# Red Hat Enterprise Linux 9

# 将 RHEL 系统直接与 Windows Active Directory 集成

将 RHEL 主机加入到 AD 中,并访问 AD 中的资源

Last Updated: 2024-06-26

# Red Hat Enterprise Linux 9 将 RHEL 系统直接与 Windows Active Directory 集成

将 RHEL 主机加入到 AD 中,并访问 AD 中的资源

## 法律通告



## 摘要

管理员可以使用系统安全服务守护进程(SSSD)或 Samba Winbind 服务将 Red Hat Enterprise Linux (RHEL)主机加入到活动目录(AD)域中,以访问 AD 资源。或者，也可以使用受管服务帐户(MSA)访问没有域集成的 AD 资源。

# 目录

# 对红帽文档提供反馈

我们感谢您对我们文档的反馈。让我们了解如何改进它。

**通过 Jira 提交反馈（需要帐户）**

1. 登录到 Jira 网站。
2. 在顶部导航栏中点 **Create**
3. 在 **Summary** 字段中输入描述性标题。
4. 在 **Description** 字段中输入您对改进的建议。包括文档相关部分的链接。
5. 点对话框底部的 **Create**。

# 第 1 章 使用 SSSD 将 RHEL 系统直接连接到 AD

您需要两个组件才能将 RHEL 系统连接到 Active Directory(AD)。一个组件 (SSSD) 与中央身份和验证源交互，另一个组件 (**realmd**) 会检测到可用的域并配置底层 RHEL 系统服务，以连接到域。

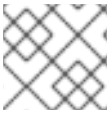
这部分论述了使用系统安全服务守护进程 (SSSD) 将 RHEL 系统连接到 Active Directory (AD)。

- 使用 SSSD 直接集成概述
- 支持直接集成的 Windows 平台
- 直接连接到 AD
- AD 供应商如何处理动态 DNS 更新
- 修改 AD 供应商的动态 DNS 设置
- AD 供应商如何处理可信域
- 使用 SSSD 覆盖 Active Directory 站点自动发现
- realm 命令

## 1.1. 使用 SSSD 直接集成概述

您可以使用 SSSD 访问用户目录用于身份验证和授权，并通过带有用户缓存的通用框架进以允许离线登录。SSSD 是高度可配置的；它提供了可插拔验证模块(PAM)和名称交换服务(NSS)集成和数据库，用于存储本地用户以及从中央服务器检索的扩展用户数据。在把 RHEL 系统与以下身份服务器类型之一连接时，推荐使用 SSSD：

- Active Directory
- RHEL 中的身份管理（IdM）
- 任何通用 LDAP 或 Kerberos 服务器

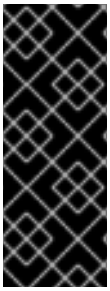


**注意**

默认情况下，直接与 SSSD 集成在一个 AD 林中正常工作。

配置 SSSD 以将 Linux 系统直接与 AD 集成的最便捷方法是使用 **realmd** 服务。它允许调用者以标准的方式配置网络身份验证和域成员资格。**realmd** 服务自动发现有关可访问 domain 和 realm 的信息，且不需要高级配置就可以加入到 domain 和 realm。

您可以使用 SSSD 与 AD 直接和间接集成，并允许您从一个集成方法切换到另一个集成方法。直接集成是将 RHEL 系统引入 AD 环境的简单方法。但是，随着 RHEL 系统的共享增加，您的部署通常需要更好地管理与身份相关的策略，如基于主机的访问控制、sudo 或 SELinux 用户映射。在初始阶段，您可以在本地配置文件中维护 RHEL 系统的这些配置。但是，在有大量系统的情况下，使用一个置备系统（如 Red Hat Satellite）可以使对配置文件进行分发和管理的任务变得更为容易。当直接集成不再可以满足环境扩展的要求时，应该考虑使用间接集成。有关从直接集成(RHEL 客户端位于 AD 域中)移到间接集成（带有到 AD 的信任的 IdM）的更多信息，请参阅 将 RHEL 客户端从 AD 域移到 IdM 服务器。



**重要**

如果 IdM 处于 FIPS 模式，IdM-AD 集成无法正常工作，因为 AD 只支持使用 RC4 或 AES HMAC-SHA1 加密，而 FIPS 模式中的 RHEL 9 默认只允许 AES HMAC-SHA2。要在 RHEL 9 中启用 AES HMAC-SHA1，请输入 **# update-crypto-policies --set FIPS:AD-SUPPORT**。

IdM 不支持更严格的 **FIPS:OSPP** 加密策略，该策略只应用于通用标准评估的系统。

有关哪个类型的集成适合您的用例的更多信息，请参阅在间接集成和直接集成之间做决定。

**其他资源**

- **realm(8)** 手册页。
- **sssd-ad(5)** 手册页。
- **sssd(8)** 手册页。

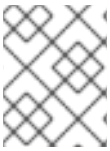
## 1.2. 支持直接集成的 WINDOWS 平台

您可以直接将 RHEL 系统与使用以下林和域功能级别的 Active Directory 网站集成：

- 林功能级别范围：Windows Server 2008 - Windows 服务器 2016
- 域功能级别范围：Windows Server 2008 - Windows 服务器 2016

在以下支持的操作系统中测试了直接集成：

- Windows Server 2022（RHEL 9.1 及更高版本）
- Windows Server 2019
- Windows Server 2016
- Windows Server 2012 R2



**注意**

Windows Server 2019 和 Windows Server 2022 没有引入新的功能级。Windows Server 2019 和 Windows Server 2022 使用的最高功能级是 Windows Server 2016。

## 1.3. 直接连接到 AD

系统安全服务守护进程 (SSSD) 是推荐的组件，用于将 Red Hat Enterprise Linux (RHEL) 系统与 Active Directory (AD) 连接。本节论述了如何使用 ID 映射（SSSD 默认使用）或者使用 POSIX 属性直接与 AD 集成。

- 用于与 AD 集成的选项：使用 ID 映射或 POSIX 属性
- 使用 SSSD 发现并加入 AD 域
- 使用 Active Directory 中定义的 POSIX 属性连接到 AD
- 使用 SSSD 连接到不同 AD 林中的多个域

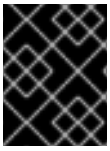
**重要**

在将您的系统加入到 AD 之前，请确保按照 基本预检查步骤：RHEL 使用 'adcli', 'realm' 和 'net' 命令加入活动目录 中的流程来正确配置您的系统。

### 1.3.1. 用于与 AD 集成的选项： 使用 ID 映射或 POSIX 属性

Linux 和 Windows 系统为用户和组群使用不同的标识符：

- Linux 使用 *用户 ID*（UID）和 *组群 ID*（GID）。请参阅 *配置基本系统设置* 中的 管理用户和组帐户简介。Linux UID 和 GID 符合 POSIX 标准。
- Windows 使用*安全 ID* (SID)。



**重要**

将 RHEL 系统连接到 AD 后，您可以使用 AD 用户名和密码进行身份验证。不要创建名称与 Windows 用户相同的 Linux 用户，因为重复名称可能会导致冲突并中断身份验证过程。

要以 AD 用户身份验证 RHEL 系统，您必须分配了 UID 和 GID。SSSD 提供了使用 ID 映射或 POSIX 属性与 AD 集成的选项。默认是使用 ID 映射。

#### 为 AD 用户自动生成新的 UID 和 GID

SSSD 可以使用 AD 用户的 SID 在名为 ID *映射的过程中以算法生成 POSIX ID*。ID 映射会在 AD 中的 SID 和 Linux 中的 ID 之间创建一个映射。

