



Red Hat Enterprise Linux 8

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

Red Hat Enterprise Linux 8 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

Legal Notice

Copyright © 2025 Red Hat, Inc.

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

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

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

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

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

Abstract

您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。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。

Table of Contents

对红帽文档提供反馈	3
第 1 章 使用 SSSD 将 RHEL 系统直接连接到 AD	4
1.1. 使用 SSSD 直接集成概述	4
1.2. 支持直接集成的 WINDOWS 平台	4
1.3. 直接连接到 AD	5
1.4. AD 供应商如何处理动态 DNS 更新	8
1.5. 修改 AD 供应商的动态 DNS 设置	9
1.6. AD 供应商如何处理可信域	10
1.7. 使用 SSSD 覆盖活动目录站点自动发现	10
1.8. REALM 命令	11
1.9. 使用 SSSD 将 RHEL 系统直接集成到 AD 所需的端口	12
第 2 章 使用 SAMBA WINBIND 将 RHEL 系统直接连接到 AD	13
2.1. 使用 SAMBA WINBIND 直接集成的概述	13
2.2. 支持直接集成的 WINDOWS 平台	13
2.3. 将 RHEL 系统添加到 AD 域中	14
2.4. REALM 命令	16
第 3 章 使用 RHEL 系统角色将 RHEL 系统加入到 ACTIVE DIRECTORY	18
3.1. 使用 AD_INTEGRATION RHEL 系统角色将 RHEL 加入到 ACTIVE DIRECTORY 域中	18
第 4 章 管理到 AD 的直接连接	21
4.1. 修改默认的 KERBEROS 主机 KEYTAB 续订间隔	21
4.2. 从 AD 域中删除 RHEL 系统	21
4.3. 在 SSSD 中设置域解析顺序，以解析简短的 AD 用户名	22
4.4. 为域用户管理登录权限	23
4.5. 在 RHEL 中应用组策略对象访问控制	25
第 5 章 使用受管服务帐户访问 AD	32
5.1. 受管服务帐户的好处	32
5.2. 为 RHEL 主机配置受管服务帐户	32
5.3. 更新受管服务帐户的密码	34
5.4. 受管服务帐户规格	35
5.5. ADCLI CREATE-MSA 命令的选项	35

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. 单击顶部导航栏中的 **Create**。
3. 在 **Summary** 字段中输入描述性标题。
4. 在 **Description** 字段中输入您的建议以改进。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

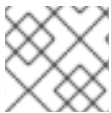
要将 RHEL 系统连接到 Active Directory (AD)，请使用：

- 用于身份和身份验证的系统安全服务守护进程(SSSD)
- **realmd** 来检测可用的域并配置底层 RHEL 系统服务。

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 高度可配置；提供可插拔验证模块(PAM)和名称交换服务(NSS)集成，以及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的数据库。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 集成只能在单个 AD 林中正常工作。

将 SSSD 配置为直接将 Linux 系统与 AD 集成的最简便方法是使用 **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域和域的信息，不需要高级配置加入域或域。

您可以使用 SSSD 与 AD 进行直接和间接集成，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 系统份额的增长，您的部署通常需要更好地集中管理身份相关的策略，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（具有信任 AD 的 IdM）的更多信息，请参阅 [将 RHEL 客户端从 AD 域移到 IdM 服务器](#)。

其它资源

- 您系统上的 **realm (8)**、**sssd-ad (5)** 和 **sssd (8)** 手册页
- [在间接集成和直接集成间进行选择](#)

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

在以下支持的操作系统中测试了直接集成：

- Windows Server 2022 (RHEL 8.7 或更高版本)

- 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别。Windows Server 2019 和 Windows Server 2022 使用的最高功能级别是 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)系统与活动目录(AD)连接。您可以使用 POSIX ID 映射(SSSD 的默认设置)或者使用 AD 中定义的 POSIX 属性直接与 AD 集成。



重要

在将您的系统加入 AD 前，请按照红帽知识库解决方案 [Basic Prechecks Steps](#) 中的步骤正确配置您的系统：RHEL 使用 'adcli'、'realm' 和 'net' 命令与活动目录结合。

