系统要求

Red Hat Enterprise Linux 7 支持身份管理。在干净的系统中安装 IdM 服务器，而无需为 DNS、Kerberos 或 Directory Server 等服务配置任何自定义配置。

重要

出于性能和稳定性的原因，红帽建议您不要在 IdM 服务器上安装其他应用程序或服务。例如，IdM 服务器可能会对系统完成，特别是当 LDAP 对象的数量很高时。另外，IdM 会在系统中集成，如果第三方应用程序更改了 IdM 依赖于的配置文件，IdM 可能会中断。

IdM 服务器安装覆盖了系统文件来设置 IdM 域。IdM 将原始系统文件备份到 **/var/lib/ipa/sysrestore/**。

名称服务缓存守护进程(NSCD)要求

红帽建议在身份管理机器上禁用 NSCD。另外，如果没有禁用 NSCD，则只为 SSSD 不缓存的映射启用 NSCD。

NSCD 和 SSSD 服务都执行缓存，当系统同时同时使用这两个服务时，可能会出现问题。有关如何避免 NSCD 和 SSSD 之间冲突的信息，请参阅 系统级身份验证指南。

必须在系统中启用 IPv6

IdM 服务器必须在内核中启用 IPv6 协议。请注意，Red Hat Enterprise Linux 7 系统中默认启用 IPv6。

如果您之前禁用 IPv6，请按照红帽知识库中如何在 Red Hat Enterprise Linux 中禁用或启用 IPv6 协议所述重新启用。

注意

IdM 不需要在您要注册为客户端的主机的内核中启用 IPv6 协议。例如，如果您的内部网络仅使用 IPv4 协议，您可以将系统安全服务守护进程(SSSD)配置为仅使用 IPv4 与 IdM 服务器通信。您可以将以下行插入到 **/etc/sssd/sssd.conf** 文件的 **[domain/_NAME_]** 部分中：

```
lookup_family_order = ipv4_only
```

有关 **lookup_family_order** 的更多信息，请参阅 **sssd.conf (5)** 手册页。

2.1.4. 在 FIPS 环境中安装服务器的先决条件

在使用 Red Hat Enterprise Linux 7.4 及更新的版本设置的环境中：

- 您可以在启用了联邦信息处理标准(FIPS)模式的系统中配置新的 IdM 服务器或副本。安装脚本自动检测启用了 FIPS 的系统，并在没有管理员干预的情况下配置 IdM。

  要在操作系统中启用 FIPS，请参阅 *安全指南* 中的启用 FIPS 模式。

重要

您不能：

  - 在之前禁用 FIPS 模式的现有 IdM 服务器上启用 FIPS 模式。
  - 当使用禁用 FIPS 模式的现有 IdM 服务器时，以 FIPS 模式安装副本。

在使用 Red Hat Enterprise Linux 7.3 及更早版本设置的环境中：

- IdM 不支持 FIPS 模式。在安装 IdM 服务器或副本前禁用您的系统 FIPS，并在安装后不要启用它。

有关 FIPS 模式的详情，请参阅 *安全指南* 中的 联邦信息处理标准(FIPS)。

2.1.5. 主机名和 DNS 配置

警告

请非常小心，并确保：

- 您有一个经过测试和功能的 DNS 服务可用
- 服务配置正确

此要求适用于具有集成 DNS 服务的 IdM 服务器以及在没有 DNS 的情况下安装的 IdM 服务器。DNS 记录对于几乎所有 IdM 域功能至关重要，包括运行 LDAP 目录服务、Kerberos 和 Active Directory 集成。

请注意，在安装后无法更改主 DNS 域和 Kerberos 域。

不要使用单标签域名，例如 **.company**：IdM 域必须由一个或多个子域和一个顶级域组成，如 **example.com** 或 **company.example.com** 。

服务器主机必须正确配置 DNS，无论 DNS 服务器是在 IdM 中集成还是外部托管。

身份管理需要将单独的 DNS 域用于服务记录。为了避免 DNS 级别（*主 IdM DNS 域*）冲突，名称为 IdM Kerberos 名称小写版本的 DNS 域无法与其他系统（如其它 IdM 或 AD 域）共享

主 IdM DNS 域必须包含其自身用于标准 IdM 服务的 SRV 记录。所需的记录有：

- _kerberos._tcp.*domain_name* *和* *_kerberos._* udp.*domain_name*的 SRV 记录
- _ldap._tcp.*domain_name*的 SRV 记录
- _kerberos.*domain_name*的 TXT 记录

当注册的客户端通过 **ipa** 命令行工具查找 IdM 提供的服务或介质服务时，它会在 **/etc/ipa/default.conf** 文件中查找由 **xmlrpc_uri** 参数指定的服务器。如果需要，它还会查找同一文件中域参数中提供的 IdM DNS **域名**，并查询该域的 _ldap._tcp.*domain_name* SRV 记录，以标识正在查找的服务器。如果 **/etc/ipa/default.conf** 文件中没有给出的域，客户端仅联系文件的 **xmlrpc_uri** 参数中设置的服务器。

请注意，IdM 客户端和服务器的主机名不需要是主 DNS 域的一部分。但是，在使用 Active Directory(AD)的信任环境中，IdM 服务器的主机名必须是 IdM 拥有的域（与 IdM 域关联的域），而不是 AD 拥有的域（与可信 AD 域关联的域）。从信任的角度来看，此关联通过 Realm 域进行管理。

有关使用 Active Directory DNS 域的主机名配置用户访问 IdM 客户端的详情，请参考 *Windows 集成指南中的* Active Directory DNS 域中的 IdM 客户端 。

验证**服务器主机名**
主机名必须是完全限定域名，如 **server.example.com**。

重要

不要使用单标签域名，例如 .company：IdM 域必须由一个或多个子域和一个顶层域组成，如 example.com 或 company.example.com。

完全限定域名必须满足以下条件：

- 它是一个有效的 DNS 名称，即只允许数字、字母字符和连字符(-)。主机名中的其他字符（如下划线(_)）会导致 DNS 失败。
- 都是小写。不允许使用大写字母。
- 完全限定域名不能解析到环回地址。它必须解析到计算机的公共 IP 地址，而不是 **127.0.0.1**。

有关其他推荐的命名实践，请参阅 *Red Hat Enterprise Linux 安全指南中的* 推荐命名实践。

要验证机器的主机名，请使用 **hostname** 工具：

```
[root@server ~]# hostname
server.example.com
```

hostname 的输出不能是 **localhost** 或 **localhost6**。

验证转发和反向 DNS 配置

1. 获取服务器的 IP 地址。**ip addr show** 命令显示 IPv4 和 IPv6 地址：
   - IPv4 地址显示在以 **inet** 开头的行上。在以下示例中，配置的 IPv4 地址为 **192.0.2.1**。
   - IPv6 地址显示在以 **inet6** 开头的行中。只有范围为 **global** 的 IPv6 地址与此流程相关。在以下示例中，返回的 IPv6 地址为 **2001:DB8::1111**。

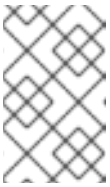
```
[root@server ~]# ip addr show
...
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
group default qlen 1000
 link/ether 00:1a:4a:10:4e:33 brd ff:ff:ff:ff:ff:ff
 inet 192.0.2.1/24 brd 192.0.2.255 scope global dynamic eth0
  valid_lft 106694sec preferred_lft 106694sec
 inet6 2001:DB8::1111/32 scope global dynamic
   valid_lft 2591521sec preferred_lft 604321sec
 inet6 fe80::56ee:75ff:fe2b:def6/64 scope link
       valid_lft forever preferred_lft forever
```

2. 使用 **dig** 工具并添加主机名，验证转发 DNS 配置。
   1. 运行 **dig +short *server.example.com* A** 命令。返回的 IPv4 地址必须与 **ip addr show** 返回的 IP 地址匹配：

```
[root@server ~]# dig +short server.example.com A
192.0.2.1
```

   2. 运行 **dig +short *server.example.com* AAAA** 命令。如果命令返回地址，它必须与 **ip addr show** 返回的 IPv6 地址匹配：

```
[root@server ~]# dig +short server.example.com AAAA
2001:DB8::1111
```



注意

如果没有为 AAAA 记录返回任何输出，它不会指出配置不正确；没有输出仅表示服务器计算机的 DNS 中未配置 IPv6 地址。如果您不打算在网络中使用 IPv6 协议，则可以继续进行安装。

3. 使用 **dig** 工具并添加 IP 地址，验证反向 DNS 配置(PTR 记录)。

    1. 运行 **dig +short -x *IPv4 address*** 命令。服务器主机名必须显示在命令输出中。例如：

    ```
    [root@server ~]# dig +short -x 192.0.2.1
    server.example.com
    ```

    2. 如果上一步中的 **dig +short -x *server.example.com* AAAA** 命令返回 IPv6 地址，则使用 dig 查询 IPv6 地址。同样，必须在命令输出中显示服务器主机名。例如：

    ```
    [root@server ~]# dig +short -x 2001:DB8::1111
    server.example.com
    ```

    注意

    如果上一步中的 **dig +short *server.example.com* AAAA** 没有显示任何 IPv6 地址，查询 AAAA 记录不会输出任何内容。在这种情况下，这是正常的行为，不代表配置不正确。

    如果显示不同的主机名或主机名，即使上一步中的 **dig +short *server.example.com*** 返回 IP 地址，这表示反向 DNS 配置不正确。

验证 DNS Forwarders 的标准合规性

当使用集成的 DNS 配置 IdM 时，建议使用 DNS 安全扩展 (DNSSEC)记录验证。通过验证来自其他服务器的已签名 DNS 记录，您可以防止 IdM 安装被欺骗地址。但是，DNSSEC 验证不是成功安装 IdM 的硬要求。

IdM 安装程序默认启用 DNSSEC 记录验证。要成功进行 DNSSEC 验证，务必要让 DNSSEC 正确配置的转发器已正确配置。在安装过程中，IdM 会检查全局转发器，如果转发器不支持 DNSSEC，则转发器上将禁用 DNSSEC 验证。

要验证您要与 IdM DNS 服务器使用的所有 DNS 转发器是否符合 DNS(EDNS0)和 DNSSEC 标准扩展机制：

```
$ dig +dnssec @IP_address_of_the_DNS_forwarder . SOA
```

命令显示的预期输出包含以下信息：

- 状态：**NOERROR**
- 标记： **ra**
- EDNS 标志： **do**

- **RRSIG** 记录必须在 **ANSWER** 部分中存在

如果输出中缺少其中任何一个项目，请检查 DNS 转发器的文档，并验证是否支持并启用了 EDNS0 和 DNSSEC。在最新版本的 BIND 服务器中，**dnssec-enable yes;** 选项必须在 **/etc/named.conf** 文件中设置。

例如，预期的输出如下：

```
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 48655
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 4096

;; ANSWER SECTION:
. 31679 IN SOA a.root-servers.net. nstld.verisign-grs.com. 2015100701 1800 900 604800 86400
. 31679 IN RRSIG SOA 8 0 86400 20151017170000 20151007160000 62530 . GNVz7SQs [...]
```

/etc/hosts 文件

重要

不要手动修改 **/etc/hosts** 文件。如果修改了 **/etc/hosts**，请确保其内容符合以下规则：

以下是正确配置了 **/etc/hosts** 文件的示例。它正确列出主机的 IPv4 和 IPv6 localhost 条目，后跟 IdM 服务器 IP 地址和主机名作为第一个条目。请注意，IdM 服务器主机名不能是 **localhost** 条目的一部分。

```
127.0.0.1 localhost.localdomain localhost
::1  localhost6.localdomain6 localhost6
192.0.2.1 server.example.com server
2001:DB8::1111 server.example.com server
```

2.1.6. 端口要求

IdM 使用多个端口与其服务通信。这些端口必须处于打开状态，且可供 IdM 使用。它们不能被其他服务使用或受防火墙阻止。

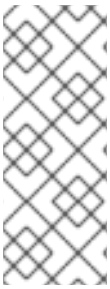
- 有关所需端口的列表，请参阅 "所需端口列表"一节。
- 有关与所需端口对应的 **firewalld** 服务列表，请参阅 "firewalld 服务列表"一节。

所需端口列表

表 2.1. 身份管理端口

服务	端口	协议
HTTP/HTTPS	80, 443	TCP
LDAP/LDAPS	389, 636	TCP
Kerberos	88, 464	TCP 和 UDP

服务	端口	协议
DNS	53	TCP 和 UDP
NTP	123	UDP



注意

不要担心 IdM 使用端口 80 和 389。

- 端口 80(HTTP)用于提供在线证书状态协议(OCSP)响应和证书撤销列表(CRL)。二者均经过数字签名，因此可防止中间人攻击。
- 端口 389(LDAP)使用 STARTTLS 和 GSSAPI 进行加密。

此外，IdM 可以侦听端口 8080，某些安装中也侦听端口 8443 和 749。但是，这三个端口仅在内部使用：尽管 IdM 保持开放，但不需要从外部访问它们。建议您不要打开端口 8080、8443 和 749，而是让端口被防火墙阻止。

firewalld 服务列表

表 2.2. firewalld 服务

服务名称	详情请查看：
freeipa-ldap	**/usr/lib/firewalld/services/freeipa-ldap.xml**
freeipa-ldaps	**/usr/lib/firewalld/services/freeipa-ldaps.xml**
dns	**/usr/lib/firewalld/services/dns.xml**

打开所需端口

1. 确保 **firewalld** 服务正在运行。
   - 查看 **firewalld** 当前是否正在运行：

     ```
     # systemctl status firewalld.service
     ```

   - 启动 **firewalld** 并将其配置为在系统引导时自动启动：

     ```
     # systemctl start firewalld.service
     # systemctl enable firewalld.service
     ```

2. 使用 **firewall-cmd** 工具打开所需的端口。选择以下选项之一：
   a. 使用 **firewall-cmd --add-port** 命令在防火墙中添加各个端口。例如，要在默认区中打开端口：

      ```
      # firewall-cmd --permanent --add-port={80/tcp,443/tcp,list_of_ports}
      ```

b. 使用 **firewall-cmd --add-service** 命令在防火墙中添加 **firewalld** 服务。例如，要在默认区中打开端口：

```
# firewall-cmd --permanent --add-service={freeipa-ldap,list_of_services}
```

有关使用 **firewall-cmd** 在系统上打开端口的详情，请参考 *安全指南中的使用* CLI 或 firewall-cmd(1) man page 中的 Runtime 和 Permanent Configuration 中的修改设置 。

3. 重新载入 **firewall-cmd** 配置以确保修改立即生效：

```
# firewall-cmd --reload
```

请注意，在生产环境的系统上重新载入 **firewalld** 可能会导致 DNS 连接超时。另请参阅《 安全指南》的使用 CLI 修改运行时和永久 配置中 *的设置*。如果需要，为了避免超时的风险并在运行的系统上永久保留修改，请使用 **firewall-cmd** 命令的 **--runtime-to-permanent** 选项，例如：

```
# firewall-cmd --runtime-to-permanent --add-port=
{80/tcp,443/tcp,389/tcp,636/tcp,88/tcp,88/udp,464/tcp,464/udp,53/tcp,53/udp,123/udp}
```

4. *可选*。要验证端口现在是否可用，请使用 **nc**、**telnet** 或 **nmap** 工具连接到端口或运行端口扫描。

注意

请注意，您还必须为传入和传出流量打开基于网络的防火墙。

2.2. 安装 IDM 服务器所需的软件包

安装没有集成 DNS 服务的服务器所需的软件包：

```
# yum install ipa-server
```

安装具有集成 DNS 服务的服务器所需的软件包：

```
# yum install ipa-server ipa-server-dns
```

注意

要确定 DNS 是否适合您的用例，请参阅 第 2.3.1 节 “确定使用集成 DNS”。

ipa-server 软件包会自动安装其他必需的软件包作为依赖项，例如：

- 389-DS-base 用于目录服务器 LDAP 服务
- Kerberos 服务的 krb5-server 软件包
- 各种特定于 IdM 的工具

2.3. 安装 IDM 服务器：简介

注意

以下部分中的安装过程和示例不是相互排斥的：您可以组合它们来实现所需的结果。例如，您可以安装具有集成 DNS 的服务器，并使用外部托管的 root CA 安装。

ipa-server-install 工具安装和配置 IdM 服务器。

在安装服务器前，请查看以下部分：

- 第 2.3.1 节 “确定使用集成 DNS”
- 第 2.3.2 节 “确定要使用的 CA 配置”

ipa-server-install 工具提供了一个非互动安装模式，它允许自动和无人值守的服务器设置。详情请查看 第 2.3.7 节 “非临时安装服务器”

ipa-server-install 安装脚本在 **/var/log/ipaserver-install.log** 中创建一个日志文件。如果安装失败，日志可帮助您辨别问题。

2.3.1. 确定使用集成 DNS

IdM 支持安装具有集成 DNS 或没有集成 DNS 的服务器。

带有集成 DNS 服务的 IdM 服务器

IdM 提供的集成 DNS 服务器并非旨在用作通用 DNS 服务器。它只支持与 IdM 部署和维护相关的功能。它不支持一些高级 DNS 功能。

红帽强烈建议 IdM 集成的 DNS 在 IdM 部署中的基本用法：当 IdM 服务器也管理 DNS 时，DNS 和原生 IdM 工具之间的紧密集成启用了自动化一些 DNS 记录管理。

请注意，即使 IdM 服务器被用作主 DNS 服务器，其他外部 DNS 服务器仍然可以用作从属服务器。

例如，如果您的环境已在使用其他 DNS 服务器，如 Active Directory 集成的 DNS 服务器，则只能将 IdM 主域委派给 IdM 集成的 DNS。您不需要将 DNS 区域迁移到 IdM 集成的 DNS。

注意

如果您需要在 Subject 备用名称 (SAN) 扩展中使用 IP 地址的 IdM 客户端发布证书，则必须使用 IdM 集成 DNS 服务。

要安装具有集成 DNS 的服务器，请参阅 第 2.3.3 节 “使用集成的 DNS 安装服务器”

没有集成 DNS 服务的 IdM 服务器

外部 DNS 服务器用于提供 DNS 服务。考虑在这些情况下在没有 DNS 的情况下安装 IdM 服务器：

- 如果您需要超出 IdM DNS 范围的高级 DNS 功能
- 在具有良好 DNS 基础架构的环境中，允许您使用外部 DNS 服务器

要在没有集成 DNS 的情况下安装服务器，请参阅 第 2.3.4 节 “安装没有集成的 DNS 的服务器”

重要

请确定您的系统满足 第 2.1.5 节 "主机名和 DNS 配置" 中描述的 DNS 要求。

集成或外部 DNS 的维护要求

使用集成的 DNS 服务器时，大部分 DNS 记录维护都是自动化的。您只需要：

- 设置从父域到 IdM 服务器的正确委托

  例如，如果 IdM 域名是 **ipa.example.com**，则必须从 **example.com** 域正确委托。

注意

您可以使用以下命令验证委托：

```
# dig @IP_address +norecurse +short ipa.example.com. NS
```

ip_address 是管理 **example.com** DNS 域的服务器的 IP 地址。如果委派正确，命令列出安装了 DNS 服务器的 IdM 服务器。

在使用外部 DNS 服务器时，您必须：

- 在 DNS 服务器中手动创建新域
- 使用 IdM 安装程序生成的来自区域文件中的记录手动填充新域
- 在安装或删除副本后手动更新记录，以及在服务配置中的任何更改后手动更新记录，例如在配置了 Active Directory 信任后

防止 DNS 简化攻击

IdM 集成的 DNS 服务器的默认配置允许所有客户端向 DNS 服务器发出递归查询。如果您的服务器部署到具有不受信任的客户端的网络中，请更改服务器配置以仅限制对授权客户端的递归。[1]

要确保只允许授权的客户端发出递归查询，请将适当的访问控制列表(ACL)语句添加到服务器上的 **/etc/named.conf** 文件中。例如：

```
acl authorized { 192.0.2.0/24; 198.51.100.0/24; };
options {
  allow-query { any; };
  allow-recursion { authorized; };
};
```

2.3.2. 确定要使用的 CA 配置

IdM 支持安装带有集成 IdM 证书颁发机构(CA)或没有 CA 的服务器。

带有集成 IdM CA 的服务器

这是适合大多数部署的默认配置。证书系统使用 *CA 签名证书* 在 IdM 域中创建和签署证书。

警告

红帽强烈建议将 CA 服务安装到多台服务器中。有关安装包括 CA 服务的初始服务器副本的详情请参考 第 4.5.4 节 “使用 CA 安装副本”。

如果您只在一个服务器中安装 CA，则在 CA 服务器失败时可能会丢失 CA 配置且无法恢复。详情请查看 第 B.2.6 节 “恢复丢失的 CA 服务器”。

IdM CA 签名证书可以是 root CA，也称为 **自签名**，或者可由外部 CA 签名。

IdM CA 是 root CA

这是默认配置。

要使用这个配置安装服务器，请参阅 第 2.3.3 节 “使用集成的 DNS 安装服务器” 和 第 2.3.4 节 “安装没有集成的 DNS 的服务器”。

外部 CA 是 root CA

IdM CA 属于外部 CA。但是，IdM 域的所有证书仍由证书系统实例颁发。

外部 CA 可以是企业 CA 或第三方 CA，如 Verisign 或 Thawte。外部 CA 可以是 root CA 或从属 CA。IdM 域中发布的证书可能会受到外部 root CA 或中间 CA 证书为属性（如有效周期或可发布证书的域）设置的限制。

要使用外部托管 root CA 安装服务器，请参阅 第 2.3.5 节 “使用外部 CA 作为 Root CA 安装服务器”

没有 CA 的服务器

当基础架构中的限制不允许使用服务器安装证书服务时，此配置选项适用于非常罕见的情况。

安装前，您必须从第三方颁发机构请求这些证书：

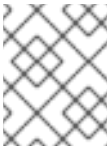
- LDAP 服务器证书和私钥
- Apache 服务器证书和私钥
- 发布 LDAP 和 Apache 服务器证书的 CA 完整 CA 证书链

警告

在没有集成 IdM CA 的情况下管理证书会带来重大维护负担。例如，您必须手动管理 IdM 服务器的 Apache Web 服务器和 LDAP 服务器证书。这包括：

- 创建和上传证书.
- 监控证书的过期日期.请注意，如果您在没有集成 CA 的情况下安装了 IdM，**certmonger** 服务不会跟踪证书。
- 在证书过期前续订证书以避免中断.

要安装没有集成 CA 的服务器，请参阅 第 2.3.6 节 “在没有 CA 的情况下安装”



注意

如果您在没有 CA 的情况下安装 IdM 域，您可以随后安装 CA 服务。要将 CA 安装到现有的 IdM 域，请参阅 第 26.8 节 “在现有 IdM 域中安装 CA”。

2.3.3. 使用集成的 DNS 安装服务器

注意

如果您不确定适合哪些 DNS 或 CA 配置，请参阅 第 2.3.1 节 “确定使用集成 DNS” 和 第 2.3.2 节 “确定要使用的 CA 配置”。

要安装具有集成 DNS 的服务器，请在安装过程中提供以下信息：

DNS forwarders

支持以下 DNS 转发器设置：

- 一个或多个转发器（非互动安装中的 **--forwarder** 选项）
- 没有转发器（非互动安装中的 **--no-forwarders** 选项）

如果您不确定是否使用 DNS 转发，请参阅 第 33.6 节 “管理 DNS 转发”。

反向 DNS 区域

支持以下反向 DNS 区域设置：

- 自动检测需要在 IdM DNS 中创建反向区域（互动安装中的默认设置，非交互式安装中的 **--auto-reverse** 选项）
- 没有反向区域自动检测（互动安装中的 **--no-reverse** 选项）

请注意，如果设置了 **-- auto-reverse 选项，则 --allow-zone- overlap** 选项将被忽略。使用选项的组合：

```
$ ipa-server-install --auto-reverse --allow-zone-overlap
```

因此，*不会*创建可与现有 DNS 区域重叠的反向区域，例如在另一个 DNS 服务器上。

对于非交互式安装，还要添加 **--setup-dns** 选项。

例 2.1. 使用集成的 DNS 安装服务器

这个过程安装服务器：

- 带有集成的 DNS
- 集成 IdM CA 作为 root CA，这是默认 CA 配置

1. 运行 **ipa-server-install** 工具。

```
# ipa-server-install
```

2. 此脚本提示配置集成的 DNS 服务。输入 **yes**。

```
Do you want to configure integrated DNS (BIND)? [no]: yes
```

3. 该脚本会提示输入几个必要的设置。

   - 要接受括号中的默认值，请按 **Enter** 键。
   - 要提供与建议的默认值不同的值，请输入所需的值。

```
Server host name [server.example.com]:
Please confirm the domain name [example.com]:
Please provide a realm name [EXAMPLE.COM]:
```

警告

红帽强烈建议 Kerberos 域名与主 DNS 域名相同，且所有字母都使用大写。例如，如果主 DNS 域是 **ipa.example.com**，请将 **IPA.EXAMPLE.COM** 用于 Kerberos 域名。

不同的命名实践将阻止您使用 Active Directory 信任，并可能导致其他负面影响。

4. 输入目录服务器超级用户 **cn=Directory Manager** 以及 **admin** IdM 系统用户帐户的密码。

```
Directory Manager password:
IPA admin password:
```

5. 此脚本提示 DNS 转发器。

```
Do you want to configure DNS forwarders? [yes]:
```

- 要配置 DNS 正向解析器，请输入 **yes**，然后按照命令行中的说明进行操作。

  安装过程会将正向解析器 IP 地址添加到安装的 IdM 服务器的 **/etc/named.conf** 文件中。

  - 有关转发策略默认设置，请查看 ipa-dns-install(1) man page 中的 **--forward-policy** 描述。
  - 详情请查看 "forward 策略"一节。

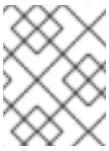
- 如果您不想使用 DNS 正向解析，请输入 **no**。

6. 脚本会提示检查是否需要配置与服务器关联的 IP 地址的任何 DNS 反向(PTR)记录。

```
Do you want to search for missing reverse zones? [yes]:
```

如果您运行搜索并发现丢失了反向区，脚本会询问您是否创建反向区以及 PTR 记录。

```
Do you want to create reverse zone for IP 192.0.2.1 [yes]:
Please specify the reverse zone name [2.0.192.in-addr.arpa.]:
Using reverse zone(s) 2.0.192.in-addr.arpa.
```



注意

使用 IdM 管理反向区是可选的。您可以改为使用外部 DNS 服务来实现这一目的。

7. 输入 **yes** 以确认服务器配置。

```
Continue to configure the system with these values? [no]: yes
```

8. 安装脚本现在配置服务器。等待操作完成。

9. 将父域中的 DNS 委托程序添加到 IdM DNS 域。例如，如果 IdM DNS 域是 **ipa.example.com**，请在 **example.com** 父域中添加一个名称服务器(NS)记录。

重要

每次安装 IdM DNS 服务器时，必须重复此步骤。

该脚本建议您备份 CA 证书并确保所需的网络端口已打开。有关 IdM 端口要求以及如何打开这些端口的说明的详情，请参考 第 2.1.6 节 "端口要求"。

测试新服务器：

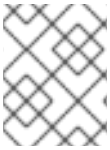
1. 使用 admin 凭据向 Kerberos 域进行身份验证。这将验证 **管理员** 是否已正确配置，并且 Kerberos 域可以访问。

```
# kinit admin
```

2. 运行一个命令，如 **ipa user-find**。在新服务器上，命令会输出唯一配置的用户： **admin**。

```
# ipa user-find admin
--------------
1 user matched
--------------
User login: admin
Last name: Administrator
Home directory: /home/admin
Login shell: /bin/bash
UID: 939000000
GID: 939000000
Account disabled: False
Password: True
Kerberos keys available: True
----------------------------
Number of entries returned 1
----------------------------
```

2.3.4. 安装没有集成的 DNS 的服务器



注意

如果您不确定适合哪些 DNS 或 CA 配置，请参阅 第 2.3.1 节 “确定使用集成 DNS” 和 第 2.3.2 节 “确定要使用的 CA 配置”。

要安装没有集成 DNS 的服务器，请在没有任何与 DNS 相关的选项的情况下运行 **ipa-server-install** 工具。

例 2.2. 安装没有集成的 DNS 的服务器

这个过程安装服务器：

- 没有集成的 DNS
- 集成 IdM CA 作为 root CA，这是默认 CA 配置

1. 运行 **ipa-server-install** 工具。

```
# ipa-server-install
```

2. 此脚本提示配置集成的 DNS 服务。按 **Enter** 键选择默认的 **no** 选项。

```
Do you want to configure integrated DNS (BIND)? [no]:
```

3. 该脚本会提示输入几个必要的设置。

   - 要接受括号中的默认值，请按 **Enter** 键。
   - 要提供与建议的默认值不同的值，请输入所需的值。

```
Server host name [server.example.com]:
Please confirm the domain name [example.com]:
Please provide a realm name [EXAMPLE.COM]:
```

警告

红帽强烈建议 Kerberos 域名与主 DNS 域名相同，且所有字母都使用大写。例如，如果主 DNS 域是 **ipa.example.com**，请将 **IPA.EXAMPLE.COM** 用于 Kerberos 域名。

不同的命名实践将阻止您使用 Active Directory 信任，并可能导致其他负面影响。

4. 输入目录服务器超级用户 **cn=Directory Manager** 以及 **admin** IdM 系统用户帐户的密码。

```
Directory Manager password:
IPA admin password:
```

5. 输入 **yes** 以确认服务器配置。

```
Continue to configure the system with these values? [no]: yes
```

6. 安装脚本现在配置服务器。等待操作完成。

7. 安装脚本生成包含 DNS 资源记录的文件：下面示例输出中的 **/tmp/ipa.system.records.UFRPto.db** 文件。将这些记录添加到现有的外部 DNS 服务器中。更新 DNS 记录的过程因特定的 DNS 解决方案而异。

```
...
Restarting the KDC
Please add records in this file to your DNS system: /tmp/ipa.system.records.UFRBto.db
Restarting the web server
...
```

重要

在将 DNS 记录添加到现有 DNS 服务器之前，服务器安装不会完成。

该脚本建议您备份 CA 证书并确保所需的网络端口已打开。有关 IdM 端口要求以及如何打开这些端口的说明的详情，请参考 第 2.1.6 节 “端口要求”。

测试新服务器：

1. 使用 admin 凭据向 Kerberos 域进行身份验证。这将验证 **管理员** 是否已正确配置，并且 Kerberos 域可以访问。

```
# kinit admin
```

2. 运行一个命令，如 **ipa user-find**。在新服务器上，命令会输出唯一配置的用户： **admin**。

```
# ipa user-find admin
--------------
```

```
1 user matched
----------
User login: admin
Last name: Administrator
Home directory: /home/admin
Login shell: /bin/bash
UID: 939000000
GID: 939000000
Account disabled: False
Password: True
Kerberos keys available: True
----------------------------
Number of entries returned 1
----------------------------
```

2.3.5. 使用外部 CA 作为 Root CA 安装服务器

注意

如果您不确定适合哪些 DNS 或 CA 配置，请参阅 第 2.3.1 节 “确定使用集成 DNS” 和 第 2.3.2 节 “确定要使用的 CA 配置”。

要安装服务器并使用外部 CA 作为 root CA 链，请使用 **ipa-server-install** 工具传递这些选项：

- **--external-ca** 指定您要使用外部 CA。
- **--external-ca-type** 指定外部 CA 的类型。详情请查看 ipa-server-install(1) man page。

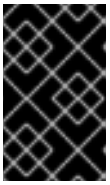
否则，大多数安装过程与 第 2.3.3 节 “使用集成的 DNS 安装服务器” 或 第 2.3.4 节 “安装没有集成的 DNS 的服务器” 中的相同。

在证书系统实例配置过程中，该工具会打印证书签名请求(CSR)的位置： **/root/ipa.csr**:

```
...

Configuring certificate server (pki-tomcatd): Estimated time 3 minutes 30 seconds
  [1/8]: creating certificate server user
  [2/8]: configuring certificate server instance
The next step is to get /root/ipa.csr signed by your CA and re-run /sbin/ipa-server-install as: /sbin/ipa-server-install --external-cert-file=/path/to/signed_certificate --external-cert-file=/path/to/external_ca_certificate
```

当发生这种情况时：

1. 将位于 **/root/ipa.csr** 中的 CSR 提交给外部 CA。这个过程根据要用作外部 CA 的服务的不同而有所不同。



重要

可能需要为证书请求适当的扩展。为身份管理生成的 CA 签名证书必须是有效的 CA 证书。这要求您在基本限制扩展 **true** 中设置 ***CA*** 参数。详情请查看 *RFC 5280* 中 *的基本约束* 部分。

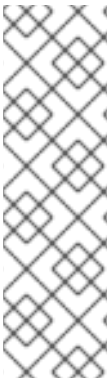
2. 在基础 64 编码 blob 中检索颁发的证书和颁发 CA 的 CA 证书链（Windows CA 的 PEM 文件或 Base_64 证书）。同样，不同的证书服务的进程会有所不同。通常，网页或通知电子邮件中的下载链接允许管理员下载所有需要的证书。

重要

确保获取 CA 的完整证书链，而不只是 CA 证书。

3. 再次运行 **ipa-server-install**，这次指定新发布的 CA 证书和 CA 链文件的位置和名称。例如：

```
# ipa-server-install --external-cert-file=/tmp/servercert20110601.pem --external-cert-file=/tmp/cacert.pem
```



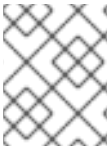
注意

ipa-server-install --external-ca 命令有时可能会失败，并显示以下错误：

```
ipa        : CRITICAL failed to configure ca instance Command '/usr/sbin/pkispawn -s CA -f /tmp/configuration_file' returned non-zero exit status 1
Configuration of CA failed
```

当设置 ***_proxy** 环境变量时，会发生此失败。有关如何解决这个问题的解决方案，请参阅 第 B.1.1 节 “外部 CA 安装失败”

2.3.6. 在没有 CA 的情况下安装



注意

如果您不确定适合哪些 DNS 或 CA 配置，请参阅 第 2.3.1 节 “确定使用集成 DNS” 和 第 2.3.2 节 “确定要使用的 CA 配置”。

要安装没有 CA 的服务器，您必须通过在 **ipa-server-install** 工具中添加选项来手动提供所需的证书。另外，大多数安装过程与 第 2.3.3 节 “使用集成的 DNS 安装服务器” 或 第 2.3.4 节 “安装没有集成的 DNS 的服务器” 中的相同。

重要

您不能使用自签名第三方服务器证书安装服务器或副本。

安装没有 CA 的 IdM 服务器所需的证书
对于没有 CA 的 IdM 服务器安装，您必须提供以下证书：

- 使用这些选项提供的 LDAP 服务器证书和私钥：
  - **--dirsrv-cert-file** 用于 LDAP 服务器证书的证书和私钥文件
  - **--dirsrv-pin** 用于访问 **--dirsrv-cert-file**中指定的文件中的私钥的密码
- 使用这些选项提供的 Apache 服务器证书和私钥：
  - **--http-cert-file** 用于 Apache 服务器证书的证书和私钥文件

- **--http-pin**，用于访问 **--http-cert-file**中指定的文件中的私钥的密码

- 发布 LDAP 和 Apache 服务器证书的 CA 的完整 CA 证书链，使用这些选项提供：
  - **--dirsrv-cert-file** 和 **--http-cert-file**用于具有完整 CA 证书链或部分证书链的证书文件

    您可以提供在 **--dirsrv-cert-file** 和 **--http-cert-file** 选项中指定的以下格式的文件：

    - Privacy-Enhanced Mail(PEM)编码的证书(RFC 7468)。请注意，IdM 安装程序接受串联的 PEM 编码对象。
    - 区分编码规则(DER)
    - PKCS #7 证书链对象
    - PKCS #8 私钥对象
    - PKCS #12 归档

    您可以多次指定 **--dirsrv-cert-file** 和 **--http-cert-file** 选项来指定多个文件。

- 如有必要，完成完整 CA 证书链的证书文件，使用这个选项提供：
  - **--ca-cert-file**，您可以多次添加这个选项

- （可选）提供外部 Kerberos 密钥分发中心(KDC)PKINIT 证书的证书文件，使用以下选项提供：
  - **--pkinit-cert-file** 用于 Kerberos KDC SSL 证书和私钥
  - **--PKINIT-pin**，密码用于解锁 Kerberos KDC 私钥

  如果您不提供 PKINIT 证书，**ipa-server-install** 使用带有自签名证书的本地 KDC 来配置 IdM 服务器。详情请查看 *第 27 章 IdM 中的 Kerberos PKINIT 身份验证*。

使用 **--dirsrv-cert-file** 和 **--http-cert-file** 提供的文件以及使用 **--ca-cert-file** 提供的文件必须包含 CA 的完整 CA 证书链，并发出 LDAP 和 Apache 服务器证书。

有关证书文件接受哪些选项的详情，请查看 ipa-server-install(1) man page。

注意

列出的命令行选项与 **--external-ca** 选项不兼容。

注意

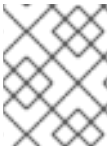
较早版本的身份管理使用 **--root-ca-file** 选项指定 root CA 证书的 PEM 文件。这不再需要，因为可信 CA 始终是 DS 和 HTTP 服务器证书的签发者。IdM 现在从 **--dirsrv-cert-file**、**--http-cert-file 和 --ca-cert-file 指定的证书自动识别 root CA** 证书。

例 2.3. 安装没有 CA 的 IdM 服务器的命令示例

```
[root@server ~]# ipa-server-install \
    --http-cert-file /tmp/server.crt \
    --http-cert-file /tmp/server.key \
    --http-pin secret \
    --dirsrv-cert-file /tmp/server.crt \
```

```
--dirsrv-cert-file /tmp/server.key \
--dirsrv-pin secret \
--ca-cert-file ca.crt
```

2.3.7. 非临时安装服务器



注意

如果您不确定适合哪些 DNS 或 CA 配置，请参阅 第 2.3.1 节 “确定使用集成 DNS” 和 第 2.3.2 节 “确定要使用的 CA 配置”。

非互动安装的最低必需选项为：

- **--ds-password** 为目录管理者(DM)（目录服务器超级用户）提供密码
- **--admin-password** 为 **admin** （IdM 管理员）提供密码
- **--realm** 提供 Kerberos 领域名
- **--unattended**，让安装进程为主机名和域名选择默认选项

另外，您还可以为这些设置提供自定义值：

- **--hostname** 作为服务器主机名
- **--domain** 作为域名

您还可以使用 **--dirsrv-config-file** 参数更改默认目录服务器设置，方法是使用带有自定义值的 LDIF 文件的路径。如需更多信息，请参阅 IdM 现在支持在服务器或副本安装过程中 *为 Red Hat Enterprise Linux 7.3* 设置独立目录服务器选项。

警告

红帽强烈建议 Kerberos 域名与主 DNS 域名相同，且所有字母都使用大写。例如，如果主 DNS 域是 **ipa.example.com**，请将 **IPA.EXAMPLE.COM** 用于 Kerberos 域名。

不同的命名实践将阻止您使用 Active Directory 信任，并可能导致其他负面影响。

如需 **ipa-server-install** 接受的选项的完整列表，请运行 **ipa-server-install --help** 命令。

例 2.4. 基本安装（不交互）

1. 运行 **ipa-server-install** 工具，提供所需的设置。例如，以下命令在没有集成 DNS 的情况下安装服务器，并使用集成 CA:

```
# ipa-server-install --realm EXAMPLE.COM --ds-password DM_password --admin-password admin_password --unattended
```

2. 设置脚本现在配置服务器。等待操作完成。

3. 安装脚本生成包含 DNS 资源记录的文件：下面示例输出中的 **/tmp/ipa.system.records.UFRPto.db** 文件。将这些记录添加到现有的外部 DNS 服务器中。更新 DNS 记录的过程因特定的 DNS 解决方案而异。

```
...
Restarting the KDC
Please add records in this file to your DNS system: /tmp/ipa.system.records.UFRBto.db
Restarting the web server
...
```

重要

在将 DNS 记录添加到现有 DNS 服务器之前，服务器安装不会完成。

该脚本建议您备份 CA 证书并确保所需的网络端口已打开。有关 IdM 端口要求以及如何打开这些端口的说明的详情，请参考 第 2.1.6 节 “端口要求”。

测试新服务器：

1. 使用 admin 凭据向 Kerberos 域进行身份验证。这将验证 **管理员** 是否已正确配置，并且 Kerberos 域可以访问。

```
# kinit admin
```

2. 运行一个命令，如 **ipa user-find**。在新服务器上，命令会输出唯一配置的用户： **admin**。

```
# ipa user-find admin
--------------
1 user matched
--------------
User login: admin
Last name: Administrator
Home directory: /home/admin
Login shell: /bin/bash
UID: 939000000
GID: 939000000
Account disabled: False
Password: True
Kerberos keys available: True
----------------------------
Number of entries returned 1
----------------------------
```

2.4. 卸载 IDM 服务器

注意

在域级别 **0** 中，这个过程不同。请参阅 第 D.3.6 节 “删除副本”。

先决条件

- 在卸载充当证书颁发机构(CA)、密钥恢复机构(KRA)或 DNS 安全扩展(DNSSEC)服务器的服务器之前，请确保这些服务正在域的另一个服务器上运行。

警告

删除充当 CA、KRA 或 DNSSEC 服务器的最后一个副本可能会严重破坏身份管理功能。

步骤

卸载 **server.example.com** ：

1. 在另一台服务器上，使用 **ipa server-del** 命令从拓扑中删除 **server.example.com** ：

```
[root@another_server ~]# ipa server-del server.example.com
```

2. 在 **server.example.com** 上，使用 **ipa-server-install --uninstall** 命令：

```
[root@server ~]# ipa-server-install --uninstall
```

3. 确保指向 **server.example.com** 的所有名称服务器(NS) DNS 记录已从 DNS 区域中删除。无论您使用由 IdM 还是外部 DNS 管理的集成 DNS，这个均适用。

2.5. 重命名服务器

设置后无法更改 IdM 服务器的主机名。但是，您可以使用其他名称将服务器替换为副本。

1. 使用 CA 和新的主机名或 IP 地址创建新服务器副本。这在 第 4 章 *安装和卸载身份管理副本* 中描述。
2. 停止初始 IdM 服务器实例。

```
[root@old_server ~]# ipactl stop
```

3. 验证所有其他副本和客户端是否像以前一样工作。
4. 卸载初始 IdM 服务器，如 所述 第 2.4 节 “卸载 IdM 服务器”

[1] 详情请查看 DNS Amplification Attacks 页面。

第 3 章 安装和卸载身份管理客户端

本章说明了如何将系统配置为作为注册到服务器的客户端计算机加入身份管理(IdM)域。

注意

有关 IdM 域中客户端和服务器的详情，请参阅 第 1.2 节 “Identity Management Domain”。

3.1. 安装客户端的先决条件

DNS 要求

使用适当的 DNS 委派.有关 IdM 中 DNS 要求的详情，请参考 第 2.1.5 节 “主机名和 DNS 配置”。

不要更改客户端上的 **resolv.conf** 文件。

端口要求

IdM 客户端连接到 IdM 服务器上的多个端口，以与其服务通信。这些端口必须在 *IdM 服务器上* 以进入方向打开。有关 IdM 需要的端口的更多信息，请参阅 第 2.1.6 节 “端口要求”。

在客户端上，以传出方向打开这些端口。如果您使用的防火墙不过滤传出数据包，如 **firewalld**，这些端口已在传出方向中可用。

名称服务缓存守护进程(NSCD)要求

红帽建议在身份管理机器上禁用 NSCD。另外，如果没有禁用 NSCD，则只为 SSSD 不缓存的映射启用 NSCD。

NSCD 和 SSSD 服务都执行缓存，当系统同时同时使用这两个服务时，可能会出现问题。有关如何避免 NSCD 和 SSSD 之间冲突的信息，请参阅 系统级身份验证指南。

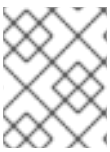
3.1.1. 安装 IdM 客户端的 RHEL 支持版本

IdM 服务器在最新的 RHEL 7 次要版本中运行的身份管理(IdM)部署支持在最新次版本中运行的客户端：

- RHEL 7

  RHEL 8

  RHEL 9



注意

如果您计划使 IdM 部署 FIPS 兼容，则 {RH} 强烈建议您将环境迁移到 RHEL 9。RHEL 9 是为 FIPS 140-3 认证的第一个主要 RHEL 版本。

3.1.2. 在 FIPS 环境中安装客户端的先决条件

在使用 Red Hat Enterprise Linux 7.4 及更新的版本设置的环境中：

- 您可以在启用了联邦信息处理标准(FIPS)模式的系统上配置新的客户端。安装脚本自动检测启用了 FIPS 的系统，并在没有管理员干预的情况下配置 IdM。

要在操作系统中启用 FIPS，请参阅 *安全指南* 中的启用 FIPS 模式。

在使用 Red Hat Enterprise Linux 7.3 及更早版本设置的环境中：

- IdM 不支持 FIPS 模式。在安装 IdM 客户端前禁用系统中的 FIPS，且不会在安装后启用它。

有关 FIPS 模式的详情，请参阅 *安全指南* 中的 联邦信息处理标准(FIPS)。

3.2. 安装客户端所需的软件包

安装 ipa-client 软件包：

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

ipa-client 软件包会自动安装其他必需的软件包作为依赖项，如系统安全服务守护进程(SSSD)软件包。

3.3. 安装客户端

ipa-client-install 工具安装并配置 IdM 客户端。安装过程要求您提供可用于注册客户端的凭证。支持以下验证方法：

授权注册客户端的用户凭证，如 admin

默认情况下，**ipa-client-install** 需要这个选项。请参阅 第 3.3.1 节 “交互式安装客户端” 以获得示例。

要直接向 **ipa-client-install** 提供用户凭证，请使用 **--principal** 和 **--password** 选项。

服务器上预生成的随机、一次性密码

要使用此验证方法，请在 **ipa-client-install** 选项中添加 **--random** 选项。请参阅 例 3.1 “使用 Random 密码以非交互方式安装客户端”。

来自之前的报名登记的主体

要使用此身份验证方法，请将 **--keytab** 选项添加到 **ipa-client-install**。详情请查看 第 3.8 节 “将客户端重新注册到 IdM 域”。

详情请查看 ipa-client-install(1) man page。

以下小节记录了基本安装场景。有关使用 **ipa-client-install** 和接受选项的完整列表的详情，请查看 ipa-client-install(1) man page。

3.3.1. 交互式安装客户端

以下流程在提示用户在需要时输入时安装客户端。用户提供授权将客户端注册到域中的用户凭据，如 **admin** 用户。

1. 运行 **ipa-client-install** 工具。

   添加 **--enable-dns-updates** 选项，如果适用，请使用客户端机器的 IP 地址更新 DNS 记录：

   - 客户端注册的 IdM 服务器已安装了集成的 DNS
   - 网络上的 DNS 服务器接受使用 GSS-TSIG 协议的 DNS 条目更新

添加 **--no-krb5-offline-passwords** 选项以禁用在 SSSD 缓存中存储 Kerberos 密码。

2. 安装脚本会尝试自动获取所有必需的设置。

   a. 如果您的系统上正确设置了 DNS 区域和 SRV 记录，脚本会自动发现所有需要的值并打印它们。输入 **yes** 以确认。

   ```
   Client hostname: client.example.com
   Realm: EXAMPLE.COM
   DNS Domain: example.com
   IPA Server: server.example.com
   BaseDN: dc=example,dc=com

   Continue to configure the system with these values? [no]: yes
   ```

   如果要使用不同值安装系统，请取消当前安装。然后再次运行 **ipa-client-install**，并使用命令行选项指定所需的值。

   详情请查看 ipa-client-install(1) man page 中的 **DNS 自动发现** 部分。

   b. 如果脚本自动获取一些设置，它会提示您输入这些值。

重要

不要使用单标签域名，例如 .company：IdM 域必须由一个或多个子域和一个顶层域组成，如 example.com 或 company.example.com。

完全限定域名必须满足以下条件：

- 它是一个有效的 DNS 名称，即只允许数字、字母字符和连字符(-)。主机名中的其他字符（如下划线(_)）会导致 DNS 失败。
- 都是小写。不允许使用大写字母。
- 完全限定域名不能解析到环回地址。它必须解析到计算机的公共 IP 地址，而不是 **127.0.0.1**。

有关其他推荐的命名实践，请参阅 *Red Hat Enterprise Linux 安全指南中的* 推荐命名实践。

3. 该脚本提示其身份用于注册客户端的用户。默认情况下，这是 **admin** 用户：

   ```
   User authorized to enroll computers: admin
   Password for admin@EXAMPLE.COM
   ```

4. 安装脚本现在配置客户端。等待操作完成。

   ```
   Client configuration complete.
   ```

5. 运行 **ipa-client-automount** 工具，它为 IdM 自动配置 NFS。详情请查看 第 34.2.1 节 “自动配置 NFS”。

3.3.2. 以非交互方式安装客户端

对于非互动安装，请使用命令行选项向 **ipa-client-install** 工具提供所有必需的信息。非互动安装的最低必需选项为：

- 指定用于注册客户端的凭证的选项；详情请参阅 第 3.3 节 “安装客户端”
- **--unattended** 允许运行安装而无需用户确认

如果您的系统上正确设置了 DNS 区域和 SRV 记录，脚本会自动发现所有其他必要值。如果脚本无法自动发现这些值，请使用命令行选项提供值。

- **--hostname** 为客户端机器指定静态主机名

重要

不要使用单标签域名，例如 .company：IdM 域必须由一个或多个子域和一个顶层域组成，如 example.com 或 company.example.com。

完全限定域名必须满足以下条件：

- 它是一个有效的 DNS 名称，即只允许数字、字母字符和连字符(-)。主机名中的其他字符（如下划线(_)）会导致 DNS 失败。
- 都是小写。不允许使用大写字母。
- 完全限定域名不能解析到环回地址。它必须解析到计算机的公共 IP 地址，而不是 **127.0.0.1**。

有关其他推荐的命名实践，请参阅 *Red Hat Enterprise Linux 安全指南中的* 推荐命名实践。

- **--server** 用于指定客户端将要注册的 IdM 服务器的主机名
- **--domain** 用于指定客户端将要注册的 IdM 服务器的 DNS 域名
- **--realm** 指定 Kerberos 域名

添加 **--enable-dns-updates** 选项，如果适用，请使用客户端机器的 IP 地址更新 DNS 记录：

- 客户端注册的 IdM 服务器已安装了集成的 DNS
- 网络上的 DNS 服务器接受使用 GSS-TSIG 协议的 DNS 条目更新

添加 **--no-krb5-offline-passwords** 选项以禁用在 SSSD 缓存中存储 Kerberos 密码。

有关 **ipa-client-install** 接受的选项的完整列表，请查看 ipa-client-install(1) man page。

例 3.1. 使用 Random 密码以非交互方式安装客户端

这个过程会在不提示用户进行任何输入的情况下安装客户端。进程包括在服务器上预生成用于授权注册的一次性密码。

1. 在现有服务器中：
   a. 以管理员身份登录：

```
$ kinit admin
```

b. 将新机器作为 IdM 主机添加。在 **ipa host-add** 命令中使用 **--random** 选项来生成随机密码。

```
$ ipa host-add client.example.com --random
--------------------------------------------------
Added host "client.example.com"
--------------------------------------------------
  Host name: client.example.com
  Random password: W5YpARl=7M.n
  Password: True
  Keytab: False
  Managed by: server.example.com
```

当使用生成的密码将机器注册到 IdM 域后，生成的密码将变为无效。注册完成后，它将被一个正确的主机 keytab 替换。

2. 在您要安装客户端的机器上，运行 **ipa-client-install**，并使用以下选项：

- **--password** 用于 **ipa host-add** 输出中的随机密码

注意

密码通常包含特殊字符。因此，将其括在单引号(')中。

- **--unattended** 允许运行安装而无需用户确认

如果您的系统上正确设置了 DNS 区域和 SRV 记录，脚本会自动发现所有其他必要值。如果脚本无法自动发现这些值，请使用命令行选项提供值。

例如：

```
# ipa-client-install --password 'W5YpARl=7M.n' --domain example.com --server
server.example.com --unattended
```

3. 运行 **ipa-client-automount** 工具，它为 IdM 自动配置 NFS。详情请查看 第 34.2.1 节 “自动配置 NFS”。

3.4. 通过 KICKSTART 设置 IDM 客户端

Kickstart 注册会在安装 Red Hat Enterprise Linux 时自动将新系统添加到 IdM 域。有关 Kickstart 的详情，请查看安装 *指南中的 Kickstart 安装*。

准备 Kickstart 客户端安装包括以下步骤：

1. 第 3.4.1 节 “在 IdM 服务器中预先创建客户端主机条目”
2. 第 3.4.2 节 “为客户端创建 Kickstart 文件”

3.4.1. 在 IdM 服务器中预先创建客户端主机条目

1. 以 admin 用户身份登录：

```
$ kinit admin
```

2. 在 IdM 服务器上创建主机条目，并为该条目设置一个临时密码：

```
$ ipa host-add client.example.com --password=secret
```

Kickstart 使用密码在客户端安装过程中进行验证，并在第一次验证尝试后过期。成功安装客户端后，它会使用它的 keytab 进行验证。

3.4.2. 为客户端创建 Kickstart 文件

用于设置 IdM 客户端的 Kickstart 文件必须包含以下内容：

- 要安装的软件包列表中的 ipa-client 软件包：

```
%packages
@ X Window System
@ Desktop
@ Sound and Video
ipa-client
...
```

详情请参阅 *安装指南中的*软件包选择。

- 安装后说明：
  - 确保已生成 SSH 密钥，然后再注册
  - 运行 **ipa-client-install** 工具，指定：
    - 访问和配置 IdM 域服务所需的所有信息
    - 在 IdM 服务器上预先创建客户端主机时设置的密码，以 第 3.4.1 节 “在 IdM 服务器中预先创建客户端主机条目”

例如：

```
%post --log=/root/ks-post.log

# Generate SSH keys to ensure that ipa-client-install uploads them to the IdM server
/usr/sbin/sshd-keygen

# Run the client install script
/usr/sbin/ipa-client-install --hostname=client.example.com --domain=EXAMPLE.COM --
enable-dns-updates --mkhomedir -w secret --realm=EXAMPLE.COM --
server=server.example.com
```

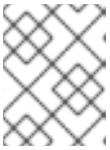
对于非交互式安装，还要添加 **--unattended** 选项。

要让客户端安装脚本为机器请求证书：

- 将 **--request-cert** 选项添加到 **ipa-client-install**。

- 将 kickstart **chroot** 环境中的 **getcert** 和 **ipa-client-install** 工具的系统总线地址设置为 **/dev/null**。要做到这一点，请在 **ipa-client-install** 指令前将这些行添加到安装后指令文件中：

```
# env DBUS_SYSTEM_BUS_ADDRESS=unix:path=/dev/null getcert list
# env DBUS_SYSTEM_BUS_ADDRESS=unix:path=/dev/null ipa-client-install
```



注意

红帽建议不要在 kickstart 注册前启动 **sshd** 服务。在注册客户端前启动 **sshd** 会自动生成 SSH 密钥，但使用上述脚本是首选的解决方案。

有关详细信息，请参阅 *安装指南中的* 安装后脚本。

有关使用 Kickstart 的详情，请参考安装 *指南中如何执行 Kickstart 安装？*有关 Kickstart 文件示例，请参阅 示例 Kickstart 配置。

3.5. 客户端安装后注意事项

3.5.1. 删除优先级管理配置

ipa-client-install 脚本不会从 **/etc/openldap/ldap.conf** 和 **/etc/sssd/sssd.conf** 文件中删除任何之前的 LDAP 和 SSSD 配置。如果在安装客户端前修改了这些文件中的配置，该脚本会添加新的客户端值，但会将它们注释掉。例如：

```
BASE   dc=example,dc=com
URI    ldap://ldap.example.com

#URI ldaps://server.example.com # modified by IPA
#BASE dc=ipa,dc=example,dc=com # modified by IPA
```

应用新的身份管理配置值：

1. 打开 **/etc/openldap/ldap.conf** 和 **/etc/sssd/sssd.conf**。
2. 删除前面的配置。
3. 取消注释新的身份管理配置。
4. 依赖于系统范围的 LDAP 配置的服务器进程可能需要重启来应用更改。使用 **openldap** 库的应用程序通常会在启动时导入配置。

3.6. 测试新客户端

检查客户端是否可以获取有关服务器上定义的用户的信息。例如，检查默认的 **admin** 用户：

```
[user@client ~]$ id admin
uid=1254400000(admin) gid=1254400000(admins) groups=1254400000(admins)
```

3.7. 卸载客户端

卸载客户端会从 IdM 域中删除客户端，以及针对系统服务（如 SSSD）的所有 IdM 特定配置。这会恢复客户端计算机以前的配置。

1. 运行 **ipa-client-install --uninstall** 命令：

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

2. 从服务器手动删除客户端主机的 DNS 条目。请参阅 第 33.4.6 节 “从 DNS 区域中删除记录”。

3.8. 将客户端重新注册到 IDM 域

如果客户端虚拟机已被销毁，并且您仍有其 keytab，您可以重新推出客户端：

- 以交互方式使用管理员凭据.请参阅 第 3.8.1 节 “使用管理员帐户以交互方式重新注册客户端”。
- 以非交互方式使用之前备份的 keytab 文件。请参阅 第 3.8.2 节 “使用客户端 keytab 以非交互方式重新注册客户端”。

注意

您只能重新注册域条目仍然活跃的客户端。如果您卸载了客户端（使用 **ipa-client-install --uninstall**）或者禁用了其主机条目（使用 **ipa host-disable**），则无法重新注册它。

在重新注册过程中，IdM 会执行以下操作：

- 吊销原始主机证书
- 生成新主机证书
- 创建新 SSH 密钥
- 生成一个新的 keytab

3.8.1. 使用管理员帐户以交互方式重新注册客户端

1. 重新创建具有相同主机名的客户端机器。
2. 在客户端机器上运行 **ipa-client-install --force-join** 命令：

```
# ipa-client-install --force-join
```

3. 该脚本提示其身份用于注册客户端的用户。默认情况下，这是 **admin** 用户：

```
User authorized to enroll computers: admin
Password for admin@EXAMPLE.COM
```

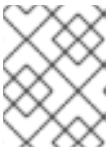
3.8.2. 使用客户端 keytab 以非交互方式重新注册客户端

使用客户端 keytab 重新注册适合自动安装，或者在使用管理员密码时不可行的情况下。

1. 备份原始客户端的 keytab 文件，例如在 **/tmp** 或 **/root** 目录中。
2. 重新创建具有相同主机名的客户端机器。

3. 重新打开客户端，并使用 **--keytab** 选项**指定 keytab** 位置：

```
# ipa-client-install --keytab /tmp/krb5.keytab
```



注意

--keytab 选项中指定的 keytab只在进行身份验证以启动注册时才使用。在重新注册过程中，IdM 为客户端生成一个新的 keytab。

3.9. 重命名客户端机器

本节介绍如何重命名 IdM 客户端。该流程涉及：

- “识别当前服务和密钥选项卡配置”一节.
- “从 IdM 域中删除客户端机器”一节.
- “使用新主机名重新注册客户端”一节.

警告

重新命名客户端是一个手动过程。除非绝对需要更改主机名，否则红帽不推荐这样做。

识别**当前服**务**和密**钥选项**卡配置**

在卸载当前客户端之前，请记下客户端的某些设置。在使用新的主机名重新注册计算机后，您将应用此配置。

1. 确定在机器上运行哪些服务：

   a. 使用 **ipa service-find** 命令，并在输出中识别带有证书的服务：

   ```
   $ ipa service-find client.example.com
   ```

   b. 此外，每个主机都有一个默认 *主机服务*，该服务不会出现在 **ipa service-find** 输出中。主机服务的服务主体（也称为 *主机主体*）*是* ***host****/client.example.com*。

2. 识别机器所属的所有主机组。

```
# ipa hostgroup-find client.example.com
```

3. 对于 **ipa service-find client.example.com** 显示的所有服务主体，请确定 **client.example.com** 上对应的 keytab 的位置。

   客户端系统上的每个服务都有一个 Kerberos 主体，格式为 *service_name/hostname@REALM*，如 **ldap/client.example.com@EXAMPLE.COM**。

从 IdM 域中删除客户端机器

1. 从 IdM 域取消注册客户端计算机。请参阅 第 3.7 节 “卸载客户端”。
2. 对于除 **/etc/krb5.keytab** 以外的每个识别的 keytab，删除旧的主体：

   ```
   [root@client ~]# ipa-rmkeytab -k /path/to/keytab -r EXAMPLE.COM
   ```

   请参阅 第 29.4 节 “删除 keytab”。
3. 在 IdM 服务器上，删除主机条目。这会删除所有服务并吊销为该主机发布的所有证书：

   ```
   [root@server ~]# ipa host-del client.example.com
   ```

此时，主机已完全从 IdM 中删除。

使用新主机名重新注册客户端

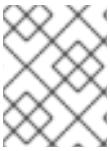
1. 根据需要重命名机器。
2. 将计算机重新注册为 IdM 客户端。请参阅 第 3.8 节 “将客户端重新注册到 IdM 域”。
3. 在 IdM 服务器中，为 “识别当前服务和密钥选项卡配置”一节 中指定的每个服务添加新的 keytab。

   ```
   [root@server ~]# ipa service-add service_name/new_host_name
   ```
4. 为在 “识别当前服务和密钥选项卡配置”一节 中分配了证书的服务生成证书。您可以做到这一点：
   - 使用 IdM 管理工具.请参阅 第 24 章 *管理用户、主机和服务的证书*。
   - 使用 **certmonger** 工具。请参阅 *系统级身份验证指南* 中的使用 certmonger 或 certmonger(8) man page。
5. 将客户端重新添加到 “识别当前服务和密钥选项卡配置”一节 中指定的主机组中。请参阅 第 13.3 节 “添加和删除用户或主机组成员”。

第 4 章 安装和卸载身份管理副本

副本通过克隆现有身份管理服务器的配置来创建。因此，服务器和其副本共享相同的核心配置。副本安装过程复制现有服务器配置，并根据该配置安装副本。

维护多个服务器副本是推荐的备份解决方案，可避免数据丢失，如 "Idup 和 Restore in IdM/IPA"知识库解决方案 中所述。



注意

另一个备份解决方案，它主要用于从副本重建 IdM 部署时，这是 **ipa-backup** 工具，如 第 9 章 *备份和恢复身份管理* 所述。

4.1. 解释 IDM 副本

要为大量客户端提供服务可用性和冗余，您可以在一个域中部署多个 IdM 服务器（称为 *副本*）。副本是初始 IdM 服务器的克隆，其功能相互相同：它们共享与用户、机器、证书和配置策略相同的内部信息。

但是，环境中只有一个服务器角色每次只能满足两个唯一的服务器角色：

- *CA 续订服务器*：此服务器管理证书颁发机构(CA)子系统证书的续订
- *CRL 生成服务器*：此服务器生成证书撤销列表(CRL)。

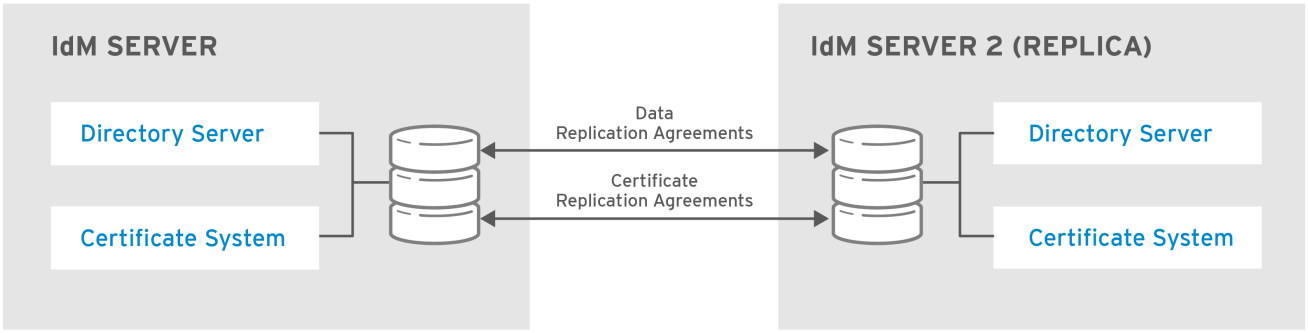
默认情况下，安装的第一个 CA 服务器同时满足 CA 续订服务器和 CRL 生成服务器角色。您可以将这些角色过渡到拓扑中的任何其他 CA 服务器，例如，如果您需要停用初始安装的服务器。这两个角色不需要由同一服务器实现。

注意

有关 IdM 拓扑中机器类型的更多信息，请参阅 第 1.2 节 “Identity Management Domain”。

复制 是在副本之间复制数据的过程。副本之间的信息使用 *多主机复制共享：通过复制* 协议加入的所有副本接收更新，因此被视为数据主服务器。

图 4.1. 服务器和副本协议



RHEL_404973_0516

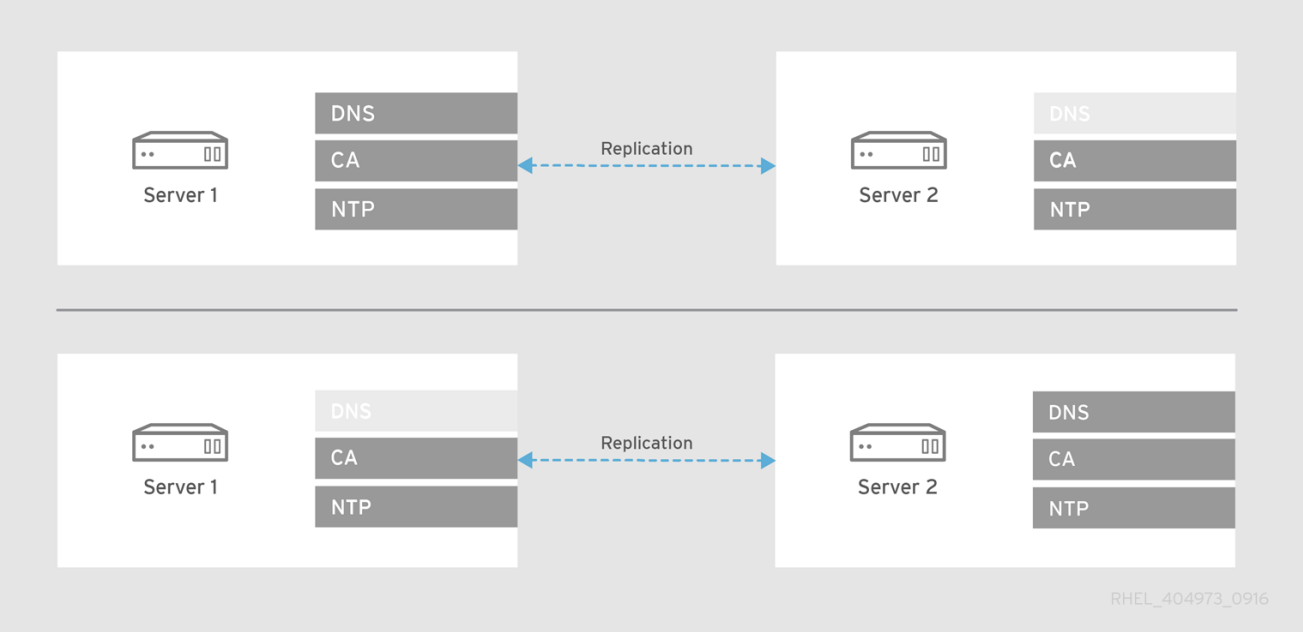
4.2. REPLICAS 的部署注意事项

4.2.1. Topology 中的服务器服务分布

IdM 服务器可以运行多个服务，如证书颁发机构(CA)或 DNS。副本可以运行与其从中创建的服务器相同的服务，但这不是必需的。

例如，您可以安装不带 DNS 服务的副本，即使初始服务器运行 DNS。同样，即使初始服务器在没有 DNS 的情况下安装，您也可以将副本设置为 DNS 服务器。

图 4.2. 带有不同服务的副本



Replicas 上的 CA 服务

如果您设置了一个没有 CA 的副本，它会将证书操作的所有请求转发到拓扑中的 CA 服务器。

警告

红帽强烈建议将 CA 服务安装到多台服务器中。有关安装包括 CA 服务的初始服务器副本的详情请参考 第 4.5.4 节 “使用 CA 安装副本”。

如果您只在一个服务器中安装 CA，则在 CA 服务器失败时可能会丢失 CA 配置且无法恢复。详情请查看 第 B.2.6 节 “恢复丢失的 CA 服务器”。

如果您在副本中设置 CA，其配置必须镜像初始服务器的 CA 配置。

- 例如，如果服务器包含集成的 IdM CA 作为 root CA，则副本还必须使用集成 CA 作为 root CA 进行安装。
- 有关支持的 CA 配置选项，请参阅 第 2.3.2 节 “确定要使用的 CA 配置”。

4.2.2. 副本拓扑建议

红帽建议遵循以下指南：

在单个 IdM 域中配置不超过 60 个副本

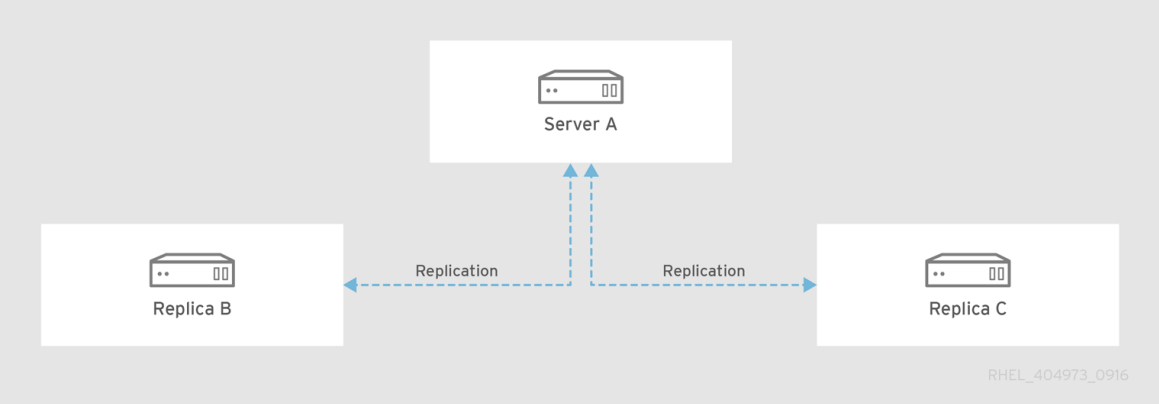
红帽保证支持具有 60 个副本或更少副本的环境。

至少配置两个，但每个副本 不超过四个 复制协议

配置额外的复制协议可确保不仅在初始副本和主服务器之间复制信息，而且还在其他副本之间复制。

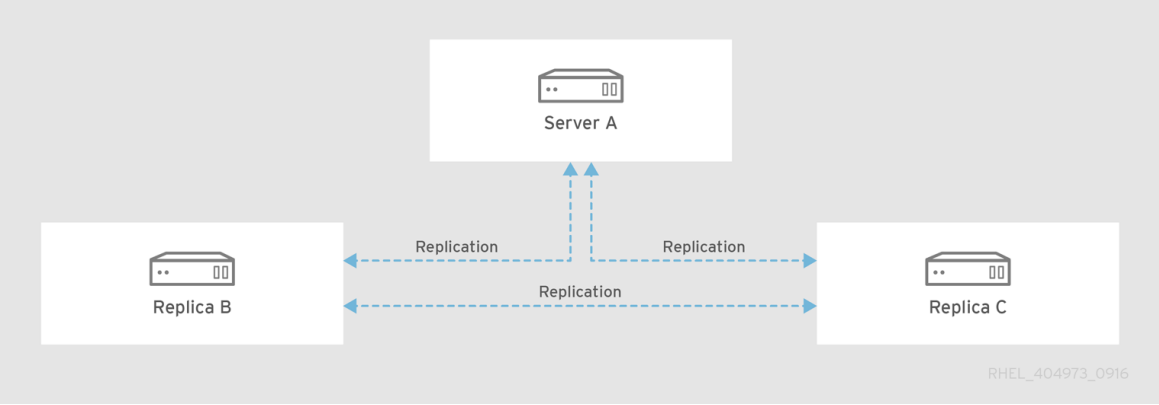
- 如果您从服务器 A 创建副本 B，然后从服务器 A 创建副本 C，则副本 B 和 C 不会被直接加入，因此在传播到副本 C 之前，必须首先将来自副本 B 的数据复制到 server A。

图 4.3. 复制协议中未加入副本 B 和 C



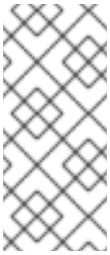
在副本 B 和副本 C 之间设置额外的复制协议可确保数据直接复制，从而提高数据可用性、一致性、故障转移容错和性能。

图 4.4. replicas B 和 Cre Joined 在复制协议中



有关管理复制协议的详情，请查看 第 6 章 *管理复制拓扑*。

不需要为每个副本配置超过四个复制协议。每个服务器有大量的复制协议不会带来显著的额外好处，因为一个消费者服务器一次只能由一个主服务器更新，因此其他协议同时闲置和等待。此外，配置太多复制协议可能会对整体性能造成负面影响。

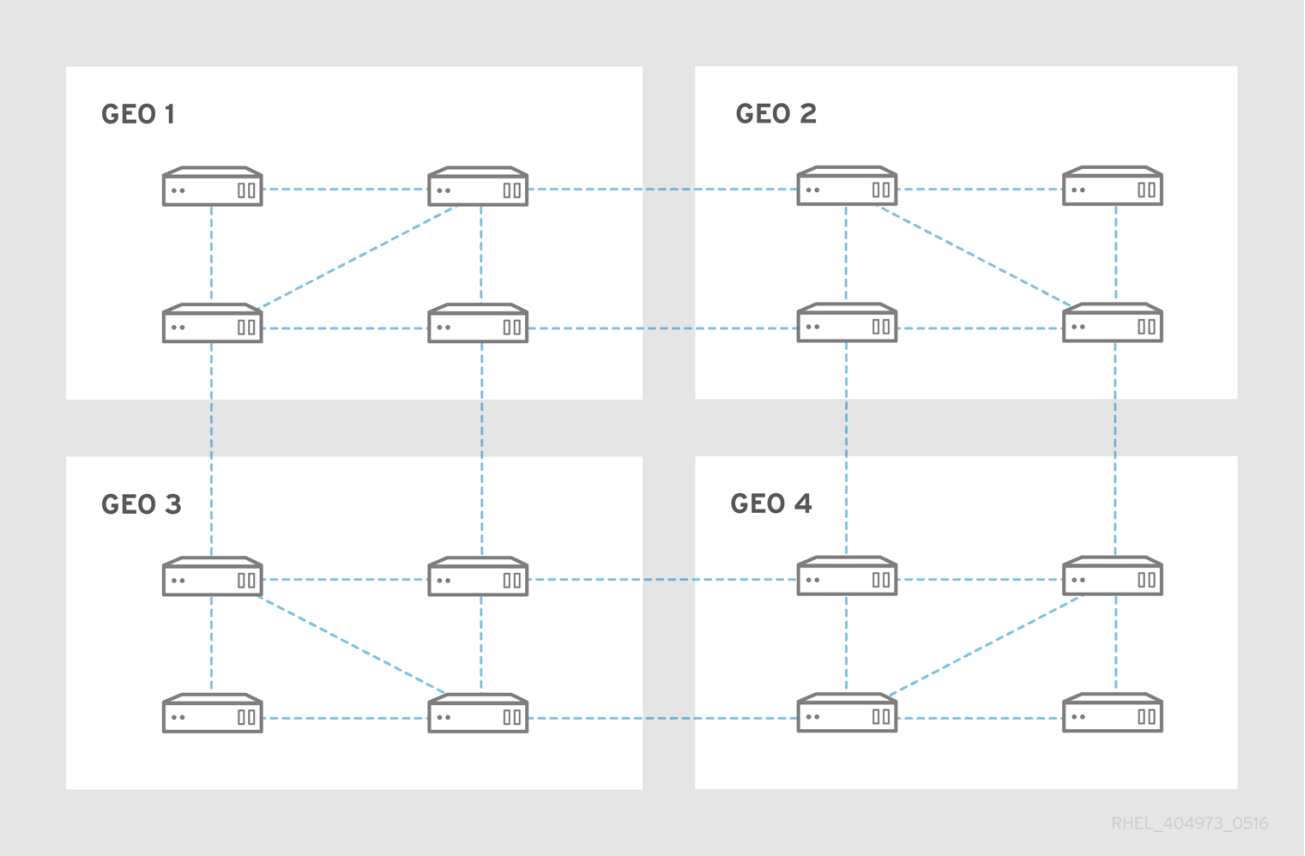


注意

ipa topologysuffix-verify 命令检查拓扑是否满足最重要的建议。运行 **ipa topologysuffix-verify --help** 获取详细信息。

命令要求您指定拓扑后缀。详情请查看 第 6.1 节 “解释复制协议、拓扑后缀和拓扑片段”。

图 4.5. 拓扑示例



4.2.2.1. 紧凑 Cell Topology

最具弹性的拓扑之一是为服务器和单元中有少量服务器的副本创建单元配置：

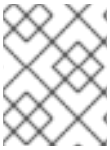
- 每个单元都是一个 *紧密的单元*，所有服务器都有复制协议。
- 每一服务器与单元 *外的其他* 服务器都有一个复制协议。这样可确保每个单元松散耦合到域中的其他单元。

达到紧张的单元拓扑：

- 每个主要办公室、数据中心或本地化都至少有一个 IdM 服务器。最好有两个 IdM 服务器。
- 每个数据中心不超过 4 个服务器.
- 在小型办公室中，而不是使用副本，使用 SSSD 将凭据和非站点 IdM 服务器缓存为数据后端。

4.2.3. Hidden Replica Mode

默认情况下，当您设置新副本时，安装程序会在 DNS 中自动创建服务(**SRV**)资源记录。这些记录可让客户端自动发现副本及其服务。隐藏的副本是一个 IdM 服务器，它具有所有运行的服务并可用。但是，它在 DNS 中没有 **SRV** 记录，并且 LDAP 服务器角色没有启用。因此，客户端无法使用服务发现来检测这些隐藏的副本。



注意

隐藏的副本功能在 Red Hat Enterprise Linux 7.7 及更新的版本中作为技术预览提供，因此不受支持。

隐藏副本主要针对可能会破坏客户端的专用服务设计。例如，IdM 的完整备份需要关闭 master 或副本中的所有 IdM 服务。因为没有客户端使用隐藏的副本，管理员可以在不影响任何客户端的情况下暂时关闭这个主机上的服务。其他用例包括 IdM API 或 LDAP 服务器上的高负载操作，如大量导入或广泛查询。

要将副本作为隐藏安装，请将 ***--hidden-replica*** 参数传递到 **ipa-replica-install** 命令。有关安装副本的详情，请参考 第 4.5 节 “创建副本：简介”。

另外，您可以更改现有副本的状态。详情请查看 第 6.5.3 节 “Hidden Replicas 的降级和升级”。

4.3. 安装副本的先决条件

副本的安装要求与 IdM 服务器的安装要求相同。确保副本机器满足 第 2.1 节 “安装服务器的先决条件” 中列出的所有先决条件。

除常规服务器要求外，还必须满足以下条件：

副本必须运行相同或更新版本的 IdM

例如，如果 master 服务器在 Red Hat Enterprise Linux 7 上运行，并使用 IdM 4.4 软件包，则副本还必须在 Red Hat Enterprise Linux 7 或更高版本中运行，并使用 IdM 版本 4.4 或更高版本。这样可确保把配置从服务器正确复制到副本。

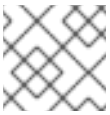
重要

IdM 不支持创建比 master 版本更早的版本副本。如果您尝试使用较早版本创建副本，则安装会失败。

副本需要打开其他端口

除了 第 2.1.6 节 “端口要求” 中描述的标准 IdM 服务器端口要求外，请确保还满足以下条件：

- 在域级别 0 上，在副本设置过程中使 *TCP 端口 22* 在主服务器上保持打开。需要此端口才能使用 SSH 连接主服务器。



注意

有关域级别的详情请参考 第 7 章 *显示和提升域级别*。

- 如果其中一个服务器正在运行 Red Hat Enterprise Linux 6 并安装了 CA，在副本配置期间和之后还打开 *TCP 端口 7389*。在纯 Red Hat Enterprise Linux 7 环境中，不需要端口 7389。

有关如何使用 **firewall-cmd** 工具打开端口的详情，请参考 第 2.1.6 节 “端口要求”。

4.4. 安装副本所需的软件包

副本软件包要求与服务器软件包要求相同。请参阅 第 2.2 节 “安装 IdM 服务器所需的软件包”。

4.5. 创建副本：简介

ipa-replica-install 工具用于从现有 IdM 服务器安装新副本。逐一安装身份管理副本。不支持同时安装多个副本。

注意

本章论述了 Red Hat Enterprise Linux 7.3 中引入的简化副本安装。这些步骤需要域级别 1（请参阅 第 7 章 *显示和提升域级别*）。

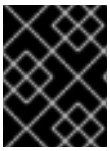
有关在域级别 0 中安装副本的文档，请参考 ???。

您可以安装新副本：

- 在现有 IdM 客户端中通过将 *客户端* 提升到副本： 请参阅 "将现有客户端提升到副本"一节
- 在尚未加入 IdM 域的机器上查看： "在不是客户端的机器上安装副本"一节

在这两种情况下，您可以通过向 **ipa-replica-install** 添加选项来自定义副本：请参阅 "使用 ipa-replica-install 为您的用例配置副本"一节。

要将副本作为隐藏安装，请将 ***--hidden-replica*** 参数传给 **ipa-replica-install**。有关隐藏副本的详情，请查看 第 4.2.3 节 "Hidden Replica Mode"。



重要

如果您要复制的 IdM 服务器与 Active Directory 有信任，在运行 **ipa-replica-install** 后将副本设置为信任代理。请参阅 *Windows 集成指南中的* 信任控制器和信任代理。

将现有客户端提升到副本

要在现有客户端上安装副本，您必须确保授权提升客户端。要实现这一点，请选择以下之一：

提供特权用户的凭证

默认特权用户为 **admin**。可以通过多种方式提供用户的凭据。您可以：

- 让 IdM 提示您以交互方式获取凭证

  注意

  这是提供特权用户的凭据的默认方式。如果在 **ipa-replica-install** 运行时没有可用的凭证，则安装会自动提示您。

- 在客户端上运行 **ipa-replica-install** 之前，以用户身份登录：

  ```
  $ kinit admin
  ```

- 将用户的主体名称和密码直接添加到 **ipa-replica-install** 中：

  ```
  # ipa-replica-install --principal admin --admin-password admin_password
  ```

将客户端添加到 ipaservers 主机组中

ipaservers 中的成员资格授予机器提升的权限，类似于特权用户的凭证。您无需提供用户的凭据。

示例：第 4.5.1 节 "使用主机密钥选项卡将客户端提升至副本."

在不是客户端的机器上安装副本

当在还没有在 IdM 域中注册的机器上运行时，**ipa-replica-install** 首先将机器注册为客户端，然后安装副本组件。

要在这种情况下安装副本，请选择以下之一：

提供特权用户的凭证

默认特权用户为 **admin**。要提供凭证，请将主体名称和密码直接添加到 **ipa-replica-install** 中：

```
# ipa-replica-install --principal admin --admin-password admin_password
```

为客户端提供随机密码

在安装副本之前，您必须在服务器上生成随机密码。在安装过程中，您不需要提供用户的凭据。

示例：第 4.5.2 节 “使用 Random 密码安装副本”

默认情况下，副本针对客户端安装程序发现的第一个 IdM 服务器安装。要根据特定服务器安装副本，请在 **ipa-replica-install** 中添加以下选项：

- **--server** 作为服务器的完全限定域名(FQDN)
- IdM DNS 域的 **--domain**

使用 ipa-replica-install 为您的用例配置副本

当不带任何选项运行时，**ipa-replica-install** 只设置基本的服务器服务。要安装其他服务，如 DNS 或证书颁发机构(CA)，请在 **ipa-replica-install** 中添加选项。

警告

红帽强烈建议将 CA 服务安装到多台服务器中。有关安装包括 CA 服务的初始服务器副本的详情请参考 第 4.5.4 节 “使用 CA 安装副本”。

如果您只在一个服务器中安装 CA，则在 CA 服务器失败时可能会丢失 CA 配置且无法恢复。详情请查看 第 B.2.6 节 “恢复丢失的 CA 服务器”。

例如，使用最显著选项安装副本的场景，请参阅：

- 第 4.5.3 节 “使用 DNS 安装副本”使用 **--setup-dns** 和 **--forwarder**
- 第 4.5.4 节 “使用 CA 安装副本”,使用 **--setup-ca**
- 第 4.5.5 节 “从没有 CA 的服务器安装 Replica”使用 **--dirsrv-cert-file**、**--dirsrv-pin**、**--http-cert-file** 和 **--http-pin**

您还可以使用 **--dirsrv-config-file** 参数更改默认目录服务器设置，方法是使用带有自定义值的 LDIF 文件的路径。如需更多信息，请参阅 IdM 现在支持在服务器或副本安装过程中 *为 Red Hat Enterprise Linux 7.3 设置独立目录服务器选项*。

有关配置副本的选项的完整列表，请查看 ipa-replica-install(1) man page。

4.5.1. 使用主机密钥选项卡将客户端提升至副本.

在此过程中，现有的 IdM 客户端使用自己的主机 keytab 来授权提升，从而将现有的 IdM 客户端提升到副本。

此过程不要求您提供管理员或目录管理器(DM)凭证。因此，它更为安全，因为不会在命令行中公开任何敏感信息。

1. 在现有服务器中：

   a. 以管理员身份登录。

   ```
   $ kinit admin
   ```

   b. 将客户端机器添加到 **ipaservers** 主机组中。

   ```
   $ ipa hostgroup-add-member ipaservers --hosts client.example.com
     Host-group: ipaservers
     Description: IPA server hosts
     Member hosts: server.example.com, client.example.com
   -------------------------
   Number of members added 1
   -------------------------
   ```

   ipaservers 中的成员资格授予机器提升的权限，类似于管理员的凭证。

2. 在客户端上运行 **ipa-replica-install** 工具。

   ```
   # ipa-replica-install
   ```

3. 另外，如果您正在复制的 IdM 服务器有 Active Directory 信任，请将副本设置为信任代理或信任控制器。详情请参阅 *Windows 集成指南中的* 信任控制器和信任代理。

4.5.2. 使用 Random 密码安装副本

在此过程中，副本会从头开始安装在尚未是 IdM 客户端的机器上。为授权注册，仅使用针对一个客户端注册的、特定于客户端的随机密码。

此过程不要求您提供管理员或目录管理器(DM)凭证。因此，它更为安全，因为不会在命令行中公开任何敏感信息。

1. 在现有服务器中：

   a. 以管理员身份登录。

   ```
   $ kinit admin
   ```

   b. 将新机器作为 IdM 主机添加。在 **ipa host-add** 命令中使用 **--random** 选项来生成用于副本安装的随机一次性密码。

   ```
   $ ipa host-add client.example.com --random
   --------------------------------------------------
   Added host "client.example.com"
   --------------------------------------------------
   ```

```
Host name: client.example.com
Random password: W5YpARl=7M.n
Password: True
Keytab: False
Managed by: server.example.com
```

当使用生成的密码将机器注册到 IdM 域后，生成的密码将变为无效。注册完成后，它将被一个正确的主机 keytab 替换。

c. 将机器添加到 **ipaservers** 主机组中。

```
$ ipa hostgroup-add-member ipaservers --hosts client.example.com
  Host-group: ipaservers
  Description: IPA server hosts
  Member hosts: server.example.com, client.example.com
-------------------------
Number of members added 1
-------------------------
```

ipaservers 中的成员资格授予设置所需服务器服务所需的机器提升特权。

2. 在您要安装副本的机器上，运行 **ipa-replica-install**，并使用 **--password** 选项提供随机密码。将密码括在单引号(')中，因为它通常包含特殊字符：

```
# ipa-replica-install --password 'W5YpARl=7M.n'
```

3. 另外，如果您正在复制的 IdM 服务器有 Active Directory 信任，请将副本设置为信任代理或信任控制器。详情请参阅 *Windows 集成指南中的* 信任控制器和信任代理。

4.5.3. 使用 DNS 安装副本

此流程适用于在客户端以及尚不属于 IdM 域的机器上安装副本。详情请查看 第 4.5 节 “创建副本：简介”。

1. 使用以下选项运行 **ipa-replica-install** ：

   - **--setup-dns** 以创建 DNS 区域（如果不存在），并将副本配置为 DNS 服务器
   - 如果您不想使用任何 forwarders，**指定 --forwarder 指定 forwarder 或 --no-** forwarder

     出于故障转移的原因，需要指定多个正向解析器，请多次使用 **--forwarder**。

   例如：

```
# ipa-replica-install --setup-dns --forwarder 192.0.2.1
```

注意

ipa-replica-install 工具接受与 DNS 设置相关的许多其他选项，如 **--no-reverse** 或 **--no-host-dns**。有关它们的详情请参考 ipa-replica-install(1) man page。

2. 如果初始服务器是在启用了 DNS 的情况下创建的，则会使用正确的 DNS 条目自动创建副本。这些条目可确保 IdM 客户端能够发现新服务器。

如果初始服务器未启用 DNS，请手动添加 DNS 记录。域服务需要以下 DNS 记录：

- **_ldap._tcp**
- **_kerberos._tcp**
- **_kerberos._udp**
- **_kerberos-master._tcp**
- **_kerberos-master._udp**
- **_ntp._udp**
- **_kpasswd._tcp**
- **_kpasswd._udp**

这个示例演示了如何验证条目是否存在：

a. 为 DOMAIN 和 NAMESERVER 变量设置适当的值：

```
# DOMAIN=example.com
# NAMESERVER=replica
```

b. 使用以下命令检查 DNS 条目：

```
# for i in _ldap._tcp _kerberos._tcp _kerberos._udp _kerberos-master._tcp _kerberos-
master._udp _ntp._udp ; do
dig @${NAMESERVER} ${i}.${DOMAIN} srv +nocmd +noquestion +nocomments
+nostats +noaa +noadditional +noauthority
done | egrep "^_"

_ldap._tcp.example.com. 86400     IN    SRV     0 100 389 server1.example.com.
_ldap._tcp.example.com. 86400     IN    SRV     0 100 389 server2.example.com.
_kerberos._tcp.example.com. 86400 IN    SRV     0 100 88  server1.example.com.
...
```

3. 将父域中的 DNS 委托程序添加到 IdM DNS 域。例如，如果 IdM DNS 域是 **ipa.example.com**，请在 **example.com** 父域中添加一个名称服务器(NS)记录。

重要

每次安装 IdM DNS 服务器时，必须重复此步骤。

4. *可选，但推荐*在副本不可用时，手动将其他 DNS 服务器添加为备份服务器。请参阅 第 33.11.1 节 “设置其他名称服务器”。当新副本是您 IdM 域中的第一个 DNS 服务器时，尤其建议这样做。

5. 另外，如果您正在复制的 IdM 服务器有 Active Directory 信任，请将副本设置为信任代理或信任控制器。详情请参阅 *Windows 集成指南中的* 信任控制器和信任代理。

4.5.4. 使用 CA 安装副本

此流程适用于在客户端以及尚不属于 IdM 域的机器上安装副本。详情请查看 第 4.5 节 “创建副本：简介”。

1. 使用 **--setup-ca** 选项运行 **ipa-replica-install**。

```
[root@replica ~]# ipa-replica-install --setup-ca
```

2. **setup-ca** 选项从初始服务器配置中复制 CA 配置，无论服务器上的 IdM CA 是根 CA 还是从属到外部 CA。

注意

有关支持的 CA 配置的详情，请参考 第 2.3.2 节 “确定要使用的 CA 配置”。

3. 另外，如果您正在复制的 IdM 服务器有 Active Directory 信任，请将副本设置为信任代理或信任控制器。详情请参阅 *Windows 集成指南中的* 信任控制器和信任代理。

4.5.5. 从没有 CA 的服务器安装 Replica

此流程适用于在客户端以及尚不属于 IdM 域的机器上安装副本。详情请查看 第 4.5 节 “创建副本：简介”。

重要

您不能使用自签名第三方服务器证书安装服务器或副本。

1. 运行 **ipa-replica-install**，并通过添加以下选项来提供所需的证书文件：

- **--dirsrv-cert-file**
- **--dirsrv-pin**
- **--http-cert-file**
- **--http-pin**

有关使用这些选项提供的文件的详情，请参考 第 2.3.6 节 “在没有 CA 的情况下安装”。

例如：

```
[root@replica ~]# ipa-replica-install \
   --dirsrv-cert-file /tmp/server.crt \
   --dirsrv-cert-file /tmp/server.key \
   --dirsrv-pin secret \
   --http-cert-file /tmp/server.crt \
   --http-cert-file /tmp/server.key \
   --http-pin secret
```

注意

不要添加 **--ca-cert-file** 选项。**ipa-replica-install** 工具从主服务器自动获取此部分证书信息。

2. 另外，如果您正在复制的 IdM 服务器有 Active Directory 信任，请将副本设置为信任代理或信任控制器。详情请参阅 *Windows 集成指南中的* 信任控制器和信任代理。

4.6. 测试新副本

在创建副本后检查复制是否按预期工作：

1. 在其中一个服务器上创建用户：

```
[admin@server1 ~]$ ipa user-add test_user --first=Test --last=User
```

2. 确保该用户在其它服务器中可见：

```
[admin@server2 ~]$ ipa user-show test_user
```

4.7. 卸载副本

请参阅 第 2.4 节 “卸载 IdM 服务器”。

部分 III. 管理：管理服务器

这部分涵盖了与管理相关的主题，如 **管理身份管理** 域中服务器之间的 **身份管理服务器** 和复制，提供 **身份管理** 拓扑的详细信息，并提供了有关如何更新系统上的 **身份管理** 软件包的说明。另外，这部分解释了如何在影响 **身份管理** 部署时手动备份和恢复 **身份管理** 系统。最后一章详细介绍了不同的内部访问控制机制。

第 5 章 管理 IDM 服务器和服务的基本知识

本章论述了可用于管理 IdM 服务器和服务的身份管理命令行和 UI 工具，包括向 IdM 进行身份验证的方法。

5.1. 启动和停止 IDM 服务器

与 IdM 服务器一起安装多种不同的服务，包括目录服务器、证书颁发机构(CA)、DNS、Kerberos 等。使用 **ipactl** 工具来停止、启动或重启整个 IdM 服务器，以及所有已安装的服务。

启动整个 IdM 服务器：

```
# ipactl start
```

停止整个 IdM 服务器：

```
# ipactl stop
```

重启整个 IdM 服务器：

```
# ipactl restart
```

如果您只想停止、启动或重启单个服务，请使用 **systemctl** 工具，如 系统管理员指南 中所述。例如，在自定义目录服务器行为时使用 **systemctl** 管理单个服务很有用：配置更改需要重启目录服务器实例，但不需要重启所有 IdM 服务。

重要

要重启多个 IdM 域服务，红帽建议使用 **ipactl**。由于与 IdM 服务器一起安装的服务之间的依赖关系，这些服务启动和停止的顺序至关重要。**ipactl** 工具确保服务以适当的顺序启动和停止。

5.2. 使用 KERBEROS 登录 IDM

IdM 使用 Kerberos 协议来支持单点登录。使用 Kerberos，用户只需提供一次正确的用户名和密码，就可以访问 IdM 服务，而系统不需要再次提示输入凭证。

默认情况下，只有作为 IdM 域成员的机器才能使用 Kerberos 向 IdM 进行身份验证。但是，也可以为 Kerberos 身份验证配置外部系统；如需更多信息，请参阅 第 5.4.4 节 “配置外部系统以进行 Kerberos 身份验证到 Web UI”。

使用 kinit

要从命令行登录到 IdM，请使用 **kinit** 工具。

注意

要使用 **kinit**，必须安装 krb5-workstation 软件包。

当不指定用户名时，**kinit** 会在当前登录到本地系统的用户的用户名下登录到 IdM。例如，如果您在本地系统中以 **local_user** 身份登录，运行 **kinit** 会尝试以 **local_user** IdM 用户身份进行身份验证：

```
[local_user@server ~]$ kinit
Password for local_user@EXAMPLE.COM:
```

注意

如果本地用户的用户名与 IdM 中的任何用户条目都不匹配，身份验证尝试会失败。

要以不同的 IdM 用户身份登录，请将所需的用户名作为参数传递给 **kinit** 工具。例如，要以 **admin** 用户身份登录：

```
[local_user@server ~]$ kinit admin
Password for admin@EXAMPLE.COM:
```

自动获取 Kerberos Tickets

pam_krb5 可插拔验证模块(PAM)和 SSSD 可以配置为在成功登录 IdM 客户端机器上的桌面环境后自动获取用户的 TGT。这样可确保登录后，用户不需要运行 **kinit**。

在 SSSD 中配置了 IdM 作为身份和身份验证供应商的 IdM 系统中，SSSD 在用户使用对应的 Kerberos 主体名称登录后自动获取 TGT。

有关配置 **pam_krb5** 的详情，请参考 pam_krb5(8) man page。有关 PAM 的常规信息，请参阅 系统级身份验证指南。

存储多个 Kerberos 票据

默认情况下，Kerberos 只会将每个登录用户的一个票据存储在凭据缓存中。每当用户运行 **kinit** 时，Kerberos 会使用新的票据覆盖当前存储的票据。例如，如果您使用 **kinit** 以 **user_A** 进行身份验证，则 **user_A** 的票据将在以 **user_B** 身份再次进行身份验证后丢失。

要获取并存储用户的另一个 TGT，请设置不同的凭据缓存，这样可确保上一个缓存的内容不会被覆盖。您可以通过以下两种方式之一完成此操作：

- 运行 **export KRB5CCNAME=*path_to_different_cache*** 命令，然后使用 **kinit** 获取 ticket。
- 运行 **kinit -c *path_to_different_cache*** 命令，然后重置 **KRB5CCNAME** 变量。

恢复存储在默认凭证缓存中的原始 TGT：

1. 运行 **kdestroy** 命令。
2. 使用 **unset $KRB5CCNAME** 命令恢复默认凭证缓存位置。

检查当前登录用户

要验证当前存储并用于身份验证的 TGT，请使用 **klist** 工具来列出缓存的票据。在以下示例中，缓存包含 **user_A** 的票据，这意味着当前只允许 **user_A** 访问 IdM 服务：

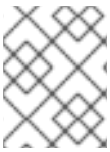
```
$ klist
Ticket cache: KEYRING:persistent:0:0
Default principal: user_A@EXAMPLE.COM

Valid starting     Expires            Service principal
11/10/2015 08:35:45  11/10/2015 18:35:45  krbtgt/EXAMPLE.COM@EXAMPLE.COM
```

5.3. IDM 命令行实用程序

IdM 的基本命令行脚本名为 **ipa**。**ipa** 脚本是多个子命令的父脚本。这些子命令随后用于管理 IdM。例如，**ipa user-add** 命令添加一个新用户：

```
$ ipa user-add user_name
```

与 UI 中的管理相比，命令行管理具有某些优势；例如，命令行实用程序允许以一致的方式自动执行和重复执行管理任务，而无需人工干预。此外，虽然大多数管理操作都从命令行和 Web UI 中可用，但某些任务只能从命令行执行。



注意

本节仅提供 **ipa** 子命令的一般信息。其它专用于 IdM 管理区域的部分提供了更多信息。例如，有关使用 **ipa** 子命令管理用户条目的详情，请参考 第 11 章 *管理用户帐户*。

5.3.1. 获取 ipa 命令的帮助信息

ipa 脚本可以显示关于特定子命令的帮助信息： *主题*。要显示可用主题列表，请使用 **ipa help topics** 命令：

```
$ ipa help topics

automember          Auto Membership Rule.
automount           Automount
caacl               Manage CA ACL rules.
...
```

要显示特定主题的帮助信息，请使用 **ipa help *topic_name*** 命令。例如，显示有关 **自动成员** 主题的信息：

```
$ ipa help automember

Auto Membership Rule.

Bring clarity to the membership of hosts and users by configuring inclusive
or exclusive regex patterns, you can automatically assign a new entries into
a group or hostgroup based upon attribute information.

...

EXAMPLES:

 Add the initial group or hostgroup:
   ipa hostgroup-add --desc="Web Servers" webservers
   ipa group-add --desc="Developers" devel
...
```

ipa 脚本还可以显示可用的 **ipa** 命令列表。要做到这一点，请使用 **ipa help 命令** ：

```
$ ipa help commands
automember-add                  Add an automember rule.
automember-add-condition            Add conditions to an automember rule.
...
```

有关单个 **ipa** 命令的详细帮助，请在命令中添加 **--help** 选项。例如：

```
$ ipa automember-add --help

Usage: ipa [global-options] automember-add AUTOMEMBER-RULE [options]

Add an automember rule.
Options:
  -h, --help            show this help message and exit
  --desc=STR            A description of this auto member rule
...
```

有关 **ipa** 工具程序的详情请参考 ipa(1) man page。

5.3.2. 设置值列表

IdM 在列表中存储条目属性。例如：

```
ipaUserSearchFields: uid,givenname,sn,telephonenumber,ou,title
```

对属性列表的任何更新都会覆盖上一个列表。例如，尝试通过仅指定此属性来添加单个属性，将之前定义的整个列表替换为单个新属性。因此，在更改属性列表时，您必须指定整个更新列表。

IdM 支持以下提供属性列表的方法：

- 在同一命令调用中多次使用相同的命令行参数。例如：

  ```
  $ ipa permission-add --permissions=read --permissions=write --permissions=delete
  ```

- 将列表括在大括号中，以允许 shell 进行扩展。例如：

  ```
  $ ipa permission-add --permissions={read,write,delete}
  ```

5.3.3. 使用特殊字符

当在包含特殊字符的 **ipa** 命令中传递命令行参数时，如 angle 括号(< 和 >)、ampersand (&)、星号或垂直栏(|)，您必须使用反斜杠(\)转义这些字符。例如，要转义星号(*)：

```
$ ipa certprofile-show certificate_profile --out=exported\*profile.cfg
```

包含未转义特殊字符的命令无法按预期工作，因为 shell 无法正确解析这些字符。

5.3.4. 搜索 IdM 条目

列出 IdM 条目

使用 **ipa114-find** 命令搜索特定类型的 IdM 条目。例如：

- 要列出所有用户，请执行以下操作：

  ```
  $ ipa user-find
  ---------------
  4 users matched
  ---------------
    ...
  ```

- 列出其指定属性包含 ***关键字的用户组*** ：

```
$ ipa group-find keyword
----------------
2 groups matched
----------------
  ...
```

要配置属性 IdM 搜索用户和用户组，请参阅 第 13.5 节 “为用户和组设置搜索属性”。

在搜索用户组时，您还可以将搜索结果限制为包含特定用户的组：

```
$ ipa group-find --user=user_name
```

您还可以搜索不包含特定用户的组：

```
$ ipa group-find --no-user=user_name
```

显示 entry Enicular Entry 的详情
使用 **ipa *-show** 命令显示特定 IdM 条目的详情。例如：

```
$ ipa host-show server.example.com
 Host name: server.example.com
 Principal name: host/server.example.com@EXAMPLE.COM
 ...
```

5.3.4.1. 调整搜索大小和时间限制

某些搜索结果（如查看用户列表）可能会返回大量条目。通过调整这些搜索操作，您可以在运行 **ipa thefind** 命令时提高整体服务器性能，如 **ipa user-find**，以及在 web UI 中显示对应的列表时。

搜索大小限制：

- 定义从客户端、IdM 命令行工具或 IdM Web UI 发送到服务器的请求的最大条目数。
- 默认值：100 个条目.

搜索时间限制：

- 定义服务器等待搜索运行的最长时间。搜索达到此限制后，服务器将停止搜索并返回在该时间发现的条目。
- 默认值：2 秒.

如果您将值设为 **-1**，IdM 在搜索时不会应用任何限制。

重要

如果设置的搜索大小或时间限制太大，则可能会对服务器性能造成负面影响。

Web UI：调整搜索大小和时间限制
为所有查询在全局范围内调整限制：

1. 选择 **IPA Server → Configuration**。

2. 在**搜索选项**区域中设置所需的值。

3. 点页面顶部的 **Save**。

命令行：调整搜索大小和时间限制
要在全局范围内调整所有查询的限制，请使用 **ipa config-mod** 命令，并添加 **--searchrecordslimit** 和 **--searchtimelimit** 选项。例如：

```
$ ipa config-mod --searchrecordslimit=500 --searchtimelimit=5
```

在命令行中，您还可以仅针对特定查询调整限值。为此，可在命令中添加 **--sizelimit** 或 **--timelimit** 选项。例如：

```
$ ipa user-find --sizelimit=200 --timelimit=120
```

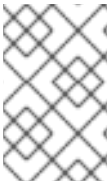
重要

请注意，使用带有 --searchrecordslimit 或 **--searchtimelimit** 选项的 **ipa config-mod** 命令调整大小或时间限制会影响 **ipa**命令返回的条目数量，如 **ipa user-find**。

除了这些限制外，还会考虑在 Directory 服务器级别上配置的设置，并且可能会实施更严格的限制。有关目录服务器限制的更多信息，请参阅 *红帽目录服务器管理指南*。

5.4. THE IDM WEB UI

Identity Management Web UI 是一个用于 IdM 管理的 Web 应用。它具有 **ipa** 命令行工具的大部分功能。因此，用户可以选择是否要从 UI 管理 IdM，还是从命令行管理 IdM。



注意

供登录用户使用的管理操作取决于用户的访问权限。对于具有管理特权的 **admin** 用户和其他用户，所有管理任务都可用。对于普通用户，只能使用与其自身用户帐户相关的一组有限操作。

5.4.1. 支持的 Web 浏览器

身份管理支持以下浏览器来连接到 Web UI：

- Mozilla Firefox 38 及更新的版本
- Google Chrome 46 及更新的版本

5.4.2. 访问 Web UI 和身份验证

Web UI 可以从 IdM 服务器和客户端机器以及 IdM 域外的计算机访问。但是，要从非域机器访问 UI，您必须首先将非 IdM 系统配置为能够连接到 IdM Kerberos 域；如需更多详情，请参阅 第 5.4.4 节 “配置外部系统以进行 Kerberos 身份验证到 Web UI”。

5.4.2.1. 访问 Web UI

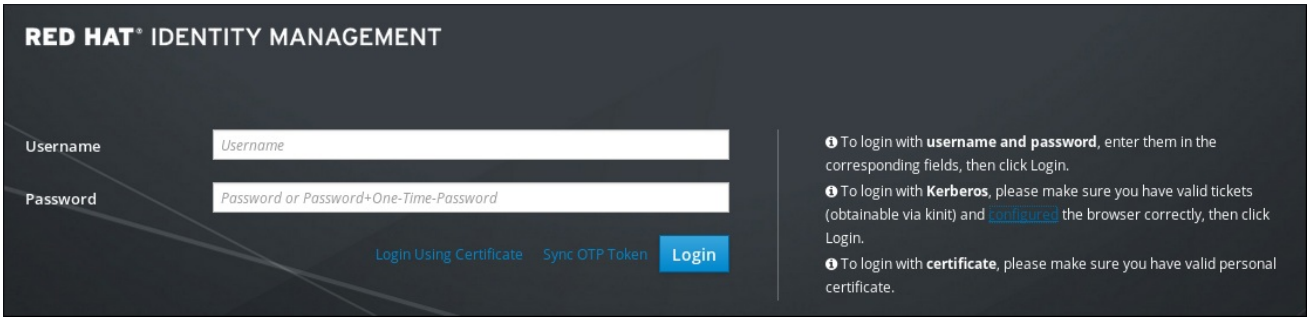
要访问 Web UI，在浏览器地址栏中输入 IdM 服务器 URL：

```
https://server.example.com
```

-

这会在浏览器中打开 IdM Web UI 登录屏幕。

图 5.1. Web UI 登录屏幕



5.4.2.2. 可用的登录方法

用户可以通过以下方式验证 Web UI：

带有活跃的 Kerberos ticket

如果用户有通过 **kinit** 工具获取的有效 TGT，请单击 **Login** 会自动验证用户。请注意，浏览器必须配置正确，以支持 Kerberos 身份验证。

有关获取 Kerberos TGT 的详情请参考 第 5.2 节 “使用 Kerberos 登录 IdM”。有关配置浏览器的详情请参考 第 5.4.3 节 “为 Kerberos 身份验证配置浏览器”。

通过提供用户名和密码

要使用用户名和密码进行身份验证，请在 Web UI 登录屏幕中输入用户名和密码。

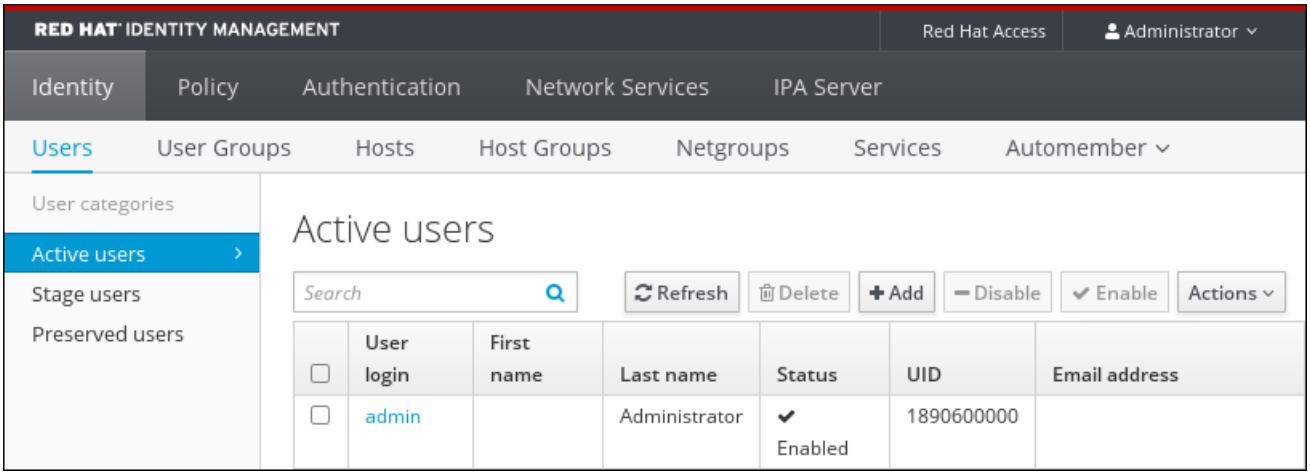
IdM 还支持一次性密码(OTP)身份验证。如需更多信息，请参阅 第 22.3 节 “一次性密码”。

使用智能卡

如需更多信息，请参阅 第 23.6 节 “使用智能卡验证身份管理 Web UI”。

用户成功进行身份验证后，会打开 IdM 管理窗口。

图 5.2. IdM Web UI 布局



5.4.2.3. Web UI 会话长度

当用户使用用户名和密码登录到 IdM Web UI 时，会话长度与在登录操作过程中获得的 Kerberos 票据的过期期限相同。

5.4.2.4. 以 AD 用户身份向 IdM Web UI 进行身份验证

Active Directory(AD)用户可以使用其用户名和密码登录 IdM Web UI。在 Web UI 中，AD 用户只能执行与其自己的用户帐户相关的一组有限操作，这与能够执行与其管理权限相关的管理操作的 IdM 用户不同。

要为 AD 用户启用 Web UI 登录，IdM 管理员必须为 Default Trust View 中的每个 AD 用户定义一个 ID 覆盖。例如：

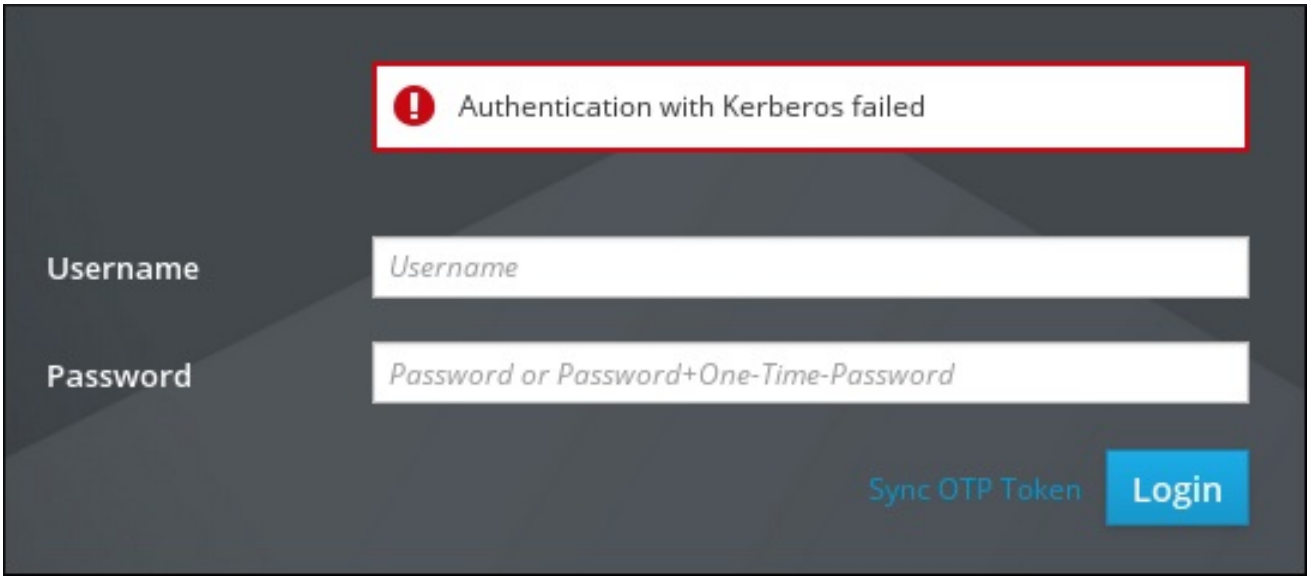
```
[admin@server ~]$ ipa idoverrideuser-add 'Default Trust View' ad_user@ad.example.com
```

有关 AD 中 ID 视图的详情，请参阅 *Windows 集成指南中的* 在 Active Directory 环境中使用 ID 视图 。

5.4.3. 为 Kerberos 身份验证配置浏览器

要使用 Kerberos 凭证启用身份验证，您必须将浏览器配置为支持 Kerberos 协商来访问 IdM 域。请注意，如果您的浏览器没有为 Kerberos 身份验证正确配置，则在点 IdM Web UI 登录屏幕上的 **Login** 后会出现错误消息。

图 5.3. Kerberos 身份验证错误



您可以通过三种方法为 Kerberos 身份验证配置浏览器：

- 从 IdM Web UI 自动进行。这个选项仅适用于 Firefox。详情请查看 “Web UI 中的自动 Firefox 配置”一节。
- 在 IdM 客户端安装过程中自动从命令行执行。这个选项仅适用于 Firefox。详情请查看 “从命令行自动配置 Firefox”一节。
- 在 Firefox 配置设置中手动使用。此选项适用于所有支持的浏览器。详情请查看 “手动浏览器配置”一节。

注意

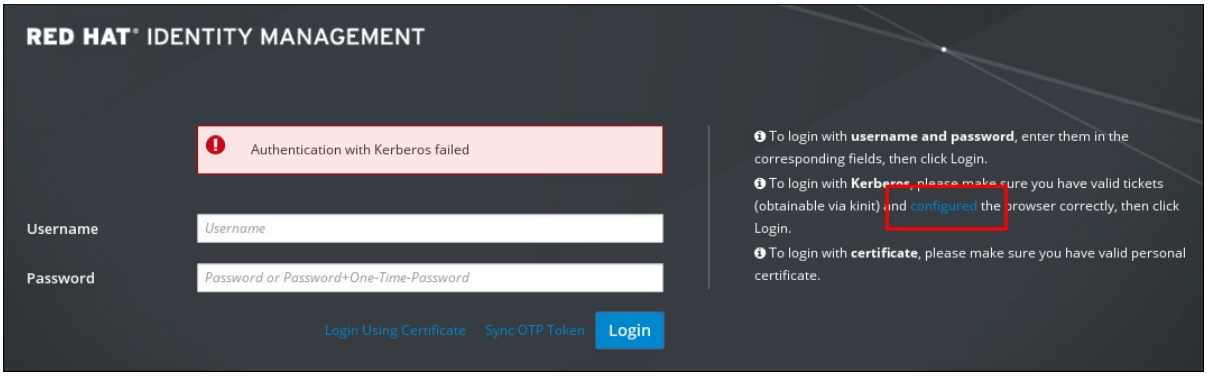
系统级身份验证指南包括对 Firefox Kerberos 配置进行故障排除。如果 Kerberos 身份验证无法正常工作，请参阅此故障排除指南以了解更多信息。

Web UI 中的自动 Firefox 配置

从 IdM Web UI 自动配置 Firefox：

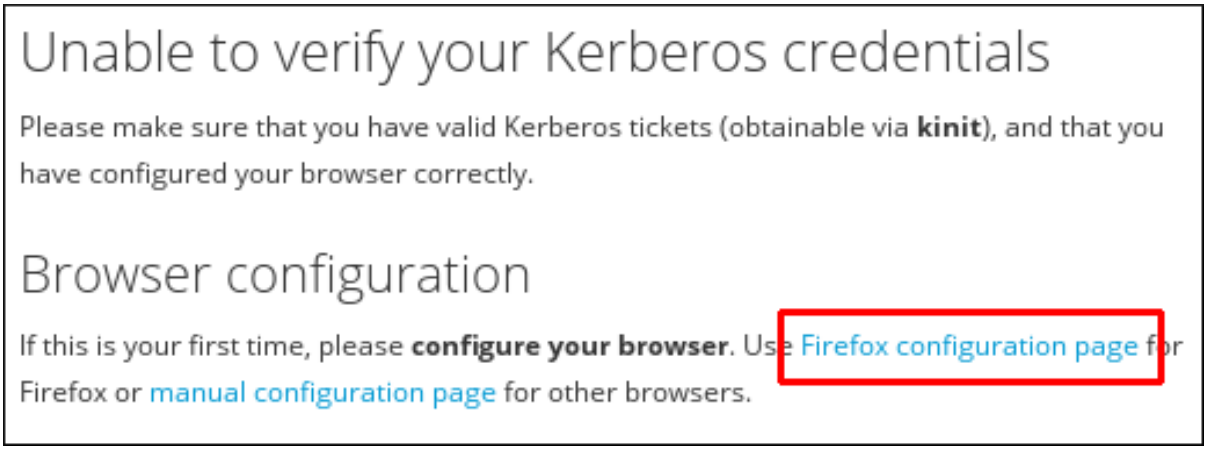
1. 在 Web UI 登录屏幕上，单击浏览器配置的链接。

图 5.4. 在 Web UI 中配置浏览器的链接



2. 选择 Firefox 配置的链接，以打开 Firefox 配置页面。

图 5.5. 指向 Firefox 配置页面的链接



3. 按照 Firefox 配置页面上的步骤操作。

从命令行自动配置 Firefox

在 IdM 客户端安装过程中，可以从命令行配置 Firefox。要做到这一点，在使用 **ipa-client-install** 工具安装 IdM 客户端时使用 **--configure-firefox** 选项：

```
# ipa-client-install --configure-firefox
```

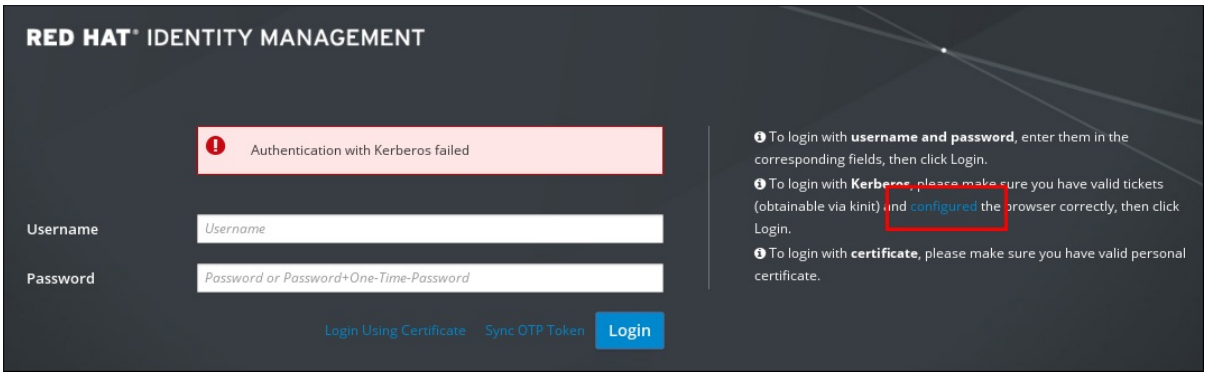
configure **-firefox** 选项创建一个全局配置文件，该文件具有默认 Firefox 设置，可启用单点登录(SSO)的 Kerberos。

手动浏览器配置

手动配置浏览器：

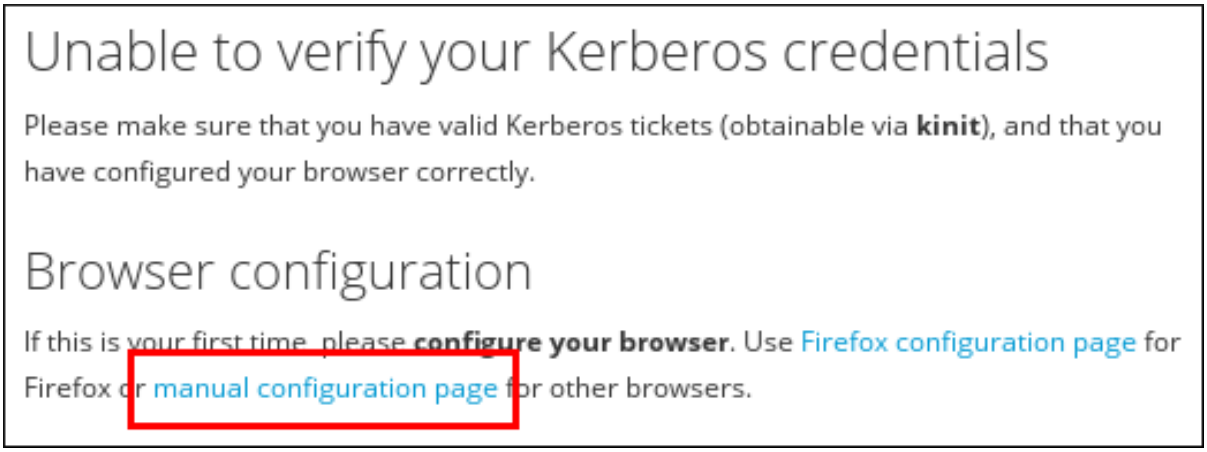
1. 在 Web UI 登录屏幕上，单击浏览器配置的链接。

图 5.6. 在 Web UI 中配置浏览器的链接



2. 选择 链接以进行手动浏览器配置。

图 5.7. 手动配置页面链接



3. 查看相关说明以配置浏览器并按照步骤操作。

5.4.4. 配置外部系统以进行 Kerberos 身份验证到 Web UI

要从不属于 IdM 域的系统启用 Web UI 的 Kerberos 身份验证，您必须在外部机器上定义 IdM 特定的 Kerberos 配置文件。当您的基础架构包含多个域或重叠域时，在外部系统上启用 Kerberos 身份验证非常有用。

创建 Kerberos 配置文件：

1. 将 **/etc/krb5.conf** 文件从 IdM 服务器复制到外部机器。例如：

```
# scp /etc/krb5.conf root@externalmachine.example.com:/etc/krb5_ipa.conf
```

警告

不要覆盖外部计算机上的现有的 **krb5.conf** 文件。

2. 在外部机器上，将终端会话设置为使用复制的 IdM Kerberos 配置文件：

```
$ export KRB5_CONFIG=/etc/krb5_ipa.conf
```

3. 在外部机器上配置浏览器，如 第 5.4.3 节 “为 Kerberos 身份验证配置浏览器” 所述。

外部系统上的用户现在可以使用 **kinit** 工具对 IdM 服务器域进行身份验证。

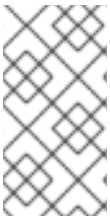
5.4.5. Web UI 中的代理服务器和端口转发

使用代理服务器访问 Web UI 不需要在 IdM 中进行任何额外的配置。

IdM 服务器不支持端口转发。但是，由于可以使用代理服务器，因此可以使用通过 OpenSSH 和 SOCKS 选项的代理转发来配置与端口转发类似的操作。这可以使用 **ssh** 工具的 **-D** 选项进行配置；有关使用 **-D** 的更多信息，请参阅 ssh(1) man page。

第 6 章 管理复制拓扑

本章论述了如何管理身份管理(IdM)域中服务器之间的复制。



注意

本章论述了 Red Hat Enterprise Linux 7.3 中引入的简化拓扑管理。这些步骤需要域级别 1（请参阅 第 7 章 *显示和提升域级别*）。

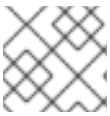
有关在域级别 0 管理拓扑的文档，请参阅 第 D.3 节 “管理副本和复制协议”。

有关安装初始副本以及复制的基本信息的详情，请参考 第 4 章 *安装和卸载身份管理副本*。

6.1. 解释复制协议、拓扑后缀和拓扑片段

复制协议
存储在 IdM 服务器上的数据会根据复制协议复制：当两台服务器配置了复制协议时，它们将共享其数据。

复制协议始终为现实：数据从第一个副本复制到另一个副本，另一个副本复制到第一个副本。



注意

详情请查看 第 4.1 节 “解释 IdM 副本”。

拓扑后缀
*拓扑后缀*存储复制的数据。IdM 支持两种类型的拓扑后缀：**domain** 和 **ca**。每个后缀代表一个单独的后端，即一个单独的复制拓扑。

配置复制协议时，它会在两个不同的服务器上加入同一类型的两个拓扑后缀。

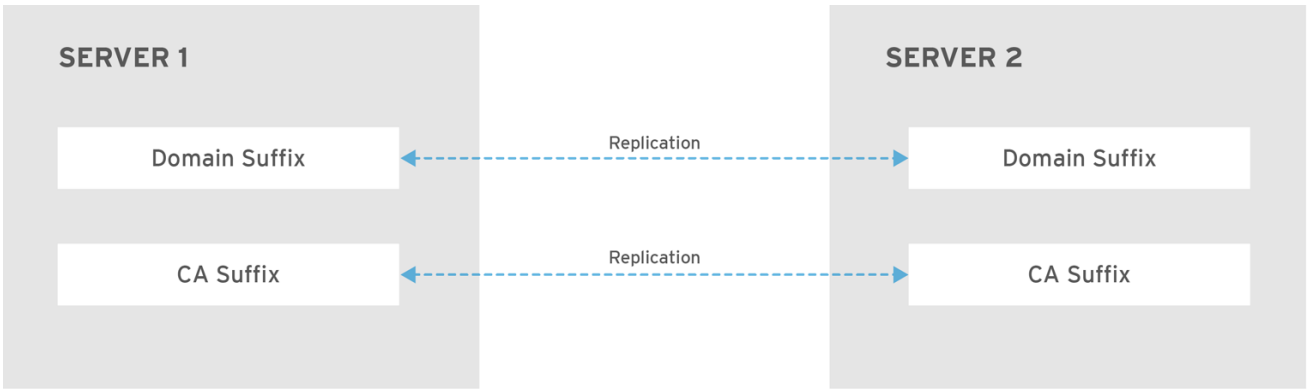
域 后缀：dc=*示例*,dc=*com*
域 后缀包含与域相关的所有数据。

当两个副本在其 **域** 后缀之间具有复制协议时，它们共享目录数据，如用户、组和策略。

ca 后缀：o=ipaca
ca 后缀包含证书系统组件的数据。它仅存在于安装有证书颁发机构 (CA) 的服务器上。

当两个副本在其 **ca** 后缀之间具有复制协议时，它们会共享证书数据。

图 6.1. 拓扑后缀



在安装新副本时，**ipa-replica-install** 脚本会在两台服务器之间设置初始拓扑片段。

例 6.1. 查看拓扑缓冲

ipa topologysuffix-find 命令显示拓扑后缀列表：

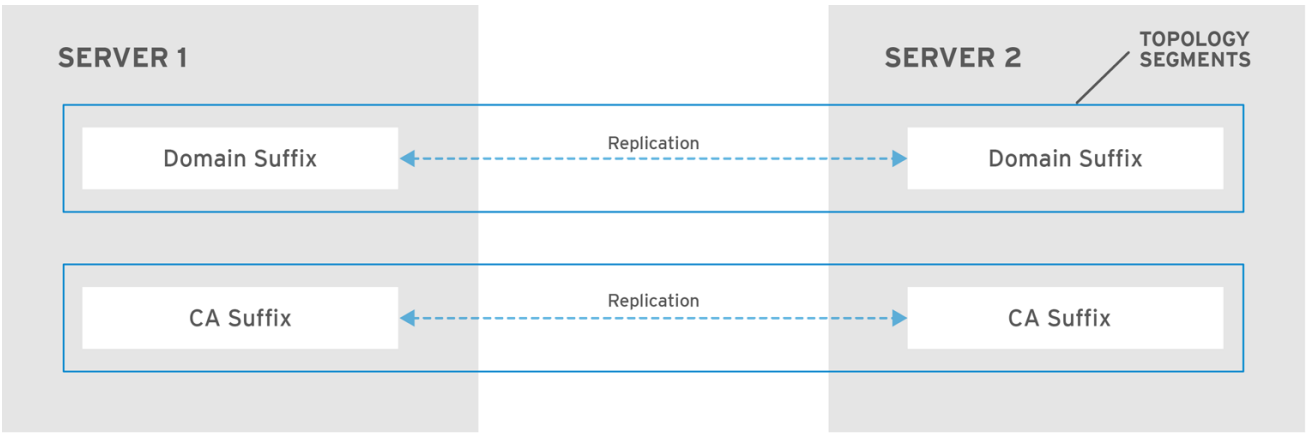
```
$ ipa topologysuffix-find
----------------------------
2 topology suffixes matched
----------------------------
  Suffix name: ca
  Managed LDAP suffix DN: o=ipaca

  Suffix name: domain
  Managed LDAP suffix DN: dc=example,dc=com
----------------------------
Number of entries returned 2
----------------------------
```

拓扑片段

当两个副本的后缀之间有复制协议时，后缀会形成 *拓扑片段*。每个拓扑片段由一个 *左侧节点* 和一个 *右节点* 组成。节点代表在复制协议中加入的服务器。

IdM 中的拓扑片段始终是双向的。每个片段代表两种复制协议：从服务器 A 到服务器 B，以及从服务器 B 到服务器 A。因此，数据会双向复制。

图 6.2. **拓扑片段**



例 6.2. 查看拓扑分段

ipa topologysegment-find 命令显示为域或 CA 后缀配置的当前拓扑片段。例如，对于域后缀：

```
$ ipa topologysegment-find
Suffix name: domain
----------------
1 segment matched
----------------
  Segment name: server1.example.com-to-server2.example.com
  Left node: server1.example.com
  Right node: server2.example.com
  Connectivity: both
----------------------------
Number of entries returned 1
----------------------------
```

在本例中，域相关的数据仅在两个服务器之间复制： **server1.example.com** 和 **server1.example.com**。

要只显示特定片段的详情，请使用 **ipa topologysegment-show** 命令：

```
$ ipa topologysegment-show
Suffix name: domain
Segment name: server1.example.com-to-server2.example.com
  Segment name: server1.example.com-to-server2.example.com
  Left node: server1.example.com
  Right node: server2.example.com
  Connectivity: both
```

6.2. WEB UI：使用拓扑图形管理复制拓扑

访问**拓扑**图**形**
Web UI 中的拓扑图显示域中服务器之间的关系：

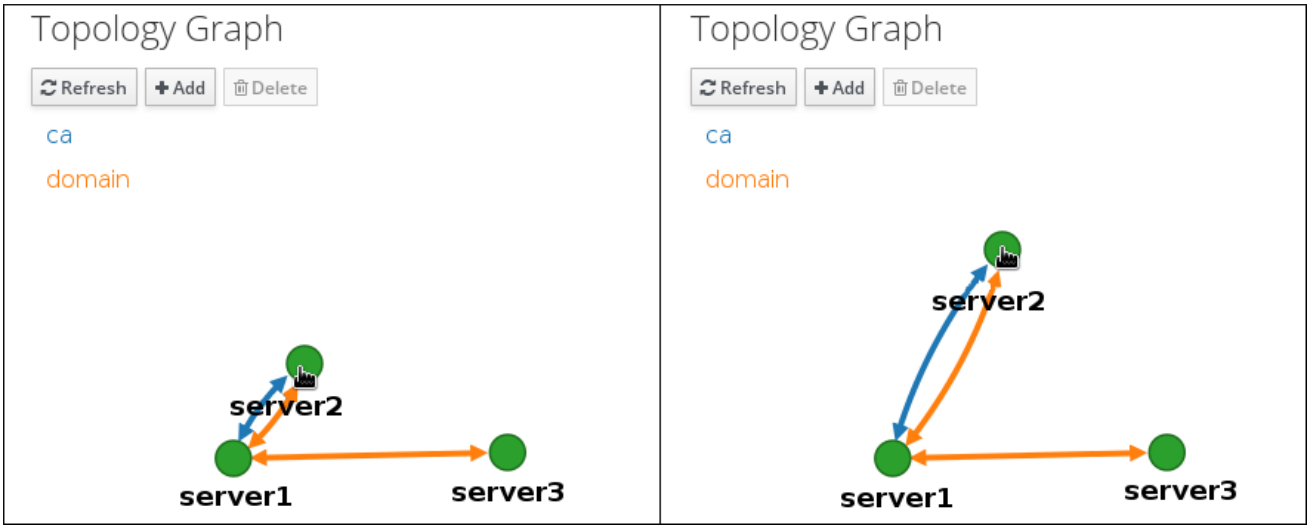
1. 选择 **IPA Server → Topology → Topology Graph**。

2. 如果您对拓扑进行任何没有立即反映在图形中的更改，点 **Refresh**。

自定义拓扑视图

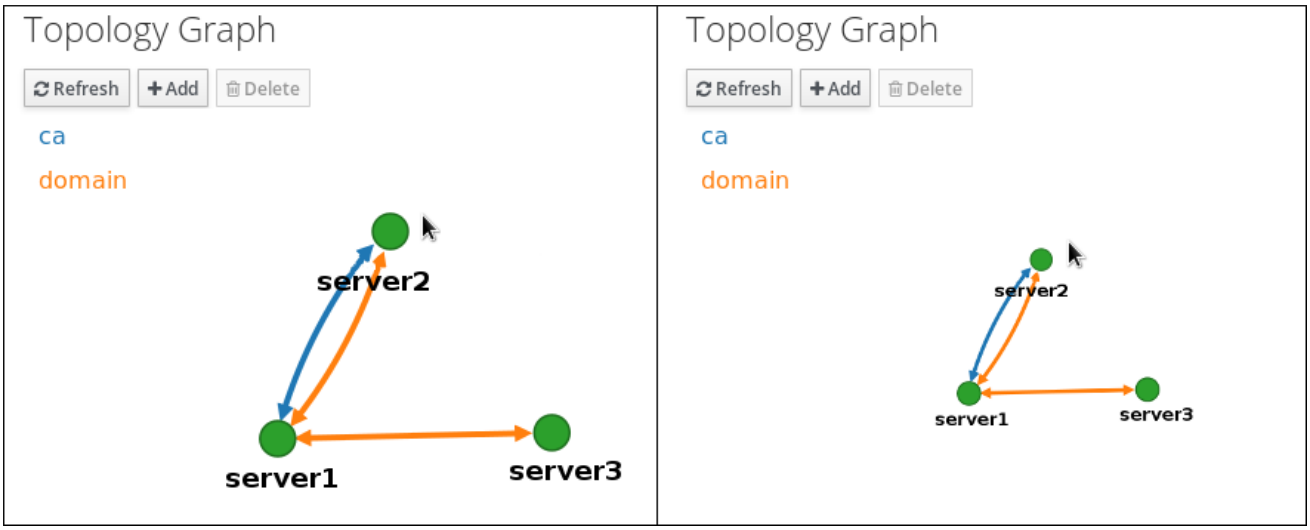
您可以通过拖动鼠标来移动独立拓扑节点：

图 6.3. 移动拓扑图形节点



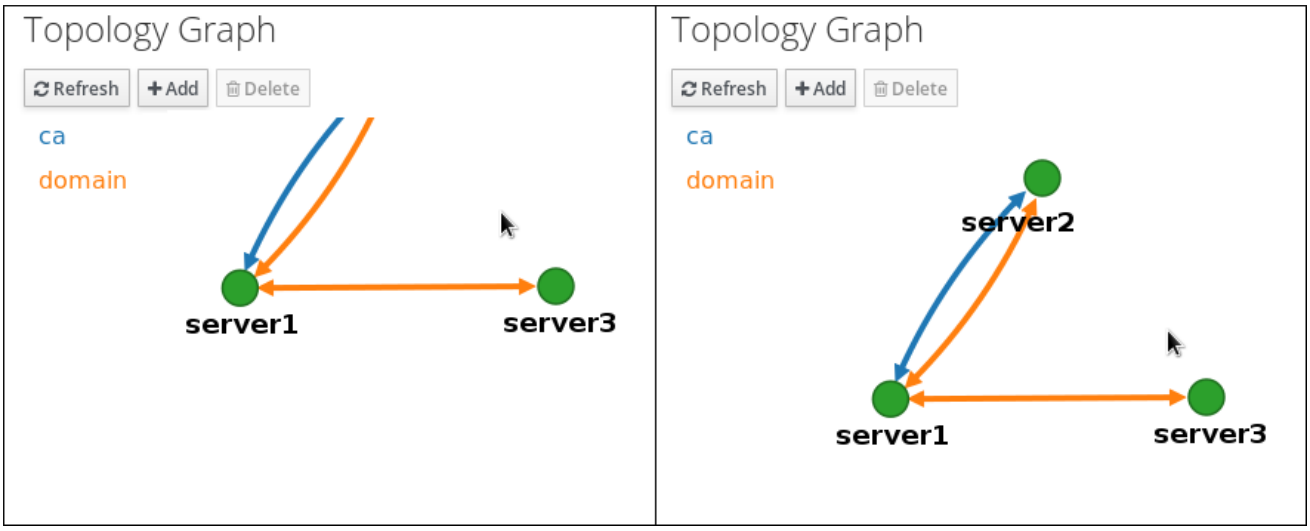
您可以使用鼠标 wheel 缩放并缩小拓扑图形：

图 6.4. 缩放拓扑图形



您可以通过按鼠标左键移动拓扑图形的 Canvas：

图 6.5. 移动拓扑图形 Canvas



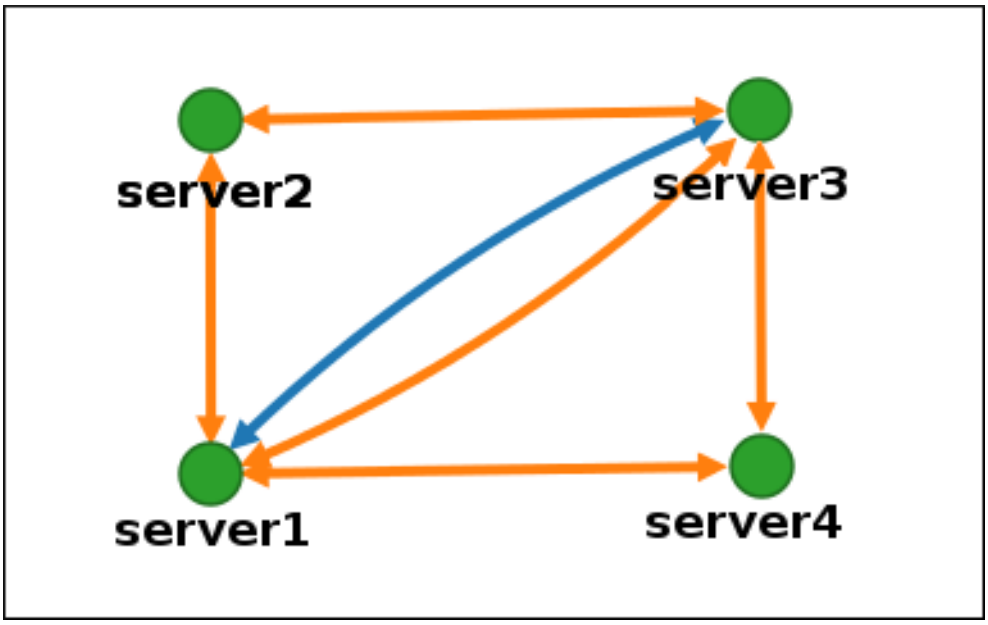
解释拓扑图形

加入域复制协议中的服务器通过灰色箭头连接。加入 CA 复制协议的服务器通过蓝色箭头连接。

拓扑图示例：推荐的拓扑图示例

图 6.6 "建议的拓扑示例" 显示四个服务器的一个可能推荐的拓扑之一：每个服务器至少连接到两个其他服务器，多个服务器是 CA 主服务器。

图 6.6. 建议的拓扑示例

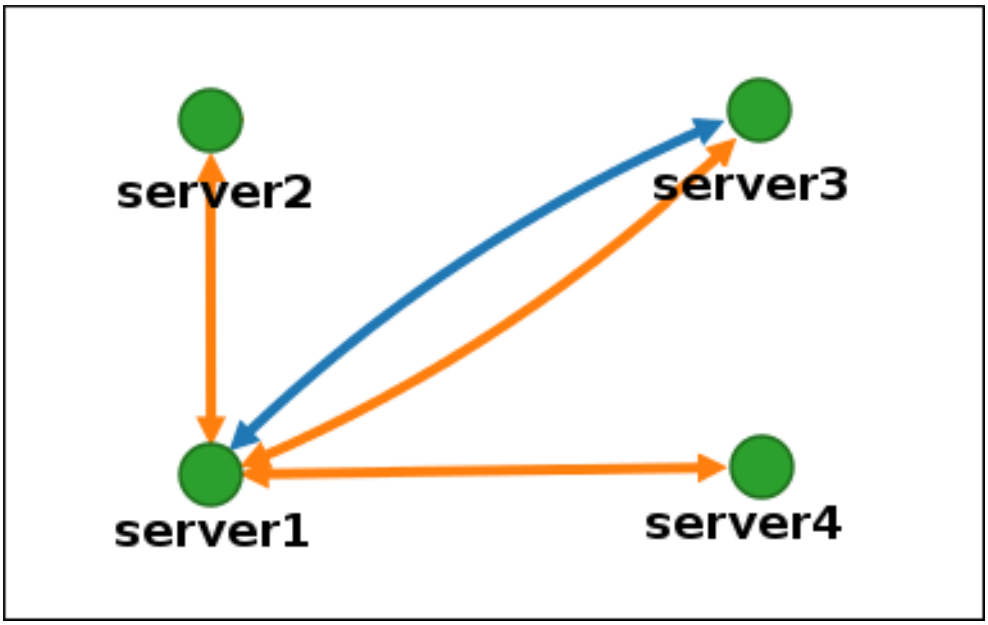


拓扑图示例：禁用拓扑

在 图 6.7 "不鼓励的拓扑示例：单点故障" 中，**server1** 是一个单点故障。所有其他服务器都与此服务器具有复制协议，但与其他任何服务器均不具有复制协议。因此，如果 **server1** 出现故障，所有其他服务器将被隔离。

避免创建如下所示的拓扑：

图 6.7. 不鼓励的拓扑示例：单点故障

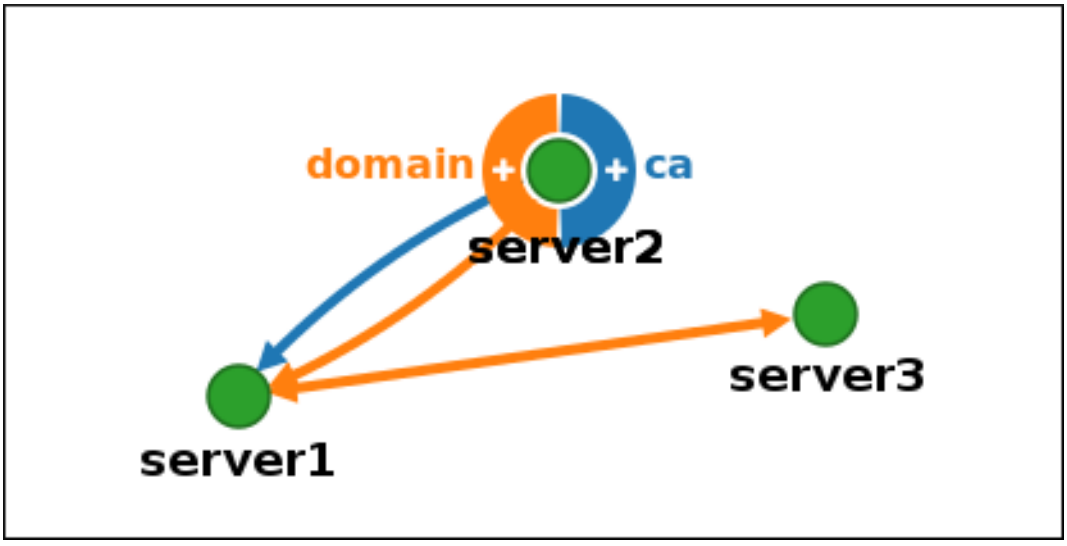


有关拓扑建议的详情请参考 第 4.2 节 “Replicas 的部署注意事项”。

6.2.1. 在两个服务器之间设置复制

1. 在拓扑图中，将鼠标悬停在其中一个服务器节点上。

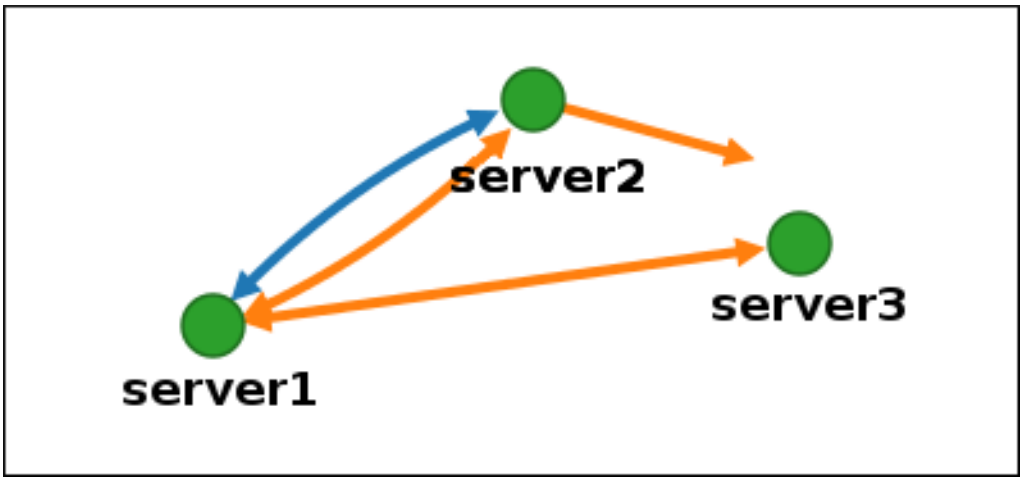
   图 6.8. 域或 CA 选项

2. 点击 **域** 或圆圈的 **ca** 部分，具体取决于您要创建的拓扑网类型。

3. 在鼠标指针下会出现代表新复制协议的新箭头。将鼠标移到其他服务器节点，然后单击该节点。

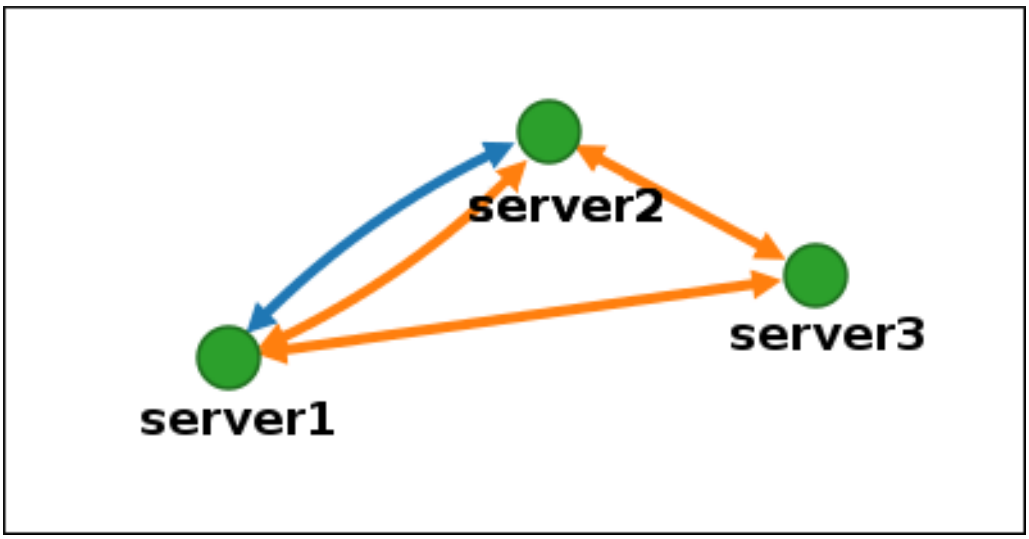
图 6.9. 创建新分段



4. 在 **Add Topology Segment** 窗口中，单击 **Add** 以确认新段的属性。

IdM 在两台服务器之间创建一个新的拓扑片段，它们将它们加入到复制协议中。拓扑图现在显示更新的复制拓扑：

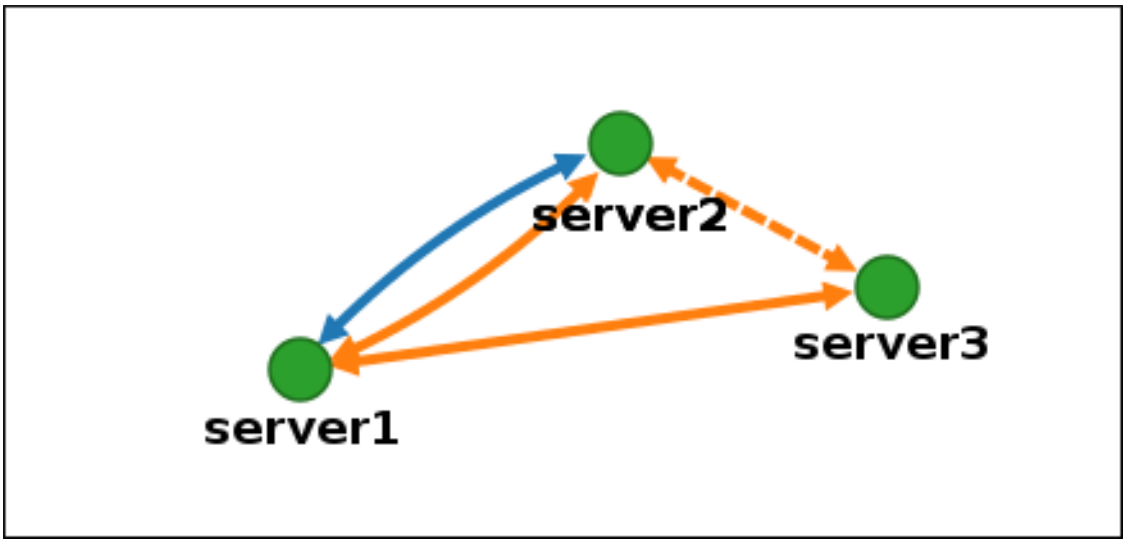
图 6.10. 新片段已创建



6.2.2. 停止两个服务器之间的复制

1. 单击代表您要删除的复制协议的箭头。这突出显示了箭头。

图 6.11. 拓扑片段突出显示

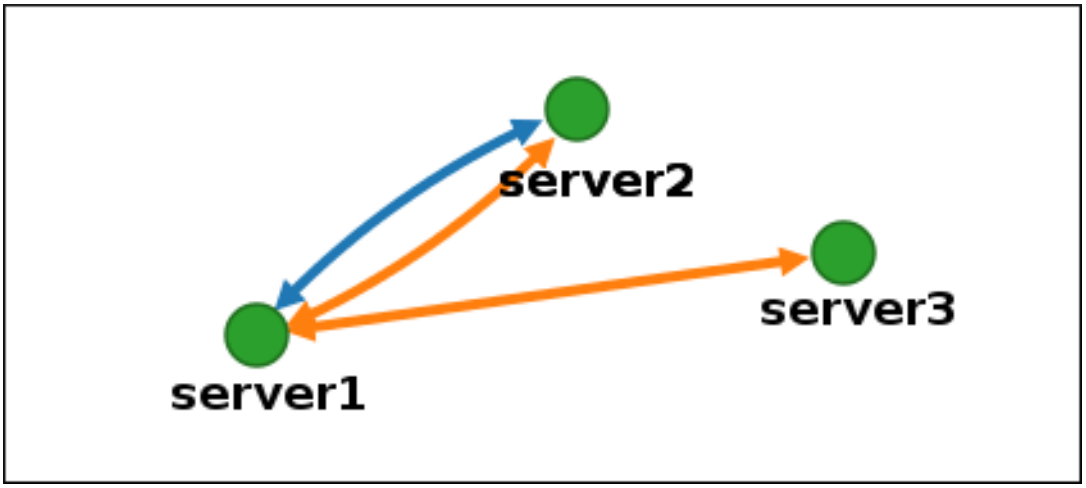


2. 单击 **Delete**。

3. 在 **Confirmation** 窗口中，单击 **OK**。

IdM 删除两个服务器之间的拓扑片段，这将删除其复制协议。拓扑图现在显示更新的复制拓扑：

图 6.12. 拓扑片段已删除



6.3. 命令行：使用 IPA TOPOLOGY ITEM 命令管理拓扑

6.3.1. 获得拓扑管理命令的帮助

显示用于管理复制拓扑的所有命令：

```
$ ipa help topology
```

要显示特定命令的详细帮助，请使用 **--help** 选项运行它：

```
$ ipa topologysuffix-show --help
```

6.3.2. 在两个服务器之间设置复制

1. 使用 **ipa topologysegment-add** 命令创建两个服务器的拓扑网段。出现提示时，提供：

   - 所需的拓扑后缀： **domain** 或 **ca**

     注意

     如果要在 **ca** 后缀之间创建网段，则两个服务器都必须安装 CA。请参阅 第 26.8 节 “在现有 IdM 域中安装 CA”。

   - 左节点和右侧节点，代表两个服务器

   - （可选）片段的自定义名称

   例如：

   ```
   $ ipa topologysegment-add
   Suffix name: domain
   Left node: server1.example.com
   Right node: server2.example.com
   Segment name [server1.example.com-to-server2.example.com]: new_segment
   ----------------------------
   Added segment "new_segment"
   ----------------------------
     Segment name: new_segment
     Left node: server1.example.com
     Right node: server2.example.com
     Connectivity: both
   ```

   添加新段将加入复制协议中的服务器。

2. *可选。*使用 **ipa topologysegment-show** 命令验证是否已配置新网段。

   ```
   $ ipa topologysegment-show
   Suffix name: domain
   Segment name: new_segment
     Segment name: new_segment
     Left node: server1.example.com
     Right node: server2.example.com
     Connectivity: both
   ```

6.3.3. 停止两个服务器之间的复制

1. 要停止复制，您必须删除服务器之间的对应复制片段。要做到这一点，您需要知道片段名称。

   如果您不知道名称，请使用 **ipa topologysegment-find** 命令显示所有片段，并在输出中找到所需的片段。出现提示时，请提供所需的拓扑后缀： **domain** 或 **ca**。例如：

   ```
   $ ipa topologysegment-find
   Suffix name: domain
   ------------------
   8 segments matched
   ------------------
     Segment name: new_segment
     Left node: server1.example.com
   ```

```
  Right node: server2.example.com
  Connectivity: both

...

----------------------------
Number of entries returned 8
----------------------------
```

2. 使用 **ipa topologysegment-del** 命令删除加入两台服务器的拓扑网段。

```
$ ipa topologysegment-del
Suffix name: domain
Segment name: new_segment
----------------------------
Deleted segment "new_segment"
----------------------------
```

删除片段会删除复制协议。

3. *可选*。使用 **ipa topologysegment-find** 命令验证网段是否不再列出。

```
$ ipa topologysegment-find
Suffix name: domain
------------------
7 segments matched
------------------
  Segment name: server2.example.com-to-server3.example.com
  Left node: server2.example.com
  Right node: server3.example.com
  Connectivity: both

...

----------------------------
Number of entries returned 7
----------------------------
```

6.4. 从拓扑中删除服务器

如果适用，IdM 不允许从拓扑中删除服务器：

- 移除的服务器是唯一连接到其他拓扑其余服务器的服务器；这会导致其他服务器被隔离，这是不允许的
- 正在删除的服务器是您最后一个 CA 或 DNS 服务器

在这些情况下，尝试会失败并显示错误。例如，在命令行中：

```
$ ipa server-del
Server name: server1.example.com
Removing server1.example.com from replication topology, please wait...
ipa: ERROR: Server removal aborted:
```

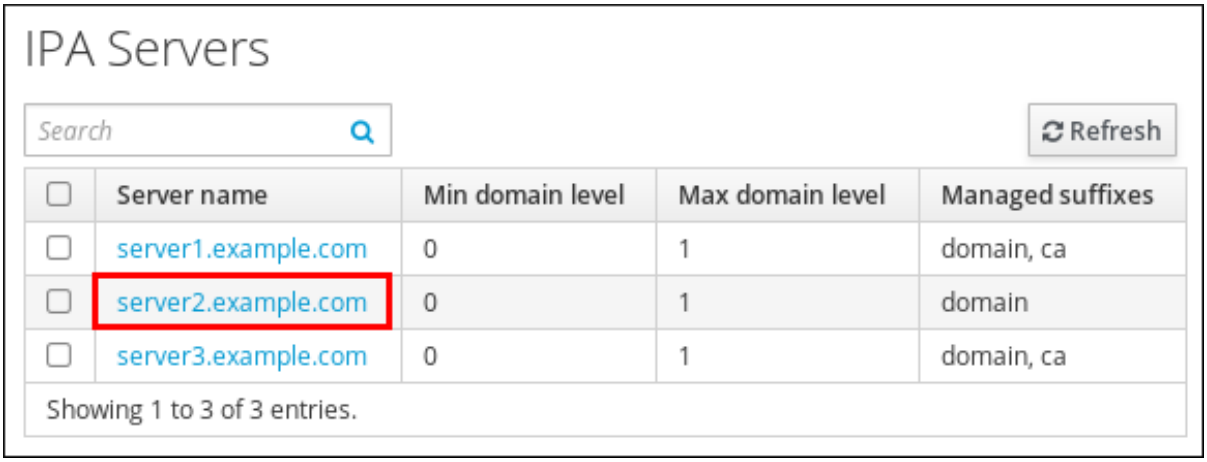
```
Removal of 'server1.example.com' leads to disconnected topology in suffix 'domain':
Topology does not allow server server2.example.com to replicate with servers:
  server3.example.com
  server4.example.com
...
```

6.4.1. Web UI：从拓扑中删除服务器

在不从机器卸载服务器组件的情况下从拓扑中删除服务器组件：

1. 选择 **IPA Server → Topology → IPA Servers**。
2. 单击要删除的服务器的名称。

图 6.13. 选择服务器



3. 单击 **Delete Server**。

6.4.2. 命令行：从拓扑中删除服务器

重要

删除服务器是一个不可逆的操作。如果您删除了服务器，将其重新引入拓扑的唯一方法是在机器上安装新副本。

删除 **server1.example.com** ：

1. 在另一台服务器上，运行 **ipa server-del** 命令来移除 **server1.example.com**。该命令删除指向服务器的所有拓扑片段：

```
[user@server2 ~]$ ipa server-del
Server name: server1.example.com
Removing server1.example.com from replication topology, please wait...
-----------------------------------------------------------
Deleted IPA server "server1.example.com"
-----------------------------------------------------------
```

2. 在 **server1.example.com** 上，运行 **ipa server-install --uninstall** 命令来从机器中卸载服务器组件。

```
[root@server1 ~]# ipa server-install --uninstall
```

6.5. 管理服务器角色

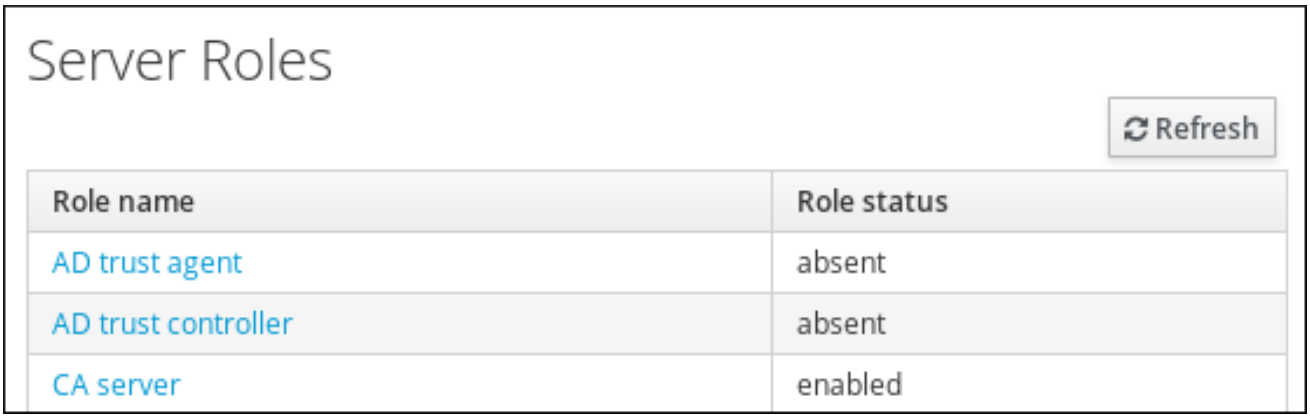
根据在 IdM 服务器中安装的服务，它可以执行各种 *服务器角色*。例如：CA 服务器、DNS 服务器或密钥恢复机构(KRA)服务器.