- 当 SSSD 检测到新的 AD 域时，它会为这个新域分配一个可用的 ID 范围。
- 当 AD 用户第一次登录到 SSSD 客户端机器时，SSSD 会在 SSSD 缓存中为用户创建一个条目，包括基于用户的 SID 和该域的 ID 范围的 UID。
- 由于 AD 用户的 ID 是以一致的方式从同一 SID 生成的，所以用户在登录到任何 Red Hat Enterprise Linux 系统时都有相同的 UID 和 GID。

请参阅使用 SSSD 发现并加入 AD 域。

**注意**

当所有客户端系统都使用 SSSD 将 SID 映射到 Linux ID 时，映射是一致的。如果有些客户端使用不同的软件，请选择以下之一：

- 确定所有客户端都使用相同的映射算法。
- 使用 AD 中定义的显式 POSIX 属性。

#### 使用 AD 中定义的 POSIX 属性

AD 可以创建并存储 POSIX 属性，如 **uidNumber**、**gidNumber**、**UNIXHomeDirectory** 或 **loginShell**。

当使用上述 ID 映射时，SSSD 会创建新的 UID 和 GID，这将覆盖 AD 中定义的值。要保留 AD 定义的值，必须在 SSSD 中禁用 ID 映射。

请参阅使用 Active Directory 中定义的 POSIX 属性连接到 AD。

### 1.3.2. 使用 SSSD 发现并加入 AD 域

按照以下流程发现 AD 域，并使用 SSSD 将 RHEL 系统连接到那个域。

**先决条件**

- 确保 AD 域控制器上的以下端口已打开，并可以被 RHEL 主机访问。

  **表 1.1. 使用 SSSD 将 Linux 系统直接集成到 AD 所需的端口**

| 服务 | 端口 | 协议 | 备注 |
| --- | --- | --- | --- |
| DNS | 53 | UDP 和 TCP | |
| LDAP | 389 | UDP 和 TCP | |
| Samba | 445 | UDP 和 TCP | 对于 AD 组策略对象 (GPO) |
| Kerberos | 88 | UDP 和 TCP | |
| Kerberos | 464 | UDP 和 TCP | **kadmin** 用来设置和更改密码 |
| LDAP 全局目录 | 3268 | TCP | 如果使用了 **id_provider = ad** 选项 |
| NTP | 123 | UDP | 可选 |

- 确保您为 DNS 使用 AD 域控制器服务器。
- 验证两个系统中的系统时间已被同步。这样可确保 Kerberos 正常工作。

**流程**

1. 安装以下软件包：

```
# dnf install samba-common-tools realmd oddjob oddjob-mkhomedir sssd adcli krb5-workstation
```

2. 要显示特定域的信息，请运行 **realm discover** 并添加您要发现的域名称：

```
# realm discover ad.example.com
ad.example.com
  type: kerberos
  realm-name: AD.EXAMPLE.COM
  domain-name: ad.example.com
  configured: no
  server-software: active-directory
  client-software: sssd
  required-package: oddjob
  required-package: oddjob-mkhomedir
```

```
required-package: sssd
required-package: adcli
required-package: samba-common
```

**realmd** 系统使用 DNS SRV 查找自动查找这个域中的域控制器。

**注意**

**realmd** 系统可以发现 Active Directory 和 Identity Management 域。如果环境中存在这两个域，您可以使用 **--server-software=active-directory** 选项将发现结果限制为特定类型的服务器。

3. 使用 **realm join** 命令配置本地 RHEL 系统。 **realmd** 套件自动编辑所有必需的配置文件。例如，对于名为 **ad.example.com** 的域：

```
# realm join ad.example.com
```

**验证步骤**

- 显示 AD 用户详情，如管理员用户：

```
# getent passwd administrator@ad.example.com
administrator@ad.example.com:*:1450400500:1450400513:Administrator:/home/administrator
@ad.example.com:/bin/bash
```

**其他资源**

- 请参阅 **realm(8)** 手册页。
- 请参阅 **nmcli(1)** 手册页。

### 1.3.3. 使用 Active Directory 中定义的 POSIX 属性连接到 AD

为获得最佳性能，请将 POSIX 属性发布到 AD 全局目录。如果全局目录中没有 POSIX 属性，SSSD 会直接连接到 LDAP 端口上的单个域控制器。

**先决条件**

- 确保 RHEL 主机上的以下端口已为 AD 域控制器打开并可以被访问。

**表 1.2. 使用 SSSD 将 Linux 系统直接集成到 AD 所需的端口**

| 服务 | 端口 | 协议 | 备注 |
|---|---|---|---|
| DNS | 53 | UDP 和 TCP | |
| LDAP | 389 | UDP 和 TCP | |
| Kerberos | 88 | UDP 和 TCP | |

| 服务 | 端口 | 协议 | 备注 |
| --- | --- | --- | --- |
| Kerberos | 464 | UDP 和 TCP | kadmin 用来设置和更改密码 |
| LDAP 全局目录 | 3268 | TCP | 如果使用了 **id_provider = ad** 选项 |
| NTP | 123 | UDP | 可选 |

- 确保您为 DNS 使用 AD 域控制器服务器。
- 验证两个系统中的系统时间已被同步。这样可确保 Kerberos 正常工作。

**流程**

1. 安装以下软件包：

```
# dnf install realmd oddjob oddjob-mkhomedir sssd adcli krb5-workstation
```

2. 使用 **realm join** 命令及 **--automatic-id-mapping=no** 选项，配置本地 RHEL 系统并禁用 ID 映射。**realmd** 套件自动编辑所有必需的配置文件。例如，对于名为 **ad.example.com** 的域：

```
# realm join --automatic-id-mapping=no ad.example.com
```

3. 如果您已经加入某个域，可以在 SSSD 中手动禁用 ID 映射：
   a. 打开 **/etc/sssd/sssd.conf** 文件：
   b. 在 AD domain 部分中，添加 **ldap_id_mapping = false** 设置。
   c. 删除 SSSD 缓存：

```
rm -f /var/lib/sss/db/*
```

   d. 重启 SSSD：

```
systemctl restart sssd
```

SSSD 现在使用 AD 中的 POSIX 属性，而不是在本地创建它们。

**注意**

您必须在 AD 中的用户配置了相关的 POSIX 属性（**uidNumber**, **gidNumber**, **unixHomeDirectory**, 和 **loginShell**）。

**验证步骤**

- 显示 AD 用户详情，如管理员用户：

```
# getent passwd administrator@ad.example.com
administrator@ad.example.com:*:10000:10000:Administrator:/home/Administrator:/bin/bash
```

**其他资源**

- 有关 ID 映射和 **ldap_id_mapping** 参数的详情，请查看 **sssd-ldap(8)** man page。

### 1.3.4. 使用 SSSD 连接到不同 AD 林中的多个域

您可以使用 Active Directory (AD) 受管服务帐户 (MSA)从一个不同、直接没有信任的林访问 AD 域。

请参阅使用受管服务帐户访问 AD。

## 1.4. AD 供应商如何处理动态 DNS 更新

Active Directory（AD）通过超时（*aging*）和删除（*scavenging*）不活跃的记录来主动维护 DNS 记录。

默认情况下，SSSD 服务会按照以下间隔刷新 RHEL 客户端的 DNS 记录：

- 身份提供程序每次上线时。
- 每次 RHEL 系统重启时。
- 在 **/etc/sssd/sssd.conf** 配置文件中的 **dyndns_refresh_interval** 选项指定的时间间隔。默认值为 **86400** 秒（24 小时）。

