Samba

Samba (https://www.samba.org/) is the standard Windows interoperability suite of programs for Linux and Unix. Since 1992, Samba has provided secure, stable and fast file and print services for all clients using the SMB/CIFS protocol, such as all versions of DOS and Windows, OS/2, Linux and many others.

To share files through Samba, see #Server section; to access files shared through Samba on other machines, please see #Client section.

Related articles

- Active Directory Integration
- Samba/Active Directory domain controller
- NFS

1 Server

1.1 Installation

Install the `samba (https://archlinux.org/packages/?name=samba)` package.

Samba is configured in the `/etc/samba/smb.conf` configuration file, which is extensively documented in `smb.conf(5) (https://man.archlinux.org/man/smb.conf.5)`.

Because the `samba (https://archlinux.org/packages/?name=samba)` package does not provide this file, one needs to create it **before** starting `smb.service`.

A documented example as in `smb.conf.default` from the Samba git repository (https://git.samba.org/samba.git/?p=samba.git;a=blob_plain;f=examples/smb.conf.default;hb=HEAD) may be used to setup `/etc/samba/smb.conf`.

Note:

- The default configuration sets `log file` to a non-writable location, which will cause errors - apply one of the following workarounds:
  - Change the log file location to a writable path: `log file = /var/log/samba/%m.log`
  - Change logging to a non-file backend solution: `logging = syslog` with `syslog only = yes`, or use `logging = systemd`
- If required; the `workgroup` specified in the `[global]` section has to match the Windows workgroup (default `WORKGROUP`).
- The example configuration file exposes the user's home directory to the network with write access. If you see this as a security risk, consider commenting out the entire `[homes]` section. See `smb.conf(5) § The [homes] section (https://man.archlinux.org/man/smb.conf.5#The_%5Bhomes%5D_section)` for details.

Tip: Whenever you modify the `smb.conf` file, run the **testparm(1) (https://man.archlinux.org/man/testparm.1)** command to check for syntactic errors.

1.1.1 Enabling and starting services

To provide basic file sharing through SMB, **enable/start** `smb.service`. See **smbd(8) (https://man.archlinux.org/man/smbd.8)** for details.

If you want to make your server accessible via NetBIOS host name, set the desired name in the `netbios name` option in `smb.conf` and **enable/start** `nmb.service`. See **nmbd(8) (https://man.archlinux.org/man/nmbd.8)** for details.

Note: `nmb.service` is not required. However, it is needed to access Samba servers by hostname (e.g. `smb://hostname/`) for some hosts. If your network is only composed of machines running Windows 10 or later, consider **installing a WSD daemon as well** for your server to appear in the "Network" view.

1.1.2 Make the server discoverable

Install the **avahi (https://archlinux.org/packages/?name=avahi)** package, then **enable/start** `avahi-daemon.service` to make the samba server discoverable with **Zeroconf**. It should work for most non-Windows file managers (macOS Finder, various GUI-based file managers on Linux & BSD etc.)

If `avahi-daemon.service` is not running, the server will still be accessible, just not discoverable, i.e. it will not show up in file managers, but you can still connect to the server directly by IP or domain.

Windows Explorer relies on the WS-Discovery protocol instead; see **#Windows 1709 or up does not discover the samba server in Network view**.

1.1.3 Configure firewall

If you are using a **firewall**, do not forget to open required ports (usually 137-139 + 445). For a complete list, see **Samba port usage (https://www.samba.org/~tpot/articles/firewall.html)**.

1.1.3.1 UFW Rule

A **Ufw** App Profile for SMB/CIFS is included by default with the default installation of UFW in `ufw-fileserver`.

Allow Samba by running `ufw allow CIFS` as root.

If you deleted the profile, create/edit `/etc/ufw/applications.d/samba` and add the following content:

```
[Samba]
title=LanManager-like file and printer server for Unix
description=The Samba software suite is a collection of programs that implements the SMB/CIFS protocol fo
r unix systems, allowing you to serve files and printers to Windows, NT, OS/2 and DOS clients. This proto
col is sometimes also referred to as the LanManager or NetBIOS protocol.
ports=137,138/udp|139,445/tcp
```

Then load the profile into UFW run `ufw app update Samba` as root.

Then finally, allow Samba by running `ufw allow Samba` as root.

1.1.3.2 firewalld service

To configure **firewalld** to allow Samba in the **home** zone, run:

```
# firewall-cmd --permanent --add-service={samba,samba-client,samba-dc} --zone=home
```

The three services listed are:

- `samba` : for sharing files with others.
- `samba-client` : to browse shares on other machines on the network.
- `samba-dc` : for **Samba/Active Directory domain controller**.

`--permanent` ensures the changes remain after `firewalld.service` is **restarted**.

1.2 Basic configuration

1.2.1 User management

The following section describes creating a local (tdbsam) database of Samba users. For user authentication and other purposes, Samba can also be bound to an Active Directory domain, can itself serve as an Active Directory domain controller, or can be used with an LDAP server.

1.2.1.1 Adding a user

Samba requires a Linux user account - you may use an existing user account or create a **new one**.

Note: The **user**/**user group** *nobody* should already exist on the system, it is used as the default `guest account` and may be used for shares containing `guest ok = yes` , thus preventing the need of user login on that share.

Although the user name is shared with Linux system, Samba uses a password separate from that of the Linux user accounts. Replace `samba_user` with the chosen Samba user account:

```
# smbpasswd -a samba_user
```

Depending on the **server role (https://www.samba.org/samba/docs/man/manpages-3/smb.conf.5.html#SERVERROLE)**, existing **File permissions and attributes** may need to be altered for the Samba user account.