6.5.1. 查看服务器角色

Web UI：查看服务器角色
有关支持的服务器角色的完整列表，请参阅 **IPA 服务器 → 拓扑 → 服务器角色**。

- **缺少** 角色状态意味着拓扑中没有服务器执行该角色。
- **启用** 角色状态意味着拓扑中的一个或多个服务器正在执行该角色。

图 6.14. Web UI 中的服务器角色



命令行：查看服务器角色
ipa config-show 命令显示所有 CA 服务器、NTP 服务器和当前的 CA 续订 master：

```
$ ipa config-show
  ...
  IPA masters: server1.example.com, server2.example.com, server3.example.com
  IPA CA servers: server1.example.com, server2.example.com
  IPA NTP servers: server1.example.com, server2.example.com, server3.example.com
  IPA CA renewal master: server1.example.com
```

ipa server-show 命令显示在特定服务器上启用的角色列表。例如，对于 *server.example.com* 上启用的角色列表：

```
$ ipa server-show
Server name: server.example.com
  ...
  Enabled server roles: CA server, DNS server, NTP server, KRA server
```

ipa server-find --servrole 搜索启用了特定服务器角色的所有服务器。例如，要搜索所有 CA 服务器：

```
$ ipa server-find --servrole "CA server"
---------------------
2 IPA servers matched
```

```
----------------------
  Server name: server1.example.com
  ...

  Server name: server2.example.com
  ...
----------------------------
Number of entries returned 2
----------------------------
```

6.5.2. 将副本提升到主 CA 服务器

注意

本节论述了在域级别 1 中更改 CA 续订 master（请参阅 第 7 章 *显示和提升域级别*）。有关在域级别 0 更改 CA 续订 master 的文档，请参考 第 D.4 节 “将副本提升到主 CA 服务器”。

如果您的 IdM 部署使用嵌入的证书颁发机构(CA)，其中一个 IdM CA 服务器充当 master CA：它管理 CA 子系统证书的续订并生成证书撤销列表(CRL)。默认情况下，master CA 是系统管理员使用 **ipa-server-install** 或 **ipa-ca-install** 命令在其上安装 CA 角色的第一个服务器。

如果您计划使 master CA 服务器离线或停用它，请提升另一个 CA 服务器作为新的 CA 续订 master：

1. 配置副本以处理 CA 子系统证书续订。
   - 有关域级别 1 的信息，请参阅 第 6.5.2.1 节 “更改当前 CA 续订 master”。
   - 有关域级别 0，请参阅 第 D.4.1 节 “更改 Which 服务器处理证书续订”。
2. 配置副本以生成 CRL。请参阅 第 6.5.2.2 节 “更改 Which Server Generates CRL”。
3. 在停用之前的 master CA 服务器前，请确保新的 master 正常工作。请参阅 第 6.5.2.3 节 “验证新 master CA 服务器是否配置正确”。

6.5.2.1. 更改当前 CA 续订 master

Web UI：更改当前 CA 续订 master

1. 选择 **IPA Server → Configuration**。
2. 在 **IPA CA renewal master** 字段中，选择新的 CA renewal master。

命令行：更改当前 CA 续订 master
使用 **ipa config-mod --ca-renewal-master-server** 命令：

```
$ ipa config-mod --ca-renewal-master-server new_ca_renewal_master.example.com
  ...
  IPA masters: old_ca_renewal_master.example.com, new_ca_renewal_master.example.com
  IPA CA servers: old_ca_renewal_master.example.com, new_ca_renewal_master.example.com
  IPA NTP servers: old_ca_renewal_master.example.com, new_ca_renewal_master.example.com
  IPA CA renewal master: new_ca_renewal_master.example.com
```

输出确认更新成功。

6.5.2.2. 更改 Which Server Generates CRL

要更改哪个服务器生成证书撤销列表(CRL)：

1. 如果您不知道当前的 CRL 生成 master，在每个 IdM 证书颁发机构(CA)上使用 **ipa-crlgen-manage status** 命令来确定是否启用了 CRL 生成：

```
# ipa-crlgen-manage status
CRL generation: enabled
```

2. 在当前的 CRL 生成 master 上，禁用此功能：

```
# ipa-crlgen-manage disable
```

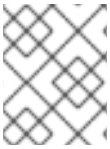
3. 在您要配置为新 CRL 生成 master 的其他 CA 主机上，启用此功能：

```
# ipa-crlgen-manage enable
```

6.5.2.3. 验证新 master CA 服务器是否配置正确

确保新的 master CA 服务器上存在 **/var/lib/ipa/pki-ca/publish/MasterCRL.bin** 文件。

该文件会根据使用 ***ca.crl.MasterCRL.autoUpdateInterval*** 参数在 **/etc/pki/pki-tomcat/ca/CS.cfg** 文件中定义的时间间隔生成。默认值为 240 分钟（4 小时）。



注意

如果您更新 ***ca.crl.MasterCRL.autoUpdateInterval*** 参数，则更改将在下一次调度的 CRL 更新后生效。

如果文件存在，则新的 master CA 服务器会被正确配置，您可以安全地忽略之前的 CA master 系统。

6.5.3. Hidden Replicas 的降级和升级

安装副本后，您可以更改副本是隐藏还是可见：

- 要将可见的副本降级到隐藏的副本：

  1. 如果副本是 CA 续订 master，请将该服务移到另一个副本。详情请查看 第 6.5.2.1 节 “更改当前 CA 续订 master”。

  2. 将副本的状态更改为 **隐藏** ：

```
# ipa server-state replica.idm.example.com --state=hidden
```

- 要将隐藏的副本提升到可见的副本，请输入：

```
# ipa server-state replica.idm.example.com --state=enabled
```

注意

隐藏的副本功能在 Red Hat Enterprise Linux 7.7 及更新的版本中作为技术预览提供，因此不受支持。

第 7 章 显示和提升域级别

域级别表示 IdM 拓扑中可用的操作和功能。

域级别 1

可用功能示例：

- 简化的 **ipa-replica-install** （请参阅 第 4.5 节 “创建副本：简介”）
- 增强的拓扑管理（请参阅 第 6 章 *管理复制拓扑*）

重要

在 Red Hat Enterprise Linux 7.3 中使用 IdM 版本 4.4 引入了域级别 1。要使用域级别 1 功能，您的所有副本都必须运行红帽企业 Linux 7.3 或更高版本。

如果您的第一个服务器安装了 Red Hat Enterprise Linux 7.3，则您的域的域级别将自动设置为 1。

如果您将所有服务器升级到 IdM 版本 4.4，则不会自动提高域级别。如果要使用域级别 1 功能，请手动提高域级别，如 第 7.2 节 “提高域级别” 所述。

域级别 0

可用功能示例：

- **ipa-replica-install** 需要在初始服务器上创建副本信息文件并将其复制到副本（请参阅 第 D.2 节 “创建副本”）
- 使用 **ipa-replica-manage** 和 **ipa-csreplica-manage** （请参阅 第 D.3 节 “管理副本和复制协议”）

7.1. 显示当前域级别

命令行：显示当前域级别

1. 以管理员身份登录：

```
$ kinit admin
```

2. 运行 **ipa domainlevel-get** 命令：

```
$ ipa domainlevel-get
-----------------------
Current domain level: 0
-----------------------
```

Web UI：显示当前域级别
选择 **IPA服务器 → 拓扑域级别**。

7.2. 提高域级别

重要

这是不可逆的操作。如果您将域级别从 **0** 增加到 **1**，则无法再次从 **1** 降级到 0。

命令行：提高域级别

1. **以管理员身份登录：**

```
$ kinit admin
```

2. **运行 ipa domainlevel-set 命令并提供所需的级别：**

```
$ ipa domainlevel-set 1
-----------------------
Current domain level: 1
-----------------------
```

Web UI：提高域级别

1. 选择 **IPA服务器 → 拓扑域级别。**

2. 单击 **Set Domain Level。**

第 8 章 更新和迁移身份管理

8.1. 更新身份管理

您可以使用 yum 工具更新系统上的身份管理软件包。

另外，如果有新的次版本 Red Hat Enterprise Linux，如 7.3，yum 会将身份管理服务器或客户端升级到这个版本。

注意

本节不论述将身份管理从 Red Hat Enterprise Linux 6 迁移到 Red Hat Enterprise Linux 7。如果要迁移，请参阅 第 8.2 节 “将身份管理从红帽企业 Linux 6 迁移到版本 7”。

8.1.1. 更新身份管理的注意事项

- 在至少一台服务器上更新 Identity Management 软件包后，拓扑中的所有其他服务器都会接收更新的 schema，即使您没有更新其软件包。这将确保任何使用新模式的新条目都可以在其他服务器之间复制。

- 不支持降级身份管理软件包。

  重要

  不要在任何 ipa ö 软件包上运行 yum downgrade 命令。

- 红帽建议只升级到下一个版本。例如，如果您想要升级到 Red Hat Enterprise Linux 7.4 的 Identity Management，我们建议从 Red Hat Enterprise Linux 7.3 的 Identity Management 升级。从较早版本升级可能会导致问题。

8.1.2. 使用 yum 更新身份管理软件包

更新服务器或客户端中的所有身份管理软件包：

```
# yum update ipa-*
```

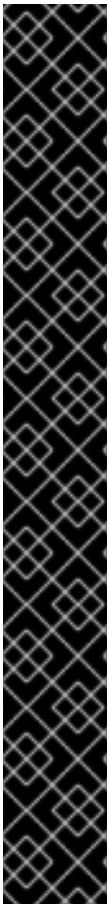
警告

在升级多个身份管理服务器时，请在每次升级之间至少等待 10 分钟。

当两个或更多个服务器同时升级，或在不同升级之间只能简短的间隔，则可能没有足够的时间来在整个拓扑间复制升级后的数据变化，从而会导致复制事件冲突。

相关信息

- 有关使用 yum 工具的详情，请参考 *系统管理员指南中的* Yum。



重要

由于 CVE-2014-3566，在 mod_nss 模块中需要禁用安全套接字层版本 3 (SSLv3)协议。您可以通过以下步骤确保：

1. 编辑 /etc/httpd/conf.d/nss.conf 文件，并将 *NSSProtocol* 参数设置为 TLSv1.0 （用于向后兼容）、TLSv1.1 和 TLSv1.2。

```
NSSProtocol TLSv1.0,TLSv1.1,TLSv1.2
```

2. 重新启动 httpd 服务。

```
# systemctl restart httpd.service
```

请注意，当启动 yum update ipa 114 命令时，Red Hat Enterprise Linux 7 中的身份管理会自动执行上述步骤来升级主软件包。

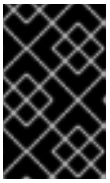
8.2. 将身份管理从红帽企业 LINUX 6 迁移到版本 7

这个步骤描述了如何将所有数据和配置从 Red Hat Enterprise Linux 6 Identity Management 迁移到 Red Hat Enterprise Linux 7 服务器。迁移步骤包括：

- 将基于红帽企业 Linux 6 的证书颁发机构(CA)主服务器迁移到红帽企业 Linux 7。
- 将所有服务转移到新的红帽企业 Linux 7 服务器.这些服务包括 CRL 和证书创建、DNS 管理或 Kerberos KDC 管理。
- 停用原始 Red Hat Enterprise Linux 6 CA master。

在以下步骤中：

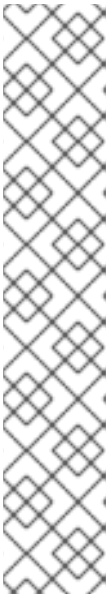
- rhel7.example.com 是 Red Hat Enterprise Linux 7 系统，它将成为新的 CA master。



重要

RHEL 7.9 唯一支持的次版本。确定在您的系统上安装了 RHEL 7.9。

- rhel6.example.com 是原始 Red Hat Enterprise Linux 6 CA master。



注意

要识别哪个 Red Hat Enterprise Linux 6 服务器是主 CA 服务器，请确定 certmonger 服务跟踪 renewal _ca_cert 命令。在 Red Hat Enterprise Linux 6 服务器上运行这个命令：

```
[root@rhel6 ~]# getcert list -d /var/lib/pki-ca/alias -n "subsystemCert cert-pki-ca" | grep post-save
post-save command: /usr/lib64/ipa/certmonger/renew_ca_cert "subsystemCert cert-pki-ca"
```

执行 renewal _ca_cert 的 post-save 操作只针对 CA master 定义。

8.2.1. 将身份管理从 Red Hat Enterprise Linux 6 迁移到 7 的先决条件

- 将 rhel6.example.com 系统更新至最新的 Red Hat Enterprise Linux 6 版本。

- 在 rhel6.example.com 系统中，升级 ipa the 软件包：

```
[root@rhel6 ~]# yum update ipa-*
```

此步骤还确保您应用了 RHBA-2015:0231-2 公告，该公告提供 bind-dyndb-ldap 软件包的 2.3-6.el6_6 版本，并附带 Red Hat Enterprise Linux 6.6 Extended Update Support (EUS)。

警告

使用早期版本的 bind-dyndb-ldap 会导致 Red Hat Enterprise Linux 6.6 DNS 服务器和 Red Hat Enterprise Linux 7 DNS 服务器间的 DNS 转发区域服务的行为不一致。

- 确保 rhel7.example.com 系统满足 第 2.1 节 “安装服务器的先决条件” 和 第 4.3 节 “安装副本的先决条件” 的要求。

- 在 rhel7.example.com 系统上，安装所需的软件包。请参阅 第 2.2 节 “安装 IdM 服务器所需的软件包”。

8.2.2. 更新 Red Hat Enterprise Linux 6 上的身份管理架构

copy-schema-to-ca.py 模式更新脚本为 rhel7.example.com 副本安装准备 rhel6.example.com。由于身份管理版本 3.1 及更高版本之间的 schema 更改，因此需要更新该架构。

1. 将 copy-schema-to-ca.py 模式更新脚本从 rhel7.example.com 系统复制到 rhel6.example.com 系统。例如：

```
[root@rhel7 ~]# scp /usr/share/ipa/copy-schema-to-ca.py root@rhel6:/root/
```

2. **在 rhel6.example.com 上运行更新的 copy-schema-to-ca.py 脚本。**

```
[root@rhel6 ~]# python copy-schema-to-ca.py
ipa        : INFO     Installed /etc/dirsrv/slapd-PKI-IPA//schema/60kerberos.ldif
[... output truncated ...]
ipa        : INFO     Schema updated successfully
```

3. **在连接到 Red Hat Enterprise Linux 7 副本前，在每个运行证书颁发机构的 Red Hat Enterprise Linux 6 IdM 副本上重复这些步骤。**

8.2.3. 安装 Red Hat Enterprise Linux 7 Replica

1. **在 rhel6.example.com 系统上，创建用于安装 rhel7.example.com 副本的副本文件。例如，要为 rhel7.example.com 创建一个副本文件，其 IP 地址为 192.0.2.1 ：**

```
[root@rhel6 ~]# ipa-replica-prepare rhel7.example.com --ip-address 192.0.2.1

Directory Manager (existing master) password:
Preparing replica for rhel7.example.com from rhel6.example.com
[... output truncated ...]
The ipa-replica-prepare command was successful
```

另请参阅 **第 D.1 节 “副本信息文件”** 和 **第 D.2 节 “创建副本”**。

2. **将副本信息文件从 rhel6.example.com 复制到 rhel7.example.com。**

```
[root@rhel6 ~]# scp /var/lib/ipa/replica-info-replica.example.com.gpg root@rhel7:/var/lib/ipa/
```

3. **如果您在 Red Hat Enterprise Linux 7.6 或更高版本中安装带有集成 CA 的新副本，请将以下条目附加到 /etc/httpd/conf.d/nss.conf 文件中的 *NSSCipherSuite* 参数的末尾：**

```
+ecdhe_rsa_aes_128_sha,+ecdhe_rsa_aes_256_sha
```

在 Red Hat Enterprise Linux 7.6 或更高版本中，IdM CA 中不再启用某些密码。如果没有将此条目添加到配置中，在 Red Hat Enterprise Linux 7.6 上设置带有集成 CA 的 IdM 服务器作为在 Red Hat Enterprise Linux 6 上运行的 master 的副本会失败，并显示 CRITICAL Failed 来配置 CA 实例 错误。

4. 使用副本文件安装 rhel7.example.com 副本。例如，以下命令使用以下选项：

- --setup-ca 用来设置证书系统组件
- --setup-dns 和 --forwarder 来配置集成的 DNS 服务器并设置转发器
- --ip-address 指定 rhel7.example.com 系统的 IP 地址

```
[root@rhel7 ~]# ipa-replica-install /var/lib/ipa/replica-info-rhel7.example.com.gpg --setup-ca --ip-address 192.0.2.1 --setup-dns --forwarder 192.0.2.20
Directory Manager (existing master) password:

Checking DNS forwarders, please wait ...
Run connection check to master
[... output truncated ...]
Client configuration complete.
```

另请参阅：

- 第 D.2 节 “创建副本”描述了使用副本信息文件创建副本
- 第 2.3.1 节 “确定使用集成 DNS” 和 第 2.3.2 节 “确定要使用的 CA 配置”

5. 验证身份管理服务是否在 rhel7.example.com 上运行。

```
[root@rhel7 ~]# ipactl status
Directory Service: RUNNING
[... output truncated ...]
ipa: INFO: The ipactl command was successful
```

8.2.4. 将 CA 服务转换到 Red Hat Enterprise Linux 7 服务器

开始前：

- 验证 rhel6.example.com 和 rhel7.example.com CA 是否都配置为主服务器。

```
[root@rhel7 ~]$ kinit admin
[root@rhel7 ~]$ ipa-csreplica-manage list
rhel6.example.com: master
rhel7.example.com: master
```

显示复制协议的详情：

```
[root@rhel7 ~]# ipa-csreplica-manage list --verbose rhel7.example.com
rhel7.example.com
last init status: None
last init ended: 1970-01-01 00:00:00+00:00
last update status: Error (0) Replica acquired successfully: Incremental update succeeded
last update ended: 2017-02-13 13:55:13+00:00
```

在 rhel6.example.com 原始 master CA 上，停止 CA 子系统证书续订：

1. **禁用跟踪原始 CA 证书。**

```
[root@rhel6 ~]# getcert stop-tracking -d /var/lib/pki-ca/alias -n "auditSigningCert cert-pki-ca"
Request "20201127184547" removed.
[root@rhel6 ~]# getcert stop-tracking -d /var/lib/pki-ca/alias -n "ocspSigningCert cert-pki-ca"
Request "20201127184548" removed.
[root@rhel6 ~]# getcert stop-tracking -d /var/lib/pki-ca/alias -n "subsystemCert cert-pki-ca"
Request "20201127184549" removed.
[root@rhel6 ~]# getcert stop-tracking -d /etc/httpd/alias -n ipaCert
Request "20201127184550" removed.
```

2. **重新配置 rhel6.example.com，以从新的 master CA 检索更新的证书。**

   a. **将续订帮助程序脚本复制到 certmonger 服务目录中，并设置适当的权限。**

```
[root@rhel6 ~]# cp /usr/share/ipa/ca_renewal /var/lib/certmonger/cas/
[root@rhel6 ~]# chmod 0600 /var/lib/certmonger/cas/ca_renewal
```

   b. **更新 SELinux 配置。**

```
[root@rhel6 ~]# restorecon /var/lib/certmonger/cas/ca_renewal
```

   c. **重新启动 certmonger。**

-

```
[root@rhel6 ~]# service certmonger restart
```

d. **检查 CA 是否已列出以检索证书。**

```
[root@rhel6 ~]# getcert list-cas
...
CA 'dogtag-ipa-retrieve-agent-submit':
        is-default: no
        ca-type: EXTERNAL
 helper-location: /usr/libexec/certmonger/dogtag-ipa-retrieve-agent-submit
```

e. **获取 CA 证书数据库 PIN。**

```
[root@rhel6 ~]# grep internal= /var/lib/pki-ca/conf/password.conf
```

f. **配置 certmonger 以跟踪外部续订的证书。这需要数据库 PIN。**

```
[root@rhel6 ~]# getcert start-tracking \
    -c dogtag-ipa-retrieve-agent-submit \
    -d /var/lib/pki-ca/alias \
    -n "auditSigningCert cert-pki-ca" \
    -B /usr/lib64/ipa/certmonger/stop_pkicad \
    -C '/usr/lib64/ipa/certmonger/restart_pkicad \
    "auditSigningCert cert-pki-ca"' \
    -T "auditSigningCert cert-pki-ca" \
    -P database_pin
New tracking request "20201127184743" added.
[root@rhel6 ~]# getcert start-tracking \
    -c dogtag-ipa-retrieve-agent-submit \
    -d /var/lib/pki-ca/alias \
    -n "ocspSigningCert cert-pki-ca" \
    -B /usr/lib64/ipa/certmonger/stop_pkicad \
    -C '/usr/lib64/ipa/certmonger/restart_pkicad \
    "ocspSigningCert cert-pki-ca"' \
    -T "ocspSigningCert cert-pki-ca" \
    -P database_pin
New tracking request "20201127184744" added.
[root@rhel6 ~]# getcert start-tracking \
    -c dogtag-ipa-retrieve-agent-submit \
    -d /var/lib/pki-ca/alias \
    -n "subsystemCert cert-pki-ca" \
    -B /usr/lib64/ipa/certmonger/stop_pkicad \
    -C '/usr/lib64/ipa/certmonger/restart_pkicad \
    "subsystemCert cert-pki-ca"' \
    -T "subsystemCert cert-pki-ca" \
    -P database_pin
New tracking request "20201127184745" added.
[root@rhel6 ~]# getcert start-tracking \
```

```
    -c dogtag-ipa-retrieve-agent-submit \
    -d /etc/httpd/alias \
    -n ipaCert \
    -C /usr/lib64/ipa/certmonger/restart_httpd \
    -T ipaCert \
    -p /etc/httpd/alias/pwdfile.txt
New tracking request "20201127184746" added.
```

将 CRL 生成从原始 rhel6.example.com CA master 移到 rhel7.example.com。

1. **在 rhel6.example.com 上，停止 CRL 生成：**

   a. **停止 CA 服务。**

   ```
   [root@rhel6 ~]# service pki-cad stop
   ```

   b. **在 rhel6.example.com 中禁用 CRL 生成。打开 /var/lib/pki-ca/conf/CS.cfg 文件，并将 *ca.crl.MasterCRL.enableCRLCache* 和 ca.crl.MasterCRLUpdates 参数的值设置为 false。**

   ```
   ca.crl.MasterCRL.enableCRLCache=false
   ca.crl.MasterCRL.enableCRLUpdates=false
   ```

   c. **启动 CA 服务。**

   ```
   [root@rhel6 ~]# service pki-cad start
   ```

2. **在 rhel6.example.com 上，将 Apache 配置为重定向 CRL 请求：**

   a. **打开 /etc/httpd/conf.d/ipa-pki-proxy.conf 文件，取消注释 RewriteRule 条目：**

   ```
   RewriteRule ^/ipa/crl/MasterCRL.bin https://rhel6.example.com/ca/ee/ca/getCRL?
   op=getCRL&crlIssuingPoint=MasterCRL [L,R=301,NC]
   ```

   注意

   不要替换 URL 中的服务器主机名。URL 必须引用本地主机名。

b. 重新启动 Apache.

```
[root@rhel6 ~]# service httpd restart
```

IdM 现在从本地 CA 获取证书撤销列表(CRL)，而不是从本地文件获取。

3. 在 rhel7.example.com 上，将 rhel7.example.com 配置为新的 CA master：

a. 配置 rhel7.example.com 以处理 CA 子系统证书续订，如 第 D.4.1 节 “更改 Which 服务器处理证书续订” 所述。

b. 将 rhel7.example.com 配置为常规证书撤销列表(CRL)，如 第 6.5.2.2 节 “更改 Which Server Generates CRL” 所述。

相关信息

- 有关 CA 子系统证书续订和 CRL 的详细信息，请参阅 第 6.5.2 节 “将副本提升到主 CA 服务器”。

8.2.5. 停止 Red Hat Enterprise Linux 6 服务器

停止 rhel6.example.com 上的所有服务，将域发现强制到新的 rhel7.example.com 服务器。

```
[root@rhel6 ~]# ipactl stop
Stopping CA Service
Stopping pki-ca:                                           [  OK  ]
Stopping HTTP Service
Stopping httpd:                                            [  OK  ]
Stopping MEMCACHE Service
Stopping ipa_memcached:                                    [  OK  ]
Stopping DNS Service
Stopping named: .                                          [  OK  ]
Stopping KPASSWD Service
Stopping Kerberos 5 Admin Server:                          [  OK  ]
Stopping KDC Service
Stopping Kerberos 5 KDC:                                   [  OK  ]
Stopping Directory Service
Shutting down dirsrv:
    EXAMPLE-COM...                                         [  OK  ]
    PKI-IPA...                                             [  OK  ]
```

之后，使用 ipa 实用程序将通过远程过程调用(RPC)联系新的服务器。

8.2.6. 在迁移主 CA 服务器后，下一步

对于拓扑中的每个 Red Hat Enterprise Linux 6 服务器：

1. 从 rhel7.example.com 创建副本文件。

注意

从 Red Hat Enterprise Linux 6 服务器安装 Red Hat Enterprise Linux 7 副本后，身份管理域的域级别会自动设置为 0。

Red Hat Enterprise Linux 7.3 引进了一种更简单的方式来安装和管理副本。要使用这些功能，您的拓扑必须在域级别 1 上。请参阅 第 7 章 *显示和提升域级别*。

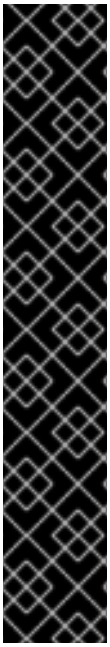
2. 使用副本文件在另一个 Red Hat Enterprise Linux 7 系统上安装新副本。

请参阅 第 4 章 *安装和卸载身份管理副本*。

要取消使用 Red Hat Enterprise Linux 6 服务器：

- 通过在 Red Hat Enterprise Linux 7 服务器中执行移除命令，从拓扑中删除服务器。

请参阅 第 2.4 节 “卸载 IdM 服务器”。



重要

客户端配置不会自动更新。如果您取消授权 IDM 服务器并使用不同的名称配置了新服务器，您应该检查整个客户端配置。特别是，您必须手动更新以下文件：

- **/etc/openldap/ldap.conf**
- **/etc/ipa/default.conf**
- **/etc/sssd/sssd.conf**

第 9 章 备份和恢复身份管理

Red Hat Enterprise Linux Identity Management 提供了手动备份和恢复 IdM 系统的解决方案，例如当服务器停止正确执行或数据丢失时。在备份过程中，系统会创建一个目录，其中包含您的 IdM 设置信息并存储它。在恢复过程中，您可以使用这个备份目录使原始 IdM 设置返回。

重要

只有在您无法从部署的其余部分中重建丢失的 IdM 服务器组部分时，才使用本章中描述的备份和恢复步骤，方法是重新安装丢失的副本作为剩余服务器的副本。

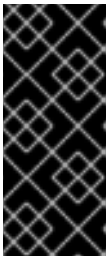
IdM/IPA 中的 "Backup 和 Restore in IdM/IPA"知识库解决方案 描述了如何通过维护多个服务器副本来避免损失。最好从具有相同数据的现有副本中重建，因为备份版本通常包含较旧的信息，因此可能会过期的信息。

备份和恢复可防止的潜在威胁情况包括：

- 机器上的灾难性硬件故障发生，机器变得无法进一步运行。在这种情况下：

  1. 从头开始重新安装操作系统。
  2. 配置相同主机名、完全限定域名(FQDN)和 IP 地址的计算机。
  3. 安装 IdM 软件包，以及与原始系统中存在的 IdM 相关的所有其他可选软件包。
  4. 恢复 IdM 服务器的完整备份。

- 在隔离的机器上进行升级会失败。操作系统仍然可以正常工作，但 IdM 数据已损坏，因此您要将 IdM 系统恢复到已知良好的状态。



重要

如果出现硬件或升级失败（如上述两项），只有在所有副本或具有特殊角色（如唯一证书颁发机构(CA)）的副本都已丢失时才从备份中进行恢复。如果仍存在具有相同数据的副本，建议删除丢失的副本，然后将其从剩余副本中重建。

- 对 LDAP 内容进行了不必要的更改，例如删除了条目，您想要恢复它们。恢复备份的 LDAP 数据会将 LDAP 条目返回到之前的状态，而不影响 IdM 系统本身。

恢复的服务器成为 IdM 的唯一信息来源；其他 master 服务器是从恢复的服务器重新初始化的。最后一次备份后创建的所有数据都将丢失。因此，您不应该使用备份和恢复解决方案进行正常的系统维护。如果可能，请始终通过将丢失的服务器重新安装为副本来重建丢失的服务器。

备份和恢复功能只能从命令行管理，在 IdM Web UI 中不可用。

9.1. 仅备份全服务器备份和恢复

IdM 提供两个备份选项：

全 IdM 服务器备份

全服务器备份会创建所有 IdM 服务器文件的备份副本以及 LDAP 数据，这使其成为独立备份。IdM 会影响数百个文件；备份过程复制的文件是整个目录和特定文件（如配置文件或日志文件）的组合，并与 IdM 依赖的各种服务直接相关。由于全服务器备份是原始文件备份，因此它会脱机执行。执行 full-server 备份的脚本停止所有 IdM 服务，以确保备份过程的安全。

有关完整服务器备份副本的文件和目录的完整列表，请查看 第 9.1.3 节 “备份期间目录和文件绑定列表”。

只数据备份

仅数据备份仅创建 LDAP 数据的备份副本以及 changelog。进程备份 IPA-REALM 实例，也可以备份多个后端或只有一个后端；后端包括 IPA 后端和 CA Dogtag 后端。这种类型的备份还会备份以 LDIF（LDAP 数据交换格式）存储的 LDAP 内容记录。仅数据备份可以在线和脱机执行。

默认情况下，IdM 将创建的备份存储在 /var/lib/ipa/backup/ 目录中。包含备份的子目录的命名约定有：

- ipa-full-YEAR-MM-DD-HH-MM-SS （全服务器备份）中的 ipa-full-YEAR-MM-DD-HH-MM-SS
- ipa-data-YEAR-MM-DD-HH-MM-SS （仅数据备份）中的 ipa-data-YEAR-MM-DD-HH-MM-SS

9.1.1. 创建备份

使用 ipa-backup 工具创建全服务器和仅数据备份，必须始终以 root 用户身份运行。

要创建全服务器备份，请运行 ipa-backup。

重要

执行全服务器备份会停止所有 IdM 服务，因为进程必须离线运行。IdM 服务将在备份完成后再次启动。

要创建仅数据备份，请运行 ipa-backup --data 命令。

您可以在 ipa-backup 中添加几个附加选项：

- --online 执行在线备份；这个选项仅适用于仅数据备份
- --logs 在备份中包含 IdM 服务日志文件

有关使用 ipa-backup 的详情，请参考 ipa-backup(1) man page。

9.1.1.1. 在备份过程中，在卷不足时工作空间不足

本节论述了如何解决 IdM 备份过程中涉及的目录存储在空闲空间不足的卷中的问题。

卷中含有 /var/lib/ipa/backup/ 的空间不足

如果 /var/lib/ipa/backup/ 目录存储在空闲空间不足的卷中，则无法创建备份。要解决这个问题，请使用以下临时解决方案之一：

- 在不同的卷上创建一个目录，并将其链接到 /var/lib/ipa/backup/。例如，如果 /home 存储在有足够可用空间的不同卷中：

  1. 创建目录，如 /home/idm/backup/ ：

     ```
     # mkdir -p /home/idm/backup/
     ```

  2. 将以下权限设置为该目录：

     ```
     # chown root:root /home/idm/backup/
     # chmod 700 /home/idm/backup/
     ```

  3. 如果 /var/lib/ipa/backup/ 包含现有的备份，请将其移到新目录中：

     ```
     # mv /var/lib/ipa/backup/* /home/idm/backup/
     ```

  4. 删除 /var/lib/ipa/backup/ 目录：

     ```
     # rm -rf /var/lib/ipa/backup/
     ```

  5. 创建 /var/lib/ipa/backup/ 链接到 /home/idm/backup/ 目录：

     ```
     # ln -s /home/idm/backup/ /var/lib/ipa/backup/
     ```

- 将存储在不同卷上的目录挂载到 /var/lib/ipa/backup/。例如，如果 /home 存储在具有足够可用空间的不同卷中，请创建 /home/idm/backup/，并将其挂载到 /var/lib/ipa/backup/ ：

  1.

1. 创建 /home/idm/backup/ 目录：

```
# mkdir -p /home/idm/backup/
```

2. 将以下权限设置为该目录：

```
# chown root:root /home/idm/backup/
# chmod 700 /home/idm/backup/
```

3. 如果 /var/lib/ipa/backup/ 包含现有的备份，请将其移到新目录中：

```
# mv /var/lib/ipa/backup/* /home/idm/backup/
```

4. 将 /home/idm/backup/ 挂载到 /var/lib/ipa/backup/ ：

```
# mount -o bind /home/idm/backup/ /var/lib/ipa/backup/
```

5. 要在系统引导时自动挂载 /home/idm/backup/ 到 /var/lib/ipa/backup/，请在 /etc/fstab 文件中附加以下内容：

```
/home/idm/backup/     /var/lib/ipa/backup/     none     bind     0 0
```

卷中含有 /tmp 的空间不足

如果因为 /tmp 目录中空间不足造成备份失败，请使用 TMPDIR 环境变量更改在备份期间创建的暂存文件的位置：

```
# TMPDIR=/path/to/backup ipa-backup
```

详情请查看 ipa-backup 命令无法完成 知识库解决方案。

9.1.2. 加密备份

您可以使用 GNU Privacy Guard(GPG)加密 IdM 备份。

创建 GPG 密钥：

1. **创建包含密钥详情的 keygen 文件，例如运行 cat >keygen <<EOF 并在命令行中提供所需的加密详情：**

```
[root@server ~]# cat >keygen <<EOF
> %echo Generating a standard key
> Key-Type: RSA
> Key-Length:2048
> Name-Real: IPA Backup
> Name-Comment: IPA Backup
> Name-Email: root@example.com
> Expire-Date: 0
> %pubring /root/backup.pub
> %secring /root/backup.sec
> %commit
> %echo done
> EOF
[root@server ~]#
```

2. **生成名为 backup 的新密钥对，并将 keygen 的内容提供给命令。以下示例生成了一个名为 /root/ backup.sec 和 / root /backup.pub 的密钥对 ：**

```
[root@server ~]# gpg --batch --gen-key keygen
[root@server ~]# gpg --no-default-keyring --secret-keyring /root/backup.sec \
      --keyring /root/backup.pub --list-secret-keys
```

要创建 GPG 加密备份，请通过提供以下选项将生成的 备份密钥传递给 ipa- backup ：

- **--GPG，它指示 ipa-backup 执行加密的备份**
- **--GPG-keyring=GPG_KEYRING，它提供了 GPG 密钥环的完整路径，而无需文件扩展名。**

例如：

```
[root@server ~]# ipa-backup --gpg --gpg-keyring=/root/backup
```

注意

如果您的系统使用 gpg2 工具生成 GPG 密钥，您可能会遇到问题，因为 gpg2 需要外部程序才能正常工作。要在这种情况下从控制台生成密钥，请在生成密钥前将 pinentry-program /usr/bin/pinentry-curses 行添加到 .gnupg/gpg-agent.conf 文件中。

9.1.3. 备份期间目录和文件绑定列表

目录：

```
/usr/share/ipa/html
/root/.pki
/etc/pki-ca
/etc/pki/pki-tomcat
/etc/sysconfig/pki
/etc/httpd/alias
/var/lib/pki
/var/lib/pki-ca
/var/lib/ipa/sysrestore
/var/lib/ipa-client/sysrestore
/var/lib/ipa/dnssec
/var/lib/sss/pubconf/krb5.include.d/
/var/lib/authconfig/last
/var/lib/certmonger
/var/lib/ipa
/var/run/dirsrv
/var/lock/dirsrv
```

文件：

```
/etc/named.conf
/etc/named.keytab
/etc/resolv.conf
/etc/sysconfig/pki-ca
/etc/sysconfig/pki-tomcat
/etc/sysconfig/dirsrv
/etc/sysconfig/ntpd
/etc/sysconfig/krb5kdc
/etc/sysconfig/pki/ca/pki-ca
/etc/sysconfig/ipa-dnskeysyncd
/etc/sysconfig/ipa-ods-exporter
/etc/sysconfig/named
/etc/sysconfig/ods
/etc/sysconfig/authconfig
/etc/ipa/nssdb/pwdfile.txt
/etc/pki/ca-trust/source/ipa.p11-kit
/etc/pki/ca-trust/source/anchors/ipa-ca.crt
/etc/nsswitch.conf
/etc/krb5.keytab
```

```
/etc/sssd/sssd.conf
/etc/openldap/ldap.conf
/etc/security/limits.conf
/etc/httpd/conf/password.conf
/etc/httpd/conf/ipa.keytab
/etc/httpd/conf.d/ipa-pki-proxy.conf
/etc/httpd/conf.d/ipa-rewrite.conf
/etc/httpd/conf.d/nss.conf
/etc/httpd/conf.d/ipa.conf
/etc/ssh/sshd_config
/etc/ssh/ssh_config
/etc/krb5.conf
/etc/ipa/ca.crt
/etc/ipa/default.conf
/etc/dirsrv/ds.keytab
/etc/ntp.conf
/etc/samba/smb.conf
/etc/samba/samba.keytab
/root/ca-agent.p12
/root/cacert.p12
/var/kerberos/krb5kdc/kdc.conf
/etc/systemd/system/multi-user.target.wants/ipa.service
/etc/systemd/system/multi-user.target.wants/sssd.service
/etc/systemd/system/multi-user.target.wants/certmonger.service
/etc/systemd/system/pki-tomcatd.target.wants/pki-tomcatd@pki-tomcat.service
/var/run/ipa/services.list
/etc/opendnssec/conf.xml
/etc/opendnssec/kasp.xml
/etc/ipa/dnssec/softhsm2.conf
/etc/ipa/dnssec/softhsm_pin_so
/etc/ipa/dnssec/ipa-ods-exporter.keytab
/etc/ipa/dnssec/ipa-dnskeysyncd.keytab
/etc/idm/nssdb/cert8.db
/etc/idm/nssdb/key3.db
/etc/idm/nssdb/secmod.db
/etc/ipa/nssdb/cert8.db
/etc/ipa/nssdb/key3.db
/etc/ipa/nssdb/secmod.db
```

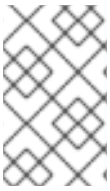
日志文件和目录：

```
/var/log/pki-ca
/var/log/pki/
/var/log/dirsrv/slapd-PKI-IPA
/var/log/httpd
/var/log/ipaserver-install.log
/var/log/kadmind.log
/var/log/pki-ca-install.log
/var/log/messages
/var/log/ipaclient-install.log
/var/log/secure
/var/log/ipaserver-uninstall.log
```

```
/var/log/pki-ca-uninstall.log
/var/log/ipaclient-uninstall.log
/var/named/data/named.run
```

9.2. 恢复备份

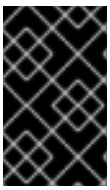
如果您有使用 ipa-backup 创建备份的目录，您可以将 IdM 服务器或 LDAP 内容恢复到执行备份时的状态。您不能在不同于最初创建备份的主机的主机上恢复备份。



注意

卸载 IdM 服务器不会自动删除此服务器的备份。

9.2.1. 从 Full-Server 或 Data-onlyly Backup 中恢复



重要

建议您先卸载服务器，然后再对其执行全服务器恢复。

全服务器和仅数据备份都使用 ipa-restore 工具进行恢复，必须始终以 root 用户身份运行。将备份传递给命令：

- 如果目录位于默认的 /var/lib/ipa/backup/ 目录中，则仅传递带有备份的目录名称。
- 如果包含备份的目录不在默认目录中，则向备份传递完整路径。例如：

```
[root@server ~]# ipa-restore /path/to/backup
```

ipa-restore 工具自动检测备份目录包含哪些类型的备份，默认执行同一类型的恢复。

您可以在 ipa-restore 中添加以下选项：

- --data 从全服务器备份中执行仅数据恢复，即，仅从包含全服务器备份的备份目录中恢复 LDAP 数据组件

- **--online 在仅在线恢复数据中恢复 LDAP 数据**

- **--instance 指定恢复哪些 389 DS 实例。Red Hat Enterprise Linux 7 中的 IdM 只使用 IPA-REALM 实例，但有可能使用单独的实例在带有独立实例的系统中创建备份；在这种情况下，--instance 允许您只恢复 IPA-REALM。例如：**

```
[root@server ~]# ipa-restore --instance=IPA-REALM /path/to/backup
```

您只能在执行仅数据恢复时使用这个选项。

- **--backend 指定恢复了哪些后端；如果没有这个选项，ipa-restore 会恢复它发现的所有后端。定义可能的后端的参数是 userRoot，它会恢复 IPA 数据后端和 ipaca，这会恢复 CA 后端。**

您只能在执行仅数据恢复时使用这个选项。

- **--no-logs 在不恢复日志文件的情况下恢复备份**

为了避免 IdM master 上的身份验证问题，请在恢复后清除 SSSD 缓存：

1. **停止 SSSD 服务：**

```
[root@server ~]# systemctl stop sssd
```

2. **从 SSSD 中删除所有缓存的内容：**

```
[root@server ~]# find /var/lib/sss/ ! -type d | xargs rm -f
```

3. **启动 SSSD 服务：**

```
[root@server ~]# systemctl start sssd
```

注意

建议您在从备份恢复后重启您的系统。

有关使用 ipa-restore 的详情，请参考 ipa-restore(1) man page。

9.2.2. 使用多个主服务器恢复

有关在多 master 复制环境中恢复 IdM 的详情，请参阅 “IdM 中的备份和恢复”。

9.2.3. 从加密备份中恢复

如果要从 GPG 加密的备份中恢复，请使用 --gpg-keyring 选项提供私钥和公钥的完整路径。例如：

```
[root@server ~]# ipa-restore --gpg-keyring=/root/backup /path/to/backup
```

第 10 章 为 IDM 用户定义访问控制

访问控制是一组安全功能，用于定义谁可以访问某些资源，如机器、服务或条目等，以及它们允许执行的操作类型。身份管理提供了多个访问控制区域，以便明确授予哪些访问类型以及授予谁。因此，身份管理区分了对域中资源的访问控制和对 IdM 配置本身的访问控制。

本章详细介绍了 IdM 服务器中用户对 IdM 服务器和其他 IdM 用户可用的不同内部访问控制机制。

10.1. IDM 条目的访问控制

访问控制定义了授予用户对其他用户或对象执行操作的权限或权限。

身份管理访问控制结构基于标准的 LDAP 访问控制。IdM 服务器中的访问是基于存储在后端目录服务器实例的 IdM 用户，它们允许访问其他 IdM 实体，也作为 LDAP 条目存储在目录服务器实例中。

访问控制指令(ACI)有三个部分：

actor

这是被授予执行操作权限的实体。在 LDAP 访问控制模型中，这称为 *绑定规则*，因为它定义了用户是，并可选择性地对绑定尝试进行其他限制，如限制尝试一天或特定机器。

目标

这将定义允许行动者对其执行操作的条目。

操作类型

操作类型 - 最后一个部分决定了用户被允许执行的操作类型。最常见的操作有 add、delete、write、read 和 search。在身份管理中，所有用户都会隐式授予 IdM 域中所有条目的读和搜索权限，对密码和 Kerberos 密钥等敏感属性的限制。匿名用户受到与安全性相关的配置的限制，如 sudo 规则和基于主机的访问控制。

当尝试任何操作时，IdM 客户端的第一个操作是发送用户凭证，作为 bind 操作的一部分。后端目录服务器检查这些用户凭证，然后检查用户帐户以查看用户是否有权限来执行所请求的操作。

10.1.1. 身份管理中的访问控制方法

要使访问控制规则简单且明确实现，身份管理会将访问控制定义分成三个类别：

自助服务规则

自助服务规则，定义用户可以根据自己的个人条目执行哪些操作。访问控制类型仅允许对条目内的属性进行写入权限；它不允许为条目本身添加或删除操作。

委派规则

委派规则，允许特定用户组对另一用户组中用户的特定属性执行写入（编辑）操作。与自助服务规则一样，这种形式的访问控制规则仅限于编辑特定属性的值；它不授予添加或删除整个条目或控制未指定属性的功能。

基于角色的访问控制

基于角色的访问控制，它会创建特殊的访问控制，然后对 IdM 域中的所有实体授予更广泛的权威。可以授予角色编辑、添加和删除权限，即可以授予对整个条目的完整控制权限，而不仅限于选择的属性。

一些角色已在身份管理中创建并可用。可以创建特殊角色来管理任何类型的条目，如主机、自动挂载配置、网络组、DNS 设置和 IdM 配置。

10.2. 定义自助服务设置

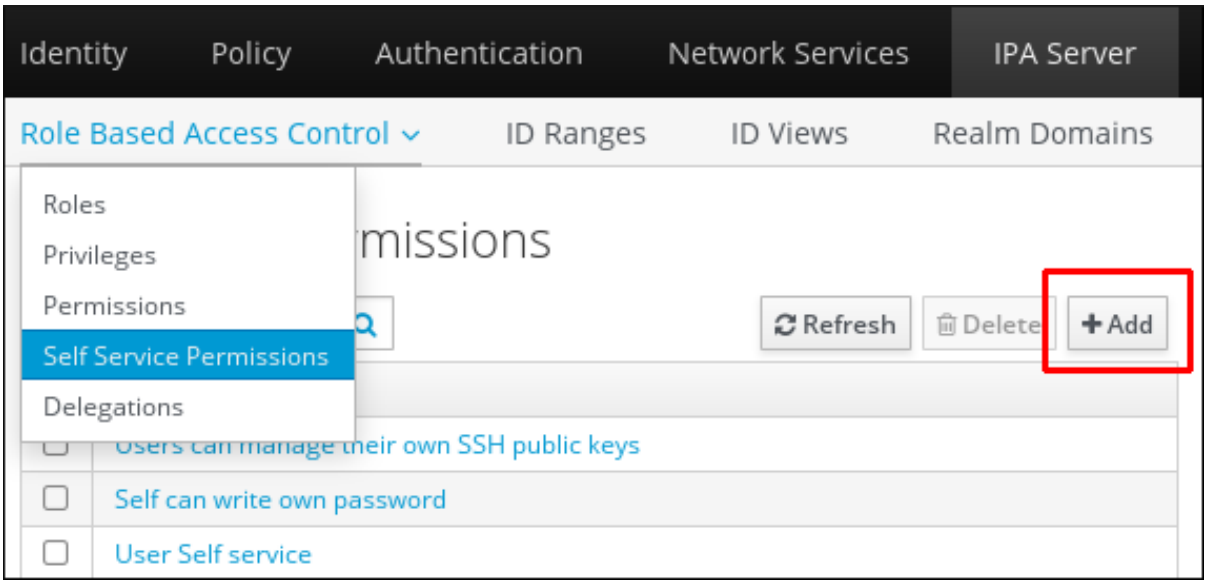
自助服务访问控制规则定义实体可以自己执行的操作。这些规则仅定义用户（或其他 IdM 实体）可在其个人条目上编辑哪些属性。

10.2.1. 从 Web UI 创建自助服务规则

1. 在顶部菜单中的 IPA Server 选项卡中，选择 基于角色的访问控制自助服务权限 子选项卡。

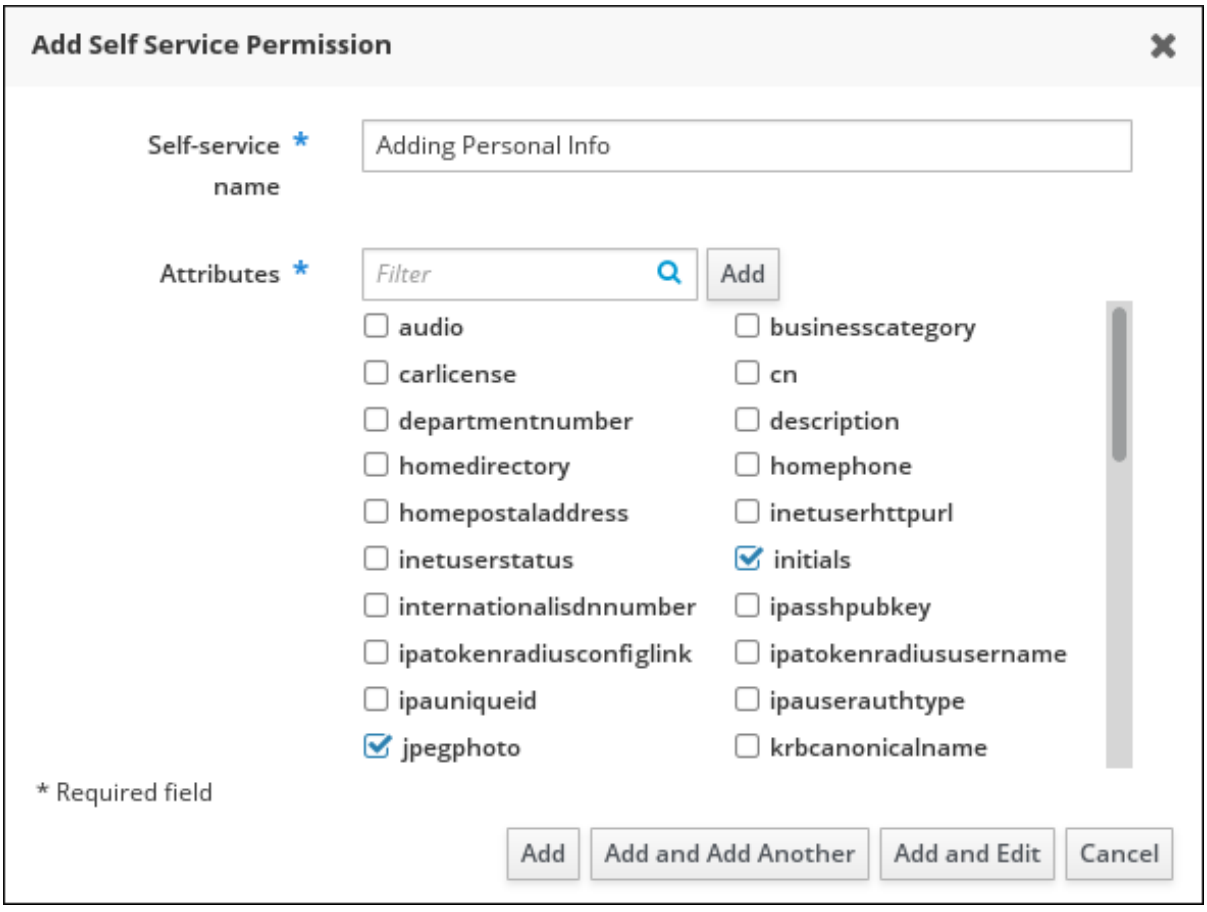
2. 单击自助服务访问控制指令列表顶部的 Add。

图 10.1. 添加当前自助服务规则



3. 在弹出窗口中输入规则的名称。允许使用空格。

图 10.2. 添加自助服务规则表格



4. 选中此 ACI 允许用户编辑的属性所对应的复选框。

5. 点 Add 按钮保存新的自助服务 ACI。

10.2.2. 从命令行创建自助服务规则

可以使用 selfservice-add 命令添加新的自助服务规则。这两个选项是必需的：

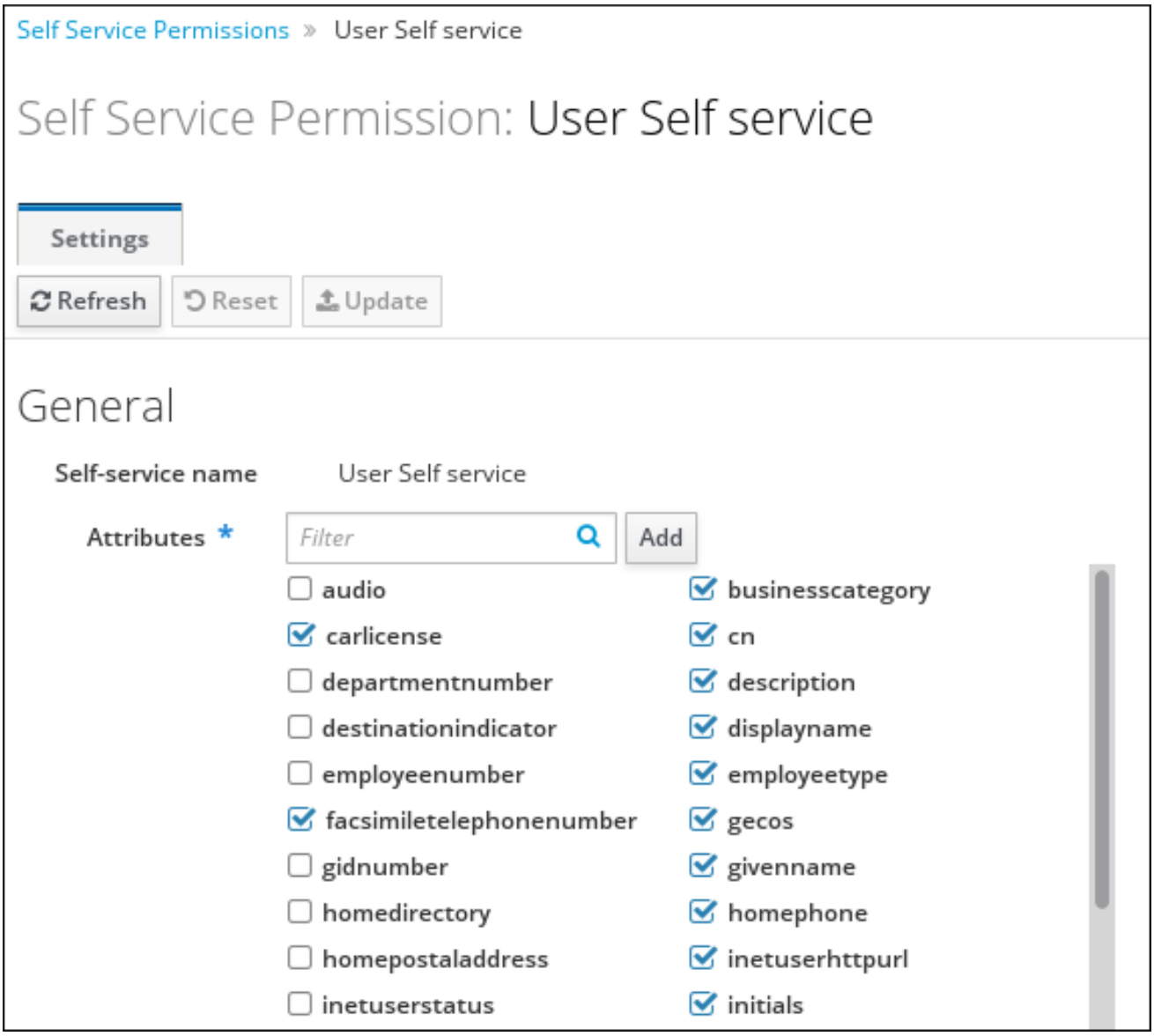
- --permissions 用于设置 ACI 授予的权限（如写入、添加或删除）
- --attrs 用于提供 ACI 授予的权限的完整属性列表。

```
[jsmith@server ~]$ ipa selfservice-add "Users can manage their own name details" --
permissions=write --attrs=givenname --attrs=displayname --attrs=title --attrs=initials
---------------------------------------------------------
Added selfservice "Users can manage their own name details"
---------------------------------------------------------
  Self-service name: Users can manage their own name details
  Permissions: write
  Attributes: givenname, displayname, title, initials
```

10.2.3. 编辑自助服务规则

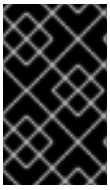
在 Web UI 中的自助服务条目中，唯一可以编辑的元素是 ACI 中包含的属性列表。可以选择或取消选中复选框。

图 10.3. 自助服务编辑页面



在命令行中，使用 ipa selfservice-mod 命令编辑自助服务规则。--attrs 选项覆盖先前支持的属性列表，因此始终包括属性的完整列表以及任何新属性。

```
[jsmith@server ~]$ ipa selfservice-mod "Users can manage their own name details" --
attrs=givenname --attrs=displayname --attrs=title --attrs=initials --attrs=surname
---------------------------------------------------------------
Modified selfservice "Users can manage their own name details"
---------------------------------------------------------------
Self-service name: Users can manage their own name details
Permissions: write
Attributes: givenname, displayname, title, initials
```



重要

在修改自助服务规则时包括所有属性，包括现有的属性。

10.3. 为用户委派权限

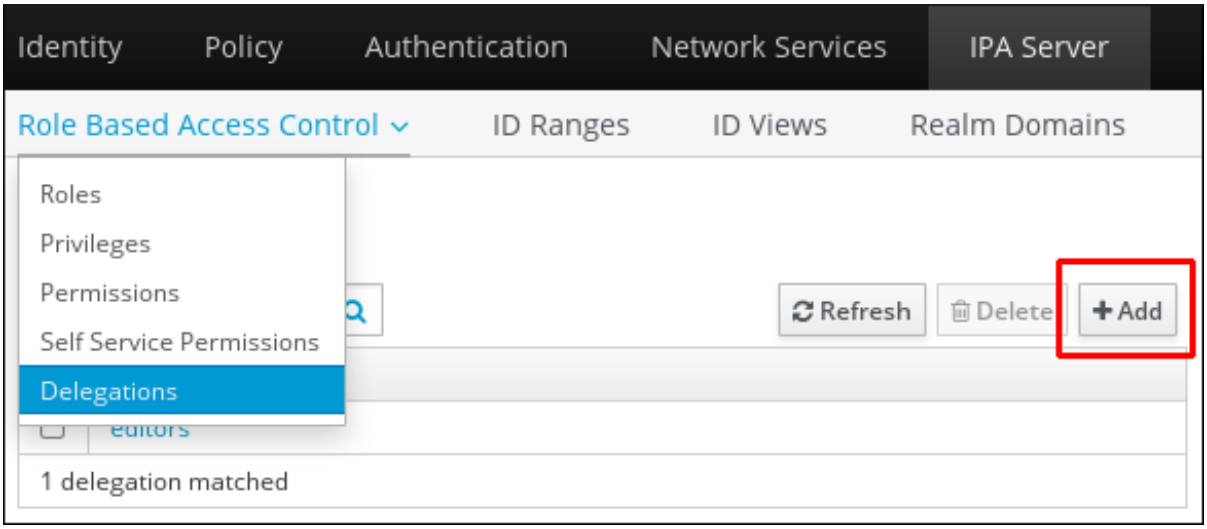
委派与角色非常相似，因为为一组用户分配了管理另一组用户条目的权限。但是，委派的授权与授予完整访问权限但仅授予特定用户属性（而非整个条目）的自助服务规则更加相似。另外，委派的颁发机构中的组是现有的 IdM 用户组，而不是为访问控制特别创建的角色。

10.3.1. 在 Web UI 中委派用户访问用户组

1. 在顶部菜单中的 **IPA Server** 选项卡中，选择 **基于角色的访问控制委派** → 子选项卡。

2. 单击委派访问控制指令列表顶部的 **Add** 链接。

图 10.4. 添加新委派



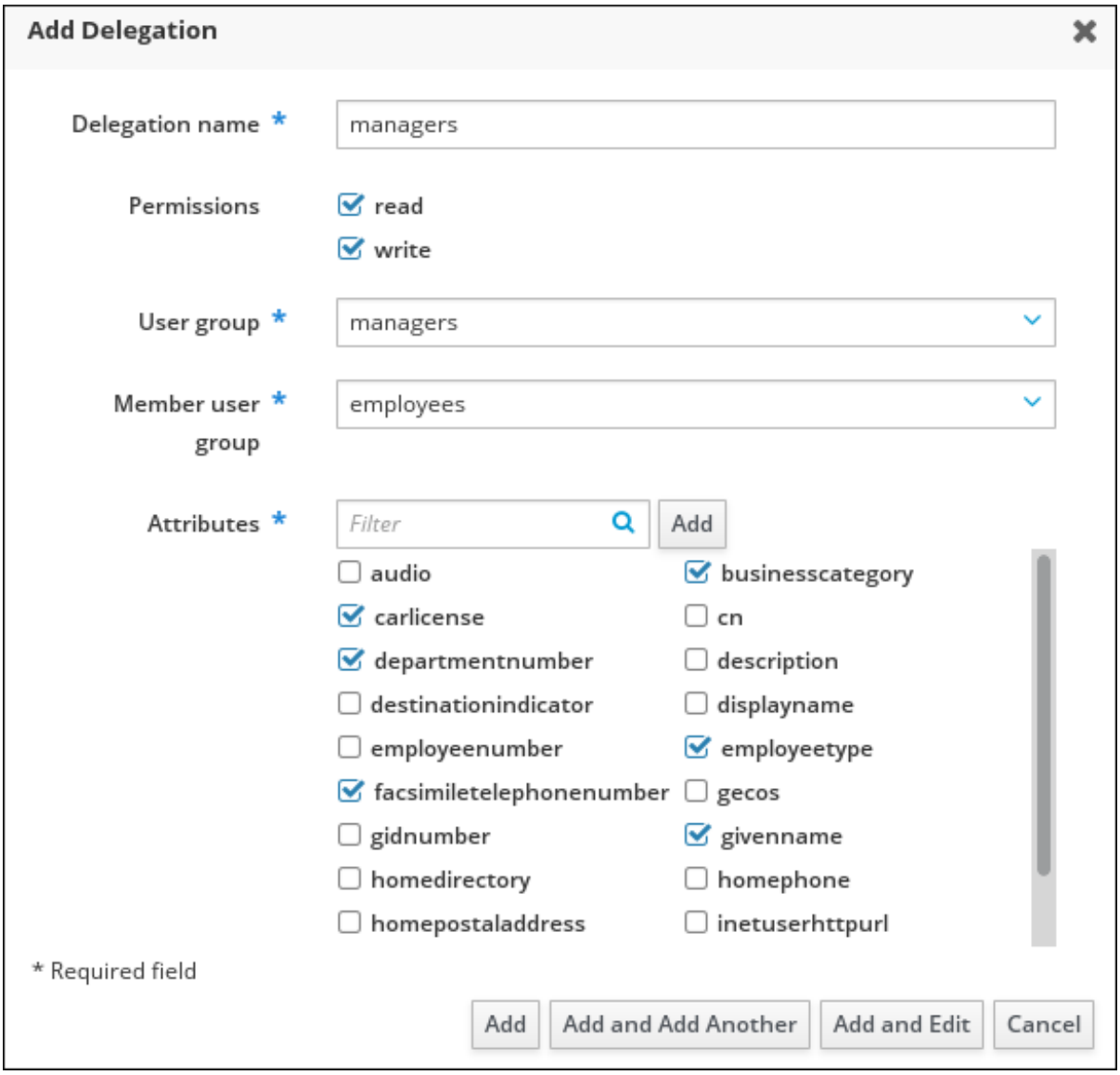
3. 将新委派命名为 **ACI**。

4. 通过选中复选框来设置权限，用户是否有权查看给定属性（读取），并添加或更改给定属性（写入）。

   某些用户可能具有查看信息，但不应能够对其进行编辑。

5. 在 **User group** 下拉菜单中，选择为用户组中用户条目 *授予权限的* 组。

图 10.5. 添加委派表格



6. **在 Member user group 下拉菜单中，选择*其条目可以被委派组的成员编辑* 的组。**

7. **在属性框中，根据授予 member 用户组权限的属性选中复选框。**

8. 单击 **Add 按钮，以保存新的委派 ACI。**

10.3.2. 在命令行中委派用户访问用户组

使用 delegation-add 命令添加新的委派访问控制规则。需要三个参数：

- **--group，*即被授予用户组中用户条目权限* 的组。**

- **--membergroup，委派组成员 *可编辑其条目* 的组。**

- **--attrs，允许成员组中的用户查看或编辑的属性。**

例如：

```
$ ipa delegation-add "basic manager attrs" --attrs=manager --attrs=title --attrs=employeetype --
attrs=employeenumber --group=engineering_managers --membergroup=engineering
-------------------------------------
Added delegation "basic manager attrs"
-------------------------------------
  Delegation name: basic manager attrs
  Permissions: write
  Attributes: manager, title, employeetype, employeenumber
  Member user group: engineering
  User group: engineering_managers
```

使用 delegation-mod 命令编辑委派规则。--attrs 选项覆盖先前支持的属性列表，因此始终包括属性的完整列表以及任何新属性。

```
[jsmith@server ~]$ ipa delegation-mod "basic manager attrs" --attrs=manager --attrs=title --
attrs=employeetype --attrs=employeenumber --attrs=displayname
---------------------------------------
Modified delegation "basic manager attrs"
---------------------------------------
  Delegation name: basic manager attrs
  Permissions: write
  Attributes: manager, title, employeetype, employeenumber, displayname
  Member user group: engineering
  User group: engineering_managers
```

重要

在修改委派规则时包括所有属性，包括现有属性。

10.4. 定义基于角色的访问控制

与自助服务和授权访问控制相比，基于角色的访问控制为用户授予截然不同的权限。基于角色的访问控制本质上是管理性的，提供修改条目的功能。

基于角色的访问控制有三个部分：*权限*、*特权和 角色*。特权由一个或多个权限组成，角色由一个或多

个特权组成。

- *权限* 定义了特定操作或一组操作（如读取、写入、添加或删除）以及这些操作应用到的 IdM LDAP 目录中的目标条目。权限是构建块；可以根据需要将其分配给多个特权。

  使用 IdM 权限，您可以控制哪些用户有权访问哪些对象，甚至控制这些对象的属性。IdM 可让您将所有匿名用户、所有经过身份验证的用户、或仅一组特权用户列入白名单或黑名单，或更改特定 IdM 功能（如用户、组或 sudo）的整个可见性。在这样的情形中，这种灵活的权限方法非常有用，例如，管理员希望将用户或组的访问限制为这些用户或组需要访问的特定部分，并使其他部分完全隐藏。

- *特权* 是可应用到角色的一组权限。例如，可以创建用于添加、编辑和删除自动挂载位置的权限。然后，可以将该权限与管理 FTP 服务相关的其他权限合并，并可用于创建与管理文件系统相关的单一特权。

  注意

  在红帽身份管理环境中，特权对原子访问控制单元具有非常具体的含义，即创建权限和角色。红帽身份管理中不存在作为常规用户临时获得额外特权的 *特权升级*。使用基于角色的访问控制(RBAC)将特权分配给用户。用户具有授予访问权限的角色，或者不授予访问权限。

  除用户外，还将特权分配到用户组、主机、主机组和网络服务。这种做法允许一组通过特定网络服务对一组主机上的一组用户进行精细控制操作。

- *角色是为角色* 指定的用户拥有的特权列表。

  重要

  角色用于对允许的操作进行分类。它们不用作实施特权升级或防止特权升级的工具。

可以创建全新的权限，以及基于现有权限或新权限创建新的权限。红帽身份管理提供以下一系列预定义角色：

表 10.1. 红帽身份管理中的预定义角色

角色	特权	Description
Helpdesk	Modify Users and Reset passwords, Modify Group membership	负责执行简单的用户管理任务
IT Security Specialist	Netgroups Administrators, HBAC Administrator, Sudo Administrator	负责管理安全策略，如基于主机的访问控制、sudo 规则
IT Specialist	Host Administrators, Host Group Administrators, Service Administrators, Automount Administrators	负责管理主机
Security Architect	Delegation Administrator, Replication Administrators, Write IPA Configuration, Password Policy Administrator	负责管理身份管理环境、创建信任、创建复制协议
User Administrator	User Administrators, Group Administrators, Stage User Administrators	负责创建用户和组

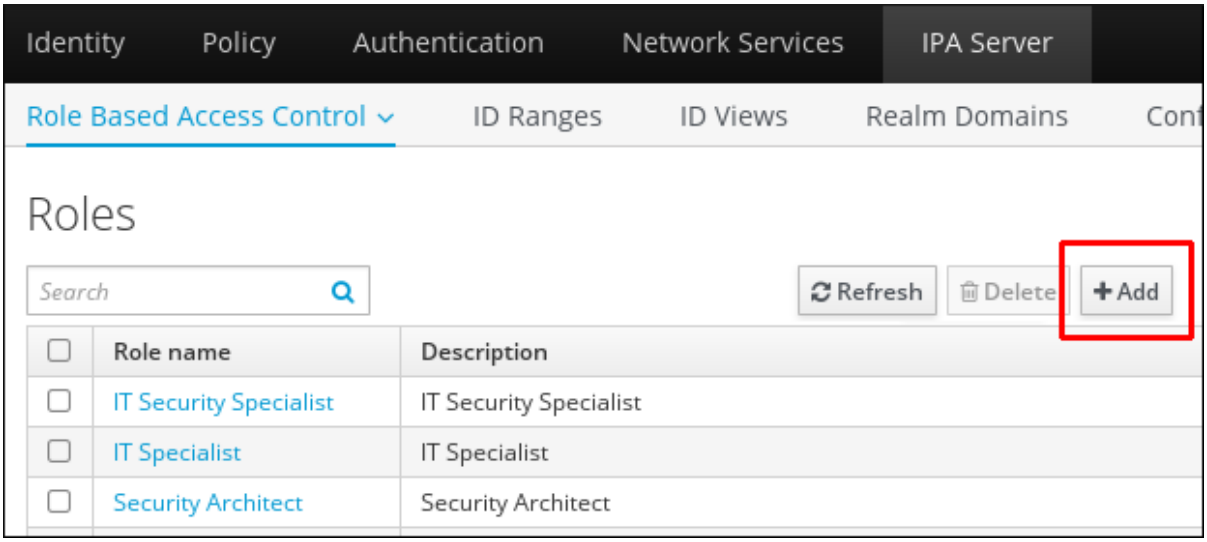
10.4.1. 角色

10.4.1.1. 在 Web UI 中创建角色

1. **打开顶部菜单中的 IPA Server 选项卡，然后选择 基于角色的访问控制 子选项卡。**

2. **单击基于角色的访问控制指令列表顶部的 Add 链接。**

图 10.6. 添加新角色



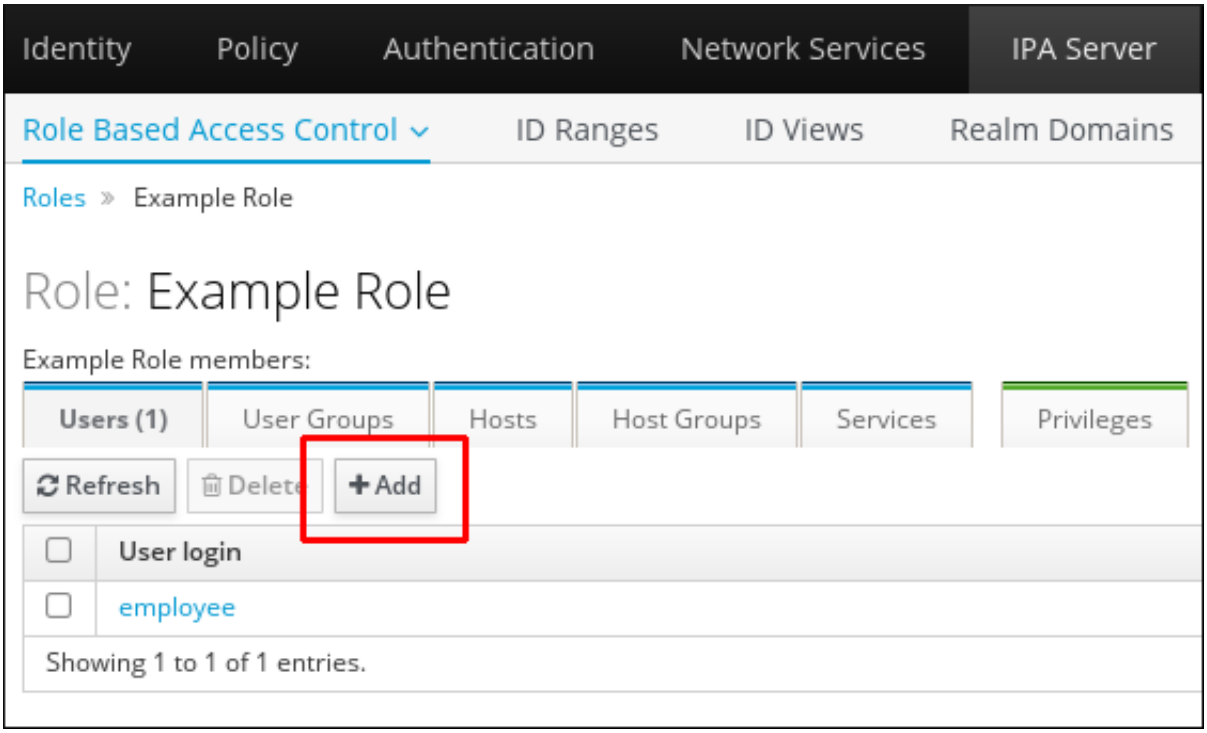
3. 输入角色名称和描述。

图 10.7. 用于添加角色的表单

4. 单击 **Add and Edit** 按钮，以保存新角色，再进入配置页面。

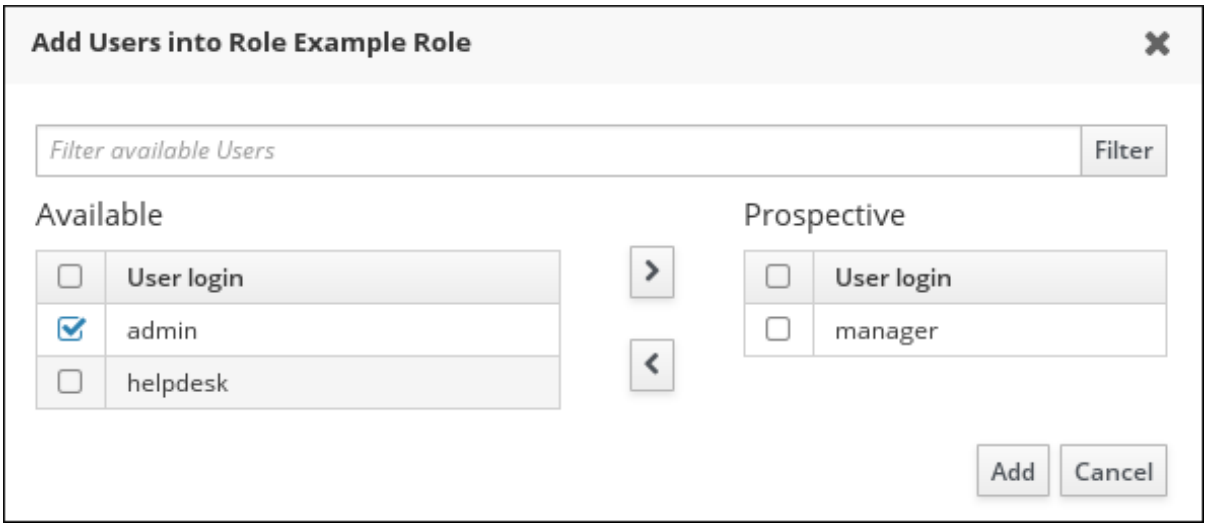
5. 在 **Users** 选项卡的顶部，或者在添加组时在 **Users Groups** 选项卡中点 **Add**。

图 10.8. 添加用户



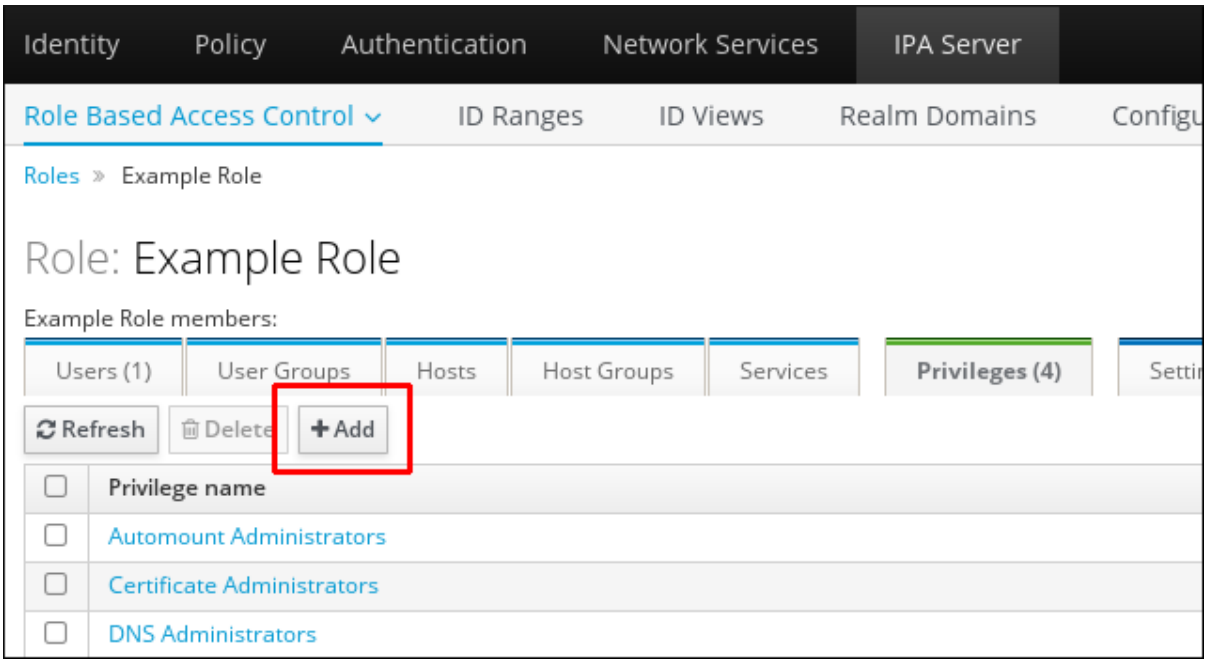
6. 选择左侧的用户，并使用 > 按钮将它们移到 **Prospective** 列中。

图 10.9. 选择用户



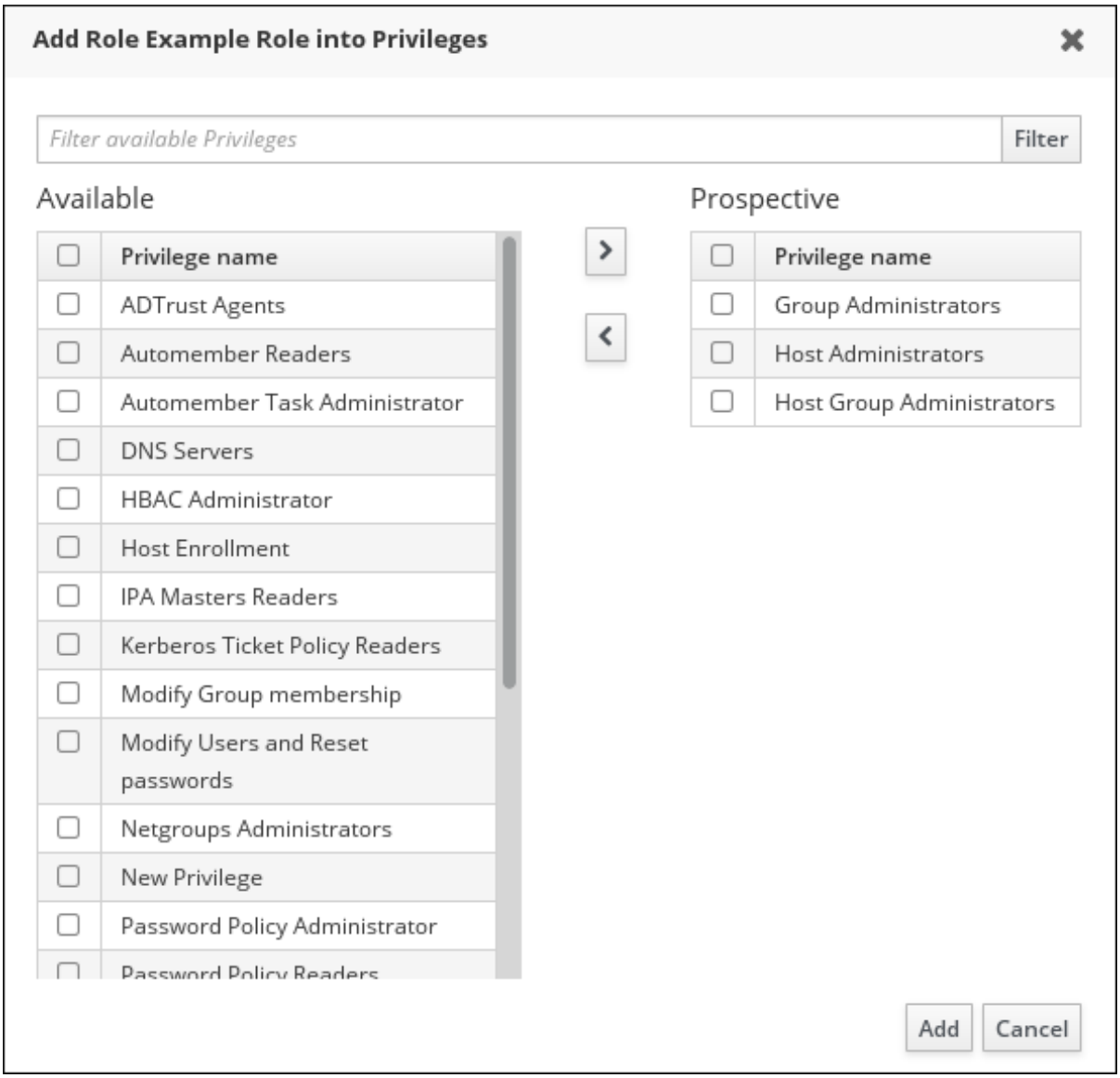
7. 在 Privileges 选项卡的顶部，单击 Add。

图 10.10. 添加特权



8. 选择左侧的特权，并使用 > 按钮将它们移到 Prospective 列中。

图 10.11. 选择特权



9. 单击 **Add** 按钮保存。

10.4.1.2. 在命令行中创建角色

1. **添加新角色：**

```
[root@server ~]# kinit admin
[root@server ~]# ipa role-add --desc="User Administrator" useradmin
  ----------------------
  Added role "useradmin"
  ----------------------
  Role name: useradmin
  Description: User Administrator
```

2. 为角色添加所需的权限：

```
[root@server ~]# ipa role-add-privilege --privileges="User Administrators" useradmin
  Role name: useradmin
  Description: User Administrator
  Privileges: user administrators
  ----------------------------
  Number of privileges added 1
----------------------------
```

3. 将所需的组添加到角色。在这种情况下，我们只添加一个单独的组 **useradmins**，该组已存在。

```
[root@server ~]# ipa role-add-member --groups=useradmins useradmin
  Role name: useradmin
  Description: User Administrator
  Member groups: useradmins
  Privileges: user administrators
  -------------------------
  Number of members added 1
-------------------------
```

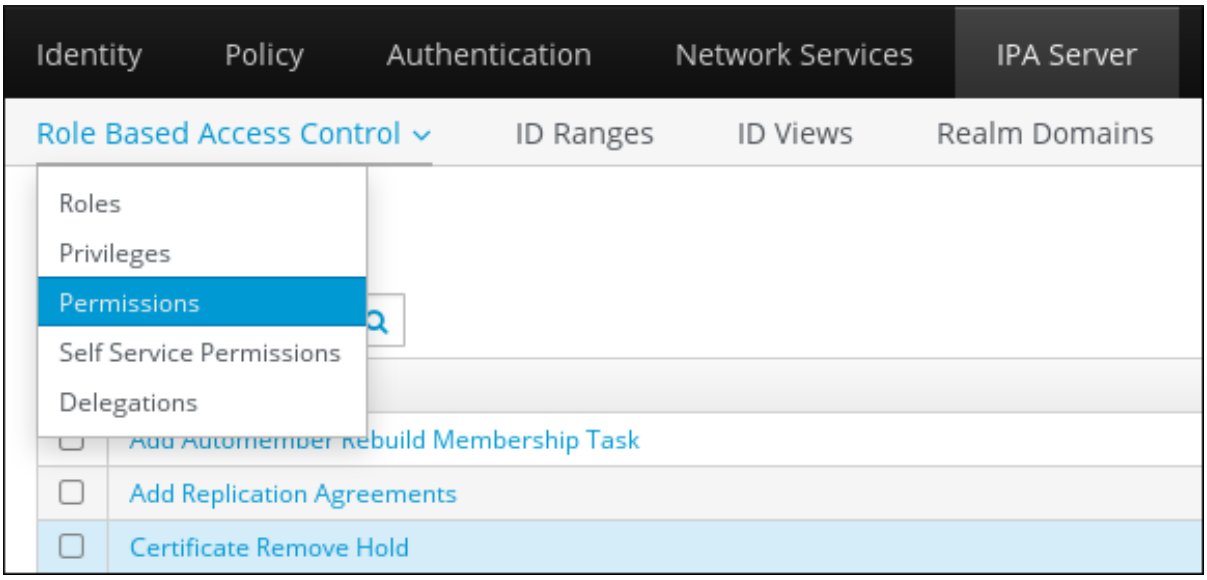
10.4.2. 权限

10.4.2.1. 从 Web UI 创建新权限

1. 打开顶部菜单中的 **IPA Server** 选项卡，然后选择 **基于角色的访问控制** 子选项卡。

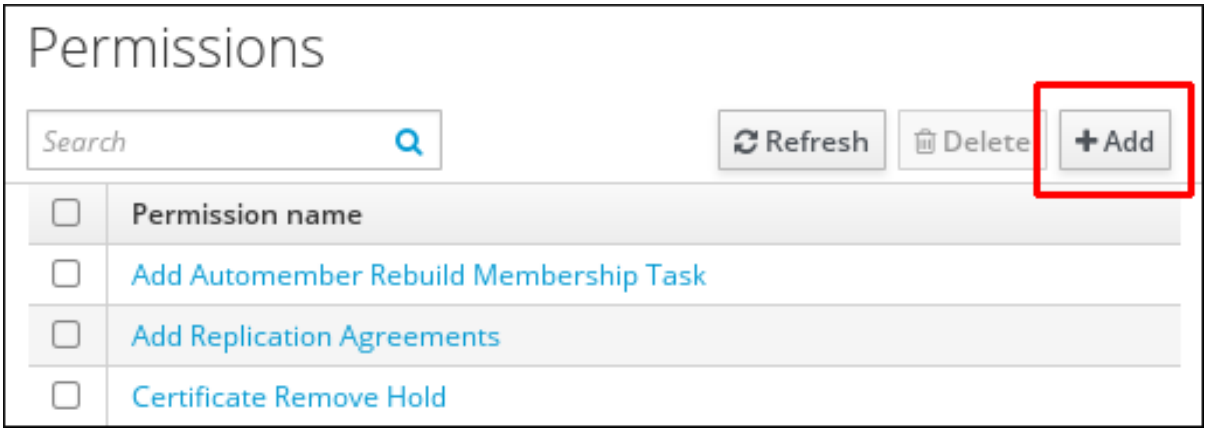
2. 选择 **Permissions** 任务链接。

图 10.12. 权限任务



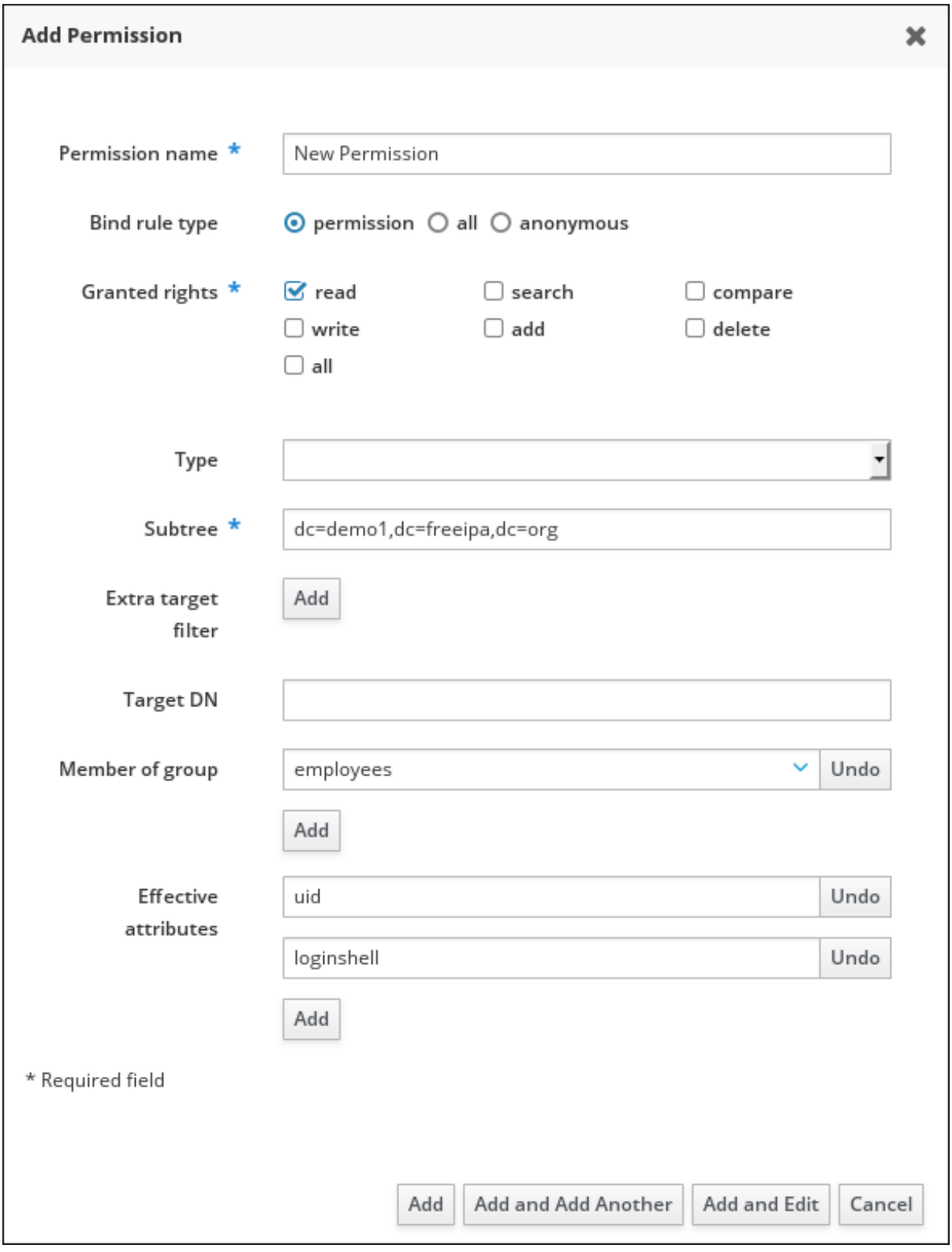
3. 单击权限列表顶部的 **Add** 按钮。

图 10.13. 添加新权限



4. 以显示的形式定义新权限的属性。

图 10.14. 添加权限表格



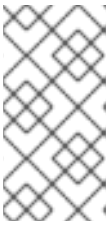
5. 点表单下的 **Add** 按钮保存权限。

您可以指定以下权限属性：

1. 输入新权限的名称。

2. 选择适当的 绑定规则类型 ：

   - permission 是默认的权限类型，通过特权和角色授予访问权限
   - all 指定权限适用于所有经过身份验证的用户
   - anonymous 指定权限适用于所有用户，包括未经身份验证的用户



注意

不能对特权添加带有非默认绑定规则类型的权限。您也不能对非默认绑定规则类型设置特权中已存在的权限。

3. 选择授予权限 中授予权限 的权限。

4. 定义方法来标别权限的目标条目：

   - Type 指定条目类型，如 user、host 或 service。如果您为 Type 设置选择了一个值，则可通过该 ACI 访问该条目类型的所有可能属性的列表将出现在 Effective Attributes 下。

     定义 Type 会将 Subtree 和 Target DN 设置为其中一个预定义的值。

   - 子树指定一个子树条目；然后，此子树条目下的每个条目都会作为目标。提供现有的子树条目，因为 Subtree 不接受通配符或不存在的域名(DN)。例如：

     ```
     cn=automount,dc=example,dc=com
     ```

   - 额外目标过滤器 使用 LDAP 过滤器来识别权限将应用到哪个条目。过滤器可以是任何有效的 LDAP 过滤器，例如：

```
(!(objectclass=posixgroup))
```

IdM 自动检查给定过滤器的有效性。如果您输入了一个无效的过滤器，在尝试保存权限后 IdM 会警告您有关这个权限的信息。

- 目标 DN 指定域名(DN)，并接受通配符。例如：

```
uid=*,cn=users,cn=accounts,dc=com
```

- 组成员 对给定组的成员设置目标过滤器。

填写过滤器设置并点击 Add 后，IdM 会验证过滤器。如果所有权限设置都正确，IdM 将执行搜索。如果某些权限设置不正确，IdM 将显示一条消息，通知您哪个设置不正确。

5. 如果设置了 Type，请从可用的 ACI 属性列表中选择 Effective attributes。如果您没有使用 Type，通过将属性写入Effective attributes 字段来手动添加属性。一次添加一个属性；若要添加多个属性，可单击 Add 来添加另一个输入字段。

重要

如果您没有为权限设置任何属性，则默认包含所有属性。

10.4.2.2. 从命令行创建新权限

要添加新权限，请发出 ipa permission-add 命令。通过提供对应的选项来指定权限的属性：

- 提供权限的名称。例如：

```
[root@server ~]# ipa permission-add "dns admin permission"
```

- --bindtype 指定绑定规则类型。此选项接受 all、anonymous 和 permission 参数。例如：

```
--bindtype=all
```

如果不使用 **--bindtype**，则类型会自动设置为 默认权限 值。

注意

不能对特权添加带有非默认绑定规则类型的权限。您也不能对非默认绑定规则类型设置特权中已存在的权限。

- **--permissions** 列出权限授予的权限。您可以使用多个 **--permissions** 选项或通过在大括号内以逗号分隔的列表中列出选项来设置多个属性。例如：

```
--permissions=read --permissions=write
--permissions={read,write}
```

- **--attrs** 提供授予权限的属性列表。您可以使用多个 **--attrs** 选项或通过在大括号内以逗号分隔的列表列出选项，来设置多个属性。例如：

```
--attrs=description --attrs=automountKey
--attrs={description,automountKey}
```

使用 **--attrs** 提供的属性必须存在，并且是给定对象类型的允许属性，否则命令会失败，并显示模式语法错误。

- **--type** 定义条目对象类型，如 **user**、**host** 或 **service**。每种类型都有自己的一组允许的属性。例如：

```
[root@server ~]# ipa permission-add "manage service" --permissions=all --
type=service --attrs=krbprincipalkey --attrs=krbprincipalname --attrs=managedby
```

- **--subtree** 提供子树条目；然后，过滤器以这个子树条目下的每个条目为目标。提供现有的子树条目；**--subtree** 不接受通配符或不存在的域名(DN)。在 目录中包含 DN。

因为 IdM 使用简化的扁平目录树结构，所以 **--subtree** 可用于将某些类型的条目作为目标，如自动挂载位置，它们是其他配置的容器或父条目。例如：

```
[root@server ~]# ipa permission-add "manage automount locations" --
subtree="ldap://ldap.example.com:389/cn=automount,dc=example,dc=com" --
permissions=write --attrs=automountmapname --attrs=automountkey --
attrs=automountInformation
```

--type 和 --subtree 选项是互斥的。

- --filter 使用 LDAP 过滤器来识别权限应用到哪个条目。IdM 自动检查给定过滤器的有效性。过滤器可以是任何有效的 LDAP 过滤器，例如：

```
[root@server ~]# ipa permission-add "manage Windows groups" --filter="(!
(objectclass=posixgroup))" --permissions=write --attrs=description
```

- 检查组是否存在后，--memberof 对给定组的成员设置目标过滤器。例如：

```
[root@server ~]# ipa permission-add ManageHost --permissions="write" --
subtree=cn=computers,cn=accounts,dc=testrelm,dc=com --attr=nshostlocation --
memberof=admins
```

- 在检查组存在后，--targetgroup 对指定的用户组设置目标。

Web UI 中提供的 Target DN 设置不会在命令行中可用。

注意

有关修改和删除权限的详情，请运行 ipa permission-mod --help 和 ipa permission-del --help 命令。

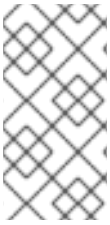
10.4.2.3. 默认管理的权限

管理权限 是预装了身份管理的权限。它们的行为与用户创建的其他权限类似，但有以下区别：

- 您不能修改其名称、位置和目标属性。
- 您不能删除它们。
- 它们有三组属性：
  - *默认* 属性，由 IdM 管理，用户无法修改它们
  - *包含* 的属性，这是用户添加的额外属性；要将 include 属性添加到受管权限，请使用 ipa permission-mod 命令提供 --includedattrs 选项指定属性
  - *排除* 的属性，它们是用户删除的属性；要将 exclude 属性添加到受管权限，请使用 ipa permission-mod 命令提供 --excludedattrs 选项指定属性

管理的权限适用于 default 和 included 属性集中显示的所有属性，但不应用到排除集中的所有属性。

如果您在修改受管权限 时使用 --attrs 选项，则包含和 exclude 属性集会自动调整，以便仅启用由 --attrs 提供的属性。



注意

虽然您无法删除受管权限，但将其绑定类型设置为 权限，并从所有特权中删除受管权限会有效地禁用它。

所有受管权限的名称都以 System 开头，例如 *System：添加 Sudo 规则* 或 *System:修改服务*。

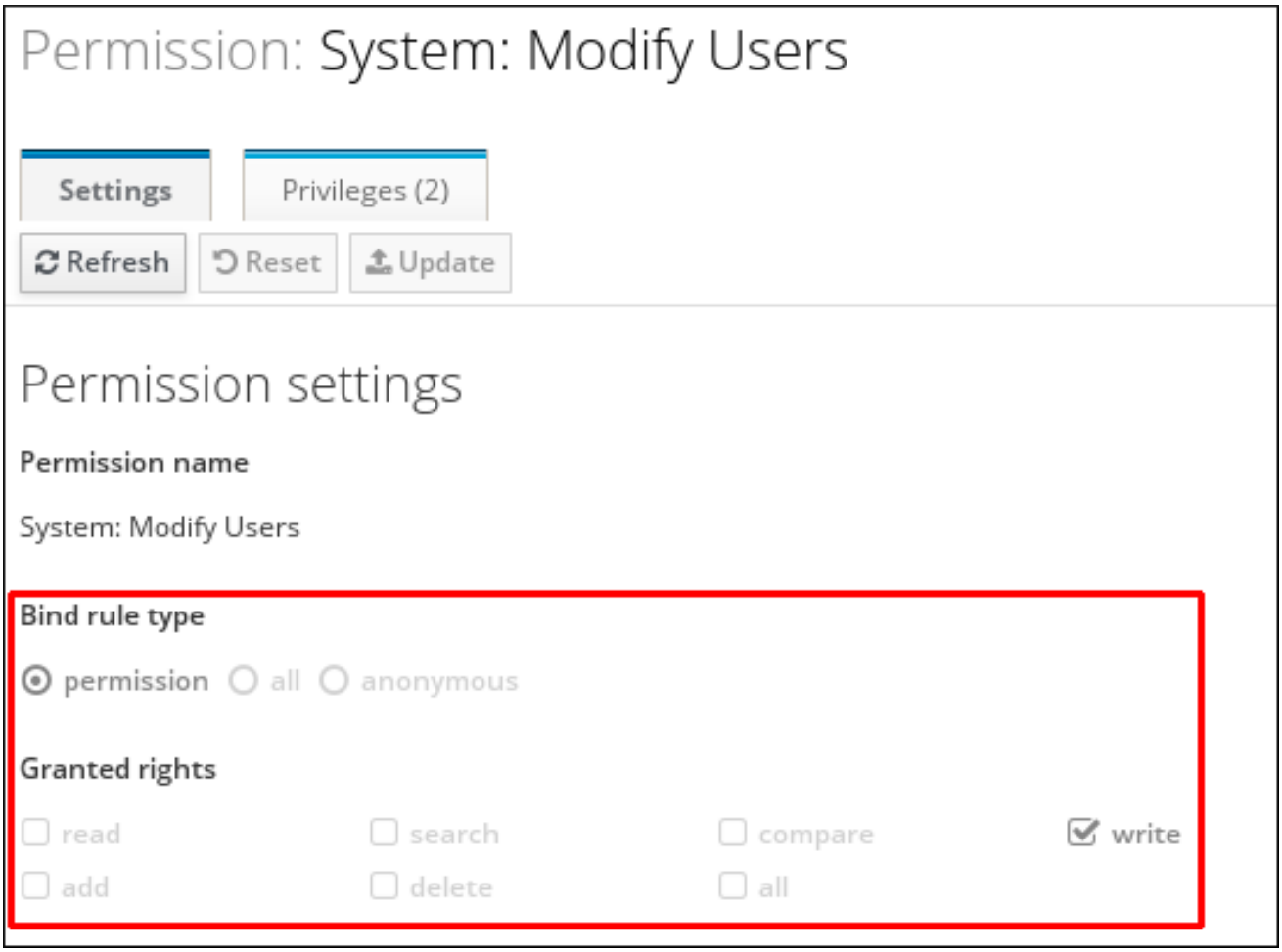
IdM 的早期版本使用不同的默认权限方案，例如，禁止用户修改默认权限，用户只能将它们分配给特权。这些默认权限大部分已转换为受管权限，但以下权限仍使用以前的方案：

- 添加自动成员重新构建成员身份任务

- 添加复制协议
- 证书删除冻结
- 从 CA 获取证书状态
- 修改 DNA 范围
- 修改复制协议
- 删除复制协议
- 请求证书
- 从不同主机请求证书
- 从 CA 检索证书
- 吊销证书
- 写入 IPA 配置

如果您尝试从 Web UI 修改受管权限，则不会修改的属性将被禁用。

图 10.15. 禁用的属性



如果您尝试从命令行修改受管权限，系统将不允许您更改无法修改的属性。例如：尝试更改默认 系统：修改 Users 权限以应用到组失败：

```
$ ipa permission-mod 'System: Modify Users' --type=group
ipa: ERROR: invalid 'ipapermlocation': not modifiable on managed permissions
```

但是，您可以使 System：修改 Users 权限，以应用到 GECOS 属性：

```
$ ipa permission-mod 'System: Modify Users' --excludedattrs=gecos
-----------------------------------------
Modified permission "System: Modify Users"
```

10.4.2.4. 较早版本的身份管理中的权限

早期版本的身份管理处理不同的权限，例如：

- 全局 IdM ACI 授予服务器所有用户（即使是匿名用户）的读取访问权限，即不是经过身份验证的用户。

- 仅可使用写入、添加和删除权限类型。读取权限也可用，但实际价值不大，因为包括未经身份验证的用户（包括未经身份验证的用户）默认具有读取访问权限。

当前的身份管理版本包含用于设置权限的选项，这些权限更加精细：

- 全局 IdM ACI 不会向未经身份验证的用户授予读取访问权限。
- 现在，可以在同一权限中添加过滤器和子树。
- 可以添加搜索和比较权限。

新的处理权限的方式大大改进了 IdM 的功能来控制用户或组访问，同时保持与较早版本的向后兼容性。从早期版本的 IdM 升级会删除所有服务器上的全局 IdM ACI，并使用 *受管权限* 替换它。

每当您修改时，通过先前方式创建的权限将自动转换为当前风格。如果您不尝试更改它们，则上一类型的权限保持不变。且权限使用当前样式，它永远不会降级到上一样式。

注意

在运行较早版本的 IdM 的服务器中，仍可以为权限分配权限。

ipa permission-show 和 ipa permission-find 命令可识别当前权限和之前样式的权限。这两个命令的输出都以当前样式显示权限，但权限本身保持不变；命令在仅输出数据之前升级权限条目，而不向 LDAP 提交更改。

以上和当前特征的权限会对所有运行 IdM 版本的服务器以及运行当前 IdM 版本的服务器产生影响。但是，您不能在运行之前版本的 IdM 服务器上使用当前权限创建或修改权限。

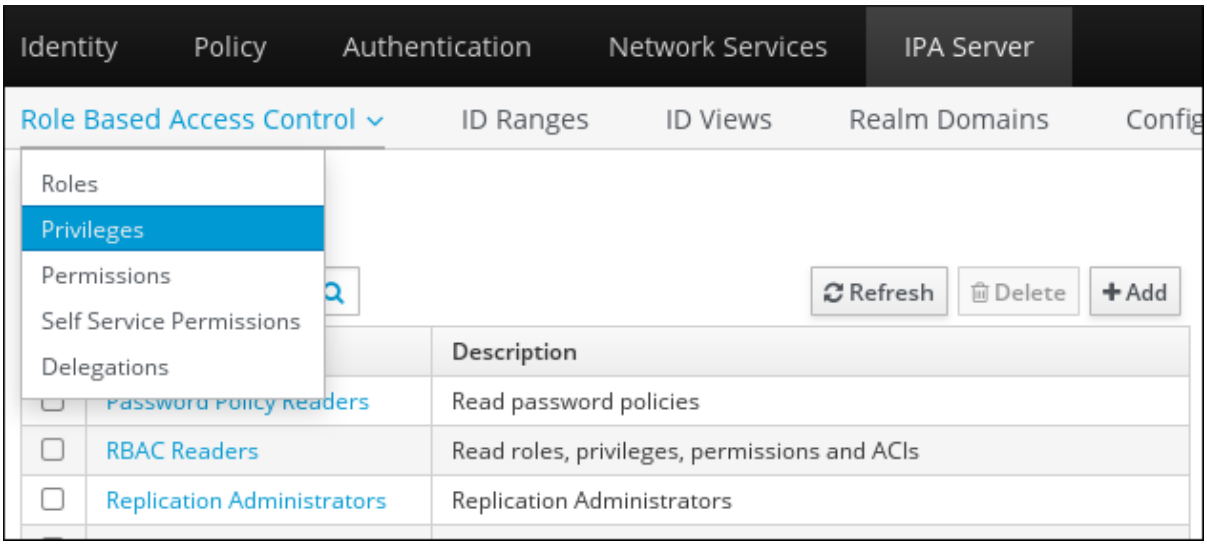
10.4.3. 权限

10.4.3.1. 从 Web UI 创建新特权

1. **打开顶部菜单中的 IPA Server 选项卡，然后选择 基于角色的访问控制 子选项卡。**

2. **选择 Privileges 任务链接。**

图 10.16. 权限任务

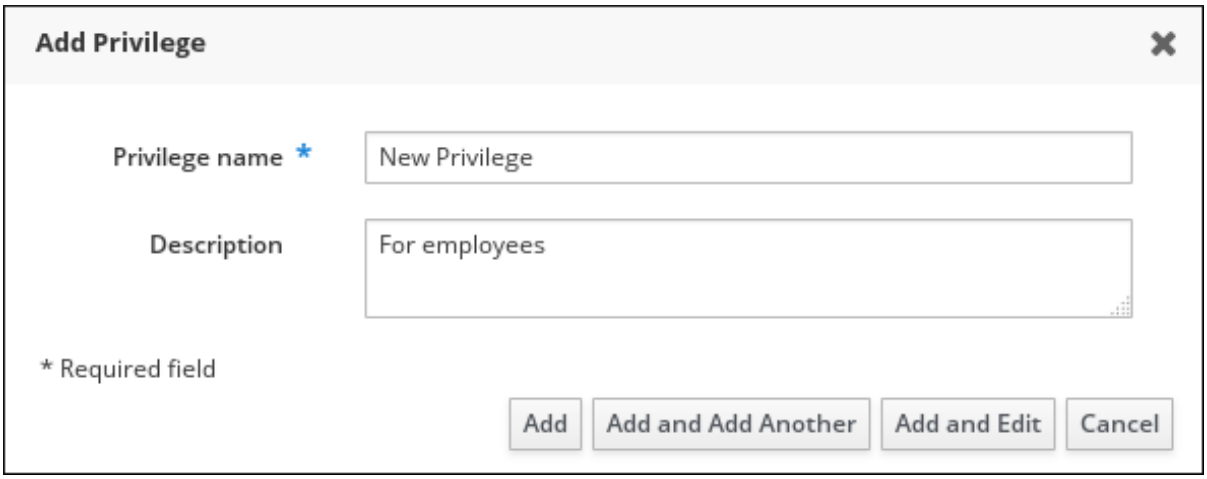


3. **单击特权列表顶部的 Add 链接。**

图 10.17. 添加新特权

4. **输入特权的名称和描述。**

图 10.18. 添加特权表格

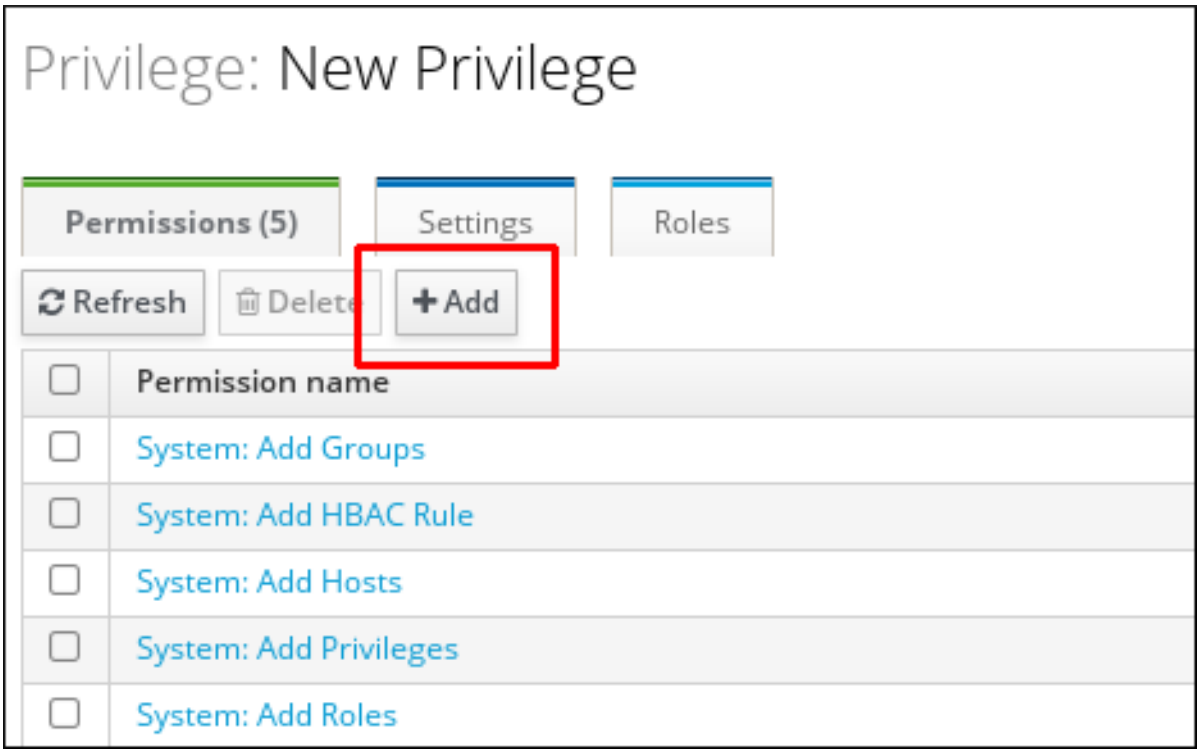


5. 单击 **Add and Edit** 按钮，以进入特权配置页面来添加权限。

6. 选择 **Permissions** 选项卡。

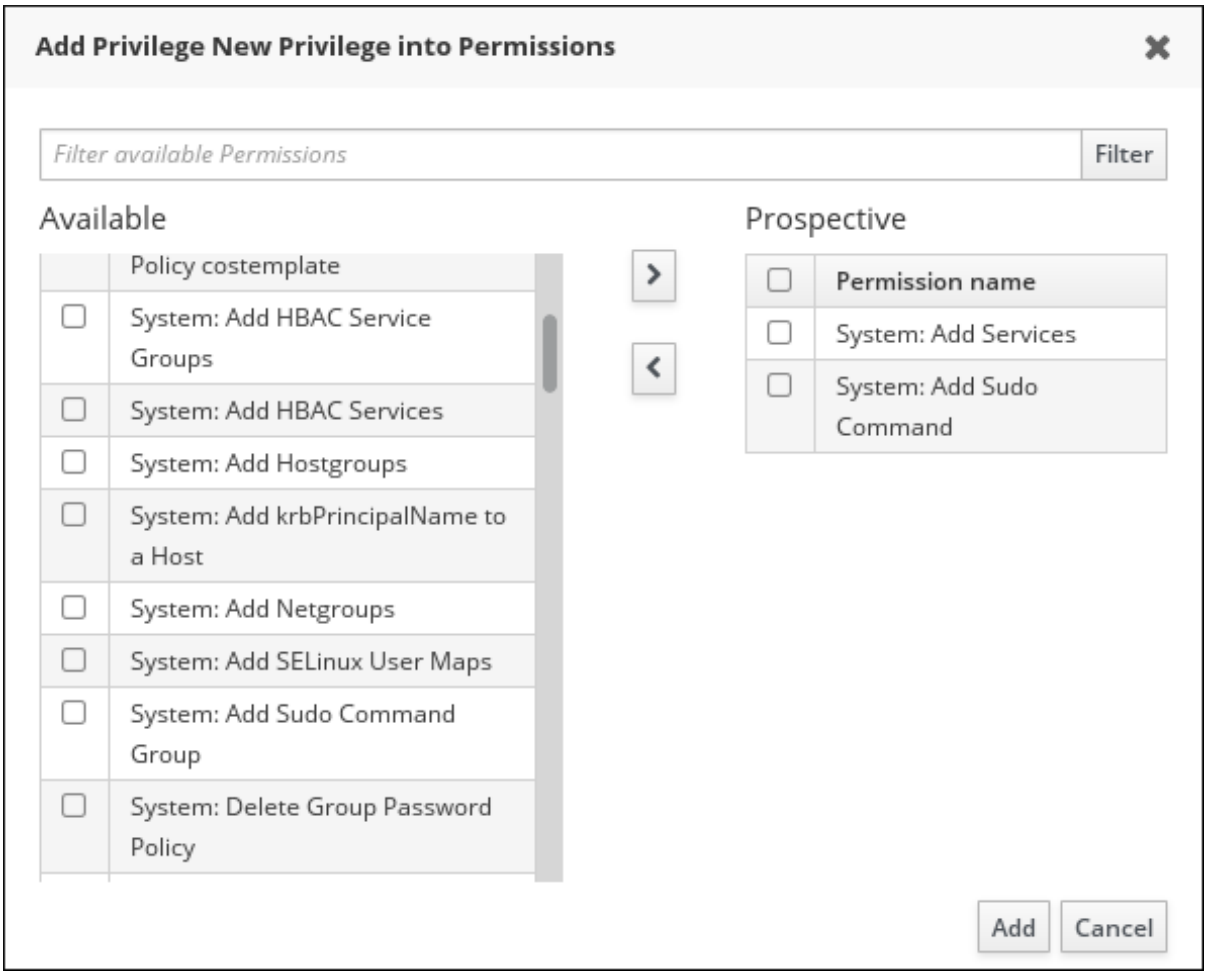
7. 单击权限列表顶部的 **Add**，以向特权添加权限。

图 10.19. 添加权限



8. 根据要添加的权限名称，点复选框，并使用 &gt；按钮将权限移到 **Prospective** 列中。

图 10.20. 选择权限



9. 单击 **Add** 按钮保存。

10.4.3.2. 从命令行创建新特权

使用 privilege-add 命令创建特权条目，然后使用 privilege- add-permission 命令将权限 添加到特权组中。

1. **创建特权条目。**

```
[jsmith@server ~]$ ipa privilege-add "managing filesystems" --desc="for filesystems"
```

2. **分配所需的权限。例如：**

```
[jsmith@server ~]$ ipa privilege-add-permission "managing filesystems" --
permissions="managing automount" --permissions="managing ftp services"
```

部分 IV. 管理：管理身份

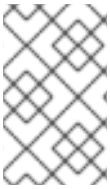
本节详细介绍了如何管理用户帐户、主机以及用户组和主机组。另外，它详细介绍了如何分配和查看唯一的 UID 和 GID 号以及用户和组模式的工作方式。本章介绍了管理服务并委派对主机和服务的访问权限。最后章节提供了如何为 身份管理 用户定义的访问控制、如何管理 Kerberos 标记和主体别名的说明，以及如何与 NIS 域和 Netgroups 集成。

第 11 章 管理用户帐户

本章涵盖了用户帐户的一般管理和配置。

11.1. 设置用户主目录

建议每个用户都配置了主目录。用户主目录的默认预期位置位于 /home/ 目录中。例如，IdM 需要具有 *user_login* 登录的用户在 /home/*user_login* 中设置主目录。



注意

您可以使用 ipa config-mod 命令更改用户主目录的默认预期位置。

IdM 不会自动为用户创建主目录。但是，您可以配置 PAM 主目录模块，以在用户登录时自动创建主目录。或者，您可以使用 NFS 共享和 automount 工具手动添加主目录。

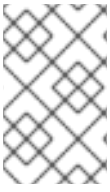
11.1.1. 使用 PAM 主目录模块自动挂载主目录

支持的 PAM 主目录模块

要将 PAM 主目录模块配置为在用户登录 IdM 域时自动为用户创建主目录，请使用以下 PAM 模块之一：

- pam_oddjob_mkhomedir
- pam_mkhomedir

IdM 首先尝试使用 pam_oddjob_mkhomedir。如果没有安装此模块，IdM 会尝试改为使用 pam_mkhomedir。



注意

不支持为 NFS 共享中的新用户自动创建主目录。

配置 PAM 主目录模块

启用 PAM 主目录模块具有本地效果。因此，您必须在需要的每个客户端和服务器中单独启用该模块。

要在服务器或客户端安装过程中配置模块，在安装机器时使用带有 ipa-server-install 或 ipa-client-install 工具的 --mkhomedir 选项。

要在已安装的服务器或客户端上配置模块，请使用 authconfig 工具。例如：

```
# authconfig --enablemkhomedir --update
```

有关使用 authconfig 创建主目录的更多信息，请参阅 系统级身份验证指南。

11.1.2. 手动挂载主目录

您可以使用 NFS 文件服务器提供一个 /home/ 目录，供 IdM 域中的所有机器使用，然后使用 automount 工具将目录挂载到 IdM 计算机上。

使用 NFS 时的潜在问题

使用 NFS 可能会对性能和安全性造成负面影响。例如，使用 NFS 可能会导致安全问题为 NFS 用户授予 root 访问权限、加载整个 /home/ 目录树的性能问题，或者为主目录使用远程服务器的网络性能问题。

为降低这些问题的影响，建议遵循以下准则：

- 使用 automount 仅挂载用户的主目录，并且仅在用户登录时挂载。不要使用它来加载整个 /home/ 树。
- 使用有限权限的远程用户创建主目录，并以此用户身份在 IdM 服务器上挂载共享。由于 IdM 服务器作为 httpd 进程运行，因此可以使用 sudo 或类似的程序授予 IdM 服务器的有限访问权限，以便在 NFS 服务器上创建主目录。

使用 NFS 和 自动挂载配置主目录

使用 NFS 共享和 自动挂载从独立位置手动将主目录添加到 IdM 服务器中 ：

1. 为用户目录映射创建一个新位置。

-

```
$ ipa automountlocation-add userdirs
Location: userdirs
```

2. 添加直接映射到新位置的 auto.direct 文件。auto.direct 文件是由 ipa-server-install 工具自动创建的 自动挂载映射。在以下示例中，挂载点为 /share ：

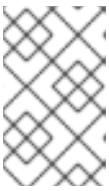
```
$ ipa automountkey-add userdirs auto.direct --key=/share --info="-ro,soft,
server.example.com:/home/share"

Key: /share
Mount information: -ro,soft, server.example.com:/home/share
```

有关在 IdM 中使用 自动挂载 的详情，请参考 第 34 章 *使用自动挂载*。

11.2. 用户生命周期

身份管理支持三种用户帐户状态： ***stage***、***active*** 和 ***preserve***。

- Stage（预发布） 用户不允许进行身份验证。这是初始状态。可能尚未设置活动用户所需的部分用户帐户属性。
- Active（活跃）用户被允许进行身份验证。所有必需的用户帐户属性都需要在这个阶段设置。
- 保留 的用户是之前 活跃的 用户。它们被视为不活动，无法向 IdM 进行身份验证。保留用户保留他们作为活跃用户的大多数帐户属性，但它们不属于任何用户组。



注意

处于 保留 状态的用户列表可以提供过去用户帐户的历史记录。

也可以从 IdM 数据库永久删除用户条目。删除用户条目会从 IdM 永久删除条目本身及其所有信息，包括组成员身份和密码。任何对用户的外部配置，如系统帐户和主目录，都不会被删除，但无法通过 IdM 来访问。

重要

删除的用户帐户无法恢复。当您删除用户帐户时，与帐户关联的所有信息都将永久丢失。

新管理员用户只能由另一个管理员（如默认的 admin 用户）创建新的管理员用户。如果您意外删除所有管理员帐户，目录管理器必须在 Directory 服务器中手动创建一个新管理员。

警告

不要删除 admin 用户。由于 admin 是 IdM 所需的预定义用户，因此此操作会导致某些命令出现问题。如果要定义和使用替代 admin 用户，在为至少一个不同的用户授予 admin 权限后，使用 ipa user-disable admin 禁用预定义的 admin 用户。

用户生命周期管理操作

若要管理用户调配，管理员可以将用户帐户从一个状态移到另一个状态。新用户帐户可以添加为 active 或 stage，但不能作为 保留。

IdM 支持以下操作来进行用户生命周期管理：

stage → active

当处于 stage 状态的帐户准备好被正确激活时，管理员会把它移到 active 状态。

Active → 保留

用户离开公司后，管理员会将帐户移到 preserved 状态。

Relded → active

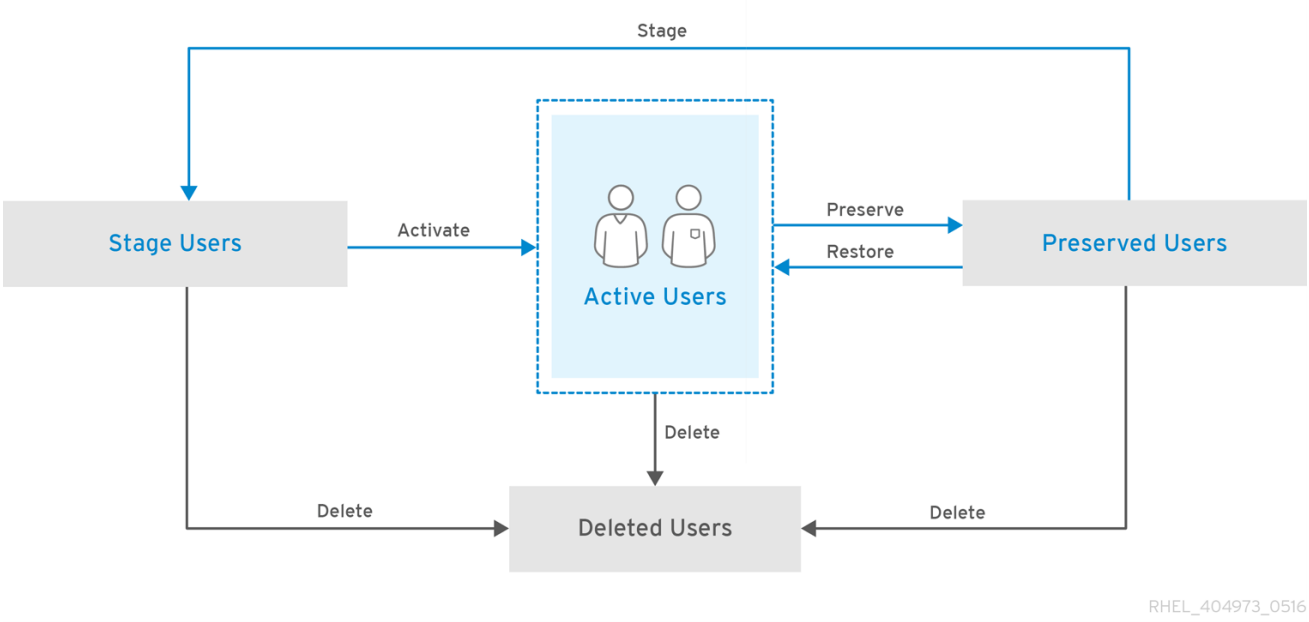
之前的用户再次加入公司。管理员通过将其从 preserved 状态移到 active 状态来恢复用户帐户。

Relded → stage

前一位用户计划再次加入公司。管理员将帐户从 preserved 状态移到 stage 状态，以准备帐户以便稍后重新激活。

您还可以从 IdM 永久删除活跃、阶段和保留的用户。请注意，您无法将 stage 用户移到 preserved 状态，您只能永久删除它们。

图 11.1. 用户生命周期操作



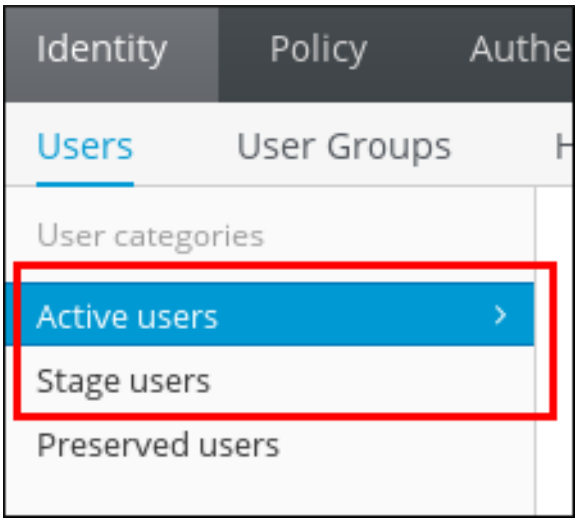
11.2.1. 添加阶段或活动用户

在 Web UI 中添加用户

1. 选择 **Identity** → **Users** 选项卡。

2. 根据您要以 active 或 stage 状态添加用户，选择 **Active users** 或 **Stage** 用户类别。

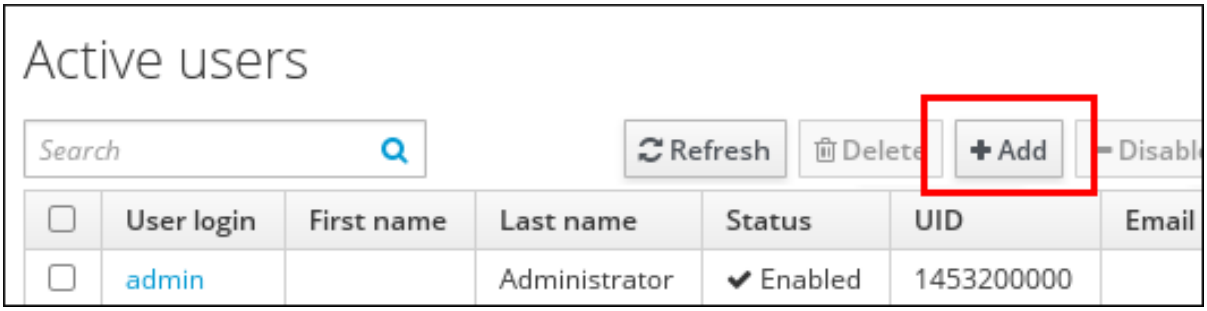
图 11.2. 选择用户类别



有关 活跃 或 stage 用户生命周期状态的更多信息，请参阅 第 11.2 节 “用户生命周期”。

3. 点 users 列表顶部的 Add。

图 11.3. 添加用户



4. 填写 Add User 表单。

请注意，如果您没有手动设置用户登录，IdM 会根据指定的名字和姓氏自动生成登录。

5. 点击 Add。

或者，单击 Add and Add Another 以开始添加其他 用户或添加和编辑 以开始编辑新用户条目。有关编辑用户条目的详情请参考 第 11.3 节 “编辑用户”。

从命令行添加用户

要添加一个处于 active 状态的新用户，请使用 ipa user-add 命令。要添加新用户处于 stage 状态，

请使用 **ipa stageuser-add** 命令。

注意

有关 活跃 或 stage 用户生命周期状态的更多信息，请参阅 第 11.2 节 “用户生命周期”。

当不带任何选项运行时，**ipa user-add** 和 **ipa stageuser-add** 会提示您输入最低所需的用户属性，并将默认值用于其他属性。或者，您也可以直接向命令添加指定各种属性的选项。

在交互式会话中，在您运行不带任何选项的 命令后，IdM 会根据提供的名字和姓氏提供自动生成的用户登录，并将其显示在方括号([ ])中。若要接受默认登录，请按 Enter 键进行确认。要指定自定义登录，请不要确认默认帐户，而是指定自定义登录。

```
$ ipa user-add
First name: first_name
Last name: last_name
User login [default_login]: custom_login
```

在 **ipa user-add** 和 **ipa stageuser-add** 中添加选项可让您为许多用户属性定义自定义值。这意味着您可以指定比在互动会话中的更多信息。例如，添加 stage 用户：

```
$ ipa stageuser-add stage_user_login --first=first_name --last=last_name --email=email_address
```

如需 **ipa user-add** 和 **ipa stageuser-add** 接受的选项的完整列表，请使用添加 **--help** 选项运行命令。

11.2.1.1. 用户名要求

IdM 支持可通过以下正则表达式描述的用户名：

```
'(?!^[0-9]+$)^[a-zA-Z0-9_.][a-zA-Z0-9_.-]*[a-zA-Z0-9_.$-]?$'
```

用户名只能包含字母、数字、_、-、.、$ 和 必须至少包含一个字母。

注意

支持以末尾的美元符号($)结尾的用户名，以启用 Samba 3.x 机器支持。

如果您添加了用户名包含大写字符的用户，IdM 会在保存名称时自动将其转换为小写。因此，IdM 始终要求用户在登录时输入其用户名全部小写。此外，不能添加用户名仅在字母校准（如用户和用户）上有所不同 的用户。

用户名的默认最大长度为 32 个字符。要更改它，请使用 ipa config-mod --maxusername 命令。例如，要将最大用户名长度增加到 64 个字符：

```
$ ipa config-mod --maxusername=64
  Maximum username length: 64
  ...
```

11.2.1.2. 定义自定义 UID 或 GID 号

如果您添加新的用户条目但没有指定自定义 UID 或 GID 号，IdM 会自动分配 ID 范围内下一个可用的 ID 号。这意味着用户的 ID 号始终是唯一的。有关 ID 范围的详情请参考 *第 14 章 唯一 UID 和 GID 编号分配*。

当您指定自定义 ID 号时，服务器不会验证自定义 ID 号是否唯一。因此，多个用户条目可能被分配了相同的 ID 号。红帽建议防止有多个 ID 号相同的条目。

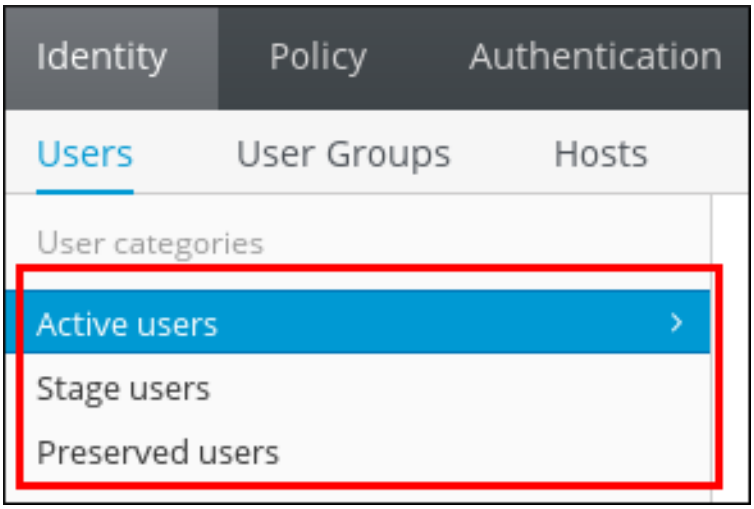
11.2.2. 列出用户和搜索用户

在 Web UI 中列出用户

1. 选择 **Identity** → **Users** 选项卡。

2. 选择 **Active users**、**Stage users** 或 **Preserved users** 类别。

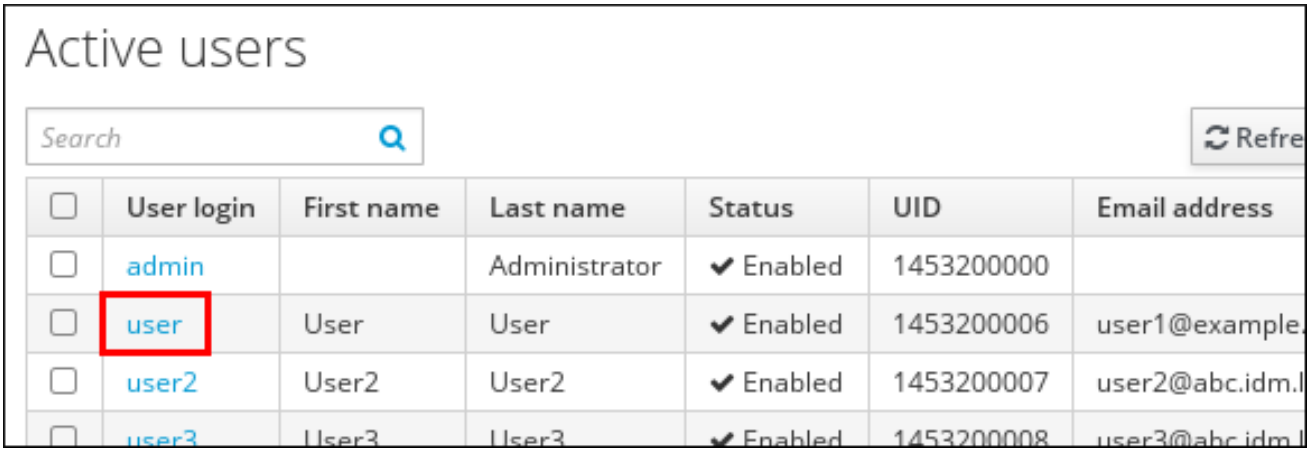
图 11.4. 列出用户



在 Web UI 中显示用户的信息

要显示用户的详细信息，请点击用户列表中的用户名称：

图 11.5. 显示用户信息



从命令行列出用户

要列出所有活动的用户，请运行 ipa user-find 命令。要列出所有 stage 用户，请使用 ipa stageuser-find 命令。要列出保留的用户，请运行 ipa user-find --preserved=true 命令。

例如：

```
$ ipa user-find
---------------
23 users matched
---------------
  User login: admin
  Last name: Administrator
  Home directory: /home/admin
  Login shell: /bin/bash
  UID: 1453200000
```

```
  GID: 1453200000
  Account disabled: False
  Password: True
  Kerberos keys available: True

  User login: user
...
```

通过在 **ipa user-find** 和 **ipa stageuser-find** 中添加选项和参数，您可以定义搜索结果并过滤搜索结果。例如，显示定义了特定标题的所有活跃用户：

```
$ ipa user-find --title=user_title
---------------
2 users matched
---------------
  User login: user
...
  Job Title: Title
...

  User login: user2
...
  Job Title: Title
...
```

同样，显示登录包含用户的所有 **stage** 用户 ：

```
$ ipa user-find user
---------------
3 users matched
---------------
User login: user
...

User login: user2
...

User login: user3
...
```

如需 **ipa user-find** 和 **ipa stageuser-find** 接受的选项的完整列表，请使用添加的 **--help** 选项运行命令。

从命令行显示用户的信息

要显示活跃或保留用户的信息，请使用 **ipa user-show** 命令：

```
$ ipa user-show user_login
```

```
User login: user_login
First name: first_name
Last name: last_name
...
```

要显示 stage 用户的信息，请使用 ipa stageuser-show 命令：

11.2.3. 激活、保留、删除和保留用户

本节论述了在不同用户生命周期状态之间移动用户帐户。有关 IdM 中生命周期状态的详情，请参考 第 11.2 节 “用户生命周期”。

在 Web UI 中管理用户生命周期

激活 stage 用户：

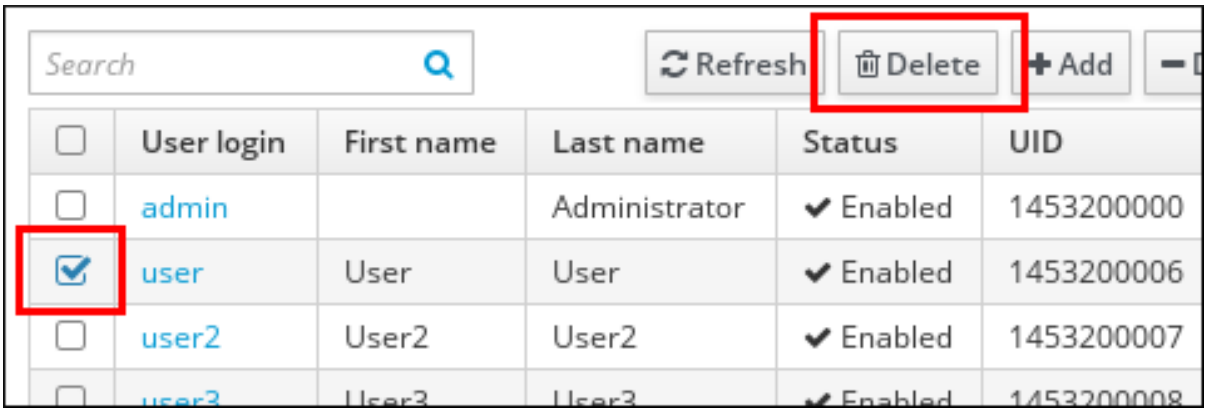
- 在 Stage users 列表中，选择要激活的用户，然后单击 Activate。

图 11.6. 激活用户

保留或删除用户：

1. 在 Active users 或 Stage 用户列表，选择用户。单击 Delete。

图 11.7. 删除用户



2. 如果您选择了活动用户，请选择 **delete** 或 **preserve**。如果选择了 **stage** 用户，则只能删除该用户。默认 **UI** 选项为 **delete**。

例如，要保留活跃的用户：

图 11.8. 在 Web UI 中选择 Delete Mode

若要确认，请单击 **Delete** 按钮。

恢复保留的用户：

- 在 **Preserved users** 列表中，选择要恢复的用户，然后单击 **Restore**。

图 11.9. 恢复用户

注意

恢复用户不会恢复之前帐户的所有属性。例如，用户的密码不会被恢复，必须再次定义。

请注意，在 Web UI 中，用户无法将用户从 preserved 状态移到 stage 状态。

从命令行管理用户生命周期

要通过从 stage 移到 active 来激活用户帐户，请使用 ipa stageuser-activate 命令。

```
$ ipa stageuser-activate user_login
-------------------------
Stage user user_login activated
-------------------------
...
```

要保留或删除用户帐户，请使用 ipa user-del 或 ipa stageuser-del 命令。

- **要从 IdM 数据库永久删除一个活动的用户，请在没有任何选项的情况下运行 ipa user-del。**

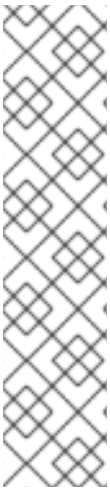
```
$ ipa user-del user_login
-------------------
Deleted user "user3"
-------------------
```

- **要保留活跃的用户帐户，请使用 --preserve 选项运行 ipa user-del。**

```
$ ipa user-del --preserve user_login
--------------------
Deleted user "user_login"
--------------------
```

- **要从 IdM 数据库永久删除 stage 用户，请运行 ipa stageuser-del。**

```
$ ipa stageuser-del user_login
--------------------------
Deleted stage user "user_login"
--------------------------
```



注意

删除多个用户时，请使用 --continue 选项强制命令继续，而不论出现什么错误。命令完成后，会将成功和失败的操作摘要输出到 stdout 标准输出流。

```
$ ipa user-del --continue user1 user2 user3
```

如果不使用 --continue，则命令会继续删除用户，直到它遇到错误，之后它停止并退出。

要通过将保留的用户帐户从 preserved 移到 active 来恢复保留的用户帐户，请使用 ipa user-undel 命令。

```
$ ipa user-undel user_login
------------------------------
Undeleted user account "user_login"
------------------------------
```

要通过将保留的用户帐户从 preserved 移到 stage 来恢复保留的用户帐户，请使用 ipa user-stage 命令。

```
$ ipa user-stage user_login
------------------------------
Staged user account "user_login"
------------------------------
```

注意

恢复用户帐户不会恢复之前帐户的所有属性。例如，用户的密码不会被恢复，必须再次定义。

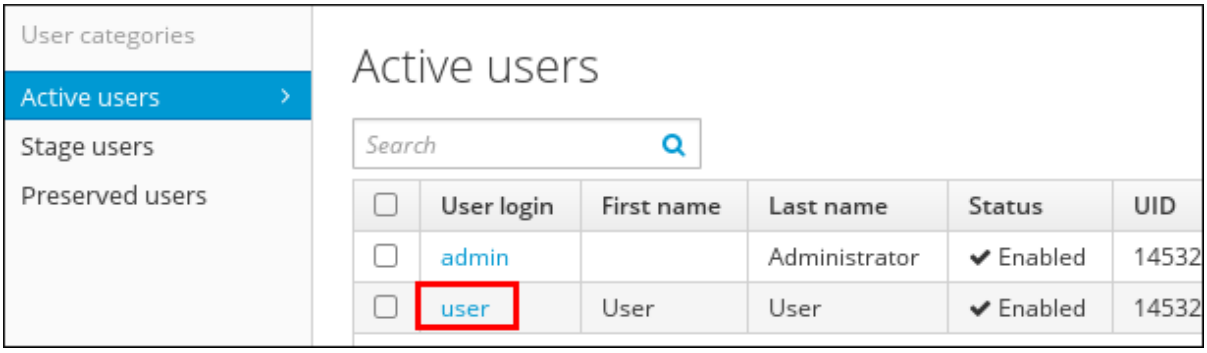
有关这些命令及其接受的选项的更多信息，请在添加 **--help** 选项的情况下运行它们。

11.3. 编辑用户

在 Web UI 中编辑用户

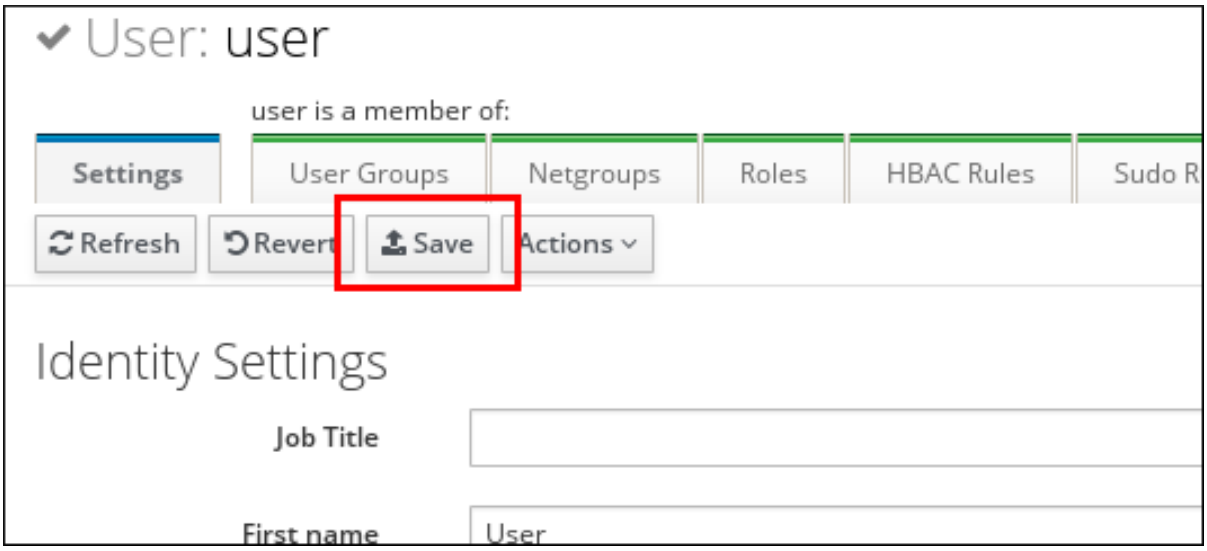
1. 选择 **Identity** → **Users** 选项卡。
2. 搜索 **Active users**,**Stage users**, 或 **Preserved users category**，以查找要编辑的用户。
3. 单击要编辑的用户的名称。

图 **11.10.** 选择要编辑的用户



4. 根据需要编辑 **user** 属性字段。
5. 点页面顶部的 **Save**。

图 11.11. 保存修改的用户属性



在 Web UI 中更新用户详情后，新值不会立即同步。可能需要大约 5 分钟后，新值才会反映在客户端系统中。

从命令行编辑用户

要修改处于 active 或 preserved 状态的用户，请使用 ipa user-mod 命令。要修改处于 stage 状态的用户，请使用 ipa stageuser-mod 命令。

ipa user-mod 和 ipa stageuser-mod 命令接受以下选项：

- 用户登录，用于标识要修改的用户帐户
- 指定新属性值的选项

有关可从命令行修改的用户条目属性的完整列表，请参阅 ipa user-mod 和 ipa stageuser-mod 接受的选项列表。要显示选项列表，请在添加 --help 选项的情况下运行命令。

只需在 ipa user-mod 或 ipa stageuser-mod 中添加属性选项会覆盖当前的属性值。例如，以下内容更改了用户的标题，或者添加了一个新的标题（如果用户尚未指定标题）：

```
$ ipa user-mod user_login --title=new_title
```

对于允许具有多个值的 LDAP 属性，IdM 也接受多个值。例如，用户可以在其用户帐户中保存两个电子邮件地址。若要添加不覆盖现有值的额外属性值，可使用 --addattr 选项和 选项来指定新属性值。例

如，要向已指定电子邮件地址的用户帐户添加新的电子邮件地址：

```
$ ipa user-mod user --addattr=mobile=new_mobile_number
--------------------
Modified user "user"
--------------------
  User login: user
...
  Mobile Telephone Number: mobile_number, new_mobile_number
...
```

要同时设置两个属性值，请使用 **--addattr** 选项两次：

```
$ ipa user-mod user --addattr=mobile=mobile_number_1 --addattr=mobile=mobile_number_2
```

ipa user-mod 命令也接受用于设置属性值的 **--setattr** 选项，以及用于删除属性值的 **--delattr** 选项。这些选项的使用方式与使用 **--addattr** 类似。详情请查看 **ipa user-mod --help** 命令的输出。

注意

要覆盖用户的当前电子邮件地址，请使用 **--email** 选项。但是，要添加额外的电子邮件地址，请使用 **mail** 选项和 **--addattr** 选项：

```
$ ipa user-mod user --email=email@example.com

$ ipa user-mod user --addattr=mail=another_email@example.com
```

11.4. 启用和禁用用户帐户

管理员可以禁用和启用活动用户帐户。禁用用户帐户可取消激活帐户。无法使用禁用的用户帐户进行身份验证。禁用帐户的用户无法登录 IdM，也不能使用 IdM 服务，如 Kerberos，或执行任何任务。

禁用的用户帐户仍然在 IdM 中存在，所有相关信息保持不变。与保留的用户帐户不同，禁用的用户帐户 保持活动状态。因此，它们会显示在 **ipa user-find** 命令的输出中。例如：

```
$ ipa user-find
...
  User login: user
  First name: User
  Last name: User
  Home directory: /home/user
  Login shell: /bin/sh
```

```
  UID: 1453200009
  GID: 1453200009
  Account disabled: True
  Password: False
  Kerberos keys available: False
...
```

可以重新启用任何禁用的用户帐户。

注意

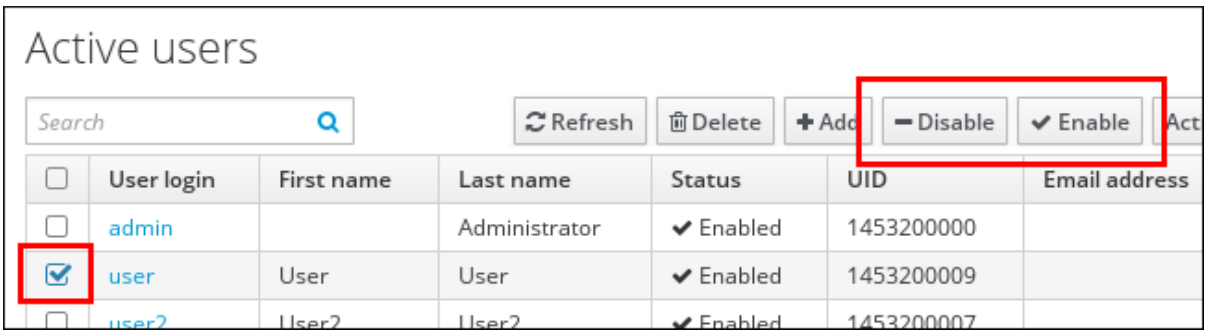
禁用用户帐户后，现有连接在用户的 Kerberos TGT 和其他票据到期之前保持有效。票据过期后，用户将无法续订。

在 Web UI 中启用和禁用用户帐户

1. 选择 **Identity → Users** 选项卡。

2. 从 **Active users** 列表中，选择所需的用户或用户，然后单击 **Disable** 或 **Enable**。

图 11.12. 禁用或启用用户帐户



从命令行禁用和启用用户帐户

要禁用用户帐户，请使用 ipa user-disable 命令。

```
$ ipa user-disable user_login
----------------------------
Disabled user account "user_login"
----------------------------
```

要启用用户帐户，请使用 ipa user-enable 命令。

```
$ ipa user-enable user_login
```

```
----------------------------
Enabled user account "user_login"
----------------------------
```

11.5. 允许非管理员用户管理用户条目

默认情况下，只有 admin 用户被允许管理用户生命周期，并禁用或启用用户帐户。要允许另一个非管理员用户执行此操作，创建一个新角色，在此角色中添加相关权限，并将非管理员用户分配到该角色。

默认情况下，IdM 包括以下与管理用户帐户相关的权限：

修改用户和重置密码

此特权包括修改各种用户属性的权限。

User Administrators

此特权包括添加活动用户、激活非活动用户、删除用户、修改用户属性和其他权限的权限。

阶段用户置备

此特权包括添加暂存用户的权限。

暂存用户管理员

此权限包括执行多个生命周期操作的权限，如添加暂存用户或在生命周期状态之间移动用户。但是，它不包括将用户移到 active 状态的权限。

有关定义角色、权限和权限的详情请参考 第 10.4 节 “定义基于角色的访问控制”。

允许不同的用户执行不同的用户管理操作

与管理用户帐户相关的不同特权可以添加到不同的用户。例如，您可以通过以下方式分隔员工帐户条目和激活的权限：

- 将一个用户配置为 *阶段用户管理员*，允许将将来的员工作为暂存用户添加到 IdM，但不允许激活它们。

- 将另一个用户配置为 *安全管理员*，允许在员工凭证在就业第一天得到验证后激活暂存用户。

要允许用户执行某些用户管理操作，请创建一个具有所需特权或特权的新角色，并将该用户分配到该角色。

例 11.1. 允许非管理员用户添加阶段用户

本例演示了如何创建仅允许添加新阶段用户但不允许执行任何其他阶段用户管理操作的用户。

1. 以 admin 用户身份登录，或允许管理基于角色的访问控制的另一个用户身份登录。

```
$ kinit admin
```

2. 创建新的自定义角色来管理添加阶段用户。

   a. 创建 系统置备 角色。

```
$ ipa role-add --desc "Responsible for provisioning stage users" "System
Provisioning"
---------------------------------
Added role "System Provisioning"
---------------------------------
Role name: System Provisioning
Description: Responsible for provisioning stage users
```

   b. 将 Stage User Provisioning 特权添加到该角色。此特权提供添加暂存用户的功能。

```
$ ipa role-add-privilege "System Provisioning" --privileges="Stage User Provisioning"
Role name: System Provisioning
Description: Responsible for provisioning stage users
Privileges: Stage User Provisioning
----------------------------
Number of privileges added 1
----------------------------
```

3. 为非管理员用户授予添加暂存用户的权限。

   a.