1.3.1. 用于与 AD 集成的选项：使用 POSIX 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 [管理用户和组群帐户简介](#)。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 POSIX ID 映射或 AD 中的 POSIX 属性与 AD 集成的选项。默认为使用 POSIX ID 映射。

1.3.2. 使用 POSIX ID 映射连接到 AD

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

- 当 SSSD 检测到新的 AD 域时，它会为新域分配一系列可用 ID。
- 当 AD 用户第一次登录 SSSD 客户端机器时，SSSD 在 SSSD 缓存中为用户创建一个条目，包括基于用户的 SID 以及该域的 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，所以在登录到任何 RHEL 系统时，用户具有相同的 UID 和 GID。



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。

如需更多信息，请参阅 **sssd-ad** 手册页中 ID 映射的部分。

1.3.2.1. 使用 SSSD 发现并加入 AD 域

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

先决条件

- 确保打开所需的端口：
 - [使用 SSSD 将 RHEL 系统直接集成到 AD 所需的端口](#)
- 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：

```
# yum 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的名称：

```
# 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 查找来自动查找此域中的域控制器。



注意

realm 系统可以发现 Active Directory 和 Identity Management 域。如果您的环境中两个域都存在，您可以使用 **--server-software=active-directory** 选项将发现结果限制为特定的服务器类型。

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

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

验证

- 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
```

其它资源

- 您系统上的 **realm (8)** 和 **nmcli (1)** 手册页

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

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

使用 POSIX 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，您必须在 SSSD 中禁用 POSIX ID 映射。

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

- 确保打开所需的端口：
 - 使用 [SSSD 将 RHEL 系统直接集成到 AD 所需的端口](#)
- 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

- 安装以下软件包：

```
# yum install realm oddjob oddjob-mkhomedir sssd adcli krb5-workstation
```

- 使用 **realm join** 命令和 **-- automatic-id-mapping=no** 选项，配置本地 RHEL 系统并禁用 POSIX ID 映射。**realm** 套件自动编辑所有必要的配置文件。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，您可以在 SSSD 中手动禁用 POSIX ID 映射：

- a. 打开 `/etc/sss/sss.conf` 文件：
- b. 在 AD 域部分中，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

您必须为 AD 中的用户配置了相关的 POSIX 属性（`uidNumber`、`gidNumber`、`unixHomeDirectory` 和 `loginShell`）。

验证

- 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
```

其它资源

- 系统中的 **sss-ldap (8)** 手册页

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)Managed Service Account(MSA)从不同的林访问 AD 域，其中它们之间没有信任。

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/sss/sss.conf` 配置文件中的 `dyndns_refresh_interval` 选项指定的间隔内。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 租期续订后更新 DNS 记录。

SSSD 使用 Kerberos/GSSAPI 为 DNS(GSS-TSIG)使用 Kerberos/GSSAPI 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。

其它资源

- 系统中 **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/sss/sss.conf** 配置文件。

