



Red Hat Enterprise Linux 9

네트워크 파일 서비스 구성 및 사용

Red Hat Enterprise Linux 9에서 네트워크 파일 서비스를 구성 및 사용하는 가이드입니다.

Red Hat Enterprise Linux 9 네트워크 파일 서비스 구성 및 사용

Red Hat Enterprise Linux 9에서 네트워크 파일 서비스를 구성 및 사용하는 가이드입니다.

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.

법적 공지

Copyright © 2022 | You need to change the HOLDER entity in the en-US/Configuring_and_using_network_file_services.ent file |.

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

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

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

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

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

초록

이 문서에서는 Samba 서버 및 NFS 서버를 포함하여 Red Hat Enterprise Linux 9에서 네트워크 파일 서비스를 구성하고 실행하는 방법을 설명합니다.

차례

보다 포괄적 수용을 위한 오픈 소스 용어 교체	5
RED HAT 문서에 관한 피드백 제공	6
1장. SAMBA를 서버로 사용	7
1.1. 다양한 SAMBA 서비스 및 모드 이해	7
1.1.1. Samba 서비스	7
1.1.2. Samba 보안 서비스	8
1.1.3. Samba 서비스 및 Samba 클라이언트 유틸리티가 구성을 로드 및 다시 로드하는 시나리오	8
1.1.4. 안전한 방식으로 Samba 구성 편집	9
1.2. SAMBA 구성 확인	10
1.2.1. testparm 유틸리티를 사용하여 smb.conf 파일 확인	10
1.3. 독립 실행형 서버로 SAMBA 설정	10
1.3.1. 독립 실행형 서버에 대한 서버 구성 설정	11
1.3.2. 로컬 사용자 계정 생성 및 활성화	12
1.4. SAMBA ID 매핑 이해 및 구성	12
1.4.1. Samba ID 범위 계획	13
1.4.2. * 기본 도메인	14
1.4.3. tdb ID 매핑 백엔드 사용	15
1.4.4. ad ID 매핑 백엔드 사용	15
1.4.5. 제거 ID 매핑 백엔드 사용	17
1.4.6. 자동 덮어쓰기 ID 매핑 백엔드 사용	19
1.5. AD 도메인 멤버 서버로 SAMBA 설정	21
1.5.1. AD 도메인에 RHEL 시스템 연결	21
1.5.2. MIT Kerberos용 로컬 인증 플러그인 사용	24
1.6. IDM 도메인 멤버에서 SAMBA 설정	24
1.6.1. 도메인 멤버에 Samba를 설치하기 위해 IdM 도메인 준비	25
1.6.2. GPO를 사용하여 Active Directory의 AES 암호화 유형 활성화	26
1.6.3. IdM 클라이언트에 Samba 서버 설치 및 구성	27
1.6.4. IdM이 새 도메인을 신뢰하는 경우 ID 매핑 구성 수동 추가	29
1.6.5. 추가 리소스	30
1.7. POSIX ACL을 사용하는 SAMBA 파일 공유 설정	30
1.7.1. POSIX ACL을 사용하는 공유 추가	30
1.7.2. POSIX ACL을 사용하는 Samba 공유에 표준 Linux ACL 설정	31
1.7.3. POSIX ACL을 사용하는 Samba 공유에서 확장 ACL 설정	32
1.8. POSIX ACL을 사용하는 공유에 권한 설정	34
1.8.1. 사용자 및 그룹 기반 공유 액세스 구성	34
1.8.2. 호스트 기반 공유 액세스 구성	35
1.9. WINDOWS ACL을 사용하는 공유 설정	35
1.9.1. SeDiskOperatorPrivilege 권한 부여	36
1.9.2. Windows ACL 지원 활성화	36
1.9.3. Windows ACL을 사용하는 공유 추가	37
1.9.4. Windows ACL을 사용하는 공유의 공유 권한 및 파일 시스템 ACL 관리	38
1.10. SMBACLS를 사용하여 SMB 공유에서 ACL 관리	38
1.10.1. 액세스 제어 항목	38
1.10.2. rootfscacIs를 사용하여 ACL 표시	41
1.10.3. ACE 마스크 계산	42
1.10.4. rootfscacIs를 사용하여 ACL 추가, 업데이트 및 제거	43
ACL 추가	43
ACL 업데이트	43
ACL 삭제	43

1.11. 사용자가 SAMBA 서버에서 디렉토리를 공유 가능	43
1.11.1. 사용자 공유 기능 활성화	44
1.11.2. 사용자 공유 추가	45
1.11.3. 사용자 공유의 설정 업데이트	46
1.11.4. 기존 사용자 공유에 대한 정보 표시	46
1.11.5. 사용자 공유 나열	47
1.11.6. 사용자 공유 삭제	48
1.12. 인증 없이 액세스를 허용하도록 공유 구성	48
1.12.1. 공유에 대한 게스트 액세스 활성화	48
1.13. MACOS 클라이언트용 SAMBA 구성	50
1.13.1. macOS 클라이언트에 파일 공유를 제공하기 위해 Samba 구성 최적화	50
1.14. SMBCLIENT 유틸리티를 사용하여 SMB 공유에 액세스	52
1.14.1. NetNamespaceclient 대화형 모드가 작동하는 방법	52
1.14.2. 대화형 모드에서 smbclient 사용	53
1.14.3. 스크립팅 모드에서 smbclient 사용	54
1.15. 인쇄 서버로 SAMBA 설정	54
1.15.1. Samba 스펴sd 서비스	54
1.15.2. Samba에서 출력 서버 지원 활성화	56
1.15.3. 수동으로 특정 프린터 공유	57
1.16. SAMBA 인쇄 서버에서 WINDOWS 클라이언트의 자동 프린터 드라이버 다운로드 설정	58
1.16.1. 프린터 드라이버에 대한 기본 정보	59
지원되는 드라이버 모델 버전	59
패키지 인식 드라이버	59
업로드를 위한 프린터 드라이버 준비	59
클라이언트에 프린터를 위한 32비트 및 64비트 드라이버 제공	60
1.16.2. 사용자가 드라이버를 업로드하고 사전 구성 가능	60
1.16.3. print\$ 공유 설정	60
1.16.4. 클라이언트가 Samba 인쇄 서버를 신뢰할 수 있도록 VDDK 만들기	63
1.16.5. 드라이버 업로드 및 프린터 사전 구성	66
1.17. FIPS 모드가 활성화된 서버에서 SAMBA 실행	67
1.17.1. FIPS 모드에서 Samba 사용 제한	67
1.17.2. FIPS 모드에서 Samba 사용	68
1.18. SAMBA 서버의 성능 튜닝	69
1.18.1. SMB 프로토콜 버전 설정	69
1.18.2. 많은 수의 파일이 포함된 디렉터리와의 공유 튜닝	69
1.18.3. 성능에 부정적인 영향을 줄 수 있는 설정	71
1.19. 기본 버전보다 SMB 버전이 필요한 클라이언트와 호환되도록 SAMBA 구성	71
1.19.1. Samba 서버에서 지원하는 최소 SMB 프로토콜 버전 설정	71
1.20. 자주 사용되는 SAMBA 명령줄 유틸리티	72
1.20.1. 네트워크 광고 조인 및 net#189 조인 명령 사용	72
1.20.2. net rpc 권한 명령 사용	74
설정할 수 있는 권한 나열	74
권한 부여	74
권한 해지	75
1.20.3. net rpc share 명령 사용	75
공유 나열	75
공유 추가	75
공유 제거	76
1.20.4. net user 명령 사용	77
도메인 사용자 계정 나열	77
도메인에 사용자 계정 추가	77
도메인에서 사용자 계정 삭제	78
1.20.5. rpcclient 유틸리티 사용	78

예제	78
1.20.6. samba-regedit 애플리케이션 사용	79
1.20.7. smbcontrol 유틸리티 사용	81
1.20.8. smbpasswd 유틸리티 사용	81
1.20.9. smbstatus 유틸리티 사용	83
1.20.10. smbtar 유틸리티 사용	84
1.20.11. wbinfo 유틸리티 사용	84
1.21. 추가 리소스	85
2장. NFS 공유 내보내기	87
2.1. NFS 소개	87
2.2. 지원되는 NFS 버전	87
기본 NFS 버전	87
마이너 NFS 버전의 기능	88
2.3. NFSV3 및 NFSV4의 TCP 및 UDP 프로토콜	89
2.4. NFS에 필요한 서비스	89
NFSv4를 사용한 RPC 서비스	90
2.5. NFS 호스트 이름 형식	90
2.6. NFS 서버 구성	91
2.6.1. /etc/exports 구성 파일	92
내보내기 항목	92
기본 옵션	93
기본 및 재정의 옵션	94
2.6.2. VDDK 유틸리티	95
일반tekton 옵션	95
2.7. NFS 및 RPCBIND	96
2.8. NFS 설치	96
2.9. NFS 서버 시작	97
2.10. NFS 및 RPCBIND 문제 해결	97
2.11. 방화벽 뒤에서 실행되도록 NFS 서버 구성	98
2.11.1. 방화벽 뒤에서 실행되도록 NFSv3- 사용 서버 구성	99
2.11.2. 방화벽 뒤에서 실행되도록 NFSv4 전용 서버 구성	101
2.11.3. 방화벽 뒤에서 실행되도록 NFSv3 클라이언트 구성	101
2.11.4. 방화벽 뒤에서 실행되도록 NFSv4 클라이언트 구성	103
2.12. 방화벽을 통해 RPC 할당량 내보내기	104
2.13. RDMA를 통한 NFS 활성화 (NFSORDMA)	104
2.14. 추가 리소스	105
3장. NFS 보안	106
3.1. AUTH_SYS 및 내보내기 컨트롤을 사용한 NFS 보안	106
3.2. AUTH_GSS를 사용한 NFS 보안	107
3.3. KERBEROS를 사용하도록 NFS 서버 및 클라이언트 구성	107
3.4. NFSV4 보안 옵션	108
3.5. 마운트된 NFS 내보내기에 대한 파일 권한	109
4장. NFS에서 PNFS SCSI 레이아웃 활성화	110
4.1. PNFS 기술	110
4.2. PNFS SCSI 레이아웃	110
클라이언트와 서버 간의 작업	111
장치 예약	111
4.3. PNFS와 호환되는 SCSI 장치 확인	111
4.4. 서버에서 PNFS SCSI 설정	112
4.5. 클라이언트에서 PNFS SCSI 설정	113
4.6. 서버에서 PNFS SCSI 예약 해제	114

보다 포괄적 수용을 위한 오픈 소스 용어 교체

Red Hat은 코드, 문서, 웹 속성에서 문제가 있는 용어를 교체하기 위해 최선을 다하고 있습니다. 먼저 마스터(master), 슬레이브(slave), 블랙리스트(blacklist), 화이트리스트(whitelist) 등 네 가지 용어를 교체하고 있습니다. 이러한 변경 작업은 작업 범위가 크므로 향후 여러 릴리스에 걸쳐 점차 구현할 예정입니다. 자세한 내용은 [CTO Chris Wright의 메시지](#)를 참조하십시오.

RED HAT 문서에 관한 피드백 제공

문서 개선을 위한 의견을 보내 주십시오. Red Hat이 어떻게 이를 개선할 수 있는지 알려 주십시오.

- 특정 문구에 대한 간단한 의견 작성 방법은 다음과 같습니다.
 1. 문서가 *Multi-page HTML* 형식으로 표시되는지 확인합니다. 또한 문서 오른쪽 상단에 **피드백** 버튼이 있는지 확인합니다.
 2. 마우스 커서를 사용하여 주석 처리하려는 텍스트 부분을 강조 표시합니다.
 3. 강조 표시된 텍스트 아래에 표시되는 **피드백 추가** 팝업을 클릭합니다.
 4. 표시된 지침을 따릅니다.
- Bugzilla를 통해 피드백을 제출하려면 새 티켓을 생성합니다.
 1. [Bugzilla](#) 웹 사이트로 이동합니다.
 2. 구성 요소로 **문서**를 사용합니다.
 3. **설명** 필드에 문서 개선을 위한 제안 사항을 기입하십시오. 관련된 문서의 해당 부분 링크를 알려주십시오.
 4. **버그 제출**을 클릭합니다.

1장. SAMBA를 서버로 사용

Samba는 Red Hat Enterprise Linux에서 SMB(Server Message Block) 프로토콜을 구현합니다. SMB 프로토콜은 파일 공유 및 공유 프린터와 같은 서버의 리소스에 액세스하는 데 사용됩니다. 또한 Samba는 Microsoft Windows에서 사용하는 DCE RPC(Distributed Computing Environment Remote Procedure Call) 프로토콜을 구현합니다.

다음과 같이 Samba를 실행할 수 있습니다.

- Active Directory(AD) 또는 NT4 도메인 구성원
- 독립 실행형 서버
- NT4 PDC(기본 도메인 컨트롤러) 또는 백업 도메인 컨트롤러(BDC)



참고

Red Hat은 NT4 도메인을 지원하는 Windows 버전이 있는 기존 설치에서만 PDC 및 BDC 모드를 지원합니다. Windows 7 및 Windows Server 2008 R2 이외의 Microsoft 운영 체제에서 NT4 도메인을 지원하지 않기 때문에 새 Samba NT4 도메인을 설정하지 않는 것이 좋습니다.

Red Hat은 Samba를 AD DC(Domain Controller)로 실행하는 것을 지원하지 않습니다.

설치 모드와는 독립적으로 디렉터리와 프린터를 선택적으로 공유할 수 있습니다. 그러면 Samba가 파일 및 인쇄 서버로 작동할 수 있습니다.

1.1. 다양한 SAMBA 서비스 및 모드 이해

이 섹션에서는 Samba에 포함된 다양한 서비스와 다양한 모드를 설명합니다.

1.1.1. Samba 서비스

Samba는 다음 서비스를 제공합니다.

smbd

이 서비스는 SMB 프로토콜을 사용하여 파일 공유 및 인쇄 서비스를 제공합니다. 또한 서비스는 리소스 잠금을 담당하고 사용자 연결을 인증합니다. 도메인 멤버를 인증하기 위해 **smbd**에는 **winbindd**가 필요합니다. **smb systemd** 서비스는 **smbd** 데몬을 시작하고 중지합니다.

smbd 서비스를 사용하려면 **samba** 패키지를 설치합니다.

nmbd

이 서비스는 NetBIOS over IPv4 프로토콜을 사용하여 호스트 이름 및 IP 확인을 제공합니다. 이름 확인 외에도 **nmbd** 서비스를 사용하면 SMB 네트워크를 검색하여 도메인, 작업 그룹, 호스트, 파일 공유 및 프린터를 찾을 수 있습니다. 이를 위해 서비스는 이 정보를 브로드캐스트 클라이언트에 직접 보고하거나 로컬 또는 마스터 브라우저로 전달합니다. **nmb systemd** 서비스는 **nmbd** 데몬을 시작하고 중지합니다.

최신 SMB 네트워크는 DNS를 사용하여 클라이언트 및 IP 주소를 확인합니다. Kerberos의 경우 작동 중인 DNS 설정이 필요합니다.

nmbd 서비스를 사용하려면 **samba** 패키지를 설치합니다.

winbindd

이 서비스는 로컬 시스템에서 AD 또는 NT4 도메인 사용자 및 그룹을 사용할 수 있도록 NSS(Name Service Switch)에 대한 인터페이스를 제공합니다. 예를 들어, 도메인 사용자는 Samba 서버 또는 다른 로컬 서비스에 호스팅된 서비스에 대해 인증할 수 있습니다. **winbind systemd** 서비스는 **winbindd** 데몬을 시작하고 중지합니다.

Samba를 도메인 멤버로 설정하는 경우, **smbd** 서비스보다 먼저 **winbindd**를 시작해야 합니다. 그렇지 않으면 도메인 사용자 및 그룹을 로컬 시스템에서 사용할 수 없습니다.

winbindd 서비스를 사용하려면 **samba-winbind** 패키지를 설치합니다.



중요

Red Hat은 도메인 사용자 및 그룹을 로컬 시스템에 제공하기 위해 **winbindd** 서비스가 있는 서버로만 Samba 실행을 지원합니다. ACL(Windows Access Control List) 지원 및 NTLM(NT LAN Manager) 대체와 같은 특정 제한 사항으로 인해 SSSD는 지원되지 않습니다.

1.1.2. Samba 보안 서비스

/etc/samba/smb.conf 파일의 **[global]** 섹션에 있는 **security** 매개 변수는 Samba가 서비스에 연결하는 사용자를 인증하는 방법을 관리합니다. Samba를 설치하는 모드에 따라 매개 변수를 다른 값으로 설정해야 합니다.

AD 도메인 구성원에서는 **security = ads**를 설정

이 모드에서 Samba는 Kerberos를 사용하여 AD 사용자를 인증합니다.

Samba를 도메인 멤버로 설정하는 방법에 대한 자세한 내용은 [Setting up Samba as an AD domain member server](#)를 참조하십시오.

독립 실행형 서버에서 **security = user** 설정

이 모드에서 Samba는 로컬 데이터베이스를 사용하여 연결 사용자를 인증합니다.

Samba를 독립 실행형 서버로 설정하는 방법에 대한 자세한 내용은 [Samba를 독립 실행형 서버로 설정](#)을 참조하십시오.

NT4 PDC 또는 BDC에서 **security = user**를 설정

이 모드에서 Samba는 로컬 또는 LDAP 데이터베이스로 사용자를 인증합니다.

NT4 도메인 멤버에서 **security = domain** 설정

이 모드에서 Samba는 사용자를 NT4 PDC 또는 BDC에 연결하는 것을 인증합니다. AD 도메인 구성원에서는 이 모드를 사용할 수 없습니다.

Samba를 도메인 멤버로 설정하는 방법에 대한 자세한 내용은 [Setting up Samba as an AD domain member server](#)를 참조하십시오.

추가 리소스

- **smb.conf(5)** 도움말 페이지의 **security** 매개변수

1.1.3. Samba 서비스 및 Samba 클라이언트 유틸리티가 구성을 로드 및 다시 로드하는 시나리오

다음은 Samba 서비스 및 유틸리티가 설정을 로드하고 다시 로드하는 경우를 설명합니다.

- Samba 서비스는 설정을 다시 로드합니다.
 - 자동으로 3분마다
 - 예를 들어, manual 요청에서 **smbcontrol all reload-config** 명령을 실행하는 경우입니다.
- Samba 클라이언트 유틸리티는 처음 시작할 때만 구성을 읽습니다.

security 등의 특정 매개 변수를 사용하려면 **smb** 서비스를 다시 시작해야 하며 다시 로드하는 것만으로는 충분하지 않습니다.

추가 리소스

- **smb.conf(5)** 도움말 페이지의 구성 변경 내용 적용 방법
- **smbd(8)**, **nmbd(8)** 및 **winbindd(8)** 도움말 페이지

1.1.4. 안전한 방식으로 Samba 구성 편집

Samba 서비스는 3분마다 구성을 자동으로 다시 로드합니다. 이 절차에서는 **testparm** 유틸리티를 사용하여 구성을 확인하기 전에 서비스가 변경 사항을 다시 로드하지 못하도록 Samba 구성을 편집하는 방법을 설명합니다.

사전 요구 사항

- Samba가 설치되어 있어야 합니다.

절차

1. **/etc/samba/smb.conf** 파일의 사본을 생성합니다.

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

2. 복사한 파일을 편집하고 원하는 대로 변경합니다.

3. **/etc/samba/samba.conf.copy** 파일에서 구성을 확인합니다.

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

testparm이 오류를 보고하면 오류를 수정하고 명령을 다시 실행합니다.

4. **/etc/samba/smb.conf** 파일을 새 구성으로 재정의합니다.

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

5. Samba 서비스가 구성을 자동으로 다시 로드하거나 구성을 수동으로 다시 로드할 때까지 기다립니다.

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

추가 리소스

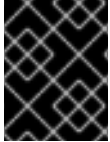
- Samba 서비스 및 Samba 클라이언트 유틸리티가 구성을 로드 및 다시 로드하는 시나리오

1.2. SAMBA 구성 확인

`/etc/samba/smb.conf` 파일을 업데이트할 때마다 Samba 구성을 확인하는 것이 좋습니다. 이 섹션에서는 이에 대해 자세히 설명합니다.

1.2.1. testparm 유틸리티를 사용하여 smb.conf 파일 확인

testparm 유틸리티는 `/etc/samba/smb.conf` 파일의 Samba 구성이 올바른지 확인합니다. 유틸리티는 잘못된 매개 변수와 값을 감지하지만 ID 매핑과 같은 잘못된 설정도 탐지합니다. **testparm**이 문제를 보고하지 않으면 Samba 서비스가 `/etc/samba/smb.conf` 파일을 로드합니다. **testparm**은 구성된 서비스가 사용 가능한지 또는 예상대로 작동하는지 확인할 수 없습니다.



중요

Red Hat은 이 파일을 수정한 후 **testparm**을 사용하여 `/etc/samba/smb.conf` 파일을 확인하는 것이 좋습니다.

사전 요구 사항

- Samba가 설치되어 있어야 합니다.
- `/etc/samba/smb.conf` 파일을 종료합니다.

절차

1. **testparm** 유틸리티를 **root** 사용자로 실행합니다.

```
# 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]
...
```

이전 예제 출력에서는 존재하지 않는 매개 변수와 잘못된 ID 매핑 구성을 보고합니다.

2. **testparm**이 구성에 잘못된 매개 변수, 값 또는 기타 오류를 보고하면 문제를 수정하고 유틸리티를 다시 실행합니다.

1.3. 독립 실행형 서버로 SAMBA 설정

Samba를 도메인의 멤버가 아닌 서버로 설정할 수 있습니다. 이 설치 모드에서 Samba는 중앙 DC가 아니라 로컬 데이터베이스로 사용자를 인증합니다. 또한 게스트 액세스를 활성화하여 사용자가 인증 없이 하나 이상의 서비스에 연결할 수 있습니다.

1.3.1. 독립 실행형 서버에 대한 서버 구성 설정

이 섹션에서는 Samba 독립 실행형 서버에 대한 서버 구성을 설정하는 방법을 설명합니다.

절차

1. **samba** 패키지를 설치합니다.

```
# dnf install samba
```

2. **/etc/samba/smb.conf** 파일을 편집하고 다음 매개변수를 설정합니다.

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

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

이 구성은 **Example-WG** 작업 그룹 내에서 **Server** 라는 독립 실행형 서버를 정의합니다. 또한 이 구성을 사용하면 최소 수준(1)에서 로깅할 수 있으며 로그 파일은 **/var/log/samba/** 디렉터리에 저장됩니다. Samba는 **log file** 매개 변수의 **%m** 매크로를 클라이언트 연결의 NetBIOS 이름으로 확장합니다. 이를 통해 각 클라이언트에 대한 개별 로그 파일이 활성화됩니다.

3. 선택적으로 파일 또는 프린터 공유를 구성합니다. 다음 내용을 참조하십시오.

- [POSIX ACL을 사용하는 공유 설정](#)
- [Windows ACL을 사용하는 공유 설정](#)
- [Samba를 인쇄 서버로 설정](#)

4. **/etc/samba/smb.conf** 파일을 확인합니다.

```
# testparm
```

5. 인증이 필요한 공유를 설정하면 사용자 계정을 생성합니다.
자세한 내용은 [로컬 사용자 계정 생성 및 활성화](#) 를 참조하십시오.
6. 필요한 포트를 열고 **firewall-cmd** 유틸리티를 사용하여 방화벽 구성을 다시 로드합니다.

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

7. **smb** 서비스를 활성화하고 시작합니다.

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

추가 리소스

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

1.3.2. 로컬 사용자 계정 생성 및 활성화

사용자가 공유에 연결할 때 인증할 수 있도록 하려면 운영 체제와 Samba 데이터베이스에서 Samba 호스트에 계정을 만들어야 합니다. Samba는 파일 시스템 개체 및 Samba 계정에서 연결 사용자를 인증하기 위해 운영 체제 계정이 ACL(액세스 제어 목록)의 유효성을 검사해야 합니다.

passdb backend = tdbsam 기본 설정을 사용하는 경우 Samba는 사용자 계정을 **/var/lib/samba/private/passdb.tdb** 데이터베이스에 저장합니다.

이 섹션의 절차에서는 **example**이라는 로컬 Samba 사용자를 만드는 방법을 설명합니다.

사전 요구 사항

- Samba는 독립 실행형 서버로 설치 및 구성됩니다.