如果非 admin 用户尚不存在，请创建一个新用户。在本例中，该用户名为 stage_user_admin。

```
$ ipa user-add stage_user_admin --password
First name: first_name
Last name: last_name
Password:
Enter password again to verify:
...
```

b. 将 stage_user_admin 用户分配给 System Provisioning 角色。

```
$ ipa role-add-member "System Provisioning" --users=stage_user_admin
Role name: System Provisioning
Description: Responsible for provisioning stage users
Member users: stage_user_admin
Privileges: Stage User Provisioning
-------------------------
Number of members added 1
-------------------------
```

c. 为确保正确配置了 System Provisioning 角色，您可以使用 ipa role-show 命令显示角色设置。

```
$ ipa role-show "System Provisioning"
--------------
1 role matched
--------------
Role name: System provisioning
Description: Responsible for provisioning stage users
Member users: stage_user_admin
Privileges: Stage User Provisioning
----------------------------
Number of entries returned 1
----------------------------
```

4. 以 stage_user_admin 用户身份测试添加新的 stage 用户。

a. 以 stage_user_admin 身份登录。请注意，如果您在前面的步骤中以一个新用户身份创建了 stage_user_admin，IdM 会要求您更改 admin 设置的初始密码。

```
$ kinit stage_user_admin
Password for stage_user_admin@EXAMPLE.COM:
Password expired.  You must change it now.
```

```
Enter new password:
Enter it again:
```

b.

要确保您的 admin 的 Kerberos 票据已被 stage_user_admin 的 Kerberos 票据替代，您可以使用 klist 工具。

```
$ klist
Ticket cache: KEYRING:persistent:0:krb_ccache_xIICQDW
Default principal: stage_user_admin@EXAMPLE.COM

Valid starting       Expires              Service principal
02/25/2016 11:42:20  02/26/2016 11:42:20  krbtgt/EXAMPLE.COM
```

c.

添加新 stage 用户。

```
$ ipa stageuser-add stage_user
First name: first_name
Last name: last_name
ipa: ERROR: stage_user: stage user not found
```

注意

预期在添加暂存用户后 IdM 报告的错误。stage_user_admin 只允许添加 stage 用户，而不可以显示有关他们的信息。因此，IdM 不会显示新添加的 stage_user 设置摘要。

stage_user_admin 用户不允许显示 stage 用户的信息。因此，当以 stage_user _ admin 身份登录时，尝试显示有关新 stage_user 用户 的信息会失败：

```
$ ipa stageuser-show stage_user
ipa: ERROR: stage_user: stage user not found
```

要显示 stage_user 的信息，您可以 以管理员身份登录 ：

```
$ kinit admin
Password for admin@EXAMPLE.COM:
$ ipa stageuser-show stage_user
  User login: stage_user
  First name: Stage
  Last name: User
...
```

11.6. 将外部置备系统用于用户和组

身份管理支持配置您的环境，以便使用用于管理身份的外部解决方案在 IdM 中置备用户和组身份。这部分论述了这类配置的示例。这个示例包括：

- 第 11.6.1 节 “配置要由外部置备系统使用的用户帐户”
- 第 11.6.2 节 “配置 IdM 以自动激活暂存用户帐户”
- 第 11.6.3 节 “配置外部置备系统的 LDAP 提供程序来管理 IdM 标识符”

11.6.1. 配置要由外部置备系统使用的用户帐户

此流程演示了如何配置两个 IdM 用户帐户供外部置备系统使用。通过使用合适的密码策略将帐户添加到组中，您可以使外部调配系统来管理 IdM 中的用户调配。

1. 创建用户 provisionator，并具有添加 stage 用户的特权。该用户帐户将供外部调配系统用于添加新的暂存用户。

   a. 添加 provisionator 用户帐户：

   ```
   $ ipa user-add provisionator --first=provisioning --last=account --password
   ```

   b. 为 provisionator 用户授予所需的特权。

   创建一个自定义角色 System Provisioning，来管理添加 stage 用户：

   ```
   $ ipa role-add --desc "Responsible for provisioning stage users" "System Provisioning"
   ```

   将 Stage User Provisioning 特权添加到该角色。这个特权提供了添加 stage 用户的能力：

   ```
   $ ipa role-add-privilege "System Provisioning" --privileges="Stage User Provisioning"
   ```

将 provisionator 用户添加到角色中：

```
$ ipa role-add-member --users=provisionator "System Provisioning"
```

2. 创建用户 activator，其具有管理用户帐户的特权。用户帐户将用于自动激活由外部调配系统添加的暂存用户。

   a. 添加 activator 用户帐户：

   ```
   $ ipa user-add activator --first=activation --last=account --password
   ```

   b. 授予 activator 用户所需的特权。

   将用户添加到默认的 User Administrator 角色中：

   ```
   $ ipa role-add-member --users=activator "User Administrator"
   ```

3. 为服务和应用程序帐户创建用户组：

   ```
   $ ipa group-add service-accounts
   ```

4. 更新组的密码策略。以下策略可防止帐户的密码过期和锁住，但通过要求复杂的密码来弥补潜在的风险：

   ```
   $ ipa pwpolicy-add service-accounts --maxlife=10000 --minlife=0 --history=0 --minclasses=4
   --minlength=20 --priority=1 --maxfail=0 --failinterval=1 --lockouttime=0
   ```

5. 将置备和激活帐户添加到服务和应用程序帐户的组中：

   ```
   $ ipa group-add-member service-accounts --users={provisionator,activator}
   ```

6. 更改用户帐户的密码：

   ```
   $ kpasswd provisionator
   $ kpasswd activator
   ```

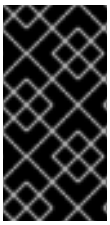
- 需要更改密码，因为新 IdM 用户的密码会立即过期。

其他资源：

- 有关添加新用户的详情请参考 第 11.2.1 节 “添加阶段或活动用户”。
- 有关授予用户管理其他用户帐户所需的权限的详情，请参考 第 11.5 节 “允许非管理员用户管理用户条目”。
- 有关管理 IdM 密码策略的详情，请参考 第 28 章 *定义密码策略*。

11.6.2. 配置 IdM 以自动激活暂存用户帐户

此流程演示了如何为激活 stage 用户创建脚本。系统在指定的时间间隔自动运行脚本。这样可确保新用户帐户被自动激活，并在创建后很快可用。



重要

该程序假定新用户帐户不需要验证，脚本才会将它们添加到 IdM。例如，如果用户已经由外部调配系统的所有者验证，则不需要验证。

这对于仅在一个 IdM 服务器上启用激活过程足够了。

1. 为激活帐户生成 keytab 文件：

```
# ipa-getkeytab -s example.com -p "activator" -k /etc/krb5.ipa-activation.keytab
```

如果您要在多个 IdM 服务器上启用激活过程，请仅在一个服务器上生成 keytab 文件。然后，将 keytab 文件复制到其他服务器上。

2. 创建一个包含以下内容的 /usr/local/sbin/ipa-activate-all 脚本来激活所有用户：

```
#!/bin/bash

kinit -k -i activator

ipa stageuser-find --all --raw | grep "  uid:" | cut -d ":" -f 2 | while read uid; do ipa stageuser-activate ${uid}; done
```

3. **编辑 ipa-activate-all 脚本的权限和所有权，使其可执行：**

```
# chmod 755 /usr/local/sbin/ipa-activate-all
# chown root:root /usr/local/sbin/ipa-activate-all
```

4. **创建包含以下内容的 systemd 单元文件 /etc/systemd/system/ipa-activate-all.service ：**

```
[Unit]
Description=Scan IdM every minute for any stage users that must be activated

[Service]
Environment=KRB5_CLIENT_KTNAME=/etc/krb5.ipa-activation.keytab
Environment=KRB5CCNAME=FILE:/tmp/krb5cc_ipa-activate-all
ExecStart=/usr/local/sbin/ipa-activate-all
```

5. **创建一个 systemd 计时器 /etc/systemd/system/ipa-activate-all.timer，其内容如下：**

```
[Unit]
Description=Scan IdM every minute for any stage users that must be activated

[Timer]
OnBootSec=15min
OnUnitActiveSec=1min

[Install]
WantedBy=multi-user.target
```

6. **启用 ipa-activate-all.timer:**

```
# systemctl enable ipa-activate-all.timer
```

其他资源：

- **有关 systemd 单元文件的更多信息，请参阅 系统管理员指南中的使用 systemd 单元文件管理服务章节。**

11.6.3. 配置外部置备系统的 LDAP 提供程序来管理 IdM 标识符

本节介绍各种用户和组管理操作的模板。使用这些模板，您可以配置置备系统的 LDAP 提供程序来管理 IdM 用户帐户。例如，您可以将系统配置为在员工离开公司后激活用户帐户。

使用 LDAP 管理用户帐户

您可以添加新用户条目、修改现有条目、在不同的生命周期状态之间移动用户，或通过编辑底层目录服务器数据库来删除用户。要编辑数据库，请使用 ldapmodify 工具。

以下 LDIF 格式的模板提供了有关使用 ldapmodify 修改的属性的信息。有关详细的示例步骤，请参阅 例 11.2 “使用 ldapmodify添加 Stage 用户” 和 例 11.3 “使用 ldapmodify保留用户”。

添加新的 stage 用户

使用 UID 和 GID 自动分配用户：

```
dn: uid=user_login,cn=staged users,cn=accounts,cn=provisioning,dc=example,dc=com
changetype: add
objectClass: top
objectClass: inetorgperson
uid: user_login
sn: surname
givenName: first_name
cn: full_name
```

静态分配使用 UID 和 GID 的用户：

```
dn: uid=user_login,cn=staged users,cn=accounts,cn=provisioning,dc=example,dc=com
changetype: add
objectClass: top
objectClass: person
objectClass: inetorgperson
objectClass: organizationalperson
objectClass: posixaccount
uid: user_login
uidNumber: UID_number
gidNumber: GID_number
sn: surname
givenName: first_name
cn: full_name
homeDirectory: /home/user_login
```

在添加 stage 用户时，您不需要指定任何 IdM 对象类。在激活用户后，IdM 自动添加这些类。

请注意，所创建的条目的可分辨名称(DN)必须以 uid=*user_login* 开头。

修改现有用户

在修改用户之前，请通过按用户登录搜索来获取用户的区分名称(DN)。在以下示例中，以下示例中的 *user_allowed_to_read* 用户是允许读取用户和组信息的用户，*密码* 是此用户的密码：

```
# ldapsearch -LLL -x -D "uid=user_allowed_to_read,cn=users,cn=accounts,dc=example, dc=com"
-w "password" -H ldap://server.example.com -b "cn=users, cn=accounts, dc=example, dc=com"
uid=user_login
```

修改用户的属性：

```
dn: distinguished_name
changetype: modify
replace: attribute_to_modify
attribute_to_modify: new_value
```

禁用用户：

```
dn: distinguished_name
changetype: modify
replace: nsAccountLock
nsAccountLock: TRUE
```

启用用户：

```
dn: distinguished_name
changetype: modify
replace: nsAccountLock
nsAccountLock: FALSE
```

保留用户：

```
dn: distinguished_name
changetype: modrdn
newrdn: uid=user_login
deleteoldrdn: 0
newsuperior: cn=deleted users,cn=accounts,cn=provisioning,dc=example
```

更新 *nssAccountLock* 属性不会对 stage 和preserved 用户造成影响。虽然更新操作成功完成，属性值仍然保持 nssAccountLock：对.

创建新组

要创建新组，请执行以下操作：

```
dn: cn=group_distinguished_name,cn=groups,cn=accounts,dc=example,dc=com
changetype: add
objectClass: top
objectClass: ipaobject
objectClass: ipausergroup
objectClass: groupofnames
objectClass: nestedgroup
objectClass: posixgroup
cn: group_name
gidNumber: GID_number
```

修改组

在修改组之前，请通过按组名称搜索来获取组的区分名称(DN)。

```
# ldapsearch -YGSSAPI  -H ldap://server.example.com -b
"cn=groups,cn=accounts,dc=example,dc=com" "cn=group_name"
```

删除现有组：

```
dn: group_distinguished_name
changetype: delete
```

将成员添加到组中：

```
dn: group_distinguished_name
changetype: modify
add: member
member: uid=user_login,cn=users,cn=accounts,dc=example,dc=com
```

从组中删除成员：

```
dn: distinguished_name
changetype: modify
delete: member
member: uid=user_login,cn=users,cn=accounts,dc=example,dc=com
```

–

不要向组中添加 stage 或 preserved 的用户。即使更新操作成功完成，也不会作为组的成员更新用户。只有活动的用户才能属于组。

例 11.2. 使用 ldapmodify添加 Stage 用户

使用标准 interorgperson 对象类添加新 stageuser 用户：

1. 使用 ldapmodify 添加用户。

```
# ldapmodify -Y GSSAPI
SASL/GSSAPI authentication started
SASL username: admin@EXAMPLE
SASL SSF: 56
SASL data security layer installed.
dn: uid=stageuser,cn=staged users,cn=accounts,cn=provisioning,dc=example
changetype: add
objectClass: top
objectClass: inetorgperson
cn: Stage
sn: User

adding new entry "uid=stageuser,cn=staged
users,cn=accounts,cn=provisioning,dc=example"
```

2. 考虑验证暂存条目的内容，以确保您的置备系统添加了所有必要的 POSIX 属性，并且暂存条目已就绪，可激活。使用 ipa stageuser-show --all --raw 命令显示新的 stage 用户的 LDAP 属性。请注意，由 nsaccountlock 属性显式禁用该用户：

```
$ ipa stageuser-show stageuser --all --raw
  dn: uid=stageuser,cn=staged users,cn=accounts,cn=provisioning,dc=example
  uid: stageuser
  sn: User
  cn: Stage
  has_password: FALSE
  has_keytab: FALSE
  nsaccountlock: TRUE
  objectClass: top
  objectClass: inetorgperson
  objectClass: organizationalPerson
  objectClass: person
```

例 11.3. 使用 ldapmodify保留用户

使用 LDAP modrdn 操作保留用户：

1. **使用 ldapmodify 工具修改用户条目。**

```
$ ldapmodify -Y GSSAPI
SASL/GSSAPI authentication started
SASL username: admin@EXAMPLE
SASL SSF: 56
SASL data security layer installed.
dn: uid=user1,cn=users,cn=accounts,dc=example
changetype: modrdn
newrdn: uid=user1
deleteoldrdn: 0
newsuperior: cn=deleted users,cn=accounts,cn=provisioning,dc=example

modifying rdn of entry "uid=user1,cn=users,cn=accounts,dc=example"
```

2. **（可选）通过列出所有保留的用户来验证用户是否已保留。**

```
$ ipa user-find --preserved=true
---------------
1 user matched
---------------
  User login: user1
  First name: first_name
  Last name: last_name
...
----------------------------
Number of entries returned 1
----------------------------
```

第 12 章 管理主机

DNS 和 Kerberos 都配置为初始客户端配置的一部分。这是必要的，因为这些服务是带 IdM 域中的机器的两个服务，并允许它识别它将连接的 IdM 服务器。在初始配置后，IdM 提供了管理这些服务的工具，以响应域服务的更改、IT 环境更改或在影响 Kerberos、证书和 DNS 服务的机器上更改。

本章论述了如何管理直接与客户端机器关联的身份服务：

- DNS 条目和设置
- 机器验证
- 主机名更改（影响域服务）

12.1. 关于主机、服务和机器身份和身份验证

注册过程的基本功能是为 IdM 目录中客户端计算机创建主机条目。*此主机条目用于建立域中其他主机甚至服务之间的关系（如 第 1 章 红帽身份管理简介所述）。这些关系是为域中的主机委派授权和控制的一部分。*

主机条目包含有关 IdM 中客户端的所有信息：

- *与主机关联的服务条目*
- *主机和服务主体*
- *访问控制规则*
- *机器信息，如物理位置和操作系统*

主机上运行的一些服务也可以属于 IdM 域。可以存储 Kerberos 主体或 SSL 证书（或两者）的任何服务都可以配置为 IdM 服务。向 IdM 域添加服务可让服务从域请求 SSL 证书或 keytab。（仅证书的公钥

存储在服务记录中。私钥是该服务的本地密钥。）

IdM 域在机器之间建立通用性，具有通用身份信息、通用策略和共享服务。属于域的任何计算机充当域的客户端，这意味着它使用域所提供的服务。IdM 域为机器提供三个主要服务：

- DNS
- Kerberos
- 证书管理

与用户一样，机器是由 IdM 管理的身份。客户端机器使用 DNS 来识别 IdM 服务器、服务和域成员。就像用户身份一样，它们存储在 IdM 服务器的 389 目录服务器实例中。与用户一样，计算机可以使用 Kerberos 或证书验证域。

从机器的角度来看，有几个任务可以访问这些域服务：

- 加入 DNS 域（机器注册）
- 管理 DNS 条目和区域
- 管理机器身份验证

IdM 中的身份验证包括机器和用户。IdM 服务器需要机器身份验证才能信任机器并接受来自该机器上安装的客户端软件的 IdM 连接。验证客户端后，IdM 服务器可以响应其请求。IdM 支持三种不同的机器身份验证方法：

- SSH 密钥。主机的 SSH 公钥已创建并上传到主机条目。从那里，系统安全服务守护进程 (SSSD)使用 IdM 作为身份提供程序，并可与 OpenSSH 和其他服务一起引用位于身份管理中的公钥。这在 第 12.5 节 “管理主机的公共 SSH 密钥” 中描述。
- 键表（或 keytab、 对称密钥集在某种程度上用户密码）和计算机证书.Kerberos 票据作为 Kerberos 服务的一部分生成，由服务器定义的策略。最初授予 Kerberos 票据、续订 Kerberos

凭证甚至销毁 Kerberos 会话也由 IdM 服务处理。Kerberos 管理包括在 第 29 章 管理 Kerberos 域 中。

- 计算机证书。在这种情况下，计算机使用 IdM 服务器的证书认证机构发布的 SSL 证书，然后存储在 IdM 的目录服务器中。证书然后发送到计算机，当它向服务器进行身份验证时会存在该证书。在客户端上，证书由名为 certmonger 的服务管理。

12.2. 关于主机条目配置属性

主机条目可以包含其系统配置之外的主机的信息，如其物理位置、MAC 地址、密钥和证书。

如果手动创建主机条目，则可以在创建此类信息时设置该信息；否则，在主机注册后，大多数此类信息都需要添加到主机条目中。

表 12.1. 主机配置属性

UI 字段	命令行选项	Description
Description	**--desc**=*description*	主机的描述。
地点	**--locality**=*locality*	主机的地理位置。
位置	**--location**=*location*	主机的物理位置，如其数据中心机架。
平台	**--platform**=*string*	主机硬件或架构.
操作系统	**--os**=*string*	主机的操作系统和版本。
MAC 地址	**--macaddress**=*address*	主机的 MAC 地址。这是一个多值属性。NIS 插件使用 MAC 地址为主机创建 NIS **ethers** 映射。
SSH 公钥	**--sshpubkey**=*string*	主机的完整 SSH 公钥。这是一个多值属性，因此可以设置多个键。
主体名称（不可编辑）	**--principalname**=*principal*	主机的 Kerberos 主体名称。除非在 **-p** 中显式设置了不同的主体，否则默认为客户端安装期间的主机名。这可以通过命令行工具进行更改，但不能在 UI 中更改。
设置一次性密码	**--password**=*string*	为主机设置可批量注册的密码。

UI 字段	命令行选项	Description
-	**--random**	生成要在批量注册时使用的随机密码。
-	**--certificate**=*string*	主机的证书 blob。
-	**--updatedns**	这会设置主机在其 IP 地址更改时是否可以动态更新其 DNS 条目。

12.3. 添加主机条目

12.3.1. 从 Web UI 添加主机条目

1. 打开 **Identity** 选项卡，然后选择 **Hosts** 子选项卡。

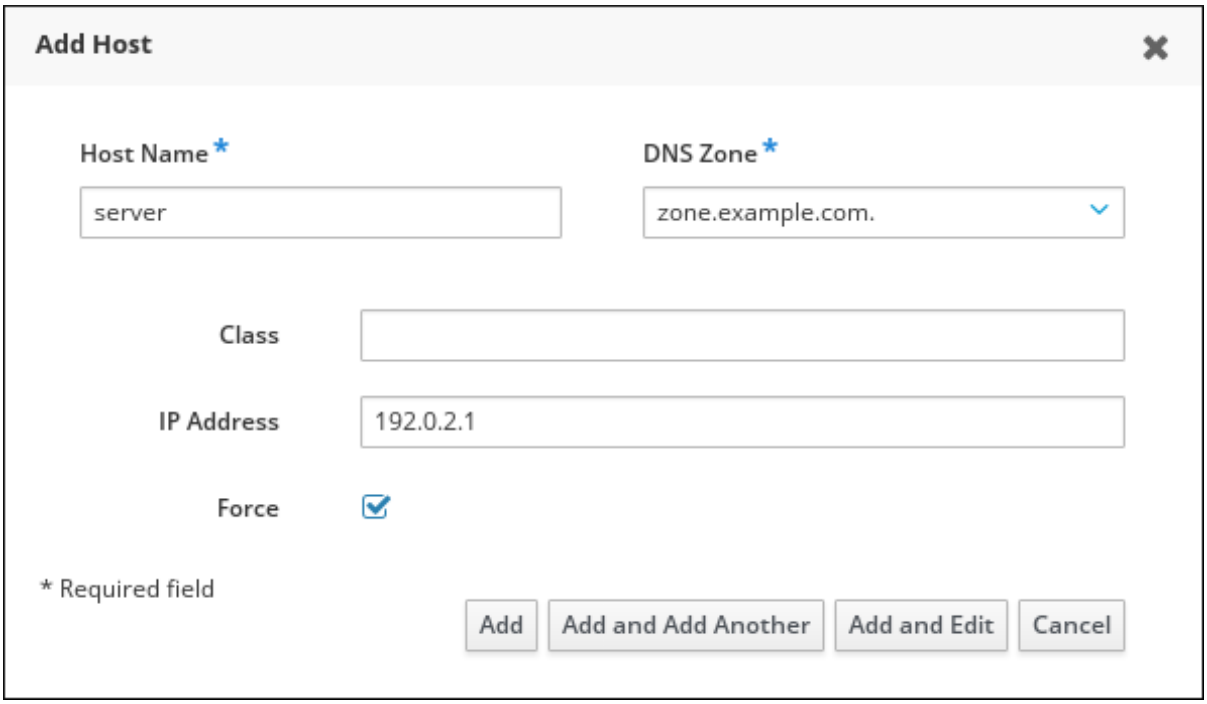
2. 单击主机列表顶部的 **Add**。

图 12.1. 添加主机条目

3. 填写机器名称，并从下拉列表中配置的区域中选择域。如果已经为主机分配了静态 IP 地址，则将它与主机条目一起包含，以便完全创建 DNS 条目。

另外，若要为某些用例向主机添加额外的值，请使用 **Class** 字段。在此属性中放置的语义用于本地解释。

图 12.2. 添加主机向导



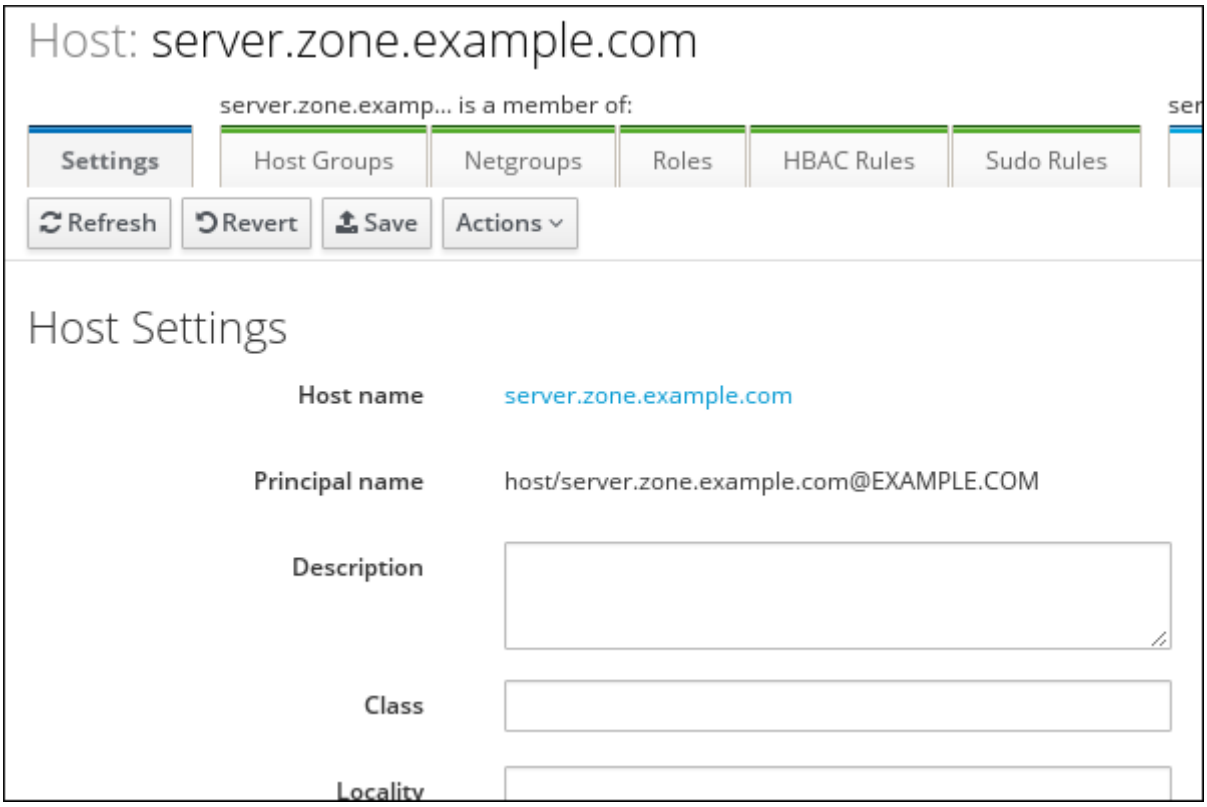
DNS 区域可以在 IdM 中创建，如 第 33.4.1 节 “添加和删除主 DNS 区域” 中所述。如果 IdM 服务器不管理 DNS 服务器，则可以在菜单区域中手动输入区，如常规文本字段。

注意

如果要跳过检查主机是否可以通过 DNS 解析，请选择 Force 复选框。

4. *点 Add and Edit 按钮直接进入展开的条目页面，并填写更多属性信息。有关主机硬件和物理位置的信息可以包含在主机条目中。*

图 12.3. 扩展的条目页面



12.3.2. 从命令行添加主机条目

主机条目使用 **host-add** 命令来创建。此命令将主机条目添加到 IdM 目录服务器中。**ipa host** 手册页中列出了带有 **host-add** 的选项的完整列表。在最基本的情况下，添加操作只需要客户端主机名将客户端添加到 **Kerberos** 域中，并在 **IdM LDAP** 服务器中创建条目：

```
$ ipa host-add client1.example.com
```

如果 **IdM** 服务器配置为管理 **DNS**，那么也可以使用 **--ip-address** 和 **--force** 选项将主机添加到 **DNS** 资源记录中。

例 12.1. 创建具有静态 **IP** 地址的主机条目

```
$ ipa host-add --force --ip-address=192.168.166.31 client1.example.com
```

通常，在配置客户端时，主机可能没有静态 **IP** 地址或 **IP** 地址可能不知道。例如，笔记本电脑可能预配置为身份管理客户端，但它们在配置时没有 **IP** 地址。使用 **DHCP** 的主机仍然可以使用 **--force** 配置 **DNS** 条目。这基本上在 **IdM DNS** 服务中创建占位符条目。当 **DNS** 服务动态更新其记录时，会检测到主机的当前 **IP** 地址并更新其 **DNS** 记录。

例 12.2. 创建具有 **DHCP** 的主机条目

```
$ ipa host-add --force client1.example.com
```

使用 host-del 命令删除主机记录。如果 IdM 域使用 DNS，-- updatedns 选项也会从 DNS 中删除主机任何类型的关联记录。

```
$ ipa host-del --updatedns client1.example.com
```

12.4. 禁用和重新启用主机条目

活动主机可由域中的其他服务、主机和用户访问。有些情况下，需要从活动中删除主机。但是，删除主机会删除该条目及所有关联的配置，并且会永久删除。

12.4.1. 禁用主机条目

禁用主机可防止域用户访问该主机，而不将其永久从域中删除。这可以通过使用 host-disable 命令来完成。

例如：

```
[jsmith@ipaserver ~]$ kinit admin
[jsmith@ipaserver ~]$ ipa host-disable server.example.com
```

重要

禁用主机条目不仅会禁用该主机。它还会禁用该主机上每个配置的服务。

12.4.2. 重新启用主机

这部分描述了如何重新启用禁用的 IdM 主机。

禁用主机会删除其活动 keytab，该选项卡将主机从 IdM 域中删除，而不影响其配置条目。

要重新启用主机，请使用 ipa-getkeytab 命令，添加：

- -s 选项来指定要从哪个 IdM 服务器请求 keytab
- -p 选项来指定主体名称
- k 选项来指定保存 keytab 的文件。

例如，要为 client.example.com 从 server.example.com 请求新的主机 keytab，并将 keytab 存储在 /etc/krb5.keytab 文件中：

```
$ ipa-getkeytab -s server.example.com -p host/client.example.com -k /etc/krb5.keytab -D "cn=directory manager" -w password
```

注意

您还可以使用管理员的凭据，指定 -D "uid=admin,cn=users,cn=accounts,dc=example,dc=com"。重要的是，凭据对应于允许为主机创建 keytab 的用户。

如果您在活动的 IdM 客户端或服务器上运行 ipa-getkeytab 命令，那么如果用户具有使用 kinit admin 获取的 TGT，您可以在没有 LDAP 凭据(-D 和 -w)的情况下运行它。若要在禁用的主机上直接运行命令，请提供 LDAP 凭据来向 IdM 服务器进行身份验证。

12.5. 管理主机的公共 SSH 密钥

OpenSSH 使用 公钥 对主机进行身份验证。一台计算机尝试访问另一台计算机并显示其密钥对。主机第一次进行身份验证时，目标计算机上的管理员必须手动批准请求。然后，机器将主机的公钥存储在 known_hosts 文件中。每当远程机器再次尝试访问目标机器时，目标机器只需检查其 known_hosts 文件，然后自动授予对批准的主机的访问权限。

这个系统有几个问题：

- known_hosts 文件将主机条目存储在主机 IP 地址、主机名和密钥的 triplet 中。如果 IP 地址发生更改（在虚拟环境和数据中心中很常见）或更新密钥，此文件可以快速过时。
- SSH 密钥必须手动分发给环境中的所有计算机。

- 管理员必须批准主机密钥才能将它们添加到配置中，但很难正确验证主机或密钥问题者，这可能会导致安全问题。

在 Red Hat Enterprise Linux 上，系统安全服务守护进程(SSSD)可以配置为缓存和检索主机 SSH 密钥，以便应用程序和服务必须只查找主机密钥的一个位置。由于 SSSD 可以使用身份管理作为其身份信息提供商之一，因此身份管理提供了密钥的通用和集中存储库。管理员无需担心分发、更新或验证主机 SSH 密钥。

12.5.1. 关于 SSH 密钥格式

当密钥上传到 IdM 条目时，密钥格式可以是 OpenSSH 样式的密钥或 原始 RFC 4253 风格的 Blob。任何 RFC 4253 风格的密钥都会自动转换为 OpenSSH 样式的密钥，然后再导入并保存到 IdM LDAP 服务器中。

IdM 服务器可以从上传的密钥 blob 中识别密钥类型，如 RSA 或 DSA 密钥。但是，在文件（如 ~/.ssh/known_hosts ）中，密钥条目由服务器的主机名和 IP 地址标识，然后输入密钥本身。例如：

```
host.example.com,1.2.3.4 ssh-rsa AAA...ZZZ==
```

这与用户公钥条目稍有不同，后者中含有 类型为 key== 注释 的元素：

```
"ssh-rsa ABCD1234...== ipaclient.example.com"
```

可以将密钥文件中的所有三个部分上传到，再查看主机条目。在这种情况下，需要重新排序 ~/.ssh/known_hosts 文件中的主机公钥条目，以匹配用户密钥的格式，输入 key== comment ：

```
ssh-rsa AAA...ZZZ== host.example.com,1.2.3.4
```

可以从公钥的内容中自动确定密钥类型，注释是可选的，从而更轻松地识别单个密钥。唯一必需的元素是公钥 Blob 本身。

12.5.2. 关于 ipa-client-install 和 OpenSSH

默认情况下，ipa-client-install 脚本在 IdM 客户端机器上配置 OpenSSH 服务器和客户端。它还将 SSSD 配置为执行主机和用户密钥缓存。基本上，只需配置客户端即可执行主机使用 SSSD、OpenSSH 和身份管理进行密钥缓存和检索所需的所有配置。

如果使用客户端安装（默认值）启用了 SSH 服务（默认值），则在 ssh 服务首次启动时会创建一个 RSA 密钥。

注意

当使用 ipa-client-install 将机器添加为 IdM 客户端时，会创建带有两个 SSH 密钥 RSA 和 DSS 的客户端。

还有额外的客户端配置选项 **--ssh-trust-dns**，可以使用 ipa-client-install 运行，并自动配置 OpenSSH 来信任存储密钥指纹的 IdM DNS 记录。

或者，也可以使用 **--no-sshd** 选项，在安装客户端时禁用 OpenSSH。这可防止安装脚本配置 OpenSSH 服务器。

另一个选项 **--no-dns-sshfp** 可防止主机使用自己的 DNS 条目创建 DNS SSHFP 记录。这可与 **--no-sshd** 选项或不带 **--no-sshd** 选项一起使用。

12.5.3. 通过 Web UI 上传主机 SSH 密钥

1. 主机的密钥可以从 **~/.ssh/known_hosts** 检索。例如：

```
server.example.com,1.2.3.4 ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAQEApvjBvSFSkTU0WQW4eOweeo0DZZ08F9Ud21xlLy6F
OhzwpXFGIyxvXZ52+siHBHbbqGL5+14N7UvElruyslIHx9LYUR/pPKSMXCGyboLy5aTNl5OQ5
EHwrhVnFDIKXkvp45945R7SKYCUtRumm0Iw6wq0XD4o+ILeVbV3wmcB1bXs36ZvC/M6riefn
9PcJmh6vNCvIsbMY6S+FhkWUTTiOXJjUDYRLIwM273FfWhzHK+SSQXeBp/zIn1gFvJhSZMR
i9HZpDoqxLbBB9QIdIw6U4MIjNmKsSI/ASpkFm2GuQ7ZK9KuMItY2AoCuIRmRAdF8iYNHBT
XNfFurGogXwRDjQ==
```

如有必要，生成主机密钥。在使用 OpenSSH 工具时，请确保使用空白密码短语，并将密钥保存到与用户的 **~/.ssh/** 目录不同的位置，因此它不会覆盖任何现有密钥。

```
[jsmith@server ~]$ ssh-keygen -t rsa -C "server.example.com,1.2.3.4"
Generating public/private rsa key pair.
Enter file in which to save the key (/home/jsmith/.ssh/id_rsa): /home/jsmith/.ssh/host_keys
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/jsmith/.ssh/host_keys.
Your public key has been saved in /home/jsmith/.ssh/host_keys.pub.
The key fingerprint is:
SHA256:GAUIDVVEgly7rs1ITWP6oguHz8BKvyZkpqCqVSsmi7c server.example.com
```

```
The key's randomart image is:
+--[ RSA 2048]----+
|             .. |
|              .+|
|         o   .* |
|        o . .. *|
|       S + .  o+|
|        E . .. .|
|       . = .  o |
|        o .  ..o|
|           .....|
+-----------------+
```

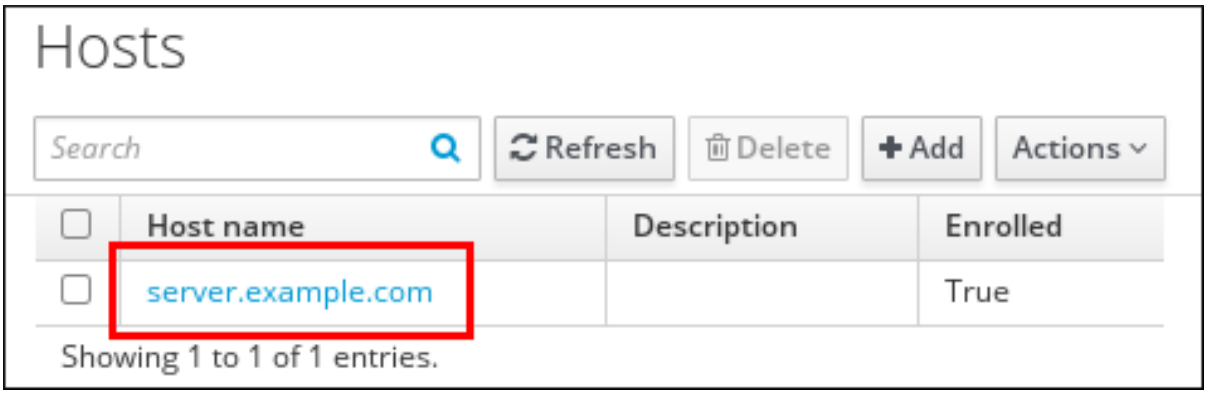
2. ***从 密钥文件复制公钥。full key 条目的格式为 主机名，IP 类型 key==。仅需要 key==，但可以存储整个条目。若要使用 条目中的所有元素，请重新排列条目，使其顺序为 key== [host name,IP]***

```
[jsmith@server ~]$ cat /home/jsmith/.ssh/host_keys.pub

ssh-rsa AAAAB3NzaC1yc2E...tJG1PK2Mq++wQ== server.example.com,1.2.3.4
```

3. ***打开 Identity 选项卡，然后选择 Hosts 子选项卡。***

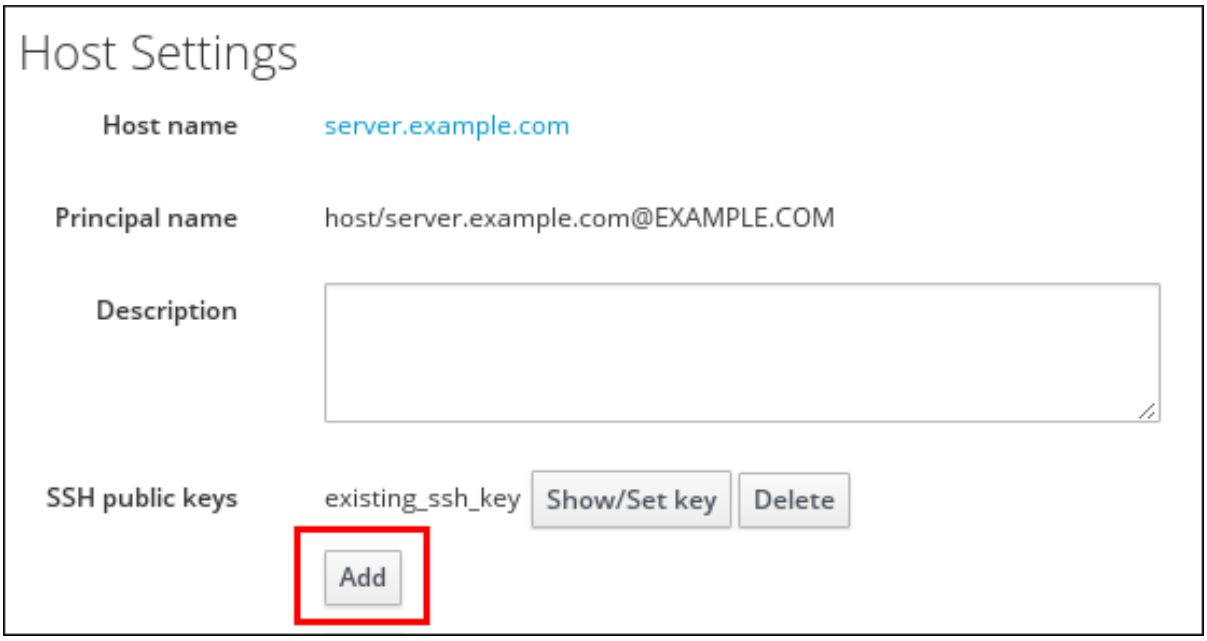
4. ***单击要编辑的主机的名称。***

图 12.4. 主机列表



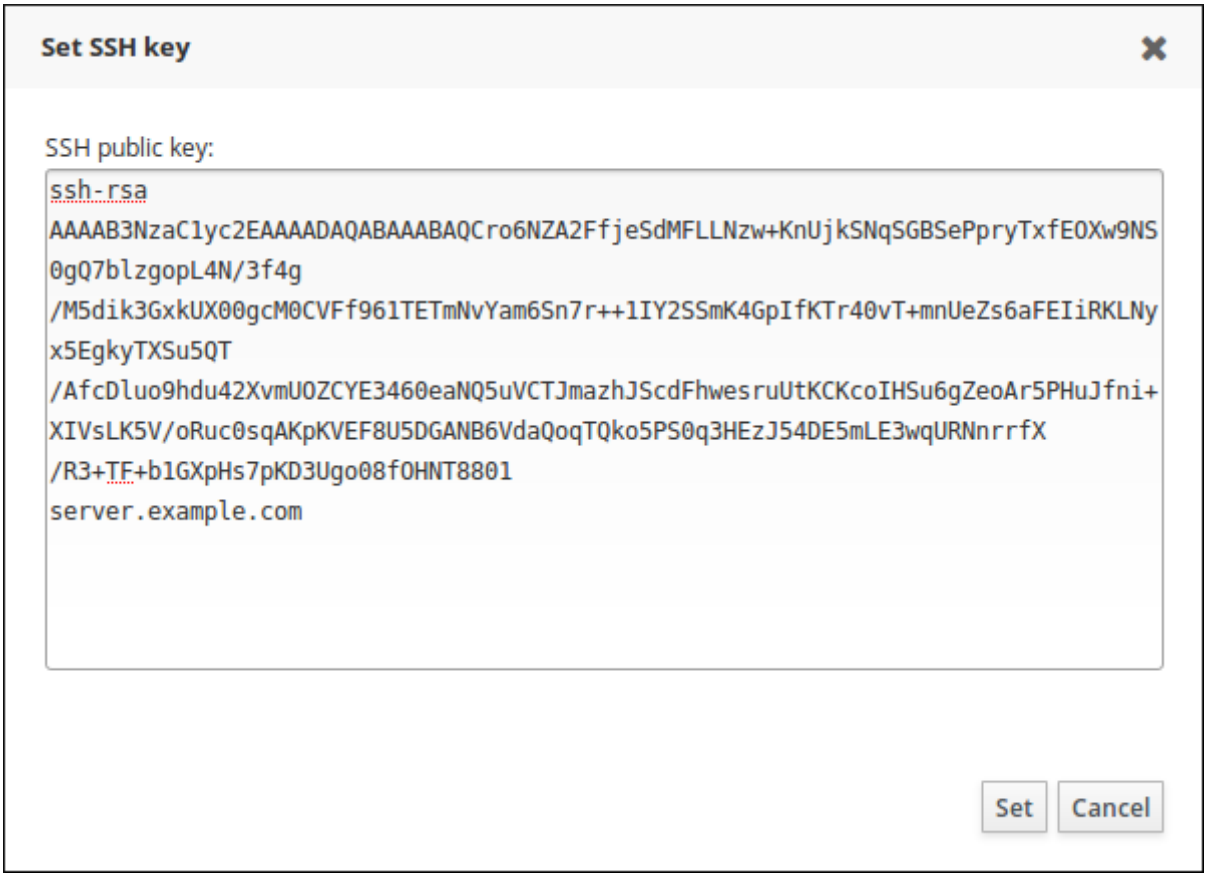
5. ***在 Settings 选项卡的 Host Settings 区域，单击 SSH 公钥 旁边的 Add。***

图 12.5. 添加 SSH 密钥



6. *粘贴主机的公钥，然后单击* **Set。**

图 12.6. 设置 SSH 密钥



SSH 公钥 区域现在显示新密钥。单击 **Show/Set key** *将打开提交的密钥。*

7. 要上传多个密钥，请单击公钥列表下的 Add 链接，并上传其他密钥。

8. 提交所有密钥后，单击主机页面顶部的 Save 以保存更改。

保存公钥后，条目将显示为密钥指纹、注释（如果包含公钥）和密钥类型[2].

上传主机密钥后，将 SSSD 配置为使用身份管理作为其身份域之一，并设置 OpenSSH 以使用 SSSD 工具来管理主机密钥，如 第 22.6 节 “配置 SSSD 为 OpenSSH 服务提供缓存” 所述。

12.5.4. 从命令行添加主机密钥

主机 SSH 密钥添加到 IdM 中的主机条目中，可以是使用 host-add 创建或稍后修改条目时。

注意

RSA 和 DSS 主机密钥由 ipa-client-install 命令创建，除非安装脚本中明确禁用了 SSH 服务。

1. 使用 --sshpubkey 选项运行 host-mod 命令，将 base64 编码的公钥上传到主机条目。

添加主机密钥也会更改主机的 DNS SSHFP 条目，因此也使用 --updatedns 选项来更新主机的 DNS 条目。

例如：

```
[jsmith@server ~]$ ipa host-mod --sshpubkey="ssh-rsa RjlzYQo==" --updatedns
host1.example.com
```

实际密钥通常也以等号(=)结尾，但时间较长。

要上传多个密钥，请输入多个 --sshpubkey 命令行参数：

```
--sshpubkey="RjlzYQo==" --sshpubkey="ZEt0TAo=="
```

注意

一个主机可以有多个公钥。

2. 上传主机密钥后，将 SSSD 配置为使用身份管理作为其身份域之一，并设置 OpenSSH 以使用 SSSD 工具来管理主机密钥，如 第 22.6 节 “配置 SSSD 为 OpenSSH 服务提供缓存” 所述。

12.5.5. 删除主机密钥

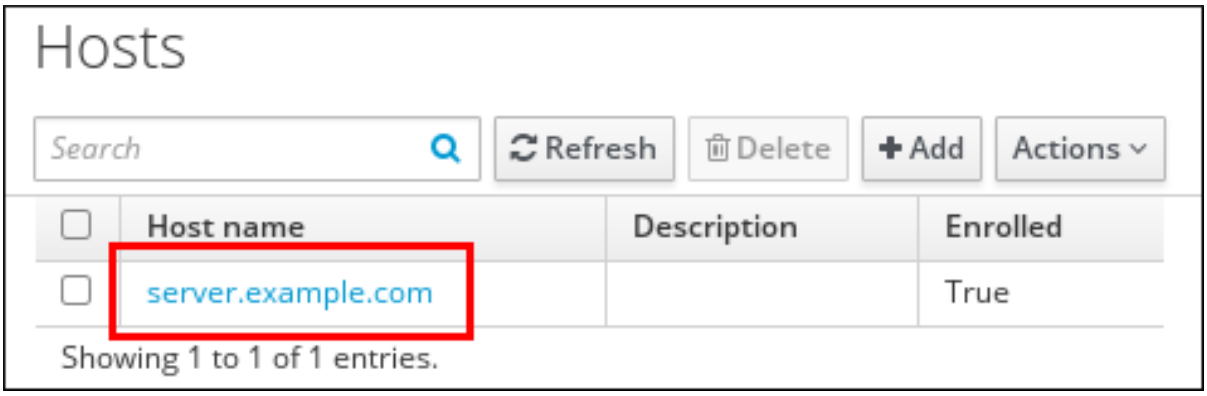
主机密钥在过期或不再有效后可将其删除。

要删除单个主机密钥，最简单的方法是通过 Web UI 删除该密钥：

1. 打开 Identity 选项卡，然后选择 Hosts 子选项卡。

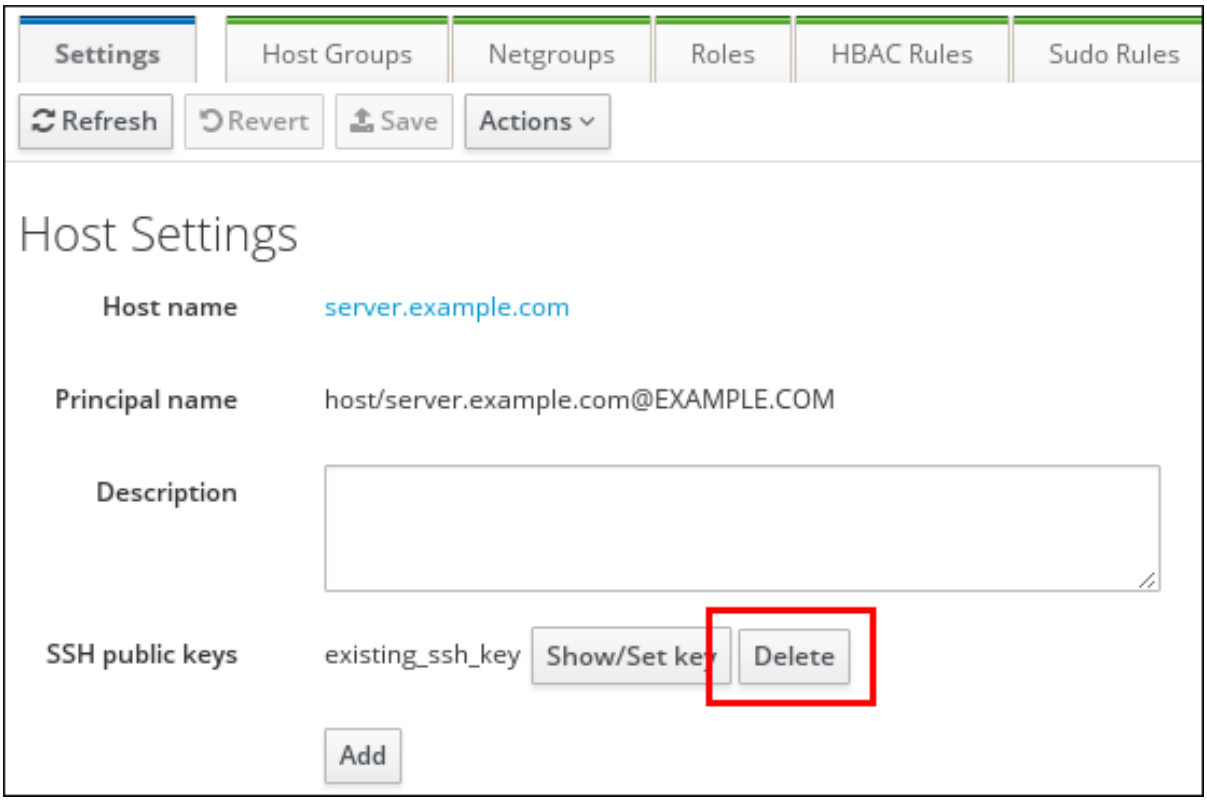
2. 单击要编辑的主机的名称。

图 12.7. 主机列表



3. 在 SSH 公钥 区域中，点 Delete by the key 的指纹将其移除。

图 12.8. 公钥删除



4. 单击主机页面顶部的 Save，以保存更改。

命令行工具可用于删除所有密钥。这可以通过运行 ipa host-mod，并将 --sshpubkey= 设置为空白值；这将删除主机 的所有 公钥。此外，使用 --updatedns 选项更新主机的 DNS 条目。例如：

```
[jsmith@server ~]$ kinit admin
[jsmith@server ~]$ ipa host-mod --sshpubkey= --updatedns host1.example.com
```

12.6. 为主机设置 ETHERS 信息

NIS 可以托管 ethers 表，它可以用来根据其平台、操作系统、DNS 域和 MAC 地址 - 保存在 IdM 中的主机条目中管理 DHCP 配置文件。

在身份管理中，每个系统都会在目录中使用对应的 ethers 条目创建，位于 ou=ethers 子树中。

```
cn=server,ou=ethers,dc=example,dc=com
```

此条目用于为 ethers 服务创建 NIS 映射，该服务可以由 IdM 中的 NIS 兼容性插件管理。

为 ethers 条目配置 NIS 映射：

1. 将 MAC 地址属性添加到主机条目。例如：

```
[jsmith@server ~]$ kinit admin
[jsmith@server ~]$ ipa host-mod --macaddress=12:34:56:78:9A:BC server.example.com
```

2. 打开 nsswitch.conf 文件。

3. 为 ethers 服务添加一行，并将其设置为使用 LDAP 进行查找。

```
ethers: ldap
```

4. 检查 ethers 信息是否可用于客户端。

```
[root@server ~]# getent ethers server.example.com
```

---

[2] 如果密钥类型不包含在上传密钥中，则从密钥本身自动确定密钥类型。

第 13 章 管理用户和组

13.1. IDM 中的用户和组如何工作

13.1.1. 用户和组是什么

用户组是一组具有常见特权、密码策略和其他特征的用户。

主机组是一组具有常见访问控制规则和其他特征的 IdM 主机。

例如，您可以定义公司部门、物理位置或访问控制需求的组。

13.1.2. 支持的组成员

IdM 中的用户组可以包括：

- IdM 用户
- 其他 IdM 用户组
- 外部用户，它们是 IdM 外部存在的用户

IdM 中的主机组可以包括：

- IdM 服务器和客户端
- 其他 IdM 主机组

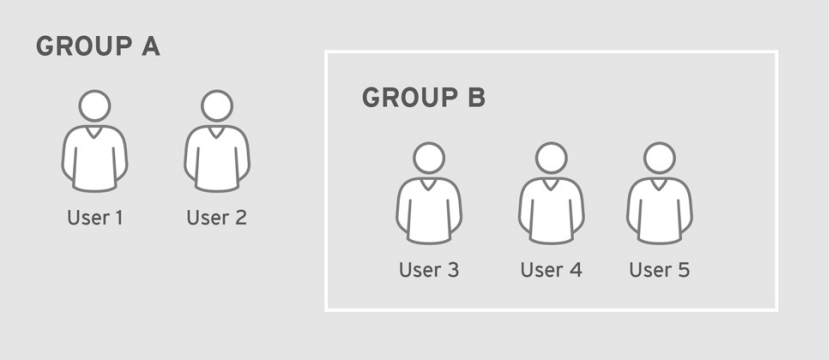
13.1.3. 直接和间接组成员

IdM 中的用户和组属性同时应用到直接成员和间接成员：当组 B 是组 A 的成员时，组 B 中的所有用户都将被视为组 A 的成员。

例如，在 图 13.1 “直接和间接组成员身份” 中：

- 用户 1 和用户 2 是组 A 的直接成员。
- 用户 3、用户 4 和用户 5 是组 A 的间接成员。

图 13.1. 直接和间接组成员身份



如果您为用户组 A 设置密码策略，该策略也会应用到用户组 B 中的所有用户。

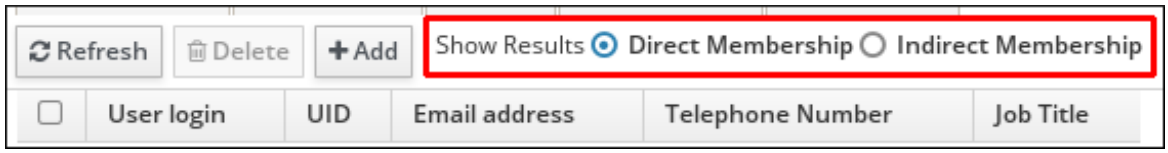
例 13.1. 查看直接组成员和间接组成员

1. 创建两个组：group_A 和 group_B。请参阅 第 13.2 节 “添加和删除用户或主机组”。
2. 添加：
   - 一个用户作为 group_A的成员
   - 另一个用户作为 group_B的成员
   - group_ b 作为 group_A的成员

请参阅 第 13.3 节 “添加和删除用户或主机组成员”。

3. 在 Web UI 中：选择 Identity → Groups。在左侧的侧边栏中列出的单独组类型，选择 User Groups，然后单击 group_A 的名称。在 Direct Membership 和 Indirect Membership 之间切换。

图 13.2. 间接和直接成员



4. 在命令行中：使用 ipa group-show 命令：

```
$ ipa group-show group_A
  ...
  Member users: user_1
  Member groups: group_B
  Indirect Member users: user_2
```

间接成员列表不包括来自可信活动目录域的外部用户。Active Directory 信任用户对象在 IdM 界面中不可见，因为它们不作为 IdM 中的 LDAP 对象存在。

13.1.4. IdM 中的用户组类型

POSIX 组（默认）

POSIX 组支持其成员的 POSIX 属性。请注意，与 Active Directory 交互的组无法使用 POSIX 属性。

非 POSIX 组

这种类型的组的所有组成员都必须属于 IdM 域。

外部组

外部组允许添加存在于 IdM 域外的身份存储中的组成员。外部存储可以是本地系统、Active Directory 域或目录服务。

非 POSIX 和外部组不支持 POSIX 属性。例如，这些组没有定义 GID。

例 13.2. 搜索不同的用户组群类型

1. 运行 **ipa group-find** 命令来显示所有用户组。
2. 运行 **ipa group-find --posix** 命令来显示所有 **POSIX** 组。
3. 运行 **ipa group-find --nonposix** 命令来显示所有非 **POSIX** 组。
4. 运行 **ipa group-find --external** 命令来显示所有外部组。

13.1.5. 默认创建的用户和组

表 13.1. 默认创建的用户和组

组名称	用户或主机	默认组成员
ipausers	用户组	所有 IdM 用户
admins	用户组	具有管理特权的用户，最初是默认的 **admin** 用户
editors	用户组	用户允许在 Web UI 中编辑其他 IdM 用户，而无需管理员用户的所有权限
trust admins	用户组	具有管理 Active Directory 信任的特权用户
ipaservers	主机组	所有 IdM 服务器主机

*将用户添加到用户组应用与组关联的特权和策略。例如，将用户添加到 **admins** 组会授予用户管理特权。*

警告

不要删除 admins 组。由于 admins 是 IdM 要求的预定义组，因此此操作会导致某些命令出现问题。

警告

将主机添加到 ipaservers 主机组时要小心。ipaservers 中的所有主机都能够将其自身提升到 IdM 服务器。

另外，当在 IdM 中创建新用户时，IdM 默认会创建用户私有组。

- 用户专用组的名称与为其创建的用户的名称相同。
- 用户是用户专用组的唯一成员。
- 专用组的 GID 与用户的 UID 相匹配。

例 13.3. 查看用户专用组

运行 **ipa group-find --private** 命令来显示所有用户私有组：

```
$ ipa group-find --private
----------------
2 groups matched
----------------
  Group name: user1
  Description: User private group for user1
  GID: 830400006

  Group name: user2
  Description: User private group for user2
```

```
  GID: 830400004
----------------------------
Number of entries returned 2
----------------------------
```

在某些情况下，最好避免创建用户专用组，例如 NIS 组或其他系统组已使用分配给用户专用组的 GID。请参阅 第 13.4 节 “禁用用户专用组”。

13.2. 添加和删除用户或主机组

要添加组，您可以使用：

- Web UI（请参阅 “Web UI：添加用户或主机组”一节）
- 命令行（请参见 “命令行：添加用户或主机组”一节）

IdM 在创建用户组时启用自定义 GID。如果您这样做，请小心以避免 ID 冲突。请参阅 第 14.6 节 “确保唯一 ID 值”。如果没有指定自定义的 GID，IdM 会自动从可用的 ID 范围内分配一个 GID。

要删除组，您可以使用：

- Web UI（请参阅 “Web UI：删除用户或主机组”一节）
- 命令行（请参见 “命令行：删除用户或主机组”一节）

请注意，删除组不会从 IdM 删除组成员。

Web UI：添加用户或主机组

1. 单击 Identity → Groups，然后在左侧栏中选择 User Groups 或 Host Groups。
2. 单击 Add 开始添加组。

3. 填写有关组的信息。

   有关用户组群类型的详情请参考 第 13.1.4 节 “IdM 中的用户组类型”。

4. 单击 Add 确认。

命令行：添加用户或主机组

1. 以管理员身份登录：

```
$ kinit admin
```

2. 要添加用户组，请使用 ipa group-add 命令。要添加主机组，请使用 ipa hostgroup-add 命令。

```
$ ipa group-add group_name
-----------------------
Added group "group_name"
-----------------------
```

默认情况下，ipa group-add 添加 POSIX 用户组。要指定不同的组类型，请在 ipa group-add 中添加选项：

- --nonposix 用来创建非 POSIX 组
- --external 用来创建外部组

有关组类型的详情请参考 第 13.1.4 节 “IdM 中的用户组类型”。

Web UI：删除用户或主机组

1. 点 Identity → Groups，再选择左侧栏中的 User Groups 或 Host Groups。

2. 选择要删除的组，然后单击 Delete。

命令行：删除用户或主机组

1. 以管理员身份登录：

```
$ kinit admin
```

2. 要删除用户组，请使用 **ipa group-del group_name** 命令。要删除主机组，请使用 **ipa hostgroup-del group_name** 命令。

```
$ ipa group-del group_name
--------------------------
Deleted group "group_name"
--------------------------
```

13.3. 添加和删除用户或主机组成员

要将成员添加到用户组中，您可以使用：

- IdM Web UI（请参阅 “Web UI：将成员添加到用户或主机组”一节）
- 命令行（请参见 “命令行：将成员添加到用户组中”一节）

重要

当添加另一个用户组作为成员时，请不要创建递归组。例如，如果组 A 是组 B 的成员，则不要将组 B 添加为组 A 的成员。递归组可能会导致无法预料的行为。

要从用户组群中删除成员，您可以使用：

- IdM Web UI（请参阅 “Web UI：从用户组中删除成员”一节）
- 命令行（请参见 “命令行：从用户组中删除成员”一节）

注意

将成员添加到用户或主机组后，更新可能需要一些时间才能分散到身份管理环境中的所有客户端。这是因为，当任何给定主机解析用户、组或网络组时，系统安全服务守护进程 (SSSD)首先查看其缓存，并且仅对缺失或过期的记录执行服务器查找。

要查看立即应用到主机组的更改，请使用 cache purge 实用程序 sss_cache 更新主机上的 SSSD 缓存。使用 sss_cache 为主机组使 SSSD 缓存中的当前记录无效，强制 SSSD 缓存从身份提供程序检索更新的记录，以便快速进行更改。

在 SSSD 缓存中清除主机组条目：

```
# sss_cache -n host_group_name
```

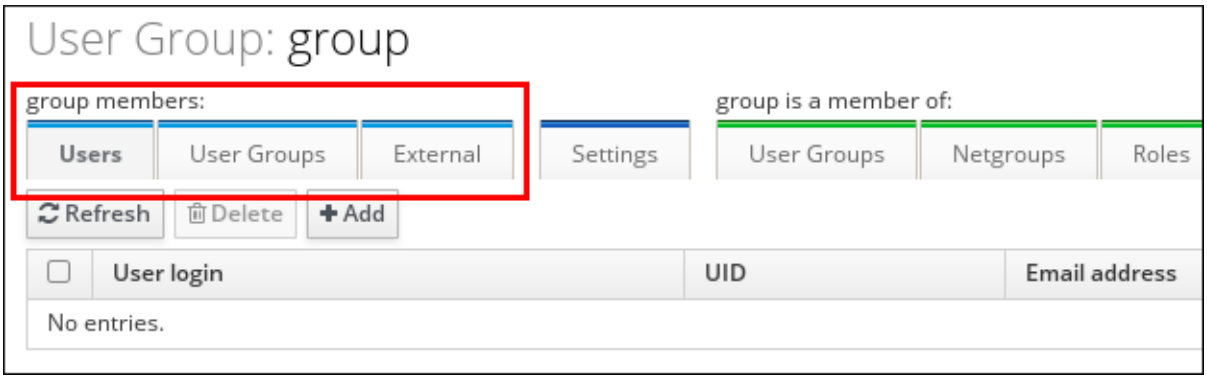
Web UI：将成员添加到用户或主机组

1. 点 Identity → Groups，再选择左侧栏中的 User Groups 或 Host Groups。

2. 单击组的名称。

3. 选择您要添加的组成员类型。例如，用户、用户组 或 External （用户组）。

图 13.3. 添加用户组群成员



4. 点击 Add。

5. 选择您要添加的成员，然后单击 **Add** 确认。

命令行：将成员添加到用户组中

1. 可选。使用 **ipa group-find** 或 **ipa hostgroup-find** 命令查找组。

2. 要将成员添加到用户组，请使用 **ipa group-add-member** 命令。要将成员添加到主机组，请使用 **ipa hostgroup-add-member** 命令。

   添加用户组群成员时，使用以下选项指定成员：

   - **--users** 添加 IdM 用户
   - **--external** 添加一个存在于 IdM 域外的用户，格式为 **DOMAIN\user_name** 或 **user_name@domain**
   - **--groups** 添加 IdM 用户组

   添加主机组 **member** 时，使用以下选项指定成员：

   - **--hosts** 添加 IdM 主机
   - **--groups** 添加 IdM 主机组

例 13.4. 将成员添加到用户组中的命令示例

将 **user1**、**user2** 和 **group1** 添加到名为 **group_name** 的组中：

```
$ ipa group-add-member group_name --users=user1 --users=user2 --groups=group1
```

要将 **ad_user** 从名为 **ad_domain** 的域添加到名为 **group_name** 的组中，您可以选择如何指定外部用户。例如：

```
$ ipa group-add-member group_name --external='AD_DOMAIN\ad_user'
$ ipa group-add-member group_name --external='ad_user@AD_DOMAIN'
$ ipa group-add-member group_name --
external='ad_user@AD_DOMAIN.EXAMPLE.COM'
```

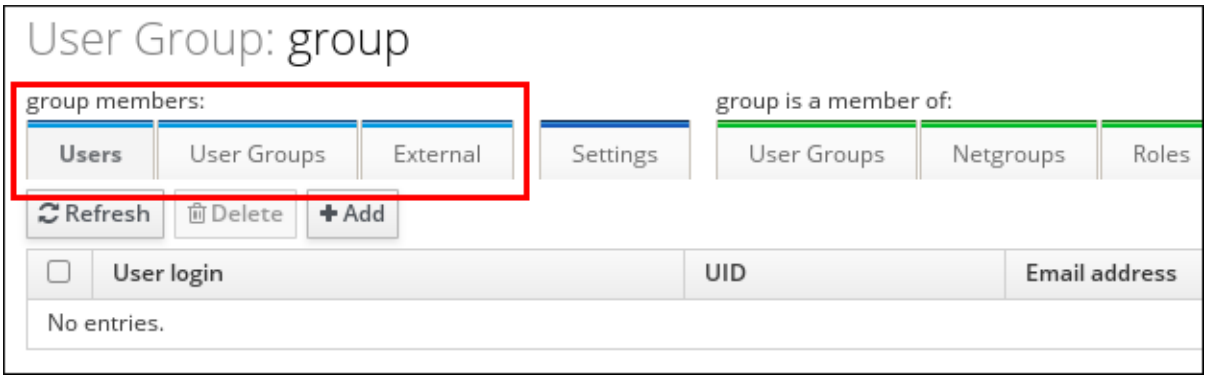
Web UI：从用户组中删除成员

1. 点 Identity → Groups，再选择左侧栏中的 User Groups 或 Host Groups。

2. 单击组的名称。

3. 选择您要删除的组成员类型。例如，用户、用户组 或 External （用户组）。

图 13.4. 删除用户组群成员



4. 选中所需成员旁边的复选框。

5. 单击 Delete。

命令行：从用户组中删除成员

1. 可选。使用 ipa group-show 或 ipa hostgroup-show 命令确认组是否包含您要删除的成员。

2. 要删除用户组成员，请使用 ipa group-remove-member 命令。要删除主机组成员，请使用 ipa hostgroup-remove-member 命令。

删除用户组群成员时，使用以下选项指定成员：

- --users 删除 IdM 用户
- --external 删除存在于 IdM 域外的用户，格式为 DOMAIN\user_name 或 user_name@domain
- --groups 删除 IdM 用户组

删除主机组 member 时，使用以下选项指定成员：

- --hosts 删除 IdM 主机
- --groups 删除 IdM 主机组

例如，要从名为 group_name 的组中删除 user1、user2 和 group1：

```
$ ipa group-remove-member group_name --users=user1 --users=user2 --groups=group1
```

13.4. 禁用用户专用组

要确定 IdM 没有为新用户创建默认用户私有组，请选择以下之一：

- 第 13.4.1 节 “创建没有用户专用组的用户”
- 第 13.4.2 节 “针对所有用户全局禁用用户专用组”

即使禁用了创建默认用户私有组，在添加新用户时 IdM 仍需要 GID。要确保添加用户成功，请参阅第 13.4.3 节 “添加用户专用组禁用的用户”。

注意

如果您要禁用因 GID 冲突而创建默认用户私有组，请考虑更改默认 UID 和 GID 分配范围。请参阅 第 14 章 唯一 UID 和 GID 编号分配。

13.4.1. 创建没有用户专用组的用户

在 ipa user-add 命令中添加 --noprivate 选项。请注意，要使命令成功，您必须指定自定义专用组。请参阅 第 13.4.3 节 “添加用户专用组禁用的用户”。

13.4.2. 针对所有用户全局禁用用户专用组

1. 以管理员身份登录：

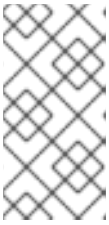
```
$ kinit admin
```

2. IdM 使用 Directory Server Managed Entries 插件来管理用户私有组。列出插件的实例：

```
$ ipa-managed-entries --list
```

3. 为确保 IdM 不会创建用户私有组，请禁用负责管理用户私有组的插件实例：

```
$ ipa-managed-entries -e "UPG Definition" disable
Disabling Plugin
```



注意

要在稍后重新启用 UPG Definition 实例，请使用 ipa-managed-entries -e "UPG Definition" enable 命令。

4. 重新启动目录服务器来加载新配置。

```
# systemctl restart dirsrv.target
```

13.4.3. 添加用户专用组禁用的用户

要确保在创建默认用户私有组时添加新用户成功，请选择以下之一：

- 添加新用户时指定自定义的 GID。GID 不必对应于已经存在的用户组。

  例如，当从命令行添加用户时，请在 ipa user-add 命令中添加 --gid 选项。

- 使用自动成员规则将用户添加到具有 GID 的现有组中。请参阅 第 13.6 节 “为用户和主机定义自动组成员资格”。

13.5. 为用户和组设置搜索属性

当使用 ipa user-find 关键字和 ipa group-find 关键字 命令搜索指定 关键字 时，IdM 仅搜索某些属性。最值得注意的是：

- 在用户搜索中：名字、姓、用户名（登录 ID）、作业标题、组织单元、电话号码、UID、电子邮件地址.

- 在组搜索中：组名称、描述.

以下流程演示了如何配置 IdM 来搜索其他属性。请注意，IdM 始终搜索默认属性。例如，即使您从用户搜索属性列表中删除作业标题属性，IdM 仍将搜索用户标题。

先决条件

在添加新属性之前，请确保此属性的 LDAP 目录中存在对应的索引。大多数标准 LDAP 属性在 LDAP 中都有索引，但如果要添加自定义属性，则必须手动创建索引。请参阅 Red Hat Directory Server 10 管理指南中的 创建标准索引。

Web UI：设置搜索属性

1. 选择 IPA Server → Configuration。

2. 在 User Options 区域中，在 User search 字段中设置用户搜索属性。

3. 在 Group Options 区域中，在 Group 搜索字段中设置组搜索 属性。

4. 点页面顶部的 Save。

命令行：设置搜索属性

使用带有以下选项的 ipa config-mod 命令：

- --usersearch 为用户定义新的搜索属性列表
- --groupsearch 为组定义一个新的搜索属性列表

例如：

```
$ ipa config-mod --usersearch="uid,givenname,sn,telephonenumber,ou,title"
$ ipa config-mod --groupsearch="cn,description"
```

13.6. 为用户和主机定义自动组成员资格

13.6.1. IdM 中的自动组成员资格工作

13.6.1.1. 自动组成员资格是什么

通过使用自动组成员身份，您可以根据用户和组的属性自动分配用户和组。例如，您可以：

- 根据员工的经理、位置或任何其他属性，将员工的用户条目划分为组。
- 根据主机的类、位置或任何其他属性来划分主机。
- 将所有用户或全部主机添加到单个全局组。

13.6.1.2. 自动组成员的好处

手动管理组成员开销

利用自动组成员身份，管理员不再手动将用户和组分配用户和组。

提高了用户和主机管理的一致性

利用自动组成员身份，根据严格定义和自动评估的标准，用户和主机分配到组中。

更轻松地管理基于组的设置

为组定义各种设置，然后应用到各个组成员，如 sudo 规则、自动挂载 或访问控制。使用自动组成员身份时，用户和主机会自动添加到指定组中，从而更加轻松地管理基于组的设置。

13.6.1.3. Automember 规则

在配置自动组成员身份时，管理员定义 自动成员规则。自动成员规则应用到特定的用户或主机组。它包括 用户或主机必须满足的条件 才能包含或排除在组中：

包含的条件

当用户或主机条目符合包含条件时，它将包含在组中。

排他条件

当用户或主机条目符合独占条件时，它不会 包含在组中。

条件被指定为 Perl 兼容的正则表达式(PCRE)格式的正则表达式。有关 PCRE 的详情，请参考 pcresyntax(3) man page。

IdM 在包含条件之前评估排他条件。在发生冲突时，排他条件优先于包含条件。

13.6.2. 添加自动成员规则

使用以下方法添加自动成员规则：

- IdM Web UI，请查看 “Web UI：添加自动成员规则”一节

- 命令行，请查看 “命令行：添加自动成员规则”一节

添加自动成员规则后：

- 将来创建的所有条目都将成为指定组的成员。如果条目满足多个自动成员规则中指定的条件，则会将其添加到所有对应的组中。

- 现有条目 不会成为 指定组的成员。请参阅 第 13.6.3 节 “将自动成员规则应用到现有用户和主机” 了解更多信息。

Web UI：添加自动成员规则

1. 选择 Identity → Automember → User group rules 或 Host group rules。

2. 点击 Add。

3. 在 Automember rule 字段中，选择规则要应用的组。点 Add and Edit。

4. 定义一个或多个包含和独占条件。详情请查看 第 13.6.1.3 节 “Automember 规则”。

   a. 在 Inclusive 或 Exclusive 部分中，点 Add。

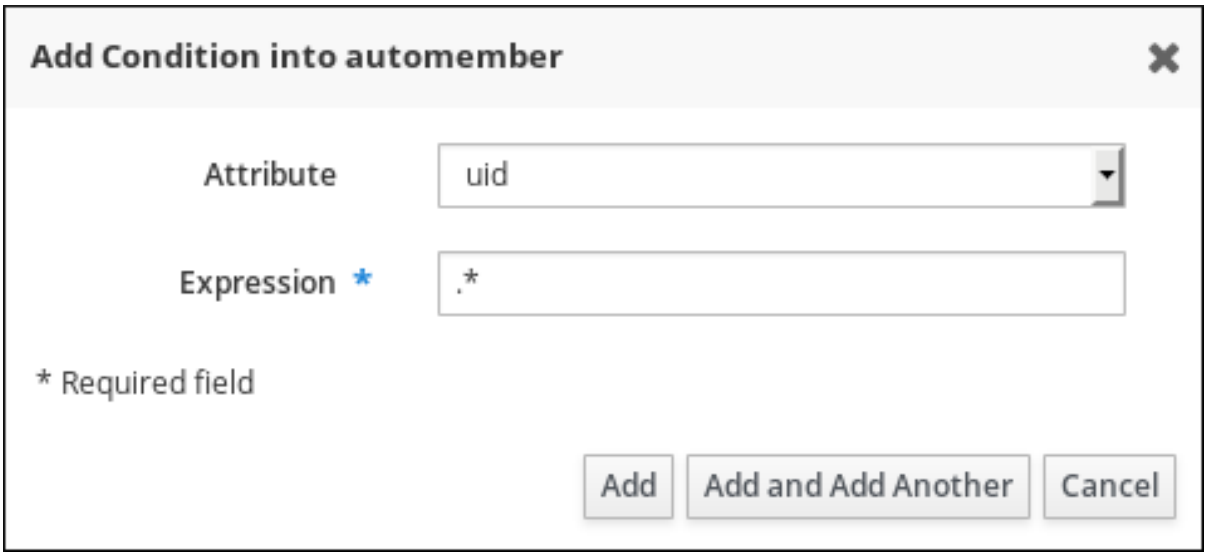
   b. 在 Attribute 字段中，选择所需的属性。

   c. 在 Expression 字段中，定义正则表达式。

   d. 点击 Add。

例如，以下条件以其用户登录属性(uid)中具有 任何值(IANA)的所有用户为目标。

图 13.5. 添加自动成员规则条件



命令行：添加自动成员规则

1. 使用 ipa automember-add 命令添加自动成员规则。在提示时，指定：

   - 自动成员规则，与目标组名称匹配。
   - 分组 Type，它指定规则是否以用户组还是主机组为目标。要以用户组为目标，请输入 group。要以主机组为目标，请输入 hostgroup。

   例如，要为名为 user_group 的用户组添加自动成员规则：

   ```
   $ ipa automember-add
   Automember Rule: user_group
   Grouping Type: group
   ---------------------------------
   Added automember rule "user_group"
   ---------------------------------
     Automember Rule: user_group
   ```

2. 定义一个或多个包含和独占条件。详情请查看 第 13.6.1.3 节 “Automember 规则”。

   a. 要添加条件，请使用 ipa automember-add-condition 命令。在提示时，指定：

      - 自动成员规则，与目标组名称匹配。

- 属性 Key，用于指定过滤器将应用到的条目属性。例如，用户的 经理。

- 分组 Type，它指定规则是否以用户组还是主机组为目标。要以用户组为目标，请输入 group。要以主机组为目标，请输入 hostgroup。

- 包含 正则表达式和 显式 正则表达式，其将一个或多个条件指定为正则表达式。如果您只想指定一个条件，请在提示输入其它条件时按 Enter 键。

例如，以下条件以其用户登录属性(uid)中具有 任何值(IANA)的所有用户为目标。

```
$ ipa automember-add-condition
Automember Rule: user_group
Attribute Key: uid
Grouping Type: group
[Inclusive Regex]: .*
[Exclusive Regex]:
---------------------------------
Added condition(s) to "user_group"
---------------------------------
  Automember Rule: user_group
  Inclusive Regex: uid=.*
----------------------------
Number of conditions added 1
----------------------------
```

b. 要删除条件，请使用 ipa automember-remove-condition 命令。

例 13.5. 命令行：创建 Automember 规则以将所有条目添加到单个组中

通过为所有用户或主机条目包含的属性创建包含条件，如 cn 或 fqdn，您可以确保以后创建的所有用户或主机将添加到单个组中。

1. 创建组，如名为 all_hosts 的主机组。请参阅 第 13.2 节 “添加和删除用户或主机组”。

2. 为新主机组添加自动成员规则。例如：

```
$ ipa automember-add
Automember Rule: all_hosts
Grouping Type: hostgroup
-------------------------------------
```

```
Added automember rule "all_hosts"
-----------------------------------
  Automember Rule: all_hosts
```

3. 添加以所有主机为目标的包含条件。在以下示例中，包含条件以 fqdn 属性中的任何值 (IANA)的主机为目标：

```
$ ipa automember-add-condition
Automember Rule: all_hosts
Attribute Key: fqdn
Grouping Type: hostgroup
[Inclusive Regex]: .*
[Exclusive Regex]:
-------------------------------
Added condition(s) to "all_hosts"
-------------------------------
  Automember Rule: all_hosts
  Inclusive Regex: fqdn=.*
-----------------------------
Number of conditions added 1
-----------------------------
```

以后添加的所有主机将自动成为 all_hosts 组的成员。

例 13.6. 命令行：为同步 AD 用户创建自动成员规则

从 Active Directory (AD)同步的 Windows 用户共享 ntUser 对象类。通过创建一个以 objectclass 属性中带有 ntUser 的所有用户的自动成员条件，您可以确保以后创建的所有同步 AD 用户都包含在 AD 用户的通用组中。

1. 为 AD 用户创建一个用户组，如 ad_users。请参阅 第 13.2 节 “添加和删除用户或主机组”。

2. 为新用户组添加自动成员规则。例如：

```
$ ipa automember-add
Automember Rule: ad_users
Grouping Type: group
-----------------------------------
Added automember rule "ad_users"
-----------------------------------
  Automember Rule: ad_users
```

3. 添加包含条件以过滤 AD 用户。在以下示例中，inclusive 条件以 objectclass 属性中的 ntUser 值的所有用户为目标：

```
$ ipa automember-add-condition
Automember Rule: ad_users
Attribute Key: objectclass
Grouping Type: group
[Inclusive Regex]: ntUser
[Exclusive Regex]:
----------------------------------
Added condition(s) to "ad_users"
----------------------------------
  Automember Rule: ad_users
  Inclusive Regex: objectclass=ntUser
----------------------------
Number of conditions added 1
----------------------------
```

以后添加的所有 AD 用户将自动成为 ad_users 用户组的成员。

13.6.3. 将自动成员规则应用到现有用户和主机

Automember 规则自动应用到添加规则后创建的用户和主机条目。它们不会追溯应用到添加规则之前存在的条目。

要将自动成员规则应用到添加规则前存在的条目，请手动重建自动成员资格。重建自动成员身份重新评估所有现有自动成员规则，并将其应用于所有条目或特定条目。

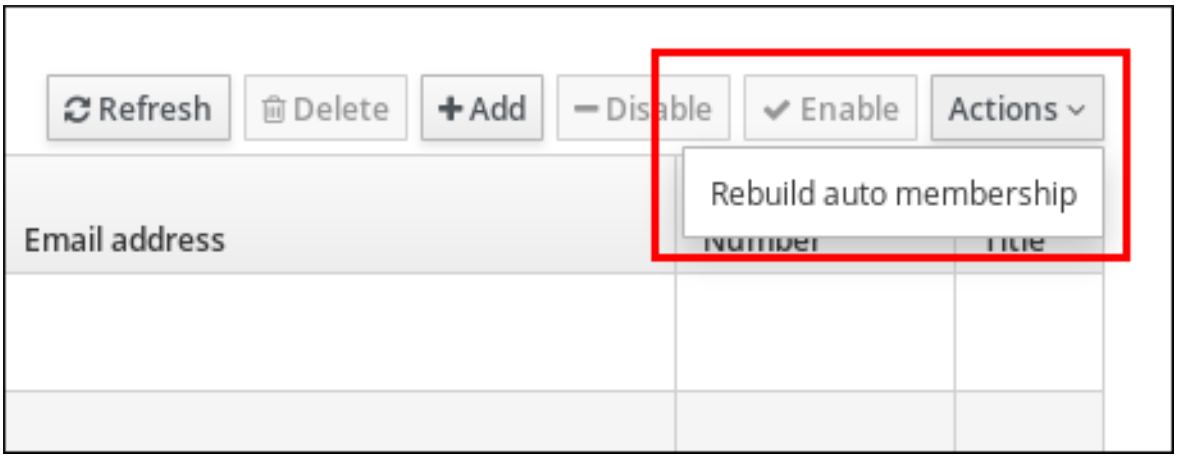
Web UI：重建现有条目的自动成员身份

重新构建所有用户或所有主机的自动成员资格：

1. 选择 Identity → Users 或 Hosts。

2. 单击 Actions → Rebuild auto membership。

图 13.6. 为所有用户或主机重建自动成员身份

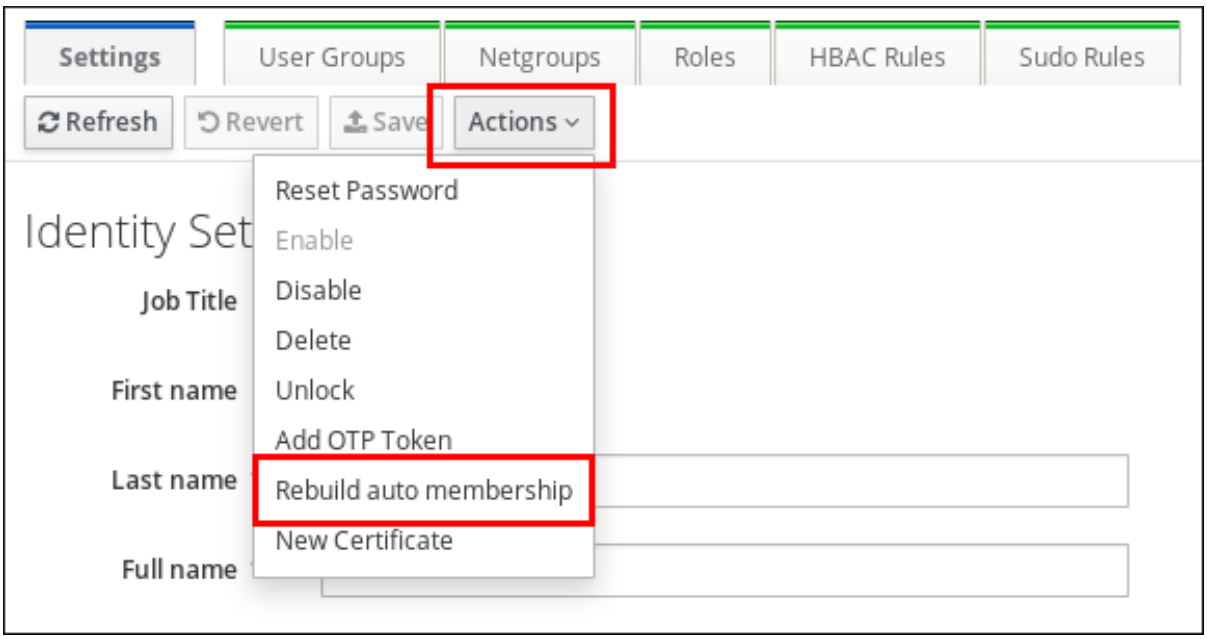


仅为单个用户或主机重建自动成员资格：

1. *选择 **Identity → Users 或 Hosts，然后单击所需的用户登录或主机名。***

2. *单击 **Actions → Rebuild auto membership。***

图 13.7. 为单个用户或主机重建自动成员身份



命令行：为现有条目重建自动成员

要为所有用户重建自动成员资格，请使用 ipa automember-rebuild --type=group 命令：

```
$ ipa automember-rebuild --type=group
----------------------------------------------------------
Automember rebuild task finished. Processed (9) entries.
----------------------------------------------------------
```

-

要为所有用户重建自动成员资格，请使用 **ipa automember-rebuild --type=hostgroup** 命令。

要为指定用户或用户重建自动成员资格，请使用 **ipa automember-rebuild --users=user** 命令：

```
$ ipa automember-rebuild --users=user1 --users=user2
----------------------------------------------------------
Automember rebuild task finished. Processed (2) entries.
----------------------------------------------------------
```

要为指定主机或主机重建自动成员资格，请使用 **ipa automember-rebuild --hosts=example.com** 命令。

13.6.4. 配置默认自动成员组

配置了默认的 **automember** 组时，与任何 **automember** 规则不匹配的用户或主机条目将自动添加到 **default** 组。

1. 使用 **ipa automember-default-group-set** 命令配置默认的自动成员组。在提示时，指定：

   - **Default (fallback) Group**，指定目标组名称。
   - **Grouping Type**，指定目标是用户组还是主机组。要以用户组为目标，请输入 **group**。要以主机组为目标，请输入 **hostgroup**。

   例如：

   ```
   $ ipa automember-default-group-set
   Default (fallback) Group: default_user_group
   Grouping Type: group
   ---------------------------------------------------
   Set default (fallback) group for automember "default_user_group"
   ---------------------------------------------------
     Default (fallback) Group:
   cn=default_user_group,cn=groups,cn=accounts,dc=example,dc=com
   ```

2. 要验证组是否已正确设置，请使用 **ipa automember-default-group-show** 命令。命令显示当前的默认的自动成员组。例如：

```
$ ipa automember-default-group-show
Grouping Type: group
  Default (fallback) Group:
cn=default_user_group,cn=groups,cn=accounts,dc=example,dc=com
```

要删除当前的默认自动成员组，请使用 ipa automember-default-group-remove 命令。

第 14 章 唯一 UID 和 GID 编号分配

IdM 服务器生成用户 ID(UID)和组 ID(GID)值，同时确保副本永远不会生成相同的 ID。如果单个机构使用多个单独的域，对唯一 UID 和 GID 的要求甚至可能跨越 IdM 域。

14.1. ID 范围

UID 和 GID 编号分为多个 ID 范围。通过为个别服务器和副本保留单独的数字范围，为条目发出的 ID 值已由另一服务器上的另一条目使用的可能性很小。

分布式数字分配(DNA)插件，作为域的后端 389 目录服务器实例的一部分，确保范围在服务器和副本之间更新和共享；插件在所有 master 和副本之间管理 ID 范围。每个服务器或副本都有一个当前的 ID 范围，并在当前范围耗尽后服务器或副本使用的额外 下一个 ID 范围。有关 DNA 目录服务器插件的更多信息，请参阅 红帽目录服务器部署指南。

14.2. 安装期间 ID 范围分配

在服务器安装过程中，ipa-server-install 命令默认会自动将随机当前 ID 范围分配给安装的服务器。设置脚本从总计 10,000 个可能的范围中随机选择 200,000 个 ID。当您决定以后合并两个独立的 IdM 域时，以这种方法选择一个随机范围可显著降低冲突 ID 的可能性。

但是，您可以使用 ipa-server-install 的以下两个选项在服务器安装过程中手动定义当前的 ID 范围：

- --idstart 为 UID 和 GID 号提供了起始值；默认情况下，会随机选择值，
- --idmax 给出 UID 和 GID 号的最大值；默认情况下，值为 --idstart 起始值加上 199,999。

如果您安装了单个 IdM 服务器，新的用户或组条目会从整个范围内接收随机 ID。当您安装新副本并且副本请求自己的 ID 范围时，服务器的初始 ID 范围会在服务器和副本之间分配：副本接收在初始 master 上可用的其余 ID 范围的一半。然后，服务器及副本将原始 ID 范围内的相应部分用于新条目。此外，如果分配到副本范围的 ID 范围内的 100 个 ID 少于 100 个 ID，则副本将保留，表示副本已接近其分配 ID 范围，则副本通过请求新的 ID 范围联系其他可用服务器。

服务器第一次使用 DNA 插件时收到 ID 范围；在此之前，服务器没有定义 ID 范围。例如，当您从主服务器创建副本时，副本不会立即收到 ID 范围。仅当副本上要分配第一个 ID 时，副本才会从初始主控机请求 ID 范围。

注意

如果初始 master 在副本请求 ID 范围之前停止运行，则副本无法通过 ID 范围的请求与主控机联系。尝试在副本中添加新用户会失败。在这种情况下，您可以找出分配给禁用的 master 的 ID 范围，并为副本手动分配 ID 范围，如 第 14.5 节 “手动 ID 范围扩展和分配新 ID 范围” 所述。

14.3. 显示当前分配 ID 范围

要显示为服务器配置了哪些 ID 范围，请使用以下命令：

- ipa-replica-manage dnarange-show 显示所有服务器上设置的当前 ID 范围，或者如果您指定了一个服务器，则仅显示指定服务器上的当前 ID 范围，例如：

```
# ipa-replica-manage dnarange-show
masterA.example.com: 1001-1500
masterB.example.com: 1501-2000
masterC.example.com: No range set

# ipa-replica-manage dnarange-show masterA.example.com
masterA.example.com: 1001-1500
```

- ipa-replica-manage dnanextrange-show 显示当前在所有服务器上设置的下一个 ID 范围，或者如果您指定了一个服务器，则仅显示指定服务器上的下一个 ID 范围，例如：

```
# ipa-replica-manage dnanextrange-show
masterA.example.com: 1001-1500
masterB.example.com: No on-deck range set
masterC.example.com: No on-deck range set

# ipa-replica-manage dnanextrange-show masterA.example.com
 masterA.example.com: 1001-1500
```

有关这两个命令的详情请参考 ipa-replica-manage(1) man page。

14.4. 删除副本后自动 ID 范围扩展

当您删除可正常工作的副本时，ipa-replica-manage del 命令会检索分配给副本的 ID 范围，并将它们作为下一个范围添加到其他可用 IdM 副本。这样可确保 ID 范围仍然可供其他副本使用。

删除副本后，您可以使用 ipa-replica-manage dnarange-show 和 ipa-replica-manage

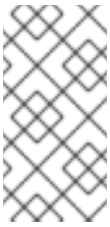
dnanextrange-show 命令验证为其他服务器配置了哪些 ID 范围，如 第 14.3 节 “显示当前分配 ID 范围” 所述。

14.5. 手动 ID 范围扩展和分配新 ID 范围

在某些情况下，需要手动调整 ID 范围：

分配的 ID 范围已被耗尽

副本已耗尽分配给它的 ID 范围，请求额外 ID 会失败，因为其他副本的 ID 范围内没有其他可用 ID。您需要扩展分配给副本的 ID 范围。这可能涉及分割现有的 ID 范围，或者超过服务器初始配置的 ID 范围。或者，您可能想要分配一个新的 ID 范围。



注意

如果您分配了新的 ID 范围，则服务器上已存在的条目的 UID 将保持不变。这不会造成问题，因为即使您更改了当前的 ID 范围，IdM 也会保留过去分配的范围的记录。

副本停止工作

当副本终止且需要删除时，不会自动检索 ID 范围，这意味着之前分配给副本的 ID 范围变得不可用。您需要恢复 ID 范围，并使其可用于其他副本。

如果要恢复属于停止工作的服务器的 ID 范围，并将其分配给其他服务器，首先使用 ipa-replica-manage dnarange-show 命令找出 ID 范围值在 第 14.3 节 “显示当前分配 ID 范围” 中描述的，然后手动将该 ID 范围分配给服务器。此外，为了避免重复的 UID 或 GID，确保之前未将恢复范围内的 ID 值分配给用户或组；您可以通过检查现有用户和组的 UID 和 GID 来执行此操作。

要手动定义 ID 范围，请使用以下命令：

- ipa-replica-manage dnarange-set 允许您为指定服务器定义当前的 ID 范围：

```
# ipa-replica-manage dnarange-set masterA.example.com 1250-1499
```

- ipa-replica-manage dnanextrange-set 允许您为指定服务器定义下一个 ID 范围：

```
# ipa-replica-manage dnanextrange-set masterB.example.com 1001-5000
```

有关这些命令的详情请参考 ipa-replica-manage(1) man page。

重要

注意不要创建重叠的 ID 范围。如果您分配给服务器或副本的任何 ID 范围重叠了，可能会导致两个不同的服务器给不同的条目分配了相同的 ID 值。

不要设置 UID 值为 1000 或更低的 ID 范围；这些值是保留给系统使用的。另外，不要设置包含 0 值的 ID 范围；SSSD 服务不会处理 0 ID 值。

手动扩展 ID 范围时，请确保新扩展范围包含在 IdM ID 范围内；您可以使用 ipa idrange-find 命令检查它。运行 ipa idrange-find -h 命令显示如何使用 ipa idrange-find 的帮助信息。

14.6. 确保唯一 ID 值

建议您避免冲突 UID 或 GID。UID 和 GID 应始终是唯一的：两个用户不应具有相同的 UID，并且两个组不应具有相同的 GID。

自动 ID 分配

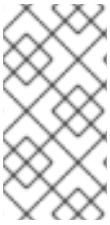
当以交互方式创建用户或组时，或者没有手动指定 ID 号，服务器会将 ID 范围内的下一个可用 ID 编号分配到用户帐户。这样可确保 UID 或 GID 始终是唯一的。

手动 ID 分配

当您手动为用户或组条目分配 ID 时，服务器不会验证指定的 UID 或 GID 唯一；如果您选择已由其他条目使用的值，它不会警告您发生冲突。

如 第 14.7 节 “修复更改的 UID 和 GID 号” 中所述，SSSD 服务不会处理 ID 相同的条目。如果两个条目共享相同的 ID 号，则搜索此 ID 仅返回第一个条目。但是，如果您搜索其他属性或运行 ipa user-find --all 命令，则返回这两个条目。

UID 和 GID 都从同一 ID 范围内选择。用户和组可以具有相同的 ID；在这种情况下不会发生任何冲突，因为 UID 和 GID 在两个不同的属性中设置： uidNumber 和 gidNumber。



注意

通过为用户和组设置相同的 ID，您可以配置用户专用组。要以这种方式为用户创建唯一系统组，请为用户和组设置相同的 ID 值，其中唯一成员是上述用户。

14.7. 修复更改的 UID 和 GID 号

当用户登录 IdM 系统或服务时，该系统中的 SSSD 会将其用户名与用户的 UID 和 GID 一起缓存。然后，SSSD 将 UID 用作用户的标识密钥。如果具有相同用户名但不同 UID 试图登录到系统的用户，SSSD 会注册两个不同的 UID，并假定存在两个不同的用户，且用户名冲突。如果用户的 UID 更改，这可能会造成问题。在这种情况下，SSSD 会错误地将修改后的 UID 的用户解释为新用户，而不是将其识别为具有不同 UID 的同一用户。如果现有用户的 UID 更改，用户就无法登录 SSSD 及关联的服务和域。这也会影响使用 SSSD 作为身份信息的客户端应用程序。

要临时解决这个问题，如果 UID 或 GID 更改，清除 SSSD 缓存，这可确保用户可以再次登录。例如，要清除指定用户的 SSSD 缓存，请使用 sss_cache 工具，如下所示：

```
[root@server ~]# sss_cache -u user
```

第 15 章 用户和组架构

创建用户条目时，会自动为其分配特定的 LDAP 对象类，这些类反过来会提供某些属性。LDAP 属性是信息存储在 目录中的方式。（在 Directory Server 部署指南和目录服务器架构 参考 中详细讨论。）

表 15.1. 默认身份管理用户对象类

对象类	描述
ipaobject ipasshuser	IdM 对象类
人 OrganizationPerson inetorgperson inetuser posixAccount	人员对象类
krbprincipalaux krbticketpolicyaux	Kerberos 对象类
mepOriginEntry	受管条目(template)对象类

用户条目可以使用多个属性：有些是手动设置的，如果未设置特定值，则根据默认值设置。还有一个选项，可以添加 表 15.1 “默认身份管理用户对象类” 中的对象类中可用属性，即使该属性没有 UI 或命令行参数。另外，也可以配置默认属性生成的或使用的值，如 第 15.4 节 “指定默认用户和组属性” 所示。

表 15.2. 默认身份管理用户属性

UI 字段	命令行选项	必需、可选或默认[a]
用户登录	*username*	必需
名	**--first**	必需
姓	**--last**	必需
全名	**--cn**	选填
显示名称	**--displayname**	选填
初始	**--initials**	Default（默认）

UI 字段	命令行选项	必需、可选或默认[a]
主目录	**--homedir**	Default（默认）
GECOS 字段	**--gecos**	Default（默认）
shell	**--shell**	Default（默认）
Kerberos 主体	**--principal**	Default（默认）
电子邮件地址	**--email**	选填
密码	**--password** [b]	选填
用户 ID 号	**--uid**	Default（默认）
组 ID 号	**--gidnumber**	Default（默认）
街道地址	**--street**	选填
City	**--city**	选填
州/省	**--state**	选填
zip 代码	**--postalcode**	选填
电话号码	**--phone**	选填
手机电话号码	**--mobile**	选填
寻呼器编号	**--pager**	选填
传真号码	**--fax**	选填
组织单元	**--orgunit**	选填
任务标题	**--title**	选填
Manager（管理者）	**--manager**	选填
汽车许可证	**--carlicense**	选填
	--noprivate	选填
SSH 密钥	**--sshpubkey**	选填

UI 字段	命令行选项	必需、可选或默认[a]
其他属性	**--addattr**	选填
部门编号	**--departmentnumber**	选填
员工号	**--employeenumber**	选填
员工类型	**--employeetype**	选填
首选语言	**--preferredlanguage**	选填

[a] 必须为每个条目设置必要属性。可选属性可以设置，而默认属性会自动添加预定义的值，除非指定了特定值。

[b] 脚本会提示输入新密码，而不是通过 参数接受值。

15.1. 关于更改默认用户和组架构

可以添加或更改用于用户和组条目的对象类和属性(第 15 章 用户和组架构)。

IdM 配置在对象类更改时提供一些验证：

- *LDAP 服务器必须知道所有对象类及其指定属性。*
- *为条目配置的所有默认属性都必须被配置的对象类支持。*

但是，IdM 模式验证存在限制。最重要的是，IdM 服务器不会检查定义的用户或组对象类是否包含 IdM 条目所需的所有对象类。例如，所有 IdM 条目都需要 ipaobject 对象类。但是，当用户或组架构发生更改时，服务器不会检查确保包含此对象类；如果对象类被意外删除，则将来的条目添加操作将失败。

此外，所有对象类更改都是原子的，而不是增量的。每次有更改时都必须定义默认对象类的整个列表。例如，公司可以创建自定义对象类别来存储员工信息，如生日和就业开始日期。管理员不能简单地将自定义对象类添加到列表中；必须设置当前默认对象类的整个列表 加上 新的对象类。更新配置时，必须始终包含 现有的 默认对象类。否则，将覆盖当前设置，这会导致严重的性能问题。

15.2. 将自定义对象类应用到新用户条目

使用应用到该条目的一组预定义的 LDAP 对象类创建用户和组帐户。属于对象类的任何属性都可以添

加到用户条目中。

虽然标准和特定于 IdM 的 LDAP 对象类将涵盖大多数部署场景，但管理员可以使用自定义属性创建自定义对象类。请注意，在管理员修改默认对象类列表后，新条目将包含自定义对象类，但旧条目不会被自动修改。

15.2.1. 使用 Web UI

1. 将所有自定义架构元素添加到身份管理使用的 389 目录服务器实例中。目录服务器管理员指南的 schema 章节 介绍了添加架构元素。

2. 打开 IPA Server 选项卡。

3. 选择 Configuration 子选项卡。

4. 滚动到 User Options 区域。

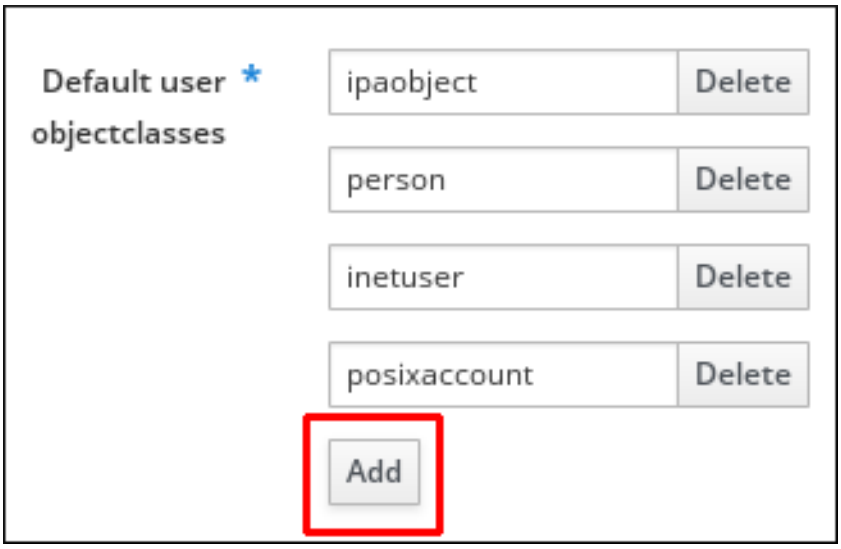
图 15.1. 服务器配置中的用户选项

5. 在用户区域的底部，单击 Add 以为另一个对象类包含一个新字段。

重要

更新配置时，始终包含 现有的 默认对象类。否则，将覆盖当前设置。如果没有包含身份管理所需的任何对象类，后续的尝试添加条目将失败，并显示对象类违反情况。

图 15.2. 更改默认用户对象类



6. 更改完成后，点 Configuration 页面顶部的 Save。

15.2.2. 从命令行

1. 将所有自定义架构元素添加到身份管理使用的 389 目录服务器实例中。目录服务器管理员指南的 schema 章节 介绍了添加架构元素。

2. 将新对象类添加到添加到条目的对象类列表中。用户对象类的选项是 --userobjectclasses。

重要

更新配置时，始终包含 现有的 默认对象类。否则，将覆盖当前设置。如果没有包含身份管理所需的任何对象类，后续的尝试添加条目将失败，并显示对象类违反情况。

所有对象类都必须包含在对象类列表中。通过 config-mod 命令传递的信息会覆盖之前的值。这可以通过 --userobjectclasses 参数指定各个对象类，或者将逗号分隔的所有对象类列在不允许有空格的大括号内，如 {attr1,attr2,attr3}。特别对于长列表而言，使用大括号比多个选项更容易。例如：

```
[bjensen@server ~]$ ipa config-mod --
userobjectclasses={top,person,organizationalperson,inetorgperson,inetuser,posixaccount,krbpr
incipalaux,krbticketpolicyaux,ipaobject,ipasshuser,employeeinfo}
```

注意

要使用大括号选项，必须打开 大括号扩展功能。要激活这个功能，请使用 set 命令：

```
# set -o braceexpand
```

15.3. 将自定义对象类应用到新组条目

与用户条目一样，管理员可以使用自定义属性创建自定义对象类。它们可以通过将对象类添加到 IdM 服务器配置来自动添加。请注意，在管理员修改默认对象类列表后，新条目将包含自定义对象类，但旧条目不会被自动修改。

15.3.1. 使用 Web UI

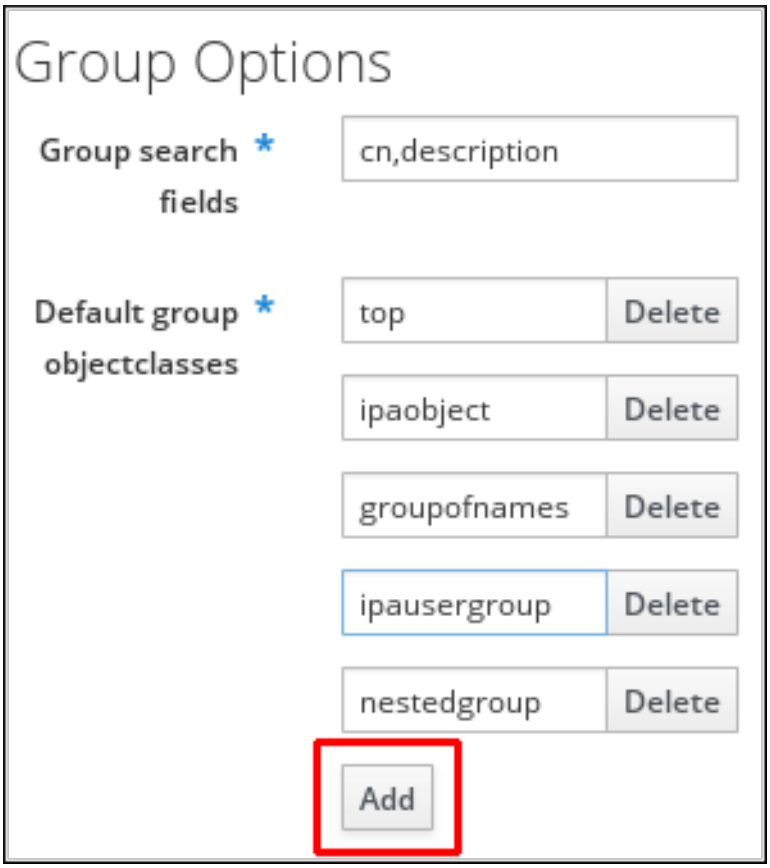
1. 将所有自定义架构元素添加到身份管理使用的 389 目录服务器实例中。目录服务器管理员指南的 schema 章节 介绍了添加架构元素。

2. 打开 IPA Server 选项卡。

3. 选择 Configuration 子选项卡。

4. 滚动到 Group Options 区域。

图 15.3. 服务器配置中的组选项



5. 点 Add 使其包含另一个对象类的新字段。

重要

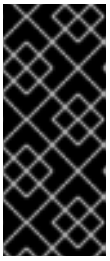
更新配置时，始终包含 现有的 默认对象类。否则，将覆盖当前设置。如果没有包含身份管理所需的任何对象类，后续的尝试添加条目将失败，并显示对象类违反情况。

6. 更改完成后，点 Configuration 页面顶部的 Save。

15.3.2. 从命令行

1. 将所有自定义架构元素添加到身份管理使用的 389 目录服务器实例中。目录服务器管理员指南的 schema 章节 介绍了添加架构元素。

2. 将新对象类添加到添加到条目的对象类列表中。组对象类的选项是 --groupobjectclasses。



重要

更新配置时，始终包含现有的默认对象类。否则，将覆盖当前设置。如果没有包含身份管理所需的任何对象类，后续的尝试添加条目将失败，并显示对象类违反情况。

所有对象类都必须包含在对象类列表中。通过 config-mod 命令传递的信息会覆盖之前的值。这可以通过 --groupobjectclasses 参数指定各个对象类，或者将逗号分隔的所有对象类列在不允许有空格的大括号内，如 {attr1,attr2,attr3}。特别对于长列表而言，使用大括号比多个选项更容易。例如：

```
[bjensen@server ~]$ ipa config-mod --
groupobjectclasses={top,groupofnames,nestedgroup,ipausergroup,ipaobject,ipasshuser,empl
oyeegroup}
```

15.4. 指定默认用户和组属性

身份管理在创建新条目时使用模板。

对于用户，模板非常具体。身份管理为 IdM 用户帐户使用多个核心属性使用默认值。这些默认值可以定义用户帐户属性的实际值（如主目录位置），也可以定义属性值的格式，如用户名长度。这些设置还定义分配给用户的对象类。

对于组，模板仅定义分配的对象类。

这些默认定义都包含在 IdM 服务器的一个配置条目 cn=ipaconfig,cn=etc,dc=example,dc=com 中。

可以使用 ipa config-mod 命令更改配置。

表 15.3. 默认用户参数

字段	命令行选项	描述
最大用户名长度	**--maxusername**	设置用户名的最大字符数。默认值为 32。
主目录的根	**--homedirectory**	设置要用于用户主目录的默认目录。默认值为 **/home**。

字段	命令行选项	描述
默认 shell	**--defaultshell**	设置供用户使用的默认 shell.默认值为 **/bin/sh**。
默认用户组群	**--defaultgroup**	设置为所有新建帐户添加到的默认组。默认值为 **ipausers**，它会在 IdM 服务器安装过程中自动创建。
默认电子邮件域	**--emaildomain**	将电子邮件地址设置为使用 来基于新帐户创建电子邮件地址。默认为 IdM 服务器域。
搜索时间限制	**--searchtimelimit**	在服务器返回结果之前，设置搜索上花费的最大时间（以秒为单位）。
搜索大小限制	**--searchrecordslimit**	设置搜索返回的最大记录数.
用户搜索字段	**--usersearch**	设置用户条目中的字段，可用作搜索字符串。列出的任何属性都有一个用于该属性的索引，因此设置太多属性可能会影响服务器性能。
组搜索字段	**--groupsearch**	设置组条目中的字段，可用作搜索字符串。
证书主题基础		设置在为客户端证书创建主题 DN 时要使用的基本 DN。这是在设置服务器时配置的。
默认用户对象类	**--userobjectclasses**	定义用于创建 IdM 用户帐户的对象类。这可以多次调用。必须提供对象类的完整列表，因为运行 命令时会覆盖该列表。
默认组对象类	**--groupobjectclasses**	定义用于创建 IdM 组帐户的对象类。这可以多次调用。必须提供对象类的完整列表，因为运行 命令时会覆盖该列表。
密码过期通知	**--pwdexpnotify**	设置服务器发送通知的密码到期前的天数（以天数为单位）。
密码插件功能		设置用户允许的密码格式。

15.4.1. 从 Web UI 查看属性

1. *打开 **IPA Server** 选项卡。*

2. *选择 **Configuration** 子选项卡。*

3. *完整的配置条目在三个部分显示，一个用于所有搜索限制，一个用于用户模板，另一个用于组模板。*

图 15.4. 设置搜索限制

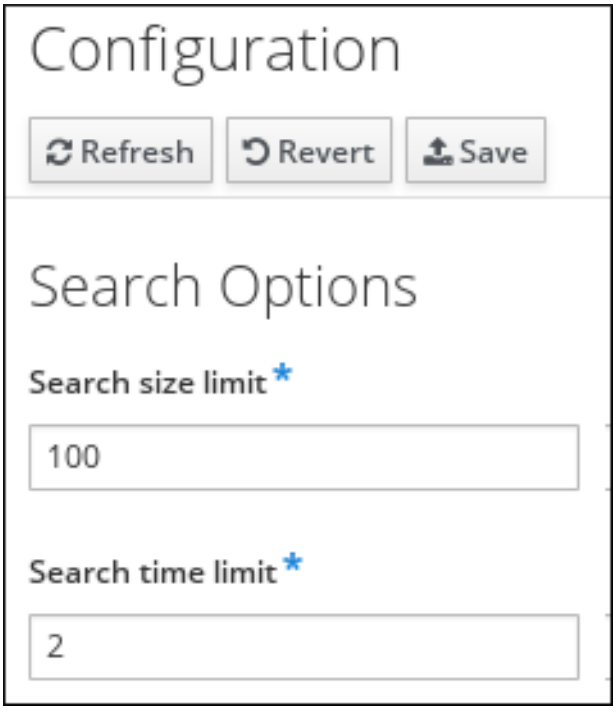
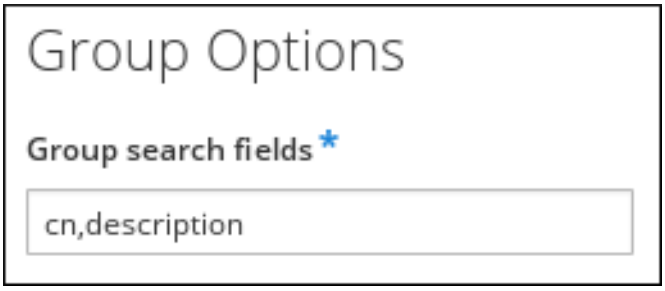


图 15.5. 用户属性

图 15.6. 组属性



15.4.2. 从命令行查看属性

config-show 命令显示适用于所有新用户帐户的当前配置。默认情况下，仅显示最常见的属性；使用 --all 选项显示完整的配置。

```
[bjensen@server ~]$ kinit admin
[bjensen@server ~]$ ipa config-show --all
dn: cn=ipaConfig,cn=etc,dc=example,dc=com
Maximum username length: 32
Home directory base: /home
Default shell: /bin/sh
Default users group: ipausers
Default e-mail domain: example.com
Search time limit: 2
Search size limit: 100
User search fields: uid,givenname,sn,telephonenumber,ou,title
Group search fields: cn,description
Enable migration mode: FALSE
Certificate Subject base: O=EXAMPLE.COM
Default group objectclasses: top, groupofnames, nestedgroup, ipausergroup, ipaobject
Default user objectclasses: top, person, organizationalperson, inetorgperson, inetuser, posixaccount,
krbprincipalaux, krbticketpolicyaux, ipaobject, ipasshuser
Password Expiration Notification (days): 4
Password plugin features: AllowNThash
SELinux user map order: guest_u:s0$xguest_u:s0$user_u:s0$staff_u:s0-
s0:c0.c1023$unconfined_u:s0-s0:c0.c1023
Default SELinux user: unconfined_u:s0-s0:c0.c1023
Default PAC types: MS-PAC, nfs:NONE
cn: ipaConfig
objectclass: nsContainer, top, ipaGuiConfig, ipaConfigObject
```

第 16 章 管理服务

主机上运行的一些服务也可以属于 IdM 域。可以存储 Kerberos 主体或 SSL 证书（或两者）的任何服务都可以配置为 IdM 服务。向 IdM 域添加服务可让服务从域请求 SSL 证书或 keytab。（仅证书的公钥存储在服务记录中。私钥是该服务的本地密钥。）

IdM 域在机器之间建立通用性，具有通用身份信息、通用策略和共享服务。属于域的任何计算机充当域的客户端，这意味着它使用域所提供的服务。IdM 域（如 第 1 章 红帽身份管理简介所述）为机器提供三个主要服务：

- DNS
- Kerberos
- 证书管理

16.1. 添加和编辑服务条目和密钥选项卡

与主机条目一样，主机（以及该主机上的任何其他服务）的服务条目必须手动添加到 IdM 域中。这个过程分为两个步骤。首先，必须创建服务条目，然后必须为该服务创建一个 keytab，它将用于访问域。

默认情况下，身份管理将其 HTTP keytab 保存到 /etc/httpd/conf/ipa.keytab。

注意

此 keytab 用于 Web UI。如果密钥存储在 ipa.keytab 中，且删除了 keytab 文件，IdM Web UI 将停止工作，因为原始密钥也会被删除。

可以为每个需要让 Kerberos 感知的服务指定相似的位置。没有必须使用的特定位置，但在使用 ipa-getkeytab 时，您应该避免使用 /etc/krb5.keytab。此文件不应包含特定于服务的 keytab；每个服务都应将其 keytab 保存在特定位置，并且应配置访问权限（以及 SELinux 规则），以便只有此服务有权访问 keytab。

16.1.1. 从 Web UI 添加服务和 keytab

1. 打开 Identity 选项卡，然后选择 Services 子选项卡。

2. 单击 services 列表顶部的 Add 按钮。

3. 从下拉菜单中选择服务类型，并为它指定一个名称。

4. 选择运行该服务的 IdM 主机的主机名。主机名用于构造完整的服务主体名称。

5. 点 Add 按钮保存新服务主体。

6. 使用 ipa-getkeytab 命令为服务主体生成并分配新的 keytab。

```
[root@ipaserver ~]# # ipa-getkeytab -s ipaserver.example.com -p HTTP/server.example.com
-k /etc/httpd/conf/krb5.keytab -e aes256-cts
```

- realm name 是可选的。IdM 服务器会自动附加配置了它的 Kerberos 域。您无法指定不同的域。

- 主机名必须解析为 DNS A 记录，然后才能用于 Kerberos。如果需要，您可以使用 --force 标志强制创建主体。

- e 参数可以包含 要在 keytab 中包含的加密类型列表。这会取代任何默认加密类型。可以多次使用 选项设置条目列表，也可通过将选项列在大括号内的逗号分隔列表中，如 --option={val1,val2,val3}。

警告

创建新密钥为指定主体重置 secret。这意味着该主体的所有其他 keytab 都会变为无效。

16.1.2. 从命令行添加服务和 keytab

1. 创建服务主体。该服务通过名称（如 service/FQDN）来识别 ：

```
# ipa service-add serviceName/hostname
```

例如：

```
$ ipa service-add HTTP/server.example.com
---------------------------------------------------------
Added service "HTTP/server.example.com@EXAMPLE.COM"
---------------------------------------------------------
  Principal: HTTP/server.example.com@EXAMPLE.COM
  Managed by: ipaserver.example.com
```

2. 使用 ipa-getkeytab 命令创建服务 keytab 文件。此命令在 IdM 域中的客户端上运行。（实际上，它可以在任何 IdM 服务器或客户端上运行，然后复制到相应机器的密钥。不过，最简便的做法是在正在创建的 计算机上运行 命令。）

命令需要 Kerberos 服务主体(-p)、IdM 服务器名称(-s)、要写入的文件(-k)和加密方法(-e)。务必将 keytab 复制到服务的相应目录。

例如：

```
# ipa-getkeytab -s server.example.com -p HTTP/server.example.com -k
/etc/httpd/conf/krb5.keytab -e aes256-cts
```

- realm name 是可选的。IdM 服务器会自动附加配置了它的 Kerberos 域。您无法指定不同的域。

- 主机名必须解析为 DNS A 记录，然后才能用于 Kerberos。如果需要，您可以使用 --force 标志强制创建主体。

- e 参数可以包含 要包含在 keytab 中的以逗号分隔的加密类型列表。这会取代任何默认加密类型。可以多次使用 选项设置条目列表，也可通过将选项列在大括号内的逗号分隔列表中，如 --option={val1,val2,val3}。

警告

ipa-getkeytab 命令重置指定主体的 secret。这意味着该主体的所有其他 keytab 都会变为无效。

16.2. 配置集群服务

IdM 服务器不知道。但是，可以通过在所有参与的主机中同步 Kerberos 密钥并配置主机上运行的服务来响应客户端使用的任何名称，将集群服务配置为 IdM 的一部分。

1. **将集群中的所有主机注册到 IdM 域中。**
2. **创建任何服务主体并生成所需的 keytab。**
3. **收集主机上为服务设置的所有 keytab，包括位于 /etc/krb5.keytab 的主机 keytab。**
4. **使用 ktutil 命令生成一个 keytab 文件，其中包含所有 keytab 文件的内容。**
   a. **对于每个文件，使用 rkt 命令从该文件中读取密钥。**
   b. **使用 wkt 命令将所有已读取的密钥写入一个新的 keytab 文件。**
5. **将每个主机上的 keytab 文件替换为新创建的组合 keytab 文件。**
6. **此时，此集群中的每个主机现在可以模拟任何其他主机。**
7. **有些服务需要额外的配置，以适应在接管故障服务时不会重置主机名的群集成员。**

   - **对于 sshd，在 /etc/ssh/sshd_config 中设置 GSSAPIStrictAcceptorCheck no。**

- 对于 mod_auth_kerb，请在 /etc/httpd/conf.d/auth_kerb.conf 中设置 KrbServiceName any。

注意

对于 SSL 服务器，当客户端连接到集群主机时，服务器证书的主题名称或替代名称必须正确显示。如果可能，在所有主机之间共享私钥。

如果每个群集成员都包含主题备用名称，其中包含所有其他群集成员的名称，则满足任何客户端连接要求。

16.3. 将相同的服务主体用于多个服务

在群集中，可以将相同的服务主体用于多个服务，分散到不同的机器上。

1. 使用 ipa-getkeytab 命令检索服务主体。

```
# ipa-getkeytab -s kdc.example.com -p HTTP/server.example.com -k
/etc/httpd/conf/krb5.keytab -e aes256-cts
```

2. 将多个服务器或服务定向为使用同一文件，或根据需要将文件复制到个别服务器。

16.4. 检索多个服务器的现有 KEYTAB

在某些情况下，如在集群环境中，不同机器在一个通用主机名上代表的服务需要相同的 keytab 文件。IdM 命令可用于检索每个主机上的同一 keytab。

要准备通用主机名和服务主体，请在 IdM 服务器上运行以下命令：

1. 以 admin 用户身份进行身份验证：

-

```
[root@ipaserver ~]# kinit admin
```

2. 为共享此主机名的所有 IP 地址添加通用的转发 DNS 记录：

```
[root@ipaserver ~]# ipa dnsrecord-add idm.example.com cluster --a-rec=
{192.0.2.40,192.0.2.41}
  Record name: cluster
    A record: 192.0.2.40, 192.0.2.41
```

3. 为通用 DNS 名称创建新主机条目对象：

```
[root@ipaserver ~]# ipa host-add cluster.idm.example.com
---------------------------------------
Added host "cluster.idm.example.com"
---------------------------------------
  Host name: cluster.idm.example.com
  Principal name: host/cluster.idm.example.com@IDM.EXAMPLE.COM
  Password: False
  Keytab: False
  Managed by: cluster.idm.example.com
```

4. 为主机添加服务主体：

```
[root@ipaserver ~]# ipa service-add HTTP/cluster.idm.example.com
------------------------------------------------------------
Added service "HTTP/cluster.idm.example.com@IDM.EXAMPLE.COM"
------------------------------------------------------------
  Principal: HTTP/cluster.idm.example.com@IDM.EXAMPLE.COM
  Managed by: cluster.idm.example.com
```

5. 将主机添加到服务中，该服务应该可以从 IdM 检索 keytab：

```
[root@ipaserver ~]# ipa service-allow-retrieve-keytab HTTP/cluster.idm.example.com --
hosts={node01.idm.example.com,node02.idm.example.com}
  Principal: HTTP/cluster.idm.example.com@IDM.EXAMPLE.COM
  Managed by: cluster.idm.example.com
  Hosts allowed to retrieve keytab: node01.idm.example.com, node02.idm.example.com
-------------------------
Number of members added 2
-------------------------
```

6. 向一个主机授予创建新 keytab 的权限：

```
[root@ipaserver ~]# ipa service-allow-create-keytab HTTP/cluster.idm.example.com --
hosts=node01.idm.example.com
Principal: HTTP/cluster.idm.example.com@IDM.EXAMPLE.COM
Managed by: cluster.idm.example.com
Hosts allowed to retrieve keytab: node01.idm.example.com, node02.idm.example.com
Hosts allowed to create keytab: node01.idm.example.com
-------------------------
Number of members added 1
-------------------------
```

在客户端中，按照以下步骤执行：

1. **使用主机 Kerberos keytab 进行身份验证：**

```
# kinit -kt /etc/krb5.keytab
```

2. 1. **在您授予相应权限的客户端中，生成新的 keytab 并将其存储在文件中：**

```
[root@node01 ~]# ipa-getkeytab -s ipaserver.idm.example.com -p
HTTP/cluster.idm.example.com -k /tmp/client.keytab
```

2. **在所有其他客户端中，通过向命令添加 -r 选项，从 IdM 服务器检索现有的 keytab：**

```
[root@node02 ~]# ipa-getkeytab -r -s ipaserver.idm.example.com -p
HTTP/cluster.idm.example.com -k /tmp/client.keytab
```

警告

请注意，如果省略 **-r** 选项，将生成一个新 **keytab**。这会使之前检索到的所有 **keytab** 无效。

16.5. 禁用和重新启用服务条目

活动服务可由域中的其他服务、主机和用户访问。有些情况下，需要从活动中删除主机或服务。但是，删除服务或主机会删除该条目及所有关联的配置，并且会永久删除。

16.5.1. 禁用服务条目

禁用服务可防止域用户访问该服务，而不将其永久从域中删除。这可以通过使用 service-disable 命令来完成。

对于服务，指定该服务的主体。例如：

```
[jsmith@ipaserver ~]$ kinit admin
[jsmith@ipaserver ~]$ ipa service-disable HTTP/server.example.com
```

重要

禁用主机条目不仅会禁用该主机。它还会禁用该主机上每个配置的服务。

16.5.2. 重新启用服务

禁用服务实质上会终止其当前活动 keytab。删除 keytab 会有效地从 IdM 域中删除该服务，而不涉及其配置条目。

要重新启用服务，只需使用 ipa-getkeytab 命令。-s 选项设定请求 keytab 的 IdM 服务器，-p 提供主体名称，-k 则提供保存 keytab 的文件。

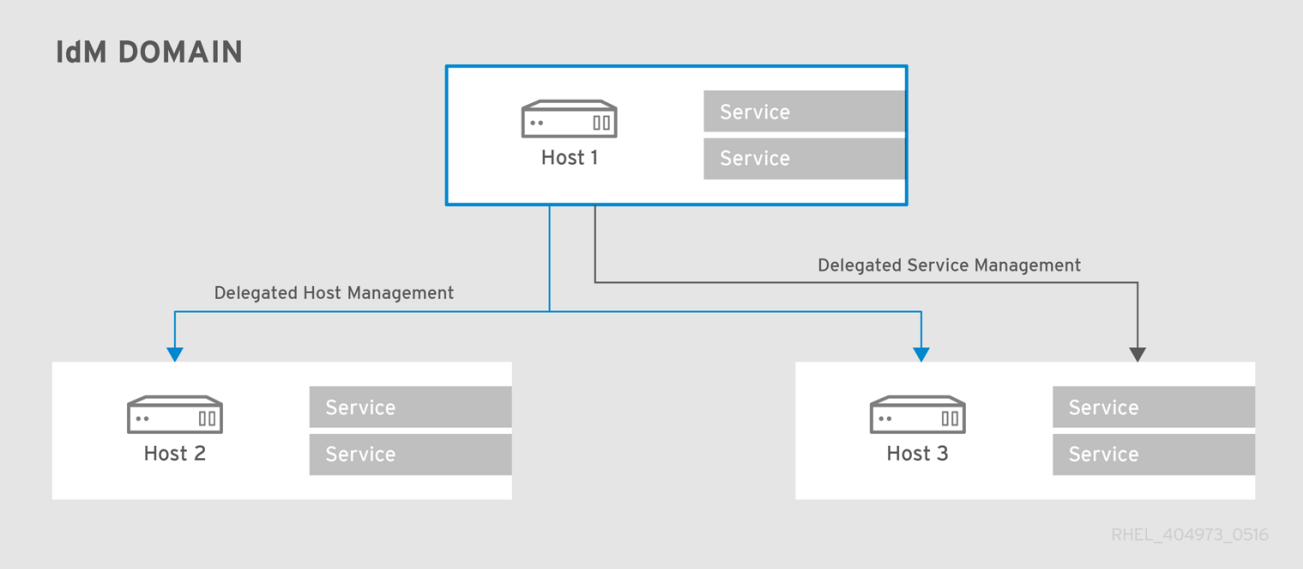
例如，请求新的 HTTP keytab：

```
[root@ipaserver ~]# ipa-getkeytab -s ipaserver.example.com -p HTTP/server.example.com -k
/etc/httpd/conf/krb5.keytab -e aes256-cts
```

要在本章上下文中 管理，意味着能够检索另一个主机或服务的 keytab 和证书。每个主机和服务都有一个 managedby 条目，它列出了哪些主机或服务可以管理它。默认情况下，主机可以管理自身及其所有服务。也可以通过更新适当的委托或提供合适的 managedby 条目来允许主机管理其他主机或服务。

只要授予该主机或 委派 了访问该服务的权限，就可以从任何 IdM 主机管理 IdM 服务。类似地， 可以将主机的权限委派给域中的其他主机。

图 17.1. 主机和服务委派



注意

如果主机通过 managedBy 条目委派给另一台主机，这并不表示主机也已被委托了对该主机上所有服务的管理。每个委托都必须独立执行。

17.1. 委派服务管理

主机使用 service-add-host 工具委派了对服务的控制：

```
# ipa service-add-host principal --hosts=hostname
```

委派该服务有两个部分：

- 使用 principal 参数指定主体。

- 使用 --hosts 选项识别具有控件的主机。

例如：

```
[root@server ~]# ipa service-add HTTP/web.example.com
[root@server ~]# ipa service-add-host HTTP/web.example.com --hosts=client1.example.com
```

主机被委派后，主机主体可用于管理服务：

```
[root@client1 ~]# kinit -kt /etc/krb5.keytab host/client1.example.com
[root@client1 ~]# ipa-getkeytab -s server.example.com -k /tmp/test.keytab -p
HTTP/web.example.com
Keytab successfully retrieved and stored in: /tmp/test.keytab
```

要为此服务创建票据，请使用委派机构在主机上创建一个证书请求：

```
[root@client1]# kinit -kt /etc/krb5.keytab host/client1.example.com
[root@client1]# openssl req -newkey rsa:2048 -subj '/CN=web.example.com/O=EXAMPLE.COM' -
keyout /etc/pki/tls/web.key -out /tmp/web.csr -nodes
Generating a 2048 bit RSA private key
.....+++
.....+++
Writing new private key to '/etc/pki/tls/private/web.key'
```

使用 cert-request 工具创建服务条目并加载认证信息：

```
[root@client1]# ipa cert-request --principal=HTTP/web.example.com web.csr
Certificate: MIICETCCAXqgA...[snip]
Subject: CN=web.example.com,O=EXAMPLE.COM
Issuer: CN=EXAMPLE.COM Certificate Authority
Not Before: Tue Feb 08 18:51:51 2011 UTC
Not After: Mon Feb 08 18:51:51 2016 UTC
Serial number: 1005
```

有关创建证书请求和使用 ipa cert-request 的更多信息，请参阅 第 24.1.1 节 “为用户、主机或服务请求新证书”。

17.2. 委派主机管理

主机通过 host-add-managedby 实用程序在其他主机上委派授权。这会创建一个 managedby 条目。创建 managedby 条目后，主机就可以检索它委托颁发机构的主机的 keytab。

1. 以 admin 用户身份登录。

```
[root@server ~]# kinit admin
```

2. 添加 managedby 条目。例如，这会将 客户端 2 的授权委派 给 client1。

```
[root@server ~]# ipa host-add-managedby client2.example.com --
hosts=client1.example.com
```

3. 获取票据作为主机 client1 ：

```
[root@client1 ~]# kinit -kt /etc/krb5.keytab host/client1.example.com
```

4. 为 client2 检索 keytab：

```
[root@client1 ~]# ipa-getkeytab -s server.example.com -k /tmp/client2.keytab -p
host/client2.example.com
Keytab successfully retrieved and stored in: /tmp/client2.keytab
```

17.3. 在 WEB UI 中委派主机或服务管理

IdM Web UI 中的每个主机和服务条目都有一个配置选项卡，用于指示哪些主机被委派对该主机或服务进行管理控制。

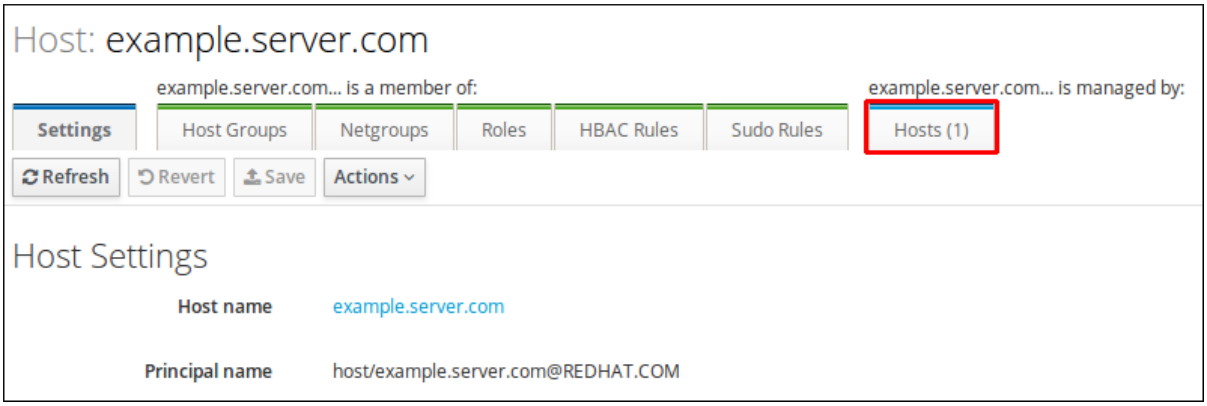
1. 打开 Identity 选项卡，然后选择 Hosts 或 Services 子选项卡。

2. 单击 您要向委派管理授予的主机或服务的名称。

3.

