# Red Hat Enterprise Linux 9

## 配置和使用网络文件服务

在 Red Hat Enterprise Linux 9 中配置和使用网络文件服务的指南。

Last Updated: 2022-07-18

# Red Hat Enterprise Linux 9 配置和使用网络文件服务

在 Red Hat Enterprise Linux 9 中配置和使用网络文件服务的指南。

Enter your first name here. Enter your surname here.
Enter your organisation's name here. Enter your organisational division here.
Enter your email address here.

## 法律通告



## 摘要

本文档论述了如何在 Red Hat Enterprise Linux 9（包括 Samba 服务器和 NFS 服务器）中配置和运行网络文件服务。

# 目录

# 使开源包含更多

红帽致力于替换我们的代码、文档和 Web 属性中存在问题的语言。我们从这四个术语开始：master、slave、黑名单和白名单。由于此项工作十分艰巨，这些更改将在即将推出的几个发行版本中逐步实施。详情请查看 CTO Chris Wright 的信息。

# 对红帽文档提供反馈

我们感谢您对文档提供反馈信息。请让我们了解如何改进文档。

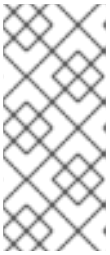
- 关于特定内容的简单评论：
  1. 请确定您使用 *Multi-page HTML* 格式查看文档。另外，确定 **Feedback** 按钮出现在文档页的右上方。
  2. 用鼠标指针高亮显示您想评论的文本部分。
  3. 点在高亮文本上弹出的 **Add Feedback**。
  4. 按照显示的步骤操作。
- 要通过 Bugzilla 提交反馈，请创建一个新的 ticket：
  1. 进入 Bugzilla 网站。
  2. 在 Component 中选择 **Documentation**。
  3. 在 **Description** 中输入您要提供的信息。包括文档相关部分的链接。
  4. 点 **Submit Bug**。

# 第 1 章 使用 SAMBA 作为服务器

Samba 在Red Hat Enterprise Linux中实现了服务器消息块(SMB)协议。SMB 协议用于访问服务器上的资源，如文件共享和共享打印机。此外，Samba 实现了Microsoft Windows 使用的分布式计算环境远程过程调用(DCE RPC)协议。

您可以以以下方式运行 Samba：

- Active Directory(AD)或 NT4 域成员
- 独立服务器
- NT4 主域控制器(PDC)或备份域控制器(BDC)



**注意**

红帽支持仅在支持 NT4 域的 Windows 版本的现有安装中支持 PDC 和 BDC 模式。红帽建议不要设置新的 Samba NT4 域，因为 Windows 7 和 Windows Server 2008 R2之后的Microsoft操作系不支持 NT4 域。

红帽不支持将 Samba 作为 AD 域控制器(DC)来运行。

有别于安装模式，您可以选择共享目录和打印机。这可让 Samba 充当文件和打印服务器。

## 1.1. 了解不同的 SAMBA 服务和模式

这部分论述了 Samba 中包含的不同服务以及您可以配置的不同模式。

### 1.1.1. Samba 服务

Samba 提供以下服务：

**smbd**

此服务使用 SMB 协议提供文件共享和打印服务。另外，该服务负责资源锁定和验证连接用户。要进行身份验证域成员，**smbd** 需要 **winbindd**。**smb systemd**服务启动并停止**smbd**守护进程。
要使用**smbd**服务，请安装**samba**软件包。

**nmbd**

此服务通过 IPv4 协议使用 NetBIOS 提供主机名和 IP 解析。除了名字解析之外，**nmbd**服务还支持浏览 SMB 网络来查找域、工作组、主机、文件共享和打印机。为此，服务可将此信息直接报告给广播客户端，或者将其转发到本地或主浏览器。**nmb systemd**服务启动并停止**nmbd**守护进程。
请注意，现代 SMB 网络使用 DNS 来解析客户端和 IP 地址。对于 Kerberos，需要一个正常工作的 DNS 设置。

要使用**nmbd**服务，请安装 **samba**软件包。

**winbindd**

该服务为名字服务交换机(NSS)提供了一个接口，以便使用本地系统上的 AD 或 NT4 域用户和组。例如，这使域用户能够对在 Samba 服务器上托管的服务或其他本地服务进行身份验证。**winbind systemd**服务启动并停止**winbindd**守护进程。
如果将 Samba 设置为域成员，则必须在**smbd**服务运行之前启动**winbindd**。否则，本地系统将无法使用域用户和组。

要使用**winbindd**服务，请安装**samba-winbind**软件包。

**重要**

红帽仅支持将 Samba 作为带有**winbindd**服务的服务器运行，以便为本地系统提供域用户和组。由于某些限制，如缺少 Windows 访问控制列表(ACL)支持和 NT LAN Manager(NTLM)回退，目前不支持 SSSD。

### 1.1.2. Samba 安全服务

**/etc/samba/smb.conf**文件中的**[global]**部分中的**security**参数管理 Samba 如何验证连接到该服务的用户的身份。根据您在其中安装 Samba 的模式，参数必须设为不同的值：

**对于 AD 域成员，设置security = ads**

在这个模式中，Samba 使用 Kerberos 来验证 AD 用户。
有关将 Samba 设置为域成员的详情，请参考 将 Samba 设置为 AD 域成员服务器。

**对于单独服务器，设置security = user**

在这个模式中，Samba 使用本地数据库验证连接用户。
有关将 Samba 设置为独立服务器的详情，请参考 将 Samba 设置为单机服务器。

**对于NT4 PDC 或 BDC，设置security = user**

在此模式中，Samba 将用户身份验证到本地或 LDAP 数据库。

**对于 NT4 域成员，设置security = domain**

在此模式中，Samba 将连接的用户验证到 NT4 PDC 或 BDC。您不能在 AD 域成员中使用这个模式。
有关将 Samba 设置为域成员的详情，请参考 将 Samba 设置为 AD 域成员服务器。

**其它资源**

- **smb.conf(5)** man page 中的 **security** 参数

### 1.1.3. Samba 服务和 Samba 客户端工具加载并重新载入其配置的情况

下面描述了 Samba 服务和工具加载并重新载入其配置：

- Samba 服务在以下情况下重新载入其配置：
  - 每 3 分钟自动进行
  - 在手动请求时，例如运行**smbcontrol all reload-config** 命令。
- Samba 客户端实用程序仅在启动时读取其配置。

请注意，某些参数（如**security** ）需要重启**smb**服务才能生效，而重新载入不足以生效。

**其它资源**

- **smb.conf(5)**手册页中的**如何应用配置更改** 部分
- **smbd(8)**、**nmbd(8)**和**winbindd(8)**手册页

### 1.1.4. 以安全的方式编辑 Samba 配置

Samba 服务每 3 分钟自动重新载入其配置。这个流程描述了在使用**testparm**工具验证配置前，如何以防止服务重新载入更改的方式编辑 Samba 配置。

**先决条件**

- 已安装 Samba。

**流程**

1. 创建**/etc/samba/smb.conf**文件的副本：

```
# cp /etc/samba/smb.conf /etc/samba/samba.conf.copy
```

2. 编辑复制的文件并进行必要的更改。
3. 验证**/etc/samba/samba.conf.*copy***文件中的配置：

```
# testparm -s /etc/samba/samba.conf.copy
```

   如果**testparm**报告错误，请修复这些错误，然后再次运行该命令。

4. 使用新配置覆盖**/etc/samba/smb.conf**文件：

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

5. 等待 Samba 服务自动重新载入其配置或手动重新载入配置：

```
# smbcontrol all reload-config
```

**其它资源**

- Samba 服务和 Samba 客户端工具加载并重新载入其配置的情况

## 1.2. 验证 SAMBA 配置

红帽建议您在每次更新**/etc/samba/smb.conf**文件后，验证 Samba 配置。本节提供有关此问题的详细信息。

### 1.2.1. 使用 testparm 工具验证 smb.conf 文件

**testparm**工具验证**/etc/samba/smb.conf**文件中的 Samba 配置是否正确。该工具不但检测无效的参数和值，还检测不正确的设置，如 ID 映射。如果**testparm**报告没有问题，Samba 服务将成功加载**/etc/samba/smb.conf**文件。请注意，**testparm**无法验证配置的服务是否可用或按预期工作。

**重要**

红帽建议在每次修改此文件后，使用**testparm**来验证**/etc/samba/smb.conf**文件。

**先决条件**

- 已安装 Samba。

- 退出**/etc/samba/smb.conf**文件。

**流程**

1. 以**root**用户身份运行**testparm**工具：

```
# testparm
Load smb config files from /etc/samba/smb.conf
rlimit_max: increasing rlimit_max (1024) to minimum Windows limit (16384)
Unknown parameter encountered: "log levell"
Processing section "[example_share]"
Loaded services file OK.
ERROR: The idmap range for the domain * (tdb) overlaps with the range of DOMAIN (ad)!

Server role: ROLE_DOMAIN_MEMBER

Press enter to see a dump of your service definitions

# Global parameters
[global]
 ...

[example_share]
 ...
```

前面的示例输出会报告不存在的参数以及不正确的 ID 映射配置。

2. 如果**testparm**报告了配置中不正确的参数、值或其他错误，请修复问题并再次运行该工具。

## 1.3. 将 SAMBA 设置为独立服务器

您可以将 Samba 设置为不是域成员的服务器。在此安装模式中，Samb身份验证到本地数据库，而不是中央DC。另外，您可以启用客户机访问，允许用户在没有身份验证的情况下连接到一个或多个服务。

### 1.3.1. 为独立服务器设置服务器配置

这部分论述了如何为 Samba 独立服务器设置服务器配置。

**流程**

1. 安装**samba**软件包：

```
# dnf install samba
```

2. 编辑**/etc/samba/smb.conf**文件并设置以下参数：

```
[global]
 workgroup = Example-WG
 netbios name = Server
 security = user

 log file = /var/log/samba/%m.log
 log level = 1
```

此配置在**Example-learning**工作组里定义了一个名为**Server**的独立服务器。此外，此配置启用了最小级别(**1**)的日志记录，日志文件将存储在**/var/log/samba/**目录中。Samba 将把 **日志文件** 参数中的**%m** 宏扩展到连接客户端的 NetBIOS 名称。这可为每个客户端启用独立的日志文件。

3. （可选）配置文件或打印机共享。请参阅：
   - 设置使用 POSIX ACL 的共享
   - 设置使用 Windows ACL 的共享
   - 将 Samba 设置为打印服务器
4. 验证**/etc/samba/smb.conf**文件：

```
# testparm
```

5. 如果您设置了需要身份验证的共享，请创建用户帐户。
   详情请参阅 创建和启用本地用户帐户。
6. 打开所需的端口并使用**firewall-cmd**工具重新载入防火墙配置：

```
# firewall-cmd --permanent --add-service=samba
# firewall-cmd --reload
```

7. 启用并启动**smb**服务：

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

**其它资源**

- **smb.conf(5)** man page

### 1.3.2. 创建并启用本地用户帐户

要让用户在连接到共享时进行身份验证，您必须在 Samba 主机上的操作系统和 Samba 数据库中创建帐户。Samba 要求操作系统帐户验证文件系统对象上的访问控制列表(ACL)和 Samba 帐户，来验证连接用户的身份。

如果您使用了**passdb backend = tdbsam** 默认设置，Samba 会将用户帐户存储在**/var/lib/samba/private/passdb.tdb**数据库中。

本节中的流程论述了如何创建名为**example**的本地 Samba 用户。

**先决条件**

- Samba 安装并配置为独立服务器。

**流程**

1. 创建操作系统帐户：

```
# useradd -M -s /sbin/nologin example
```

此命令添加了**example**帐户，而不创建主目录如果帐户仅用于对 Samba 进行身份验证，请将**/sbin/nologin**命令指定为 shell，以防止帐户在本地登录。

2. 为操作系统帐户设置密码以启用它：

```
# passwd example
Enter new UNIX password: password
Retype new UNIX password: password
passwd: password updated successfully
```

Samba 不会使用操作系统帐户中的密码集进行身份验证。然而，您需要设置密码才能启用帐户。如果一个帐户被禁用，当这个用户连接时，Samba 会拒绝访问。

3. 将用户添加到 Samba 数据库，并为帐户设置密码：

```
# smbpasswd -a example
New SMB password: password
Retype new SMB password: password
Added user example.
```

当使用此帐户连接到 Samba 共享时，使用此密码进行验证。

4. 启用 Samba 帐户：

```
# smbpasswd -e example
Enabled user example.
```

## 1.4. 了解并配置 SAMBA ID 映射

Windows 域通过唯一安全标识符(SID)来区分用户和组。但是，Linux 需要为每个用户和组群有唯一的 UID 和 GID。如果您以域成员身份运行 Samba，**winbindd**服务负责向操作系统提供域用户和组的信息。

要启用**winbindd**服务来向 Linux提供唯一的用户和组ID，您必须在**/etc/samba/smb.conf**文件中为以下情况配置 ID 映射：

- 本地数据库（默认域）
- Samba 服务器所属的 AD 或 NT4 域
- 每个用户必须能够访问这个 Samba 服务器上的资源的可信域

Samba 为特定配置提供不同的 ID 映射后端。最常用的后端是：

| 后端 | 使用案例 |
| --- | --- |
| **tdb** | ***仅限默认域** |
| **ad** | 仅限 AD 域 |
| **rid** | AD 和 NT4 域 |
| **autorid** | AD、NT4 和 ***默认域** |

### 1.4.1. 规划 Samba ID 范围

无论您在 AD 中是否存储了 Linux UID 和 GID，还是将 Samba 配置为生成它们，每个域配置都需要一个唯一的 ID 范围，其不得与任何其他域重叠。

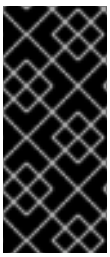
**警告**

如果您设置了重叠 ID 范围，Samba 无法正常工作。

**例 1.1. 唯一的 ID 范围**

以下显示了默认(*)、**AD-DOM**和**TRUST-DOM**域的非重叠 ID 映射范围。

```
[global]
...
idmap config * : backend = tdb
idmap config * : range = 10000-999999

idmap config AD-DOM:backend = rid
idmap config AD-DOM:range = 2000000-2999999

idmap config TRUST-DOM:backend = rid
idmap config TRUST-DOM:range = 4000000-4999999
```



**重要**

每个域只能分配一个范围。因此，在域范围之间有足够的空间。这可让您在域扩展后扩展范围。

如果您稍后给某个域分配了一个不同的范围，那么之前由这些用户和组创建的文件和目录的所有权将会丢失。

### 1.4.2. * 默认域

在域环境中，您可以为以下每个情况添加一个 ID 映射配置：

- Samba 服务器所属的域
- 每个可以访问 Samba 服务器的可信域

但是，对于所有其他对象，Samba 会从默认域分配 ID。这包括：

- 本地 Samba 用户和组
- Samba 内置帐户和组，如**BUILTIN\Administrators**

**重要**

您必须按照本节所述配置默认域，才可以使Samba正常运行。

默认域后端必须可写，才能永久存储分配的 ID。

对于默认域，您可以使用以下后端之一：

**tdb**

当您将默认域配置为使用**tdb**后端时，请设置一个足够大的 ID 范围，以包含将来要创建的对象，这些对象不属于已定义的域ID映射配置的一部分。
例如，在**/etc/samba/smb.conf**文件中的**[global]**部分中设置以下内容：

```
idmap config * : backend = tdb
idmap config * : range = 10000-999999
```

详情请查看使用 TDB ID 映射后端。

**autorid**

当您将默认域配置为使用**autorid**后端时，为域添加额外的 ID 映射配置是可选的。
例如，在**/etc/samba/smb.conf**文件中的**[global]**部分中设置以下内容：

```
idmap config * : backend = autorid
idmap config * : range = 10000-999999
```

详情请查看使用 自动 ID 映射后端。

### 1.4.3. 使用 tdb ID 映射后端

**winbindd**服务默认使用可写的**tdb** ID 映射后端来存储安全标识符(SID)、UID 以及 GID 映射表。这包括本地用户、组和内置主体。

仅将此后端用于*默认域。例如：

```
idmap config * : backend = tdb
idmap config * : range = 10000-999999
```

**其它资源**

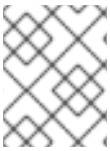
- * 默认域。

### 1.4.4. 使用 ad ID 映射后端

本节论述了如何将 Samba AD 成员配置为使用**ad** ID 映射后端。

**ad** ID 映射后端实现了一个只读 API，以便从 AD 读取帐户和组信息。它具有以下优点：

- 所有用户和组群设置都集中存储在 AD 中。
- 使用这个后端的所有 Samba 服务器中的用户和组群 ID 是一致的。
- ID 不会存储在本地数据库中（本地数据库可能会被损坏），因此文件所有者不会丢失。



**注意**

**ad** ID 映射后端不支持具有单向信任的Active Directory域。如果您使用单向信任在 Active Directory 中配置域成员，请使用以下一种 ID 映射后端： **tdb**、**delete** 或 **autorid**。

后端从 AD 读取以下属性：

| AD 属性名称 | 对象类型 | 映射到 |
|---|---|---|
| **sAMAccountName** | 用户和组群 | 用户和组名称，取决于对象 |
| **uidNumber** | User | 用户 ID（UID） |
| **gidNumber** | 组 | 组 ID（GID） |
| **loginShell**[a] | User | 用户 shell 的路径 |
| **unixHomeDirectory**[a] | User | 用户主目录的路径 |
| **primaryGroupID**[b] | User | 主组群 ID |

[a] 如果您设置了 **idmap config *DOMAIN*:unix_nss_info = yes**，则 Samba 只读取这个属性。

[b] 如果您设置了 **idmap config *DOMAIN*:unix_primary_group = yes**，则 Samba 只读取这个属性。

**先决条件**

- 用户和组必须在 AD 中设置唯一的 ID，并且 ID 必须在 **/etc/samba/smb.conf** 文件中配置的范围之内。其ID 不在范围之内的对象在 Samba 服务器上不可用。
- 用户和组必须在 AD 中设置所有必需的属性。如果缺少所需的属性，该用户或组将无法在 Samba 服务器中可用。所需的属性取决于您的配置。
- 已安装 Samba。
- Samba 配置（除了ID 映射）位于 **/etc/samba/smb.conf** 文件中。

**流程**

1. 编辑 **/etc/samba/smb.conf** 文件中的 **[global]** 部分：
   a. 如果默认域(*)不存在，请为其添加 ID 映射配置。例如：

```
idmap config * : backend = tdb
idmap config * : range = 10000-999999
```

   b. 为 AD 域启用 **ad** ID 映射后端：

```
idmap config DOMAIN : backend = ad
```

c. 设置分配给 AD 域中用户和组的 ID 范围。例如：

```
idmap config DOMAIN : range = 2000000-2999999
```

**重要**

范围不得与这个服务器上的任何其他域配置重叠。此外，范围必须足够大，以便包含将来分配的所有 ID。详情请查看 规划 Samba ID 范围。

d. 当从AD读取属性时，使用RFC 2307模式来设置 Samba：

```
idmap config DOMAIN : schema_mode = rfc2307
```

e. 要让 Samba 从对应的 AD 属性读取登录 shell 和用户主目录的路径，请设置：

```
idmap config DOMAIN : unix_nss_info = yes
```

或者，您可以设置适用于所有用户的统一的域范围的主目录路径和登录 shell。例如：

```
template shell = /bin/bash
template homedir = /home/%U
```

f. 默认情况下，Samba 使用用户对象的 **primaryGroupID** 属性作为 Linux 上用户的主组。或者，您可以将 Samba 配置为使用**gidNumber**属性中设置的值：

```
idmap config DOMAIN : unix_primary_group = yes
```

2. 验证**/etc/samba/smb.conf**文件：

```
# testparm
```

3. 重新载入 Samba 配置：

```
# smbcontrol all reload-config
```

**其它资源**

- * 默认域
- **smb.conf(5)** and **idmap_ad(8)** man pages
- **smb.conf(5)** 手册页中的 VARI **STITUTIONS** 部分