절차

1. 운영 체제 계정을 생성합니다.

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

이 명령은 홈 디렉터리를 생성하지 않고 **example** 계정을 추가합니다. 계정이 Samba로 인증하는 데만 사용되는 경우 계정이 로컬에 로그인하지 못하도록 **/sbin/nologin** 명령을 쉘로 할당합니다.

2. 활성화하려면 암호를 운영 체제 계정으로 설정합니다.

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

Samba는 운영 체제 계정에 설정된 암호를 사용하여 인증하지 않습니다. 그러나 계정을 활성화하려면 암호를 설정해야 합니다. 계정이 비활성화되어 있으면 Samba는 이 사용자가 연결하면 액세스를 거부합니다.

3. 사용자를 Samba 데이터베이스에 추가하고 암호를 계정으로 설정합니다.

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

이 계정을 사용하여 Samba 공유에 연결할 때 인증하려면 이 암호를 사용합니다.

4. Samba 계정을 활성화합니다.

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

1.4. SAMBA ID 매핑 이해 및 구성

Windows 도메인은 고유한 SID(보안 식별자)를 통해 사용자와 그룹을 구분합니다. 그러나 Linux에는 사용자 및 그룹마다 고유한 UID와 GID가 필요합니다. 도메인 구성원으로 Samba를 실행하는 경우 **winbindd** 서비스는 도메인 사용자 및 그룹에 대한 정보를 운영 체제에 제공합니다.

winbindd 서비스를 활성화하여 Linux에 사용자와 그룹에 고유한 ID를 제공하려면 `/etc/samba/smb.conf` 파일에 ID 매핑을 구성해야 합니다.

- 로컬 데이터베이스(기본 도메인)
- Samba 서버가 멤버인 AD 또는 NT4 도메인
- 사용자가 이 Samba 서버의 리소스에 액세스할 수 있어야 하는 각 신뢰할 수 있는 도메인

Samba는 특정 구성에 대해 다양한 ID 매핑 백엔드를 제공합니다. 가장 자주 사용되는 백엔드는 다음과 같습니다.

백엔드	사용 사례
tdb	* 기본 도메인만 해당
ad	AD 도메인만 해당
rid	AD 및 NT4 도메인
autorid	AD, NT4 및 * 기본 도메인

1.4.1. Samba ID 범위 계획

Linux UID 및 GID를 AD에 저장하는지 여부 또는 이를 생성하도록 Samba를 구성하는지 여부에 관계없이 각 도메인 설정에는 다른 도메인과 겹치지 않아야 하는 고유한 ID 범위가 필요합니다.



주의

중복 ID 범위를 설정하면 Samba가 올바르게 작동하지 않습니다.

예 1.1. 고유 ID 범위

다음은 기본값(*), **AD-DOM**, 및 **TRUST-DOM** 도메인에 대한 인수 이외의 ID 매핑 범위를 보여줍니다.

```
[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
```

중요

도메인당 하나의 범위만 할당할 수 있습니다. 따라서 도메인 범위 사이에 충분한 공간을 남겨 두십시오. 이렇게 하면 나중에 도메인이 확장되는 경우 범위를 확장할 수 있습니다.

나중에 도메인에 다른 범위를 할당하면 이러한 사용자와 그룹이 이전에 만든 파일과 디렉터리의 소유권이 손실됩니다.

1.4.2. * 기본 도메인

도메인 환경에서는 다음 각각에 대해 하나의 ID 매핑 구성을 추가합니다.

- Samba 서버가 멤버인 도메인
- Samba 서버에 액세스할 수 있는 신뢰할 수 있는 각 도메인

그러나 다른 모든 개체에 대해 Samba는 기본 도메인의 ID를 할당합니다. 여기에는 다음이 포함됩니다.

- 로컬 Samba 사용자 및 그룹
- **BUILTIN\Administrators**와 같은 Samba 기본 제공 계정 및 그룹

중요

Samba가 올바르게 작동하도록 하려면 이 섹션에 설명된 대로 기본 도메인을 구성해야 합니다.

할당된 ID를 영구적으로 저장하려면 기본 도메인 백엔드에 쓸 수 있어야 합니다.

기본 도메인의 경우 다음 백엔드 중 하나를 사용할 수 있습니다.

tdb

tdb 백엔드를 사용하도록 기본 도메인을 구성하는 경우 나중에 생성될 오브젝트를 포함할 ID 범위를 설정하고 정의된 도메인 ID 매핑 구성의 일부가 아닌 ID 범위를 설정합니다.

예를 들어 **/etc/samba/smb.conf** 파일의 **[global]** 섹션에서 다음을 설정합니다.

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

자세한 내용은 [TDB ID 매핑 백엔드 사용](#) 을 참조하십시오.

autorid

autorid 백엔드를 사용하도록 기본 도메인을 구성하는 경우 도메인에 대한 ID 매핑 구성을 추가하는 것은 선택 사항입니다.

예를 들어 **/etc/samba/smb.conf** 파일의 **[global]** 섹션에서 다음을 설정합니다.

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

자세한 내용은 [Autorid ID 매핑 백엔드 사용](#) 을 참조하십시오.

1.4.3. tdb ID 매핑 백엔드 사용

winbindd 서비스는 기본적으로 쓰기 가능한 **tdb** ID 매핑 백엔드를 사용하여 SID(보안 식별자), UID 및 GID 매핑 테이블을 저장합니다. 여기에는 로컬 사용자, 그룹 및 기본 제공 주체가 포함됩니다.

이 백엔드는 * 기본 도메인에만 사용합니다. 예를 들어 다음과 같습니다.

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

추가 리소스

- [* 기본 도메인](#).

1.4.4. ad ID 매핑 백엔드 사용

이 섹션에서는 **ad** ID 매핑 백엔드를 사용하도록 Samba AD 멤버를 구성하는 방법을 설명합니다.

ad ID 매핑 백엔드는 읽기 전용 API를 구현하여 AD에서 계정 및 그룹 정보를 읽습니다. 이는 다음과 같은 이점을 제공합니다.

- 모든 사용자 및 그룹 설정은 AD에 중앙에 저장됩니다.
- 이 백엔드를 사용하는 모든 Samba 서버에서 사용자 및 그룹 ID가 일관되게 유지됩니다.
- ID는 손상될 수 있는 로컬 데이터베이스에 저장되지 않으므로 파일 소유권을 분실할 수 없습니다.



참고

ad ID 매핑 백엔드는 단방향 트러스트가 있는 Active Directory 도메인을 지원하지 않습니다. 단방향 트러스트를 사용하여 Active Directory에 도메인 멤버를 구성하는 경우 **tdb**, **rid**, **autorid**와 같은 ID 매핑 백엔드 중 하나를 대신 사용합니다.

에드혹 백엔드는 AD에서 다음 속성을 읽습니다.

AD 속성 이름	오브젝트 유형	매핑 대상
sAMAccountName	사용자 및 그룹	사용자 또는 그룹 이름 (오브젝트에 따라)
uidNumber	사용자	사용자 ID(UID)
gidNumber	그룹	그룹 ID(GID)
loginShell [a]	사용자	사용자 셸의 경로
unixHomeDirectory [a]	사용자	사용자의 홈 디렉터리 경로

AD 속성 이름	오브젝트 유형	매핑 대상
primaryGroupID ^[b]	사용자	기본 그룹 ID
[a] Samba는 idmap config DOMAIN:unix_nss_info = yes 를 설정하는 경우에만 이 속성을 읽습니다.		
[b] Samba는 idmap config DOMAIN:unix_primary_group = yes 를 설정하는 경우에만 이 속성을 읽습니다.		

사전 요구 사항

- 사용자와 그룹 모두 AD에 고유한 ID를 설정해야 하며 ID는 **/etc/samba/smb.conf** 파일에 구성된 범위 내에 있어야 합니다. 범위를 벗어나는 ID가 있는 개체는 Samba 서버에서 사용할 수 없습니다.
- 사용자와 그룹은 AD에서 모든 필수 속성을 설정해야 합니다. 필수 속성이 없으면 Samba 서버에서 사용자 또는 그룹을 사용할 수 없습니다. 필수 속성은 구성에 따라 다릅니다.. 전제 조건
- Samba가 설치되어 있어야 합니다.
- ID 매핑을 제외한 Samba 구성이 **/etc/samba/smb.conf** 파일에 있습니다.

절차

1. **/etc/samba/smb.conf** 파일에서 **[global]** 섹션을 편집합니다.

- a. 없는 경우 기본 도메인 (*)의 ID 매핑 구성을 추가합니다. 예를 들어 다음과 같습니다.

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

- b. AD 도메인의 **ad** ID 매핑 백엔드를 활성화합니다.

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

- c. AD 도메인의 사용자와 그룹에 할당된 ID 범위를 설정합니다. 예를 들어 다음과 같습니다.

```
idmap config DOMAIN : range = 2000000-29999999
```



중요

범위는 이 서버의 다른 도메인 구성과 겹치지 않아야 합니다. 또한 범위는 나중에 할당되는 모든 ID를 포함할 만큼 충분히 커야 합니다. 자세한 내용은 [Planning Samba ID 범위](#)를 참조하십시오.

- d. AD에서 속성을 읽을 때 Samba가 [RFC 2307](#) 스키마를 사용하도록 설정합니다.

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

- e. Samba가 해당 AD 속성에서 로그인 셸 및 사용자 홈 디렉터리 경로를 읽을 수 있도록 하려면 다음을 설정합니다.

■

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

또는 모든 사용자에게 적용되는 균일한 도메인 전체 홈 디렉터리 경로 및 로그인 셸을 설정할 수 있습니다. 예를 들어 다음과 같습니다.

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

- f. 기본적으로 Samba는 사용자 오브젝트의 **primaryGroupID** 속성을 Linux의 사용자 기본 그룹으로 사용합니다. 또는 대신 **gidNumber** 특성에 설정된 값을 사용하도록 Samba를 구성할 수 있습니다.

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

2. **/etc/samba/smb.conf** 파일을 확인합니다.

```
# testparm
```

3. Samba 구성을 다시 로드합니다.

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

추가 리소스

- * 기본 도메인
- **smb.conf(5)** 및 **idmap_ad(8)** man 페이지
- **smb.conf(5)** 도움말 페이지의 **VARIABLE SUBSTITUTIONS** 섹션

1.4.5. 제거 ID 매핑 백엔드 사용

이 섹션에서는 **rid** ID 매핑 백엔드를 사용하도록 Samba 도메인 멤버를 구성하는 방법에 대해 설명합니다.

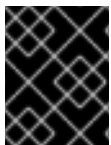
Samba는 Windows SID의 상대 식별자(RID)를 사용하여 Red Hat Enterprise Linux에서 ID를 생성할 수 있습니다.



참고

RID는 SID의 마지막 부분입니다. 예를 들어 사용자의 SID가 **S-1-5-21-5421822485-11512151-421485315-30014** 이면 **30014** 가 해당하는 RID입니다.

rid ID 매핑 백엔드는 AD 및 NT4 도메인의 알고리즘 매핑 체계를 기반으로 계정과 그룹 정보를 계산하는 읽기 전용 API를 구현합니다. 백엔드를 구성할 때 **idmap config *DOMAIN* : range** 매개변수에서 가장 낮고 가장 높은 RID를 설정해야 합니다. Samba는 이 매개 변수에 설정된 것보다 낮은 RID를 가진 사용자 또는 그룹을 매핑하지 않습니다.



중요

읽기 전용 백엔드인 **rid**는 **BUILTIN** 그룹과 같은 새 ID를 할당할 수 없습니다. 따라서 * 기본 도메인에 이 백엔드를 사용하지 마십시오.

remove 백엔드를 사용할 수 있는 이점

- 구성된 범위 내에 RID가 있는 모든 도메인 사용자 및 그룹을 도메인 구성원에서 자동으로 사용할 수 있습니다.
- ID, 홈 디렉터리 및 로그인 셸을 수동으로 할당할 필요는 없습니다.

remove 백엔드를 사용하는 단점

- 모든 도메인 사용자는 동일한 로그인 셸 및 홈 디렉터리가 할당됩니다. 그러나 변수를 사용할 수 있습니다.
- 사용자 및 그룹 ID는 모두 동일한 ID 범위 설정으로 **remove 백엔드를 사용하는 경우에만 Samba** 도메인 구성원에서 동일합니다.
- 도메인 멤버는 개별 사용자 또는 그룹을 제외할 수 없습니다. 구성된 범위 밖의 사용자와 그룹만 제외됩니다.
- formula에 따라 **winbindd** 서비스는 ID를 계산하기 위해 사용하는 표현식에 따라 다른 도메인의 오브젝트에 RID가 동일한 경우 다중 도메인 환경에서 중복 ID가 발생할 수 있습니다.

사전 요구 사항

- Samba가 설치되어 있어야 합니다.
- ID 매핑을 제외한 Samba 구성이 **/etc/samba/smb.conf** 파일에 있습니다.

절차

1. **/etc/samba/smb.conf** 파일에서 **[global]** 섹션을 편집합니다.

- a. 없는 경우 기본 도메인 (*)의 ID 매핑 구성을 추가합니다. 예를 들어 다음과 같습니다.

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

- b. 도메인에 대한 ID 매핑 제거를 활성화합니다.

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

- c. 앞으로 할당할 모든 RID를 포함할 수 있을 만큼 큰 범위를 설정합니다. 예를 들어 다음과 같습니다.

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

Samba는 이 도메인의 RID가 범위에 속하지 않는 사용자 및 그룹을 무시합니다.



중요

범위는 이 서버의 다른 도메인 구성과 겹치지 않아야 합니다. 또한 범위는 나중에 할당되는 모든 ID를 포함할 만큼 충분히 커야 합니다. 자세한 내용은 [Planning Samba ID 범위](#)를 참조하십시오.

- d. 매핑된 모든 사용자에게 할당될 셸 및 홈 디렉터리 경로를 설정합니다. 예를 들어 다음과 같습니다.

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

2. `/etc/samba/smb.conf` 파일을 확인합니다.

```
# testparm
```

3. Samba 구성을 다시 로드합니다.

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

추가 리소스

- * 기본 도메인
- `smb.conf(5)` 도움말 페이지의 **VARIABLE SUBSTITUTIONS** 섹션
- RID의 로컬 ID 계산에서 `idmap_rid(8)` 매뉴얼 페이지를 참조하십시오.

1.4.6. 자동 덮어쓰기 ID 매핑 백엔드 사용

이 섹션에서는 **autorid** ID 매핑 백엔드를 사용하도록 Samba 도메인 멤버를 구성하는 방법에 대해 설명합니다.

auto rid 백엔드는 remove ID 매핑 백엔드와 유사하게 작동하지만 다른 도메인의 ID를 자동으로 할당할 수 있습니다. 이를 통해 다음과 같은 상황에서 **autorid** 백엔드를 사용할 수 있습니다.

- * 기본 도메인의 경우에만
- * 기본 도메인 및 추가 도메인의 경우 각 추가 도메인에 대한 ID 매핑 구성을 생성하지 않고도 * 기본 도메인 및 추가 도메인의 경우
- 특정 도메인 전용



참고

기본 도메인에 **autorid**를 사용하는 경우 도메인에 대한 ID 매핑 구성을 추가하는 것은 선택 사항입니다.

이 섹션의 일부는 Samba Wiki에 게시된 [idmap 구성](#) 문서에서 채택되었습니다. 라이선스: [CC BY 4.0](#). 작성자 및 기여자: Wiki 페이지의 [기록](#) 탭을 참조하십시오.

autorid 백엔드를 사용할 경우의 이점

- 구성된 범위 내에 계산된 UID 및 GID가 있는 모든 도메인 사용자 및 그룹은 도메인 멤버에서 자동으로 사용할 수 있습니다.
- ID, 홈 디렉터리 및 로그인 셸을 수동으로 할당할 필요는 없습니다.
- 다중 도메인 환경의 여러 오브젝트에 동일한 RID가 있는 경우에도 중복 ID가 없습니다.

단점

- 사용자 및 그룹 ID는 Samba 도메인 구성원에서 동일하지 않습니다.
- 모든 도메인 사용자는 동일한 로그인 셸 및 홈 디렉터리가 할당됩니다. 그러나 변수를 사용할 수 있습니다.
- 도메인 멤버는 개별 사용자 또는 그룹을 제외할 수 없습니다. 계산된 UID 또는 GID가 구성된 범위 외부에 있는 사용자 및 그룹만 제외됩니다.

사전 요구 사항

- Samba가 설치되어 있어야 합니다.
- ID 매핑을 제외한 Samba 구성이 **/etc/samba/smb.conf** 파일에 있습니다.

절차

1. **/etc/samba/smb.conf** 파일에서 **[global]** 섹션을 편집합니다.

- a. * 기본 도메인에 대해 자동 ID 매핑 백엔드를 활성화합니다.

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

- b. 기존 및 향후 오브젝트의 ID를 모두 할당할 수 있을 만큼 큰 범위를 설정합니다. 예를 들어 다음과 같습니다.

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

Samba는 이 도메인에서 계산된 ID가 범위에 속하지 않는 사용자 및 그룹을 무시합니다.



주의

범위를 설정하고 Samba가 이를 사용하기 시작하면 범위의 상한만 늘릴 수 있습니다. 범위에 대한 다른 모든 변경으로 인해 새 ID 할당으로 인해 파일 소유권이 손실될 수 있습니다.

- c. 선택적으로 범위 크기를 설정합니다. 예를 들어 다음과 같습니다.

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

Samba는 **idmap config * : range** 매개변수에 설정된 범위의 모든 ID를 취할 때까지 각 도메인의 오브젝트에 대해 이 수의 연속 ID를 할당합니다.



참고

rangesize를 설정하는 경우 그에 따라 범위를 조정해야 합니다. 범위는 여러 범위 크기여야 합니다.

- d. 매핑된 모든 사용자에게 할당될 셸 및 홈 디렉터리 경로를 설정합니다. 예를 들어 다음과 같습니다.

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

- e. 필요한 경우 도메인에 대한 ID 매핑 구성을 추가합니다. 개별 도메인에 대한 구성이 없는 경우 Samba는 이전에 구성한 * 기본 도메인 **의 자동 백엔드** 설정을 사용하여 ID를 계산합니다.



중요

범위는 이 서버의 다른 도메인 구성과 겹치지 않아야 합니다. 또한 범위는 나중에 할당되는 모든 ID를 포함할 만큼 충분히 커야 합니다. 자세한 내용은 [Planning Samba ID 범위](#)를 참조하십시오.

2. **/etc/samba/smb.conf** 파일을 확인합니다.

```
# testparm
```

3. Samba 구성을 다시 로드합니다.

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

추가 리소스

- **idmap_autorid(8)** 매뉴얼 페이지의 **MAPPING FORMULAS** 섹션
- **idmap_autorid(8)** 매뉴얼 페이지의 **rangesize** 매개변수 설명
- **smb.conf(5)** 도움말 페이지의 **VARIABLE SUBSTITUTIONS** 섹션

1.5. AD 도메인 멤버 서버로 SAMBA 설정

AD 또는 NT4 도메인을 실행 중인 경우 Samba를 사용하여 Red Hat Enterprise Linux 서버를 도메인에 멤버로 추가하여 다음을 얻을 수 있습니다.

- 다른 도메인 구성원의 도메인 리소스에 액세스
- **sshd**와 같은 로컬 서비스에 도메인 사용자를 인증합니다
- 공유 디렉토리 및 서버에서 호스트된 프린터 파일 역할을 파일 및 인쇄 서버

1.5.1. AD 도메인에 RHEL 시스템 연결

Samba Winbind는 RHEL(Red Hat Enterprise Linux) 시스템을 Active Directory(AD)와 연결하기 위한 SSSD(System Security Services Daemon) 대안입니다. 이 섹션에서는 **realmd**를 사용하여 Samba Winbind를 구성하여 RHEL 시스템을 AD 도메인에 연결하는 방법을 설명합니다.

절차

1. AD에 Kerberos 인증을 위한 더 이상 사용되지 않는 RC4 암호화 유형이 필요한 경우 RHEL에서 이러한 암호에 대한 지원을 활성화합니다.

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

2. 다음 패키지를 설치합니다.

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

3. 도메인 구성원에서 디렉터리 또는 프린터를 공유하려면 **samba** 패키지를 설치합니다.

```
# dnf install samba
```

4. 기존 **/etc/samba/smb.conf** Samba 구성 파일을 백업합니다.

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

5. 도메인에 가입합니다. 예를 들어 **ad.example.com**이라는 도메인에 가입하려면 다음을 수행합니다.

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

이전 명령을 사용하면 **realm** 유틸리티가 자동으로 수행됩니다.

- **ad.example.com** 도메인 멤버십에 대한 **/etc/samba/smb.conf** 파일을 만듭니다.
 - 사용자 및 그룹 조회에 대한 **winbind** 모듈을 **/etc/nsswitch.conf** 파일에 추가합니다.
 - **/etc/pam.d/** 디렉토리에서 PAM(Pluggable Authentication Module) 구성 파일을 업데이트합니다.
 - **winbind** 서비스를 시작하고 시스템이 부팅될 때 서비스가 시작됩니다.
6. 선택적으로 **/etc/samba/smb.conf** 파일에서 대체 ID 매핑 백엔드 또는 사용자 지정된 ID 매핑 설정을 설정합니다.

자세한 내용은 [Samba ID 매핑 이해 및 구성](#)을 참조하십시오.

1. **winbind** 서비스가 실행 중인지 확인합니다.

```
# systemctl status winbind
```

```
...
```

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



중요

Samba를 활성화하여 도메인 사용자 및 그룹 정보를 쿼리하려면 **smb** 를 시작하기 전에 **winbind** 서비스를 실행해야 합니다.

2. 디렉터리 및 프린터를 공유하는 **samba** 패키지를 설치한 경우 **smb** 서비스를 활성화하고 시작합니다.

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

3. 선택적으로 Active Directory에 로컬 로그인을 인증하는 경우 **winbind_krb5_localauth** 플러그인을 활성화합니다. [MIT Kerberos 용으로 로컬 권한 부여 플러그인 사용](#)을 참조하십시오.

검증 단계

1. AD 도메인의 AD 관리자 계정과 같은 AD 사용자의 세부 정보를 표시합니다.

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

2. AD 도메인에서 도메인 사용자 그룹의 멤버를 쿼리합니다.

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

3. 선택적으로 파일 및 디렉터리에 대한 권한을 설정할 때 도메인 사용자와 그룹을 사용할 수 있는지 확인합니다. 예를 들어 **/srv/samba/example.txt** 파일의 소유자를 **AD\administrator**로 설정하고 그룹을 **AD\Domain Users**로 설정하려면 다음을 수행합니다.

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

4. Kerberos 인증이 예상대로 작동하는지 확인합니다.

- a. AD 도메인 멤버에서 **administrator@AD.EXAMPLE.COM** 주체의 티켓을 받습니다.