单击主机或服务条目右侧的 Hosts 子选项卡。这是列出 可以管理 所选主机或服务的主机的选项卡。

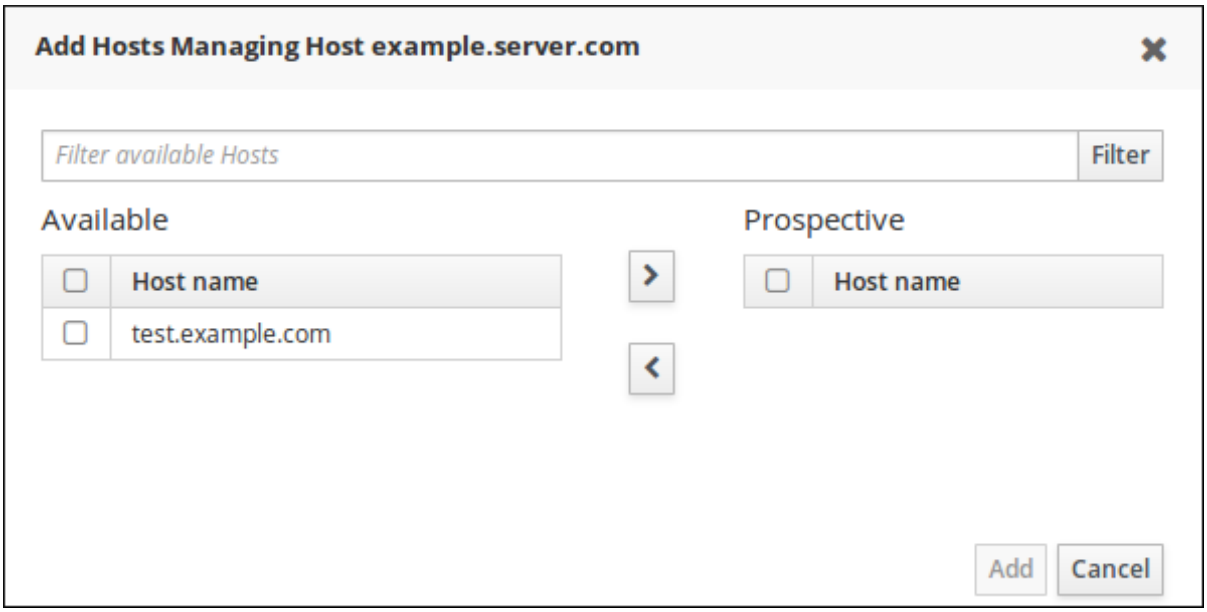
图 17.2. 主机子选项卡



4. 单击列表顶部的 Add 链接。

5. 单击要将主机或服务进行管理的主机的名称。点击右箭头按钮 & gt;，将主机移到选择框。

图 17.3. 主机/服务委派管理



6. 单击 Add 按钮，以关闭选择框并保存委派设置。

17.4. 访问委派的服务

对于服务和主机，如果客户端已委派授权，它可以在本地计算机上获取该主体的 keytab。对于 services，这的格式为 service/hostname@REALM。对于主机，服务是 主机。

使用 kinit 时，使用 -k 选项来加载 keytab，并使用 -t 选项指定 keytab。例如：

访问主机：

```
[root@server ~]# kinit -kt /etc/krb5.keytab host/ipa.example.com@EXAMPLE.COM
```

访问服务：

```
[root@server ~]# kinit -kt /etc/httpd/conf/krb5.keytab HTTP/ipa.example.com@EXAMPLE.COM
```

第 18 章 ID 视图

通过 ID 视图，您可以为 POSIX 用户或组属性指定新值，并定义要应用新值的客户端主机或主机。

例如，您可以使用 ID 视图来：

- 为不同环境定义不同的属性值；请参阅 第 18.3 节 “在不同主机上为用户帐户定义不同的属性值”
- 将之前生成的属性值替换为不同的值

重要

您只能将 ID 视图应用到 IdM 客户端，而不应用到 IdM 服务器。

对 SSSD 性能的潜在影响

应用 ID 视图可能会对 SSSD 性能造成负面影响，因为某些优化和 ID 视图不能同时运行。例如，ID 视图会防止 SSSD 优化在服务器上查找组的过程：

- 使用 ID 视图时，如果组名称已被覆盖，SSSD 必须检查返回的组成员名称列表中的每个成员。
- 如果没有 ID 视图，SSSD 只能从组对象的成员属性收集用户名。

当 SSSD 缓存为空或清除缓存后，这会使所有条目都无效，这主要会显示此负面影响。

其它资源

在涉及 Active Directory 的环境中，ID 视图也有多种用例。详情请查看《 Windows 集成指南 》中的"从同步到信任迁移 "一章。

18.1. ID 查看可以覆盖的属性

ID 视图由用户和组 ID 覆盖组成。覆盖定义新属性值。

用户和组 ID 覆盖可定义以下属性的新值：

用户属性

- *登录名(uid)*
- *GECOS 条目(gecos)*
- *UID 号(uidNumber)*
- *GID 号(gidNumber)*
- *登录 shell(loginShell)*
- *主目录（homeDirectory）*
- *SSH 公钥(ipaSshPubkey)*
- *证书(userCertificate)*

组属性

- *组名(cn)*
- *组 GID 号(gidNumber)*

18.2. 获取 ID VIEW 命令的帮助

要显示用于管理 ID 视图和覆盖的所有命令：

```
$ ipa help idviews
```

要显示特定命令的详细帮助信息，请在命令中添加 **--help** 选项：

```
$ ipa idview-add --help
```

18.3. 在不同主机上为用户帐户定义不同的属性值

管理员可以创建多个 ID 视图来覆盖用户帐户使用的属性值，并将这些 ID 视图应用到不同的客户端主机。Example:服务帐户配置为在不同主机上进行身份验证时使用不同的 SSH 公钥。

本节包含以下步骤：

- 第 18.3.1 节 “Web UI：覆盖特定主机的属性值”
- 第 18.3.2 节 “命令行：覆盖特定主机的属性值”

该流程演示了如何为名为 **host1.example.com** 的客户端主机创建 ID 视图。若要覆盖其他主机上的属性值，可使用流程创建多个 ID 视图，每个主机对应一个。

在以下步骤中：

- **user** 是需要覆盖其属性的用户帐户
- **host1.example.com** 是应用 ID 视图的主机

重要

创建新 ID 视图后，在应用 ID 视图的所有客户端中重启 SSSD。

如果新 ID 视图更改了 UID 或 GID，则也会清除这些客户端上的 SSSD 缓存。

18.3.1. Web UI：覆盖特定主机的属性值

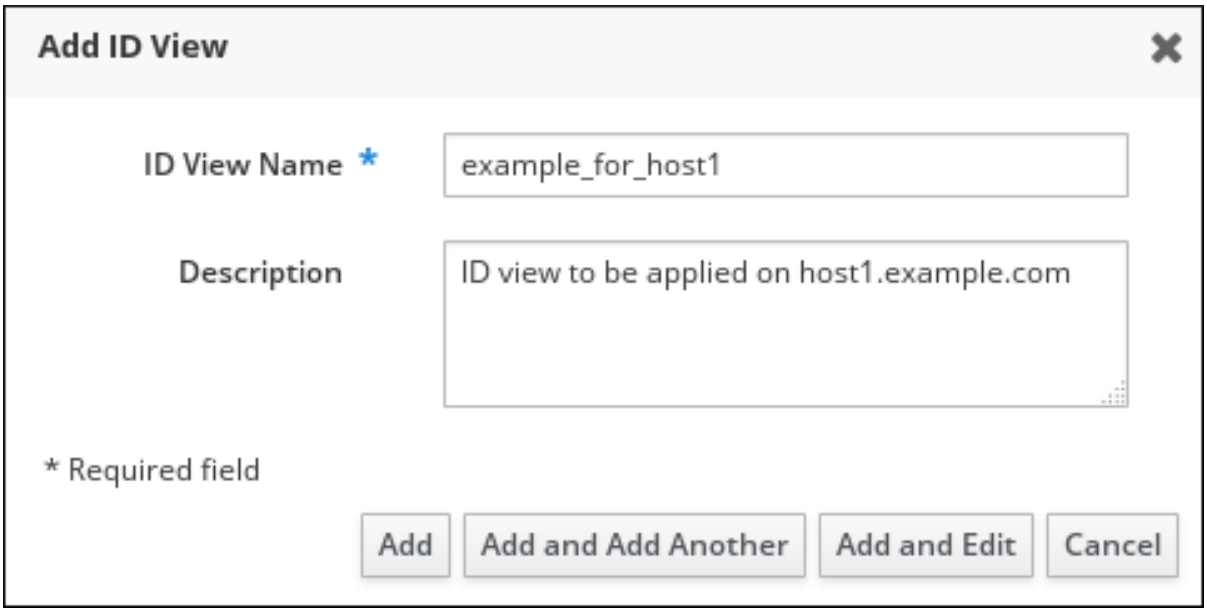
要管理 ID 视图，请首先以 IdM 管理员身份登录 IdM Web UI。

创建新 ID 视图

1. 在 Identity 选项卡下，选择 ID 视图 子选项卡。

2. 点 Add 并为 ID 视图提供一个名称。

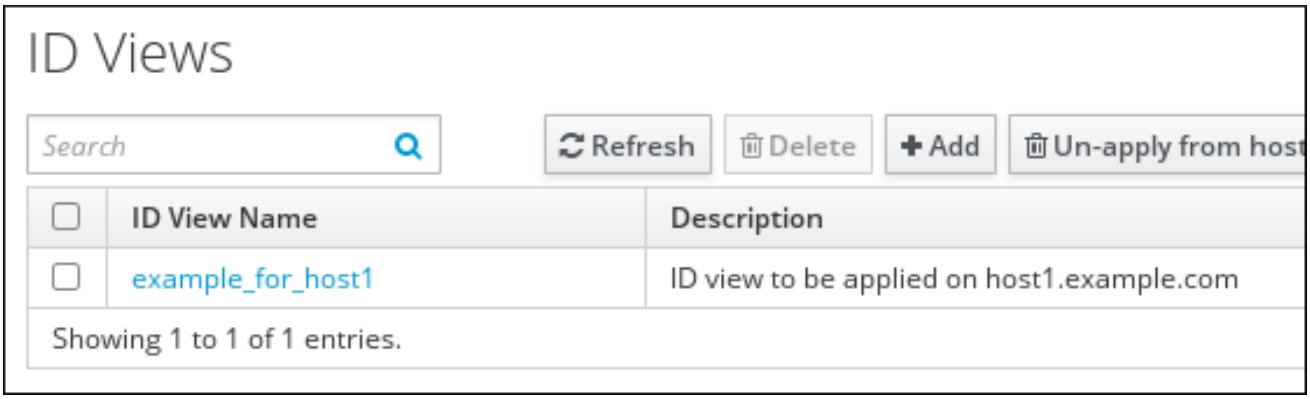
图 18.1. 添加 ID 视图



3. 单击 Add 确认。

新的 ID 视图现在显示在 ID 视图列表中。

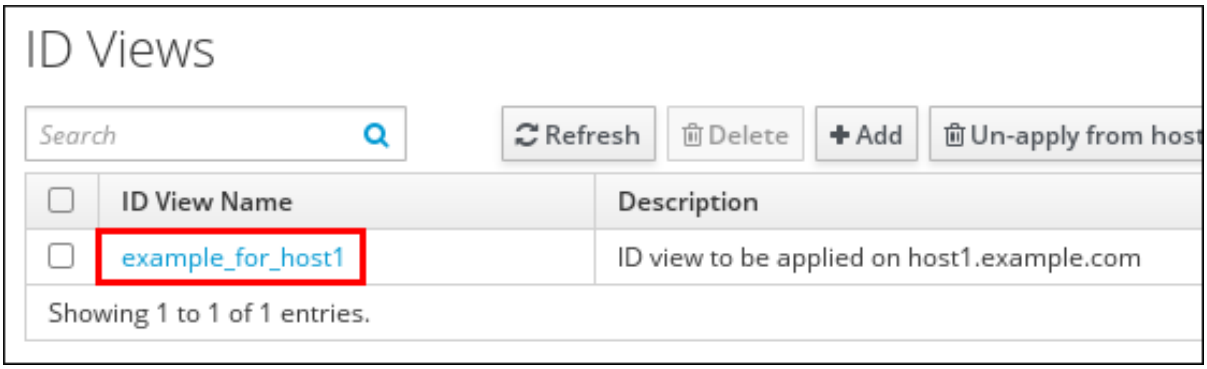
图 18.2. ID 视图列表



添加用户覆盖到 ID 视图

1. **在 ID 视图列表中，单击 ID 视图的名称。**

图 18.3. 编辑 ID 视图

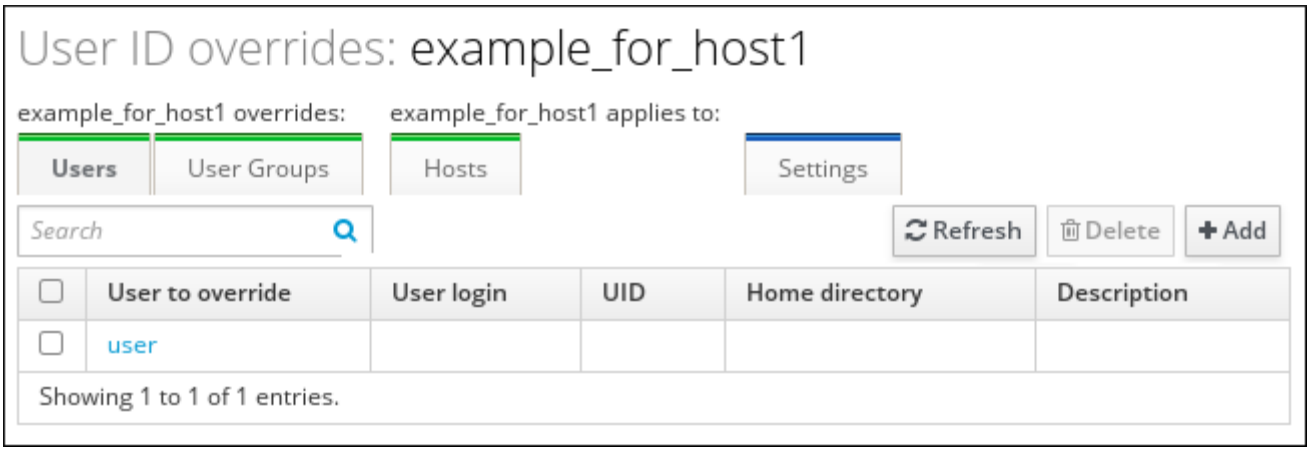


2. **在 Users 选项卡下，点 Add 添加用户覆盖。**

3. **选择要覆盖其属性值的用户帐户，然后单击 Add。**

用户覆盖现在显示在 example_for_host1 ID 视图页中。

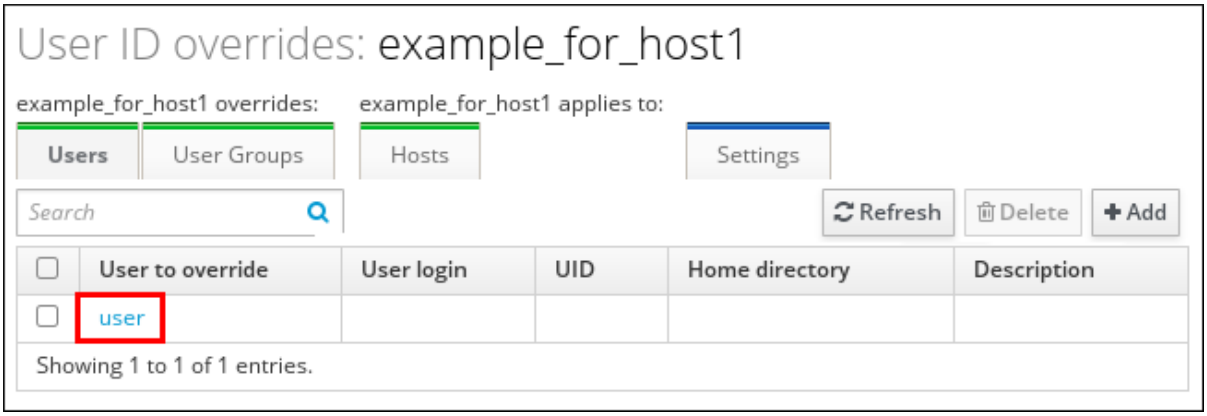
图 18.4. 覆盖列表



指定覆盖的属性

1. 单击您要用来更改属性值的覆盖。

图 18.5. 编辑覆盖

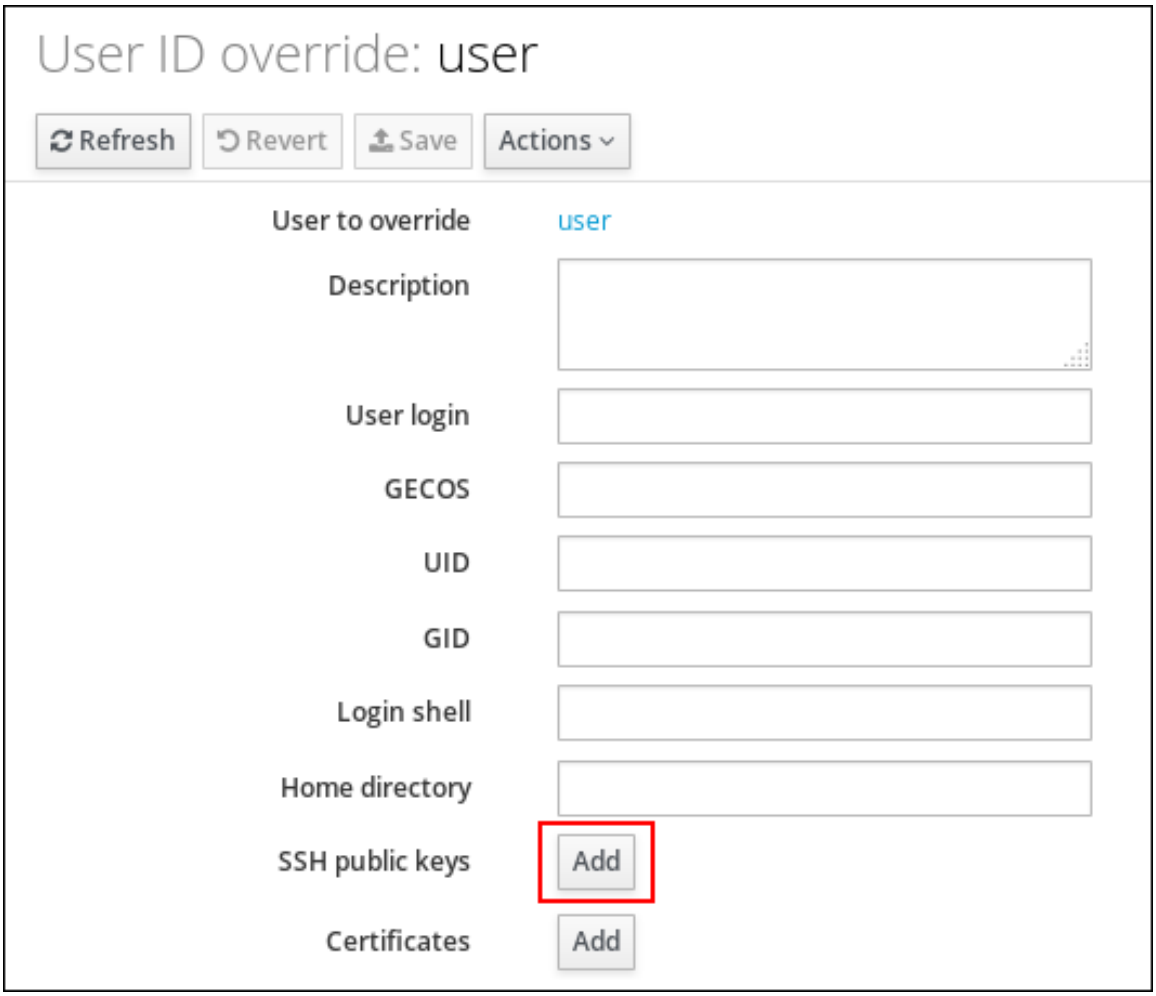


2. 定义 属性的新值。

   例如，覆盖用户帐户使用的 SSH 公钥：

   a. 点 SSH 公钥：添加。

图 18.6. 添加 SSH 公钥



b. *粘贴到公钥中。*

注意

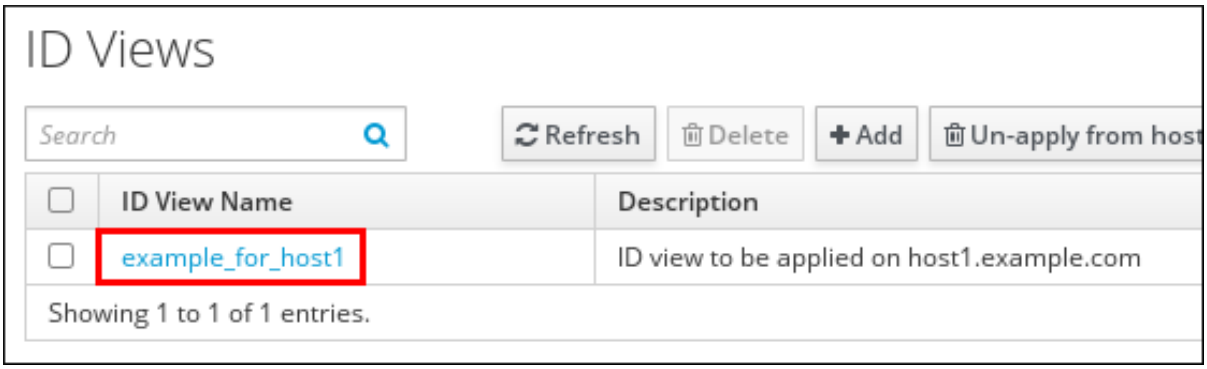
有关将 SSH 密钥添加到 IdM 的详情，请参考 第 22.5 节 “管理用户的公共 SSH 密钥”。

3. *点 Save 以更新覆盖。*

将 ID 视图应用到特定主机

1. *在 ID 视图列表中，单击 ID 视图的名称。*

图 18.7. 编辑 ID 视图



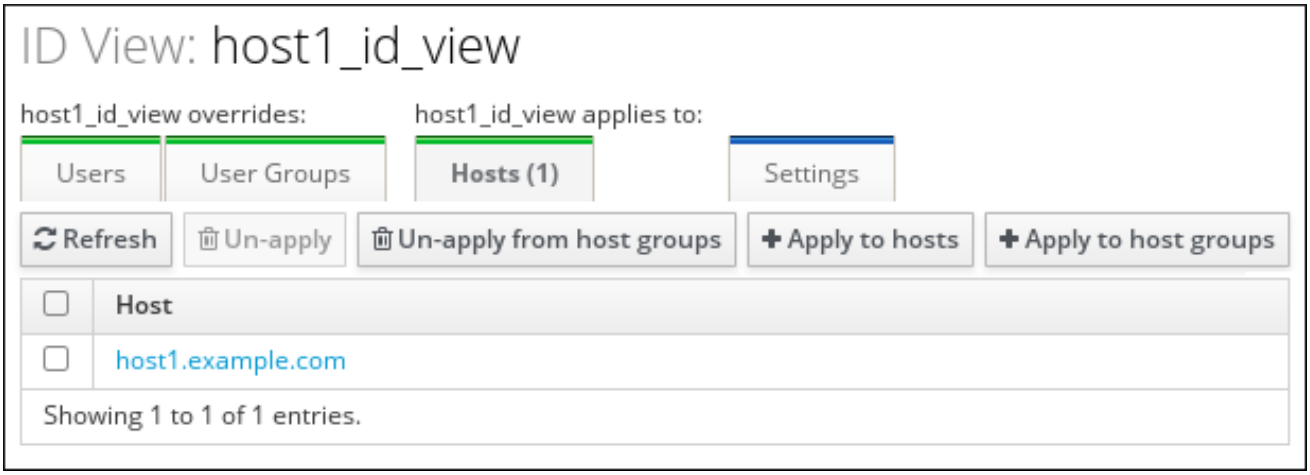
2. *在 Hosts 选项卡下，单击 Apply to hosts。*

3. *选择 host1.example.com 主机，并将它移到 Prospective 列中。*

4. *点应用。*

该主机现在显示在 ID 视图应用到的主机列表中。

图 18.8. 列出主机以哪个 ID 查看应用



18.3.2. 命令行：覆盖特定主机的属性值

在管理 ID 视图前，请以 IdM 管理员身份请求票据。例如：

```
$ kinit admin
```

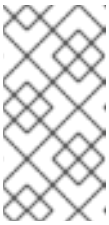
1. *创建新的 ID 视图。例如，创建一个名为 example_for_host1 的 ID 视图：*

```
$ ipa idview-add example_for_host1
---------------------------
Added ID View "example_for_host1"
---------------------------
  ID View Name: example_for_host1
```

2. 将用户覆盖添加到 example_for_host1 ID 视图。ipa idoverrideuser-add 命令需要 ID 视图的名称和用户覆盖。

- 若要指定新属性值，也可添加对应的命令行选项。如需可用选项的列表，请运行 ipa idoverrideuser-add --help。例如，使用 --sshpubkey 选项覆盖 SSH 公钥值：

```
$ ipa idoverrideuser-add example_for_host1 user --sshpubkey="ssh-rsa
AAAAB3NzaC1yrRqFE...gWRL71/miPIZ user@example.com"
----------------------------
Added User ID override "user"
----------------------------
  Anchor to override: user
  SSH public key: ssh-rsa
                  AAAB3NzaC1yrRqFE...gWRL71/miPIZ
    user@example.com
```



注意

有关将 SSH 密钥添加到 IdM 的详情，请参考 第 22.5 节 “管理用户的公共 SSH 密钥”。

- ipa idoverrideuser-add --certificate 命令替换指定 ID 视图中帐户的所有现有证书。要附加额外的证书，请使用 ipa idoverrideuser-add-cert 命令：

```
$ ipa idoverrideuser-add-cert example_for_host1 user --certificate="MIIEATCC..."
```

- 使用 ipa idoverrideuser-mod 命令，您还可以为现有用户覆盖指定新的属性值。

- 使用 ipa idoverrideuser-del 命令删除用户覆盖。

注意

如果您使用此命令删除 SSH 密钥覆盖，则不会立即从缓存中删除 SSH 密钥。使用默认缓存超时值(entry_cache_timeout = 5400)时，密钥会在缓存中保留 1 小时和半小时。

3. 将 example_for_host1 应用到 host1.example.com 主机：

```
$ ipa idview-apply example_for_host1 --hosts=host1.example.com
-----------------------------
Applied ID View "example_for_host1"
-----------------------------
hosts: host1.example.com
---------------------------------------------
Number of hosts the ID View was applied to: 1
---------------------------------------------
```

注意

ipa idview-apply 命令也接受 --hostgroups 选项。选项将 ID 视图应用到属于指定主机组的主机，但不会将 ID 视图与主机组本身相关联。相反，--hostgroups 选项会展开指定主机组的成员，并将 --hosts 选项分别应用到其中的每一个成员。

第 19 章 为 IDM 用户定义访问控制

访问控制是一组安全功能，用于定义谁可以访问某些资源，如机器、服务或条目等，以及它们允许执行的操作类型。身份管理提供了多个访问控制区域，以便明确授予哪些访问类型以及授予谁。因此，身份管理区分了对域中资源的访问控制和对 IdM 配置本身的访问控制。

有关 IdM 中用户对 IdM 服务器和其他 IdM 用户可用的不同内部访问控制机制的详情，请参考 第 10 章 为 IdM 用户定义访问控制。

第 20 章 管理 KERBEROS 标记和主要别名

20.1. 服务和主机的 KERBEROS 标记

您可以使用各种 Kerberos 标记来定义 Kerberos ticket 行为的特定方面。您可以将这些标记添加到服务和托管 Kerberos 主体中。

Identity Management(IdM)中的主体接受以下 Kerberos 标记：

OK_AS_DELEGATE

使用此标志指定为委派而受信任的 Kerberos 票据。

Active directory (AD)客户端检查 Kerberos 票据上的 OK_AS_DELEGATE 标志，以确定用户凭据是否可以转发或委派给特定的服务器。AD 将票据授予票据(TGT)仅转发到设置了 OK_AS_DELEGATE 的服务或主机。使用此标志，系统安全服务守护进程(SSSD)可以将 AD 用户 TGT 添加到 IdM 客户端机器的默认 Kerberos 凭证缓存中。

REQUIRES_PRE_AUTH

使用此标志指定只允许预先验证的票据与主体进行身份验证。

设置 REQUIRES_PRE_AUTH 标志后，密钥分发中心(KDC)需要额外的身份验证：KDC 仅针对 REQUIRES_PRE_AUTH 主体发出 TGT。

您可以清除 REQUIRES_PRE_AUTH 来禁用所选服务或主机的预身份验证，这样可降低 KDC 的负载，但也会稍微增加对长期密钥进行强度攻击的可能性。

OK_TO_AUTH_AS_DELEGATE

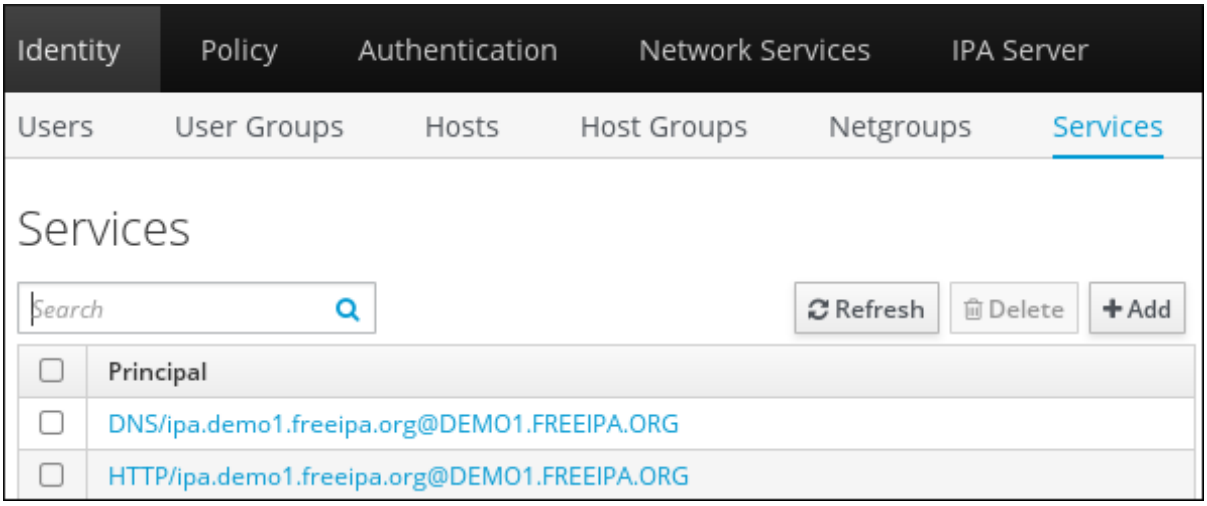
使用 OK_TO_AUTH_AS_AS_DELEGATE 标志来指定允许该服务代表用户获取 kerberos 票据。请注意，虽然这足以执行协议转换，以便代表用户获取其他票据，但服务需要 OK_AS_DELEGATE 标志以及密钥分发中心上允许的对应策略决定。

20.1.1. 从 Web UI 设置 Kerberos 标记

要将 OK_AS_DELEGATE、REQUIRES_PRE_AUTH 或 OK_TO_AUTH_AS_AS_DELEGATE 添加到主体：

1. *选择 **Services** 子选项卡，可通过 **Identity main** 选项卡访问。*

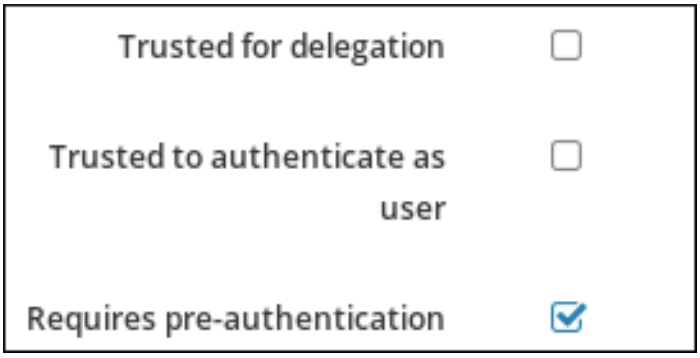
图 20.1. 服务列表



2. *点击您要添加标记的服务。*

3. *检查您要设置的选项。例如，要设置 **REQUIRES_PRE_AUTH** 标志，请检查 **Requires pre-authentication** 选项：*

*图 20.2. 添加 **REQUIRES_PRE_AUTH** 标志*



*下表列出了 **Web UI** 中的 **Kerberos** 标记的名称和对应名称：*

*表 20.1. **Web UI** 中的 **Kerberos** 标记的映射*

Kerberos 标志名称	Web UI 选项
OK_AS_DELEGATE	**受委托的信任**
REQUIRES_PRE_AUTH	**需要预身份验证**

Kerberos 标志名称	Web UI 选项
OK_TO_AUTH_AS_DELEGATE	受信任以用户身份进行身份验证

20.1.2. 从命令行设置和删除 Kerberos 标记

要从命令行或删除标志在主体中添加标志，请在 **ipa service-mod** 命令中添加以下选项之一：

- **--ok-as-delegate for OK_AS_DELEGATE**
- **--requires-pre-auth for REQUIRES_PRE_AUTH**
- **--ok-to-auth-as-delegate for OK_TO_AUTH_AS_DELEGATE**

要添加标志，请将对应的选项设置为 **1**。例如，要将 **OK_AS_DELEGATE** 标志添加到 **service/ipa.example.com@EXAMPLE.COM** 主体：

```
$ ipa service-mod service/ipa.example.com@EXAMPLE.COM --ok-as-delegate=1
```

要删除标志或禁用标志，请将对应的选项设置为 **0**。例如，要为 **test/ipa.example.com@EXAMPLE.COM** 主体禁用 **REQUIRES_PRE_AUTH** 标志：

```
$ ipa service-mod test/ipa.example.com@EXAMPLE.COM --requires-pre-auth=0
```

20.1.3. 从命令行显示 Kerberos 标记

要找出当前是否为主体设置了 **OK_AS_DELEGATE** ：

1. 运行 **kvno** 工具。
2. 运行 **klist -f** 命令。

OK_AS_DELEGATE 由 **klist -f** 输出中的 **O** 字符表示：

```
$ kvno test/ipa.example.com@EXAMPLE.COM
$ klist -f
Ticket cache: KEYRING:persistent:0:0
Default principal: admin@EXAMPLE.COM

Valid starting  Expires   Service principal
02/19/2014 09:59:02 02/20/2014 08:21:33 test/ipa/example.com@EXAMPLE.COM
    Flags: FATO
```

表 20.2. kerberos 标记的缩写

Kerberos 标志名称	缩写
OK_AS_DELEGATE	O
REQUIRES_PRE_AUTH	一个
OK_TO_AUTH_AS_DELEGATE	F

要找出当前为主体设置了哪些标记，请使用 **kadmin.local** 工具。当前标志显示在 **kadmin.local** 输出的 **Attributes** 行中，例如：

```
# kadmin.local
kadmin.local: getprinc test/ipa.example.com
Principal: test/ipa.example.com@EXAMPLE.COM
Expiration date: [never]
...
Attributes: REQUIRES_PRE_AUTH OK_AS_DELEGATE OK_TO_AUTH_AS_DELEGATE
Policy: [none]
```

20.2. 管理用户、主机和服务的 KERBEROS 主要别名

当您创建新用户、主机或服务时，会自动添加采用以下格式的 **Kerberos** 主体：

- ***user_name@REALM***
- ***host/host_name@REALM***
- ***service_name/host_name@REALM***

在某些情况下，管理员最好使用户、主机或服务能够使用别名与 **Kerberos** 应用程序进行身份验证，

例如：

- 用户名已更改，但该用户应当能使用之前和新用户名进行登录。
- 即使 IdM Kerberos 域与电子邮件域不同，用户也需要使用电子邮件地址登录。

请注意，如果您重命名了一个用户，对象会保留别名和之前的规范主名称。

20.2.1. Kerberos 主要别名

添加 Kerberos 主体别名

要将别名名称 **useralias** 添加到帐户 用户，请输入：

```
[root@ipaserver ~]# ipa user-add-principal user useralias
--------------------------------
Added new aliases to user "user"
--------------------------------
        User login: user
   Principal alias: user@IDM.EXAMPLE.COM, useralias@IDM.EXAMPLE.COM
```

要为主机或服务添加别名，请分别使用 **ipa host-add-principal** 或 **ipa service-add-principal** 命令。

如果您使用别名名称进行验证，请将 **-C** 选项传给 **kinit** 命令：

```
[root@ipaserver ~]# kinit -C useralias
Password for user@IDM.EXAMPLE.COM:
```

删除 Kerberos 主体别名

要从帐户用户中删除别名 **user alias**，请输入：

```
[root@ipaserver ~]# ipa user-remove-principal user useralias
--------------------------------
Removed aliases from user "user"
--------------------------------
  User login: user
  Principal alias: user@IDM.EXAMPLE.COM
```

要从主机或服务中删除别名，请分别使用 **ipa host-remove-principal** 或 **ipa service-remove-**

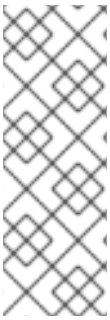
principal 命令。

请注意，您无法删除规范的主名称：

```
[root@ipaserver ~]# ipa user-show user
  User login: user
  ...
  Principal name: user@IDM.EXAMPLE.COM
  ...

[root@ipaserver ~]# ipa user-remove-principal user user
ipa: ERROR: invalid 'krbprincipalname': at least one value equal to the canonical principal name must be present
```

20.2.2. Kerberos Enterprise Principal Alias

企业级别名可以使用任何域后缀，但用户主体名称(UPN)后缀、NetBIOS 名称或可信 Active Directory 林域的域名除外。



注意

在添加或删除企业主体别名时，请使用两个反斜杠(\\)转义 @ 符号。否则，shell 会将 @ 符号解析为 Kerberos 域名称的一部分，并导致以下错误：

```
ipa: ERROR: The realm for the principal does not match the realm for this IPA server
```

添加 Kerberos 企业主要别名

将企业主体别名 user@example.com 添加到 用户帐户 中：

```
[root@ipaserver ~]# ipa user-add-principal user user\\@example.com
--------------------------------
Added new aliases to user "user"
--------------------------------
        User login: user
    Principal alias: user@IDM.EXAMPLE.COM, user\@example.com@IDM.EXAMPLE.COM
```

要为主机或服务添加企业别名，请分别使用 ipa host-add-principal 或 ipa service-add-principal 命令。

如果您使用企业主体名称进行验证，请将 -E 选项传给 kinit 命令：

```
[root@ipaserver ~]# kinit -E user@example.com
Password for user\@example.com@IDM.EXAMPLE.COM:
```

删除 Kerberos 企业主要别名

要从帐户 用户 中删除企业主体别名 user@example.com，请输入：

```
[root@ipaserver ~]# ipa user-remove-principal user user\\@example.com
--------------------------------
Removed aliases from user "user"
--------------------------------
  User login: user
  Principal alias: user@IDM.EXAMPLE.COM
```

要从主机或服务中删除别名，请分别使用 ipa host-remove-principal 或 ipa service-remove-principal 命令。

第 21 章 与 NIS 域和网络组集成

21.1. 关于 NIS 和身份管理

在 UNIX 环境中，网络信息服务(NIS)是集中管理身份和身份验证的常用方法。NIS，最初命名为 Yellow Pages (YP)，集中管理身份验证和身份信息，例如：

- 用户和密码
- 主机名和 IP 地址
- POSIX 组.

对于现代网络基础架构，NIS 被视为过于不安全，因为它既不提供主机身份验证，也不通过网络发送的数据进行加密。为了临时解决这个问题，NIS 通常与其他协议集成，以增强安全性。

如果您使用身份管理(IdM)，您可以使用 NIS 服务器插件连接无法完全迁移到 IdM 的客户端。IdM 将网络组和其他 NIS 数据集成到 IdM 域中。另外，您可以轻松地将用户和主机身份从 NIS 域迁移到 IdM。

身份管理中的 NIS

NIS 对象集成并存储在目录服务器后端中，以符合 RFC 2307。IdM 在 LDAP 目录中创建 NIS 对象，客户端使用加密的 LDAP 连接检索它们，例如：系统安全服务守护进程(SSSD)或 nss_ldap。

IdM 管理网络组、帐户、组、主机和其他数据。IdM 使用 NIS 侦听器将密码、组和网络组映射到 IdM 条目。

身份管理中的 NIS 插件

对于 NIS 支持，IdM 使用 slapi-nis 软件包提供的以下插件：

NIS 服务器插件

NIS 服务器插件使 IdM 集成的 LDAP 服务器能够充当客户端的 NIS 服务器。在此角色中，目录服务器会根据配置动态生成和更新 NIS 映射。使用插件，IdM 使用 NIS 协议作为 NIS 服务器提供客户端。

详情请查看 第 21.2 节 “在身份管理中启用 NIS”。

架构兼容性插件

Schema 兼容性插件使 Directory 服务器后端能够提供存储在目录信息树(DIT)中的条目的备用视图。这包括添加、丢弃或重命名属性值，以及从树中的多个条目检索属性值（可选）。

详情请查看 /usr/share/doc/slapi-nis-版本/sch-getting-started.txt 文件。

21.1.1. 身份管理中的 NIS Netgroups

NIS 实体可以存储在网络组中.与 UNIX 组相比， netgroups 支持：

- 嵌套组（组作为其他组的成员）。
- 对主机进行分组.

netgroup 定义以下信息集合：host、user 和 domain。这一集被称为 三者。这三个字段可以包含：

- 个值。
- 短划线(-)，指定 "no valid value"
- 无值.空字段指定通配符。

```
(host.example.com,,nisdomain.example.com)
(-,user,nisdomain.example.com)
```

当客户端请求 NIS netgroup 时， IdM 会转换 LDAP 条目：

- 并使用 NIS 插件通过 NIS 协议将其发送到客户端。

- 符合 RFC 2307 或 RFC 2307 bis 的 LDAP 格式。

21.1.1.1. 显示 NIS Netgroup 条目

IdM 将用户和组存储在 memberUser 属性中，以及 memberHost 中的主机和主机组。以下示例显示了 IdM 目录服务器组件中的 netgroup 条目：

例 21.1. 目录服务器中的 NIS 条目

```
dn: ipaUniqueID=d4453480-cc53-11dd-ad8b-0800200c9a66,cn=ng,cn=alt,...
...
cn: netgroup1
memberHost: fqdn=host1.example.com,cn=computers,cn=accounts,...
memberHost: cn=VirtGuests,cn=hostgroups,cn=accounts,...
memberUser: cn=demo,cn=users,cn=accounts,...
memberUser: cn=Engineering,cn=groups,cn=accounts,...
nisDomainName: nisdomain.example.com
```

在 IdM 中，您可以使用 ipa netgroup the 命令管理 netgroup 条目。例如，显示 netgroup 条目：

例 21.2. 显示 Netgroup Entry

```
[root@server ~]# ipa netgroup-show netgroup1
Netgroup name: netgroup1
Description: my netgroup
NIS domain name: nisdomain.example.com
Member Host: VirtGuests
Member Host: host1.example.com
Member User: demo
Member User: Engineering
```

21.2. 在身份管理中启用 NIS

在身份管理中启用 NIS：

1. 启用 NIS 侦听器和兼容性插件：

```
[root@ipaserver ~]# ipa-nis-manage enable
[root@ipaserver ~]# ipa-compat-manage enable
```

2. 可选 ：为 NIS 远程过程调用(RPC)设置固定端口。

在使用 NIS 时，客户端必须知道要使用的 IdM 服务器上的哪些端口来建立连接。使用默认设置，IdM 在服务器启动时绑定到未使用的随机端口。此端口发送到客户端用于请求端口号的端口映射器服务。

对于更严格的防火墙配置，您可以设置固定端口。例如，要将端口设置为 514 ：

```
[root@ipaserver ~]# ldapmodify -x -D 'cn=directory manager' -W
dn: cn=NIS Server,cn=plugins,cn=config
changetype: modify
add: nsslapd-pluginarg0
nsslapd-pluginarg0: 514
```

注意

您可以在 1024 以下设置任何未使用的端口号。

3. 启用并启动端口映射器服务：

```
[root@ipaserver ~]# systemctl enable rpcbind.service
[root@ipaserver ~]# systemctl start rpcbind.service
```

4. 重启 Directory 服务器：

```
[root@ipaserver ~]# systemctl restart dirsrv.target
```

21.3. 创建 NETGROUPS

21.3.1. 添加 Netgroup

要添加 Netgroup，您可以使用：

- IdM Web UI （请参阅 “Web UI：添加 Netgroup”一节）

- 命令行（请参见 “命令行：添加 Netgroup”一节）

Web UI：添加 Netgroup

1. 选择 **Identity** → **Groups** → **Netgroups**

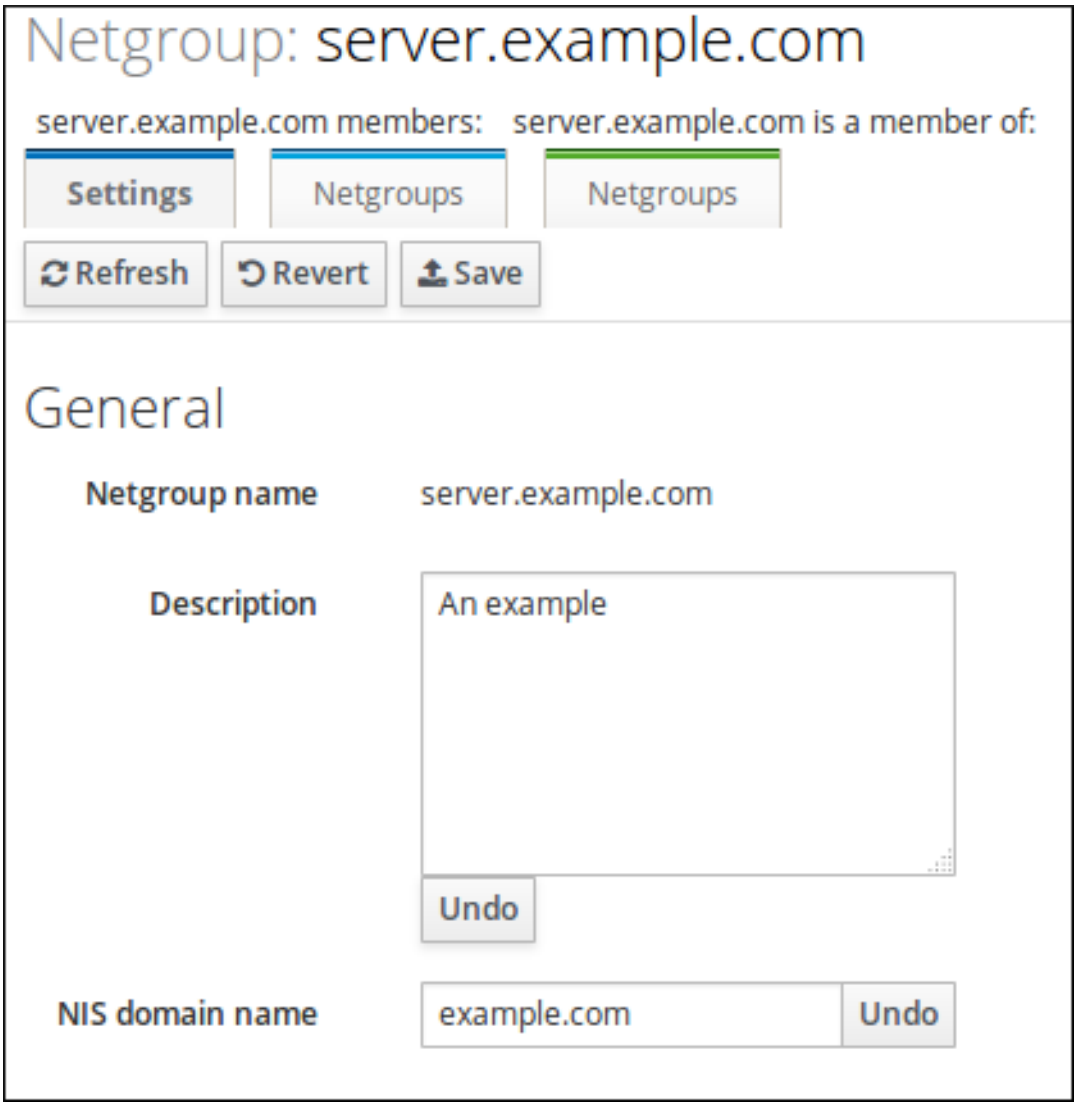
2. 点击 **Add**。

3. 输入唯一名称，也可选择性地输入描述。组名称是用于 IdM 域中的 netgroup 的标识符。以后您无法更改。

4. 点 **Add and Edit** 保存更改并开始编辑该条目。

5. 默认 **NIS** 域设置为 **IdM** 域名。另外，您可以在 **NIS** 域名字段中输入备用 **NIS** 域的名称。

图 21.1. netgroup 选项卡



NIS 域名 字段设置 netgroup triple 中显示的域。它不会影响身份管理 NIS 侦听程序响应的 NIS 域。

6. *添加成员，如 “Web UI：将成员添加到网络组中”一节 所述。*

7. *点击 Save。*

命令行：添加 Netgroup

您可以使用 ipa netgroup-add 命令添加新的 netgroup。指定：

- *组名称。*

- （可选）描述。

- （可选）如果 NIS 域名与 IdM 域名不同。

注意

nisdomain 选项设置 netgroup triple 中出现的域。它不会影响身份管理侦听器响应的 NIS 域。

例如：

```
[root@server ~]# ipa netgroup-add --desc="Netgroup description" --nisdomain="example.com"
example-netgroup
```

要将成员添加到 netgroup，请参阅 “命令行：将成员添加到网络组中”一节。

21.3.2. 将成员添加到网络组中

在用户和主机旁边，netgroups 可以包含用户组、主机组和其他网络组（嵌套组）作为其成员。根据组的大小，您为子组的成员创建嵌套组后最多可能需要几分钟，才能将 显示为父组的成员。

要将成员添加到 Netgroup，您可以使用：

- IdM Web UI （请参阅 “Web UI：将成员添加到网络组中”一节）

- 命令行（请参见 “命令行：将成员添加到网络组中”一节）

警告

不要创建递归嵌套组。例如，如果 GroupA 是 GroupB 的成员，请不要将 GroupB 添加为 GroupA 的成员。不支持递归组，并可能导致无法预测的行为。

Web UI：将成员添加到网络组中

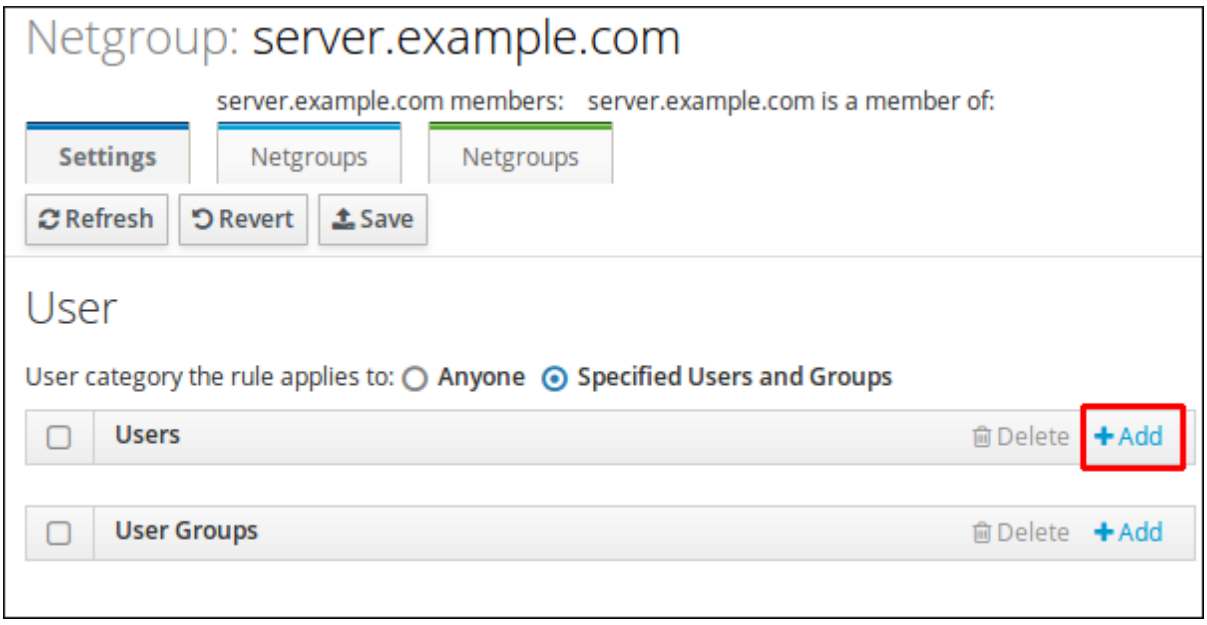
使用 Web UI 将成员添加到 netgroup 中：

1. *选择 **Identity** → **Groups** → **Netgroups***

2. *单击要向其添加成员的 **netgroup** 的名称。*

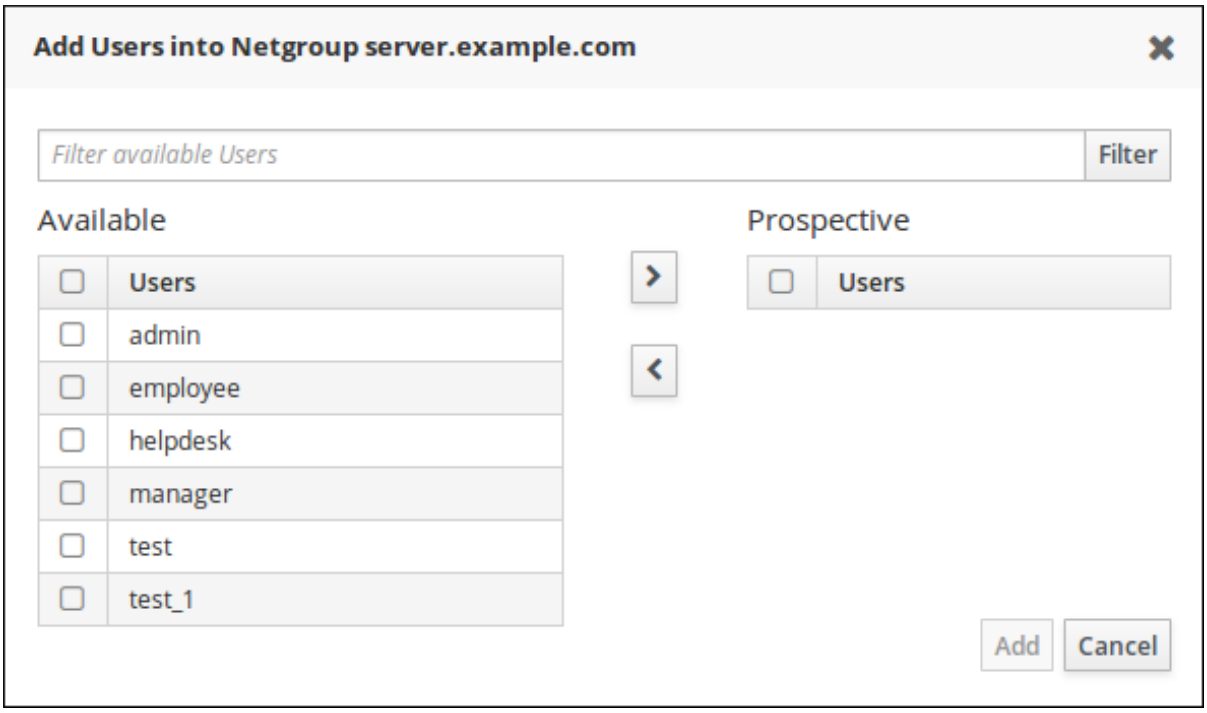
3. *单击所需成员类型旁边的 **Add**。*

图 21.2. Netgroup 选项卡中的 User Menu



4. *选择您要添加的成员，然后点击 **&gt；确认**。*

图 21.3. 在 Netgroup 选项卡中添加 User Menu



5. 点击 **Add**。

命令行：将成员添加到网络组中

创建 **netgroup** 后，您可以使用 **ipa netgroup-add-member** 命令添加成员：

```
# ipa netgroup-add-member --users=user_name --groups=group_name --hosts=host_name \
    --hostgroups=host_group_name --netgroups=netgroup_name group_nameame
```

要设置多个成员，请在一组大括号内使用逗号分隔的列表。例如：

```
[root@server ~]# ipa netgroup-add-member --users={user1;user2,user3} \
    --groups={group1,group2} example-group
```

21.4. 向 NIS 客户端公开自动挂载映射

如果已经定义了任何自动挂载映射，您必须手动将它们添加到 **IdM** 中的 **NIS** 配置中。这样可确保将映射公开给 **NIS** 客户端。

NIS 服务器由 **IdM LDAP** 目录中的特殊插件条目管理。**NIS** 服务器使用的每个 **NIS** 域和映射均作为该容器中的子条目添加。**NIS** 域条目包含：

- NIS 域的名称
- NIS 映射的名称
- 有关如何查找要用作 NIS 映射的内容的目录条目的信息
- 有关将哪些属性用作 NIS 映射的键和值的信息

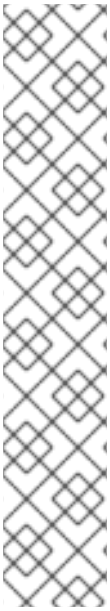
其中大多数设置对于每个映射都相同。

21.4.1. 添加自动挂载映射

IdM 将自动挂载映射（根据自动挂载位置分组）存储在 IdM 目录树的 cn=automount 分支中。您可以使用 LDAP 协议添加 NIS 域和映射。

例如，要在 example.com 域的默认位置中添加名为 auto.example 的自动挂载映射：

```
[root@server ~]# ldapadd -h server.example.com -x -D "cn=Directory Manager" -W

dn: nis-domain=example.com+nis-map=auto.example,cn=NIS Server,cn=plugins,cn=config
objectClass: extensibleObject
nis-domain: example.com
nis-map: auto.example
nis-filter: (objectclass=automount)
nis-key-format: %{automountKey}
nis-value-format: %{automountInformation}
nis-base: automountmapname=auto.example,cn=default,cn=automount,dc=example,dc=com
```



注意

将 nis-domain 属性设置为 NIS 域的名称。

nis-base 属性中设置的值必须对应：

- 到使用 ipa automountmap047 命令设置的现有自动挂载映射。
- 使用 ipa automountlocation the 命令设置的现有自动挂载位置。

设置条目后，您可以验证自动挂载映射：

```
[root@server ~]# ypcat -k -d example.com -h server.example.com auto.example
```

21.5. 从 NIS 迁移到 IDM

从现有 NIS 服务器迁移到身份管理(IdM)需要以下步骤：

1. 在身份管理中启用 NIS Listener
2. 从 NIS 导出并导入现有数据

21.5.1. 在 IdM 中准备 Netgroup 条目

在迁移前，请确定在当前 NIS 服务器中管理哪些身份：

用户条目

确定哪些应用程序正在使用 NIS 提供的用户信息.虽然某些工具（如 sudo ）需要 NIS 网络组，但有些实用程序可以使用常规 UNIX 组。

迁移：

1. 在 IdM 中创建对应的用户帐户。请参阅 第 21.5.3.1 节 “迁移用户条目”。

2. 如果额外需要 netgroups：

   a. 添加 netgroups。请参阅 第 21.3.1 节 “添加 Netgroup”。

   b. 将用户添加到 netgroups。请参阅 第 21.5.3.4 节 “迁移 Netgroup Entries”。

主机条目

当您在 IdM 中创建主机组时，会自动创建对应的 shadow NIS 组。不要在这些影子 NIS 组中使用 ipa netgroup-* 命令。仅使用 ipa netgroup 047 命令管理通过 netgroup-add 命令创建 的原生 网络组。

对于直接转换

如果每个用户和主机条目都必须使用相同的名称，您可以在 IdM 中使用相同名称创建条目：

1. 为 netgroup 中引用的每个用户创建一个条目。

2. 为 netgroup 中引用的每一主机创建一个条目。

3. 创建名称与原始 netgroup 的名称相同的 netgroup。

4. 将用户和主机添加为 netgroup 的直接成员。如果用户和主机是组或主机组的成员，您也可以将这些组添加到 netgroup。

21.5.2. 在身份管理中启用 NIS Listener

请参阅 第 21.2 节 “在身份管理中启用 NIS”。

21.5.3. 导出和导入现有 NIS 数据

NIS 服务器可以包含有关用户、组、主机、网络组和自动挂载映射的信息。您可以将这些条目类型迁移到 IdM。

在以下部分中，我们使用 ypcat 命令从当前 NIS 服务器导出数据，并使用输出来使用对应的 ipa114-add 命令将条目导入到 IdM。

- 确保安装 yp-tools 软件包，因为它提供了迁移脚本中使用的 ypcat 命令：

```
[root@nis-server ~]# yum install yp-tools -y
```

21.5.3.1. 迁移用户条目

NIS passwd 映射包含有关用户的信息，如名称、UID、主组、GECOS、shell 和主目录。使用此数据将 NIS 用户帐户迁移到 IdM:

1. 可选 ：如果您需要弱密码支持，请参阅 第 21.5.4 节 “为 NIS 用户身份验证启用弱密码哈希”。

2. 使用以下内容创建 /root/nis-users.sh 脚本：

```
#!/bin/sh
# $1 is the NIS domain, $2 is the NIS master server
ypcat -d $1 -h $2 passwd > /dev/shm/nis-map.passwd 2>&1

IFS=$'\n'
for line in $(cat /dev/shm/nis-map.passwd) ; do
 IFS=' '
 username=$(echo $line | cut -f1 -d:)
 # Not collecting encrypted password because we need cleartext password
 # to create kerberos key
 uid=$(echo $line | cut -f3 -d:)
 gid=$(echo $line | cut -f4 -d:)
 gecos=$(echo $line | cut -f5 -d:)
 homedir=$(echo $line | cut -f6 -d:)
 shell=$(echo $line | cut -f7 -d:)

 # Now create this entry
 echo passw0rd1 | ipa user-add $username --first=NIS --last=USER \
     --password --gidnumber=$gid --uid=$uid --gecos="$gecos" --homedir=$homedir \
     --shell=$shell
 ipa user-show $username
done
```

3. 以 IdM admin 用户身份进行身份验证：

```
[root@nis-server ~]# kinit admin
```

4. 运行脚本。例如：

```
[root@nis-server ~]# sh /root/nis-users.sh nisdomain nis-master.example.com
```

注意

此脚本将硬编码值用于名字、姓氏，并将密码设置为 passw0rd1。用户必须在下一次登录时更改临时密码。

21.5.3.2. 迁移组条目

NIS 组映射包含有关组的信息，如组名称、GID 或组成员。使用此数据将 NIS 组迁移到 IdM:

1. 使用以下内容创建 /root/nis-groups.sh 脚本：

```
#!/bin/sh
# $1 is the NIS domain, $2 is the NIS master server
ypcat -d $1 -h $2 group > /dev/shm/nis-map.group 2>&1

IFS=$'\n'
for line in $(cat /dev/shm/nis-map.group); do
 IFS=' '
 groupname=$(echo $line | cut -f1 -d:)
 # Not collecting encrypted password because we need cleartext password
 # to create kerberos key
 gid=$(echo $line | cut -f3 -d:)
 members=$(echo $line | cut -f4 -d:)

 # Now create this entry
 ipa group-add $groupname --desc=NIS_GROUP_$groupname --gid=$gid
 if [ -n "$members" ]; then
  ipa group-add-member $groupname --users={$members}
 fi
 ipa group-show $groupname
done
```

2. 以 IdM admin 用户身份进行身份验证：

```
[root@nis-server ~]# kinit admin
```

3. 运行脚本。例如：

```
[root@nis-server ~]# sh /root/nis-groups.sh nisdomain nis-master.example.com
```

21.5.3.3. 迁移主机条目

NIS 主机映射包含有关主机的信息，如主机名和 IP 地址。使用此数据将 NIS 主机条目迁移到 IdM：

1. 使用以下内容创建 /root/nis-hosts.sh 脚本：

```
#!/bin/sh
# $1 is the NIS domain, $2 is the NIS master server
ypcat -d $1 -h $2 hosts | egrep -v "localhost|127.0.0.1" > /dev/shm/nis-map.hosts 2>&1

IFS=$'\n'
for line in $(cat /dev/shm/nis-map.hosts); do
 IFS=' '
 ipaddress=$(echo $line | awk '{print $1}')
 hostname=$(echo $line | awk '{print $2}')
 master=$(ipa env xmlrpc_uri | tr -d '[:space:]' | cut -f3 -d: | cut -f3 -d/)
 domain=$(ipa env domain | tr -d '[:space:]' | cut -f2 -d:)
 if [ $(echo $hostname | grep "\." |wc -l) -eq 0 ] ; then
  hostname=$(echo $hostname.$domain)
 fi
 zone=$(echo $hostname | cut -f2- -d.)
 if [ $(ipa dnszone-show $zone 2>/dev/null | wc -l) -eq 0 ] ; then
  ipa dnszone-add --name-server=$master --admin-email=root.$master
 fi
 ptrzone=$(echo $ipaddress | awk -F. '{print $3 "." $2 "." $1 ".in-addr.arpa."}')
 if [ $(ipa dnszone-show $ptrzone 2>/dev/null | wc -l) -eq 0 ] ; then
  ipa dnszone-add  $ptrzone --name-server=$master --admin-email=root.$master
 fi
 # Now create this entry
 ipa host-add $hostname --ip-address=$ipaddress
 ipa host-show $hostname
done
```

2. 以 IdM admin 用户身份进行身份验证：

```
[root@nis-server ~]# kinit admin
```

3. 运行脚本。例如：

```
[root@nis-server ~]# sh /root/nis-hosts.sh nisdomain nis-master.example.com
```

注意

此脚本不会迁移特殊主机配置，如别名。

21.5.3.4. 迁移 Netgroup Entries

NIS netgroup 映射包含有关网络组的信息。使用此数据将 NIS 网络组迁移到 IdM：

1. 使用以下内容创建 /root/nis-netgroups.sh 脚本：

```
#!/bin/sh
# $1 is the NIS domain, $2 is the NIS master server
ypcat -k -d $1 -h $2 netgroup > /dev/shm/nis-map.netgroup 2>&1

IFS=$'\n'
for line in $(cat /dev/shm/nis-map.netgroup); do
 IFS=' '
 netgroupname=$(echo $line | awk '{print $1}')
 triples=$(echo $line | sed "s/^$netgroupname //")
 echo "ipa netgroup-add $netgroupname --desc=NIS_NG_$netgroupname"
 if [ $(echo $line | grep "(," | wc -l) -gt 0 ]; then
  echo "ipa netgroup-mod $netgroupname --hostcat=all"
 fi
 if [ $(echo $line | grep ",," | wc -l) -gt 0 ]; then
  echo "ipa netgroup-mod $netgroupname --usercat=all"
 fi

 for triple in $triples; do
  triple=$(echo $triple | sed -e 's/-//g' -e 's/(//' -e 's/)//')
  if [ $(echo $triple | grep ",.*," | wc -l) -gt 0 ]; then
   hostname=$(echo $triple | cut -f1 -d,)
   username=$(echo $triple | cut -f2 -d,)
   domain=$(echo $triple | cut -f3 -d,)
   hosts=""; users=""; doms="";
   [ -n "$hostname" ] && hosts="--hosts=$hostname"
   [ -n "$username" ] && users="--users=$username"
   [ -n "$domain"   ] && doms="--nisdomain=$domain"
   echo "ipa netgroup-add-member $netgroup $hosts $users $doms"
  else
   netgroup=$triple
   echo "ipa netgroup-add $netgroup --desc=NIS_NG_$netgroup"
  fi
 done
done
```

2. 以 IdM admin 用户身份进行身份验证：

```
[root@nis-server ~]# kinit admin
```

3. 运行脚本。例如：

```
[root@nis-server ~]# sh /root/nis-netgroups.sh nisdomain nis-master.example.com
```

21.5.3.5. 迁移自动挂载映射

自动挂载映射是一系列嵌套条目和宏条目，用于定义位置（父条目）、关联的键和映射。将 NIS 自动挂载映射迁移到 IdM：

1. 使用以下内容创建 /root/nis-automounts.sh 脚本：

```
#!/bin/sh
# $1 is for the automount entry in ipa

ipa automountlocation-add $1

# $2 is the NIS domain, $3 is the NIS master server, $4 is the map name
ypcat -k -d $2 -h $3 $4 > /dev/shm/nis-map.$4 2>&1

ipa automountmap-add $1 $4

basedn=$(ipa env basedn | tr -d '[:space:]' | cut -f2 -d:)
cat > /tmp/amap.ldif <<EOF
dn: nis-domain=$2+nis-map=$4,cn=NIS Server,cn=plugins,cn=config
objectClass: extensibleObject
nis-domain: $2
nis-map: $4
nis-base: automountmapname=$4,cn=$1,cn=automount,$basedn
nis-filter: (objectclass=*)
nis-key-format: %{automountKey}
nis-value-format: %{automountInformation}
EOF
ldapadd -x -h $3 -D "cn=Directory Manager" -W -f /tmp/amap.ldif

IFS=$'\n'
for line in $(cat /dev/shm/nis-map.$4); do
 IFS=" "
 key=$(echo "$line" | awk '{print $1}')
 info=$(echo "$line" | sed -e "s#^$key[ \t]*##")
 ipa automountkey-add nis $4 --key="$key" --info="$info"
done
```

脚本导出 NIS 自动挂载信息，为自动挂载位置和相关映射生成 LDAP 数据交换格式 (LDIF)，并将 LDIF 文件导入到 IdM 目录服务器。详情请查看 第 21.4 节 “向 NIS 客户端公开自动挂载映射”。

2. 以 IdM admin 用户身份进行身份验证：

```
[root@nis-server ~]# kinit admin
```

3. 运行脚本。例如：

```
[root@nis-server ~]# sh /root/nis-automounts.sh location nisdomain \
    nis-master.example.com map_name
```

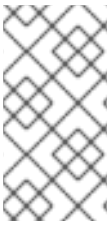
21.5.4. 为 NIS 用户身份验证启用弱密码哈希

使用 Directory Server 组件的默认设置，存储在 userPassword 属性中的密码将使用 salted 安全哈希算法(SSHA)进行哈希处理。如果您的 NIS 客户端需要弱的密码哈希算法，请更新密码存储方案设置。

启用弱密码散列方案只会影响存储在 userPassword 属性中的密码。请注意，Kerberos 不使用这个属性，因此 Kerberos 加密不受此设置的影响。

例如，启用 CRYPT 哈希密码：

```
[root@server ~]# ldapmodify -D "cn=Directory Manager" -W -p 389 -h ipaserver.example.com -x
dn: cn=config
changetype: modify
replace: passwordStorageScheme
passwordStorageScheme: crypt
```



注意

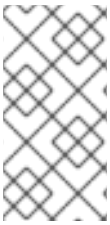
由于密码哈希无法解密，因此 Directory 服务器不会转换现有的密码哈希。服务器仅将新密码存储应用到更改存储方案后设置的密码。

部分 V. 管理：管理身份验证

这部分提供了有关如何设置和管理智能卡验证的指令。此外，它还涵盖了与证书相关的主题，如发布证书、配置基于证书的身份验证，以及控制 身份管理中的 证书的有效性。

第 22 章 用户身份验证

本章论述了管理用户身份验证机制，包括如何管理用户的密码、SSH 密钥和证书以及如何配置一次性密码(OTP)和智能卡验证的信息。



注意

有关如何使用 Kerberos 登录身份管理(IdM)的文档，请参考 第 5 章 管理 IdM 服务器和服务的基本知识。

22.1. 用户密码

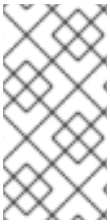
22.1.1. 更改和重置用户密码

没有更改其他用户密码的权限只能更改他们自己的个人密码。个人密码以这种方式更改：

- 必须满足 IdM 密码策略。有关配置密码策略的详情请参考 第 28 章 定义密码策略。

管理员和具有密码更改权限的用户可以为新用户设置初始密码，并为现有用户重置密码。密码以这种方式更改：

- 不必满足 IdM 密码策略
- 第一次成功登录后过期。发生这种情况时，IdM 会提示用户立即更改过期的密码。要禁用此行为，请参阅 第 22.1.2 节 “在下一个登录账户为密码更改启用密码重置”。



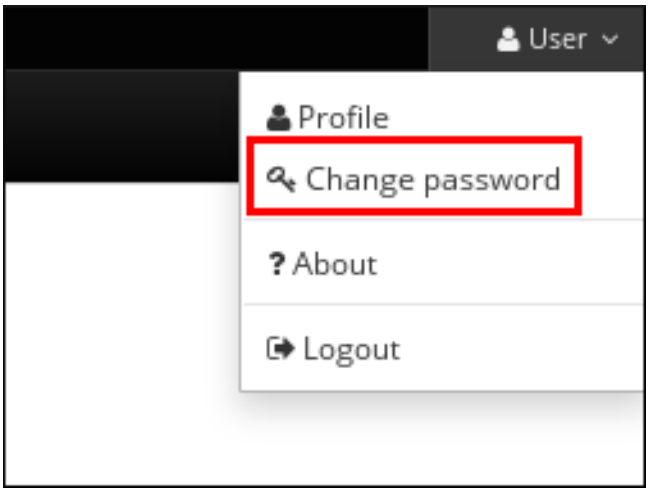
注意

LDAP 目录管理器(DM)用户可以使用 LDAP 工具更改用户密码。新密码可以覆盖任何 IdM 密码策略。DM 设置的密码在第一次登录后不会过期。

22.1.1.1. Web UI：更改您自己的个人密码

1. 在右上角，单击 User name → Change password。

图 22.1. 重置密码



2. *输入新密码。*

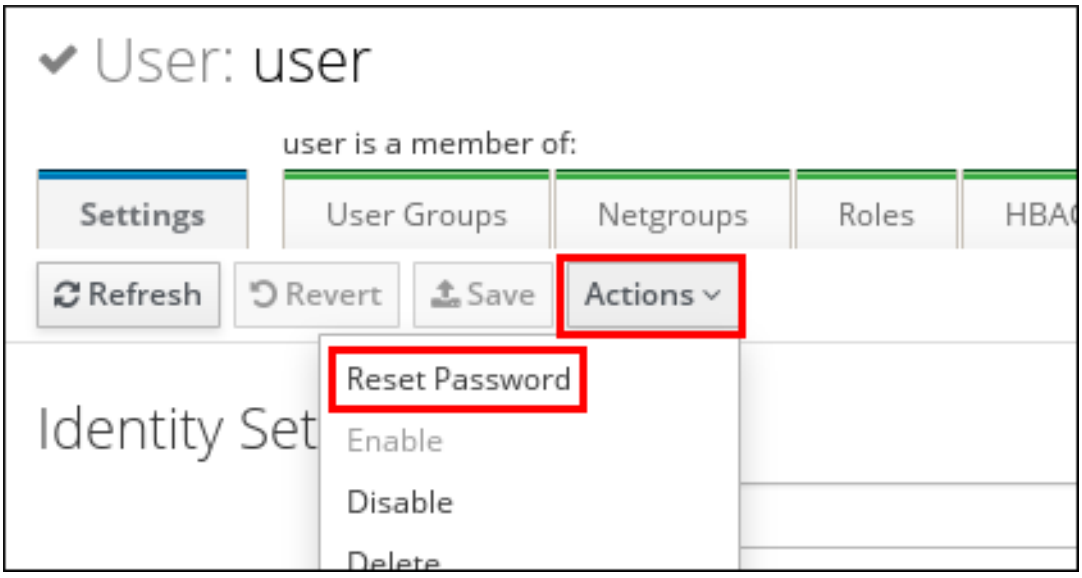
22.1.1.2. Web UI：重置另一个用户的密码

1. *选择 **Identity** → **Users**。*

2. *单击要编辑的用户的名称。*

3. *单击 **Actions** → **Reset password**。*

图 22.2. 重置密码



4. *输入新密码，然后单击 **Reset Password**。*

图 22.3. 确认新密码

22.1.1.3. 命令行：更改或重置其他用户的密码

要更改您自己的个人密码或更改或重置其他用户的密码，请在 **ipa user-mod** 命令中添加 **--password** 选项。命令将提示您输入新密码。

```
$ ipa user-mod user --password
Password:
Enter Password again to verify:
--------------------
Modified user "user"
--------------------
...
```

22.1.2. 在下一个登录账户为密码更改启用密码重置

默认情况下，当管理员重置另一个用户的密码时，密码会在第一次成功登录后过期。详情请查看 第 22.1.1 节 “更改和重置用户密码”。

为确保管理员在首次使用时设置的密码不会过期，请在域中的每个身份管理服务器上进行这些更改：

- 编辑密码同步条目： **cn=ipa_pwd_extop,cn=plugins,cn=config**。
- 在 **passSyncManagersDNs** 属性中指定管理用户帐户。属性是多值的。

例如，使用 **ldapmodify** 实用程序指定 **admin** 用户：

```
$ ldapmodify -x -D "cn=Directory Manager" -W -h ldap.example.com -p 389

dn: cn=ipa_pwd_extop,cn=plugins,cn=config
```

```
changetype: modify
add: passSyncManagersDNs
passSyncManagersDNs: uid=admin,cn=users,cn=accounts,dc=example,dc=com
```

警告

仅指定需要这些额外权限的用户。在 passSyncManagerDNs 下列出的所有用户都可以：

- 执行密码更改操作而无需随后的密码重置
- 绕过密码策略，以便不会应用强度或历史记录强制

22.1.3. 密码失败后解锁用户帐户

如果用户尝试多次使用错误的密码登录，IdM 将锁定用户帐户，这会阻止用户登录。请注意，IdM 不会显示用户帐户已被锁定的任何警告信息。

注意

有关设置允许失败的确切次数和锁定持续时间的详情，请参考 第 28 章 定义密码策略。

IdM 在经过指定的时间后自动解锁用户帐户。或者，管理员可以手动解锁用户帐户。

手动解锁用户帐户

要解锁用户帐户，请使用 **ipa user-unlock** 命令。

```
$ ipa user-unlock user
----------------------
Unlocked account "user"
----------------------
```

之后，用户可以再次登录。

22.1.3.1. 检查用户帐户的状态

要显示用户的失败登录尝试次数，请使用 ipa user-status 命令。如果显示的数字超过允许的失败登录尝试次数，则会锁定用户帐户。

```
$ ipa user-status user
-----------------------
Account disabled: False
-----------------------
  Server: example.com
  Failed logins: 8
  Last successful authentication: 20160229080309Z
  Last failed authentication: 20160229080317Z
  Time now: 2016-02-29T08:04:46Z
----------------------------
Number of entries returned 1
----------------------------
```

默认情况下，IdM 在 Red Hat Enterprise Linux 7.4 及之后的版本中取消启动，不会存储用户最后一次成功 Kerberos 身份验证的时间戳。要启用这个功能，请查看 第 22.2 节 “启用最后成功 Kerberos 身份验证的跟踪”。

22.2. 启用最后成功 KERBEROS 身份验证的跟踪

出于性能原因，在 Red Hat Enterprise Linux 7.4 及之后的版本上运行的 IdM 不会存储用户最后一次成功的 Kerberos 身份验证的时间戳。因此，某些命令（如 ipa user-status ）不会显示时间戳。

启用跟踪用户最近一次成功的 Kerberos 身份验证：

1. 显示当前启用的密码插件功能：

```
# ipa config-show | grep "Password plugin features"
  Password plugin features: AllowNThash, KDC:Disable Last Success
```

在以下步骤中，您需要功能的名称(KDC:Disable Last Success 除外)。

2. 将每个功能的 --ipaconfigstring=feature 参数传递给当前启用的 ipa config-mod 命令，但 KDC:Disable Last Success 除外：

```
# ipa config-mod --ipaconfigstring='AllowNThash'
```

这个命令只启用 **AllowNThash** 插件。要启用多个功能，请多次指定 **--ipaconfigstring=feature** 参数。例如，启用 **AllowNThash** 和 **KDC:Disable Lockout** 功能：

```
# ipa config-mod --ipaconfigstring='AllowNThash' --ipaconfigstring='KDC:Disable Lockout'
```

3. 重启 **IdM:**

```
# ipactl restart
```

22.3. 一次性密码

重要

用于 OTP 验证的 IdM 解决方案仅支持运行 Red Hat Enterprise Linux 7.1 或更高版本的客户端。

一次性密码(OTP)是一种仅对一个身份验证会话有效的密码，在使用后会无效。不同于传统的静态密码，身份验证令牌生成的 OTP 会不断更改。OTP 作为双因素验证的一部分：

1. 用户使用传统密码进行身份验证。
2. 用户提供由可识别的 **OTP** 令牌生成的 **OTP** 代码。

双因素身份验证被认为比仅使用传统密码进行身份验证更为安全。即使潜在的入侵者在登录期间截获 OTP，被拦截的 OTP 也会在那个时候无效，因为它只能用于成功验证一次。

警告

以下安全性和其他限制目前与 IdM 中的 OTP 支持相关：

- 最重要的安全限制是可能出现跨系统重播攻击的漏洞。复制是异步的，因此 OTP 代码可以在复制期间重复使用。个用户可以同时登录两台服务器。但是，由于综合加密，此漏洞通常很难被利用。
- 无法使用不支持 OTP 验证的客户端获取票据(TGT)。这可能会影响某些用例，如使用 mod_auth_kerb 模块或通用安全服务 API (GSSAPI)进行身份验证。
- 如果启用了 FIPS 模式，则无法在 IdM 解决方案中使用密码 + OTP。

22.3.1. IdM 中的 OTP 身份验证如何工作

22.3.1.1. IdM 中支持的 OTP 令牌

软件和硬件令牌

IdM 同时支持软件和硬件令牌。

用户管理和管理员管理的令牌

用户可以管理自己的令牌，或者管理员可以为其管理自己的令牌：

用户管理的令牌

用户可以完全控制身份管理中的用户管理令牌：允许他们创建、编辑或删除其令牌。

管理员管理的令牌

管理员将管理员管理的令牌添加到用户帐户。用户本身具有此类令牌的只读访问权限：他们没有管理或修改令牌的权限，而且无需以任何方式配置令牌。

请注意，如果令牌目前是唯一活跃的令牌，则用户无法删除或取消激活令牌。作为管理员，您无法删

除或取消激活最后一个活跃令牌，但您可以删除或取消激活其他用户的最后一个活跃令牌。

支持的 OTP 算法

身份管理支持以下两个标准的 OTP 机制：

- 基于 HMAC 的一次性密码(HOTP)算法是基于计数器的。HMAC 代表哈希消息身份验证代码。
- 基于时间的一次性密码(TOTP)算法是 HOTP 的扩展，来支持基于时间的移动因子。

22.3.1.2. 可用的 OTP 身份验证方法

启用 OTP 身份验证时，您可以从以下验证方法中选择：

双因素身份验证（密码 + OTP）

使用此方法时，始终需要用户输入标准密码和 OTP 代码。

密码

使用此方法时，用户仍可以选择仅使用标准密码进行身份验证。

RADIUS 代理服务器身份验证

有关为 OTP 验证配置 RADIUS 服务器的详情，请参考 第 22.3.7 节 “从专有 OTP 解决方案进行迁移”。

全局和用户特定身份验证方法

您可以全局或单独用户配置这些身份验证方法：

- 默认情况下，特定于用户的验证方法设置优先于全局设置。如果没有为用户设置身份验证方法，则将应用全局定义的方法。
- 您可以为任何用户禁用按用户的身份验证方法设置。这样可确保 IdM 忽略每个用户的设置，并且始终为用户应用全局设置。

组合多个身份验证方法

如果您一次设置多个方法，则其中任一方法都足以成功验证。例如：

- 如果您同时配置了双因素和密码身份验证，则用户必须提供密码（第一因素），但在使用命令行时，提供 OTP（第二个因子）是可选的：

```
First Factor:
Second Factor (optional):
```

- 在 Web UI 中，用户仍然必须提供这两个因素。

注意

单个主机或服务可以配置为需要特定的身份验证方法，如 OTP。如果您试图仅使用第一个因素对此类主机或服务进行身份验证，您将被拒绝访问。请参阅 第 22.4 节 “根据用户身份验证的方式限制对服务和主机的访问”。

但是，当配置了 RADIUS 和另一个验证方法时，会出现一个小的异常：

- Kerberos 将始终使用 RADIUS，但 LDAP 不会用到。LDAP 仅识别密码和双因素身份验证方法。

- 如果您使用外部的双因素身份验证供应商，请使用应用程序中的 Kerberos。如果您要仅允许用户使用密码进行身份验证，请使用 LDAP。建议应用利用 Apache 模块和 SSSD，允许配置 Kerberos 或 LDAP。

22.3.1.3. GNOME 密钥环服务支持

IdM 将 OTP 身份验证与 GNOME 密钥环服务集成。请注意，GNOME Keyring 集成要求用户单独输入第一个和第二个因素：

```
First factor: static_password
Second factor: one-time_password
```

22.3.1.4. 使用 OTP 进行离线身份验证

IdM 支持离线 OTP 身份验证。但是，为了能够脱机登录，用户必须首先通过输入静态密码和 OTP 分别在系统联机时进行验证：

```
First factor: static_password
Second factor: one-time_password
```

如果在在线登录时单独输入了这两个密码，那么即使中央身份验证服务器不可用，用户也可以进行身份验证。请注意，只有在用户离线验证时，IdM 才会提示输入传统静态密码。

IdM 还支持在 First factor 提示符的一个字符串中同时输入静态密码和 OTP。但请注意，这与离线 OTP 身份验证不兼容。如果用户在单个提示符中输入了这两个因素，IdM 始终必须在身份验证时联系中央身份验证服务器，这需要系统在线。

重要

如果您在同样离线操作的设备（如笔记本电脑）中使用 OTP 验证，红帽建议单独输入静态密码和 OTP 以确保可以使用离线验证。否则，IdM 不允许您在系统离线后登录。

如果要从 OTP 离线验证中受益，除了单独输入静态和 OTP 密码外，还要确保满足以下条件：

- /etc/sssd/sssd.conf 文件中的 cache_credentials 选项设置为 True，这将启用缓存第一个因素密码。
- 第一个因素静态密码满足 /etc/sssd/sssd.conf 中设置的 cache_credentials_minimal_first_factor_length 选项中定义的密码长度要求。默认最小长度为 8 个字符。有关这个选项的详情请参考 sssd.conf(5) man page。

请注意，即使 /etc/sssd/sssd.conf 中的 krb5_store_password_if_offline 选项被设置为 true，SSSD 不会在系统再次上线时尝试刷新 Kerberos ticket-granting ticket (TGT)，因为 OTP 可能已在此时无效。要在这种情形中获取 TGT，用户必须使用这两个因素再次进行身份验证。

22.3.2. 在 FIPS 模式运行的 IdM 服务器中配置 RADIUS 代理所需的设置

在联邦信息处理标准(FIPS)模式中，OpenSSL 默认禁用 MD5 摘要算法。因此，因为 RADIUS 协议需要 MD5 在 RADIUS 客户端和 RADIUS 服务器间加密 secret，因此 FIPS 模式的 MD5 不可用会导致 IdM RADIUS 代理服务器失败。

如果 RADIUS 服务器与 IdM master 在相同的主机上运行，您可以解决这个问题并在安全边界中启用 MD5，请执行以下步骤：

1. 使用以下内容创建 /etc/systemd/system/radiusd.service.d/ipa-otp.conf 文件：

```
[Service]
Environment=OPENSSL_FIPS_NON_APPROVED_MD5_ALLOW=1
```

2. 重新载入 systemd 配置：

```
# systemctl daemon-reload
```

3. 启动 radiusd 服务：

```
# systemctl start radiusd
```

22.3.3. 启用两个事实器身份验证

有关与 OTP 相关的可用验证方法的详情，请参考 第 22.3.1.2 节 “可用的 OTP 身份验证方法”。

启用双因素验证：

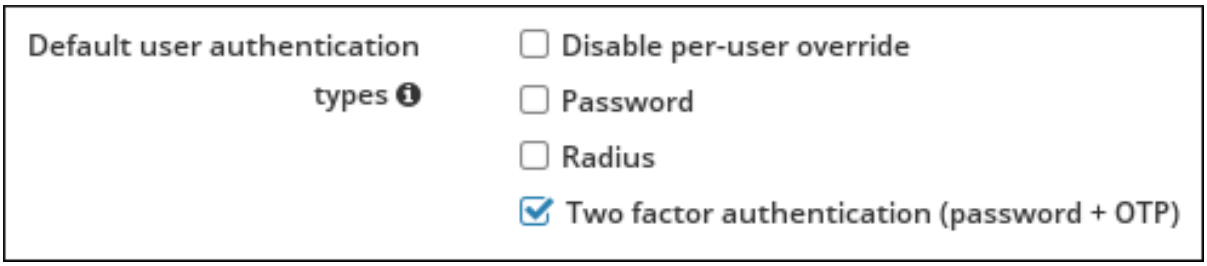
- Web UI，请参阅 “Web UI：启用两个事实器身份验证”一节。
- 命令行请查看 “命令行：启用两个事实器身份验证”一节。

Web UI：启用两个事实器身份验证

要为所有用户全局设置身份验证方法：

1. 选择 IPA Server → Configuration。
2. 在 User Options 区域中，选择所需的默认用户身份验证类型。

图 22.4. 用户身份验证方法



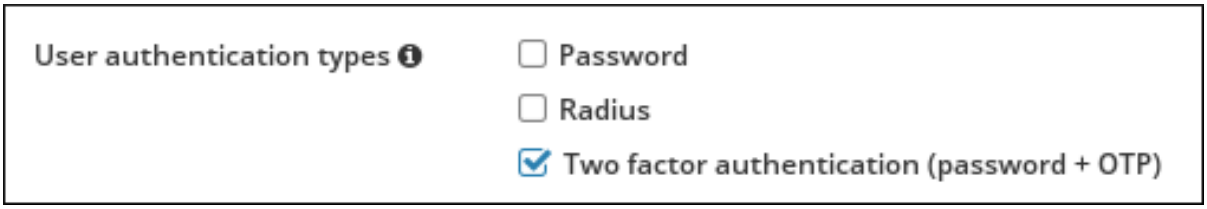
要确保按用户设置没有覆盖全局设置，请选择 **Disable per-user override**。如果您没有选择 **Disable per-user 覆盖**，则每个用户配置的身份验证方法优先于全局设置。

以每个用户为基础单独设置验证方法：

1. 选择 **Identity → Users**，然后单击要编辑的用户名称。

2. 在 帐户设置 区域中，选择所需的 用户身份验证类型。

图 22.5. 用户身份验证方法



命令行：启用两个事实器身份验证

要为所有用户全局设置身份验证方法：

1. 运行 **ipa config-mod --user-auth-type** 命令。例如，将全局身份验证方法设置为双因素验证：

```
$ ipa config-mod --user-auth-type=otp
```

如需 **--user-auth-type** 接受的值列表，请运行 **ipa config-mod --help** 命令。

2. 要禁用每个用户覆盖，因此请确保不使用每个用户的设置覆盖全局设置，同时添加 **--user-auth-type=disabled** 选项。例如，要将全局身份验证方法设置为双因素验证，并禁用每个用户覆盖：

```
$ ipa config-mod --user-auth-type=otp --user-auth-type=disabled
```

如果没有设置 **--user-auth-type=disabled**，则每个用户配置的身份验证方法优先于全局设置。

为指定用户单独设置身份验证方法：

- 运行 **ipa user-mod --user-auth-type** 命令。例如，要设置 该用户 需要使用双因素身份验证：

```
$ ipa user-mod user --user-auth-type=otp
```

要设置多种身份验证方法，请多次添加 **--user-auth-type**。例如，要为所有用户全局配置密码和双因素身份验证：

```
$ ipa config-mod --user-auth-type=otp --user-auth-type=password
```

22.3.4. 添加用户管理的软件令牌

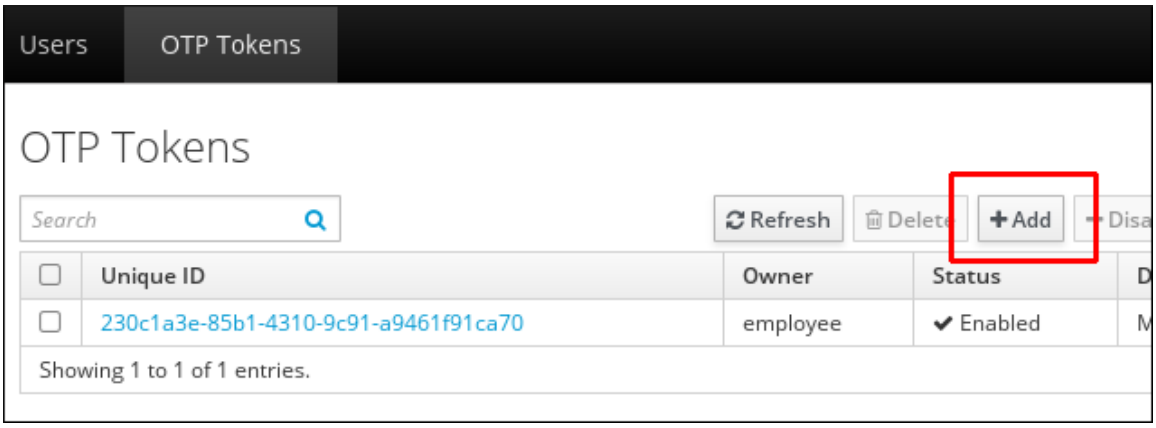
1. 使用标准密码登录。

2. 确保在您的移动设备上安装了 **FreeOTP Authenticator** 应用程序。要下载 **FreeOTP Authenticator**，请参阅 **FreeOTP 源页面**。

3. 在 **IdM Web UI** 或命令行中创建软件令牌。

   - 要在 **Web UI** 中创建令牌，请单击 **OTP 令牌** 选项卡下的 **Add**。如果您以管理员身份登录，则可通过 **Authentication** 选项卡访问 **OTP Tokens** 选项卡。

图 22.6. 为用户添加 OTP 令牌



- *要从命令行创建令牌，请运行 **ipa otptoken-add** 命令。*

```
$ ipa otptoken-add
------------------
Added OTP token ""
------------------
  Unique ID: 7060091b-4e40-47fd-8354-cb32fecd548a
  Type: TOTP
...
```

*有关 **ipa otptoken-add** 的更多信息，请运行带有 **--help** 选项的命令。*

4. *Web UI 或命令行中会显示 802.11 代码。使用 FreeOTP Authenticator 扫描 QR 代码，将令牌置备到移动设备。*

22.3.5. 添加用户管理的 YubiKey 硬件令牌

可编程性硬件令牌（如 YubiKey 令牌）只能从命令行添加。以拥有令牌的用户添加 YubiKey 硬件令牌：

1. *使用标准密码登录。*
2. *插入您的 YubiKey 令牌。*
3. *运行 **ipa otptoken-add-yubikey** 命令。*

- 如果 YubiKey 有一个空插槽，该命令会自动选择空插槽。

- 如果没有可用的空插槽，则必须使用 **--slot** 选项手动选择一个插槽。例如：

```
$ ipa otptoken-add-yubikey --slot=2
```

请注意，这会覆盖所选的插槽。

22.3.6. 以管理员身份为用户添加令牌

以管理员身份添加软件令牌：

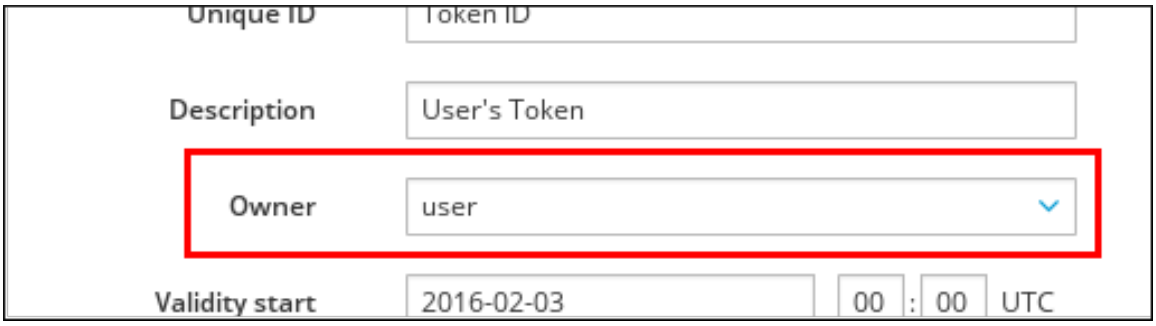
1. 确保您已以管理员身份登录。

2. 确保在移动设备上安装了 **FreeOTP Authenticator** 应用程序。要下载 **FreeOTP Authenticator**，请参阅 **FreeOTP 源页面**。

3. 在 IdM Web UI 或命令行中创建软件令牌。

- 要在 **Web UI** 中创建令牌，请选择 **Authentication → OTP Tokens**，再单击 **OTP** 令牌列表顶部的 **Add**。在 **Add OTP Token** 表单中，选择令牌的所有者。

图 22.7. 添加管理员管理的软件令牌



- 要从命令行创建令牌，请使用 **--owner** 选项运行 **ipa otptoken-add** 命令。例如：

```
$ ipa otptoken-add --owner=user
----------------
```

```
Added OTP token ""
------------------
  Unique ID: 5303baa8-08f9-464e-a74d-3b38de1c041d
  Type: TOTP
...
```

4. Web UI 或命令行中会显示 802.11 代码。使用 FreeOTP Authenticator 扫描 QR 代码，将令牌置备到移动设备。

以管理员身份添加可编程性硬件令牌，如 YubiKey 令牌：

1. 确保您已以管理员身份登录。

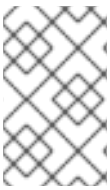
2. 插入 YubiKey 令牌。

3. 使用 --owner 选项运行 ipa otptoken-add-yubikey 命令。例如：

```
$ ipa otptoken-add-yubikey --owner=user
```

22.3.7. 从专有 OTP 解决方案进行迁移

为启用将大型部署从专有 OTP 解决方案迁移到 IdM 原生 OTP 解决方案，IdM 提供了一种将 OTP 验证卸载到第三方 RADIUS 服务器的方法。管理员创建一组 RADIUS 代理，每个代理只能引用单个 RADIUS 服务器。如果需要解决多台服务器，建议创建一个虚拟 IP 解决方案，指向多个 RADIUS 服务器。此类解决方案需要在 RHEL IdM 外部构建，并使用 keepalived 守护进程（例如：然后，管理员将这些代理集中的一个分配给用户。只要用户设置了 RADIUS 代理，IdM 会绕过所有其他身份验证机制。



注意

IdM 不提供对第三方系统中令牌的任何令牌管理或同步支持。

为 OTP 验证配置 RADIUS 服务器，并将用户添加到代理服务器中：

1. 确保启用了 radius 用户身份验证方法。详情请查看 第 22.3.3 节 “启用两个事实器身份验证”。

2. 运行 ipa radiusproxy-add proxy_name -- secretsecret 命令来添加 RADIUS 代理。命令提示您插入所需信息。

   RADIUS 代理的配置要求客户端和服务器之间使用通用机密来打包凭据。在 --secret 参数中指定此 secret。

3. 运行 ipa user-mod radiususer --radius=proxy_name 命令将用户分配给添加的代理。

4. 如果需要，通过运行 ipa user-mod radiususer --radius-username=radius_user 命令将用户名配置为发送到 RADIUS。

因此，用户 OTP 身份验证将开始通过 RADIUS 代理服务器进行处理。

注意

要在启用了 FIPS 模式的 IdM master 上运行 RADIUS 服务器，请执行 第 22.3.2 节 “在 FIPS 模式运行的 IdM 服务器中配置 RADIUS 代理所需的设置” 中描述的步骤。

当用户准备好迁移到 IdM 原生 OTP 系统时，您只需删除用户的 RADIUS 代理分配。

22.3.7.1. 在低网络中运行 RADIUS 服务器时更改 KDC 的超时值

在某些情况下，比如在较慢的网络中运行 RADIUS 代理，IdM KDC 会在 RADIUS 服务器响应前关闭连接，因为等待用户进入令牌时连接超时。

更改 KDC 的超时设置：

1. 更改 /var/kerberos/krb5kdc/kdc.conf 文件中的 [otp] 部分中的 timeout 参数的值。例如，要将超时设置为 120 秒：

```
[otp]
DEFAULT = {
  timeout = 120
  ...
}
```

2. 重启 krb5kdc 服务：

```
# systemctl restart krb5kdc
```

22.3.8. 将当前凭证提升为两次身份验证

如果同时配置了密码和双因素验证，但您仅使用密码进行身份验证，您可能会被拒绝访问某些服务或主机（请参阅 第 22.4 节 “根据用户身份验证的方式限制对服务和主机的访问”）。在这种情况下，通过再次进行身份验证，将您的凭证从一个因素提升到双因素身份验证：

1. 锁定屏幕.锁定屏幕的默认键盘快捷方式是 Super key+L。

2. 解锁屏幕.当系统询问凭据时，请使用 password 和 OTP。

22.3.9. 重新同步 OTP 令牌

请参阅 第 B.4.3 节 “OTP 令牌不同步”。

22.3.10. 替换丢失的 OTP 令牌

以下流程描述了丢失 OTP 令牌的用户如何替换令牌：

1. 作为管理员，为用户启用密码和 OTP 身份验证：

```
[admin@server]# ipa user-mod --user-auth-type=password --user-auth-type=otp user_name
```

2. 用户现在可以添加新的令牌。例如，要添加在描述 中设置的新令牌 的新令牌：

```
[user@server]# ipa otptoken-add --desc="New Token"
```

如需了解更多详细信息，请输入添加 ipa otptoken-add --help 参数的命令。

3. 用户现在可以删除旧的令牌：

a.

另外，还可列出与帐户关联的令牌：

```
[user@server]# ipa otptoken-find
--------------------
2 OTP tokens matched
--------------------
  Unique ID: 4ce8ec29-0bf7-4100-ab6d-5d26697f0d8f
  Type: TOTP
  Description: New Token
  Owner: user

  Unique ID: e1e9e1ef-172c-4fa9-b637-6b017ce79315
  Type: TOTP
  Description: Old Token
  Owner: user
----------------------------
Number of entries returned 2
----------------------------
```

b.

删除旧令牌。例如，使用 e1e9e1ef-172c-4fa9-b637-6b017ce79315 ID 删除令牌：

```
[user@server]# # ipa otptoken-del e1e9e1ef-172c-4fa9-b637-6b017ce79315
-----------------------------------------------------
Deleted OTP token "e1e9e1ef-172c-4fa9-b637-6b017ce79315"
-----------------------------------------------------
```

4.

作为管理员，只为用户启用 OTP 验证：

```
[admin@server]# ipa user-mod --user-auth-type=otp user_name
```

22.4. 根据用户身份验证的方式限制对服务和主机的访问

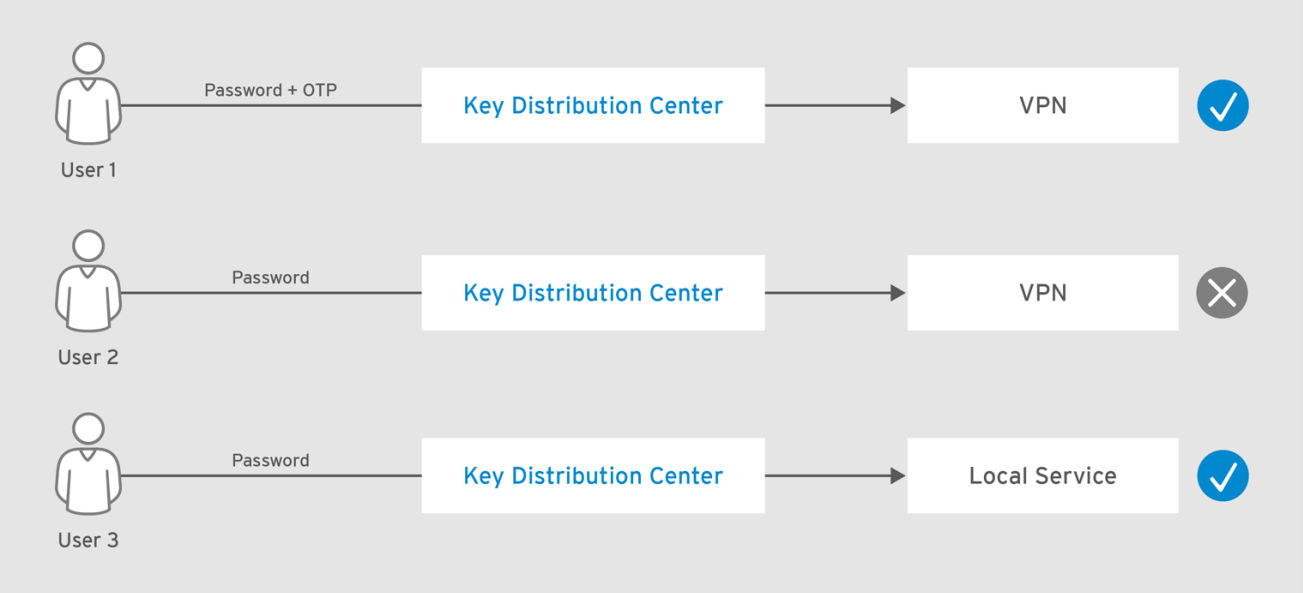
IdM 支持的身份验证机制在身份验证强度上有所不同。例如，使用一次性密码(OTP)和标准密码进行身份验证被视为比仅使用标准密码进行身份验证更安全。本节介绍如何根据用户身份验证方式限制对服务和主机的访问。

例如，您可以配置：

- 对安全性至关重要的服务，如 VPN，需要强大的身份验证方法

- 非关键服务（如本地登录）以允许使用较弱但更为方便的身份验证方法进行身份验证

图 22.8. 使用不同方法验证示例



RHEL_404973_1016

身份验证标识符

对服务和主机的访问通过 身份验证指示器 定义：

- 服务或主机条目中包含的指标决定了用户可用于访问该服务或主机的身份验证方法。
- 用户票证(TGT)中包含的指标显示了用于获取票据的验证方法。

如果主体中的指示符与 TGT 中的指示符不匹配，则会拒绝用户访问。

22.4.1. 配置主机或服务以需要特定的身份验证方法

使用以下命令配置主机或服务：

- Web UI，请参见 “Web UI：配置主机或服务以需要特定的身份验证方法”一节
- 命令行，请查看 “命令行：配置主机或服务以需要特定的身份验证方法”一节

Web UI：配置主机或服务以需要特定的身份验证方法

1. 选择 **Identity → Hosts 或 Identity → Services。**

2. 单击所需主机或服务的名称。

3. 在 **Authentication indicators 下，选择所需的验证方法。**

   - 例如，选择 **OTP 可确保只有使用带有密码的有效 OTP 代码的用户才能访问主机或服务。**

   - 如果您选择 **OTP 和 RADIUS，则 OTP 或 RADIUS 都足以允许访问。**

4. 点页面顶部的 **Save。**

命令行：配置主机或服务以需要特定的身份验证方法

1. 可选。使用 **ipa host-find 或 ipa service-find 命令来识别主机或服务。**

2. 使用带有 **--auth-ind 选项的 ipa host-mod 或 ipa service-mod 命令来添加所需的身份验证指标。有关 --auth-ind 接受的值列表，请查看 ipa host-mod --help 或 ipa service-mod --help 命令的输出。**

   例如，**--auth-ind=otp 确保只有使用有效 OTP 代码的用户才被允许访问主机或服务：**

```
$ ipa host-mod server.example.com --auth-ind=otp
----------------------------------------------------------
Modified host "server.example.com"
----------------------------------------------------------
  Host name: server.example.com
  ...
  Authentication Indicators: otp
  ...
```

   如果您同时为 **OTP 和 RADIUS 添加指示器，OTP 或 RADIUS 将足以允许访问。**

22.4.2. 更改 Kerberos 身份验证标识符

默认情况下，身份管理(IdM)使用 pkinit 指示器来使用 PKINIT 预身份验证插件进行 Kerberos 身份验证的证书映射。如果您需要更改身份验证供应商，Kerberos 分发中心(KDC)插入到票据授予票据(TGT)中，请修改提供 PKINIT 功能的所有 IdM master 上的配置，如下所示：

1. 在 /var/kerberos/krb5kdc/kdc.conf 文件中，将 pkinit_indicator 参数添加到 [kdcdefaults] 部分：

```
# pkinit_indicator = indicator
```

您可以设置指示符以下值：

- OTP 用于两个因素身份验证
- RADIUS 用于基于 RADIUS 的身份验证
- PKINIT 用于智能卡验证

2. 重启 krb5kdc 服务：

```
# systemctl restart krb5kdc
```

22.5. 管理用户的公共 SSH 密钥

通过身份管理，您可以将 SSH 公共密钥上传到用户条目。有权访问对应私有 SSH 密钥的用户可以使用 ssh 登录 IdM 机器，而无需使用 Kerberos 凭据。如果正确配置了 pam_krb5，或者 SSSD 用作 IdM 服务器的身份提供程序，用户在登录后收到 Kerberos ticket-granting ticket (TGT)；更多详情，请参阅 “自动获取 Kerberos Tickets”一节。

请注意，如果用户从不可用其 SSH 私钥文件的计算机登录，则用户仍可提供其 Kerberos 凭据进行身份验证。

自动缓存和检索 SSH 密钥

在 IdM 服务器或客户端安装过程中，SSSD 会在机器上自动配置，以缓存和检索用户和主机 SSH 密钥。这允许 IdM 作为 SSH 密钥的通用集中式存储库。

如果在安装过程中没有配置服务器或客户端，您可以在机器上手动配置 SSSD。有关如何执行此操作的详情请参考 第 22.6 节 “配置 SSSD 为 OpenSSH 服务提供缓存”。请注意，SSSD 缓存 SSH 密钥需要本地机器上的管理权限。

SSH 密钥格式要求

IdM 接受以下两种 SSH 密钥格式：

openssh 样式键

有关此格式的详情，请参阅 RFC 4716。

原始 RFC 4253-style 键

有关此格式的详情，请参阅 RFC 4253。

请注意，IdM 会自动将 RFC 4253 样式的密钥转换为 OpenSSH 样式的密钥，然后再将其保存到 IdM LDAP 服务器中。

密钥文件（如 id_rsa.pub ）包含三个部分：密钥类型、密钥本身以及附加注释或标识符。在以下示例中，密钥类型是 RSA，注释将密钥与 client.example.com 主机名相关联：

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQDMM4xPu54Kf2dx7C4Ta2F7vnIzuL1i6P21TTKniSkjFuA+r

qW06588e7v14Im4VejwnNk352gp49A62qSVOzp8IKA9xdtyRmHYCTUvmkcyspZvFRI713zfRKQVFyJO
qHmW/m
dCmak7QBxYou2ELSPhH3pe8MYTQIulKDSu5Zbsrqedg1VGkSJxf7mDnCSPNWWzAY9AFB9Lmd2m
2xZmNgVAQEQ
nZXNMaIlroLD/51rmMSkJGHGb1O68kEq9Z client.example.com
```

将密钥上传到 IdM 时，您可以上传所有三个关键部分，或者只上传密钥本身。如果您只上传密钥本身，IdM 会自动从上传的密钥中识别密钥类型，如 RSA 或 DSA。

22.5.1. 生成 SSH 密钥

您可以使用 OpenSSH ssh-keygen 工具生成 SSH 密钥。实用程序显示有关公钥位置的信息。例如：

```
$ ssh-keygen -t rsa -C user@example.com
Generating public/private rsa key pair.
Enter file in which to save the key (/home/user/.ssh/id_rsa):
Created directory '/home/user/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/user/.ssh/id_rsa.
Your public key has been saved in /home/user/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:GAUIDVVEgly7rs1ITWP6oguHz8BKvyZkpqCqVSsmi7c user@example.com
The key's randomart image is:
+--[ RSA 2048]----+
|                 |
|     + .         |
|    + =   .      |
|     =   +       |
|    . E S..      |
|   .    . .o     |
|    . .  . oo.   |
|   . o .  +.+o   |
|    o  .o..o+o   |
+-----------------+
```

要上传用户的 SSH 密钥，请使用存储在显示文件中的公钥字符串。

22.5.2. 上传用户 SSH 密钥

22.5.2.1. Web UI：上传用户 SSH 密钥

1. 选择 **Identity → Users**。
2. 单击要编辑的用户的名称。
3. 在 **Account Settings** 区域的 **Settings** 选项卡下，点 **SSH 公钥：添加**。

图 22.9. 帐户设置中的 SSH 公钥

4. *粘贴为 Base 64 编码的公钥字符串，然后单击 Set。*

图 22.10. 在公钥中粘贴

5. *点页面顶部的 Save。*

22.5.2.2. 命令行：上传用户 SSH 密钥

使用 ipa user-mod 命令，并使用 --sshpubkey 选项传递 Base 64 编码的公钥字符串。

例如，上传密钥类型、密钥本身和主机名标识符：

```
$ ipa user-mod user --sshpubkey="ssh-rsa AAAAB3Nza...SNc5dv== client.example.com"
```

要上传多个密钥，请多次使用 --sshpubkey。例如，上传两个 SSH 密钥：

```
--sshpubkey="AAAAB3Nza...SNc5dv==" --sshpubkey="RjlzYQo...ZEt0TAo="
```

注意

您可以使用命令重定向并指向包含 键的文件，而不是手动将密钥粘贴到命令行中。例如：

```
$ ipa user-mod user --sshpubkey="$(cat ~/.ssh/id_rsa.pub)" --sshpubkey="$(cat ~/.ssh/id_rsa2.pub)"
```

22.5.3. 删除用户密钥

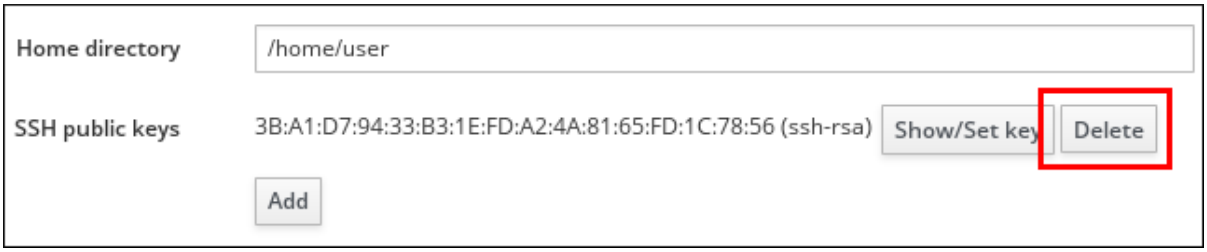
删除 SSH 密钥：

- 使用 Web UI，请参阅 第 22.5.3.1 节 “Web UI：删除用户 SSH 密钥”
- 使用命令行，请查看 第 22.5.3.2 节 “命令行：删除用户 SSH 密钥”

22.5.3.1. Web UI：删除用户 SSH 密钥

1. 选择 **Identity** → **Users**。
2. 单击要编辑的用户的名称。
3. 在 **Account Settings** 区域的 **Settings** 选项卡下，单击您要删除的密钥旁边的 **Delete**。

图 22.11. 删除用户 SSH 公钥



4. 点页面顶部的 **Save**。

22.5.3.2. 命令行：删除用户 SSH 密钥

要删除分配给用户帐户的所有 SSH 密钥，请在 ipa user-mod 命令中添加 --sshpubkey 选项，而不指定任何密钥：

```
$ ipa user-mod user --sshpubkey=
```

如果您只想删除特定的 SSH 密钥或密钥，请使用 --sshpubkey 选项指定您要保留的密钥或密钥。

注意

此命令不会立即从缓存中删除 SSH 密钥。使用默认缓存超时值 (entry_cache_timeout = 5400)时，密钥会在缓存中保留 1 小时和半小时。

22.6. 配置 SSSD 为 OPENSSH 服务提供缓存

系统安全服务守护进程(SSSD)为多个系统服务提供接口，包括 OpenSSH。

本节论述了如何将 SSSD 配置为为机器和用户缓存 SSH 密钥。

22.6.1. SSSD 如何通过 OpenSSH 工作

OpenSSH 是一种 SSH 协议实施。OpenSSH 基于用于识别身份验证实体 的公钥-私钥对，在两个系统之间创建安全的加密连接。详情请参阅 系统管理员指南中的 OpenSSH。

SSSD 可以充当机器和用户的 SSH 公钥的凭据缓存。在这个设置中：

1. OpenSSH 配置为引用 SSSD 以检查缓存的密钥。
2. SSSD 使用身份管理(IdM)域，IdM 存储公钥和主机信息。

注意

只有 IdM 域中的 Linux 机器才能使用 SSSD 作为 OpenSSH 的重要缓存。其他计算机（包括 Windows 机器）不能.

SSSD 如何管理主机密钥

要管理主机密钥，SSSD 会执行以下操作：

1. 从主机系统检索公钥。
2. 将主机密钥存储在 /var/lib/sss/pubconf/known_hosts 文件中。
3. 建立与主机计算机的连接。

有关所需配置步骤的详情，请查看 第 22.6.2 节 “将 OpenSSH 配置为将 SSSD 用于主机密钥”。

SSSD 如何管理用户密钥

要管理用户密钥，SSSD 会执行以下操作：

1. 从 IdM 域中的用户条目检索用户的公钥。
2. 以标准授权密钥格式，将用户密钥存储为 .ssh/sss_authorized_keys 文件。

有关所需配置步骤的详情，请查看 第 22.6.3 节 “将 OpenSSH 配置为为用户密钥使用 SSSD”。

22.6.2. 将 OpenSSH 配置为将 SSSD 用于主机密钥

您可以针对每个用户或整个系统更改配置。

1. 打开所需的配置文件。

a.
要更改特定于用户的配置，请打开 ~/.ssh/config 文件。

b.
要更改系统范围的配置，请打开 /etc/ssh/sshd_config 文件。

2.
使用 ProxyCommand 选项指定将用来连接到 SSH 客户端的命令（带有所需参数和主机名的 sss_ssh_knownhostsproxy 工具）。

有关 sss_ssh_knownhostsproxy 的详情，请查看 sss_ssh_knownhostsproxy(1) man page。

3.
使用 GlobalKnownHostsFile 选项指定 SSSD 主机文件的位置：/var/lib/sss/pubconf/known_hosts。此文件将被使用，而不是默认的 OpenSSH known_hosts 文件。

以下示例将 SSH 配置为在 SSSD 域中查找公钥，并通过提供的端口和主机进行连接：

```
ProxyCommand /usr/bin/sss_ssh_knownhostsproxy -p %p %h
GlobalKnownHostsFile /var/lib/sss/pubconf/known_hosts
```

有关配置 SSH 和配置文件的详情，请参考 ssh_config(5) man page。

22.6.3. 将 OpenSSH 配置为为用户密钥使用 SSSD

您可以更改整个系统的配置。

1.
打开 /etc/ssh/sshd_config 文件。

2.
使用 AuthorizedKeysCommand 选项指定将要执行的命令来检索用户密钥。

3.
使用 AuthorizedKeysCommandUser 选项指定在其下运行命令的帐户下的用户。

以下示例将 SSH 配置为在用户帐户下运行 sss_ssh_authorizedkeys 实用程序。

```
AuthorizedKeysCommand /usr/bin/sss_ssh_authorizedkeys
AuthorizedKeysCommandUser user
```

有关 sss_ssh_authorizedkeys 的详情，请查看 sss_ssh_authorizedkeys(1) man page。

有关配置 SSH 和配置文件的详情，请参考 ssh_config(5) man page。

22.7. 身份管理中的智能卡身份验证

有关身份管理中的智能卡验证的详情，请参考 第 23 章 身份管理中的智能卡身份验证。

22.8. 用户证书

有关用户证书的详情请参考 第 24 章 管理用户、主机和服务的证书。

第 23 章 身份管理中的智能卡身份验证

使用基于智能卡的验证是使用密码进行验证的替代选择。用户凭证存储在智能卡中，而特殊软件和硬件则用于访问它们。用户将智能卡放入读卡器，并为智能卡提供 PIN 代码。

本章论述了如何在身份管理中配置基于智能卡的身份验证，以及用户如何使用智能卡验证身份管理。

23.1. 从智能卡导出证书

导出证书：

1. 将智能卡放在读取器中。

2. 使用以下命令列出智能卡上的证书。在输出中，找到用于身份验证的证书，并注意其 nickname：

```
$ certutil -L -d /etc/pki/nssdb/ -h all
Certificate Nickname         Trust Attributes
                             SSL,S/MIME,JAR/XPI

my_certificate               CT,C,C
```

3. 使用证书 nickname 将证书提取到文件中。例如，将 Base64 格式的证书提取到名为 user.crt 的文件：

```
$ certutil -L -d /etc/pki/nssdb/ -n 'my_certificate' -r | base64 -w 0 > user.crt
```

base64 工具是 coreutils 软件包的一部分。

23.2. 在身份管理中配置证书映射规则

23.2.1. 在智能卡中配置身份验证的证书映射规则

证书映射规则是允许用户在身份管理(IdM)管理员无法访问某些用户证书时，允许用户使用证书进行身份验证的便捷方式。缺少访问权限的原因通常是因为证书是由外部证书颁发机构颁发的。一个特殊用例由 IdM 域处于信任关系的 Active Directory(AD)证书系统发布的证书代表。

如果 IdM 环境较大且有大量使用智能卡的用户，使用证书映射规则就会比较方便。在这种情况下，添加完整证书可能会比较复杂。大多数情况下，主题和发行者都是可预测的，因此提前添加比完整证书更容易。作为系统管理员，您可以创建证书映射规则，并在将证书映射数据添加到特定用户之前将证书映射数据添加到用户条目。签发证书后，用户将能够使用该证书登录，即使完整证书没有上传到其条目中。

另外，由于证书必须定期续订，证书映射规则可降低管理开销。当用户的证书被更新时，管理员不必更新用户条目。例如，如果映射基于 Subject 和 Issuer 值，如果新证书的主题和签发者与旧证书相同，则映射仍适用。如果使用完整证书，则管理员必须将新证书上传到用户条目以替换旧证书。

设置证书映射：

1. 管理员必须将证书映射数据（通常是签发者和主题）加载到用户帐户中。

2. 管理员必须创建证书映射规则，以便用户成功登录到 IdM:

   - 其帐户包含证书映射数据条目
   - 哪个证书映射数据条目与证书的信息匹配

   有关组成映射规则的单个组件以及如何获取和使用它们的详情，请参阅 IdM 中的身份映射规则的组件，以及从证书中排除签发者以在匹配规则中使用的证书。

23.2.1.1. 使用 Active Directory 域进行信任的证书映射规则

本节概述了如果 IdM 部署与 Active Directory(AD)域存在信任关系时可能的不同证书映射用例。

证书映射规则是一个便捷的方法，可为具有可信 AD 证书系统发布的智能卡证书的用户启用对 IdM 资源的访问。根据 AD 配置，可能会出现以下情况：

- 如果证书由 AD 颁发，但用户和证书存储在 IdM 中，则在 IdM 端执行身份验证请求的映射和整个处理。有关配置这种情况的详情请参考 第 23.2.2 节 “为存储在 IdM 中的用户配置证书映射”。

- 如果用户存储在 AD 中，则会在 AD 中处理身份验证请求。有三个不同的子案例：

  - AD 用户条目包含整个证书。有关如何在此场景中配置 IdM 的详情请参考 第 23.2.3 节 “为用户配置证书映射，Whose AD User Entry 包含 Whole Certificate”。

  - AD 配置为将用户证书映射到用户帐户。在本例中，AD 用户条目不包含整个证书，而是包含名为 altSecurityIdentities 的属性。有关如何在此场景中配置 IdM 的详情请参考 第 23.2.4 节 “如果将 AD 配置为将用户证书映射到用户帐户，则配置证书映射”。

  - AD 用户条目既不包含整个证书，也不包含映射数据。在这种情况下，唯一的解决方案是使用 ipa idoverrideuser-add 命令将整个证书添加到 IdM 中的 AD 用户的 ID 覆盖中。详情请查看 第 23.2.5 节 “如果 AD 用户输入不包含证书或映射数据，则配置证书映射”。

23.2.1.2. IdM 中身份管理规则的组件

本节介绍了 IdM 中身份映射规则的组件以及如何配置它们。每个组件都有一个可覆盖的默认值。您可以在 web UI 或命令行中定义组件。在命令行中，身份映射规则是使用 ipa certmaprule-add 命令创建的。

映射规则

映射规则组件将证书与一个或多个用户帐户关联（或映射）。规则定义一个 LDAP 搜索过滤器，它将证书与预期用户帐户相关联。

不同证书颁发机构(CA)发布的证书可能有不同的属性，可以在不同的域中使用。因此，IdM 不会无条件地应用映射规则，而只应用到适当的证书。使用匹配规则定义适当的证书。

请注意，如果您将映射规则选项留空，则证书将在 userCertificate 属性中作为 DER 编码的二进制文件进行搜索。

使用 --maprule 选项，在命令行中定义映射规则。

匹配规则

域列表指定您希望 IdM 在处理身份映射规则时搜索用户的身份域。如果未指定 选项，IdM 仅在 IdM 客户端所属本地域中搜索用户。

使用 --domain 选项，在命令行中定义域。

优先级

当多个规则适用于证书时，优先级最高的规则优先。所有其他规则将被忽略。

- 数字值越低，身份映射规则的优先级越高。例如，具有优先级 1 的规则的优先级高于优先级 2 的规则。
- 如果规则没有定义优先级值，它具有最低的优先级。

使用 --priority 选项，在命令行中定义映射规则优先级。

> **例 23.1. 证书映射规则示例**
>
> 要定义，使用命令行，一个名为 simple_rule 的证书映射规则，允许对 EXAMPLE.ORG 机构智能卡 CA 发布的证书进行身份验证，只要该证书上的 Subject 与 IdM 中用户帐户中的 certmapdata 条目匹配：
>
> ```
> # ipa certmaprule-add simple_rule --matchrule '<ISSUER>CN=Smart Card
> CA,O=EXAMPLE.ORG' --maprule '(ipacertmapdata=X509:<I>{issuer_dn!nss_x500}<S>
> {subject_dn!nss_x500})'
> ```

23.2.1.3. 从匹配规则中的证书获取颁发者

这个步骤描述了如何从证书获取签发者信息，以便您可以将其复制并粘贴到证书映射规则的匹配规则中。要获得匹配规则所需的签发者格式，请使用 openssl x509 命令。

先决条件

- 您有 .pem 或 .crt 格式的用户证书。

步骤

1. 从证书获取用户信息。使用 openssl 证书显示和签名工具：

- 用于防止请求编码版本的输出的 **-noout** 选项
- 输出签发者名称的 **-issuer** 选项
- **in** 选项指定 要从中读取证书的输入文件名
- 使用 **RFC2253** 值的 **-nameopt** 选项，首先显示最具体相对可分辨名称**(RDN)**的输出

如果输入文件包含身份管理证书，命令的输出会显示使用机构 信息 定义了发行者：

```
# openssl x509 -noout -issuer -in idm_user.crt -nameopt RFC2253
issuer=CN=Certificate Authority,O=REALM.EXAMPLE.COM
```

如果输入文件包含一个 **Active Directory** 证书，命令的输出会显示使用 域组件 信息定义了发行者：

```
# # openssl x509 -noout -issuer -in ad_user.crt -nameopt RFC2253
issuer=CN=AD-WIN2012R2-CA,DC=AD,DC=EXAMPLE,DC=COM
```

2. （可选）要在命令行中基于匹配规则创建一个新的映射规则，该规则指定证书签发者必须是 **ad.example.com** 域的提取 **AD-WIN2012R2-CA**，证书上的主题必须与 **IdM** 中用户帐户中的 **certmapdata** 条目匹配：

```
# ipa certmaprule-add simple_rule --matchrule '<ISSUER>CN=AD-WIN2012R2-
CA,DC=AD,DC=EXAMPLE,DC=COM' --maprule '(ipacertmapdata=X509:<I>
{issuer_dn!nss_x500}<S>{subject_dn!nss_x500})'
```

其它信息

有关 **certmap** 命令的详情，包括匹配规则和映射规则支持的格式的信息，以及优先级和域字段的说明，请参阅 **sss-certmap(5) man page**。

23.2.2. 为存储在 IdM 中的用户配置证书映射

这部分论述了当在 **IdM** 中存储证书身份验证的用户时，系统管理员在 **IdM** 中启用证书映射所需的步骤。

先决条件

- 用户在 IdM 中有一个帐户。
- 管理员具有要添加到用户条目的完整证书或证书映射数据。

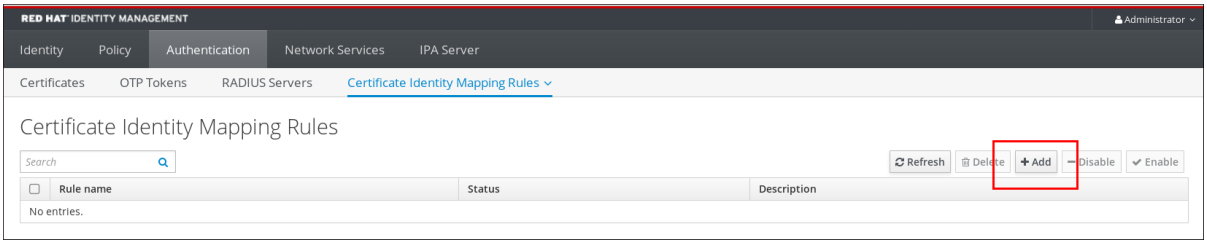
23.2.2.1. 在 IdM 中添加证书映射规则

本节论述了如何设置证书映射规则，以便具有与映射规则中指定的条件以及证书映射数据条目中指定的证书的 IdM 用户可以进行身份验证到 IdM。

23.2.2.1.1. 在 IdM Web UI 中添加证书映射规则

1. 以管理员身份登录 IdM Web UI。
2. 导航到 **Authentication** → **Certificate Identity Mapping Rules** → **Certificate Identity Mapping Rules**。
3. 点击 **Add**。

图 23.1. 在 IdM Web UI 中添加新证书映射规则



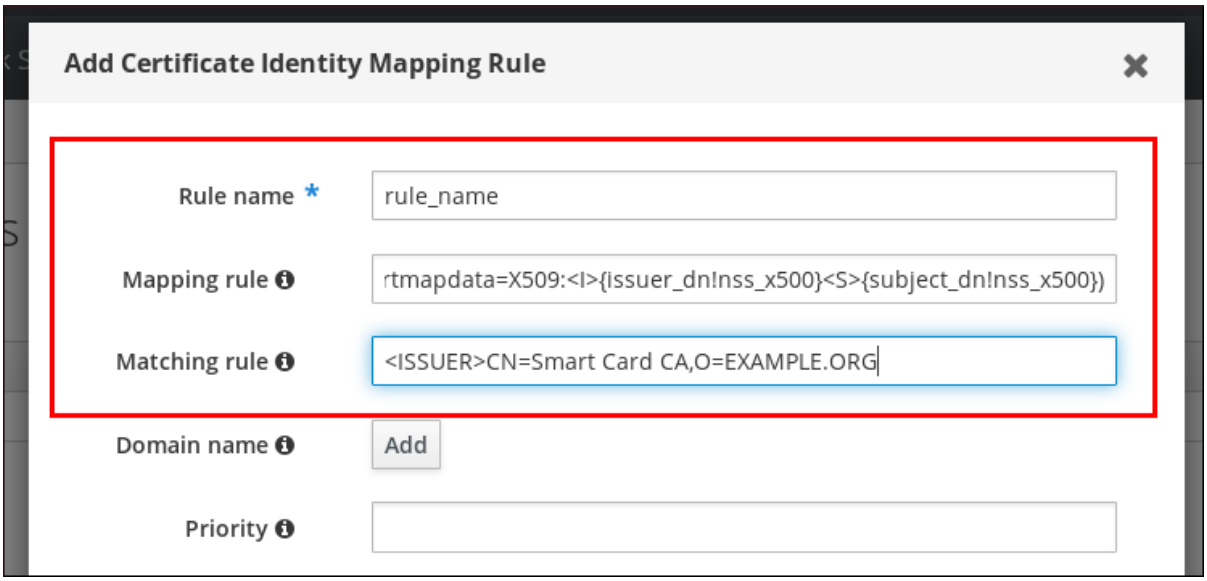
4. 输入规则名称。
5. 输入映射规则。例如，要让 IdM 搜索提供给它们的任何证书中的 Issuer 和 Subject 条目，并根据在提供的证书的两个条目中找到的信息进行验证或不进行验证，请输入：

```
(ipacertmapdata=X509:<I>{issuer_dn!nss_x500}<S>{subject_dn!nss_x500})
```

6. 输入匹配的规则。例如，只允许 EXAMPLE.ORG 机构的智能卡 CA 发布的证书来验证用户到 IdM，请输入：

```
<ISSUER>CN=Smart Card CA,O=EXAMPLE.ORG
```

图 23.2. 在 IdM Web UI 中输入证书映射规则的详情



7. 单击对话框底部的 **Add**，以添加该规则并关闭该框。

8. 系统安全服务守护进程(SSSD)定期重新读取证书映射规则。要强制立即载入新创建的规则，请重启 **SSSD**：

```
# systemctl restart sssd
```

现在，您设置了证书映射规则，将它在智能卡证书中找到的映射规则中指定的数据类型与 IdM 用户条目中的证书映射数据进行比较。找到匹配项后，它将验证匹配的用户。

23.2.2.1.2. 使用命令行添加证书映射规则

1. 获取管理员凭证：

```
# kinit admin
```

2. 输入映射规则，以及映射规则所基于的匹配规则。例如，要让 IdM 搜索所呈现的任何证书中的 发行者和 Subject 条目，并基于所显示证书的两个条目中找到的信息进行身份验证，仅识别由 EXAMPLE.ORG 机构的 智能卡 CA 发布的证书：

```
# ipa certmaprule-add rule_name --matchrule '<ISSUER>CN=Smart Card
```

```
CA,O=EXAMPLE.ORG' --maprule '(ipacertmapdata=X509:<I>{issuer_dn!nss_x500}<S>
{subject_dn!nss_x500})'
----------------------------------------------------
Added Certificate Identity Mapping Rule "rule_name"
----------------------------------------------------
  Rule name: rule_name
  Mapping rule: (ipacertmapdata=X509:<I>{issuer_dn!nss_x500}<S>{subject_dn!nss_x500})
  Matching rule: <ISSUER>CN=Smart Card CA,O=EXAMPLE.ORG
  Enabled: TRUE
```

3. 系统安全服务守护进程(SSSD)定期重新读取证书映射规则。要强制立即载入新创建的规则，请重启 SSSD：

```
# systemctl restart sssd
```

现在，您设置了证书映射规则，将它在智能卡证书中找到的映射规则中指定的数据类型与 IdM 用户条目中的证书映射数据进行比较。找到匹配项后，它将验证匹配的用户。

23.2.2.2. 在 IdM 中添加证书映射数据到用户条目

这部分论述了如何输入证书映射数据到 IdM 用户条目，以便用户可以使用多个证书进行身份验证，只要它们都包含证书映射数据条目中指定的值。

23.2.2.2.1. 在 IdM Web UI 中添加证书映射数据到用户条目

1. 以管理员身份登录 IdM Web UI。
2. 导航到 Users → Active users，再单击用户条目。
3. 找到 证书映射数据 选项，然后单击 Add。
4. 如果您有用户的证书，请由您处理：
   a. 在命令行界面中，使用 cat 工具或文本编辑器显示证书：

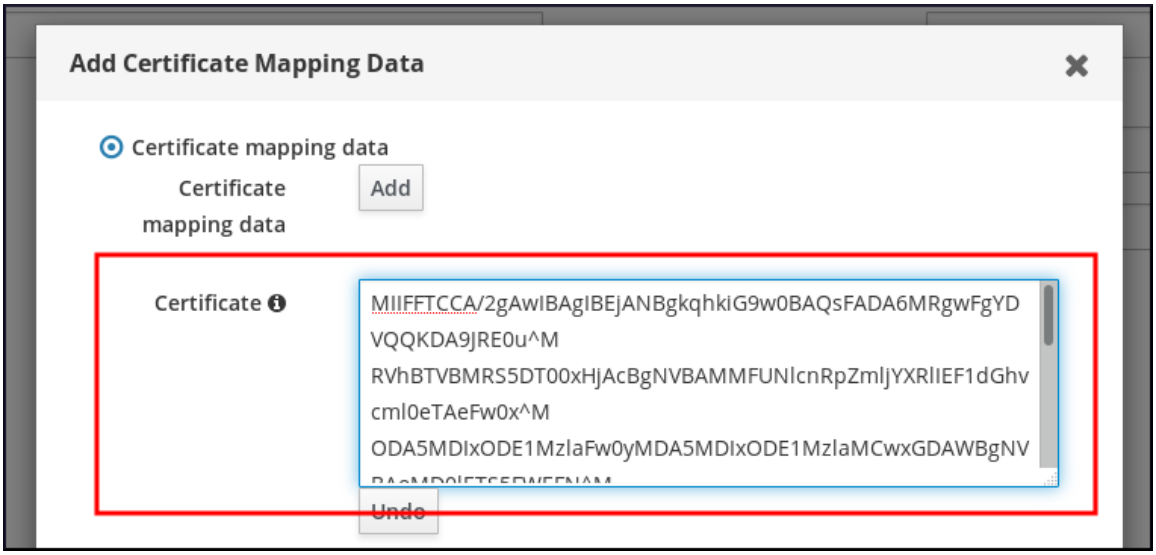
```
# [root@server ~]# cat idm_user_certificate.pem
-----BEGIN CERTIFICATE-----
MIIFFTCCA/2gAwIBAgIBEjANBgkqhkiG9w0BAQsFADA6MRgwFgYDVQQKDA9JRE0u
RVhBTVBMRS5DT00xHjAcBgNVBAMMFUNlcnRpZmljYXRlIEF1dGhvcml0eTAeFw0x
```

```
ODA5MDIxODE1MzlaFw0yMDA5MDIxODE1MzlaMCwxGDAWBgNVBAoMD0lETS5FWEFN
[...output truncated...]
```

b. 复制证书。

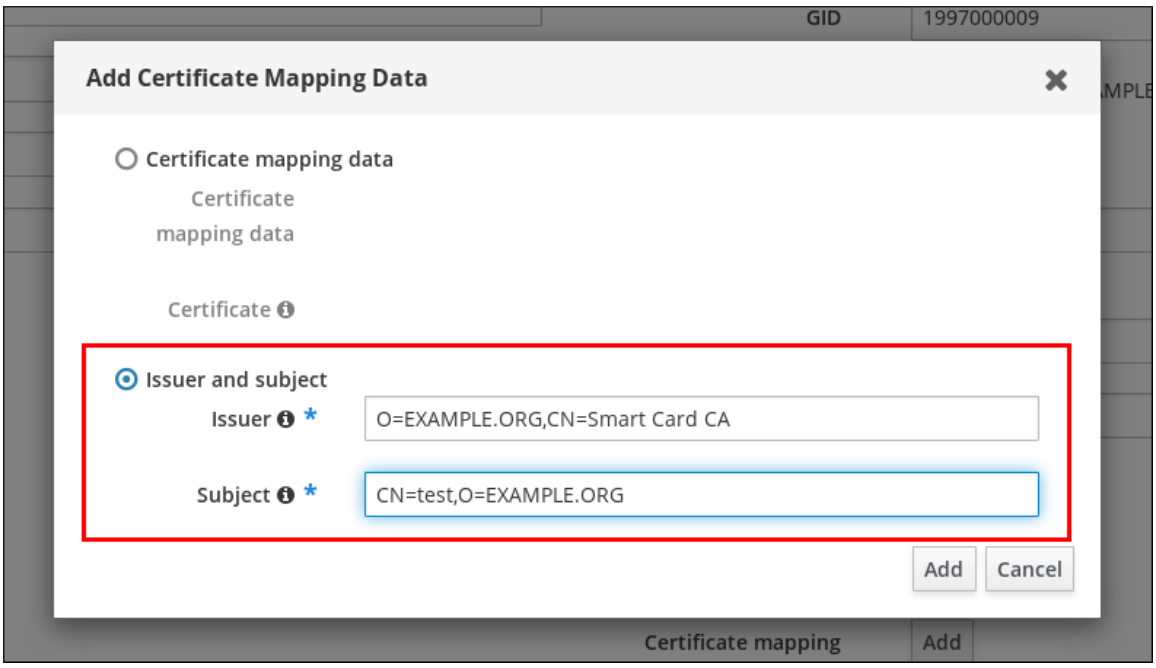
c. 在 IdM Web UI 中，单击 Certificate 旁边的 Add，并将证书粘贴到打开的窗口中。

图 23.3. 添加用户证书映射数据：证书



或者，如果您还没有用户的证书，但知道证书的 Issuer 和 Subject 信息，请检查 Issuer 和 subject 单选按钮，并在两个框中输入值。

图 23.4. 添加用户证书映射数据：签发者和主题



5. 点击 **Add**。

6. 另外，如果您能够以 **.pem** 格式访问整个证书，请验证是否已链接用户和证书：

   a. 使用 **sss_cache** 工具使 **SSSD** 缓存中用户的记录无效，并强制重新载入用户信息：

   ```
   # sss_cache -u user_name
   ```

   b. 使用包含 **IdM** 用户证书的文件名称运行 **ipa certmap-match** 命令：

   ```
   # ipa certmap-match idm_user_cert.pem
   ----------
   1 user matched
   ----------
    Domain: IDM.EXAMPLE.COM
    User logins: idm_user
   ----------------------------
   Number of entries returned 1
   ----------------------------
   ```

输出会确认您现在已向用户添加证书映射数据，且存在 第 23.2.2.1 节 “在 IdM 中添加证书映射规则” 中定义的对应映射规则。这意味着，您可以使用与定义的证书映射数据匹配的任何证书，以作为用户进行身份验证。

23.2.2.2.2. 使用命令行添加证书映射数据到用户条目

1. 获取管理员凭证：

   ```
   # kinit admin
   ```

2. 如果您有用户的证书，请使用 **ipa user-add-cert** 命令将证书添加到用户帐户中：

   ```
   # CERT=`cat idm_user_cert.pem | tail -n +2 | head -n -1 | tr -d '\r\n'\`
   # ipa user-add-certmapdata idm_user --certificate $CERT
   ```

   或者，如果您还没有用户的证书，但知道用户证书的 **Issuer** 和 **Subject** 信息：

   ```
   # ipa user-add-certmapdata idm_user --subject "O=EXAMPLE.ORG,CN=test" --issuer
   ```

```
"CN=Smart Card CA,O=EXAMPLE.ORG"
----------------------------------------------
Added certificate mappings to user "idm_user"
----------------------------------------------
  User login: idm_user
  Certificate mapping data: X509:<I>O=EXAMPLE.ORG,CN=Smart Card
CA<S>CN=test,O=EXAMPLE.ORG
```

3. 另外，如果您能够以 .pem 格式访问整个证书，请验证是否已链接用户和证书：

   a. 使用 sss_cache 工具使 SSSD 缓存中用户的记录无效，并强制重新载入用户信息：

   ```
   # sss_cache -u user_name
   ```

   b. 使用包含 IdM 用户证书的文件名称运行 ipa certmap-match 命令：

   ```
   # ipa certmap-match idm_user_cert.pem
   --------------
   1 user matched
   --------------
    Domain: IDM.EXAMPLE.COM
    User logins: idm_user
   ----------------------------
   Number of entries returned 1
   ----------------------------
   ```

23.2.3. 为用户配置证书映射，Whose AD User Entry 包含 Whole Certificate

本节论述了在 IdM 部署中信任 Active Directory(AD)启用证书映射所需的步骤，用户存储在 AD 中，AD 中的用户条目包含整个证书。

先决条件

- 用户在 IdM 中没有帐户。
- 用户在 AD 中有一个包含证书的帐户。
- IdM 管理员有权访问 IdM 证书映射规则可以基于的数据。

23.2.3.1. 为用户添加证书映射规则，而 Whose AD User Entry 包含使用 IdM Web UI 的 Whole Certificate

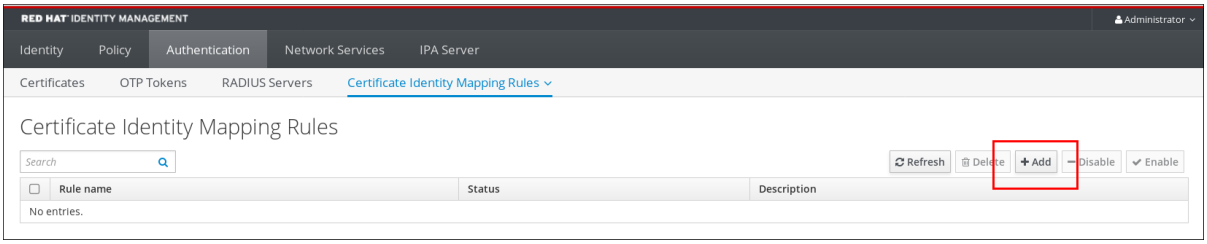
在 IdM Web UI 中添加证书映射规则：

1. 以管理员身份登录 IdM Web UI。

2. 导航到 Authentication → Certificate Identity Mapping Rules → Certificate Identity Mapping Rules。

3. 点击 Add。

图 23.5. 在 IdM Web UI 中添加新证书映射规则



4. 输入规则名称。

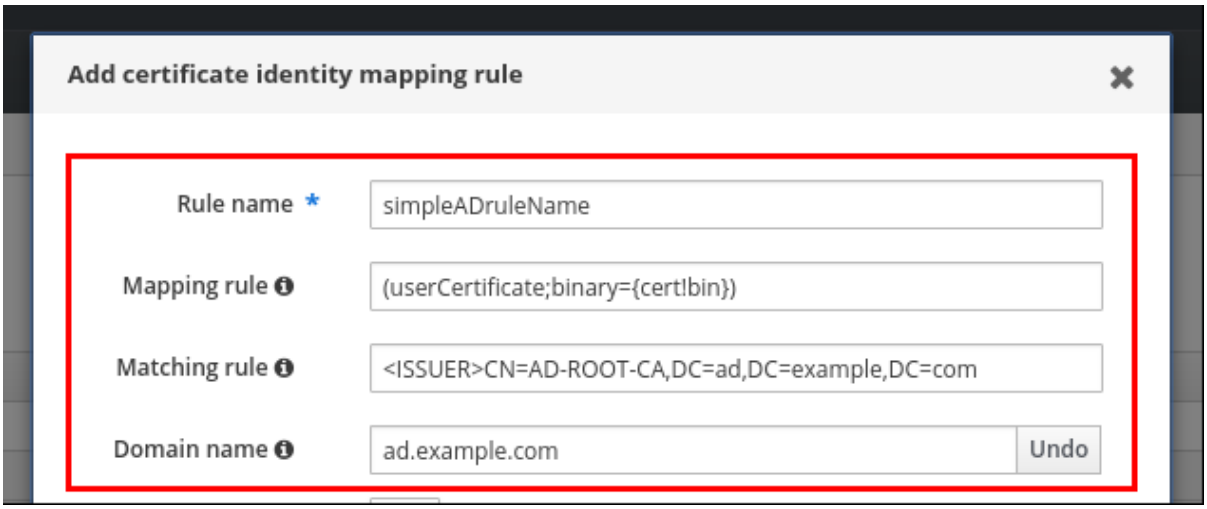
5. 输入映射规则。与 AD 中可用内容相比，有向 IdM 呈现的完整证书进行身份验证：

```
(userCertificate;binary={cert!bin})
```

6. 输入匹配的规则。例如，只允许 AD.EXAMPLE.COM 域的 AD-ROOT-CA 发布的证书进行身份验证：

```
<ISSUER>CN=AD-ROOT-CA,DC=ad,DC=example,DC=com
```

图 23.6. 为在 AD 中存储证书的用户映射证书映射规则



7. *点击 **Add**。*

8. *系统安全服务守护进程(SSSD)定期重新读取证书映射规则。要强制立即载入新创建的规则，请重启 **SSSD**：*

```
# systemctl restart sssd
```

23.2.3.2. 为用户 Whose AD User Entry 添加证书映射规则，使用命令行包含整个证书

使用命令行添加证书映射规则：

1. *获取管理员凭证：*

```
# kinit admin
```

2. *输入映射规则，以及映射规则所基于的匹配规则。要获得与 AD 中可用证书相比的用于身份验证的完整证书，只允许 AD.EXAMPLE.COM 域的 AD-ROOT-CA 发布的证书来进行验证：*

```
# ipa certmaprule-add simpleADrule --matchrule '<ISSUER>CN=AD-ROOT-
CA,DC=ad,DC=example,DC=com' --maprule '(userCertificate;binary={cert!bin})' --domain
ad.example.com
----------------------------------------------------
Added Certificate Identity Mapping Rule "simpleADrule"
----------------------------------------------------
  Rule name: simpleADrule
  Mapping rule: (userCertificate;binary={cert!bin})
```

```
Matching rule: <ISSUER>CN=AD-ROOT-CA,DC=ad,DC=example,DC=com
Domain name: ad.example.com
Enabled: TRUE
```

3. 系统安全服务守护进程(SSSD)定期重新读取证书映射规则。要强制立即载入新创建的规则，请重启 SSSD：

```
# systemctl restart sssd
```

23.2.4. 如果将 AD 配置为将用户证书映射到用户帐户，则配置证书映射

本节论述了在 IdM 部署中信任 Active Directory(AD)启用证书映射所需的步骤，用户存储在 AD 中，AD 中的用户条目包含证书映射数据。

前提条件

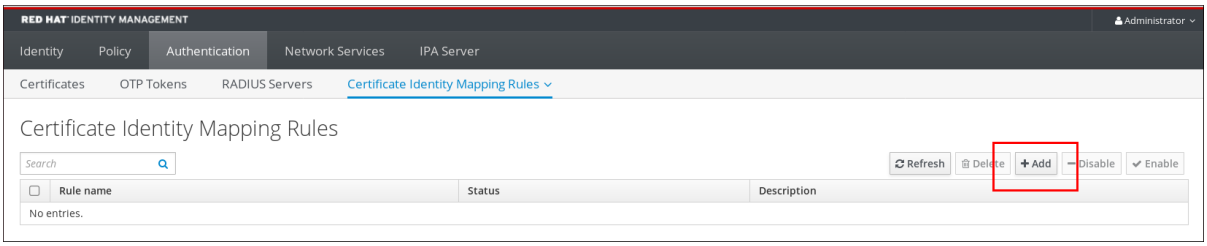
- 用户在 IdM 中没有帐户。
- 用户在 AD 中有一个帐户，其中包含 altSecurityIdentities 属性，即 IdM certmapdata 属性的 AD 等效。
- IdM 管理员有权访问 IdM 证书映射规则可以基于的数据。

23.2.4.1. 如果受信任的 AD 域被配置为映射用户证书，则使用 Web UI 添加证书映射规则

如果可信 AD 域被配置为映射用户证书，请添加证书映射规则：

1. 以管理员身份登录 IdM Web UI。
2. 导航到 Authentication → Certificate Identity Mapping Rules → Certificate Identity Mapping Rules。
3. 点击 Add。

图 23.7. 在 IdM Web UI 中添加新证书映射规则



4. *输入规则名称。*

5. *输入映射规则。例如，要使 AD DC 搜索所呈现的任何证书中的 Issuer 和 Subject 条目，并决定根据所出示证书的两个条目中提供的信息进行验证：*

```
(altSecurityIdentities=X509:<I>{issuer_dn!ad_x500}<S>{subject_dn!ad_x500})
```

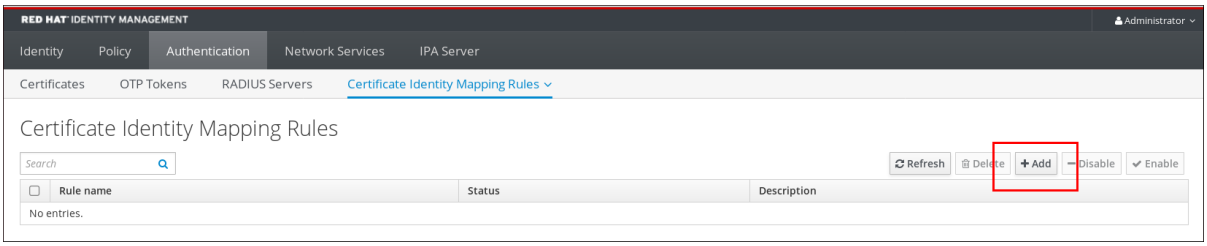
6. *输入匹配的规则。例如，要只允许 AD.EXAMPLE.COM 域的 AD-ROOT-CA 发布的证书来向 IdM 验证用户：*

```
<ISSUER>CN=AD-ROOT-CA,DC=ad,DC=example,DC=com
```

7. *输入域：*

```
ad.example.com
```

图 23.8. 如果为映射配置了 AD，则证书映射规则



8. *点击 Add。*

9. *系统安全服务守护进程(SSSD)定期重新读取证书映射规则。要强制立即载入新创建的规则，请重启 SSSD：*

```
# systemctl restart sssd
```

23.2.4.2. 如果将受信任的 AD 域配置为映射用户证书，则使用命令行添加证书映射规则

使用命令行添加证书映射规则：

1. 获取管理员凭证：

```
# kinit admin
```

2. 输入映射规则，以及映射规则所基于的匹配规则。例如，要让 AD 搜索所提供的任何证书中的 Issuer 和 Subject 条目，并且只允许 AD.EXAMPLE.COM 域的 AD-ROOT-CA 发布的证书：

```
# ipa certmaprule-add ad_configured_for_mapping_rule --matchrule '<ISSUER>CN=AD-
ROOT-CA,DC=ad,DC=example,DC=com' --maprule '(altSecurityIdentities=X509:<I>
{issuer_dn!ad_x500}<S>{subject_dn!ad_x500})' --domain=ad.example.com
----------------------------------------------------
Added Certificate Identity Mapping Rule "ad_configured_for_mapping_rule"
----------------------------------------------------
  Rule name: ad_configured_for_mapping_rule
  Mapping rule: (altSecurityIdentities=X509:<I>{issuer_dn!ad_x500}<S>
{subject_dn!ad_x500})
  Matching rule: <ISSUER>CN=AD-ROOT-CA,DC=ad,DC=example,DC=com
  Domain name: ad.example.com
  Enabled: TRUE
```

3. 系统安全服务守护进程(SSSD)定期重新读取证书映射规则。要强制立即载入新创建的规则，请重启 SSSD：

```
# systemctl restart sssd
```

23.2.4.3. 检查 AD 分片中的证书映射数据

altSecurityIdentities 属性是与 IdM 中的 certmapdata 用户属性等效的 Active Directory(AD)。当将可信 AD 域配置为将用户帐户映射到用户帐户时，IdM 系统管理员需要检查 AD 中的用户条目是否正确设置了 altSecurityIdentities 属性。

要检查 AD 是否包含 AD 中存储的用户的正确信息，请使用 ldapsearch 命令。

例如，要检查 adserver. ad.example.com 服务器，使得 altSecurityIdentities 属性在 ad_user 的用户条目中设置，并且 matchrule 认为 ad_user 用来向 AD 进行身份验证的证书是由 ad.example.com

域的 AD-ROOT-CA 以及主题为 <S<> DC=com 签发的。
DC=example,DC=ad,CN=Users,CN=ad_user ：

```
$ ldapsearch -o ldif-wrap=no -LLL -h adserver.ad.example.com \
-p 389 -D cn=Administrator,cn=users,dc=ad,dc=example,dc=com \
-W -b cn=users,dc=ad,dc=example,dc=com "(cn=ad_user)" \
altSecurityIdentities
Enter LDAP Password:
dn: CN=ad_user,CN=Users,DC=ad,DC=example,DC=com
altSecurityIdentities: X509:<I>DC=com,DC=example,DC=ad,CN=AD-ROOT-
CA<S>DC=com,DC=example,DC=ad,CN=Users,CN=ad_user
```

23.2.5. 如果 AD 用户输入不包含证书或映射数据，则配置证书映射

本节论述了在 IdM 部署中信任 Active Directory(AD)时启用证书映射所需的步骤，用户存储在 AD 中，AD 中的用户条目既不包含整个证书，也不包含整个证书或证书映射数据。

先决条件

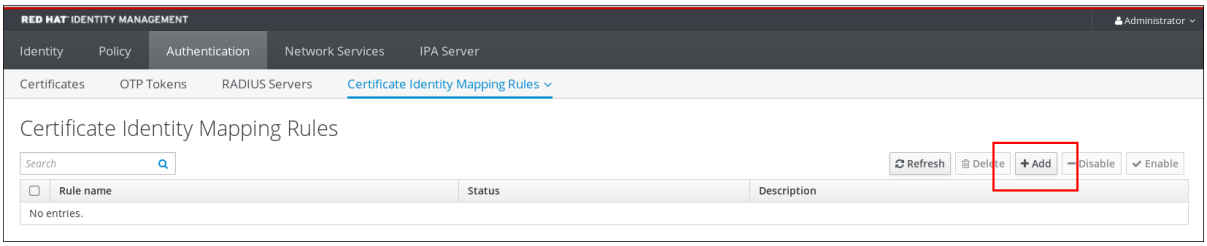
- 用户在 IdM 中没有帐户。
- 用户在 AD 中有一个帐户，它不包含整个证书和 altSecurityIdentities 属性，即 IdM certmapdata 属性的 AD 等效。
- IdM 管理员具有整个 AD 用户证书，用于在 IdM 中添加到 AD 用户的用户 ID 覆盖 中。

23.2.5.1. 如果 AD 用户条目不包含证书或映射数据，则使用 Web UI 添加证书映射规则

如果 AD 用户条目不包含证书或映射数据，则使用 web UI 添加证书映射规则：

1. 以管理员身份登录 IdM Web UI。
2. 导航到 Authentication → Certificate Identity Mapping Rules → Certificate Identity Mapping Rules。
3. 点击 Add。

图 23.9. 在 IdM Web UI 中添加新证书映射规则



4. *输入规则名称。*

5. *输入映射规则。与 IdM 中 AD 用户条目的用户 ID 覆盖条目中存储的证书相比，要让 IdM 为 IdM 提供的整个证书进行身份验证：*

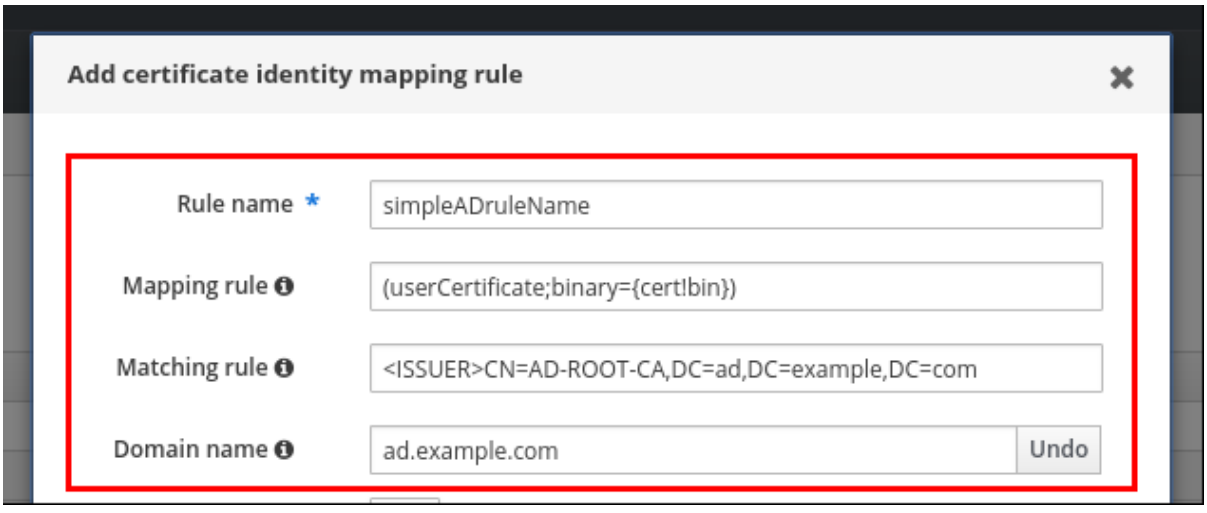
```
(userCertificate;binary={cert!bin})
```

6. *输入匹配的规则。例如，只允许 AD.EXAMPLE.COM 域的 AD-ROOT-CA 发布的证书进行身份验证：*

```
<ISSUER>CN=AD-ROOT-CA,DC=ad,DC=example,DC=com
```

7. *输入域名。例如，要在 ad.example.com 域中搜索用户：*

图 23.10. 无证书的用户的证书映射规则或在 AD 中映射数据存储



8. *点击 Add。*

9. *系统安全服务守护进程(SSSD)定期重新读取证书映射规则。要强制立即载入新创建的规则，请重启 SSSD：*

```
# systemctl restart sssd
```

23.2.5.2. 如果 AD 用户条目不包含证书或映射数据，则使用命令行添加证书映射规则

如果 AD 用户条目不包含证书或映射数据，请使用命令行添加证书映射规则：

1. 获取管理员凭证：

```
# kinit admin
```

2. 输入映射规则，以及映射规则所基于的匹配规则。要获得与存储在 IdM 中的 AD 用户条目的用户 ID 覆盖条目中的证书相比的用于认证的整个证书，只允许 AD.EXAMPLE.COM 域的 AD-ROOT-CA 发布的证书进行验证：

```
# ipa certmaprule-add simpleADrule --matchrule '<ISSUER>CN=AD-ROOT-
CA,DC=ad,DC=example,DC=com' --maprule '(userCertificate;binary={cert!bin})' --domain
ad.example.com
----------------------------------------------------------
Added Certificate Identity Mapping Rule "simpleADrule"
----------------------------------------------------------
  Rule name: simpleADrule
  Mapping rule: (userCertificate;binary={cert!bin})
  Matching rule: <ISSUER>CN=AD-ROOT-CA,DC=ad,DC=example,DC=com
  Domain name: ad.example.com
  Enabled: TRUE
```

3. 系统安全服务守护进程(SSSD)定期重新读取证书映射规则。要强制立即载入新创建的规则，请重启 SSSD：