### 1.4.5. 使用网格 ID 映射后端

这部分论述了如何配置 Samba 域成员以使用**rid** ID映射后端。

Samba 可以使用 Windows SID 的相对标识符(RID)，以便在 Red Hat Enterprise Linux 上生成 ID。

**注意**

RID 是 SID 的最后部分。例如，如果用户的 SID 是**S-1-5-21-5421822485-1151247151-421485315-30014**，那么**30014**是对应的 RID。

**rid**ID映射后端实施了一个只读 API，以便根据 AD 和 NT4 域的算法映射方案计算帐户和组信息。当配置后端时，您必须在 **idmap config *DOMAIN* : *range***参数中设置最低和最高的 RID。Samba 不会映射比这个参数中设置低或更高 RID 的用户或组。

**重要**

作为只读后端，**rid**无法分配新的ID，例如为**BUILTIN**组。因此，请勿将此后端用于 * 默认域。

**使用网格后端的好处**

- 所有在配置范围内具有 RID 的域用户和组都会自动在域成员中可用。
- 您不需要手动分配 ID、主目录和登录 shell。

**使用网格后端的缺陷**

- 所有域用户可以获得相同的登录 shell 和主目录。但是，您可以使用变量。
- 如果它们都使用具有相同ID范围设置的**rid**后端，那么用户和组ID只在 Samba 域成员之间是相同的。
- 您不能阻止单独的用户或组在域成员中可用。只有超出配置范围以外的用户和组才会包括。
- 根据 **winbindd** 服务用于计算 ID 的公式，如果不同域中的对象有相同的 RID，那么在多域环境中可能会有重复ID的事情发生 。

**先决条件**

- 已安装 Samba。
- Samba 配置（除了ID 映射）位于 **/etc/samba/smb.conf** 文件中。

**流程**

1. 编辑 **/etc/samba/smb.conf** 文件中的 **[global]** 部分：

   a. 如果默认域(*)不存在，请为其添加 ID 映射配置。例如：

   ```
   idmap config * : backend = tdb
   idmap config * : range = 10000-999999
   ```

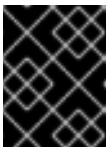
   b. 为域启用**rid**ID映射后端：

   ```
   idmap config DOMAIN : backend = rid
   ```

   c. 设置一个足够大的范围，以包括将来将要分配的所有RID。例如：

   ```
   idmap config DOMAIN : range = 2000000-2999999
   ```

Samba 会忽略此域中其RID不在范围内的用户和组。



**重要**

范围不得与这个服务器上的任何其他域配置重叠。此外，范围必须足够大，以便包含将来分配的所有 ID。详情请查看 规划 Samba ID 范围。

d. 设置分配给所有映射用户的 shell 和主目录路径。例如：

```
template shell = /bin/bash
template homedir = /home/%U
```

2. 验证**/etc/samba/smb.conf**文件：

```
# testparm
```

3. 重新载入 Samba 配置：

```
# smbcontrol all reload-config
```

**其它资源**

- * 默认域
- **smb.conf(5)** 手册页中的 VARI **STITUTIONS** 部分
- 从 RID 中计算本地 ID，请查看 **idmap_rid(8)** man page

### 1.4.6. 使用自动 ID 映射后端

这部分描述了如何配置 Samba 域成员，以便使用**autorid**ID映射后端。

**autorid**后端的工作方式与**rid**ID 映射后端类似，但可以为不同的域自动分配 ID。这可让您在以下情况下使用**autorid**后端：

- 仅用于*默认域
- 对于*默认域和附加域，不需要为每个附加域创建 ID 映射配置
- 只适用于特定域

**注意**

如果您对默认域使用**autorid**，为域添加额外的 ID 映射配置是可选的。

本节的部分内容来自在 Samba Wiki 中发布的 idmap config autorid 文档。许可证： CC BY 4.0。作者和贡献者：请参阅 Wiki页面上的历史选项卡。

**使用自动扩展后端的好处**

- 所有在配置范围内计算 UID 和 GID 的域用户和组都会在域成员中自动可用。
- 您不需要手动分配 ID、主目录和登录 shell。

- 没有重复的 ID，即使多域环境中的多个对象有相同的 RID。

**缺陷**

- 在 Samba 域成员中用户和组群 ID 不相同。
- 所有域用户可以获得相同的登录 shell 和主目录。但是，您可以使用变量。
- 您不能阻止单独的用户或组在域成员中可用。只有计算 UID 或 GID 不在配置范围内的用户和组才会包括。

**先决条件**

- 已安装 Samba。
- Samba 配置（除了ID 映射）位于 **/etc/samba/smb.conf** 文件中。

**流程**

1. 编辑 **/etc/samba/smb.conf** 文件中的 **[global]** 部分：
   a. 为 * 默认域启用**autorid**ID映射后端：

```
idmap config * : backend = autorid
```

   b. 设置一个足够大的范围来为所有现有和将来的对象分配 ID。例如：

```
idmap config * : range = 10000-999999
```

   Samba 忽略在此域中计算 ID 不在范围范围内的用户和组。

**警告**

设置范围并开始使用 Samba 后，您只能增加范围的上限。对范围的任何其他变化都可能会导致分配新的 ID，从而会丢失文件的所有者信息。

   c. 另外，还可设置范围大小。例如：

```
idmap config * : rangesize = 200000
```

   Samba 会为每个域的对象分配这个连续的ID号，直到 **idmap config * : *range***参数中设置的范围内的所有ID分配完。

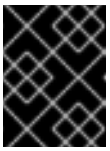
**注意**

如果设置 rangesize，则需要相应地调整范围。范围必须是 rangesize 的倍数。

   d. 设置分配给所有映射用户的 shell 和主目录路径。例如：

```
template shell = /bin/bash
template homedir = /home/%U
```

e. 另外，还可为域添加额外的 ID 映射配置。如果没有针对单个域的配置，Samba 则使用之前配置的 * 默认域中的 **autorid** 后端设置来计算 ID。



**重要**

范围不得与这个服务器上的任何其他域配置重叠。此外，范围必须足够大，以便包含将来分配的所有 ID。详情请查看 规划 Samba ID 范围。

2. 验证**/etc/samba/smb.conf**文件：

```
# testparm
```

3. 重新载入 Samba 配置：

```
# smbcontrol all reload-config
```

**其它资源**

- 在 **idmap_autorid(8)手册页中的 idmap_autorid(8) man page 中的 THEptING FORMULAS** 部分
- **idmap_autorid(8)** man page 中的 **rangesize** 参数描述
- **smb.conf(5)** 手册页中的 VARI **STITUTIONS** 部分

## 1.5. 将 SAMBA 设置为 AD 域成员服务器

如果您正在运行 AD 或 NT4 域，请使用 Samba 将 Red Hat Enterprise Linux 服务器添加为域的成员，以便可以：

- 访问其他域成员上的域资源
- 对本地服务（如 **sshd**）验证域用户
- 托管在服务器上的共享目录和打印机，以充当文件和打印服务器

### 1.5.1. 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。这部分论述了如何使用 **realmd** 配置 Samba Winbind 将 RHEL 系统加入到 AD 域中。

**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。

详情请参阅 了解和配置 Samba ID 映射。

1. 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。

2. 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
```

3. 另外，如果您要验证Active Directory的本地登录，请启用**winbind_krb5_localauth**插件。请参阅在 MIT Kerberos 中使用本地授权插件。

**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。请参阅使用 GPO 在 Active Directory 中启用 AES 加密类型。请注意，这可能会对您的 AD 中的其他服务产生影响。
- **realm(8)** man page

### 1.5.2. 使用 MIT Kerberos 的本地授权插件

**winbind**服务向域成员提供Active Directory用户。在某些情况下，管理员希望域用户能够对域成员上运行的本地服务（如SSH服务器）启用身份验证。当使用 Kerberos 来验证域用户时，启用**winbind_krb5_localauth**插件，通过**winbind** 服务将 Kerberos 主体正确映射到Active Directory帐户。

例如，如果Active Directory用户的**sAMAccountName**属性设置为**EXAMPLE**，并且用户尝试使用小写的用户名进行日志记录，Kerberos将返回大写的用户名。因此，条目不匹配，身份验证失败。

使用**winbind_krb5_localauth**插件时，帐户名称会被正确映射。请注意，这只适用于 GSSAPI 身份验证，不适用于获取初始票据授权票据(TGT)。

**先决条件**

- Samba 配置为 Active Directory 的成员。
- Red Hat Enterprise Linux 对 Active Directory 进行身份验证。
- **winbind**服务在运行。

**流程**

编辑**/etc/krb5.conf**文件，并添加以下部分：

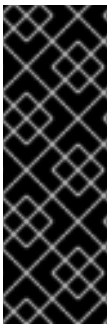
```
[plugins]
localauth = {
    module = winbind:/usr/lib64/samba/krb5/winbind_krb5_localauth.so
    enable_only = winbind
}
```

**其它资源**

- **winbind_krb5_localauth(8)** man page.

## 1.6. 在 IDM 域成员中设置 SAMBA

本节描述了如何在加入到 Red Hat Identity Management(IdM)域的主机上设置 Samba。来自IdM的用户，以及来自受信任的 Active Directory(AD)域的用户(如果有的话)可以访问 Samba 提供的共享和打印机服务。



**重要**

对 IdM 域成员使用 Samba 是一种不受支持的技术预览特性，且包含了某些限制。例如，由于 IdM 信任控制器不支持全局目录服务，注册了AD的 Windows 主机无法在 Windows 中找到 IdM 用户和组。另外，IdM Trust Controller 不支持使用分布式计算环境/远程过程调用（DCE/RPC）协议解析 IdM 组。因此，AD 用户只能访问 IdM 客户端的 Samba 共享和打印机。

我们鼓励在 IdM 域成员中部署 Samba 的用户向红帽提供反馈意见。

**先决条件**

- 主机作为 IdM 域的客户端加入。

### 1.6.1. 准备 IdM 域以便在域成员中安装 Samba

在IdM客户端上设置Samba之前，必须在IdM服务器上使用**ipa-adtrust-install**工具来准备IdM域。

**注意**

运行**ipa-adtrust-install**命令的任何系统都会自动成为 AD 信任控制器。但是，您必须在 IdM 服务器上只运行一次 **ipa-adtrust-install** 。

**先决条件**

- IdM 服务器已安装。
- 您需要 root 权限才能安装软件包并重新启动 IdM 服务。

**步骤**

1. 安装所需的软件包：

```
[root@ipaserver ~]# dnf install ipa-server-trust-ad samba-client
```

2. 以 IdM 管理用户身份进行身份验证：

```
[root@ipaserver ~]# kinit admin
```

3. 运行 **ipa-adtrust-install** 工具：

```
[root@ipaserver ~]# ipa-adtrust-install
```

如果 IdM 安装了集成的 DNS 服务器，则会自动创建 DNS 服务记录。

如果您在没有集成 DNS 服务器的情况下安装了 IdM，**ipa-adtrust-install** 会打印一个服务记录列表，您必须手动将它们添加到 DNS，然后才能继续操作。

4. 该脚本提示您 **/etc/samba/smb.conf** 已存在，并将被重写：

```
WARNING: The smb.conf already exists. Running ipa-adtrust-install will break your existing
Samba configuration.

Do you wish to continue? [no]: yes
```

5. 该脚本提示您配置 **slapi-nis** 插件，这是一个兼容插件，允许旧的 Linux 客户端与受信任的用户一起工作：

```
Do you want to enable support for trusted domains in Schema Compatibility plugin?
This will allow clients older than SSSD 1.9 and non-Linux clients to work with trusted users.

Enable trusted domains support in slapi-nis? [no]: yes
```

6. 提示时，输入 IdM 域的 NetBIOS 名称，或者按 **Enter** 接受推荐的名称：

```
Trust is configured but no NetBIOS domain name found, setting it now.
Enter the NetBIOS name for the IPA domain.
Only up to 15 uppercase ASCII letters, digits and dashes are allowed.
Example: EXAMPLE.

NetBIOS domain name [IDM]:
```

7. 系统会提示您运行 SID 生成任务，以便为任何现有用户创建 SID：

```
Do you want to run the ipa-sidgen task? [no]: yes
```

这是一个资源密集型任务，因此如果您有大量的用户，您可以在其他时间运行此操作。

8. **（可选）** 默认情况下，对于 Windows Server 2008 及更高版本，动态 RPC 端口范围定义为 **49152-65535**。如果需要为您的环境定义一个不同的动态 RPC 端口范围，请将 Samba 配置为使用不同的端口，并在防火墙设置中开放这些端口。以下示例将端口范围设置为**55000-65000**。

```
[root@ipaserver ~]# net conf setparm global 'rpc server dynamic port range' 55000-65000
[root@ipaserver ~]# firewall-cmd --add-port=55000-65000/tcp
[root@ipaserver ~]# firewall-cmd --runtime-to-permanent
```

9. 重启**ipa**服务：

```
[root@ipaserver ~]# ipactl restart
```

10. 使用**smbclient**工具来验证 Samba 是否响应 IdM 端的 Kerberos 身份验证：

```
[root@ipaserver ~]# smbclient -L server.idm.example.com -U user_name --use-kerberos=required
lp_load_ex: changing to config backend registry
   Sharename       Type      Comment
   ---------       ----      -------
   IPC$            IPC       IPC Service (Samba 4.15.2)
...
```

### 1.6.2. 使用 GPO 在 Active Directory 中启用 AES 加密类型

这部分描述了如何使用组策略对象(GPO)在 Active Directory(AD)中启用 AES 加密类型。RHEL 上的某些功能（如在 IdM 客户端上运行 Samba 服务器）需要这个加密类型。

请注意，RHEL 不再支持弱 DES 和 RC4 加密类型。

**先决条件**

- 以可编辑组策略的用户身份登录到 AD。
- 计算机上安装了组**策略管理控制台**。

**步骤**

1. 打开组**策略管理控制台**。
2. 右键单击**默认域策略**，然后选择**编辑**。打开组**策略管理编辑器**。
3. 导航到 **计算机配置 → 策略 → Windows 设置 → 安全设置 → 本地策略 → 安全选项**。
4. 双击 **网络安全：配置 Kerberos 策略允许的加密类型**。
5. 选择**AES256_HMAC_SHA1**和可选的**未来加密类型**。
6. 点**确定**。
7. 关闭组**策略管理编辑器**。
8. 对**默认域控制器策略**重复上述步骤。

9. 等待 Windows 域控制器(DC)自动应用组策略。或者，如果要在 DC 上手动应用 GPO，请使用具有管理员权限的帐户输入以下命令：

```
C:\> gpupdate /force /target:computer
```

### 1.6.3. 在 IdM 客户端中安装和配置 Samba 服务器

这部分论述了如何在在 IdM 域注册的客户端中安装和配置 Samba。

**先决条件**

- IdM 服务器和客户端必须在 RHEL 9.0 或更高版本中运行。
- 已准备好 IdM 域，如 为在域成员上安装 Samba 准备 IdM 域 中所述。
- 如果 IdM 具有配置了 AD 的信任，请为 Kerberos 启用 AES 加密类型。例如，使用组策略对象(GPO)来启用 AES 加密类型。详情请参阅 使用 GPO 在活动目录中启用 AES 加密 。

**流程**

1. 安装**ipa-client-samba**软件包：

```
[root@idm_client]# dnf install ipa-client-samba
```

2. 使用**ipa-client-samba**工具准备客户端并创建初始 Samba 配置：

```
[root@idm_client]# ipa-client-samba
Searching for IPA server...
IPA server: DNS discovery
Chosen IPA master: idm_server.idm.example.com
SMB principal to be created: cifs/idm_client.idm.example.com@IDM.EXAMPLE.COM
NetBIOS name to be used: IDM_CLIENT
Discovered domains to use:

 Domain name: idm.example.com
NetBIOS name: IDM
        SID: S-1-5-21-525930803-952335037-206501584
   ID range: 212000000 - 212199999

 Domain name: ad.example.com
NetBIOS name: AD
        SID: None
   ID range: 1918400000 - 1918599999

Continue to configure the system with these values? [no]: yes
Samba domain member is configured. Please check configuration at /etc/samba/smb.conf
and start smb and winbind services
```

3. 默认情况下，**ipa-client-samba**会自动将**[homes]**部分添加到**/etc/samba/smb.conf**文件中，该文件在用户连接时动态共享用户的主目录。如果用户在这个服务器上没有主目录，或者您不想共享主目录，请从**/etc/samba/smb.conf**中删除以下行：

```
[homes]
   read only = no
```

4. 共享目录和打印机。详情请查看：
   - 设置使用 POSIX ACL 的 Samba 文件共享
   - 设置使用 Windows ACL 的共享
   - 将 Samba 设置为打印服务器
5. 在本地防火墙中打开 Samba 客户端所需的端口：

```
[root@idm_client]# firewall-cmd --permanent --add-service=samba-client
[root@idm_client]# firewall-cmd --reload
```

6. 启用并启动**smb**和**winbind**服务：

```
[root@idm_client]# systemctl enable --now smb winbind
```

**验证步骤**

在安装了 **samba-client** 软件包的不同 IdM 域成员中运行以下验证步骤：

- 使用 Kerberos 身份验证列出 Samba 服务器中的共享：

```
$ smbclient -L idm_client.idm.example.com -U user_name --use-kerberos=required
lp_load_ex: changing to config backend registry

    Sharename       Type      Comment
    ---------       ----      -------
    example         Disk
    IPC$            IPC       IPC Service (Samba 4.15.2)
...
```

**其它资源**

- **ipa-client-samba(1)** man page。

### 1.6.4. 如果 IdM 信任新域，请手动添加 ID 映射配置

Samba 需要一个 ID 映射配置，用户可从该域访问资源。在 IdM 客户端上运行的现有 Samba 服务器上，在管理员向 Active Directory(AD)域添加了新的信任后，您必须手动添加 ID 映射配置。

**先决条件**

- 您在 IdM 客户端中配置了 Samba。之后，IdM 增加了一个新的信任。
- 在可信 AD 域中必须禁用 Kerberos 的 DES 和 RC4 加密类型。为了安全起见，RHEL 9 不支持这些弱加密类型。

**步骤**

1. 使用主机的 keytab 进行身份验证：

```
[root@idm_client]# kinit -k
```

2. 使用**ipa idrange-find**命令来显示新域的基本 ID 和 ID 范围大小。例如，以下命令显示了**ad.example.com**域的值：

```
[root@idm_client]# ipa idrange-find --name="AD.EXAMPLE.COM_id_range" --raw
---------------
1 range matched
---------------
  cn: AD.EXAMPLE.COM_id_range
  ipabaseid: 1918400000
  ipaidrangesize: 200000
  ipabaserid: 0
  ipanttrusteddomainsid: S-1-5-21-968346183-862388825-1738313271
  iparangetype: ipa-ad-trust
----------------------------
Number of entries returned 1
----------------------------
```

在后续步骤中，您需要**ipabaseid**和**ipaidrangesize**属性的值。

3. 要计算可用最高的 ID，请使用以下公式：

```
maximum_range = ipabaseid + ipaidrangesize - 1
```

使用上一步中的值，**ad.example.com**域的最大可用 ID 是**1918599999**(1918400000 + 200000 - 1)。

4. 编辑**/etc/samba/smb.conf**文件，并将域的 ID 映射配置添加到**[global]**部分：

```
idmap config AD : range = 1918400000 - 1918599999
idmap config AD : backend = sss
```

将**ipabaseid**属性的值指定为最小值，将上一步中的计算值指定为该范围的最大值。

5. 重启**smb**和**winbind**服务：

```
[root@idm_client]# systemctl restart smb winbind
```

**验证步骤**

- 使用 Kerberos 身份验证列出 Samba 服务器中的共享：

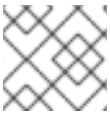
```
$ smbclient -L idm_client.idm.example.com -U user_name --use-kerberos=required
lp_load_ex: changing to config backend registry