**注意**

如果将 **dyndns_refresh_interval** 选项设置为与 DHCP 租期相同的间隔，您可以在 IP 租期被续订后更新 DNS 记录。

SSSD 使用 DNS 的 Kerberos/GSSAPI(GSS-TSIG)向 AD 服务器发送动态 DNS 更新。这意味着您只需要启用到 AD 的安全连接。

**其他资源**

- **sssd-ad(5)** 手册页。

## 1.5. 修改 AD 供应商的动态 DNS 设置

系统安全服务守护进程(SSSD)服务将 Red Hat Enterprise Linux(RHEL)客户端的 DNS 记录以默认间隔刷新到 AD 环境。以下流程调整这些间隔。

**先决条件**

- 您已使用 SSSD 服务将 RHEL 主机加入到 Active Directory 环境。
- 您需要 **root** 权限来编辑 **/etc/sssd/sssd.conf** 配置文件。

**流程**

1. 在文本编辑器中打开 **/etc/sssd/sssd.conf** 配置文件。

2. 在您的 AD 域的 **[domain]** 部分添加以下选项，将 DNS 记录刷新间隔设置为 12 小时，禁用更新 PTR 记录，并将 DNS 记录时间设置为 1 小时。

```
[domain/ad.example.com]
id_provider = ad
...
dyndns_refresh_interval = 43200
dyndns_update_ptr = false
dyndns_ttl = 3600
```

3. 保存并关闭 **/etc/sssd/sssd.conf** 配置文件。

4. 重启 SSSD 服务以载入配置更改。

```
[root@client ~]# systemctl restart sssd
```

**注意**

您可以通过将 **sssd.conf** 文件中的 **dyndns_update** 选项设置为 **false** 来禁用动态 DNS 更新：

```
[domain/ad.example.com]
id_provider = ad
...

dyndns_update = false
```

**其他资源**

- AD 供应商如何处理动态 DNS 更新
- **sssd-ad(5)** man page

## 1.6. AD 供应商如何处理可信域

如果您在 **/etc/sssd/sssd.conf** 配置文件中设置了 **id_provider = ad** 选项，SSSD 会处理可信域，如下所示：

- SSSD 只支持单个 AD 林中的域。如果 SSSD 需要从多个林访问多个域，请考虑使用带有信任的 IPA（首选方式）或 **winbindd** 服务，而不是 SSSD。
- 默认情况下，SSSD 会发现林中的所有域，并在一个可信域中的对象请求到达时，SSSD 会尝试解析它。
  如果可信域无法访问或地理距离非常长，这使得它们的速度较慢，您可以在 **/etc/sssd/sssd.conf** 中设置 **ad_enabled_domains** 参数，以限制 SSSD 从哪些可信域解析对象。
- 默认情况下，您必须使用完全限定的用户名从可信域解析用户。

**其他资源**

- **sssd.conf(5)** 手册页。

## 1.7. 使用 SSSD 覆盖 ACTIVE DIRECTORY 站点自动发现

Active Directory(AD)林可能非常大，带有许多不同的域控制器、域、子域和物理站点。AD 使用**站点**的概念来识别其域控制器的物理位置。这可让客户端连接到地理上接近的域控制器，这会增加客户端性能。

本节介绍了 SSSD 如何使用自动发现来查找 AD 站点连接，以及如何手动覆盖自动发现并手动指定站点。

### 1.7.1. SSSD 如何处理 AD 站点自动发现

默认情况下，SSSD 客户端使用自动发现（autodiscovery）功能查找其 AD 站点并连接到最接近的域控制器。此过程由这些步骤组成：

1. SSSD 执行 SRV 查询来查找域中的域控制器(DC)。SSSD 从 SSSD 配置文件中的 **dns_discovery_domain** 或 **ad_domain** 选项读取发现域。
2. SSSD 在 3 个批处理中对这些数据中心执行 Connection-Less LDAP(CLDAP)ping 命令，以避免 ping 过多的 DC，并避免无法访问的 DCs 出现超时。如果 SSSD 在任何这些批处理过程中收到站点和林信息，它会跳过批处理的其余部分。
3. SSSD 创建并保存特定于站点和备份服务器的列表。

### 1.7.2. 覆盖 AD 站点自动发现

要覆盖自动发现过程，请通过将 **ad_site** 选项添加到 **/etc/sssd/sssd.conf** 文件的 **[domain]** 部分，指定要连接的 AD 站点。这个示例将客户端配置为连接到 **ExampleSite** AD 站点。

**先决条件**

- 已使用 SSSD 服务将 RHEL 主机加入到 Active Directory 环境中。
- 您可以以 **root** 用户身份进行身份验证，以便可以编辑 **/etc/sssd/sssd.conf** 配置文件。

**流程**

1. 在文本编辑器中打开 **/etc/sssd/sssd.conf** 文件。
2. 将 **ad_site** 选项添加到 AD 域的 **[domain]** 部分：

```
[domain/ad.example.com]
id_provider = ad
...
ad_site = ExampleSite
```

3. 保存并关闭 **/etc/sssd/sssd.conf** 配置文件。
4. 重启 SSSD 服务以载入配置更改：

```
# systemctl restart sssd
```

## 1.8. REALM 命令

**realmd** 系统有两个主要任务：

- 在一个域中管理系统注册。
- 控制允许哪些域用户访问本地系统资源。

在 **realmd** 中，使用命令行工具 **realm** 来运行命令。大多数 **realm** 命令要求用户指定程序要执行的操作，以及要执行操作的域或用户帐户等实体。

**表 1.3. realmd 命令**

| 命令 | 描述 |
| --- | --- |
| *realm 命令* | |
| discover | 对网络中的域运行发现扫描。 |
| join | 将系统添加到指定的域中。 |
| leave | 从指定的域中删除系统。 |
| list | 列出系统的所有配置域，或者所有发现和配置的域。 |
| *登录命令* | |
| permit | 启用特定用户或配置域中的所有用户访问本地系统。 |
| deny | 限制特定用户或配置域中的所有用户访问本地系统。 |

**其他资源**

- **realm(8)** 手册页。

# 第 2 章 使用 SAMBA WINBIND 将 RHEL 系统直接连接到 AD

您需要两个组件才能将 RHEL 系统连接到 AD。一个组件（Samba Winbind）与 AD 身份和验证源交互，另一个组件（**realmd**）检测可用的域并配置底层 RHEL 系统服务，即 Samba Winbind 以连接到 AD 域。

这部分论述了使用 Samba Winbind 将 RHEL 系统连接到 Active Directory(AD)。

- 使用 Samba Winbind 直接集成的概述
- 支持直接集成的 Windows 平台
- 将 RHEL 系统添加到 AD 域中
- realm 命令

## 2.1. 使用 SAMBA WINBIND 直接集成的概述

Samba Winbind 在 Linux 系统中模拟 Windows 客户端并与 AD 服务器沟通。

您可以使用 **realmd** 服务配置 Samba Winbind：

- 以标准的方式配置网络身份验证和域成员资格。
- 自动发现有关可访问 domain 和 realm 的信息。
- 不需要高级配置加入domain 或 realm。

请注意：

- 在多林 AD 设置中直接与 Winbind 集成需要双向信托。
- 远程林必须信任本地林，以确保 **idmap_ad** 插件正确处理远程林用户。

Samba 的 **winbindd** 服务为 Name Service Switch(NSS)提供了一个接口，并允许域用户在登录到本地系统时向 AD 进行身份验证。

使用 **winbindd** 的优势是，您可以在不安装其他软件的情况下，增强配置以共享目录和打印机。详情请参阅 Deploying Different of Servers Guide 中的有关使用 Samba 作为服务器的部分。

**其他资源**

- 请参阅 **realmd** man page。
- 请参阅 **winbindd** man page。

## 2.2. 支持直接集成的 WINDOWS 平台

您可以直接将 RHEL 系统与使用以下林和域功能级别的 Active Directory 网站集成：

- 林功能级别范围：Windows Server 2008 - Windows 服务器 2016
- 域功能级别范围：Windows Server 2008 - Windows 服务器 2016