```
# systemctl restart sssd
```

23.2.5.3. 使用 Web UI 将证书添加到 AD 用户的 ID 覆盖中

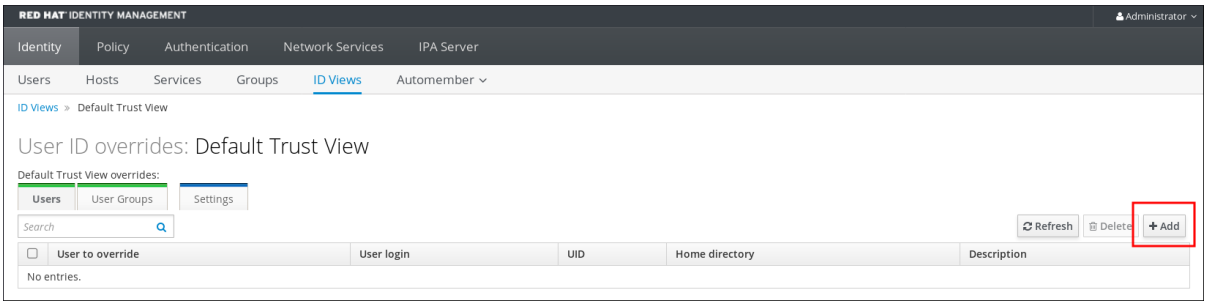
如果 AD 中的用户条目不包含证书或映射数据，则使用 web UI 将证书添加到 AD 用户的 ID 覆盖中：

1. 以管理员身份登录 IdM Web UI。

2. *导航到 Identity → ID Views → Default Trust View。*

3. *点击 Add。*

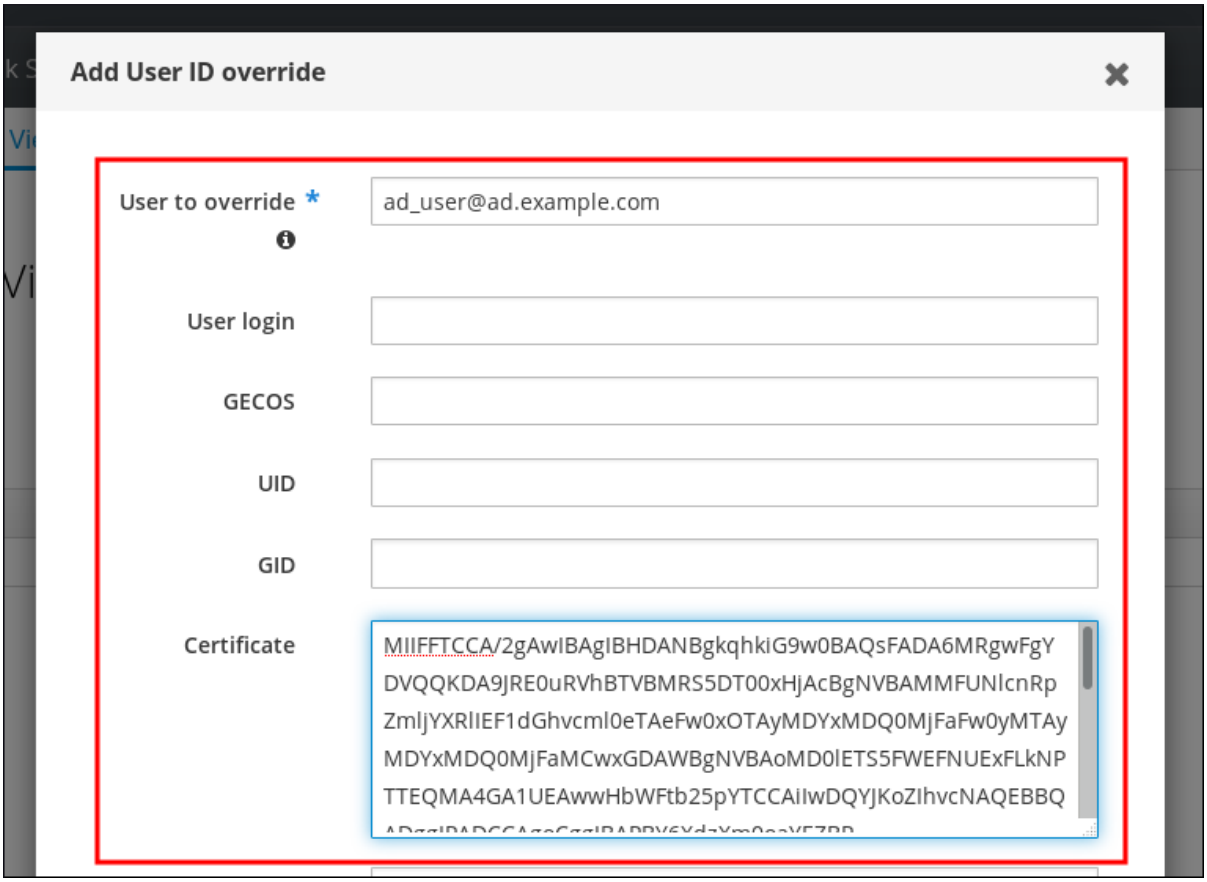
图 23.11. 在 IdM Web UI 中添加新用户 ID 覆盖



4. *在 User to override 字段中，输入以下格式的用户名：user_name@domain_name*

5. *将用户的证书复制并粘贴到 Certificate 字段中。*

图 23.12. 为 AD 用户配置用户 ID 覆盖



6. *（可选）验证用户和证书是否已链接：*

a. 使用 sss_cache 工具使 SSSD 缓存中用户的记录无效，并强制重新载入用户信息：

```
# sss_cache -u ad_user@ad.example.com
```

b. 输入 ipa certmap-match 命令，以及包含 AD 用户证书的文件的名称：

```
# ipa certmap-match ad_user_cert.pem
--------------
1 user matched
--------------
 Domain: AD.EXAMPLE.COM
 User logins: ad_user@ad.example.com
----------------------------
Number of entries returned 1
----------------------------
```

输出确认您将证书映射数据添加到 ad_user@ad.example.com，并且存在对应的映射规则。这意味着，您可以使用与定义的证书映射数据匹配的证书作为 ad_user@ad.example.com 进行身份验证。

23.2.5.4. 使用命令行在 AD 用户的 ID 覆盖中添加证书

如果 AD 中的用户条目不包含证书或映射数据，则使用命令行在 AD 用户的 ID 覆盖中添加证书：

1. 获取管理员凭证：

```
# kinit admin
```

2. 使用 ipa idoverrideuser-add-cert 命令将用户的证书添加到用户帐户中：

```
# CERT=`cat ad_user_cert.pem | tail -n +2 | head -n -1 | tr -d '\r\n'\`
# ipa idoverrideuser-add-cert ad_user@ad.example.com --certificate $CERT
```

3. （可选）验证用户和证书是否已链接：

a. 使用 sss_cache 工具使 SSSD 缓存中用户的记录无效，并强制重新载入用户信息：

```
# sss_cache -u ad_user@ad.example.com
```

b. 输入 ipa certmap-match 命令，以及包含 AD 用户证书的文件的名称：

```
# ipa certmap-match ad_user_cert.pem
--------------
1 user matched
--------------
 Domain: AD.EXAMPLE.COM
 User logins: ad_user@ad.example.com
----------------------------
Number of entries returned 1
----------------------------
```

输出确认您将证书映射数据添加到 ad_user@ad.example.com，并且存在对应的映射规则。这意味着，您可以使用与定义的证书映射数据匹配的证书作为 ad_user@ad.example.com 进行身份验证。

23.2.6. 将 Several Identity Mapping 规则合并到一个

要将多个身份映射规则组合成一个组合规则，请使用 | （或）字符在单个映射规则前前，并使用 （） 方括号将它们分隔，例如：

```
$ ipa certmaprule-add ad_cert_for_ipa_and_ad_users \ --maprule='(|(ipacertmapdata=X509:<I>
{issuer_dn!nss_x500}<S>{subject_dn!nss_x500})(altSecurityIdentities=X509:<I>{issuer_dn!ad_x500}
<S>{subject_dn!ad_x500}))' \ --matchrule='<ISSUER>CN=AD-ROOT-
CA,DC=ad,DC=example,DC=com' \ --domain=ad.example.com
```

在上例中，--maprule 选项中的过滤器定义包含以下条件：

- ipacertmapdata=X509:<I>{issuer_dn!nss_x500}<S>{ subject_dn!nss_x500} 是一个过滤器，它将主题和签发者从智能卡证书链接到 IdM 用户帐户中的 ipacertmapdata 属性的值，如 第 23.2.2.1 节 “在 IdM 中添加证书映射规则” 所述。
- altSecurityIdentities=X509:<I>{issuer_dn!ad_x500}<S>{ subject_dn!ad_x500} 是一个过滤器，它将主题和签发者从智能卡证书链接到 AD 用户帐户中的 altSecurityIdentities 属性的值，如 第 23.2.4 节 “如果将 AD 配置为将用户证书映射到用户帐户，则配置证书映射” 所述。
- 添加 --domain=ad.example.com 选项意味着映射到给定证书的用户不仅在本地 idm.example.com 域中搜索，也在 ad.example.com 域中搜索。

--maprule 选项中的过滤器定义接受逻辑运算符 |（或），以便您可以指定多个条件。在这种情况下，规则会映射至少满足其中一个条件的所有用户帐户。

```
$ ipa certmaprule-add ipa_cert_for_ad_users \
  --maprule='(|(userCertificate;binary={cert!bin})(ipacertmapdata=X509:<I>{issuer_dn!nss_x500}<S>
{subject_dn!nss_x500})(altSecurityIdentities=X509:<I>{issuer_dn!ad_x500}<S>
{subject_dn!ad_x500}))' \
  --matchrule='<ISSUER>CN=Certificate Authority,O=REALM.EXAMPLE.COM' \
  --domain=idm.example.com --domain=ad.example.com
```

在上例中，--maprule 选项中的过滤器定义包含以下条件：

- userCertificate;binary={cert!bin} 是一个过滤器，它返回包含整个证书的用户条目。对于 AD 用户，第 23.2.5 节 “如果 AD 用户输入不包含证书或映射数据，则配置证书映射” 中详细介绍了创建这种过滤器类型。

- ipacertmapdata=X509:<I>{issuer_dn!nss_x500}<S>{ subject_dn!nss_x500} 是一个过滤器，它将主题和签发者从智能卡证书链接到 IdM 用户帐户中的 ipacertmapdata 属性的值，如第 23.2.2.1 节 “在 IdM 中添加证书映射规则” 所述。

- altSecurityIdentities=X509:<I>{issuer_dn!ad_x500}<S>{ subject_dn!ad_x500} 是一个过滤器，它将主题和签发者从智能卡证书链接到 AD 用户帐户中的 altSecurityIdentities 属性的值，如 第 23.2.4 节 “如果将 AD 配置为将用户证书映射到用户帐户，则配置证书映射” 所述。

--maprule 选项中的过滤器定义接受逻辑运算符 |（或），以便您可以指定多个条件。在这种情况下，规则会映射至少满足其中一个条件的所有用户帐户。

23.3. 使用智能卡向身份管理客户端进行身份验证

作为身份管理服务器中具有多个角色帐户的身份管理用户，您可以使用智能卡验证加入到身份管理域的桌面客户端系统。这可让您将客户端系统用作所选角色。

有关支持的选项的基本概述，请参阅：

- 第 23.3.1 节 “身份管理客户端上支持的基于智能卡的身份验证选项”

有关配置环境以启用身份验证的详情，请参考：

- 第 23.3.2 节 “为智能卡身份验证准备身份管理客户端”

有关如何验证的详情请参考：

- 第 23.3.3 节 “使用智能卡在身份管理客户端中进行身份验证，使用控制台登录”

23.3.1. 身份管理客户端上支持的基于智能卡的身份验证选项

在使用身份管理客户端上的智能卡进行身份验证时，身份管理中的用户可以使用以下选项：

本地验证

本地验证包括以下身份验证：

- 文本控制台
- 图形控制台，如 Gnome Display Manager(GDM)
- 本地身份验证服务，如 su 或 sudo

使用 ssh 进行远程身份验证

智能卡中的证书与受 PIN 保护的 SSH 私钥一起存储。

不支持使用其他服务（如 FTP）进行基于智能卡的验证。

23.3.2. 为智能卡身份验证准备身份管理客户端

作为身份管理管理员，执行以下步骤：

1. 在服务器上，创建一个 shell 脚本来配置客户端。

   a. 使用 ipa-advise config-client-for-smart-card-auth 命令，并将其输出保存到文件中：

   ```
   # ipa-advise config-client-for-smart-card-auth > client_smart_card_script.sh
   ```

   b. 打开 脚本文件，并检查其内容。

   c. 使用 chmod 实用程序为文件添加执行权限：

   ```
   # chmod +x client_smart_card_script.sh
   ```

2. 将 脚本复制到客户端并运行它。使用签署智能卡证书的证书颁发机构(CA)添加 PEM 文件的路径：

   ```
   # ./client_smart_card_script.sh CA_cert.pem
   ```

另外，如果外部证书颁发机构(CA)在智能卡上签名证书，请将智能卡 CA 添加为可信 CA：

1. 在 Identity Management 服务器中安装 CA 证书：

   ```
   # ipa-cacert-manage -n "SmartCard CA" -t CT,C,C install ca.pem
   # ipa-certupdate
   ```

   在所有副本和客户端上也重复 ipa-certupdate。

2. 重启 HTTP 服务器：

   ```
   # systemctl restart httpd
   ```

   在所有副本上重复 systemctl restart httpd。

注意

SSSD 可让管理员使用 certificate_verification 参数调整证书验证过程，例如，如果证书中定义的在线证书状态协议(OCSP)服务器无法从客户端访问。如需更多信息，请参阅 sssd.conf(5) man page。

23.3.3. 使用智能卡在身份管理客户端中进行身份验证，使用控制台登录

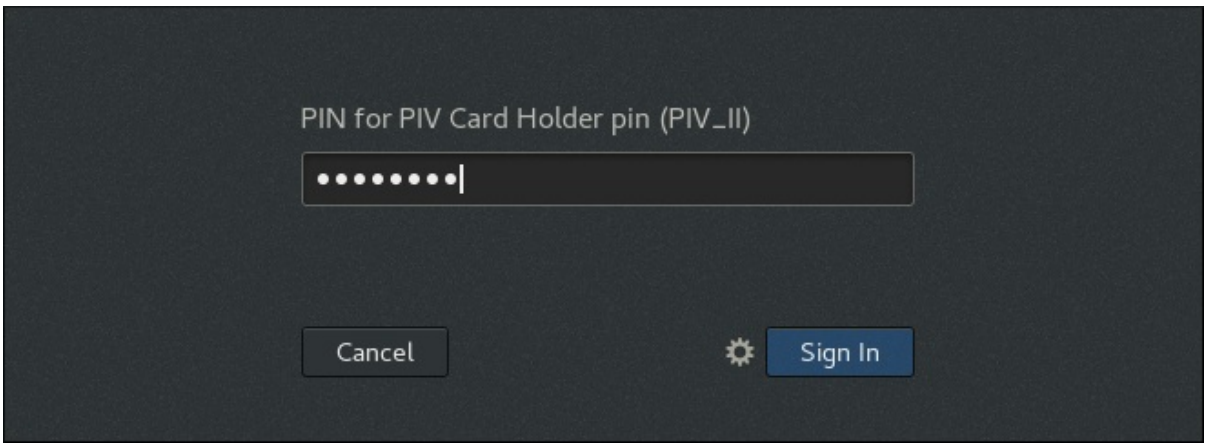
要以 Identity Management 用户身份进行身份验证，请输入用户名和 PIN。

- **从命令行登录时：**

```
client login: idm_user
PIN for PIV Card Holder pin (PIV_II) for user idm_user@idm.example.com:
```

- **使用 Gnome Desktop Manager(GDM)登录时，GDM 会在选择所需用户后提示您输入智能卡 PIN：**

图 23.13. 在 Gnome Desktop Manager 中输入智能卡 PIN



要以 Active Directory 用户身份进行身份验证，请使用 NetBIOS 域名的格式输入用户名：AD.EXAMPLE.COM\ad_user or ad_user@AD.EXAMPLE.COM.

如果身份验证失败，请参阅 第 A.4 节 “调查智能卡身份验证失败”。

23.3.4. 从本地系统向远程系统进行身份验证

在本地系统中执行以下步骤：

1. 插入智能卡。

2. 启动 ssh，并使用 -I 选项指定 PKCS the 库：

   - 作为身份管理用户：

     ```
     $ ssh -I /usr/lib64/opensc-pkcs11.so -l idm_user server.idm.example.com

     Enter PIN for 'PIV_II (PIV Card Holder pin)':
     Last login: Thu Apr  6 12:49:32 2017 from 10.36.116.42
     ```

   - 作为 Active Directory 用户：

     ```
     $ ssh -I /usr/lib64/opensc-pkcs11.so -l ad_user@ad.example.com
     server.idm.example.com

     Enter PIN for 'PIV_II (PIV Card Holder pin)':
     Last login: Thu Apr  6 12:49:32 2017 from 10.36.116.42
     ```

3. 可选。使用 id 实用程序检查您是否以预期用户身份登录。

   - 作为身份管理用户：

     ```
     $ id
     uid=1928200001(idm_user) gid=1928200001(idm_user) groups=1928200001(idm_user)
     context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
     ```

   - 作为 Active Directory 用户：

     ```
     $ id
     uid=1171201116(ad_user@ad.example.com)
     gid=1171201116(ad_user@ad.example.com)
     groups=1171201116(ad_user@ad.example.com),1171200513(domain
     users@ad.example.com) context=unconfined_u:unconfined_r:unconfined_t:s0-
     s0:c0.c1023
     ```

如果身份验证失败，请参阅 第 A.4 节 “调查智能卡身份验证失败”。

23.3.5. 其它资源

- 使用带有智能卡的 ssh 的身份验证不会在远程系统上获得票据授予票据(TGT)。要在远程系统上获取 TGT，管理员必须在本地系统上配置 Kerberos 并启用 Kerberos 委派。有关所需配置的示例，请查看 此 Kerberos 知识库条目。

- 有关使用 OpenSSH 的智能卡验证的详情，请参考 安全指南中的使用智能卡为 OpenSSH 提供凭证 。

23.4. 为智能卡身份验证配置用户名 HINT 策略

作为身份管理管理员，您可以为与多个帐户关联的智能卡配置 用户名提示 策略。

23.4.1. 身份管理中的用户名提示

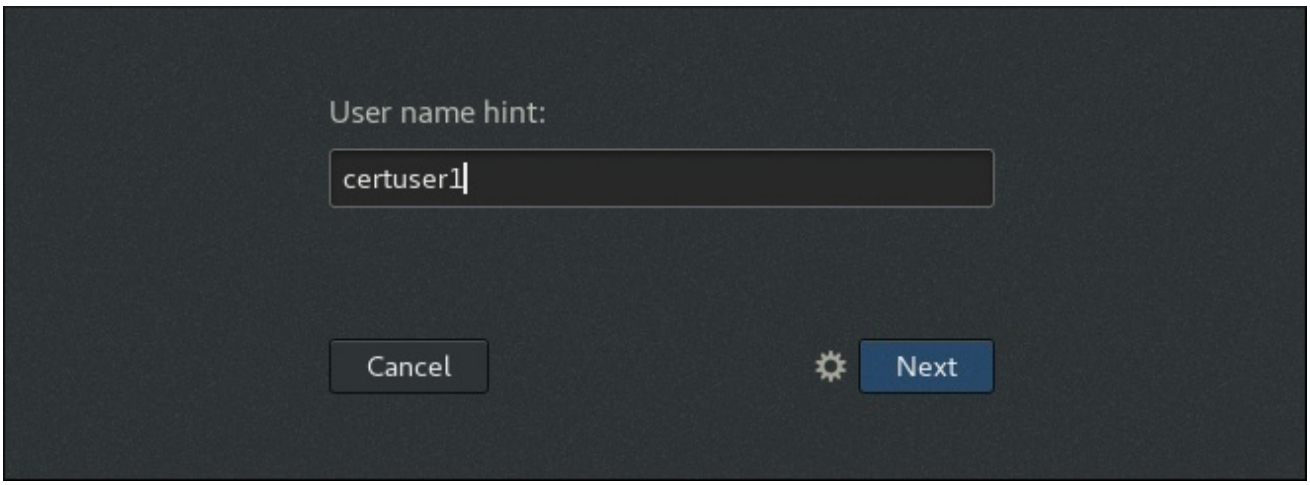
用户名提示策略将 Identity Management 配置为提示智能卡用户输入其用户名。当用户使用与身份管理中的多个用户帐户匹配的智能卡证书验证时，会出现以下情况之一：

- 如果启用了用户名提示策略，系统会提示用户输入用户名，然后就可以进行身份验证。

- 如果禁用了用户名提示策略，身份验证会在不提示的情况下失败。

身份管理将用户名提示添加到默认提示输入智能卡 PIN 而不要求用户名的应用程序。在 Red Hat Enterprise Linux 中，这仅是 Gnome Desktop Manager(GDM)登录名。

图 23.14. Gnome Desktop Manager 中的用户名提示



默认情况下，身份管理不会将用户名提示添加到要求输入用户名的应用程序中，例如：

- Identity Management Web UI 身份验证，因为 GUI 始终显示 Username 字段
- SSH 身份验证，因为 ssh 使用当前用户的登录名称或 -l 选项提供的名称或 username@host 格式
- 控制台身份验证，其中提供登录名称

在这些情况下，始终允许使用与多个用户匹配的证书进行身份验证。

23.4.2. 在身份管理中启用用户名提示

Identity Management 管理员会集中设置用户名提示策略。该策略适用于注册到身份管理域中的所有主机。

在任何身份管理系统上执行这些步骤。

命令行：在身份管理中启用用户名提示

1. 以 Identity Management 管理员身份登录：

```
$ kinit admin
Password for admin@IDM.EXAMPLE.COM:
```

2. 使用带有 --promptusername=True 选项的 ipa certmapconfig-mod 命令启用用户名提示。

```
$ ipa certmapconfig-mod --promptusername=TRUE
Prompt for the username: TRUE
```

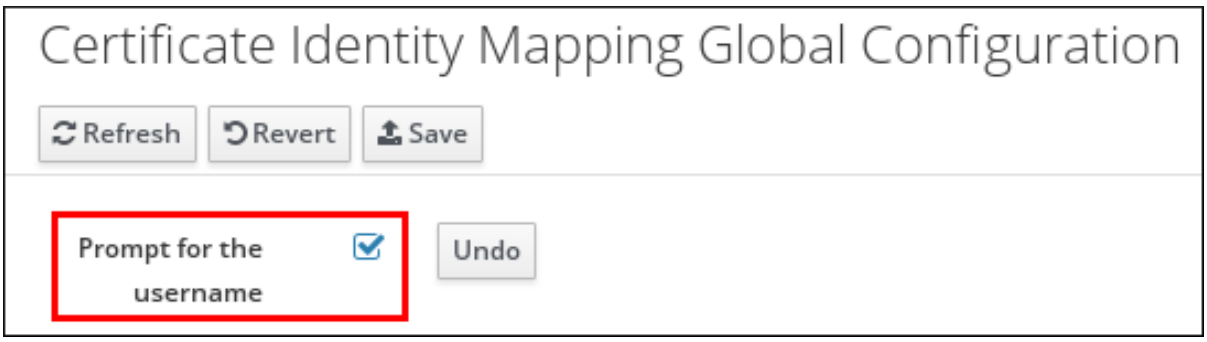
要禁用用户名提示，请使用 --promptusername=False 选项。

Web UI：在身份管理中启用用户名提示

1. 单击 **Authentication → Certificate Identity Mapping Rules → Certificate Identity Mapping Global Configuration**。

2. 选择 **Prompt** 作为用户名，然后单击 **Save** 。

图 **23.15.** 在 **Web UI** 中启用用户名提示



其它资源

- 有关 **ipa certmapconfig-mod** 命令的详情，请使用 **--help** 选项执行它。

23.5. 身份管理中的 PKINIT 智能卡身份验证

身份管理用户可以在加入身份管理的桌面客户端系统中使用智能卡进行身份验证，并自动获得 **Kerberos** 票据授权票据(TGT)。用户可以使用该票据从客户端进行进一步的单点登录(SSO)身份验证。

23.5.1. 为 PKINIT 身份验证准备身份管理客户端

作为身份管理管理员，请在您希望用户进行身份验证的客户端上执行以下步骤：

1. 在服务器上，创建一个 **shell** 脚本来配置客户端。

   a. 使用 **ipa-advise config-client-for-smart-card-auth** 命令，并将其输出保存到文件中：

   ```
   # ipa-advise config-client-for-smart-card-auth > client_smart_card_script.sh
   ```

   b. 打开 脚本文件，并检查其内容。

c.

使用 chmod 实用程序为文件添加执行权限：

```
# chmod +x client_smart_card_script.sh
```

2.

将 脚本复制到客户端并运行它。使用签署智能卡证书的证书颁发机构(CA)添加 PEM 文件的路径：

```
# ./client_smart_card_script.sh CA_cert.pem
```

3.

确保已安装 krb5-pkinit 软件包。

另外，如果外部证书颁发机构(CA)在智能卡上签名证书，请将智能卡 CA 添加为可信 CA：

1.

在 Identity Management 服务器中安装 CA 证书：

```
# ipa-cacert-manage -n "SmartCard CA" -t CT,C,C install ca.pem
# ipa-certupdate
```

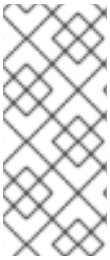
在所有副本和客户端上也重复 ipa-certupdate。

2.

重启 HTTP 服务器：

```
# systemctl restart httpd
```

在所有副本上重复 systemctl restart httpd。



注意

SSSD 可让管理员使用 certificate_verification 参数调整证书验证过程，例如，如果证书中定义的在线证书状态协议(OCSP)服务器无法从客户端访问。如需更多信息，请参阅 sssd.conf(5) man page。

23.5.2. 作为身份管理用户：在身份管理客户端上使用 PKINIT 进行身份验证

在身份管理客户端中使用 kinit 工具进行身份验证：

```
$ kinit -X X509_user_identity='PKCS11:opensc-pkcs11.so' idm_user
```

X 选项将 opensc-pkcs11.so 模块指定为预身份验证属性。详情请查看 kinit(1) man page。

23.5.3. 作为 Active Directory 用户：在身份管理客户端上使用 PKINIT 进行身份验证

先决条件

以管理员身份，配置环境以支持 Active Directory 用户的 PKINIT 身份验证：

- 配置 Active Directory 服务器，以信任签发智能卡证书的证书颁发机构(CA)。在 NTAuth 存储中导入 CA（请参阅 Microsoft 支持），并将 CA 添加为可信 CA。详情请参阅 Active Directory 文档。

- 将 Kerberos 客户端配置为信任发布智能卡证书的 CA：

  1. 在身份管理客户端上，打开 /etc/krb5.conf 文件。

  2. 在该文件中添加以下行：

     ```
     [libdefaults]
     [... file truncated ...]
      pkinit_eku_checking = kpServerAuth
      pkinit_kdc_hostname = adserver.ad.domain.com
     ```

- 如果用户证书不包含证书撤销列表(CRL)分发点扩展，请配置 Active Directory 以忽略撤销错误：

  1. 将以下 REG 格式的内容保存到纯文本文件中，然后双击该文件将其导入到 Windows 注册表：

     ```
     Windows Registry Editor Version 5.00

     [HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Kdc]
     "UseCachedCRLOnlyAndIgnoreRevocationUnknownErrors"=dword:00000001
     ```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\LSA\Kerberos\Parameters]
"UseCachedCRLOnlyAndIgnoreRevocationUnknownErrors"=dword:00000001
```

或者，使用 regedit.exe 应用程序手动设置值。

2. 重启 Windows 系统以应用更改。

步骤

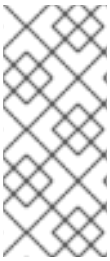
在身份管理客户端上使用 kinit 工具进行身份验证。使用用户名和域名指定 Active Directory 用户：

```
$ kinit -X X509_user_identity='PKCS11:opensc-pkcs11.so' ad_user@AD.DOMAIN.COM
```

X 选项将 opensc-pkcs11.so 模块指定为预身份验证属性。详情请查看 kinit(1) man page。

23.6. 使用智能卡验证身份管理 WEB UI

作为在身份管理服务器中拥有多个角色帐户的 Identity Management 用户，您可以使用智能卡作为所选角色与身份管理 Web UI 进行身份验证。这可让您将 Web UI 用作所选角色。



注意

只有身份管理用户才能使用智能卡登录 Web UI。Active Directory 用户可以使用其用户名和密码登录。详情请查看 第 5.4.2.4 节 “以 AD 用户身份向 IdM Web UI 进行身份验证”。

有关配置环境以启用身份验证的详情，请参考：

- 第 23.6.1 节 “在 Web UI 中为智能卡身份验证准备身份管理服务器”
- 第 23.6.2 节 “为智能卡身份验证准备浏览器”

有关如何验证的详情请参考：

- 第 23.6.3 节 “以身份管理用户身份使用智能卡向身份管理 Web UI 进行身份验证”

23.6.1. 在 Web UI 中为智能卡身份验证准备身份管理服务器

作为身份管理管理员：

1. 在身份管理服务器上，创建 shell 脚本来配置服务器。

   a. 使用 ipa-advise config-server-for-smart-card-auth 命令，并将其输出保存到文件中：

   ```
   # ipa-advise config-server-for-smart-card-auth > server_smart_card_script.sh
   ```

   b. 打开 脚本文件，并检查其内容。

   c. 使用 chmod 实用程序为文件添加执行权限：

   ```
   # chmod +x server_smart_card_script.sh
   ```

2. 在 Identity Management 域中的所有服务器上运行 脚本。

3. 确保已安装 sssd-dbus 软件包。

另外，如果外部证书颁发机构(CA)签署了智能卡中的证书：

1. 在身份管理服务器中，将 CA 证书添加到 HTTP 服务器使用的 NSS 数据库中：

   ```
   # ipa-cacert-manage -n "SmartCard CA" -t CT,C,C install ca.pem
   # ipa-certupdate
   ```

   在所有副本和客户端上重复 ipa-certupdate。

2. 重启 HTTP 服务器和 Kerberos 服务器：

```
# systemctl restart httpd
# systemctl restart krb5kdc
```

对所有副本重复这些命令。

23.6.2. 为智能卡身份验证准备浏览器

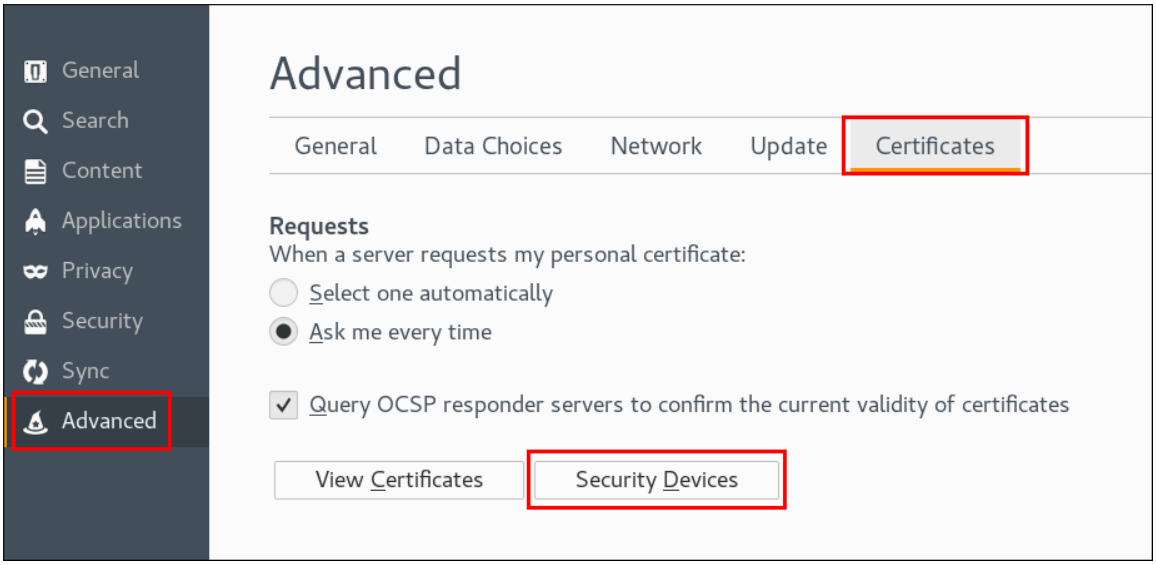
要配置浏览器以进行智能卡身份验证，请在用户启动 Web 浏览器以访问 Web UI 的客户端上执行这些步骤。浏览器所在的系统不需要是身份管理域的一部分。在此流程中，我们使用 Firefox 浏览器。

1. 启动 Firefox。

2. 将 Firefox 配置为从智能卡读取证书。

   a. 选择 Edit → Preferences → Advanced → Certificates → Security Devices

   图 23.16. 在 Firefox 中配置安全设备

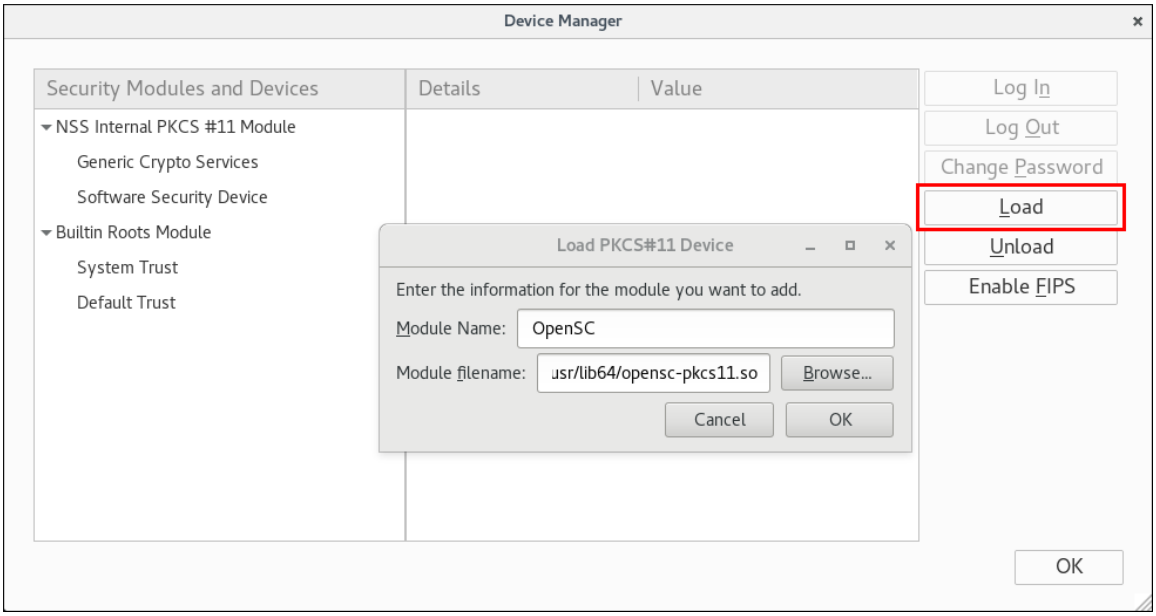


   b. 单击 Load。在 Load PKCS the Device 窗口中填写以下信息：

   - 模块名称 ：OpenSC

- 模块文件名 ：/usr/lib64/opensc-pkcs11.so

图 23.17. Firefox 中的设备管理器



c. 点 OK 确认。然后单击 OK 以关闭设备管理器。

Firefox 现在可以使用智能卡证书进行验证。

23.6.3. 以身份管理用户身份使用智能卡向身份管理 Web UI 进行身份验证

验证：

1. 将智能卡插入到智能卡读取器中。

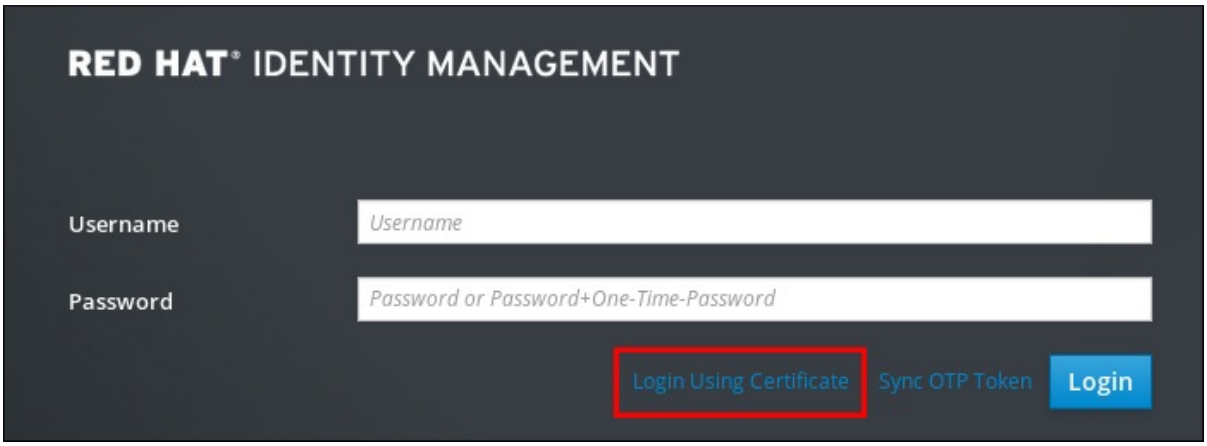
2. 在浏览器中，导航到位于 https://ipaserver.example.com/ipa/ui 的 Identity Management Web UI。

3. 如果智能卡证书链接到一个用户帐户，请不要填写 Username 字段。

   如果智能卡证书链接到多个用户帐户，请填写 Username 字段来指定所需的帐户。

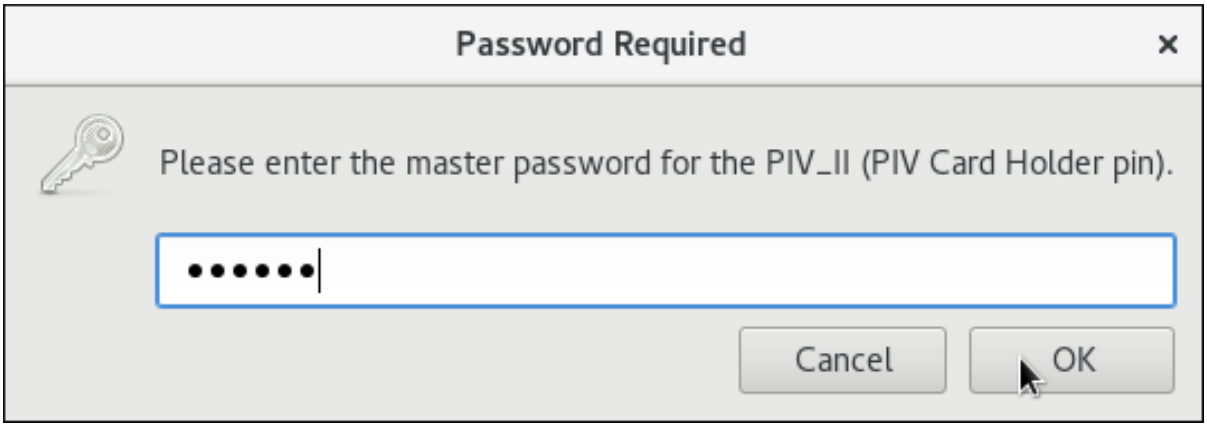
4. *单击 "登录使用证书 "*

图 23.18. 在身份管理 Web UI 中使用证书登录



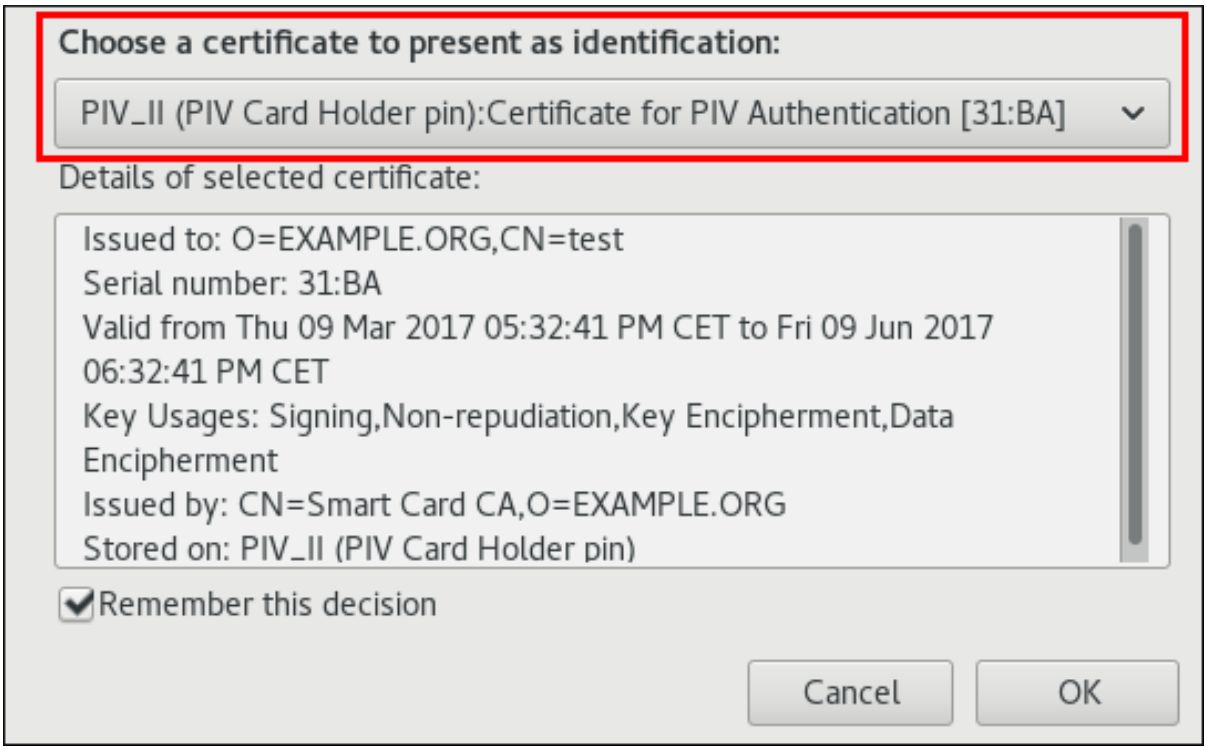
5. *提示时输入智能卡 PIN。*

图 23.19. 输入智能卡 PIN



6. *此时会打开一个新窗口，建议使用证书。选择智能卡证书。*

图 23.20. 选择智能卡证书



您现在作为与智能卡证书对应的用户进行身份验证。

注意

如果管理员重置用户的密码，IdM Web UI 会拒绝访问，直到用户设置新密码，例如，使用 kinit 实用程序。

其它资源

- 如果身份验证失败，请参阅 第 A.4 节 “调查智能卡身份验证失败”。

23.6.4. 其它资源

- 有关身份管理 Web UI 的详情，请参考 第 5.4 节 “The IdM Web UI”。

23.7. 将身份管理智能卡身份验证与 WEB 应用程序集成

作为一名开发人员，其应用通过身份管理 Web 基础架构 Apache 模块将身份管理服务器用作身份验证后端，您可以配置应用程序，使用户的身份验证具有多个角色帐户与其智能卡相关联。这使得这些用户能够在允许的角色帐户下使用应用。

23.7.1. 使用智能卡 Web 应用程序身份验证的先决条件

在运行 Apache Web 应用程序的服务器中：

- 将服务器注册为身份管理域中的客户端。
- 安装 sssd-dbus 和 mod_lookup_identity 软件包。
- 确保 Apache 具有使用 mod_nss 模块配置的可正常工作的 HTTPS 连接。

23.7.2. 为 Web 应用程序配置身份管理智能卡身份验证

1. 在 /etc/httpd/conf.d/nss.conf 文件中的 mod_nss 配置中启用 TLS 重新协商：

```
NSSRenegotiation
NSSRequireSafeNegotiation on
```

2. 确保为 mod_nss 证书数据库中的客户端证书信任签发用户的 CA。数据库的默认位置为 /etc/httpd/alias。

3. 添加 Web 应用。在此过程中，我们使用包含登录页面和受保护区域几乎最小的示例。

   - /login 端点仅允许用户提供用户名，并将用户发送到应用的受保护的部分。
   - /app 端点检查 REMOTE_USER 环境变量。如果登录成功，变量包含已登录用户的 ID。否则，将取消设置 变量。

4. 创建一个目录，并将其组设置为 apache，并将模式设置为至少 750。在此过程中，我们使用名为 /var/www/app/ 的目录。

5. 创建一个文件，并将其组设置为 apache，并将模式设置为至少 750。在此过程中，我们使用名为 /var/www/app/login.py 的文件。

将以下内容保存到文件中：

```
#! /usr/bin/env python

def application(environ, start_response):
    status = '200 OK'
    response_body = """
<!DOCTYPE html>
<html>
    <head>
        <title>Login</title>
    </head>
    <body>
        <form action='/app' method='get'>
            Username: <input type='text' name='username'>
            <input type='submit' value='Login with certificate'>
        </form>
    </body>
</html>
"""
    response_headers = [
        ('Content-Type', 'text/html'),
        ('Content-Length', str(len(response_body)))
    ]
    start_response(status, response_headers)
    return [response_body]
```

6.

*创建一个文件，并将其组设置为 **apache**，并将模式设置为至少 **750**。在此过程中，我们使用名为 **/var/www/app/protected.py** 的文件。*

将以下内容保存到文件中：

```
#! /usr/bin/env python

def application(environ, start_response):
    try:
        user = environ['REMOTE_USER']
    except KeyError:
        status = '400 Bad Request'
        response_body = 'Login failed.\n'
    else:
        status = '200 OK'
        response_body = 'Login succeeded. Username: {}\n'.format(user)

    response_headers = [
        ('Content-Type', 'text/plain'),
        ('Content-Length', str(len(response_body)))
```

```
    ]
    start_response(status, response_headers)
    return [response_body]
```

7. 为您的应用创建配置文件。在此过程中，我们使用包含以下内容的名为 **/etc/httpd/conf.d/app.conf** 的文件：

```
<IfModule !lookup_identity_module>
   LoadModule lookup_identity_module modules/mod_lookup_identity.so
</IfModule>

WSGIScriptAlias /login /var/www/app/login.py
WSGIScriptAlias /app /var/www/app/protected.py

<Location "/app">
   NSSVerifyClient require
   NSSUserName SSL_CLIENT_CERT
   LookupUserByCertificate On
   LookupUserByCertificateParamName "username"
</Location>
```

在这个文件中：

- 第一个部分加载 **mod_lookup_identity** （如果尚未加载）。
- 下一部分将 **/login** 和 **/app** 端点映射到相应的 **Web** 服务器网关接口**(WSGI)**脚本。
- 最后部分为 **/app** 端点配置 **mod_nss**，以便在 **TLS** 握手期间需要客户端证书并使用它。另外，它还配置一个可选的请求参数 用户名 来查找用户身份。

23.8. 在从 KDC 获取请求时强制执行特定身份验证指示器

要强制使用特定的验证指示符，请执行以下操作：

- 主机对象，执行：

```
# ipa host-mod host_name --auth-ind=indicator
```

- 一个 Kerberos 服务，执行：

```
# ipa service-mod service/host_name --auth-ind=indicator
```

要设置多个身份验证指标，请多次指定 **--auth-ind** 参数。

警告

将身份验证指标设置为 HTTP/IdM_master 服务会导致 IdM master 失败。另外，IdM 提供的工具不会允许您恢复 master。

例 23.2. 在特定主机上强制 pkinit 标识符

以下命令配置只有通过智能卡验证的用户才能获取 host.idm.example.com 主机的服务票据：

```
# ipa host-mod host.idm.example.com --auth-ind=pkinit
```

以上设置可确保请求 Kerberos 票据的用户的票据授予票据(TGT)包含 pkinit 身份验证指标。

第 24 章 管理用户、主机和服务的证书

身份管理(IdM)支持两种类型的证书颁发机构(CA)：

集成的 IdM CA

集成的 CA 可以为用户、主机和服务创建、吊销和发布证书。如需了解更多详细信息，请参阅 第 24.1 节 “使用集成的 IdM CA 管理证书”。

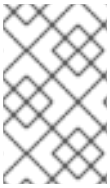
IdM 支持创建轻量级子 CA。如需了解更多详细信息，请参阅 第 26.1 节 “轻量级子 CA”

外部 CA

外部 CA 是集成的 IdM CA 以外的 CA。

使用 IdM 工具，您可以将这些 CA 发布的证书添加到用户、服务或主机，以及删除它们。如需了解更多详细信息，请参阅 第 24.2 节 “管理由外部 CA 发布的证书”。

每个用户、主机或服务都可以分配多个证书。



注意

有关 IdM 服务器的 CA 配置的详情，请参考 第 2.3.2 节 “确定要使用的 CA 配置”。

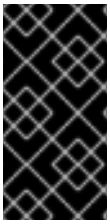
24.1. 使用集成的 IDM CA 管理证书

24.1.1. 为用户、主机或服务请求新证书

使用以下命令请求证书：

- IdM Web UI，请查看 “Web UI：请求新证书”一节。
- 命令行请查看 “命令行：请求新证书”一节。

请注意，您必须使用第三方工具自行生成证书请求。以下流程使用 certutil 和 openSSL 工具。



重要

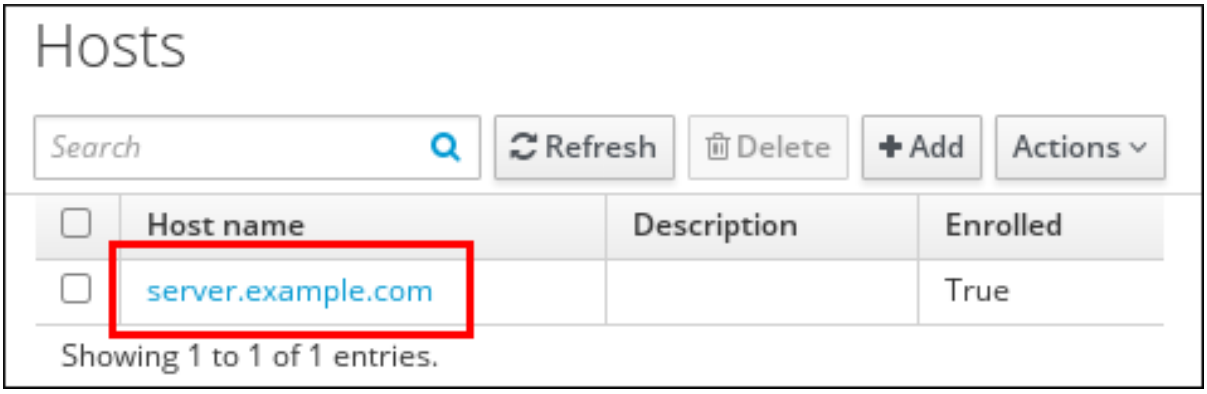
通常运行在存储私钥的专用服务节点上的服务。将服务的私钥复制到 IdM 服务器被视为不安全。因此，在为服务请求证书时，在服务节点上创建 CSR。

Web UI：请求新证书

1. 在 Identity 选项卡下，选择 Users、Hosts 或 Services 子选项卡。

2. 单击用户、主机或服务的名称，来打开其配置页面。

图 24.1. 主机列表



3. 单击 Actions → New Certificate。

4. 可选：选择发布 CA 和配置文件 ID。

5. 按照屏幕上的说明操作，以使用 certutil。

6. 单击 Issue。

命令行：请求新证书

在标准情况下使用 certutil 请求新证书 - 请参阅 第 24.1.1.1 节 “使用 certutil 请求新证书”。使用 openSSL 请求新证书，以启用 Kerberos 别名来使用主机或服务证书 - 请参阅 第 24.1.1.2 节 “使用

OpenSSL 准备多个 SAN 字段的证书请求”。

24.1.1.1. 使用 certutil 请求新证书

1. 为证书数据库创建一个临时目录：

```
# mkdir ~/certdb/
```

2. 创建新的临时证书数据库，例如：

```
# certutil -N -d ~/certdb/
```

3. 创建证书签名请求(CSR)，并将输出重定向到文件。例如，要为 4096 位证书创建 CSR，并将主题设为 CN=server.example.com,O=EXAMPLE.COM ：

```
# certutil -R -d ~/certdb/ -a -g 4096 -s "CN=server.example.com,O=EXAMPLE.COM" -8
server.example.com > certificate_request.csr
```

4. 将证书请求提交到 CA。详情请查看 第 24.1.1.4 节 “将证书请求提交到 IdM CA”。

24.1.1.2. 使用 OpenSSL 准备多个 SAN 字段的证书请求

1. 为 Kerberos 主体 test /server.example.com 创建一个或多个别名，如test1/ server.example.com、test2/ server.example.com。详情请查看 第 20.2.1 节 “Kerberos 主要别名”。

2. 在 CSR 中，为 dnsName(server.example.com)和 otherName(test2/server.example.com)添加 subjectAltName。要做到这一点，请配置 openssl.conf 文件，使其包含以下指定 UPN otherName 和 subjectAltName 的行：

```
otherName=1.3.6.1.4.1.311.20.2.3;UTF8:test2/server.example.com@EXAMPLE.COM
DNS.1 = server.example.com
```

3. 使用 openssl 创建证书请求：

```
openssl req -new -newkey rsa:2048 -keyout test2service.key -sha256 -nodes -out
certificate_request.csr -config openssl.conf
```

4. 将证书请求提交到 CA。详情请查看 第 24.1.1.4 节 “将证书请求提交到 IdM CA”。

24.1.1.3. 使用 Certmonger 请求新证书

您可以使用 certmonger 服务从 IdM CA 请求证书。详情请查看《系统级身份验证指南》中的请求 CA 签名证书 Through SCEP 部分。

24.1.1.4. 将证书请求提交到 IdM CA

将证书请求文件提交到在 IdM 服务器上运行的 CA。务必指定与新签发的证书关联的 Kerberos 主体：

```
# ipa cert-request certificate_request.csr --principal=host/server.example.com
```

IdM 中的 ipa cert-request 命令使用以下默认值：

- 证书配置文件： caIPAserviceCert

  要选择自定义配置集，请在 ipa cert-request 命令中使用 --profile-id 选项。

  有关创建自定义证书配置集的详情，请参考 第 24.4.1 节 “创建证书配置文件”。

- 集成的 CA： ipa (IdM root CA)

  要选择一个子 CA，请在 ipa cert-request 命令中使用 --ca 选项。

详情请查看 ipa cert-request --help 命令的输出。

24.1.2. 使用集成的 IdM CA 撤销证书

如果您需要在证书过期前无效，您可以撤销证书。使用以下方法撤销证书：

- **IdM Web UI，请查看 “Web UI：撤销证书”一节**
- **命令行，请查看 “命令行：撤销证书”一节**

已吊销的证书是无效的，不能用于身份验证。所有撤销都是永久性的，但原因 6：证书已保留.

表 24.1. 吊销原因

ID	原因	解释
0	未指定	
1	密钥泄露	签发证书的密钥不再被信任。 可能的原因是：丢失令牌，非正常访问文件。
2	CA 泄露	签发证书的 CA 不再被信任。
3	隶属关系更改了	可能的原因： ● 某人已离开公司或迁移到另一个部门。 ● 主机或服务正在被停用。
4	被取代	较新的证书替换了当前的证书。
5	停止操作	主机或服务将被停用。
6	证书冻结	证书被临时吊销。您可稍后恢复证书。
8	从 CRL 中删除	证书不再包含在证书吊销列表(CRL)中。
9	特权收回	用户、主机或服务不再被允许使用证书。
10	属性授权(AA)泄露	AA 证书不再被信任。

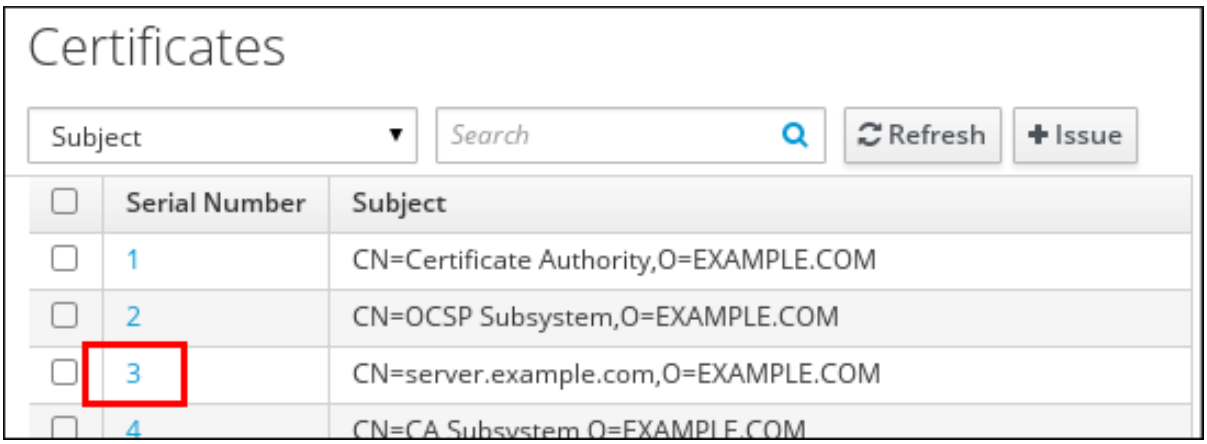
Web UI：撤销证书

撤销证书：

1. 打开 Authentication 选项卡，然后选择 证书 子选项卡。

2. 单击证书的序列号，来打开证书信息页面。

图 24.2. 证书列表



3. 单击 Actions → Revoke Certificate。

4. 选择吊销的原因，然后单击 Revoke。详情请查看 表 24.1 “吊销原因”。

命令行：撤销证书

使用 ipa cert-revoke 命令，并指定：

- 证书序列号
- 标识撤销原因的数字；详情请查看 表 24.1 “吊销原因”

例如，由于原因 1，要吊销序列号为 1032 的证书：主要总结：

```
$ ipa cert-revoke 1032 --revocation-reason=1
```

24.1.3. 使用集成的 IdM CA 恢复证书

如果您因为原因 6 撤销了证书：证书已保留，您可以重新恢复。使用以下方法恢复证书：

- IdM Web UI，请查看 “Web UI：恢复证书”一节

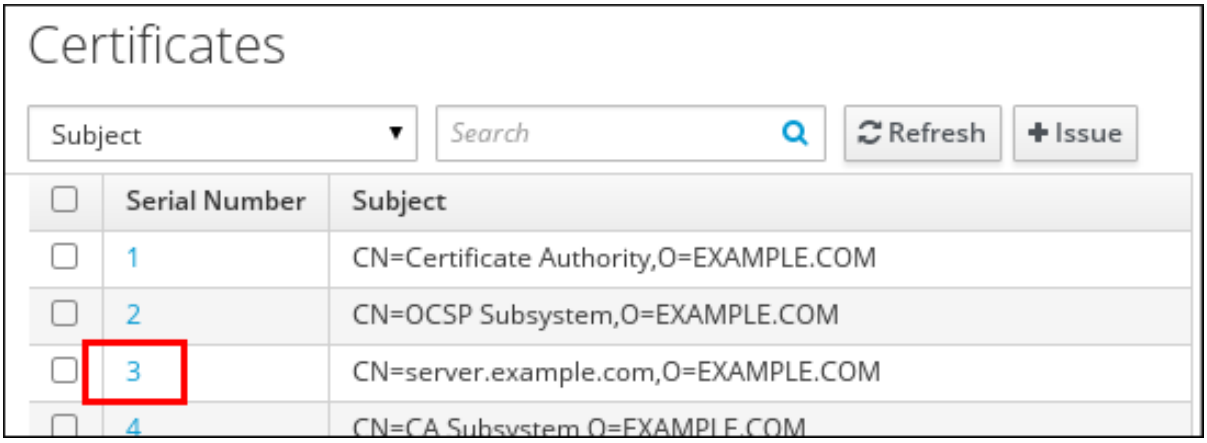
- 命令行，请查看 “命令行：恢复证书”一节

Web UI：恢复证书

1. 打开 Authentication 选项卡，然后选择 证书 子选项卡。

2. 单击证书的序列号，来打开证书信息页面。

图 24.3. 证书列表



3. 单击 Actions → Restore Certificate。

命令行：恢复证书

使用 ipa cert-remove-hold 命令并指定证书序列号。例如：

```
$ ipa cert-remove-hold 1032
```

24.2. 管理由外部 CA 发布的证书

24.2.1. 命令行：添加和删除由外部 CA 发布的证书

为用户、主机或服务添加证书：

- ipa user-add-cert

- ipa host-add-cert
- ipa service-add-cert

从用户、主机或服务中删除证书：

- ipa user-remove-cert
- ipa host-remove-cert
- ipa service-remove-cert

从 IdM 中删除外部 CA 发布的证书后，不会撤销它。这是因为证书没有存在于 IdM CA 数据库中。您只能从外部 CA 端手动撤销这些证书。

这些命令需要您指定以下信息：

- 用户、主机或服务的名称
- Base64 编码的 DER 证书

要以交互方式运行命令，请在不添加任何选项的情况下执行这些命令。

要直接通过 命令提供所需的信息，请使用命令行参数和选项：

```
$ ipa user-add-cert user --certificate=MIQTPrajQAwg...
```

注意

您可以将证书转换为 **DER** 格式，然后重新编码为 **base64**，而不是将证书复制并粘贴到命令行中。例如，要将 **user_cert.pem** 证书添加到 **user** ：

```
$ ipa user-add-cert user --certificate="$(openssl x509 -outform der -in user_cert.pem | base64 -w 0)"
```

24.2.2. Web UI：添加和删除由外部 CA 发布的证书

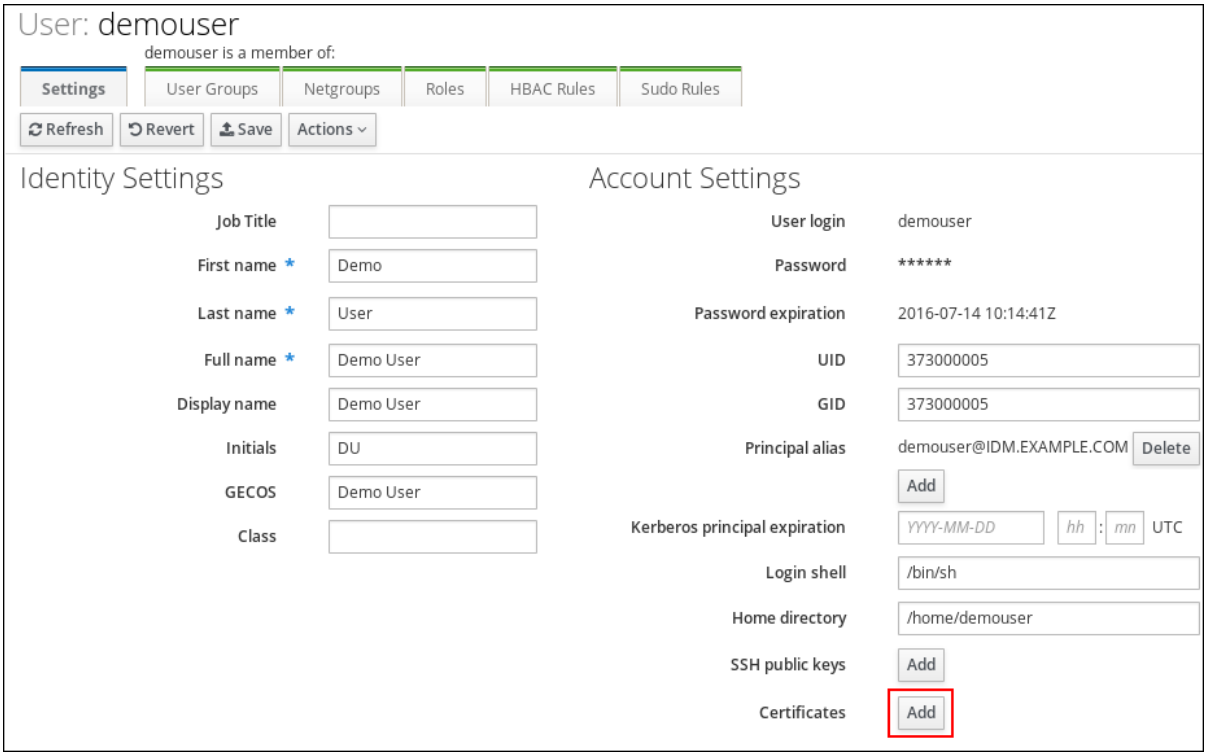
为用户、主机或服务添加证书：

1. 打开 **Identity** 选项卡，然后选择 **Users**、**Hosts** 或 **Services** 子选项卡。

2. 单击用户、主机或服务的名称以打开其配置页面。

3. 单击 **Certificates** 条目旁边的 **Add**。

图 **24.4.** 添加证书到用户帐户



4. 将 **Base64** 或 **PEM** 编码格式的证书粘贴到文本字段中，然后单击 **Add**。

5. 单击 **Save** 以保存更改。

从用户、主机或服务中删除证书：

1. 打开 **Identity** 选项卡，然后选择 **Users**、**Hosts** 或 **Services** 子选项卡。

2. 单击用户、主机或服务的名称以打开其配置页面。

3. 单击要删除的证书旁边的 **Actions**，然后选择 **Delete**。

4. 单击 **Save** 以保存更改。

24.3. 列出和显示证书

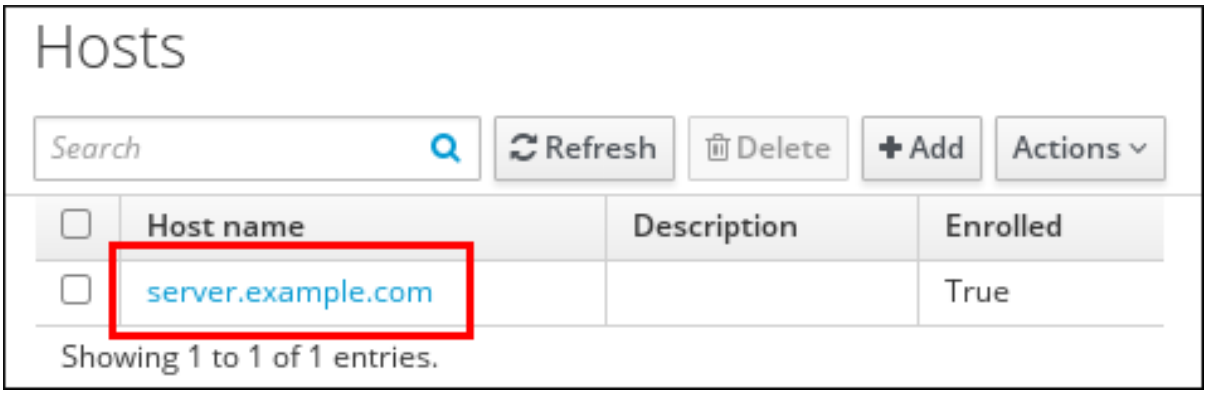
在 Web UI 中列出和显示证书

列出分配给用户、主机或服务条目的证书：

1. 打开 **Identity** 选项卡，然后选择 **Users**、**Hosts** 或 **Services** 子选项卡。

2. 单击用户、主机或服务的名称以打开其配置页面。

图 24.5. 主机列表



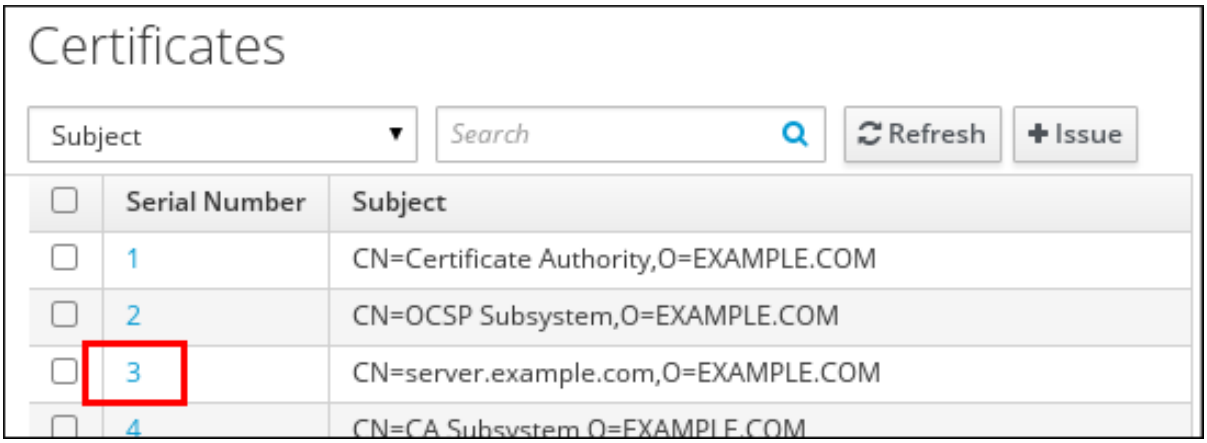
3. 配置页面中列出了分配给该条目的所有证书。此外，单击 **Show** 显示特定证书。

列出在 IdM 服务器中注册的所有证书：

1. **打开 Authentication 选项卡，然后选择 证书 子选项卡。**

2. **证书部分会显示所有 证书 的列表。要显示特定证书，请单击其序列号。**

图 24.6. 证书列表



从命令行列出证书

要列出 IdM 数据库中的所有证书，请运行 ipa cert-find 命令。

```
$ ipa cert-find
-----------------------
10 certificates matched
-----------------------
  Serial number (hex): 0x1
  Serial number: 1
  Status: VALID
  Subject: CN=Certificate Authority,O=EXAMPLE.COM
...
----------------------------
Number of entries returned 10
----------------------------
```

您可以通过指定某些证书属性来过滤搜索结果，如签发日期或有效期日期。例如，要按问题日期间隔搜索，请使用 --issuedon-from 或 --issuedon- 选项来指定起始点和端点或一段时间。

```
ipa cert-find --issuedon-from=2020-01-07 --issuedon-to=2020-02-07
```

如需用于过滤搜索证书的选项列表，请使用 --help 选项运行 ipa cert-find。

从命令行显示证书

要显示证书，请使用 ipa cert-show 命令并指定序列号。

```
$ ipa cert-show 132
Serial number: 132
  Certificate:
MIIDtzCCAp+gAwIBAgIBATANBgkqhkiG9w0BAQsFADBBMR8wHQYDVQQKExZMQUIu
...
LxIQjrEFtJmoBGb/TWRlwGEWy1ayr4iTEf1ayZ+RGNylLaIEAtk9RLjEjg==
  Subject: CN=Certificate Authority,O=EXAMPLE.COM
  Issuer: CN=Certificate Authority,O=EXAMPLE.COM
  Not Before: Sun Jun 08 05:51:11 2014 UTC
  Not After: Thu Jun 08 05:51:11 2034 UTC
  Serial number (hex): 0x132
  Serial number: 132
```

要显示分配给用户、主机或服务条目的证书，请使用 ipa cert-show 并指定该条目。例如，显示分配给用户的证书：

```
$ ipa user-show user
  User login: user
  ...
  Certificate: MIICfzCCAWcCAQA...
  ...
```

您还可以通过将 --out 选项添加到 ipa cert-show 来将证书保存到文件中。

```
$ ipa cert-show certificate_serial_number --out=path_to_file
```

请注意，如果 user、host 或 服务具有多个证书，则 --out 选项将导出全部证书。证书或证书作为 PEM 对象导出。

24.4. 证书配置文件

证书配置文件定义了属于特定配置文件的证书内容，以及签发证书、注册方式以及用于注册的输入和输出表单的限制。单个证书配置文件与签发特定类型的证书相关联。可以为 IdM 中的用户、服务和主机定义不同的证书配置文件。

CA 使用证书签名中的证书配置集来决定：

- CA 是否可以接受证书签名请求(CSR)
- 证书中应存在哪些功能和扩展

IdM 默认包括以下两个证书配置文件： caIPAserviceCert 和 IECUserRoles。另外，也可以导入自定义配置集。

自定义证书配置文件允许针对特定不相关的用途签发证书。例如，可以将特定配置集的使用限制为仅一个用户或一个组，从而防止其他用户和组使用该配置文件发布证书以进行身份验证。

有关支持的证书配置文件配置的详情，请参阅 Red Hat Certificate System Administration Guide 中的默认 参考 和约束 参考。

注意

通过组合证书配置文件和 CA ACL 第 24.5 节 “证书颁发机构 ACL 规则”，管理员可以定义和控制自定义证书配置集的访问。有关使用配置集和 CA ACL 发布用户证书的描述，请参考 第 24.6 节 “使用证书配置文件和 ACL 来向 IdM CA 签发用户证书”。

24.4.1. 创建证书配置文件

有关创建证书配置文件的详情，请查看 Red Hat Certificate System 9 管理指南中的 以下文档：

- 设置证书配置文件 部分说明了如何创建新证书配置文件以及如何构建它们。
- 证书和 CRL 附录的默认、约束和扩展列出了 您可以在证书配置集中使用的对象标识符(OID)其他字段。

24.4.2. 从命令行管理证书配置文件

用于管理 IdM 配置集的 certprofile 插件允许特权用户导入、修改或删除 IdM 证书配置文件。要显示插件支持的所有命令，请运行 ipa certprofile 命令：

```
$ ipa certprofile
Manage Certificate Profiles
```

```
...

EXAMPLES:

  Import a profile that will not store issued certificates:
    ipa certprofile-import ShortLivedUserCert \
      --file UserCert.profile --desc "User Certificates" \
      --store=false

  Delete a certificate profile:
    ipa certprofile-del ShortLivedUserCert
...
```

请注意，要执行 certprofile 操作，您必须以具有所需权限的用户进行操作。IdM 默认包括以下证书配置集相关的权限：

系统：读取证书配置文件

允许用户读取所有配置集属性。

系统：导入证书配置集

允许用户将证书配置集导入到 IdM 中。

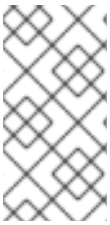
系统：删除证书配置集

允许用户删除现有证书配置文件。

系统：修改证书配置集

允许用户修改配置集属性并禁用或启用配置集。

所有这些权限都包含在默认的 CA Administrator 特权中。有关基于 IdM 角色的访问控制和管理权限的更多信息，请参阅 第 10.4 节 “定义基于角色的访问控制”。



注意

在请求证书时，可将 --profile-id 选项添加到 ipa cert-request 命令中，以指定要使用的配置集。如果没有指定配置文件 ID，则默认 caIPAserviceCert 配置集用于证书。

本节只描述了使用 ipa certprofile 命令进行配置集管理的重要方面。有关命令的完整信息，请使用添加 --help 选项来运行，例如：

```
$ ipa certprofile-mod --help
Usage: ipa [global-options] certprofile-mod ID [options]

Modify Certificate Profile configuration.
Options:
  -h, --help     show this help message and exit
  --desc=STR     Brief description of this profile
  --store=BOOL   Whether to store certs issued using this profile
...
```

导入证书配置集

要将新证书配置文件导入到 IdM，请使用 ipa certprofile-import 命令。运行不带任何选项的命令将启动一个交互式会话，其中 certprofile-import 脚本会提示您输入导入证书所需的信息。

```
$ ipa certprofile-import

Profile ID: smime
Profile description: S/MIME certificates
Store issued certificates [True]: TRUE
Filename of a raw profile. The XML format is not supported.: smime.cfg
-----------------------
Imported profile "smime"
-----------------------
  Profile ID: smime
  Profile description: S/MIME certificates
  Store issued certificates: TRUE
```

ipa certprofile-import 命令接受几个命令行选项。最值得注意的是：

--file

此选项将包含配置文件配置的文件直接传递给 ipa certprofile-import。例如：

```
$ ipa certprofile-import --file=smime.cfg
```

--store

此选项设置 Store 颁发的证书 属性。它接受两个值：

- true，它为客户端提供发布的证书，并将其存储在目标 IdM 主体的 userCertificate 属性中。

- false，它为客户端提供发布的证书，但不将其存储在 IdM 中。在发出多个短期证书时，最常使用这个选项。

如果已使用 ipa certprofile-import 指定的配置集 ID 已在使用中，或者配置集内容不正确，则导入会失败。例如，如果缺少所需的属性，或者提供的文件中定义的配置集 ID 与 ipa certprofile-import 指定的配置集 ID 不匹配，则导入会失败。

要获取新配置集的模板，您可以使用 --out 选项运行 ipa certprofile-show 命令，该命令将指定的现有配置集导出到文件中。例如：

```
$ ipa certprofile-show caIPAserviceCert --out=file_name
```

然后，您可以根据需要编辑导出的文件并将其导入为新配置集。

显示证书配置集

要显示当前存储在 IdM 中的所有证书配置文件，请使用 ipa certprofile-find 命令：

```
$ ipa certprofile-find
------------------
3 profiles matched
------------------
  Profile ID: caIPAserviceCert
  Profile description: Standard profile for network services
  Store issued certificates: TRUE

  Profile ID: IECUserRoles
...
```

要显示特定配置集的信息，请使用 ipa certprofile-show 命令：

```
$ ipa certprofile-show profile_ID
  Profile ID: profile_ID
  Profile description: S/MIME certificates
  Store issued certificates: TRUE
```

修改证书配置集

要修改现有证书配置文件，请使用 ipa certprofile-mod 命令。使用 ipa certprofile-mod 接受的命令行选项，通过命令传递所需的修改。例如，要修改配置集的描述并更改 IdM 是否存储发布的证书：

```
$ ipa certprofile-mod profile_ID --desc="New description" --store=False
```

```
-----------------------------------
Modified Certificate Profile "profile_ID"
-----------------------------------
  Profile ID: profile_ID
  Profile description: New description
  Store issued certificates: FALSE
```

要更新证书配置集配置，请使用 --file 选项导入包含更新配置的文件。例如：

```
$ ipa certprofile-mod profile_ID --file=new_configuration.cfg
```

删除证书配置集

要从 IdM 中删除现有证书配置文件，请使用 ipa certprofile-del 命令：

```
$ ipa certprofile-del profile_ID
-----------------------
Deleted profile "profile_ID"
-----------------------
```

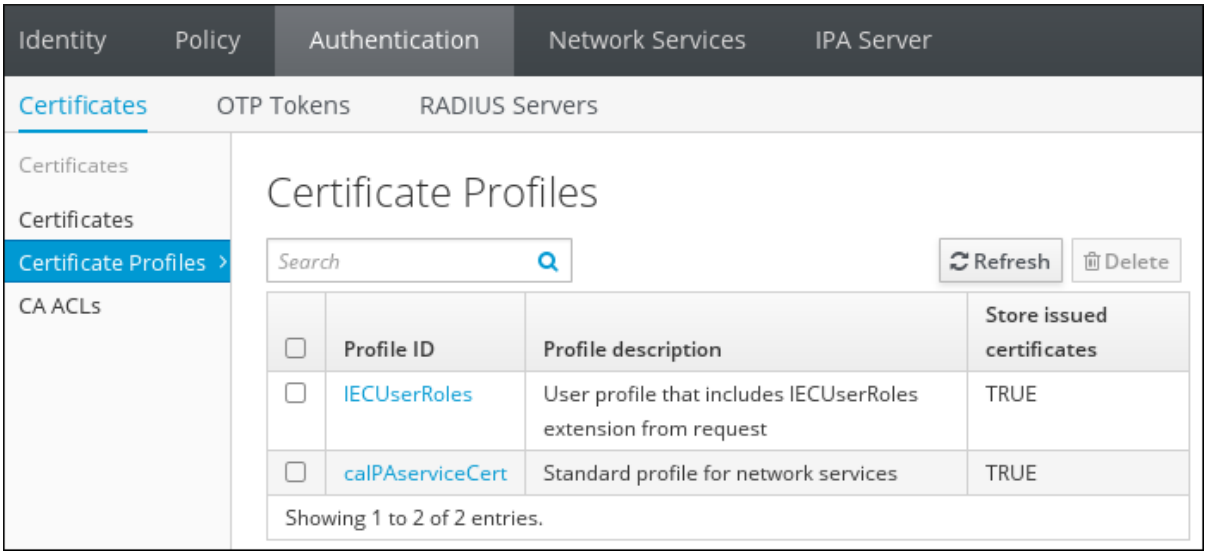
24.4.3. 从 Web UI 中的证书配置文件管理

从 IdM Web UI 管理证书配置集：

1. **打开 Authentication 选项卡和 Certificates 子选项卡。**

2. **打开 Certificate Profiles 部分。**

图 24.7. Web UI 中的证书配置文件管理

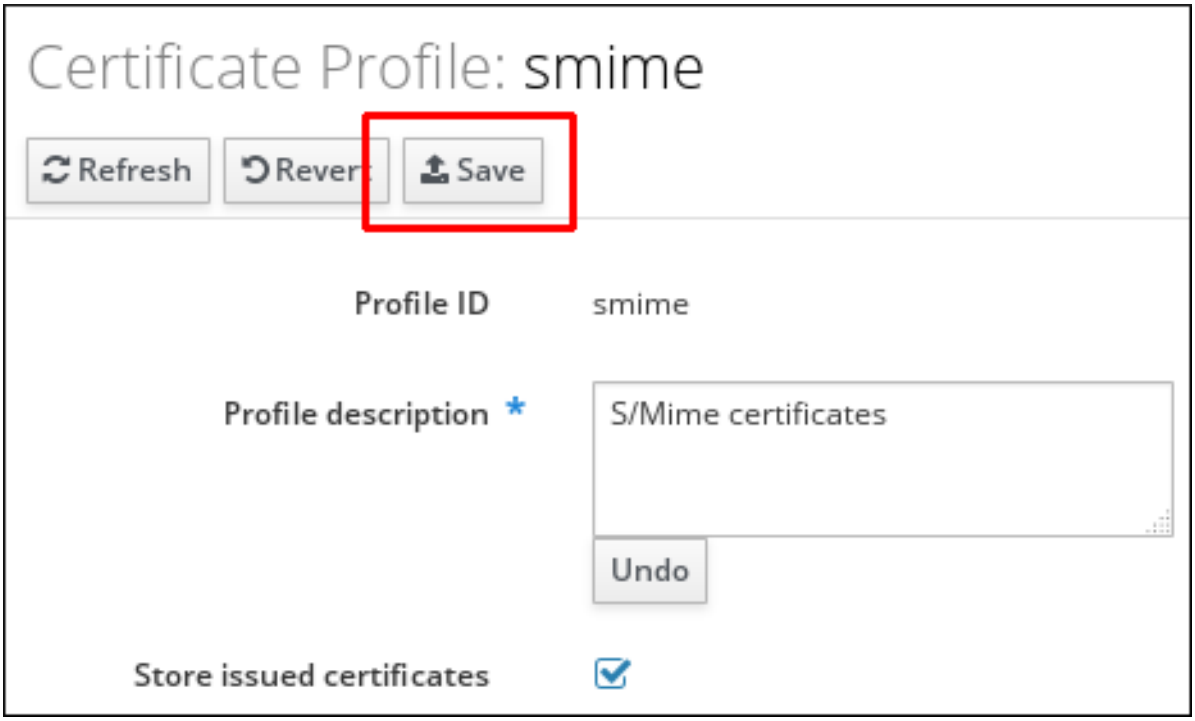


在 Certificate Profiles 部分中，您可以显示有关现有配置集的信息，修改其属性或删除所选配置集。

例如，要修改现有证书配置集：

1. 单击配置文件的名称，以打开配置文件配置页面。
2. 在配置集配置页面中，填写所需信息。
3. 单击 Save 以确认新配置。

图 24.8. 在 Web UI 中修改证书配置集



如果您启用 Store 发布的证书 选项，发布的证书将传送到客户端，并存储在目标 IdM 主体的 userCertificate 属性中。如果禁用该选项，发布的证书将传送到客户端，但不存储在 IdM 中。在发出多个较短的证书时，存储证书通常会被禁用。

请注意，一些证书配置集管理操作目前在 web UI 中不可用：

- 无法在 web UI 中导入证书配置文件。要导入证书，请使用 ipa certprofile-import 命令。

- 无法设置、添加或删除属性和值对。要修改属性和值对，请使用 ipa certprofile-mod 命令。

- 无法导入更新的证书配置集配置。要导入包含更新的配置集配置的文件，请使用 ipa certprofile-mod --file=file_name 命令。

有关用于管理证书配置集的命令的详情请参考 第 24.4.2 节 “从命令行管理证书配置文件”。

24.4.4. 使用证书配置集升级 IdM 服务器

升级 IdM 服务器时，服务器中包含的配置集都会被导入并启用。

如果升级多个服务器副本，则第一个升级副本的配置文件会被导入。在其他副本中，IdM 会检测到是否存在其他配置集，且不会导入它们，或解决两组配置集之间的任何冲突。如果您在副本上定义了自定义配置集，请确保在所有副本上的配置集在升级前保持一致。

24.5. 证书颁发机构 ACL 规则

证书颁发机构访问控制列表(CA ACL)规则定义哪些配置文件可用于向哪些用户、服务或主机发布证书。通过关联配置集、主体和组，CA ACL 允许主体或组使用特定配置集请求证书：

- ACL 允许访问多个配置集

- ACL 可以关联有多个用户、服务、主机、用户组和主机组

例如，利用 CA ACL，管理员可以将配置文件的使用限制为仅属于伦敦办事处相关组成员的员工。

注意

通过将 第 24.4 节 “证书配置文件” 和 CA ACL 中描述的证书配置集与 CA ACL 相结合，管理员可以定义和控制自定义证书配置集的访问。有关使用配置集和 CA ACL 发布用户证书的描述，请参考 第 24.6 节 “使用证书配置文件和 ACL 来向 IdM CA 签发用户证书”。

24.5.1. 从命令行进行 CA ACL 管理

用于管理 CA ACL 规则的 caacl 插件允许特权用户添加、显示、修改或删除指定的 CA ACL。要显示插件支持的所有命令，请运行 ipa caacl 命令：

```
$ ipa caacl
Manage CA ACL rules.

...

EXAMPLES:

  Create a CA ACL "test" that grants all users access to the
  "UserCert" profile:
    ipa caacl-add test --usercat=all
    ipa caacl-add-profile test --certprofiles UserCert

  Display the properties of a named CA ACL:
    ipa caacl-show test

  Create a CA ACL to let user "alice" use the "DNP3" profile on "DNP3-CA":
    ipa caacl-add alice_dnp3
    ipa caacl-add-ca alice_dnp3 --cas DNP3-CA
    ipa caacl-add-profile alice_dnp3 --certprofiles DNP3
    ipa caacl-add-user alice_dnp3 --user=alice
...
```

请注意，要执行 caacl 操作，您必须以具有所需权限的用户进行操作。IdM 默认包括以下与 CA ACL 相关的权限：

系统：阅读 CA ACL

允许用户读取 CA ACL 的所有属性。

系统：添加 CA ACL

允许用户添加新的 CA ACL。

系统：删除 CA ACL

允许用户删除现有 CA ACL。

系统：修改 CA ACL

允许用户修改 CA ACL 的属性并禁用或启用 CA ACL。

系统：管理 CA ACL 成员资格

允许用户管理 CA、配置文件、用户、主机和服务成员资格。

所有这些权限都包含在默认的 CA Administrator 特权中。有关基于 IdM 角色的访问控制和管理权限的更多信息，请参阅 第 10.4 节 “定义基于角色的访问控制”。

本节仅描述了将 ipa caacl 命令用于 CA ACL 管理最重要的方面。有关命令的完整信息，请使用添加 --help 选项来运行，例如：

```
$ ipa caacl-mod --help
Usage: ipa [global-options] caacl-mod NAME [options]

Modify a CA ACL.
Options:
  -h, --help            show this help message and exit
  --desc=STR            Description
  --cacat=['all']       CA category the ACL applies to
  --profilecat=['all']  Profile category the ACL applies to
...
```

创建 CA ACL

要创建新 CA ACL，请使用 ipa caacl-add 命令。运行不带任何选项的命令将启动一个交互式会话，其中 ipa caacl-add 脚本会提示您输入有关新 CA ACL 所需的信息。

```
$ ipa caacl-add
ACL name: smime_acl
------------------------
Added CA ACL "smime_acl"
------------------------
  ACL name: smime_acl
  Enabled: TRUE
```

默认启用新的 CA ACL。

ipa caacl-add 接受的最显著选项是将 CA ACL 与 CA、证书配置文件、用户、主机或服务类别关联的选项：

- --cacat

- --profilecat
- --usercat
- --hostcat
- --servicecat

IdM 仅接受带有这些选项的所有值，它们将 CA ACL 与所有 CA、配置集、用户、主机或服务相关联。例如，要将 CA ACL 与所有用户和用户组关联：

```
$ ipa caacl-add ca_acl_name --usercat=all
```

CA、配置集、用户、主机和服务类别是将特定对象或对象组添加到 CA ACL 的替代方法，如 “将条目添加到 CA ACL 中并从 CA ACL 中删除条目”一节 所述。请注意，无法使用类别并添加同一类型的对象或组；例如，您无法使用 --usercat=all 选项，然后使用 ipa caacl-add-user --users=user_name 命令将用户添加到 CA ACL 中。

注意

如果用户或组没有添加到对应的 CA ACL，则使用证书配置集为用户或组请求证书会失败。例如：

```
$ ipa cert-request CSR-FILE --principal user --profile-id profile_id
ipa: ERROR Insufficient access: Principal 'user' is not permitted to use CA '.' with
profile 'profile_id' for certificate issuance.
```

您必须将用户或组添加到 CA ACL 中，如 “将条目添加到 CA ACL 中并从 CA ACL 中删除条目”一节 所述，或者将 CA ACL 与所有用户类别 关联。

显示 CA ACL

要显示所有 CA ACL，请使用 ipa caacl-find 命令：

```
$ ipa caacl-find
----------------
2 CA ACLs matched
```

```
----------------
  ACL name: hosts_services_caIPAserviceCert
  Enabled: TRUE
...
```

请注意，ipa caacl-find 接受 --cacat、--profilecat、--usercat、--hostcat 和 --servicecat 选项，可用于过滤带有对应 CA、证书配置文件、主机或服务类别的搜索 CA ACL 的结果。请注意，IdM 只接受带有这些选项 的所有 类别。有关选项的详情请参考 “创建 CA ACL”一节。

要显示特定 CA ACL 的信息，请使用 ipa caacl-show 命令：

```
$ ipa caacl-show ca_acl_name
  ACL name: ca_acl_name
  Enabled: TRUE
  Host category: all
...
```

修改 CA ACL

要修改现有 CA ACL，请使用 ipa caacl-mod 命令。使用 ipa caacl-mod 接受的命令行选项传递所需的修改。例如，要修改 CA ACL 的描述并将 CA ACL 与所有证书配置集相关联：

```
$ ipa caacl-mod ca_acl_name --desc="New description" --profilecat=all
----------------------------
Modified CA ACL "ca_acl_name"
----------------------------
  ACL name: smime_acl
  Description: New description
  Enabled: TRUE
  Profile category: all
```

ipa caacl-mod 接受的最显著选项是 --cacat、--profilecat、--usercat、--hostcat、和 --servicecat 选项。有关这些选项的描述请查看 “创建 CA ACL”一节。

禁用和启用 CA ACL

要禁用 CA ACL，请使用 ipa caacl-disable 命令：

```
$ ipa caacl-disable ca_acl_name
---------------------------
Disabled CA ACL "ca_acl_name"
---------------------------
```

禁用的 CA ACL 不会被应用，且无法用于请求证书。禁用 CA ACL 不会将其从 IdM 中删除。

要启用禁用的 CA ACL，请使用 ipa caacl-enable 命令：

```
$ ipa caacl-enable ca_acl_name
----------------------------
Enabled CA ACL "ca_acl_name"
----------------------------
```

删除 CA ACL

要删除现有的 CA ACL，请使用 ipa caacl-del 命令：

```
$ ipa caacl-del ca_acl_name
```

将条目添加到 CA ACL 中并从 CA ACL 中删除条目

使用 ipa caacl-add the ipa caacl-remove the 命令，您可以向 CA ACL 添加新条目或删除现有条目。

ipa caacl-add-ca 和 ipa caacl-remove-ca

添加或删除 CA。

ipa caacl-add-host 和 ipa caacl-remove-host

添加或删除主机或主机组.

ipa caacl-add-profile 和 ipa caacl-remove-profile

添加或删除配置文件。

ipa caacl-add-service 和 ipa caacl-remove-service

添加或删除服务。

ipa caacl-add-user 和 ipa caacl-remove-user

添加或删除用户或组。

例如：

```
$ ipa caacl-add-user ca_acl_name --groups=group_name
```

请注意，无法向 CA ACL 添加对象或对象组，也可以根据需要使用同一对象的类别，如 “创建 CA ACL”一节 所述；这些设置是相互排斥的。例如，如果您试图在通过 --usercat=all 选项指定的 CA ACL 上运行 ipa caacl-add-user --users=user_name 命令，命令会失败：

```
$ ipa caacl-add-user ca_acl_name --users=user_name
ipa: ERROR: users cannot be added when user category='all'
```

注意

如果用户或组没有添加到对应的 CA ACL，则使用证书配置集为用户或组请求证书会失败。例如：

```
$ ipa cert-request CSR-FILE --principal user --profile-id profile_id
ipa: ERROR Insufficient access: Principal 'user' is not permitted to use CA '.' with
profile 'profile_id' for certificate issuance.
```

您必须将用户或组添加到 CA ACL 中，或者将 CA ACL 与所有 用户类别关联，如 “创建 CA ACL”一节 所述。

有关这些命令所需语法和可用选项的详细信息，可通过添加 --help 选项来运行命令。例如：

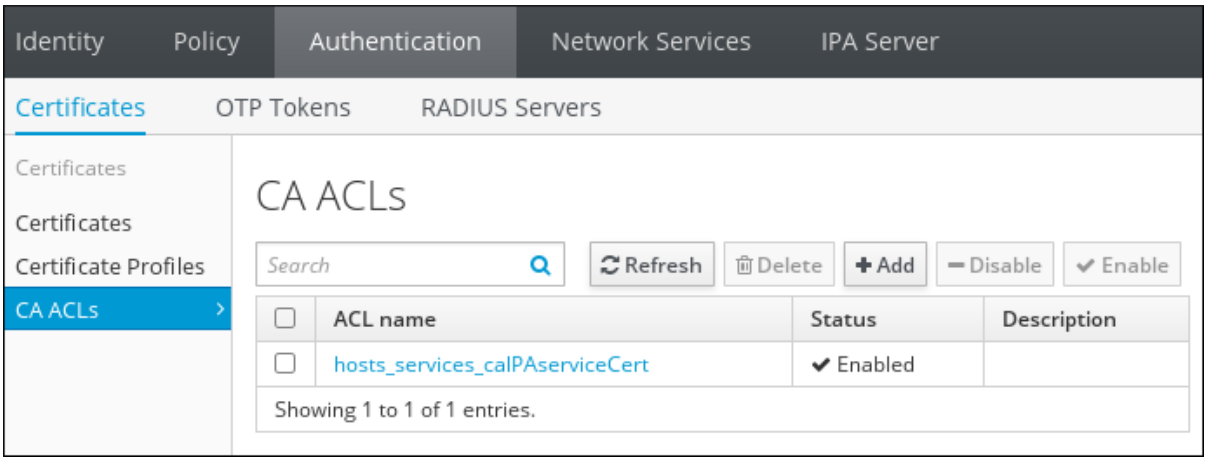
```
$ ipa caacl-add-user --help
```

24.5.2. Web UI 中的 CA ACL 管理

从 IdM Web UI 管理 CA ACL：

1. 打开 Authentication 选项卡和 Certificates 子选项卡。
2. 打开 CA ACL 部分。

图 24.9. Web UI 中的 CA ACL 规则管理



在 CA ACL 部分中，您可以添加新的 CA ACL，显示有关现有 CA ACL 的信息，修改其属性，以及启用、禁用或删除所选 CA ACL。

例如，要修改现有 CA ACL：

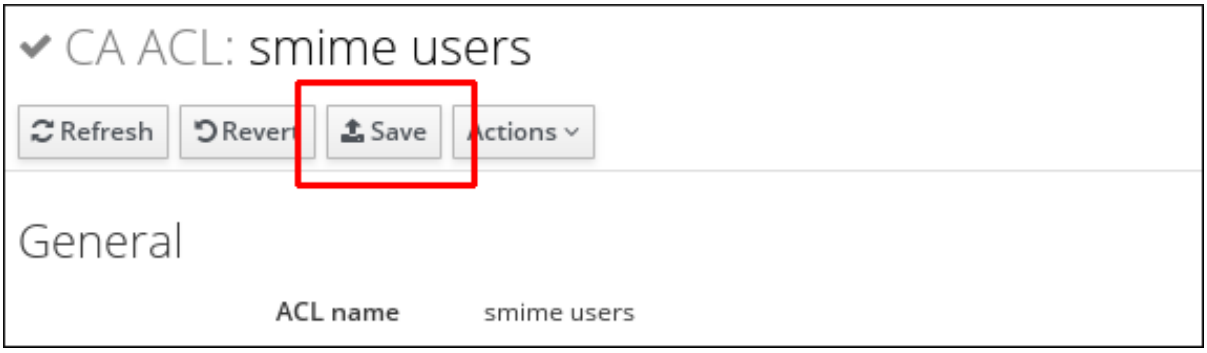
1. *单击 CA ACL 的名称以打开 CA ACL 配置页面。*

2. *在 CA ACL 配置页面中，填写所需信息。*

   Profiles 和 Permitted to have certificate issued 部分，允许您将 CA ACL 与证书配置文件、用户或用户组、主机或主机组或服务相关联。您可以使用 Add 按钮添加这些对象，或者选择 Anyone 选项将 CA ACL 与所有用户、主机或服务关联。

3. *单击 Save 以确认新配置。*

图 24.10. 在 Web UI 中修改 CA ACL 规则



24.6. 使用证书配置文件和 ACL 来向 IDM CA 签发用户证书

当证书颁发机构访问控制列表(CA ACL)允许时，用户可以为自己请求证书。以下流程使用证书配置集和 CA ACL，它们在 第 24.4 节 “证书配置文件” 和 第 24.5 节 “证书颁发机构 ACL 规则” 中单独描述。有关使用证书配置文件和 CA ACL 的详情，请查看这些部分。

从命令行向用户签发证书

1. 创建或导入用于处理用户证书请求的新自定义证书配置集。例如：

```
$ ipa certprofile-import certificate_profile --file=certificate_profile.cfg --store=True
```

2. 添加将用于向用户条目请求证书的新证书颁发机构(CA)ACL。例如：

```
$ ipa caacl-add users_certificate_profile --usercat=all
```

3. 将自定义证书配置文件添加到 CA ACL。

```
$ ipa caacl-add-profile users_certificate_profile --certprofiles=certificate_profile
```

4. 为用户生成证书请求。例如，使用 OpenSSL:

```
$ openssl req -new -newkey rsa:2048 -days 365 -nodes -keyout private.key -out cert.csr -subj '/CN=user'
```

5. 运行 ipa cert-request 命令，使 IdM CA 为用户发布新证书。

```
$ ipa cert-request cert.csr --principal=user --profile-id=certificate_profile
```

（可选）将 --ca sub-CA_name 选项传给命令，以从子 CA 请求证书，而不是 root CA ipa。

要确保新发布的证书分配给用户，您可以使用 ipa user-show 命令：

```
$ ipa user-show user
  User login: user
  ...
  Certificate: MIICfzCCAWcCAQA...
  ...
```

在 Web UI 中向用户发布证书

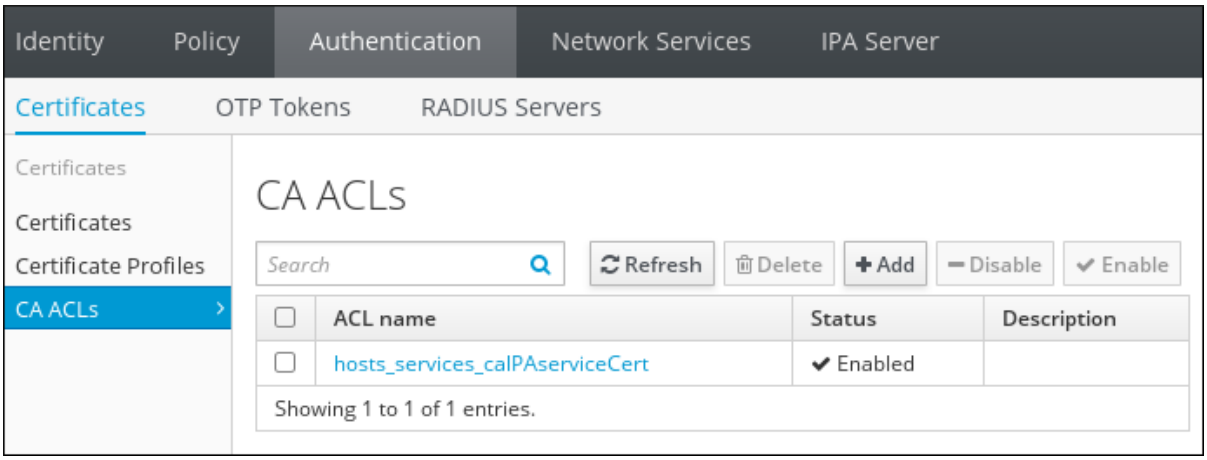
1. 创建或导入用于处理用户证书请求的新自定义证书配置集。只能从命令行导入配置集，例如：

```
$ ipa certprofile-import certificate_profile --file=certificate_profile.txt --store=True
```

有关证书配置集的详情请参考 第 24.4 节 “证书配置文件”。

2. 在 Web UI 中，在 Authentication 选项卡下，打开 CA ACL 部分。

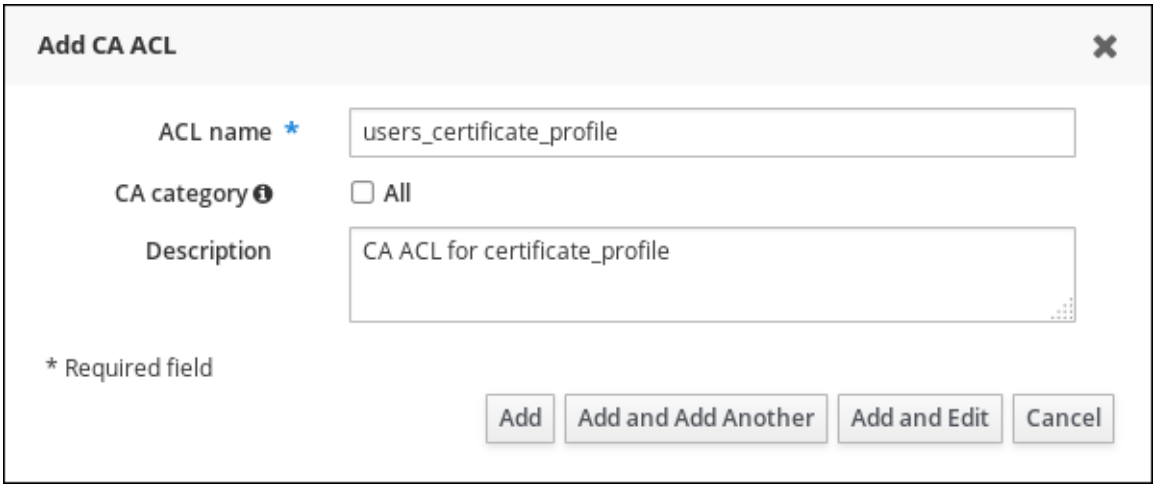
图 24.11. Web UI 中的 CA ACL 规则管理



点证书颁发机构(CA) ACL 列表顶部的 Add 来添加新的 CA ACL，允许请求用户条目的证书。

a. 在打开的 Add CA ACL 窗口中，填写有关新 CA ACL 所需的信息。

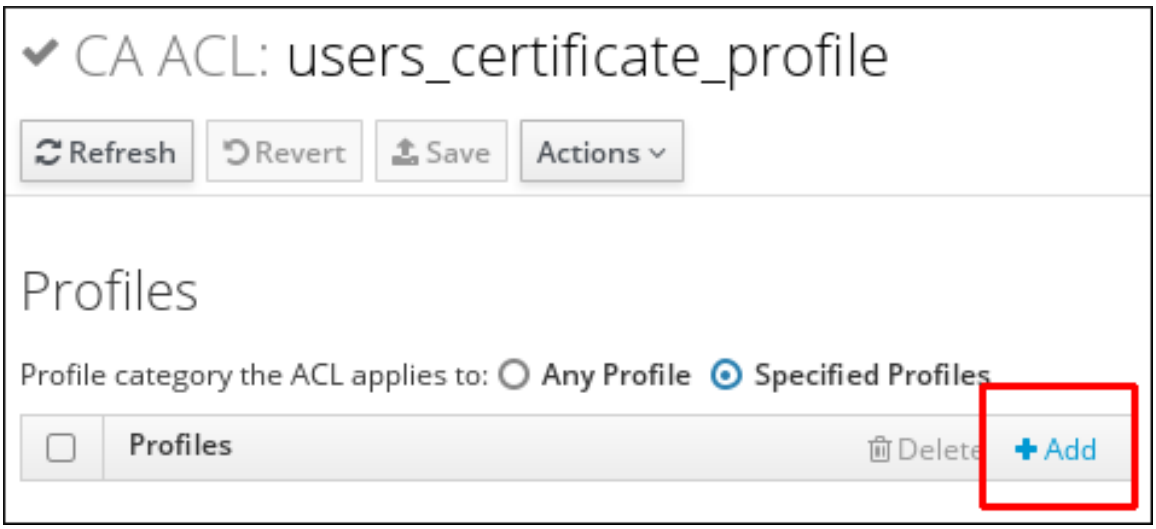
图 24.12. 添加新 CA ACL



然后，单击 **Add and Edit** 以直接进入 **CA ACL** 配置页面。

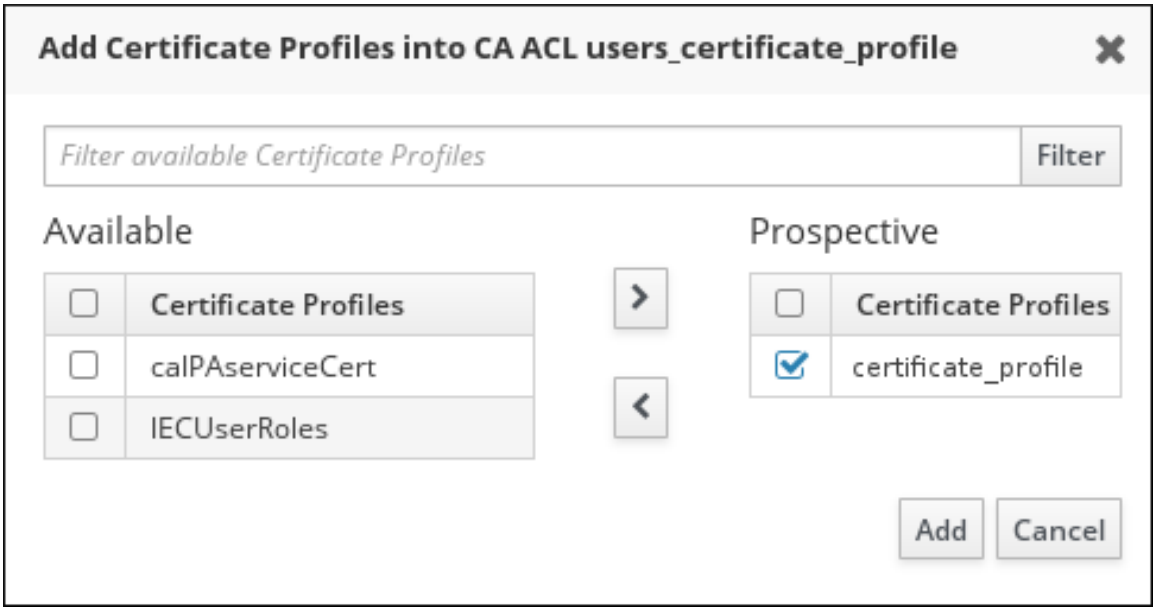
b. 在 **CA ACL** 配置页面中，滚动到 **Profiles** 部分，然后点击 **profile** 列表顶部的 **Add**。

图 24.13. 在 CA ACL 中添加证书配置集



c. 选择配置文件并将其移到 **Prospective** 列中，将自定义证书配置文件添加到 **CA ACL** 中。

图 24.14. 选择证书配置文件

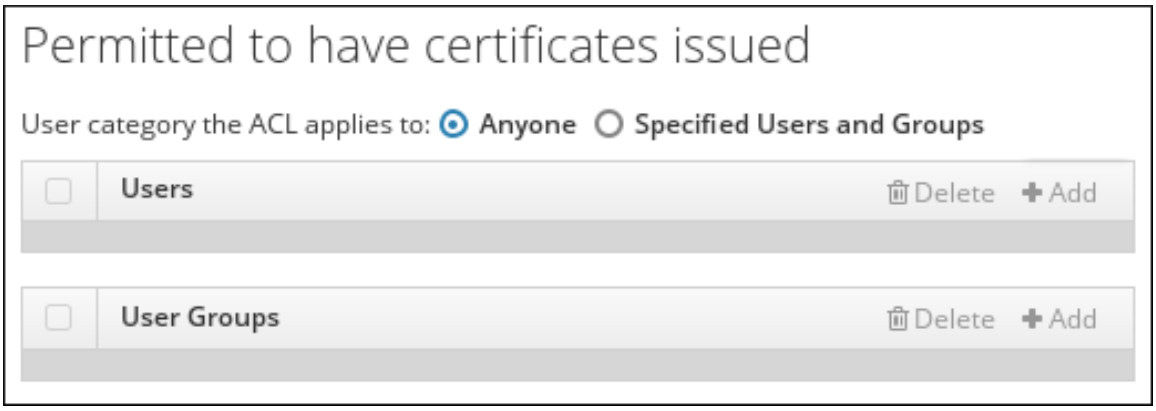


然后点击 **Add**。

d. 滚动到 **Permitted to have certificate issued** 部分，以将 **CA ACL** 与用户或用户组关联。

您可以使用 Add 按钮添加用户或组，或者选择 Anyone 选项将 CA ACL 与所有用户关联。

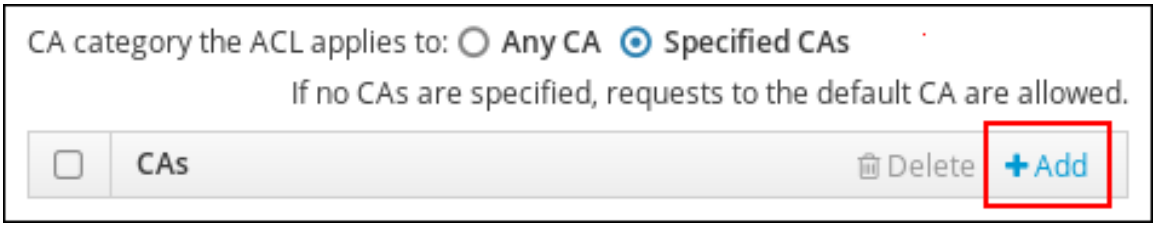
图 24.15. 在 CA ACL 中添加用户



e. 在 Permitted to have certificate issued 部分，您可以将 CA ACL 与一个或多个 CA 关联。

您可以使用 Add 按钮添加 CA，或者选择 任何 CA 选项将 CA ACL 与所有 CA 关联。

图 24.16. 在 CA ACL 中添加 CA



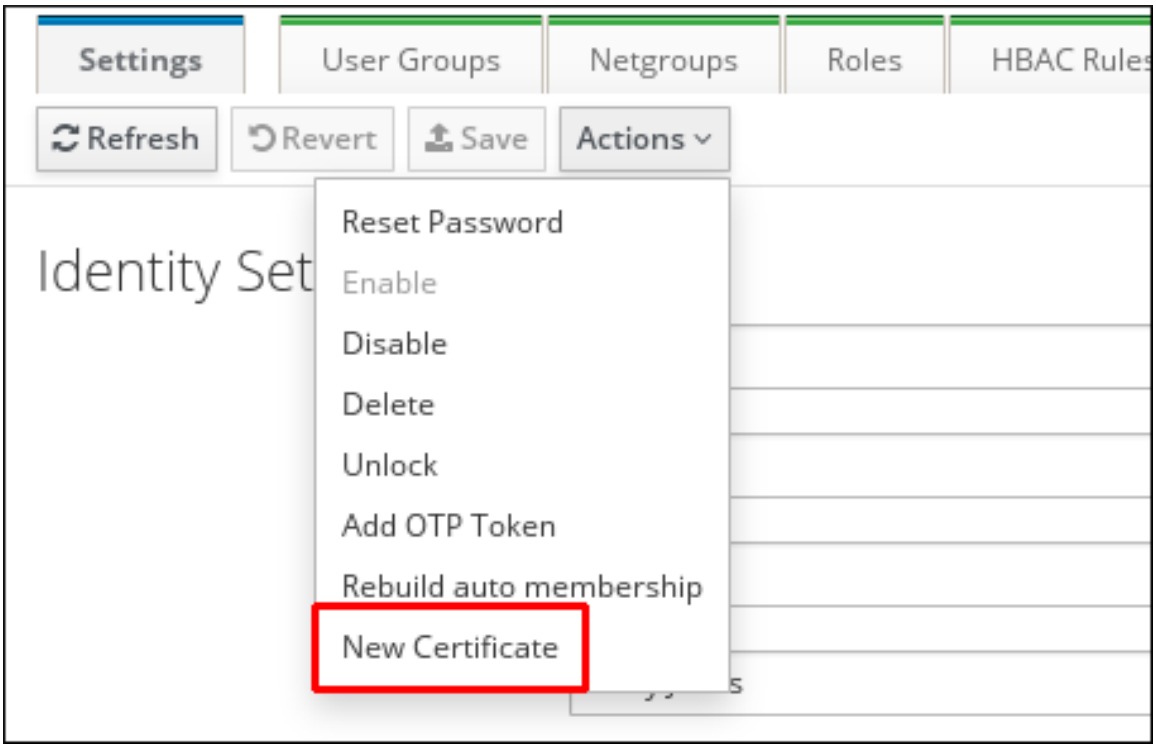
f. 在 CA ACL 配置页面的顶部，单击 Save 以确认 CA ACL 的更改。

3. 为用户请求新证书。

a. 在 Identity 选项卡和 Users 子选项卡下，选择请求证书的用户。单击用户的用户名，以打开用户输入配置页面。

b. 单击用户配置页面顶部的 Actions，然后单击 New Certificate。

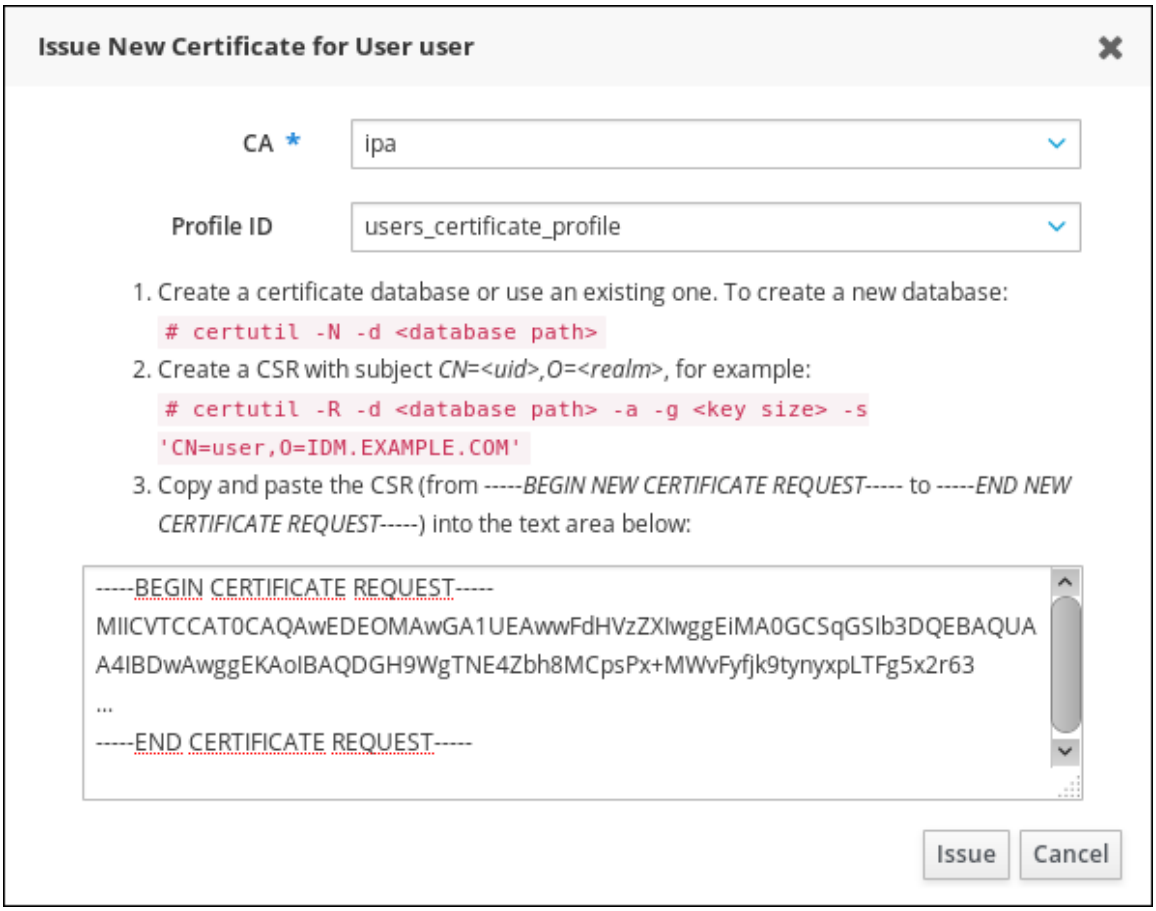
图 24.17. 为用户请求证书



c.

填写所需信息。

图 24.18. 向用户发布证书



然后单击 **Issue**。

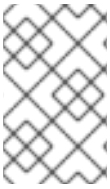
之后，新发布的证书在用户配置页面中可见。

第 25 章 使用 VAULT 存储身份验证 SECRET

密码库是用于存储、检索、共享和恢复机密的安全位置。机密是安全敏感型数据，只能被有限人或实体访问。例如，secret 包括：

- 密码
- pins
- SSH 私有密钥

用户和服务可以从注册到身份管理(IdM)域的任何计算机访问存储在密码库中的机密。



注意

Vault 只能从命令行使用，不能从 IdM Web UI 使用。

Vault 的用例包括：

存储用户的个人 secret

详情请查看 第 25.4 节 “存储用户的个人机密”。

为服务存储 secret

详情请查看 第 25.5 节 “在 Vault 中存储服务 secret”。

存储供多个用户使用的通用 secret

详情请查看 第 25.6 节 “为多个用户存储通用 secret”。

请注意，要使用 vault，必须满足 第 25.2 节 “使用 Vault 的先决条件” 中描述的条件。

25.1. VAULT 如何工作

25.1.1. Vault 所有者、成员和管理员

IdM 区分以下 vault 用户类型：

Vault 所有者

Vault 所有者是具有密码库上基本管理特权的用户或服务。例如，vault 所有者可以修改密码库的属性或添加新的 vault 成员。

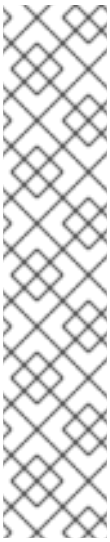
每个密码库必须至少有一个所有者。个密码库也可以有多个所有者。

Vault 成员

Vault 成员是用户或服务，可以访问由其他用户或服务创建的库。

Vault 管理员

Vault 管理员对所有密码库具有不受限制的访问权限，并允许执行所有 vault 操作。



注意

对称和非对称密码库通过密码或密钥进行保护，并应用特殊的访问控制规则（请参阅 第 25.1.2 节 “Standard、Symmetric 和 nonsymmetric Vaults”）。管理员必须满足以下规则：

- 使用对称和非对称库访问 secret
- 更改或重置 vault 密码或密钥

Vault 管理员是具有 Vault 管理员特权的任何用户。有关定义用户权限的信息，请参阅 第 10.4 节 “定义基于角色的访问控制”。

特定所有者和成员特权取决于密码库的类型。详情请查看 第 25.1.2 节 “Standard、Symmetric 和 nonsymmetric Vaults”。

Vault 用户

有些命令的输出（如 ipa vault-show 命令）也会为用户 vault 显示 Vault 用户 ：

```
$ ipa vault-show my_vault
  Vault name: my_vault
  Type: standard
  Owner users: user
  Vault user: user
```

vault 用户代表密码库所在的容器的用户。有关 vault 容器和用户 vault 的详情，请查看 第 25.1.4 节 “Vault 容器的不同类型” 和 第 25.1.3 节 “用户、服务和共享 Vault”。

25.1.2. Standard、Symmetric 和 nonsymmetric Vaults

以下 vault 类型基于安全性和访问控制级别：

标准库

Vault 所有者和 vault 成员可以存档和检索机密，而无需使用密码或密钥。

对称库

密码库中的机密通过对称密钥进行保护。Vault 成员和 vault 所有者可以存档和检索机密，但它们必须提供 vault 密码。

非对称库

密码库中的机密通过非对称密钥进行保护。用户使用公钥归档机密，再使用私钥检索该机密。Vault 成员只能存档机密，而 vault 拥有者可以同时存档和检索机密。

25.1.3. 用户、服务和共享 Vault

以下 vault 类型基于所有权：

用户密码库：用户的专用密码库

所有者：单个用户.

任何用户都可以拥有一个或多个用户库。

服务库：服务的专用密码库

所有者：一项服务.

任何服务都可以拥有一个或多个服务库。

共享库

所有者：创建密码库的 vault 管理员.其他 vault 管理员也拥有对密码库的完全访问权限。

共享库可供多个用户或服务使用。

25.1.4. Vault 容器的不同类型

vault 容器是密码库的集合。

IdM 提供以下默认 vault 容器：

用户容器：用户的私有容器

此容器存储：特定用户的用户库。

服务容器：服务的私有容器

此容器存储：特定服务的服务库。

共享容器

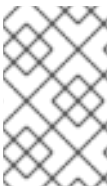
此容器存储：可由多个用户或服务共享的库。

当为用户或服务创建第一个私有密码库时，IdM 会自动为每个用户或服务创建用户和服务容器。删除用户或服务后，IdM 会删除容器及其内容。

25.2. 使用 VAULT 的先决条件

要启用 vault，请在 IdM 域的一个或多个服务器中安装密钥恢复授权机构(KRA)证书系统组件：

```
# ipa-kra-install
```



注意

要使 Vault 服务具有高可用性，请在两个 IdM 服务器或更高服务器上安装 KRA。

25.3. 获取 VAULT 命令帮助

显示用于管理 vault 和 vault 容器的所有命令：

```
$ ipa help vault
```

要显示特定命令的详细帮助信息，请在命令中添加 **--help** 选项：

```
$ ipa vault-add --help
```

带有 **vault not found** 错误的 vault 命令故障

有些命令要求您使用以下选项指定 vault 的所有者或类型：

- **--user 或 --service** 指定您要查看的 vault 的所有者

  ```
  $ ipa vault-show user_vault --user user
  ```

- **--shared** 指定您要查看的 vault 是一个共享库

例如，如果您试图在不添加 **--user** 的情况下查看其他用户的 vault，IdM 会通知您它没有找到库：

```
[admin@server ~]$ ipa vault-show user_vault
ipa: ERROR: user_vault: vault not found
```

25.4. 存储用户的个人机密

本节介绍用户如何创建一个或多个专用库来安全地存储个人机密。然后，用户在需要时在域中的任何计算机上检索 secret。例如，用户可以将个人证书归档到密码库中，从而安全地将证书存储在中央位置。

本节包括以下步骤：

- 第 25.4.1 节 “归档用户的个人机密”
- 第 25.4.2 节 “检索用户的个人机密”

在流程中：

- 用户是想要创建密码库的用户
- my_vault 是用于存储用户证书的库
- vault 类型 是标准的，因此访问存档证书不需要用户提供 vault 密码
- secret.txt 是包含用户希望存储在密码库中的证书的文件
- secret_exported.txt 是用户将存档证书导出到的文件

25.4.1. 归档用户的个人机密

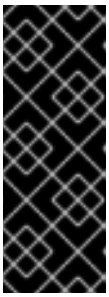
创建专用用户密码库，并将您的证书存储在其中。vault 类型是 standard，它可确保在访问证书时您不需要进行身份验证。

1. 以用户 身份登录：

```
$ kinit user
```

2. 使用 ipa vault-add 命令来创建标准密码库：

```
$ ipa vault-add my_vault --type standard
----------------------
Added vault "my_vault"
----------------------
  Vault name: my_vault
  Type: standard
  Owner users: user
  Vault user: user
```



重要

确保同一用户创建了用户的第一个用户密码库。例如，如果其他用户（如 admin）为 user1 创建第一个用户 vault，则用户 vault 容器的所有者也是 admin，user1 无法访问用户 vault 或创建新用户库。另请参阅 第 B.5.1 节 “用户无法访问其 Vault，因为使用无效"添加"权限”。

3. 使用 ipa vault-archive --in 命令将 secret.txt 文件归档到密码库中：

```
$ ipa vault-archive my_vault --in secret.txt
-----------------------------------
Archived data into vault "my_vault"
-----------------------------------
```

注意

一个密码库只能存储一个 secret。

25.4.2. 检索用户的个人机密

从您的私有标准密码库导出证书。

1. 以用户 身份登录：

```
$ kinit user
```

2. 使用 ipa vault-retrieve --out 命令检索密码库的内容，并将它们保存到 secret_exported.txt 文件中。

```
$ ipa vault-retrieve my_vault --out secret_exported.txt
---------------------------------------
Retrieved data from vault "my_vault"
---------------------------------------
```

25.5. 在 VAULT 中存储服务 SECRET

本节介绍管理员如何使用 vault 将服务机密安全地存储在中央位置。服务机密使用服务公钥加密。然后，服务使用其在域中的任何计算机上的私钥来检索机密。只有服务和管理员可以访问该 secret。

本节包括以下步骤：

- 第 25.5.1 节 “创建用户 Vault 以存储服务密码”
- 第 25.5.2 节 “从用户 Vault 置备服务密码到服务实例”
- 第 25.5.3 节 “为服务实例检索服务密码”
- 第 25.5.4 节 “更改服务 Vault 密码”

在流程中：

- admin 是管理服务密码的管理员
- http_password 是管理员创建的私有用户库的名称
- password.txt 是包含服务密码的文件
- password_vault 是为服务创建的库
- http/server.example.com 是正在归档密码的服务

- *service-public.pem 是用于加密 password_vault中存储的密码的服务公钥*

25.5.1. 创建用户 Vault 以存储服务密码

创建管理员拥有的用户 vault，并使用它来存储服务密码。vault 类型是标准的，它可确保管理员在访问密码库的内容时无需进行身份验证。

1. *以管理员身份登录：*

```
$ kinit admin
```

2. *创建标准用户库：*

```
$ ipa vault-add http_password --type standard
----------------------------
Added vault "http_password"
----------------------------
  Vault name: http_password
  Type: standard
  Owner users: admin
  Vault user: admin
```

3. *将服务密码归档到密码库中：*

```
$ ipa vault-archive http_password --in password.txt
----------------------------------------
Archived data into vault "http_password"
----------------------------------------
```

警告

将密码归档到密码库后，从系统中删除 password.txt。

25.5.2. 从用户 Vault 置备服务密码到服务实例

使用为服务创建的非对称密码库，将服务密码调配到服务实例。

1. **以管理员身份登录：**

```
$ kinit admin
```

2. **获取服务实例的公钥。例如，使用 openssl 工具：**

   a. **生成 service-private.pem 私钥。**

```
$ openssl genrsa -out service-private.pem 2048
Generating RSA private key, 2048 bit long modulus
.+++
.....................................+++
e is 65537 (0x10001)
```

   b. **根据私钥生成 service-public.pem 公钥。**

```
$ openssl rsa -in service-private.pem -out service-public.pem -pubout
writing RSA key
```

3. **创建非对称 vault 作为服务实例库，并提供公钥：**

```
$ ipa vault-add password_vault --service HTTP/server.example.com --type asymmetric --
public-key-file service-public.pem
----------------------------
Added vault "password_vault"
----------------------------
Vault name: password_vault
Type: asymmetric
Public key: LS0tLS1C...S0tLS0tCg==
Owner users: admin
Vault service: HTTP/server.example.com@EXAMPLE.COM
```

   归档到密码库的密码将通过 密钥进行保护。

4. **从管理员的专用密码库中检索服务密码，然后将其归档到新服务库中：**

```
$ ipa vault-retrieve http_password --out password.txt
----------------------------------------
Retrieved data from vault "http_password"
----------------------------------------
```

```
$ ipa vault-archive password_vault --service HTTP/server.example.com --in password.txt
----------------------------------
Archived data into vault "password_vault"
----------------------------------
```

这将使用服务实例公钥加密密码。

警告

将密码归档到密码库后，从系统中删除 password.txt。

对需要密码的每个服务实例重复这些步骤。为每个服务实例创建新的非对称密码库。

25.5.3. 为服务实例检索服务密码

服务实例可以使用本地存储的服务私钥检索服务 vault 密码。

1. 以管理员身份登录：

```
$ kinit admin
```

2. 获取该服务的 Kerberos ticket：

```
# kinit HTTP/server.example.com -k -t /etc/httpd/conf/ipa.keytab
```

3. 检索服务 vault 密码：

```
$ ipa vault-retrieve password_vault --service HTTP/server.example.com --private-key-file
service-private.pem --out password.txt
```

```
-----------------------------------
Retrieved data from vault "password_vault"
-----------------------------------
```

25.5.4. 更改服务 Vault 密码

如果服务实例遭到破坏，请通过更改服务 vault 密码将它隔离，然后仅将新密码重新调配到非编译服务实例。

1. **在管理员的用户库中归档新密码：**

```
$ ipa vault-archive http_password --in new_password.txt
-----------------------------------------
Archived data into vault "http_password"
-----------------------------------------
```

这会覆盖存储在密码库中的当前密码。

2. **将新密码重新调配到除被破坏的实例外的每个服务实例。**

   a. **从管理员的 vault 中检索新密码：**

```
$ ipa vault-retrieve http_password --out password.txt
-----------------------------------------
Retrieved data from vault "http_password"
-----------------------------------------
```

   b. **将新密码归档到服务实例库中：**

```
$ ipa vault-archive password_vault --service HTTP/server.example.com --in password.txt
-----------------------------------
Archived data into vault "password_vault"
-----------------------------------
```

警告

将密码归档到密码库后，从系统中删除 password.txt。

25.6. 为多个用户存储通用 SECRET

本节介绍管理员如何创建共享密码库并允许其他用户访问密码库中的机密。管理员会将常用密码归档到密码库中，其他用户能够检索域内任何计算机上的密码。

本节包括以下步骤：

- 第 25.6.2 节 “以 Member 用户身份从共享 Vault 检索 secret”
- 第 25.6.1 节 “使用通用 secret 创建共享 Vault”

在流程中：

- shared_vault 是用于存储通用密码的库
- admin 是创建共享密码库的管理员
- vault 类型是 标准，因此访问存档的密码不需要用户提供 vault 密码
- secret.txt 是包含通用 secret 的文件
- user1 和 user2 是允许访问密码库的用户

25.6.1. 使用通用 secret 创建共享 Vault

创建一个共享密码库，并使用它来存储共同的机密。添加将要作为 vault 成员访问机密的用户。vault 类型是标准的，它可确保任何访问 secret 的用户都不需要进行身份验证。

1. *以管理员身份登录：*

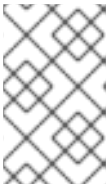
```
$ kinit admin
```

2. *创建共享库：*

```
$ ipa vault-add shared_vault --shared --type standard
----------------------------
Added vault "shared_vault"
----------------------------
  Vault name: shared_vault
  Type: standard
  Owner users: admin
  Shared vault: True
```

3. *将机密存档到密码库中。添加 --shared 选项以指定 vault 位于共享容器中：*

```
$ ipa vault-archive shared_vault --shared --in secret.txt
-----------------------------------
Archived data into vault "shared_vault"
-----------------------------------
```



注意

个密码库只能存储一个 secret。

4. *添加 user1 和 user2 作为 vault 成员：*

```
ipa vault-add-member shared_vault --shared --users={user1,user2}
Vault name: shared_vault
Type: standard
Owner users: admin
Shared vault: True
Member users: user1, user2
-------------------------
Number of members added 2
-------------------------
```

25.6.2. 以 Member 用户身份从共享 Vault 检索 secret

以 vault 的成员用户身份登录，再使用密码从密码库中导出 文件。

1. 以 user1 成员用户身份登录：

```
$ kinit user1
```

2. 从共享密码库中检索 secret：

```
$ ipa vault-retrieve shared_vault --shared --out secret_exported.txt
----------------------------------------
Retrieved data from vault "shared_vault"
----------------------------------------
```

25.7. 更改 VAULT 的密码或公钥

vault 的所有者可以更改 vault 的密码。根据密码库是对称还是非对称，该命令会有所不同：

- 更改对称密码库：

```
# ipa vault-mod --change-password
Vault name: example_symmetric_vault
Password: old_password
New password: new_password
Enter New password again to verify: new_password
----------------------
Modified vault "example_symmetric_vault"
----------------------
  Vault name: example_symmetric_vault
  Type: symmetric
  Salt: dT+M+4ik/ltgnpstmCG1sw==
  Owner users: admin
  Vault user: admin
```

- 更改非对称库的公钥：

```
# ipa vault-mod example_asymmetric_vault --private-key-file=old_private_key.pem --public-key-file=new_public_key.pem
-------------------------------
Modified vault "example_assymmetric_vault"
```

```
--------------------------------
 Vault name: example_assymmetric_vault
 Typ: asymmetric
 Public key: ...
 Owner users: admin
 Vault user: admin
```

第 26 章 管理证书和证书颁发机构

26.1. 轻量级子 CA

如果您的 IdM 安装配置了集成证书系统(CS)证书颁发机构(CA)，您可以创建轻量级子 CA。它们允许您配置服务，如虚拟专用网络(VPN)网关，以仅接受由一个子 CA 发布的证书。同时，您可以将其他服务配置为仅接受由不同子 CA 或 root CA 发布的证书。

如果您撤销子 CA 的中间证书，此子 CA 发布的所有证书都自动无效。

如果您使用集成的 CA 设置 IdM，则自动创建的 ipa CA 是证书系统的根 CA。您创建的所有子 CA 都从属到这个 root CA。

26.1.1. 创建轻量级子 CA

有关创建子 CA 的详情，请参考

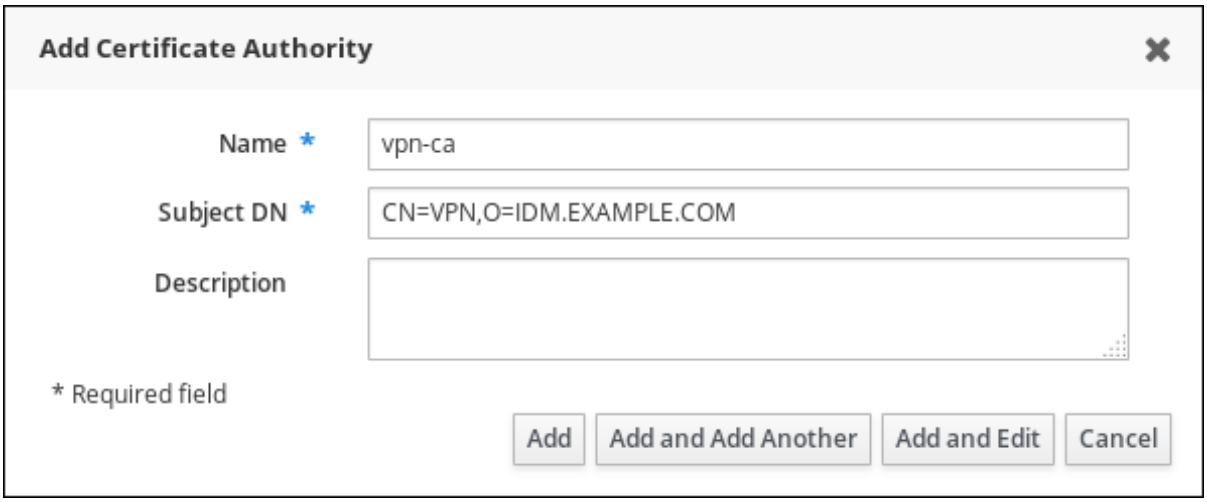
- “从 Web UI 创建子 CA”一节
- “从命令行创建子 CA”一节

从 Web UI 创建子 CA

要创建一个名为 vpn-ca 的新子 CA ：

1. 打开 Authentication 选项卡，然后选择 证书 子选项卡。
2. 选择 证书授权 并单击 添加。
3. 输入 CA 的名称和主题 DN。

图 26.1. 添加 CA



主题 DN 在 IdM CA 基础架构中必须是唯一的。

从命令行创建子 CA

要创建一个名为 vpn-ca 的新子 CA，请输入：

```
[root@ipaserver ~]# ipa ca-add vpn-ca --subject="CN=VPN,O=IDM.EXAMPLE.COM"
-------------------
Created CA "vpn-ca"
-------------------
  Name: vpn-ca
  Authority ID: ba83f324-5e50-4114-b109-acca05d6f1dc
  Subject DN: CN=VPN,O=IDM.EXAMPLE.COM
  Issuer DN: CN=Certificate Authority,O=IDM.EXAMPLE.COM
```

Name

CA 的名称。

颁发机构 ID

自动创建，为 CA 单独创建 ID。

主题 DN

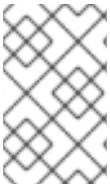
主题区分名称(DN)。主题 DN 在 IdM CA 基础架构中必须是唯一的。

签发者 DN

发布子 CA 证书的父 CA。所有子 CA 都是作为 IdM root CA 的子 CA 创建的。

要验证新的 CA 签名证书是否已成功添加到 IdM 数据库中，请运行：

```
[root@ipaserver ~]# certutil -d /etc/pki/pki-tomcat/alias/ -L

Certificate Nickname                                         Trust Attributes
                                                             SSL,S/MIME,JAR/XPI

caSigningCert cert-pki-ca                                    CTu,Cu,Cu
Server-Cert cert-pki-ca                                      u,u,u
auditSigningCert cert-pki-ca                                 u,u,Pu
caSigningCert cert-pki-ca ba83f324-5e50-4114-b109-acca05d6f1dc u,u,u
ocspSigningCert cert-pki-ca                                  u,u,u
subsystemCert cert-pki-ca                                    u,u,u
```



注意

当新 CA 证书安装证书系统实例时，它们会自动传输到所有副本。

26.1.2. 删除轻量级子 CA

有关删除子 CA 的详情，请参考

- “从 Web UI 中删除子 CA”一节
- “从命令行删除子 CA”一节

从 Web UI 中删除子 CA

1. 打开 Authentication 选项卡，然后选择 证书 子选项卡。
2. 选择 证书颁发机构。
3. 选择要删除的子 CA，然后单击" 删除 "。
4. 单击 Delete 确认。

从命令行删除子 CA

要删除子 CA，请输入：

```
[root@ipaserver ~]# ipa ca-del vpn-ca
-------------------
Deleted CA "vpn-ca"
-------------------
```

26.2. 续订证书

有关以下方面的详情：

- 自动证书续订，请参见 第 26.2.1 节 “自动续订证书”
- 手动证书续订，请参阅 第 26.2.2 节 “手动续订 CA 证书”

26.2.1. 自动续订证书

certmonger 服务会在过期前自动更新以下证书 28 天：

- IdM CA 作为 root CA 发布的 CA 证书
- 由内部 IdM 服务使用的集成的 IdM CA 发布的子系统和服务器证书

要自动更新子 CA CA 证书，必须在 certmonger 跟踪列表中列出它们。更新跟踪列表：

```
[root@ipaserver ~]# ipa-certupdate
trying https://idmserver.idm.example.com/ipa/json
Forwarding 'schema' to json server 'https://idmserver.idm.example.com/ipa/json'
trying https://idmserver.idm.example.com/ipa/json
Forwarding 'ca_is_enabled' to json server 'https://idmserver.idm.example.com/ipa/json'
Forwarding 'ca_find/1' to json server 'https://idmserver.idm.example.com/ipa/json'
Systemwide CA database updated.
Systemwide CA database updated.
The ipa-certupdate command was successful
```

注意

如果您使用外部 CA 作为 root CA，则必须手动更新证书，如 第 26.2.2 节 “手动续订 CA 证书” 所述。certmonger 服务无法自动更新外部 CA 签名的证书。

如需有关 certmonger 如何监控证书过期日期的更多信息，请参阅 系统级身份验证指南中的使用 certmonger 跟踪证书。

要验证自动续订是否按预期工作，请检查 /var/log/messages 文件中的 certmonger 日志消息：

- 续订证书后，certmonger 记录消息类似于以下内容，以指示续订操作已成功或失败：

```
Certificate named "NSS Certificate DB" in token "auditSigningCert cert-pki-ca" in database "/var/lib/pki-ca/alias" renew success
```

- 当证书接近其过期时，certmonger 会记录以下信息：

```
certmonger: Certificate named "NSS Certificate DB" in token "auditSigningCert cert-pki-ca" in database "/var/lib/pki-ca/alias" will not be valid after 20160204065136.
```

26.2.2. 手动续订 CA 证书

您可以使用 ipa-cacert-manage 工具手动续订：

- 自签名 IdM CA 证书
- 外部签名的 IdM CA 证书

使用 ipa-cacert-manage renewal 命令更新的证书使用与旧证书相同的密钥对和主题名称。续订证书不会删除之前的版本来启用证书滚动。

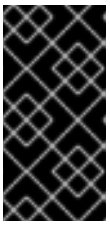
详情请查看 ipa-cacert-manage(1) man page。

26.2.2.1. 手动续订自签名 IdM CA 证书

1. 运行 ipa-cacert-manage renewal 命令。命令不要求您指定证书的路径。

2. 更新的证书现在存在于 LDAP 证书存储中，并存在于 /etc/pki/pki-tomcat/alias NSS 数据库中。

3. 在所有服务器和客户端上运行 ipa-certupdate 工具，以使用 LDAP 中的新证书的信息来更新它们。您必须为每个服务器和客户端单独运行 ipa-certupdate。



重要

手动安装证书后始终运行 ipa-certupdate。如果不这样做，则证书不会分发到其他计算机上。

要确保正确安装了更新的证书，请使用 certutil 实用程序列出数据库中的证书。例如：

```
# certutil -L -d /etc/pki/pki-tomcat/alias
```

26.2.2.2. 手动续订外部签名的 IdM CA 证书

1. 运行 ipa-cacert-manage renewal --external-ca 命令。

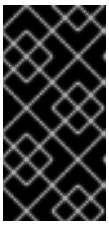
2. 该命令创建 /var/lib/ipa/ca.csr CSR 文件。将 CSR 提交到外部 CA 以获取签发更新 CA 证书。

3. 再次运行 ipa-cacert-manage renewal，这一次使用 --external-cert-file 选项指定更新的 CA 证书和外部 CA 证书链文件。例如：

```
# ipa-cacert-manage renew --external-cert-file=/tmp/servercert20110601.pem --external-cert-file=/tmp/cacert.pem
```

4. 更新的 CA 证书和外部 CA 证书链现在存在于 LDAP 证书存储中，以及 /etc/pki/pki-tomcat/alias/ NSS 数据库中。

5. 在所有服务器和客户端上运行 ipa-certupdate 工具，以使用 LDAP 中的新证书的信息来更新它们。您必须为每个服务器和客户端单独运行 ipa-certupdate。



重要

手动安装证书后始终运行 **ipa-certupdate**。如果不这样做，则证书不会分发到其他计算机上。

要确保正确安装了更新的证书，请使用 **certutil** 实用程序列出数据库中的证书。例如：

```
# certutil -L -d /etc/pki/pki-tomcat/alias/
```

26.2.3. IdM 离线时续订过期的系统证书

如果系统证书已过期，IdM 无法启动。IdM 支持使用 **ipa-cert-fix** 工具更新系统证书。

前提条件

- 通过在主机上输入 **ipactl start --ignore-service-failures** 命令来确保 LDAP 服务正在运行。

过程 26.1. 在 IdM 服务器上续订所有过期的系统证书

1. 在 IdM 域中的 CA 中：

   a. 启动 **ipa-cert-fix** 工具以分析系统并列出过期的证书：

   ```
   # ipa-cert-fix
   ...
   The following certificates will be renewed:

   Dogtag sslserver certificate:
     Subject: CN=ca1.example.com,O=EXAMPLE.COM 201905222205
     Serial:  13
     Expires: 2019-05-12 05:55:47
   ...
   Enter "yes" to proceed:
   ```

   b. 输入 **yes** 以开始续订过程：

```
Enter "yes" to proceed: yes
Proceeding.
Renewed Dogtag sslserver certificate:
  Subject: CN=ca1.example.com,O=EXAMPLE.COM 201905222205
  Serial:  268369925
  Expires: 2021-08-14 02:19:33
...

Becoming renewal master.
The ipa-cert-fix command was successful
```

ipa-cert-fix 更新所有过期证书前最多可能需要一分钟的时间。

注意

如果您在不是续订 master 的 CA 主机上运行 ipa-cert-fix 工具，并且实用程序续订共享证书，则此主机会自动成为域中的新续订 master。域中必须始终只有一个续订 master，以避免不一致。

c. **（可选）验证所有服务是否都在运行：**

```
# ipactl status
Directory Service: RUNNING
krb5kdc Service: RUNNING
kadmin Service: RUNNING
httpd Service: RUNNING
ipa-custodia Service: RUNNING
pki-tomcatd Service: RUNNING
ipa-otpd Service: RUNNING
ipa: INFO: The ipactl command was successful
```

2. **在 IdM 域中的其他服务器上：**

a. **使用 --force 参数重启 IdM：**

```
# ipactl restart --force
```

使用 --force 参数时，ipactl 工具会忽略单独的启动失败。例如，如果服务器也是 CA，pki-tomcat 服务将无法启动。这是预期并忽略的，因为使用了 --force 参数。

b.

重启后，验证 **certmonger** 服务是否更新证书：

```
# getcert list | egrep '^Request|status:|subject:'
Request ID '20190522120745':
        status: MONITORING
        subject: CN=IPA RA,O=EXAMPLE.COM 201905222205
Request ID '20190522120834':
        status: MONITORING
        subject: CN=Certificate Authority,O=EXAMPLE.COM 201905222205
...
```

请注意，在 **certmonger** 更新副本上的共享证书前可能需要一些时间。

c.

如果服务器也是 **CA**，以上命令会报告 **pki-tomcat** 服务使用的证书的 **CA_UNREACHABLE** ：

```
Request ID '20190522120835':
        status: CA_UNREACHABLE
        subject: CN=ca2.example.com,O=EXAMPLE.COM 201905222205
...
```

要续订此证书，请使用 **ipa-cert-fix** 工具：

```
# ipa-cert-fix
Dogtag sslserver certificate:
  Subject: CN=ca2.example.com,O=EXAMPLE.COM
  Serial:  3
  Expires: 2019-05-11 12:07:11

Enter "yes" to proceed: yes
Proceeding.
Renewed Dogtag sslserver certificate:
  Subject: CN=ca2.example.com,O=EXAMPLE.COM 201905222205
  Serial:  15
  Expires: 2019-08-14 04:25:05

The ipa-cert-fix command was successful
```

26.3. 手动安装 CA 证书

要将新证书安装到 IdM，请使用 **ipa-cacert-manage install** 命令。例如，命令允许您在接近其到期日期时更改当前证书。

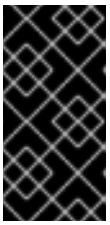
1.

运行 ipa-cacert-manage install 命令，并指定包含证书的文件的路径。该命令接受 PEM 格式的证书文件：

```
[root@server ~]# ipa-cacert-manage install /etc/group/cert.pem
```

证书现在包括在 LDAP 证书存储中。

2. 在所有服务器和客户端上运行 ipa-certupdate 工具，以使用 LDAP 中的新证书的信息来更新它们。您必须为每个服务器和客户端单独运行 ipa-certupdate。



重要

手动安装证书后始终运行 ipa-certupdate。如果不这样做，则证书不会分发到其他计算机上。

ipa-cacert-manage install 命令可使用以下选项：

-n

指定证书的 nickname；默认值是证书的主题名称

-t

以 certutil 格式指定证书的信任标志；默认值为 C， 即。有关指定信任标记的格式的详情请参考 ipa-cacert-manage(1) man page。

26.4. 更改证书链

您可以使用 ipa-cacert-manage 续订 CA 证书来修改证书链。

自签名 CA 证书 → 外部签名的 CA 证书

将 --external-ca 选项添加到 ipa-cacert-manage 续订。这会续订自签名 CA 证书作为外部签名的 CA 证书。

有关使用这个选项运行命令的详情请参考 第 26.2.2 节 “手动续订 CA 证书”。

外部签名的 CA 证书 → 自签名 CA 证书

将 --self-signed 选项添加到 ipa-cacert-manage 续订。这会续订外部签名的 CA 证书作为自签名 CA 证书。

26.5. 允许 IDM 使用过期的证书启动

IdM 管理服务器证书过期后，大多数 IdM 服务都无法访问。您可以配置底层 Apache 和 LDAP 服务，以允许对服务的 SSL 访问，即使证书已过期。

如果您允许使用过期证书进行有限访问：

- Apache、Kerberos、DNS 和 LDAP 服务将继续工作。这些服务处于活动状态后，用户可以登录 IdM 域。
- 需要 SSL 访问的客户端服务仍将失败。例如，sudo 将失败，因为它需要在 IdM 客户端上需要 SSSD，SSSD 需要 SSL 来联系 IdM。

重要

此流程仅作临时解决方案。尽快续订所需的证书，然后恢复上述更改。

1. 配置 Apache 服务器的 mod_nss 模块，使其不强制执行有效的证书。

   a. 打开 /etc/httpd/conf.d/nss.conf 文件。

   b. 将 NSSEnforceValidCerts 参数设置为 off ：

   ```
   NSSEnforceValidCerts off
   ```

2. 重新启动 Apache.