   Sharename       Type      Comment
   ---------       ----      -------
   example         Disk
   IPC$            IPC       IPC Service (Samba 4.15.2)
...
```

### 1.6.5. 其它资源

- 请参阅安装身份管理客户端

## 1.7. 设置使用 POSIX ACL 的 SAMBA 文件共享

作为 Linux 服务，Samba 支持与 POSIX ACL 的共享。它们允许您使用诸如**chmod**等工具在 Samba 服务器上本地管理权限。如果共享是存储在支持扩展属性的文件系统中，您可以使用多个用户和组定义 ACL。



**注意**

如果您需要使用精细的 Windows ACL，请参阅设置使用 Windows ACL 的共享。

这个部分的内容基于 Samba Wiki 中发布的 Setting up a Share Using POSIX ACLs 文档。许可证： CC BY 4.0。作者和贡献者：请参阅 Wiki 页面上的历史选项卡。

### 1.7.1. 添加使用 POSIX ACL 的共享

这部分描述了如何创建名为**example**的共享，该共享提供了**/srv/samba/example/**目录的内容，并使用了 POSIX ACL。

#### 先决条件

Samba 采用以下模式之一设置：

- 独立服务器
- 域成员

#### 流程

1. 如果不存在，请创建文件夹。例如：

```
# mkdir -p /srv/samba/example/
```

2. 如果您在**enforcing**模式下运行 SELinux，请在目录中设置**samba_share_t**上下文：

```
# semanage fcontext -a -t samba_share_t "/srv/samba/example(/.*)?"
# restorecon -Rv /srv/samba/example/
```

3. 在目录中设置文件系统 ACL。详情请查看：
   - 在使用 POSIX ACL 的 Samba 共享中设置标准 ACL
   - 在使用 POSIX ACL 的共享中设置扩展 ACL。
4. 将示例共享添加到**/etc/samba/smb.conf**文件中。例如，添加启用了共享的写操作：

```
[example]
 path = /srv/samba/example/
 read only = no
```

**注意**

无论文件系统 ACL 是什么；如果您没有设置**read only = no**，Samba 会以只读模式共享该目录。

5. 验证**/etc/samba/smb.conf**文件：

```
# testparm
```

6. 打开所需的端口，并使用**firewall-cmd**工具重新加载防火墙配置：

```
# firewall-cmd --permanent --add-service=samba
# firewall-cmd --reload
```

7. 重启**smb**服务：

```
# systemctl restart smb
```

### 1.7.2. 在使用 POSIX ACL 的 Samba 共享中设置标准 Linux ACL

Linux 中的标准 ACL 支持为一个所有者、一个组和所有其他未定义用户设置权限。您可以使用**chown**、**chgrp**和**chmod**工具来更新 ACL。如果您需要精确控制，请使用更复杂的 POSIX ACL，请参阅

在使用 POSIX ACL 的 Samba 共享中设置扩展 ACL 。在使用 POSIX ACL 的 Samba 共享中设置扩展 ACL。以下步骤将**/srv/samba/example/**目录的所有者设置为**root**用户，将读写权限赋予**Domain Users**组，并拒绝所有其他用户的访问。

**先决条件**

- 存在要设置 ACL 的 Samba 共享。

**流程**

```
# chown root:"Domain Users" /srv/samba/example/
# chmod 2770 /srv/samba/example/
```

**注意**

对目录启用 set-group-ID(SGID)位会自动对目录组的所有新文件和子目录设置默认组，而不是通常的行为，将其设置为创建新目录条目的用户的主组。

**其它资源**

- **chown(1)** 和 **chmod(1)** man page

### 1.7.3. 在使用 POSIX ACL 的 Samba 共享中设置扩展的 ACL

如果文件系统中保存了共享目录的支持扩展 ACL，您可以使用它们设置复杂的权限。扩展 ACL 可以包含多个用户和组群的权限。

扩展 POSIX ACL 可让您使用多个用户和组配置复杂的 ACL。但是，您只能设置以下权限：

- 无权限
- 读权限
- 写权限

- 全控制

如果您需要更细粒度的 Windows 权限，如 **创建文件夹 / 追加数据**，请将共享配置为使用 Windows ACL。

请参阅 设置使用 Windows ACL 的共享。

以下流程演示了如何在共享中启用扩展 ACL。另外，它还包含有关设置扩展 ACL 的示例。

**先决条件**

- 存在要设置 ACL 的 Samba 共享。

**流程**

1. 在**/etc/samba/smb.conf**文件中的共享部分启用以下参数，以启用扩展 ACL 的 ACL 继承：

```
inherit acls = yes
```

   详情请查看 **smb.conf(5)**手册页中的参数描述。

2. 重启**smb**服务：

```
# systemctl restart smb
```

3. 在目录中设置 ACL。例如：

   **例 1.2. 设置扩展 ACL**

   以下步骤为**Domain Admins**组设置读、写和执行权限，为**Domain Users**组设置读和执行权限，并拒绝其他人对**/srv/samba/example/**目录的访问：

   1. 为主用户帐户组禁用自动授予权限：

```
# setfacl -m group::--- /srv/samba/example/
# setfacl -m default:group::--- /srv/samba/example/
```

      目录的主组还被映射到动态**CREATOR GROUP**主体。当您对Samba 共享使用扩展 POSIX ACL 时，主体会被自动添加，您无法将其删除。

   2. 设置目录中的权限：

      a. 对**Domain Admins**组赋予读、写和执行权限：

```
# setfacl -m group:"DOMAIN\Domain Admins":rwx /srv/samba/example/
```

      b. 对**Domain Users**组赋予读和执行权限：

```
# setfacl -m group:"DOMAIN\Domain Users":r-x /srv/samba/example/
```

      c. **other** ACL条目设置权限，以拒绝与其他 ACL 条目不匹配的用户的访问：

```
# setfacl -R -m other::--- /srv/samba/example/
```

   这些设置只适用于这个目录。在 Windows 中，这些 ACL 映射到仅**此文件**夹模式。

3. 要使上一步中设置的权限被在此目录中创建的新文件系统对象继承，请执行以下操作：

```
# setfacl -m default:group:"DOMAIN\Domain Admins":rwx /srv/samba/example/
# setfacl -m default:group:"DOMAIN\Domain Users":r-x /srv/samba/example/
# setfacl -m default:other::--- /srv/samba/example/
```

使用这些设置，现在将主体的仅**此文件**夹模式设置为**此文件**夹、**子文件**夹**和文件**模式。

Samba 将流程中设置的权限映射到以下 Windows ACL:

| 主体 | 权限 | 适用于 |
|---|---|---|
| *domain**Domain*Admins | 全控制 | 这个文件夹、子文件夹和文件 |
| *Domain*\Domain Users | 读和执行 | 这个文件夹、子文件夹和文件 |
| **每个人**[a] | 无 | 这个文件夹、子文件夹和文件 |
| *所有者*（*Unix 用户**所有者*） [b] | 全控制 | 只限于这个文件夹 |
| *primary_group* (*Unix 用户**primary_group*) [c] | 无 | 只限于这个文件夹 |
| **创建者所有者**[d] [e] | 全控制 | 只适用于子文件夹和文件 |
| **创建者组** [e][f] | 无 | 只适用于子文件夹和文件 |

[a] Samba从**othe** ACL 条目映射此主体的权限。

[b] Samba 将目录的所有者映射到此条目。

[c] Samba 将目录的主组群映射到这个条目。

[d] 在新文件系统对象中，创建者会自动继承这个主体的权限。

[e] 在使用 POSIX ACL 的共享中不支持从 ACL 配置或删除这些主体。

[f] 在新文件系统对象中，创建器的主组群自动继承这个主体的权限。

## 1.8. 对使用 POSIX ACL 的共享设置权限

另外，要限制或赋予对 Samba 共享的访问权限，您可以在**/etc/samba/smb.conf**文件的共享部分设置某些参数。

**注意**

如果用户、组或主机能够访问共享，则进行基于共享的权限管理。这些设置不会影响文件系统 ACL。

使用基于共享的设置来限制对共享的访问，例如拒绝特定主机的访问。

**先决条件**

- 与 POSIX ACL 的共享已被设置。

### 1.8.1. 配置基于用户和组群的共享访问权限

基于用户和组的访问控制，使您能够赋予或拒绝特定用户和组对共享的访问权限。

**先决条件**

- 已存在您要设置用户或组群访问的 Samba 共享。

**流程**

1. 例如，要在 **用户帐户** 访问时允许 **Domain Users** 组的所有成员访问共享，请在共享的配置中添加以下参数：

```
valid users = +DOMAIN\"Domain Users"
invalid users = DOMAIN\user
```

**invalid users** 参数的优先级高于**valid users** 参数。例如，如果 **user**帐户是 **Domain Users** 组的成员，则在使用上例时会拒绝此帐户的访问。

2. 重新载入 Samba 配置：

```
# smbcontrol all reload-config
```

**其它资源**

- **smb.conf(5)** man page

### 1.8.2. 配置基于主机的共享访问权限

基于主机的访问控制允许您根据客户端的主机名、IP 地址或 IP 范围授予或拒绝对共享的访问。

以下流程解释了如何启用 **127.0.0.1** IP 地址、**192.0.2.0/24** IP 范围，以及**client1.example.com** 主机来访问共享，另外拒绝了对 **client2.example.com** 主机的访问：

**先决条件**

- 已存在您要设置基于主机的访问的 Samba 共享。

**流程**

1. 在 **/etc/samba/smb.conf** 文件的共享配置中添加以下参数：

```
hosts allow = 127.0.0.1 192.0.2.0/24 client1.example.com
hosts deny = client2.example.com
```

**hosts deny** 参数的优先级高于 **hosts allow**。例如，如果 **client1.example.com** 解析为 **hosts allow** 参数中列出的 IP 地址，那么此主机的访问将被拒绝。

2. 重新载入 Samba 配置：

```
# smbcontrol all reload-config
```

**其它资源**

- **smb.conf(5)** man page

## 1.9. 设置使用 WINDOWS ACL 的共享

Samba 支持在共享和文件系统对象中设置 Windows ACL。这可让您：

- 使用精细 Windows ACL
- 使用 Windows 管理共享权限和文件系统 ACL

或者，您可以将共享配置为使用 POSIX ACL。

详情请参阅 设置使用 POSIX ACL 的 Samba 文件共享 。

这个部分的内容基于 Samba Wiki 中发布的 Setting up a Share Using Windows ACLs 文档。许可证： CC BY 4.0。作者和贡献者：请参阅 Wiki 页面上的历史选项卡。

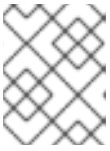
### 1.9.1. 授予 SeDiskOperatorPrivilege 特权

只有被赋予了 **SeDiskOperatorPrivilege** 特权的用户和组才能对使用了 Windows ACL 的共享配置权限。

**流程**

1. 例如，要对 ***DOMAIN*\Domain Admins** 组赋予 **SeDiskOperatorPrivilege** 特权：

```
# net rpc rights grant "DOMAIN\Domain Admins" SeDiskOperatorPrivilege -U
"DOMAIN\administrator"
Enter DOMAIN\administrator's password:
Successfully granted rights.
```



**注意**

在域环境中，对域组赋予 **SeDiskOperatorPrivilege**。这可让您通过更新用户的组成员资格来集中管理特权。

2. 列出所有被赋予了 **SeDiskOperatorPrivilege** 的用户和组：

```
# net rpc rights list privileges SeDiskOperatorPrivilege -U "DOMAIN\administrator"
Enter administrator's password:
SeDiskOperatorPrivilege:
  BUILTIN\Administrators
  DOMAIN\Domain Admins
```

### 1.9.2. 启用 Windows ACL 支持

要配置支持 Windows ACL 的共享，您必须在 Samba 中启用此功能。

**先决条件**

- 在 Samba 服务器中配置了一个用户共享。

**流程**

1. 要全局启用所有共享，请在 **/etc/samba/smb.conf** 文件的 **[global]** 部分添加以下设置：

```
vfs objects = acl_xattr
map acl inherit = yes
store dos attributes = yes
```

   或者，您可以通过将相同的参数添加到共享部分来启用对单个共享的 Windows ACL 支持。

2. 重启**smb**服务：

```
# systemctl restart smb
```

### 1.9.3. 添加使用 Windows ACL 的共享

这部分描述了如何创建名为 **example** 的共享，其共享了 **/srv/samba/example/** 目录的内容，并使用了 Windows ACL。

**流程**

1. 如果不存在，请创建文件夹。例如：

```
# mkdir -p /srv/samba/example/
```

2. 如果您在**enforcing**模式下运行 SELinux，请在目录中设置**samba_share_t**上下文：

```
# semanage fcontext -a -t samba_share_t "/srv/samba/example(/.*)?"
# restorecon -Rv /srv/samba/example/
```

3. 将示例共享添加到**/etc/samba/smb.conf**文件中。例如，添加启用了共享的写操作：

```
[example]
 path = /srv/samba/example/
 read only = no
```

**注意**

无论文件系统 ACL 是什么；如果您没有设置**read only = no**，Samba 会以只读模式共享该目录。

4. 如果您没有在**[global]** 部分中对所有共享启用 Windows ACL 支持，那么请在 **[example]** 部分中添加以下参数来为这个共享启用此特性：

```
vfs objects = acl_xattr
map acl inherit = yes
store dos attributes = yes
```

5. 验证**/etc/samba/smb.conf**文件：

```
# testparm
```

6. 打开所需的端口，并使用**firewall-cmd**工具重新加载防火墙配置：

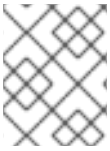
```
# firewall-cmd --permanent --add-service=samba
# firewall-cmd --reload
```

7. 重启**smb**服务：

```
# systemctl restart smb
```

### 1.9.4. 管理使用 Windows ACL 的共享的共享权限和文件系统 ACL

要在使用 Windows ACL 的 Samba 共享上管理共享权限和文件系统 ACL，请使用 Windows 应用程序，如计**算机管理**。详情请查看 Windows 文档。或者，使用 **smbcacls** 工具来管理 ACL。



**注意**

要从 Windows 修改文件系统权限，您必须使用赋予了 **SeDiskOperatorPrivilege** 特权的帐户。

**其它资源**

- 使用 smbcacls 在 SMB 共享中管理 ACL
- Grating SeDiskOperatorPrivilege 权限

## 1.10. 使用 SMBCACLS 在 SMB 共享中管理 ACL

**smbcacls** 工具可以列出、设置和删除存储在 SMB 共享中的文件和目录的 ACL。您可以使用 **smbcacls** 来管理文件系统 ACL：

- 在使用高级 Windows ACL 或 POSIX ACL 的本地或远程 Samba 服务器中
- 在 Red Hat Enterprise Linux 上，远程管理在 Windows 上托管的共享的 ACL

### 1.10.1. 访问控制条目

文件系统对象的每个 ACL 条目都包含以下格式的访问控制条目(ACE)：

```
security_principal:access_right/inheritance_information/permissions
```

**例 1.3. 访问控制条目**

如果 **AD\Domain Users** 组对Windows 上的**此文件夹、子文件夹和文件**拥有**修改**权限，那么ACL 将包含以下 ACE：

```
AD\Domain Users:ALLOWED/OI|CI/CHANGE
```

ACE 包含以下部分：

**安全主体**

安全主体是 ACL 中权限的用户、组群或 SID。

**访问权利**

定义是否赋予或拒绝了对对象的访问权限。该值可以是 **ALLOWED** 或 **DENIED**。

**继承信息**

存在以下值：

**表 1.1. 继承设置**

| 值 | 描述 | 映射到 |
|---|---|---|
| **OI** | 对象实例 | 这个文件夹和文件 |
| **CI** | 容器继承 | 这个文件夹和子文件夹 |
| **IO** | 只继承 | ACE 不适用于当前文件或目录 |
| **ID** | 继承 | ACE 从父目录中继承 |

另外，这些值可以合并如下：

**表 1.2. 继承设置组合**

| 值组合 | 映射到 Windows 应用于 设置 |
|---|---|
| **OI\|CI** | 这个文件夹、子文件夹和文件 |
| **OI\|CI\|IO** | 只适用于子文件夹和文件 |
| **CI\|IO** | 只使用子文件夹 |
| **OI\|IO** | 仅限文件 |

**权限**

这个值可以是代表一个或多个 Windows 权限的十六进制值，也可以是一个 **smbcacls** 别名：

- 代表一个或多个 Windows 权限的十六进制值。
  下表以十六进制格式显示了高级 Windows 权限及其对应的值：

  **表 1.3. 十六进制格式的Windows权限及其相应的smbcacls值**

| Windows 权限 | 十六进制值 |
|---|---|
| 全控制 | **0x001F01FF** |

| Windows 权限 | 十六进制值 |
| --- | --- |
| 遍历文件夹 / 执行文件 | **0x00100020** |
| 列出文件夹 / 读数据 | **0x00100001** |
| 读取属性 | **0x00100080** |
| 读取扩展属性 | **0x00100008** |
| 创建文件 / 写数据 | **0x00100002** |
| 创建文件夹/附加数据 | **0x00100004** |
| 写入属性 | **0x00100100** |
| 写扩展属性 | **0x00100010** |
| 删除子文件夹和文件 | **0x00100040** |
| 删除 | **0x00110000** |
| 读取权限 | **0x00120000** |
| 更改权限 | **0x00140000** |
| 获取所有权 | **0x00180000** |

可以使用位 **OR** 操作将多个权限组合为一个十六进制值。

详情请参阅 ACE 掩码计算。

- **smbcacls** 别名。下表显示了可用的别名：

  **表 1.4. 现有 smbcacls 别名及其对应的 Windows 权限**

| smbcacls 别名 | 映射至 Windows 权限 |
| --- | --- |
| **R** | 读 |
| **READ** | 读和执行 |

| smbcacls 别名 | 映射至 Windows 权限 |
| --- | --- |
| **W** | 特殊：<br>○ 创建文件 / 写数据<br>○ 创建文件夹/附加数据<br>○ 写入属性<br>○ 写扩展属性<br>○ 读取权限 |
| **D** | 删除 |
| **P** | 更改权限 |
| **O** | 获取所有权 |
| **X** | 遍历 / 执行 |
| **CHANGE** | 修改 |
| **FULL** | 全控制 |

**注意**

设置权限时，您可以组合单例别名。例如，您可以设置 **RD** 来应用 Windows 权限 **读** 和 **删除**。但是，您既不能组合多个非字母别名，也无法组合别名和十六进制值。

## 1.10.2. 使用 smbcacls 显示 ACL

要显示 SMB 共享的 ACL，请使用 **smbcacls** 工具。如果您运行不带任何操作参数的 **smbcacls**，如 **--add**，那么工具会显示文件系统对象的 ACL。

**流程**

例如，列出 **//server/example** 共享的根目录的 ACL：

```
# smbcacls //server/example / -U "DOMAIN\administrator"
Enter DOMAIN\administrator's password:
REVISION:1
CONTROL:SR|PD|DI|DP
OWNER:AD\Administrators
GROUP:AD\Domain Users
ACL:AD\Administrator:ALLOWED/OI|CI/FULL
ACL:AD\Domain Users:ALLOWED/OI|CI/CHANGE
ACL:AD\Domain Guests:ALLOWED/OI|CI/0x00100021
```

命令的输出会显示：

- **REVISION** ：安全描述符的内部 Windows NT ACL 修订版
- **CONTROL** ：安全描述符控制
- **OWNER** ：安全描述符所有者的名称或 SID
- **GROUP** ：安全描述符组的名称或 SID
- **ACL** 条目.详情请参阅 访问控制条目。

### 1.10.3. ACE 掩码计算

在大多数情况下，当添加或更新 ACE 时，您可以使用 Existing **smbcacls** 别名中列出的 smbcacls 别名及其对应的 Windows 权限。

但是，如果您要设置 Windows 权限和对应 smbcacls 值（以十六进制格式 ）中列出的高级 Windows 权限，则必须使用位范围 **OR** 操作来计算正确的值。您可以使用以下 shell 命令计算值：

```
# echo $(printf '0x%X' $(( hex_value_1 | hex_value_2 | ... )))
```

**例 1.4. 计算 ACE 掩码**

您需要设置以下权限：

- 遍历文件夹/执行文件(0x00100020)
- 列出文件夹/读数据(0x00100001)
- 读属性(0x00100080)

要计算上面权限的十六进制值，请输入：