流程

1. 在文本编辑器中打开 **/etc/sss/sss.conf** 配置文件。
2. 将以下选项添加到 AD 域的 **[domain]** 部分，将 DNS 记录刷新闻隔设置为 12 小时，禁用更新 PTR 记录，并将 DNS 记录时间到 Live(TTL)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/sss/sss.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
```

其它资源

- [AD 供应商如何处理动态 DNS 更新](#)
- 系统中 **sssd-ad (5)** 手册页

1.6. AD 供应商如何处理可信域

如果您在 **/etc/sss/sss.conf** 配置文件中设置了 **id_provider = ad** 选项，则 SSSD 会将可信域按如下处理：

- SSSD 只支持单个 AD 林中的域。如果 SSSD 需要从多个地区访问多个域，请考虑使用带有信任（首选）的 IPA 或 **winbindd** 服务而不是 SSSD。
- 默认情况下，SSSD 会发现林中的所有域，如果可信域中的对象请求到达，SSSD 会尝试解析它。
如果可信域无法访问或在地理位置上造成速度较慢，您可以在 **/etc/sss/sss.conf** 中设置 **ad_enabled_domains** 参数，以限制可信域 SSSD 从哪个位置解析对象。
- 默认情况下，您必须使用完全限定用户名从可信域解析用户。

其它资源

- 您系统上的 **sss.conf (5)** 手册页

1.7. 使用 SSSD 覆盖活动目录站点自动发现

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。这能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 **autodiscovery** 查找其 AD 站点并连接到最近的域控制器。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，避免从 unreachable DCs 执行超时。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 autodiscovery 过程，请通过将 **ad_site** 选项添加到 **/etc/sss/sss.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，以便您可以编辑 **/etc/sss/sss.conf** 配置文件。

流程

1. 在文本编辑器中打开 **/etc/sss/sss.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/sss/sss.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 系统有两个主要的任务领域：

- 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** 命令要求用户指定实用程序应执行的操作，以及要为其执行操作的实体（如域或用户帐户）。

表 1.1. realmd 命令

命令	描述
<i>Realm 命令</i>	
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。

命令	描述
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。

其它资源

- 您系统上的 **realm (8)** 手册页

1.9. 使用 SSSD 将 RHEL 系统直接集成到 AD 所需的端口

以下端口必须打开，并可以被 AD 域控制器和 RHEL 主机访问。

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	
LDAPS	636	TCP	可选
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 id_provider = ad 选项
LDAPS 全局目录	3269	TCP	可选
NTP	123	UDP	可选
NTP	323	UDP	可选

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

要将 RHEL 系统连接到 Active Directory (AD)，请使用：

- Samba Winbind 与 AD 身份和验证源交互
- **realmd** 来检测可用的域并配置底层 RHEL 系统服务。

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。

您可以通过以下方法使用 **realmd** 服务来配置 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 集成需要双向信任。
- 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** 服务为名称服务交换机(NSS)提供接口，并让域用户在登录本地系统时对 AD 进行身份验证。

使用 **winbindd** 的优势在于，您可以增强共享目录和打印机的配置，而无需安装其他软件。

其它资源

- [使用 Samba 作为服务器](#)
- 系统上的 **realmd** 手册页
- 系统上的 **winbindd** man page

2.2. 支持直接集成的 WINDOWS 平台

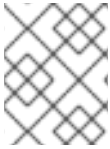
您可以直接将 RHEL 系统与使用以下林和域功能级别的 Active Directory 网站集成：

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

在以下支持的操作系统中测试了直接集成：

- Windows Server 2022 (RHEL 8.7 或更高版本)
- 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别。Windows Server 2019 和 Windows Server 2022 使用的最高功能级别是 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。

先决条件

- 主机使用 DNS 服务器来解析 AD 域。
- 主机上的时间与 AD 中的时间同步，时区设置是正确的。

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：

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

3. 要在域成员中共享目录或打印机，请安装 **samba** 软件包：

```
# yum 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：

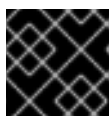
- 为 **ad.example.com** 域中的成员创建 **/etc/samba/smb.conf** 文件
- 将用于用户和组查找的 **winbind** 模块添加到 **/etc/nsswitch.conf** 文件中
- 更新 **/etc/pam.d/** 目录中的可插拔验证模块(PAM)配置文件
- 启动 **winbind** 服务，并使服务在系统引导时启动