在以下支持的操作系统中测试了直接集成：

- Windows Server 2022（RHEL 9.1 及更高版本）

- Windows Server 2019
- Windows Server 2016
- Windows Server 2012 R2

**注意**

Windows Server 2019 和 Windows Server 2022 没有引入新的功能级。Windows Server 2019 和 Windows Server 2022 使用的最高功能级是 Windows Server 2016。

## 2.3. 将 RHEL 系统添加到 AD 域中

Samba Winbind 是系统安全服务守护进程(SSSD)的替代方案，用于连接带有 Active Directory(AD)的 Red Hat Enterprise Linux(RHEL)系统。您可以使用 **realmd** 将 RHEL 系统加入到 AD 域来配置 Samba Winbind 。

**流程**

1. 如果您的 AD 需要弃用的 RC4 加密类型进行 Kerberos 验证，请在 RHEL 中启用对这些密码的支持：

```
# update-crypto-policies --set DEFAULT:AD-SUPPORT
```

2. 安装以下软件包：

```
# dnf install realmd oddjob-mkhomedir oddjob samba-winbind-clients \
    samba-winbind samba-common-tools samba-winbind-krb5-locator
```

3. 要在域成员中共享目录或打印机，请安装**samba** 软件包：

```
# dnf install samba
```

4. 备份现有的**/etc/samba/smb.conf** Samba 配置文件：

```
# mv /etc/samba/smb.conf /etc/samba/smb.conf.bak
```

5. 加入域。例如，要加入名为**ad.example.com**的域：

```
# realm join --membership-software=samba --client-software=winbind ad.example.com
```

使用上面的命令，**realm**工具会自动：

- 为**ad.example.com**域中的成员创建**/etc/samba/smb.conf**文件
- 将用于用户和组查找的**winbind**模块添加到**/etc/nsswitch.conf**文件中
- 更新**/etc/pam.d/**目录中的可插拔验证模块(PAM)配置文件
- 启动**winbind**服务，并使服务在系统引导时启动

6. 另外，在**/etc/samba/smb.conf**文件中设置备用的 ID 映射后端或自定义 ID 映射设置。

详情请查看了解和配置 Samba ID 映射

1. 编辑 **/etc/krb5.conf** 文件并添加以下部分：

```
[plugins]
  localauth = {
    module = winbind:/usr/lib64/samba/krb5/winbind_krb5_localauth.so
    enable_only = winbind
  }
```

2. 验证**winbind**服务是否运行：

```
# systemctl status winbind
...
  Active: active (running) since Tue 2018-11-06 19:10:40 CET; 15s ago
```

**重要**

要启用 Samba 来查询域用户和组信息，必须在启动**smb**之前运行**winbind**服务。

3. 如果您安装了**samba**软件包来共享目录和打印机，请启用并启动**smb**服务：

```
# systemctl enable --now smb
```

**验证步骤**

1. 显示 AD 用户的详情，如 AD 域中的 AD 管理员帐户：

```
# getent passwd "AD\administrator"
AD\administrator:*:10000:10000::/home/administrator@AD:/bin/bash
```

2. 查询 AD 域中的域用户组成员：

```
# getent group "AD\Domain Users"
   AD\domain users:x:10000:user1,user2
```

3. 另外，还可在设置文件和目录权限时验证您可以使用域用户和组。例如，将**/srv/samba/example.txt**文件的所有者设置为**AD\administrator**，组设置为**AD\Domain Users**：

```
# chown "AD\administrator":"AD\Domain Users" /srv/samba/example.txt
```

4. 验证 Kerberos 验证是否如预期正常工作：

   a. 对于 AD 域成员，为**administrator@AD.EXAMPLE.COM**主体获取一个ticket：

```
# kinit administrator@AD.EXAMPLE.COM
```

   b. 显示缓存的 Kerberos ticket：

```
# klist
Ticket cache: KCM:0
Default principal: administrator@AD.EXAMPLE.COM
```

```
Valid starting       Expires              Service principal
01.11.2018 10:00:00  01.11.2018 20:00:00
krbtgt/AD.EXAMPLE.COM@AD.EXAMPLE.COM
        renew until 08.11.2018 05:00:00
```

5. 显示可用域：

```
# wbinfo --all-domains
BUILTIN
SAMBA-SERVER
AD
```

**其它资源**

- 如果您不想使用弃用的 RC4 密码，可以在 AD 中启用 AES 加密类型。查看
- 使用 GPO 在 Active Directory 中启用 AES 加密类型
- **realm(8)** man page

## 2.4. REALM 命令

**realmd** 系统有两个主要任务：

- 在一个域中管理系统注册。
- 控制允许哪些域用户访问本地系统资源。

在 **realmd** 中，使用命令行工具 **realm** 来运行命令。大多数 **realm** 命令要求用户指定程序要执行的操作，以及要执行操作的域或用户帐户等实体。

**表 2.1. realmd 命令**

| 命令 | 描述 |
|---|---|
| *realm 命令* | |
| discover | 对网络中的域运行发现扫描。 |
| join | 将系统添加到指定的域中。 |
| leave | 从指定的域中删除系统。 |
| list | 列出系统的所有配置域，或者所有发现和配置的域。 |
| *登录命令* | |
| permit | 启用特定用户或配置域中的所有用户访问本地系统。 |
| deny | 限制特定用户或配置域中的所有用户访问本地系统。 |

**其他资源**

**其他资源**

- **realm(8)** 手册页。

# 第 3 章 使用 RHEL 系统角色将 RHEL 系统直接集成到 AD 中

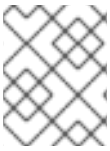
使用 **ad_integration** 系统角色，您可以使用 Red Hat Ansible Automation Platform 自动化 RHEL 系统与活动目录(AD)的直接集成。

## 3.1. AD_INTEGRATION RHEL 系统角色

使用 **ad_integration** 系统角色，您可以将 RHEL 系统直接连接到活动目录(AD)。

该角色使用以下组件：

- SSSD 与中央身份和身份验证源交互
- **realmd** 来检测可用的 AD 域，并配置底层 RHEL 系统服务（在本例中为 SSSD）来连接到所选 AD 域



**注意**

**ad_integration** 角色用于使用没有身份管理(IdM)环境的直接 AD 集成的部署。对于 IdM 环境，请使用 **ansible-freeipa** 角色。

**其他资源**

- **/usr/share/ansible/roles/rhel-system-roles.ad_integration/README.md** 文件
- **/usr/share/doc/rhel-system-roles/ad_integration/** directory
- 使用 SSSD 将 RHEL 系统直接连接到 AD

# 第 4 章 管理到 AD 的直接连接

您可以使用系统安全服务守护进程(SSSD)或 Samba Winbind 将 Red Hat Enterprise Linux(RHEL)系统连接到 Active Directory(AD)。这部分论述了如何在 RHEL 系统已配置为 AD 客户端时修改和管理您的到 AD 的连接。

**先决条件**

- 您已使用 SSSD 或 Samba Winbind 将 RHEL 系统连接到 Active Directory 域。

## 4.1. 修改默认的 KERBEROS 主机 KEYTAB 续订间隔

如果安装了 **adcli** 软件包，SSSD 会在 AD 环境中自动更新 Kerberos 主机 keytab 文件。如果机器帐户密码早于配置的值，守护进程会每天检查并在需要时更新它。

默认续订间隔为 30 天。要更改默认值，请按照以下步骤执行。

**流程**

1. 在 **/etc/sssd/sssd.conf** 文件中的 AD 供应商中添加以下参数：

```
ad_maximum_machine_account_password_age = value_in_days
```

2. 重启 SSSD：

```
# systemctl restart sssd
```

3. 要禁用自动 Kerberos 主机 keytab 续订，设置 **ad_maximum_machine_account_password_age = 0**。

**其他资源**

- **adcli(8)**
- **sssd.conf(5)**
- SSSD 服务失败，并显示错误 'Failed to initialize credentials using keytab [MEMORY:/etc/krb5.keytab]: Preauthentication failed.'