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

- b. 캐시된 Kerberos 티켓을 표시합니다.

```
# 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. 사용 가능한 도메인을 표시합니다.

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

추가 리소스

- 더 이상 사용되지 않는 RC4 암호를 사용하지 않으려면 AD에서 AES 암호화 유형을 활성화할 수 있습니다. VDD [K를 사용하여 Active Directory에서 AES 암호화 유형 활성화를 참조하십시오](#). 이는 AD의 다른 서비스에 영향을 줄 수 있습니다.
- **realm(8)** 매뉴얼 페이지

1.5.2. MIT Kerberos용 로컬 인증 플러그인 사용

winbind 서비스는 Active Directory 사용자를 도메인 구성원에게 제공합니다. 관리자는 특정 상황에서 도메인 사용자가 도메인 구성원에서 실행 중인 SSH 서버와 같은 로컬 서비스에 인증할 수 있도록 설정하려고 합니다. Kerberos를 사용하여 도메인 사용자를 인증할 때 **winbind_krb5_localauth** 플러그인을 활성화하여 **winbind** 서비스를 통해 Kerberos 주체를 Active Directory 계정에 올바르게 매핑하십시오.

예를 들어 Active Directory 사용자의 **sAMAccountName** 속성이 **EXAMPLE** 으로 설정되어 있고 사용자가 사용자 이름 소문자로 로그인하려고 하면 Kerberos는 대문자로 사용자 이름을 반환합니다. 이로 인해 항목이 일치하지 않고 인증이 실패합니다.

winbind_krb5_localauth 플러그인을 사용하면 계정 이름이 올바르게 매핑됩니다. 이는 GSSAPI 인증에만 적용되며 초기 티켓 부여 티켓(TGT)은 받지 않습니다.

사전 요구 사항

- Samba는 Active Directory의 멤버로 구성되어 있습니다.
- Red Hat Enterprise Linux는 Active Directory에 대한 로그인 시도를 인증합니다.
- **winbind** 서비스가 실행 중입니다.

절차

/etc/krb5.conf 파일을 편집하고 다음 섹션을 추가합니다.

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

추가 리소스

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

1.6. IDM 도메인 멤버에서 SAMBA 설정

이 섹션에서는 Red Hat IdM(Identity Management) 도메인에 연결된 호스트에서 Samba를 설정하는 방법을 설명합니다. IdM의 사용자 및 또한 신뢰할 수 있는 AD(Active Directory) 도메인에서 사용 가능한 경우 Samba에서 제공하는 공유 및 프린터 서비스에 액세스할 수 있습니다.

중요

IdM 도메인 멤버에서 Samba를 사용하는 것은 지원되지 않는 기술 프리뷰 기능이며 특정 제한 사항이 포함되어 있습니다. 예를 들어, IdM 신뢰 컨트롤러는 Active Directory 글로벌 카탈로그 서비스를 지원하지 않으며 DMCE/원격 프로시저 호출(DCE/RPC) 프로토콜을 사용하여 IdM 그룹 해결을 지원하지 않습니다. 결과적으로 AD 사용자는 다른 IdM 클라이언트에 로그인할 때 IdM 클라이언트에서 호스팅되는 Samba 공유 및 프린터에만 액세스할 수 있습니다. Windows 시스템에 로그인한 AD 사용자는 IdM 도메인 멤버에서 호스팅되는 Samba 공유에 액세스할 수 없습니다.

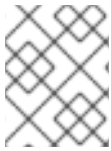
IdM 도메인 구성원에 Samba를 배포하는 고객은 Red Hat에 피드백을 제공하는 것이 좋습니다.

사전 요구 사항

- 호스트는 IdM 도메인에 클라이언트로 결합됩니다.

1.6.1. 도메인 멤버에 Samba를 설치하기 위해 IdM 도메인 준비

IdM 클라이언트에 Samba를 설정하려면 IdM 서버에서 **ipa-adtrust-install** 유틸리티를 사용하여 IdM 도메인을 준비해야 합니다.



참고

ipa-adtrust-install 명령을 실행하는 시스템은 자동으로 AD 신뢰 컨트롤러가 됩니다. 그러나 IdM 서버에서 **ipa-adtrust-install** 을 한 번만 실행해야 합니다.

사전 요구 사항

- IdM 서버가 설치되어 있어야 합니다.
- 패키지를 설치하고 IdM 서비스를 다시 시작하려면 루트 권한이 필요합니다.

절차

1. 필수 패키지를 설치합니다.

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

2. IdM 관리자로 인증합니다.

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

3. **ipa-adtrust-install** 유틸리티를 실행합니다.

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

IdM이 통합된 DNS 서버와 함께 설치된 경우 DNS 서비스 레코드가 자동으로 생성됩니다.

통합된 DNS 서버 없이 IdM을 설치한 경우, **ipa-adtrust-install** 은 DNS에 수동으로 추가해야 하는 서비스 레코드 목록을 인쇄합니다.

4. 스크립트에서 **/etc/samba/smb.conf**가 이미 존재하고 다시 작성됨을 묻는 메시지를 표시합니다.

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

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

5. 스크립트에서 이전 Linux 클라이언트가 신뢰할 수 있는 사용자로 작업할 수 있는 호환성 플러그인인 **slapi-nis** 플러그인을 구성하도록 프롬프트를 표시합니다.

```
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. 메시지가 표시되면 IdM 도메인의 NetBIOS 이름을 입력하거나 **Enter** 를 눌러 제안된 이름을 수락합니다.

```
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. SID 생성 작업을 실행하여 기존 사용자의 SID를 생성하라는 메시지가 표시됩니다.

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

이는 리소스 집약적인 작업이므로 사용자가 많은 경우 한 번에 이 작업을 실행할 수 있습니다.

8. (선택 사항) 기본적으로 Dynamic RPC 포트 범위는 Windows Server 2008 이상에서는 **49152-65535** 로 정의됩니다. 환경에 대해 다른 Dynamic RPC 포트 범위를 정의해야 하는 경우 다른 포트를 사용하도록 Samba를 구성하고 방화벽 설정에서 해당 포트를 엽니다. 다음 예제에서는 포트 범위를 **55000-65000**으로 설정합니다.

```
[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. **ipa** 서비스를 다시 시작하십시오.

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

10. **smbclient** 유틸리티를 사용하여 Samba가 IdM 측에서 Kerberos 인증에 응답하는지 확인합니다.

```
[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. GPO를 사용하여 Active Directory의 AES 암호화 유형 활성화

이 섹션에서는 그룹 정책 개체(GPO)를 사용하여 AD(Active Directory)의 AES 암호화 유형을 활성화하는 방법을 설명합니다. IdM 클라이언트에서 Samba 서버를 실행하는 등의 RHEL의 특정 기능에는 이 암호화 유형이 필요합니다.

RHEL은 더 이상 약한 DES 및 RC4 암호화 유형을 지원하지 않습니다.

사전 요구 사항

- 그룹 정책을 편집할 수 있는 사용자로 AD에 로그인되어 있습니다.
- 그룹 정책 관리 콘솔이 컴퓨터에 설치되어 있습니다.

절차

1. 그룹 정책 관리 콘솔을 엽니다.
2. 기본 도메인 정책에서 마우스 오른쪽 버튼으로 클릭하여 편집을 선택합니다. 그룹 정책 관리 편집기가 열립니다.
3. 컴퓨터 구성 → 정책 → **Windows** 설정 → 보안 설정 → 로컬 정책 → 보안 옵션으로 이동합니다.
4. 네트워크 보안 을 두 번 클릭합니다. **Kerberos** 정책에 허용되는 암호화 유형을 구성합니다.
5. **AES256_HMAC_SHA1**을 선택하고 선택적으로 **Future** 암호화 유형을 선택합니다.
6. **OK**를 클릭합니다.
7. 그룹 정책 관리 편집기 를 닫습니다.
8. 기본 도메인 컨트롤러 정책에 대한 단계를 반복합니다.
9. Windows 도메인 컨트롤러(DC)가 그룹 정책을 자동으로 적용할 때까지 기다립니다. 또는 DC에서 GPO를 수동으로 적용하려면 관리자 권한이 있는 계정을 사용하여 다음 명령을 입력합니다.

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

1.6.3. IdM 클라이언트에 Samba 서버 설치 및 구성

이 섹션에서는 IdM 도메인에 등록된 클라이언트에 Samba를 설치하고 구성하는 방법을 설명합니다.

사전 요구 사항

- IdM 서버와 클라이언트 모두 RHEL 9.0 이상에서 실행해야 합니다.
- IdM 도메인은 도메인 멤버에 Samba를 설치하기 위해 IdM 도메인 준비에 설명된 대로 준비됩니다.
- IdM에 AD로 신뢰가 구성된 경우 Kerberos에 대해 AES 암호화 유형을 활성화합니다. 예를 들어 GPO(그룹 정책 오브젝트)를 사용하여 AES 암호화 유형을 활성화합니다. 자세한 내용은 VDDK를 사용하여 Active Directory에서 AES 암호화 활성화를 참조하십시오 .

절차

1. **ipa-client-samba** 패키지를 설치합니다.

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

2. **ipa-client-samba** 유틸리티를 사용하여 클라이언트를 준비하고 초기 Samba 구성을 생성합니다.

```
[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

- 기본적으로 **ipa-client-915**는 사용자가 연결할 때 사용자의 홈 디렉터리를 동적으로 공유하는 `/etc/controlPlane/dpdk.conf` 파일에 **[homes]** 섹션을 자동으로 추가합니다. 이 서버에 홈 디렉터리가 없거나 공유하려는 경우 `/etc/samba/smb.conf`에서 다음 행을 제거하십시오.

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

- 디렉터리와 프린터를 공유합니다. 자세한 내용은 다음을 참조하십시오.

- [POSIX ACL을 사용하는 Samba 파일 공유 설정](#)
- [Windows ACL을 사용하는 공유 설정](#)
- [인쇄 서버로 Samba 설정](#)

- 로컬 방화벽에서 Samba 클라이언트에 필요한 포트를 엽니다.

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

- smb** 및 **winbind** 서비스를 활성화하고 시작합니다.

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

검증 단계

samba-client 패키지가 설치된 다른 IdM 도메인 멤버에서 다음 확인 단계를 실행합니다.

- Kerberos 인증을 사용하여 Samba 서버의 공유를 나열합니다.

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

Sharename	Type	Comment
<i>example</i>	Disk	
IPC\$	IPC	IPC Service (Samba 4.15.2)

...

추가 리소스

- **ipa-client-cnv(1)** 매뉴얼 페이지.

1.6.4. IdM이 새 도메인을 신뢰하는 경우 ID 매핑 구성 수동 추가

Samba에는 사용자가 리소스에 액세스하는 각 도메인에 대한 ID 매핑 구성이 필요합니다. IdM 클라이언트에서 실행 중인 기존 Samba 서버에서 관리자가 Active Directory(AD) 도메인에 새 신뢰를 추가한 후 ID 매핑 구성을 수동으로 추가해야 합니다.

사전 요구 사항

- IdM 클라이언트에 Samba가 구성되어 있습니다. 그 후 IdM에 새로운 신뢰가 추가되었습니다.
- Kerberos에 대한 DES 및 RC4 암호화 유형은 신뢰할 수 있는 AD 도메인에서 비활성화해야 합니다. 보안상의 이유로 RHEL 9는 이러한 약한 암호화 유형을 지원하지 않습니다.

절차

1. 호스트의 keytab을 사용하여 인증합니다.

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

2. **ipa idrange-find** 명령을 사용하여 새 도메인의 기본 ID와 ID 범위 크기를 모두 표시합니다. 예를 들어 다음 명령은 **ad.example.com** 도메인의 값을 표시합니다.

```
[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
-----
```

다음 단계에서 **ipabaseid** 및 **ipaidrangesize** 속성의 값이 필요합니다.

3. 사용 가능한 최고 ID를 계산하려면 다음 공식을 사용합니다.

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

이전 단계의 값을 사용하면 **ad.example.com** 도메인에서 사용 가능한 가장 높은 ID는 **1918599999** ($1918400000 + 200000 - 1$)입니다.

4. **/etc/samba/smb.conf** 파일을 편집하고 도메인의 ID 매핑 구성을 **[global]** 섹션에 추가합니다.

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

ipabaseid 속성의 값을 가장 낮은 값으로 지정하고 이전 단계에서 계산된 값을 범위의 가장 높은 값으로 지정합니다.

5. **smb** 및 **winbind** 서비스를 다시 시작합니다.

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

검증 단계

- Kerberos 인증을 사용하여 Samba 서버의 공유를 나열합니다.

```
$ 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. 추가 리소스

- [Identity Management 클라이언트 설치](#)를 참조하십시오.

1.7. POSIX ACL을 사용하는 SAMBA 파일 공유 설정

Linux 서비스로 Samba는 POSIX ACL과 공유를 지원합니다. **chmod**와 같은 유틸리티를 사용하여 Samba 서버에서 권한을 로컬로 관리할 수 있습니다. 확장 속성을 지원하는 파일 시스템에 공유가 저장된 경우 여러 사용자 및 그룹을 사용하여 ACL을 정의할 수 있습니다.



참고

대신 세분화된 Windows ACL을 사용해야 하는 경우 Windows ACL을 [사용하는 공유 설정 단원](#)을 참조하십시오.

이 섹션의 일부는 Samba440에 게시된 [POSIX ACL 설명서를 사용하여 공유](#) 설정에서 채택되었습니다. 라이선스: [CC BY 4.0](#). 작성자 및 기여자: Wiki 페이지의 [기록](#) 탭을 참조하십시오.

1.7.1. POSIX ACL을 사용하는 공유 추가

이 섹션에서는 **/srv/controlPlane/example** / 디렉터리의 콘텐츠를 제공하고 **POSIX ACL**을 사용하는 **example**이라는 공유를 만드는 방법을 설명합니다.

사전 요구 사항

Samba는 다음 모드 중 하나로 설정되었습니다.

- [독립 실행형 서버](#)
- [도메인 멤버](#)

절차

1. 폴더가 없는 경우 해당 폴더를 생성합니다. 예를 들어 다음과 같습니다.

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

2. **enforcing** 모드에서 SELinux를 실행하는 경우 디렉터리에 **samba_share_t** 컨텍스트를 설정합니다.

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

3. 디렉터리에 파일 시스템 ACL을 설정합니다. 자세한 내용은 다음을 참조하십시오.

- [POSIX ACL을 사용하는 Samba 공유에서 표준 ACL 설정](#)
- [POSIX ACL을 사용하는 공유에 확장 ACL 설정](#).

4. 예제 공유를 **/etc/samba/smb.conf** 파일에 추가합니다. 예를 들어 공유 쓰기 기능을 추가하려면 다음을 수행합니다.

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



참고

파일 시스템 ACL에 관계없이 **read only = no**를 설정하지 않으면 Samba는 읽기 전용 모드로 디렉토리를 공유합니다.

5. **/etc/samba/smb.conf** 파일을 확인합니다.

```
# testparm
```

6. 필요한 포트를 열고 **firewall-cmd** 유틸리티를 사용하여 방화벽 구성을 다시 로드합니다.

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

7. **smb** 서비스를 다시 시작하십시오.

```
# systemctl restart smb
```

1.7.2. POSIX ACL을 사용하는 Samba 공유에 표준 Linux ACL 설정

Linux의 표준 ACL은 하나의 소유자, 한 그룹, 기타 정의되지 않은 모든 사용자에게 대한 권한을 설정합니다. **chown**, **chgrp** 및 **chmod** 유틸리티를 사용하여 ACL을 업데이트할 수 있습니다. 정확한 제어가 필요한 경우 보다 복잡한 POSIX ACL을 사용하십시오.

[POSIX ACL을 사용하는 Samba 공유에서 확장 ACL 설정](#).

다음 절차에서는 **/srv/initiate/example/** 디렉터리의 소유자를 **root** 사용자에게 설정하고, **Domain Users** 그룹에 읽기 및 쓰기 권한을 부여하고, 기타 모든 사용자에게 대한 액세스를 거부합니다.

사전 요구 사항

- ACL을 설정하려는 Samba 공유가 있습니다.

절차

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



참고

디렉토리에 set-group-ID(SGID) 비트를 활성화하면 새 디렉토리 항목을 생성한 사용자의 기본 그룹으로 설정하는 일반적인 동작이 아니라 모든 새 파일 및 하위 디렉토리에 대한 기본 그룹이 자동으로 설정됩니다.

추가 리소스

- **chown(1)** 및 **kernel(1)** 매뉴얼 페이지

1.7.3. POSIX ACL을 사용하는 Samba 공유에서 확장 ACL 설정

공유 디렉터리가 에 저장된 파일 시스템이 확장 ACL을 지원하는 경우 이를 사용하여 복잡한 권한을 설정할 수 있습니다. 확장 ACL에는 여러 사용자와 그룹에 대한 권한이 포함될 수 있습니다.

확장된 POSIX ACL을 사용하면 여러 사용자 및 그룹으로 복잡한 ACL을 구성할 수 있습니다. 그러나 다음 권한만 설정할 수 있습니다.

- 액세스 권한 없음
- 읽기 액세스
- 쓰기 액세스
- 완전 제어

Create folder / append data와 같은 세분화된 Windows 권한이 필요한 경우 Windows ACL을 사용하도록 공유를 구성합니다.

[Windows ACL을 사용하는 공유](#) 설정을 참조하십시오.

다음 절차는 공유에서 확장 ACL을 활성화하는 방법을 보여줍니다. 확장 ACL 설정에 대한 예제도 포함되어 있습니다.

사전 요구 사항

- ACL을 설정하려는 Samba 공유가 있습니다.

절차

1. **/etc/controlPlane/tekton.conf** 파일의 **share** 섹션에서 다음 매개 변수를 활성화하여 확장 ACL에 대한 ACL 상속을 활성화합니다.

```
inherit acls = yes
```

자세한 내용은 **smb.conf(5)** 도움말 페이지의 매개 변수 설명을 참조하십시오.

2. **smb** 서비스를 다시 시작하십시오.

```
# systemctl restart smb
```

3. 디렉터리에 ACL을 설정합니다. 예를 들어 다음과 같습니다.

예 1.2. 확장 ACL 설정

다음 절차에서는 **Domain Admins** 그룹에 대한 읽기, 쓰기, 실행 권한, **Domain Users** 그룹에 대한 읽기 및 실행 권한을 설정하고, **/srv/tekton/example/** 디렉터리의 다른 모든 사용자에게 대한 액세스를 거부합니다.

1. 사용자 계정의 기본 그룹에 대한 자동 분리 권한을 비활성화합니다.

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

디렉터리의 기본 그룹은 동적 **CREATOR GROUP** principal에 추가로 매핑됩니다. Samba 공유에 확장 POSIX ACL을 사용하면 이 보안 주체가 자동으로 추가되고 제거할 수 없습니다.

2. 디렉터리에 대한 권한을 설정합니다.

- a. **Domain Admins** 그룹에 읽기, 쓰기 및 실행 권한을 부여합니다.

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

- b. **Domain Users** 그룹에 읽기 및 실행 권한을 부여합니다.

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

- c. 기타 ACL 항목과 일치하지 않는 사용자에게 대한 액세스를 거부하려면 other ACL 항목에 대한 권한을 설정합니다.

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

이러한 설정은 이 디렉터리에만 적용됩니다. Windows에서는 이러한 ACL 이 이 폴더만 모드로 매핑됩니다.

3. 이전 단계에서 설정한 권한을 이 디렉터리에 생성된 새 파일 시스템 개체에서 상속할 수 있도록 하려면 다음을 실행합니다.

```
# 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/
```

이러한 설정을 사용하면 보안 주체의 **This folder only** mode가 이제 이 폴더, 하위 폴더 및 파일로 설정됩니다.

Samba는 절차에 설정된 권한을 다음 Windows ACL에 매핑합니다.

보안 주체	액세스	적용 대상
도메인\Domain 관리자	완전 제어	이 폴더, 하위 폴더 및 파일
도메인\Domain 사용자	읽기 & 실행	이 폴더, 하위 폴더 및 파일

보안 주체	액세스	적용 대상
Everyone ^[a]	없음	이 폴더, 하위 폴더 및 파일
소유자 (Unix User\owner) ^[b]	완전 제어	이 폴더만
primary_group (Unix User\tekton_group) ^[c]	없음	이 폴더만
CREATOR OWNER ^{[d][e]}	완전 제어	하위 폴더 및 파일만
CREATOR GROUP ^{[e][f]}	없음	하위 폴더 및 파일만

[a] Samba는 이 주체에 대한 권한을 **other** ACL 항목에서 매핑합니다.

[b] Samba는 디렉터리의 소유자를 이 항목에 매핑합니다.

[c] Samba는 디렉터리의 기본 그룹을 이 항목에 매핑합니다.

[d] 새 파일 시스템 오브젝트에서 작성자는 이 보안 주체의 권한을 자동으로 상속합니다.

[e] POSIX ACL을 사용하는 공유에서는 지원되지 않는 ACL에서 이러한 보안 주체를 구성하거나 제거합니다.

[f] 새 파일 시스템 오브젝트에서 작성자의 기본 그룹은 이 보안 주체의 권한을 자동으로 상속합니다.

1.8. POSIX ACL을 사용하는 공유에 권한 설정

필요한 경우 Samba 공유에 대한 액세스 권한을 제한하거나 부여하려면 **/etc/tekton/tekton.conf** 파일의 공유 섹션에서 특정 매개 변수를 설정할 수 있습니다.



참고

공유 기반 권한은 사용자, 그룹 또는 호스트가 공유에 액세스할 수 있는 경우 관리합니다. 이러한 설정은 파일 시스템 ACL에 영향을 미치지 않습니다.

공유 기반 설정을 사용하여 공유에 대한 액세스를 제한합니다(예: 특정 호스트의 액세스 거부).

사전 요구 사항

- POSIX ACL과의 공유가 설정되었습니다.

1.8.1. 사용자 및 그룹 기반 공유 액세스 구성

사용자 및 그룹 기반 액세스 제어를 사용하면 특정 사용자 및 그룹의 공유에 대한 액세스 권한을 부여하거나 거부할 수 있습니다.

사전 요구 사항

- 사용자 또는 그룹 기반 액세스 권한을 설정하려는 Samba 공유가 있습니다.

절차

1. 예를 들어 **Domain Users** 그룹의 모든 멤버가 **사용자** 계정에 대해 액세스가 거부되는 동안 공유에 액세스할 수 있도록 하려면 공유 구성에 다음 매개 변수를 추가합니다.

```
valid users = +DOMA\*Domain Users"
invalid users = DOMA\Muser
```

유효하지 않은 users 매개 변수는 **유효한 users** 매개 변수보다 우선 순위가 높습니다. 예를 들어 사용자 계정이 **Domain Users** 그룹의 멤버인 경우 이전 예제를 사용할 때 이 계정에 대한 액세스가 거부됩니다.

2. Samba 구성을 다시 로드합니다.

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

추가 리소스

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

1.8.2. 호스트 기반 공유 액세스 구성

호스트 기반 액세스 제어를 사용하면 클라이언트의 호스트 이름, IP 주소 또는 IP 범위를 기반으로 공유에 대한 액세스 권한을 부여하거나 거부할 수 있습니다.

다음 절차에서는 **127.0.0.1** IP 주소, **192.0.2.0/24** IP 범위, **client1.example.com** 호스트가 공유에 액세스하는 방법을 설명하고 **client2.example.com** 호스트의 액세스를 추가로 거부하는 방법을 설명합니다.

사전 요구 사항

- 호스트 기반 액세스 권한을 설정하려는 Samba 공유가 있습니다.

절차

1. 다음 매개 변수를 **/etc/ smb.conf** 파일의 공유 구성에 추가합니다.

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

hosts deny 매개 변수는 **hosts** 허용 보다 우선 순위가 높습니다. 예를 들어 **client1.example.com** 이 **hosts allow** 매개 변수에 나열된 IP 주소로 확인되면 이 호스트에 대한 액세스가 거부됩니다.

2. Samba 구성을 다시 로드합니다.

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

추가 리소스

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

1.9. WINDOWS ACL을 사용하는 공유 설정

Samba는 공유 및 파일 시스템 오브젝트에서 Windows ACL 설정을 지원합니다. 이를 통해 다음을 수행할 수 있습니다.

- 세분화된 Windows ACL 사용
- Windows를 사용하여 공유 권한 및 파일 시스템 ACL 관리

또는 POSIX ACL을 사용하도록 공유를 구성할 수 있습니다.

자세한 내용은 [POSIX ACL을 사용하는 Samba 파일 공유](#) 설정을 참조하십시오.

이 섹션의 일부는 Samba440에 게시된 [Windows ACL 문서를 사용하여 공유 설정](#) 문서에서 채택되었습니다. 라이선스: [CC BY 4.0](#). 작성자 및 기여자: Wiki 페이지의 [기록](#) 탭을 참조하십시오.

1.9.1. SeDiskOperatorPrivilege 권한 부여

SeDiskOperatorPrivilege 권한이 부여된 사용자 및 그룹만 Windows ACL을 사용하는 공유에 대한 권한을 구성할 수 있습니다.

절차

1. 예를 들어, **SeDiskOperatorPrivilege** 권한을 **DOMAIN\Domain Admins** 그룹에 부여하려면 다음을 수행합니다.

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



참고

도메인 환경에서 **SeDiskOperatorPrivilege** 을 도메인 그룹에 부여합니다. 이를 통해 사용자의 그룹 멤버십을 업데이트하여 권한을 중앙에서 관리할 수 있습니다.

2. **SeDiskOperatorPrivilege** 이 부여된 모든 사용자 및 그룹을 나열하려면 다음을 수행합니다.

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

1.9.2. Windows ACL 지원 활성화

Windows ACL을 지원하는 공유를 구성하려면 Samba에서 이 기능을 활성화해야 합니다.

사전 요구 사항

- 사용자 공유는 Samba 서버에 구성됩니다.

절차

1. 모든 공유에 대해 전역적으로 활성화하려면 `/etc/samba/smb.conf` 파일의 **[global]** 섹션에 다음 설정을 추가합니다.

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

또는 공유의 섹션에 동일한 매개변수를 추가하여 개별 공유에 대해 Windows ACL 지원을 활성화할 수 있습니다.

2. **smb** 서비스를 다시 시작하십시오.