```
# echo $(printf '0x%X' $(( 0x00100020 | 0x00100001 | 0x00100080 )))
0x1000A1
```

设置或更新 ACE 时使用返回的值。

### 1.10.4. 使用 smbcacls 添加、更新和删除 ACL

根据您传递给 **smbcacls** 工具的参数，您可以添加、更新和删除文件或目录的 ACL。

**添加 ACL**
要对 **//server/example** 共享的根添加ACL，该共享将**此文件夹、子文件夹和文件**的 **CHANGE** 权限赋予 **AD\Domain Users** 组：

```
# smbcacls //server/example / -U "DOMAIN\administrator --add ACL:"AD\Domain Users":ALLOWED/OI|CI/CHANGE
```

**更新 ACL**
更新 ACL 与添加新的 ACL 类似。您可以使用 **--modify** 参数和现有的安全主体来覆盖 ACL，以便更新 ACL。如果 **smbcacls** 在 ACL 列表中找到了安全主体，那么工具会更新这些权限。否则，命令会失败并报错：

```
ACL for SID principal_name not found
```

例如，要更新 **AD\Domain Users** 组的权限，并将其设置为对 **此文件夹、子文件夹和文件**的**READ**权限，请执行以下操作：

```
# smbcacls //server/example / -U "DOMAIN\administrator --modify ACL:"AD\Domain
Users":ALLOWED/OI|CI/READ
```

**删除 ACL**
要删除 ACL，请将带有确切ACL的 **--delete** 参数传递给 **smbcacls** 工具。例如：

```
# smbcacls //server/example / -U "DOMAIN\administrator --delete ACL:"AD\Domain
Users":ALLOWED/OI|CI/READ
```

## 1.11. 允许用户在 SAMBA 服务器上共享目录

在 Samba 服务器上，你可以配置用户共享目录,而无需root权限。

### 1.11.1. 启用用户共享功能

在用户可以共享目录之前，管理员必须在 Samba 中启用用户共享。

例如，仅允许本地 **example** 组的成员创建用户共享：

**流程**

1. 如果本地 **example** 组不存在，请创建它：

```
# groupadd example
```

2. 为 Samba 准备目录以存储用户共享定义并正确设置其权限。例如：
   a. 创建目录：

```
# mkdir -p /var/lib/samba/usershares/
```

   b. 为 **example** 组设置写权限：

```
# chgrp example /var/lib/samba/usershares/
# chmod 1770 /var/lib/samba/usershares/
```

   c. 设置粘性位以防止用户重命名或删除此目录中其他用户存储的文件。

3. 编辑 **/etc/samba/smb.conf** 文件，并将以下内容添加到 **[global]** 部分：
   a. 设置您配置用来存储用户共享定义的目录的路径。例如：

```
usershare path = /var/lib/samba/usershares/
```

   b. 设置允许在这个服务器上创建多少个用户共享 Samba。例如：

```
usershare max shares = 100
```

如果您对 **usershare max shares** 参数使用默认值 **0**，则用户共享将被禁用。

c. 另外，还可设置绝对目录路径列表。例如，要配置 Samba 只允许共享 **/data** 和 **/srv** 目录的子目录，请设置：

```
usershare prefix allow list = /data /srv
```

有关您可以设置的更多用户共享相关参数的列表，请参阅 **smb.conf(5)** 手册页中的 **用户共享** 部分。

4. 验证**/etc/samba/smb.conf**文件：

```
# testparm
```

5. 重新载入 Samba 配置：

```
# smbcontrol all reload-config
```

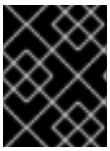
用户现在可以创建用户共享。

### 1.11.2. 添加用户共享

在 Samba 中启用了用户共享功能后，用户可以通过运行 **net usershare add** 命令在 Samba 服务器上共享目录，而无需 **root** 权限。

**net usershare add** 命令的说明：

**net usershare add** share_name path [[ comment ] | [ ACL ]] [ guest_ok=y|n ]



**重要**

如果在创建用户共享时设置了 ACL，您必须在 ACL 之前指定comment参数。要设置空的 comment，请在双引号中使用空字符串。

请注意，如果管理员在 **/etc/samba/smb.conf** 文件的 **[global]** 部分中设置了 **usershare allow guests = yes**，用户只能对用户共享启用 guest访问。

**例 1.5. 添加用户共享**

用户想要在 Samba 服务器上共享 **/srv/samba/** 目录。该共享应命名为 **example**，未设置任何 comment，应该可以被 guest 用户访问。此外，对 **AD\Domain Users** 组的共享权限应设置为可完全访问，对其他用户设置为读权限。要添加此共享，请以用户身份运行：

```
$ net usershare add example /srv/samba/ "" "AD\Domain Users":F,Everyone:R
guest_ok=yes
```

### 1.11.3. 更新用户共享的设置

要更新用户共享的设置，请使用具有相同共享名称和新设置的 **net usershare add** 命令覆盖共享。

请参阅添加用户共享。

### 1.11.4. 显示现有用户共享的信息

用户可以在 Samba 服务器上输入 **net usershare info** 命令，来显示用户共享及其设置。

**先决条件**

- 在 Samba 服务器中配置了一个用户共享。

**流程**

1. 显示任意用户创建的所有用户共享：

```
$ net usershare info -l
[share_1]
path=/srv/samba/
comment=
usershare_acl=Everyone:R,host_name\user:F,
guest_ok=y
...
```

   若要只列出运行命令的用户所创建的共享，请省略 **-l** 参数。

2. 若要只显示关于特定共享的信息，请将共享名称或通配符传给命令。例如，显示名称以 **share_** 开头的共享的信息：

```
$ net usershare info -l share_*
```

### 1.11.5. 列出用户共享

如果您想只列出可用的用户共享，而不列出它们的设置，请使用 **net usershare list** 命令。

**先决条件**

- 在 Samba 服务器中配置了一个用户共享。

**流程**

1. 列出任意用户创建的共享：

```
$ net usershare list -l
share_1
share_2
...
```

   若要只列出运行命令的用户所创建的共享，请省略 **-l** 参数。

2. 若要只列出特定的共享，请将共享名称或通配符传给命令。例如，只列出名称以 **share_** 开头的共享 ：

```
$ net usershare list -l share_*
```

### 1.11.6. 删除用户共享

要删除用户共享，请以创建共享的用户身份或以 **root** 用户身份，使用**net usershare delete** 命令。

**先决条件**

- 在 Samba 服务器中配置了一个用户共享。

**流程**

```
$ net usershare delete share_name
```

## 1.12. 配置共享以允许不进行身份验证的访问

在某些情况下，您想要共享一个用户无需身份验证即可连接到的目录。若要对此进行配置，请对共享启用 guest访问。

**警告**

不需要身份验证的共享可能会造成安全隐患。

### 1.12.1. 启用对共享的客户机访问

如果对共享启用了 guest 访问，Samba 会将guest连接映射到**guest account**参数中设置的操作系统帐户。如果至少满足以下条件之一，Guest用户就可以访问此共享上的文件：

- 该帐户在文件系统 ACL 中列出
- **other** 用户的 POSIX 权限允许这样做

**例 1.6. 客户端共享权限**

如果您将 Samba 配置为将guest帐户映射到 **nobody** （这是默认值），那么以下示例中的 ACL：

- 允许 guest 用户读 **file1.txt**
- 允许 guest 用户读和修改 **file2.txt**
- 防止 guest 用户读或修改 **file3.txt**

```
-rw-r--r--. 1 root       root      1024 1. Sep 10:00 file1.txt
-rw-r-----. 1 nobody     root      1024 1. Sep 10:00 file2.txt
-rw-r-----. 1 root       root      1024 1. Sep 10:00 file3.txt
```

**流程**

1. 编辑 **/etc/samba/smb.conf** 文件：
   a. 如果这是您在这个服务器上设置的第一个客户机共享：

i. 在 **[global]** 部分中设置**map to guest = Bad User** ：

```
[global]
        ...
        map to guest = Bad User
```

使用这个设置，Samba 将拒绝使用错误密码的登录尝试，除非用户名不存在。如果指定的用户名不存在，并且对共享启用了 guest 访问，那么 Samba 会将连接视为guest登录。

ii. 默认情况下，Samba 将 guest 帐户映射到 Red Hat Enterprise Linux 上的 **nobody** 帐户。另外，您也可以设置另外一个帐户。例如：

```
[global]
        ...
        guest account = user_name
```

此参数中设置的帐户必须在 Samba 服务器中本地存在。出于安全考虑，红帽建议使用没有分配有效 shell 的帐户。

b. 在 **[example]** 共享部分中添加 **guest ok = yes** 设置：

```
[example]
        ...
        guest ok = yes
```

2. 验证**/etc/samba/smb.conf**文件：

```
# testparm
```

3. 重新载入 Samba 配置：

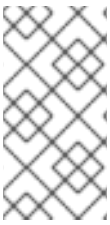
```
# smbcontrol all reload-config
```

## 1.13. 为 MACOS 客户端配置 SAMBA

**fruit** 虚拟文件系统(VFS)Samba 模块提供了与 Apple 服务器消息块(SMB)客户端增强了的兼容性。

### 1.13.1. 优化 Samba 配置，以便为 macOS 客户端提供文件共享

本节描述了如何为托管在服务器上的所有 Samba 共享配置**fruit**模块，以为 macOS 客户端优化 Samba 文件共享。



**注意**

红帽建议全局启用 **fruit** 模块。当客户端建立了到服务器的第一个连接时，使用 macOS 的客户端通过服务器消息块版本 2(SMB2)Apple(AAPL)协议扩展与服务器进行协商。如果客户端第一次连接到未启用 AAPL 扩展的共享，那么客户端不会对服务器的任何共享使用扩展。

**先决条件**

- Samba 配置为文件服务器。

**流程**

1. 编辑 **/etc/samba/smb.conf** 文件，并在 **[global]** 部分启用 **fruit**和**streams_xattr** VFS 模块：

```
vfs objects = fruit streams_xattr
```

**重要**

在启用 **streams_xattr** 之前，您必须启用 **fruit**模块。**fruit** 模块使用备用数据流 (ADS)。因此，您也必须启用 **streams_xattr** 模块。

2. 另外，要对共享提供 macOS Time Machine 支持，请在 **/etc/samba/smb.conf** 文件中的共享配置中添加以下设置：

```
fruit:time machine = yes
```

3. 验证**/etc/samba/smb.conf**文件：

```
# testparm
```

4. 重新载入 Samba 配置：

```
# smbcontrol all reload-config
```

**其它资源**

- **vfs_fruit(8)** 手册页.
- 配置文件共享：
  - 设置使用 POSIX ACL 的 Samba 文件共享
  - 设置使用 Windows ACL 的共享。

## 1.14. 使用 SMBCLIENT 实用程序访问 SMB 共享

smbclient 工具可让您访问 SMB 服务器中的文件共享，类似于命令行 FTP 客户端。例如，您可以使用它来向共享上传文件和从共享下载文件。

**先决条件**

- **samba-client** 软件包已安装。

### 1.14.1. smbclient 互动模式如何工作

例如，使用 ***DOMAIN\user*** 帐户对在 **server** 上托管的**example**共享进行身份验证：

```
# smbclient -U "DOMAIN\user" //server/example
Enter domain\user's password:
Try "help" to get a list of possible commands.
smb: \>
```

在 **smbclient** 成功连接到共享后，工具进入互动模式并显示以下提示：

```
smb: \>
```

要在互动 shell 中显示所有可用命令，请输入：

```
smb: \> help
```

要显示特定命令的帮助信息，请输入：

```
smb: \> help command_name
```

**其它资源**

- **smbclient(1)** 手册页

### 1.14.2. 在互动模式中使用 smbclient

如果您使用不带 **-c** 参数的 **smbclient**，那么工具将进入交互模式。下面的步骤演示了如何连接到 SMB 共享并从子目录中下载文件。

**流程**

1. 连接到共享：

```
# smbclient -U "DOMAIN\user_name" //server_name/share_name
```

2. 进到 **/example/** 目录：

```
smb: \> d /example/
```

3. 列出目录中的文件：

```
smb: \example\> ls
  .                   D        0  Thu Nov 1 10:00:00 2018
  ..                  D        0  Thu Nov 1 10:00:00 2018
  example.txt         N  1048576  Thu Nov 1 10:00:00 2018

        9950208 blocks of size 1024. 8247144 blocks available
```

4. 下载 **example.txt** 文件：

```
smb: \example\> get example.txt
getting file \directory\subdirectory\example.txt of size 1048576 as example.txt (511975,0
KiloBytes/sec) (average 170666,7 KiloBytes/sec)
```

5. 从共享断开：

```
smb: \example\> exit
```

### 1.14.3. 在脚本模式中使用 smbclient

如果将 **-c** 参数传给 **smbclient**，那么你可对远程 SMB 共享自动执行命令。这可让您在脚本中使用 **smbclient**。

下面的步骤演示了如何连接到 SMB 共享并从子目录中下载文件。

**流程**

- 使用以下命令连接到共享，进到 **example** 目录，下载 **example.txt** 文件：

```
# smbclient -U DOMAIN\user_name //server_name/share_name -c "cd /example/ ; get example.txt ; exit"
```

## 1.15. 将 SAMBA 设置为打印服务器

如果您将 Samba 设置为打印服务器，那么网络中的客户端可以使用 Samba 进行打印。此外，如果进行了配置， Windows 客户端可以从 Samba 服务器下载驱动程序。

本节的部分内容摘自在 Samba Wiki 中发布的将Samba设置为打印服务器文档。许可证： CC BY 4.0。作者和贡献者：请参阅 Wiki 页面上的历史选项卡。

**先决条件**

Samba 采用以下模式之一设置：

- 独立服务器
- 域成员

### 1.15.1. Samba spoolssd 服务

Samba **spoolssd**是一种集成到 **smbd** 服务中的服务。在 Samba 配置中启用**spoolssd**，可以显著提高具有大量作业或打印机的打印服务器的性能。

如果没有 **spoolsd**，Samba 就会对 **smbd** 进程进行分叉，并为每个打印作业初始化 **printcap** 缓存。如果打印机数量很大，**smbd** 可能会在缓存初始化过程中有几秒钟没有响应。**spoolssd**服务允许您启动预分叉的**smbd**进程，这些进程在处理打印作业时不会出现任何延迟。主 **spoolsd smbd** 进程使用较少的内存，分叉并终止子进程。

以下流程解释了如何启用 **spoolsd** 服务。

**流程**

1. 编辑 **/etc/samba/smb.conf** 文件中的 **[global]** 部分：
   a. 添加以下参数：

```
rpc_server:spoolss = external
rpc_daemon:spoolssd = fork
```

   b. 另外，您可以设置以下参数：

| 参数 | 默认 | 描述 |
| --- | --- | --- |
| **spoolssd:prefork_min_children** | 5 | 最小子进程数量 |

| 参数 | 默认 | 描述 |
| --- | --- | --- |
| **spoolssd:prefork_max_children** | 25 | 子进程的最大数量 |
| **spoolssd:prefork_spawn_rate** | 5 | Samba 将此参数中设置的新子进程的数量进行分叉，最多为 **spoolsd:prefork_max_children** 中设置的值（如果新的连接已建立） |
| **spoolssd:prefork_max_allowed_clients** | 100 | 子进程服务的客户端数 |
| **spoolssd:prefork_child_min_life** | 60 | 子进程的最低生命周期（以秒为单位）。60 秒是最小的。 |

2. 验证**/etc/samba/smb.conf**文件：

```
# testparm
```

3. 重启**smb**服务：

```
# systemctl restart smb
```

重启服务后，Samba 会自动启动 **smbd** 子进程：

```
# ps axf
...
30903 smbd
30912  \_ smbd
30913      \_ smbd
30914      \_ smbd
30915      \_ smbd
...
```

### 1.15.2. 在 Samba 中启用打印服务器支持

这部分论述了如何在 Samba 中启用打印服务器支持。

**流程**

1. 在 Samba 服务器上，设置 CUPS 并将打印机添加到 CUPS 后端。有关在 CUPS 中配置打印机的详情，请查看打印服务器上的 CUPS web 控制台(https://*print_server_host_name*:631/help)中提供的文档。

**注意**

如果CUPS安装在本地 Samba 打印服务上，Samba 只能将打印作业转发到 CUPS。

2. 编辑 **/etc/samba/smb.conf** 文件：

a. 如果要启用 **spoolsd** 服务，请在 **[global]** 部分中添加以下参数：

```
rpc_server:spoolss = external
rpc_daemon:spoolssd = fork
```

b. 要配置打印后端，请添加 **[printers]** 部分：

```
[printers]
        comment = All Printers
        path = /var/tmp/
        printable = yes
        create mask = 0600
```

**重要**

**[printers]** 共享名称是写死的，不能更改。

3. 验证**/etc/samba/smb.conf**文件：

```
# testparm
```

4. 打开所需的端口，并使用**firewall-cmd**工具重新加载防火墙配置：

```
# firewall-cmd --permanent --add-service=samba
# firewall-cmd --reload
```

5. 重启**smb**服务：

```
# systemctl restart smb
```

重启服务后，Samba 会自动共享在 CUPS 后端中配置的所有打印机。如果想要仅手动共享特定打印机，请参阅 手动共享特定打印机。

### 1.15.3. 手动共享特定的打印机

如果您将 Samba 配置为打印服务器，默认情况下，Samba 会共享在 CUPS 后端中配置的所有打印机。以下流程解释了如何只共享特定的打印机。

**先决条件**

- Samba 被设置为打印服务器

**流程**

1. 编辑 **/etc/samba/smb.conf** 文件：

a. 在 **[global]** 部分中，通过以下设置禁用自动打印机共享：

```
load printers = no
```

b. 为您要共享的每个打印机添加一段。例如，要在 Samba 中将 CUPS 后端中名为 **example** 的打印机共享为 **Example-Printer**，请添加以下部分：

```
[Example-Printer]
        path = /var/tmp/
        printable = yes
        printer name = example
```

您不需要为每个打印机单独设置 spool 目录。您可以在打印机的**path** 参数中设置与您在 **[printers]** 部分中设置的完全相同的 spool 目录。

2. 验证**/etc/samba/smb.conf**文件：

```
# testparm
```

3. 重新载入 Samba 配置：

```
# smbcontrol all reload-config
```

## 1.16. 在 SAMBA 打印服务器中为 WINDOWS 客户端设置自动打印机驱动程序下载

如果您在为 Windows 客户端运行 Samba 打印服务器，你可以上传驱动程序并预配置打印机。如果用户连接到打印机,Windows 会自动在客户端本地下载并安装驱动程序。用户不需要本地管理员权限进行安装。另外，Windows 应用预配置的驱动程序设置，如纸匣的数量。

本节的部分内容摘自 Samba Wiki 上发布的为 Windows 客户端设置自动打印机驱动程序下载文档。许可证： CC BY 4.0。作者和贡献者：请参阅 Wiki 页面上的历史选项卡。

**先决条件**

- Samba 被设置为打印服务器

### 1.16.1. 有关打印机驱动程序的基本信息

本节提供有关打印机驱动程序的一般信息。

**支持的驱动程序模型版本**
Samba 只支持 Windows 2000 及更高版本支持的，以及 Windows Server 2000 及更高版本支持的打印机驱动程序模型版本 3。Samba 不支持 Windows 8 和 Windows Server 2012 中引入的驱动程序模型版本 4。但是，这些及之后的 Windows 版本也支持版本 3 驱动程序。

**包感知驱动程序**
Samba 不支持包感知驱动程序。

**准备上传的打印机驱动程序**
在您将驱动程序上传到 Samba 打印服务器之前：

- 如果驱动程序采用压缩格式提供，请解包它。
- 有些驱动程序需要启动一个设置应用程序，以便在 Windows 主机上本地安装驱动程序。在某些情况下，安装程序会在设置运行期间将单个文件提取到操作系统的临时文件夹中。使用驱动程序文件上传：
  a. 启动安装程序。
  b. 将临时文件夹中的文件复制到新位置。

c. 取消安装。

请您的打印机厂商提供支持上传到打印服务器的驱动程序。

**为客户端提供 32 位和 64 位驱动**
要为 32 位和 64 位 Windows 客户端提供打印机的驱动程序，您必须上传两个架构具有完全相同名称的驱动程序。例如，如果您上传名为**Example PostScript**的 32 位驱动程序和名为**Example PostScript (v1.0)** 的 64 位驱动程序，则名称不匹配。因此，您只能为打印机分配其中一个驱动程序，且该驱动程序无法对这两个架构都适用。

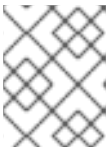
### 1.16.2. 启用用户上传和预配置驱动程序

要上传和预配置打印机驱动程序，用户或组需要被赋予 **SePrintOperatorPrivilege** 特权。用户必须被添加到 **printadmin** 组中。在安装 **samba** 软件包时，Red Hat Enterprise Linux 会自动创建这个组。**printadmin** 组被分配了低于 1000 的最小可用动态系统 GID。

**流程**

1. 例如，要对 **printadmin** 组赋予 **SePrintOperatorPrivilege** 特权：