## 4.2. 从 AD 域中删除 RHEL 系统

按照以下流程，从 AD 域中删除直接集成到活动目录(AD)中的 Red Hat Enterprise Linux (RHEL)系统。

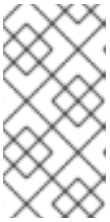
**先决条件**

- 您已使用系统安全服务守护进程(SSSD)或 Samba Winbind 将 RHEL 系统连接到 AD。

**流程**

1. 使用 **realm leave** 命令从身份域中删除系统。该命令从 SSSD 和本地系统中删除域配置。

```
# realm leave ad.example.com
```



**注意**

当客户端离开域时，AD 不会删除帐户，仅删除本地客户端配置。要删除 AD 帐户，请使用 **--remove** 选项运行该命令。最初，会在没有凭证的情况下尝试进行连接，但如果没有有效的 Kerberos 票据，则会提示您输入用户密码。您必须具有从 Active Directory 中删除帐户的权限。

2. 使用带有 **-U** 选项的 **realm leave** 命令指定一个不同的用户从身份域中删除系统。默认情况下，**realm leave** 命令作为默认管理员执行。对于 AD，管理员帐户名为 **Administrator**。如果使用其他用户加入到域中，可能需要以该用户身份执行删除操作。

```
# realm leave [ad.example.com] -U [AD.EXAMPLE.COM\user]'
```

命令首先在没有凭证的情况下尝试连接，但是如果需要，它会提示输入密码。

**验证步骤**

- 验证域不再被配置：

```
# realm discover [ad.example.com]
ad.example.com
   type: kerberos
   realm-name: EXAMPLE.COM
   domain-name: example.com
   configured: no
   server-software: active-directory
   client-software: sssd
   required-package: oddjob
   required-package: oddjob-mkhomedir
   required-package: sssd
   required-package: adcli
   required-package: samba-common-tools
```

**其他资源**

- 请参阅 **realm(8)** 手册页。

## 4.3. 在 SSSD 中设置域解析顺序来解析简短 AD 用户名

默认情况下，您必须指定完全限定的用户名，如 **ad_username@ad.example.com** 和 **group@ad.example.com**，以解决与 SSSD 服务连接到 AD 的 RHEL 主机上的 Active Directory(AD)用户和组。

此流程在 SSSD 配置中设置域解析顺序，以便您可以使用短名称（如 **ad_username** ）解析 AD 用户和组。这个示例配置示例按以下顺序搜索用户和组：

1. Active Directory(AD) 子域 **subdomain2.ad.example.com**
2. AD 子域 **subdomain1.ad.example.com**
3. AD root 域 **ad.example.com**

**先决条件**

- 您已使用 SSSD 服务将 RHEL 主机直接连接到 AD。

**流程**

1. 在文本编辑器中打开 **/etc/sssd/sssd.conf** 文件。
2. 在文件的 **[sssd]** 部分中设置 **domain_resolution_order** 选项。

```
domain_resolution_order = subdomain2.ad.example.com, subdomain1.ad.example.com,
ad.example.com
```

3. 保存并关闭该文件。
4. 重启 SSSD 服务以加载新的配置设置。

```
[root@ad-client ~]# systemctl restart sssd
```

**验证步骤**

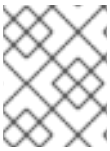
- 验证您只能使用短名称从第一个域检索用户信息。

```
[root@ad-client ~]# id <user_from_subdomain2>
uid=1916901142(user_from_subdomain2) gid=1916900513(domain users)
groups=1916900513(domain users)
```

## 4.4. 为域用户管理登录权限

默认情况下，会应用域端的访问控制，这意味着 Active Directory(AD)用户的登录策略在 AD 域本身中定义。可覆盖此默认行为，以便使用客户端的访问控制。使用客户端侧访问控制时，登录权限仅由本地策略定义。

如果域应用客户端访问控制，您可以使用 **realmd** 为来自该域的用户配置基本的允许或拒绝访问规则。



**注意**

访问规则可以允许或拒绝对系统中所有服务的访问。必须在特定系统资源或域中设置更具体的访问规则。

### 4.4.1. 启用对域中用户的访问

默认情况下，Active Directory(AD)用户的登录策略在 AD 域本身中定义。按照以下流程覆盖此默认行为，并配置 RHEL 主机，以便为 AD 域中的用户启用访问权限。

**重要**

不建议默认允许所有用户访问，只使用 realm 权限 **-x** 指定要被拒绝的用户反之，红帽建议为所有用户维护默认的 no access 策略，且只使用域允许为所选用户授予访问权限。

**先决条件**

- 您的 RHEL 系统是 Active Directory 域的成员。

**流程**

1. 授予对所有用户的访问权限：

```
# realm permit --all
```

2. 授予对特定用户的访问权限：

```
$ realm permit aduser01@example.com
$ realm permit 'AD.EXAMPLE.COM\aduser01'
```

目前，您只能允许访问主域中的用户，而不允许访问可信域中的用户。这是因为用户登录必须包含域名，SSSD 目前无法提供 **realmd** 可用子域的信息。

**验证步骤**

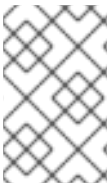
1. 使用 SSH 以 **aduser01@example.com** 用户身份登录到服务器：

```
$ ssh aduser01@example.com@server_name
[aduser01@example.com@server_name ~]$
```

2. 使用 ssh 命令第二次访问同一服务器，此时与 **aduser02@example.com** 用户身份进行以下操作：

```
$ ssh aduser02@example.com@server_name
Authentication failed.
```

请注意 **aduser02@example.com** 用户是如何拒绝对该系统的访问的。您只为 **aduser01@example.com** 用户授予登录系统的权限。由于指定的登录策略，来自该 Active Directory 域的所有其他用户都将被拒绝。



**注意**

如果在 **sssd.conf** 文件中将 **use_fully_qualified_names** 设置为 true，则所有请求都必须使用完全限定域名。但是，如果您将 **use_fully_qualified_names** 设置为 false，则可以在请求中使用完全限定名称，但输出中只会显示简化的版本。

**其他资源**

- 请参阅 **realm(8)** 手册页。

### 4.4.2. 拒绝对域中用户的访问

默认情况下，Active Directory(AD)用户的登录策略在 AD 域本身中定义。按照以下流程覆盖此默认行为，并将 RHEL 主机配置为拒绝对 AD 域中用户的访问。

**重要**

仅允许访问特定用户或组比拒绝访问某些用户或组而允许访问其他所有用户或组要安全。因此，不建议默认允许所有用户访问，只使用 realm 权限 **-x** 指定要被拒绝的用户反之，红帽建议为所有用户维护默认的 no access 策略，且只使用域允许为所选用户授予访问权限。

**先决条件**

- 您的 RHEL 系统是 Active Directory 域的成员。

**流程**

1. 拒绝对域内所有用户的访问：

```
# realm deny --all
```

此命令可防止 **realm** 帐户登录到本地计算机。使用 **realm permit** 来限制登陆到特定账户。

2. 验证域用户的 **login-policy** 被设置为 **deny-any-login** ：

```
[root@replica1 ~]# realm list
example.net
  type: kerberos
  realm-name: EXAMPLE.NET
  domain-name: example.net
  configured: kerberos-member
  server-software: active-directory
  client-software: sssd
  required-package: oddjob
  required-package: oddjob-mkhomedir
  required-package: sssd
  required-package: adcli
  required-package: samba-common-tools
  login-formats: %U@example.net
  login-policy: deny-any-login
```