```
# systemctl restart smb
```

1.9.3. Windows ACL을 사용하는 공유 추가

이 섹션에서는 `/srv/controlPlane/example` 디렉터리의 콘텐츠를 공유하며 **Windows ACL**을 사용하는 **example**이라는 공유를 생성하는 방법을 설명합니다.

절차

1. 폴더가 없는 경우 해당 폴더를 생성합니다. 예를 들어 다음과 같습니다.

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

2. **enforcing** 모드에서 SELinux를 실행하는 경우 디렉터리에 **samba_share_t** 컨텍스트를 설정합니다.

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

3. 예제 공유를 `/etc/samba/smb.conf` 파일에 추가합니다. 예를 들어 공유 쓰기 기능을 추가하려면 다음을 수행합니다.

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



참고

파일 시스템 ACL에 관계없이 **read only = no**를 설정하지 않으면 Samba는 읽기 전용 모드로 디렉토리를 공유합니다.

4. 모든 공유의 **[global]** 섹션에서 Windows ACL 지원을 활성화하지 않은 경우 **[example]** 섹션에 다음 매개변수를 추가하여 이 공유에 대해 이 기능을 활성화합니다.

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

5. `/etc/samba/smb.conf` 파일을 확인합니다.

```
# testparm
```

6. 필요한 포트를 열고 **firewall-cmd** 유틸리티를 사용하여 방화벽 구성을 다시 로드합니다.

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

7. **smb** 서비스를 다시 시작하십시오.

```
# systemctl restart smb
```

1.9.4. Windows ACL을 사용하는 공유의 공유 권한 및 파일 시스템 ACL 관리

Windows ACL을 사용하는 Samba 공유에서 공유 권한 및 파일 시스템 ACL을 관리하려면 **컴퓨터 관리**와 같은 Windows 애플리케이션을 사용합니다. 자세한 내용은 Windows 설명서를 참조하십시오. 또는 **smbcacls** 유틸리티를 사용하여 ACL을 관리합니다.



참고

Windows에서 파일 시스템 권한을 수정하려면 **SeDiskOperatorPrivilege** 권한이 부여된 계정을 사용해야 합니다.

추가 리소스

- [smbcacls를 사용하여 SMB 공유에서 ACL 관리](#)
- [SeDiskOperatorPrivilege 권한 부여](#)

1.10. SMBACLS를 사용하여 SMB 공유에서 ACL 관리

smbcacls 유틸리티는 SMB 공유에 저장된 파일 및 디렉터리의 ACL을 나열, 설정 및 삭제할 수 있습니다. **smbcacls**를 사용하여 파일 시스템 ACL을 관리할 수 있습니다.

- 고급 Windows ACL 또는 POSIX ACL을 사용하는 로컬 또는 원격 Samba 서버에서
- Windows에서 호스팅되는 공유에 대한 ACL을 원격으로 관리하려면 Red Hat Enterprise Linux에서

1.10.1. 액세스 제어 항목

파일 시스템 오브젝트의 각 ACL 항목에는 다음 형식으로 ACE(Access Control Entries)가 포함되어 있습니다.

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

예 1.3. 액세스 제어 항목

AD\Domain Users 그룹에 이 폴더, 하위 폴더 및 **Windows**의 파일에 적용되는 수정 권한이 있는 경우 ACL에 다음 ACE가 포함됩니다.

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

ACE에는 다음 부분이 포함되어 있습니다.

보안 주체

보안 주체는 ACL의 권한이 적용되는 사용자, 그룹 또는 SID입니다.

액세스 권한

오브젝트에 대한 액세스 허용 또는 거부 여부를 정의합니다. 값은 **ALLOWED** 또는 **DENIED** 일 수 있습니다.

상속 정보

다음 값이 있습니다.

표 1.1. 상속 설정

값	설명	매핑
OI	Object Inherit	이 폴더 및 파일
CI	Container Inherit	이 폴더 및 하위 폴더
IO	상속만	현재 파일 또는 디렉터리에는 ACE가 적용되지 않습니다.
ID	inherited	상위 디렉터리에서 ACE를 상속했습니다.

또한 값은 다음과 같이 결합할 수 있습니다.

표 1.2. 상속 설정 조합

값 조합	Windows에 매핑하여 설정할 수 있습니다.
OI CI	이 폴더, 하위 폴더 및 파일
OI CI IO	하위 폴더 및 파일만
CI IO	하위 폴더만 해당
OI IO	파일만

권한

이 값은 하나 이상의 Windows 권한 또는 rootfs **cacls** 별칭을 나타내는 16진수 값일 수 있습니다.

- 하나 이상의 Windows 권한을 나타내는 16진수 값입니다. A hex value that represents one or more Windows permissions.

다음 표에서는 고급 Windows 권한 및 해당 값을 16진수 형식으로 표시합니다.

표 1.3. Windows 권한 및 해당 rootfscacls 값 16진수 형식

Windows 권한	Hex 값
완전 제어	0x001F01FF

Windows 권한	Hex 값
트래버스 폴더/실행 파일	0x00100020
목록 / 읽기 데이터	0x00100001
읽기 속성	0x00100080
확장 속성 읽기	0x00100008
파일 생성 / 쓰기	0x00100002
폴더 생성 / 데이터 추가	0x00100004
쓰기 속성	0x00100100
확장 속성 작성	0x00100010
하위 폴더 및 파일 삭제	0x00100040
delete	0x00110000
읽기 권한	0x00120000
권한 변경	0x00140000
소유권 가져오기	0x00180000

비트 단위 **OR** 작업을 사용하여 여러 권한을 단일 16진수 값으로 결합할 수 있습니다.

자세한 내용은 [ACE 마스크 계산](#)을 참조하십시오.

- Net **Namespacecacs** 별칭. 다음 표에서 사용 가능한 별칭을 표시합니다.

표 1.4. 기존 pvccacIs 별칭 및 해당 Windows 권한

pvccacIs 별칭	Windows 권한에 매핑
R	읽기
READ	읽기 & 실행

pvccacIs 별칭	Windows 권한에 매핑
W	특별함: ○ 파일 생성 / 쓰기 ○ 폴더 생성 / 데이터 추가 ○ 쓰기 속성 ○ 확장 속성 작성 ○ 읽기 권한
D	delete
P	권한 변경
O	소유권 가져오기
X	트래버스 / 실행
변경	수정
FULL	완전 제어



참고

권한을 설정할 때 단일 문자 별칭을 결합할 수 있습니다. 예를 들어 **RD** 를 설정하여 Windows 권한 **읽기** 및 **삭제** 를 적용할 수 있습니다. 그러나 여러 개의 단일 문자 별칭을 결합하거나 별칭과 hex 값을 결합할 수 없습니다.

1.10.2. rootfscacIs를 사용하여 ACL 표시

SMB 공유의 ACL을 표시하려면 rootfs **cacIs** 유틸리티를 사용합니다. **--add** 와 같은 operation 매개변수 없이 NetNamespace **cacIs** 를 실행하면 유틸리티에서 파일 시스템 오브젝트의 ACL을 표시합니다.

절차

예를 들어 **//server/example** 공유의 루트 디렉터리의 ACL을 나열하려면 다음을 수행합니다.

```
# smbcacls //server/example / -U "DOMAINAdministrator"
Enter DOMAINAdministrator'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
```

명령 출력이 표시됩니다.

- **REVISION:** 보안 설명자의 내부 Windows NT ACL 개정
- **CONTROL:** 보안 설명자 제어
- 보안 설명자 소유자의 이름 또는 SID
-
- **GROUP:** 보안 설명자 그룹의 이름 또는 SID
-
- **ACL 항목.** 자세한 내용은 [액세스 제어 항목](#)을 참조하십시오.

1.10.3. ACE 마스크 계산

대부분의 경우 **ACE**를 추가하거나 업데이트할 때 [Existingtekton cacs](#) 별칭과 해당 **Windows** 권한에 나열된 [rootfscacs](#) 별칭을 사용합니다.

그러나 **Windows** 권한에 나열된 고급 **Windows** 권한을 설정하고 **16진수 형식의 해당 rootfscacs 값**을 설정하려는 경우, 비트 단위 **OR** 작업을 사용하여 올바른 값을 계산해야 합니다. 다음 **shell** 명령을 사용하여 값을 계산할 수 있습니다.

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

예 1.4. ACE mask 계산

다음 권한을 설정하려고 합니다.

- 트래버스 폴더/실행 파일(0x00100020)
- 목록 / 읽기 데이터(0x00100001)
- 읽기 속성(0x00100080)

이전 권한에 대한 **16진수 값**을 계산하려면 다음을 입력합니다.

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

ACE를 설정하거나 업데이트할 때 반환된 값을 사용합니다.

1.10.4. rootfscaccls를 사용하여 ACL 추가, 업데이트 및 제거

qcow caccls 유틸리티에 전달하는 매개변수에 따라 파일 또는 디렉터리에서 **ACL**을 추가, 업데이트 및 제거할 수 있습니다.

ACL 추가

이 폴더에 대한 **CHANGE** 권한을 부여하는 **//server/example** 공유의 루트에 **ACL**을 추가하려면 **AD\Domain Users** 그룹에 파일을 추가합니다.

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

ACL 업데이트

ACL을 업데이트하는 것은 새 **ACL**을 추가하는 것과 유사합니다. 기존 보안 주체와 함께 **--modify** 매개 변수를 사용하여 **ACL**을 재정의하여 **ACL**을 업데이트합니다. **pvc caccls**가 **ACL** 목록에서 보안 주체를 발견하면 유틸리티가 권한을 업데이트합니다. 그렇지 않으면 명령이 오류와 함께 실패합니다.

ACL for SID *principal_name* not found

예를 들어 **AD\Domain Users** 그룹의 권한을 업데이트하고 이 폴더, 하위 폴더 및 파일에 대해 **READ**로 설정하려면 다음을 수행합니다.

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

ACL 삭제

ACL을 삭제하려면 정확한 **ACL**을 사용하는 **--delete** 매개 변수를 **smbcaccls** 유틸리티에 전달합니다. 예를 들어 다음과 같습니다.

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

1.11. 사용자가 SAMBA 서버에서 디렉토리를 공유 가능

Samba 서버에서는 사용자가 루트 권한 없이 디렉토리를 공유할 수 있도록 구성할 수 있습니다.

1.11.1. 사용자 공유 기능 활성화

사용자가 디렉토리를 공유할 수 있으려면 관리자가 **Samba**에서 사용자 공유를 활성화해야 합니다.

예를 들어 로컬 **example** 그룹의 멤버만 활성화하여 사용자 공유를 생성하려면 다음을 수행합니다.

절차

1. 로컬 **example** 그룹이 없는 경우 해당 그룹을 생성합니다.

```
# groupadd example
```

2. **Samba**에서 사용자 공유 정의를 저장하고 권한을 올바르게 설정하도록 디렉토리를 준비합니다. 예를 들어 다음과 같습니다.

- a. 디렉토리를 만듭니다.

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

- b. **example** 그룹에 대한 쓰기 권한을 설정합니다.

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

- c. **Sticky bit**를 설정하면 사용자가 이 디렉토리의 다른 사용자가 저장한 파일의 이름을 변경하거나 삭제할 수 없습니다.

3. **/etc/samba/smb.conf** 파일을 편집하고 **[global]** 섹션에 다음을 추가합니다.

- a. 사용자 공유 정의를 저장하도록 구성된 디렉토리의 경로를 설정합니다. 예를 들어 다음과 같습니다.

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

- b. 이 서버에서 만들 수 있는 **Samba** 공유 사용자 공유 수를 설정합니다. 예를 들어 다음과 같습니다.

```
usershare max shares = 100
```

usershare max shares 매개변수에 기본값 **0**을 사용하는 경우 사용자 공유가 비활성화됩니다.

c.

선택적으로 절대 디렉터리 경로 목록을 설정합니다. 예를 들어 **Samba**가 **/data** 및 **/srv** 디렉터리의 하위 디렉터리만 공유할 수 있도록 하려면 다음을 설정합니다.

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

설정할 수 있는 추가 사용자 공유 관련 매개변수 목록은 **smb.conf(5)** 도움말 페이지의 **USERSHARES** 섹션을 참조하십시오.

4.

/etc/samba/smb.conf 파일을 확인합니다.

```
# testparm
```

5.

Samba 구성을 다시 로드합니다.

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

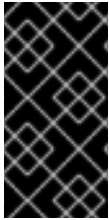
이제 사용자가 사용자 공유를 생성할 수 있습니다.

1.11.2. 사용자 공유 추가

Samba에서 사용자 공유 기능을 활성화한 후 사용자는 **net usershare add** 명령을 실행하여 **root** 권한 없이 **Samba** 서버의 디렉토리를 공유할 수 있습니다.

net usershare add 명령의 개요입니다.

```
net usershare add share_name path [[ comment ] | [ ACLs ]] [ guest_ok=y|n ]
```



중요

사용자 공유를 생성할 때 **ACL**을 설정하는 경우 **ACL**보다 먼저 **comment** 매개변수를 지정해야 합니다. 빈 주석을 설정하려면 이중 따옴표로 빈 문자열을 사용합니다.

사용자가 사용자 공유에 대해 게스트 액세스를 활성화할 수 있습니다. 관리자가 사용자 공유 영역의 **usershare allow guests = yes** 를 **/etc/exporter/ECDHE.conf** 파일의 **[global]** 섹션에 있는 경우에만 활성화할 수 있습니다.

예 1.5. 사용자 공유 추가

사용자가 **Samba** 서버에서 **/srv/samba/** 디렉터리를 공유하려고 합니다. 공유 이름은 **example** 이고, 주석을 설정하지 않으며, 게스트 사용자가 액세스할 수 있어야 합니다. 또한 공유 권한은 **AD\Domain Users** 그룹에 대한 전체 액세스 권한과 기타 사용자의 읽기 권한으로 설정해야 합니다. 이 공유를 추가하려면 사용자로 실행합니다.

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

1.11.3. 사용자 공유의 설정 업데이트

사용자 공유의 설정을 업데이트하려면 **net usershare add** 명령을 동일한 공유 이름 및 새 설정으로 사용하여 공유를 재정의합니다.

[사용자 공유 추가](#)를 참조하십시오.

1.11.4. 기존 사용자 공유에 대한 정보 표시

사용자는 **Samba** 서버에 **net usershare info** 명령을 입력하여 사용자 공유 및 해당 설정을 표시할 수 있습니다.

사전 요구 사항

- 사용자 공유는 **Samba** 서버에 구성됩니다.

절차

1.

사용자가 만든 모든 사용자 공유를 표시하려면 다음을 수행하십시오.

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

명령을 실행하는 사용자가 생성한 공유만 나열하려면 -l 매개 변수를 생략합니다.

2.

특정 공유에 대한 정보만 표시하려면 공유 이름 또는 와일드카드 카드를 명령에 전달합니다. 예를 들어 이름이 **share_**로 시작하는 공유에 대한 정보를 표시하려면 다음을 수행합니다.

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

1.11.5. 사용자 공유 나열

Samba 서버에서의 설정 없이 사용 가능한 사용자 공유만 나열하려면 **net usershare list** 명령을 사용합니다.

사전 요구 사항

•

사용자 공유는 **Samba** 서버에 구성됩니다.

절차

1.

사용자가 생성한 공유를 나열하려면 다음을 수행하십시오.

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

명령을 실행하는 사용자가 생성한 공유만 나열하려면 -l 매개 변수를 생략합니다.

2.

특정 공유만 나열하려면 공유 이름 또는 와일드카드를 명령에 전달합니다. 예를 들어 이름이 **share_**로 시작되는 공유만 나열하려면 다음을 수행하십시오.

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

1.11.6. 사용자 공유 삭제

사용자 공유를 삭제하려면 **net usershare delete** 명령을 공유를 만든 사용자 또는 **root** 사용자로 사용합니다.

사전 요구 사항

- 사용자 공유는 **Samba** 서버에 구성됩니다.

절차

```
$ net usershare delete share_name
```

1.12. 인증 없이 액세스를 허용하도록 공유 구성

특정 상황에서는 인증 없이 사용자가 연결할 수 있는 디렉토리를 공유하려고 합니다. 이를 구성하려면 공유에서 게스트 액세스를 활성화합니다.



주의

인증이 필요하지 않은 공유는 보안 위험이 될 수 있습니다.

1.12.1. 공유에 대한 게스트 액세스 활성화

공유에 게스트 액세스가 활성화된 경우 **Samba**는 게스트 계정 매개변수에 설정된 운영 체제 계정에 게스트 연결을 매핑합니다. 게스트 사용자는 다음 조건 중 하나 이상이 충족되는 경우 이 공유의 파일에 액세스할 수 있습니다.

- 이 계정은 파일 시스템 **ACL**에 나열됩니다.
- 다른 사용자에게 대한 **POSIX** 권한에서 허용

예 1.6. 게스트 공유 권한

게스트 계정을 **nobody**에 매핑하도록 **Samba**를 구성한 경우 기본값인 경우 다음 예제의 **ACL**입니다.

- 게스트 사용자가 **file1.txt**를 읽을 수 있도록 허용
- 게스트 사용자가 **file2.txt**를 읽고 수정할 수 있도록 허용
- 게스트 사용자가 **file3.txt**를 읽거나 수정하는 것을 방지

```
-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
```

절차

1. `/etc/samba/smb.conf` 파일을 편집합니다.
 - a. 이 서버에서 설정한 첫 번째 게스트 공유인 경우:
 - i. **[global]** 섹션에서 **guest = Bad User**에 **map**을 설정합니다.

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

이 설정에서 **Samba**는 사용자 이름이 존재하지 않는 한 잘못된 암호를 사용하는 로그인 시도를 거부합니다. 지정된 사용자 이름이 없고 공유에서 게스트 액세스가 활성화된 경우 **Samba**는 연결을 게스트 로그인으로 처리합니다.

ii.

기본적으로 **Samba**는 게스트 계정을 **Red Hat Enterprise Linux**의 **nobody** 계정에 매핑합니다. 또는 다른 계정을 설정할 수 있습니다. 예를 들어 다음과 같습니다.

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

이 매개 변수에 설정된 계정은 **Samba** 서버에 로컬로 존재해야 합니다. 보안상의 이유로 유효한 셸이 할당되지 않은 계정을 사용하는 것이 좋습니다.

b.

[example] share 섹션에 **guest ok = yes** 설정을 추가합니다.

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

2.

/etc/samba/smb.conf 파일을 확인합니다.

```
# testparm
```

3.

Samba 구성을 다시 로드합니다.

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

1.13. MACOS 클라이언트용 SAMBA 구성

CFS (Virtual File System) Samba 모듈은 **Apple** 서버 메시지 블록(**SMB**) 클라이언트와의 향상된 호환성을 제공합니다.

1.13.1. macOS 클라이언트에 파일 공유를 제공하기 위해 Samba 구성 최적화

이 섹션에서는 **macOS** 클라이언트의 **Samba** 파일 공유를 최적화하기 위해 서버에서 호스팅되는 모든 **Samba** 공유에 대해유프 모듈을 구성하는 방법에 대해 설명합니다.



참고

Red Hat은 전 세계적으로 대체 모듈을 활성화할 것을 권장합니다. macOS를 사용하는 클라이언트는 클라이언트가 서버에 대한 첫 번째 연결을 설정할 때 서버 서버 메시지 블록 2(SMB2) Apple (APL) 프로토콜 확장을 협상합니다. 클라이언트가 먼저 AAPL 확장 기능을 사용하지 않고 공유에 연결하면 클라이언트는 서버 공유를 위해 확장 기능을 사용하지 않습니다.

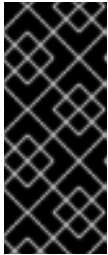
사전 요구 사항

- Samba는 파일 서버로 구성됩니다.

절차

1. `/etc/controlPlane/ECDHE.conf` 파일을 편집하고 `[global]` 섹션에서 대체 및 `streams_xattr` `tekton` 모듈을 활성화합니다.

```
vfs objects = fruit streams_xattr
```



중요

`streams_xattr` 을 활성화하기 전에 대체 모듈을 활성화해야 합니다. 유사 모듈은 대체 데이터 스트림(ADS)을 사용합니다. 이러한 이유로 `streams_xattr` 모듈도 활성화해야 합니다.

2. 선택적으로 공유에서 macOS Time Machine 지원을 제공하려면 `/etc/samba/smb.conf` 파일의 공유 구성에 다음 설정을 추가합니다.

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

3. `/etc/samba/smb.conf` 파일을 확인합니다.

```
# testparm
```

4. Samba 구성을 다시 로드합니다.

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

추가 리소스

- **vfs_fruit(8)** 매뉴얼 페이지.
- 파일 공유 구성:
 - **POSIX ACL**을 사용하는 **Samba** 파일 공유 설정
 - **Windows ACL**을 사용하는 공유 설정.

1.14. SMBCLIENT 유틸리티를 사용하여 SMB 공유에 액세스

smbclient 유틸리티를 사용하면 명령줄 **FTP** 클라이언트와 마찬가지로 **SMB** 서버의 파일 공유에 액세스할 수 있습니다. 예를 들어 공유 영역에 파일을 업로드하고 다운로드할 수 있습니다.

사전 요구 사항

- **samba-client** 패키지가 설치되어 있습니다.

1.14.1. NetNamespaceclient 대화형 모드가 작동하는 방법

예를 들어 **DOMAIN\user** 계정을 사용하여 서버에서 호스팅되는 예제 공유를 인증하려면 다음을 수행합니다.

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

Net Namespaceclient 가 공유에 성공적으로 연결된 후 유틸리티가 대화형 모드로 전환되고 다음 프롬프트가 표시됩니다.

```
smb: \>
```

대화형 셸에서 사용 가능한 모든 명령을 표시하려면 다음을 입력합니다.

```
smb: \> help
```

특정 명령에 대한 도움말을 표시하려면 다음을 입력합니다.

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

추가 리소스

- [pvcclient\(1\) 도움말 페이지](#)

1.14.2. 대화형 모드에서 smbclient 사용

-c 매개변수 없이 **NetNamespaceclient** 를 사용하는 경우 유틸리티가 대화형 모드를 시작합니다. 다음 절차에서는 **SMB** 공유에 연결하고 하위 디렉터리에서 파일을 다운로드하는 방법을 보여줍니다.

절차

1. 공유에 연결합니다.

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

2. */example/* 디렉터리로 변경합니다.

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

3. 디렉터리에 파일을 나열합니다.