6. 可选：在 **/etc/samba/smb.conf** 文件中设置替代的 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](#)
7. 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
  }
```

8. 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。

9. 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

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. 可选：验证当您对文件和目录设置权限时，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](#)
- 您系统上的 **realm (8)** 手册页

2.4. REALM 命令

realmd 系统有两个主要的任务领域：

- 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** 命令要求用户指定实用程序应执行的操作，以及要为其执行操作的实体（如域或用户帐户）。

表 2.1. realmd 命令

命令	描述
<i>Realm 命令</i>	
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。
<i>登录命令</i>	
permit	启用特定用户或配置域中的所有用户访问本地系统。

命令	描述
deny	限制特定用户或配置域中的所有用户访问本地系统。

其它资源

- 您系统上的 **realm (8)** 手册页

第 3 章 使用 RHEL 系统角色将 RHEL 系统加入到 ACTIVE DIRECTORY

如果您的组织使用 Microsoft Active Directory (AD) 集中管理用户、组和其他资源，您可以将 Red Hat Enterprise Linux (RHEL) 主机加入到这个 AD 中。例如，AD 用户可以登录 RHEL，您可以在 RHEL 主机上为经过身份验证的 AD 用户提供服务。通过使用 **ad_integration** RHEL 系统角色，您可以将 Red Hat Enterprise Linux 系统集成到活动目录(AD)域中。



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。

3.1. 使用 AD_INTEGRATION RHEL 系统角色将 RHEL 加入到 ACTIVE DIRECTORY 域中

您可以使用 **ad_integration** RHEL 系统角色来自动化将 RHEL 加入到活动目录(AD)域的过程。

先决条件

- 您已准备好控制节点和受管节点
- 以可在受管主机上运行 playbook 的用户登录到控制节点。
- 用于连接到受管节点的帐户具有 **sudo** 权限。
- 受管节点使用 DNS 服务器来解析 AD DNS 条目。
- AD 帐户的凭证，该帐户有权将计算机加入到域中。
- 确保打开所需的端口：
 - 使用 **SSSD** 将 RHEL 系统直接集成到 AD 所需的端口

流程

1. 将您的敏感变量存储在一个加密文件中：

- a. 创建 vault：

```
$ ansible-vault create ~/vault.yml
New Vault password: <vault_password>
Confirm New Vault password: <vault_password>
```

- b. 在 **ansible-vault create** 命令打开编辑器后，以 **<key>: <value>** 格式输入敏感数据：

```
usr: administrator
pwd: <password>
```

- c. 保存更改，并关闭编辑器。Ansible 加密 vault 中的数据。

2. 创建一个包含以下内容的 playbook 文件，如 **~/playbook.yml**：

```

---
- name: Active Directory integration
  hosts: managed-node-01.example.com
  vars_files:
    - ~/vault.yml
  tasks:
    - name: Join an Active Directory
      ansible.builtin.include_role:
        name: redhat.rhel_system_roles.ad_integration
      vars:
        ad_integration_user: "{{ usr }}"
        ad_integration_password: "{{ pwd }}"
        ad_integration_realm: "ad.example.com"
        ad_integration_allow_rc4_crypto: false
        ad_integration_timesync_source: "time_server.ad.example.com"

```

示例 playbook 中指定的设置包括以下内容：

ad_integration_allow_rc4_crypto: <true/false>

配置角色是否激活受管节点上的 **AD-SUPPORT** 加密策略。默认情况下，RHEL 不支持弱 RC4 加密，但如果 AD 中的 Kerberos 仍然需要 RC4，您可以通过设置 **ad_integration_allow_rc4_crypto: true** 来启用这个加密类型。如果 Kerberos 使用 AES 加密，则省略此变量或将其设置为 **false**。

ad_integration_timesync_source: <time_server>

指定用于时间同步的 NTP 服务器。Kerberos 需要 AD 域控制器和域成员之间同步时间，以防止重播攻击。如果省略此变量，**ad_integration** 角色不会利用 **timesync** RHEL 系统角色在受管节点上配置时间同步。

有关 playbook 中使用的所有变量的详情，请查看控制节点上的 **/usr/share/ansible/roles/rhel-system-roles.ad_integration/README.md** 文件。

3. 验证 playbook 语法：

```
$ ansible-playbook --ask-vault-pass --syntax-check ~/playbook.yml
```

请注意，这个命令只验证语法，不会防止错误但有效的配置。

4. 运行 playbook：

```
$ ansible-playbook --ask-vault-pass ~/playbook.yml
```

验证

- 检查 AD 用户（如 **管理员**）是否在受管节点本地可用：