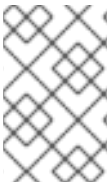
3. 使用 **-x** 选项拒绝对特定用户的访问：

```
$ realm permit -x 'AD.EXAMPLE.COM\aduser02'
```

**验证步骤**

- 使用 SSH 以 **aduser01@example.net** 用户身份登录到服务器。

```
$ ssh aduser01@example.net@server_name
Authentication failed.
```



**注意**

如果在 **sssd.conf** 文件中将 **use_fully_qualified_names** 设置为 true，则所有请求都必须使用完全限定域名。但是，如果您将 **use_fully_qualified_names** 设置为 false，则可以在请求中使用完全限定名称，但输出中只会显示简化的版本。

**其他资源**

- 请参阅 **realm(8)** 手册页。

## 4.5. 在 RHEL 中应用组策略对象访问控制

*组策略对象* (GPO)是存储在 Microsoft Active Directory(AD)中的访问控制设置的集合，适用于 AD 环境中的计算机和用户。通过在 AD 中指定 GPO，管理员可以定义 Windows 客户端和 Red Hat Enterprise Linux(RHEL)主机加入 AD 时的登录策略。

以下小节介绍了如何在您的环境中管理 GPO：

- SSSD 如何解释 GPO 访问控制规则
- SSSD 支持的 GPO 设置列表
- 控制 GPO 强制的 SSSD 选项列表
- 更改 GPO 访问控制模式
- 为 RHEL 主机创建和配置 GPO

### 4.5.1. SSSD 如何解释 GPO 访问控制规则

默认情况下，SSSD 从 Active Directory(AD)域控制器检索组策略对象(GPO)，并评估它们来确定是否允许用户登录到加入 AD 的特定 RHEL 主机。

SSSD 将 AD *Windows Logon Rights* 映射到可插拔验证模块(PAM)服务名称，以在 GNU/Linux 环境中强制实施这些权限。

作为 AD Administrator，您可以通过在 *安全过滤器*中列出它们，将 GPO 规则的范围限制到特定用户、组或主机。

#### 主机过滤的限制

旧版本的 SSSD 不评估 AD GPO 安全过滤器中的主机。

- **RHEL 8.3.0 和更新版本：** SSSD 支持安全过滤器中的用户、组和主机。
- **RHEL 版本早于 8.3.0：** SSSD 忽略主机条目，并且仅支持安全过滤器中的用户和组。
  为确保 SSSD 将基于 GPO 的访问控制应用到特定主机，请在 AD 域中创建一个新的机构单元(OU)，将系统移到新的 OU 中，然后将 GPO 链接到此 OU。

#### 按组过滤的限制

SSSD 目前不支持 Active Directory 的内置组，如带有安全标识符(SID) **S-1-5-32-544** 的 **Administrators**。红帽建议您在 AD GPOs 中针对 RHEL 主机使用 AD 内置组。

#### 其它资源

- 有关 Windows GPO 选项及其对应的 SSSD 选项列表，请参阅 SSSD 支持的 GPO 设置列表 。

### 4.5.2. SSSD 支持的 GPO 设置列表

下表显示了与 Windows 上的 *Group Policy Management Editor* 中指定的 Active Directory GPO 选项对应的 SSSD 选项。

**表 4.1. SSSD 检索的 GPO 访问控制选项**

| GPO 选项 | 对应的 sssd.conf 选项 |
| --- | --- |
| 允许本地登陆<br>拒绝本地登陆 | **ad_gpo_map_interactive** |
| 允许通过 Remote Desktop Services 登陆<br>拒绝通过 Remote Desktop Services 登陆 | **ad_gpo_map_remote_interactive** |
| 从网络访问此计算机<br>拒绝从网络访问这个计算机 | **ad_gpo_map_network** |
| 允许作为批处理作业登陆<br>拒绝作为批处理作业登录 | **ad_gpo_map_batch** |
| 允许作为服务登陆<br>拒绝作为服务登录 | **ad_gpo_map_service** |

**其他资源**

- 有关这些 **sssd.conf** 设置的更多信息，如映射到 GPO 选项的可插拔验证模块(PAM)服务，请参阅 **sssd-ad (5)** 手册页条目。

### 4.5.3. 控制 GPO 强制的 SSSD 选项列表

您可以设置以下 SSSD 选项来限制 GPO 规则的范围。

**ad_gpo_access_control 选项**

您可以在 **/etc/sssd/sssd.conf** 文件中设置 **ad_gpo_access_control** 选项来选择基于 GPO 访问控制的三种不同模式。

**表 4.2. ad_gpo_access_control 值表**

| ad_gpo_access_control 的值 | 行为 |
| --- | --- |
| **enforcing** | 基于 GPO 的访问控制规则会被评估并强制执行。<br>**这是 RHEL 8 中的默认设置。** |
| **permissive** | 基于 GPO 的访问控制规则会被评估 但**不会**强制；每次访问时都会记录一个 **syslog** 信息。这是 RHEL 7 中的默认设置。<br>这个模式是测试策略调整的理想模式,同时允许用户继续登录。 |
| **disabled** | 基于 GPO 的访问控制规则不被评估也不被强制执行。 |

**ad_gpo_implicit_deny 选项**

**ad_gpo_implicit_deny** 选项默认设置为 **False**。在这个默认状态下，如果没有找到适用的 GPOs，则允许用户进行访问。如果将这个选项设置为 **True**，则必须使用 GPO 规则明确允许用户访问。

您可以使用此功能来强化安全性，但小心不要意外拒绝访问。红帽建议在 **ad_gpo_access_control** 设置为 **permissive** 时测试此功能。

以下两个表演示了，根据在 AD 服务器端定义的允许和拒绝登陆权限和 **ad_gpo_implicit_deny** 的值，一个用户被允许或拒绝访问。

**表 4.3. ad_gpo_implicit_deny 设置为 False （默认） 的登录行为**

| 允许规则 | 拒绝规则 | 结果 |
| --- | --- | --- |
| 缺少 | 缺少 | 允许所有用户 |
| 缺少 | 存在 | 仅允许没有拒绝规则的用户 |
| 存在 | 缺少 | 只允许有允许规则的用户 |
| 存在 | 存在 | 只允许有允许规则而不在拒绝规则中的用户 |

**表 4.4. ad_gpo_implicit_deny 设置为 True 的登录行为**

| 允许规则 | 拒绝规则 | 结果 |
| --- | --- | --- |
| 缺少 | 缺少 | 没有用户被允许 |
| 缺少 | 存在 | 没有用户被允许 |
| 存在 | 缺少 | 只允许有允许规则的用户 |
| 存在 | 存在 | 只允许有允许规则而不在拒绝规则中的用户 |

**其它资源**

- 有关在 SSSD 中更改 GPO 强制模式的步骤，请参阅更改 GPO 访问控制模式。
- 有关各种操作的 GPO 模式的详情，请参阅 **sssd-ad(5)** 手册页中的 **ad_gpo_access_control** 条目。