```
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. *example.txt* 파일을 다운로드합니다.

```
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.

공유에서 연결을 끊습니다.

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

1.14.3. 스크립팅 모드에서 **smbclient** 사용

-c 매개 변수를 **qcow client**에 전달하는 경우 원격 **SMB** 공유에서 명령을 자동으로 실행할 수 있습니다. 이를 통해 스크립트에서 **smbclient**를 사용할 수 있습니다.

다음 절차에서는 **SMB** 공유에 연결하고 하위 디렉터리에서 파일을 다운로드하는 방법을 보여줍니다.

절차

- 다음 명령을 사용하여 공유에 연결하고 **example** 디렉터리로 변경하고 **example.txt** 파일을 다운로드합니다.

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

1.15. 인쇄 서버로 **SAMBA** 설정

Samba를 출력 서버로 설정하면 네트워크의 클라이언트가 **Samba**를 사용하여 출력할 수 있습니다. 또한 **Windows** 클라이언트가 구성된 경우 **Samba** 서버에서 드라이버를 다운로드할 수 있습니다.

이 섹션의 일부는 [Samba Wiki](#)에 게시된 [Print Server 설명서](#)로 **Samba** 설정에서 채택되었습니다. 라이선스: [CC BY 4.0](#). 작성자 및 기여자: [Wiki](#) 페이지의 [기록](#) 탭을 참조하십시오.

사전 요구 사항

Samba는 다음 모드 중 하나로 설정되었습니다.

- 독립 실행형 서버
- 도메인 멤버

1.15.1. Samba 스펙 **sd** 서비스

Samba 스푼은 **rootfsd** 서비스에 통합된 서비스입니다. **Samba** 구성에서 **spoolsd** 를 활성화하여 많은 작업 또는 프린터가 있는 인쇄 서버의 성능을 크게 향상시킬 수 있습니다.

spoolssd 가 없으면 **Samba**는 **rootfs d** 프로세스를 분기하고 각 출력 작업의 **printcap** 캐시를 초기화합니다. 많은 수의 프린터가 있는 경우 캐시가 초기화되는 동안 **rootfs d** 서비스가 여러 초 동안 응답하지 않을 수 있습니다. **spoolssd** 서비스를 사용하면 지연 없이 출력 작업을 처리하는 사전 예약된 처리된 프로세스를 시작할 수 있습니다. 기본 스푼링 **d** 프로세스는 낮은 양의 메모리를 사용하며 포크를 사용하고 하위 프로세스를 종료합니다.

다음 절차에서는 스푼 서비스를 활성화하는 방법을 설명합니다.

절차

1.

/etc/samba/smb.conf 파일에서 **[global]** 섹션을 편집합니다.

a.

다음 매개변수를 추가합니다.

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

b.

선택적으로 다음 매개변수를 설정할 수 있습니다.

매개변수	기본값	설명
spoolssd:prefork_min_children	5	최소 하위 프로세스 수
spoolssd:prefork_max_children	25	최대 하위 프로세스 수
spoolssd:prefork_spawn_rate	5	Samba는 새 연결이 설정된 경우 이 매개변수에 설정된 새 하위 프로세스 수를 spoolssd:prefork_max_databind 에 설정된 값까지 포크합니다.
spoolssd:prefork_max_allowed_clients	100	클라이언트의 수, 자식 프로세스
spoolssd:prefork_child_min_life	60	몇 초 안에 자식 프로세스의 최소 수명을 지정합니다. 최소 60초가 됩니다.

2.

`/etc/samba/smb.conf` 파일을 확인합니다.

```
# testparm
```

3.

smb 서비스를 다시 시작하십시오.

```
# systemctl restart smb
```

서비스를 다시 시작한 후 **Samba**는 고유 한 하위 프로세스를 자동으로 시작합니다.

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

1.15.2. Samba에서 출력 서버 지원 활성화

이 섹션에서는 **Samba**에서 출력 서버 지원을 활성화하는 방법을 설명합니다.

절차

1.

Samba 서버에서 **CUPS**를 설정하고 **CUPS** 백엔드에 프린터를 추가합니다. **CUPS**에서 프린터를 구성하는 방법에 대한 자세한 내용은 출력 서버의 **CUPS** 웹 콘솔 (https://print_server_host_name:631/help)에 제공된 문서를 참조하십시오.



참고

Samba는 **CUPS**가 **Samba** 출력 서버에 로컬로 설치된 경우에만 출력 작업을 **CUPS**로 전달할 수 있습니다.

2.

`/etc/samba/smb.conf` 파일을 편집합니다.

a.

spoolssd 서비스를 활성화하려면 **[global]** 섹션에 다음 매개변수를 추가합니다.

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

b.

인쇄 백엔드를 구성하려면 **[errorss]** 섹션을 추가합니다.

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



중요

[databinds] 공유 이름은 하드 코딩되며 변경할 수 없습니다.

3.

/etc/samba/smb.conf 파일을 확인합니다.

```
# testparm
```

4.

필요한 포트를 열고 **firewall-cmd** 유틸리티를 사용하여 방화벽 구성을 다시 로드합니다.

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

5.

smb 서비스를 다시 시작하십시오.

```
# systemctl restart smb
```

서비스를 다시 시작한 후 **Samba**는 **CUPS** 백엔드에 구성된 모든 프린터를 자동으로 공유합니다. 특정 프린터만 수동으로 공유하려면 수동으로 특정 **프린터 공유**를 참조하십시오.

1.15.3. 수동으로 특정 프린터 공유

Samba를 출력 서버로 구성한 경우 기본적으로 **Samba**는 **CUPS** 백엔드에 구성된 모든 프린터를 공유합니다. 다음 절차에서는 특정 프린터만 공유하는 방법을 설명합니다.

사전 요구 사항

- **Samba가 인쇄 서버로 설정**

절차

1.

`/etc/samba/smb.conf` 파일을 편집합니다.

a.

[global] 섹션에서 설정을 설정하여 자동 프린터 공유를 비활성화합니다.

```
load printers = no
```

b.

공유할 각 프린터에 대해 섹션을 추가합니다. 예를 들어 **CUPS** 백엔드에 **example** 이라는 프린터를 **Samba**에서 **Example- printer**로 공유하려면 다음 섹션을 추가합니다.

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

각 프린터마다 개별 스푼 디렉터리가 필요하지 않습니다. **[915s]** 섹션에서 설정한 것과 동일한 **spool** 디렉터리를 프린터의 **path** 매개변수에 설정할 수 있습니다.

2.

`/etc/samba/smb.conf` 파일을 확인합니다.

```
# testparm
```

3.

Samba 구성을 다시 로드합니다.

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

1.16. SAMBA 인쇄 서버에서 WINDOWS 클라이언트의 자동 프린터 드라이버 다운로드 설정

Windows 클라이언트용 **Samba** 출력 서버를 실행하는 경우 드라이버를 업로드하고 프린터를 사전 설정할 수 있습니다. 사용자가 프린터에 연결하면 **Windows**가 클라이언트에서 로컬로 드라이버를 다운로드하여 설치합니다. 사용자는 설치에 대한 로컬 관리자 권한이 필요하지 않습니다. 또한 **Windows**는 자동 마운 수와 같은 사전 구성된 드라이버 설정을 적용합니다.

이 섹션의 일부는 **Samba Wiki**에 게시된 [Windows 클라이언트용 자동 printer Driver Downloads](#)에

서 채택되었습니다. 라이선스: [CC BY 4.0](#). 작성자 및 기여자: [Wiki 페이지의 기록](#) 탭을 참조하십시오.

사전 요구 사항

- **Samba가 인쇄 서버로 설정**

1.16.1. 프린터 드라이버에 대한 기본 정보

이 섹션에서는 프린터 드라이버에 대한 일반 정보를 제공합니다.

지원되는 드라이버 모델 버전

Samba는 **Windows 2000** 이상 및 **Windows Server 2000** 이상에서 지원되는 프린터 드라이버 모델 버전 3만 지원합니다. **Samba**는 **Windows 8** 및 **Windows Server 2012**에 도입된 드라이버 모델 버전 4를 지원하지 않습니다. 그러나 이러한 버전과 이후의 **Windows** 버전에서는 버전 3 드라이버도 지원합니다.

패키지 인식 드라이버

Samba는 패키지 인식 드라이버를 지원하지 않습니다.

업로드를 위한 프린터 드라이버 준비

드라이버를 **Samba** 출력 서버에 업로드하려면 먼저 다음을 수행합니다.

- 압축된 형식으로 제공되는 경우 드라이버의 압축을 풉니다.
- 일부 드라이버는 **Windows** 호스트에서 로컬로 드라이버를 설치하는 설정 애플리케이션을 시작해야 합니다. 특정 상황에서 설치 프로그램은 설치 프로그램이 실행되는 동안 운영 체제의 임시 폴더에 개별 파일을 추출합니다. 드라이버 파일을 업로드에 사용하려면 다음을 수행합니다.
 - a. 설치 프로그램을 시작합니다.
 - b. 임시 폴더의 파일을 새 위치로 복사합니다.
 - c. 설치를 취소합니다.

인쇄 서버에 업로드를 지원하는 드라이버를 프린터 제조업체에 요청합니다.

클라이언트에 프린터를 위한 32비트 및 64비트 드라이버 제공

32비트 및 64비트 Windows 클라이언트 모두에 대한 프린터 드라이버를 제공하려면 두 아키텍처에서 정확히 동일한 이름으로 드라이버를 업로드해야 합니다. 예를 들어 **ExampleECDHE(v1.0)**라는 **32비트** 드라이버를 업로드하고 이름이 **Example ECDHE (v1.0)** 인 **64비트** 드라이버를 업로드하는 경우 이름이 일치하지 않습니다. 따라서 드라이버 중 하나만 프린터에 할당할 수 있으며 두 아키텍처에서 드라이버를 사용할 수 없습니다.

1.16.2. 사용자가 드라이버를 업로드하고 사전 구성 가능

프린터 드라이버를 업로드하고 사전 구성할 수 있으려면 사용자 또는 그룹에 **Se printOperatorPrivilege** 권한이 부여되어야 합니다. 사용자를 **printadmin** 그룹에 추가해야 합니다. **Red Hat Enterprise Linux**는 **samba** 패키지를 설치할 때 이 그룹을 자동으로 생성합니다. **printadmin** 그룹에는 **1000** 미만의 사용 가능한 동적 시스템 **GID**가 할당됩니다.

절차

1.

예를 들어 **SeprintOperatorPrivilege** 권한을 **printadmin** 그룹에 부여하려면 다음을 수행합니다.

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



참고

도메인 환경에서 **SeprintOperatorPrivilege** 을 도메인 그룹에 부여합니다. 이를 통해 사용자의 그룹 멤버십을 업데이트하여 권한을 중앙에서 관리할 수 있습니다.

2.

SeprintOperatorPrivilege 이 부여된 모든 사용자 및 그룹을 나열하려면 다음을 수행합니다.

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

1.16.3. print\$ 공유 설정

Windows 운영 체제는 인쇄 서버에서 **print\$** 라는 공유에서 프린터 드라이버를 다운로드합니다. 이 공유 이름은 Windows에서 하드 코딩되며 변경할 수 없습니다.

다음 절차에서는 `/var/lib/controlPlane/drivers/` 디렉토리를 **print\$** 로 공유하고 로컬 **printadmin** 그룹의 멤버를 활성화하여 프린터 드라이버를 업로드하는 방법을 설명합니다.

절차

1.

[print\$] 섹션을 `/etc/controlPlane/ECDHE.conf` 파일에 추가합니다.

```
[print$]
path = /var/lib/samba/drivers/
read only = no
write list = @printadmin
force group = @printadmin
create mask = 0664
directory mask = 2775
```

다음 설정을 사용합니다.

- **printadmin** 그룹의 멤버만 프린터 드라이버를 공유에 업로드할 수 있습니다.
- 새로 생성된 파일과 디렉토리의 그룹이 **printadmin**으로 설정됩니다.
- 새 파일의 권한은 **664**로 설정됩니다.
- 새 디렉토리의 권한은 **2775**로 설정됩니다.

2.

모든 프린터에 대한 64비트 드라이버만 업로드하려면 다음 설정을 `/etc/controlPlane/octets.conf` 파일의 **[global]** 섹션에 추가합니다.

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

이 설정을 사용하지 않으면 Windows에 최소 32비트 버전을 업로드한 드라이버만 표시됩니다.

3. `/etc/samba/smb.conf` 파일을 확인합니다.

```
# testparm
```

4. Samba 구성 다시 로드

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

5. `printadmin` 그룹이 없는 경우 해당 그룹을 생성합니다.

```
# groupadd printadmin
```

6. `SeprintOperatorPrivilege` 권한을 `printadmin` 그룹에 부여합니다.

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

7. `enforcing` 모드에서 SELinux를 실행하는 경우 디렉터리에 `samba_share_t` 컨텍스트를 설정합니다.

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

8. `/var/lib/samba/drivers/` 디렉터리에 대한 권한을 설정합니다.

- POSIX ACL을 사용하는 경우 다음을 설정합니다.

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

- Windows ACL을 사용하는 경우 다음을 설정합니다.

보안 주체	액세스	적용 대상
CREATOR OWNER	완전 제어	하위 폴더 및 파일만
Authenticated Users	읽기 및 실행, 폴더 내용 나열, 읽기	이 폴더, 하위 폴더 및 파일
printadmin	완전 제어	이 폴더, 하위 폴더 및 파일

Windows에서 ACL 설정에 대한 자세한 내용은 Windows 설명서를 참조하십시오.

추가 리소스

- [사용자가 드라이버를 업로드하고 사전 설정하도록 활성화합니다.](#)

1.16.4. 클라이언트가 Samba 인쇄 서버를 신뢰할 수 있도록 VDDK 만들기

보안상의 이유로 최근 Windows 운영 체제는 클라이언트가 신뢰할 수 없는 서버에서 패키지 인식 프린터 드라이버를 다운로드하지 못하도록 합니다. 인쇄 서버가 AD의 멤버인 경우 도메인에 GPO(그룹 정책 개체)를 만들어 Samba 서버를 신뢰할 수 있습니다.

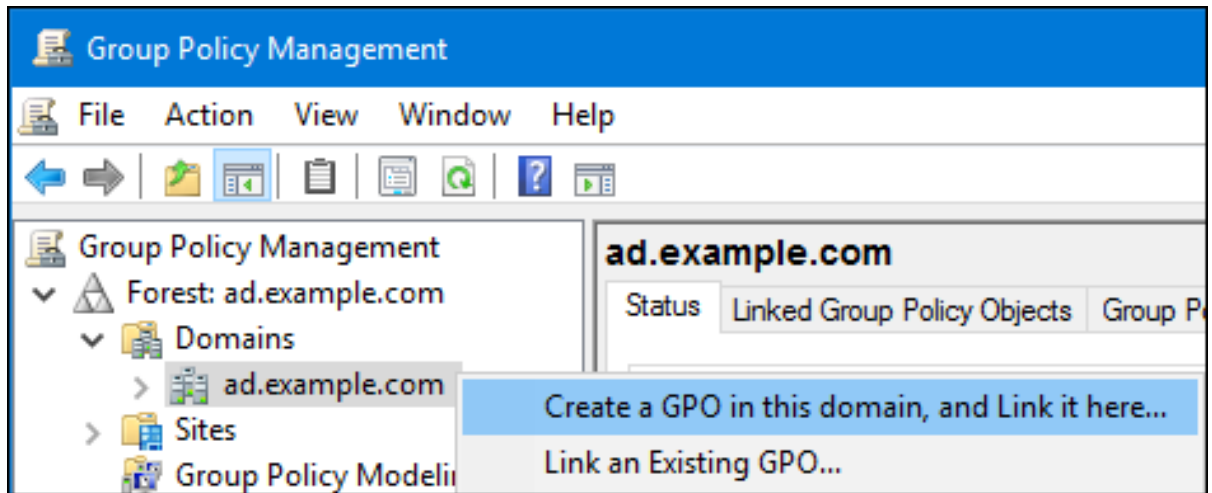
사전 요구 사항

- Samba 출력 서버는 AD 도메인의 멤버입니다.
- VDDK를 생성하는 데 사용하는 Windows 컴퓨터에는 Windows Remote Server Administration Tools(RSAT)가 설치되어 있어야 합니다. 자세한 내용은 Windows 설명서를 참조하십시오.

절차

1. AD 도메인 Administrator 사용자와 같은 그룹 정책을 편집할 수 있는 계정을 사용하여 Windows 컴퓨터에 로그인합니다.
2. Group Policy Management Console을 엽니다.
3. AD 도메인을 마우스 오른쪽 버튼으로 클릭하고 이 도메인에서 VDDK 만들기를 선택하고 여

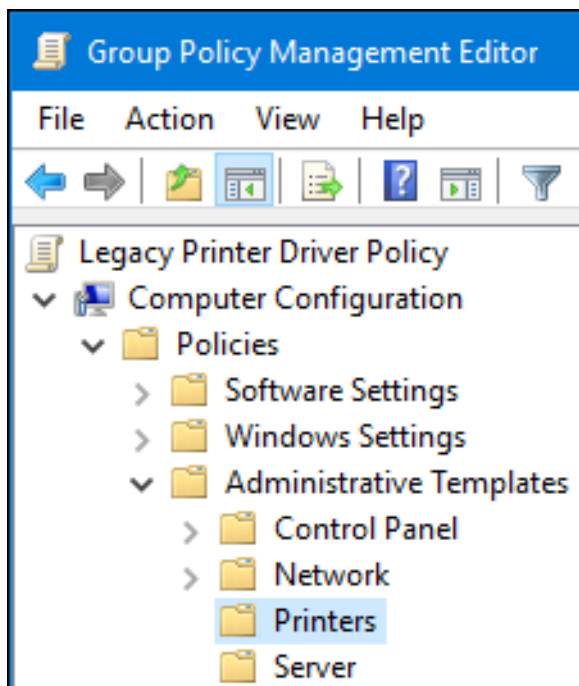
기에 연결합니다.



4. **Legacy printer Driver Policy**와 같은 **VDDK**의 이름을 입력하고 확인을 클릭합니다. 새 **GPO**가 도메인 항목에 표시됩니다.

5. 새로 만든 **NetNamespace**를 마우스 오른쪽 버튼으로 클릭하고 **편집**을 선택하여 그룹 정책 관리 편집기를 엽니다.

6. **Computer Configuration** → **Policies** → **Administrative Templates** → 이동합니다.



7. 창 오른쪽에 있는 **Point 및 Print Restriction**을 두 번 클릭하여 정책을 편집합니다.

a.

정책을 활성화하고 다음 옵션을 설정합니다.

i.

Users(사용자)는 이러한 서버를 가리키거나 출력할 수 있으며 **Samba** 인쇄 서버의 정규화된 도메인 이름(**FQDN**)을 이 옵션 옆에 있는 필드에 입력합니다.

ii.

보안 프롬프트 아래에 있는 두 확인란의 경우 경고 또는 설명 프롬프트가 표시되지 않음을 선택합니다.

Point and Print Restrictions

Point and Print Restrictions

☐ Not Configured Comment:
☒ Enabled
☐ Disabled

Supported on: **At least Windows Vista**

Options:

☒ Users can only point and print to these servers:
 Enter fully qualified server names separated by semicolons

☐ Users can only point and print to machines in their forest

Security Prompts:

When installing drivers for a new connection:

When updating drivers for an existing connection:

b.

확인을 클릭합니다.

8.

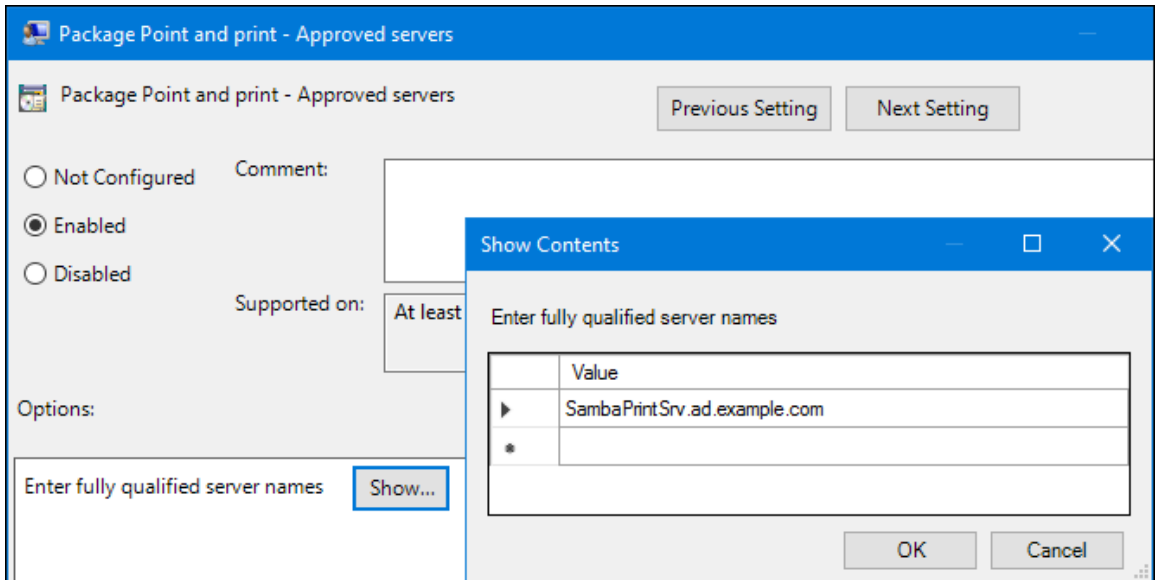
패키지 포인트 및 인쇄 - 승인된 서버를 두 번 클릭하여 정책을 편집합니다.

a.

정책을 활성화하고 **Show** (표시) 버튼을 클릭합니다.

b.

Samba 출력 서버의 **FQDN**을 입력합니다.



c.

OK 를 클릭하여 **Show Contents** 및 정책의 속성 창을 모두 닫습니다.

9.

그룹 정책 관리 편집기 를 닫습니다.

10.

그룹 정책 관리 콘솔을 닫습니다.

Windows 도메인 멤버가 그룹 정책을 적용한 후 사용자가 프린터에 연결하면 **Samba** 서버에서 프린터 드라이버가 자동으로 다운로드됩니다.

추가 리소스



그룹 정책을 사용하는 방법은 **Windows** 설명서를 참조하십시오.

1.16.5. 드라이버 업로드 및 프린터 사전 구성

Windows 클라이언트에서 **Print Management** 애플리케이션을 사용하여 **Samba** 인쇄 서버에서 호스팅되는 드라이버 및 사전 설정 프린터를 업로드합니다. 자세한 내용은 **Windows** 설명서를 참조하십시오.

1.17. FIPS 모드가 활성화된 서버에서 SAMBA 실행

이 섹션에서는 **FIPS** 모드가 활성화된 상태에서 **Samba**를 실행할 때의 제한 사항을 간략하게 설명합니다. 또한 **Samba**를 실행하는 **Red Hat Enterprise Linux** 호스트에서 **FIPS** 모드를 활성화하는 절차도 제공합니다.

1.17.1. FIPS 모드에서 Samba 사용 제한

다음 **Samba** 모드 및 기능은 표시된 조건에서 **FIPS** 모드에서 작동합니다.

- **Samba**는 **AD(Active Directory)** 또는 **AES** 암호를 사용하는 **Kerberos** 인증을 사용하는 **IdM(Red Hat Identity Management)** 환경에서만 사용됩니다.
- **Active Directory** 도메인 멤버의 파일 서버로 **Samba**. 그러나 이를 위해서는 클라이언트가 **Kerberos**를 사용하여 서버에 인증해야 합니다.

FIPS의 보안 강화로 인해 **FIPS** 모드가 활성화된 경우 다음과 같은 **Samba** 기능 및 모드가 작동하지 않습니다.

- **RC4** 암호가 차단되어 있으므로 **NT LAN Manager(NTLM)** 인증
- 서버 메시지 블록 버전 1 (**SMB1**) 프로토콜
- 독립 실행형 파일 서버 모드(**SDSC** 인증을 사용)
- **NT4** 스타일 도메인 컨트롤러
- **NT4** 스타일 도메인 멤버입니다. **Red Hat**은 백그라운드에서 계속해서 기본 도메인 컨트롤러 (**PDC**) 기능 **IdM**을 지원합니다.
- **Samba** 서버에 대한 암호 변경 **Active Directory** 도메인 컨트롤러에 대해 **Kerberos**를 사용하여 암호 변경만 수행할 수 있습니다.

다음 기능은 **FIPS** 모드에서 테스트되지 않으므로 **Red Hat**에서 지원하지 않습니다.

- **Samba**를 출력 서버로 실행

1.17.2. FIPS 모드에서 Samba 사용

이 섹션에서는 **Samba**를 실행하는 **RHEL** 호스트에서 **FIPS** 모드를 활성화하는 방법을 설명합니다.

사전 요구 사항

- **Samba**는 **Red Hat Enterprise Linux** 호스트에서 구성됩니다.
- **Samba**는 **FIPS** 모드에서 지원되는 모드에서 실행됩니다.

절차

1. **RHEL**에서 **FIPS** 모드를 활성화합니다.

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

2. 서버를 재부팅합니다.

```
# reboot
```

3. **testparm** 유틸리티를 사용하여 구성을 확인합니다.