```
# net rpc rights grant "printadmin" SePrintOperatorPrivilege -U
"DOMAIN\administrator"
Enter DOMAIN\administrator's password:
Successfully granted rights.
```



**注意**

在域环境中，将 **SePrintOperatorPrivilege** 赋予域组。这可让您通过更新用户的组成员资格来集中管理特权。

2. 列出所有被赋予了 **SePrintOperatorPrivilege** 的用户和组：

```
# net rpc rights list privileges SePrintOperatorPrivilege -U "DOMAIN\administrator"
Enter administrator's password:
SePrintOperatorPrivilege:
  BUILTIN\Administrators
  DOMAIN\printadmin
```

### 1.16.3. 设置 print$ 共享

Windows 操作系统从打印服务器上名为 **print$** 的共享中下载打印机驱动程序。这个共享名称在 Windows 中硬编码，无法更改。

以下流程解释了如何将 **/var/lib/samba/drivers/** 目录共享为**print$**，并使本地 **printadmin** 组成员能够上传打印机驱动程序。

**流程**

1. 在 **/etc/samba/smb.conf** 文件中添加 **[print$]** 部分：

```
[print$]
        path = /var/lib/samba/drivers/
        read only = no
```

```
write list = @printadmin
force group = @printadmin
create mask = 0664
directory mask = 2775
```

使用这些设置：

- 只有 **printadmin** 组成员才能将打印机驱动程序上传到共享。
- 新创建的文件和目录的组将被设为 **printadmin**。
- 新文件的权限将被设置为 **664**。
- 新目录的权限将被设置为 **2775**。

2. 要只为所有打印机上传 64 位驱动程序，请在 **/etc/samba/smb.conf** 文件的 **[global]** 部分包含此设置：

```
spoolss: architecture = Windows x64
```

如果没有这个设置，Windows 只显示您上传的至少 32 位版本的驱动程序。

3. 验证**/etc/samba/smb.conf**文件：

```
# testparm
```

4. 重新载入 Samba 配置

```
# smbcontrol all reload-config
```

5. 如果 **printadmin** 组不存在，就创建它：

```
# groupadd printadmin
```

6. 将 **SePrintOperatorPrivilege** 特权赋予 **printadmin** 组。

```
# net rpc rights grant "printadmin" SePrintOperatorPrivilege -U
"DOMAIN\administrator"
Enter DOMAIN\administrator's password:
Successfully granted rights.
```

7. 如果您在**enforcing**模式下运行 SELinux，请在目录中设置**samba_share_t**上下文：

```
# semanage fcontext -a -t samba_share_t "/var/lib/samba/drivers(/.)?" # *restorecon -
Rv /var/lib/samba/drivers/
```

8. 对 **/var/lib/samba/drivers/** 目录设置权限：

- 如果使用 POSIX ACL，请设置：

```
# chgrp -R "printadmin" /var/lib/samba/drivers/
# chmod -R 2775 /var/lib/samba/drivers/
```

- 如果使用 Windows ACL，请设置：

| 主体 | 权限 | 适用于 |
| --- | --- | --- |
| **创建者所有者** | 全控制 | 只适用于子文件夹和文件 |
| **通过身份验证的用户** | 读和执行、列出目录内容、读 | 这个文件夹、子文件夹和文件 |
| **printadmin** | 全控制 | 这个文件夹、子文件夹和文件 |

有关在 Windows 上设置 ACL 的详情，请查看 Windows 文档。

**其它资源**

- 启用用户上传和预配置驱动程序。

### 1.16.4. 创建 GPO 以启用客户端信任 Samba 打印服务器

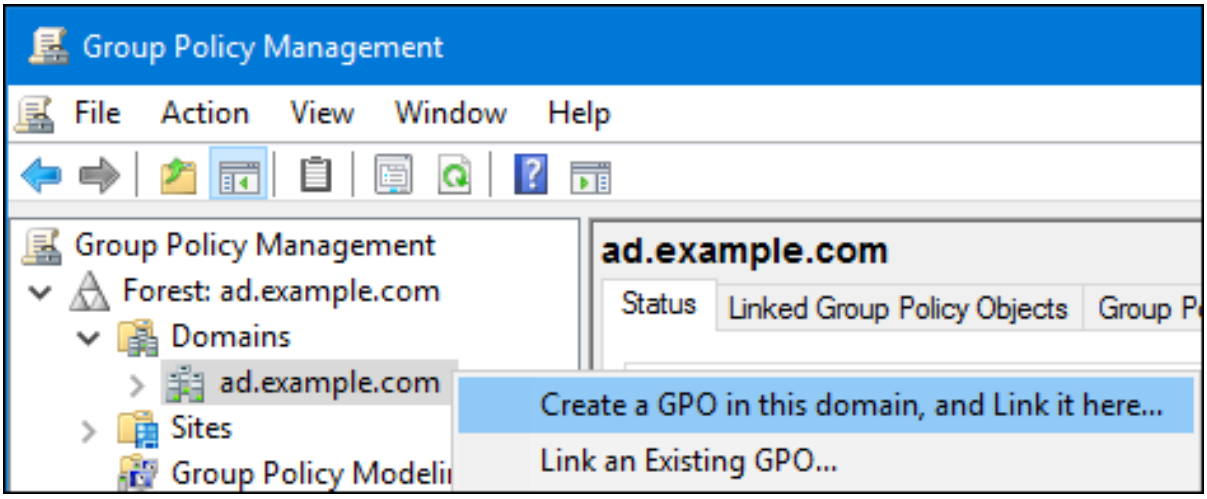
出于安全考虑，最近的 Windows 操作系统会阻止客户端从不受信任的服务器下载非包感知的打印机驱动程序。如果您的打印服务器是 AD 中的成员，您可以在域中创建一个组策略对象(GPO)来信任 Samba 服务器。

**先决条件**

- Samba 打印服务器是 AD 域的成员。
- 您用来创建 GPO 的 Windows 计算机必须安装有 Windows 远程服务器管理工具(RSAT)。详情请查看 Windows 文档。

**流程**

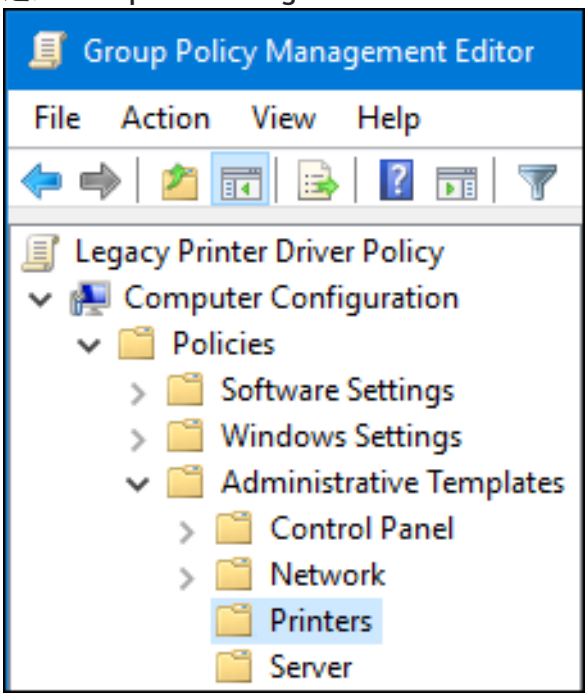
1. 使用允许编辑组策略的帐户（如 AD 域 **Administrator** 用户）登录到 Windows 计算机。
2. 打开 **组策略管理控制台**。
3. 右键单击 AD 域并选择 **创建此域中的 GPO，并链接到此处**。




4. 为 GPO 输入一个名称，如 **Legacy Printer Driver Policy**，并点击 **OK**。新的 GPO 将在域条目下显示。

5. 右键单击新创建的 GPO，然后选择 **编辑** 以打开 组**策略管理编辑器**。

6. 进入 **Computer Configuration → Policies → Administrative Templates → Printers**。

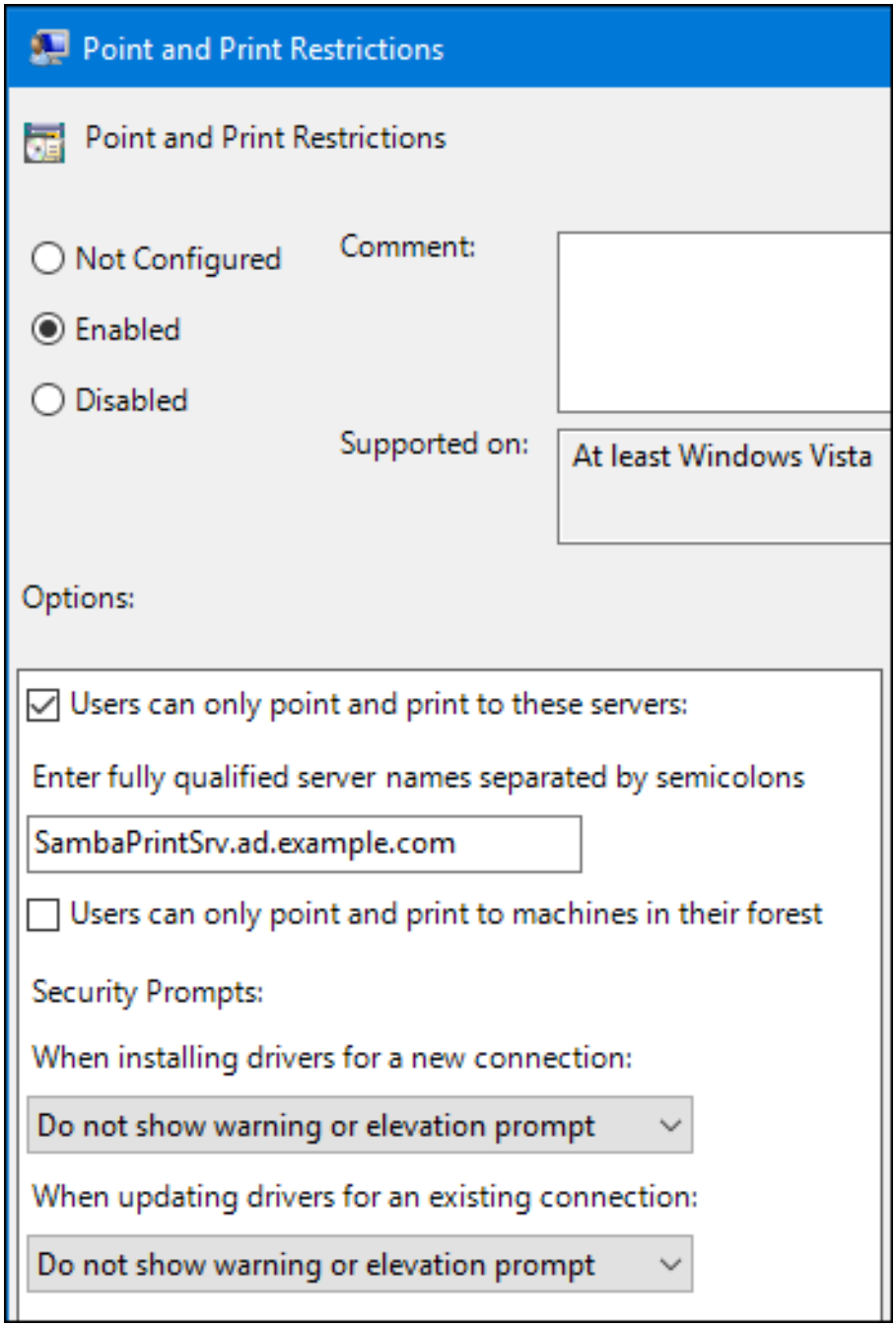



7. 在窗口的右侧，双击 **指向和打印限制** 来编辑策略：

   a. 启用策略并设置以下选项：

      i. 选择 **用户只能指向并打印到这些服务器**，再将 Samba 打印服务器的完全限定域名 (FQDN)输入到此选项旁边的字段。

      ii. 在 **安全提示** 下的两个复选框中，选择 **不显示警告 或 高程提示**。

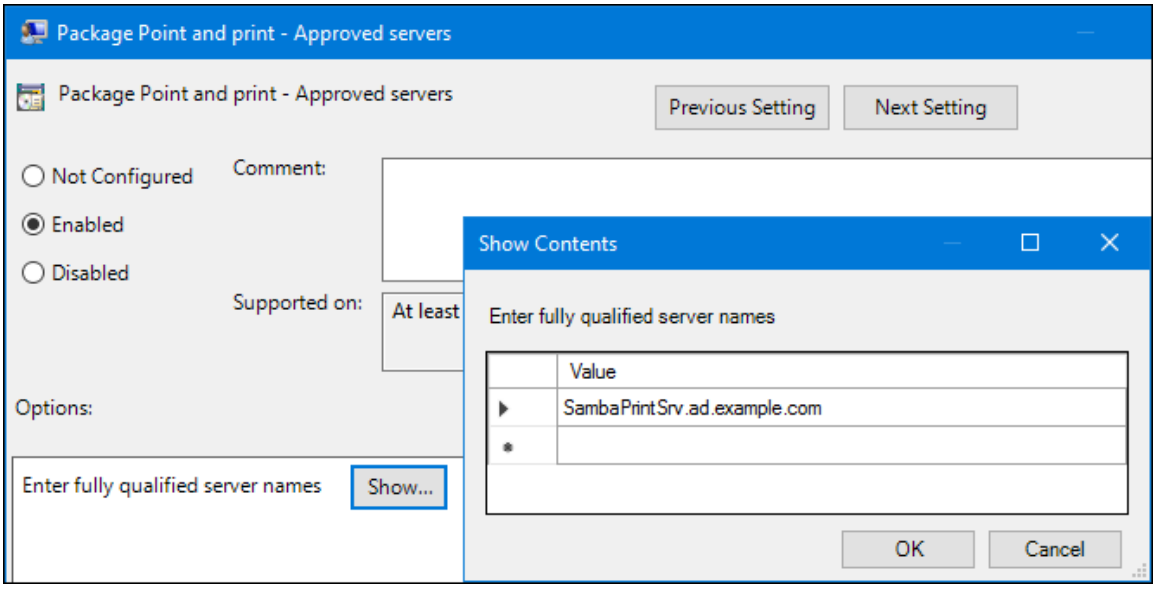



b. 点击确定。

8. 双击 **包指向和打印 - 已批准的服务器** 来编辑策略：

a. 启用策略并单击 **显示** 按钮。

b. 输入 Samba 打印服务器的 FQDN。




c. 单击 **OK**，关闭 **显示内容** 和策略的属性窗口。

9. 关闭 **组策略管理编辑器**。

10. 关闭 **组策略管理控制台**。

在 Windows 域成员应用了组策略后，用户连接到打印机时会自动从 Samba 服务器下载打印机驱动程序。

**其它资源**

- 有关使用组策略，请参阅 Windows 文档。

### 1.16.5. 上传驱动程序和预配置打印机

在 Windows 客户端使用 **打印管理** 应用程序上传托管在 Samba 打印服务器上的驱动程序和预配置打印机。详情请查看 Windows 文档。

## 1.17. 在启用了 FIPS 模式的服务器上运行 SAMBA

本节概述了在启用了 FIPS 模式的情况下运行 Samba 的限制。还提供了在运行 Samba 的 Red Hat Enterprise Linux 主机上启用 FIPS 模式的流程。

### 1.17.1. 在 FIPS 模式中使用 Samba 的限制

在指定条件下，以下 Samba 模式和功能在 FIPS 模式下工作：

- Samba 仅在 Active Directory(AD)或使用AES密码进行Kerberos身份验证的红帽身份管理(IdM)环境中作为域成员。
- Samba 作为 Active Directory 域成员上的文件服务器。但是，这需要客户端使用 Kerberos 向服务器进行身份验证。

由于 FIPS 的安全性增强，如果启用了 FIPS 模式，以下 Samba 特性和模式将无法正常工作：

- NT LAN Manager(NTLM)验证，因为 RC4 密码已被阻止
- 服务器消息块版本 1(SMB1)协议

- 独立文件服务器模式，因为它使用了 NTLM 身份验证
- NT4 风格的域控制器
- NT4 风格的域成员.请注意,红帽继续支持后台使用的主域控制器（PDC）功能 IdM。
- 针对Samba 服务器的密码修改.您只能对 Active Directory 域控制器使用 Kerberos 进行密码修改。

以下特性没有在 FIPS 模式下测试，因此红帽不支持：

- 将 Samba 作为打印服务器来运行

### 1.17.2. 在 FIPS 模式下使用 Samba

这部分描述了如何在运行 Samba 的 RHEL 主机上启用 FIPS 模式。

**先决条件**

- 在 Red Hat Enterprise Linux 主机上配置了Samba 。
- Samba 以 FIPS 模式支持的模式运行。

**流程**

1. 在 RHEL 中启用 FIPS 模式：

```
# fips-mode-setup --enable
```

2. 重启服务器：

```
# reboot
```

3. 使用 **testparm** 工具来验证配置：

```
# testparm -s
```

如果命令显示任何错误或不兼容，请修复它们以确保 Samba 正常工作。

**其它资源**

- 第 1.17.1 节 “在 FIPS 模式中使用 Samba 的限制”

## 1.18. 调整 SAMBA 服务器的性能

本章描述了在某些情况下，什么设置可以提高 Samba 的性能，以及哪些设置可能会对性能造成负面影响。

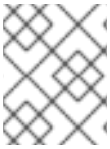
本节的部分内容来自在 Samba Wiki 中发布的 Performance Tuning 文档。许可证： CC BY 4.0。作者和贡献者：请参阅 Wiki 页面上的历史选项卡。

**先决条件**

- Samba 被设置为文件或打印服务器

### 1.18.1. 设置 SMB 协议版本

每个新的 SMB 版本都添加了特性并提高了协议的性能。最新的 Windows 和 Windows 服务器操作系统始终支持最新的协议版本。如果 Samba 也使用最新的协议版本，那么连接到 Samba 的 Windows 客户端将从性能改进中受益。在 Samba 中，server max protocol的默认值被设置为最新支持的稳定的 SMB 协议版本。



**注意**

要始终拥有最新的稳定的 SMB 协议版本，请不要设置 **server max protocol** 参数。如果手动设置参数，则需要修改 SMB 协议的每个新版本的设置，以便启用最新的协议版本。

以下流程解释了如何对 **server max protocol** 参数使用默认值。

**步骤**

1. 从 **/etc/samba/smb.conf** 文件的 **[global]** 部分中删除 **server max protocol** 参数。
2. 重新载入 Samba 配置

```
# smbcontrol all reload-config
```

### 1.18.2. 与包含大量文件的目录调整共享

Linux 支持区分大小写的文件名。因此，在搜索或访问文件时，Samba需要针对大小写文件名来扫描目录。您可以将共享配置为只以小写或大写来创建新文件，这可以提高性能。

**先决条件**

- Samba 配置为文件服务器

**步骤**

1. 将共享上的所有文件重命名为小写。

**注意**

使用这个过程中的设置，名称不为小写的文件将不再显示。

2. 在共享部分中设置以下参数：

```
case sensitive = true
default case = lower
preserve case = no
short preserve case = no
```

有关参数的详情，请查看 **smb.conf(5)** 手册页 中的描述。

3. 验证**/etc/samba/smb.conf**文件：