```

$ ansible managed-node-01.example.com -m command -a 'getent passwd
administrator@ad.example.com'
administrator@ad.example.com:*:1450400500:1450400513:Administrator:/home/administrator
@ad.example.com:/bin/bash

```

其它资源

- `/usr/share/ansible/roles/rhel-system-roles.ad_integration/README.md` 文件
- `/usr/share/doc/rhel-system-roles/ad_integration/` directory
- [Ansible vault](#)

第 4 章 管理到 AD 的直接连接

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)域后，您可以管理 Kerberos 续订、域成员资格、用户访问权限和组策略对象(GPO)等密钥设置。

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 环境中自动续订 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/sss/sssd.conf** 文件中向 AD 供应商添加以下参数：

```
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，请设置 **ad_maximum_machine_account_password_age = 0**。

其它资源

- **adcli(8)**
- **sss.conf(5)**
- [SSSD 服务失败并显示 'Failed to initialize credentials using keytab \[MEMORY:/etc/krb5.keytab\]:preauthentication failed.'](#)(Red Hat Knowledgebase)

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 left** 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. 将 **-U** 选项与 **realm left** 命令搭配使用，以指定其他用户从身份域中删除系统。默认情况下，**realm left** 命令以默认管理员身份执行。对于 AD，管理员帐户名为 **Administrator**。如果使用其他用户加入域，则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]
```

命令首先尝试在没有凭据的情况下进行连接，但是如果需要，它会提示输入密码。

验证

- 验证不再配置域：

```
# 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
```

其它资源

- 您系统上的 **realm (8)** 手册页

4.3. 在 SSSD 中设置域解析顺序，以解析简短的 AD 用户名

默认情况下，您必须指定完全限定用户名（如 **ad_username@ad.example.com** 和 **group@ad.example.com**）来解析使用 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 **子域1.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/sss/sss.conf` 文件。
2. 在文件的 `[sss]` 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

- 验证您只能使用简短名称从第一个域中检索用户的用户信息。

```
[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。此默认行为可以被覆盖，以便使用客户端访问控制。使用客户端访问控制时，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。您可以覆盖此默认行为，并配置 RHEL 主机来为 AD 域中的用户启用访问权限。



重要

不建议默认允许访问 all，而只拒绝对具有域允许 **-x** 的特定用户。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 当前无法提供有关可用子域的**信息**。

验证

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** 用户如何拒绝对该系统的访问。您只为 **aduser01@example.com** 用户授权可以登录到系统。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

如果您在 **sssd.conf** 文件中将 **use_fully_qualified_names** 设置为 **true**，则所有请求必须使用完全限定域名。但是，如果您将 **use_fully_qualified_names** 设为 **false**，则可以在请求中使用完全限定的名称，但输出中仅显示简化的版本。

其它资源

- 您系统上的 **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。您可以覆盖此默认行为，并将 RHEL 主机配置为拒绝对 AD 域中的用户访问。



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，不建议默认允许访问 **all**，而只拒绝对具有域允许 **-x** 的特定用户。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
```

此命令将阻止 **realm** 帐户登录本地计算机。使用 **realm 允许** 将登录限制为特定的帐户。

2. 验证域用户的 **login-policy** 是否已 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
```

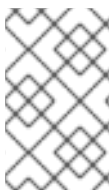
3. 使用 **-x** 选项拒绝对特定用户的访问：

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

验证

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

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



注意

如果您在 **sssd.conf** 文件中将 **use_fully_qualified_names** 设置为 **true**，则所有请求必须使用完全限定域名。但是，如果您将 **use_fully_qualified_names** 设为 **false**，则可以在请求中使用完全限定的名称，但输出中仅显示简化的版本。

其它资源

- 您系统上的 **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)中的访问控制设置集合，可应用于 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)主机加入到 AD 的登录策略。

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)，并对其进行评估，以确定是否允许用户登录加入 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 环境中强制实现这些权限。