### 4.5.4. 更改 GPO 访问控制模式

此流程更改了在加入到 Active Directory(AD)环境的 RHEL 主机上如何评估和强制实施基于 GPO 的访问控制规则。

在本例中，您要将 GPO 操作模式从 **enforcing** （默认）改为 **permissive**，用于测试。

**重要**

如果您看到以下错误，则 Active Directory 用户因为基于 GPO 的访问控制而无法登录：

- 在 **/var/log/secure** 中：

```
Oct 31 03:00:13 client1 sshd[124914]: pam_sss(sshd:account): Access denied for user aduser1: 6 (Permission denied)
Oct 31 03:00:13 client1 sshd[124914]: Failed password for aduser1 from 127.0.0.1 port 60509 ssh2
Oct 31 03:00:13 client1 sshd[124914]: fatal: Access denied for user aduser1 by PAM account configuration [preauth]
```

- 在 **/var/log/sssd/sssd__example.com_.log** 中：

```
(Sat Oct 31 03:00:13 2020) [sssd[be[example.com]]] [ad_gpo_perform_hbac_processing] (0x0040): GPO access check failed: [1432158236](Host Access Denied)
(Sat Oct 31 03:00:13 2020) [sssd[be[example.com]]] [ad_gpo_cse_done] (0x0040): HBAC processing failed: [1432158236](Host Access Denied}
(Sat Oct 31 03:00:13 2020) [sssd[be[example.com]]] [ad_gpo_access_done] (0x0040): GPO-based access control failed.
```

如果这是非预计的行为，您可以临时将 **ad_gpo_access_control** 设置为 **permissive** 来调试 AD 中的适当的 GPO 设置。

**先决条件**

- 已使用 SSSD 将 RHEL 主机加入到 AD 环境中。
- 编辑 **/etc/sssd/sssd.conf** 配置文件需要 **root** 权限。

**流程**

1. 停止 SSSD 服务。

```
[root@server ~]# systemctl stop sssd
```

2. 在文本编辑器中打开 **/etc/sssd/sssd.conf** 文件。
3. 在 AD 域的 **domain** 部分中，将 **ad_gpo_access_control** 设置为 **permissive**。

```
[domain/example.com]
ad_gpo_access_control=permissive
...
```

4. 保存 **/etc/sssd/sssd.conf** 文件。
5. 重启 SSSD 服务以加载配置更改。

```
[root@server ~]# systemctl restart sssd
```

**其它资源**

- 有关不同 GPO 访问控制模式列表，请参阅 SSSD 选项列表来控制 GPO 强制 。

### 4.5.5. 在 AD GUI 中为 RHEL 主机创建和配置 GPO

组策略对象 (GPO)是存储在 Microsoft Active Directory(AD)中的访问控制设置的集合，适用于 AD 环境中的计算机和用户。以下流程在 AD 图形用户界面(GUI)中创建 GPO，以控制对直接集成到 AD 域的 RHEL 主机的登录访问。

**先决条件**

- 已使用 SSSD 将 RHEL 主机加入到 AD 环境中。
- 您有 AD Administrator 特权才能使用 GUI 更改 AD。

**流程**

1. 在 **Active Directory Users 和 Computers** 中，创建一个机构单元(OU)来与新的 GPO 关联：
   a. 右键点击域。
   b. 选择 **New**。
   c. 选择 **Organizational Unit**。
2. 点代表 RHEL 主机的 Computer Object 名称（加入 Active Directory 时创建），并将其拖到新的 OU 中。通过在自己的 OU 中带有 RHEL 主机，GPO 以这个主机为目标。
3. 在 **Group Policy Management Editor** 中，为您创建的 OU 创建新的 GPO：
   a. 展开 **Forest**。
   b. 展开 **Domains**。
   c. 展开您的域。
   d. 右键点击新的 OU。
   e. 选择 **Create a GPO in this domain**。
4. 为新的 GPO 指定名称，如 **Allow SSH access** 或 **Allow Console/GUI access** 并点 **OK**。
5. 编辑新的 GPO：
   a. 在 **Group Policy Management** 编辑器中选择 OU。
   b. 点鼠标右键，选择 **Edit**。
   c. 选择 **User Rights Assignment**。
   d. 选择 **Computer Configuration**
   e. 选择 **Policies**。
   f. 选择 **Windows Settings**。
   g. 选择 **Security Settings**。

h. 选择 **Local Policies**。

i. 选择 **User Rights Assignment**。

6. 分配登录权限：

a. 双击 **Allow log on locally** 以授予本地控制台/GUI 访问权限。

b. 双击 **Allow log on remote Desktop Services** 以授予 SSH 访问权限。

7. 将您要访问这些策略的用户添加到策略本身：

a. 点 **Add User or Group**。

b. 在空白字段中输入用户名。

c. 点击 **确定**。

**其他资源**

- 如需有关组策略对象的更多信息，请参阅 Microsoft 文档中的组策略对象。

### 4.5.6. 其它资源

- 有关将 RHEL 主机加入活动目录环境的更多信息，请参阅 使用 SSSD 将 RHEL 系统直接连接到 AD。

# 第 5 章 使用受管服务帐户访问 AD

Active Directory(AD)受管服务帐户(MSA)允许您在 AD 中创建与特定计算机对应的帐户。您可以使用 MSA 作为特定用户主体连接到 AD 资源，而无需将 RHEL 主机加入到 AD 域中。

本节讨论以下主题：

- 使用受管服务帐户的好处
- 为 RHEL 主机配置受管服务帐户
- 更新受管服务帐户的密码
- 受管服务帐户规格
- adcli create-msa 命令的选项

## 5.1. 使用受管服务帐户的好处

如果要允许 RHEL 主机在不加入的情况下访问 Active Directory(AD)域，您可以使用 Managed Service Account(MSA)访问该域。MSA 是 AD 中的一个帐户，对应于特定的计算机，您可将其用来连接到 AD 资源作为特定用户主体。

例如，如果 AD 域 **production.example.com** 与 **lab.example.com** AD 域有一个单向信任关系，则应用以下条件：

- **lab** 域信任来自 **production** 域的用户和主机。
- **production** 域**不信任来自 lab** 域中的用户和主机。

这意味着主机加入了 **lab** 域（如 **client.lab.example.com**）将无法通过信任访问来自 **production** 域的资源。

如果要为 **client.lab.example.com** 主机创建一个例外，可以使用 **adcli** 程序在 **production.example.com** 域中为 **client** 主机创建一个 MSA。要使用 MSA 的 Kerberos 主体进行身份验证，您可以从 **client** 主机在 **production** 域中执行安全 LDAP 搜索。

## 5.2. 为 RHEL 主机配置受管服务帐户

这个过程从 **lab.example.com** Active Directory(AD)域为主机创建受管服务帐户(MSA)，并配置 SSSD，以便您可以访问 **production.example.com** AD 域并进行身份验证。

**注意**

如果您需要从 RHEL 主机访问 AD 资源，红帽建议您使用 **realm** 命令将 RHEL 主机加入到 AD 域中。请参阅使用 SSSD 将 RHEL 系统直接连接到 AD 。

只有在满足以下条件之一时才执行这个步骤：

- 您不能将 RHEL 主机加入到 AD 域中，您希望在 AD 中为该主机创建帐户。
- 您已将 RHEL 主机加入到 AD 域，且您需要访问您加入的域的主机凭证无效，例如：有一个单向信任。

**先决条件**

- 确保 RHEL 主机上的以下端口已为 AD 域控制器打开并可以被访问。

| 服务 | 端口 | 协议 |
|---|---|---|
| DNS | 53 | TCP, UDP |
| LDAP | 389 | TCP, UDP |
| LDAPS（可选） | 636 | TCP, UDP |
| Kerberos | 88 | TCP, UDP |

- 您有在 **production.example.com** 域中创建 MSA 的 AD Administrator 的密码。
- 您有运行 **adcli** 命令以及修改 **/etc/sssd/sssd.conf** 配置文件所需的 root 权限。
- *（可选）*您已安装了 **krb5-workstation** 软件包，其中包括 **klist** 诊断实用程序。