```
# testparm -s
```

명령에서 오류 또는 비호환성을 표시하는 경우 이를 수정하여 **Samba**가 올바르게 작동하는지 확인합니다.

추가 리소스

- [1.17.1절. “FIPS 모드에서 Samba 사용 제한”](#)

1.18. SAMBA 서버의 성능 튜닝

이 장에서는 특정 상황에서 **Samba**의 성능을 향상시킬 수 있는 설정과 성능에 부정적인 영향을 미칠 수 있는 설정에 대해 설명합니다.

이 섹션의 일부는 **Sambasouth**에 게시된 [성능 튜닝](#) 문서에서 채택되었습니다. 라이선스: **CC BY 4.0**.
 작성자 및 기여자: **Wiki** 페이지의 [기록](#) 탭을 참조하십시오.

사전 요구 사항

- **Samba**가 파일 또는 인쇄 서버로 설정

1.18.1. SMB 프로토콜 버전 설정

각각의 새로운 **SMB** 버전은 기능을 추가하고 프로토콜의 성능을 향상시킵니다. 최근의 **Windows** 및 **Windows Server** 운영 체제는 항상 최신 프로토콜 버전을 지원합니다. 또한 **Samba**가 최신 프로토콜 버전을 사용하는 경우 **Samba**에 연결된 **Windows** 클라이언트는 성능 개선의 이점을 누릴 수 있습니다. **Samba**에서 서버 **max** 프로토콜의 기본값은 지원되는 최신 **stable SMB** 프로토콜 버전으로 설정됩니다.



참고

항상 안정적인 최신 **SMB** 프로토콜 버전을 사용하려면 **server max protocol** 매개 변수를 설정하지 마십시오. 매개 변수를 수동으로 설정하는 경우 최신 프로토콜 버전을 사용하도록 새 버전의 **SMB** 프로토콜을 사용하여 설정을 수정해야 합니다.

다음 절차에서는 **server max protocol** 매개 변수에서 기본값을 사용하는 방법을 설명합니다.

절차

1. **/etc/ controlPlane/octets.conf** 파일의 **[global]** 섹션에서 **server max protocol** 매개 변수를 제거합니다.
2. **Samba** 구성 다시 로드

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

1.18.2. 많은 수의 파일이 포함된 디렉터리와의 공유 튜닝

Linux는 대/소문자를 구분하지 않는 파일 이름을 지원합니다. 따라서 파일을 검색하거나 액세스할 때 **Samba**에서 대문자 및 소문자 파일 이름의 디렉터리를 스캔해야 합니다. 소문자 또는 대문자에서만 새 파일을 만들어 성능을 개선하도록 공유를 구성할 수 있습니다.

사전 요구 사항

- **Samba**가 파일 서버로 구성되어 있습니다.

절차

1. 공유의 모든 파일의 이름을 소문자로 바꿉니다.



참고

이 절차의 설정을 사용하여 소문자가 아닌 이름이 있는 파일은 더 이상 표시되지 않습니다.

2. 공유 섹션에서 다음 매개변수를 설정합니다.

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

매개변수에 대한 자세한 내용은 **rootfs.conf(5)** 매뉴얼 페이지에서 해당 설명을 참조하십시오.

3. **/etc/samba/smb.conf** 파일을 확인합니다.

```
# testparm
```

4. **Samba** 구성을 다시 로드합니다.

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

이러한 설정을 적용한 후 이 공유에서 새로 생성된 모든 파일의 이름은 소문자를 사용합니다. 이러한 설정으로 인해 **Samba**는 더 이상 대문자 및 소문자로 디렉터리를 스캔할 필요가 없으므로 성능이 향상됩니다.

니다.

1.18.3. 성능에 부정적인 영향을 줄 수 있는 설정

기본적으로 **Red Hat Enterprise Linux**의 커널은 높은 네트워크 성능을 위해 조정됩니다. 예를 들어 커널은 버퍼 크기에 자동 튜닝 메커니즘을 사용합니다. `/etc/ controlPlane/octets.conf` 파일에서 **socket options** 매개 변수를 설정하면 이러한 커널 설정이 재정의됩니다. 결과적으로 이 매개 변수를 설정하면 대부분의 경우 **Samba** 네트워크 성능이 저하됩니다.

커널에서 최적화된 설정을 사용하려면 `/etc/ controlPlane/octets.conf`의 `[global]` 섹션에서 **socket options** 매개 변수를 제거합니다.

1.19. 기본 버전보다 **SMB** 버전이 필요한 클라이언트와 호환되도록 **SAMBA** 구성

Samba는 지원하는 최소 서버 메시지 블록(**SMB**) 버전에 대해 적절하고 안전한 기본값을 사용합니다. 그러나 이전 **SMB** 버전이 필요한 클라이언트가 있는 경우 이를 지원하도록 **Samba**를 구성할 수 있습니다.

1.19.1. **Samba** 서버에서 지원하는 최소 **SMB** 프로토콜 버전 설정

Samba에서 `/etc/ controlPlane/ECDHE.conf` 파일의 **server min protocol** 매개 변수는 **Samba** 서버가 지원하는 최소 서버 메시지 블록(**SMB**) 프로토콜 버전을 정의합니다. 이 섹션에서는 최소 **SMB** 프로토콜 버전을 변경하는 방법에 대해 설명합니다.



참고

기본적으로 **RHEL 8.2** 이상의 **Samba**는 **SMB2** 및 최신 프로토콜 버전만 지원합니다. **Red Hat**은 더 이상 사용되지 않는 **SMB1** 프로토콜을 사용하지 않을 것을 권장합니다. 그러나 환경에 **SMB1**이 필요한 경우 **server min protocol** 매개 변수를 **NT1**로 수동으로 설정하여 **SMB1**을 다시 활성화할 수 있습니다.

사전 요구 사항



Samba가 설치 및 구성되어 있습니다.

절차

1.

`/etc/steps/ECDHE.conf` 파일을 편집하고 **server min protocol** 매개 변수를 추가하고, 서버에서 지원해야 하는 최소 **SMB** 프로토콜 버전으로 매개 변수를 설정합니다. 예를 들어 최소 **SMB** 프로토콜 버전을 **SMB3**로 설정하려면 다음을 추가합니다.

```
server min protocol = SMB3
```

2.

smb 서비스를 다시 시작하십시오.

```
# systemctl restart smb
```

추가 리소스

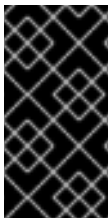
- **smb.conf(5) man page**

1.20. 자주 사용되는 SAMBA 명령줄 유틸리티

이 장에서는 **Samba** 서버로 작업할 때 자주 사용되는 명령에 대해 설명합니다.

1.20.1. 네트워크 광고 조인 및 net#189 조인 명령 사용

net 유틸리티의 **join** 하위 명령을 사용하여 **Samba**를 **AD** 또는 **NT4** 도메인에 결합할 수 있습니다. 도메인에 참여하려면 **/etc/ controlPlane/ECDHE.conf** 파일을 수동으로 생성하고 선택적으로 **PAM**과 같은 추가 구성을 업데이트해야 합니다.



중요

realm 유틸리티를 사용하여 도메인에 가입하는 것이 좋습니다. **realm** 유틸리티는 관련된 모든 구성 파일을 자동으로 업데이트합니다.

절차

1.

다음 설정을 사용하여 **/etc/ controlPlane/ECDHE.conf** 파일을 수동으로 생성합니다.

- **AD** 도메인 멤버의 경우:
- ```
[global]
workgroup = domain_name
security = ads
passdb backend = tdbsam
realm = AD_REALM
```

•

NT4 도메인 멤버의 경우:

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

2.

\* 기본 도메인에 대한 ID 매핑 구성을 추가하고 `/etc/requests/ECDHE.conf` 파일의 `[global]` 섹션에 조인할 도메인에 대한 ID 매핑 구성을 추가합니다.

3.

`/etc/samba/smb.conf` 파일을 확인합니다.

```
testparm
```

4.

도메인 관리자로 도메인에 가입하십시오.

•

AD 도메인에 가입하려면 다음을 수행합니다.

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

•

NT4 도메인에 가입하려면 다음을 수행합니다.

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

5.

`/etc/nsswitch.conf` 파일의 `passwd` 및 `group database` 항목에 `winbind` 소스를 추가합니다.

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

6.

`winbind` 서비스를 활성화하고 시작합니다.

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

7.

선택적으로 `authselect` 유틸리티를 사용하여 **PAM**을 구성합니다.

자세한 내용은 **authselect(8)** 도움말 페이지를 참조하십시오.

8.

**AD 환경의 경우 Kerberos 클라이언트를 구성합니다.**

자세한 내용은 **Kerberos 클라이언트 설명서**를 참조하십시오.

추가 리소스

- [Samba를 도메인에 조인](#) 합니다.
- [Samba ID 매핑 이해 및 구성](#).

### 1.20.2. net rpc 권한 명령 사용

**Windows**에서는 계정 및 그룹에 권한을 할당하여 공유 또는 프린터 드라이버 업로드 또는 업로드와 같은 특수 작업을 수행할 수 있습니다. **Samba** 서버에서는 **net rpc rights** 명령을 사용하여 권한을 관리할 수 있습니다.

설정할 수 있는 권한 나열

사용 가능한 모든 권한 및 소유자를 나열하려면 **net rpc rights list** 명령을 사용합니다. 예를 들어 다음과 같습니다.

```
net rpc rights list -U "DOMAINadministrator"
Enter DOMAINadministrator'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
```

권한 부여

계정 또는 그룹에 권한을 부여하려면 **net rpc rights grant** 명령을 사용합니다.

예를 들어, **DOMAINprintadmin** 그룹에 **SeprintOperatorPrivilege** 권한을 부여합니다.

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

### 권한 해지

계정 또는 그룹의 권한을 취소하려면 **net NetNamespace rights revoke** 명령을 사용합니다.

예를 들어 **DOMAIN\printadmin** 그룹에서 **SeprintOperatorPrivilege** 권한을 취소하려면 다음을 수행합니다.

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

### 1.20.3. net rpc share 명령 사용

**net rpc share** 명령은 로컬 또는 원격 **Samba** 또는 **Windows** 서버에서 공유를 나열, 추가 및 제거하는 기능을 제공합니다.

### 공유 나열

**SMB** 서버의 공유를 나열하려면 **net rpc share list** 명령을 사용합니다. 선택적으로 명령에 **-S server\_name** 매개 변수를 전달하여 원격 서버의 공유를 나열합니다. 예를 들어 다음과 같습니다.

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



### 참고

**/etc/tekton/octets.conf** 파일의 섹션에 **browseable = no setable**이 설정된 **Samba** 서버에서 호스팅되는 공유는 출력에 표시되지 않습니다.

### 공유 추가

**net rpc share add** 명령을 사용하면 **SMB** 서버에 공유를 추가할 수 있습니다.

예를 들어 **C:\example** 디렉터리를 공유하는 원격 **Windows** 서버에 **example**이라는 공유를 추가하려면 다음을 실행합니다.

```
net rpc share add example="C:\example" -U "DOMAINadministrator" -S server_name
```



참고

**Windows** 디렉터리 이름을 지정할 때 경로에서 후행 백슬래시를 생략해야 합니다.

명령을 사용하여 **Samba** 서버에 공유를 추가하려면 다음을 수행합니다.

- **-U** 매개변수에 지정된 사용자에게는 대상 서버에 부여된 **SeDiskOperatorPrivilege** 권한이 있어야 합니다.
- **/etc/samba/smb.conf** 파일에 공유 섹션을 추가하고 **Samba**를 다시 로드하는 스크립트를 작성해야 합니다. 스크립트는 **/etc/tekton/ECDHE.conf**의 **[global]** 섹션의 **add share command** 매개변수에서 설정해야 합니다. 자세한 내용은 **qcow.conf(5)** 매뉴얼 페이지의 **add share** 명령 설명을 참조하십시오.

## 공유 제거

**net rpc share delete** 명령을 사용하면 **SMB** 서버에서 공유를 제거할 수 있습니다.

예를 들어 원격 **Windows** 서버에서 **example**이라는 공유를 제거하려면 다음을 수행합니다.

```
net rpc share delete example -U "DOMAINadministrator" -S server_name
```

명령을 사용하여 **Samba** 서버에서 공유를 제거하려면 다음을 수행합니다.

- **-U** 매개변수에 지정된 사용자에게는 **SeDiskOperatorPrivilege** 권한이 부여되어야 합니다.
- **/etc/samba/smb.conf** 파일에서 공유 섹션을 제거하고 **Samba**를 다시 로드하는 스크립트를 작성해야 합니다. 스크립트는 **/etc/tekton/ECDHE.conf**의 **[global]** 섹션의 **delete share command** 매개변수에서 설정해야 합니다. 자세한 내용은 **rootfs.conf(5)** 매뉴얼 페이지의 **delete share** 명령 설명을 참조하십시오.

### 1.20.4. net user 명령 사용

**net user** 명령을 사용하면 **AD DC** 또는 **NT4 PDC**에서 다음 작업을 수행할 수 있습니다.

- 모든 사용자 계정 나열
- 사용자 추가
- 사용자 제거



참고

**AD** 도메인용 광고 또는 **NT4** 도메인용 **providers**와 같은 연결 방법을 지정하는 것은 도메인 사용자 계정을 나열할 때만 필요합니다. 다른 사용자 관련 하위 명령은 연결 방법을 자동으로 감지할 수 있습니다.

명령에 **-U user\_name** 매개 변수를 전달하여 요청된 작업을 수행할 수 있는 사용자를 지정합니다.

도메인 사용자 계정 나열

**AD** 도메인의 모든 사용자를 나열하려면 다음을 수행하십시오.

```
net ads user -U "DOMAINadministrator"
```

**NT4** 도메인의 모든 사용자를 나열하려면 다음을 수행하십시오.

```
net rpc user -U "DOMAINadministrator"
```

도메인에 사용자 계정 추가

**Samba** 도메인 멤버에서 **net user add** 명령을 사용하여 사용자 계정을 도메인에 추가할 수 있습니다.

예를 들어 **user** 계정을 도메인에 추가합니다.

1. 계정을 추가합니다.

```
net user add user password -U "DOMAINadministrator"
User user added
```

2.

필요한 경우 원격 프로시저 호출(RPC) 셸을 사용하여 AD DC 또는 NT4 PDC에서 계정을 활성화할 수 있습니다. 예를 들어 다음과 같습니다.

```
net rpc shell -U DOMAINadministrator -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
```

도메인에서 사용자 계정 삭제

**Samba** 도메인 멤버에서는 **net user delete** 명령을 사용하여 도메인에서 사용자 계정을 제거할 수 있습니다.

예를 들어 도메인에서 **user** 계정을 제거하려면 다음을 수행합니다.

```
net user delete user -U "DOMAINadministrator"
User user deleted
```

#### 1.20.5. rpcclient 유틸리티 사용

**rpcclient** 유틸리티를 사용하면 로컬 또는 원격 **SMB** 서버에서 클라이언트측 **MS-RPC** 기능을 수동으로 실행할 수 있습니다. 그러나 대부분의 기능은 **Samba**에서 제공하는 별도의 유틸리티에 통합되어 있습니다. **MS-PRC** 함수 테스트를 위해서만 **client** 를 사용하십시오.

사전 요구 사항

- **samba-client** 패키지가 설치되어 있습니다.

예제

예를 들어 **rpcclient** 유틸리티를 사용하여 다음을 수행할 수 있습니다.

- 프린터 **Spool** 하위 시스템(SPOOLSS)을 관리합니다.

예 1.7. **printer**에 드라이버 할당

```
rpcclient server_name -U "DOMAINadministrator" -c 'setdriver "printer_name"
"driver_name"
Enter DOMAINadministrators password:
Successfully set printer_name to driver driver_name.
```

•

SMB 서버에 대한 정보를 검색합니다.

예 1.8. 모든 파일 공유 및 공유 printer 나열

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

•

SCC(Security Account Manager Remote) 프로토콜을 사용하여 작업을 수행합니다.

예 1.9. SMB 서버에 사용자 나열

```
rpcclient server_name -U "DOMAINadministrator" -c 'enumdomusers'
Enter DOMAINadministrators password:
user:[user1] rid:[0x3e8]
user:[user2] rid:[0x3e9]
```

독립 실행형 서버 또는 도메인 구성원에 대해 명령을 실행하면 로컬 데이터베이스의 사용자가 나열됩니다. AD DC 또는 NT4 PDC에 대해 명령을 실행하면 도메인 사용자가 나열됩니다.

추가 리소스

•

NetNamespacedclient(1) 도움말 페이지

#### 1.20.6. samba-regedit 애플리케이션 사용

프린터 구성과 같은 특정 설정은 Samba 서버의 레지스트리에 저장됩니다. ncurses 기반 samba-regedit 애플리케이션을 사용하여 Samba 서버의 레지스트리를 편집할 수 있습니다.

Path: ...AL\_MACHINE/SOFTWARE/Microsoft/Windows NT/CurrentVersion/Print/Printers/

| Key              | Value                                            |
|------------------|--------------------------------------------------|
| <b>Name</b>      | <b>Name</b>   <b>Type</b>   <b>Data</b>          |
| +Example-Printer | Attributes   REG_DWORD   0x00001848 (6216)       |
|                  | ChangeID   REG_DWORD   0x00160374 (1442676)      |
|                  | Datatype   REG_SZ   RAW                          |
|                  | Default Priority   REG_DWORD   0x00000001 (1)    |
|                  | Description   REG_SZ                             |
|                  | Location   REG_SZ                                |
|                  | Name   REG_SZ   Example-Printer                  |
|                  | Parameters   REG_SZ                              |
|                  | Port   REG_SZ   Samba Printer Port               |
|                  | Print Processor   REG_SZ   winprint              |
|                  | Printer Driver   REG_SZ   Example Printer Driver |
|                  | Priority   REG_DWORD   0x00000001 (1)            |
|                  | Security   REG_BINARY   (248 bytes)              |
|                  | Separator File   REG_SZ                          |
|                  | Share Name   REG_SZ   Example-Printer            |
|                  | StartTime   REG_DWORD   0x00000000 (0)           |
|                  | Status   REG_DWORD   0x00000000 (0)              |
|                  | UntilTime   REG_DWORD   0x00000000 (0)           |

[n] New Value [d] Del Value [ENTER] Edit [b] Edit binary **VALUES**  
 [TAB] Switch sections [q] Quit [UP] List up [DOWN] List down [/] Search [x] Next

사전 요구 사항

- **samba-client** 패키지가 설치되어 있습니다.

절차

애플리케이션을 시작하려면 다음을 입력합니다.

```
samba-regedit
```

다음 키를 사용합니다.

- 커서가 위 및 커서: 레지스트리 트리와 값을 이동합니다.
- 키를 열거나 값을 편집합니다.Opens a key or edits a value.
- 탭: 키와 값 창 사이를 전환합니다.

- **Ctrl+C:** 애플리케이션을 닫습니다.

### 1.20.7. smbcontrol 유틸리티 사용

**Net Namespacecontrol** 유틸리티를 사용하면 명령 메시지를 해당 서비스, **nmbd**, **winbindd** 또는 모든 서비스에 보낼 수 있습니다. 이러한 제어 메시지는 예를 들어 구성을 다시 로드하도록 서비스에 지시합니다.

이 섹션의 절차에서는 **reload-config** 메시지 유형을 **all** 대상으로 전송하여 **rootfsd**, **nmbd**, **winbindd** 서비스의 구성을 다시 로드하는 방법을 보여줍니다.

사전 요구 사항

- **samba-common-tools** 패키지가 설치되어 있습니다.

절차

```
smbcontrol all reload-config
```

추가 리소스

- **pvccontrol(1)** 도움말 페이지

### 1.20.8. smbpasswd 유틸리티 사용

**smbpasswd** 유틸리티는 로컬 **Samba** 데이터베이스에서 사용자 계정과 암호를 관리합니다.

사전 요구 사항

- **samba-common-tools** 패키지가 설치되어 있습니다.

절차

1.

명령을 사용자로 실행하는 경우, **rootfs passwd** 는 명령을 실행하는 사용자의 **Samba** 암호를 변경합니다. 예를 들어 다음과 같습니다.

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

2.

**root** 사용자로 **smbpasswd**를 실행하는 경우 유틸리티를 사용하여 다음을 수행할 수 있습니다.

•

새 사용자를 생성합니다.

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



참고

사용자를 **Samba** 데이터베이스에 추가하려면 먼저 로컬 운영 체제에서 계정을 만들어야 합니다. 기본 시스템 설정 구성 가이드 [의 명령줄 섹션에서 새 사용자 추가](#)를 참조하십시오.

•

**Samba** 사용자를 활성화합니다.

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

•

**Samba** 사용자를 비활성화합니다.

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

•

사용자를 삭제합니다.

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

추가 리소스

- **NetNamespaced(8) 도움말 페이지**

### 1.20.9. smbstatus 유틸리티 사용

**Net Namespacestatus** 유틸리티에서 다음을 보고합니다.

- **Samba 서버에 대한 각 Daemon 데몬의 PID당 연결** 이 보고서에는 사용자 이름, 기본 그룹, **SMB** 프로토콜 버전, 암호화 및 서명 정보가 포함됩니다.
- **Samba 공유당 연결.** 이 보고서에는 **rootfs d** 데몬 의 **PID**, 연결 시스템의 **IP**, 연결이 설정된 타임스탬프, 암호화, 서명 정보가 포함됩니다.
- **손상된 파일 목록.** 보고서 항목에는 **opportunistic 잠금(oplock)** 유형과 같은 추가 세부 정보가 포함됩니다.

### 사전 요구 사항

- **samba** 패키지가 설치되어 있습니다.
- **smbd** 서비스가 실행 중입니다.

### 절차

```
smbstatus
```

```
Samba version 4.15.2
```

```
PID Username Group Machine Protocol Version Encryption
Signing
```

```
.....
```

```
-
```

```
963 DOMAINadministrator DOMAINdomain 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 |

추가 리소스

- [pvcstatus\(1\) 도움말 페이지](#)

#### 1.20.10. smbtar 유틸리티 사용

**Net Namespacetar** 유틸리티는 **SMB** 공유 또는 하위 디렉터리의 콘텐츠를 백업하고 해당 콘텐츠를 **tar** 아카이브에 저장합니다. 또는 **media device**에 내용을 쓸 수 있습니다.

사전 요구 사항

- **samba-client** 패키지가 설치되어 있습니다.

절차

- 다음 명령을 사용하여 **//server/example/** 공유에서 **demo** 디렉터리의 콘텐츠를 백업하고 해당 콘텐츠를 **/root/example.tar** 아카이브에 저장합니다.

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

추가 리소스

- [pvctar\(1\) 매뉴얼 페이지](#)

#### 1.20.11. wbinfo 유틸리티 사용

**wbinfo** 유틸리티는 **winbindd** 서비스에서 생성 및 사용하는 정보를 쿼리하고 반환합니다.

사전 요구 사항

- **samba-winbind-clients** 패키지가 설치되어 있습니다.

## 절차

예를 들어 **wbinfo**를 사용하여 다음을 수행할 수 있습니다.

- 목록 도메인 사용자:

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

- 도메인 그룹을 나열합니다.

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

- 사용자의 **SID**를 표시합니다.