作为 AD Administrator，您可以通过 *在安全过滤器* 中列出 GPO 规则的范围，将 GPO 规则的范围限制为特定用户、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 或更高版本**：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 会忽略主机条目，并且只支持安全过滤器中的用户和组。
为确保 SSSD 应用基于 GPO 的访问控制到特定主机，请在 AD 域中创建新的机构单元(OU)，将系统移到新 OU，然后将 GPO 链接到这个 OU。

按组过滤的限制

SSSD 目前不支持 Active Directory 的内置组，如具有安全标识符(SID) **S-1-5-32-544** 的管理员。红帽建议您在 AD GPOs 中以 RHEL 主机为目标使用 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
允许本地登录 拒绝本地日志	ad_gpo_map_interactive
允许通过 Remote Desktop Services 登录 通过 Remote Desktop Services 在上拒绝登录	ad_gpo_map_remote_interactive
从网络访问此计算机 拒绝从网络访问此计算机	ad_gpo_map_network
允许以批处理作业身份登录 拒绝作为批处理作业登录	ad_gpo_map_batch

GPO 选项	对应的 sssd.conf 选项
允许作为服务登录 拒绝作为服务登录	ad_gpo_map_service

其它资源

- 系统中 **sssd-ad (5)** 手册页

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/sss/sss.conf` 文件中设置 **ad_gpo_access_control** 选项，以选择基于 GPO 的访问控制在三种不同模式中运行。

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。 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 强制实施；每次被拒绝的访问都会记录 syslog 消息。这是 RHEL 7 中的默认设置。 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 的访问控制规则不评估，也不强制执行。

ad_gpo_implicit_deny 选项

默认情况下，**ad_gpo_implicit_deny** 选项设置为 **False**。在此默认状态下，如果找不到适用的 GPO，则允许用户访问。如果将这个选项设置为 **True**，则必须明确允许用户使用 GPO 规则进行访问。

您可以使用此功能来强化安全性，但请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 AD 服务器端定义的允许和拒绝登录权限以及 **ad_gpo_implicit_deny** 的值，允许或拒绝用户访问。

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

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

允许规则	拒绝规则	结果
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

- [更改 GPO 访问控制模式](#)
- 系统中 **sssd-ad (5)** 手册页

4.5.4. 更改 GPO 访问控制模式

此流程改变了在加入 Active Directory(AD)环境的 RHEL 主机上评估并实施基于 GPO 的访问控制规则的方式。

在本例中，您将为测试目的将 GPO 操作模式从 **enforcing**（默认）更改为 **permissive**。

重要

如果您看到以下错误，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/sss/sssd__example.com_.log` 中：

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

如果这是不需要的行为，您可以临时将 `ad_gpo_access_control` 设置为 `permissive`，在对 AD 中的正确 GPO 设置进行故障排除时，如这个过程所述。

先决条件

- 已使用 SSSD 将 RHEL 主机加入到 AD 环境中。
- 编辑 `/etc/sss/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/sss/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/sss/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

- [控制 GPO 强制的 SSSD 选项列表](#)

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 用户和 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。通过让 RHEL 主机在自己的 OU 中，GPO 将此主机作为目标。
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 Editor 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 local to grant local console/GUI access**。
 - 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**确定**。

其它资源

- Microsoft 文档中的 [组策略对象](#)

4.5.6. 其它资源

- [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)Managed Service Accounts(MSA)允许您在 AD 中创建与特定计算机对应的帐户。您可以将 MSA 以特定用户主体的形式连接到 AD 资源，而无需将 RHEL 主机加入到 AD 域中。

5.1. 受管服务帐户的好处

如果要允许 RHEL 主机在没有加入的情况下访问 Active Directory(AD)域，您可以使用 Managed Service Account(MSA)访问该域。MSA 是 AD 中的帐户，对应于一个特定的计算机，作为特定用户主体，可用于连接到 AD 资源。

例如，如果 AD 域 **production.example.com** 与 **lab.example.com** AD 域具有单向信任关系，则适用以下条件：