   ```
   # systemctl restart httpd.service
   ```

3. 确保为 LDAP 目录服务器禁用了有效检查。要做到这一点，请验证 nsslapd-validate-cert 属性是否已设置为 warn ：

```
# ldapsearch -h server.example.com -p 389 -D "cn=directory manager" -w secret -LLL -b
cn=config -s base "(objectclass=*)" nsslapd-validate-cert

dn: cn=config
nsslapd-validate-cert: warn
```

如果属性没有设为 warn，请修改它：

```
# ldapmodify -D "cn=directory manager" -w secret -p 389 -h server.example.com

dn: cn=config
changetype: modify
replace: nsslapd-validate-cert
nsslapd-validate-cert: warn
```

4. 重新启动目录服务器。

```
# systemctl restart dirsrv.target
```

26.6. 为 HTTP 或 LDAP 安装第三方证书

为 Apache Web 服务器和 Directory 服务器安装新的 SSL 服务器证书，或两者均将当前 SSL 证书替换为新的 SSL 证书。要做到这一点，您需要：

- 您的私有 SSL 密钥（以下流程中的ssl.key ）
- 您的 SSL 证书（以下流程中的ssl.crt ）

有关接受密钥和证书的格式列表，请查看 ipa-server-certinstall(1) man page。

先决条件

ssl.crt 证书必须由您要载入证书的服务已知的 CA 签名。如果没有这种情况，请将签名 ssl.crt 的 CA 的 CA 证书安装到 IdM 中，如 第 26.3 节 “手动安装 CA 证书” 所述。

这样可确保 IdM 识别 CA，因此接受 ssl.crt。

安装第三方证书

1. 使用 ipa-server-certinstall 工具安装证书。指定您要安装的位置：

   - --HTTP 在 Apache Web 服务器中安装证书
   - --dirsrv 在目录服务器上安装证书

   例如，要将 SSL 证书安装到两者中：

```
# ipa-server-certinstall --http --dirsrv ssl.key ssl.crt
```

2. 重新启动您安装证书的服务器。

   - 重启 Apache Web 服务器：

```
# systemctl restart httpd.service
```

   - 重启目录服务器：

```
# systemctl restart dirsrv@REALM.service
```

3. 要验证证书是否已正确安装，请确保它存在于证书数据库中。

   - 显示 Apache 证书数据库：

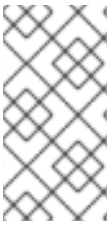
```
# certutil -L -d /etc/httpd/alias
```

   - 显示 Directory 服务器证书数据库：

```
# certutil -L -d /etc/dirsrv/slapd-REALM/
```

26.7. 配置 OCSP 回复

与 IdM 服务器集成的每个 CA 使用内部在线证书状态协议(OCSP)响应程序。允许访问 OCSP 响应器的 IdM 服务位于 http://ca-server.example.com/ca/ocsp 中。客户端可以连接到此 URL 以检查证书的有效性。



注意

有关 OCSP 的详情，请查看 Red Hat 证书系统文档。例如：2.2.4。撤销 计划、安装和部署指南中的 证书和检查状态。

26.7.1. 更改 CRL 更新间隔

默认情况下，IdM CA 会每四个小时自动生成 CRL 文件。更改这个间隔：

1. 停止 CA 服务器。

```
# systemctl stop pki-tomcatd@pki-tomcat.service
```

2. 打开 /var/lib/pki/pki-tomcat/conf/ca/CS.cfg 文件，并将 ca.crl.MasterCRL.autoUpdateInterval 值改为新的 interval 设置。例如，要每 60 分钟生成 CRL：

```
ca.crl.MasterCRL.autoUpdateInterval=60
```

注意

如果您更新 ca.crl.MasterCRL.autoUpdateInterval 参数，则更改将在下一次调度的 CRL 更新后生效。

3. 启动 CA 服务器。

```
# systemctl start pki-tomcatd@pki-tomcat.service
```

26.8. 在现有 IDM 域中安装 CA

如果在没有证书颁发机构(CA)的情况下安装了 IdM 域，您可以随后安装 CA 服务。根据您的环境，您可以安装 IdM 证书服务器 CA 或使用外部 CA。

注意

有关支持的 CA 配置的详情，请参考 第 2.3.2 节 “确定要使用的 CA 配置”。

安装 IdM 证书服务器

1. 使用以下命令安装 IdM 证书服务器 CA：

```
[root@ipa-server ~] ipa-ca-install
```

2. 在所有服务器和客户端上运行 ipa-certupdate 工具，以使用 LDAP 中的新证书的信息来更新它们。您必须为每个服务器和客户端单独运行 ipa-certupdate。

重要

手动安装证书后始终运行 ipa-certupdate。如果不这样做，则证书不会分发到其他计算机上。

安装外部 CA

外部 CA 的后续安装由多个步骤组成：

1. 启动安装：

```
[root@ipa-server ~] ipa-ca-install --external-ca
```

在这一步后，显示保存了证书签名请求(CSR)。将 CSR 提交给外部 CA，并将发布的证书复制到 IdM 服务器。

2. 继续安装，将证书和到外部 CA 文件的完整路径传递给 ipa-ca-install ：

-

```
[root@ipa-server ~]# ipa-ca-install --external-cert-file=/root/master.crt --external-cert-file=/root/ca.crt
```

3. 在所有服务器和客户端上运行 ipa-certupdate 工具，以使用 LDAP 中的新证书的信息来更新它们。您必须为每个服务器和客户端单独运行 ipa-certupdate。

重要

手动安装证书后始终运行 ipa-certupdate。如果不这样做，则证书不会分发到其他计算机上。

CA 安装不会将 LDAP 和 Web 服务器的现有服务证书替换为由新安装的 CA 发布的证书。有关如何替换证书的详情请参考 第 26.9 节 “替换 Web 服务器和 LDAP 服务器的证书”。

26.9. 替换 WEB 服务器和 LDAP 服务器的证书

替换 web 服务器和 LDAP 服务器的服务证书：

1. 请求新证书。您可以使用以下命令完成此操作：

   - 集成的 CA：详情请参阅 第 24.1.1 节 “为用户、主机或服务请求新证书”。
   - 外部 CA：生成私钥和证书签名请求(CSR)。例如，使用 OpenSSL：

```
$ openssl req -new -newkey rsa:2048 -days 365 -nodes -keyout new.key -out new.csr -subj '/CN=idmserver.idm.example.com,O=IDM.EXAMPLE.COM'
```

     将 CSR 提交到外部 CA。这个过程根据要用作外部 CA 的服务的不同而有所不同。

2. 替换 Apache Web 服务器的私钥和证书：

```
[root@ipaserver ~]# ipa-server-certinstall -w --pin=password new.key new.crt
```

3. ***替换 LDAP 服务器的私钥和证书：***

```
[root@ipaserver ~]# ipa-server-certinstall -d --pin=password new.key new.cert
```

第 27 章 IDM 中的 KERBEROS PKINIT 身份验证

Kerberos(PKINIT)中用于初始身份验证的公钥加密是 Kerberos 的预验证机制。从 Red Hat Enterprise Linux 7.4 开始，身份管理(IdM)服务器包括 Kerberos PKINIT 身份验证的机制。以下小节概述了 IdM 中的 PKINIT 实施，并描述了如何在 IdM 中配置 PKINIT。

27.1. 不同 IDM 版本中的默认 PKINIT 状态

IdM 服务器上的默认 PKINIT 配置取决于 Red Hat Enterprise Linux (RHEL)和证书颁发机构(CA)配置中 IdM 的版本。请参阅 表 27.1 “IdM 版本中的默认 PKINIT 配置”。

表 27.1. IdM 版本中的默认 PKINIT 配置

RHEL 版本	CA 配置	PKINIT 配置
7.3 及更早版本	没有 CA	本地 PKINIT：IdM 仅将 PKINIT 用于服务器上的内部目的。
7.3 及更早版本	带有集成 CA	IdM 会尝试使用集成 IdM CA 签名的证书来配置 PKINIT。 如果尝试失败，则 IdM 仅配置本地 PKINIT。
7.4 及更新的版本	没有 CA 没有提供给 IdM 的外部 PKINIT 证书	本地 PKINIT：IdM 仅将 PKINIT 用于服务器上的内部目的。
7.4 及更新的版本	没有 CA 为 IdM 提供的外部 PKINIT 证书	IdM 使用外部 Kerberos 密钥分发中心(KDC)证书和 CA 证书来配置 PKINIT。
7.4 及更新的版本	带有集成 CA	IdM 使用 IdM CA 签名的证书来配置 PKINIT。

在域级别 0 上，PKINIT 被禁用。默认行为是本地 PKINIT：IdM 仅将 PKINIT 用于服务器上的内部目的。另请参阅 第 7 章 显示和提升域级别。

27.2. 显示当前 PKINIT 配置

IdM 提供多个命令，可用于查询域中的 PKINIT 配置。

要确定域中的 PKINIT 状态，请使用 ipa pkinit-status 命令：

```
$ ipa pkinit-status
  Server name: server1.example.com
  PKINIT status: enabled
  [...output truncated...]
  Server name: server2.example.com
  PKINIT status: disabled
  [...output truncated...]
```

要在登录的服务器上确定 PKINIT 状态，请使用 ipa-pkinit-manage status 命令：

```
# ipa-pkinit-manage status
PKINIT is enabled
The ipa-pkinit-manage command was successful
```

命令显示 PKINIT 配置状态为 启用或禁用 ：

- **启用 ：PKINIT 使用由集成的 IdM CA 或外部 PKINIT 证书签名的证书进行配置。另请参阅 第 27.1 节 “不同 IdM 版本中的默认 PKINIT 状态”。**
- **disabled:IdM 仅将 PKINIT 用于 IdM 服务器上的内部目的。**

要显示支持 IdM 客户端的 PKINIT 的活跃 Kerberos 密钥分发中心(KDC)的 IdM 服务器，请在任何服务器上使用 ipa config-show 命令：

```
$ ipa config-show
  Maximum username length: 32
  Home directory base: /home
  Default shell: /bin/sh
  Default users group: ipausers
  [...output truncated...]
  IPA masters capable of PKINIT: server1.example.com
  [...output truncated...]
```

其它资源

- **有关报告 PKINIT 状态的命令行工具的更多详细信息，请使用 ipa help pkinit 命令。**

27.3. 在 IDM 中配置 PKINIT

如果您的 IdM 服务器在禁用 PKINIT 的情况下运行，请使用以下步骤启用它。例如，如果您使用 ipa-server-install 或 ipa-replica-install 工具传递 --no-pkinit 选项，则禁用 PKINIT 的服务器。

先决条件

- 确保所有安装了证书颁发机构(CA)的 IdM 服务器都在同一域级别运行。详情请查看 第 7 章 显示和提升域级别。

步骤

1. 如果您在没有 CA 的情况下使用 IdM，请使用 ipa-server-certinstall 工具安装外部 Kerberos 密钥分发中心(KDC)证书。KDC 证书必须满足以下条件：

   - 它使用通用名称 CN=fully_qualified_domain_name,certificate_subject_base 发布。
   - 它包括 Kerberos 主体 krbtgt/REALM_NAME@REALM_NAME。
   - 它包含用于 KDC 验证的对象标识符(OID)：1.3.6.1.5.2.3.5.

   ```
   # ipa-server-certinstall --kdc kdc.pem
   # systemctl restart krb5kdc.service
   ```

   详情请查看 ipa-server-certinstall(1) man page。

2. 启用 PKINIT:

   ```
   $ ipa-pkinit-manage enable
   Configuring Kerberos KDC (krb5kdc)
     [1/1]: installing X509 Certificate for PKINIT
   Done configuring Kerberos KDC (krb5kdc).
   The ipa-pkinit-manage command was successful
   ```

   如果您使用 IdM CA，命令会从 CA 请求 PKINIT KDC 证书。

3. 要验证新的 PKINIT 状态，请参阅 第 27.2 节 “显示当前 PKINIT 配置”。

27.4. 其它资源

- *有关 Kerberos PKINIT 的详细信息，MIT Kerberos 文档中的 PKINIT 配置.*

- *有关在 IdM 中配置 PKINIT 智能卡验证的文档，请参考 第 23.5 节 “身份管理中的 PKINIT 智能卡身份验证”。*

部分 VI. 管理：管理策略

这部分提供了如何定义密码策略、管理 Kerberos 域、使用 sudo 工具、如何配置基于主机的访问控制和定义 SELinux 用户映射的说明。

第 28 章 定义密码策略

本章论述了身份管理(IdM)中的密码策略是什么以及如何管理它们。

28.1. 什么是密码策略以及为什么它们有用

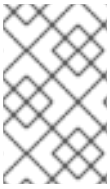
密码策略是密码 必须满足的一组规则。

例如，密码策略可以定义最短密码长度和最长密码期限。受此类策略影响的所有用户都必须设置足够长的密码并足够频繁地进行更改。

密码策略有助于降低发现和滥用用户密码的风险。

28.2. 在 IDM 中密码策略如何工作

所有用户都必须具有用于向 Identity Management(IdM)Kerberos 域进行身份验证的密码。IdM 中的密码策略定义这些用户密码必须满足的要求。



注意

IdM 密码策略在底层 LDAP 目录中设置，但也由 Kerberos 密钥分发中心(KDC)执行。

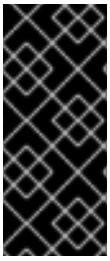
28.2.1. 支持的密码策略属性

表 28.1 “密码策略属性” 列出 IdM 中密码策略可以定义的属性。

表 28.1. 密码策略属性

属性	介绍	示例
Max lifetime	密码在用户必须重置之前有效的最长时间（以天数为单位）。	最大生命周期 = 90 用户密码仅有效 90 天。之后，IdM 会提示用户更改它们。
Min lifetime	必须在两个密码更改操作之间传递的最小时间（以小时为单位）。	Min Life = 1 用户更改密码后，他们必须至少等待 1 小时后再重新更改密码。

属性	介绍	示例
History size	先前存储的密码数量.用户无法重复利用其密码历史记录中的密码。	History size = 0 用户可以重复使用之前的任何密码。
Character classes	用户必须在密码中使用的不同字符类别的数量。字符类为： • 大写字符 • 小写字符 • 数字 • 特殊字符，如逗号(,)、句号(.)、星号(*) • 其他 UTF-8 字符 当一个字符连续使用三次或更多次时，会将该字符类减一。例如： • **Secret1** 有 3 个字符类：大写、小写、数字 • **Secret111** 有 2 个字符类：大写、小写、数字以及重复使用 **1** 的 -1 损失	字符类 = 0 需要的默认类数为 0。要配置数字，请使用 **--minclasses** 选项运行 **ipa pwpolicy-mod** 命令。该命令将所需的字符类数设置为 1： `$ ipa pwpolicy-mod --minclasses=1` 另请参阅下表中的 重要 备注。
Min length	密码中的最少字符数.	Min length = 8 用户不能使用少于 8 个字符的密码。
Max failures	IdM 锁定用户帐户前允许的失败登录的最多次数。另请参阅 第 22.1.3 节 “密码失败后解锁用户帐户”。	Max failures = 6 IdM 会锁定用户所在行中输入错误密码 7 次的用户帐户。
Failure reset interval	IdM 重置当前失败登录尝试次数的时间（以秒为单位）。	Failure reset interval = 60 如果用户在 **Max failures** 定义的登录尝试失败的次数超过 1 分钟，用户可以尝试再次登录，而不会造成用户帐户锁定的风险。
锁定持续时间	在 **Max failures** 中定义的失败登录尝试次数后，用户帐户锁定的时间（以秒为单位）。另请参阅 第 22.1.3 节 “密码失败后解锁用户帐户”。	Lockout duration = 600 锁定帐户的用户在 10 分钟内无法登录。



重要

如果您一组不同的硬件可能不能使用国际字符和符号，则字符类要求应为英语字母和常用符号。有关密码中字符类策略的更多信息，请参阅红帽知识库 中的哪些字符在密码中有效？

28.2.2. 全局和特定于组的密码策略

默认密码策略是 全局密码策略。除了全局策略外，您还可以创建其他 组密码策略。

全局密码策略

安装初始 IdM 服务器会自动使用默认设置创建全局密码策略。

全局策略规则应用到所有用户，无组密码策略。

组密码策略

组密码策略应用到对应用户组的所有成员。

对于任何用户，一次只能有一个密码策略生效。如果用户分配了多个密码策略，其中一个将优先于优先级。请参阅 第 28.2.3 节 “密码策略优先级”。

28.2.3. 密码策略优先级

每个组密码策略都有一个 优先级 集。值越低，策略优先级越高。支持最低的优先级值为 0。

- 如果多个密码策略适用于某个用户，则优先级最低的策略优先。其他策略中定义的所有规则都将被忽略。
- 优先级值最低的密码策略适用于所有密码策略属性，即使策略中没有定义的属性也是如此。

全局密码策略没有设置优先级值。当没有为用户设置组策略时，它充当回退策略。全局策略不能优先于组策略。

表 28.2 “根据优先级应用密码策略属性示例” 演示了密码策略优先级在属于两个组并定义了策略的用户示例上工作。

表 28.2. 根据优先级应用密码策略属性示例

	Max lifetime	Min length
组 A 的策略（优先级 0）	60	10
组 B 的策略（优先级 1）	90	0（无限制）
	↓	↓
用户（组 A 和组 B 的成员）	60	10

注意

ipa pwpolicy-show --user=user_name 命令显示哪个策略当前对特定用户生效。

28.3. 添加新密码策略

添加新密码策略时，您必须指定：

- **策略要应用到的用户组（请参阅 第 28.2.2 节 “全局和特定于组的密码策略”）**
- **优先级（请参阅 第 28.2.3 节 “密码策略优先级”）**

使用以下方法添加新密码策略：

- **Web UI，请参见 “Web UI：添加新密码策略”一节**
- **命令行，请查看 “命令行：添加新密码策略”一节**

Web UI：添加新密码策略

1. **选择 Policy → Password Policies。**
2. **点击 Add。**

3. 定义用户组和优先级。

4. 单击 **Add** 确认。

要配置新密码策略的属性，请参阅 第 28.4 节 “修改密码策略属性”。

命令行：添加新密码策略

1. 使用 **ipa pwpolicy-add** 命令。指定用户组群和优先级：

```
$ ipa pwpolicy-add
Group: group_name
Priority: priority_level
```

2. 可选。使用 **ipa pwpolicy-find** 命令来验证策略是否已成功添加：

```
$ ipa pwpolicy-find
```

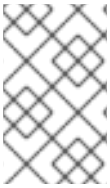
要配置新密码策略的属性，请参阅 第 28.4 节 “修改密码策略属性”。

28.4. 修改密码策略属性

重要

当您修改密码策略时，新规则仅适用于新密码。这些更改不会追溯应用到现有密码。

要使更改生效，用户必须更改其现有密码，或者管理员必须重置其他用户的密码。请参阅 第 22.1.1 节 “更改和重置用户密码”。



注意

有关安全 用户密码的建议，请参阅安全指南 中的密码安全性。

要修改密码策略，请使用以下命令：

- Web UI，请参见 “Web UI：修改密码策略”一节
- 命令行，请查看 “命令行：修改密码策略”一节

请注意，将密码策略属性设置为 0 表示没有属性限制。例如，如果您将最长生命周期设定为 0，则用户密码永远不会过期。

Web UI：修改密码策略

1. 选择 Policy → Password Policies。
2. 点您要更改的策略。
3. 更新所需的属性。有关可用属性的详情请参考 第 28.2.1 节 “支持的密码策略属性”。
4. 单击 Save 以确认更改。

命令行：修改密码策略

1. 使用 ipa pwpolicy-mod 命令更改策略的属性。

   a. 例如，要更新全局密码策略，并将最小密码长度设置为 10 ：

   ```
   $ ipa pwpolicy-mod --minlength=10
   ```

   b. 要更新组策略，请将组名称添加到 ipa pwpolicy-mod。例如：

   ```
   $ ipa pwpolicy-mod group_name --minlength=10
   ```

2. 可选。使用 ipa pwpolicy-show 命令显示新的策略设置。

a. 显示全局策略：

```
$ ipa pwpolicy-show
```

b. 要显示组策略，请将组名称添加到 ipa pwpolicy-show ：

```
$ ipa pwpolicy-show group_name
```

28.5. 使用立即生效更改密码过期日期

当现有密码更改或用户输入新密码时，IdM 将应用密码策略规则。请参阅 第 28.4 节 “修改密码策略属性”。

要强制立即更改用户密码的过期日期，请重置 LDAP 中的 krbPasswordExpiration 属性值。例如，对于单个用户：

1. 使用 ldapmodify 工具：

```
# ldapmodify -D "cn=Directory Manager" -w secret -h server.example.com -p 389 -vv

dn: uid=user_name,cn=users,cn=accounts,dc=example,dc=com
changetype: modify
replace: krbPasswordExpiration
krbPasswordExpiration: 20160203203734Z
```

krbPasswordExpiration 格式遵循此模板：

- year (2016)
- month (02)
- 第一天(03)
- 当前的时间（小时、分钟和秒）(20:37:34)

- *时区(Z)*

2. *按 Ctrl+D 确认并将更改发送到服务器。*

要一次编辑多个条目，请使用 -f 选项和 ldapmodify 来引用 LDIF 文件。

第 29 章 管理 KERBEROS 域

本章论述了管理身份管理服务器的 Kerberos 密钥分发中心(KDC)组件。

重要

不要使用 kadmin 或 kadmin.local 工具来管理身份管理 Kerberos 策略。按照本指南所述，使用原生身份管理命令行工具。

如果您尝试使用上述 Kerberos 工具管理身份管理策略，其中一些操作不会影响其 Directory 服务器实例中存储的身份管理配置。

29.1. 管理 KERBEROS 票据策略

身份管理中的 Kerberos ticket 策略设置对票据持续时间和续订的限制。使用以下步骤，您可以为在身份管理服务器上运行的 Kerberos 密钥分发中心(KDC)配置 Kerberos ticket 策略。

29.1.1. 确定 Kerberos Ticket 的生命周期

当身份管理服务器确定身份管理客户端代表 user_name 请求 Kerberos 票据后授予的票据生命周期时，会考虑几个参数。首先，客户端评估发生，它根据 /etc/krb5.conf 文件中的 kinit 命令和 ticket_lifetime 设置计算要请求的值。然后，该值将发送到进行服务器端评估的身份管理服务器。如果请求的生命周期低于全局设置允许的范围，则会授予请求的生命周期。否则，授予的生命周期是全局设置允许的值。

客户端代表 user_name 请求的生命周期，如下所示：

在客户端

- 如果您使用 -l 选项在 kinit 命令本身中显式状态 user_name 的值，例如：

```
$ kinit user_name -l 90000
```

此例中为 90000 秒，客户端代表 user_name 请求该值。

- 否则，如果没有将 lifetime 值作为 kinit user_name命令的参数 传递，客户端 /etc/krb5.conf 文件中的 ticket_lifetime 设置的值供客户端代表 user_name 使用。如果在

/etc/krb5.conf 文件中没有指定值，则使用初始票据请求的默认 IdM 值，即 1 天。

在服务器端

服务器端进行一个双阶段评估：

1. 如果存在这些策略，客户端请求的值将与 user_name- 特定 Kerberos ticket 策略的 --maxlife 设置进行比较，并选择这两个策略的较低值。如果 user_name- 特定的 Kerberos ticket 策略不存在，客户端发送的值将与全局 Kerberos ticket 策略的 --maxlife 设置进行比较，并选择两者的较低值。有关全局和用户特定 Kerberos ticket 策略的详情，请参考 第 29.1.2 节 “全局和特定于用户的 Kerberos 票据策略”。

2. 上一步中选择的值与两个其他值进行比较：

   - /var/kerberos/krb5kdc/kdc.conf 文件中的 max_life 设置的值
   - LDAP 条目的 krbMaxTicketLife 属性中设置的值带有可分辨名称(DN): krbPrincipalName=krbtgt/REALM_NAME@ REALM_NAME@REALM_NAME,cn=REALM_NAME,cn=kerberos,domain_name

   在授予 user_name 的 Kerberos ticket 的生命周期内，最终选择了三个值中最低的值。

29.1.2. 全局和特定于用户的 Kerberos 票据策略

您可以重新定义全局 Kerberos ticket 策略，并针对个人用户定义其他策略。

全球 Kerberos ticket 策略

全局策略适用于身份管理 Kerberos 域内发布的所有票据。

用户特定的 Kerberos ticket 策略

特定于用户的策略仅应用到关联的用户帐户。例如，特定于用户的 Kerberos 票据策略可为 admin 用户定义较长的最大票据生命周期。

特定于用户的策略优先于全局策略。

29.1.3. 配置全局 Kerberos 票据策略

要配置全局 Kerberos ticket 策略，您可以使用：

- Identity Management web UI: see “Web UI：配置全局 Kerberos 票据策略”一节
- 命令行：请查看 “命令行：配置全局 Kerberos 票据策略”一节

表 29.1. 支持的 Kerberos Ticket 策略属性

属性	介绍	示例
最大续订数	用户可在 Kerberos 票据到期后续订 Kerberos 票据的时间（以秒为单位）。续订期后，用户必须使用 **kinit** 工具登录才能获取新的票据。 要续订票据，请使用 **kinit -R** 命令。	最大续订 = 604800 票据到期后，用户可以在接下来的 7 天（604,800 秒）内续订。
最大生命周期	Kerberos 票据的生命周期（以秒为单位）。Kerberos ticket 处于活动状态的时间段。	最大生命周期 = 86400 票据在签发后 24 小时（86,400 秒）过期。

Web UI：配置全局 Kerberos 票据策略

1. 选择 **Policy** → **Kerberos Ticket Policy**。
2. 定义所需的值：
   a. 在 **Max renew** 字段中，输入 Kerberos 票据的最大续订周期。
   b. 在 **Max Life** 字段中，输入 Kerberos 票据的最大生命周期。

图 29.1. 配置全局 Kerberos 票据策略

3. *点击 Save。*

命令行：配置全局 Kerberos 票据策略

修改全局 Kerberos ticket 策略：

- *使用 ipa krbtpolicy-mod 命令，并至少传递以下选项之一：*
  - *--maxrenew 定义 Kerberos 票据的最大续订周期*
  - *--maxlife 定义 Kerberos ticket 的最长生命周期*

例如，要更改最长生命周期：

```
$ ipa krbtpolicy-mod --maxlife=80000
Max life: 80000
Max renew: 604800
```

将全局 Kerberos ticket 策略重置为原始默认值：

1. *使用 ipa krbtpolicy-reset 命令。*

2. 可选。使用 ipa krbtpolicy-show 命令验证当前的设置。

有关 ipa krbtpolicy-mod 和 ipa krbtpolicy-reset 的详细信息，请使用它们传递 --help 选项。

29.1.4. 配置用户特定的 Kerberos 票据策略

要修改特定用户的 Kerberos ticket 策略：

1. 使用 ipa krbtpolicy-mod user_name 命令，并至少传递以下选项之一：

   - --maxrenew 定义 Kerberos 票据的最大续订周期
   - --maxlife 定义 Kerberos ticket 的最长生命周期

   如果您仅定义其中一个属性，身份管理将对另一个属性应用全局 Kerberos ticket 策略值。

   例如，要更改 admin 用户的最大生命周期：

   ```
   $ ipa krbtpolicy-mod admin --maxlife=160000
   Max life: 80000
   Max renew: 604800
   ```

2. 可选。使用 ipa krbtpolicy-show user_name 命令显示指定用户的当前值。

新策略会立即对用户请求的下一个 Kerberos 票据（如使用 kinit 工具时）生效。

要重置用户特定的 Kerberos 票据策略，请使用 ipa krbtpolicy-reset user_name 命令。命令清除特定于用户的值，之后身份管理应用全局策略值。

有关 ipa krbtpolicy-mod 和 ipa krbtpolicy-reset 的详细信息，请使用它们传递 --help 选项。

29.2. 重新打包 KERBEROS 主体

重新打包 Kerberos 主体会将带有更高密钥版本号(KVNO)的新 keytab 条目添加到主体的 keytab。原始条目保留在 keytab 中，但不再用于发出票据。

1. 查找所需时间段内发布的所有 keytab。例如，以下命令使用 ldapsearch 工具显示 2016 年 1 月 1 日 1 日和 11:59 PM 在 2016 年 12 月 31 日创建的所有主机和服务主体：

```
# ldapsearch -x -b "cn=computers,cn=accounts,dc=example,dc=com" "(&
(krblastpwdchange>=20160101000000)(krblastpwdchange<=20161231235959))" dn
krbprincipalname
```

```
# ldapsearch -x -b "cn=services,cn=accounts,dc=example,dc=com" "(&
(krblastpwdchange>=20160101000000)(krblastpwdchange<=20161231235959))" dn
krbprincipalname
```

   - searchbase (-b)定义 ldapsearch 查找主体的子树：
     - 主机主体存储在 cn=computers,cn=accounts,dc=example,dc=com 子树下。
     - 服务主体存储在 cn=services,cn=accounts,dc=example,dc=com 子树下。
   - krblastpwdchange 参数按上次更改日期过滤搜索结果。参数接受 GMT 中日期和 HHMMSS 格式的 YYYYMMDD 格式。
   - 指定 dn 和 krbprincipalname 属性将搜索结果限制为条目名称和主体。

2. 对于需要重新密钥主体的每个服务和主机，请使用 ipa-getkeytab 工具来检索新的 keytab 条目。传递以下选项：

   - 用于指定主体的 --principal (-p)

     用于指定原始 keytab 位置的 --key tab(-k)

用于指定身份管理服务器主机名的 **--server (-s)**

例如：

- 在 **/etc/krb5.keytab** 的默认位置中使用 **keytab** 重新密钥主机主体：

```
# ipa-getkeytab -p host/client.example.com@EXAMPLE.COM -s server.example.com -k
/etc/krb5.keytab
```

- 在 **/etc/httpd/conf/ipa.keytab** 的默认位置重新密钥 **Apache** 服务的 **keytab**：

```
# ipa-getkeytab -p HTTP/client.example.com@EXAMPLE.COM -s server.example.com -
k /etc/httpd/conf/ipa.keytab
```

重要

某些服务（如 **NFS** 版本 **4**）只支持一组有限的加密类型。将适当的参数传递给 **ipa-getkeytab** 命令，以正确配置 **keytab**。

3. 可选。验证您是否成功更新了主体。使用 **klist** 实用程序列出所有 **Kerberos** 票据。例如，要列出 **/etc/krb5.keytab** 中的所有 **keytab** 条目：

```
# klist -kt /etc/krb5.keytab
Keytab: WRFILE:/etc/krb5.keytab
KVNO Timestamp         Principal
---- ----------------- --------------------------------------------------------
   1 06/09/16 05:58:47 host/client.example.com@EXAMPLE.COM(aes256-cts-hmac-sha1-
96)
   2 06/09/16 11:23:01 host/client.example.com@EXAMPLE.COM(aes256-cts-hmac-sha1-
96)
   1 03/09/16 13:57:16 krbtgt/EXAMPLE.COM@EXAMPLE.COM(aes256-cts-hmac-sha1-96)
   1 03/09/16 13:57:16 HTTP/server.example.com@EXAMPLE.COM(aes256-cts-hmac-sha1-
96)
   1 03/09/16 13:57:16 ldap/server.example.com@EXAMPLE.COM(aes256-cts-hmac-sha1-
96)
```

输出显示 **client.example.com** 的 **keytab** 条目使用更高的 **KVNO** 重新密钥。原始 **keytab** 仍存在于数据库中，与之前的 **KVNO** 一起。

针对之前的 keytab 发出的票据继续起作用，而使用 KVNO 最高的密钥签发新的票据。这可以避免对系统操作造成任何干扰。

29.3. 保护 KEYTABS

要防止 Kerberos keytab 被具有服务器访问权限的其他用户影响，请将对 keytab 的访问权限限制为 keytab 所有者。建议在检索后立即对 keytab 进行保护。

例如，要保护位于 /etc/httpd/conf/ipa.keytab 的 Apache keytab：

1. 将文件的所有者设置为 apache。

```
# chown apache /etc/httpd/conf/ipa.keytab
```

2. 将文件的权限设置为 0600。这会将读取、写入和执行权限授予给所有者。

```
# chmod 0600 /etc/httpd/conf/ipa.keytab
```

29.4. 删除 KEYTAB

删除 keytab 并创建新的 keytab 是必需的，例如，当您取消滚动和重新注册主机时，或者遇到 Kerberos 连接错误时。

要删除主机上的所有 keytab，请使用 ipa-rmkeytab 工具，并传递这些选项：

- 用于指定 Kerberos 域的 --realm (-r)
- 用于指定 keytab 文件的路径的 --key tab(-k)

```
# ipa-rmkeytab --realm EXAMPLE.COM --keytab /etc/krb5.keytab
```

要删除特定服务的 keytab，请使用 --principal (-p)选项来指定服务主体：

```
# ipa-rmkeytab --principal ldap/client.example.com --keytab /etc/krb5.keytab
```

29.5. 其它资源

- 有关身份管理服务器托管的 Kerberos KDC 的概述，请查看 第 1.2.1.1 节 “IdM 服务器托管的服务”。

- 有关 Kerberos 的 Red Hat 文档，请参阅 系统级身份验证指南中的 使用 Kerberos。

- 有关 Kerberos 概念的更多信息，请参阅 MIT Kerberos 文档。

第 30 章 使用 SUDO

身份管理提供了一种机制，可在 IdM 域中可预测且一致地应用 sudo 策略。IdM 域中的每个系统都可以配置为 sudo 客户端。

30.1. 身份管理中的 SUDO 工具

sudo 实用程序提供对指定用户的管理访问权限。当信任用户之前使用 sudo 管理命令时，会提示他们自己的密码。然后，当它们经过身份验证并假定允许 命令时，将像 root 用户一样执行管理命令。有关 sudo 的更多信息，请参阅 系统管理员指南。

30.1.1. sudo的身份管理 LDAP 架构

IdM 具有 sudo 条目的专用 LDAP 模式。架构支持：

- 主机组和网络组.请注意，sudo 只支持 netgroups。
- sudo 命令组，包含多个命令。

注意

由于 sudo 不支持主机组或命令组，因此 IdM 会在创建 sudo 规则时将 IdM sudo 配置转换为原生 sudo 配置。例如，IdM 为每个主机组创建对应的 shadow netgroup，它允许 IdM 管理员创建引用主机组的 sudo 规则，而本地 sudo 命令则使用对应的 netgroup。

默认情况下，sudo 信息无法通过 LDAP 匿名使用。因此，IdM 在 uid=sudo,cn=sysaccounts,cn=etc,$SUFFIX 中定义了默认的 sudo 用户。您可以在位于 /etc/ sudo - ldap.conf 的 LDAP sudo 配置文件中更改此用户。

30.1.2. NIS 域名要求

必须为 netgroups 和 sudo 设置 NIS 域名才能正常工作。sudo 配置需要 NIS 格式的 netgroups 和 netgroups 的 NIS 域名。但是，IdM 并不要求 NIS 域实际存在。也不要求安装 NIS 服务器。

注意

ipa-client-install 工具默认自动将 NIS 域名设置为 IdM 域名。

30.2. 身份管理中的 SUDO 规则

使用 sudo 规则，您可以定义 谁可以 执行什么、位置 以及谁。

- 有权使用 sudo 的用户。
- 可用于 sudo 的命令 是什么。
- 其中 是允许用户使用 sudo 的目标主机。
- 用户假定 要 执行任务的系统或其他用户身份。

30.2.1. sudo 规则中的外部用户和主机

IdM 接受 sudo 规则中的外部实体。外部实体是存储在 IdM 域外部的实体，如不属于 IdM 域的用户或主机。

例如，您可以使用 sudo 规则为 IdM 中的 IT 组的成员授予 root 访问权限，其中 root 用户不是 IdM 域中定义的用户。或者，例如，管理员可以阻止对网络中某些主机的访问，但不属于 IdM 域。

30.2.2. sudo 规则的用户组支持

您可以使用 sudo 授予对 IdM 中整个用户组的访问权限。IdM 支持 Unix 和非 POSIX 组。请注意，创建非 POSIX 组可能会导致访问问题，因为非 POSIX 组中的任何用户都会从组中继承非 POSIX 权限。

30.2.3. 支持 sudoers 选项

IdM 支持 sudoers 选项。有关可用 sudoers 选项的完整列表，请查看 sudoers(5) man page。

请注意，IdM 不允许 sudoers 选项中的空格或换行符。因此，可以单独添加它们，而不是在逗号分隔的列表中提供多个选项。例如，要从命令行添加两个 sudoers 选项：

```
$ ipa sudorule-add-option sudo_rule_name
Sudo Option: first_option
$ ipa sudorule-add-option sudo_rule_name
Sudo Option: second_option
```

类似地，请确保在一行上提供长选项。例如，从命令行：

```
$ ipa sudorule-add-option sudo_rule_name
Sudo Option: env_keep="COLORS DISPLAY EDITOR HOSTNAME HISTSIZE INPUTRC KDEDIR
LESSSECURE LS_COLORS MAIL PATH PS1 PS2 XAUTHORITY"
```

30.3. 配置位置以查找 SUDO 策略

sudo 配置的集中 IdM 数据库使 IdM 中定义的 sudo 策略对所有域主机全局可用。在 Red Hat Enterprise Linux 7.1 系统中，ipa-server-install 和 ipa-client-install 工具会自动将系统配置为使用 IdM 定义的策略，方法是将 SSSD 设置为 sudo 的数据供应商。

查找 sudo 策略的位置在 /etc/nsswitch.conf 文件的 sudoers 行中定义。在运行 Red Hat Enterprise Linux 7.1 及更新版本的 IdM 系统中，nsswitch.conf 中的默认 sudoers 配置是：

```
sudoers: files sss
```

files 选项指定系统使用 /etc/sudoers 本地 SSSD 配置文件中定义的 sudo 配置。sss 选项指定使用 IdM 中定义的 sudo 配置。

30.3.1. 在 IdM 的 Earlier 版本中将主机配置为使用 IdM sudo 策略

要在运行早于 7.1 的 Red Hat Enterprise Linux 版本的 IdM 系统上实施 IdM 定义的 sudo 策略，请手动配置本地机器。您可以使用 SSSD 或 LDAP 执行此操作。红帽强烈建议使用基于 SSSD 的配置。

30.3.1.1. 使用 SSSD 将 sudo 策略应用到主机

在对 sudo 规则使用 SSSD 的每个系统中执行这些步骤：

1. 配置 sudo 以查找 sudoers 文件的 SSSD。

```
# vim /etc/nsswitch.conf

sudoers:  files sss
```

保留 files 选项可让 sudo 在检查 SSSD 以了解 IdM 配置前检查其本地配置。

2. 将 sudo 添加到本地 SSSD 客户端管理的服务列表中。

```
# vim /etc/sssd/sssd.conf

[sssd]
config_file_version = 2
services = nss, pam, sudo
domains = IPADOMAIN
```

3. 在 sudo 配置中为 NIS 域设置名称。sudo 使用 NIS 风格的网络组，因此 sudo 必须在系统配置中设置 NIS 域名才能查找 IdM sudo 配置中使用的主机组。

    1. 如果 rhel-domainname 服务尚未启用，以确保 NIS 域名在重启后将保留。

    ```
    # systemctl enable rhel-domainname.service
    ```

    2. 设置要与 sudo 规则一起使用的 NIS 域名。

    ```
    # nisdomainname example.com
    ```

    3. 将系统验证设置配置为永久保留 NIS 域名.例如：

    ```
    # echo "NISDOMAIN=example.com" >> /etc/sysconfig/network
    ```

    这会更新带有 NIS 域的 /etc/sysconfig/network 和 /etc/yp.conf 文件。

    4. 重启 rhel-domainname 服务：

```
# systemctl restart rhel-domainname.service
```

4. （可选）在 SSSD 中启用调试以显示它使用的 LDAP 设置。

```
[domain/IPADOMAIN]
debug_level = 6
....
```

SSSD 用于操作的 LDAP 搜索基础记录在 sssd_DOMAINNAME.log 日志中。

30.3.1.2. 使用 LDAP 将 sudo 策略应用到主机

重要

仅对不使用 SSSD 的客户端使用基于 LDAP 的配置。红帽建议使用基于 SSSD 的配置配置所有其他客户端，如 第 30.3.1.1 节 “使用 SSSD 将 sudo 策略应用到主机” 所述。

有关使用 LDAP 应用 sudo 策略的详情，请参考 Red Hat Enterprise Linux 6 Identity Management Guide 中的使用 LDAP 将 sudo 策略应用到主机。

基于 LDAP 的配置主要适用于基于 Red Hat Enterprise Linux 7 之前的版本的 Red Hat Enterprise Linux。因此，它只在 Red Hat Enterprise Linux 6 文档中被描述。

30.4. 添加 SUDO 命令、命令组和规则

30.4.1. 添加 sudo 命令

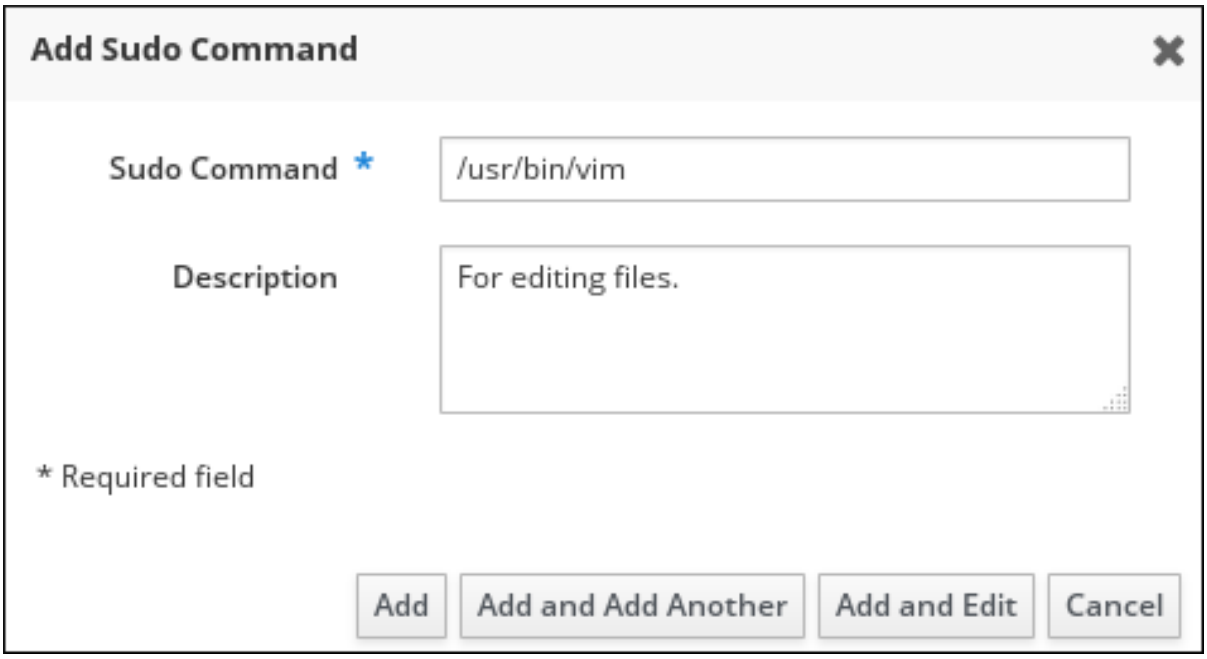
在 Web UI 中添加 sudo 命令

1. 在 Policy 选项卡下，单击 Sudo → Sudo Commands。

2. 单击列表顶部的 Add。

3. 填写关于 命令的信息。输入命令可执行文件的完整系统路径。

图 30.1. 添加新的 sudo 命令



4. *点击 Add。或者，单击 Add and Add Another 以开始添加另一个条目或 Add and Edit 以开始编辑新条目。*

从命令行添加 sudo 命令

要添加 sudo 命令，请使用 ipa sudocmd-add 命令。提供命令可执行文件的完整系统路径。例如，要添加 /usr/bin/less 命令和描述：

```
$ ipa sudocmd-add /usr/bin/less --desc="For reading log files"
----------------------------------
Added sudo command "/usr/bin/less"
----------------------------------
  sudo Command: /usr/bin/less
  Description: For reading log files
```

30.4.2. 添加 sudo 命令组

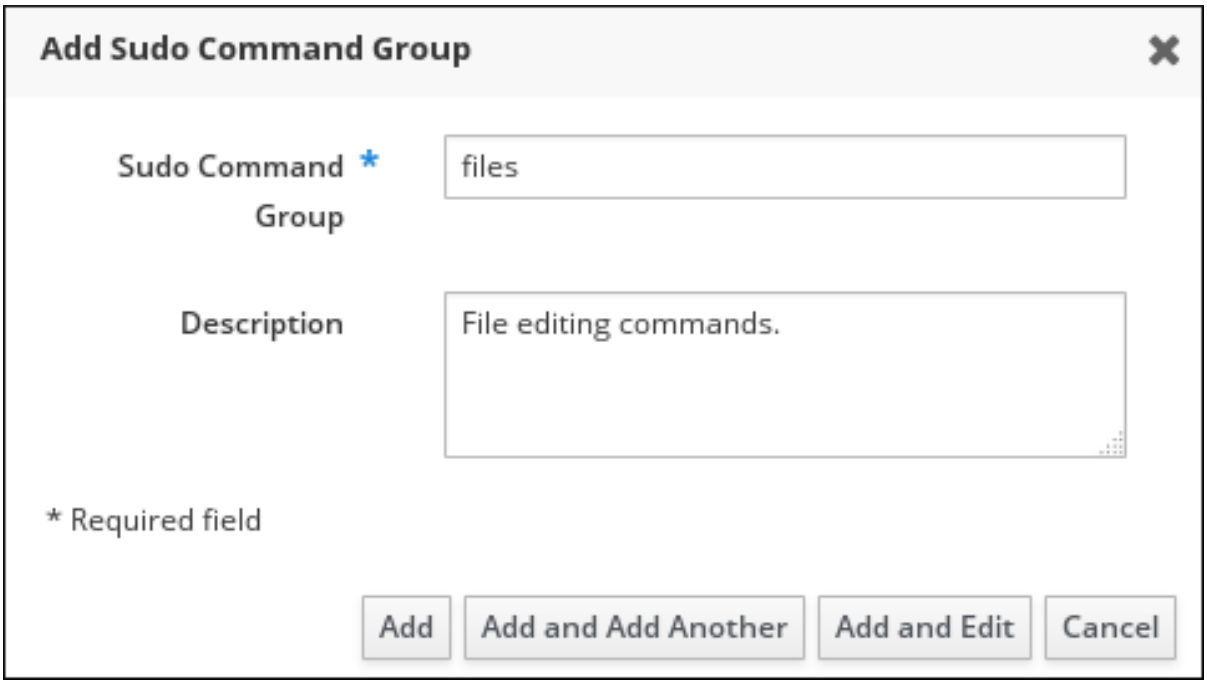
在 Web UI 中添加 sudo 命令组

1. *在 Policy 选项卡下，单击 Sudo → Sudo Command Groups。*

2. *单击列表顶部的 Add。*

3. *填写关于 命令组的信息。*

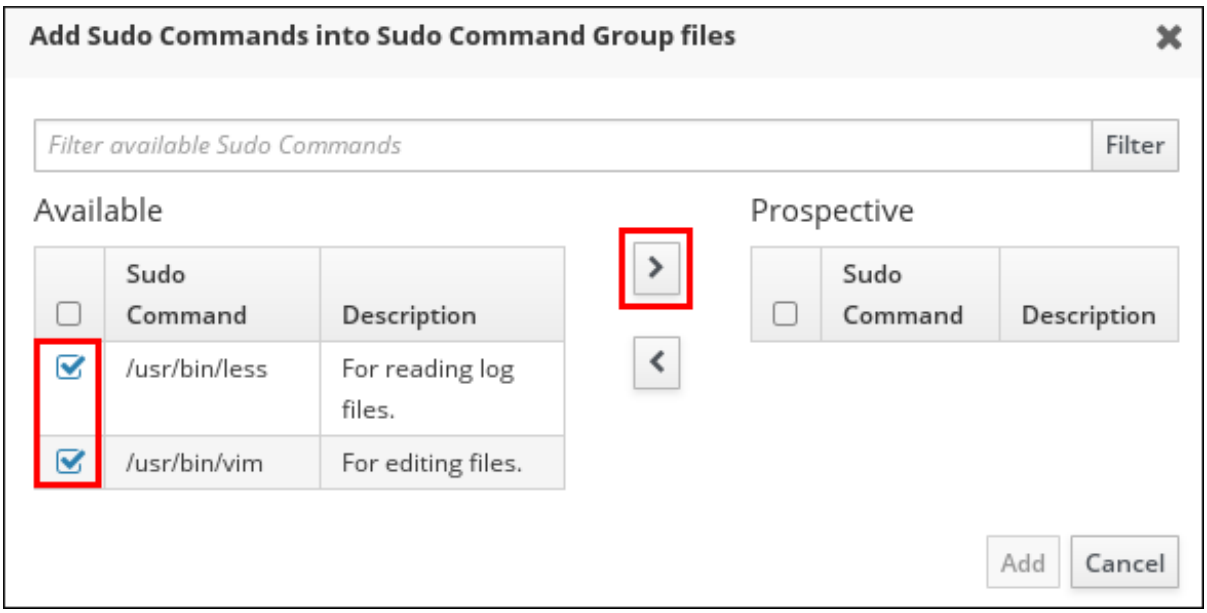
图 30.2. 添加新 sudo 命令组



4. *单击 **Add and Edit** 以开始编辑命令组。*

5. *在 **Sudo Commands** 选项卡下，单击 **Add** 将 sudo 命令添加到组中。选择所需命令并使用 **>** 按钮将它们移到 **Prospective** 列中。*

图 30.3. 在 sudo 命令组中添加命令



6. *点击 **Add**。*

从命令行添加 sudo 命令组

1. 使用 **ipa sudocmdgroup-add** 命令创建命令组。例如，要创建 **files** 命令组并添加其描述：

```
$ ipa sudocmdgroup-add files --desc="File editing commands"
----------------------------------
Added sudo command group "files"
----------------------------------
  sudo Command Group: files
  Description: File editing commands
```

2. 使用 **ipa sudocmdgroup-add-member** 命令在组中包括 **sudo** 命令。请注意，您只能包含已添加到 IdM 的命令，如 第 30.4.1 节 “添加 sudo 命令” 所述。

```
$ ipa sudocmdgroup-add-member files --sudocmds "/usr/bin/vim"
  sudo Command Group: files
  Description: File editing commands
  Member sudo commands: /usr/bin/vim
-------------------------
Number of members added 1
-------------------------
```

30.4.3. 添加 sudo 规则

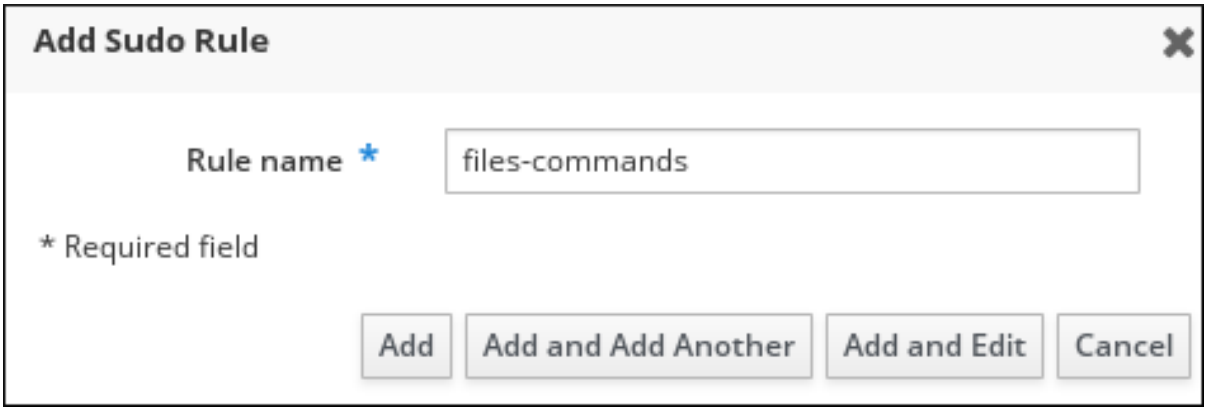
在 Web UI 中添加 sudo 规则

1. 在 **Policy** 选项卡下，单击 **Sudo** → **Sudo Rules**。

2. 单击列表顶部的 **Add**。

3. 输入规则的名称。

图 30.4. 命名一个新的 sudo 规则



4.

点击 Add。或者，单击 Add and Add Another 以开始添加另一个条目或 Add and Edit 以开始编辑新条目。

有关如何编辑新的 sudo 规则的详情，请参考 第 30.6 节 “修改 sudo 规则”。

从命令行添加 sudo 规则

要添加新的 sudo 规则，请使用 ipa sudorule-add 命令。例如，添加名为 files-commands 的规则：

```
$ ipa sudorule-add files-commands
-------------------------------
Added Sudo Rule "files-commands"
-------------------------------
  Rule name: files-commands
  Enabled: TRUE
```

有关使用 ipa sudorule-add 及其接受的选项的更多信息，请使用 --help 选项运行命令。

有关如何编辑新的 sudo 规则的详情，请参考 第 30.6 节 “修改 sudo 规则”。

有关添加新的 sudo 规则并从命令行编辑它的完整示例，请参考 例 30.1 “从命令行添加和修改新的 sudo 规则”。

30.5. 修改 SUDO 命令和命令组

在 Web UI 中修改 sudo 命令和命令组

1. 在 Policy 选项卡下，单击 Sudo → Sudo commands 或 Sudo → Sudo Command Groups。

2. 单击命令或命令组的名称，以显示其配置页面。

3. 根据需要更改设置。在某些配置页面中，保存按钮 位于页面的顶部。在这些页面中，您必须单击 按钮以确认更改。

从命令行修改 sudo 命令和命令组

要修改命令或命令组，请使用以下命令：

- ipa sudocmd-mod
- ipa sudocmdgroup-mod

在以上命令中添加命令行选项，以更新 sudo 命令或命令行组属性。例如，要为 /usr/bin/less 命令添加新描述：

```
$ ipa sudocmd-mod /usr/bin/less --desc="For reading log files"
---------------------------------------
Modified Sudo Command "/usr/bin/less"
---------------------------------------
  Sudo Command: /usr/bin/less
  Description: For reading log files
  Sudo Command Groups: files
```

有关这些命令及其接受的选项的更多信息，请在添加 --help 选项的情况下运行它们。

30.6. 修改 SUDO 规则

在 Web UI 中修改 sudo 规则

1. 在 Policy 选项卡下，单击 Sudo → Sudo Rules。
2. 单击规则的名称，以显示其配置页面。
3. 根据需要更改设置。在某些配置页面中，保存按钮 位于页面的顶部。在这些页面上，单击 按钮以确认更改。

sudo 规则配置页面包含多个配置区域：

常规 区域

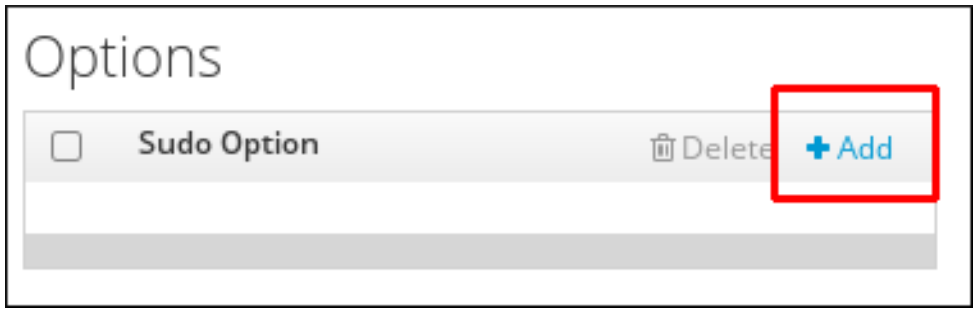
在此区域中，您可以修改规则的描述和 sudo 顺序。sudo order 字段接受整数，并定义 IdM 评估规则的顺序。首先评估具有最高 sudo 顺序 值的规则。

Options 区域

在这个区域中，您可以在规则中添加 sudoers 选项。

1. 点选项列表上方的 Add。

图 30.5. 添加 sudo 选项



2. 输入 sudoers 选项。例如，要指定 sudo 不会提示用户进行身份验证，请添加 !authenticate 选项：

图 30.6. 输入 sudoers 选项

有关 sudoers 选项的详情请参考 sudoers(5) man page。

3. 点击 Add。

Who 区域

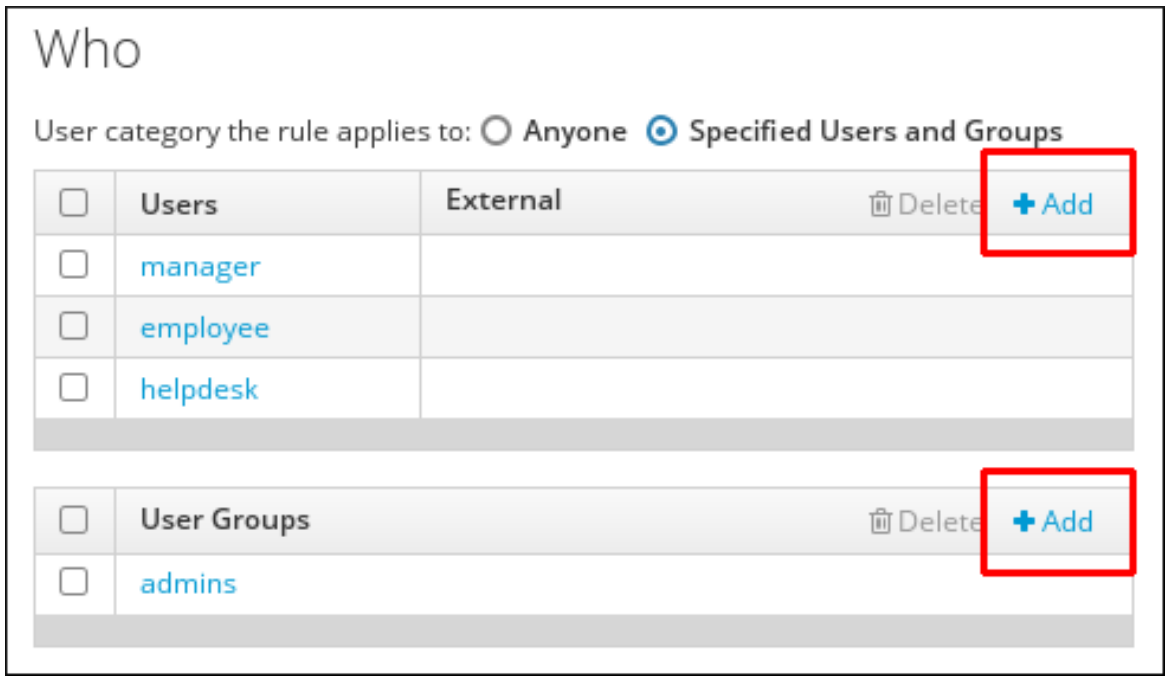
在此区域中，您可以选择将 sudo 规则应用到的用户或用户组。这些用户将有权使用规则中定义的 sudo。

要指定所有系统用户可以使用规则中定义的 sudo，请选择 Anyone。

要仅将规则应用到特定的用户或组，请选择 指定的用户和组，然后按照以下步骤操作：

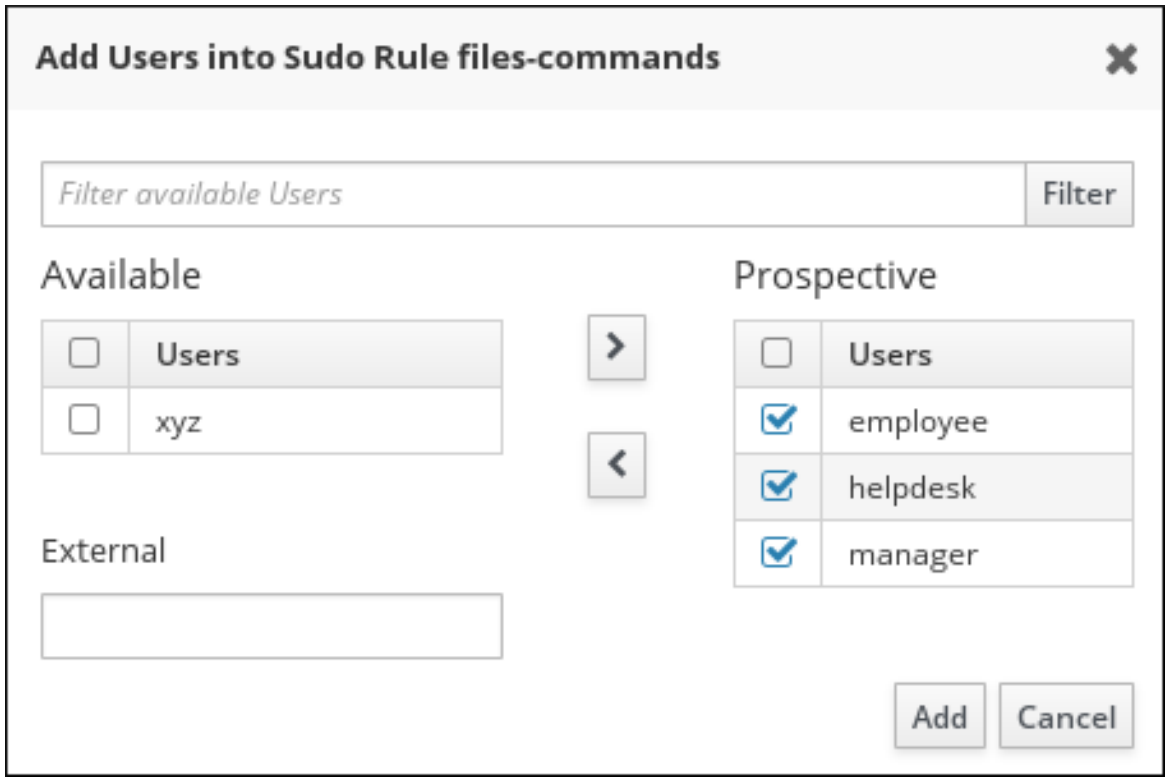
1. *点用户或用户组列表上方的 Add。*

图 30.7. 将用户添加到 sudo 规则



2. *选择要添加到规则的用户或用户组，然后点击 > 箭头按钮将它们移到 Prospective 列中。要添加外部用户，请在 External 字段中指定用户，然后点击 > 箭头按钮。*

图 30.8. 为 sudo 规则选择用户



3. *点击 Add。*

访问此主机 区域

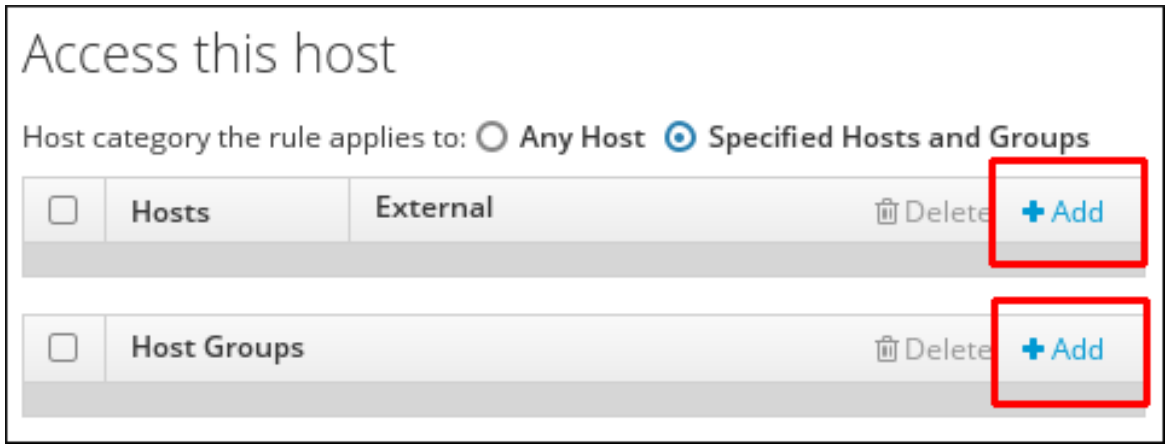
在此区域中，您可以选择 sudo 规则将生效的主机。这些是授予用户 sudo 权限的主机。

要指定该规则将在所有主机中生效，请选择 Anyone。

要仅将规则应用到特定的主机或主机组，请选择指定的 主机和组，然后按照以下步骤操作：

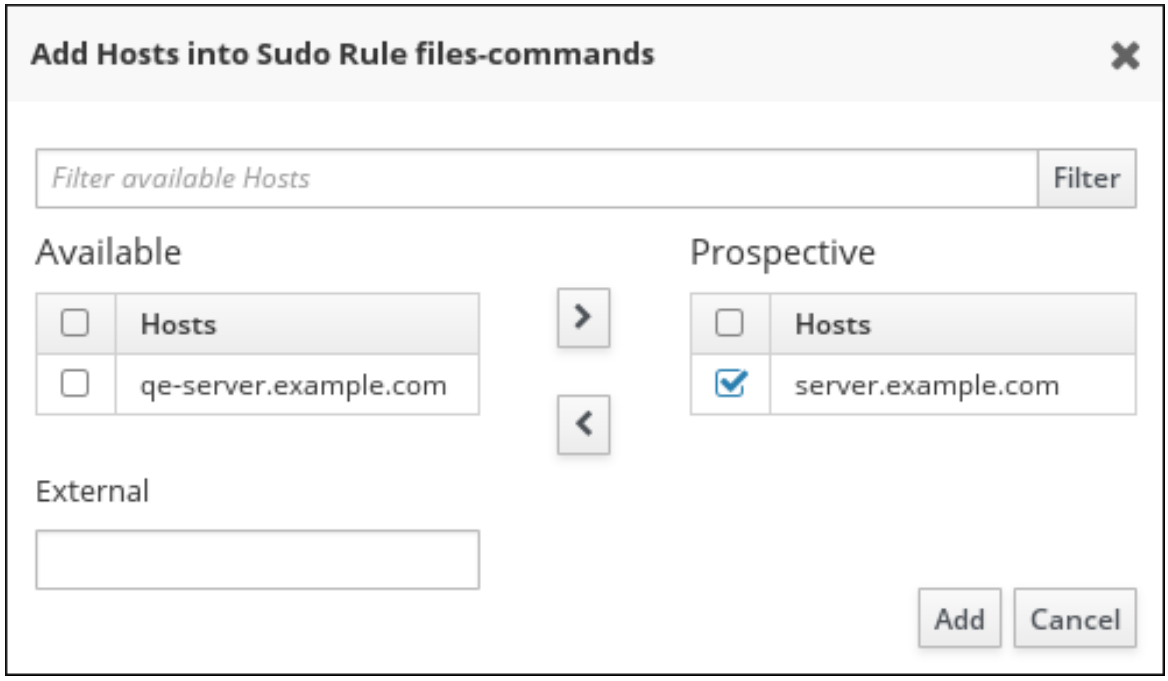
1. *点主机列表上方的 Add。*

图 30.9. 将主机添加到 sudo 规则



2. *选择要包含在该规则中的主机或主机组，然后点击 > 箭头按钮将它们移到 Prospective 列中。要添加外部主机，请在 External 字段中指定主机，然后点击 > 箭头按钮。*

图 30.10. 为 sudo 规则选择主机



3. *点击 Add。*

运行命令 区域

在这个区域中，您可以选择要包含在 sudo 规则中的命令。您可以指定允许或拒绝用户使用特定的命令。

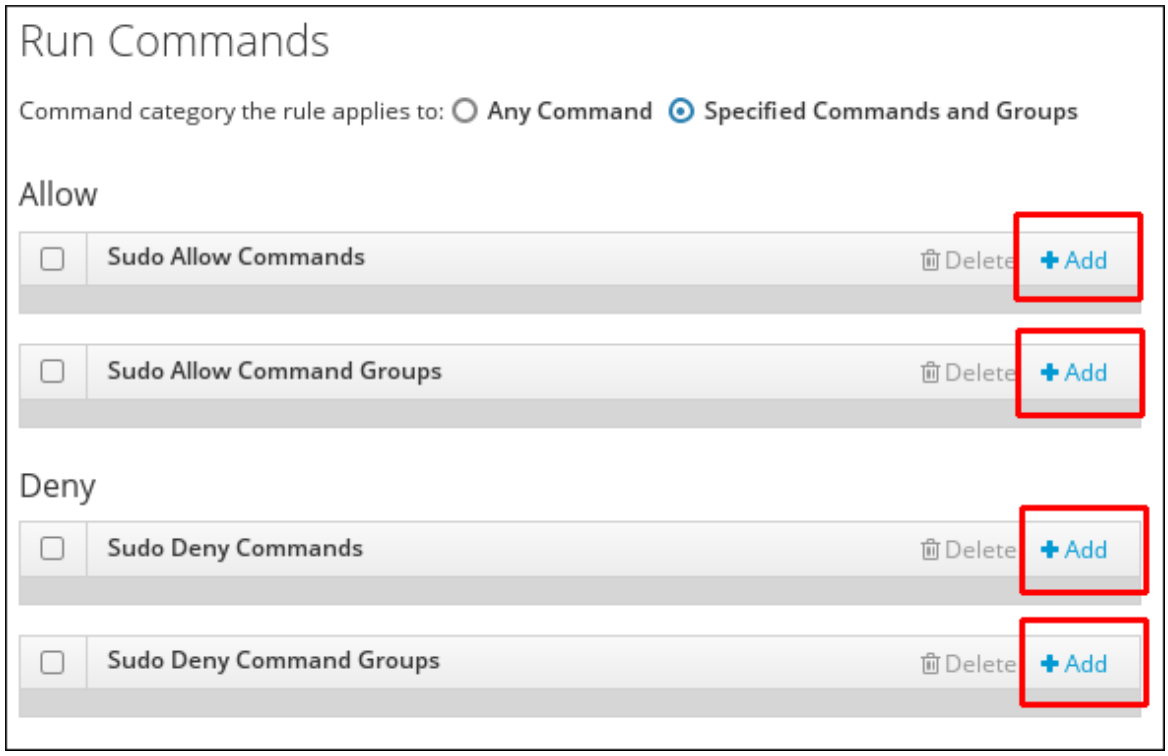
要指定允许用户使用带有 sudo 的任何命令，请选择 任何命令。

要将规则与特定的命令或命令组关联，请选择 指定的命令 和组，然后按照以下步骤操作：

1. 点 Add 按钮之一来添加命令或命令组。

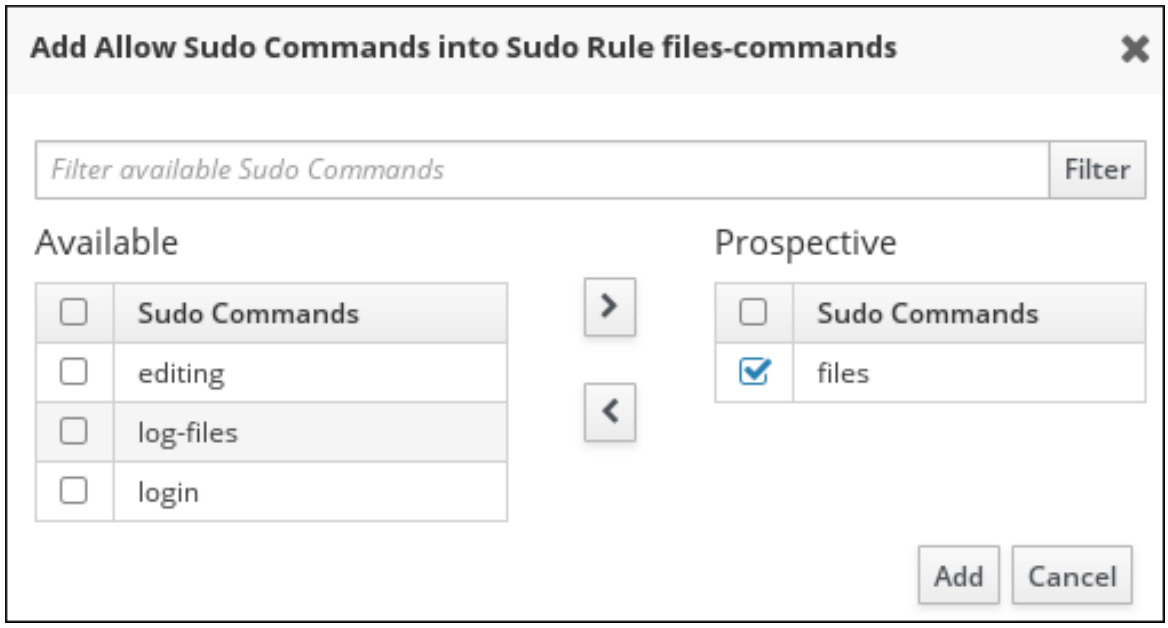
   要指定允许的命令或命令组，请使用 Allow 区域。要指定被拒绝的命令或命令组，请使用 Deny 区域。

   图 30.11. 在 sudo 规则中添加命令



2. 选择要包含在规则中的命令或命令组，然后点击 > 箭头按钮将它们移到 Prospective 列中。

图 30.12. 为 sudo 规则选择命令



3. *点击 Add。*

As Whom 区域

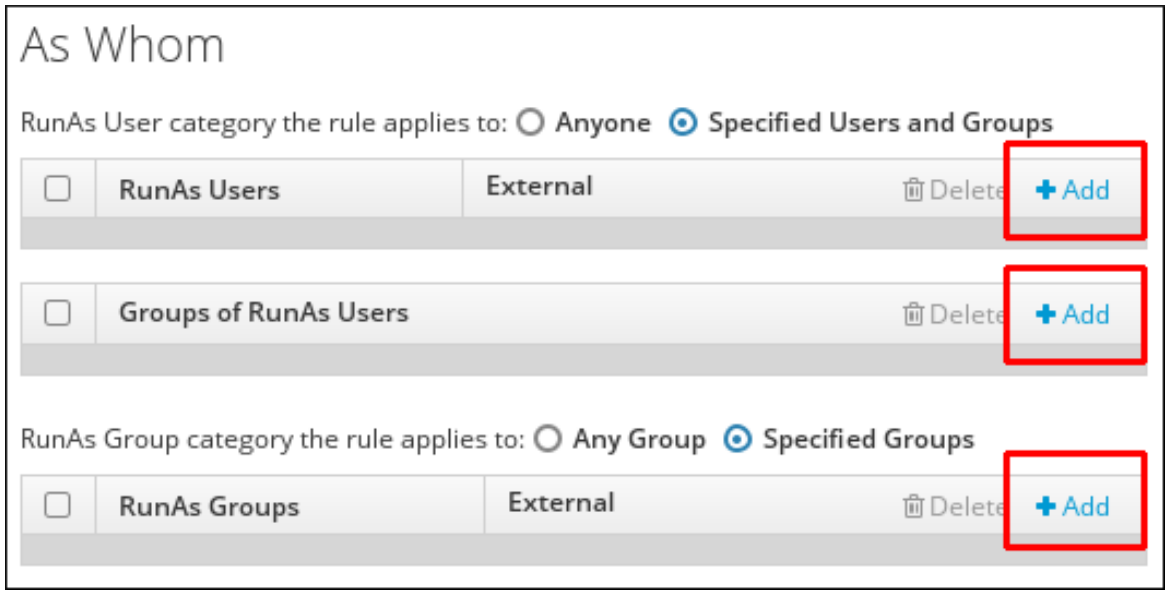
在此区域中，您可以配置 sudo 规则，以特定、非 root 用户身份运行给定的命令。

请注意，如果添加了一组 RunAs 用户，则将使用 组成员的 UID 来运行命令。如果添加 RunAs 组，则将使用组 的 GID 来运行 命令。

要指定将以系统上的任何用户身份运行该规则，请选择 Anyone。要指定该规则将作为系统上的任何组运行，请选择 任何组。

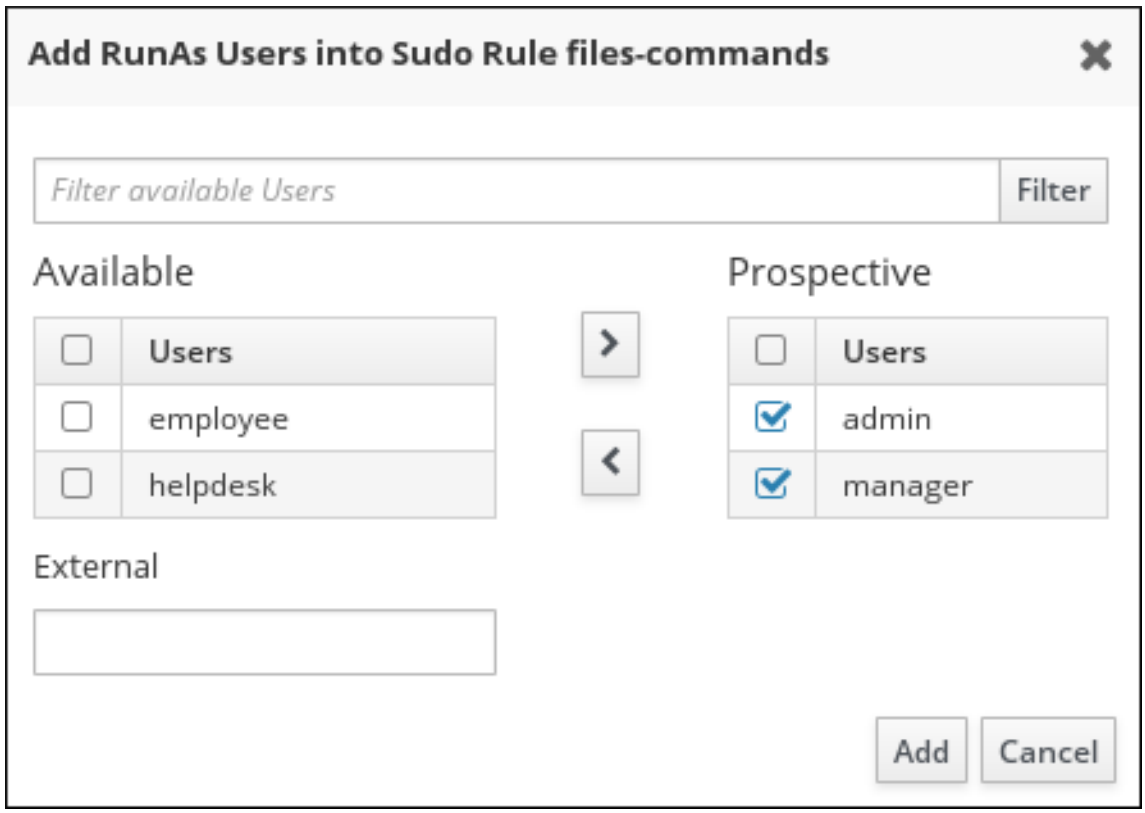
1. *点 users 列表上方的 Add。*

图 30.13. 将 sudo 规则配置为以特定用户身份执行命令



2. *选择所需的用户或组，并使用 > 箭头按钮将它们移到 Prospective 列中。要添加外部实体，请在 External 字段中指定它，然后点击 > 箭头按钮。*

图 30.14. 为命令选择用户



3. *点击 Add。*

从命令行修改 sudo 规则

IdM 命令行工具允许您配置多个 sudo 规则区域：

常规 sudo 规则管理

要更改 sudo 规则的常规配置，请使用 ipa sudorule-mod 命令。命令接受的最常见选项有：

- 用于更改 sudo 规则描述的 --desc 选项。例如：

```
$ ipa sudorule-mod sudo_rule_name --desc="sudo_rule_description"
```

- 用于定义指定规则顺序的 --order 选项。例如：

```
$ ipa sudorule-mod sudo_rule_name --order=3
```

- 指定实体类别的选项： --usercat （用户类别）、 --hostcat（主机 类别）、 --cmdcat（命令类别）、 --runasusercat (run-as user category)和 --runasgroupcat (run-as group category)。这些选项仅接受 所有将规则 与所有用户、主机、命令、运行用户或 run-as 组关联的值。

  例如，要指定所有用户都能够使用 sudo _rule 规则中定义的 sudo：

```
$ ipa sudorule-mod sudo_rule --usercat=all
```

  请注意，如果该规则已经与特定实体关联，您必须在定义对应的 所有类别 前将其删除。例如，如果 sudo_rule 之前使用 ipa sudorule-add-user 命令与特定用户关联，您必须首先使用 ipa sudorule-remove-user 命令删除用户。

如需更多详细信息以及 ipa sudorule-mod 接受的选项的完整列表，请使用 --help 选项运行命令。

管理 sudo 选项

要添加 sudoers 选项，请使用 ipa sudorule-add-option 命令。

例如，要根据 files-commands 规则使用 sudo 来指定用户进行验证，请添加 !authenticate 选项：

```
$ ipa sudorule-add-option files-commands
Sudo Option: !authenticate
-----------------------------------------------------
Added option "!authenticate" to Sudo Rule "files-commands"
-----------------------------------------------------
```

有关 sudoers 选项的详情请参考 sudoers(5) man page。

要删除 sudoers 选项，请使用 ipa sudorule-remove-option 命令。例如：

```
$ ipa sudorule-remove-option files-commands
Sudo Option: authenticate
-----------------------------------------------------------
Removed option "authenticate" from Sudo Rule "files-commands"
-----------------------------------------------------------
```

管理被授予使用 sudo的权限

要指定单个用户，请在 ipa sudorule-add-user 命令中添加 --users 选项。要指定用户组，请将 --groups 选项添加到 ipa sudorule-add-user 中。

例如，要将 user 和 user_group 添加到 files-commands 规则中：

```
$ ipa sudorule-add-user files-commands --users=user --groups=user_group
...
-------------------------
Number of members added 2
-------------------------
```

要删除单独的用户或组，请使用 ipa sudorule-remove-user。例如，要删除用户：

```
$ ipa sudorule-remove-user files-commands
[member user]: user
[member group]:
...
---------------------------
Number of members removed 1
---------------------------
```

管理用户在什么位置被授予 sudo 权限

要指定主机，请在 ipa sudorule-add-host 命令中添加 --hosts 选项。要指定主机组，请将 --hostgroups 选项添加到 ipa sudorule-add-host。

例如，要将 example.com 和 host_group 添加到 files-commands 规则中：

```
$ ipa sudorule-add-host files-commands --hosts=example.com --hostgroups=host_group
...
-------------------------
Number of members added 2
-------------------------
```

要删除主机或主机组，请使用 ipa sudorule-remove-host 命令。例如：

```
$ ipa sudorule-remove-host files-commands
[member host]: example.com
[member host group]:
...
---------------------------
Number of members removed 1
---------------------------
```

管理可与 sudo一起使用的命令

您可以指定允许或拒绝用户使用特定的命令。

要指定允许的命令或命令组，请在 ipa sudorule-add-allow-command 中添加 --sudocmds 或 --sudocmdgroups 选项。要指定被拒绝的命令或命令组，请在 ipa sudorule-add-deny-command 命令中添加 --sudocmds 或 --sudocmdgroups 选项。

例如，要在 files-commands 规则中添加 /usr/bin/less 命令和 files 命令组：

```
$ ipa sudorule-add-allow-command files-commands --sudocmds=/usr/bin/less --
sudocmdgroups=files
...
-------------------------
Number of members added 2
-------------------------
```

要从规则中删除命令或命令组，请使用 ipa sudorule-remove-allow-command 或 ipa sudorule-remove-deny-command 命令。例如：

```
$ ipa sudorule-remove-allow-command files-commands
[member sudo command]: /usr/bin/less
[member sudo command group]:
...
```

```
---------------------------
Number of members removed 1
---------------------------
```

请注意，--sudocmds 选项只接受添加到 IdM 的命令，如 第 30.4.1 节 “添加 sudo 命令” 所述。

以谁方式运行 sudo 命令

要将组中的单个用户或用户的 UID 用作运行命令的身份，请使用 ipa sudorule-add-runasuser 命令的 --users 或 --groups 选项。

要使用用户组的 GID 作为命令的身份，请使用 ipa sudorule-add-runasgroup --groups 命令。

如果没有指定用户或组，sudo 命令将以 root 用户身份运行。

例如，指定将使用 用户的身份 在 sudo 规则中执行命令：

```
$ ipa sudorule-add-runasuser files-commands --users=user
...
RunAs Users: user
...
```

有关 ipa sudorule the 命令的更多信息，请参阅 ipa help sudorule 命令的输出或运行带有 --help 选项的特定命令。

例 30.1. 从命令行添加和修改新的 sudo 规则

在所选服务器上允许特定的用户组使用 sudo 和任何命令：

1. 为 admin 用户或允许管理 sudo 规则的任何其他用户获取 Kerberos 票据。

```
$ kinit admin
Password for admin@EXAMPLE.COM:
```

2. 向 IdM 添加新的 sudo 规则。

```
$ ipa sudorule-add new_sudo_rule --desc="Rule for user_group"
--------------------------------
Added Sudo Rule "new_sudo_rule"
--------------------------------
  Rule name: new_sudo_rule
  Description: Rule for user_group
  Enabled: TRUE
```

3. **定义谁：指定将使用 sudo 规则的用户组。**

```
$ ipa sudorule-add-user new_sudo_rule --groups=user_group
  Rule name: new_sudo_rule
  Description: Rule for user_group
  Enabled: TRUE
  User Groups: user_group
------------------------
Number of members added 1
------------------------
```

4. **定义：指定授予用户 sudo 权限的主机组的位置。**

```
$ ipa sudorule-add-host new_sudo_rule --hostgroups=host_group
  Rule name: new_sudo_rule
  Description: Rule for user_group
  Enabled: TRUE
  User Groups: user_group
  Host Groups: host_group
------------------------
Number of members added 1
------------------------
```

5. **定义允许用户运行任何 sudo 命令的：，将所有命令类别添加到规则中。**

```
$ ipa sudorule-mod new_sudo_rule --cmdcat=all
-------------------------------
Modified Sudo Rule "new_sudo_rule"
-------------------------------
  Rule name: new_sudo_rule
  Description: Rule for user_group
  Enabled: TRUE
  Command category: all
  User Groups: user_group
  Host Groups: host_group
```

6. **要让 sudo 命令以 root 用户身份执行，请不要指定任何作为用户或组运行的运行。**

7. 添加 !authenticate sudoers 选项，以指定在使用 sudo 命令时不需要用户进行身份验证。

```
$ ipa sudorule-add-option new_sudo_rule
Sudo Option: !authenticate
-----------------------------------------------------
Added option "!authenticate" to Sudo Rule "new_sudo_rule"
-----------------------------------------------------
  Rule name: new_sudo_rule
  Description: Rule for user_group
  Enabled: TRUE
  Command category: all
  User Groups: user_group
  Host Groups: host_group
  Sudo Option: !authenticate
```

8. 显示新的 sudo 规则配置以验证其是否正确。

```
$ ipa sudorule-show new_sudo_rule
  Rule name: new_sudo_rule
  Description: Rule for user_group
  Enabled: TRUE
  Command category: all
  User Groups: user_group
  Host Groups: host_group
  Sudo Option: !authenticate
```

30.7. 列出并显示 SUDO 命令、命令组和规则

在 Web UI 中列出并显示 sudo 命令、命令组和规则

1. 在 Policy 选项卡下，单击 Sudo，再选择 Sudo Rules、Sudo Commands 或 Sudo Command Groups。

2. 单击规则、命令或命令组的名称，以显示其配置页面。

从命令行列出并显示 sudo 命令、命令组和规则

要列出所有命令、命令组和规则，请使用以下命令：

- ipa sudocmd-find

- ipa sudocmdgroup-find
- ipa sudorule-find

要显示特定命令、命令组或规则的信息，请使用以下命令：

- ipa sudocmd-show
- ipa sudocmdgroup-show
- ipa sudorule-show

例如，要显示有关 /usr/bin/less 命令的信息：

```
$ ipa sudocmd-show /usr/bin/less
  Sudo Command: /usr/bin/less
  Description: For reading log files.
  Sudo Command Groups: files
```

有关这些命令及其接受的选项的更多信息，请在添加 --help 选项的情况下运行它们。

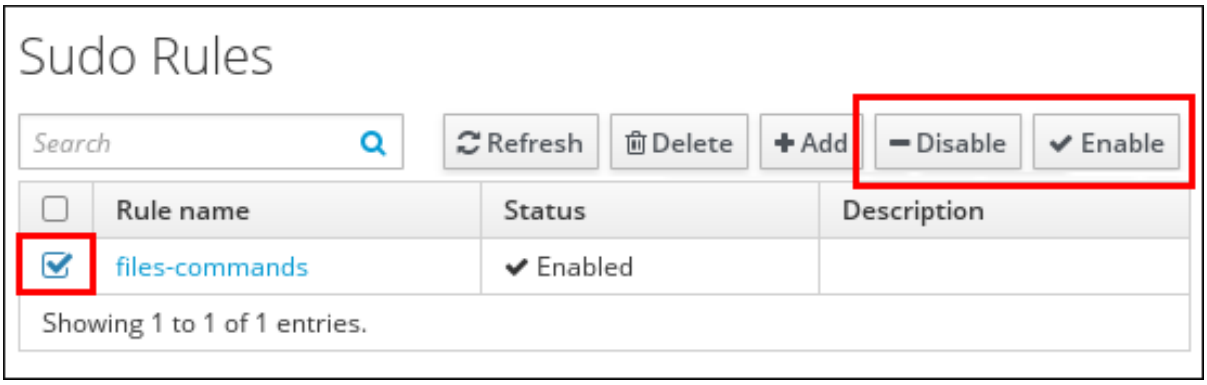
30.8. 禁用并启用 SUDO 规则

禁用 sudo 规则会临时停用它。禁用的规则不会从 IdM 中删除，可以再次启用。

从 Web UI 禁用并启用 sudo 规则

1. 在 Policy 选项卡下，单击 Sudo → Sudo Rule。
2. 选择要禁用的规则，然后单击 Disable 或 Enable。

图 30.15. 禁用或启用 sudo 规则



从命令行禁用和启用 sudo 规则

要禁用规则，请使用 ipa sudo-rule-disable 命令。

```
$ ipa sudorule-disable sudo_rule_name
----------------------------------
Disabled Sudo Rule "sudo_rule_name"
----------------------------------
```

要重新启用规则，请使用 ipa sudorule-enable 命令。

```
$ ipa sudorule-enable sudo_rule_name
----------------------------------
Enabled Sudo Rule "sudo_rule_name"
----------------------------------
```

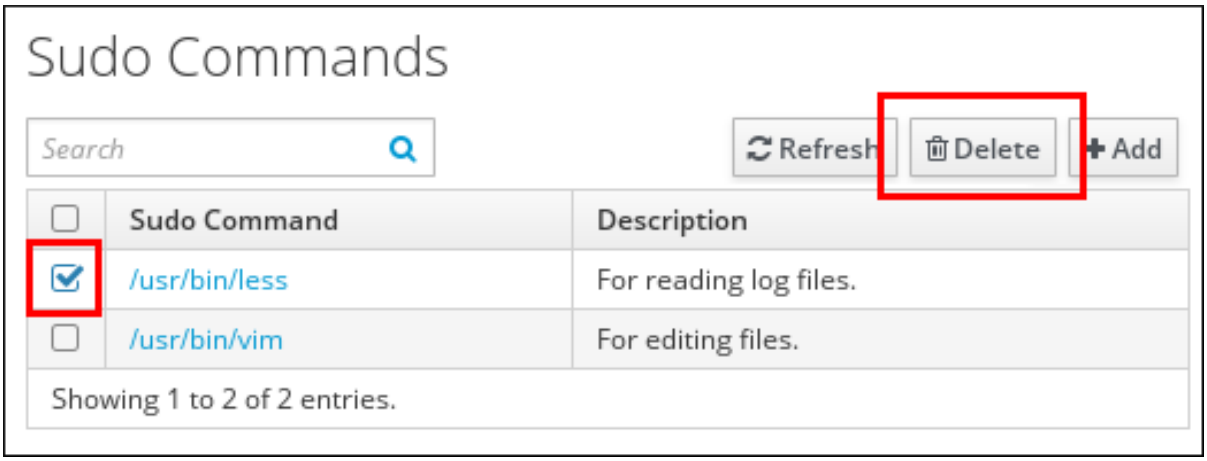
30.9. 删除 SUDO 命令、命令组和规则

在 Web UI 中删除 sudo 命令、命令组和规则

1. *在 Policy 选项卡下，单击 Sudo，再选择 Sudo Rules、Sudo Commands 或 Sudo Command Groups。*

2. *选择要删除的命令、命令组或规则，然后单击 Delete。*

图 30.16. 删除 sudo 命令



从命令行删除 sudo 命令、命令组和规则

要删除命令、命令组或规则，请使用以下命令：

- ***ipa sudocmd-del***
- ***ipa sudocmdgroup-del***
- ***ipa sudorule-del***

*有关这些命令及其接受的选项的更多信息，请在添加 **--help** 选项的情况下运行它们。*

30.10. 其它资源

有关在将身份管理环境迁移到 Red Hat Enterprise Linux 7 中的新环境时导入和导出 sudo 规则的详情，请参考知识库解决方案。

第 31 章 配置基于主机的访问控制

本章论述了身份管理(IdM)中 基于主机的访问控制 (HBAC)，并解释如何使用 HBAC 管理 IdM 域中的访问控制。

31.1. IDM 中基于主机的访问控制如何工作

基于主机的访问控制通过使用指定的服务（或服务组中的服务）定义哪些用户（或用户组）可以访问指定的主机（或主机组）。例如，您可以：

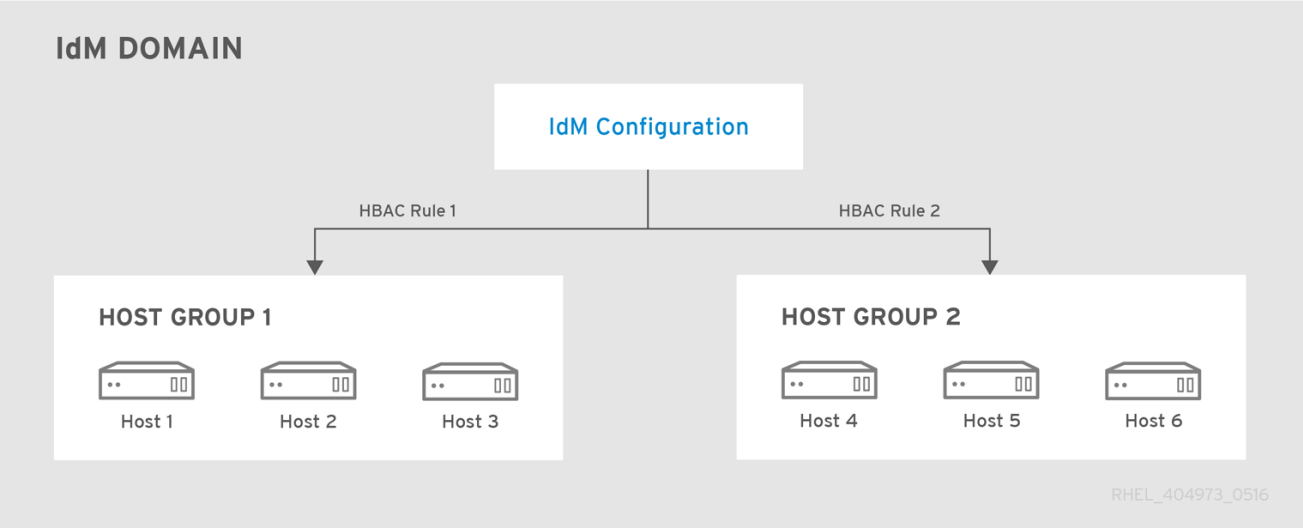
- 将您域中对指定系统的访问权限限制为特定用户组的成员。
- 仅允许使用特定的服务访问域中的系统。

管理员使用一组允许名为 HBAC 规则 的规则来配置基于主机的访问控制。默认情况下，IdM 配置成一个名为 allow_all 的默认 HBAC 规则，该规则允许在整个 IdM 域中进行通用访问。

将 HBAC 规则应用到组

要集中和简化访问控制管理，您可以将 HBAC 规则应用到整个用户、主机或服务组，而不是单个用户、主机或服务。

图 31.1. 主机组和基于主机的访问控制



将 HBAC 规则应用到组时，请考虑 使用自动成员规则。请参阅 第 13.6 节 “为用户和主机定义自动组成员资格”。

31.2. 在 IDM 域中配置基于主机的访问控制

为基于主机的访问控制配置域：

1. 创建 HBAC 规则
2. 测试新的 HBAC 规则
3. 禁用默认的 allow_all HBAC 规则

重要

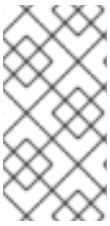
在创建自定义 HBAC 规则前，不要禁用 allow_all 规则。如果您这样做，则没有用户能够访问任何主机。

31.2.1. 创建 HBAC 规则

要创建 HBAC 规则，您可以使用：

- IdM Web UI（请参阅 “Web UI：创建 HBAC 规则”一节）
- 命令行（请参见 “命令行：创建 HBAC 规则”一节）

有关示例，请参阅 “HBAC 规则示例”一节。



注意

IdM 将用户的主组存储为 gidNumber 属性的数字值，而不是指向 IdM 组对象的链接。因此，HBAC 规则只能引用用户的补充组，而不能引用其主组。

Web UI：创建 HBAC 规则

1. 选择 基于策略的访问控制 → HBAC 规则。

2. 单击 **Add** 以开始添加新规则。

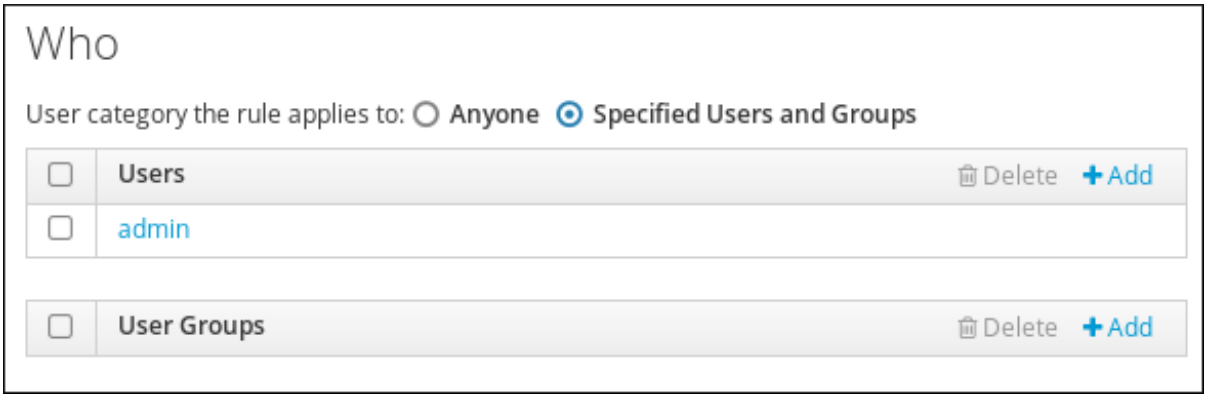
3. 输入规则的名称，然后点 **Add** 和 **Edit** 以直接进入 **HBAC** 规则配置页面。

4. 在 **Who** 区域，指定目标用户。

   - 要将 **HBAC** 规则应用到指定的用户或组，请选择 **指定的用户和组。然后点击 Add 添加用户或组。**

   - 要将 **HBAC** 规则应用到所有用户，请选择 **Anyone。**

图 **31.2.** 为 **HBAC** 规则指定目标用户



5. 在 **Accessing** 区域中指定目标主机：

   - 要将 **HBAC** 规则应用到指定的主机或主机组，请选择 **指定的主机和组。然后单击 Add 以添加主机或主机组。**

   - 要将 **HBAC** 规则应用到所有主机，请选择 **任何主机。**

6. 在 **Via Service** 区域中，指定目标 **HBAC** 服务：

   - 要将 **HBAC** 规则应用到指定的服务或组，请选择**指定的服务和组。然后点击 Add 添加服务或组。**

- 要将 HBAC 规则应用到所有服务，请选择 任何 Service。

注意

默认情况下，只有最常见的服务和服务组是为 HBAC 规则配置。

- 要显示当前可用的服务列表，请选择 基于策略的访问控制 → HBAC 服务。
- 要显示当前可用的服务组列表，请选择 基于策略的访问控制 → HBAC Service Groups。

要添加更多服务和服务组，请参阅 第 31.3 节 “为自定义 HBAC 服务添加 HBAC Service Entries” 和 第 31.4 节 “添加 HBAC 服务组”。

7. 更改 HBAC 规则配置页面中的某些设置，突出显示页面顶部的 Save 按钮。如果发生此情况，请单击 按钮以确认更改。

命令行：创建 HBAC 规则

1. 使用 ipa hbacrule-add 命令添加规则。

```
$ ipa hbacrule-add
Rule name: rule_name
---------------------------
Added HBAC rule "rule_name"
----------------------------
  Rule name: rule_name
  Enabled: TRUE
```

2. 指定目标用户。

- 要将 HBAC 规则应用到指定的用户或组，请使用 ipa hbacrule-add-user 命令。

例如，要添加组：

```
$ ipa hbacrule-add-user
Rule name: rule_name
[member user]:
[member group]: group_name
  Rule name: rule_name
  Enabled: TRUE
  User Groups: group_name
-------------------------
Number of members added 1
-------------------------
```

要添加多个用户或组，请使用 --users 和 --groups 选项：

```
$ ipa hbacrule-add-user rule_name --users=user1 --users=user2 --users=user3
  Rule name: rule_name
  Enabled: TRUE
  Users: user1, user2, user3
-------------------------
Number of members added 3
-------------------------
```

- **要将 HBAC 规则应用到所有用户，请使用 ipa hbacrule-mod 命令并指定 所有用户 类别：**

```
$ ipa hbacrule-mod rule_name --usercat=all
------------------------------
Modified HBAC rule "rule_name"
------------------------------
  Rule name: rule_name
  User category: all
  Enabled: TRUE
```

注意

如果 HBAC 规则与单个用户或组关联，ipa hbacrule-mod --usercat=all 会失败。在这种情况下，使用 ipa hbacrule-remove-user 命令删除用户和组。

详情请参阅使用 **--help** 选项运行 **ipa hbacrule-remove-user**。

3. **指定目标主机。**

- 要将 HBAC 规则应用到指定的主机或主机组，请使用 ipa hbacrule-add-host 命令。

  例如，要添加单个主机：

```
$ ipa hbacrule-add-host
Rule name: rule_name
[member host]: host.example.com
[member host group]:
  Rule name: rule_name
  Enabled: TRUE
  Hosts: host.example.com
-------------------------
Number of members added 1
-------------------------
```

  要添加多个主机或组，请使用 --hosts 和 --hostgroups 选项：

```
$ ipa hbacrule-add-host rule_name --hosts=host1 --hosts=host2 --hosts=host3
  Rule name: rule_name
  Enabled: TRUE
  Hosts: host1, host2, host3
-------------------------
Number of members added 3
-------------------------
```

- 要将 HBAC 规则应用到所有主机，请使用 ipa hbacrule-mod 命令并指定 所有主机 类别：

```
$ ipa hbacrule-mod rule_name --hostcat=all
-----------------------------
Modified HBAC rule "rule_name"
-----------------------------
  Rule name: rule_name
  Host category: all
  Enabled: TRUE
```

注意

如果 HBAC 规则与单个主机或主机组关联，ipa hbacrule-mod --hostcat=all 会失败。在这种情况下，使用 ipa hbacrule-remove-host 命令删除主机和组。

详情请参阅使用 --help 选项运行 ipa hbacrule-remove-host。

4. 指定目标 HBAC 服务。

- 要将 HBAC 规则应用到指定的服务或组，请使用 ipa hbacrule-add-service 命令。

例如，要添加单个服务：

```
$ ipa hbacrule-add-service
Rule name: rule_name
[member HBAC service]: ftp
[member HBAC service group]:
Rule name: rule_name
Enabled: TRUE
Services: ftp
-------------------------
Number of members added 1
-------------------------
```

要添加多个服务或组，您可以使用 --hbacsvcs 和 --hbacsvcgroups 选项：

```
$ ipa hbacrule-add-service rule_name --hbacsvcs=su --hbacsvcs=sudo
  Rule name: rule_name
  Enabled: TRUE
  Services: su, sudo
-------------------------
Number of members added 2
-------------------------
```

注意

只有最常见的服务和服务组是为 HBAC 规则配置。要添加更多，请查看 第 31.3 节 “为自定义 HBAC 服务添加 HBAC Service Entries” 和 第 31.4 节 “添加 HBAC 服务组”。

- 要将 HBAC 规则应用到所有服务，请使用 ipa hbacrule-mod 命令并指定 所有服务 类别：

```
$ ipa hbacrule-mod rule_name --servicecat=all
-----------------------------
Modified HBAC rule "rule_name"
-----------------------------
  Rule name: rule_name
  Service category: all
  Enabled: TRUE
```

注意

如果 HBAC 规则与各个服务或组关联，ipa hbacrule-mod --servicecat=all 会失败。在这种情况下，使用 ipa hbacrule-remove-service 命令删除服务和组。

详情请参阅使用 --help 选项运行 ipa hbacrule-remove-service。

5. 可选。验证 HBAC 规则是否已正确添加。

a. 使用 ipa hbacrule-find 命令来验证 HBAC 规则是否已添加到 IdM。

b. 使用 ipa hbacrule-show 命令验证 HBAC 规则的属性。

详情请使用 --help 选项运行命令。

HBAC 规则示例

例 31.1. 使用任何服务授予单用户对所有主机的访问权限

要允许 admin 用户使用任何服务访问域中的所有系统，请创建一个新的 HBAC 规则并设置：

- 要管理 的用户
- 主机到任何主机（在 Web UI 中），或使用 --hostcat=all 与 ipa hbacrule-add （添加规则时）或 ipa hbacrule-mod

- 服务到任何服务（在 Web UI 中），或使用 --servicecat=all 与 ipa hbacrule-add （添加规则时）或 ipa hbacrule-mod

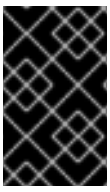
例 31.2. 确保只能用于访问主机的特定服务

要确保所有用户都必须使用 sudo相关服务来访问名为 host.example.com 的主机，请创建一个新的 HBAC 规则并设置：

- 用户到 任何人 （在 Web UI 中），或使用 --usercat=all 与 ipa hbacrule-add （添加规则时）或 ipa hbacrule-mod
- 到 host.example.com的主机
- HBAC 服务组为 Sudo，这是 sudo 和相关服务的默认组

31.2.2. 测试 HBAC 规则

IdM 可让您在各种情况下使用模拟场景测试 HBAC 配置。通过执行这些模拟测试运行，您可以在生产中部署 HBAC 规则之前发现错误配置问题或安全风险。



重要

在开始在生产环境中使用自定义 HBAC 规则前，请始终测试它们。

请注意，IdM 不会测试 HBAC 规则对可信 Active Directory(AD)用户的影响。因为 AD 数据没有存储在 IdM LDAP 目录中，所以 IdM 在模拟 HBAC 场景时无法解析 AD 用户的组成员身份。

要测试 HBAC 规则，您可以使用：

- IdM Web UI（请参阅 “Web UI：测试 HBAC 规则”一节）

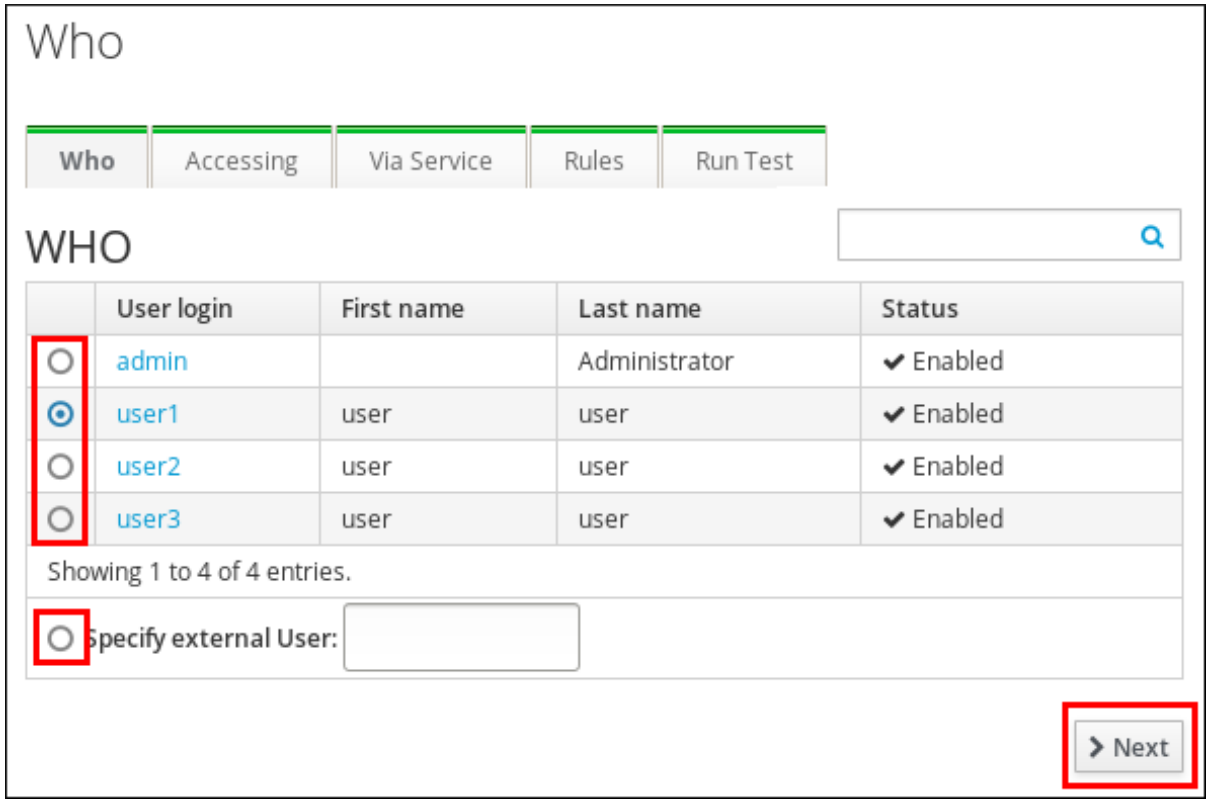
- 命令行（请参见 *"命令行：测试 HBAC 规则"一节*）

Web UI：测试 HBAC 规则

1. 选择 **Policy** → **Host-Based Access** → **HBAC Test**。

2. 在 **Who** 屏幕中：指定您要执行测试的身份下的用户，然后单击 **Next**。

图 31.3. 为 HBAC 测试指定目标用户



3. 在 **Accessing** 屏幕中：指定用户可访问的主机，然后单击 **Next**。

4. 在 **Via Service** 屏幕中：指定用户要使用的服务，然后单击 **Next**。

5. 在 **Rules** 屏幕中：选择您要测试的 **HBAC** 规则，然后点 **Next**。如果不选择任何规则，则会测试所有规则。

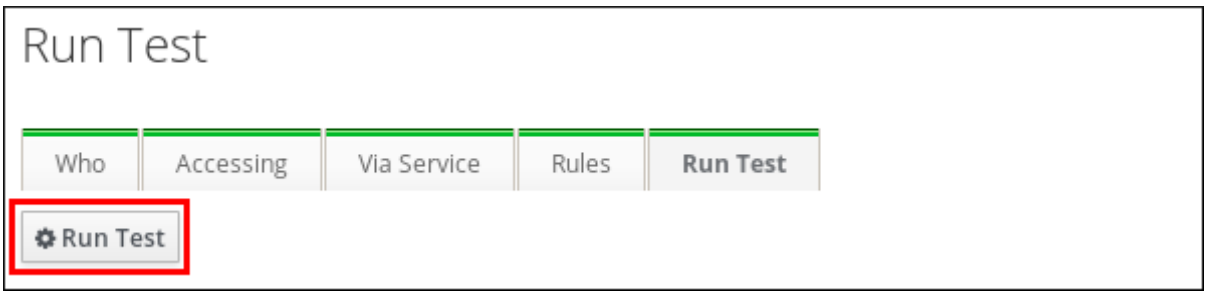
选择 **Include Enabled**，以对所有状态为 **Enabled** 的规则运行测试。选择 **Include Disabled** 在状态为 **Disabled** 的所有规则上运行测试。要查看并更改 **HBAC** 规则的状态，请选择 **基于策略的访问控制** → **HBAC** 规则。

重要

如果测试在多个规则上运行，则如果至少一个所选规则允许访问，它将成功传递。

6. **在 Run Test 屏幕中：单击 Run Test。**

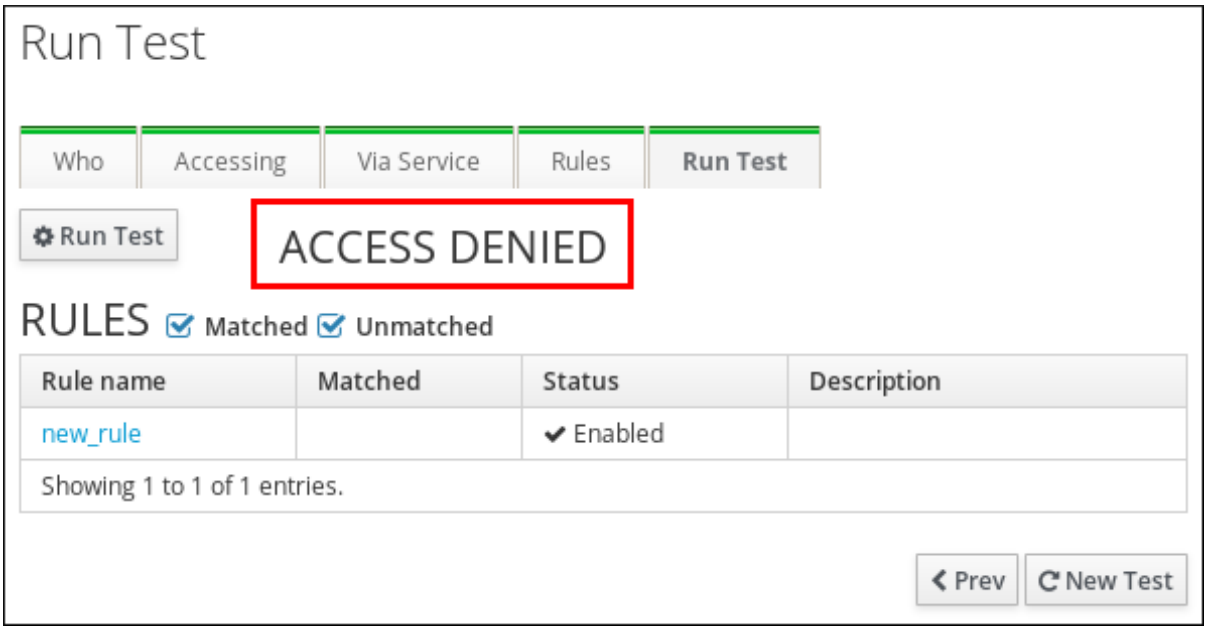
图 31.4. 运行 HBAC 测试



7. **查看测试结果：**

- **如果您看到 ACCESS DENIED，该用户没有在测试中授予访问权限。**
- **如果您看到 ACCESS GRANTED，用户可以成功访问主机。**

图 31.5. 查看 HBAC 测试结果



默认情况下，在显示测试结果时，IdM 会列出所有经过测试的 HBAC 规则。

- 选择 Matched 以显示允许成功访问的规则。
- 选择 Unmatched 来显示阻止访问的规则。

命令行：测试 HBAC 规则

使用 ipa hbactest 命令并至少指定：

- 要执行测试其身份的用户
- 用户将尝试访问的主机
- 用户将尝试使用的服务

例如，在以交互方式指定这些值时：

```
$ ipa hbactest
User name: user1
Target host: example.com
Service: sudo
----------------------
Access granted: False
----------------------
Not matched rules: rule1
```

默认情况下，IdM 在启用了 其状态的所有 HBAC 规则上运行测试。指定不同的 HBAC 规则：

- 使用 --rules 选项定义一个或多个 HBAC 规则。
- 使用 -- disabled 选项测试其状态禁用的所有 HBAC 规则。

要查看 HBAC 规则的当前状态，请运行 ipa hbacrule-find 命令。

例 31.3. 从命令行测试 HBAC 规则

在以下测试中，名为 rule2 的 HBAC 规则阻止 user1 使用 sudo 服务访问 example.com ：

```
$ ipa hbactest --user=user1 --host=example.com --service=sudo --rules=rule1
--------------------
Access granted: False
--------------------
  Not matched rules: rule1
```

例 31.4. 从命令行测试多个 HBAC 规则

测试多个 HBAC 规则时，如果至少一条规则允许用户成功访问，则测试将通过。

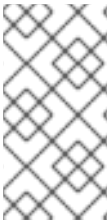
```
$ ipa hbactest --user=user1 --host=example.com --service=sudo --rules=rule1 --rules=rule2
--------------------
Access granted: True
--------------------
  Matched rules: rule2
  Not matched rules: rule1
```

在输出中：

- 匹配的规则 列出了允许成功访问的规则。
- 不匹配规则 会列出阻止访问的规则。

31.2.3. 禁用 HBAC 规则

禁用 HBAC 规则将停用该规则，但不会删除该规则。如果禁用 HBAC 规则，您可以稍后重新启用它。



注意

例如，在首次配置自定义 HBAC 规则后，禁用 HBAC 规则非常有用。为确保新配置不会被默认的 allow_all HBAC 规则覆盖，您必须禁用 allow_all。

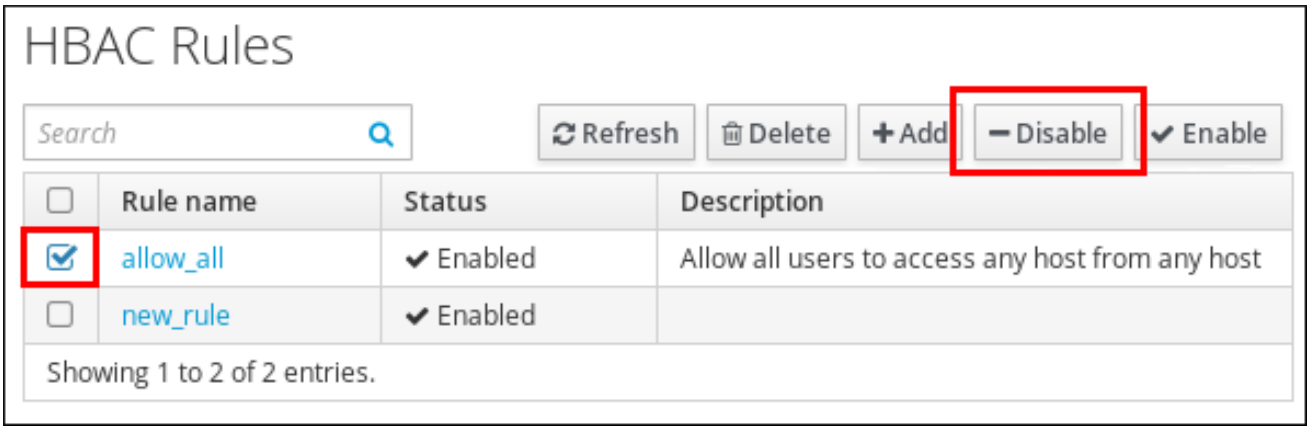
要禁用 HBAC 规则，您可以使用：

- IdM Web UI（请参阅 "Web UI：禁用 HBAC 规则"一节）
- 命令行（请参见 "命令行：禁用 HBAC 规则"一节）

Web UI：禁用 HBAC 规则

1. 选择 基于策略的访问控制 → HBAC 规则。
2. 选择您要禁用的 HBAC 规则，然后单击 Disable。

图 31.6. 禁用 allow_all HBAC 规则



命令行：禁用 HBAC 规则

使用 ipa hbacrule-disable 命令。例如，禁用 allow_all 规则：

```
$ ipa hbacrule-disable allow_all
-----------------------------
Disabled HBAC rule "allow_all"
-----------------------------
```

31.3. 为自定义 HBAC 服务添加 HBAC SERVICE ENTRIES

默认情况下，只有最常见的服务和服务组是为 HBAC 规则配置。但是，您还可以将任何其他可插拔验证模块(PAM)服务配置为 HBAC 服务。这可让您在 HBAC 规则中定义自定义 PAM 服务。

注意

将服务添加为 HBAC 服务与向域添加服务不同。在域中添加服务（在 第 16.1 节 “添加和编辑服务条目和密钥选项卡”中描述）会使服务成为可识别的资源供域中其他资源使用，但它不允许您在 HBAC 规则中使用该服务。

要添加 HBAC 服务条目，您可以使用：

- IdM Web UI（请参阅 “Web UI：添加 HBAC 服务条目”一节）
- 命令行（请参见 “命令行：添加 HBAC 服务条目”一节）

Web UI：添加 HBAC 服务条目

1. 选择 Policy → Host-Based Access → HBAC Services。
2. 点 Add 添加 HBAC 服务条目。
3. 输入服务的名称，然后单击 Add。

命令行：添加 HBAC 服务条目

使用 ipa hbacsvc-add 命令。例如，要为 tftp 服务添加一个条目：

```
$ ipa hbacsvc-add tftp
-------------------------
Added HBAC service "tftp"
-------------------------
  Service name: tftp
```

31.4. 添加 HBAC 服务组

HBAC 服务组可以简化 HBAC 规则管理：您可以添加整个服务组，而不是将个别服务添加到 HBAC

规则中。

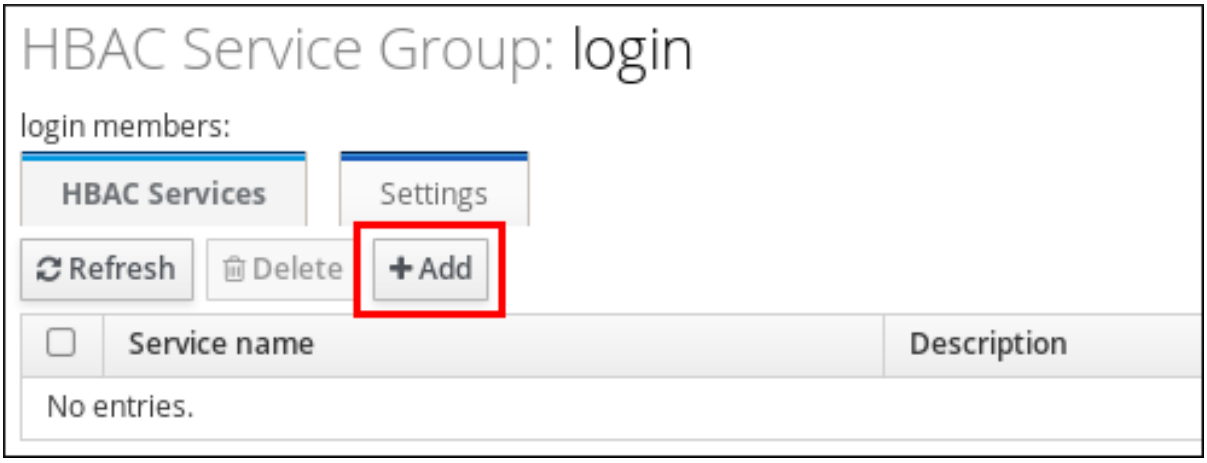
要添加 HBAC 服务组，您可以使用：

- IdM Web UI（请参阅 “Web UI：添加 HBAC 服务组”一节）
- 命令行（请参见 “命令行：添加 HBAC 服务组”一节）

Web UI：添加 HBAC 服务组

1. 选择 Policy → Host-Based Access → HBAC Service Groups。
2. 点 Add 添加 HBAC 服务组。
3. 输入服务组的名称，然后点 Add 和 Edit。
4. 在服务组配置页面中，点 Add 将 HBAC 服务添加为组的成员。

图 31.7. 将 HBAC 服务添加到 HBAC 服务组



命令行：添加 HBAC 服务组

1. 使用 ipa hbacsvcgroup-add 命令添加一个 HBAC 服务组。例如，要添加名为 login 的组：

```
$ ipa hbacsvcgroup-add
Service group name: login
```

```
-------------------------------
Added HBAC service group "login"
-------------------------------
  Service group name: login
```

2. **使用 ipa hbacsvcgroup-add-member 命令将 HBAC 服务添加为组的成员。例如，要将 sshd 服务添加到 login 组中：**

```
$ ipa hbacsvcgroup-add-member
Service group name: login
[member HBAC service]: sshd
  Service group name: login
  Member HBAC service: sshd
-------------------------
Number of members added 1
-------------------------
```

第 32 章 定义 SELINUX 用户映射

安全增强型 Linux(SELinux)可设置有关用户可以访问进程、文件、目录和系统设置的规则。系统管理员和系统应用都可以定义 安全上下文，以限制或允许来自其他应用的访问。

作为在身份管理域中定义集中式安全策略的一部分，身份管理提供了一种将 IdM 用户映射到现有 SELinux 用户上下文的方法，并根据主机的 SELinux 策略授予或限制对 IdM 域中客户端和服务的访问权限。

32.1. 关于身份管理、SELINUX 和映射用户

身份管理不会在系统上创建或修改 SELinux 上下文。相反，它使用与目标主机上现有上下文匹配的字符串，作为将域中 IdM 用户映射到系统中的 SELinux 用户的基础。

增强安全性的 Linux 为进程如何与系统上的其他资源交互定义了内核级、强制访问控制。根据系统上进程的预期行为及其安全影响，将设置称为策略的特定规则。这与更高级别的自主访问控制相反，它们主要关注文件所有权和用户身份。系统上的每个资源都被分配一个上下文。资源包括用户、应用程序、文件和进程。

系统用户与 SELinux 角色 关联。该角色同时分配了多层安全上下文(MLS)和多类别安全上下文(MCS)。MLS 和 MCS 上下文限制用户，以便他们只能访问系统上的某些进程、文件和操作。

要获得可用 SELinux 用户的完整列表：

```
[root@server1 ~]# semanage user -l

                Labelling  MLS/       MLS/
SELinux User    Prefix     MCS Level  MCS Range                SELinux Roles

guest_u         user       s0         s0                       guest_r
root            user       s0         s0-s0:c0.c1023           staff_r sysadm_r system_r unconfined_r
staff_u         user       s0         s0-s0:c0.c1023           staff_r sysadm_r system_r unconfined_r
sysadm_u        user       s0         s0-s0:c0.c1023           sysadm_r
system_u        user       s0         s0-s0:c0.c1023           system_r unconfined_r
unconfined_u    user       s0         s0-s0:c0.c1023           system_r unconfined_r
user_u          user       s0         s0                       user_r
xguest_u        user       s0         s0                       xguest_r
```

有关 Red Hat Enterprise Linux 中的 SELinux 的详情，请查看 Red Hat Enterprise Linux 7

SELinux 用户和管理员指南。

SELinux 用户和策略在系统级别而非网络级别发挥作用。这意味着 SELinux 用户是在每个系统上独立配置的。尽管在很多情况下这是可以接受的，因为 SELinux 具有常见的系统用户和 SELinux 感知服务定义自己的策略，因此当远程用户和系统访问本地资源时，会导致问题。可以在不知晓远程用户和角色的实际 SELinux 用户和角色的情况下为远程用户和服务分配默认的 guest 上下文。

身份管理可以将身份域与本地 SELinux 服务集成。身份管理可以根据主机将 IdM 用户映射到 每个主机配置的 SELinux 角色，或者 基于 HBAC 规则。映射 SELinux 和 IdM 用户改进了用户管理：

- 远程用户可以根据其 IdM 组分配授予适当的 SELinux 用户上下文。这也允许管理员一致地将相同的策略应用到同一用户，而无需创建本地帐户或重新配置 SELinux。
- 与用户关联的 SELinux 上下文是集中的。
- 可以通过基于 IdM 主机的访问控制规则等设置来计划以及与域范围内的安全策略相关的 SELinux 策略。
- 管理员可以在环境范围内实现可见性，并控制 SELinux 中如何分配用户和系统。

SELinux 用户映射定义了三个部分存在的两个独立关系：系统的 SELinux 用户、IdM 用户和 IdM 主机。首先，SELinux 用户映射定义了 SELinux 用户和 IdM 主机（本地或远程系统）之间的关系。其次，它定义了 SELinux 用户和 IdM 用户之间的关系。

此安排允许管理员为同一 IdM 用户设置不同的 SELinux 用户，具体取决于他们正在访问的主机。

SELinux 映射规则的核心是 SELinux 系统用户。每个映射首先与 SELinux 用户关联。可用于映射的 SELinux 用户在 IdM 服务器中配置，因此有一个中央和通用列表。这样，IdM 定义一组它知道的 SELinux 用户，并可在登录时与 IdM 用户关联。默认情况下，它们分别为：

- unconfined_u （也用作 IdM 用户的默认）
- guest_u

- xguest_u
- user_u
- staff_u

但是，可以修改这个默认列表，以及从中央 IdM SELinux 用户列表添加或删除任何 原生 SELinux 用户（请参阅 第 32.1 节 “关于身份管理、SELinux 和映射用户”）。

在 IdM 服务器配置中，每个 SELinux 用户都只配置其用户名，也配置其 MLS 和 MCS 范围，SELinux_user:MLS[:MCS]。IPA 服务器使用这种格式在配置映射时标识 SELinux 用户。

IdM 用户和主机配置非常灵活。可以明确且单独分配用户和主机到 SELinux 用户映射，或者可以将用户组或主机组显式分配到该映射。

您还可以将 SELinux 映射规则与基于主机的访问控制规则相关联，以简化管理，以避免在两个位置重复相同的规则，并使规则保持同步。只要基于主机的访问控制规则定义了用户和主机，就可以将它用于 SELinux 用户映射。基于主机的访问控制规则（在 第 31 章 配置基于主机的访问控制中描述）有助于将 SELinux 用户映射与 IdM 中的其他访问控制集成，并帮助限制或允许远程用户的基于主机的用户访问，以及定义本地安全上下文。

注意

如果基于主机的访问控制规则与 SELinux 用户映射关联，则在从 SELinux 用户映射配置中删除之前，无法删除基于主机的访问控制规则。

SELinux 用户映射与系统安全服务守护进程(SSSD)和 pam_selinux 模块一起工作。当远程用户尝试登录机器时，SSSD 会检查其 IdM 身份提供程序以收集用户信息，包括任何 SELinux 映射。然后 PAM 模块处理用户，并为它分配相应的 SELinux 用户上下文。SSSD 缓存允许映射离线工作。

32.2. 配置 SELINUX 用户映射顺序和默认值

SELinux 用户映射是客户端 SELinux 用户和 IdM 用户之间的关联。

可用的 SELinux 用户映射顺序是 IdM 服务器配置的一部分。SELinux 用户映射顺序是 SELinux 用户的列表，按照从最多到最少限制的顺序的顺序。SELinux 用户条目本身具有以下格式：

```
SELinux_user:MLS[:MCS]
```

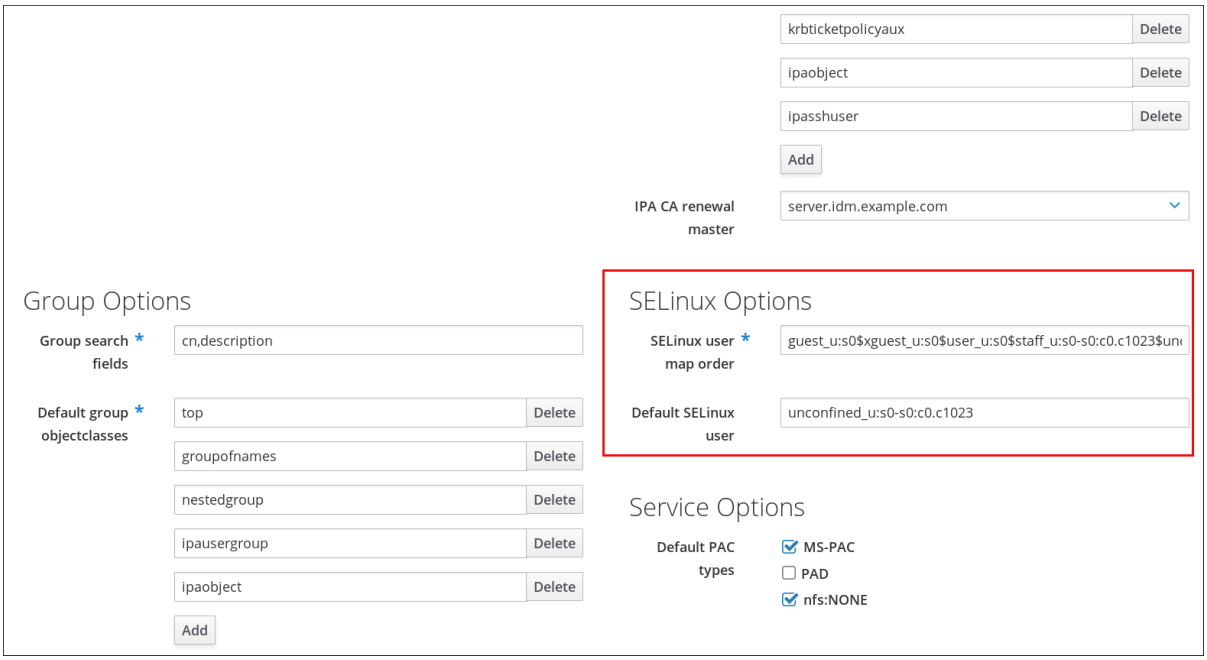
单个用户条目用美元符号($)分隔。

由于用户条目不需要具有 SELinux 映射，因此可能会取消映射许多条目。IdM 服务器配置会设置默认 SELinux 用户，这是整个 SELinux 映射列表中的一个用户，用于未映射的 IdM 用户条目。这样，即使未映射的 IdM 用户也具有正常运行的 SELinux 上下文。未映射 IdM 用户条目的默认 SELinux 用户是 unconfined_u，这是 Red Hat Enterprise Linux 上系统用户的默认 SELinux 用户。

此配置定义可用系统 SELinux 用户的映射顺序。这不定义任何 IdM 用户 SELinux 策略。必须定义 IdM 用户 - SELinux 用户映射，然后用户被添加到映射中。详情请查看 第 32.3 节 “映射 SELinux 用户和 IdM 用户”。

32.2.1. 在 Web UI 中

1. 在顶部菜单中，点 IPA Server main 选项卡和 Configuration 子选项卡。
2. 滚动到服务器配置区域列表的底部，以 SELINUX OPTIONS。
3. 编辑 SELinux 用户配置、SELinux 用户映射顺序、默认 SELinux 用户 或两者。



4. 单击页面顶部的 **Update** 链接，以保存更改。

32.2.2. 在 CLI 中

要查看 SELinux 用户列表，请在 IdM 服务器配置中设置，这些用户可以被映射：

```
[user1]@server ~]$ ipa config-show
...
SELinux user map order: guest_u:s0$xguest_u:s0$user_u:s0$staff_u:s0-
s0:c0.c1023$unconfined_u:s0-s0:c0.c1023
Default SELinux user: unconfined_u:s0-s0:c0.c1023
```

要编辑 SELinux 用户设置，请使用 **config-mod** 命令：

例 32.1. SELinux 用户列表

要编辑可用于映射的 SELinux 用户列表，请使用 **--ipaselinuxusermaporder** 选项。该列表将 SELinux 用户从最多排序到限制最低级别，例如：

```
[user1@server ~]$ ipa config-mod --ipaselinuxusermaporder="unconfined_u:s0-
s0:c0.c1023$guest_u:s0$xguest_u:s0$user_u:s0-s0:c0.c1023$staff_u:s0-s0:c0.c1023"
```

注意

用于取消映射条目的默认 SELinux 用户必须包含在用户映射列表中，否则编辑操作失败。类似地，如果编辑了默认设置，则必须将其更改为 SELinux 映射列表中的用户，或者必须首先更新映射列表。

例 32.2. 默认 SELinux 用户

IdM 用户不需要将特定的 SELinux 用户映射到其帐户。但是，本地系统仍然检查用于 IdM 用户帐户的 SELinux 用户的 IdM 条目。

要修改默认 SELinux 用户，请使用 **--ipaselinuxusermapdefault** 选项。例如：

```
[user1@server ~]$ ipa config-mod --ipaselinuxusermapdefault="guest_u:s0"
```

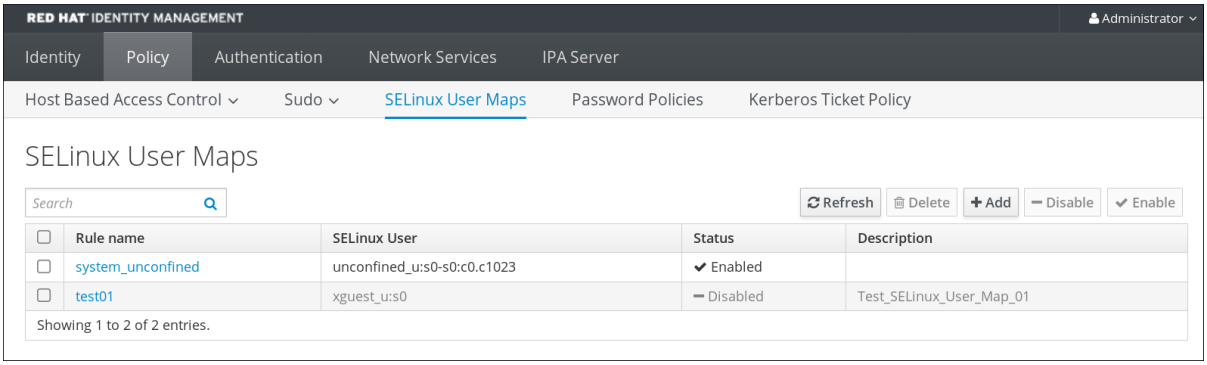
32.3. 映射 SELINUX 用户和 IDM 用户

SELinux 映射将本地系统中的 SELinux 用户上下文与域中的 IdM 用户或用户相关联。SELinux 映射有三个部分：SELinux 用户上下文和 IdM 用户主机对。IdM 用户主机对可以通过两种方式之一定义：可以为显式用户或主机组、显式主机或主机组设置它；或使用基于主机的访问控制规则来定义。

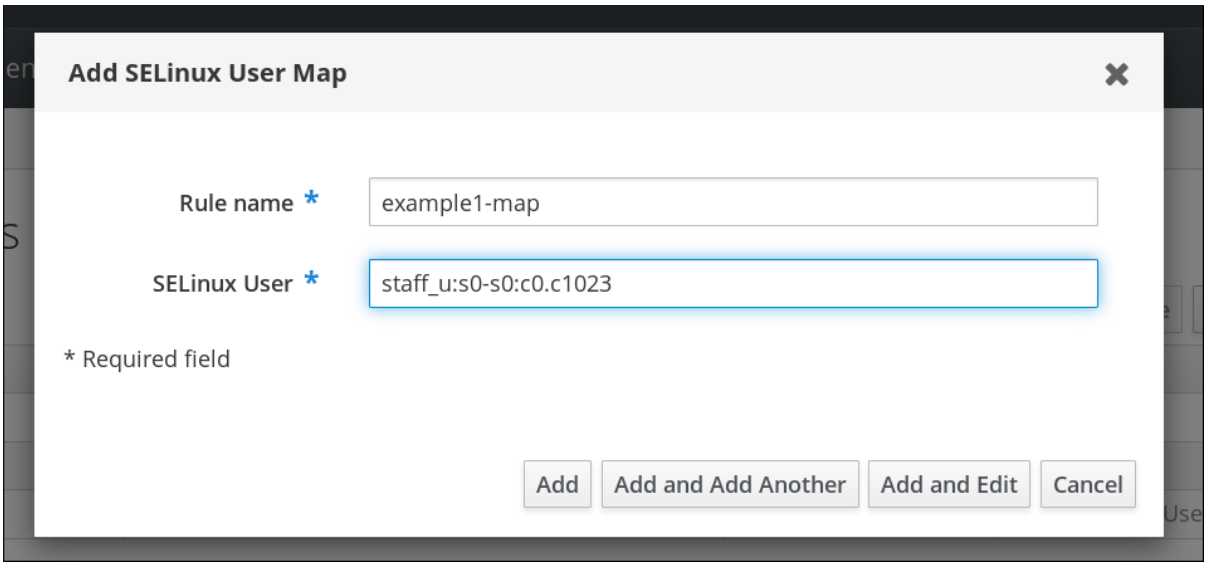
32.3.1. 在 Web UI 中

1. 在顶部菜单中，点 Policy main 选项卡和 SELinux User Mappings 子选项卡。

2. 在映射列表中，点 Add 按钮创建新映射。



3. 输入映射的名称和 SELinux 用户。SELinux 用户的格式必须与如何在 IdM 服务器配置中显示的格式相同。SELinux 用户的格式为 SELinux_user:MLS[:MCS]。



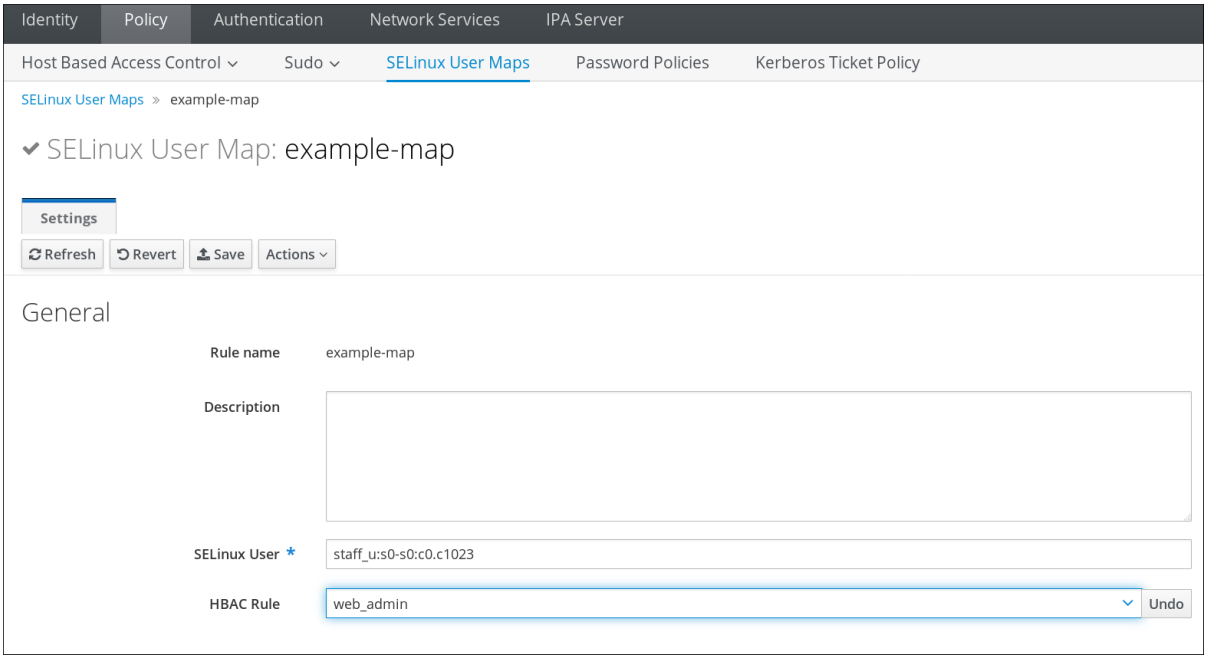
4. 点 Add and Edit 添加 IdM 用户信息。

5.