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

- 도메인 및 신뢰에 대한 정보를 표시합니다.

```
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 |

## 추가 리소스

- **wbinfo(1)** 도움말 페이지

## 1.21. 추가 리소스

- **Red Hat Samba** 패키지에는 모든 **Samba** 명령 및 구성 파일의 수동 페이지가 포함되어 있으며 패키지가 설치됩니다. 예를 들어, 이 파일에서 설정할 수 있는 모든 구성 매개변수를 설명하는 `/etc/controlPlane/ECDHE.conf` 파일의 도움말 페이지를 표시하려면 다음을 수행합니다.  
  
# man smb.conf
- `/usr/share/docs/915-version/` 디렉터리에는 **Samba** 프로젝트에서 제공하는 일반 문서, 예제 스크립트 및 **LDAP** 스키마 파일이 포함되어 있습니다.
- **Red Hat Cluster Storage 관리 가이드**: **Samba** 설정에 대한 정보를 제공하고 **GlusterFS** 볼륨에 저장된 디렉터리를 공유할 수 있는 **Clustered Trivial Database(CDTB)**를 제공합니다.
- **Red Hat Enterprise Linux에 SMB 공유 영역 마운트.**

## 2장. NFS 공유 내보내기

시스템 관리자는 **NFS** 서버를 사용하여 네트워크를 통해 시스템의 디렉터리를 공유할 수 있습니다.

### 2.1. NFS 소개

이 섹션에서는 **NFS** 서비스의 기본 개념에 대해 설명합니다.

**NFS**(네트워크 파일 시스템)를 사용하면 원격 호스트에서 파일 시스템을 네트워크를 통해 마운트하고 로컬에 마운트된 것처럼 해당 파일 시스템과 상호 작용할 수 있습니다. 이를 통해 네트워크의 중앙 집중식 서버에 리소스를 통합할 수 있습니다.

**NFS** 서버는 `/etc/exports` 구성 파일을 참조하여 내보낸 모든 파일 시스템에 액세스할 수 있는지 여부를 확인합니다. 확인되면 모든 파일 및 디렉터리 작업을 사용자가 사용할 수 있습니다.

### 2.2. 지원되는 NFS 버전

이 섹션에는 **Red Hat Enterprise Linux**에서 지원되는 **NFS** 버전과 해당 기능이 나열되어 있습니다.

현재 **Red Hat Enterprise Linux 9**는 다음과 같은 **NFS** 주요 버전을 지원합니다.

- NFS 버전 3(NFSv3)**은 안전한 비동기 쓰기를 지원하며 이전 **NFSv2**보다 오류 처리 시 더 강력합니다. 64비트 파일 크기와 오프셋도 지원하므로 클라이언트가 2GB 이상의 파일 데이터에 액세스할 수 있습니다.
- NFS 버전 4(NFSv4)**는 방화벽 및 인터넷을 통해 작동하며, 더 이상 **NetNamespace bind** 서비스가 필요하지 않으며, **ACL**(액세스 제어 목록)을 지원하며 상태 저장 작업을 활용합니다.

**NFS 버전 2(NFSv2)**는 **Red Hat**에서 더 이상 지원하지 않습니다.

#### 기본 NFS 버전

**Red Hat Enterprise Linux 9**의 기본 **NFS** 버전은 4.2입니다. **NFS** 클라이언트는 기본적으로 **NFSv4.2**를 사용하여 마운트하려고 시도하며 서버가 **NFSv4.2**를 지원하지 않는 경우 **NFSv4.1**로 대체합니다. 나중에 마운트는 **NFSv4.0**으로 전환된 다음 **NFSv3**에 속합니다.

## 마이너 NFS 버전의 기능

다음은 Red Hat Enterprise Linux 9에서 NFSv4.2의 기능입니다.

### 서버 측 복사

NFS 클라이언트가 `copy_file_range()` 시스템 호출을 사용하여 네트워크 리소스를 사용하지 않고 데이터를 효율적으로 복사할 수 있습니다.

### 스파스 파일

파일에 하나 이상의 홀을 가질 수 있으며, 이는 할당되지 않았거나 초기화되지 않은 데이터 블록으로만 0으로 구성됩니다. NFSv4.2의 `lseek()` 작업은 `trying_ECDHE()` 및 `seek_data()` 를 지원하므로 애플리케이션이 스파스 파일의 홀 위치를 매핑할 수 있습니다.

### 공간 예약

스토리지 서버가 여유 공간을 예약할 수 있도록 허용하여 서버가 공간이 부족해질 수 있습니다. NFSv4.2는 `space`를 예약하기 위해 `allocate()` 작업, `deallocate()` 작업을 지원하여 공간을 할당 해제하고 `fallocate()` 작업을 지원하여 파일의 공간을 미리 할당하거나 할당 해제합니다.

### 레이블이 지정된 NFS

NFS 파일 시스템의 개별 파일에 대해 클라이언트와 서버 간에 SELinux 레이블을 사용할 수 있도록 데이터 액세스 권한을 시행하고, SELinux 레이블을 활성화합니다.

### 레이아웃 개선 사항

일부 병렬 NFS(pNFS) 서버가 더 나은 성능 통계를 수집할 수 있도록 하는 `layoutstats()` 작업을 제공합니다.

다음은 NFSv4.1의 기능은 다음과 같습니다.

- 네트워크의 성능과 보안이 향상되고 pNFS에 대한 클라이언트 쪽 지원도 포함됩니다.
- 콜백에 대한 별도의 TCP 연결이 더 이상 필요하지 않으므로, 이를 통해 NFS 서버에서 클라이언트에 연결할 수 없는 경우에도 위임을 부여할 수 있습니다. 예를 들어 NAT 또는 방화벽이 방해하는 경우입니다.
- 의미 체계가 정확히 한 번 (재부팅 작업을 제외하고) 이전의 문제로 인해 응답이 손실되고 작업이 두 번 전송되면 이전 작업에서 부정확한 결과를 반환할 수 없습니다.

### 2.3. NFSV3 및 NFSV4의 TCP 및 UDP 프로토콜

NFSv4에는 IP 네트워크를 통해 실행되는 TCP(Transmission Control Protocol)가 필요합니다.

NFSv3는 이전 Red Hat Enterprise Linux 버전에서 UDP(User Datagram Protocol)를 사용할 수도 있습니다. Red Hat Enterprise Linux 9에서는 UDP를 통한 NFS가 더 이상 지원되지 않습니다. 기본적으로 UDP는 NFS 서버에서 비활성화되어 있습니다.

### 2.4. NFS에 필요한 서비스

이 섹션에서는 NFS 서버를 실행하거나 NFS 공유를 마운트하는 데 필요한 시스템 서비스를 나열합니다. Red Hat Enterprise Linux는 이러한 서비스를 자동으로 시작합니다.

Red Hat Enterprise Linux는 커널 수준 지원 및 서비스 프로세스를 결합하여 NFS 파일 공유를 제공합니다. 모든 NFS 버전은 클라이언트와 서버 간의 원격 프로시저 호출(RPC)에 의존합니다. NFS 파일 시스템을 공유하거나 마운트하기 위해 구현된 NFS 버전에 따라 다음 서비스가 함께 작동합니다.

#### nfsd

공유 NFS 파일 시스템에 대해 요청하는 NFS 서버 커널 모듈입니다.

#### rpcbind

로컬 RPC 서비스의 포트 예약을 허용합니다. 그런 다음 이러한 포트를 사용할 수 있도록 (또는 광고)하여 해당 원격 RPC 서비스에서 액세스할 수 있습니다. Net Namespacebind 서비스는 RPC 서비스에 대한 요청에 응답하고 요청된 RPC 서비스에 대한 연결을 설정합니다. NFSv4에서는 사용되지 않습니다.

#### rpc.mountd

이 프로세스는 NFS 서버에서 NFSv3 클라이언트의 MOUNT 요청을 처리하는 데 사용됩니다. 요청된 NFS 공유가 현재 NFS 서버에서 내보내고 클라이언트가 액세스할 수 있는지 확인합니다. 마운트 요청이 허용되는 경우 nfs-mountd 서비스는 Success 상태로 응답하고 이 NFS 공유에 대한 File-Handle을 NFS 클라이언트로 다시 제공합니다.

#### rpc.nfsd

이 프로세스를 사용하면 서버가 정의하는 명시적 NFS 버전 및 프로토콜을 사용할 수 있습니다. NFS 클라이언트가 연결할 때마다 서버 스레드를 제공하는 등 NFS 클라이언트의 동적 요구사항을 충족하기 위해 Linux 커널과 함께 작동합니다. 이 프로세스는 nfs-server 서비스에 해당합니다.

#### lockd

이는 클라이언트와 서버 모두에서 실행되는 커널 스레드입니다. NFSv3 클라이언트가 서버의 파

일을 잠글 수 있도록 **NLM(Network Lock Manager)** 프로토콜을 구현합니다. **NFS** 서버가 실행될 때마다 **NFS** 파일 시스템이 마운트될 때마다 자동으로 시작됩니다.

### rpc.statd

이 프로세스는 정상적으로 중단하지 않고 **NFS** 서버를 다시 시작할 때 **NFS** 클라이언트에 알리는 **NSS(Network Status Monitor) RPC** 프로토콜을 구현합니다. **Net Namespace-statd** 서비스는 **nfs-server** 서비스에 의해 자동으로 시작되며 사용자 구성이 필요하지 않습니다. **NFSv4**에서는 사용되지 않습니다.

### rpc.rquotad

이 프로세스는 원격 사용자에게 대한 사용자 할당량 정보를 제공합니다. **quota-rpc** 패키지에서 제공하는 **NetNamespace-rquotad** 서비스는 **nfs-server** 를 시작할 때 사용자가 시작해야 합니다.

### rpc.idmapd

이 프로세스는 유선 **NFSv4** 이름( *사용자@도메인*)과 로컬 **UID**와 **GID** 사이에 매핑되는 **NFSv4** 클라이언트와 **server upcalls**를 제공합니다. **NFSv4**에서 **idmapd** 가 작동하려면 **/etc/idmapd.conf** 파일을 구성해야 합니다. 최소한 **NFSv4** 매핑 도메인을 정의하는 **Domain** 매개 변수를 지정해야 합니다. **NFSv4** 매핑 도메인이 **DNS** 도메인 이름과 동일한 경우 이 매개 변수를 건너뛸 수 있습니다. **ID**가 올바르게 작동하려면 클라이언트와 서버는 **NFSv4** 매핑 도메인에 동의해야 합니다.

**NFSv4** 서버만 **nfs-idmapd** 서비스에 의해 시작되는 **NetNamespace.idmapd** 를 사용합니다. **NFSv4** 클라이언트는 **ID** 매핑을 수행하기 위해 커널에 의해 호출되는 인증 키 기반 **nfsidmap** 유틸리티를 사용합니다. **nfsidmap** 에 문제가 있는 경우 클라이언트는 **NetNamespace.idmapd** 를 사용하도록 돌아갑니다.

## NFSv4를 사용한 RPC 서비스

마운트 및 잠금 프로토콜이 **NFSv4** 프로토콜에 통합되었습니다. 또한 서버는 잘 알려진 **TCP** 포트 **2049**에서 수신 대기합니다. 따라서 **NFSv4**는 **NetNamespace bind,lockd**, **NetNamespace -statd** 서비스와 상호 작용할 필요가 없습니다. **nfs-mountd** 서비스는 내보내기를 설정하기 위해 **NFS** 서버에 여전히 필요하지만 **over-the-wire** 작업과는 관련이 없습니다.

### 추가 리소스

- **NetNamespace bind**를 사용하지 않고 **NFSv4** 전용 서버 구성.

## 2.5. NFS 호스트 이름 형식

이 섹션에서는 **NFS** 공유를 마운트하거나 내보낼 때 호스트를 지정하는 데 사용할 수 있는 다양한 형식에 대해 설명합니다.

다음 형식으로 호스트를 지정할 수 있습니다.

#### 단일 머신

다음 중 하나:

- 정규화된 도메인 이름(서버가 확인할 수 있음)
- 호스트 이름(서버에서 확인할 수 있는)
- IP 주소.

#### IP 네트워크

다음 형식 중 하나가 유효합니다.

- ***a.b.c.d/z*** 입니다. 여기서 ***a.b.c.d*** 는 네트워크이고 ***z*** 는 넷마스크의 비트 수입니다(예: 10.0.0 ).
- ***a.b.c.d /netmask.b.c.d*** 는 네트워크이고 ***넷마스크*** 는 넷마스크입니다(예: 192.168.100.8/255.255.255.0 ).

#### netgroups

**@group-name** 형식. 여기서 **group-name** 은 NIS netgroup 이름입니다.

## 2.6. NFS 서버 구성

이 섹션에서는 NFS 서버에서 내보내기를 구성하는 두 가지 방법의 구문 및 옵션에 대해 설명합니다.

- **/etc/exports** 구성 파일을 수동으로 편집
- 명령줄 에서 **VDD K** 유틸리티 사용

### 2.6.1. /etc/exports 구성 파일

**/etc/exports** 파일은 원격 호스트로 내보낼 파일 시스템을 제어하고 옵션을 지정합니다. 다음 구문 규칙을 따릅니다.

- 빈 줄은 무시됩니다.
- 주석을 추가하려면 해시 표시(#)로 행을 시작합니다.
- 긴 줄을 백슬래시(\)로 래핑할 수 있습니다.
- 내보낸 각 파일 시스템은 고유한 개별 행에 있어야 합니다.
- 내보낸 파일 시스템 뒤에 배치된 권한 있는 호스트 목록은 공백 문자로 구분해야 합니다.
- 각 호스트에 대한 옵션은 호스트와 첫 번째 괄호를 구분하여 호스트 식별자 뒤에 직접 배치해야 합니다.

#### 내보내기 항목

내보낸 파일 시스템의 각 항목에는 다음과 같은 구조가 있습니다.

```
export host(options)
```

각 호스트에 대한 특정 옵션과 함께 여러 호스트를 지정할 수도 있습니다. 이를 위해 다음과 같이 공백으로 구분된 목록과 동일한 줄에 각 호스트 이름 뒤에 해당 옵션(파티프로)을 차례로 표시합니다.

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

이 구조에서:

**export**

내보낼 디렉토리

**host**

내보내기를 공유할 호스트 또는 네트워크

## 옵션

호스트에 사용할 옵션

### 예 2.1. 간단한 `/etc/exports` 파일

가장 간단한 형식의 `/etc/exports` 파일은 내보낸 디렉터리 및 액세스할 수 있는 호스트만 지정합니다.

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

여기서 **bob.example.com** 은 **NFS** 서버에서 `/exported/directory/` 를 마운트할 수 있습니다. 이 예제에 옵션이 지정되지 않았기 때문에 **NFS**는 기본 옵션을 사용합니다.

## 중요

`/etc/exports` 파일의 형식은 특히 공백 문자를 사용하는 것과 관련하여 매우 정확합니다. 내보낸 파일 시스템을 항상 공백 문자를 사용하여 호스트 및 호스트와 서로 분리해야 합니다. 주석 행을 제외하고 파일에는 다른 공백 문자가 없어야 합니다.

예를 들어 다음 두 줄은 동일한 것을 의미하지 않습니다.

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

첫 번째 줄에서는 **bob.example.com** 의 사용자만 `/home` 디렉토리에 대한 읽기 및 쓰기 권한을 허용합니다. 두 번째 행을 사용하면 **bob.example.com** 의 사용자가 디렉토리를 읽기 전용(기본값)으로 마운트할 수 있으며 나머지 사용자는 읽기/쓰기로 마운트할 수 있습니다.

## 기본 옵션

내보내기 항목의 기본 옵션은 다음과 같습니다.

## ro

내보낸 파일 시스템은 읽기 전용입니다. 원격 호스트는 파일 시스템에서 공유하는 데이터를 변경

할 수 없습니다. 호스트가 파일 시스템(즉, 읽기 및 쓰기)을 변경할 수 있도록 하려면 **rw** 옵션을 지정합니다.

## sync

이전 요청의 변경 전에 **NFS** 서버는 요청에 응답하지 않습니다. 대신 비동기 쓰기를 활성화하려면 **async** 옵션을 지정합니다.

## wdelay

다른 쓰기 요청이 무미한 것으로 의심되는 경우 **NFS** 서버는 디스크에 쓰기를 지연합니다. 이렇게 하면 별도의 쓰기 명령으로 디스크에 액세스해야 하는 횟수가 줄어들어 쓰기 오버헤드가 줄어들어 성능을 향상시킬 수 있습니다. 이를 비활성화하려면 기본 동기화 옵션도 지정된 경우에만 사용할 수 있는 **no\_wdelay** 옵션을 지정합니다.

## root\_squash

이렇게 하면 원격에 연결된 루트 사용자가 루트 권한을 갖는 것을 방지할 수 있습니다. 대신, **NFS** 서버는 사용자에게 **ID**를 할당 하지 않습니다. 이를 통해 원격 루트 사용자의 권한을 가장 낮은 로컬 사용자에게 효과적으로 "스쿼시"하여 원격 서버에 무단 쓰기를 방지할 수 있습니다. 루트 스쿼시를 비활성화하려면 **no\_root\_squash** 옵션을 지정합니다.

모든 원격 사용자(**root** 포함)를 스쿼시하려면 **all\_squash** 옵션을 사용합니다. **NFS** 서버가 특정 호스트의 원격 사용자에게 할당해야 하는 사용자 및 그룹 **ID**를 지정하려면 다음과 같이 **anonuid** 및 **anongid** 옵션을 각각 사용합니다.

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

여기서 **uid** 및 **gid**는 각각 사용자 **ID** 번호와 그룹 **ID** 번호입니다. **anonuid** 및 **anongid** 옵션을 사용하면 원격 **NFS** 사용자가 공유할 특수 사용자 및 그룹 계정을 생성할 수 있습니다.

기본적으로 **ACL**(액세스 제어 목록)은 **Red Hat Enterprise Linux**에서 **NFS**에서 지원합니다. 이 기능을 비활성화하려면 파일 시스템을 내보낼 때 **no\_acl** 옵션을 지정합니다.

## 기본 및 재정의 옵션

내보낸 모든 파일 시스템의 각 기본값을 명시적으로 재정의해야 합니다. 예를 들어 **rw** 옵션을 지정하지 않으면 내보낸 파일 시스템이 읽기 전용으로 공유됩니다. 다음은 두 가지 기본 옵션을 재정의하는 **/etc/exports**의 샘플 행입니다.

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

이 예에서 **192.168.0.3**은 **/another/exported/directory/** **read** 및 **write**를 마운트할 수 있으며 디스크에 대한 모든 쓰기는 비동기적입니다.

## 2.6.2. VDDK 유틸리티

VDDK 유틸리티를 사용하면 루트 사용자가 NFS 서비스를 다시 시작하지 않고 디렉토리를 선택적으로 내보내거나 내보내기 해제할 수 있습니다. 적절한 옵션을 고려할 때 **cryptsetup** 유틸리티는 내보낸 파일 시스템을 **/var/lib/nfs/xtab**에 씁니다. **nfs-mountd** 서비스는 파일 시스템에 대한 액세스 권한을 결정할 때 **xtab** 파일을 참조하므로 내보낸 파일 시스템 목록에 대한 변경 사항이 즉시 적용됩니다.

### 일반 **tekton** 옵션

다음은 VDDK에서 사용할 수 있는 일반적으로 사용되는 옵션 목록입니다.

**-r**

**/var/lib/nfs/etab**에 새 내보내기 목록을 구성하여 **/etc/exports**에 나열된 모든 디렉토리를 내보내도록 합니다. 이 옵션은 **/etc/exports**를 변경하여 내보내기 목록을 효과적으로 새로 고칩니다.

**-a**

**NetNamespace**로 전달되는 다른 옵션에 따라 모든 디렉토리를 내보내거나 내보내지 않도록 합니다. 다른 옵션이 지정되지 않은 경우 **ImageCon**은 **/etc/exports**에 지정된 모든 파일 시스템을 내보냅니다.

**-o file-systems**

**/etc/exports**에 나열되지 않은 내보낼 디렉토리를 지정합니다. 파일 시스템을 내보낼 추가 파일 시스템으로 교체합니다. 이러한 파일 시스템은 **/etc/exports**에 지정된 방식과 동일하게 포맷해야 합니다. 이 옵션은 내보낸 파일 시스템 목록에 영구적으로 추가하기 전에 내보낸 파일 시스템을 테스트하는 데 주로 사용됩니다.

**-i**

**/etc/exports**는 무시합니다. 명령줄에 제공된 옵션만 내보낸 파일 시스템을 정의하는 데 사용됩니다.

**-u**

모든 공유 디렉토리를 연결을 해제합니다. VDDK **-ua** 명령은 모든 NFS 서비스를 유지하는 동안 NFS 파일 공유를 일시 중지합니다. NFS 공유를 다시 활성화하려면 **exportfs -r**을 사용합니다.

**-v**

**export** 또는 **unexported** 파일 시스템을 내보낼 때 자세한 내용이 표시되는 자세한 내용이 있는 자세한 정보 표시 작업입니다.

옵션을 **exportfs** 유틸리티로 전달하지 않으면 현재 내보낸 파일 시스템 목록이 표시됩니다.

추가 리소스

- [NFS 호스트 이름 형식.](#)

## 2.7. NFS 및 RPCBIND

이 섹션에서는 **NFSv3**에 필요한 **rpcbind** 서비스의 목적에 대해 설명합니다.

**rpcbind** 서비스는 **RPC**(원격 프로시저 호출) 서비스를 수신 대기하는 포트에 매핑합니다. **RPC** 프로세스는 시작 시 **rpcbind**에 알리고, 수신 대기 중인 포트를 등록하고, 제공할 것으로 예상되는 **RPC** 프로그램 번호를 등록합니다. 클라이언트 시스템은 서버의 **rpcbind**에 특정 **RPC** 프로그램 번호를 사용하여 연결합니다. **rpcbind** 서비스는 클라이언트를 적절한 포트 번호로 리디렉션하여 요청된 서비스와 통신할 수 있도록 합니다.

**RPC** 기반 서비스는 들어오는 클라이언트 요청과 관련된 모든 연결을 만들기 위해 **rpcbind**를 사용하므로 이러한 서비스가 시작되기 전에 **rpcbind**를 사용할 수 있어야 합니다.

**rpcbind**의 액세스 제어 규칙은 모든 **RPC** 기반 서비스에 영향을 미칩니다. 또는 각 **NFS RPC** 데몬에 대한 액세스 제어 규칙을 지정할 수도 있습니다.

추가 리소스

- [rpc.mountd\(8\) man page](#)
- [rpc.statd\(8\) man page](#)

## 2.8. NFS 설치

이 절차에서는 **NFS** 공유를 마운트하거나 내보내는 데 필요한 모든 패키지를 설치합니다.

절차

- **nfs-utils** 패키지를 설치합니다.

```
dnf install nfs-utils
```

## 2.9. NFS 서버 시작

다음 절차에서는 **NFS** 공유를 내보내는 데 필요한 **NFS** 서버를 시작하는 방법을 설명합니다.

사전 요구 사항

- **NFSv3** 연결을 지원하는 서버의 경우 **NetNamespacebind** 서비스가 실행되고 있어야 합니다. **rpcbind** 가 활성화 상태인지 확인하려면 다음 명령을 사용합니다.

```
$ systemctl status rpcbind
```

서비스가 중지되면 서비스를 시작하고 활성화합니다.

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

절차

- **NFS** 서버를 시작하고 부팅 시 자동으로 시작되도록 활성화하려면 다음 명령을 사용합니다.

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

추가 리소스

- [NFSv4 전용 서버 구성.](#)

## 2.10. NFS 및 RPCBIND 문제 해결

**rpcbind** 서비스는 **RPC** 서비스와 통신에 사용되는 포트 번호 간의 조정을 제공하므로 문제 해결 시 **rpcbind** 를 사용하여 현재 **RPC** 서비스의 상태를 확인하는 것이 유용합니다. **rpcinfo** 유틸리티는 포트 번호, **RPC** 프로그램 번호, 버전 번호 및 **IP** 프로토콜 유형(**TCP** 또는 **UDP**)이 있는 각 **RPC** 기반 서비스를 보여줍니다.

절차

1. **rpcbind** 에 적절한 **NFS RPC** 기반 서비스가 활성화되어 있는지 확인하려면 다음 명령을 사용합니다.

```
rpcinfo -p
```

## 예 2.2. rpcinfo -p 명령 출력

다음은 이 명령의 샘플 출력입니다.

```
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
```

**NFS** 서비스 중 하나가 올바르게 시작되지 않으면 **rpcbind** 는 클라이언트의 **RPC** 요청을 해당 서비스에 대한 **RPC** 요청을 올바른 포트에 매핑할 수 없습니다.

2.

대부분의 경우 **NFS**가 **rpcinfo** 출력에 없는 경우 **NFS**를 다시 시작하면 서비스가 **rpcbind** 에 올바르게 등록되고 작업을 시작합니다.

```
systemctl restart nfs-server
```

추가 리소스

- 

**NFSv4 전용 서버 구성.**

## 2.11. 방화벽 뒤에서 실행되도록 NFS 서버 구성

**NFS**에는 **RPC** 서비스에 대한 포트를 동적으로 할당하고 방화벽 규칙을 구성하는 문제를 일으킬 수 있는 **rpcbind** 서비스가 필요합니다. 다음 섹션에서는 지원하려는 경우 방화벽에서 작동하도록 **NFS** 버전을 구성하는 방법을 설명합니다.

- **NFSv3**

여기에는 **NFSv3**를 지원하는 모든 서버가 포함됩니다.

- **NFSv3- 전용 서버**
- **NFSv3 및 NFSv4를 모두 지원하는 서버**

- **NFSv4-전용**

#### 2.11.1. 방화벽 뒤에서 실행되도록 **NFSv3- 사용 서버 구성**

다음 절차에서는 방화벽 뒤에서 실행되도록 **NFSv3**를 지원하는 서버를 구성하는 방법을 설명합니다. 여기에는 **NFSv3** 및 **NFSv4**를 모두 지원하는 **NFSv3** 전용 서버 및 서버가 포함됩니다.

##### 절차

1. 클라이언트가 방화벽 뒤에서 **NFS** 공유에 액세스할 수 있도록 하려면 **NFS** 서버에서 다음 명령을 실행하여 방화벽을 구성합니다.

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

2. 다음과 같이 **/etc/nfs.conf** 파일에서 **RPC** 서비스 **nlockmgr** 에서 사용할 포트를 지정합니다.