```
# testparm
```

4. 重新载入 Samba 配置：

```
# smbcontrol all reload-config
```

应用了这些设置后，此共享上所有新创建的文件的名称都使用小写。由于这些设置，Samba 不再需要针对大小写来扫描目录，这样可以提高性能。

### 1.18.3. 可能会对性能造成负面影响的设置

默认情况下，Red Hat Enterprise Linux 中的内核会根据高网络性能进行了微调。例如，内核对缓冲区大小使用自动轮询机制。在 **/etc/samba/smb.conf** 文件中设置 **socket options** 参数会覆盖这些内核设置。因此，设置此参数会在大多数情况下降低 Samba 网络性能。

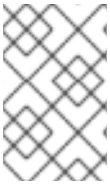
要使用内核的优化的设置，请从 **/etc/samba/smb.conf** 中的 **[global]** 部分删除 **socket options** 参数。

## 1.19. 将 SAMBA 配置为与需要 SMB 版本低于默认版本的客户端兼容

Samba 对它支持的最小服务器消息块(SMB)版本使用合理的安全默认值。但是，如果您的客户端需要较旧的 SMB 版本，您可以配置 Samba 来支持它。

### 1.19.1. 设置 Samba 服务器支持的最小 SMB 协议版本

在 Samba 中，**/etc/samba/smb.conf** 文件中的 **server min protocol** 参数定义了 Samba 服务器支持的最小服务器消息块(SMB)协议版本。这部分论述了如何更改最小 SMB 协议版本。



**注意**

默认情况下，RHEL 8.2 及之后版本上的 Samba 只支持 SMB2 和更新的协议版本。红帽建议不要使用已弃用的 SMB1 协议。但是，如果您的环境需要 SMB1，您可以手动将 **server min protocol** 参数设置为 **NT1** 来重新启用 SMB1。

**先决条件**

- 已安装并配置 Samba。

**流程**

1. 编辑 **/etc/samba/smb.conf** 文件，添加 **server min protocol** 参数，并将参数设置为服务器应支持的最小 SMB 协议版本。例如，要将 SMB 协议的最小版本设置为 **SMB3**，请添加：

```
server min protocol = SMB3
```

2. 重启**smb**服务：

```
# systemctl restart smb
```

**其它资源**

- **smb.conf(5)** man page

## 1.20. 经常使用 SAMBA 命令行工具

本章论述了使用 Samba 服务器时经常使用的命令。

### 1.20.1. 使用 net ads join 和 net rpc join 命令

使用 **net** 工具的 **join** 子命令，您可以将 Samba 加入到 AD 或 NT4 域。要加入域，您必须手动创建 **/etc/samba/smb.conf** 文件，并有选择地更新其他配置，如 PAM。

**重要**

红帽建议使用 **realm** 工具来加入域。**realm** 工具自动更新所有涉及的配置文件。

**流程**

1. 使用以下设置手动创建 **/etc/samba/smb.conf** 文件：
   - 对于 AD 域成员：

     ```
     [global]
     workgroup = domain_name
     security = ads
     passdb backend = tdbsam
     realm = AD_REALM
     ```

   - 对于 NT4 域成员：

     ```
     [global]
     workgroup = domain_name
     security = user
     passdb backend = tdbsam
     ```

2. 为 * 默认域和要加入到 **/etc/samba/smb.conf** 文件中 **[global]** 部分的域添加 ID 映射配置。
3. 验证**/etc/samba/smb.conf**文件：

   ```
   # testparm
   ```

4. 以域管理员身份加入域：
   - 加入 AD 域：

     ```
     # net ads join -U "DOMAIN\administrator"
     ```

   - 要加入 NT4 域：

     ```
     # net rpc join -U "DOMAIN\administrator"
     ```

5. 将 **winbind** 源追加到 **/etc/nsswitch.conf** 文件中的 **passwd** 和 **group** 数据库条目中：

   ```
   passwd:     files winbind
   group:      files winbind
   ```

6. 启用并启动 **winbind** 服务：

   ```
   # systemctl enable --now winbind
   ```

7. （可选）使用 **authselect** 工具来配置 PAM。
   详情请查看 **authselect(8)** 手册页。

8. 另外，对于 AD 环境，配置 Kerberos 客户端。
   详情请查看您的 Kerberos 客户端文档。

**其它资源**

- 将 Samba 加入到域 中。
- 了解并配置 Samba ID 映射。

### 1.20.2. 使用 net rpc right 命令

在 Windows 中，您可以为帐户和组分配特权来执行特殊操作，如对共享设置 ACL 或上传打印机驱动程序。在 Samba 服务器上，您可以使用 **net rpc permissions** 命令来管理特权。

**列出您可以设置的权限**
若要列出所有可用的特权及其所有者，可使用 **net rpc permissions list** 命令。例如：

```
# net rpc rights list -U "DOMAIN\administrator"
Enter DOMAIN\administrator's password:
    SeMachineAccountPrivilege  Add machines to domain
     SeTakeOwnershipPrivilege  Take ownership of files or other objects
           SeBackupPrivilege  Back up files and directories
          SeRestorePrivilege  Restore files and directories
    SeRemoteShutdownPrivilege  Force shutdown from a remote system
     SePrintOperatorPrivilege  Manage printers
        SeAddUsersPrivilege  Add users and groups to the domain
      SeDiskOperatorPrivilege  Manage disk shares
        SeSecurityPrivilege  System security
```

**授予权限**
若要为帐户或组赋予特权，可使用 **net rpc rights grant** 命令。

例如，将**SePrintOperatorPrivilege** 特权赋予 ***DOMAIN*\printadmin** 组 ：

```
# net rpc rights grant "DOMAIN\printadmin" SePrintOperatorPrivilege -U
"DOMAIN\administrator"
Enter DOMAIN\administrator's password:
Successfully granted rights.
```

**撤销权限**
若要从帐户或组撤销特权，可使用 **net rpc rights revoke** 命令。

例如，要对***DOMAIN*\printadmin** 组撤销 **SePrintOperatorPrivilege** 特权：

```
# net rpc rights remoke "DOMAIN\printadmin" SePrintOperatorPrivilege -U
"DOMAIN\administrator"
Enter DOMAIN\administrator's password:
Successfully revoked rights.
```

### 1.20.3. 使用 net rpc share 命令

**net rpc share** 命令提供了在本地或远程 Samba 或 Windows 服务器上列出、添加和删除共享的功能。

**列出共享**
若要列出 SMB 服务器上的共享，请使用 **net rpc share list** 命令。（可选）将 **-S *server_name*** 参数传给命令，以列出远程服务器的共享。例如：

```
# net rpc share list -U "DOMAIN\administrator" -S server_name
Enter DOMAIN\administrator's password:
IPC$
share_1
share_2
...
```

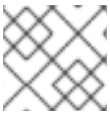
**注意**

在 **/etc/samba/smb.conf** 文件中设置了 **browseable = no** 的、托管在 Samba 服务器上的共享不会显示在输出中。

**添加共享**
**net rpc share add** 命令允许您向 SMB 服务器添加共享。

例如，要在共享 **C:\example** 目录的远程 Windows 服务器中添加一个名为 **example** 的共享：

```
# net rpc share add example="C:\example" -U "DOMAIN\administrator" -S server_name
```



**注意**

在指定 Windows 目录名称时，您必须省略路径中的结尾反斜杠。

使用命令在 Samba 服务器中添加共享：

- 在 **-U** 参数中指定的用户必须拥有在目标服务器上赋予了 **SeDiskOperatorPrivilege** 的特权。
- 您必须编写一个脚本，其在 **/etc/samba/smb.conf** 文件中添加共享部分，并重新加载 Samba 。该脚本必须在 **/etc/samba/smb.conf** 的 **[global]** 部分中的 **add share command** 参数中设置。详情请查看 **smb.conf(5)** 手册页中的 **add share command** 描述。

**删除共享**
**net rpc share delete** 命令允许您从 SMB 服务器中删除共享。

例如，要从远程 Windows 服务器中删除名为 example 的共享：

```
# net rpc share delete example -U "DOMAIN\administrator" -S server_name
```

使用命令从 Samba 服务器中删除共享：

- 在 **-U** 参数中指定的用户必须被赋予了 **SeDiskOperatorPrivilege** 特权。
- 您必须编写一个脚本，其从 **/etc/samba/smb.conf** 文件中删除共享的部分，并重新加载 Samba 。该脚本必须在 **/etc/samba/smb.conf** 的 **[global]** 部分中的 **delete share command** 参数中设置。详情请查看 **smb.conf(5)** 手册页中的 **delete share command** 描述。

### 1.20.4. 使用 net user 命令

**net user** 命令允许您在 AD DC 或 NT4 PDC 中执行以下操作：

- 列出所有用户帐户
- 添加用户
- 删除用户

**注意**

只有在列出域用户帐户时，才需要指定连接方法，如 AD 域的**ads** 或 NT4 域的 **rpc** 。其他用户相关的子命令可以自动探测连接方法。

将 **-U *user_name*** 参数传给命令，以指定允许执行所请求的操作的用户。

**列出域用户帐户**
列出 AD 域中的所有用户：

```
# net ads user -U "DOMAIN\administrator"
```

列出 NT4 域中的所有用户：

```
# net rpc user -U "DOMAIN\administrator"
```

**在域中添加用户帐户**
在 Samba 域成员中，您可以使用 **net user add** 命令将用户帐户添加到域。

例如，将 **user** 账户添加到域：

1. 添加帐户：

```
# net user add user password -U "DOMAIN\administrator"
User user added
```

2. （可选）使用远程过程调用(RPC)shell 来启用 AD DC 或 NT4 PDC 中的帐户。例如：

```
# net rpc shell -U DOMAIN\administrator -S DC_or_PDC_name
Talking to domain DOMAIN (S-1-5-21-1424831554-512457234-5642315751)

net rpc> user edit disabled user: no
Set user's disabled flag from [yes] to [no]

net rpc> exit
```

**从域中删除用户帐户**
对于 Samba 域成员，您可以使用 **net user delete** 命令从域中删除用户帐户。

例如，从域中删除 **user** 账户：

```
# net user delete user -U "DOMAIN\administrator"
User user deleted
```

### 1.20.5. 使用 rpcclient 工具

The **rpcclient** 工具可让您在本地或远程 SMB 服务器上手动执行客户端 Microsoft 远程过程调用(MS-RPC)功能。但是，大部分特性都已集成到 Samba 提供的单独工具中。使用 **rpcclient** 只用于测试 MS-PRC 功能。

**先决条件**

- **samba-client** 软件包已安装。

**例子**
例如，您可以使用 **rpcclient** 工具来：

- 管理打印机假脱机子系统(SPOOLSS)。

  **例 1.7. 将驱动程序分配给打印机**

  ```
  # rpcclient server_name -U "DOMAIN\administrator" -c 'setdriver "printer_name" "driver_name"'
  Enter DOMAIN\administrators password:
  Successfully set printer_name to driver driver_name.
  ```

- 检索有关 SMB 服务器的信息。

  **例 1.8. 列出所有文件共享和共享的打印机**

  ```
  # rpcclient server_name -U "DOMAIN\administrator" -c 'netshareenum'
  Enter DOMAIN\administrators password:
  netname: Example_Share
   remark:
   path:   C:\srv\samba\example_share\
   password:
  netname: Example_Printer
   remark:
   path:   C:\var\spool\samba\
   password:
  ```

- 使用安全帐户管理器远程(SAMR)协议来执行操作。

  **例 1.9. 在 SMB 服务器中列出用户**

  ```
  # rpcclient server_name -U "DOMAIN\administrator" -c 'enumdomusers'
  Enter DOMAIN\administrators password:
  user:[user1] rid:[0x3e8]
  user:[user2] rid:[0x3e9]
  ```

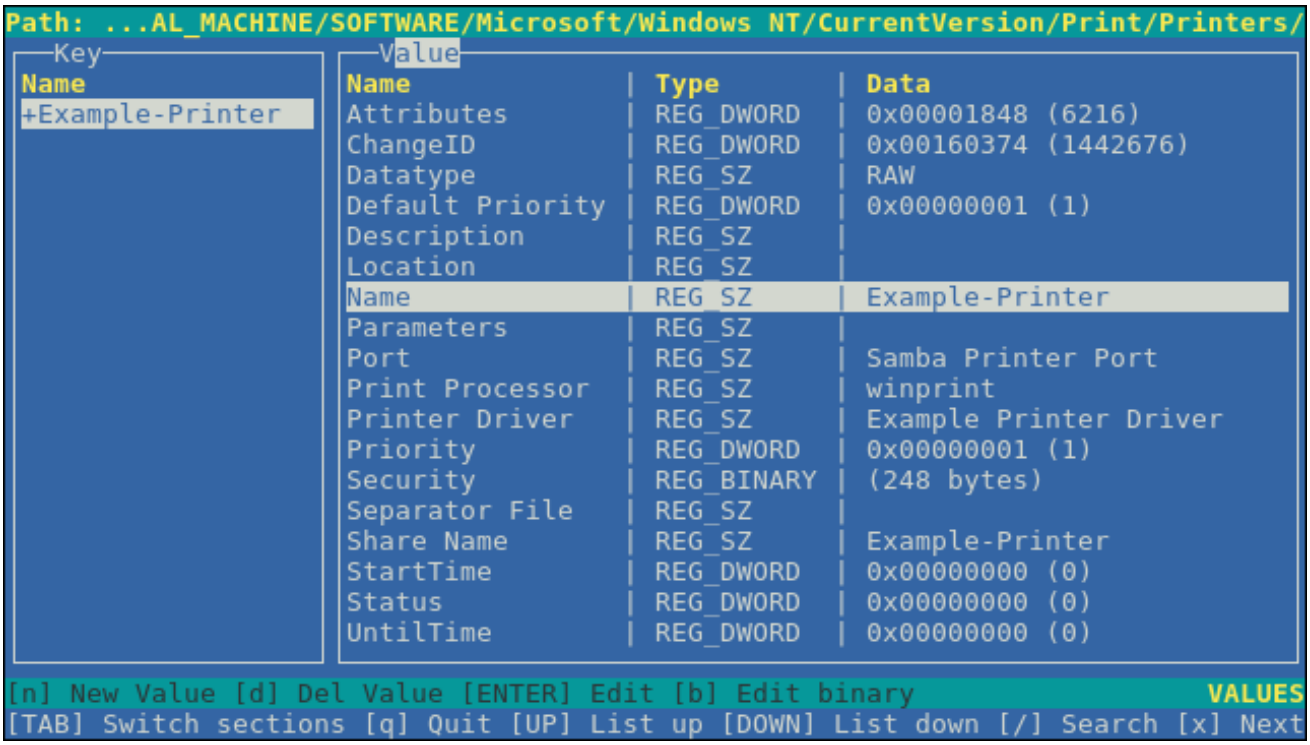
  如果您针对独立服务器或域成员运行命令，它将列出本地数据库中的用户。针对 AD DC 或 NT4 PDC 运行命令列出域用户。

**其它资源**

- **rpcclient(1)** man page

### 1.20.6. 使用 samba-regedit 应用程序

某些设置（如打印机配置）存储在 Samba 服务器上的注册表中。您可以使用基于 ncurses 的 **samba-regedit** 应用程序来编辑 Samba 服务器的注册表。




**先决条件**

- **samba-client** 软件包已安装。

**流程**

要启动应用程序，请输入：

```
# samba-regedit
```

使用以下键：

- 上键和下键：在注册表树和值中进行导航。
- **Enter**：打开关键字或编辑值。
- 选项卡 ：在 **Key** 和 **Value** 窗格间切换。
- **Ctrl+C**：关闭应用程序。

### 1.20.7. 使用 smbcontrol 工具

**smbcontrol** 工具允许您向 **smbd**、**nmbd**、**winbindd 或** 所有这些服务发送命令消息。这些控制消息指示服务重新载入其配置。

本节中的流程演示了如何通过将 **reload-config** 消息类型发送到 **所有** 目的地来重新加载 **smbd** 、**nmbd**、**winbindd** 服务的配置。

**先决条件**

- **samba-common-tools** 软件包已安装。

**流程**

```
# smbcontrol all reload-config
```

**其它资源**

- **smbcontrol(1)** man page

### 1.20.8. 使用 smbpasswd 工具

**smbpasswd** 工具管理本地 Samba 数据库中的用户帐户和密码。

**先决条件**

- **samba-common-tools** 软件包已安装。

**流程**

1. 如果您以用户身份运行命令，**smbpasswd** 将修改运行命令的用户的 Samba 密码。例如：

```
[user@server ~]$ smbpasswd
New SMB password: password
Retype new SMB password: password
```

2. 如果以 **root** 用户身份运行 **smbpasswd**，例如，您可以使用该工具来：

   - 创建一个新用户：

```
[root@server ~]# smbpasswd -a user_name
New SMB password: password
Retype new SMB password: password
Added user user_name.
```

**注意**

在将用户添加到 Samba 数据库之前，您必须先在本地操作系统中创建帐户。有关配置基本系统设置指南，请参阅 使用命令行的 Adding a new user 部分。

   - 启用 Samba 用户：

```
[root@server ~]# smbpasswd -e user_name
Enabled user user_name.
```

   - 禁用 Samba 用户：

```
[root@server ~]# smbpasswd -x user_name
Disabled user user_name
```

   - 删除用户：

```
[root@server ~]# smbpasswd -x user_name
Deleted user user_name.
```

**其它资源**

- **smbpasswd(8)** 手册页

### 1.20.9. 使用 smbstatus 工具

**smbstatus** 工具报告，关于：

- 每个 **smbd** 守护进程的每个 PID 到 Samba 服务器的连接。此报告包括用户名、主组群、SMB 协议版本、加密和签名信息。
- 每个 Samba 共享的连接。此报告包括 **smbd** 守护进程的 PID、连接机器的 IP、连接建立的时间戳、加密和签名信息。
- 锁定文件列表。报告条目包括更多详情，如 Opportunistic lock(oplock)类型

**先决条件**

- **samba** 软件包已安装。
- **smbd** 服务在运行。

**流程**

```
# smbstatus

Samba version 4.15.2
PID  Username              Group             Machine                       Protocol Version  Encryption
Signing
----------------------------------------------------------------------------------------------------------------
-
963  DOMAIN\administrator  DOMAIN\domain users  client-pc  (ipv4:192.0.2.1:57786)  SMB3_02
-           AES-128-CMAC

Service  pid  Machine    Connected at                Encryption  Signing:
---------------------------------------------------------------------
example  969  192.0.2.1  Thu Nov  1 10:00:00 2018 CEST  -           AES-128-CMAC

Locked files:
Pid  Uid    DenyMode   Access    R/W     Oplock      SharePath          Name      Time
--------------------------------------------------------------------------------------------------
969  10000  DENY_WRITE 0x120089  RDONLY  LEASE(RWH)  /srv/samba/example  file.txt  Thu
Nov  1 10:00:00 2018
```

**其它资源**

- **smbstatus(1)** man page

### 1.20.10. 使用 smbtar 工具

**smbtar** 工具备份 SMB 共享的内容或其子目录，并将内容存储在 **tar** 存档中。或者，您可以将内容写入磁带设备。

**先决条件**

- **samba-client** 软件包已安装。

**流程**

- 使用以下命令备份 **//server/example/** 共享中 **demo** 目录的内容，并将内容存储在 **/root/example.tar** 归档中：

```
# smbtar -s server -x example -u user_name -p password -t /root/example.tar
```

**其它资源**

- **smbtar(1)** 手册页

### 1.20.11. 使用 wbinfo 工具

**wbinfo** 工具查询并返回 **winbindd** 服务创建和使用的信息。

**先决条件**

- **samba-winbind-clients** 软件包已安装。

**流程**

例如，您可以使用 **wbinfo** 来：

- 列出域用户：

```
# wbinfo -u
AD\administrator
AD\guest
...
```

- 列出域组：

```
# wbinfo -g
AD\domain computers
AD\domain admins
AD\domain users
...
```

- 显示用户的 SID：

```
# wbinfo --name-to-sid="AD\administrator"
S-1-5-21-1762709870-351891212-3141221786-500 SID_USER (1)
```

- 显示域和信任的信息：