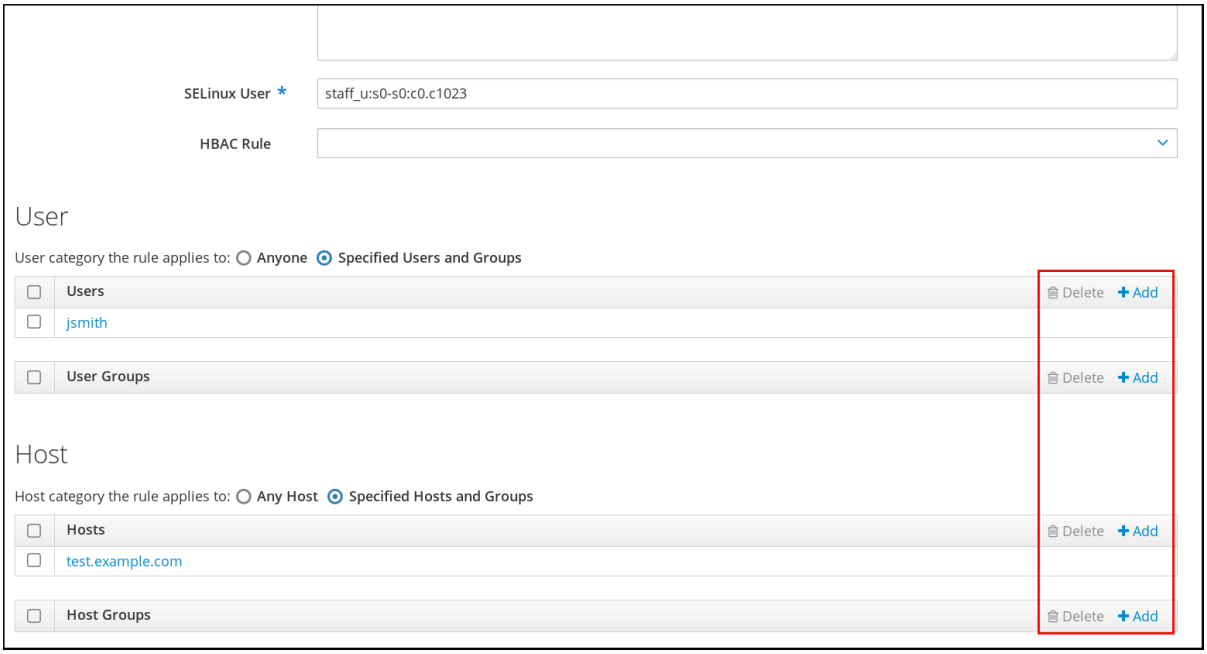
要设置基于主机的访问控制规则，请从配置 常规 区域中的下拉菜单中选择规则。使用基于主机的访问控制规则还引入了对远程用户可用于访问目标计算机的主机的访问控制。只能分配基于主机的访问控制规则。

注意

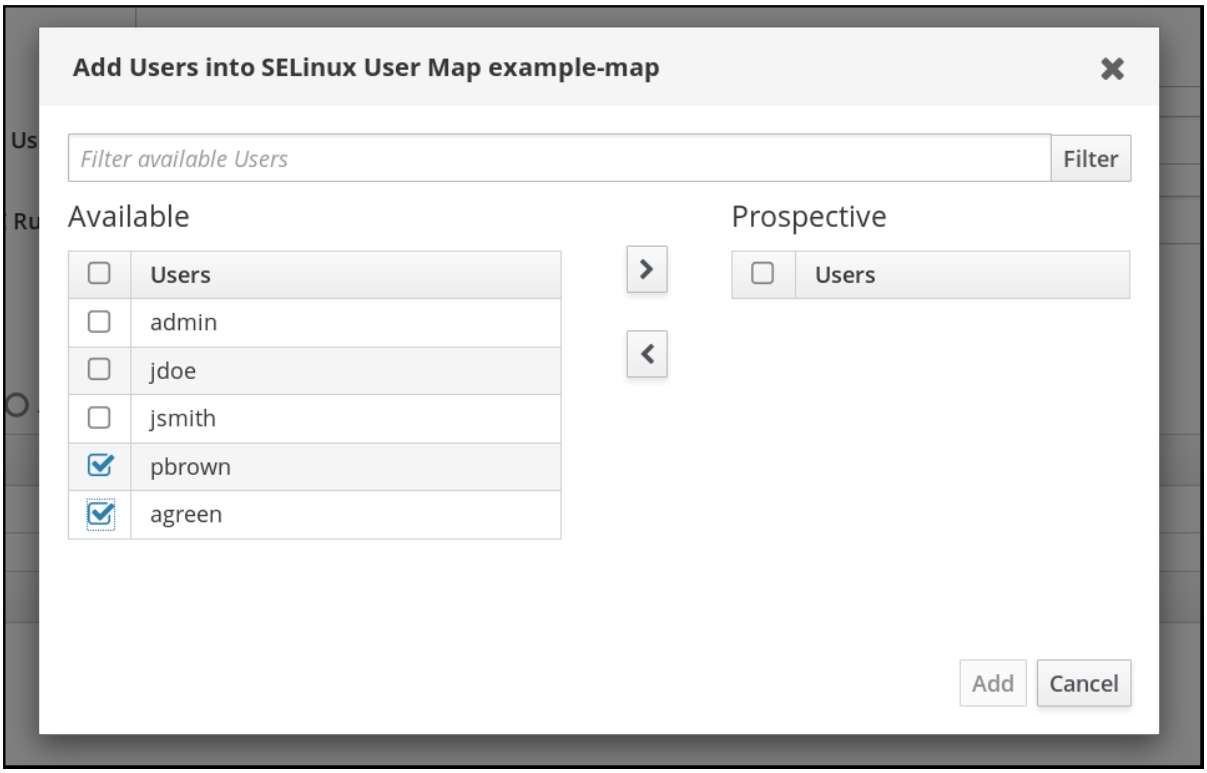
基于主机的访问控制规则必须包含用户和主机，而不只是服务。



*或者，向下滚动 **Users** 和 **Hosts** 区域，然后单击 **Add** 链接，以将用户、用户组、主机或主机组分配给 **SELinux** 映射。*



选择左侧的用户（或主机或主机组），点击右箭头按钮(>>)将它们移到 Prospective 列中，然后点击 Add 按钮将它们添加到规则中。



注意

只能使用一个选项：可以指定基于主机的访问控制规则，或者手动设置用户和主机。这两个选项不能同时使用。

6. 点顶部的 Update 链接，将更改保存到 SELinux 用户映射。

32.3.2. 在 CLI 中

SELinux 映射规则有三个基本部分：

- SELinux 用户： --selinuxuser
- 与 SELinux 用户关联的用户或用户组： --users 或 --groups
- 与 SELinux 用户关联的主机或主机组： --hosts 或 --hostgroups

- 或者，基于主机的访问控制规则指定其中的主机和用户： **--hbacrule**

可以使用 **selinuxusermap-add** 命令一次创建有所有信息的规则。在分别使用 **selinuxusermap-add-user** 和 **selinuxusermap-add-host** 命令创建后，可以将用户和主机添加到规则中。

例 32.3. 创建新的 SELinux 映射

--selinuxuser 值必须是 SELinux 用户名，就像它出现在 IdM 服务器配置中一样。SELinux 用户的格式为 **SELinux_user:MLS[:MCS]**。

必须指定用户或用户组以及主机或主机组才能使 SELinux 映射生效。**user**、**host** 和 **group** 选项可以多次使用，也可以使用逗号分隔的花括号内列出一次，如 **--option={val1,val2,val3}**。

```
[user1@server ~]$ ipa selinuxusermap-add --selinuxuser="xguest_u:s0" selinux1
[user1@server ~]$ ipa selinuxusermap-add-user --users=user1 --users=user2 --users=user3
selinux1
[user1@server ~]$ ipa selinuxusermap-add-host --hosts=server.example.com --
hosts=test.example.com selinux1
```

例 32.4. 使用基于主机的访问控制规则创建 SELinux map

hbacrule 值标识用于映射的基于主机的访问控制规则。使用基于主机的访问控制规则可介绍远程用户可用于访问目标计算机的主机的访问控制，以及在远程用户登录目标计算机后应用 SELinux 上下文。

访问控制规则必须相应地指定用户和主机，以便 SELinux 映射可以构建 SELinux 用户、IdM 用户和主机 triple。

只能指定基于主机的访问控制规则。

```
[user1@server ~]$ ipa selinuxusermap-add --hbacrule=webserver --selinuxuser="xguest_u:s0"
selinux1
```

第 31 章 配置基于主机的访问控制 中描述了基于主机的访问控制规则。

例 32.5. 在 SELinux 映射中添加用户

可以将用户和主机添加到现有的映射中。这可以通过特定命令(selinuxusermap-add-user 或 selinuxusermap-add-host)完成。

```
[user1@server ~]$ ipa selinuxusermap-add-user --users=user1 selinux1
```

如果 selinuxusermap-mod 命令与 --hbacrule 选项一起使用，则新的 SELinux 映射会覆盖以前的 SELinux 映射。

例 32.6. 从 SELinux 映射中删除用户

通过使用 selinuxusermap-remove-host 或 selinuxusermap-remove-user 命令从 SELinux 映射中删除特定用户或主机。例如：

```
[user1@server ~]$ ipa selinuxusermap-remove-user --users=user2 selinux1
```

部分 VII. 管理：管理网络服务

本节讨论如何管理与 身份管理 集成的 域名服务 (DNS)，以及如何使用 自动挂载 在多个系统中管理、组织和访问目录。

第 33 章 管理 DNS

无需集成 DNS 服务即可安装身份管理服务器，以便其使用外部 DNS 服务或配置 DNS。详情请查看 第 2.3 节 “安装 IdM 服务器：简介” 和 第 2.3.1 节 “确定使用集成 DNS”。

如果在域中配置 DNS 服务，IdM 为管理员提供了极大的灵活性和 DNS 设置控制能力。例如，可以使用原生 IdM 工具管理域的 DNS 条目，如主机条目、位置或记录，客户端也可以动态更新自己的 DNS 记录。

BIND 版本 9.9 版大多数文档资料和教程也适用于 IdM DNS，因为大多数配置选项在 BIND 和 IdM 中的工作方式相同。本章主要侧重于 BIND 和 IdM 之间的显著差异。

33.1. 身份管理中的 BIND

IdM 将 BIND DNS 服务器版本 9.9 与用于数据复制的 LDAP 数据库以及使用 GSS-TSIG 协议的 DNS 更新签名的 Kerberos 集成 [3].这可以使用 IdM 工具进行方便的 DNS 管理，同时提高弹性，因为 IdM 集成的 DNS 服务器支持多主控机操作，允许所有 IdM 集成的 DNS 服务器接受来自客户端的 DNS 更新，且无单点故障。

默认 IdM DNS 配置适用于无法从公共互联网访问的内部网络。如果可以从公共互联网访问 IdM DNS 服务器，红帽建议应用适用于 BIND 服务的常规强化功能，如《Red Hat Enterprise Linux 网络指南》中所述。

注意

无法在 chroot 环境中运行 BIND 与 IdM 集成。

Red Hat Enterprise Linux 中的 DNS (域名系统)协议的 BIND (Berkeley 互联网 名称域)实现包括 命名的 DNS 服务器。named-pkcs11 是构建了对 PKCS the 加密标准的原生支持的 BIND DNS 服务器版本。

与 IdM 集成的 BIND 使用 bind-dyndb-ldap 插件与目录服务器通信。IdM 为 BIND 服务在 /etc/named.conf 文件中创建一个 dynamic-db 配置部分，它为 BIND named-pkcs11 服务配置 bind-dyndb-ldap 插件。

标准 BIND 和 IdM DNS 之间最显著的区别是 IdM 将所有 DNS 信息存储为 LDAP 条目。每个域名都以 LDAP 条目表示，每个资源记录存储为 LDAP 条目的 LDAP 属性。例如，以下 client1.example.com. 域

名包含三个 A 记录和一个 AAAA 记录：

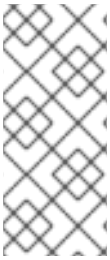
```
dn: idnsname=client1,idnsname=example.com.,cn=dns,dc=idm,dc=example,dc=com
objectclass: top
objectclass: idnsrecord
idnsname: client1
Arecord: 192.0.2.1
Arecord: 192.0.2.2
Arecord: 192.0.2.3
AAAArecord: 2001:DB8::ABCD
```

重要

要编辑 DNS 数据或 BIND 配置，请始终使用本章中描述的 IdM 工具。

33.2. 支持的 DNS 区域类型

IdM 支持两种 DNS 区类型： master 和 forward zone。



注意

本指南使用区域类型的 BIND 术语，这与用于 Microsoft Windows DNS 的术语不同。BIND 中的主区域与 正向查找区域和 Microsoft Windows DNS 中的 反向查找区域 的作用相同。BIND 中的转发区域与 Microsoft Windows DNS 中的 条件转发 器相同。

Master DNS 区域

主 DNS 区域包含权威 DNS 数据，可以接受动态 DNS 更新。此行为等同于标准 BIND 配置中的类型 master 设置。master zone 使用 ipa dnszone the 命令进行管理。

根据标准的 DNS 规则，每个 master 区域都必须包含 SOA 和 NS 记录。IdM 在创建 DNS 区域时自动生成这些记录，但必须手动将 NS 记录复制到父区域，以创建适当的委托。

根据标准的 BIND 行为，为 master 区域指定的转发配置仅影响服务器不是权威名称的查询。

例 33.1. DNS 转发方案示例

IdM 服务器包含 test.example. master 区域。此区域包含 sub.test.example. 名称的 NS 委派记录。此外，test.example. 区域使用 192.0.2.254 转发器 IP 地址进行配置。

> 查询名称 不存在.test.example. 的客户端会收到 NXDomain 回答，并且不会发生转发，因为 IdM 服务器对此名称具有权威。
>
> 另一方面，查询 sub.test.example. 名称将转发到配置的转发器 192.0.2.254，因为 IdM 服务器对此名称没有权威。

转发 DNS 区域

转发 DNS 区域不包含任何权威数据。所有属于转发 DNS 区域的名称查询都转发到指定的转发器。此行为等同于标准 BIND 配置中的 type forward 设置。转发区使用 ipa dnsforwardzone114 命令进行管理。

33.3. DNS 配置优先级

可以在三个不同的级别上配置许多 DNS 配置选项：

特定于区的配置

IdM 中定义的特定区的配置级别最高。特定于区域的配置使用 ipa dnszone the 和 ipa dnsforwardzone the 命令进行管理。

全局 DNS 配置

如果没有定义特定于区的配置，IdM 将使用存储在 LDAP 中的全局 DNS 配置。全局 DNS 配置使用 ipa dnsconfig114 命令进行管理。全局 DNS 配置中定义的设置适用于所有 IdM DNS 服务器。

配置 /etc/named.conf

在每个 IdM DNS 服务器的 /etc/named.conf 文件中定义的配置具有最低优先级。它特定于每个服务器，必须手动编辑。

/etc/named.conf 文件通常仅用于指定 DNS 转发到本地 DNS 缓存；其他选项使用命令管理上述特定于区域和全局 DNS 配置。

可以在多个级别同时配置 DNS 选项。在这种情况下，优先级最高的配置优先于较低级别中定义的配置。

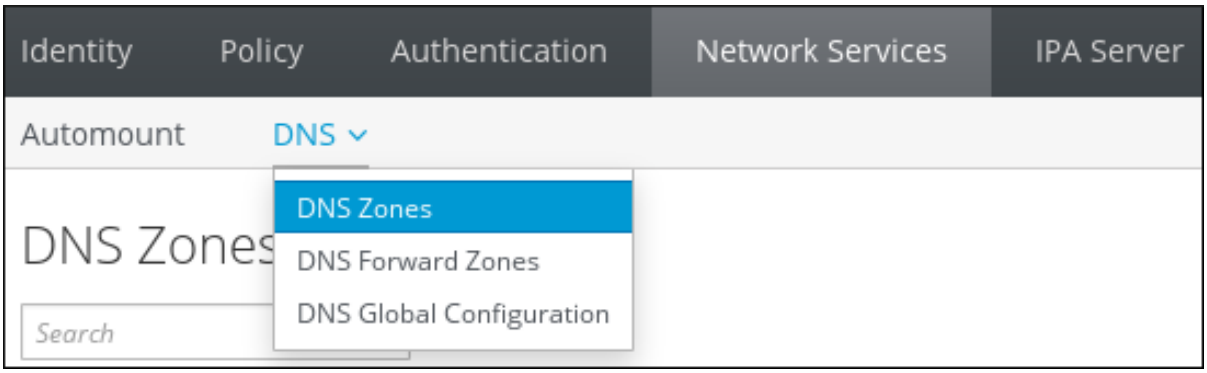
33.4. 管理主 DNS 区域

33.4.1. 添加和删除主 DNS 区域

在 Web UI 中添加主 DNS 区域

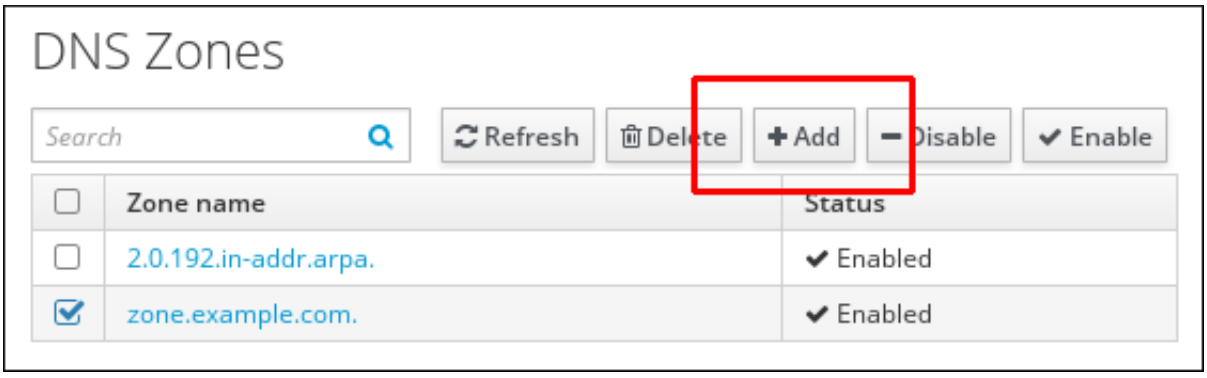
1. 打开 Network Services 选项卡，然后选择 DNS 子选项卡，后跟 DNS Zones 部分。

图 33.1. 管理 DNS 主区



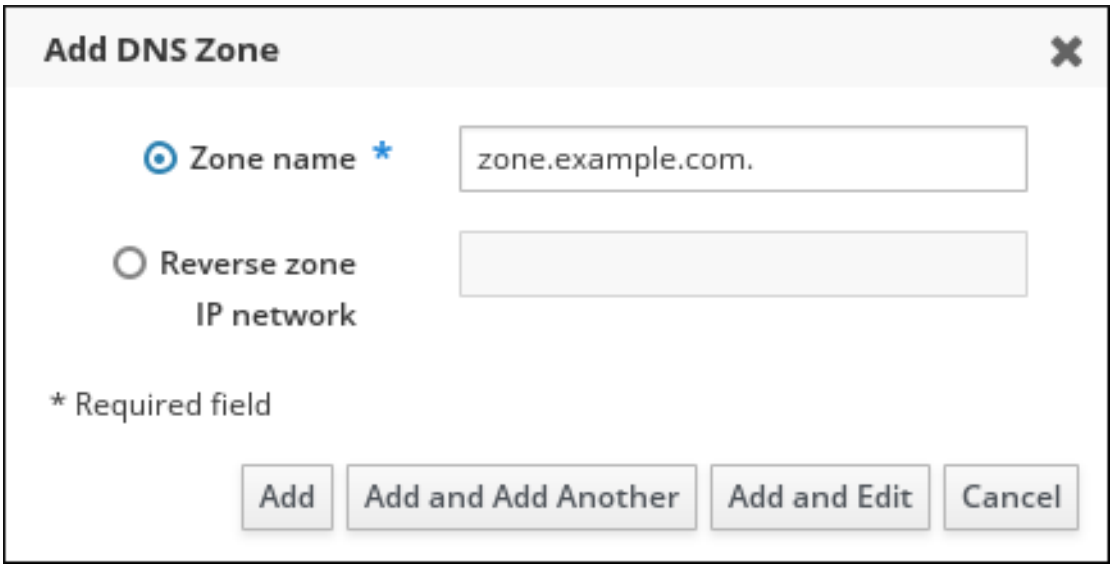
2. 要添加新的 master 区域，请单击所有区域列表顶部的 Add。

图 33.2. 添加主 DNS 区域



3. 提供区域名称，然后单击 Add。

图 33.3. 输入新的 master 区域



从命令行添加主 DNS 区域

ipa dnszone-add 命令向 DNS 域添加新区域。添加新区要求您指定新子域的名称。您可以使用以下命令直接传递子域名称：

```
$ ipa dnszone-add newserver.example.com
```

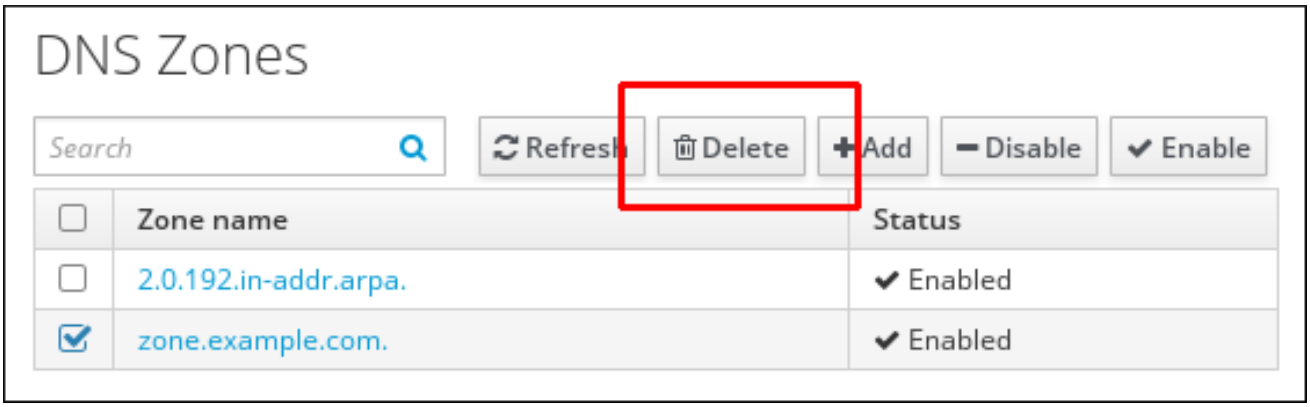
如果没有将名称传递给 ipa dnszone-add，脚本会自动提示它。

ipa dnszone-add 命令也接受各种命令行选项。如需这些选项的完整列表，请运行 ipa dnszone-add --help 命令。

删除主 DNS 区域

要在 Web UI 中删除 master DNS 区域，在所有区列表中，按区域名称选择复选框，然后单击 Delete。

图 33.4. 删除主 DNS 区域



要从命令行删除主 DNS 区域，请使用 ipa dnszone-del 命令。例如：

```
$ ipa dnszone-del server.example.com
```

33.4.2. 为主 DNS 区域添加额外的配置

IdM 使用特定默认配置创建新区域，如刷新周期、传输设置或缓存设置。

DNS 区域配置属性

可用的区设置列在 表 33.1 “zone 属性” 中。除了为区域设置实际信息外，设置还定义 DNS 服务器如何处理 授权起始 (SOA)记录条目，以及如何从 DNS 名称服务器更新其记录。

表 33.1. zone 属性

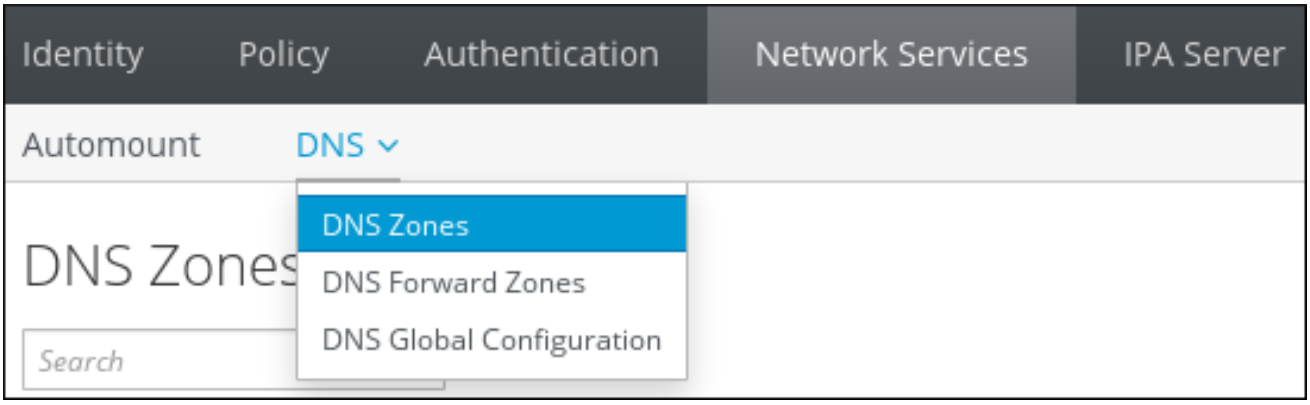
属性	命令行选项	描述
权威名称服务器	**--name-server**	设置主 DNS 名称服务器的域名，也称为 SOA MNAME。 默认情况下，每个 IdM 服务器在 SOA MNAME 字段中公告其自身。因此，使用 **--name-server** 存储在 LDAP 中的值将被忽略。
管理员电子邮件地址	**--admin-email**	设置要用于区域管理员的电子邮件地址。这默认为主机上的 root 帐户。
SOA 串行	**--serial**	在 SOA 记录中设置序列号。请注意，IdM 会自动设置版本号，用户不应该修改它。
SOA 刷新	**--refresh**	设置辅助 DNS 服务器在从主 DNS 服务器请求更新前等待的时间间隔（以秒为单位）。
SOA 重试	**--retry**	在重试失败的刷新操作前将等待的时间（以秒为单位）。
SOA 过期	**--expire**	设置辅助 DNS 服务器在结束操作尝试前尝试执行刷新更新的时间，以秒为单位。
SOA 最小值	**--minimum**	根据 RFC 2308，为负缓存设置生存时间(TTL)值（以秒为单位）。
SOA 时间生存	**--ttl**	为区域 apex 的记录设置 TTL（以秒为单位）。例如，**在区域 example.com** 中，配置了名称 **example.com** 下的所有记录（A、NS 或 SOA），但不影响其他域名，如 **test.example.com**。

属性	命令行选项	描述
默认生存时间	**--default-ttl**	为区域中的所有值（以秒为单位）设置默认生存时间(TTL)值（以秒为单位）。更改生效后，需要在所有 IdM DNS 服务器上重新启动 **named-pkcs11** 服务。
BIND 更新策略	**--update-policy**	设置允许 DNS 区域中的客户端的权限。 如需有关 更新策略语法的更多信息，请参阅 *BIND 9 管理员参考手册中*的动态 更新策略。
动态更新	**--dynamic-update**=TRUE|FALSE	为客户端启用 DNS 记录的动态更新。 请注意，如果设置为 false，IdM 客户端计算机将无法添加或更新其 IP 地址。请参阅 第 33.5.1 节 “启用动态 DNS 更新” 了解更多信息。
允许传输	**--allow-transfer**=*string*	提供允许传输给定区的 IP 地址或网络名称列表，用分号(;)分隔。 默认情况下禁用区域传送。默认的 **--allow-transfer** 值为 **none**。
允许查询	**--allow-query**	提供允许发出 DNS 查询的 IP 地址或网络名称列表，用分号(;)分隔。
允许 PTR 同步	**--allow-sync-ptr**=1|0	设置区域的 A 或 AAAA 记录（正向记录）是否将自动与 PTR（反向）记录同步。
区域转发器	**--forwarder**=*IP_address*	指定专门为 DNS 区域配置的转发器。这与 IdM 域中使用的任何全局转发器分开。 要指定多个转发器，请多次使用 选项。
forward 策略	**--forward-policy**=none|only|first	指定 forward 策略。有关支持的策略的详情，请参考 “forward 策略”一节

在 Web UI 中编辑区域配置

要从 Web UI 管理 DNS master 区域，请打开 网络服务 选项卡，然后选择 DNS 子选项卡，后跟 DNS Zones 部分。

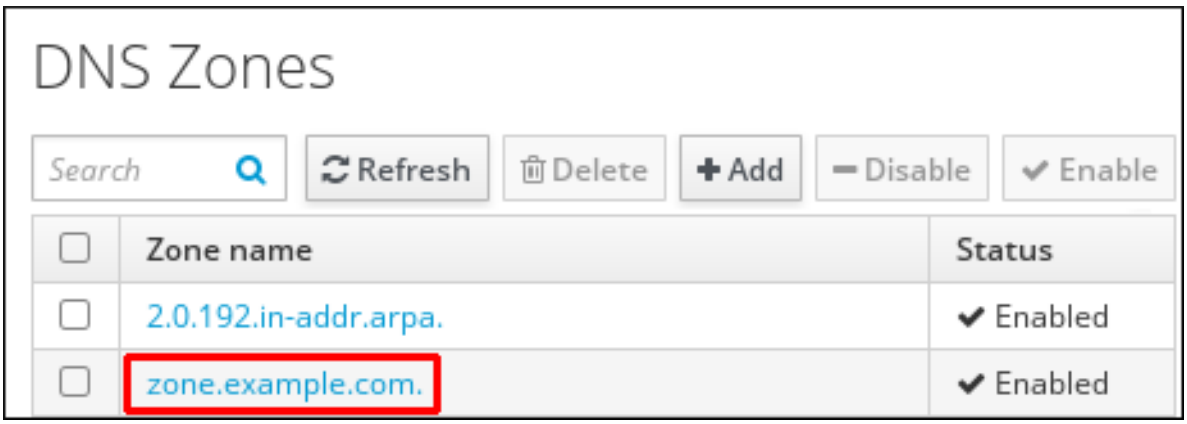
图 33.5. DNS 主区管理



要在 DNS Zones 部分编辑现有的 master 区域：

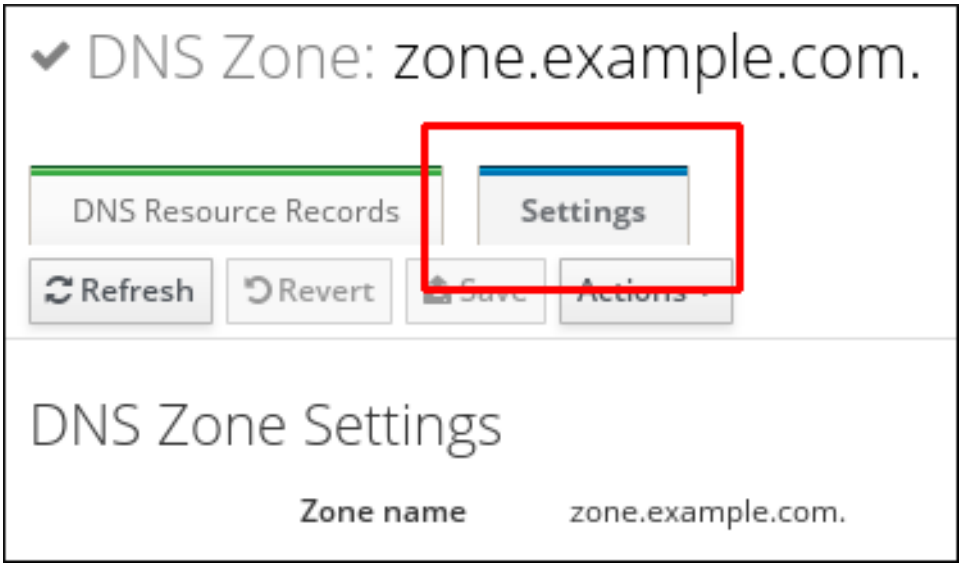
1. *单击所有区域列表中的区域名称，以打开 DNS 区域页面。*

图 33.6. 编辑 master zone



2. *单击 Settings，然后根据需要更改区域配置。*

图 33.7. Master zone Edit 页面中的 Settings 选项卡



有关可用设置的详情，请参考 表 33.1 “zone 属性”。

3. *单击 Save 以确认新配置。*

注意

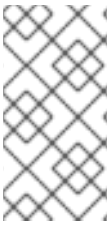
如果您要将所有区域的默认时间更改为 live(TTL)，在所有 IdM DNS 服务器上重新启动 named-pkcs11 服务，以使更改生效。所有其他设置都会立即自动激活。

从命令行编辑区域配置

要从命令行修改现有 master DNS 区域，请使用 ipa dnszone-mod 命令。有关可用设置的详情，请参考 表 33.1 “zone 属性”。

如果 DNS 区域条目中不存在属性，ipa dnszone-mod 命令会添加属性。如果 属性存在，则 命令使用指定的值覆盖当前值。

有关 ipa dnszone-mod 及其选项的详细信息，请运行 ipa dnszone-mod --help 命令。



注意

如果您要将所有区域的默认时间更改为 live(TTL)，在所有 IdM DNS 服务器上重新启动 named-pkcs11 服务，以使更改生效。所有其他设置都会立即自动激活。

33.4.3. 启用区传输

名称服务器维护区域的权威数据；对区域进行的更改必须在 DNS 域的名称服务器之间发送和分发。区域传送 将所有资源记录从一个名称服务器复制到另一个名称服务器。

IdM 支持根据 RFC 5936 (AXFR)和 RFC 1995(IXFR)标准进行区域传输。

重要

IdM 集成的 DNS 是多主控机。IdM 区域中的 SOA 序列号不会在 IdM 服务器间同步。因此，请将 DNS 从属服务器配置为仅使用一个 IdM 主服务器。这可防止由非同步 SOA 序列号导致的区域传送失败。

在 UI 中启用区传输

打开 DNS 区页面，如 “在 Web UI 中编辑区域配置”一节 所述，并切换到 Settings 选项卡。

在 Allow transfer 下，指定将区域记录传输到的名称服务器。

图 33.8. 启用区传输

单击 DNS 区域页面顶部的 Save，以确认新配置。

从命令行启用区域传输

要从命令行启用区域传送，请将 --allow-transfer 选项添加到 ipa dnszone-mod 命令中。使用 --allow-transfer 指定区域记录传输到的名称服务器的列表。例如：

```
[user@server ~]$ ipa dnszone-mod --allow-transfer="192.0.2.1;198.51.100.1;203.0.113.1"
example.com
```

在 绑定 服务中启用区传输后，可以按名称传输 IdM DNS 区域，如 dig 工具：

```
[root@server ~]# dig @ipa-server zone_name AXFR
```

33.4.4. 在 DNS 区域中添加记录

IdM 支持许多不同的记录类型。以下四项最常使用：

一个

这是主机名和普通 IPv4 地址的基本映射。A 记录的记录名称是主机名，如 www。A 记录的 IP Address 值是一个标准的 IPv4 地址，如 192.0.2.1。

有关 A 记录的更多信息，请参阅 RFC 1035。

AAAA

这是主机名和 IPv6 地址的基本映射。AAAA 记录的记录名称是主机名，如 www。IP Address 值是一个标准的十六进制 IPv6 地址，如 2001:DB8::1111。

有关 AAAA 记录的更多信息，请参阅 RFC 3596。

SRV

服务(SRV)资源记录将服务名称映射到 提供该特定服务的服务器的 DNS 名称。例如，此记录类型可以将 LDAP 目录等服务映射到管理该服务的服务器。

SRV 记录的记录名称格式为 _service._protocol，如 _ldap._tcp。SRV 记录的配置选项包括目标服务的优先级、权重、端口号和主机名。

有关 SRV 记录的更多信息，请参阅 RFC 2782。

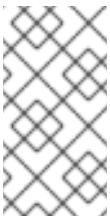
PTR

指针记录类型(PTR)记录添加反向 DNS 记录，该记录将 IP 地址映射到域名。

注意

IPv4 地址的所有反向 DNS 查找都使用在 in-addr.arpa. 域中定义的反向条目。反向地址（人类可读形式）与常规 IP 地址正好相反，其中 in-addr.arpa. 域附加到该地址。例如，对于网络地址 192.0.2.0/24，反向区域为 2.0.192.in-addr.arpa。

PTR 记录的记录名称必须使用 RFC 1035 指定的标准格式，以 RFC 2317 和 RFC 3596 为单位进行扩展。主机名值必须是您要为其创建记录的主机的规范主机名。如需更多信息，请参阅 例 33.8 “PTR 记录”。



注意

也可以为 IPv6 地址配置反向区域，包括 ip6.arpa. 域中的区域。有关 IPv6 反向区的更多信息，请参阅 RFC 3596。

在添加 DNS 资源记录时，请注意，许多记录需要不同的数据。例如，CNAME 记录需要主机名，而 A 记录需要 IP 地址。在 Web UI 中，会自动更新用于添加新记录的表单中的字段，以反映当前选定的记录类型所需的数据。

DNS 通配符支持

IdM 支持 DNS 区域中的特殊记录，作为通配符。

例 33.2. 演示 DNS 通配符结果

1. 在您的 DNS 区域 example.com 中配置以下内容：
   - 通配符 A 记录：
   - mail.example.com 的 192.168.1.0/24 记录，但没有此主机的 A 记录。
   - demo.example.com 没有记录。
2. 查询现有和不存在的 DNS 记录和类型。您将收到以下结果：

```
# host -t MX mail.example.com.
mail.example.com mail is handled by 10 server.example.com.

# host -t MX demo.example.com.
demo.example.com. has no MX record.

# host -t A mail.example.com.
mail.example.com has no A record

# host -t A demo.example.com.
random.example.com has address 192.168.1.1
```

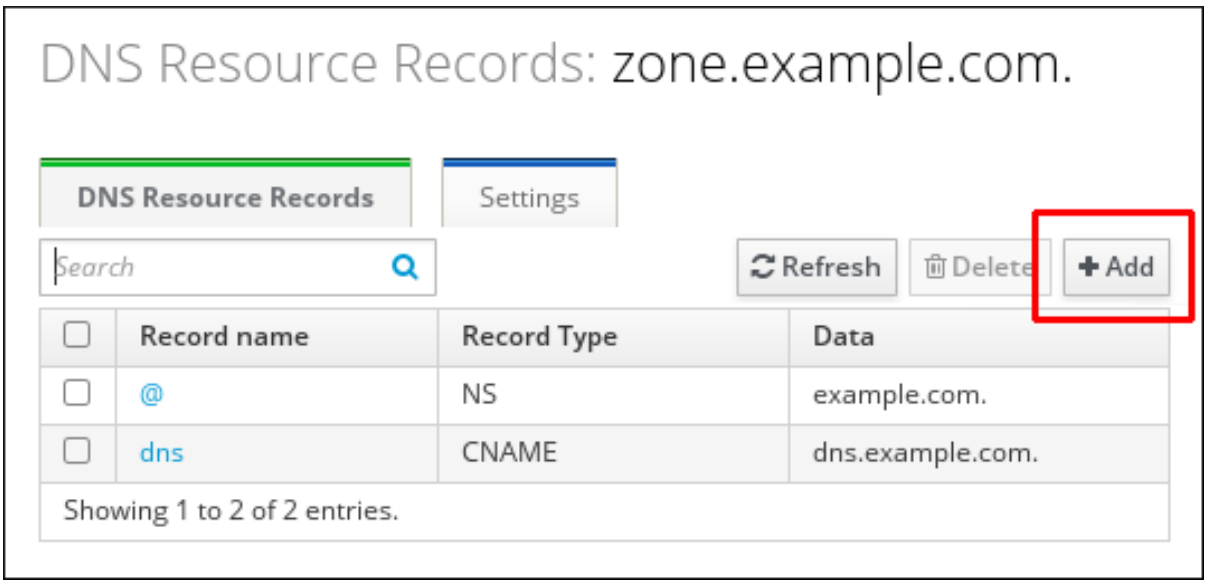
如需了解更多详细信息，请参阅 RFC1034。

从 Web UI 添加 DNS 资源记录

1. 打开 DNS 区页面，如 “在 Web UI 中编辑区域配置”一节 所述。

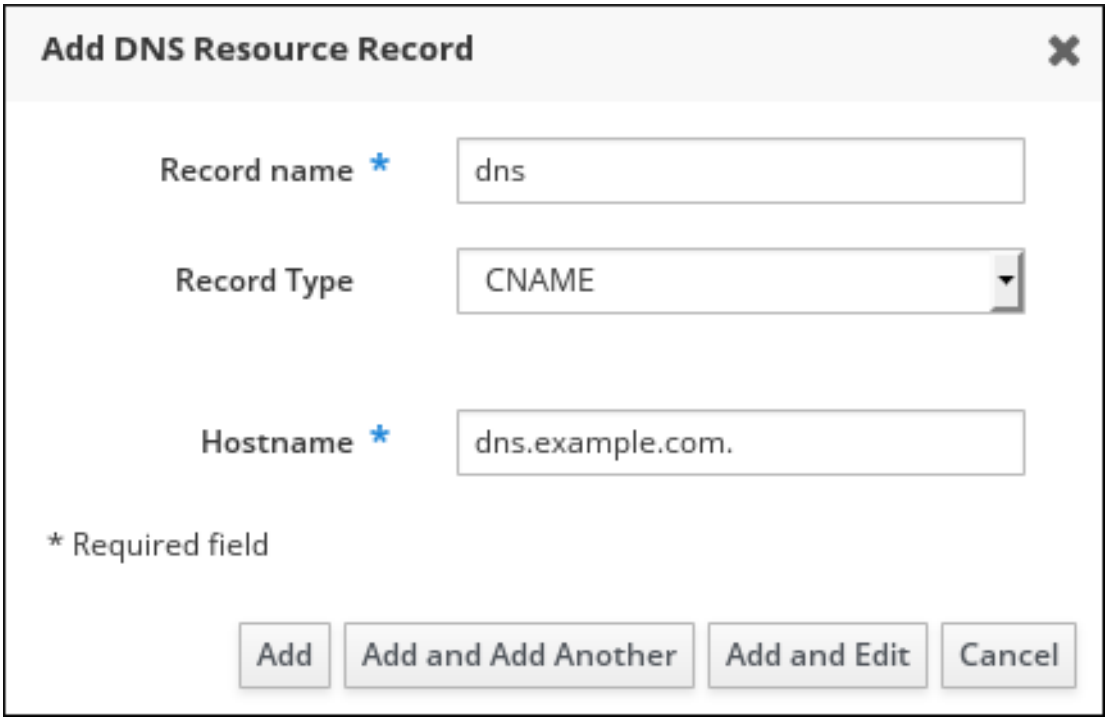
2. 在 DNS Resource Records 部分，点 Add 来添加新记录。

图 33.9. 添加新的 DNS 资源记录



3. 根据需要，选择要创建和填写其他字段的记录类型。

图 33.10. 定义新的 DNS 资源记录



4. 单击 Add 以确认新记录。

从命令行添加 DNS 资源记录

要从命令行添加任何类型的 DNS 资源记录，请使用 ipa dnsrecord-add 命令。该命令遵循以下语法：

```
$ ipa dnsrecord-add zone_name record_name --record_type_option=data
```

zone_name 是记录要添加到的 DNS 区域的名称。record_name 是新 DNS 资源记录的标识符。

表 33.2 “常用 ipa dnsrecord-add 选项” 列出最常见的资源记录类型选项：A(IPv4)、AAAA(IPv6)、SRV 和 PTR。可以通过多次使用 选项和同一命令调用来设置条目列表，或者在 Bash 中列出以逗号分隔的大括号中的选项，如 --option={val1,val2,val3}。

有关如何使用 ipa dnsrecord-add 以及 IdM 支持哪些 DNS 记录类型的更多详细信息，请运行 ipa dnsrecord-add --help 命令。

表 33.2. 常用 ipa dnsrecord-add 选项

常规记录选项	
选项	描述
--ttl=number	设置记录的生存时间。
--structured	解析原始 DNS 记录并以结构化格式返回。

"a"记录选项	
选项	描述
--a-rec=ARECORD	通过 A 记录的列表.
--a-ip-address=string	提供 记录的 IP 地址。

"AAAA"记录选项	
选项	描述
--aaaa-rec=AAAARECORD	传递 AAAA(IPv6)记录列表。

"AAAA"记录选项	
--aaaa-ip-address=string	为 记录指定 IPv6 地址。

"PTR"记录选项	
选项	描述
--ptr-rec=PTRRECORD	传递 PTR 记录列表.
--ptr-hostname=string	为记录指定主机名。

"SRV"记录选项	
选项	描述
--srv-rec=SRVRECORD	传递 SRV 记录列表。
--srv-priority=number	设置记录的优先级。服务类型可以有多个 SRV 记录。优先级(0 - 65535)设置记录的等级；优先级越低，优先级越高。服务必须首先使用优先级最高的 记录。
--srv-weight=number	设置记录的权重。这有助于确定具有相同优先级的 SRV 记录的顺序。设定的权重应最多增加 100 个，表示使用特定记录的几率（以百分比表示）。
--srv-port=number	为目标主机上服务提供服务的端口。
--srv-target=string	提供目标主机的域名。如果服务在域中不可用，这可以是单个句点(.)。

33.4.5. 从命令行添加或修改 DNS 资源记录的示例

例 33.3. 添加 IPv4 记录

以下示例创建了 IP 地址为 192.0.2.123 的记录 www.example.com。

```
$ ipa dnsrecord-add example.com www --a-rec 192.0.2.123
```

例 33.4. 添加 IPv4 通配符记录

以下示例创建了一个通配符 A 记录，其 IP 地址为 192.0.2.123 ：

```
$ ipa dnsrecord-add example.com "*" --a-rec 192.0.2.123
```

例 33.5. 修改 IPv4 记录

在创建记录时，指定 A 记录值的选项是 --a-record。但是，在修改 A 记录时，--a-record 选项用于指定 A 记录的当前值。新值使用 --a-ip-address 选项设置。

```
$ ipa dnsrecord-mod example.com www --a-rec 192.0.2.123 --a-ip-address 192.0.2.1
```

例 33.6. 添加 IPv6 记录

以下示例创建记录 www.example.com，其 IP 地址为 2001:db8::1231:5675。

```
$ ipa dnsrecord-add example.com www --aaaa-rec 2001:db8::1231:5675
```

例 33.7. 添加 SRV 记录

在以下示例中，_ldap._tcp 定义 SRV 记录的服务类型和连接协议。srv-rec 选项定义优先级、权重、端口和目标值。

例如：

```
[root@server ~]# ipa dnsrecord-add server.example.com _ldap._tcp --srv-rec="0 51 389 server1.example.com."
[root@server ~]# ipa dnsrecord-add server.example.com _ldap._tcp --srv-rec="1 49 389 server2.example.com."
```

权重值（本例中为51 和 49 ）添加到 100，并代表使用特定记录的可能性（百分比）。

例 33.8. PTR 记录

添加反向 DNS 记录时，与添加其他 DNS 记录的用法相比，与 ipa dnsrecord-add 命令一起使用的区域名称相反：

```
$ ipa dnsrecord-add reverseNetworkIpAddress hostIpAddress --ptr-rec FQDN
```

通常，hostIpAddress 是给定网络中 IP 地址的最后一个八进制数。

例如，这会为 server4.example.com 添加 IPv4 地址为 192.0.2.4 的 PTR 记录：

```
$ ipa dnsrecord-add 2.0.192.in-addr.arpa 4 --ptr-rec server4.example.com.
```

下一个示例在 0.0.0.0.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa 中添加一个反向 DNS 条目。主机 server2.example.com 的 IPv6 反向区，IP 地址为 2001:DB8::1111 ：

```
$ ipa dnsrecord-add 0.0.0.0.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa. 1.1.1.0.0.0.0.0.0.0.0.0.0.0.0.0 --ptr-rec server2.example.com.
```

33.4.6. 从 DNS 区域中删除记录

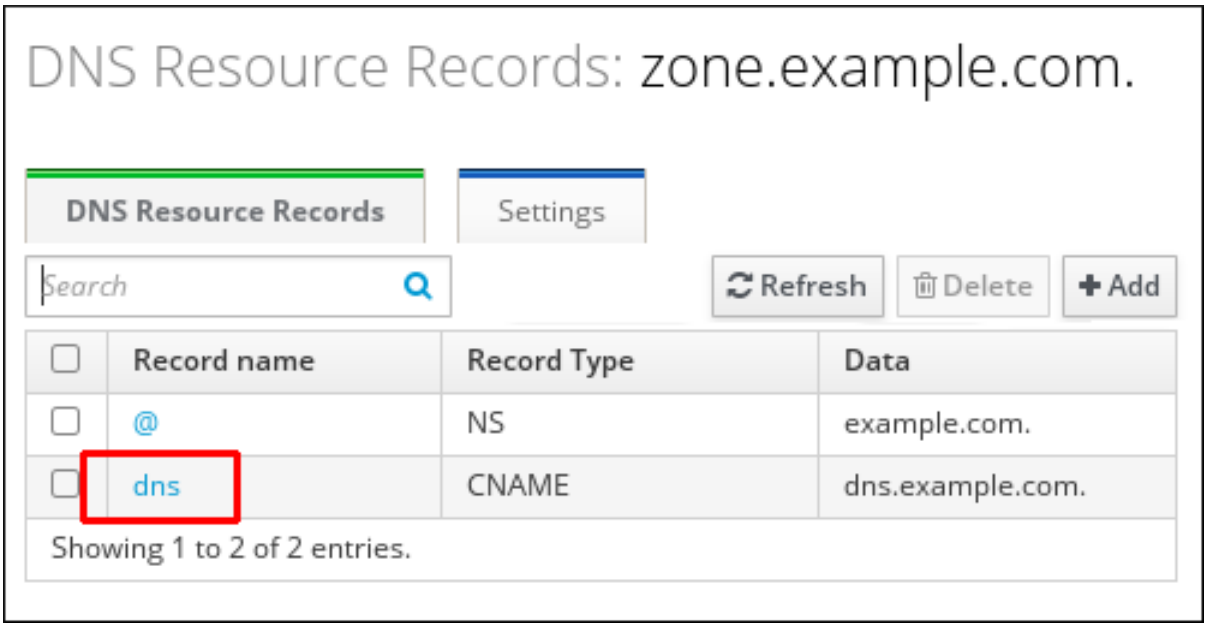
删除 Web UI 中的记录

要只从资源记录中删除特定的记录类型：

1. 打开 DNS 区页面，如 “在 Web UI 中编辑区域配置”一节 所述。

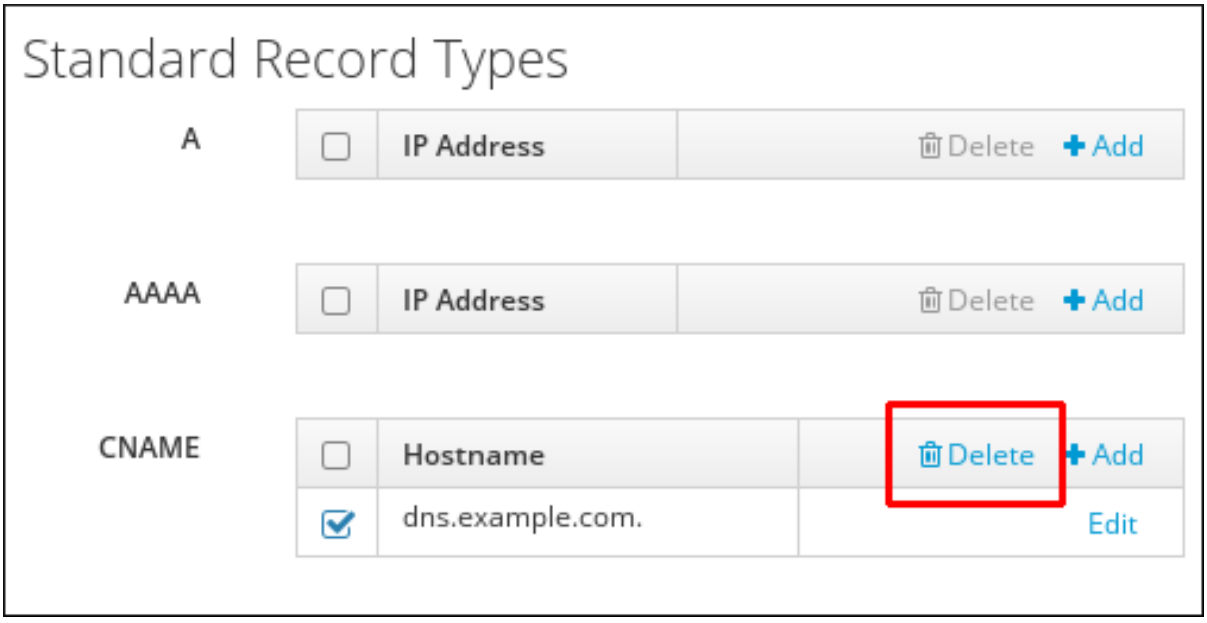
2. 在 DNS Resource Records 部分，点击资源记录的名称。

图 33.11. 选择 DNS 资源记录



3. 按要删除的记录类型的名称选中复选框。

图 33.12. 删除 DNS 资源记录



之后，仅删除所选的记录类型；其他配置将保持不变。

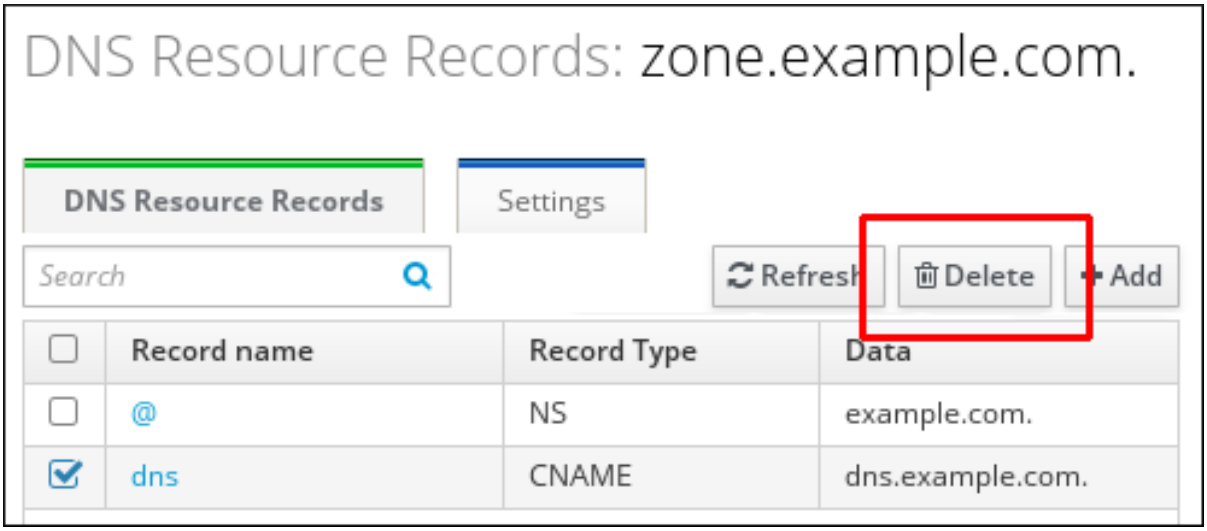
删除区中资源的所有记录：

1. 打开 DNS 区页面，如 *“在 Web UI 中编辑区域配置”一节* 所述。

2.

在 DNS Resource Records 部分中，按要删除的资源记录名称选择复选框，然后单击区域记录顶部的 Delete。

图 33.13. 删除基本资源记录



之后，整个资源记录将被删除。

从命令行删除记录

要从区中删除记录，请使用 ipa dnsrecord-del 命令，并将 --recordType-rec 选项与记录值一起添加。

例如，删除 A 类型记录：

```
$ ipa dnsrecord-del example.com www --a-rec 192.0.2.1
```

如果您在没有任何选项的情况下运行 ipa dnsrecord-del，该命令会提示输入要删除的记录的信息。请注意，通过 命令传递 --del-all 选项可删除该区域的所有关联记录。

有关如何使用 ipa dnsrecord-del 以及命令可接受的选项列表的详细信息，请运行 ipa dnsrecord-del --help 命令。

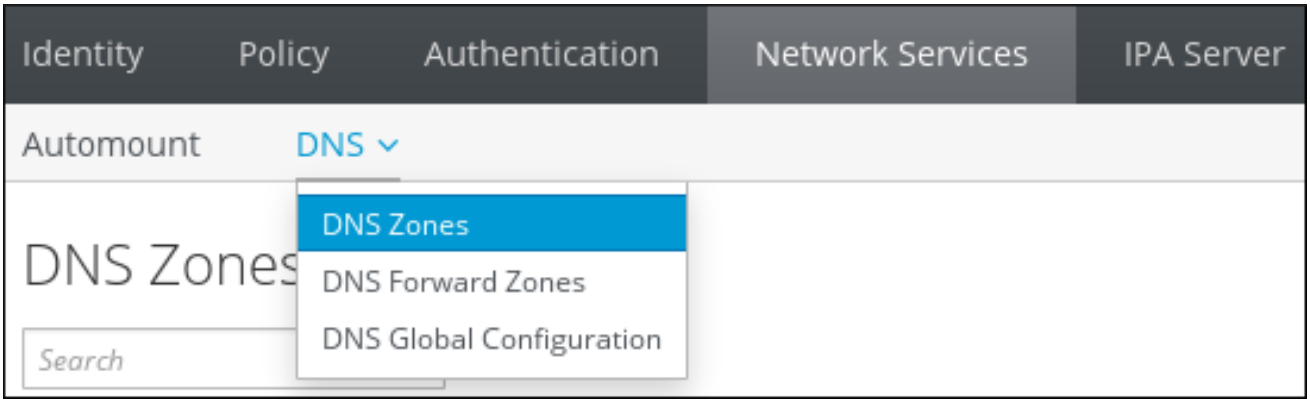
33.4.7. 禁用和启用区域

IdM 允许管理员禁用和启用 DNS 区域。当删除 DNS 区（如 “删除主 DNS 区域”一节 所述）完全删除区条目以及所有相关的配置时，禁用区会在没有从 IdM 永久删除区的情况下将其从活动中删除。也可以再次启用禁用的区域。

在 Web UI 中禁用和启用区域

要从 Web UI 管理 DNS 区域，请打开 网络服务 选项卡，然后选择 DNS 子选项卡，后跟 DNS Zones 部分。

图 33.14. 管理 DNS 区域



若要禁用某个区域，可选中区域名称旁边的复选框，然后单击 Disable。

图 33.15. 禁用 DNS 区域

类似地，若要启用禁用的区域，可选中区域名称旁边的复选框，然后单击 启用。

从命令行禁用和启用 DNS 区域

要从命令行禁用 DNS 区域，请使用 ipa dnszone-disable 命令。例如：

```
[user@server ~]$ ipa dnszone-disable zone.example.com
----------------------------------------
Disabled DNS zone "example.com"
----------------------------------------
```

要重新启用禁用的区域，请使用 ipa dnszone-enable 命令。

33.5. 管理动态 DNS 更新

33.5.1. 启用动态 DNS 更新

IdM 中新 DNS 区域默认禁用动态 DNS 更新。禁用动态更新后，ipa-client-install 脚本无法添加指向新客户端的 DNS 记录。

注意

启用动态更新可能会造成安全风险。但是，如果您的环境中可以接受启用动态更新，您可以将其简化客户端安装。

启用动态更新需要以下内容：

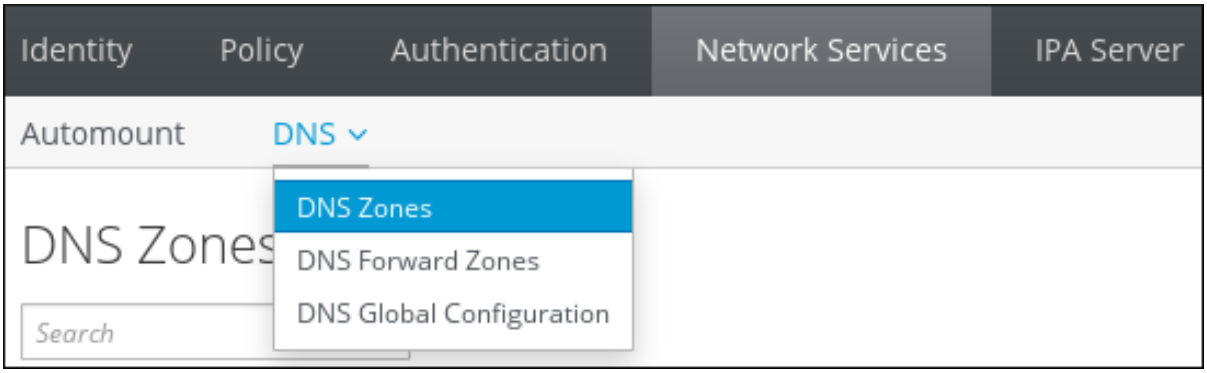
- DNS 区域必须配置为允许动态更新
- 必须将本地客户端配置为发送动态更新

33.5.1.1. 配置 DNS 区域以允许动态更新

在 Web UI 中启用动态 DNS 更新

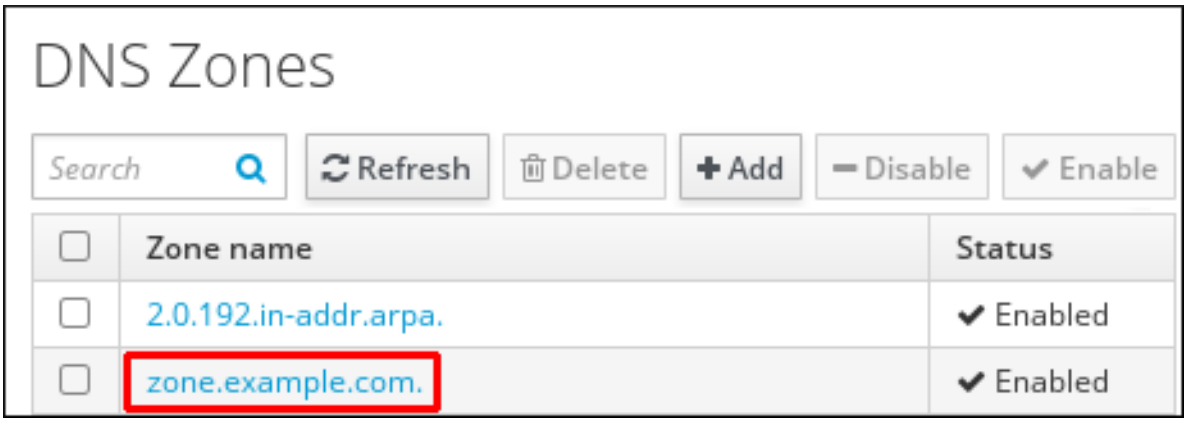
1. 打开 Network Services 选项卡，然后选择 DNS 子选项卡，后跟 DNS Zones 部分。

图 33.16. DNS 区域管理



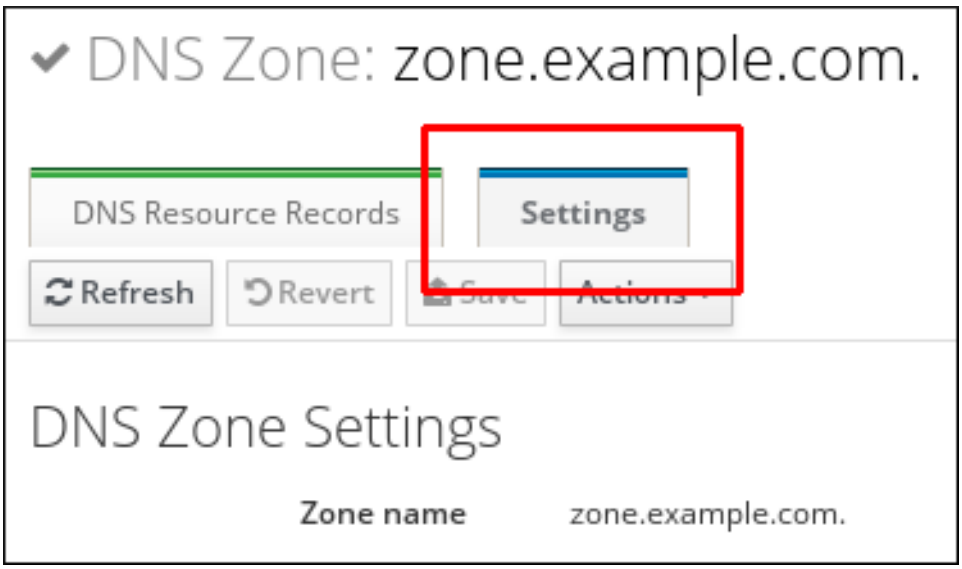
2. 单击所有区域列表中的区域名称，以打开 DNS 区域页面。

图 33.17. 编辑 master zone



3. *点 Settings 切换到 DNS zone settings 选项卡。*

图 33.18. Master zone Edit 页面中的 Settings 选项卡



4. *向下滚动到 Dynamic update 字段，并将值设为 True。*

图 33.19. 启用动态 DNS 更新

5. 单击页面顶部的 Save，以确认新配置。

从命令行启用动态 DNS 更新

要允许从命令行对 DNS 区域进行动态更新，请使用 ipa dnszone-mod 命令和 --dynamic-update=TRUE 选项。例如：

```
[user@server ~]$ ipa dnszone-mod server.example.com --dynamic-update=TRUE
```

33.5.1.2. 将客户端配置为发送动态更新

客户端会在域中注册时自动设置来发送 DNS 更新，方法是将 --enable-dns-updates 选项与 ipa-client-install 脚本搭配使用。

```
[root@client ~]# ipa-client-install --enable-dns-updates
```

DNS 区域具有为其 SOA 配置中为记录设置的生存时间(TTL)值。但是，动态更新的 TTL 由系统安全服务守护进程(SSSD)在本地系统上管理。要更改动态更新的 TTL 值，请编辑 SSSD 文件以设置值；默认值为 1200 秒。

1. 打开 SSSD 配置文件。

```
[root@server ~]# vim /etc/sssd/sssd.conf
```

2. 找到 IdM 域的 domain 部分。

```
[domain/ipa.example.com]
```

3. 如果没有为客户端启用动态更新，则将 dyndns_update 值设置为 true。

```
dyndns_update = true
```

4. 添加或编辑 dyndns_ttl 参数，以以秒为单位设置值。

```
dyndns_ttl = 2400
```

33.5.2. 同步 A/AAAA 和 PTR 记录

和 AAAA 记录与反向区域中的 PTR 记录分开配置。由于这些记录是独立配置的，因此 A/AAAA 记录可以存在，而无需对应的 PTR 记录，反之亦然。

PTR 同步工作有一些 DNS 设置要求：

- 正向和反向区域都必须由 IdM 服务器管理。
- 两个区域都必须启用动态更新。

  第 33.5.1 节 “启用动态 DNS 更新” 中介绍了启用动态更新。
- 必须为 master 正向和反向区域启用 PTR 同步。
- 只有请求客户端的名称与 PTR 记录中的名称匹配时，才会更新 PTR 记录。

重要

通过 IdM Web UI 所做的更改、通过 IdM 命令行工具或直接编辑 LDAP 条目 不会 更新 PTR 记录。仅 DNS 服务本身所做的更改会触发 PTR 记录同步。

警告

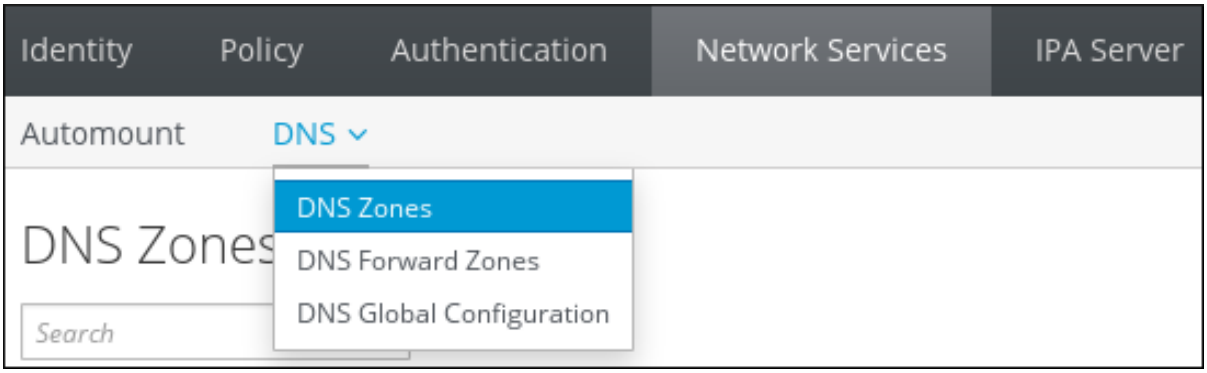
客户端系统可以更新自己的 IP 地址。这意味着，被入侵的客户端可以通过更改其 IP 地址来覆盖 PTR 记录。

33.5.2.1. 在 Web UI 中配置 PTR 记录同步

请注意，PTR 记录同步必须在存储 A 或 AAAA 记录的区域上配置，而不是在 PTR 记录所在的反向 DNS 区域中进行配置。

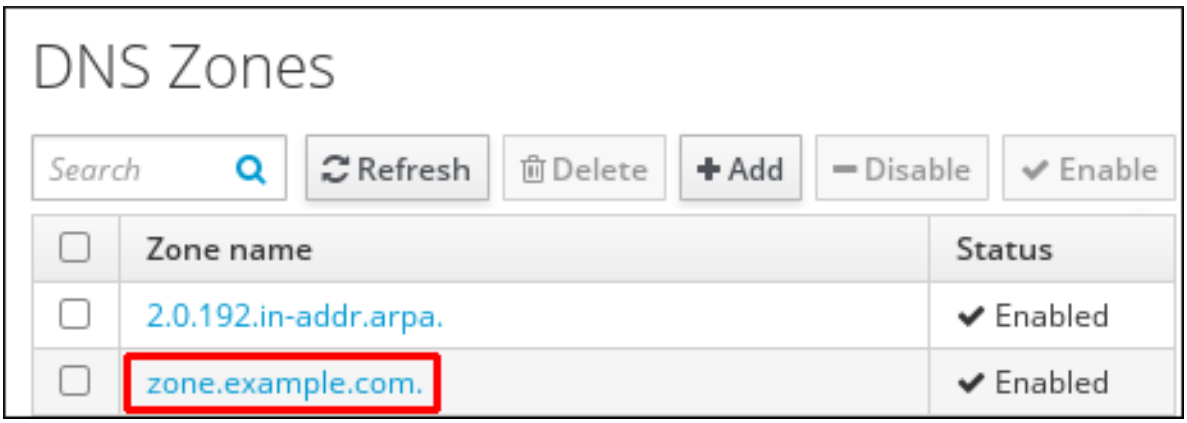
1. 打开 Network Services 选项卡，然后选择 DNS 子选项卡，后跟 DNS Zones 部分。

图 33.20. DNS 区域管理



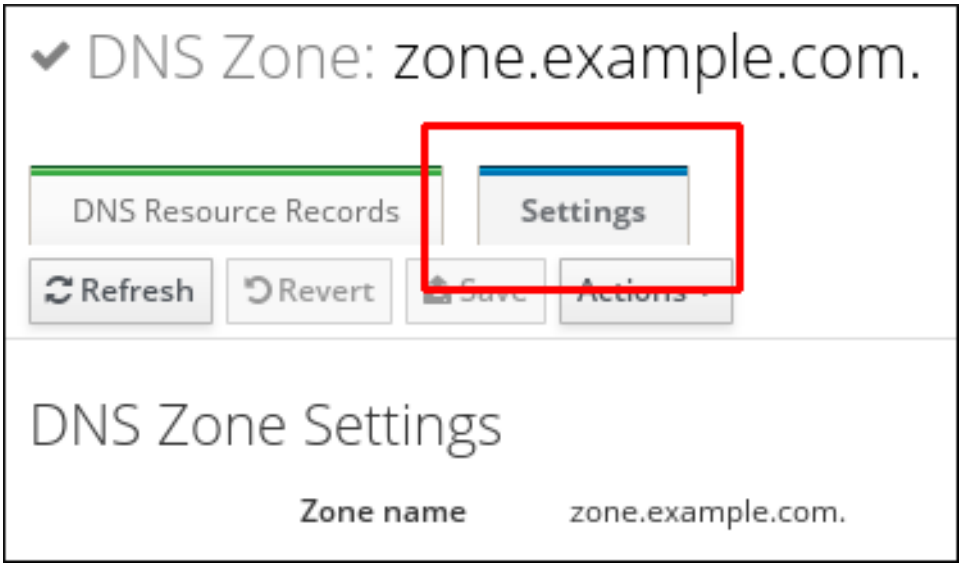
2. 单击所有区域列表中的区域名称，以打开 DNS 区域页面。

图 33.21. 编辑 DNS 区域



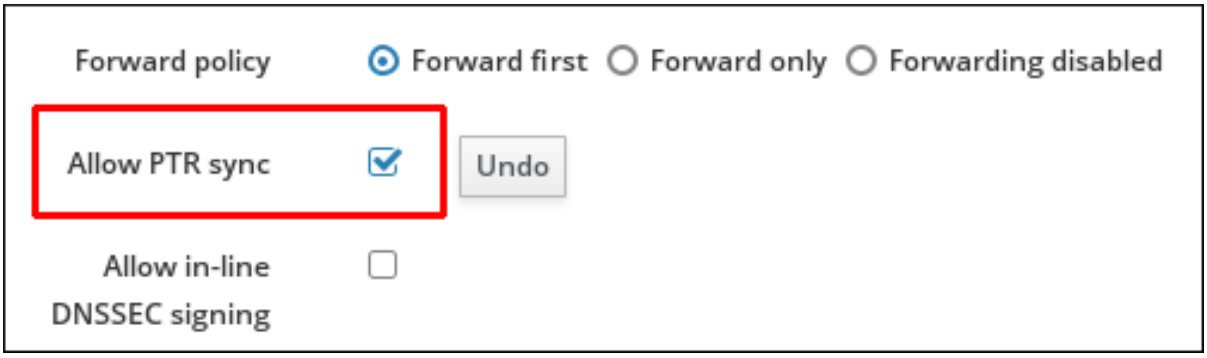
3. 点 Settings 切换到 DNS zone settings 选项卡。

图 33.22. Master zone Edit 页面中的 Settings 选项卡



4. 选择 **Allow PTR sync** 复选框。

图 33.23. 启用 PTR 同步



5. 单击页面顶部的 **Save**，以确认新配置。

33.5.2.2. 使用命令行配置 PTR 记录同步

您可以为特定区域配置 **PTR** 记录同步，也可以使用命令行为所有区域配置 **PTR** 记录同步。

33.5.2.2.1. 为特定区配置 PTR 记录同步

例如，要为 **idm.example.com** 转发区配置 **PTR** 记录同步：

1. 为转发区启用动态更新：

```
# ipa dnszone-mod idm.example.com. --dynamic-update=TRUE
```

2. 配置 **forward** 区的更新策略：

```
# ipa dnszone-mod idm.example.com. --update-policy='grant IDM.EXAMPLE.COM krb5-self * A; grant IDM.EXAMPLE.COM krb5-self * AAAA; grant IDM.EXAMPLE.COM krb5-self * SSHFP;'
```

3. 为 **forward** 区域启用 **PTR** 记录同步：

```
# ipa dnszone-mod idm.example.com. --allow-sync-ptr=True
```

4. 为反向区启用动态更新：

```
# ipa dnszone-mod 2.0.192.in-addr.arpa. --dynamic-update=TRUE
```

33.5.2.2.2. 为所有区域全局配置 PTR 记录同步

您可以使用以下方法之一为 IdM 管理的所有区启用 PTR 同步：

- 同时为所有服务器上的所有区启用 PTR 同步：

```
# ipa dnsconfig-mod --allow-sync-ptr=true
```

- 启用每个服务器同步：

1. 将 sync_ptr yes; 设置为 /etc/named.conf 文件中的 dyndb "ipa" "/usr/lib64/bind/ldap.so" 部分：

```
dyndb "ipa" "/usr/lib64/bind/ldap.so" {
  ...
  sync_ptr yes;
};
```

2. 重启 IdM:

```
# ipactl restart
```

3. 在每个安装了 DNS 服务的 IdM 服务器上重复这些步骤。

33.5.3. 更新 DNS 动态更新策略

IdM 服务器维护的 DNS 域可以根据 RFC 3007 接受 DNS 动态更新[4].

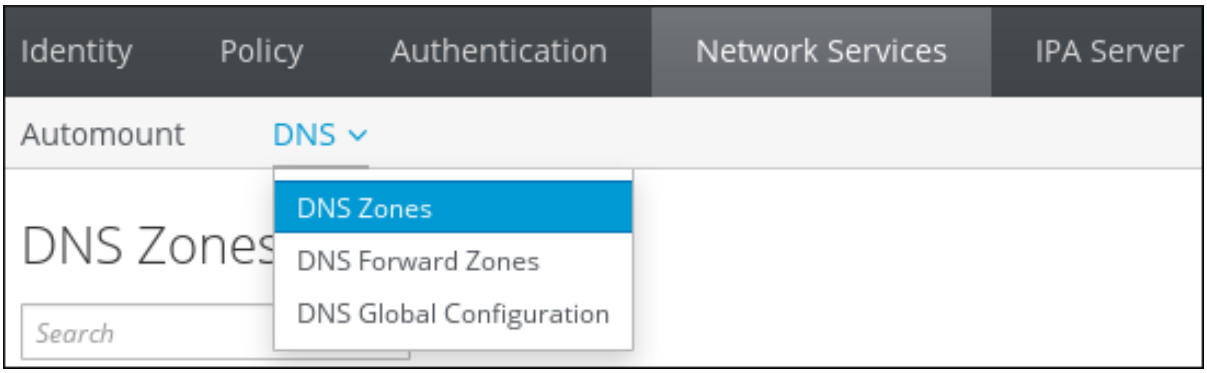
确定特定客户端可修改哪些记录的规则遵循与 /etc/named.conf 文件中的 update-policy 语句相同的语法。如需有关动态更新策略的更多信息，请参阅 BIND 9 文档。

请注意，如果为 DNS 区域禁用动态 DNS 更新，则会拒绝所有 DNS 更新，而不会反映动态更新策略声明。有关启用动态 DNS 更新的详情请参考 第 33.5.1 节 “启用动态 DNS 更新”。

在 Web UI 中更新 DNS 更新策略

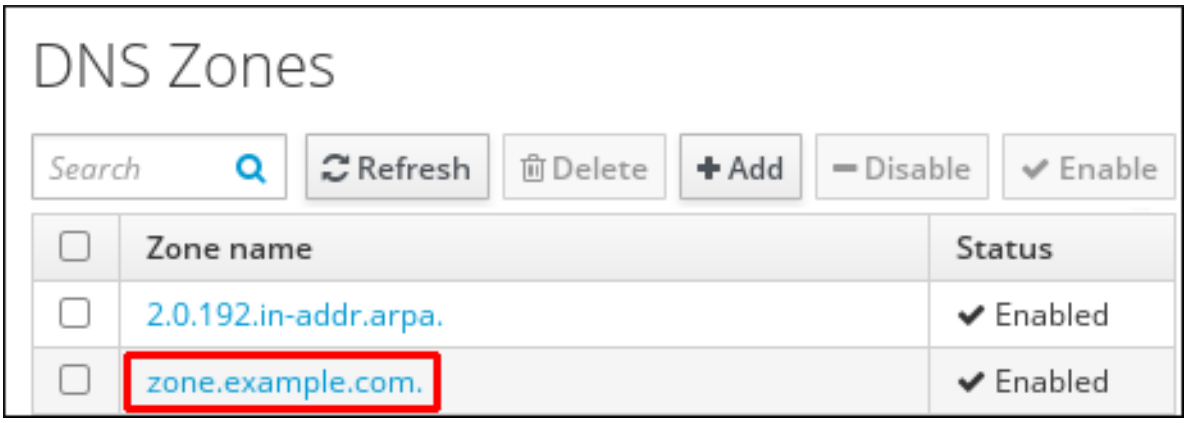
1. *打开 Network Services 选项卡，然后选择 DNS 子选项卡，后跟 DNS Zones 部分。*

图 33.24. DNS 区域管理



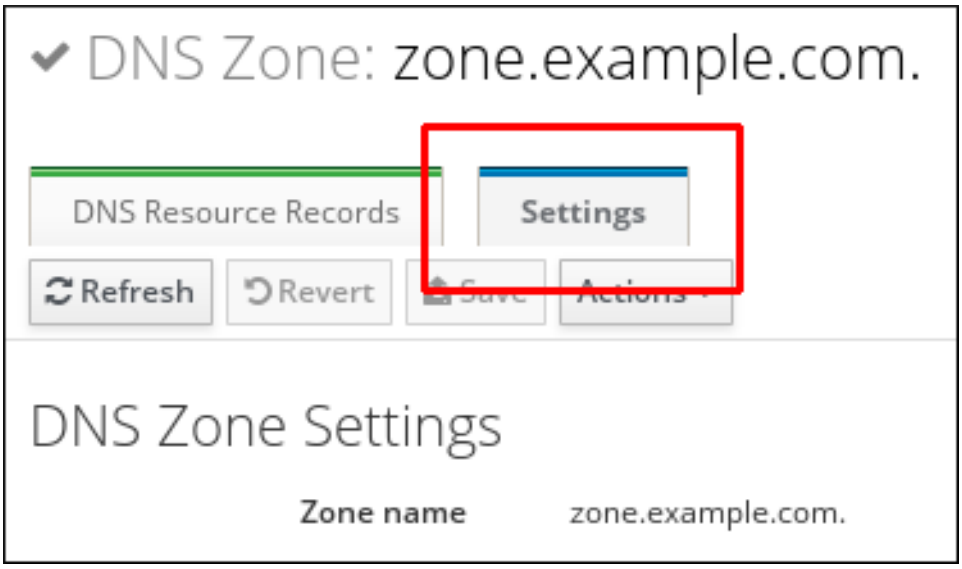
2. *单击所有区域列表中的区域名称，以打开 DNS 区域页面。*

图 33.25. 编辑 DNS 区域



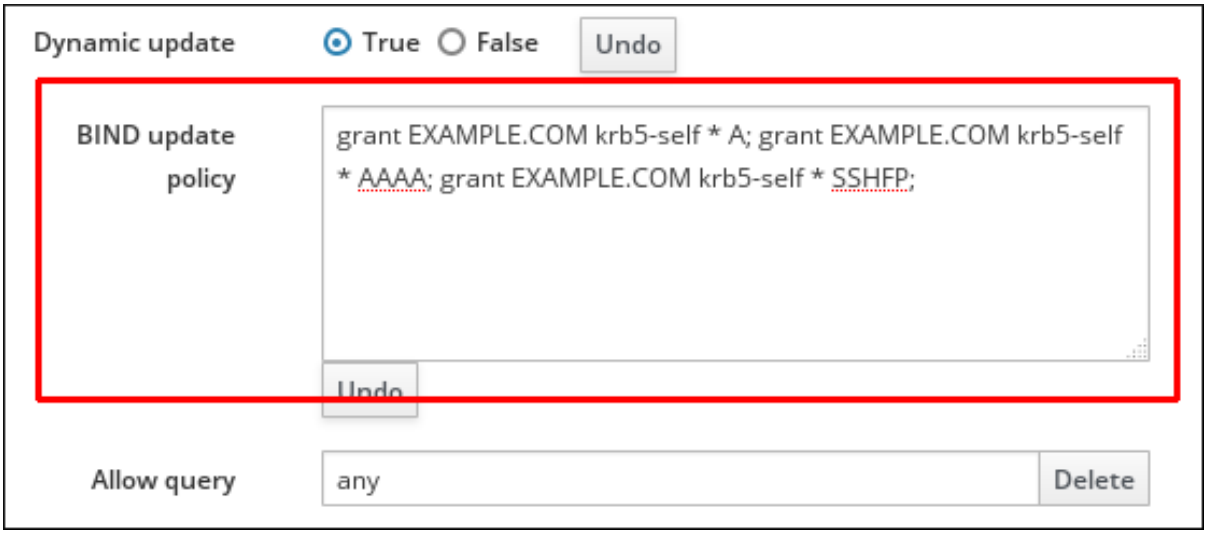
3. *点 Settings 切换到 DNS zone settings 选项卡。*

图 33.26. Master zone Edit 页面中的 Settings 选项卡



4. *在 BIND 更新策略文本框中的分号列表中设置所需的更新策略。*

图 33.27. DNS 更新策略设置



5. *单击 DNS 区域页面顶部的 Save，以确认新配置。*

从命令行更新 DNS 更新策略

要从命令行设置 DNS 更新策略，请使用 --update-policy 选项并在 选项后添加 语句中的访问控制规则。例如：

```
$ ipa dnszone-mod zone.example.com --update-policy "grant EXAMPLE.COM  krb5-self * A; grant EXAMPLE.COM krb5-self * AAAA; grant EXAMPLE.COM  krb5-self * SSHFP;"
```

33.6. 管理 DNS 转发

DNS 转发会影响 DNS 查询的回答方式.默认情况下，与 IdM 集成的 BIND 服务配置为充当权威和递归 DNS 服务器。

当 DNS 客户端查询属于 IdM 服务器具有权威的 DNS 区域的名称时，BIND 会回复配置区域中包含的数据。权威数据始终优先于任何其他数据。

当 DNS 客户端查询 IdM 服务器不是权威的名称时，BIND 会尝试使用其他 DNS 服务器解析查询。如果没有定义转发器，BIND 会询问 Internet 上的根服务器，并使用递归解析算法回答 DNS 查询。

在某些情况下，不建议让 BIND 直接联系其他 DNS 服务器，并根据 Internet 上的可用数据执行递归。这些情况包括：

- 拆分 DNS 配置，也称为 DNS 视图 配置，其中 DNS 服务器向不同的客户端返回不同的答案。拆分 DNS 配置是典型的环境，即部分 DNS 名称在公司网络内可用，但不从外部提供。
- 防火墙限制对 Internet 上 DNS 的访问的配置。
- 在 DNS 级别上带有集中过滤或日志记录的配置。
- 配置，它转发到本地 DNS 缓存，这有助于优化网络流量。

在这种配置中，BIND 不会对公共 Internet 使用完全递归。相反，它使用另一个 DNS 服务器（所谓的转发器 ）来解析查询。当 BIND 配置为使用转发器时，查询和答案会在 IdM 服务器和转发器之间来回转发，IdM 服务器则充当非权威数据的 DNS 缓存。

forward 策略

IdM 支持 第一个 和 唯一的 标准 BIND 转发策略，以及 none IdM 特定的转发策略。

转发第一（默认）

DNS 查询转发到配置的转发器。如果查询因为服务器错误或超时而失败，BIND 将使用 Internet 上的服务器返回递归解析。forward first 策略是默认策略。它适用于流量优化。

仅转发

DNS 查询转发到配置的转发器。如果查询因为服务器错误或超时而失败，BIND 会向客户端返回错误。对于采用拆分 DNS 配置的环境，建议使用 forward only 策略。

无：转发禁用

DNS 查询不会被转发。禁用转发仅作为全局转发配置的特定区域覆盖。这个选项等同于在 BIND 配置中指定一个空转发器列表。

转发不会合并 IdM 和其他 DNS 服务器中的数据

转发无法用于将 IdM 中的数据与其他 DNS 服务器的数据组合。您只能在 IdM DNS 中转发对 master 区的特定子区的查询：请参阅 “IdM DNS Master 区域中的区委派”一节。

默认情况下，如果查询的 DNS 名称属于 IdM 服务器对其具有权威的区域，BIND 服务不会将查询转发到其他服务器。在这种情况下，如果无法在 IdM 数据库中找到查询的 DNS 名称，则会返回 NXDOMAIN 回答。不使用转发。

> 例 33.9. Scenario 示例
>
> IdM 服务器对 test.example 具有权威。DNS 区域.BIND 配置为将查询转发到 IP 地址 192.0.2.254 的 DNS 服务器。
>
> 客户端发送对 不存在.test.example的查询时。DNS 名称，BIND 检测到 IdM 服务器对 test.example. 区域具有权威，并且不会将查询转发到 192.0.2.254. 服务器。因此，DNS 客户端会收到 NXDomain 回答，通知用户查询的域不存在。

IdM DNS Master 区域中的区委派

可以在 IdM DNS 中转发对 master 区的特定子区的查询。例如，如果 IdM DNS 处理区域 idm.example.com，您可以将 sub_zone1.idm.example.com 子区的颁发机构委派给不同的 DNS 服务器。要实现此行为，您需要按上文所述使用转发以及将子区域委派给其他 DNS 服务器的名称服务器记录。在以下示例中，sub_zone1 是子区，192.0.2.1 是子区委托给的 DNS 服务器的 IP 地址：

```
$ ipa dnsrecord-add idm.example.com. sub_zone1 --ns-rec=192.0.2.1
```

添加 forward 区域后类似如下：

```
$ ipa dnsforwardzone-add sub_zone1.idm.example.com. --forwarder 192.0.2.1
```

33.6.1. 配置全局转发器

全局转发 器是 DNS 服务器，用于解析 IdM 服务器不具有权威的所有 DNS 查询，如 第 33.6 节 “管理 DNS 转发” 所述。

管理员可以通过以下两种方式为全局转发配置 IP 地址和转发策略：

使用 ipa dnsconfig-mod 命令或 IdM Web UI

使用这些原生 IdM 工具设置的配置会立即应用到所有 IdM DNS 服务器。如 第 33.3 节 “DNS 配置优先级” 所述，全局 DNS 配置的优先级高于 /etc/named.conf 文件中定义的本地配置。

通过编辑 /etc/named.conf 文件

在每个 IdM DNS 服务器上手动编辑 /etc/named.conf，允许在每个服务器上使用不同的全局转发器和策略。请注意，在更改 /etc/named.conf 后，必须重启 BIND 服务。

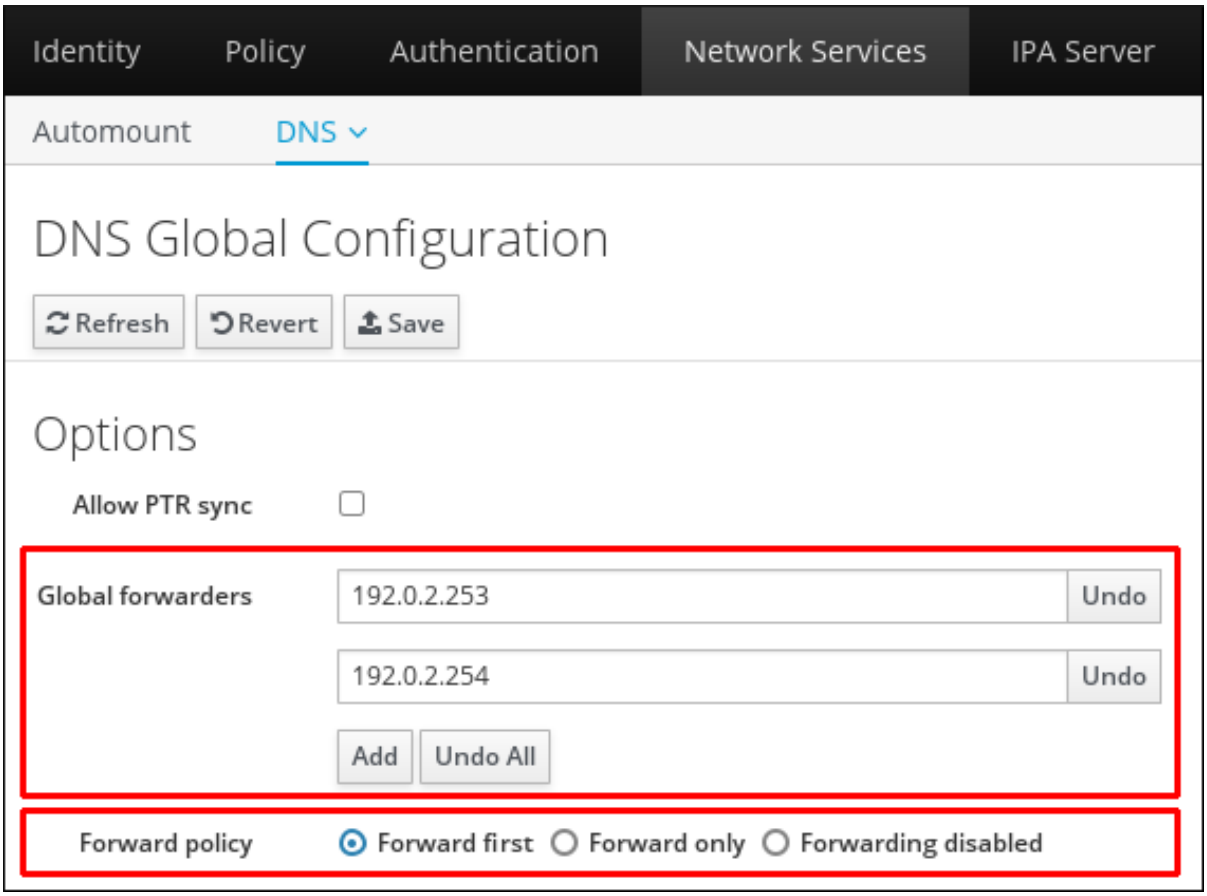
在 Web UI 中配置 Forwarders

在 IdM Web UI 中定义 DNS 全局配置：

1. 单击 Network Services 选项卡，然后选择 DNS 子选项卡，后跟 DNS Global Configuration 部分。

2. 要添加新的全局转发器，请点 Add 并输入 IP 地址。要定义新的转发策略，请从可用策略列表中选择它。

图 33.28. 在 Web UI 中编辑全局 DNS 配置



3. 单击 **Save** 以确认新配置。

从命令行配置转发器

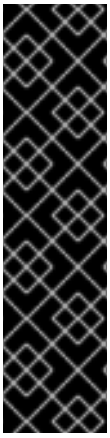
要从命令行设置全局转发器的全局列表，请使用 **ipa dnsconfig-mod** 命令。它通过编辑 LDAP 数据来编辑 DNS 全局配置。**ipa dnsconfig-mod** 命令及其选项会一次性影响所有 IdM DNS 服务器，并覆盖任何本地配置。

例如，要使用 **ipa dnsconfig-mod** 编辑全局转发器列表：

```
[user@server ~]$ ipa dnsconfig-mod --forwarder=192.0.2.254
  Global forwarders: 192.0.2.254
```

33.6.2. 配置转发区

转发区域不包含任何权威数据，并指示名称服务器仅将属于特定区域的查询转发到配置的转发器。



重要

除非绝对需要，否则不要使用 forward 区域。限制对覆盖全局转发配置的使用。在大多数情况下，不需要只配置全局转发 （第 33.6.1 节 “配置全局转发器” 描述）和转发区。

forward zone 是一个非标准的解决方案，使用它们可能会导致意外和有问题的行为。在创建新的 DNS 区域时，红帽建议始终使用 NS 记录使用标准 DNS 委派，并避免转发区域。

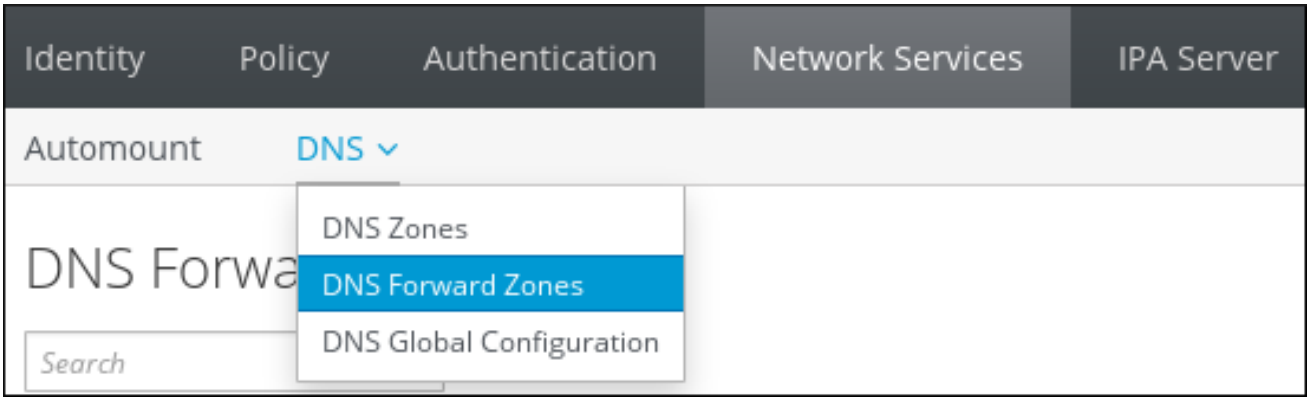
有关支持的转发策略的详情请参考 “forward 策略”一节。

有关 BIND 服务的详情，请查看 Red Hat Enterprise Linux 网络指南，BIND 9 管理员参考手册，包含在 /usr/share/doc/bind-version_number/ 目录或外部源 [5] .

在 Web UI 中配置转发区

要在 Web UI 中管理转发区域，请单击 Network Services 选项卡，然后选择 DNS 子选项卡，后跟 DNS Forward Zones 部分。

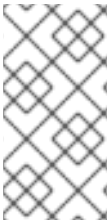
图 33.29. 管理 DNS 转发区域



在 DNS Forward Zones 部分中，管理员可以处理有关转发区域的所有必要操作：显示当前转发区域列表、添加新的转发区域、删除区、显示转发区，允许每个转发区修改转发器和转发策略，以及禁用或启用转发区域。

从命令行配置转发区

要从命令行管理转发区域，请使用下面描述的 ipa dnsforwardzone the 命令。



注意

ipa dnsforwardzone the 命令的行为与用于管理 master 区的 ipa dnszone the 命令一致。

ipa dnsforwardzone the 命令接受几个选项，特别是 --forwarder、--forward-policy 和 --name-from-ip 选项。有关可用选项的详情，请参考 表 33.1 “zone 属性” 或添加 --help 选项运行命令，例如：

```
ipa dnsforwardzone-add --help
```

添加转发区

使用 dnsforwardzone-add 命令添加新的转发区域。如果转发策略没有设置为 none，则需要至少指定一个转发器。

```
[user@server ~]$ ipa dnsforwardzone-add zone.test. --forwarder=172.16.0.1 --
forwarder=172.16.0.2 --forward-policy=first

Zone name: zone.test.
Zone forwarders: 172.16.0.1, 172.16.0.2
Forward policy: first
```

修改转发区

使用 dnsforwardzone-mod 命令修改转发区域。如果转发策略不是任何，则需要至少指定一个转发器。可以通过多种方式进行修改。

```
[user@server ~]$ ipa dnsforwardzone-mod zone.test. --forwarder=172.16.0.3

Zone name: zone.test.
Zone forwarders: 172.16.0.3
Forward policy: first
```

```
[user@server ~]$ ipa dnsforwardzone-mod zone.test. --forward-policy=only

Zone name: zone.test.
Zone forwarders: 172.16.0.3
Forward policy: only
```

显示转发区域

使用 dnsforwardzone-show 命令显示指定转发区域的信息。

```
[user@server ~]$ ipa dnsforwardzone-show zone.test.
```

```
Zone name: zone.test.
Zone forwarders: 172.16.0.5
Forward policy: first
```

查找转发区域

使用 dnsforwardzone-find 命令查找指定的转发区域。

```
[user@server ~]$ ipa dnsforwardzone-find zone.test.

Zone name: zone.test.
Zone forwarders: 172.16.0.3
Forward policy: first
----------------------------
Number of entries returned 1
----------------------------
```

删除转发区

使用 dnsforwardzone-del 命令删除指定的转发区域。

```
[user@server ~]$ ipa dnsforwardzone-del zone.test.

----------------------------
Deleted forward DNS zone "zone.test."
----------------------------
```

启用和禁用转发区域

使用 dnsforwardzone-enable 和 dnsforwardzone-disable 命令来启用和禁用转发区。请注意，转发区会被默认启用。

```
[user@server ~]$ ipa dnsforwardzone-enable zone.test.

----------------------------
Enabled forward DNS zone "zone.test."
----------------------------
```

```
[user@server ~]$ ipa dnsforwardzone-disable zone.test.

----------------------------
Disabled forward DNS zone "zone.test."
----------------------------
```

添加和删除权限

使用 dnsforwardzone-add-permission 和 dnsforwardzone-remove-permission 命令来添加或删除系统权限。

```
[user@server ~]$ ipa dnsforwardzone-add-permission zone.test.

---------------------------------------------------------
Added system permission "Manage DNS zone zone.test."
---------------------------------------------------------
  Manage DNS zone zone.test.
```

```
[user@server ~]$ ipa dnsforwardzone-remove-permission zone.test.

---------------------------------------------------------
Removed system permission "Manage DNS zone zone.test."
---------------------------------------------------------
  Manage DNS zone zone.test.
```

33.7. 管理反向 DNS 区域

可以通过以下两种方式识别反向 DNS 区域：

- 根据区域名称，格式为 reverse_ipv4_address.in-addr.arpa 或 reverse_ipv6_address.ip6.arpa。

  反向 IP 地址通过反转 IP 地址的组件的顺序来创建。例如，如果 IPv4 网络是 192.0.2.0/24，反向区域名称为 2.0.192.in-addr.arpa.（带有尾部句点）。

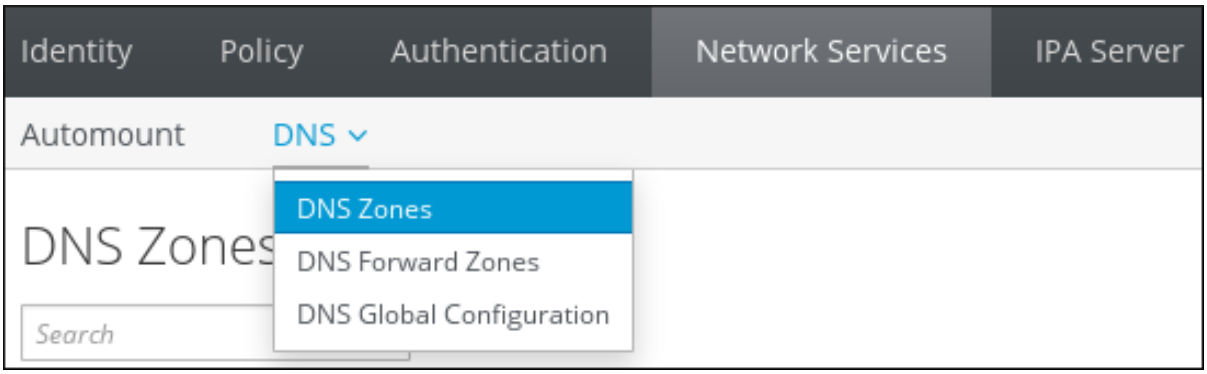
- 根据网络地址，格式为 network_ip_address/subnet_mask_bit_count

  要通过其 IP 网络创建反向区域，请将网络信息设置为（正向式）IP 地址，并使用子网掩码位计数。对于 IPv4 地址，位数必须是 8 的倍数，或者 IPv6 地址的倍数。

在 Web UI 中添加反向 DNS 区域

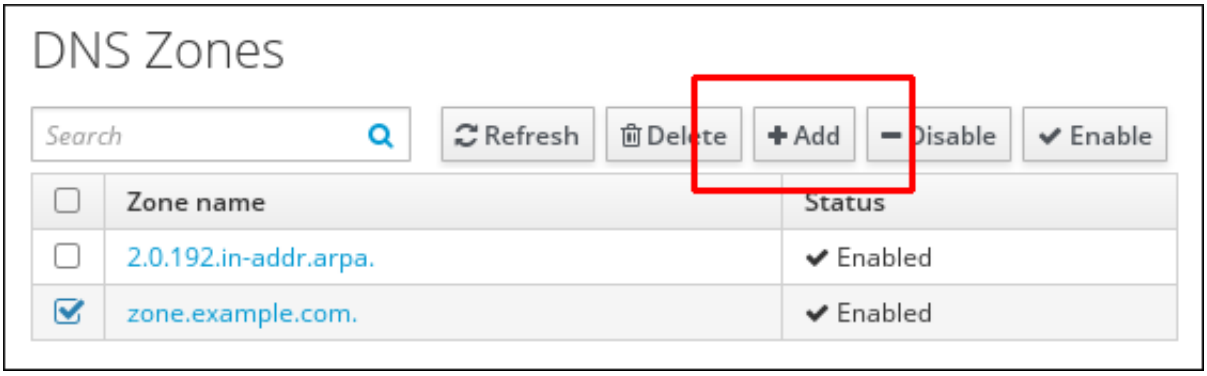
1. 打开 Network Services 选项卡，然后选择 DNS 子选项卡，后跟 DNS Zones 部分。

图 33.30. DNS 区域管理



2. *单击所有区域列表顶部的 **Add**。*

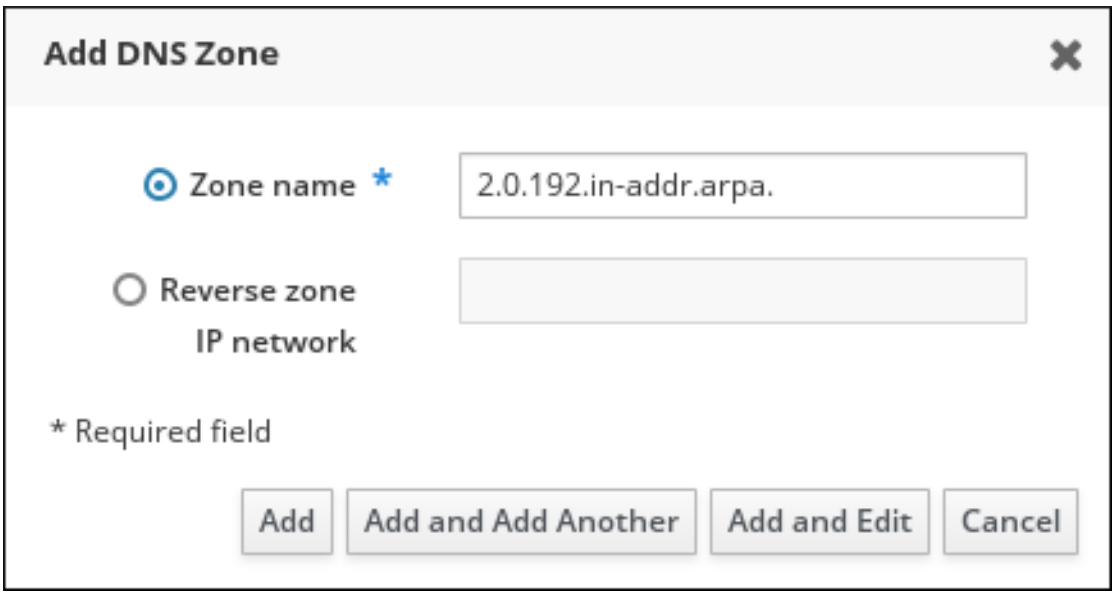
图 33.31. 添加反向 DNS 区域



3. *填写区域名称或反向区域 IP 网络。*

   a. *例如，按区名称添加反向 DNS 区域：*

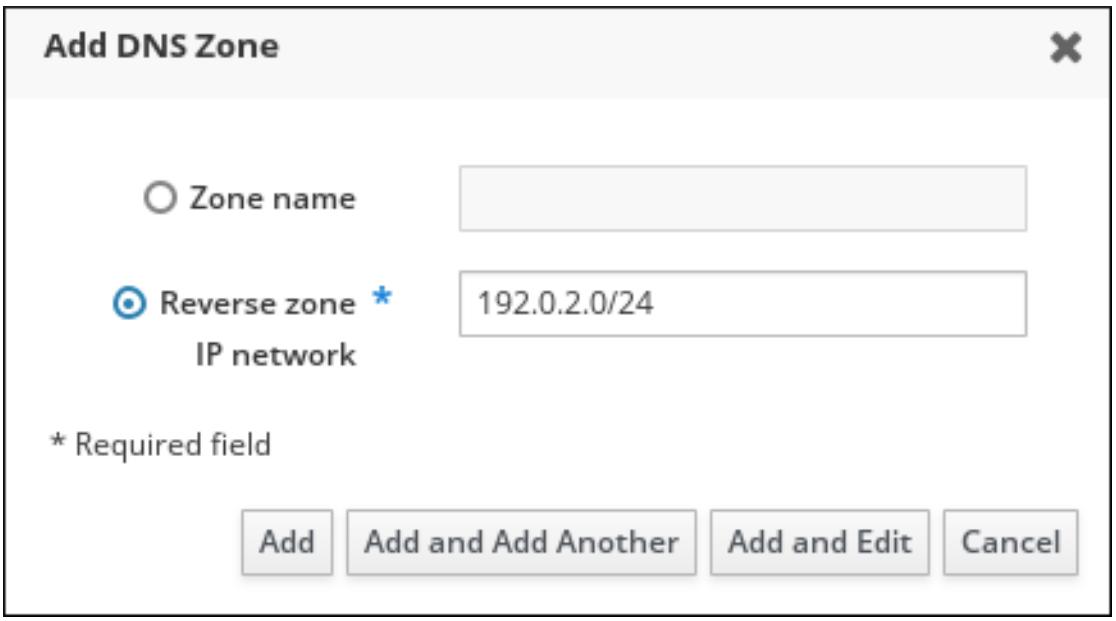
图 33.32. 根据名称创建反向区域



b.

或者，通过反向区 IP 网络添加反向 DNS 区域：

图 33.33. 通过 IP 网络创建反向区域



Reverse zone IP network 字段的验证器会在输入过程中警告有关无效的网络地址。输入完整网络地址后，该警告将消失。

4.

单击 Add 以确认新的反向区域。

从命令行添加反向 DNS 区域

要从命令行创建反向 DNS 区域，请使用 ipa dnszone-add 命令。

例如，要根据区名称创建反向区：

```
[user@server]$ ipa dnszone-add 2.0.192.in-addr.arpa.
```

或者，通过 IP 网络创建反向区：

```
[user@server ~]$ ipa dnszone-add --name-from-ip=192.0.2.0/24
```

反向 DNS 区域的其他管理操作

第 33.4 节 “管理主 DNS 区域” 描述其他区域管理操作，其中一些也适用于反向 DNS 区域管理，如编辑或禁用和启用 DNS 区域。

33.8. 定义 DNS 查询策略

要解析 DNS 域中的主机名，DNS 客户端向 DNS 名称服务器发出查询。对于某些安全上下文或性能，建议限制客户端可以查询区域中的 DNS 记录。

可以在创建区域时配置 DNS 查询，或使用 ipa dnszone-mod 命令的 --allow-query 选项设置允许发出查询的客户端列表。

例如：

```
[user@server ~]$ ipa dnszone-mod --allow-query=192.0.2.0/24;2001:DB8::/32;203.0.113.1
example.com
```

默认的 --allow-query 值是 any，它允许任何客户端查询区域。

33.9. DNS 位置

33.9.1. 基于 DNS 的服务发现

基于 DNS 的服务发现是一种进程，客户端使用 DNS 协议在提供特定服务（如 LDAP 或 Kerberos）的网络中查找服务器。种典型的操作类型是允许客户端在最接近的网络基础架构中定位身份验证服务器，因为它们可提供更高的吞吐量和更低的网络延迟，从而降低总成本。

服务发现的主要优点是：

- 不需要为客户端配置专用服务器的名称。
- DNS 服务器用作策略的中央提供程序。使用相同的 DNS 服务器的客户端有权访问相同的策略，如服务提供商及其首选顺序。

在 IdM 域中，LDAP、Kerberos 和其他服务的 DNS 服务记录(SRV 记录)存在。例如，以下命令查询 DNS 服务器以获取在 IdM DNS 域中提供基于 TCP 的 Kerberos 服务的主机：

例 33.10. DNS 位置独立结果

```
$ dig -t SRV +short _kerberos._tcp.idm.example.com
0 100 88 idmserver-01.idm.example.com.
0 100 88 idmserver-02.idm.example.com.
```

输出包含以下信息：

- 0 （优先级）：目标主机的优先级.首选值较低。
- 100 (权重).为具有相同优先级的条目指定相对权重。如需更多信息，请参阅 RFC 2782 第 3 节。
- 88 （端口号）：服务的端口号。
- 提供服务的主机的规范名称。

在上例中，返回的两个主机名具有相同的优先级和权重。在这种情况下，客户端使用结果列表中的随机条目。

当客户端查询在 DNS 位置配置的 DNS 服务器时，输出会有所不同。对于分配到某个位置的 IdM 服务器，会返回定制值。在以下示例中，客户端查询位置 germany 中的 DNS 服务器：

例 33.11. 基于 DNS Location 的结果

```
$ dig -t SRV +short _kerberos._tcp.idm.example.com
_kerberos._tcp.germany._locations.idm.example.com.
0 100 88 idmserver-01.idm.example.com.
50 100 88 idmserver-02.idm.example.com.
```

IdM DNS 服务器自动返回一个 DNS 别名(CNAME)，指向一个 DNS 位置特定的 SRV 记录（首选本地服务器）。此 CNAME 记录显示在输出的第一行中。在上例中，主机 idmserver-01.idm.example.com 具有最低的优先级值，因此首选。idmserver-02.idm.example.com 具有更高的优先级，因此仅在首选主机不可用的情况下用作备份。

33.9.2. DNS 位置的部署注意事项

对于对主 IdM DNS 域具有权威的 IdM DNS 服务器，IdM 可以生成特定于位置的 SRV 记录。因为每个 IdM DNS 服务器都会生成特定于位置的 SRV 记录，所以您必须在每个 DNS 位置至少安装一个 IdM DNS 服务器。

客户端与 DNS 位置的关联性仅由客户端收到的 DNS 记录定义。因此，如果客户端执行 DNS 服务发现从 IdM DNS 服务器解析特定于位置的记录，您可以将 IdM DNS 服务器与非 IdM DNS 从属服务器合并，并递归器。

在大多数带有混合 IdM 和非 IdM DNS 服务的部署中，DNS 递归器会使用往返时间指标自动选择最接近的 IdM DNS 服务器。通常，这可确保使用非 IdM DNS 服务器的客户端正在获取最接近的 DNS 位置的记录，从而使用最佳 IdM 服务器集。

33.9.2.1. 生存 DNS 时间(TTL)

客户端可以在区域的配置中设置的大量时间缓存 DNS 资源记录。由于这种缓存，客户端可能无法接收更改，直到生存时间(TTL)值过期。IdM 中的默认 TTL 值为 1 天。

如果您的客户端计算机在站点间漫游，您应该调整 IdM DNS 区的 TTL 值。将该值设置为小于客户端在站点之间漫游的时间。这样可确保客户端上缓存的 DNS 条目在重新连接到另一个站点之前过期，从而查询 DNS 服务器来刷新特定位置的 SRV 记录。

有关如何修改 DNS 区的默认 TTL 的详情请参考 第 33.4.2 节 “为主 DNS 区域添加额外的配置”。

33.9.3. 创建 DNS 位置

从 Web UI 创建 DNS 位置

1. 打开 IPA Server 选项卡，然后选择 Topology 子选项卡。
2. 单击导航栏中的 IPA Locations。
3. 单击位置列表顶部的 Add。
4. 填写位置名称。

5. 单击 **添加** 按钮以保存位置。

为要添加的更多位置重复上述步骤。

从命令行创建 DNS 位置

例如，要创建新位置 **germany**，请输入：

```
[root@server ~]# ipa location-add germany
----------------------------
Added IPA location "germany"
----------------------------
  Location name: germany
```

为所有要添加的位置重复此步骤。

33.9.4. 将 IdM 服务器分配给 DNS 位置

通过 Web UI 将 IdM 服务器分配给 DNS 位置

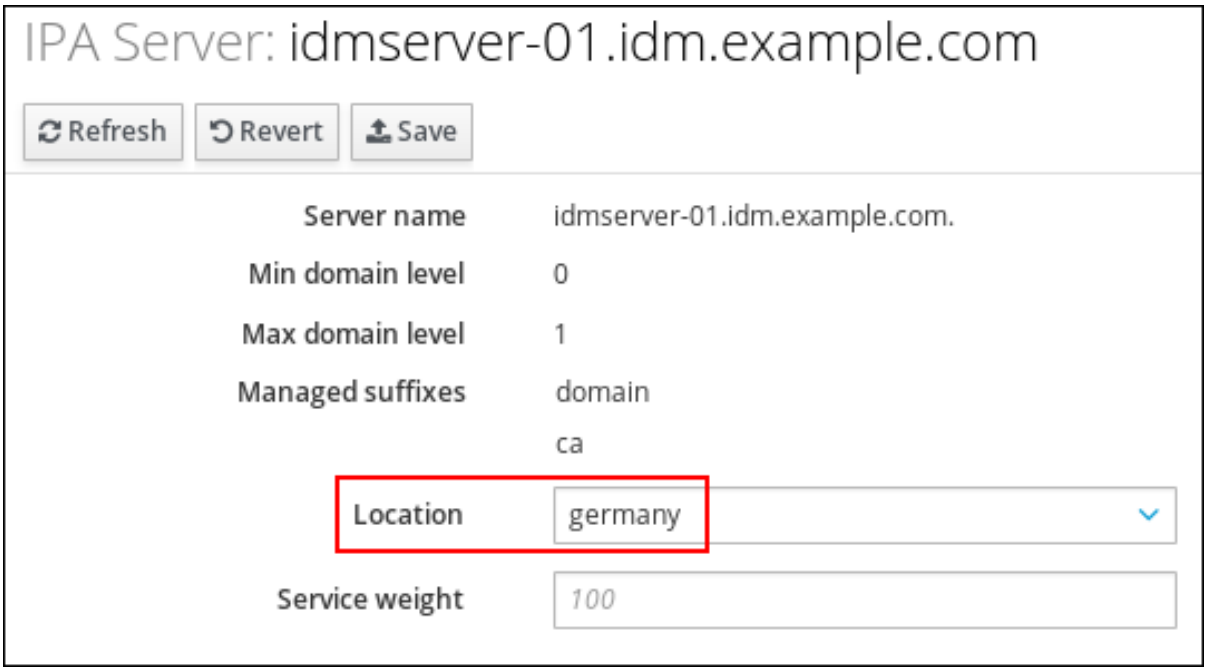
1. 打开 **IPA Server** 选项卡，然后选择 **Topology** 子选项卡。

2. 单击导航中的 **IPA Servers**。

3. 单击 **IdM** 服务器名称。

4. 选择 **DNS** 位置，并选择性地设置服务权重：

图 33.34. 将服务器分配到 DNS 位置



5. *点击 **Save**。*

6. *在您在前面的步骤中指定的主机上重启 **named-pkcs11** 服务：*

```
[root@idmserver-01 ~]# systemctl restart named-pkcs11
```

对您要为其分配 DNS 位置的其他 IdM 服务器重复这些步骤。

从命令行将 IdM 服务器分配给 DNS 位置

1. *可选：列出所有配置的 DNS 位置：*

```
[root@server ~]# ipa location-find
-----------------------
2 IPA locations matched
-----------------------
Location name: australia
Location name: germany
----------------------------
Number of entries returned: 2
----------------------------
```

2. *将服务器分配到 DNS 位置。例如，要将位置 germany 分配给服务器 idmserver-01.idm.example.com，请运行：*

```
[root@server ~]# ipa server-mod idmserver-01.idm.example.com --location=germany
ipa: WARNING: Service named-pkcs11.service requires restart on IPA server
idmserver-01.idm.example.com to apply configuration changes.
----------------------------------------------------
Modified IPA server "idmserver-01.idm.example.com"
----------------------------------------------------
Servername: idmserver-01.idm.example.com
Min domain level: 0
Max domain level: 1
Location: germany
Enabled server roles: DNS server, NTP server
```

3. 在您在前面的步骤中指定的主机上重启 **named-pkcs11** 服务：

```
[root@idmserver-01 ~]# systemctl restart named-pkcs11
```

对您要为其分配 DNS 位置的其他 IdM 服务器重复这些步骤。

33.9.5. 将客户端配置为在同一位置中使用 IdM 服务器

IdM 服务器被分配给 DNS 位置，如 第 33.9.4 节 “将 IdM 服务器分配给 DNS 位置” 所述。现在，您可以将客户端配置为使用与 IdM 服务器位于同一位置的 DNS 服务器：

- 如果 DHCP 服务器为客户端分配 DNS 服务器 IP 地址，请配置 DHCP 服务。有关在您的 DHCP 服务中分配 DNS 服务器的详情，请查看 DHCP 服务文档。
- 如果您的客户端没有从 DHCP 服务器接收 DNS 服务器 IP 地址，请手动设置客户端网络配置中的 IP 地址。有关在 Red Hat Enterprise Linux 中配置网络的详情，请参考《红帽企业 Linux 网络指南》中的"配置网络连接设置 "章节。

注意

如果您将客户端配置为使用分配给不同位置的 DNS 服务器，客户端会联系两个位置的 IdM 服务器。

例 33.12. 根据客户端位置的不同名称服务器条目

以下示例显示了位于不同位置的客户端的 /etc/resolv.conf 文件中的不同名称服务器条目：

布拉格中的客户端：

```
nameserver 10.10.0.1
nameserver 10.10.0.2
```

拉丁美洲客户：

```
nameserver 10.50.0.1
nameserver 10.50.0.3
```

Oslo 中的客户端：

```
nameserver 10.30.0.1
```

林中的客户端：

```
nameserver 10.30.0.1
```

如果每个 DNS 服务器都被分配给 IdM 中的一个位置，客户端将使用其位置中的 IdM 服务器。

33.10. 使用外部 DNS 时系统性更新 DNS 记录

在使用外部 DNS 时，身份管理不会在拓扑更改后自动更新 DNS 记录。以下流程解释了如何系统地更新由外部 DNS 服务管理的 DNS 记录，从而减少了手动 DNS 更新的需求。

有关基本概述请查看 第 33.10.1 节 “在身份管理中更新外部 DNS”。

有关流程和示例，请参阅：

- 第 33.10.2 节 “GUI：更新外部 DNS 记录” 如果您使用 GUI 管理外部 DNS 记录
- 第 33.10.3 节 “命令行：使用 nsupdate更新外部 DNS 记录” 如果您使用 nsupdate 工具管理外部 DNS 记录

33.10.1. 在身份管理中更新外部 DNS

更新 DNS 记录会删除旧的或无效的 DNS 记录并添加新记录。

您必须在拓扑更改后更新 DNS 记录，例如：

- 安装或卸载副本后
- 在身份管理服务器中安装 CA、DNS、KRA 或 Active Directory 信任后

33.10.2. GUI：更新外部 DNS 记录

1. 显示您必须更新的记录。使用 ipa dns-update-system-records --dry-run 命令。

```
$ ipa dns-update-system-records --dry-run
 IPA DNS records:
  _kerberos-master._tcp.example.com. 86400 IN SRV 0 100 88 ipa.example.com.
  _kerberos-master._udp.example.com. 86400 IN SRV 0 100 88 ipa.example.com.
[... output truncated ...]
```

2. 使用外部 DNS GUI 更新记录。

33.10.3. 命令行：使用 nsupdate更新外部 DNS 记录

这部分论述了如何使用 nsupdate 工具手动更新外部 DNS 记录。您也可以使用脚本中本节中的命令来自动化此过程。

使用 nsupdate的 DNS 记录生成文件

1. 使用带有 --out 选项的 ipa dns-update-system-records --dry-run 命令。选项指定要生成的文件路径：

```
$ ipa dns-update-system-records --dry-run --out dns_records_file.nsupdate
 IPA DNS records:
  _kerberos-master._tcp.example.com. 86400 IN SRV 0 100 88 ipa.example.com.
  _kerberos-master._udp.example.com. 86400 IN SRV 0 100 88 ipa.example.com.
[... output truncated ...]
```

生成的文件包含 nsupdate 工具接受的格式所需的 DNS 记录。

2. 生成的记录依赖于：

- 自动检测要更新记录的区域
- 自动检测区域的权威服务器

如果您使用 atypical DNS 设置，或者缺少区域委派，nsupdate 可能无法找到正确的区域和服务器。在这种情况下，在生成的文件开头添加以下选项：

- server 指定 nsupdate 将记录发送到的权威 DNS 服务器的服务器名称或端口
- zone 指定 nsupdate 放置记录的区域名称

Example:

```
$ cat dns_records_file.nsupdate
zone example.com.
server 192.0.2.1
; IPA DNS records
update delete _kerberos-master._tcp.example.com. SRV
update add _kerberos-master._tcp.example.com. 86400 IN SRV 0 100 88 ipa.example.com.
[... output truncated ...]
```

将动态 DNS 更新请求提交到名称服务器

使用 nsupdate 发送请求时，请确保正确保护它。您可以使用以下机制保护请求：

事务签名(TSIG)协议

TSIG 可让您将 nsupdate 与共享密钥搭配使用。请参阅 过程 33.1, “使用 TSIG 发送 nsupdate 请求安全”。

TSIG 的 GSS 算法(GSS-TSIG)

GSS-TSIG 使用 GSS-API 接口来获取 secret TSIG 密钥。GSS-TSIG 是 TSIG 协议的扩展。请

查看 过程 33.2, “使用 GSS-TSIG 发送 nsupdate 请求安全”

过程 33.1. 使用 TSIG 发送 nsupdate 请求安全

1. 请确定您满足以下先决条件：

   - 您的 DNS 服务器必须配置为 TSIG。请参阅以下服务器配置示例：BIND、PowerDNS
   - DNS 服务器及其客户端都必须具有共享密钥。

2. 运行 nsupdate，并使用以下选项之一提供共享 secret：

   - -k 提供 TSIG 身份验证密钥：

     ```
     $ nsupdate -k tsig_key.file dns_records_file.nsupdate
     ```

   - -Y 从密钥名称和 Base64 编码的共享 secret 中生成签名：

     ```
     $ nsupdate -y algorithm:keyname:secret dns_records_file.nsupdate
     ```

过程 33.2. 使用 GSS-TSIG 发送 nsupdate 请求安全

1. 请确定您满足以下先决条件：

   - 您的 DNS 服务器必须配置 GSS-TSIG。请参阅以下服务器配置示例：BIND、PowerDNS、Windows DNS.

   注意

   此流程假定 Kerberos V5 协议用作 GSS-API 的技术。

2. 要提交 DNS 更新请求，请使用允许更新记录的主体进行身份验证，并使用 -g 选项运行 nsupdate 来启用 GSS-TSIG 模式：

```
$ kinit principal_allowed_to_update_records@REALM
$ nsupdate -g dns_records_file.nsupdate
```

其它资源

- **nsupdate(8) man page**
- **RFC 2845 描述 TSIG 协议**
- **RFC 3645 描述了 GSS-TSIG 算法**

33.11. 在现有服务器中安装 DNS 服务

可以将 DNS 服务安装到最初没有安装的 IdM 服务器中。为此，请确保安装了 ipa-server-dns 软件包，然后使用 ipa-dns-install 工具。

使用 ipa-dns-install 配置 DNS 服务遵循与使用 ipa-server-install 工具安装 DNS 相同的原则，如第 2.3.3 节 “使用集成的 DNS 安装服务器” 所述。

有关 ipa-dns-install 的详情，请参考 ipa-dns-install(1) man page。

33.11.1. 设置其他名称服务器

33.11.1.1. 设置其他名称服务器

IdM 将新配置的 IdM DNS 服务器添加到 /etc/resolv.conf 文件中的 DNS 服务器列表中。建议手动将其他 DNS 服务器添加为备份服务器，以防 IdM 服务器不可用。例如：

```
search example.com

; the IdM server
nameserver 192.0.2.1

; backup DNS servers
nameserver 198.51.100.1
nameserver 198.51.100.2
```

有关配置 /etc/resolv.conf 的详情，请查看 resolv.conf(5) man page。

---

[3] 有关 GSS-TSIG 的更多信息，请参阅 RFC 3545。

[4] 有关 RFC 3007 的完整文本，请参阅 http://tools.ietf.org/html/rfc3007

[5] 如需更多信息，请参阅 BIND 9 配置参考。

第 34 章 使用自动挂载

自动挂载是一种在多个系统之间管理、组织和访问目录的方法。每当请求访问该目录时，自动挂载 自动挂载目录。这在 IdM 域中正常工作良好，因为它允许域中客户端的目录易于共享。这在用户主目录中尤为重要，请参阅 第 11.1 节 “设置用户主目录”。

在 IdM 中，自动挂载可用于内部 LDAP 目录，以及 DNS 服务（如果已配置）。

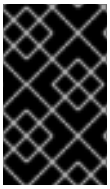
34.1. 关于自动挂载和 IDM

自动挂载提供了与目录组织方式一致的结构。每个目录称为 挂载点 或 密钥。将多个键分组在一起创建一个 映射，映射根据其物理或概念性 位置 相关联。

automount 的基本配置文件是 /etc 目录中的 auto.master 文件。如有必要，可以在单独的服务器位置有多个 auto.master 配置文件。

当在服务器上配置 autofs 工具且服务器是 IdM 域中的客户端时，自动挂载的所有配置信息都存储在 IdM 目录中。autofs 配置不存储在单独的文本文件中，而是将包含映射、位置和密钥存储为 LDAP 条目。例如，默认映射文件 auto.master 存储为：

```
dn: automountmapname=auto.master,cn=default,cn=automount,dc=example,dc=com
objectClass: automountMap
objectClass: top
automountMapName: auto.master
```



重要

身份管理可用于现有的 autofs 部署，但不设置或配置 autofs 本身。

每个新位置都作为容器条目添加到 cn=automount,dc=example,dc=com 下，每个映射则存储在该位置下。

与其他 IdM 域服务一样，自动挂载可以原生地用于 IdM。自动挂载配置可由 IdM 工具管理：

- 用于位置的 ipa automountlocation the 命令，

- 用于 直接和间接映射 的 ipa automountmap the command ,
- 用于 密钥的 ipa automountkey the 命令。

要使自动挂载在 IdM 域中工作，必须将 NFS 服务器配置为 IdM 客户端。Red Hat Enterprise Linux Storage Administration Guide 中介绍了配置 NFS 本身。

34.2. 配置自动挂载

在身份管理中，配置自动挂载条目（如位置和映射）需要现有的 autofs/NFS 服务器。创建自动挂载条目不会创建底层 autofs 配置。autofs 可以使用 LDAP 或 SSSD 作为数据存储手动配置，也可以自动配置。

注意

在更改自动挂载配置前，请测试至少一个用户，可以成功从命令行挂载其 /home 目录。确保 NFS 正常工作，以便稍后对潜在的 IdM 自动挂载配置错误进行故障排除。

34.2.1. 自动配置 NFS

在将系统配置为 IdM 客户端后，其中包含配置为域客户端一部分的 IdM 服务器和副本，可将 autofs 配置为使用 IdM 域作为其 NFS 域，并启用 autofs 服务。

默认情况下，ipa-client-automount 工具自动配置 NFS 配置文件 /etc/sysconfig/nfs 和 /etc/idmapd.conf。它还将 SSSD 配置为管理 NFS 的凭据。如果 ipa-client-automount 命令不带任何选项运行，它会运行 DNS 发现扫描来识别可用的 IdM 服务器，并创建一个名为 default 的默认位置。

```
[root@ipa-server ~]# ipa-client-automount
Searching for IPA server...
IPA server: DNS discovery
Location: default
Continue to configure the system with these values? [no]: yes
Configured /etc/nsswitch.conf
Configured /etc/sysconfig/nfs
Configured /etc/idmapd.conf
Started rpcidmapd
Started rpcgssd
Restarting sssd, waiting for it to become available.
Started autofs
```

可以使用 IdM 服务器并创建一个自动挂载位置，而不是默认位置：

```
[root@server ~]# ipa-client-automount --server=ipaserver.example.com --location=boston
```

除了设置 NFS 外，ipa-client-automount 工具将 SSSD 配置为缓存自动挂载映射，以防外部 IdM 存储无法访问。配置 SSSD 有两个方面：

- 它将服务配置信息添加到 SSSD 配置中。IdM 域条目被授予 autofs 提供程序和挂载位置的设置。

  ```
  autofs_provider = ipa
  ipa_automount_location = default
  ```

  NFS 被添加到支持的服务列表中（服务 = nss、pam、autofs...）并给出空白配置条目 ([autofs])。

- Name Service Switch(NSS)服务信息已更新，以首先检查 SSSD 是否有自动挂载信息，然后再检查本地文件。

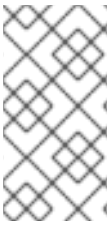
  ```
  automount: sss files
  ```

可能存在一些实例，如高安全性环境，客户端不适合缓存自动挂载映射。在这种情况下，ipa-client-automount 命令可以使用 --no-sssd 选项运行，该选项会更改所有必需的 NFS 配置文件，但不会更改 SSSD 配置。

```
[root@server ~]# ipa-client-automount --no-sssd
```

如果使用 --no-sssd，由 ipa-client-automount 更新的配置文件列表会有所不同：

- 该命令更新 /etc/sysconfig/autofs，而不是 /etc/sysconfig/nfs。
- 命令使用 IdM LDAP 配置 /etc/autofs_ldap_auth.conf。
- 命令将 /etc/nsswitch.conf 配置为使用 LDAP 服务进行自动挂载映射。



注意

ipa-client-automount 命令只能运行一次。如果配置中存在错误，需要手动编辑配置文件。

34.2.2. 手动配置 autofs 以使用 SSSD 和身份管理

1. 编辑 /etc/sysconfig/autofs 文件，以指定 autofs 搜索的模式属性：

```
#
# Other common LDAP naming
#
MAP_OBJECT_CLASS="automountMap"
ENTRY_OBJECT_CLASS="automount"
MAP_ATTRIBUTE="automountMapName"
ENTRY_ATTRIBUTE="automountKey"
VALUE_ATTRIBUTE="automountInformation"
```

2. 指定 LDAP 配置。执行此操作有两种方法：最简单的方法是让自动挂载服务自行发现 LDAP 服务器和位置：

```
LDAP_URI="ldap:///dc=example,dc=com"
```

或者，明确设置要使用的 LDAP 服务器，以及用于 LDAP 搜索的基本 DN：

```
LDAP_URI="ldap://ipa.example.com"
SEARCH_BASE="cn=location,cn=automount,dc=example,dc=com"
```

注意

location 的默认值为 default。如果添加了其他位置(第 34.5 节 “配置位置”)，那么客户端可以被指向使用这些位置。

3. 编辑 /etc/autofs_ldap_auth.conf 文件，以便 autofs 允许客户端通过 IdM LDAP 服务器进行身份验证。

- 将 authrequired 更改为 yes。
-

将 NFS 客户端服务器的 Kerberos 主机主体设置为 host/fqdn@REALM。主体名称用于连接 IdM 目录，来作为 GSS 客户端身份验证的一部分。

```
<autofs_ldap_sasl_conf
    usetls="no"
    tlsrequired="no"
    authrequired="yes"
    authtype="GSSAPI"
    clientprinc="host/server.example.com@EXAMPLE.COM"
    />
```

如有必要，请运行 klist -k 来获取确切的主机主体信息。

4. 将 autofs 配置为 SSSD 管理的服务之一。

a. 打开 SSSD 配置文件。

```
[root@server ~]# vim /etc/sssd/sssd.conf
```

b. 将 autofs 服务添加到 SSSD 处理的服务列表中。

```
[sssd]
services = nss,pam,autofs
```

c. 创建一个新的 [autofs] 部分。这可以留空；autofs 服务的默认设置可与大多数基础架构配合使用。

```
[nss]

[pam]

[sudo]

[autofs]

[ssh]

[pac]
```

d.

*（可选）为 **autofs** 条目设置搜索基础。默认情况下，这是 **LDAP** 搜索库，但可以在 **ldap_autofs_search_base** 参数中指定子树。*

```
[domain/EXAMPLE]
...
ldap_search_base = "dc=example,dc=com"
ldap_autofs_search_base = "ou=automount,dc=example,dc=com"
```

5.

*重启 **SSSD**：*

```
[root@server ~]# systemctl restart sssd.service
```

6.

*检查 **/etc/nsswitch.conf** 文件，以便 **SSSD** 被列为自动挂载配置的源：*

```
automount: sss files
```

7.

*重启 **autofs**：*

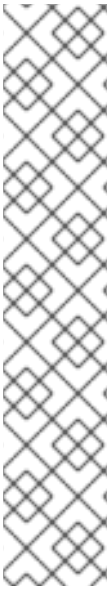
```
[root@server ~]# systemctl restart autofs.service
```

8.

*通过列出用户的 **/home** 目录来测试配置：*

```
[root@server ~]# ls /home/userName
```

*如果这没有挂载远程文件系统，请检查 **/var/log/messages** 文件是否有错误。如有必要，通过将 **LOGGING** 参数设置为 **debug** 来提高 **/etc/sysconfig/autofs** 文件中的 **debug** 级别。*



注意

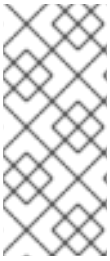
如果自动挂载存在问题，则跨引用 IdM 实例的 389 目录服务器访问日志的自动挂载尝试，这将显示尝试的访问、用户和搜索基础。

还可以通过 debug 登录在前台运行自动挂载。

```
automount -f -d
```

这将直接打印调试日志信息，而无需使用自动挂载的日志对 LDAP 访问日志进行交叉检查。

34.2.3. 在 Solaris 中配置自动挂载



注意

Solaris 对 autofs 配置使用不同的模式，与身份管理使用的架构不同。身份管理使用 2307bis 风格的自动挂载模式，该模式是为 389 目录服务器定义的（在 IdM 的内部目录服务器实例中使用）。

1. 如果 NFS 服务器在 Red Hat Enterprise Linux 上运行，请在 NFSv3 的 Solaris 机器上指定最大支持版本。编辑 /etc/default/nfs 文件并设置以下参数：

```
NFS_CLIENT_VERSMAX=3
```

2. 使用 ldapclient 命令将主机配置为使用 LDAP：

```
ldapclient -v manual -a authenticationMethod=none
   -a defaultSearchBase=dc=example,dc=com
   -a defaultServerList=ipa.example.com
   -a serviceSearchDescriptor=passwd:cn=users,cn=accounts,dc=example,dc=com
   -a serviceSearchDescriptor=group:cn=groups,cn=compat,dc=example,dc=com
   -a
serviceSearchDescriptor=auto_master:automountMapName=auto.master,cn=location,cn=aut
omount,dc=example,dc=com?one
   -a
serviceSearchDescriptor=auto_home:automountMapName=auto_home,cn=location,cn=auto
mount,dc=example,dc=com?one
   -a objectClassMap=shadow:shadowAccount=posixAccount
   -a searchTimelimit=15
   -a bindTimeLimit=5
```

3. *启用 自动挂载 ：*

```
# svcadm enable svc:/system/filesystem/autofs
```

4. *测试配置。*

a. *检查 LDAP 配置：*

```
# ldapclient -l auto_master

dn:
automountkey=/home,automountmapname=auto.master,cn=location,cn=automount
,dc=example,dc=com
objectClass: automount
objectClass: top
automountKey: /home
automountInformation: auto.home
```

b. *列出用户的 /home 目录：*

```
# ls /home/userName
```

34.3. 设置 KERBEROS 感知 NFS 服务器

1. *如果您的任何 NFS 客户端只支持弱加密，如 Red Hat Enterprise Linux 5 客户端：*

a. *更新 IdM 服务器 Kerberos 配置，以启用弱 des-cbc-crc 加密类型：*

```
$ ldapmodify -x -D "cn=directory manager" -w password -h ipaserver.example.com -p
389

dn: cn=REALM_NAME,cn=kerberos,dc=example,dc=com
```

```
changetype: modify
add: krbSupportedEncSaltTypes
krbSupportedEncSaltTypes: des-cbc-crc:normal
-
add: krbSupportedEncSaltTypes
krbSupportedEncSaltTypes: des-cbc-crc:special
-
add: krbDefaultEncSaltTypes
krbDefaultEncSaltTypes: des-cbc-crc:special
```

b. **在 NFS 服务器中，在 NFS 服务器的 /etc/krb5.conf 文件中添加以下条目以启用弱加密支持：**

```
allow_weak_crypto = true
```

2. **获取Kerberos ticket：**

```
[root@nfs-server ~]# kinit admin
```

3. **如果 NFS 主机计算机尚未作为客户端添加到 IdM 域，请创建主机条目。请参阅 第 12.3 节 “添加主机条目”。**

4. **创建 NFS 服务条目：**

```
[root@nfs-server ~]# ipa service-add nfs/nfs-server.example.com
```

如需更多信息，请参阅 第 16.1 节 “添加和编辑服务条目和密钥选项卡”。

5. **使用以下 ipa-getkeytab 命令为 NFS 服务器检索 NFS 服务 keytab，该命令可将密钥保存在 /etc/krb5.keytab 文件中：**

```
[root@nfs-server ~]# ipa-getkeytab -s ipaserver.example.com -p nfs/nfs-server.example.com
-k /etc/krb5.keytab
```

如果您的任何 NFS 客户端只支持弱加密，还会将 -e des-cbc-crc 选项传递给 命令，以请求 DES 加密 keytab。

6. **通过检查服务条目，在 IdM 中使用 keytab 验证 NFS 服务是否已正确配置：**

```
[root@nfs-server ~]# ipa service-show nfs/nfs-server.example.com
  Principal name: nfs/nfs-server.example.com@IDM.EXAMPLE.COM
  Principal alias: nfs/nfs-server.example.com@IDM.EXAMPLE.COM
  Keytab: True
  Managed by: nfs-server.example.com
```

7. **安装 nfs-utils 软件包：**

```
[root@nfs-server ~]# yum install nfs-utils
```

8. **运行 ipa-client-automount 工具来配置 NFS 设置：**

```
[root@nfs-server ~] ipa-client-automount
Searching for IPA server...
IPA server: DNS discovery
Location: default
Continue to configure the system with these values? [no]: yes
Configured /etc/sysconfig/nfs
Configured /etc/idmapd.conf
Started rpcidmapd
Started rpcgssd
Restarting sssd, waiting for it to become available.
Started autofs
```

默认情况下，这个命令启用安全 NFS 并将 /etc/idmapd.conf 文件中的 Domain 参数设置为 IdM DNS 域。如果您使用不同的域，请使用 --idmap-domain domain_name 参数指定它。

9. **将 nfs-idmapd 服务配置为在系统引导时自动启动：**

```
# systemctl enable nfs-idmapd
```

10. **编辑 /etc/exports 文件并使用 krb5p Kerberos 安全设置添加共享：**

```
/export  *(rw,sec=krb5:krb5i:krb5p)
/home  *(rw,sec=krb5:krb5i:krb5p)
```

这个示例在启用了 Kerberos 身份验证时以读写模式共享 /export 和 /home 目录。

11. **重新导出共享目录：**

```
[root@nfs-server ~]# exportfs -rav
```

12. （可选）将 NFS 服务器配置为 NFS 客户端。请参阅 第 34.4 节 “设置 Kerberos 感知 NFS 客户端”。

34.4. 设置 KERBEROS 感知 NFS 客户端

1. 如果 NFS 客户端只支持弱加密（如 Red Hat Enterprise Linux 5 客户端），请在服务器的 /etc/krb5.conf 文件中设置以下条目以允许弱加密：

```
allow_weak_crypto = true
```

2. 如果 NFS 客户端没有注册为 IdM 域中的客户端，请设置所需的主机条目，如 第 12.3 节 “添加主机条目” 所述。

3. 安装 nfs-utils 软件包：

```
[root@nfs-client ~]# yum install nfs-utils
```

4. 在运行 IdM 工具前获取 Kerberos 票据。

```
[root@nfs-client ~]# kinit admin
```

5. 运行 ipa-client-automount 工具来配置 NFS 设置：

```
[root@nfs-client ~] ipa-client-automount
Searching for IPA server...
IPA server: DNS discovery
Location: default
Continue to configure the system with these values? [no]: yes
Configured /etc/sysconfig/nfs
Configured /etc/idmapd.conf
Started rpcidmapd
Started rpcgssd
Restarting sssd, waiting for it to become available.
Started autofs
```

默认情况下，这会在 /etc/sysconfig/nfs 文件中启用安全 NFS，并在 /etc/idmapd.conf 文件中的 Domain 参数中设置 IdM DNS 域。

6. 将服务配置为在系统引导时自动启动：

```
[root@nfs-client ~]# systemctl enable rpc-gssd.service
[root@nfs-client ~]# systemctl enable rpcbind.service
```

7. 在 /etc/fstab 文件中添加下列条目，以便在系统引导时从 nfs-server.example.com 主机挂载 NFS 共享：

```
nfs-server.example.com:/export  /mnt          nfs4  sec=krb5p,rw
nfs-server.example.com:/home    /home  nfs4  sec=krb5p,rw
```

这些设置将 Red Hat Enterprise Linux 配置为将 /export 共享挂载到 /mnt，将 /home 共享挂载到 /home 目录。

8. 如果挂载点不存在，则进行创建：

```
# mkdir -p /mnt/
# mkdir -p /home
```

9. 挂载 NFS 共享：

```
[root@nfs-client ~]# mount /mnt/
[root@nfs-client ~]# mount /home
```

命令使用 /etc/fstab 条目中的信息。

10. 配置 SSSD 以续订 Kerberos 票据：

    a. 在 /etc/sssd/sssd.conf 文件的 IdM 域部分中设置以下参数，以配置 SSSD 以自动更新票据：

```
[domain/EXAMPLE.COM]
...
krb5_renewable_lifetime = 50d
krb5_renew_interval = 3600
```

b.

重启 SSSD：

```
[root@nfs-client ~]# systemctl restart sssd
```

重要

pam_oddjob_mkhomedir 模块不支持在 NFS 共享上自动创建主目录。因此，您必须在包含主目录的共享根目录中手动在服务器上创建主目录。

34.5. 配置位置

位置是一组映射，全部存储在 auto.master 中，并且位置可以存储多个映射。位置条目仅充当映射条目的容器；它本身并不是自动挂载配置。

重要

身份管理没有设置或配置 autofs。这必须单独完成。身份管理可用于现有的 autofs 部署。

34.5.1. 通过 Web UI 配置位置

1.

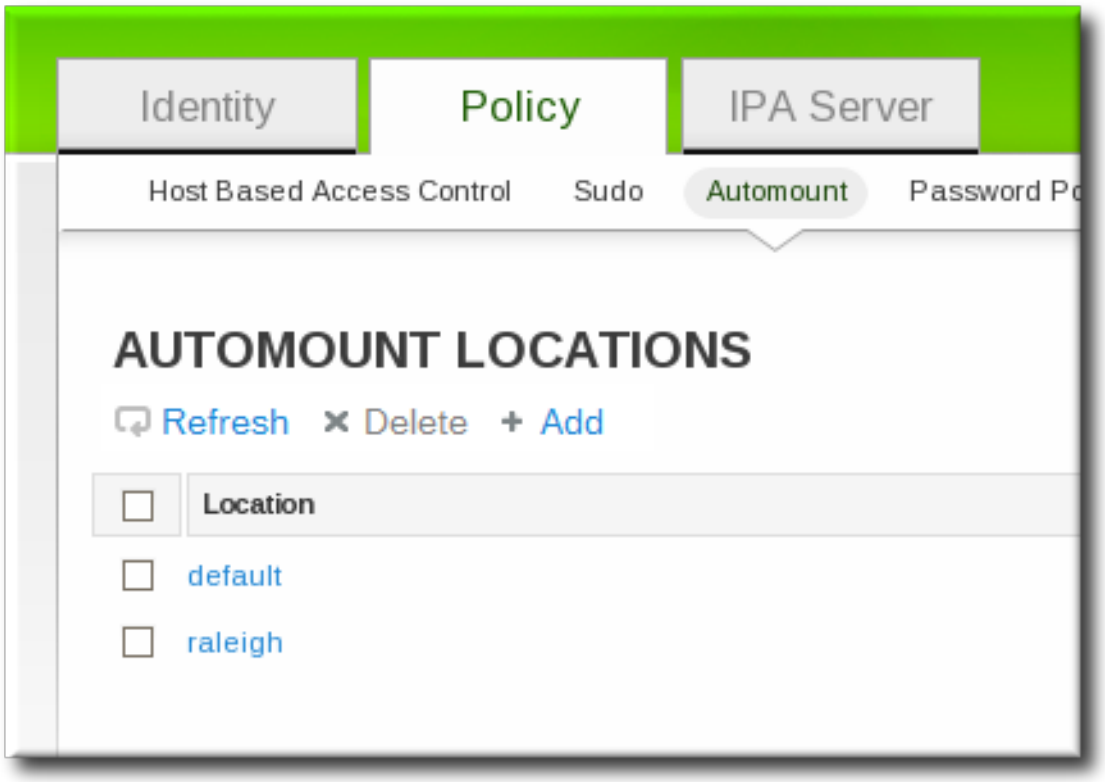
点 Policy 选项卡。

2.

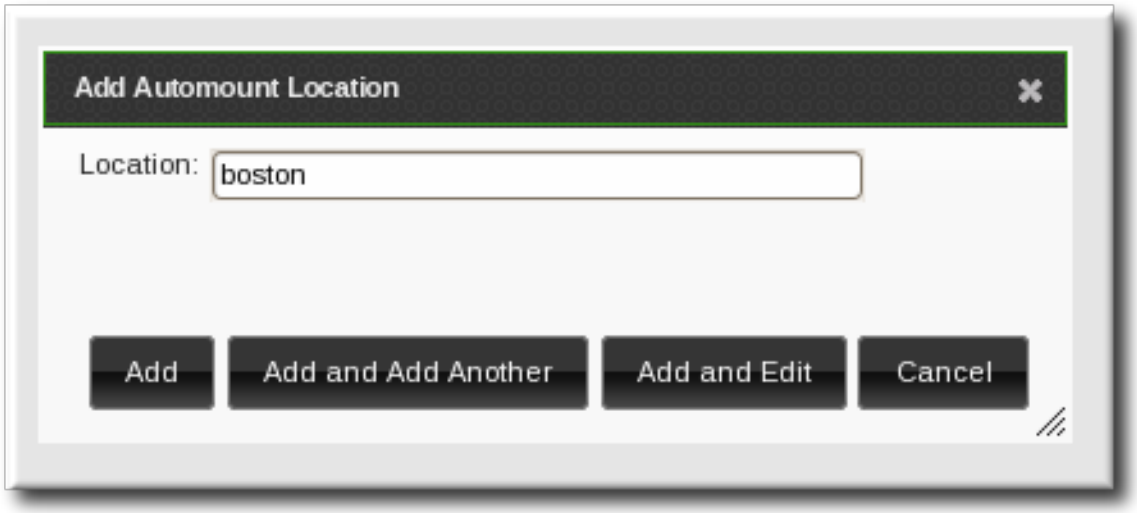
点 Automount 子选项卡。

3.

单击自动挂载位置列表顶部的 Add 链接。



4. *输入新位置的名称。*



5. *单击* **Add and Edit** *按钮，以转至新位置的映射配置。按照* *第 34.6.1.1 节 “使用 Web UI 配置直接映射”* *和* *第 34.6.2.1 节 “从 Web UI 配置间接映射”* *所述创建映射。*

34.5.2. 通过命令行配置位置

要创建映射，可使用 **automountlocation-add** *并指定位置名称。*

```
$ ipa automountlocation-add location
```

例如：

```
$ ipa automountlocation-add raleigh
----------------------------------
Added automount location "raleigh"
----------------------------------
  Location: raleigh
```

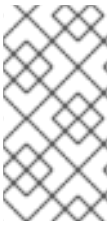
创建新位置时，会自动为其创建两个映射，auto.master 和 auto.direct。auto.master 是位置的所有自动挂载映射的根映射。auto.direct 是直接挂载的默认映射，并挂载到 /- 上。

要查看为位置配置的所有映射，就像它们部署到文件系统中一样，请使用 automountlocation-tofiles 命令：

```
$ ipa automountlocation-tofiles raleigh
/etc/auto.master:
/-      /etc/auto.direct
---------------------------
/etc/auto.direct:
```

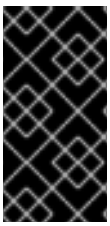
34.6. 配置映射

配置映射不仅创建映射，它通过密钥将挂载点关联，并分配访问目录时应使用的挂载选项。IdM 支持直接和间接映射。



注意

不同的客户端可以使用不同的映射集。映射集使用树结构，因此无法 在位置之间共享映射。



重要

身份管理没有设置或配置 autofs。这必须单独完成。身份管理可用于现有的 autofs 部署。

34.6.1. 配置直接映射

直接映射定义文件挂载点的确切位置（即绝对路径）。在位置条目中，可通过前面的正斜杠标识直接映射：

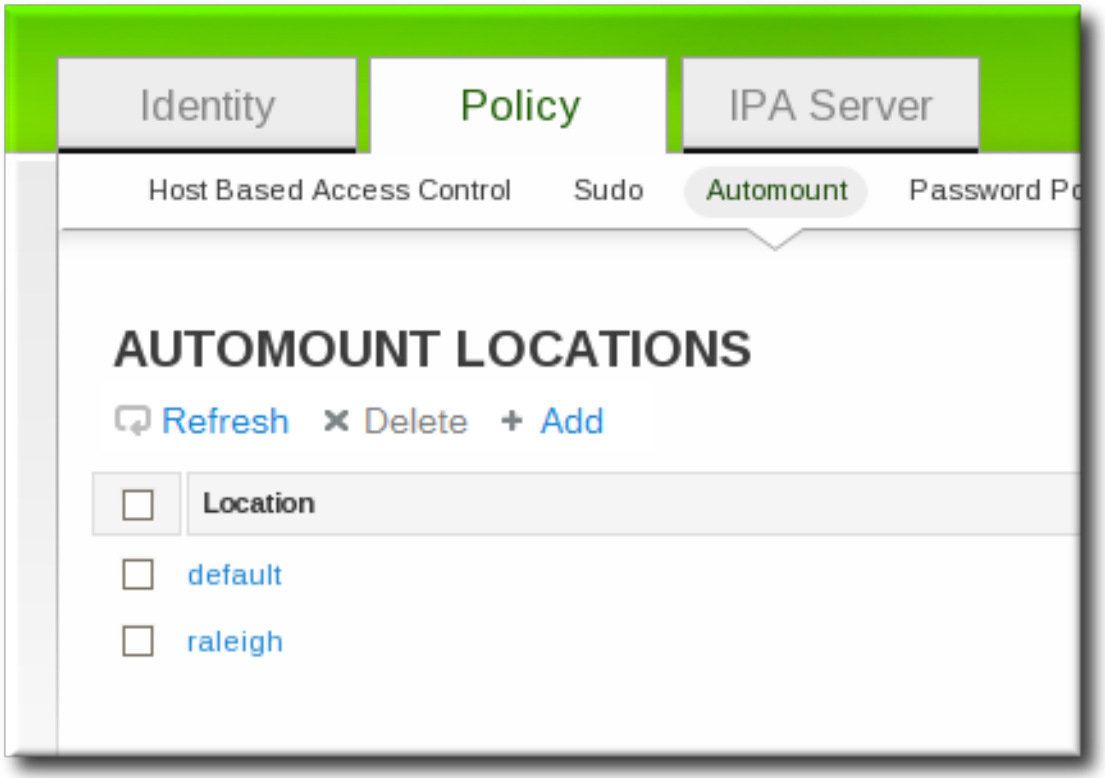
```
-------------------------
/etc/auto.direct:
/shared/man server.example.com:/shared/man
```

34.6.1.1. 使用 Web UI 配置直接映射

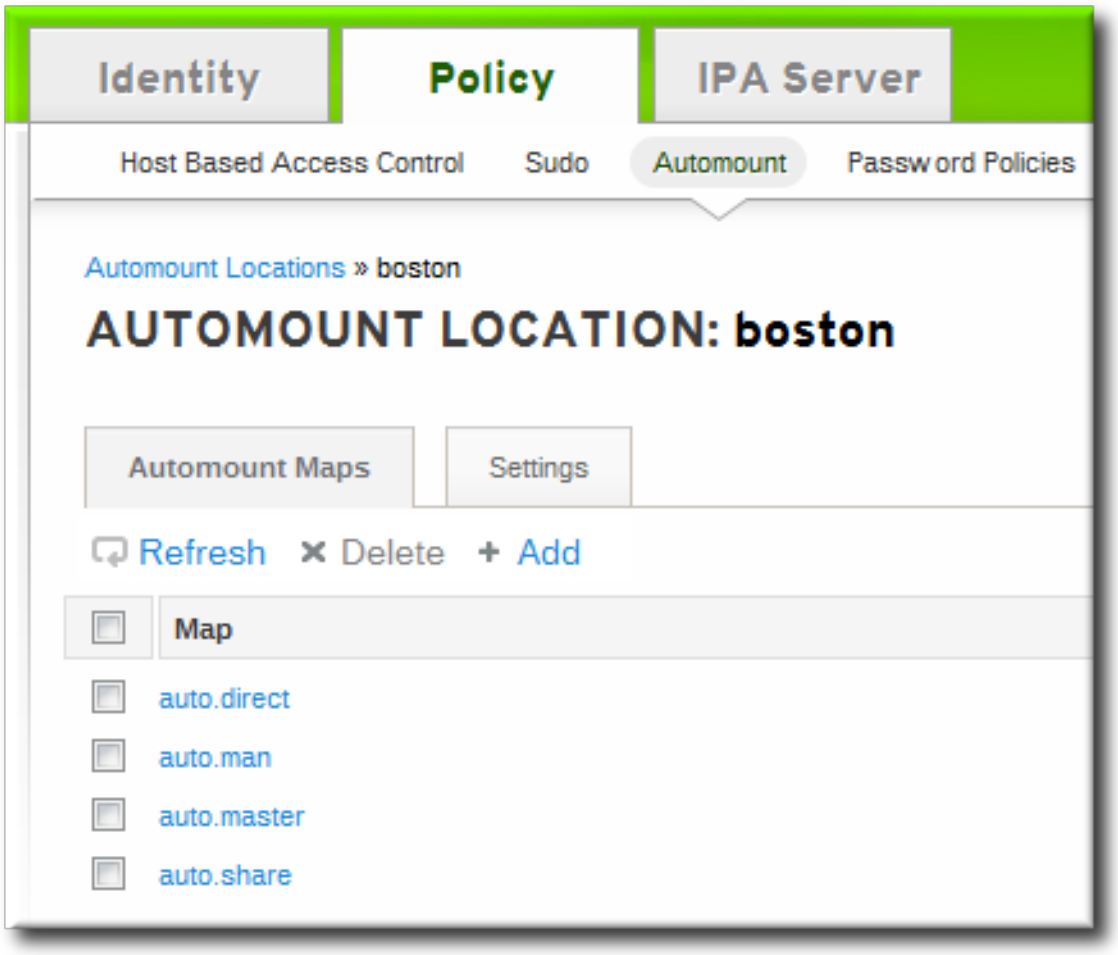
1. 点 Policy 选项卡。

2. 点 Automount 子选项卡。

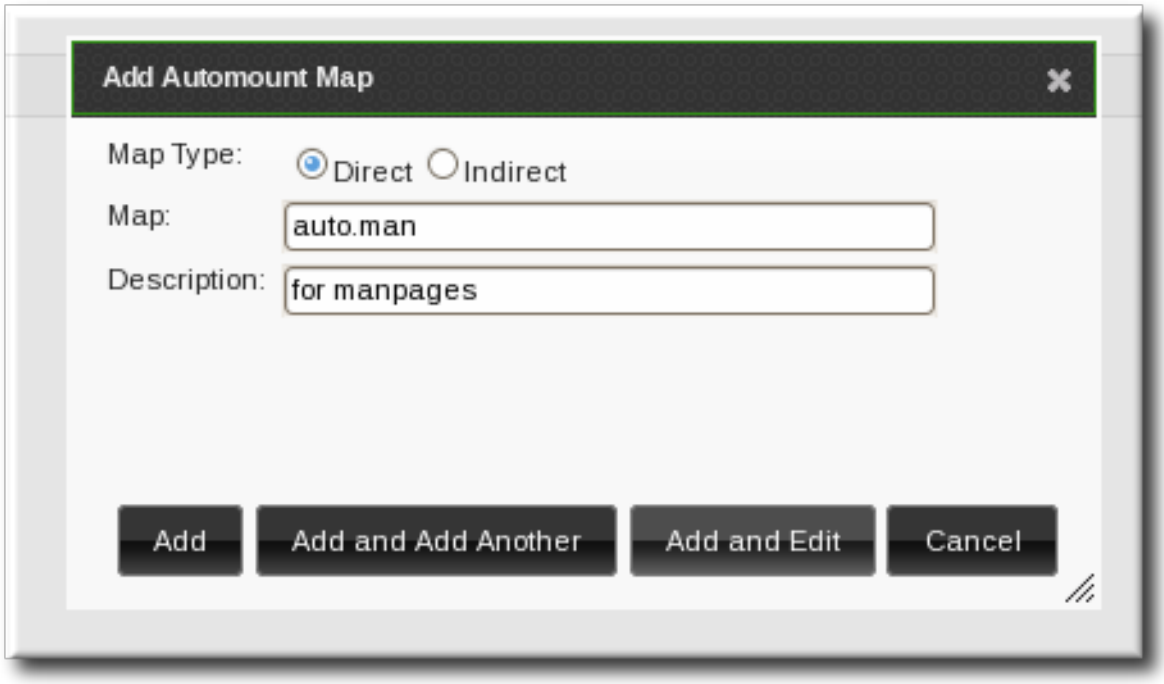
3. 单击要向其添加映射的自动挂载位置的名称。



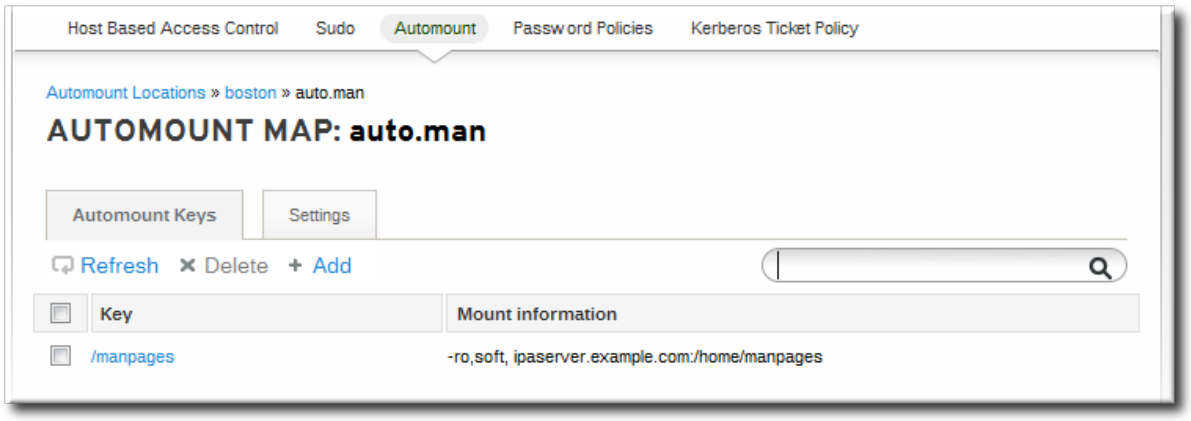
4. 在 Automount Maps 选项卡中，点 + Add 链接来创建新映射。



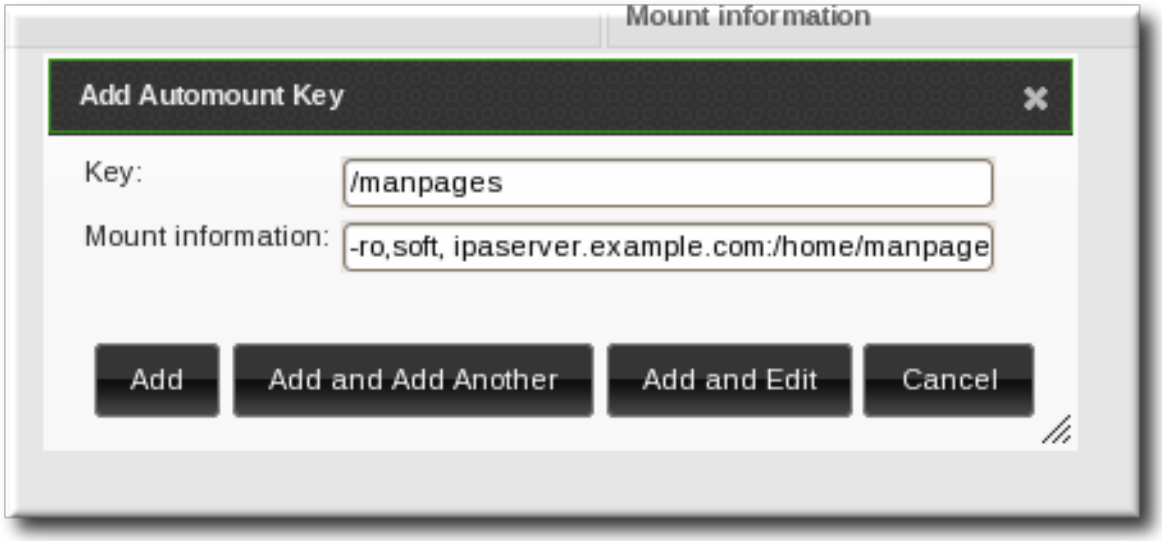
5. *在弹出窗口中，选择 **Direct** 单选按钮并输入新映射的名称。*



6. *在 **Automount Keys** 选项卡中，点 **+ Add** 链接为映射创建新密钥。*



7. *输入挂载点。键在密钥名称中定义实际的挂载点。Info 字段设置目录的网络位置，以及要使用的任何 挂载选项。*



8. *点 Add 按钮保存新密钥。*

34.6.1.2. 从命令行配置直接映射

键定义实际挂载点（在密钥名称中）和任何选项。map 是基于其密钥格式的直接或间接映射。

每个位置都使用 auto.direct 项目创建。最简单的配置是通过在现有直接映射条目中添加自动挂载密钥来定义直接映射。也可以创建不同的直接映射条目。

将直接映射的密钥添加到位置的 auto.direct 文件。--key 选项标识挂载点，-- info 提供目录的网络位置，以及要使用的任何 挂载选项。例如：

```
$ ipa automountkey-add raleigh auto.direct --key=/share --
```

```
info="ro,soft,ipaserver.example.com:/home/share"
  Key: /share
  Mount information: ro,soft,ipaserver.example.com:/home/share
```

mount manpage 中介绍了挂载选项 http://linux.die.net/man/8/mount。

在 Solaris 上，使用 ldapclient 命令添加直接映射和密钥直接添加 LDAP 条目：

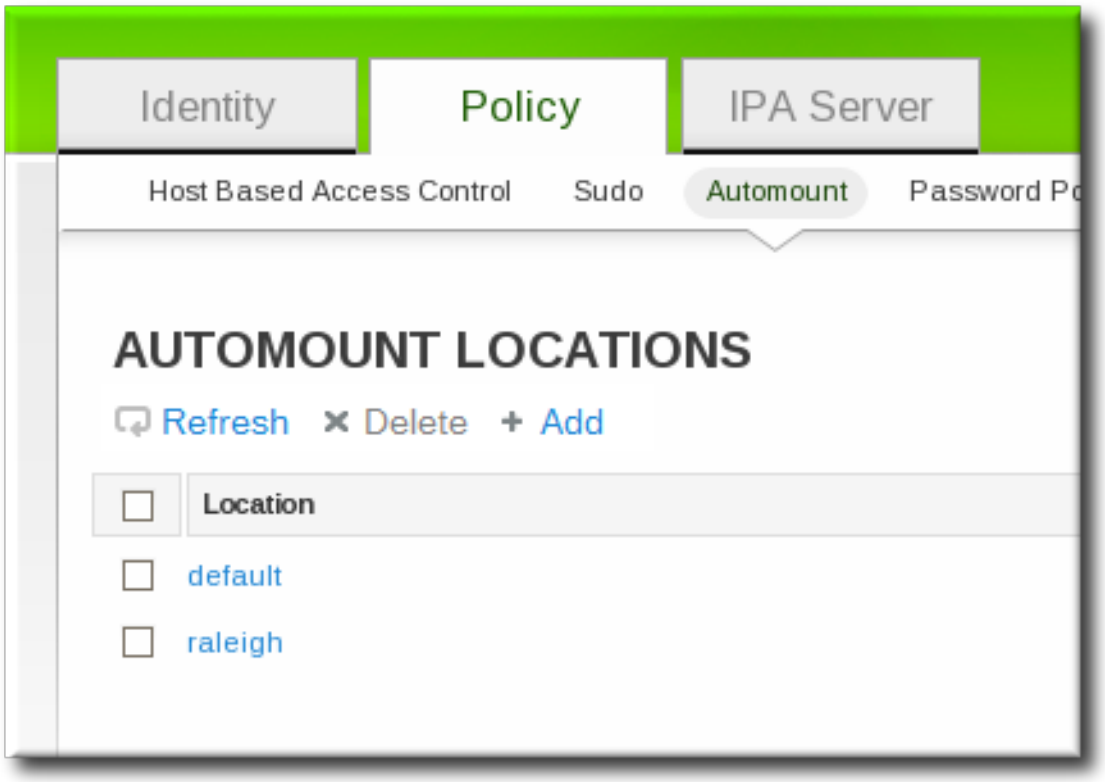
```
ldapclient -a
serviceSearchDescriptor=auto_direct:automountMapName=auto.direct,cn=location,cn=automount,dc
=example,dc=com?one
```

34.6.2. 配置间接映射

间接映射实质上指定映射的相对路径。父条目为所有间接映射设置基础目录。间接映射键将设置一个子目录；每当加载间接映射位置时，该密钥将附加到基础目录中。例如，如果主目录为 /docs，且键是 man，则映射为 /docs/man。

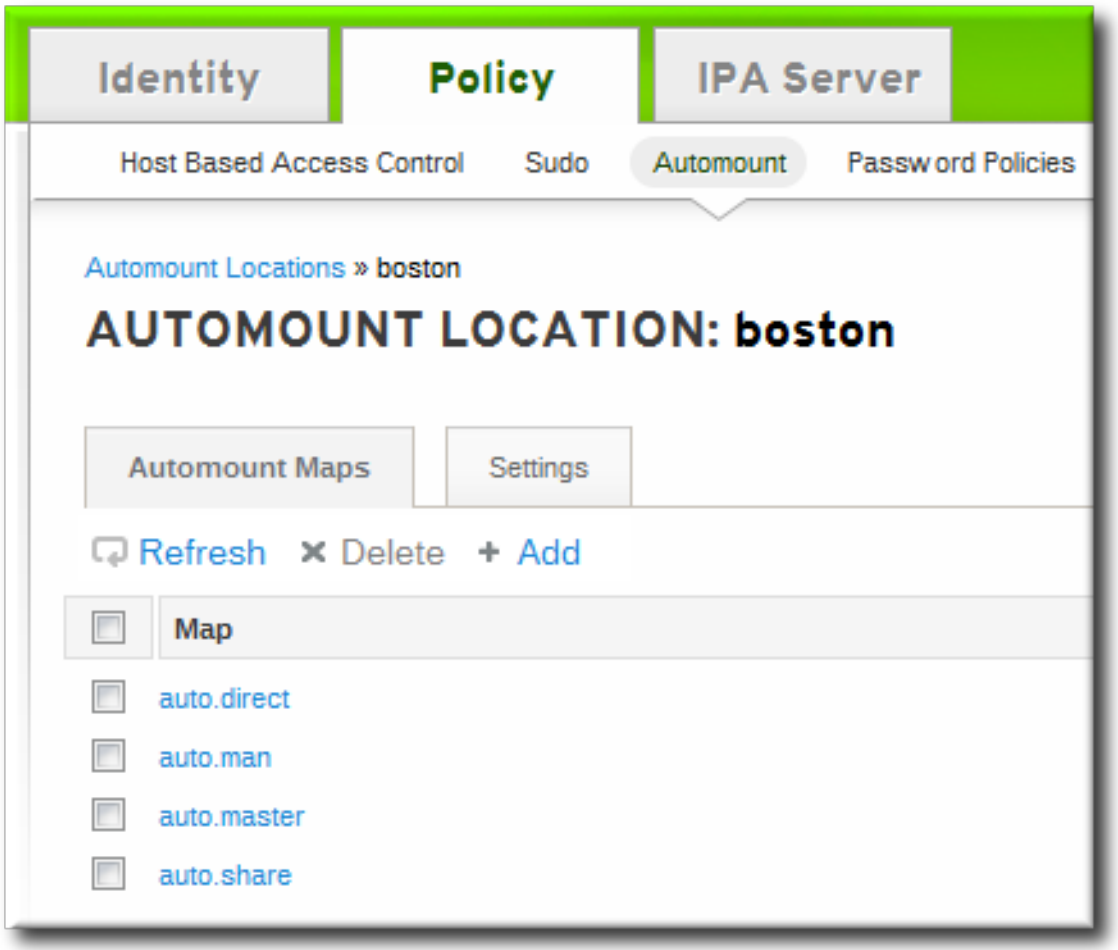
34.6.2.1. 从 Web UI 配置间接映射

1. 点 Policy 选项卡。
2. 点 Automount 子选项卡。
3. 单击要向其添加映射的自动挂载位置的名称。

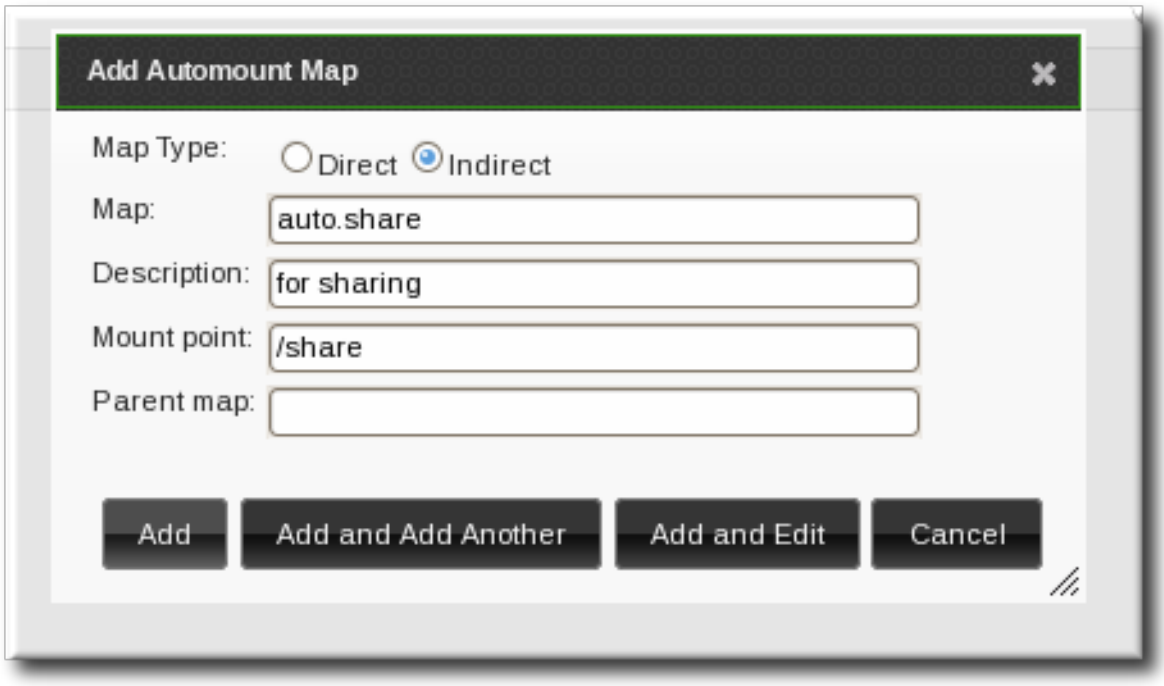


4.