```
[lockd]

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

또는 **/etc/modprobe.d/lockd.conf** 파일에 **nlm\_tcpport** 및 **nlm\_udpport**를 지정할 수 있습니다.

3.

NFS 서버에서 다음 명령을 실행하여 방화벽에서 지정된 포트를 엽니다.

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

4.

다음과 같이 `/etc/nfs.conf` 파일의 `[statd]` 섹션을 편집하여 `NetNamespace.statd`에 정적 포트를 추가합니다.

```
[statd]

port=port-number
```

5.

NFS 서버에서 다음 명령을 실행하여 방화벽에서 추가된 포트를 엽니다.

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

6.

방화벽 구성을 다시 로드합니다.

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

7.

먼저 `rpc-statd` 서비스를 다시 시작한 다음 `nfs-server` 서비스를 다시 시작합니다.

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

또는 `/etc/modprobe.d/lockd.conf` 파일에 `lockd` 포트를 지정한 경우:

a.

`/proc/sys/fs/nfs/nlm_tcpport` 및 `/proc/sys/fs/nfs/nlm_udpport`의 현재 값을 업데이트합니다.

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

b.

`rpc-statd` 및 `nfs-server` 서비스를 다시 시작합니다.

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

### 2.11.2. 방화벽 뒤에서 실행되도록 NFSv4 전용 서버 구성

다음 절차에서는 방화벽 뒤에서 실행되도록 **NFSv4** 전용 서버를 구성하는 방법을 설명합니다.

#### 절차

1.

클라이언트가 방화벽 뒤에서 **NFS** 공유에 액세스할 수 있도록 하려면 **NFS** 서버에서 다음 명령을 실행하여 방화벽을 구성합니다.

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

2.

방화벽 구성을 다시 로드합니다.

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

3.

**nfs-server**를 다시 시작합니다.

```
systemctl restart nfs-server
```

### 2.11.3. 방화벽 뒤에서 실행되도록 NFSv3 클라이언트 구성

방화벽 뒤에서 실행되도록 **NFSv3** 클라이언트를 구성하는 절차는 방화벽 뒤에서 실행되도록 **NFSv3** 서버를 구성하는 절차와 유사합니다.

구성 중인 시스템이 **NFS** 클라이언트와 **NFS** 서버 모두인 경우 **NFSv3-enabled** 서버 구성에 설명된 절차를 수행하여 방화벽 뒤에서 실행합니다.

다음 절차에서는 방화벽 뒤에서 실행하기 위해서만 **NFS** 클라이언트인 시스템을 구성하는 방법을 설명합니다.

#### 절차

1.

클라이언트가 방화벽 뒤에 있을 때 **NFS** 서버에서 **NFS** 클라이언트에 콜백을 수행할 수 있도록 하려면 **NFS** 클라이언트에서 다음 명령을 실행하여 방화벽에 **NetNamespace -bind** 서비스를

추가합니다.

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

2.

다음과 같이 `/etc/nfs.conf` 파일에서 **RPC** 서비스 `nlockmgr` 에서 사용할 포트를 지정합니다.

```
[lockd]
```

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

또는 `/etc/modprobe.d/lockd.conf` 파일에 `nlm_tcpport` 및 `nlm_udpport`를 지정할 수 있습니다.

3.

**NFS** 클라이언트에서 다음 명령을 실행하여 방화벽에서 지정된 포트를 엽니다.

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

4.

다음과 같이 `/etc/nfs.conf` 파일의 `[statd]` 섹션을 편집하여 `NetNamespace.statd`에 정적 포트를 추가합니다.

```
[statd]
```

```
port=port-number
```

5.

**NFS** 클라이언트에서 다음 명령을 실행하여 방화벽에서 추가된 포트를 엽니다.

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

6.

방화벽 구성을 다시 로드합니다.

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

7.

`rpc-statd` 서비스를 다시 시작합니다.

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

또는 `/etc/modprobe.d/lockd.conf` 파일에 `lockd` 포트를 지정한 경우:

a.

`/proc/sys/fs/nfs/nlm_tcpport` 및 `/proc/sys/fs/nfs/nlm_udpport` 의 현재 값을 업데이트합니다.

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

b.

`rpc-statd` 서비스를 다시 시작합니다.

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

#### 2.11.4. 방화벽 뒤에서 실행되도록 NFSv4 클라이언트 구성

클라이언트가 **NFSv4.0**을 사용하는 경우에만 이 절차를 수행하십시오. 이 경우 **NFSv4.0** 콜백용 포트를 열어야 합니다.

이후 프로토콜 버전에서는 서버가 클라이언트에서 시작한 동일한 연결에서 콜백을 수행하기 때문에 **NFSv4.1** 이상에는 이 절차가 필요하지 않습니다.

##### 절차

1.

**NFSv4.0** 콜백이 방화벽을 통과하도록 허용하려면 `/proc/sys/fs/nfs_callback_tcpport`를 설정하고 서버가 다음과 같이 클라이언트의 해당 포트에 연결할 수 있습니다.

```
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.

**NFS** 클라이언트에서 다음 명령을 실행하여 방화벽에서 지정된 포트를 엽니다.

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

3.

방화벽 구성을 다시 로드합니다.

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

## 2.12. 방화벽을 통해 RPC 할당량 내보내기

디스크 할당량을 사용하는 파일 시스템을 내보내는 경우 할당량 원격 프로시저 호출(RPC) 서비스를 사용하여 디스크 할당량 데이터를 **NFS** 클라이언트에 제공할 수 있습니다.

### 절차

1.

**rpc-rquotad** 서비스를 활성화하고 시작합니다.

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



참고

**rpc-rquotad** 서비스는 활성화된 경우 **nfs-server** 서비스를 시작한 후 자동으로 시작됩니다.

2.

할당량 **RPC** 서비스를 방화벽 뒤에서 액세스할 수 있도록 하려면 **TCP**(또는 **UDP**가 활성화된 경우 **UDP**) 포트 **875**를 열어야 합니다. 기본 포트 번호는 **/etc/services** 파일에 정의됩니다.

**/etc/sysconfig/rpc-rquotad** 파일의 **RPCRQUOTADOPTS** 변수에 **-p port-number** 를 추가하여 기본 포트 번호를 덮어쓸 수 있습니다.

3.

기본적으로 원격 호스트는 할당량만 읽을 수 있습니다. 클라이언트가 할당량을 설정할 수 있도록 허용하려면 **/etc/sysconfig/rpc-rquotad** 파일의 **RPCRQUOTADOPTS** 변수에 **-S** 옵션을 추가합니다.

4.

**/etc/sysconfig/rpc-rquotad** 파일의 변경 사항을 적용하려면 **rpc-rquotad**를 다시 시작하십시오.

```
systemctl restart rpc-rquotad
```

## 2.13. RDMA를 통한 NFS 활성화 (NFSORDMA)

**RDMA**(Remote Direct Memory Access) 서비스는 Red Hat Enterprise Linux 9의 **RDMA** 가능 하드웨어에서 자동으로 작동합니다.

### 절차

1. **rdma-core** 패키지를 설치합니다.

```
dnf install rdma-core
```

2. **/etc/rdma/modules/rdma.conf** 파일에서 **xprtrdma** 및 **svcrdma** 가 있는 행을 주석 처리했는지 확인합니다.

```
NFS over RDMA client support
xprtrdma
NFS over RDMA server support
svcrdma
```

3. **NFS** 서버에서 디렉토리 **/mnt/nfsordma** 를 만들고 **/etc/exports** 로 내보냅니다.

```
mkdir /mnt/nfsordma
echo "/mnt/nfsordma *(fsid=0,rw,async,insecure,no_root_squash)" >> /etc/exports
```

4. **NFS** 클라이언트에서 서버 IP 주소(예: **172.31.0.186**)를 사용하여 **nfs-share**를 마운트합니다.

```
mount -o rdma,port=20049 172.31.0.186:/mnt/nfs-share /mnt/nfs
```

5. **nfs-server** 서비스를 다시 시작합니다.

```
systemctl restart nfs-server
```

#### 추가 리소스

- [RFC 5667 표준](#)

#### 2.14. 추가 리소스

- [Linux NFS wiki](#)

### 3장. NFS 보안

**NFS** 보안 위험을 최소화하고 서버에서 데이터를 보호하려면 서버에서 **NFS** 파일 시스템을 내보내거나 클라이언트에 마운트할 때 다음 섹션을 고려하십시오.

#### 3.1. AUTH\_SYS 및 내보내기 컨트롤을 사용한 NFS 보안

**NFS**는 내보낸 파일에 대한 액세스를 제어하기 위해 다음과 같은 기존 옵션을 제공합니다.

- 서버는 **IP** 주소 또는 호스트 이름으로 파일 시스템을 마운트할 수 있는 호스트를 제한합니다.
- 서버는 로컬 사용자에게 대해 수행하는 것과 동일한 방식으로 **NFS** 클라이언트의 사용자에게 대해 파일 시스템 권한을 적용합니다. 일반적으로 **NFS**는 클라이언트에 사용자의 **UID** 및 **GID**를 나타내는 **AUTH\_UNIX**라고도 하는 **AUTH\_SYS** 호출 메시지(**AUTH\_UNIX** 라고도 함)를 사용하여 이 작업을 수행합니다. 이는 악성 또는 잘못 구성된 클라이언트가 이 문제를 쉽게 가져올 수 있고 사용자가 하지 않아야 하는 파일에 액세스할 수 있음을 의미합니다.

잠재적인 위험을 제한하기 위해 관리자는 액세스를 일반 사용자 및 그룹 **ID**로 읽기 전용 또는 스쿼시 사용자 권한으로 제한하는 경우가 많습니다. 이러한 솔루션은 원래 의도한 방식으로 **NFS** 공유를 사용하지 못하도록 합니다.

또한 공격자가 **NFS** 파일 시스템을 내보내는 시스템에서 사용하는 **DNS** 서버를 제어하는 경우 특정 호스트 이름 또는 정규화된 도메인 이름과 연결된 시스템을 인증되지 않은 시스템에 가리킬 수 있습니다. 이 시점에서 권한이 없는 시스템은 **NFS** 마운트에 대한 추가 보안을 제공하기 위해 사용자 이름 또는 암호 정보가 교환되지 않으므로 **NFS** 공유를 마운트할 수 있는 시스템입니다.

**NFS**를 통해 디렉토리를 내보낼 때 와일드카드를 사용하여 의도한 것보다 더 많은 시스템을 포함시킬 수 있으므로 와일드카드를 사용해야 합니다.

#### 추가 리소스

- **NFS** 및 **rpcbind** 를 보호하려면 다음을 사용합니다 (예: **nftables** 및 **firewalld** ).
- **nft(8)** 매뉴얼 페이지

- **firewalld-cmd(1) 매뉴얼 페이지**

### 3.2. AUTH\_GSS를 사용한 NFS 보안

모든 버전의 NFS는 **RPCSEC\_GSS** 및 **Kerberos** 메커니즘을 지원합니다.

**AUTH\_SYS**와 달리 **RPCSEC\_GSS Kerberos** 메커니즘과 달리 서버는 클라이언트에 의존하지 않고 어떤 사용자가 파일에 액세스하는지 올바르게 나타냅니다. 대신 암호화는 서버에 사용자를 인증하는 데 사용되며 악의적인 클라이언트가 사용자의 **Kerberos** 자격 증명을 사용하지 않고도 사용자를 가장하지 못하게 합니다. **RPCSEC\_GSS Kerberos** 메커니즘을 사용하는 것은 **Kerberos**를 설정한 후 추가 설정이 필요하지 않기 때문에 마운트를 보호하는 가장 간단한 방법입니다.

### 3.3. KERBEROS를 사용하도록 NFS 서버 및 클라이언트 구성

**Kerberos**는 대칭 암호화와 신뢰할 수 있는 타사인 **KDC**를 사용하여 클라이언트와 서버가 서로 인증할 수 있는 네트워크 인증 시스템입니다. **Red Hat**은 **IdM(Identity Management)**을 사용하여 **Kerberos**를 설정하는 것이 좋습니다.

사전 요구 사항

- **KDC(Kerberos Key Distribution Centre)**가 설치 및 구성되어 있습니다.

절차

1.

- **NFS 서버 측에서 `nfs/hostname.domain@REALM` 주체를 생성합니다.**
- **server 및 클라이언트 측에서 `host/hostname.domain@REALM principal`를 생성합니다.**
- **클라이언트 및 서버의 키탭에 해당 키를 추가합니다.**

2.

서버 측에서 **sec=** 옵션을 사용하여 원하는 보안 플레이버를 활성화합니다. 모든 보안 플레이버 및 비 암호화 마운트를 활성화하려면 다음을 수행합니다.

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

**sec=** 옵션과 함께 사용할 유효한 보안 플레이버는 다음과 같습니다.

- **sys:** 아무 암호화 보호, 기본값
  - **NetNamespace5:** 인증만 가능
  - **NetNamespace5i:** 무결성 보호
    - 사용자 인증에 **Kerberos V5**를 사용하고 데이터 변조를 방지하기 위해 보안 체크섬을 사용하여 **NFS** 작업의 무결성 검사를 수행합니다.
  - **krb5p:** 개인 정보 보호
    - **Kerberos V5**를 사용하여 사용자 인증, 무결성 검사 및 **NFS** 트래픽을 암호화하여 트래픽 스니핑을 방지합니다. 이는 가장 안전한 설정이지만 성능 오버헤드가 가장 많습니다.
3. 클라이언트 측에서 마운트 옵션에 따라 **sa =krb5** (또는 **home =krb5i** 또는 **PS=krb5p**)를 추가합니다.

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

추가 리소스

- [jenkinsfile 5 보안 NFS에서 root로 파일 만들기](#). 권장되지 않음.
- [exports\(5\) 도움말 페이지](#)
- [nfs\(5\) 도움말 페이지](#)

### 3.4. NFSV4 보안 옵션

**NFSv4**에는 **Microsoft Windows NT** 모델의 기능과 광범위한 배포로 인해 **POSIX** 모델이 아닌 **Microsoft Windows NT** 모델을 기반으로 하는 **ACL** 지원이 포함되어 있습니다.

**NFSv4**의 또 다른 중요한 보안 기능은 파일 시스템 마운트에 **MOUNT** 프로토콜 사용을 제거하는 것입니다. **MOUNT** 프로토콜은 프로토콜에서 파일을 처리하는 방식 때문에 보안 위험이 발생했습니다.

### 3.5. 마운트된 NFS 내보내기에 대한 파일 권한

**NFS** 파일 시스템이 원격 호스트에서 읽기 또는 쓰기로 마운트되면 각 공유 파일에 있는 각 공유 파일에 대한 권한만 보호됩니다. 동일한 사용자 ID 값을 공유하는 두 사용자가 다른 클라이언트 시스템에 동일한 **NFS** 파일 시스템을 마운트하는 경우 서로의 파일을 수정할 수 있습니다. 또한 클라이언트 시스템에서 **root**로 로그인한 사용자는 **su -** 명령을 사용하여 **NFS** 공유의 모든 파일에 액세스할 수 있습니다.

기본적으로 **ACL**(액세스 제어 목록)은 **Red Hat Enterprise Linux**의 **NFS**에서 지원합니다. **Red Hat**은 이 기능을 계속 사용하도록 권장합니다.

기본적으로 **NFS**는 파일 시스템을 내보낼 때 루트 스쿼시를 사용합니다. 이렇게 하면 로컬 시스템의 **root** 사용자로 **NFS** 공유에 액세스하는 사용자의 사용자 ID를 **nobody**로 설정합니다. 루트 스쿼싱은 기본 옵션 **root\_squash**에서 제어합니다. 이 옵션에 대한 자세한 내용은 [NFS 서버 구성](#)을 참조하십시오.

**NFS** 공유를 읽기 전용으로 내보내는 경우 **all\_squash** 옵션을 사용하는 것이 좋습니다. 이 옵션을 사용하면 내보낸 파일 시스템에 액세스하는 모든 사용자가 **nobody** 사용자의 사용자 ID를 사용할 수 있습니다.

## 4장. NFS에서 PNFS SCSI 레이아웃 활성화

데이터에 액세스하기 위해 **pNFS SCSI** 레이아웃을 사용하도록 **NFS** 서버와 클라이언트를 구성할 수 있습니다. **pNFS SCSI**는 파일에 대한 장기적인 단일 클라이언트 액세스를 포함하는 사용 사례에서 유용합니다.

### 사전 요구 사항

- 클라이언트와 서버 모두 동일한 블록 장치에 **SCSI** 명령을 보낼 수 있어야 합니다. 즉, 블록 장치는 공유 **SCSI** 버스에 있어야 합니다.
- 블록 장치에는 **XFS** 파일 시스템이 포함되어야 합니다.
- **SCSI** 장치는 **SCSI-3** 기본 명령 사양에 설명된 대로 **SCSI** 영구 예약을 지원해야 합니다.

### 4.1. PNFS 기술

**pNFS** 아키텍처는 **NFS**의 확장성을 향상시킵니다. 서버가 **pNFS**를 구현하면 클라이언트는 동시에 여러 서버를 통해 데이터에 액세스할 수 있습니다. 이로 인해 성능이 향상될 수 있습니다.

**pNFS**는 **RHEL**에서 다음과 같은 스토리지 프로토콜 또는 레이아웃을 지원합니다.

- **Files**
- **Flexfiles**
- **SCSI**

### 4.2. PNFS SCSI 레이아웃

**SCSI** 레이아웃은 **pNFS** 블록 레이아웃 작업에 따라 빌드됩니다. 레이아웃은 **SCSI** 장치 간에 정의됩니다. **SCSI** 영구 예약을 지원할 수 있어야 하는 일련의 고정 크기 블록을 논리 단위(**LU**)로 포함합니다. **LU** 장치는 **SCSI** 장치 식별자로 식별됩니다.

**pNFS SCSI**는 파일에 대해 시간이 오래 걸리는 단일 클라이언트 액세스를 포함하는 사용 사례에서 효과적입니다. 예를 들어 메일 서버 또는 클러스터가 있는 가상 시스템이 있을 수 있습니다.

#### 클라이언트와 서버 간의 작업

**NFS** 클라이언트가 파일에서 읽거나 여기에 쓰는 경우 클라이언트는 **LAYOUTGET** 작업을 수행합니다. 서버는 **SCSI** 장치에 있는 파일의 위치로 응답합니다. 클라이언트는 **GETDEVICEINFO**의 추가 작업을 수행하여 사용할 **SCSI** 장치를 결정해야 할 수 있습니다. 이러한 작업이 올바르게 작동하는 경우 클라이언트는 **READ** 및 **WRITE** 작업을 서버로 보내는 대신 **I/O** 요청을 **SCSI** 장치로 직접 실행할 수 있습니다.

클라이언트 간의 오류 또는 경합으로 인해 서버가 레이아웃을 불러오거나 클라이언트에 발행하지 못할 수 있습니다. 이러한 경우 클라이언트는 **I/O** 요청을 **SCSI** 장치로 직접 보내는 대신 **READ** 및 **WRITE** 작업을 서버에 실행하기 위해 대체됩니다.

작업을 모니터링하려면 **pNFS SCSI 레이아웃 기능 모니터링**을 참조하십시오.

#### 장치 예약

**pNFS SCSI**는 예약을 할당하여 펜싱을 처리합니다. 서버가 클라이언트에 레이아웃을 발행하기 전에 등록된 클라이언트만 장치에 액세스할 수 있도록 **SCSI** 장치를 예약합니다. 클라이언트에서 해당 **SCSI** 장치에 대해 명령을 실행할 수 있지만 장치에 등록되지 않은 경우 해당 장치의 클라이언트에서 많은 작업이 실패합니다. 예를 들어 서버에 클라이언트에 해당 장치의 레이아웃이 지정되지 않은 경우 클라이언트의 **blkid** 명령은 **XFS** 파일 시스템의 **UUID**를 표시하지 못합니다.

서버에서 자체 영구 예약을 제거하지 않습니다. 이렇게 하면 클라이언트 및 서버를 다시 시작할 때마다 장치의 파일 시스템 내의 데이터를 보호합니다. **SCSI** 장치를 다시 사용하려면 **NFS** 서버에서 영구 예약을 수동으로 제거해야 할 수 있습니다.

### 4.3. PNFS와 호환되는 SCSI 장치 확인

이 절차에서는 **SCSI** 장치가 **pNFS SCSI** 레이아웃을 지원하는지 확인합니다.

#### 사전 요구 사항

- **sg3\_utils** 패키지를 설치합니다.

```
dnf install sg3_utils
```

#### 절차

•

서버와 클라이언트 모두에서 적절한 **SCSI** 장치 지원이 있는지 확인합니다.

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

**Persist through Power loss Active(PTP L\_A)** 비트가 설정되었는지 확인합니다.

#### 예 4.1. pNFS SCSI를 지원하는 SCSI 장치

다음은 pNFS SCSI를 지원하는 SCSI 장치의 **sg\_persist** 출력 예입니다. **POST PL\_A** 비트는 1을 보고합니다.

```
inquiry cdb: 12 00 00 00 24 00
Persistent Reservation In cmd: 5e 02 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
```

#### 추가 리소스

•

**sg\_persist(8)** 매뉴얼 페이지

#### 4.4. 서버에서 pNFS SCSI 설정

이 절차에서는 pNFS SCSI 레이어아웃을 내보내도록 NFS 서버를 구성합니다.

#### 절차

1. 서버에서 **SCSI** 장치에 생성된 **XFS** 파일 시스템을 마운트합니다.
2. **NFS** 버전 4.1 이상을 내보내도록 **NFS** 서버를 구성합니다. `/etc/nfs.conf` 파일의 `[nfsd]` 섹션에서 다음 옵션을 설정합니다.

```
[nfsd]
vers4.1=y
```

3. **pnfs** 옵션을 사용하여 **NFS**를 통해 **XFS** 파일 시스템을 내보내도록 **NFS** 서버를 구성합니다.

예 4.2. **pNFS SCSI**를 내보낼 `/etc/exports`의 항목

`/etc/exports` 구성 파일의 다음 항목은 `/exported/directory/`에 마운트된 파일 시스템을 **pNFS SCSI** 레이아웃으로 `allowed.example.com` 클라이언트에 내보냅니다.

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

#### 추가 리소스

- [NFS 공유 내보내기.](#)

#### 4.5. 클라이언트에서 **PNFS SCSI** 설정

이 절차에서는 **pNFS SCSI** 레이아웃을 마운트하도록 **NFS** 클라이언트를 구성합니다.

##### 사전 요구 사항

- **NFS** 서버는 **pNFS SCSI**를 통해 **XFS** 파일 시스템을 내보내도록 구성됩니다. 서버에 **pNFS SCSI** 설정을 참조하십시오.

##### 절차

- 클라이언트에서 **NFS** 버전 4.1 이상을 사용하여 내보낸 **XFS** 파일 시스템을 마운트합니다.

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

**NFS 없이 직접 XFS 파일 시스템을 마운트하지 마십시오.**

#### 추가 리소스

- [NFS 공유 마운트](#).

#### 4.6. 서버에서 PNFS SCSI 예약 해제

이 절차에서는 **NFS** 서버가 **SCSI** 장치에 보유하는 영구 예약을 해제합니다. 이렇게 하면 더 이상 **pNFS SCSI**를 내보낼 필요가 없는 경우 **SCSI** 장치의 용도를 변경할 수 있습니다.

서버에서 예약을 제거해야 합니다. 다른 **IT Nexus**에서 제거할 수 없습니다.

#### 사전 요구 사항

- **sg3\_utils** 패키지를 설치합니다.

```
dnf install sg3_utils
```

#### 절차

1. 서버에서 기존 예약을 쿼리합니다.

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

**예 4.3. /dev/sda에서 예약 쿼리**

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

LIO-ORG block_1 4.0
Peripheral device type: disk
PR generation=0x8, Reservation follows:
Key=0x1000000000000000
scope: LU_SCOPE, type: Exclusive Access, registrants only
```

2. 서버에서 기존 등록을 제거합니다.

```
sg_persist --out \
```

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

#### 예 4.4. /dev/sda에서 예약 제거

```
sg_persist --out \
--release \
--param-rk=0x1000000000000000 \
--prout-type=6 \
/dev/sda
```

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

#### 추가 리소스

- ***sg\_persist(8) 매뉴얼 페이지***