```
# wbinfo --trusted-domains --verbose
Domain Name   DNS Domain          Trust Type  Transitive  In   Out
BUILTIN                           None        Yes         Yes  Yes
server                            None        Yes         Yes  Yes
DOMAIN1       domain1.example.com  None       Yes         Yes  Yes
DOMAIN2       domain2.example.com  External   No          Yes  Yes
```

**其它资源**

- **wbinfo(1)** man page

## 1.21. 其它资源

- Red Hat Samba 软件包包括所有 Samba 命令的说明页以及安装该软件包的配置文件。例如，显示 **/etc/samba/smb.conf** 文件的手册页，该手册页解释了你可以在此文件中设置的所有配置参数：

```
# man smb.conf
```

- **/usr/share/docs/samba-version/** 目录包含由 Samba 项目提供的常规文档、示例脚本和 LDAP 模式文件。

- 红帽集群存储管理指南 ：提供关于设置 Samba 和集群普通数据库(CDTB)以共享存储在 GlusterFS 卷上的目录的信息。

- 在 Red Hat Enterprise Linux 中挂载 SMB 共享 。

# 第 2 章 导出 NFS 共享

作为系统管理员，您可以使用 NFS 服务器来通过网络共享系统上的目录。

## 2.1. NFS 简介

这部分解释了 NFS 服务的基本概念。

网络文件系统(NFS)允许远程主机通过网络挂载文件系统，并像它们挂载在本地那样与这些文件系统进行交互。这可让您将资源整合到网络的集中服务器中。

NFS 服务器参考 **/etc/exports** 配置文件，来确定是否允许客户端访问任何导出的文件系统。一旦被验证，所有文件和目录操作都对用户有效。

## 2.2. 支持的 NFS 版本

这部分列出了 Red Hat Enterprise Linux 支持 NFS 版本及其特性。

目前，Red Hat Enterprise Linux 9 支持以下 NFS 主版本：

- 与 NFSv2 相比，NFS 版本 3（NFSv3）支持安全异步写入操作，并在处理错误时更可靠。它也支持 64 位文件大小和偏移，允许客户端访问超过 2 GB 文件数据。
- NFS 版本 4(NFSv4)通过防火墙，并在 Internet 上工作，不再需要 **rpcbind** 服务，支持访问控制列表(ACL)，并且使用有状态操作。

红帽不再支持 NFS 版本 2(NFSv2)。

### 默认 NFS 版本

Red Hat Enterprise Linux 9 中的默认 NFS 版本为 4.2。NFS 客户端默认试图使用 NFSv4.2 挂载，并在服务器不支持 NFSv4.2 时回退到 NFSv4.1。之后挂载会返回 NFSv4.0，然后回退到 NFSv3。

### 次要 NFS 版本的特性

以下是 Red Hat Enterprise Linux 9 中的 NFSv4.2 的功能：

**服务器端复制**

使用 **copy_file_range()** 系统调用，可以使 NFS 客户端高效地复制数据，而不浪费网络资源。

**稀疏文件**

使文件有一个或者多个 *洞（hole）*，它们是不分配或者未初始化的数据块，只由 0 组成。NFSv4.2 中的 **lseek()** 操作支持 **seek_hole()** 和 **seek_data()**，这使得应用程序能够在稀疏文件中映射漏洞的位置。

**保留空间**

允许存储服务器保留空闲空间，这会防止服务器耗尽空间。NFSv4.2 支持 **allocate()** 操作来保留空间，支持 **deallocate()** 操作来释放空间，支持 **fallocate()** 操作来预分配和释放文件中的空间。

**标记的 NFS**

强制实施数据访问权限，并为 NFS 文件系统上的各个文件在客户端和服务器之间启用 SELinux 标签。

**布局增强**

提供 **layoutstats()** 操作，它可让一些并行 NFS(pNFS)服务器收集更好的性能统计数据。

以下是 NFSv4.1 的功能：

- 增强性能和网络安全，同时包括对 pNFS 的客户端支持。

- 对于回调不再需要单独的 TCP 连接，回调允许 NFS 服务器在无法联系客户端的情况下授予委托：例如，当 NAT 或防火墙干扰时。
- 只提供一次语义（除重启操作外），防止出现先前的问题，即如果回复丢失，且操作被发送了两次，则某些操作有时会返回不准确的结果。

## 2.3. NFSV3 和 NFSV4 中的 TCP 和 UDP 协议

NFSv4 需要通过 IP 网络运行的传输控制协议（TCP）。

NFSv3 还可以使用早期 Red Hat Enterprise Linux 版本中的用户数据报协议(UDP)。在 Red Hat Enterprise Linux 9 中不再支持通过 UDP 的 NFS。默认情况下，UDP 在 NFS 服务器中被禁用。

## 2.4. NFS 所需的服务

这部分列出了运行 NFS 服务器或挂载 NFS 共享所需的系统服务。Red Hat Enterprise Linux 会自动启动这些服务。

Red Hat Enterprise Linux 使用内核级支持和服务流程组合提供 NFS 文件共享。所有 NFS 版本都依赖于客户端和服务器间的远程过程调用（RPC）。要共享或者挂载 NFS 文件系统，下列服务根据所使用的 NFS 版本而定：

**nfsd**

为共享 NFS 文件系统请求的 NFS 服务器内核模块。

**rpcbind**

接受本地 RPC 服务的端口保留。这些端口随后可用（或公布出去），这样相应的远程 RPC 服务可以访问它们。**rpcbind** 服务响应对 RPC 服务的请求，并建立到请求的 RPC 服务的连接。这不能与 NFSv4 一起使用。

**rpc.mountd**

NFS 服务器使用这个进程来处理来自 NFSv3 客户端的 **MOUNT** 请求。它检查所请求的 NFS 共享是否目前由 NFS 服务器导出，并且允许客户端访问它。如果允许挂载请求，**nfs-mountd** 服务会回复 Success 状态，并将此 NFS 共享的文件句柄返回给 NFS 客户端。

**rpc.nfsd**

这个过程启用了要定义的服务器公告的显式 NFS 版本和协议。它与 Linux 内核一起使用，来满足 NFS 客户端的动态需求，例如，在每次连接 NFS 客户端时提供服务器线程。这个进程对应于 **nfs-server** 服务。

**lockd**

这是一个在客户端和服务器中运行的内核线程。它实现网络锁管理器(NLM)协议，它允许 NFSv3 客户端锁住服务器上的文件。每当运行 NFS 服务器以及挂载 NFS 文件系统时，它会自动启动。

**rpc.statd**

这个进程实现网络状态监控器(NSM)RPC 协议，该协议可在 NFS 服务器没有正常关机而重启时通知 NFS 客户端。**rpc-statd** 服务由 **nfs-server** 服务自动启动，不需要用户配置。这不能与 NFSv4 一起使用。

**rpc.rquotad**

这个过程为远程用户提供用户配额信息。启动 **nfs-server** 时，用户也必须启动 **quota-rpc** 软件包提供的 **rpc-rquotad** 服务。

**rpc.idmapd**

此进程为 NFSv4 客户端和服务器提供上行调用，这些调用在线上 NFSv4 名称（***user@domain***形式的字符串）和本地 UID 和 GID 之间进行映射。要使 **idmapd** 与 NFSv4 正常工作，必须配置 **/etc/idmapd.conf** 文件。至少应指定 **Domain** 参数，该参数定义 NFSv4 映射域。如果 NFSv4 映射域

与 DNS 域名相同，可以跳过这个参数。客户端和服务器必须同意 NFSv4 映射域才能使 ID 映射正常工作。
只有 NFSv4 服务器使用 **rpc.idmapd**，它由 **nfs-idmapd** 服务启动。NFSv4 客户端使用基于密钥环的 **nfsidmap** 工具，内核按需调用它来执行 ID 映射。如果 **nfsidmap** 有问题，客户端将退回使用 **rpc.idmapd**。

**NFSv4 的 RPC 服务**
挂载和锁定协议已合并到 NFSv4 协议中。该服务器还会监听已知的 TCP 端口 2049。因此，NFSv4 不需要与 **rpcbind**、**lockd** 和 **rpc-statd** 服务进行交互。NFS 服务器上仍然需要 **nfs-mountd** 服务来设置导出，但不涉及任何线上操作。

**其它资源**

- 仅配置没有 **rpcbind**的服务器。

## 2.5. NFS 主机名格式

这部分论述了在挂载或导出 NFS 共享时用来指定主机的不同格式。

您可以使用以下格式指定主机：

**单台机器**
以下任意一种：

- 完全限定域名（可由服务器解析）
- 主机名（可由服务器解析）
- IP 地址。

**IP 网络**
以下格式之一有效：

- ***a.b.c.d/z***，其中 ***a.b.c.d*** 是网络，***z*** 是子网掩码中的位数，例如 **192.168.0.0/24**。
- ***a.b.c.d/netmask***，其中 ***a.b.c.d*** 是网络，***netmask*** 是子网掩码；例如 **192.168.100.8/255.255.255.0**。

Netgroups
***@group-name*** 格式，其中 ***group-name*** 是 NIS netgroup 名称。

## 2.6. NFS 服务器配置

这部分论述了在 NFS 服务器中配置导出的语法和选项：

- 手动编辑 **/etc/exports** 配置文件
- 在命令行上使用 **exportfs** 工具

### 2.6.1. /etc/exports 配置文件

**/etc/exports** 文件控制哪些文件系统被导出到远程主机，并指定选项。它遵循以下语法规则：

- 空白行将被忽略。
- 要添加注释，以井号(#)开始一行。
- 您可以使用反斜杠(\)换行长行。
- 每个导出的文件系统都应该独立。
- 所有在导出的文件系统后放置的授权主机列表都必须用空格分开。
- 每个主机的选项必须在主机标识符后直接放在括号中，没有空格分离主机和第一个括号。

**导出条目**
导出的文件系统的每个条目都有以下结构：

```
export host(options)
```

您还可以指定多个主机以及每个主机的特定选项。要做到这一点，在同一行中列出主机列表（以空格分隔），每个主机名带有其相关的选项（在括号中），如下所示：

```
export host1(options1) host2(options2) host3(options3)
```

在这个结构中：

*export*
导出的目录

***主机***
导出要共享的主机或网络

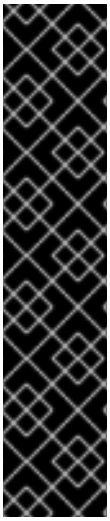
*选项*
用于主机的选项

**例 2.1. 一个简单的 /etc/exports 文件**

在最简单的形式中，**/etc/exports** 文件只指定导出的目录和允许访问它的主机：

```
/exported/directory bob.example.com
```

这里，**bob.example.com** 可以挂载 NFS 服务器的 **/exported/directory/**。因为在这个示例中没有指定选项，所以 NFS 使用默认选项。



**重要**

**/etc/exports** 文件的格式要求非常精确，特别是在空格字符的使用方面。需要将导出的文件系统与主机、不同主机间使用空格分隔。但是，除了注释行外，文件中不应该包括其他空格。

例如，下面两行并不具有相同的意义：

```
/home bob.example.com(rw)
/home bob.example.com (rw)
```

第一行仅允许来自 **bob.example.com** 的用户读写 **/home** 目录。第二行允许来自 **bob.example.com** 的用户以只读方式挂载目录（默认），而其他用户可以将其挂载为读/写。

**默认选项**

导出条目的默认选项有：

**ro**

导出的文件系统是只读的。远程主机无法更改文件系统中共享的数据。要允许主机更改文件系统（即读写），指定 rw 选项。

**sync**

在将之前的请求所做的更改写入磁盘前，NFS 服务器不会回复请求。要启用异步写，请指定 **async** 选项。

**wdelay**

如果 NFS 服务器预期另外一个写入请求即将发生，则 NFS 服务器会延迟写入磁盘。这可以提高性能，因为它可减少不同的写命令访问磁盘的次数，从而减少写开销。要禁用此功能，请指定 **no_wdelay** 选项，该选项仅在指定了默认 sync 选项时才可用。

**root_squash**

这可以防止远程连接的 root 用户（与本地连接相反）具有 root 特权；相反，NFS 服务器为他们分配用户 ID **nobody**。这可以有效地将远程 root 用户的权限"挤压"成最低的本地用户，从而防止在远程服务器上可能的未经授权的写操作。要禁用 root 挤压，请指定 **no_root_squash** 选项。
要挤压每个远程用户（包括 root 用户），请使用 **all_squash** 选项。要指定 NFS 服务器应该分配给来自特定主机的远程用户的用户和组 ID，请分别使用 **anonuid** 和 **anongid** 选项，如下所示：

```
export host(anonuid=uid,anongid=gid)
```

这里，*uid* 和 *gid* 分别是用户 ID 号和组 ID 号。 **anonuid** 和 **anongid** 选项允许您为要共享的远程 NFS 用户创建特殊的用户和组帐户。

默认情况下，Red Hat Enterprise Linux 下的 NFS 支持访问控制列表(ACL)。要禁用此功能，请在导出文件系统时指定 **no_acl** 选项。

**默认和覆盖选项**

每个导出的文件系统的默认值都必须被显式覆盖。例如，如果没有指定 **rw** 选项，则导出的文件系统将以只读形式共享。以下是 **/etc/exports** 中的示例行，其覆盖两个默认选项：

```
/another/exported/directory 192.168.0.3(rw,async)
```

在此示例中，**192.168.0.3** 可以以读写形式挂载 **/another/exported/directory/**，并且所有对磁盘的写入都是异步的。

### 2.6.2. exportfs 工具

**exportfs** 工具使 root 用户能够有选择地导出或取消导出目录，而无需重启 NFS 服务。给定合适的选项后，**exportfs** 工具将导出的文件系统写到 **/var/lib/nfs/xtab**。由于 **nfs-mountd** 服务在决定访问文件系统的特权时参考 **xtab** 文件，所以对导出的文件系统列表的更改会立即生效。

**常用的 exportfs 选项**
以下是用于 **exportfs** 的常用选项列表：

**-r**

通过在 **/var/lib/nfs/etab** 中构建新的导出列表，将 **/etc/exports** 中列出的所有目录导出。如果对 **/etc/exports** 做了任何更改，这个选项可以有效地刷新导出列表。

**-a**

根据哪些其它选项传给了 **exportfs**，将导出或取消导出所有目录。如果没有指定其他选项，**exportfs** 会导出 **/etc/exports** 中指定的所有文件系统。

**-o *file-systems***

指定没有在 **/etc/exports** 中列出的要导出的目录。使用要导出的额外文件系统替换 *file-systems*。这些文件系统的格式化方式必须与 **/etc/exports** 中指定的方式相同。此选项通常用于在将其永久添加到导出的文件系统列表之前测试导出的文件系统。

**-i**

忽略 **/etc/exports** ；只有命令行上指定的选项才会用于定义导出的文件系统。

**-u**

不导出所有共享目录。命令 **exportfs -ua** 可暂停 NFS 文件共享，同时保持所有 NFS 服务正常运行。要重新启用 NFS 共享，请使用 **exportfs -r**。

**-v**

详细操作，当执行 **exportfs** 命令时，更详细地显示正在导出的或取消导出的文件系统。

如果没有选项传给 **exportfs** 工具，它将显示当前导出的文件系统列表。

**其它资源**

- NFS 主机名格式。

## 2.7. NFS 和 RPCBIND

本节解释 NFSv3 所需的 **rpcbind** 服务的用途。

**rpcbind** 服务将远程过程调用(RPC)服务映射到其侦听的端口。RPC 进程在启动时通知 **rpcbind**，注册它们正在侦听的端口以及它们期望提供服务的 RPC 程序号。然后，客户端系统会使用特定的 RPC 程序号联系服务器上的 **rpcbind**。**rpcbind** 服务将客户端重定向到正确的端口号，这样它就可以与请求的服务进行通信。

由于基于 RPC 的服务依赖 **rpcbind** 来与所有传入的客户端请求建立连接，因此 **rpcbind** 必须在这些服务启动之前可用。

**rpcbind** 的访问控制规则会影响所有基于 RPC 的服务。另外，也可以为每个 NFS RPC 守护进程指定访问控制规则。

**其它资源**

- **rpc.mountd(8)** man page

- **rpc.statd(8)** man page

## 2.8. 安装 NFS

这个过程安装挂载或导出 NFS 共享所需的所有软件包。

**流程**

- 安装 **nfs-utils** 软件包：

```
# dnf install nfs-utils
```

## 2.9. 启动 NFS 服务器

这个步骤描述了如何启动 NFS 服务器,这是导出 NFS 共享所必需的。

**先决条件**

- 对于支持 NFSv3 连接的服务器，**rpcbind** 服务必须处于运行状态。要验证 **rpcbind** 是否处于活动状态，请使用以下命令：

```
$ systemctl status rpcbind
```

如果停止该服务，启动并启用该服务：

```
$ systemctl enable --now rpcbind
```

**流程**

- 要启动 NFS 服务器，并使其在引导时自动启动，请使用以下命令：

```
# systemctl enable --now nfs-server
```

**其它资源**

- 配置只使用 NFSv4 的服务器。

## 2.10. NFS 和 RPCBIND 故障排除

由于 **rpcbind** 服务在 RPC 服务和用于与之通信的端口号之间提供协调，因此在排除故障时，使用 **rpcbind** 查看当前 RPC 服务的状态非常有用。**rpcinfo** 工具显示每个基于 RPC 的服务，以及其端口号、RPC 程序号、版本号和 IP 协议类型（TCP 或 UDP）。

**流程**

1. 要确保为 **rpcbind** 启用了正确的基于 NFS RPC 的服务，请使用以下命令：

```
# rpcinfo -p
```

**例 2.2. rpcinfo -p 命令输出**

下面是一个这个命令的输出示例：

```
program vers proto   port  service
 100000    4   tcp    111  portmapper
 100000    3   tcp    111  portmapper
 100000    2   tcp    111  portmapper
 100000    4   udp    111  portmapper
 100000    3   udp    111  portmapper
 100000    2   udp    111  portmapper
 100005    1   udp  20048  mountd
 100005    1   tcp  20048  mountd
 100005    2   udp  20048  mountd
 100005    2   tcp  20048  mountd
 100005    3   udp  20048  mountd
 100005    3   tcp  20048  mountd
 100024    1   udp  37769  status
 100024    1   tcp  49349  status
 100003    3   tcp   2049  nfs
 100003    4   tcp   2049  nfs
 100227    3   tcp   2049  nfs_acl
 100021    1   udp  56691  nlockmgr
 100021    3   udp  56691  nlockmgr
 100021    4   udp  56691  nlockmgr
 100021    1   tcp  46193  nlockmgr
 100021    3   tcp  46193  nlockmgr
 100021    4   tcp  46193  nlockmgr
```

如果其中一个 NFS 服务没有正确启动，**rpcbind** 将不能将来自客户端的对该服务的 RPC 请求映射到正确的端口。

2. 在很多情况下，如果 NFS 没有出现在 **rpcinfo** 输出中，重启 NFS 会使服务正确使用 **rpcbind** 注册，并开始工作：

```
# systemctl restart nfs-server
```

**其它资源**

- 配置只使用 NFSv4 的服务器。

## 2.11. 将 NFS 服务器配置为在防火墙后运行

NFS 需要 **rpcbind** 服务，该服务为 RPC 服务动态分配端口，并可能导致配置防火墙规则时出现问题。下面的部分描述了如何在防火墙后配置 NFS 版本（如果要支持）：

- NFSv3
  这包括支持 NFSv3 的任何服务器：
  - NFSv3-only 服务器
  - 支持 NFSv3 和 NFSv4 的服务器
- 只使用 NFSv4

### 2.11.1. 将 NFSv3-enabled 服务器配置为在防火墙后运行

下面的步骤描述了如何将支持 NFSv3 的服务器配置为在防火墙后运行。这包括支持 NFSv3 和 NFSv4 的 NFSv3-only 服务器和服务器。

**流程**

1. 要允许客户端访问防火墙后面的 NFS 共享，请在 NFS 服务器上运行以下命令来配置防火墙：