在 Automount Maps 选项卡中，点 + Add 链接来创建新映射。



5. 在弹出窗口中，选择 Indirect 单选按钮，并为间接映射输入所需信息：



- 新映射的名称
- 挂载点。Mount 字段设置要用于所有间接映射键的基域。
- （可选）父映射.默认父项是 auto.master，但如果应使用另一个映射，可以在 Parent Map 字段中指定。

6. 点 Add 按钮保存新密钥。

34.6.2.2. 从命令行配置间接映射

直接映射和间接映射的主要区别在于，间接键前面没有正斜杠。

```
---------------------------
/etc/auto.share:
man     ipa.example.com:/docs/man
---------------------------
```

1. 创建一个间接映射，以使用 automountmap-add-indirect 命令设置基本条目。mount 选项设置用于所有间接映射密钥的基础目录。默认父条目为 auto.master，但如果应使用另一个映

射，则可以使用 --parentmap 选项指定。

```
$ ipa automountmap-add-indirect location mapName --mount=directory [--
parentmap=mapName]
```

例如：

```
$ ipa automountmap-add-indirect raleigh auto.share --mount=/share
-------------------------------
Added automount map "auto.share"
-------------------------------
```

2. **为挂载位置添加间接密钥：**

```
$ ipa automountkey-add raleigh auto.share --key=docs --
info="ipa.example.com:/export/docs"
-------------------------
Added automount key "docs"
-------------------------
  Key: docs
  Mount information: ipa.example.com:/export/docs
```

3. **要验证配置，请使用 automountlocation-tofiles 检查位置文件列表：**

```
$ ipa automountlocation-tofiles raleigh
/etc/auto.master:
/-      /etc/auto.direct
/share  /etc/auto.share
--------------------------
/etc/auto.direct:
--------------------------
/etc/auto.share:
man     ipa.example.com:/export/docs
```

在 Solaris 上，使用 ldapclient 命令添加映射来直接添加 LDAP 条目：

```
ldapclient -a
serviceSearchDescriptor=auto_share:automountMapName=auto.share,cn=location,cn=automount,dc
=example,dc=com?one
```

34.6.3. 导入自动挂载映射

如果存在自动挂载映射，可以将它们导入到 IdM 自动挂载配置中。

```
ipa automountlocation-import location map_file [--continuous]
```

唯一需要的信息是 IdM 自动挂载位置，以及映射文件的完整路径和名称。--continuous 选项告知 automountlocation-import 命令继续通过映射文件，即使命令遇到错误。

例如：

```
$ ipa automountlocation-import raleigh /etc/custom.map
```

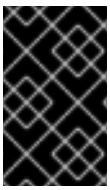
部分 VIII. SECURITY HARDENING

这部分提供了以安全的方式使用 身份管理 的建议实践。

第 35 章 为身份管理配置 TLS

本文档论述了如何将身份管理服务器配置为在 Red Hat Enterprise Linux 7.3 及更高版本中需要 TLS 协议版本 1.2。

TLS 1.2 被认为比之前的 TLS 版本更安全。如果您的 IdM 服务器部署在一个安全要求高的环境中，您可以使用比 TLS 1.2 不太安全的协议将其配置为禁止通信。



重要

在您要使用 TLS 1.2 的每个 IdM 服务器上重复这些步骤。

35.1. 配置 HTTPD 守护进程

1. 打开 /etc/httpd/conf.d/nss.conf 文件，并为 NSSProtocol 和 NSSCipherSuite 条目设置以下值：

```
NSSProtocol TLSv1.2
NSSCipherSuite
+ecdhe_ecdsa_aes_128_sha,+ecdhe_ecdsa_aes_256_sha,+ecdhe_rsa_aes_128_sha,+ecdhe
_rsa_aes_256_sha,+rsa_aes_128_sha,+rsa_aes_256_sha
```

或者，使用以下命令为您设置值：

```
# sed -i 's/^NSSProtocol .*/NSSProtocol TLSv1.2/' /etc/httpd/conf.d/nss.conf
# sed -i 's/^NSSCipherSuite .*/NSSCipherSuite
+ecdhe_ecdsa_aes_128_sha,+ecdhe_ecdsa_aes_256_sha,+ecdhe_rsa_aes_128_sha,+ecdhe
_rsa_aes_256_sha,+rsa_aes_128_sha,+rsa_aes_256_sha/' /etc/httpd/conf.d/nss.conf
```

2. 重启 httpd 守护进程：

```
# systemctl restart httpd
```

35.2. 配置目录服务器组件

使用 ldapmodify 工具自动配置 DS：

1. 使用 ldapmodify 为您进行配置更改：

```
ldapmodify -h localhost -p 389 -D 'cn=directory manager' -W << EOF
dn: cn=encryption,cn=config
changeType: modify
replace: sslVersionMin
sslVersionMin: TLS1.2
EOF
```

2. 重启 DS 以加载新配置：

```
# systemctl restart dirsrv@EXAMPLE-COM.service
```

手动配置目录服务器(DS)：

1. 停止 DS:

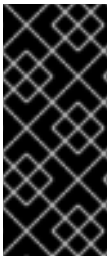
```
# systemctl stop dirsrv@EXAMPLE-COM.service
```

2. 打开 /etc/dirsrv/slapd-EXAMPLE-COM/dse.ldif 文件，并修改 cn=encryption,cn=config 条目来设置以下内容：

```
sslVersionMin: TLS1.2
```

3. 启动 DS:

```
# systemctl start dirsrv@EXAMPLE-COM.service
```



重要

在手动编辑 dse.ldif 文件之前，请确保先关闭服务器。DS 仅在启动时读取一次此文件，因此，如果通过 LDAP 进行更改，服务器运行期间的任何手动更改都将丢失。仅对于无法更改无法动态更改的属性，才建议编辑 dse.ldif 文件。

35.3. 配置证书服务器组件

1. 要手动配置证书服务器(CS)，请打开 /etc/pki/pki-tomcat/server.xml 文件。将

sslVersionRangeStream 和 sslVersionRangeDatagram 参数的所有出现的值设置为以下值：

```
sslVersionRangeStream="tls1_2:tls1_2"
sslVersionRangeDatagram="tls1_2:tls1_2"
```

或者，使用以下命令替换您的值：

```
# sed -i 's/tls1_[01]:tls1_2/tls1_2:tls1_2/g' /etc/pki/pki-tomcat/server.xml
```

2. *重启 CS:*

```
# systemctl restart pki-tomcatd@pki-tomcat.service
```

35.4. 结果

身份管理服务器配置为需要 TLS 1.2。仅支持旧 TLS 版本的身份管理客户端无法再与身份管理服务器通信。

第 36 章 禁用匿名绑定

访问域资源和运行客户端工具始终需要 Kerberos 身份验证。但是，IdM 服务器使用的后端 LDAP 目录默认允许匿名绑定。这可能会向未授权的用户打开所有域配置，包括用户、计算机、组、服务、网络组和 DNS 配置的信息。

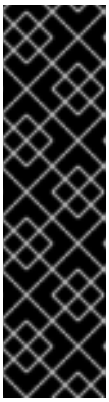
可以使用 LDAP 工具重置 nsslapd-allow-anonymous-access 属性来禁用 389 目录服务器实例的匿名绑定。

警告

某些客户端依赖于匿名绑定来发现 IdM 设置。此外，对于不使用身份验证的传统客户端，compat 树可能会中断。

1. 将 nsslapd-allow-anonymous-access 属性更改为 rootdse。

```
$ ldapmodify -x -D "cn=Directory Manager" -W -h server.example.com -p 389 -ZZ
Enter LDAP Password:
dn: cn=config
changetype: modify
replace: nsslapd-allow-anonymous-access
nsslapd-allow-anonymous-access: rootdse

modifying entry "cn=config"
```



重要

可以完全允许(on)或完全阻止（关闭）匿名访问。但是，完全阻止匿名访问也会阻止外部客户端检查服务器配置。LDAP 和 Web 客户端不一定是域客户端，因此它们将匿名连接以读取 root DSE 文件来获取连接信息。

rootdse 允许访问 root DSE 和服务器配置，而无需访问 目录数据。

2. 重启 389 目录服务器实例以加载新设置。

```
# systemctl restart dirsrv.target
```

其他资源：

- ***Red Hat Directory Server Administration Guide* 中的 *使用命令行管理条目*。**

部分 IX. 性能调优

这部分提供了优化 身份管理 性能的建议实践。

第 37 章 用于批量调配条目的性能调优

使用常规工作流添加大量条目可能会非常慢，如用于添加用户的 第 11 章 管理用户帐户。本章论述了如何调整流程以确保尽快完成调配。

作为流程的一部分：

- 身份管理(IdM)读取要从 LDIF 文件置备的条目，然后将其导入到目标 IdM LDAP 实例。
- 管理员为某些属性设置自定义值，如缓存大小，并禁用 MemberOf 和 Schema 兼容性插件。该流程包括在置备的条目上运行 fixup-memberof.pl 插件，以补补禁用 MemberOf。

此流程经过设计和测试，以调配下列条目类型：用户、用户组、主机、主机组、sudo 规则和基于主机的访问控制(HBAC)规则。

批量调配的建议和前提条件

建议：

- 当置备大量条目（ 10,000 或以上）时，不允许任何 LDAP 客户端访问调配条目或依赖服务器中信息的服务器。例如，您可以禁用服务器上的端口 389 和 636，并使用 LDAPI 在 Unix 套接字上工作。

  原因：MemberOf 插件在服务器上禁用，这意味着服务器上的成员资格信息无效。

- 停止调配期间不需要运行的应用。

  原因：这有助于在机器上释放尽可能多的内存。可用内存将由文件系统缓存使用，从而提高调配的性能。

  请注意，以下步骤已包含停止 IdM 服务的步骤，只重启 Directory Server(DS)实例。IdM 服务（特别是 tomcat ）消耗大量内存，但在调配过程中不使用。

- 在只有一个服务器的新 IdM 部署中运行这个步骤。仅在置备完成后创建副本。

  原因：调配吞吐量比复制快得多。在具有多个服务器的部署中，副本的信息将变得显著过时。

先决条件：

- 生成包含您要置备的条目的 LDIF 文件。例如，如果您要迁移现有的 IdM 部署，请使用 ldapsearch 工具导出所有条目来创建 LDIF 文件。

  有关 LDIF 格式的详情，请参阅 红帽目录服务器 10 管理指南中的 关于 LDIF 文件格式。

备份当前 DS 调优参数值

1. 检索 DS 调优参数的当前值：

   - 数据库缓存大小和数据库锁定：

     ```
     # ldapsearch -D "cn=directory manager" -w secret -b "cn=config,cn=ldbm
     database,cn=plugins,cn=config" nsslapd-dbcachesize nsslapd-db-locks

     ...
     nsslapd-dbcachesize: 10000000
     nsslapd-db-locks: 50000
     ...
     ```

   - 条目缓存大小和 DN 缓存大小：

     ```
     # ldapsearch -D "cn=directory manager" -w secret -b "cn=userRoot,cn=ldbm
     database,cn=plugins,cn=config" nsslapd-cachememsize nsslapd-dncachememsize

     ...
     nsslapd-cachememsize: 10485760
     nsslapd-dncachememsize: 10485760
     ...
     ```

2. 记录获取的值。在完成调配后，您要将参数重置回这些值。

调整数据库、域条目和 DN 缓存大小

对于数据库缓存大小：

1. *确定所需的值。*

   建议的值通常在 200 MB 到 500 MB 之间。适合您的用例的值取决于系统中可用的内存：

   - *超过 8 GB 内存 → 500 MB*
   - *8 GB - 4 GB 内存 → 200 MB*
   - *小于 4 GB 内存 → 100 MB*

2. *使用此模板设置确定的值：*

```
dn: cn=config,cn=ldbm database,cn=plugins,cn=config
changetype: modify
replace: nsslapd-dbcachesize
nsslapd-dbcachesize: db_cache_size_in_bytes
```

有关使用 ldapmodify 工具修改 LDAP 属性的示例，请参阅 例 37.1 “使用 ldapmodify 更改 LDAP 属性”。

例 37.1. 使用 ldapmodify 更改 LDAP 属性

1. *运行 ldapmodify 命令，然后添加语句来修改属性值。例如：*

```
# ldapmodify -D "cn=directory manager" -w secret -x
dn: cn=config,cn=ldbm database,cn=plugins,cn=config
changetype: modify
replace: nsslapd-dbcachesize
nsslapd-dbcachesize: 200000000
```

2. *按 Ctrl+D 确认并将更改发送到服务器。如果操作成功完成，则会显示以下信息：*

```
modifying entry "cn=config,cn=ldbm database,cn=plugins,cn=config"
```

对于域条目缓存大小：

1. *确定所需的值。*

   建议的值介于 100 MB 到 400 MB 之间。适当的值取决于系统中可用的内存：

   - *超过 4 GB 内存 → 400 MB*
   - *2 GB - 4 GB 内存 → 200 MB*
   - *小于 2 GB 内存 → 100 MB*

   如果您要置备大型静态组，建议条目缓存足够大，以适应所有条目：group 和 members。

2. *使用此模板设置确定的值：*

```
dn: cn=userRoot,cn=ldbm database,cn=plugins,cn=config
changetype: modify
replace: nsslapd-cachememsize
nsslapd-cachememsize: entry_cache_size_in_bytes
```

对于域名(DN)缓存大小：

1. *为获得最佳性能，建议 DN 缓存适合调配条目的所有 DN。估算适合您的用例的值：*

   a. *确定 文件中所有 DN 条目的数量。DN 条目位于以 dn: 开头的行中。例如，使用 192.168.1.0/24 grep、sed 和 wc ：*

```
# grep '^dn: ' ldif_file | sed 's/^dn: //' | wc -l
92200
```

b. 确定 LDIF 文件中所有 DN 条目字符串的大小。

```
# grep '^dn: ' ldif_file | sed 's/^dn: //' | wc -c
9802460
```

c. 获取平均 DN 大小：将所有 DN 条目字符串的大小除以文件中所有 DN 条目的数量。

例如：9,802,460 / 92,200 ≈ 106

d. 获取平均内存大小：将平均 DN 大小乘以 2，然后在结果中添加 32。

例如：(106 * 2) + 32 = 244

e. 获取适当的 DN 缓存大小：将平均内存大小乘以 LDIF 文件中的 DN 条目总数。

例如：244 * 92,200 = 22,496,800

2. 使用此模板设置确定的值：

```
dn: cn=userRoot,cn=ldbm database,cn=plugins,cn=config
changetype: modify
Replace: nsslapd-dncachememsize
Nsslapd-dncachememsize: dn_cache_size
```

禁用不必要的服务和调整数据库锁定

1. 禁用 MemberOf 和 Schema 兼容性插件：

```
dn: cn=MemberOf Plugin,cn=plugins,cn=config
changetype: modify
replace: nsslapd-pluginEnabled
nsslapd-pluginEnabled: off
```

```
dn: cn=Schema Compatibility,cn=plugins,cn=config
changetype: modify
replace: nsslapd-pluginEnabled
nsslapd-pluginEnabled: off
```

禁用 MemberOf 可显著加快调配速度。禁用 Schema 兼容性还有助于缩短操作的持续时间。

有关使用 ldapmodify 工具修改 LDAP 属性的示例，请参阅 例 37.1 “使用 ldapmodify 更改 LDAP 属性”。

2. 如果您的拓扑中没有安装副本（如 “批量调配的建议和前提条件”一节中建议），禁用 Content Synchronization 和 Retro Changelog 插件：

```
dn: cn=Content Synchronization,cn=plugins,cn=config
changetype: modify
replace: nsslapd-pluginEnabled
nsslapd-pluginEnabled: off
```

```
dn: cn=Retro Changelog Plugin,cn=plugins,cn=config
changetype: modify
replace: nsslapd-pluginEnabled
nsslapd-pluginEnabled: off
```

禁用这些额外的插件有助于提高调配的性能。

3. 停止 IdM 服务器。这也会停止 DS 实例。

```
# ipactl stop
```

需要停止 DS 以设置下一步中的数据库锁定数量。稍后您将重新启动它。

4. 调整数据库锁定的数量。适当的值等于调配条目数量的一半。

- 最小值为 10,000
- 最大值为 200,000

因为 DS 已停止，所以您必须修改 /etc/dirsrv/slapd-EXAMPLE-COM/dse.ldif 文件来设置值：

```
dn: cn=config,cn=ldbm database,cn=plugins,cn=config
...
nsslapd-db-locks: db_lock_number
```

IdM 在计算成员资格时访问大量数据库页面。它访问的页面越多，调配所需的锁定越多。

5. 启动 DS:

```
# systemctl start dirsrv.target
```

导入条目

将新条目从 LDIF 文件导入到 IdM LDAP 实例。例如，使用 ldapadd 工具：

```
# ldapadd -D "binddn" -y password_file -f ldif_file
```

有关使用 ldapadd 的详情，请查看 ldapadd(1) man page。

重新启用禁用服务和恢复原始属性值

1. 启用成员：

```
dn: cn=MemberOf Plugin,cn=plugins,cn=config
changetype: modify
replace: nsslapd-pluginEnabled
nsslapd-pluginEnabled: on
```

有关使用 ldapmodify 工具修改 LDAP 属性的示例，请参阅 例 37.1 “使用 ldapmodify 更改 LDAP 属性”。

2. 重启 DS:

```
# systemctl restart dirsrv.target
```

此时需要重新启动 DS，因为您在上一步中启用了 MemberOf。

3. 使用 (objectClass =114)过滤器运行 fixup-memberof.pl 脚本，在所有置备的条目上重新生成和更新 memberOf 属性。例如：

```
# fixup-memberof.pl -D "cn=directory manager" -j password_file -Z server_id -b "suffix" -f "
(objectClass=*)" -P LDAP
```

需要运行 **fixup-memberof.pl**，因为在导入条目时 **MemberOf** 插件被禁用。要能够继续调配，脚本必须成功完成。

有关 **fixup-memberof.pl** 的详情请参考 **fixup-memberof.pl(8) man page**。

4. 启用 **Schema Compatibility** 插件：

```
dn: cn=Schema Compatibility,cn=plugins,cn=config
changetype: modify
replace: nsslapd-pluginEnabled
nsslapd-pluginEnabled: on
```

5. 如果您在 *“禁用不必要的服务和调整数据库锁定”一节* 中禁用了 **Content Synchronization** 和 **Retro Changelog** 插件，请重新启用它们：

```
dn: cn=Content Synchronization,cn=plugins,cn=config
changetype: modify
replace: nsslapd-pluginEnabled
nsslapd-pluginEnabled: on
```

```
dn: cn=Retro Changelog Plugin,cn=plugins,cn=config
changetype: modify
replace: nsslapd-pluginEnabled
nsslapd-pluginEnabled: on
```

6. 恢复您在 *“备份当前 DS 调优参数值”一节* 中备份的数据库缓存、条目缓存和 **DN** 缓存大小的原始值：

```
dn: cn=config,cn=ldbm database,cn=plugins,cn=config
changetype: modify
replace: nsslapd-dbcachesize
nsslapd-dbcachesize: backup_db_cache_size

dn: cn=userRoot,cn=ldbm database,cn=plugins,cn=config
changetype: modify
Replace: nsslapd-dncachememsize
Nsslapd-dncachememsize: backup_dn_cache_size
```

```
-
replace: nsslapd-cachememsize
nsslapd-cachememsize: backup_entry_cache_size
```

7. **停止 DS:**

```
# systemctl stop dirsrv.target
```

8. **恢复您在 "备份当前 DS 调优参数值"一节 中备份的数据库锁定的原始值。因为 DS 已停止，所以您必须修改 /etc/dirsrv/slapd-EXAMPLE-COM/dse.ldif 文件来设置值：**

```
dn: cn=config,cn=ldbm database,cn=plugins,cn=config
...
nsslapd-db-locks: backup_db_lock_number
```

9. 启动 **IdM 服务器：**

```
# ipactl start
```

这会启动包括 **DS 在内的所有 IdM 服务。**

第 38 章 身份管理中的故障转移、负载平衡和高可用性

身份管理(IdM)附带自己的故障转移、负载平衡和高可用性功能，如 LDAP 身份域和证书复制，以及 系统安全服务守护进程 (SSSD)提供的服务发现和故障转移支持。

因此，IdM 带有：

- 客户端故障转移功能
- 服务器侧服务可用性

客户端故障转移功能

SSSD 从客户端自动发现的 DNS 服务器获取服务(SRV)资源记录。根据 SRV 记录，SSSD 维护可用 IdM 服务器的列表，包括有关这些服务器连接的信息。如果一个 IdM 服务器离线或超载，SSSD 已经知道要与之通信的其他服务器。

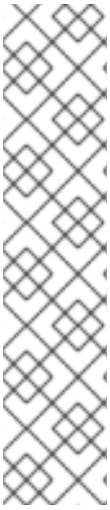
如果 DNS 自动发现不可用，则 IdM 客户端应至少配置有固定的 IdM 服务器列表，以便在出现故障时从中检索 SRV 记录。

在安装 IdM 客户端期间，安装程序会在客户端主机名的父域搜索 _ldap._tcp.DOMAIN DNS SRV 记录。这样，安装程序会检索 IdM 服务器的主机名，该主机名最方便地与客户端通信，并使用其域来配置客户端组件。

服务器侧服务可用性

IdM 允许在地理分散的数据中心中复制服务器，以缩短 IdM 客户端和最接近访问的服务器的路径。复制服务器允许为更多客户端分散负载和扩展。

IdM 复制机制提供主动/主动服务可用性。所有 IdM 副本的服务都随时可用。



注意

不建议将 IdM 与其他负载均衡结合使用，但不建议使用 HA 软件。许多第三方高可用性 (HA)解决方案假定主动/被动情况，并导致 IdM 可用性不需要的服务中断。其他解决方案使用虚拟 IP 或每个集群服务使用一个主机名。所有这些方法通常不适用于 IdM 所提供的服务。另外，它们与 Kerberos 的集成效果也不好，从而降低了部署的整体安全性和稳定性。

还不建议在 IdM 主控机上部署其他不相关的服务，特别是这些服务应该具有高可用性，并使用修改网络配置来提供 HA 功能的解决方案。

有关使用 Kerberos 进行身份验证时使用负载均衡器的详情，请查看以下博客文章。

部分 X. MIGRATION（迁移）

这部分提供了将部署从其他解决方案迁移到身份管理的建议实践。

第 39 章 从 LDAP 目录迁移到 IDM

作为管理员，您之前为身份验证和身份查找部署了 LDAP 服务器，现在您要将后端迁移到身份管理。您需要使用 IdM 迁移工具来传输用户帐户，包括密码和组，而不丢失数据。此外，您还想避免对客户端进行昂贵的配置更新。

此处描述的迁移过程假定一个简单的部署场景，在 LDAP 和 IdM 中有一个名字空间的简单部署场景。对于更复杂的环境，如多个命名空间或自定义模式，请联系红帽支持服务。

39.1. LDAP 到 IDM 迁移概述

从 LDAP 服务器迁移到身份管理的实际迁移部分 - 将数据从一个服务器移动到另一台服务器的过程比较简单。此过程很简单：移动数据、移动密码和移动客户端。

迁移的最昂贵的部分是决定如何将客户端配置为使用身份管理。对于基础架构中的每个客户端，您需要决定正在使用哪些服务（如 Kerberos 和 SSSD），以及最终 IdM 部署中可以使用哪些服务。

辅助但重要的考虑是计划如何迁移密码。除了密码外，身份管理还需要每个用户帐户的 Kerberos 哈希。第 39.1.2 节 “计划密码迁移” 中介绍了密码的一些注意事项和迁移路径。

39.1.1. 规划客户端配置

身份管理可以支持多种不同的客户端配置，具有不同功能、灵活性和安全性。根据 每个客户端的操作系统、功能区域（如开发计算机、生产服务器或用户笔记本电脑）和 IT 维护优先事项，确定最适合每个客户端的配置。

重要

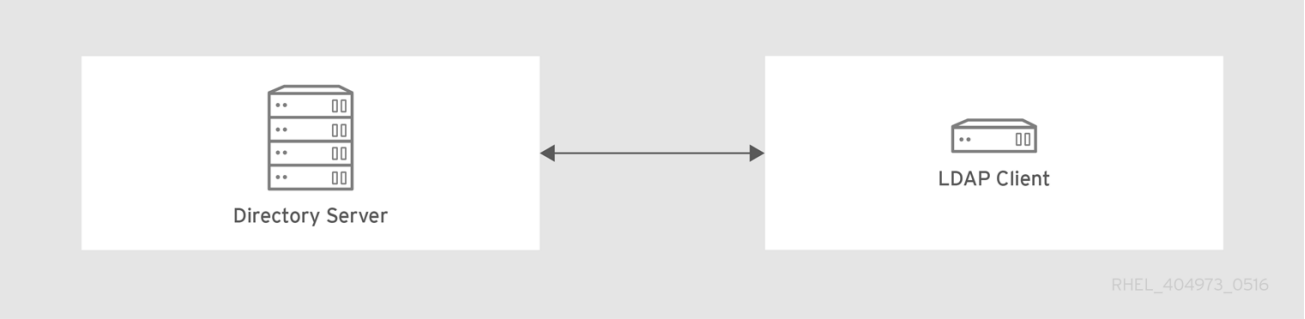
不同的客户端配置 不是相互排斥的。大多数环境都混合有客户端用于连接 IdM 域的不同方法。管理员必须决定哪种方案最适合每个客户端。

39.1.1.1. 初始客户端配置（预迁移）

在决定您要在身份管理中使用客户端配置的位置前，首先在迁移前确定您在哪里。

要迁移的所有 LDAP 部署的初始状态是存在提供身份和身份验证服务的 LDAP 服务。

图 39.1. 基本 LDAP 目录和客户端配置



Linux 和 Unix 客户端使用 PAM_LDAP 和 NSS_LDAP 库直接连接到 LDAP 服务。这些库允许客户端从 LDAP 目录检索用户信息，就像数据存储在 /etc/ passwd 或 / etc /shadow 中一样。（在现实环境中，如果客户端使用 LDAP 进行身份查找，并且使用 Kerberos 进行身份验证或其他配置，则基础架构可能更为复杂。）

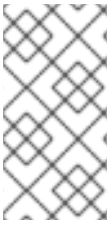
LDAP 目录和 IdM 服务器之间存在结构性差异，特别是在模式支持和目录树的结构方面。（有关这些差异的更多背景信息，请参阅 第 1.1.2 节 “将身份管理与标准 LDAP 目录进行比较”。）虽然这些差异可能会影响数据（特别是目录树，这会影响条目名称），但它们对客户端配置的影响很少，因此它对将客户端迁移到身份管理的影响很少。

39.1.1.2. 推荐的 Red Hat Enterprise Linux 客户端的配置

Red Hat Enterprise Linux 有一个名为 系统安全服务守护进程 (SSSD)的服务。SSSD 使用特殊的 PAM 和 NSS 库(pam_sss 和 nss_sss)，允许 SSSD 与身份管理紧密集成，并利用身份管理中的完整身份验证和身份功能。SSSD 具有许多有用的功能，如缓存身份信息，因此即使在中央服务器丢失了连接的情况下，用户也可以登录；这些内容在 系统级身份验证指南中 描述。

与通用 LDAP 目录服务（使用 pam_ldap 和 nss_ldap）不同，SSSD 通过定义 域 在身份和身份验证信息之间建立关系。SSSD 中的域定义四个后端功能：身份验证、身份查找、访问和密码更改。然后，SSSD 域配置为使用 供应商 为这四个功能中的任何一个（或全部）提供信息。域配置中始终需要一个身份提供程序。其他三个提供程序是可选的；如果未定义身份验证、访问或密码提供程序，则将身份提供程序用于该功能。

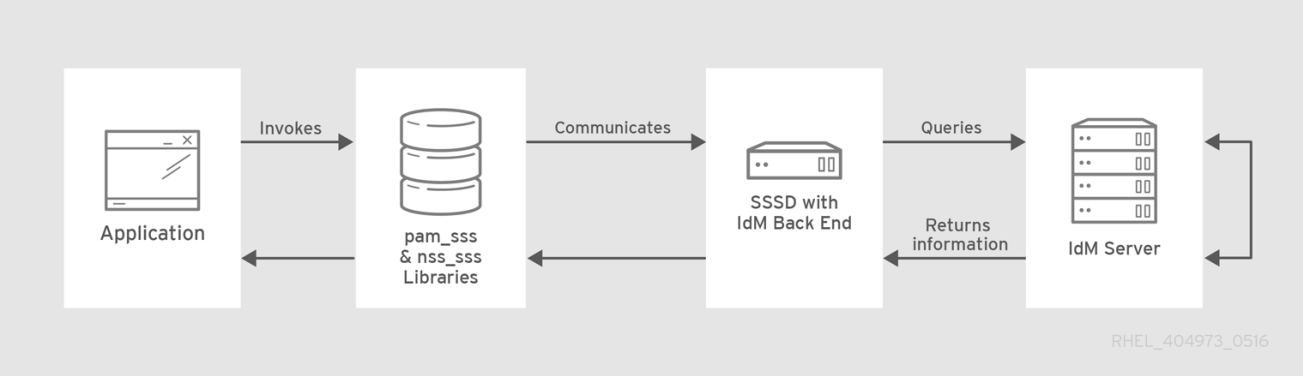
SSSD 可以对所有后端功能使用身份管理。这是理想的配置，因为它提供了完整的身份管理功能，这与通用 LDAP 身份提供程序或 Kerberos 身份验证不同。例如，在日常操作过程中，SSSD 在身份管理中强制执行基于主机的访问控制规则和安全功能。



注意

在迁移过程中，从 LDAP 目录到身份管理，SSSD 可以无缝地迁移用户密码，而无需额外的用户交互。

图 39.2. 客户端和带有 IdM 后端的 SSSD



ipa-client-install 脚本会自动将 SSSD 配置为对所有四个后端服务使用 IdM，因此默认使用推荐的配置设置 Red Hat Enterprise Linux 客户端。

注意

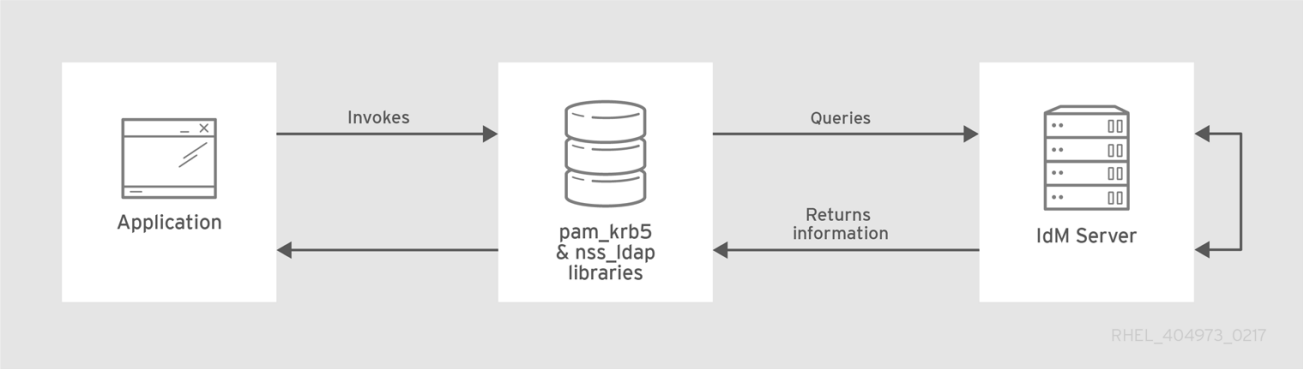
此客户端配置只支持 Red Hat Enterprise Linux 6.1 及更新的版本和 Red Hat Enterprise Linux 5.7，它支持 SSSD 和 ipa-client 的最新版本。可以配置旧版本的 Red Hat Enterprise Linux，如 第 39.1.1.3 节 “备用支持的配置” 所述。

39.1.1.3. 备用支持的配置

UNIX 和 Linux 系统，如 Mac、Solaris、HP-UX、OSPX 和 Scientific Linux 支持 IdM 管理的所有服务，但不使用 SSSD。同样，旧的 Red Hat Enterprise Linux 版本(6.1 和 5.6)支持 SSSD，但有一个旧版本，它不支持 IdM 作为身份提供程序。

当无法在系统上使用 SSSD 的现代版本时，可以将客户端配置为连接到 IdM 服务器，就像它是身份查找的 LDAP 目录服务（使用 nss_ldap），以及 IdM （就像使用 pam_krb5）一样。

图 39.3. 客户端与带有 LDAP 和 Kerberos 的 IdM



如果 Red Hat Enterprise Linux 客户端使用旧版本的 SSSD，则 SSSD 仍可配置为使用 IdM 服务器作为其身份提供者及其 Kerberos 身份验证域；这在 系统级身份验证指南 的 SSSD 配置部分中进行了描

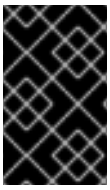
述。

任何 IdM 域客户端都可以使用 nss_ldap 和 pam_krb5 连接到 IdM 服务器。对于某些维护情况和 IT 结构，可能需要将 LDAP 用于身份和验证(nss_ldap 和 pam_ldap)。但是，通常最好将最安全的配置用于客户端。这意味着 SSSD 或 LDAP 用于身份和 Kerberos（用于身份验证）。

39.1.2. 计划密码迁移

很可能影响 LDAP 到身份管理迁移的最可见问题是迁移用户密码。

身份管理（默认）使用 Kerberos 进行身份验证，并且要求每个用户除标准用户密码外，每个用户还有存储在身份管理目录服务器中的 Kerberos 哈希。要生成这些哈希，用户需要以明文形式对 IdM 服务器提供用户密码。当您创建用户时，密码会在哈希化并存储在身份管理之前以明文形式提供。但是，当用户从 LDAP 目录迁移时，相关的用户密码已被哈希处理，因此无法生成对应的 Kerberos 密钥。



重要

用户无法对 IdM 域进行身份验证或访问 IdM 资源，直到它们有 Kerberos 哈希。

如果用户没有 Kerberos 哈希[6]，即使他有用户帐户，该用户也无法登录 IdM 域。迁移密码有三个选项：强制更改密码、使用 Web 页面和使用 SSSD。

从现有系统迁移用户会提供更畅的过渡，但还需要在迁移和转换过程中并行管理 LDAP 目录和 IdM。如果您不保留密码，可以更快地执行迁移，但需要管理员和用户进行更多手动操作。

39.1.2.1. 方法 1：使用临时密码和要求更改

当在身份管理中更改密码时，将使用适当的 Kerberos 哈希创建它们。因此，管理员的一种替代方案是在迁移用户帐户时重置所有用户帐户，从而强制用户更改其密码。新用户被分配一个临时密码，在第一次登录时更改密码。没有迁移密码。

详情请查看 第 22.1.1 节 “更改和重置用户密码”。

39.1.2.2. 方法 2：使用 Migration Web 页面

当它以迁移模式运行时，身份管理在其 Web UI 中有一个特殊的网页，它将捕获明文密码并创建适当

的 Kerberos 哈希。

> https://ipaserver.example.com/ipa/migration

管理员可以告诉用户对此网页进行身份验证一次，该页面将使用密码和相应的 Kerberos 哈希正确更新其用户帐户，而无需更改密码。

39.1.2.3. 方法 3：使用 SSSD（推荐）

SSSD 可以与 IdM 一起使用，通过生成所需的用户密钥来缓解用户对迁移的影响。对于具有大量用户或用户不应使用密码更改负担的部署，这是最佳方案。

1. 用户使用 SSSD 登录到机器。
2. SSSD 尝试对 IdM 服务器执行 Kerberos 身份验证。
3. 尽管用户存在于系统中，但 不支持错误密钥类型的身份验证会失败，因为 Kerberos 哈希尚不存在。
4. 然后 SSSD 对安全连接执行纯文本 LDAP 绑定。
5. IdM 截获此绑定请求。如果用户有 Kerberos 主体，但没有 Kerberos 哈希，则 IdM 身份提供者会生成哈希，并将其存储在用户条目中。
6. 如果身份验证成功，SSSD 会断开与 IdM 的连接，并再次尝试 Kerberos 身份验证。这一次，请求会成功，因为条目中存在哈希。

整个进程对用户完全透明；就用户所知，他们只需登录客户端服务即可正常工作。

39.1.2.4. 迁移 Cleartext LDAP 密码

尽管大多数部署中 LDAP 密码都存储有加密方式，但可能存在某些用户或某些环境对用户条目使用明文密码。

当用户从 LDAP 服务器迁移到 IdM 服务器时，他们的明文密码不会迁移。身份管理不允许明文密码。相反，会为用户创建 Kerberos 主体，而 keytab 设置为 true，密码设置为过期。这意味着身份管理要求用户在下次登录时重置密码。

注意

如果对密码进行哈希处理，密码将通过 SSSD 和迁移网页成功迁移，如 第 39.1.2.2 节 “方法 2：使用 Migration Web 页面” 和 第 39.1.2.3 节 “方法 3：使用 SSSD（推荐）” 中所示。

39.1.2.5. 自动重置密码，但无需满足要求

如果原始目录中的用户密码不符合身份管理中定义的密码策略，则必须在迁移后重置密码。

当用户第一次尝试 kinit 到 IdM 域时，会自动重置密码。

```
[jsmith@server ~]$ kinit
Password for jsmith@EXAMPLE.COM:
Password expired.  You must change it now.
Enter new password:
Enter it again:
```

39.1.3. 迁移注意事项和要求

当您计划从 LDAP 服务器迁移到身份管理时，请确保您的 LDAP 环境能够使用身份管理迁移脚本。

39.1.3.1. 支持迁移的 LDAP 服务器

从 LDAP 服务器到身份管理的迁移过程使用特殊的脚本 ipa migrate-ds 来执行迁移。此脚本对 LDAP 目录和 LDAP 条目的结构有一定的预期，以便能工作。仅支持 LDAPv3 兼容目录服务的迁移，其中包括几个通用目录：

- Sun ONE 目录服务器
- Apache 目录服务器
- OpenLDAP

从 LDAP 服务器迁移到身份管理已使用红帽目录服务器和 OpenLDAP 进行了测试。

注意

Microsoft Active Directory 不支持使用 迁移脚本进行迁移，因为它不是符合 LDAPv3 的目录。如需从 Active Directory 迁移的帮助，请联系红帽专业服务。

39.1.3.2. 迁移环境要求

Red Hat Directory Server 和 Identity Management 有很多不同的配置场景，其中任何一种情况可能会影响迁移过程。对于本章中的迁移过程示例，以下是有关环境的假设：

- 正在将一个 LDAP 目录域迁移到一个 IdM 域。不涉及整合。
- 用户密码作为哈希存储在 LDAP 目录中。有关支持的哈希列表，请参阅 Table 19.2 中的 passwordStorageScheme 属性。Red Hat Directory Server 10 管理指南中的 与密码策略相关的属性。
- LDAP 目录实例既是身份存储和身份验证方法。客户端机器配置为使用 pam_ldap 或 nss_ldap 连接到 LDAP 服务器。
- 条目仅使用标准 LDAP 模式。包含自定义对象类或属性的条目不会迁移到身份管理。

39.1.3.3. 迁移 - IdM 系统要求

对于中等大小的目录（大约 10,000 个用户和 10 个组），需要具有足够强大的目标系统(IdM 系统)来允许迁移继续进行。迁移的最低要求是：

- 4 个内核
- 4GB RAM
- 30GB 磁盘空间

- 2MB 的 SASL 缓冲大小(IdM 服务器的默认)

  如果出现迁移错误，请增大缓冲大小：

```
[root@ipaserver ~]# ldapmodify -x -D 'cn=directory manager' -w password -h
ipaserver.example.com -p 389

dn: cn=config
changetype: modify
replace: nsslapd-sasl-max-buffer-size
nsslapd-sasl-max-buffer-size: 4194304

modifying entry "cn=config"
```

  设置 **nsslapd-sasl-max-buffer-size** 值（以字节为单位）。

39.1.3.4. 关于 Sudo 规则的注意事项

如果您已在 LDAP 中使用 sudo，则必须手动迁移存储在 LDAP 中的 sudo 规则。红帽建议在 IdM 中重新创建 netgroups 作为 hostgroups。IdM 为不使用 SSSD sudo 提供者的 sudo 配置自动将 hostgroups 显示为传统的 netgroups。

39.1.3.5. 迁移工具

身份管理使用特定命令 **ipa migrate-ds** 驱动迁移过程，以便 LDAP 目录数据被正确格式化并导入到 IdM 服务器中。使用 **ipa migrate-ds** 时，远程系统用户（由 **--bind-dn** 选项指定）需要具有对 **userPassword** 属性的读取访问权限，否则不会迁移密码。

身份管理服务器必须配置为在迁移模式下运行，然后可以使用迁移脚本。详情请查看 第 39.3 节 “将 LDAP 服务器迁移到身份管理”。

39.1.3.6. 提高迁移性能

LDAP 迁移基本上是 IdM 服务器中 389 目录服务器实例的专用导入操作。调整 389 目录服务器实例以获得更好的导入操作性能，有助于提高整体迁移性能。

有两个参数会直接影响导入性能：

-

nsslapd-cachememsize 属性定义条目缓存允许的大小。这是一个缓冲区，自动设置为总缓存内存大小的 80%。对于大型导入操作，可以增大此参数（或许内存缓存本身），以便更有效地处理大量条目或具有更大属性的条目。

有关如何使用 ldapmodify 修改属性的详情，请参阅 Red Hat Directory Server 10 性能调节指南中的设置 条目缓存大小。https://access.redhat.com/documentation/zh-cn/red_hat_directory_server/11/html-single/performance_tuning_guide/index#tuning-entry-cache

- 系统 ulimit 配置选项设置系统用户允许的最大进程数。处理大型数据库可能会超过限制。如果发生这种情况，增加值：

```
[root@server ~]# ulimit -u 4096
```

如需更多信息，请参阅红帽目录服务器 性能调优指南，网址为 https://access.redhat.com/documentation/zh-cn/red_hat_directory_server/11/html-single/performance_tuning_guide/index。

39.1.3.7. 迁移序列

迁移到身份管理时有四个主要步骤，但顺序略有不同，具体取决于您要首先迁移服务器还是客户端。

使用基于客户端的迁移时，SSSD 用于在配置 IdM 服务器时更改客户端配置：

1. 部署 SSSD.
2. 重新配置客户端以连接到当前的 LDAP 服务器，然后故障转移到 IdM。
3. 安装 IdM 服务器。
4. 使用 IdM ipa migrate-ds 脚本迁移用户数据。这会从 LDAP 目录导出数据、IdM 模式的格式，然后将它导入到 IdM。
5. 使 LDAP 服务器离线，并允许客户端透明地切换到身份管理。

通过服务器迁移，LDAP 到身份管理迁移首先是：

1. 安装 IdM 服务器。

2. 使用 IdM ipa migrate-ds 脚本迁移用户数据。这会从 LDAP 目录导出数据，为 IdM 模式格式化数据，然后将其导入到 IdM 中。

3. 可选。部署 SSSD.

4. 重新配置客户端来连接到 IdM。无法简单地替换 LDAP 服务器。IdM 目录树 - 因此用户条目 DN - 与之前的目录树不同。

   虽然需要重新配置客户端，但不需要立即重新配置客户端。更新的客户端可以指向 IdM 服务器，而其他客户端则指向旧的 LDAP 目录，从而在数据迁移后可允许合理的测试和过渡阶段。

注意

不要长时间并行运行 LDAP 目录服务和 IdM 服务器。这增加了两个服务之间用户数据不一致的风险。

这两个进程都提供常规迁移过程，但可能并不在每个环境中都有效。在尝试迁移真实 LDAP 环境之前，设置测试 LDAP 环境并测试迁移过程。

39.2. 使用 IPA MIGRATE-DS的示例

数据迁移是使用 ipa migrate-ds 命令执行的。最简单的方式是利用 目录的 LDAP URL 来根据常见的默认设置迁移和导出数据。

```
ipa migrate-ds ldap://ldap.example.com:389
```

迁移的条目

migrate-ds 命令只迁移包含 gidNumber 属性的帐户，这是 posixAccount 对象类和 sn 属性所需的帐户，这是 person 对象类所需的。

自定义流程

ipa migrate-ds 命令允许您自定义如何识别和导出数据。如果原始目录树具有唯一的结构，或者应排除条目中的某些条目或属性，则这很有用。如需了解更多详细信息，请将 --help 传递给 命令。

bind DN

默认情况下，DN "cn=Directory Manager" 用于绑定到远程 LDAP 目录。将 --bind-dn 选项传递给 命令，以指定自定义绑定 DN。有关详情请参考 第 39.1.3.5 节 “迁移工具”。

命名上下文更改

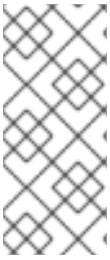
如果目录服务器命名上下文与身份管理中使用的 Directory 服务器命名上下文不同，则对象的基本 DN 会被转换。例如： uid=user,ou=Person,dc=ldap,dc=example,dc=com 被迁移到 uid=user,ou=Person,dc=idm,dc=example,dc=com。将 --base-dn 传递给 ipa migrate-ds 命令，来设置远程 LDAP 服务器上用于迁移的基本 DN。

39.2.1. 迁移特定子树

默认目录结构将人员条目置于 ou=People 子树中，并将组条目置于 ou=Groups 子树中。这些子树是这些不同类型的目录数据的容器条目。如果没有通过 migrate-ds 命令传递选项，则实用程序假定给定的 LDAP 目录使用 ou=People 和 ou=Groups 结构。

许多部署可能具有完全不同的目录结构（或者可能只想导出目录树的某些部分）。管理员可以使用两个选项来指定源 LDAP 服务器上的不同用户或组子树的 RDN：

- *--user-container*
- *--group-container*



注意

在这两种情况下，子树都必须是 RDN，且必须相对于基本 DN。例如，可以使用 --user-container =ou=Employees 进行迁移 >ou=Employees,dc=example,dc=com 目录树。

例如：

```
[root@ipaserver ~]# ipa migrate-ds --user-container=ou=employees \
                    --group-container="ou=employee groups" \
  ldap://ldap.example.com:389
```

将 **--scope** 选项传递给 **ipa migrate-ds** 命令，以设置范围：

- **onelevel** ：默认.仅迁移指定容器中的条目。
- 子树 ：指定容器和所有子容器中的条目都已迁移。
- 基本 ：只有指定的对象本身会被迁移。

39.2.2. 具体包括或排除实体

默认情况下，**ipa migrate-ds** 脚本会导入具有 **person** 对象类的每个用户条目，以及带有 **groupOfUniqueNames** 或 **groupOfNames** 对象类的每个组条目。

在某些迁移路径中，可能需要导出特定类型的用户和组，或者需要取消特定用户和组。

种选择是积极设置要包含 哪些类型的 用户和组。这是通过设置在查找用户或组条目时要搜索的对象类来完成的。

当环境中存在用于不同用户类型的自定义对象类时，这是一个非常有用的选项。例如，这只迁移具有自定义 **fullTimeEmployee** 对象类的用户：

```
[root@ipaserver ~]# ipa migrate-ds --user-objectclass=fullTimeEmployee
ldap://ldap.example.com:389
```

由于不同类型的组，这对于仅迁移某些类型组（如用户组）来说也非常有用，同时排除其他类型的组，如证书组。例如：

```
[root@ipaserver ~]# ipa migrate-ds --group-objectclass=groupOfNames --group-
objectclass=groupOfUniqueNames ldap://ldap.example.com:389
```

积极指定要基于对象类进行迁移的用户和组隐式地排除所有其他用户和组迁移。

或者，迁移所有用户和组条目也很有用，但只有少量条目除外。可以排除特定用户或组群帐户，而该类型的所有其他帐户都将被迁移。例如，这不包括 hobbies 组和两个用户：

```
[root@ipaserver ~]# ipa migrate-ds --exclude-groups="Golfers Group" --exclude-users=jsmith --
exclude-users=bjensen ldap://ldap.example.com:389
```

exclude 语句应用到与 uid 中模式匹配的用户，以及在 cn 属性中与其匹配的组。

指定要迁移的对象类可以和排除特定条目一起使用。例如，这特别包括具有 fullTimeEmployee 对象类的用户，但排除了三个管理者：

```
[root@ipaserver ~]# ipa migrate-ds --user-objectclass=fullTimeEmployee --exclude-users=jsmith --
exclude-users=bjensen --exclude-users=mreynolds ldap://ldap.example.com:389
```

39.2.3. 排除条目属性

默认情况下，用户或组条目的每个属性和对象类将被迁移。在有些情况下，由于带宽和网络约束，可能不太现实，或者由于属性数据不再相关。例如，如果在用户加入 IdM 域时为其分配新的用户证书，则不需要迁移 userCertificate 属性。

migrate-ds 可以使用以下几个不同的选项忽略特定的对象类和属性：

- --user-ignore-objectclass
- --user-ignore-attribute
- --group-ignore-objectclass
- --group-ignore-attribute

例如，要为用户排除 userCertificate 属性和 strongAuthenticationUse 对象类，为组排除 groupOfCertificate 对象类：

```
[root@ipaserver ~]# ipa migrate-ds --user-ignore-attribute=userCertificate --user-ignore-objectclass=strongAuthenticationUser --group-ignore-objectclass=groupOfCertificates ldap://ldap.example.com:389
```

注意

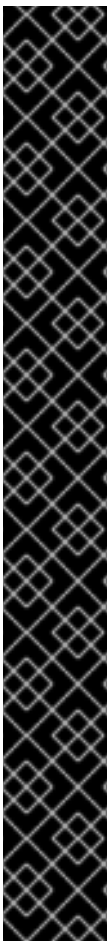
确保不要忽略任何必需的属性。另外，在排除对象类时，请确保排除该对象类仅支持的任何属性。

39.2.4. 将架构设置为使用

身份管理使用 RFC2307bis 模式来定义用户、主机、主机组和其他网络身份。但是，如果用作迁移源的 LDAP 服务器改为使用 RFC2307 模式，请将 **--schema** 选项传给 **ipa migrate-ds** 命令：

```
[root@ipaserver ~]# ipa migrate-ds --schema=RFC2307 ldap://ldap.example.com:389
```

39.3. 将 LDAP 服务器迁移到身份管理



重要

这是一个一般迁移过程，但可能并不在每个环境中都有效。

强烈建议您在尝试迁移真实 LDAP 环境前设置测试 LDAP 环境并测试迁移过程。验证迁移是否已正确完成：

- 使用 **ipa user-add** 命令在 IdM 上创建测试用户，并将迁移的用户的输出与 **test** 用户进行比较。确保迁移的用户包含 **test** 用户中存在的最小属性和对象类集合。

  ```
  $ ipa user-add TEST_USER
  ```

- 将迁移的用户的输出（在 IdM 上 10 月）与源用户（原始 LDAP 服务器上十之一）进行比较。确保导入的属性没有加倍，且具有预期值。

  ```
  $ ipa user-show --all TEST_USER
  ```

1. 在与现有 LDAP 目录不同的机器上安装 IdM 服务器，包括任何自定义 LDAP 目录模式。

注意

自定义用户或组模式在 IdM 中的支持有限。它们可能会导致迁移期间出现问题，因为对象定义不兼容。

2. 禁用 compat 插件。

```
[root@server ~]# ipa-compat-manage disable
```

如果在迁移过程中需要兼容性树提供的数据，则不需要这一步。

3. 重启 IdM 目录服务器实例。

```
[root@server ~]# systemctl restart dirsrv.target
```

4. 配置 IdM 服务器来允许迁移：

```
[root@server ~]# ipa config-mod --enable-migration=TRUE
```

5. 运行 IdM 迁移脚本 ipa migrate-ds。在最基本的方面，这只需要迁移 LDAP 目录实例的 LDAP URL：

```
[root@server ~]# ipa migrate-ds ldap://ldap.example.com:389
```

传递 LDAP URL 会利用常见的默认设置迁移所有目录数据。通过指定其他选项（如 第 39.2 节 “使用 ipa migrate-ds的示例” 所述），可以选择性地迁移用户和组数据。

如果上一步中没有禁用 compat 插件，请将 --with-compat 选项传给 ipa migrate-ds。

导出信息后，该脚本会添加所有必需的 IdM 对象类和属性，并在属性中转换 DN 以匹配 IdM 目录树（如果命名上下文不同）。例如： uid=user,ou=Person,dc=ldap,dc=example,dc=com 被迁移到 uid=user,ou=Person,dc=idm,dc=example,dc=com。

6. 如果在迁移前禁用了 compat 插件，请重新启用 compat 插件。

```
[root@server ~]# ipa-compat-manage enable
```

7. 重启 IdM 目录服务器实例。

```
[root@server ~]# systemctl restart dirsrv.target
```

8. 禁用迁移模式：

```
[root@server ]# ipa config-mod --enable-migration=FALSE
```

9. 可选。重新配置非 SSSD 客户端以使用 Kerberos 身份验证(pam_krb5)而不是 LDAP 身份验证(pam_ldap)。在所有用户都已迁移之前，使用 PAM_LDAP 模块；然后可以使用 PAM_KRB5。如需更多信息，请参阅《系统级身份验证指南》中的 配置 Kerberos 客户端。

10. 用户可以通过两种方式生成其哈希 Kerberos 密码：它们都在没有用户互动的情况下迁移用户密码，如 第 39.1.2 节 “计划密码迁移” 所述。

    a. 使用 SSSD：

       i. 将已安装 SSSD 的客户端从 LDAP 后端移到 IdM 后端，并将它们注册为 IdM 的客户端。这会下载所需的密钥和证书。

          在 Red Hat Enterprise Linux 客户端上，可以使用 ipa-client-install 命令来实现。例如：

          ```
          [root@server ~]# ipa-client-install --enable-dns-update
          ```

    b. 使用 IdM 迁移网页：

       i. 指示用户使用迁移网页登录到 IdM：

          ```
          https://ipaserver.example.com/ipa/migration
          ```

11. 要监控用户迁移过程，请查询现有的 LDAP 目录，以查看哪些用户帐户具有密码，但还没有 Kerberos 主键。

```
[user@server ~]$ ldapsearch -LL -x -D 'cn=Directory Manager' -w secret -b
'cn=users,cn=accounts,dc=example,dc=com' '(&(!(krbprincipalkey=*))(userpassword=*))' uid
```

注意

在过滤器两边包含单引号，以便 shell 不会对其进行解释。

12. 完成所有客户端和用户的迁移后，停用 LDAP 目录。

39.4. 通过 SSL 迁移

在迁移过程中加密 LDAP 和 IdM 之间的数据传输：

1. 将签发远程 LDAP 服务器证书的 CA 证书存储在 IdM 服务器上的文件中。例如：/etc/ipa/remote.crt。

2. 按照 第 39.3 节 “将 LDAP 服务器迁移到身份管理” 中描述的步骤操作。但是，对于在迁移过程中加密的 LDAP 连接，请使用 URL 中的 ldaps 协议，并将 --ca-cert-file 选项传给命令。例如：

```
[root@ipaserver ~]# ipa migrate-ds --ca-cert-file=/etc/ipa/remote.crt
ldaps://ldap.example.com:636
```

---

[6] 可以在身份管理中使用 LDAP 身份验证，而不是 Kerberos 身份验证，这意味着用户不需要 Kerberos 哈希。但是，这限制了身份管理的功能，我们不推荐使用。

第 40 章 从非 RHEL LINUX 发行版上的 FREEIPA 迁移到 RHEL 7 上的 IDM

要将非 RHEL Linux 发行版上的 FreeIPA 部署迁移到 RHEL 7 服务器上的身份管理(IdM)部署，您必须首先将新的 RHEL 7 IdM 证书颁发机构(CA)副本添加到现有 FreeIPA 环境中，将与证书相关的角色传送到它，然后停用非 RHEL FreeIPA 服务器。

重要

不支持使用 Convert2RHEL 工具执行非 RHEL FreeIPA 服务器的原位升级到 RHEL 7 IdM 服务器。

先决条件

- 您已确定了非 RHEL FreeIPA 证书颁发机构(CA)续订服务器的域级别。如需更多信息，请参阅 显示当前域级别。
- 您已在要成为新的 CA 续订服务器的系统中安装了 RHEL 7.9。

步骤

要执行迁移，请按照 将身份管理从 Red Hat Enterprise Linux 6 迁移到版本 7 相同的步骤，使用您的非 RHEL FreeIPA CA 服务器作为 RHEL 6 服务器：

1. 如果原始非 RHEL CA 续订服务器正在运行 FreeIPA 版本 3.1 或更早版本，请更新身份管理架构。要显示已安装的 FreeIPA 版本，请使用 ipa --version 命令。
2. 配置 RHEL 7 服务器，并将其作为 IdM 副本添加到非 RHEL Linux 发行版的当前 FreeIPA 环境中。如果您的域的域级别为 0，请参阅安装 RHEL 7 Replica。如果域级别为 1，请按照 创建 Replica 中描述的步骤操作：简介。
3. 使 RHEL 7 复制 CA 续订服务器，停止在非 RHEL 服务器上生成证书撤销列表(CRL)，并将 CRL 请求重定向到 RHEL 7 副本。详情请参阅 将 CA 服务转换到 Red Hat Enterprise Linux 7 服务器。
4. 停止原始的非 RHEL FreeIPA CA 续订服务器，以强制域发现新的 RHEL 7 服务器。详情请参阅 停止 Red Hat Enterprise Linux 6 服务器。
5.

在其他 RHEL 7 系统上安装新副本并弃用非 RHEL 服务器。详情请参阅 迁移 master CA 服务器后的后续步骤。

重要

红帽建议在您的拓扑中只有一个主 RHEL 版本的 IdM 副本。因此，不要延迟停用旧服务器。

其他资源

- 将身份管理从 Red Hat Enterprise Linux 6 迁移到版本 7

附录 A. 故障排除：常规指南

本附录描述了确定问题根本原因的一般步骤，例如查询日志和服务状态。

注意

有关特定问题及其解决方案的列表，请参阅 附录 B, 故障排除：特定问题的解决方案。

当您遇到问题时，您会采取什么措施？

- 使用 ipa 工具执行命令
- 使用 kinit进行身份验证
- 对 IdM Web UI 进行身份验证
- 使用智能卡进行身份验证
- 启动服务

如果您知道 IdM 的具体区域导致了这个问题，请按照以下链接进行操作：

- DNS
- 复制

如果本指南没有帮助您查找和修复问题，请继续归档客户案例，请在问题单报告中包括您使用这些故障排除过程确定的任何显著错误输出。另 请参阅联系红帽技术支持。

A.1. 在执行 IPA 实用程序时调查失败

基本故障排除

1. *将 --verbose (-v)选项添加到 命令。这将显示调试信息。*

2. *将 -vv 选项添加到 命令。这会显示 JSON 响应和请求。*

高级故障排除

图 A.1 “执行 ipa cert-show 命令的构架” 显示用户使用 IdM 命令行工具时哪些组件交互。查询这些组件可帮助您调查问题的发生位置以及导致它的原因。

1. *使用以下实用程序：*

   - *用于检查 IdM 服务器或客户端的 DNS 解析 的主机*
   - *Ping 以检查 IdM 服务器是否可用*
   - *iptables 检查 IdM 服务器上的当前防火墙配置*
   - *检查当前时间 的日期*
   - *nc 尝试连接到所需端口，如下所示 第 2.1.6 节 “端口要求”*

   有关使用这些实用程序的详情，请查看其 man page。

2. *将 KRB5_TRACE 环境变量设置为 /dev/stdout 文件，将 trace-logging 输出发送到 /dev/stdout ：*

   ```
   $ KRB5_TRACE=/dev/stdout ipa cert-find
   ```

   查看 Kerberos 密钥分发中心(KDC)日志： /var/log/krb5kdc.log。

3. 查看 Apache 错误日志：

   a. 在服务器上启用调试级别：打开 /etc/ipa/server.conf 文件，并将 debug=True 选项添加到 [global] 部分。

   b. 重启 httpd 服务：

   ```
   # systemctl restart httpd.service
   ```

   c. 再次运行失败的命令。

   d. 查看服务器上的 httpd 错误日志： /var/log/httpd/error_log。

   使用 -vvv 选项运行 命令，以显示 HTTP 请求和响应。

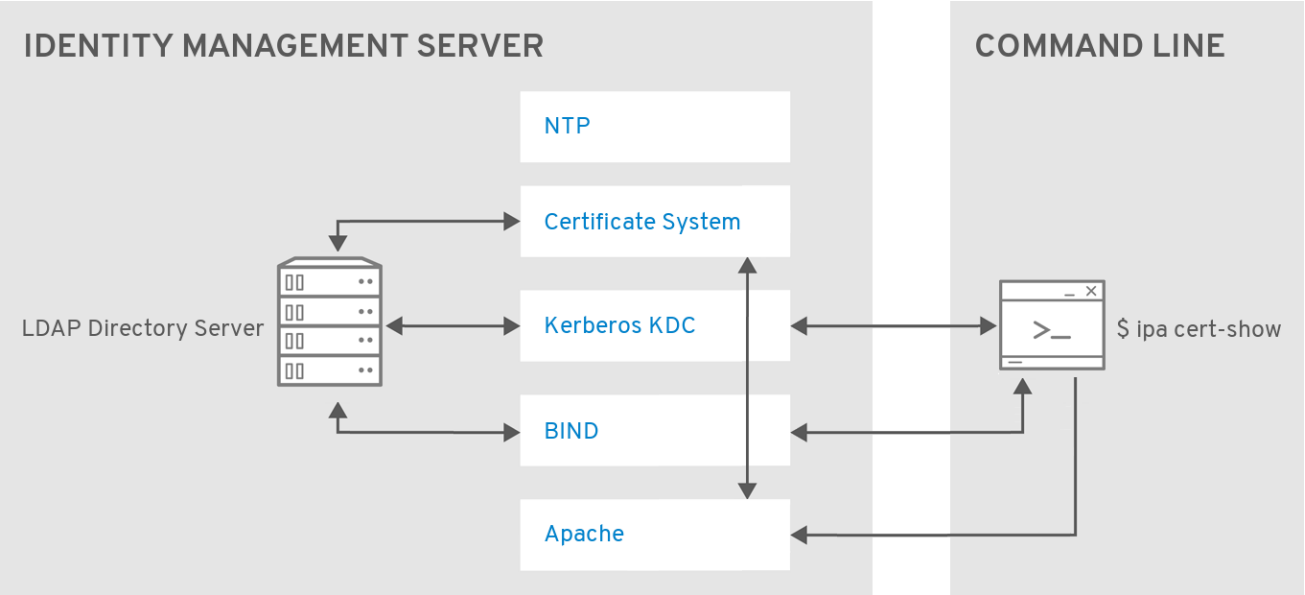
4. 查看 Apache 访问日志： /var/log/httpd/access_log。

   查看证书系统组件的日志：

   - /var/log/pki/pki-ca-spawn.time_of_installation.log
   - /var/log/pki/pki-tomcat/ca/debug
   - /var/log/pki/pki-tomcat/ca/system
   - /var/log/pki/pki-tomcat/ca/selftests.log
   - 使用 job journalctl -u pki-tomcatd@pki-tomcat.service 命令查看日志 日志。

5. 查看目录服务器访问日志：/var/log/dirsrv/slapd-IPA-EXAMPLE-COM/access。

图 A.1. 执行 ipa cert-show 命令的构架



相关信息

- 有关各种身份管理日志文件的描述，请参阅 第 C.2 节 “身份管理日志文件和目录”。

A.2. 调查 KINIT 身份验证失败

常规故障排除

1. 在 IdM 客户端上，显示 kinit 进程的调试信息：

```
$ KRB5_TRACE=/dev/stdout kinit admin
```

2. 验证：

   - 客户端转发记录在服务器和受影响的客户端中都正确：

```
# host client_fully_qualified_domain_name
```

   - 服务器转发记录在服务器和受影响的客户端中都正确：

```
# host server_fully_qualified_domain_name
```

```
# host server_IP_address
```

主机 server_IP_address 命令必须返回一个完全限定的主机名，在末尾带有结尾点，例如：

```
server.example.com.
```

3. 查看客户端上的 /etc/hosts 文件，并确保：

   - 文件中的所有服务器条目都正确
   - 在所有服务器条目中，第一个名称是完全限定域名

   另请参阅 “/etc/hosts 文件”一节。

4. 请确定您满足 第 2.1.5 节 “主机名和 DNS 配置” 中的其他条件。

5. 在 IdM 服务器上，确保 krb5kdc 和 dirsrv 服务正在运行：

```
# systemctl status krb5kdc
# systemctl status dirsrv.target
```

6. 查看 Kerberos 密钥分发中心(KDC)日志： /var/log/krb5kdc.log。

7. 如果 KDC 在 /etc/krb5.conf 文件中硬编码（该文件明确设置了 KDC 指令并使用 dns_lookup_kdc = false 设置），请在每个 master 服务器上使用 ipactl status 命令。使用以下命令检查以 KDC 列出的每台服务器上的 IdM 服务状态：

```
# ipactl status
Directory Service: RUNNING
krb5kdc Service: RUNNING
kadmin Service: RUNNING
named Service: RUNNING
httpd Service: RUNNING
ipa-custodia Service: RUNNING
ntpd Service: RUNNING
pki-tomcatd Service: RUNNING
```

```
ipa-otpd Service: RUNNING
ipa-dnskeysyncd Service: RUNNING
ipa: INFO: The ipactl command was successful
```

对错误进行故障排除 无法查找域的 KDC

如果在 获取初始凭据时显示 Cannot find KDC for realm "EXAMPLE.COM" 的 kinit 身份验证失败，这表示 KDC 未在服务器上运行，或者客户端已配置 DNS。在这种情况下，请尝试以下步骤：

1. **如果在 /etc/krb5.conf 文件中启用了 DNS 发现(dns_lookup_kdc = true 设置)，请使用 dig 工具检查以下记录是否可以解析：**

```
$ dig -t TXT _kerberos.ipa.example.com
$ dig -t SRV _kerberos._udp.ipa.example.com
$ dig -t SRV _kerberos._tcp.ipa.example.com
```

在以下示例中，以上 dig 命令之一失败并显示此输出：

```
; <<>> DiG 9.11.0-P2-RedHat-9.11.0-6.P2.fc25 <<>> -t SRV
_kerberos._tcp.ipa.server.example
;; global options: +cmd
;; connection timed out; no servers could be reached
```

输出显示 named 服务没有在主服务器中运行。

2. **如果 DNS 查找失败，请继续 第 A.6 节 “DNS 故障排除” 中的步骤。**

相关信息

- **有关各种身份管理日志文件的描述，请参阅 第 C.2 节 “身份管理日志文件和目录”。**

A.3. 调查 IDM WEB UI 身份验证失败

1. **确保用户可以使用 kinit 工具从命令行进行身份验证。如果身份验证失败，请参阅 第 A.2 节 “调查 kinit 身份验证失败”。**

2. **确保受影响服务器上的 httpd 和 dirsrv 服务正在运行：**

```
# systemctl status httpd.service
# systemctl status dirsrv@IPA-EXAMPLE-COM.service
```

3. 确保 /var/log/audit/audit.log 和 /var/log/messages 文件中没有相关的 SELinux Access Vector Cache (AVC)消息。

   如需了解有关解析 AVC 消息的详细信息，请参阅红帽知识库 中的 CLI 中的基本 SELinux 故障排除。

4. 确保在您要进行身份验证的浏览器中启用了 cookies。

5. 确保 IdM 服务器与您进行身份验证的系统之间的时间差最多为 5 分钟。

6. 查看 Apache 错误日志： /var/log/httpd/error_log。

7. 为身份验证过程启用详细日志记录，以帮助诊断问题。有关如何在 Firefox 中启用详细登录的建议，请参阅 系统级身份验证指南中的 Firefox Kerberos 配置故障排除。

如果您在使用证书登录时遇到问题：

1. 在 /etc/httpd/conf.d/nss.conf 文件中，将 LogLevel 属性改为 info。

2. 重启 Apache 服务器：

```
# systemctl restart httpd
```

3. 再次尝试使用 证书登录。

4. 查看 Apache 错误日志： /var/log/httpd/error_log。

   日志显示 mod_lookup_identity 模块记录的消息，包括有关模块在登录尝试期间是否成功匹配用户的信息。

相关信息

- 有关各种身份管理日志文件的描述，请参阅 第 C.2 节 “身份管理日志文件和目录”。

A.4. 调查智能卡身份验证失败

1. 打开 /etc/sssd/sssd.conf 文件，并将 debug_level 选项设置为 2。

2. 查看 sssd_pam.log 和 sssd_EXAMPLE.COM.log 文件。如果您在文件中看到超时错误消息，请参阅 第 B.4.4 节 “带有超时错误消息的智能卡验证失败”。

A.5. 检查服务失败为何启动

1. 查看无法启动的服务的日志。请参阅 第 C.2 节 “身份管理日志文件和目录”。

   例如，目录服务器的日志位于 /var/log/dirsrv/slapd-IPA-EXAMPLE-COM/errors。

2. 确保在其上运行该服务的服务器具有完全限定域名(FQDN)。请参阅 “验证服务器主机名”一节。

3. 如果 /etc/hosts 文件包含运行该服务的服务器的条目，请确保首先列出完全限定域名。另请参阅 “/etc/hosts 文件”一节。

4. 请确定您满足 第 2.1.5 节 “主机名和 DNS 配置” 中的其他条件。

5. 确定 keytab 中含有哪些键，用于服务验证。例如，对于 dirsrv 服务票据：

```
# klist -kt /etc/dirsrv/ds.keytab
Keytab name: FILE:/etc/dirsrv/ds.keytab
KVNO Timestamp           Principal
---- ------------------- ------------------------------------------------------
   2 01/10/2017 14:54:39 ldap/server.example.com@EXAMPLE.COM
   2 01/10/2017 14:54:39 ldap/server.example.com@EXAMPLE.COM
   [... output truncated ...]
```

a. 确保显示的主体与系统的 FQDN 匹配。

b. 确保上面显示的服务 keytab 中显示的密钥版本(KVNO)与服务器 key 选项卡中的 KVNO 相匹配。显示服务器 keytab：

```
$ kinit admin
$ kvno ldap/server.example.com@EXAMPLE.COM
```

c. 验证客户端上的正向（A、AAAA 或两者）和反向记录是否与显示的系统名称和服务主体匹配。

6. 验证客户端上的正向（A、AAAA 或两者）和反向记录是否正确。

7. 确保客户端和服务器上的系统时间差异最多为 5 分钟。

8. 在 IdM 管理服务器证书过期后，服务可能无法启动。检查是否是问题单中的原因：

a. 使用 getcert list 命令列出由 certmonger 工具跟踪的所有证书。

b. 在输出中，找到 IdM 管理证书： ldap 和 httpd 服务器证书。

c. 检查标记为 status 的字段并 过期。

```
# getcert list
Number of certificates and requests being tracked: 8.
[... output truncated ...]
Request ID '20170421124617':
 status: MONITORING
 stuck: no
 key pair storage: type=NSSDB,location='/etc/dirsrv/slapd-IPA-EXAMPLE-
COM',nickname='Server-Cert',token='NSS Certificate DB',pinfile='/etc/dirsrv/slapd-IPA-
EXAMPLE-COM/pwdfile.txt'
 certificate: type=NSSDB,location='/etc/dirsrv/slapd-IPA-EXAMPLE-
COM',nickname='Server-Cert',token='NSS Certificate DB'
 CA: IPA
 issuer: CN=Certificate Authority,O=IPA.EXAMPLE.COM
 subject: CN=ipa.example.com,O=IPA.EXAMPLE.COM
 expires: 2019-04-22 12:46:17 UTC
```

```
[... output truncated ...]
Request ID '20170421130535':
 status: MONITORING
 stuck: no
 key pair storage: type=NSSDB,location='/etc/httpd/alias',nickname='Server-
Cert',token='NSS Certificate DB',pinfile='/etc/httpd/alias/pwdfile.txt'
 certificate: type=NSSDB,location='/etc/httpd/alias',nickname='Server-Cert',token='NSS
Certificate DB'
 CA: IPA
 issuer: CN=Certificate Authority,O=IPA.EXAMPLE.COM
 subject: CN=ipa.example.com,O=IPA.EXAMPLE.COM
 expires: 2019-04-22 13:05:35 UTC
[... output truncated ...]
```

如果您需要启动该服务，即使证书已过期，请参阅 第 26.5 节 “允许 IdM 使用过期的证书启动”。

A.6. DNS 故障排除

1. 许多 DNS 问题是由错误配置造成的。因此，请确保满足 第 2.1.5 节 “主机名和 DNS 配置” 中的条件。

2. 使用 dig 工具检查 DNS 服务器的响应：

```
# dig _ldap._tcp.ipa.example.com. SRV

; <<>> DiG 9.9.4-RedHat-9.9.4-48.el7 <<>> _ldap._tcp.ipa.example.com. SRV
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 17851
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 1, ADDITIONAL: 5

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;_ldap._tcp.ipa.example.com. IN SRV

;; ANSWER SECTION:
_ldap._tcp.ipa.example.com. 86400 IN SRV        0 100 389 ipaserver.ipa.example.com.

;; AUTHORITY SECTION:
ipa.example.com.        86400 IN NS       ipaserver.ipa.example.com.

;; ADDITIONAL SECTION:
ipaserver.ipa.example.com. 86400 IN A 192.0.21
ipaserver.ipa.example.com 86400 IN AAAA 2001:db8::1
```

3. 使用 host 实用程序执行 DNS 名称查找：

```
$ host server.ipa.example.com
server.ipa.example.com. 86400 IN A 192.0.21
server.ipa.example.com 86400 IN AAAA 2001:db8::1
```

4. 使用 ipa dnszone-show 命令查看 LDAP 中的 DNS 记录：

```
$ ipa dnszone-show zone_name
$ ipa dnsrecord-show zone_name record_name_in_the_zone
```

有关使用 IdM 工具管理 DNS 的详情，请参考 第 33 章 管理 DNS。

5. 重启 BIND 以强制使用 LDAP 进行重新同步：

```
$ systemctl restart named-pkcs11
```

6. 获取所需 DNS 记录的列表：

```
$ ipa dns-update-system-records --dry-run
```

使用 dig 实用程序检查 DNS 中是否存在显示的记录。如果使用身份管理 DNS，请使用 ipa dns-update-system-records 命令更新任何缺少的记录。

A.7. 复制故障排除

在至少两台服务器上测试复制（请参见 第 4.6 节 “测试新副本”）。如果没有将一个 IdM 服务器上的更改复制到另一个服务器：

1. 请确定您满足 第 2.1.5 节 “主机名和 DNS 配置” 中的条件。

2. 确保两台服务器都可以解析彼此的正向和反向 DNS 记录：

```
[root@server1 ~]# dig +short server2.example.com A
[root@server1 ~]# dig +short server2.example.com AAAA
[root@server1 ~]# dig +short -x server2_IPv4_or_IPv6_address
```

```
[root@server2 ~]# dig +short server1.example.com A
[root@server2 ~]# dig +short server1.example.com AAAA
[root@server2 ~]# dig +short -x server1_IPv4_or_IPv6_address
```

3. **确保两台服务器上的时间差最多为 5 分钟。**

4. **查看两个服务器上的目录服务器错误日志：/var/log/dirsrv/slapd-SERVER-EXAMPLE-COM/errors。**

5. **如果您看到与 Kerberos 相关的错误，请确保 Directory Server keytab 正确，并可使用它查询其他服务器（本例中为server2）：**

```
[root@server1 ~]# kinit -kt /etc/dirsrv/ds.keytab ldap/server1.example.com
[root@server1 ~]# klist
[root@server1 ~]# ldapsearch -Y GSSAPI -h server1.example.com -b "" -s base
[root@server1 ~]# ldapsearch -Y GSSAPI -h server2_FQDN. -b "" -s base
```

相关信息

- **有关各种身份管理日志文件的描述，请参阅 第 C.2 节 “身份管理日志文件和目录”。**

附录 B. 故障排除：特定问题的解决方案

关于故障排除建议：

- 服务器，请查看 第 B.1 节 “身份管理服务器”
- 副本，请查看 第 B.2 节 “Identity Management Replicas”
- 客户端，请查看 第 B.3 节 “身份管理客户端”
- 身份验证，请查看 第 B.4 节 “登录和身份验证问题”
- Vaults, see 第 B.5 节 “Vaults”

B.1. 身份管理服务器

B.1.1. 外部 CA 安装失败

ipa-server-install --external-ca 命令失败并显示以下错误：

```
ipa         : CRITICAL failed to configure ca instance Command '/usr/sbin/pkispawn -s CA -f
/tmp/configuration_file' returned non-zero exit status 1
Configuration of CA failed
```

env|grep proxy 命令显示如下变量：

```
env|grep proxy
http_proxy=http://example.com:8080
ftp_proxy=http://example.com:8080
https_proxy=http://example.com:8080
```

这意味着：

*_proxy 环境变量会阻止安装服务器。

解决此问题：

1. 使用以下 shell 脚本取消设置 *_proxy 环境变量：

```
# for i in ftp http https; do unset ${i}_proxy; done
```

2. 运行 pkidestroy 工具以删除失败的 CA 子系统安装：

```
# pkidestroy -s CA -i pki-tomcat; rm -rf /var/log/pki/pki-tomcat  /etc/sysconfig/pki-tomcat
/etc/sysconfig/pki/tomcat/pki-tomcat  /var/lib/pki/pki-tomcat  /etc/pki/pki-tomcat /root/ipa.csr
```

3. 删除失败的 IdM 服务器安装：

```
# ipa-server-install --uninstall
```

4. 重新运行 ipa-server-install --external-ca。

B.1.2. 名为 Daemon Fails 来启动

安装带有集成 DNS 的 IdM 服务器后， named-pkcs11 无法启动。/var/log/messages 文件包含与 named-pkcs11 服务和 ldap.so 库相关的错误消息：

```
ipaserver named[6886]: failed to dynamically load driver 'ldap.so': libldap-2.4.so.2: cannot open
shared object file: No such file or directory
```

这意味着：

bind-chroot 软件包已安装，并阻止 named-pkcs11 服务启动。

解决此问题：

1. 卸载 bind-chroot 软件包。

```
# yum remove bind-chroot
```

2. 重启 IdM 服务器。

```
# ipactl restart
```

B.1.3. 在禁用 IPv6 的系统中安装服务器故障

当试图在禁用 IPv6 的系统中安装 IdM 服务器时，在安装过程中会出现以下错误：

```
CRITICAL Failed to restart the directory server
Command '/bin/systemctl restart dirsrv@EXAMPLE.service' returned non-zero exit status 1
```

这意味着：

安装和运行服务器要求在网络上启用 IPv6。请参阅 第 2.1.3 节 “系统要求”。

解决此问题：

在 系统上启用 IPv6。详情请查看 红帽知识库中如何在 Red Hat Enterprise Linux 中禁用或启用 IPv6 协议。

请注意，Red Hat Enterprise Linux 7 系统中默认启用 IPv6。

B.2. IDENTITY MANAGEMENT REPLICAS

本指南描述了 Red Hat Enterprise Linux 中身份管理的常见复制问题。

其他资源：

- 有关如何测试复制是否正常工作的建议，请参阅 第 4.6 节 “测试新副本”。
- 有关如何解决复制冲突的建议，请参阅 Red Hat Enterprise Linux 6 Identity Management Guide 中的 解决 复制 冲突。详情请参阅目录服务器 管理指南中的保证命名冲突
- 目录服务器 repl-monitor 脚本显示复制的状态，这有助于对复制问题进行故障排除。如需更多信息，请参阅目录服务器 管理指南中的 监控 复制拓扑。
- 要验证两个目录服务器实例是否已同步，请参阅 目录服务器管理指南。

B.2.1. 对 AD 用户进行身份验证，以防出现新的副本失败

在 Identity Management - Active Directory 信任设置中安装新副本后，尝试根据 IdM 副本验证 Active Directory(AD)用户失败。

这意味着：

副本既不是信任控制器，也不是信任代理。因此，它无法提供来自 AD 信任的信息。

解决此问题：

将副本配置为信任代理。请参阅 Windows 集成指南中的 信任控制器和信任代理。

B.2.2. 目录服务器日志中使用 SASL、GSS-API 和 Kerberos 错误启动副本

当副本启动时，一系列 SASL bind 错误记录在 Directory Server(DS)日志中。错误状态表示 GSS-API 连接失败，因为它找不到凭证缓存：

```
slapd_ldap_sasl_interactive_bind - Error: could not perform interactive bind for id [] mech [GSSAPI]: error -2 (Local error) (SASL(-1): generic failure: GSSAPI Error: Unspecified GSS failure. Minor code may provide more information (Credentials cache file '/tmp/krb5cc_496' not found)) ...
```

另外，可能会出现其他消息表示服务器无法获取主机主体的 Kerberos 凭证：

```
set_krb5_creds - Could not get initial credentials for principal [ldap/ replica1.example.com] in keytab [WRFILE:/etc/dirsrv/ds.keytab]: -1765328324 (Generic error)
```

这意味着：

IdM 使用 GSS-API 进行 Kerberos 连接。DS 实例将 Kerberos 凭据缓存保留在内存中。当 DS 进程结束时（如 IdM 副本停止时），凭据缓存将被销毁。

当副本重启时，DS 会在 KDC 服务器启动前启动。鉴于此启动顺序，当 DS 启动时，Kerberos 凭据尚未保存在凭据缓存中，这就是导致错误的原因。

初始失败后，DS 重新尝试在 KDC 启动后建立 GSS-API 连接。第二次尝试会成功，并确保副本按预期工作。

只要成功建立 GSS-API 连接并且副本可以正常工作，您可以忽略描述的启动错误。以下消息显示连接成功：

```
Replication bind with GSSAPI auth resumed
```

B.2.3. DNS Forward 记录与反向地址不匹配

在配置新副本时，安装会失败，并显示一系列证书错误，后跟 DNS 转发记录与反向地址不匹配的 DNS 错误。

```
ipa: DEBUG: approved_usage = SSLServer intended_usage = SSLServer
ipa: DEBUG: cert valid True for "CN=replica.example.com,O=EXAMPLE.COM"
ipa: DEBUG: handshake complete, peer = 192.0.2.2:9444
Certificate operation cannot be completed: Unable to communicate with CMS (Not Found)

...

ipa: DEBUG: Created connection context.ldap2_21534032
ipa: DEBUG: Destroyed connection context.ldap2_21534032
The DNS forward record replica.example.com. does not match the reverse address
replica.example.org
```

这意味着：

多个主机名用于单个 PTR 记录。DNS 标准允许这样的配置，但会导致 IdM 副本安装失败。

解决此问题：

验证 DNS 配置，如 “验证转发和反向 DNS 配置”一节 所述。

B.2.4. 序列号未找到错误

注意

此解决方案适用于域级别 0。详情请查看 第 7 章 显示和提升域级别。

指出没有找到证书序列号的错误会出现在复制的服务器上：

```
Certificate operation cannot be completed: EXCEPTION (Certificate serial number 0x2d not found)
```

这意味着：

两个副本之间的证书复制协议已被删除，但仍存在数据复制协议。两个副本仍然发出证书，但证书的相关信息不再被复制。

示例情况：

1. 副本 A 向主机签发证书。

2. 证书不会复制到副本 B，因为副本没有建立证书复制协议。

3. 用户尝试使用副本 B 来管理主机。

4. 副本 B 返回一个错误，它无法验证主机的证书序列号。这是因为，副本 B 在其数据目录中具有主机的信息，但它在其证书目录中没有主机证书。

解决此问题：

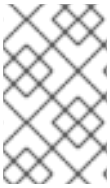
1. 使用 ipa-csreplica-manage connect 命令在两个副本之间启用证书服务器复制。请参阅 第 D.3.3 节 “创建和删除复制协议”。

2. 重新初始化其中一个副本以同步它们。请参阅 第 D.3.5 节 “重新初始化副本”。

警告

重新初始化的 会使用另一个副本中的数据覆盖重新初始化的副本中的数据。某些信息可能会丢失。

B.2.5. 清理副本更新向量(RUV)错误



注意

此解决方案适用于域级别 0。详情请查看 第 7 章 显示和提升域级别。

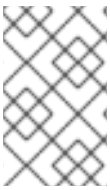
从 IdM 拓扑中删除副本后，过时的 RUV 记录现在出现在一个或多个剩余副本中。

可能的原因：

- 副本已被删除且没有正确删除其复制协议，如 “删除复制协议”一节 所述。
- 当另一个副本离线时，副本已被删除。

这意味着：

其他副本仍期望从删除的副本接收更新。



注意

第 D.3.6 节 “删除副本” 中描述了删除副本的正确步骤。

解决此问题：

清理副本中预期接收更新的 RUV 记录。

1. 使用 ipa-replica-manage list-ruv 命令列出过时的 RUV 的详细信息。该命令显示副本 ID：

```
# ipa-replica-manage list-ruv
server1.example.com:389: 6
server2.example.com:389: 5
server3.example.com:389: 4
server4.example.com:389: 12
```

2. 使用 ipa-replica-manage clean-ruv replica_ID 命令清除损坏的 RUV。该命令将删除与指定副本关联的所有 RUV。

为每个带有过时 RUV 的副本重复该命令。例如：

```
# ipa-replica-manage clean-ruv 6
# ipa-replica-manage clean-ruv 5
# ipa-replica-manage clean-ruv 4
# ipa-replica-manage clean-ruv 12
```

警告

使用 ipa-replica-manage clean-ruv 时要非常小心。针对有效的副本 ID 运行 命令将破坏与复制数据库中该副本关联的所有数据。

如果发生了这种情况，请重新初始化另一个副本的副本，如 第 D.3.5 节 “重新初始化副本” 所述。

3. **再次运行 ipa-replica-manage list-ruv。**

   - **如果命令不再显示任何损坏的 RUV，则成功清理了记录。**
   - **如果命令仍显示损坏的 RUV，请使用此任务手动清除它们：**

```
dn: cn=clean replica_ID, cn=cleanallruv, cn=tasks, cn=config
objectclass: extensibleObject
replica-base-dn: dc=example,dc=com
replica-id: replica_ID
replica-force-cleaning: no
cn: clean replica_ID
```

如果您不确定在哪个副本上清理 RUV：

1. **搜索所有服务器以获取活动副本 ID。制作未损坏且可靠的副本 ID 列表。**

   要查找有效副本的 ID，请在拓扑中运行这个 LDAP 查询：

```
# ldapsearch -p 389 -h ldM_node -D "cn=directory manager" -W -b "cn=config" "
(objectclass=nsds5replica)" nsDS5ReplicaId
```

2. **在每个服务器上运行 ipa-replica-manage list-ruv。请注意，任何不在未损坏副本 ID 列表中的副本 ID。**

3. 为每个损坏的副本 ID 运行 ipa-replica-manage clean-ruv replica_ID。

B.2.6. 恢复丢失的 CA 服务器

注意

此解决方案适用于域级别 0。详情请查看 第 7 章 显示和提升域级别。

您只安装了一台 CA 服务器。此服务器出现故障，现已丢失。

这意味着：

IdM 域的 CA 配置不再可用。

解决此问题：

如果您有原始 CA 服务器的备份，您可以恢复服务器并在副本上安装 CA。

1. 从备份中恢复 CA 服务器.详情请查看 第 9.2 节 “恢复备份”。

   这使得 CA 服务器可供副本使用。

2. 删除初始服务器和副本之间的复制协议，以避免复制冲突。请参阅 第 D.3.3 节 “创建和删除复制协议”。

3. 在副本上安装 CA。请参阅 第 6.5.2 节 “将副本提升到主 CA 服务器”。

4. 停用原始 CA 服务器。请参阅 第 D.3.6 节 “删除副本”。

如果您没有原始 CA 服务器的备份，则 CA 配置会在服务器出现故障且无法恢复时丢失。

B.3. 身份管理客户端

这部分论述了 Red Hat Enterprise Linux 中 IdM 的常见客户端问题。

其他资源：

- 要验证您的 /etc/sssd.conf 文件，请参阅 系统级身份验证指南 中的 SSSD 配置 验证。

B.3.1. 使用外部 DNS 时，客户端无法解决反向查找

外部 DNS 服务器为 IdM 服务器返回错误的主机名。以下与 IdM 服务器相关的错误会出现在 Kerberos 数据库中：

```
Jun 30 11:11:48 server1 krb5kdc[1279](info): AS_REQ (4 etypes {18 17 16 23}) 192.0.2.1:
NEEDED_PREAUTH: admin EXAMPLE COM for krbtgt/EXAMPLE COM EXAMPLE COM, Additional
pre-authentication required
Jun 30 11:11:48 server1 krb5kdc[1279](info): AS_REQ (4 etypes {18 17 16 23}) 192.0.2.1: ISSUE:
authtime 1309425108, etypes {rep=18 tkt=18 ses=18}, admin EXAMPLE COM for krbtgt/EXAMPLE
COM EXAMPLE COM
Jun 30 11:11:49 server1 krb5kdc[1279](info): TGS_REQ (4 etypes {18 17 16 23}) 192.0.2.1:
UNKNOWN_SERVER: authtime 0,  admin EXAMPLE COM for
HTTP/server1.wrong.example.com@EXAMPLE.COM, Server not found in Kerberos database
```

这意味着：

外部 DNS 名称服务器返回 IdM 服务器的主机名，或者返回 no 答案。

解决此问题：

1. 验证您的 DNS 配置，并确保 IdM 使用的 DNS 域已正确委派。详情请查看 第 2.1.5 节 “主机名和 DNS 配置”。

2. 验证您的反向(PTR)DNS 记录设置。详情请查看 第 33 章 管理 DNS。

B.3.2. 客户端未添加到 DNS 区域

在运行 ipa-client-install 工具时，nsupdate 工具无法将客户端添加到 DNS 区。

这意味着：

DNS 配置不正确。

解决此问题：

1. *验证从父区域到 IdM 的 DNS 委派的配置。详情请查看 第 2.1.5 节 “主机名和 DNS 配置”。*

2. *确保 IdM 区域中允许动态更新。详情请查看 第 33.5.1 节 “启用动态 DNS 更新”。*

有关在 IdM 中管理 DNS 的详情，请参考 第 33.7 节 “管理反向 DNS 区域”。有关在 Red Hat Enterprise Linux 中管理 DNS 的详情，请参考 《网络指南》 中的编辑区域文件。

B.3.3. 客户端连接问题

用户无法登录计算机。尝试访问用户和组信息（如使用 lsblk passwd admin 命令）失败。

这意味着：

客户端身份验证问题通常表示系统安全服务守护进程(SSSD)服务有问题。

解决此问题：

检查 /var/log/sssd/ 目录中的 SSSD 日志。目录包含 DNS 域的日志文件，如 sssd_example.com.log。

如果日志没有包含足够信息，请提高日志级别：

1. *在 /etc/sssd/sssd.conf 文件中，查找 [domain/example.com] 部分。调整 debug_level 选项，以在日志中记录更多信息。*

```
debug_level = 9
```

2. *重启 sssd 服务。*

```
# systemctl start sssd
```

3. *再次检查 sssd_example.com.log。文件现在包含更多错误消息。*

B.4. 登录和身份验证问题

B.4.1. 在运行 ipa 命令时 Kerberos GSS 失败

在安装后，当尝试运行 ipa 命令时会出现 Kerberos 错误。例如：

> ipa: ERROR: Kerberos error: ('Unspecified GSS failure. Minor code may provide more information', 851968)/('Decrypt integrity check failed', -1765328353)

这意味着：

DNS 没有正确配置。

解决此问题：

验证您的 DNS 配置。

- 有关 IdM 服务器的 DNS 要求，请参阅 第 2.1.5 节 “主机名和 DNS 配置”。
- 有关 Active Directory 信任的 DNS 要求，请参阅 Windows 集成指南中的 DNS 和 Realm 设置。

B.4.2. 使用 GSS-API 时 SSH 连接失败

用户无法使用 SSH 登录 IdM 机器。

这意味着：

当 SSH 尝试使用 GSS-API 作为安全方法连接到 IdM 资源时，GSS-API 首先验证 DNS 记录。SSH 失败的原因通常是错误的反向 DNS 条目。不正确的记录可防止 SSH 找到 IdM 资源。

解决此问题：

如 第 2.1.5 节 “主机名和 DNS 配置” 所述，验证您的 DNS 配置。

作为临时解决方案，您还可以在 SSH 配置中禁用反向 DNS 查找。为此，请在 /etc/ssh/ssh_config 文件中将 GSSAPITrustDNS 设置为 no。SSH 不使用反向 DNS 记录，而是将给定用户名直接传递给 GSS-API。

B.4.3. OTP 令牌不同步

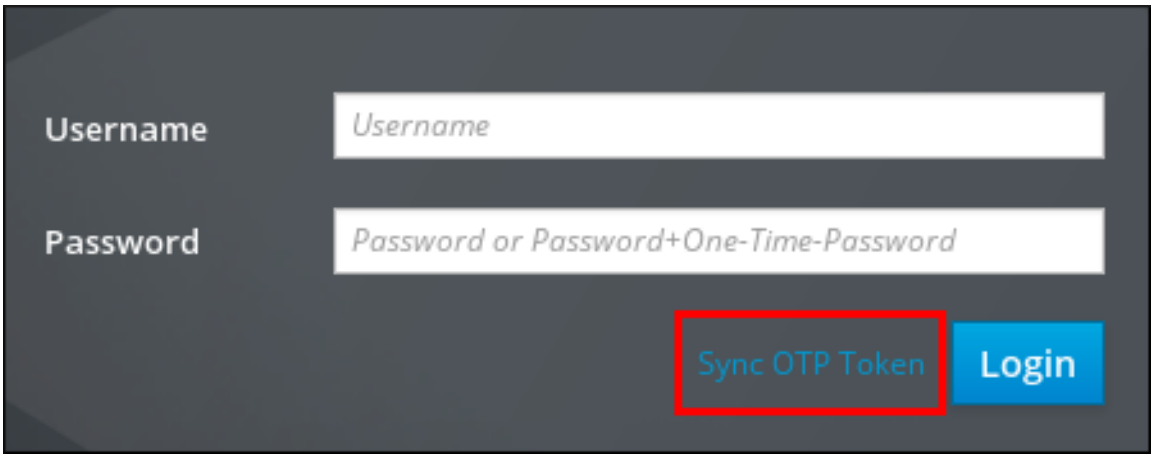
使用 OTP 进行身份验证会失败，因为令牌已被取消同步。

解决此问题：

重新同步令牌。任何用户都可以重新同步其令牌类型，以及用户是否有权修改令牌设置。

1. 在 IdM Web UI 中：在登录页面上，单击 **Sync OTP Token**。

图 B.1. 同步 OTP 令牌



在命令行中：运行 **ipa otptoken-sync** 命令。

2. 提供重新同步令牌所需的信息。例如，IdM 将要求您提供标准密码以及令牌生成的两个后续令牌代码。

注意

即使标准密码已过期，则重新同步也可正常工作。使用过期密码重新同步令牌后，登录到 IdM，让系统提示您更改密码。

B.4.4. 带有超时错误消息的智能卡验证失败

sssd_pam.log 和 sssd_EXAMPLE.COM.log 文件包含超时错误消息，例如：

```
Wed Jun 14 18:24:03 2017) [sssd[pam]] [child_handler_setup] (0x2000):
Setting up signal handler up for pid [12370]
(Wed Jun 14 18:24:03 2017) [sssd[pam]] [child_handler_setup] (0x2000): Signal
handler set up for pid [12370]
(Wed Jun 14 18:24:08 2017) [sssd[pam]] [pam_initgr_cache_remove] (0x2000):
```

```
[idmeng] removed from PAM initgroup cache
(Wed Jun 14 18:24:13 2017) [sssd[pam]] [p11_child_timeout] (0x0020): Timeout
reached for p11_child.
(Wed Jun 14 18:24:13 2017) [sssd[pam]] [pam_forwarder_cert_cb] (0x0040):
get_cert request failed.
(Wed Jun 14 18:24:13 2017) [sssd[pam]] [pam_reply] (0x0200): pam_reply called
with result [4]: System error.
```

这意味着：

当使用转发智能卡读取器或在线证书状态协议(OCSP)时，您可能需要调整某些默认值，以便用户能够使用智能卡进行身份验证。

解决此问题：

在您要验证的服务器和客户端中，在 /etc/sssd/sssd.conf 文件中进行这些更改：

1. **在 [pam] 部分中，将 p11_child_timeout 值增加到 60 秒。**

2. **在 [domain/EXAMPLE.COM] 部分中，将 krb5_auth_timeout 值增加到 60 秒。**

3. **如果您在证书中使用 OCSP，请确保 OCSP 服务器可访问。如果 OCSP 服务器无法直接访问，请通过在 /etc/sssd/sssd.conf 中添加以下选项来配置代理 OCSP 服务器：**

   ```
   certificate_verification = ocsp_default_responder=http://ocsp.proxy.url,
   ocsp_default_responder_signing_cert=nickname
   ```

   使用 /etc/pki/nssdb/ 目录中的 OCSP 签名证书的别名替换 nickname。

   有关这些选项的详情，请查看 sssd.conf(5) man page。

4. **重启 SSSD：**

   ```
   # systemctl restart sssd.service
   ```

B.5. VAULTS

B.5.1. 用户无法访问其 Vault，因为使用无效"添加"权限

用户无法访问自己的用户库或添加新的用户库。此时会出现以下出错信息：

```
ipa: ERROR: Insufficient access: Insufficient 'add' privilege to add the entry
'cn=testvault,cn=user,cn=users,cn=vaults,cn=kra,dc=example,dc=com'.
```

这意味着：

用户的 vault 容器归另一用户所有。通常，这种情况是在另一个用户（如 admin ）后为第一个用户创建第一个用户库。第一个用户然后无法访问自己的 vault 容器中的任何 vault。

解决此问题：

将预期用户添加为 vault 容器的所有者：

1. 以 admin 用户身份登录。

```
$ kinit admin
```

2. 添加 user 作为容器所有者：

```
$ ipa vaultcontainer-add-owner --user=user --users=user
  Owner users: admin, user
  Vault user: user
-------------------------
Number of owners added 1
-------------------------
```

admin 和用户 现在都能够访问用户的 vault 容器，因为它们都是容器的所有者。

3. 可选。验证用户现在可以创建新用户库：

```
$ kinit user
$ ipa vault-add testvault2
-------------------------
Added vault "testvault2"
-------------------------
```

其它资源

- *第 25.4 节 “存储用户的个人机密”*

附录 C. 身份管理文件和日志的参考

C.1. 身份管理配置文件和目录

表 C.1. IdM 服务器和客户端配置文件和目录

目录或文件	描述
/etc/ipa/	主 IdM 配置目录。
/etc/ipa/default.conf	IdM 的主要配置文件。当服务器和客户端使用 **ipa** 实用程序时被引用。
/etc/ipa/server.conf	默认情况下，不存在可选的配置文件。IdM 服务器启动时引用。 如果文件存在，它将优先于 **/etc/ipa/default.conf**。
/etc/ipa/cli.conf	默认情况下，不存在可选的配置文件。当用户使用 **ipa** 工具时引用。 如果文件存在，它将优先于 **/etc/ipa/default.conf**。
/etc/ipa/ca.crt	IdM 服务器的 CA 发布的 CA 证书。
~/.ipa/	用户第一次运行 IdM 命令时，在本地系统中创建的特定于用户的 IdM 目录。 用户可以通过在 **~./ipa/** 中创建特定于用户的 **default.conf**、**server.conf** 或 **cli.conf** 文件来设置单独的配置覆盖。
/etc/sssd/sssd.conf	配置 IdM 域以及 SSSD 使用的 IdM 服务。
/usr/share/sssd/sssd.api.d/sssd-ipa.conf	IdM 相关的 SSSD 选项及其值的 schema。
/etc/gssproxy/	配置 GSS-Proxy 协议的目录。该目录包含每个 GSS-API 服务的文件，以及一个通用 **/etc/gssproxy/gssproxy.conf** 文件。
/etc/certmonger/certmonger.conf	此配置文件包含证书守护进程的默认设置，用于监控证书是否即将到期。
/etc/custodia/custodia.conf	管理 IdM 应用的 secret 的 Custodia 服务的配置文件。

表 C.2. 系统服务文件和目录

目录或文件	描述
/etc/sysconfig/	**systemd**- 特定文件

表 C.3. Web UI 文件和目录

目录或文件	描述
/etc/ipa/html/	IdM Web UI 使用的 HTML 文件的符号链接。
/etc/httpd/conf.d/ipa.conf	由 Apache 主机用于 Web UI 应用的配置文件。
/etc/httpd/conf.d/ipa-rewrite.conf	
/etc/httpd/conf/ipa.keytab	Web 服务器使用的 keytab 文件。
/usr/share/ipa/	Web UI 使用的所有 HTML 文件、脚本和样式表的目录。
/usr/share/ipa/ipa.conf	
/usr/share/ipa/updates/	包含 IdM 的 LDAP 数据、配置和模式更新。
/usr/share/ipa/html/	包含 Web UI 使用的 HTML 文件、JavaScript 文件和样式表。
/usr/share/ipa/migration/	包含 HTML 页面、样式表和 Python 脚本，用于在迁移模式下运行 IdM 服务器。
/usr/share/ipa/ui/	包含 UI 用来执行 IdM 操作的脚本。
/etc/httpd/conf.d/ipa-pki-proxy.conf	用于 web-server-to-Certificate-System 桥接的配置文件。

表 C.4. Kerberos 文件和目录

目录或文件	描述
/etc/krb5.conf	Kerberos 服务配置文件.
/var/lib/sss/pubconf/krb5.include.d/	包括 Kerberos 客户端配置的 IdM 特定覆盖。

表 C.5. 目录服务器文件和目录

目录或文件	描述
/var/lib/dirsrv/slapd-*REALM_NAME*/	与 IdM 服务器使用的 Directory 服务器实例关联的数据库。
/etc/sysconfig/dirsrv	**dirsrv systemd** 服务的特定于 IdM 的配置。
/etc/dirsrv/slapd-*REALM_NAME*/	与 IdM 服务器使用的 Directory 服务器实例关联的配置和模式文件。

表 C.6. 证书系统文件和目录

目录或文件	描述
/etc/pki/pki-tomcat/ca/	IdM CA 实例的主目录。
/var/lib/pki/pki-tomcat/conf/ca/CS.cfg	IdM CA 实例的主配置文件。

表 C.7. 缓存文件和目录

目录或文件	描述
~/.cache/ipa/	包含 IdM 客户端的每台服务器 API 模式。IdM 将客户端上的 API 模式缓存一小时。

表 C.8. 系统备份文件和目录

目录或文件	描述
/var/lib/ipa/sysrestore/	包含安装 IdM 服务器时重新配置的系统文件和脚本的备份。包括 NSS、Kerberos 的原始 **.conf** 文件(**krb5.conf** 和 **kdc.conf**)和 NTP。
/var/lib/ipa-client/sysrestore/	包含安装 IdM 客户端时重新配置的系统文件和脚本的备份。通常，这是 SSSD 身份验证服务的 **sssd.conf** 文件。

C.2. 身份管理日志文件和目录

表 C.9. IdM 服务器和客户端日志文件及目录

目录或文件	描述
/var/log/ipaserver-install.log	IdM 服务器的安装日志。
/var/log/ipareplica-install.log	IdM 副本的安装日志。
/var/log/ipaclient-install.log	IdM 客户端的安装日志。
/var/log/sssd/	SSSD 的日志文件.
~/.ipa/log/cli.log	用于 XML-RPC 调用和 **ipa** 工具的响应返回的错误的日志文件。在运行工具的系统用户 *在主目录中创建*，其用户名可能与 IdM 用户不同。
/etc/logrotate.d/	DNS、SSSD、Apache、Tomcat 和 Kerberos 的日志轮转策略。

目录或文件	描述
/etc/pki/pki-tomcat/logging.properties	这个链接指向 **/usr/share/pki/server/conf/logging.properties** 的默认证书颁发机构日志记录配置。

表 C.10. Apache 服务器日志文件

目录或文件	描述
/var/log/httpd/	Apache Web 服务器的日志文件.
/var/log/httpd/access_log	Apache 服务器的标准访问和错误日志.特定于 IdM 的消息与 Apache 消息一起记录，因为 IdM Web UI 和 XML-RPC 命令行界面使用 Apache。
/var/log/httpd/error_log	
详情请查看 Apache 文档中的日志文件。	

表 C.11. 证书系统日志文件

目录或文件	描述
/var/log/pki/pki-ca-spawn.*time_of_installation*.log	IdM CA 的安装日志。
/var/log/pki/pki-kra-spawn.*time_of_installation*.log	IdM KRA 的安装日志。
/var/log/pki/pki-tomcat/	PKI 操作日志的顶级目录。包含 CA 和 KRA 日志。
/var/log/pki/pki-tomcat/ca/	具有与证书操作相关的日志的目录。在 IdM 中，这些日志用于服务主体、主机以及使用证书的其他实体。
/var/log/pki/pki-tomcat/kra	带有与 KRA 相关的日志的目录。
/var/log/messages	包括证书错误消息以及其他系统信息。
详情请参阅《红帽认证 *系统管理 指南* 》*中的配置* 子系统日志。	

表 C.12. 目录服务器日志文件

目录或文件	描述
/var/log/dirsrv/slapd-*REALM_NAME*/	与 IdM 服务器使用的 Directory 服务器实例关联的日志文件。此处记录的大多数操作数据都与服务器复制交互相关。

目录或文件	描述
/var/log/dirsrv/slapd-*REALM_NAME*/access	包含有关域目录服务器实例试图访问和操作的详细信息。
/var/log/dirsrv/slapd-*REALM_NAME*/errors	
/var/log/dirsrv/slapd-*REALM_NAME*/audit	包含目录服务器配置中启用了审计时所有目录服务器操作的审计跟踪。
详情请查看 Red Hat Directory Server 文档中的 监控服务器和数据库活动和日志文件参考。	

表 C.13. Kerberos 日志文件

目录或文件	描述
/var/log/krb5kdc.log	Kerberos KDC 服务器的主日志文件。
/var/log/kadmind.log	Kerberos 管理服务器的主要日志文件。
这些文件的位置在 **krb5.conf** 文件中配置。它们在某些系统上可能会有所不同。	

表 C.14. DNS 日志文件

目录或文件	描述
/var/log/messages	包括 DNS 错误消息以及其他系统信息。 默认情况下不启用此文件中的 DNS 日志记录。若要启用它，请运行 sVirt **/usr/sbin/rndc querylog** 命令。若要禁用日志记录，请再次运行 命令。

表 C.15. custodia 日志文件

目录或文件	描述
/var/log/custodia/	Custodia 服务的日志文件目录。

其它资源

- ***有关如何使用 journalctl 工具的信息，请参阅 系统管理员指南中的使用 日志。您可以使用 journalctl 查看 systemd 单元文件的日志输出。***

C.3. IDM 域服务和日志轮转

多个 IdM 域服务使用系统 logrotate 服务来处理日志轮转和压缩：

- 命名 (DNS)
- httpd (Apache)
- tomcat
- sssd
- krb5kdc (Kerberos 域控制器)

logrotate 配置文件存储在 /etc/logrotate.d/ 目录中。

例 C.1. 默认 httpd 日志轮转文件位于 /etc/logrotate.d/httpd

```
/var/log/httpd/*log {
    missingok
    notifempty
    sharedscripts
    delaycompress
    postrotate
        /sbin/service httpd reload > /dev/null 2>/dev/null || true
    endscript
}
```

警告

大多数服务的 logrotate 策略文件都会创建一个新的日志文件，其名称、默认所有者和默认权限与上一日志相同。但是，使用 named 和 tomcat 的文件，特殊的 create 规则会设置具有显式权限以及用户和组所有权的行为。

不要更改拥有 named 和 tomcat 日志文件的权限或用户和组。IdM 操作和 SELinux 设置都需要此项。更改日志轮转策略或文件的所有权可能会导致 IdM 域服务失败。

其它资源

- *IdM 作为后端使用的 389 目录服务器实例，而 Dogtag 证书系统都有自己的内部日志轮转策略。请参阅 红帽目录服务器 10 管理指南中的 配置子系统日志。*

- *有关其他潜在的日志轮转设置的详情，如压缩设置或日志文件大小，请参阅 系统管理员 指南中的日志轮转设置或 logrotate(8) man page。*

附录 D. 在域级别 0 管理副本

本附录描述了在域级别 0 中管理副本（请参阅 第 7 章 显示和提升域级别）。有关在域级别 1 中管理副本的文档，请参阅：

- 第 4.5 节 “创建副本：简介”
- 第 6 章 管理复制拓扑

D.1. 副本信息文件

在副本创建过程中，ipa-replica-prepare 实用程序在 /var/lib/ipa/ 目录中创建一个名为 after after replica server 的副本 信息文件。副本信息文件是 GPG 加密的文件，其包含主服务器的域和配置信息。

ipa-replica-install 副本设置脚本根据副本信息文件中包含的信息配置目录服务器实例，并启动 副本初始化过程，其中脚本将通过主服务器的数据复制到副本。副本信息文件只能在创建副本的特定计算机上安装副本。它不能用于在多个机器上创建多个副本。

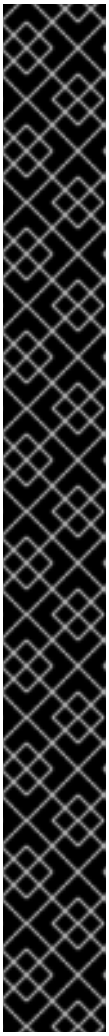
D.2. 创建副本

以下小节描述了最显著的副本安装场景。

- 流程和示例不是相互排斥的；可以同时使用 CA、DNS 和其他命令行选项。以下部分中的示例单独调用，以便更清晰地了解每个配置区域所需的内容。
- ipa-replica-install 工具也接受多个其他选项。如需完整的列表，ipa-replica-install(1) man page。

D.2.1. 在没有 DNS 的情况下安装 Replica

1. 在 master IdM 服务器上，运行 ipa-replica-prepare 工具并添加 副本 计算机的完全限定域名(FQDN)。请注意，ipa-replica-prepare 脚本不会验证 IP 地址，或者验证副本的 IP 地址是否可以被其他服务器访问。



重要

不要使用单标签域名，例如 .company：IdM 域必须由一个或多个子域和一个顶层域组成，如 example.com 或 company.example.com。

完全限定域名必须满足以下条件：

- 它是一个有效的 DNS 名称，即只允许数字、字母字符和连字符(-)。主机名中的其他字符（如下划线(_)）会导致 DNS 失败。
- 都是小写。不允许使用大写字母。
- 完全限定域名不能解析到环回地址。它必须解析到计算机的公共 IP 地址，而不是 127.0.0.1。

有关其他推荐的命名实践，请参阅 Red Hat Enterprise Linux 安全指南中的推荐命名实践。

如果 master 服务器配置了集成的 DNS，请使用 --ip-address 选项指定副本机器的 IP 地址。然后，安装脚本会询问您是否要为副本配置反向区域。只有 IdM 服务器配置了集成的 DNS 时，才传递 --ip-address。否则，没有要更新的 DNS 记录，当 DNS 记录操作失败时，会尝试创建副本失败。

出现提示时，输入初始主服务器的目录管理器(DM)密码。ipa-replica-prepare 的输出显示副本信息文件的位置。例如：

```
[root@server ~]# ipa-replica-prepare replica.example.com --ip-address 192.0.2.2
Directory Manager (existing master) password:

Do you want to configure the reverse zone? [yes]: no
Preparing replica for replica.example.com from server.example.com
Creating SSL certificate for the Directory Server
Creating SSL certificate for the dogtag Directory Server
Saving dogtag Directory Server port
Creating SSL certificate for the Web Server
Exporting RA certificate
Copying additional files
Finalizing configuration
Packaging replica information into /var/lib/ipa/replica-info-replica.example.com.gpg
Adding DNS records for replica.example.com
```

```
Waiting for replica.example.com. A or AAAA record to be resolvable
This can be safely interrupted (Ctrl+C)
The ipa-replica-prepare command was successful
```

警告

副本信息文件包含敏感信息。采取适当步骤确保正确保护它们。

有关可以添加到 ipa-replica-prepare 的其他选项，请查看 ipa-replica-prepare(1) man page。

2. **在副本机器上安装 ipa-server 软件包。**

```
[root@replica ~]# yum install ipa-server
```

3. **将副本信息文件复制到副本机器中：**

```
[root@server ~]# scp /var/lib/ipa/replica-info-replica.example.com.gpg
root@replica:/var/lib/ipa/
```

4. **在副本机器上，运行 ipa-replica-install 工具并添加复制信息文件的位置，以启动副本初始化过程。提示时输入原始主服务器的目录管理器和管理密码，并等待副本安装脚本完成。**

```
[root@replica ~]# ipa-replica-install /var/lib/ipa/replica-info-replica.example.com.gpg
Directory Manager (existing master) password:

Run connection check to master
Check connection from replica to remote master 'server.example.com':

...

Connection from replica to master is OK.
Start listening on required ports for remote master check
Get credentials to log in to remote master
admin@MASTER.EXAMPLE.COM password:

Check SSH connection to remote master

...
```

```
Connection from master to replica is OK.

...

Configuring NTP daemon (ntpd)
  [1/4]: stopping ntpd
  [2/4]: writing configuration

...

Restarting Directory server to apply updates
  [1/2]: stopping directory server
  [2/2]: starting directory server
Done.
Restarting the directory server
Restarting the KDC
Restarting the web server
```

注意

如果安装的副本文件与当前主机名不匹配，则副本安装脚本会显示警告信息并要求确认。在某些情况下，如在多主机器中，您可以确认继续使用不匹配的主机名。

有关可添加到 ipa-replica-install 的命令行选项，请查看 ipa-replica-prepare(1) man page。请注意，其中一个选项 ipa-replica-install 接受是 --ip-address 选项。当添加到 ipa-replica-install 时，--ip-address 仅接受与本地接口关联的 IP 地址。

D.2.2. 使用 DNS 安装副本

要安装带有集成 DNS 的副本，请按照 第 D.2.1 节 “在没有 DNS 的情况下安装 Replica” 中描述的 DNS 安装步骤进行，但将这些选项添加到 ipa-replica-install ：

- --setup-dns
- --forwarder

详情请查看 第 4.5.3 节 “使用 DNS 安装副本”。

例如：

```
[root@replica ~]# ipa-replica-install /var/lib/ipa/replica-info-replica.example.com.gpg --setup-dns --forwarder 198.51.100.0
```

运行 ipa-replica-install 后，确保创建了正确的 DNS 条目，并选择性地将其他 DNS 服务器添加为备份服务器。详情请查看 第 4.5.3 节 “使用 DNS 安装副本”。

D.2.3. 使用 Various CA 配置安装 Replica

警告

红帽强烈建议将 CA 服务安装到多台服务器中。有关安装包括 CA 服务的初始服务器副本的详情请参考 第 4.5.4 节 “使用 CA 安装副本”。

如果您只在一个服务器中安装 CA，则在 CA 服务器失败时可能会丢失 CA 配置且无法恢复。详情请查看 第 B.2.6 节 “恢复丢失的 CA 服务器”。

从安装了证书系统 CA 的服务器安装 Replica

要在初始服务器配置了集成 Red Hat Certificate System 实例时在副本上设置 CA （不管它是 root CA，还是从属到外部 CA），请遵循 第 D.2.1 节 “在没有 DNS 的情况下安装 Replica” 中描述的基本安装过程，但将 --setup-ca 选项添加到 ipa-replica-install 工具中。setup -ca 选项从初始服务器配置中复制 CA 配置。

```
[root@replica ~]# ipa-replica-install /var/lib/ipa/replica-info-replica.example.com.gpg --setup-ca
```

从没有安装证书系统 CA 的服务器安装 Replica

对于无 CA 副本安装，请按照 第 D.2.1 节 “在没有 DNS 的情况下安装 Replica” 中描述的基本流程，但在初始服务器上运行 ipa-replica-prepare 工具时添加以下选项：

- --dirsrv-cert-file
- --dirsrv-pin
- --http-cert-file

- --http-pin

详情请查看 第 4.5.5 节 “从没有 CA 的服务器安装 Replica”。

例如：

```
[root@server ~]# ipa-replica-prepare replica.example.com --dirsrv-cert-file /tmp/server.key --dirsrv-pin secret --http-cert-file /tmp/server.crt --http-cert-file /tmp/server.key --http-pin secret --dirsrv-cert-file /tmp/server.crt
```

D.2.4. 添加额外的复制协议

使用 ipa-replica-install 安装副本会在主服务器和副本之间创建初始复制协议。要将副本连接到其他服务器或副本，请使用 ipa-replica-manage 工具添加额外的协议。

如果 master 服务器和新副本安装了 CA，也会创建一个 CA 的复制协议。要向其他服务器或副本添加额外的 CA 复制协议，请使用 ipa-csreplica-manage 工具。

有关添加额外复制协议的详情请参考 第 D.3 节 “管理副本和复制协议”。

D.3. 管理副本和复制协议

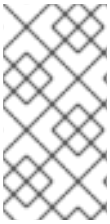
本章详细介绍了复制协议，并描述了如何管理它们。

注意

有关设置额外复制协议的指南，请参阅 第 4.2.2 节 “副本拓扑建议”。

D.3.1. 解释复制协议

副本加入到 复制协议中，后者 之间复制数据。复制协议为实：数据从第一个副本复制到另一个副本，以及从其他副本复制到第一个副本。



注意

初始复制协议由 ipa-replica-install 脚本在两个副本之间设置。有关安装初始副本的详情，请参阅 第 4 章 安装和卸载身份管理副本。

复制协议的类型

身份管理支持以下三种类型的复制协议：

- 用于复制目录数据（如用户、组和策略）的复制协议。您可以使用 ipa-replica-manage 工具管理这些协议。
- 复制协议以复制证书服务器数据.您可以使用 ipa-csreplica-manage 工具来管理这些协议。
- 同步协议，以与 Active Directory 服务器复制用户信息。本指南中未描述这些协议。有关同步 IdM 和 Active Directory 的文档，请参阅 Windows 集成指南中的 同步 Active Directory 和身份管理用户。

ipa-replica-manage 和 ipa-csreplica-manage 工具使用相同的格式和参数。本章后续小节描述了使用这些实用程序执行的最显著的复制管理操作。有关该工具的详情请参考 ipa-replica-manage(1) 和 ipa-csreplica-manage(1) man page。

D.3.2. 列出复制协议

要列出当前为副本配置的目录数据复制协议，请使用 ipa-replica-manage list 命令：

1. 运行不带参数的 ipa-replica-manage list，以列出复制拓扑中的所有副本。在输出中，找到所需的副本：

```
$ ipa-replica-manage list
server1.example.com: master
server2.example.com: master
server3.example.com: master
server4.example.com: master
```

2. 将副本的主机名添加到 ipa-replica-manage 列表中，以列出复制协议。

```
$ ipa-replica-manage list server1.example.com
server2.example.com: replica
server3.example.com: replica
```

输出显示 **server1.example.com** 发送更新的副本。

要列出证书服务器复制协议，请使用 **ipa-csreplica-manage list** 命令。

D.3.3. 创建和删除复制协议

创建复制协议

要创建新的复制协议，请使用 **ipa-replica-manage connect** 命令：

```
$ ipa-replica-manage connect server1.example.com server2.example.com
```

该命令将创建一个新的从 **server1.example.com** 到 **server2.example.com**，再从 **server2.example.com** 到 **server1.example.com** 的复制协议。

如果您只使用 **ipa-replica-manage connect** 指定一个服务器，IdM 会在本地主机和指定服务器之间创建一个复制协议。

要创建新证书服务器复制协议，请使用 **ipa-csreplica-manage connect** 命令。

删除复制协议

要删除复制协议，请使用 **ipa-replica-manage disconnect** 命令：

```
$ ipa-replica-manage disconnect server1.example.com server4.example.com
```

此命令将禁用从 **server1.example.com** 复制到 **server4. example.com**，以及 **server4.example.com** 到 **server1.example.com** 的复制。

ipa-replica-manage disconnect 命令只删除复制协议。它将两个服务器保留在身份管理复制拓扑中。要删除与副本相关的所有复制协议和数据，请使用 **ipa-replica-manage del** 命令，该命令从身份管理域完全删除副本。

```
$ ipa-replica-manage del server2.example.com
```

要删除证书服务器复制协议，请使用 **ipa-csreplica-manage disconnect** 命令。同样，要删除两个服务器之间的所有证书复制协议和数据，请使用 **ipa-csreplica-manage del** 命令。

D.3.4. 启动手动复制更新

相互之间的直接复制协议副本之间的数据更改几乎可以即时复制。但是，没有在直接复制协议中加入的副本不会以最快的速度接收更新。

在某些情况下，可能需要手动启动计划外复制更新。例如，在使副本脱机以进行维护前，等待计划更新的所有排队更改必须发送到一个或多个其他副本。在这种情况下，您可以在使副本离线前启动手动复制更新。

要手动启动复制更新，请使用 **ipa-replica-manage force-sync** 命令。运行 命令的本地主机是接收更新的副本。要指定发送更新的副本，请使用 **--from** 选项。

```
$ ipa-replica-manage force-sync --from server1.example.com
```

要为证书服务器数据启动复制更新，请使用 **ipa-csreplica-manage force-sync** 命令。

D.3.5. 重新初始化副本

如果副本已脱机一段时间或其数据库已损坏，您可以 重新初始化 它。重新初始化类似于初始化，如 第 4.5 节 “创建副本：简介” 所述。重新初始化使用一组更新的数据集刷新副本。例如，如果需要从备份中进行权威恢复，则可以使用重新初始化。

注意

等待常规复制更新或启动手动复制更新将无助于此情况。在这些复制更新期间，副本仅向彼此发送更改的条目。与重新初始化不同，复制更新不会刷新整个数据库。

要在副本上重新初始化数据复制协议，请使用 **ipa-replica-manage re-initialize** 命令。运行 命令的本地主机是重新初始化的副本。要指定获取数据的副本，请使用 **--from** 选项：

```
$ ipa-replica-manage re-initialize --from server1.example.com
```

要重新初始化证书服务器复制协议，请使用 **ipa-csreplica-manage re-initialize** 命令。

D.3.6. 删除副本

删除或降级副本 会从 拓扑中删除 IdM 副本，使其不再处理 IdM 请求。它还将主机计算机本身从 IdM 域中删除。

要删除副本，请在副本中执行这些步骤：

1. 列出 IdM 域的所有复制协议。在输出中，记下副本的主机名。

```
$ ipa-replica-manage list
server1.example.com: master
server2.example.com: master
server3.example.com: master
server4.example.com: master
```

2. 使用 **ipa-replica-manage del** 命令删除为副本配置的所有协议，以及所有关于副本的数据。

```
$ ipa-replica-manage del server3.example.com
```

3. 如果副本配置了自己的 CA，则使用 **ipa-csreplica-manage del** 命令删除所有证书服务器复制协议。

```
$ ipa-csreplica-manage del server3.example.com
```

注意

只有在副本本身配置了 IdM CA 时，才需要此步骤。如果只有主服务器或其他副本配置了 CA，则不需要这样做。

4. 卸载 IdM 服务器软件包。

```
$ ipa-server-install --uninstall -U
```

D.4. 将副本提升到主 CA 服务器

如果您的 IdM 部署使用嵌入的证书颁发机构(CA)，其中一个 IdM CA 服务器充当 master CA：它管理 CA 子系统证书的续订并生成证书撤销列表(CRL)。默认情况下，master CA 是系统管理员使用 ipa-

server-install 或 ipa-ca-install 命令在其上安装 CA 角色的第一个服务器。

如果您计划使 master CA 服务器离线或取消启用它，请 提升 副本以将其位置作为主 CA：

- 确保副本配置为处理 CA 子系统证书续订。请参阅 第 D.4.1 节 “更改 Which 服务器处理证书续订”。
- 配置副本以生成 CRL。请参阅 第 6.5.2.2 节 “更改 Which Server Generates CRL”。

D.4.1. 更改 Which 服务器处理证书续订

要更改哪些服务器处理证书续订，请在 IdM 服务器中按照以下流程：

1. 确定哪个服务器是当前的续订 master：

   - 在 Red Hat Enterprise Linux 7.3 及更新的版本中：

     ```
     $ ipa config-show | grep "CA renewal master"
     IPA CA renewal master: server.example.com
     ```

   - 在 Red Hat Enterprise Linux 7.2 及更早版本中：

     ```
     $ ldapsearch -H ldap://$HOSTNAME -D 'cn=Directory Manager' -W -b
     'cn=masters,cn=ipa,cn=etc,dc=example,dc=com' '(&(cn=CA)
     (ipaConfigString=caRenewalMaster))' dn
     ...
     # CA, server.example.com, masters, ipa, etc, example.com
     dn: cn=CA,cn=server.example.com,cn=masters,cn=ipa,cn=etc,dc=example,dc=com
     ...
     ```

   在这两个示例中，server.example.com 是当前的续订 master。

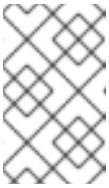
2. 设置其他服务器来处理证书续订：

- 在 Red Hat Enterprise Linux 7.4 及更新的版本中：

```
# ipa config-mod --ca-renewal-master-server new_server.example.com
```

- 在 Red Hat Enterprise Linux 7.3 及更早版本中：

```
# ipa-csreplica-manage set-renewal-master
```



注意

此命令设置运行命令的服务器，作为新的续订 master。

这些命令还会自动重新配置以前的 CA 从续订 master 克隆。

附录 E. 身份管理服务器端口注意事项

E.1. 身份管理组件和相关服务

表 E.1 “身份管理组件和相关服务” 列出个别身份管理服务从外部公开的端口。

表 E.1. 身份管理组件和相关服务

组件	服务	允许访问的端口
身份管理框架*	基于 Apache 的 Web 服务以及到其他服务的路由	HTTPS 端口 *443(TCP/TCP6)*
LDAP 目录服务器*	389-DS 实例	端口 *389(TCP/TCP6)* ：使用 StartTLS 扩展或 SASL GSSAPI 来保护连接 端口 *636(TCP/TCP6)* ：通过 SSL 的普通 LDAP 流量 端口 *389(UDP)* ：无连接 LDAP 访问，促进与 Active Directory 服务的集成
Kerberos 密钥分发中心*	krb5kdc	端口 *88（TCP/TCP6 和 UDP/UDP6）* ：普通的 Kerberos 流量 port *464（TCP/TCP6 和 UDP/UDP6）*:Kerberos 密码更改协议访问
Kerberos 管理员守护进程*	kadmind	端口 *749(TCP/TCP6)*:Kerberos 远程管理协议在内部使用
custodia 密钥管理*	custodia	HTTPS 端口 *443(TCP/TCP6)*: 作为身份管理框架的一部分
系统安全服务守护进程*	sssd	HTTPS 端口 *443(TCP/TCP6)*: 作为身份管理框架的一部分
MS-KKDCP 代理**	通过 HTTPS 对 Kerberos 的代理访问	HTTPS 端口 *443(TCP/TCP6)*: 作为身份管理框架的一部分
证书颁发机构	Tomcat 顶部的 Dogtag 实例	HTTPS 端口 *443(TCP/TCP6)*: 作为身份管理框架的一部分 HTTP 访问通过端口 *80(TCP/TCP6)* 进行内部重定向到端口 *8080(TCP/TCP6)*，根据为身份管理设置的 Apache 规则；检索的信息是 OCSP 响应器和证书状态（证书撤销列表） 在内部，HTTPS 通过端口 *8443(TCP/TCP6)* 访问：用于 CA 管理目的 在 IPA 主机上，使用端口 *8005 和 8009(TCP/TCP6)* 在 *127.0.0.1* 和 *::1* 本地接口地址上运行证书颁发机构服务的组件

组件	服务	允许访问的端口
DNS	named	端口 *53（TCP/TCP6 和 UDP/UDP6）*: 标准 DNS 解析器 端口 *953(TCP/TCP6)*:*127.0.0.1* 和 *::1* 本地接口地址的 BIND 服务远程控制
Active Directory 集成	Samba 服务（smbd、winbindd）	端口 *135(TCP/TCP6)*:DCE RPC 端点映射器（smbd 守护进程） 端口 *138(TCP/TCP6)*，NetBIOS 数据报服务（可选，需要 nmbd 守护进程才能运行） 端口 *139(TCP/TCP6)*，NetBIOS 会话服务（smbd 守护进程） 端口 *445(TCP/TCP6)*、基于 TCP/TCP6（smbd 守护进程）的 SMB 协议. 为 DCE RPC 端点服务动态打开端口 *49152-65535(TCP/TCP6)*
证书颁发机构 Vault	Dogtag 实例的 KRA 组件	HTTPS 端口 *443(TCP/TCP6)*: 作为身份管理框架的一部分 HTTP 访问通过端口 *80(TCP/TCP6)*，但内部通过 Apache 规则重定向到端口 *8080(TCP/TCP6)* ：对于 OCSP 响应器和证书状态(Certificate Revocation List) 在内部，HTTPS 通过端口 *8443(TCP/TCP6)* 访问：用于 CA 管理目的 在 IPA 主机上，使用端口 *8005 和 8009(TCP/TCP6)* 在 *127.0.0.1* 和 *::1* 本地接口地址上运行证书颁发机构服务的组件

**** 标记有星号的服务在每个身份管理部署中都处于活跃状态。***

***** MS-KKDCP 代理组件是可选的，但默认启用。***

附录 F. IDM 中的显著变化

某些 IdM 版本引入了新命令或替换现有命令。另外，有时配置或安装过程有很大变化。本附录描述了最重要的更改。

如需了解更详细的更改列表，请参阅 Red Hat Enterprise Linux(RHEL)7 发行注记。

在 RHEL 7.7 上运行的 IdM 4.6

- 添加了 ipa-cert-fix 工具，以便在 IdM 离线时续订系统证书。详情请查看 第 26.2.3 节 “IdM 离线时续订过期的系统证书”。
- IdM 现在支持证书的 SAN 扩展中的 IP 地址：在某些情况下，管理员需要在 Subject Alternative Name(SAN)扩展中使用 IP 地址签发证书。从此发行版本开始，如果该地址在 IdM DNS 服务中进行管理并与主题主机或服务主体关联，管理员可以在 SAN 扩展中设置 IP 地址。
- IdM 现在防止使用单标签域名，如 .company。IdM 域必须由一个或多个子域和一个顶层域组成，如 example.com 或 company.example.com。
- 有关此发行版本的详情，请查看 Red Hat Enterprise Linux 7.7 发行注记 中的以下部分：
  - 新功能 - 身份验证和互操作性
  - 重要的程序错误修复 - 身份验证和互操作性

在 RHEL 7.6 上运行的 IdM 4.6

- 有关此发行版本的更改，请参阅 Red Hat Enterprise Linux 7.6 发行注记 中的以下部分：
  - 新功能 - 身份验证和互操作性
  - 重要的程序错误修复 - 身份验证和互操作性

在 RHEL 7.5 上运行的 IdM 4.5

- 有关此发行版本的更改，请参阅 Red Hat Enterprise Linux 7.5 发行注记 中的以下部分：
  - 新功能 - 身份验证和互操作性
  - 重要的程序错误修复 - 身份验证和互操作性

在 RHEL 7.4 上运行的 IdM 4.5

- 此版本将客户端 HTTPS 连接的 SSL 后端从网络安全服务(NSS)改为 OpenSSL。因此，注册授权机构(RA)现在将其证书存储在 /var/lib/ipa/ 目录中，而不是 NSS 数据库。
- 有关此发行版本的详情，请查看 Red Hat Enterprise Linux 7.4 发行注记 中的以下部分：
  - 新功能 - 身份验证和互操作性
  - 重要的程序错误修复 - 身份验证和互操作性

在 RHEL 7.3 上运行的 IdM 4.4

- 新的 ipa replica-manage clean-dangling-ruv 命令可让管理员从卸载的副本中删除所有相对更新向量(RUV)。
- 新的 ipa server-del 命令可让管理员卸载 IdM 服务器。
- 此版本中介绍的以下命令可让管理员管理 IdM 证书颁发机构(CA)：
  - ipa ca-add
  - ipd ca-del
  - ipa ca-enable

- *ipa ca-disble*
- *ipa ca-find*
- *ipa ca-mod*
- *ipa ca-show*

- *此版本中介绍的以下命令替换了 ipa-replica manage 命令来管理复制协议：*
  - *ipa topology-configure*
  - *ipa topologysegment-mod*
  - *ipa topologysegment-del*
  - *ipa topologysuffix-add*
  - *ipa topologysuffix-show*
  - *ipa topologysuffix-verify*

- *此版本引入的以下命令可让管理员显示存储在 cn=masters,cn=ipa,cn=etc,domain_suffix 条目中的 IdM 服务器列表：*
  - *ipa server-find*
  - *ipa server-show*

- certmonger 帮助程序脚本已从 /usr/lib64/ipa/certmonger/ 移到 /usr/libexec/ipa/certmonger/ 目录中。

- 这个版本引入了域级别，并使用以下命令来显示和设置域级别：

  - ipa domainlevel-set

  - ipa domainlevel-show

- 有关此版本中的详情，请查看 Red Hat Enterprise Linux 7.3 发行注记 中的以下部分：

  - 新功能 - 身份验证和互操作性

  - 重要的程序错误修复 - 身份验证和互操作性

在 RHEL 7.2 上运行的 IdM 4.2

- 支持多个证书配置集和用户证书：身份管理现在支持多个配置文件来发布服务器和其他证书，而不是只支持单个服务器证书配置文件。配置文件存储在 Directory 服务器中，并在 IdM 副本之间共享。另外，管理员现在可以向个人用户发布证书。在以前的版本中，只能向主机和服务发布证书。

- 有关此版本中的进一步更改，请参阅 Red Hat Enterprise Linux 7.2 发行注记 中的新功能 - 身份验证和互操作性 部分。

在 RHEL 7.1 上运行的 IdM 4.1

- 此版本中介绍的以下命令替换了 ipa-getkeytab -r 命令来检索 keytabs 并设置检索权限：

  - ipa-host-allow-retrieve-keytab

  - ipa-host-disallow-retrieve-keytab

- *ipa-host-allow-create-keytab*
- *ipa-host-disallow-create-keytab*
- *ipa-service-allow-retrieve-keytab*
- *ipa-service-disallow-retrieve-keytab*
- *ipa-service-allow-create-keytab*
- *ipa-service-disallow-create-keytab*

- *有关此版本中的进一步更改，请参阅 Red Hat Enterprise Linux 7.1 发行注记 中的新功能 - 身份验证和互操作性 部分。*

在 RHEL 7.0 上运行的 IdM 3.3

- *有关此发行版本中的变化，请参阅 Red Hat Enterprise Linux 7.0 发行注记 中的新功能 和互操作性 部分。*

附录 G. 修订历史记录

请注意，修订号与本手册的版本相关，与 Red Hat Enterprise Linux 版本号无关。

修订 7.0-53 Tue Feb 16 2021 Florian Delehaye
各种说明，特别是 IdM 服务和配置文件的说明，以及其他细微修正。

修订 7.0-52 Tue Sep 29 2020 Florian Delehaye
发布 7.9 GA 的文档版本。

修订 7.0-51 Tue Mar 31 2020 Florian Delehaye
发布 7.8 GA 的文档版本。

修订 7.0-50 Wed Aug 28 2019 Marc Muehlfeld
*在 IdM 附录中添加了 Notable 更改。*少许更新。

修订 7.0-49 Tue Aug 06 2019 Marc Muehlfeld
7.7 GA 出版物的文档版本。

修订 7.0-48 Fri Jun 21 2019 Marc Muehlfeld
添加了在 *IdM 离线时续订过期的系统证书* 部分。

修订 7.0-47 Thu Jun 13 2019 Marc Muehlfeld
添加了有关配置隐藏副本的内容。

修订 7.0-46 Wed Jun 04 2019 Marc Muehlfeld
添加了部分 *启用最近成功 Kerberos 身份验证的跟踪*.多个小编辑.

修订 7.0-45 Tue Apr 09 2019 Marc Muehlfeld
添加了 *Web UI Session Length*，并添加了两个有关身份验证指示器和几个小编辑的部分。

修订 7.0-44 Thu Nov 22 2018 Filip Hanzelka
*在安装 和卸载 IdM 服务器 章节中添加了身份管理组件和相关服务以及副*编辑。

修订 7.0-43 Mon Oct 29 2018 Lucie Maňásková
准备发布 7.6 GA 的文档.

修订 7.0-42 Tue Jun 26 2018 Lucie Maňásková
使用集成式 IdM CA 更新了管理证书.其他更新.

修订 7.0-41 Fri Apr 23 2018 Filip Hanzelka
添加了确定 Kerberos 票据的生命周期. *其他小修复。*

修订 7.0-40 Fri Apr 6 2018 Lucie Maňásková
准备 7.5 GA 发布文档.

修订 7.0-39 Wed Mar 14 2018 Filip Hanzelka
细微更新.

修订 7.0-38 Wed Feb 28 2018 Lucie Maňásková
细微更新.

修订 7.0-37 Mon Feb 12 2018 Aneta Šteflová Petrová
添加了 *用户无法访问其 Vault，因为使用无效"添加"特权*.其他小修复。

修订 7.0-36 Mon Jan 29 2018 Aneta Šteflová Petrová
更新 *定义 SELinux 用户映射*.其他小修复。

修订 7.0-35 Fri Dec 15 2017 Aneta Šteflová Petrová
更新 *的管理主机*.其他小修复。

修订 7.0-34 Mon Dec 4 2017 Aneta Šteflová Petrová
在 IdM 中添加了 Kerberos PKINIT 身份验证.*更新了 IdM 用户的访问控制*。其他小修复。

修订 7.0-33 Mon Nov 20 2017 Aneta Šteflová Petrová
更新了章节 *用户和组架构*， 以及 *定义密码策略*.

修订 7.0-32 Mon Oct 9 2017 Aneta Šteflová Petrová
小修复.

修订 7.0-31 Tue Sep 12 2017 Aneta Šteflová Petrová
更新了几个 Web UI 屏幕截图和程序。*身份管理中智能卡身份验证的*次要更新.

修订 7.0-30 Mon Aug 28 2017 Aneta Šteflová Petrová
更新了身份管理和 *身份管理配置文件和目录中的智能卡身份验证*。

修订 7.0-29 Tue Jul 18 2017 Aneta Šteflová Petrová
7.4 GA 出版物的文件版本。

修订 7.0-28 Mon Apr 24 2017 Aneta Šteflová Petrová
更新和合并管理用户组、主机组和自动成员身份。其他次要更新.

修订 7.0-27 Mon Apr 10 2017 Aneta Šteflová Petrová
添加了为身份管理配置 TLS.各种小修复和更新。

修订 7.0-26 Mon Mar 27 2017 Aneta Šteflová Petrová
添加了客户端安装后注意事项和启用密码重置.其他次要更新.

修订 7.0-25 Mon Feb 27 2017 Aneta Šteflová Petrová
有关管理 Kerberos 域、升级和 HBAC 的章节已更新。各个章节中的其他更新.

修订 7.0-24 Wed Dec 7 2016 Aneta Šteflová Petrová
更新了自动成员和密码策略章节。添加了 NIS 支持插件的描述。其他次要更新.

修订 7.0-23 Tue Oct 18 2016 Aneta Šteflová Petrová
7.3 GA 出版物版本。

修订 7.0-22 Fri Jul 29 2016 Aneta Petrová
在 上使用 vault 添加一章.

修订 7.0-21 Thu Jul 28 2016 Marc Muehlfeld
更新的简介，其他小修复.

修订 7.0-19 Tue Jun 28 2016 Aneta Petrová
更新的示意图.向简介章节添加了使用 IdM 的优势部分。其他小修复和优化.

修订 7.0-18 Fri Jun 10 2016 Aneta Petrová
更新了简介、服务器安装和故障排除章节。其他修复.

修订 7.0-17 Fri May 27 2016 Aneta Petrová

添加了用户生命周期图。

修订 7.0-16 Thu Mar 24 2016 Aneta Petrová
添加了用户生命周期.更新了用户帐户、用户身份验证和管理副本章节。

修订 7.0-15 Thu Mar 03 2016 Aneta Petrová
更新了多个 DNS 部分。将 PAM 服务的域限制到系统级身份验证指南中.

修订 7.0-14 Tue Feb 09 2016 Aneta Petrová
添加了智能卡、ID 视图和 OTP。将卸载过程移到安装章节中。其他次要更新.

修订 7.0-13 Thu Nov 19 2015 Aneta Petrová
证书配置文件管理和将副本提升到 master 的次要更新。

修订 7.0-12 Fri Nov 13 2015 Aneta Petrová
7.2 GA 版本的版本，更新到 DNS 和其他部分。

修订 7.0-11 Thu Nov 12 2015 Aneta Petrová
7.2 GA 版本。

修订 7.0-10 Fri Mar 13 2015 Tomáš Čapek
最新编辑 7.1 的异步更新.

修订 7.0-8 Wed Feb 25 2015 Tomáš Čapek
7.1 GA 版本.

修订 7.0-6 Fri Dec 05 2014 Tomáš Čapek
重新构建 以更新启动页面上的排序顺序。

修订 7.0-4 Wed Jun 11 2014 Ella Deon Ballard
初始版本.