- 实验 域信任 **production** 域中的用户和主机。
- 生产 域 不信任 实验 域中的用户和主机。

这意味着主机加入了 实验 域，如 **client.lab.example.com**，无法通过信任从 生产 域访问资源。

如果要为 **client.lab.example.com** 主机创建一个例外，您可以使用 **adcli** 实用程序为 **production.example.com** 域中的 客户端主机 创建 MSA。通过与 MSA 的 Kerberos 主体进行身份验证，您可以从 客户端主机 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

此流程从 **lab.example.com** Active Directory(AD)域为主机创建一个受管服务帐户(MSA)，并配置 SSSD，以便您可以访问并验证 **production.example.com** AD 域。



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，而您想要在 AD 中为该主机创建帐户。
- 您已将 RHEL 主机加入到 AD 域中，您需要访问另一个 AD 域，其中来自您加入的域的主机凭证无效，比如使用单向信任。

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

服务	端口	协议
Kerberos	88	TCP, UDP

- 您有 AD Administrator 的密码，它有权在 **production.example.com** 域中创建 MSA。
- 您有运行 **adcli** 命令所需的 root 权限，并修改 **/etc/sss/sss.conf** 配置文件。
- 可选：已安装 **krb5-workstation** 软件包，其中包括 **klist** 诊断实用程序。

流程

1. 在 **production.example.com** AD 域中为主机创建一个 MSA。

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

2. 显示创建的 Kerberos keytab 中的 MSA 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/sss/sss.conf** 文件并选择要添加的相应 SSSD 域配置：

- 如果 MSA 对应于来自不同林的 AD 域，请创建一个名为 **[domain/<name_of_domain>]** 的新域部分，并输入 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
...
```



警告

即使具有现有的信任关系，**sss-ad** 还需要第二个林中的 MSA。

- 如果 MSA 对应于本地林的 AD 域，则以 `[domain/root.example.com/sub-domain.example.com]` 格式创建新子域部分，并输入关于 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

- 验证您可以检索 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)中是否有适用于主机的 MSA。

其它资源

- [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

管理服务帐户(MSA)具有一个复杂的密码，由 Active Directory(AD)自动维护。默认情况下，如果 System Services Security Daemon(SSSD)会在 Kerberos keytab 中自动更新 MSA 密码（如果它早于 30 天），它会使其使用 AD 中的密码保持最新状态。此流程解释了如何手动更新您的 MSA 密码。

先决条件

- 您之前已在 `production.example.com` AD 域中为主机创建了 MSA。
- 可选：已安装 **krb5-workstation** 软件包，其中包括 **klist** 诊断实用程序。

流程

- 可选：在 Kerberos keytab 中显示 MSA 的当前密钥版本号(KVNO)。当前 KVNO 为 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 域中的 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

- 验证您已在 Kerberos keytab 中递增了 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 工具创建的 Managed 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 主体保存在名为 **<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 个字符是来自数字的 3 个随机字符和小写 ASCII 范围 (附加到您提供的短主机名) 的后缀, 使用 **!** 字符作为分隔符。例如, 带有短名称 **myhost** 的主机接收以下规格的 MSA :

规格	订阅价值
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** 工具的全局选项外, 您还可以指定以下选项来专门控制它如何处理受管服务帐户 (MSA)。

-N, --computer-name

在 Active Directory(AD)域中要创建的 MSA 的简短非分配名称。如果没有指定名称，则 **--host-fqdn** 或其默认值的第一部分会带有一个随机后缀。

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

创建 MSA 的 Organizational Unit(OU)的完整可分辨名称。如果没有指定这个值，则 MSA 在默认位置 **OU=CN=Managed Service Accounts,DC=EXAMPLE,DC=COM** 会创建。

-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 凭据的主机 keytab 的路径。如果没有指定这个值，则默认位置 **/etc/krb5.keytab** 与添加为后缀（如 **/etc/krb5.keytab.domain.example.com**）的小写 Active Directory 域名一起使用。

--use-ldaps

通过安全 LDAP(LDAPS)频道创建 MSA。

--verbose

创建 MSA 时打印详细信息。

--show-details

创建 MSA 后打印出有关的信息。

--show-password

创建 MSA 后打印 MSA 密码。