```
firewall-cmd --permanent --add-service mountd
firewall-cmd --permanent --add-service rpc-bind
firewall-cmd --permanent --add-service nfs
```

2. 指定 **/etc/nfs.conf** 文件中 RPC 服务 **nlockmgr** 使用的端口，如下所示：

```
[lockd]

port=tcp-port-number
udp-port=udp-port-number
```

   或者，您可以在 **/etc/modprobe.d/lockd.conf** 文件中指定 **nlm_tcpport** 和 **nlm_udpport**。

3. 在 NFS 服务器中运行以下命令打开防火墙中指定的端口：

```
firewall-cmd --permanent --add-port=<lockd-tcp-port>/tcp
firewall-cmd --permanent --add-port=<lockd-udp-port>/udp
```

4. 通过编辑 **/etc/nfs.conf** 文件的 **[statd]** 部分为 **rpc.statd** 添加静态端口，如下所示：

```
[statd]

port=port-number
```

5. 在 NFS 服务器中运行以下命令，在防火墙中打开添加的端口：

```
firewall-cmd --permanent --add-port=<statd-tcp-port>/tcp
firewall-cmd --permanent --add-port=<statd-udp-port>/udp
```

6. 重新载入防火墙配置：

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

7. 首先重启 **rpc-statd** 服务，然后重启 **nfs-server** 服务：

```
# systemctl restart rpc-statd.service
# systemctl restart nfs-server.service
```

   或者，如果您在 **/etc/modprobe.d/ lockd.conf** 文件中指定锁定端口：

   a. 更新 **/proc/sys/fs/nfs/nlm_tcpport** 和 **/proc/sys/fs/nfs/nlm_udpport** 的当前值：

```
# sysctl -w fs.nfs.nlm_tcpport=<tcp-port>
# sysctl -w fs.nfs.nlm_udpport=<udp-port>
```

b. 重启 **rpc-statd** 和 **nfs-server** 服务：

```
# systemctl restart rpc-statd.service
# systemctl restart nfs-server.service
```

### 2.11.2. 将只使用 NFSv4 的服务器配置为在防火墙后运行

下面的步骤描述了如何将只使用 NFSv4 的服务器配置为在防火墙后运行。

**流程**

1. 要允许客户端在防火墙后访问 NFS 共享，在 NFS 服务器中运行以下命令配置防火墙：

```
firewall-cmd --permanent --add-service nfs
```

2. 重新载入防火墙配置：

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

3. 重启 nfs-server：

```
# systemctl restart nfs-server
```

### 2.11.3. 将 NFSv3 客户端配置为在防火墙后运行

将 NFSv3 客户端配置为在防火墙后运行的步骤类似于将 NFSv3 服务器配置为在防火墙后运行。

如果您要配置的机器既是 NFS 客户端和服务器，请按照 将 NFSv3-enabled 服务器配置为在防火墙后运行 中所述的步骤进行。

以下流程描述了如何配置只在防火墙后运行的 NFS 客户端的机器。

**流程**

1. 要在客户端位于防火墙后允许 NFS 客户端对 NFS 客户端执行回调，请在 NFS 客户端上运行以下命令将 **rpc-bind** 服务添加到防火墙中：

```
firewall-cmd --permanent --add-service rpc-bind
```

2. 指定 **/etc/nfs.conf** 文件中 RPC 服务 **nlockmgr** 使用的端口，如下所示：

```
[lockd]

port=port-number
udp-port=upd-port-number
```

或者，您可以在 **/etc/modprobe.d/lockd.conf** 文件中指定 **nlm_tcpport** 和 **nlm_udpport**。

3. 在 NFS 客户端中运行以下命令打开防火墙中指定的端口：

```
firewall-cmd --permanent --add-port=<lockd-tcp-port>/tcp
firewall-cmd --permanent --add-port=<lockd-udp-port>/udp
```

4. 通过编辑 **/etc/nfs.conf** 文件的 **[statd]** 部分为 **rpc.statd** 添加静态端口，如下所示：

```
[statd]

port=port-number
```

5. 在 NFS 客户端中运行以下命令，在防火墙中打开添加的端口：

```
firewall-cmd --permanent --add-port=<statd-tcp-port>/tcp
firewall-cmd --permanent --add-port=<statd-udp-port>/udp
```

6. 重新载入防火墙配置：

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

7. 重启 **rpc-statd** 服务：

```
# systemctl restart rpc-statd.service
```

或者，如果您在 **/etc/modprobe.d/ lockd.conf** 文件中指定锁定端口：

a. 更新 **/proc/sys/fs/nfs/nlm_tcpport** 和 **/proc/sys/fs/nfs/nlm_udpport** 的当前值：

```
# sysctl -w fs.nfs.nlm_tcpport=<tcp-port>
# sysctl -w fs.nfs.nlm_udpport=<udp-port>
```

b. 重启 **rpc-statd** 服务：

```
# systemctl restart rpc-statd.service
```

### 2.11.4. 将 NFSv4 客户端配置为在防火墙后运行

仅在客户端使用 NFSv4.0 时执行此步骤。在这种情况下，需要为 NFSv4.0 回调打开端口。

NFSv4.1 或更高版本不需要这个过程，因为在后续协议版本中，服务器在客户端发起的同一连接上执行回调。

**流程**

1. 要允许 NFSv4.0 回调通过防火墙，请设置 **/proc/sys/fs/nfs_callback_tcpport** 并允许服务器连接到客户端上的该端口，如下所示：

```
# echo "fs.nfs.nfs_callback_tcpport = <callback-port>" >/etc/sysctl.d/90-nfs-callback-port.conf
# sysctl -p /etc/sysctl.d/90-nfs-callback-port.conf
```

2. 在 NFS 客户端中运行以下命令打开防火墙中指定的端口：

```
firewall-cmd --permanent --add-port=<callback-port>/tcp
```

3. 重新载入防火墙配置：

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

## 2.12. 通过防火墙导出 RPC 配额

如果您导出使用磁盘配额的文件系统，您可以使用配额远程过程调用(RPC)服务来给 NFS 客户端提供磁盘配额数据。

**流程**

1. 启用并启动 **rpc-rquotad** 服务：

```
# systemctl enable --now rpc-rquotad
```

**注意**

如果启用了 **rpc-rquotad** 服务，其会在启动 nfs-server 服务后自动启动。

2. 为了使配额 RPC 服务可在防火墙后访问，需要打开 TCP（如果启用了 UDP，则为 UDP）端口 875。默认端口号定义在 **/etc/services** 文件中。
   您可以通过将 **-p port-number** 附加到 **/etc/sysconfig/rpc-rquotad** 文件中的 **RPCRQUOTADOPTS** 变量来覆盖默认端口号。

3. 默认情况下，远程主机只能读配额。如果要允许客户端设置配额，请将 **-S** 选项附加到 **/etc/sysconfig/rpc-rquotad** 文件中的 **RPCRQUOTADOPTS** 变量中。

4. 重启 **rpc-rquotad** ，以使 **/etc/sysconfig/rpc-rquotad** 文件中的更改生效：

```
# systemctl restart rpc-rquotad
```

## 2.13. 启用通过 RDMA(NFSORDMA) 的 NFS

如果存在支持 RDMA 的硬件，则在 Red Hat Enterprise Linux 9 中，远程直接内存访问(RDMA)服务会自动工作。

**流程**

1. 安装 **rdma-core** 软件包：

```
# dnf install rdma-core
```

2. 重启 **nfs-server** 服务：

```
# systemctl restart nfs-server
```

**其它资源**

- RFC 5667 标准

## 2.14. 其它资源

- Linux NFS wiki

# 第 3 章 保护 NFS

为最大程度地降低 NFS 安全风险并保护服务器上的数据，在服务器上导出 NFS 文件系统或将其挂载到客户端上时，请考虑以下部分：

## 3.1. 带有 AUTH_SYS 和导出控制的 NFS 安全性

NFS 提供以下传统选项来控制对导出文件的访问：

- 服务器限制哪些主机可以通过 IP 地址或主机名挂载哪些文件系统。
- 服务器对 NFS 客户端上的用户强制执行文件系统权限的方式与对本地用户强制执行权限的方式相同。传统上，NFS 使用 **AUTH_SYS** 调用消息（也称为 **AUTH_UNIX**）来执行此操作，该消息依赖于客户端来声明用户的 UID 和 GID。请注意，这意味着恶意或者错误配置的客户端可能会轻松地利用这个问题，导致用户可以访问不应该被访问的文件。

为限制潜在的风险，管理员通常将访问权限限制为只读或将用户权限挤压成普通用户和组 ID。不幸的是，这些解决方案会阻止 NFS 共享以最初预期的方式使用。

另外，如果攻击者获得了对导出 NFS 文件系统的系统所使用的 DNS 服务器的控制，它们可将与特定主机名或完全限定域名关联的系统指向未授权的机器。此时，未经授权的机器 *是* 允许挂载 NFS 共享的系统，因为没有交换用户名或密码信息来为 NFS 挂载提供额外的安全。

在通过 NFS 导出目录时应谨慎使用通配符，因为通配符的范围可能包含比预期更多的系统。

**其它资源**

- 要保护 NFS 和 **rpcbind**，例如，可使用 **nftables** 和 **firewalld**。
- **nft(8)** man page
- **firewalld-cmd(1)** man page

## 3.2. 带有 AUTH_GSS 的 NFS 安全性

NFS 的所有版本都支持 RPCSEC_GSS 和 Kerberos 机制。

与 AUTH_SYS 不同，使用 RPCSEC_GSS Kerberos 机制，服务器不依赖于客户端就可以正确地表示哪个用户正在访问文件。相反，加密用于向服务器验证用户的身份，这可防止恶意的客户端在没有用户的 Kerberos 凭据的情况下模拟该用户。使用 RPCSEC_GSS Kerberos 机制是保护挂载的最直接方法，因为配置了 Kerberos 后不需要额外的设置。

## 3.3. 配置 NFS 服务器和客户端使用 KERBEROS

Kerberos 是一种网络身份验证系统，其允许客户端和服务器使用对称加密和信任的第三方 KDC 来互相进行身份验证。红帽建议使用身份管理(IdM)来设置 Kerberos。

**先决条件**

- Kerberos 密钥分发中心(**KDC**)已安装和配置 。

**流程**

1. - 在 NFS 服务器端创建 **nfs/hostname.*domain@REALM*** 主体。

- 在服务器和客户端端创建 **host/hostname.*domain@REALM*** 主体。
- 将对应的密钥添加到客户端和服务器的 keytab 中。

2. 在服务器端，使用 **sec=** 选项来启用所想要的安全类别。启用所有安全类型和非加密挂载：

```
/export *(sec=sys:krb5:krb5i:krb5p)
```

与 **sec=** 选项一起使用的有效安全类型为：

- **sys**: 无加密保护，默认值
- **krb5** ：仅用于验证
- **krb5i** ：完整性保护
  - 使用 Kerberos V5 进行用户身份验证，并使用安全校验和执行 NFS 操作的完整性检查，以防止数据篡改。
- **krb5p** ：隐私保护
  - 使用 Kerberos V5 进行用户身份验证、完整性检查及加密 NFS 流量以防止流量嗅探。这是最安全的设置，但它也会涉及最大的性能开销。

3. 在客户端，将 **sec=krb5** （或 **sec=krb5i** 或 **sec=krb5p**，取决于设置 ）添加到挂载选项：

```
# mount -o sec=krb5 server:/export /mnt
```

**其它资源**

- 在 krb5 保护的 NFS 上以 root 用户身份创建文件 .不建议。
- **exports(5)** 手册页
- **nfs(5)** 手册页

## 3.4. NFSV4 安全选项

NFSv4 包括基于 Microsoft Windows NT 模型，而非 POSIX 模型的 ACL 支持，因为 Microsoft Windows NT 模型的功能和广泛的部署。

NFSv4 的另一个重要安全功能是，对挂载文件系统删除了 **MOUNT** 协议的使用。**MOUNT** 协议存在安全风险，因为协议处理文件句柄的方式。

## 3.5. 挂载的 NFS 导出的文件权限

远程主机一旦将 NFS 文件系统挂载为读取或读写，则保护每个共享文件的唯一方法就是其权限。如果共享同一用户 ID 值的两个用户在不同的客户端系统上挂载相同的 NFS 文件系统，他们可以修改彼此的文件。此外，在客户端系统上以 root 身份登录的任何人都可以使用 **su -** 命令来访问 NFS 共享的任何文件。

默认情况下，Red Hat Enterprise Linux 的 NFS 支持访问控制列表（ACL）。红帽建议启用此功能。

默认情况下，NFS 在导出文件系统时使用 *root squashing*。这会将本地机器上以 root 用户身份访问 NFS 共享的任何人的用户 ID 设为 **nobody**。Root squashing 由默认选项 **root_squash** 控制；有关此选项的更多信息，请参阅 NFS 服务器配置。

将 NFS 共享导出为只读时，请考虑使用 **all_squash** 选项。这个选项使访问导出的文件系统的每个用户都使用 **nobody** 用户的用户 ID。

# 第 4 章 在 NFS 中启用 PNFS SCSI 布局

您可以将 NFS 服务器和客户端配置为使用 pNFS SCSI 布局访问数据。

**先决条件**

- 客户端和服务器必须能够向同一个块设备发送 SCSI 命令。就是说块设备必须位于共享的 SCSI 总线中。
- 块设备必须包含 XFS 文件系统。
- SCSI 设备必须支持 SCSI Persistent Reservations，如 SCSI-3 Ppriary Commands 规格中所述。

## 4.1. PNFS 技术

pNFS 构架提高了 NFS 的可伸缩性。当服务器实现 pNFS 时，客户端可以同时通过多个服务器访问数据。这可提高性能。

pNFS 支持 RHEL 中的以下存储协议或布局：

- 文件
- Flexfiles
- SCSI

## 4.2. PNFS SCSI 布局

SCSI 布局基于 pNFS 块布局的工作。布局在 SCSI 设备中定义。它包含一系列固定大小的块来作为逻辑单元(LU)，这些逻辑单元必须能够支持 SCSI 持久保留。LU 设备识别通过其 SCSI 设备识别。

在涉及长时间的单客户端访问文件的用例中，pNFS SCSI 表现良好。例如：邮件服务器或者虚拟机。

**客户端和服务器间的操作**
当 NFS 客户端从文件读取或写入文件时，客户端会执行 **LAYOUTGET** 操作。服务器会使用文件在 SCSI 设备中的位置进行响应。客户端可能需要执行 **GETDEVICEINFO** 的额外操作，以确定要使用哪个 SCSI 设备。如果这些操作正常工作，客户端可以直接向 SCSI 设备发出 I/O 请求，而不必向服务器发送 **READ** 和 **WRITE** 操作。

客户端之间的错误或争用可能会导致服务器重新调用布局，或者不将它们发送给客户端。在这些情况下，客户端回退到向服务器发出 **READ** 和 **WRITE** 操作，而不是直接向 SCSI 设备发送 I/O 请求。

要监控操作，请参阅 监控 pNFS SCSI 布局功能。

**设备保留**
pNFS SCSI 通过分配保留来处理保护。在服务器向客户端发送布局之前，它会保留 SCSI 设备，以确保只有注册的客户端才可以访问该设备。如果客户端可以向那个 SCSI 设备发送命令，但没有在该设备上注册，那么该设备上的客户端的许多操作都会失败。例如，如果服务器没有向客户端提供该设备的布局，则客户端上的 **blkid** 命令将无法显示 XFS 文件系统的 UUID。

服务器不会删除其自身的持久性保留。这样可在重启客户端和服务器后保护该设备中的文件系统中的数据。为了重新使用 SCSI 设备，您可能需要手动删除 NFS 服务器中的持久性保留。

## 4.3. 检查与 PNFS 兼容的 SCSI 设备

这个过程检查 SCSI 设备是否支持 pNFS SCSI 布局。

**先决条件**

- 安装 **sg3_utils** 软件包：

```
# dnf install sg3_utils
```

**流程**

- 在服务器和客户端中检查正确的 SCSI 设备支持：

```
# sg_persist --in --report-capabilities --verbose path-to-scsi-device
```

确保设置了 *Persist Through Power Los Active*（**PTPL_A**)位。

**例 4.1. 支持 pNFS SCSI 的 SCSI 设备**

以下是支持 pNFS SCSI 的 SCSI 设备的 **sg_persist** 输出示例。**PTPL_A** 位报告 **1**。

```
    inquiry cdb: 12 00 00 00 24 00
    Persistent Reservation In cmd: 5e 02 00 00 00 00 00 20 00 00
  LIO-ORG   block11          4.0
  Peripheral device type: disk
Report capabilities response:
  Compatible Reservation Handling(CRH): 1
  Specify Initiator Ports Capable(SIP_C): 1
  All Target Ports Capable(ATP_C): 1
  Persist Through Power Loss Capable(PTPL_C): 1
  Type Mask Valid(TMV): 1
  Allow Commands: 1
  Persist Through Power Loss Active(PTPL_A): 1
    Support indicated in Type mask:
      Write Exclusive, all registrants: 1
      Exclusive Access, registrants only: 1
      Write Exclusive, registrants only: 1
      Exclusive Access: 1
      Write Exclusive: 1
      Exclusive Access, all registrants: 1
```

**其它资源**

- **sg_persist(8)** man page

## 4.4. 在服务器中设置 PNFS SCSI

这个过程将 NFS 服务器配置为导出 pNFS SCSI 布局。

**流程**

1. 在服务器中挂载在 SCSI 设备中创建的 XFS 文件系统。

2. 将 NFS 服务器配置为导出 NFS 版本 4.1 或更高版本。在 **/etc/nfs.conf** 文件的 **[nfsd]** 部分设置以下选项：

```
[nfsd]

vers4.1=y
```

3. 配置 NFS 服务器，来使用 **pnfs** 选项通过 NFS 导出 XFS 文件系统：

**例 4.2. /etc/exports 中的条目导出 pNFS SCSI**

**/etc/exports** 配置文件中的以下条目将挂载于 **/exported/directory/** 的文件系统导出到 **allowed.example.com** 客户端，来作为 pNFS SCSI 布局：

```
/exported/directory allowed.example.com(pnfs)
```

**其它资源**

- 导出 NFS 共享.

## 4.5. 在客户端中设置 PNFS SCSI

这个过程将 NFS 客户端配置为挂载 pNFS SCSI 布局。

**先决条件**

- NFS 服务器被配置为通过 pNFS SCSI 导出 XFS 文件系统。请参阅在服务器 中设置 pNFS SCSI。

**流程**

- 在客户端中使用 NFS 版本 4.1 或更高版本挂载导出的 XFS 文件系统：

```
# mount -t nfs -o nfsvers=4.1 host:/remote/export /local/directory
```

不要在没有 NFS 的情况下直接挂载 XFS 文件系统。

**其它资源**

- 挂载 NFS 共享。

## 4.6. 在服务器中释放 PNFS SCSI 保留

此流程释放 NFS 服务器在 SCSI 设备中拥有的持久保留。这可让您在不再需要导出 pNFS SCSI 时重新使用 SCSI 设备。

您必须从服务器中删除保留。它不能从不同的 IT Nexus 中删除。

**先决条件**

- 安装 **sg3_utils** 软件包：

```
# dnf install sg3_utils
```

**流程**

1. 在服务器上查询现有保留：

```
# sg_persist --read-reservation path-to-scsi-device
```

**例 4.3. 在 /dev/sda 中查询保留**

```
# *sg_persist --read-reservation /dev/sda*

  LIO-ORG   block_1          4.0
  Peripheral device type: disk
  PR generation=0x8, Reservation follows:
    Key=0x100000000000000
    scope: LU_SCOPE,  type: Exclusive Access, registrants only
```

2. 删除服务器上的现有注册：

```
# sg_persist --out \
            --release \
            --param-rk=reservation-key \
            --prout-type=6 \
            path-to-scsi-device
```

**例 4.4. 删除 /dev/sda 中的保留**

```
# sg_persist --out \
              --release \
              --param-rk=0x100000000000000 \
              --prout-type=6 \
              /dev/sda

  LIO-ORG   block_1          4.0
  Peripheral device type: disk
```

**其它资源**

- **sg_persist(8)** man page