**流程**

1. 在 **production.example.com** AD 域中为主机创建 MSA。

```
[root@client ~]# adcli create-msa --domain=production.example.com
```

2. 显示创建的 Kerberos keytab 中 MSA 的信息。记录 MSA 名称：

```
[root@client ~]# klist -k /etc/krb5.keytab.production.example.com
Keytab name: FILE:/etc/krb5.keytab.production.example.com
KVNO Principal
---- --------------------------------------------------------------
   2 CLIENT!S3A$@PRODUCTION.EXAMPLE.COM (aes256-cts-hmac-sha1-96)
   2 CLIENT!S3A$@PRODUCTION.EXAMPLE.COM (aes128-cts-hmac-sha1-96)
```

3. 打开 **/etc/sssd/sssd.conf** 文件并选择要添加的适当 SSSD 域配置：

   - 如果 MSA 对应于一个**来自不同林的 AD 域**，请创建一个名为 **[domain/<name_of_domain>]** 的新域部分，然后输入有关 MSA 和 keytab 的信息。最重要的选项为 **ldap_sasl_authid**,**ldap_krb5_keytab**, 和 **krb5_keytab**:

```
[domain/production.example.com]
ldap_sasl_authid = CLIENT!S3A$@PRODUCTION.EXAMPLE.COM
ldap_krb5_keytab = /etc/krb5.keytab.production.example.com
krb5_keytab = /etc/krb5.keytab.production.example.com
ad_domain = production.example.com
krb5_realm = PRODUCTION.EXAMPLE.COM
access_provider = ad
...
```

   - 如果 MSA 对应于一个**来自本地林的 AD 域**，请使用 **[domain/root.example.com/sub-domain.example.com]** 格式创建一个新的子域部分，然后输入有关 MSA 和 keytab 的信息。最重要的选项为 **ldap_sasl_authid**,**ldap_krb5_keytab**, 和 **krb5_keytab**:

```
[domain/ad.example.com/production.example.com]
```

```
ldap_sasl_authid = CLIENT!S3A$@PRODUCTION.EXAMPLE.COM
ldap_krb5_keytab = /etc/krb5.keytab.production.example.com
krb5_keytab = /etc/krb5.keytab.production.example.com
ad_domain = production.example.com
krb5_realm = PRODUCTION.EXAMPLE.COM
access_provider = ad
...
```

**验证步骤**

- 验证您可以检索一个 Kerberos ticket-granting ticket(TGT)作为 MSA：

```
[root@client ~]# kinit -k -t /etc/krb5.keytab.production.example.com 'CLIENT!S3A$'
[root@client ~]# klist
Ticket cache: KCM:0:54655
Default principal: CLIENT!S3A$@PRODUCTION.EXAMPLE.COM

Valid starting       Expires              Service principal
11/22/2021 15:48:03  11/23/2021 15:48:03
krbtgt/PRODUCTION.EXAMPLE.COM@PRODUCTION.EXAMPLE.COM
```

- 在 AD 中，验证您在 Managed Service Accounts Organizational units (OU) 中的具有主机的 MSA。

**其他资源**

- 使用 SSSD 将 RHEL 系统直接连接到 AD

## 5.3. 更新受管服务帐户的密码

管理的服务帐户(MSA)具有由 Active Directory(AD)自动维护的复杂密码。默认情况下，System Services Security Daemon(SSSD)会在 Kerberos keytab 早于 30 天时自动更新 MSA 密码，这会使用 AD 中的密码保持最新状态。此流程解释了如何手动更新 MSA 的密码。

**先决条件**

- 您之前为 production.example.com AD 域中的主机创建了 MSA。
- *（可选）* 您已安装了 **krb5-workstation** 软件包，其中包括 **klist** 诊断实用程序。

**流程**

1. *（可选）* 在 Kerberos keytab 中显示 MSA 的当前密钥版本号(KVNO)。当前 KVNO 是 2.

```
[root@client ~]# klist -k /etc/krb5.keytab.production.example.com
Keytab name: FILE:/etc/krb5.keytab.production.example.com
KVNO Principal
---- --------------------------------------------------------------------
   2 CLIENT!S3A$@PRODUCTION.EXAMPLE.COM (aes256-cts-hmac-sha1-96)
   2 CLIENT!S3A$@PRODUCTION.EXAMPLE.COM (aes128-cts-hmac-sha1-96)
```

2. 更新 **production.example.com** AD 域中 MSA 的密码。

```
[root@client ~]# adcli update --domain=production.example.com --host-
keytab=/etc/krb5.keytab.production.example.com --computer-password-lifetime=0
```

**验证步骤**

- 验证在 Kerberos keytab 中已递增了 KVNO：

```
[root@client ~]# klist -k /etc/krb5.keytab.production.example.com
Keytab name: FILE:/etc/krb5.keytab.production.example.com
KVNO Principal
---- --------------------------------------------------------------
   3 CLIENT!S3A$@PRODUCTION.EXAMPLE.COM (aes256-cts-hmac-sha1-96)
   3 CLIENT!S3A$@PRODUCTION.EXAMPLE.COM (aes128-cts-hmac-sha1-96)
```

## 5.4. 受管服务帐户规格

**adcli** 实用程序创建的受管服务帐户(MSA)具有以下规格：

- 它们不能有额外的服务主体名称(SPN)。
- 默认情况下，MSA 的 Kerberos 主体存储在名为 ***<default_keytab_location>.<Active_Directory_domain>*** 的 Kerberos keytab 中，如 ***/etc/krb5.keytab.production.example.com***。
- MSA 名称限制为 20 个字符或更少。最后 4 个字符是一个由 来自数字和大写 ASCII 范围中的 3 个随机字符组成的后缀，附加到您提供的短主机名中，使用 **!** 字符作为分隔符。例如，一个带有短名称 **myhost** 的主机会有一个带有以下规则的 MSA：

| 规格 | 值 |
|---|---|
| 通用名称(CN)属性 | **myhost!A2c** |
| NetBIOS 名称 | **myhost!A2c$** |
| sAMAccountName | **myhost!A2c$** |
| **production.example.com** AD 域中的 Kerberos 主体 | **myhost!A2c$@PRODUCTION.EXAMPLE.COM** |

## 5.5. ADCLI CREATE-MSA 命令的选项

除了您可以传递给 **adcli** 实用程序的全局选项外，您还可以指定以下选项以专门控制它如何处理受管服务帐户(MSA)。

**-N, --computer-name**

将在 Active Directory(AD)域中创建的 MSA 短名称。如果您没有指定名称，则为 **--host-fqdn** 的第一部分，或者将其默认与随机后缀一起使用。

**-O, --domain-ou=OU=*<path_to_OU>***

要创建 MSA 的机构单元(OU)的完整可辨识名称。如果没有指定这个值，则在默认位置 **OU=CN=Managed Service Accounts,DC=EXAMPLE,DC=COM** 中创建 MSA。

**-H, --host-fqdn=host**

覆盖本地机器的完全限定域名。如果没有指定这个选项，则使用本地机器的主机名。

**-K, --host-keytab=*<path_to_keytab>***

用于存储 MSA 凭证的主机 keytab 的路径。如果没有指定这个值，则默认位置 **/etc/krb5.keytab** 与小写的 Active Directory 域名一起使用，如 **/etc/krb5.keytab.domain.example.com**。

**--use-ldaps**

通过安全 LDAP(LDAPS)频道创建 MSA。

**--verbose**

创建 MSA 时打印详细的信息。

**--show-details**

创建 MSA 后打印出有关 MSA 的信息。

**--show-password**

创建 MSA 后打印 MSA 密码。