If you want the new user only to be allowed to remotely access the file server shares through Samba, you can restrict other login options：

- disabling shell - `usermod --shell /usr/bin/nologin --lock samba_user`
- disabling SSH logons - edit `/etc/ssh/sshd_config` , change option `AllowUsers`

Also see Security for hardening your system.

1.2.1.2 Listing users

Samba users can be listed using the pdbedit(8) (https://man.archlinux.org/man/pdbedit.8) command:

```
# pdbedit -L -v
```

1.2.1.3 Changing user password

To change a user password, use `smbpasswd`:

```
# smbpasswd samba_user
```

1.2.2 Creating an anonymous share

1. Create a Linux user which anonymous Samba users will be mapped to.

```
# useradd guest -s /bin/nologin
```

Note: The username can be any valid Linux username, not just "guest". This user does not need to be a Samba user.

2. Add the following to `/etc/samba/smb.conf`:

/etc/samba/smb.conf

```
...
[global]
security = user
map to guest = bad user
guest account = guest

[guest_share]
    comment = guest share
    path = /tmp/
    public = yes
    only guest = yes
    writable = yes
    printable = no
```

Anonymous users will now be mapped to the Linux user `guest` and have the ability to access any directories defined in `guest_share.path`, which is configured to be `/tmp/` in the example above.

Note: The share name does not have to have "guest" in it. It can be any valid Samba share name.

Make sure that the Linux user `guest` has the proper permissions to access files in `guest_share.path`.

Also, make sure shares have been properly defined as per the *Share Definitions* section of smb.conf.default (https://git.samba.org/samba.git/?p=samba.git;a=blob_plain;f=examples/smb.conf.default;hb=HEAD).

1.3 Advanced configuration

1.3.1 Enable symlink following

Warning: Enabling the `follow symlinks` option can be a security risk.

/etc/samba/smb.conf

```
...
[global]
   follow symlinks = yes
   wide links = yes
   unix extensions = no
```

Then, restart `smb.service`.

Note: When using AppArmor, if the symlink points to a directory outside the user's home or the usershare directory, then you need to modify the AppArmor profile permissions.

1.3.2 Enable server-side copy for macOS clients

Server-side copy eliminates the need to transfer data between the server and the client when copying files on the server. This is enabled by default, but it doesn't work with macOS clients. If you have macOS clients, you need to add the following configuration to `smb.conf` and then restart `smb.service`.

/etc/samba/smb.conf

```
...
[global]
   fruit:copyfile = yes
```

1.3.3 Enable Usershares

Note: This is an optional feature. Skip this section if you do not need it.

Usershares is a feature that gives non-root users the capability to add, modify, and delete their own share definitions. See **smb.conf(5) § USERSHARES (https://man.archlinux.org/man/smb.conf.5#USERSHARES)**.

1. Create a directory for usershares:

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

2. Create a user group:

```
# groupadd -r sambashare
```

3. Change the owner of the directory to `root` and the group to `sambashare`:

```
# chown root:sambashare /var/lib/samba/usershares
```

4. Change the permissions of the `usershares` directory so that users in the group `sambashare` can create files. This command also sets **sticky bit**, which is important to prevent users from deleting usershares of other users:

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

Set the following parameters in the `smb.conf` configuration file:

```
/etc/samba/smb.conf
```

```
[global]
  usershare path = /var/lib/samba/usershares
  usershare max shares = 100
  usershare allow guests = yes
  usershare owner only = yes
```

Add the user to the *sambashare* group. Replace `your_username` with the name of your user:

```
# gpasswd sambashare -a your_username
```

Restart `smb.service` and `nmb.service` services.

Log out and log back in.

If you want to share paths inside your home directory you must make it accessible for the group *others*.

In the GUI, you can use **Thunar** or **Dolphin** - right click on any directory and share it on the network.

In the CLI, use one of the following commands, replacing italic *sharename*, *user*, ... :

```
# net usershare add sharename abspath [comment] [user:{R|D|F}] [guest_ok={y|n}]
# net usershare delete sharename
# net usershare list wildcard-sharename
# net usershare info wildcard-sharename
```

1.3.4 Set and forcing permissions

Permissions may be applied to both the server and shares:

```
/etc/samba/smb.conf
```

```
[global]
  ;inherit owner = unix only ; Inherit ownership of the parent directory for new files and directories
  ;inherit permissions = yes ; Inherit permissions of the parent directory for new files and directories
  create mask = 0664
  directory mask = 2755
  force create mode = 0644
  force directory mode = 2755
  ...

[media]
```

```
  comment = Media share accessible by greg and pcusers
  path = /path/to/media
  valid users = greg @pcusers
  force group = +pcusers
  public = no
  writable = yes
  create mask = 0664
  directory mask = 2775
  force create mode = 0664
  force directory mode = 2775

[public]
  comment = Public share where archie has write access
  path = /path/to/public
  public = yes
  read only = yes
  write list = archie
  printable = no

[guests]
  comment = Allow all users to read/write
  path = /path/to/guests
  public = yes
  only guest = yes
  writable = yes
  printable = no
```

See **smb.conf(5) (https://man.archlinux.org/man/smb.conf.5)** for a full overview of possible permission flags and settings.

1.3.5 Restrict protocols for better security

Warning: By default, Samba versions prior to 4.11 allow connections using the outdated and insecure SMB1 protocol. When using one these Samba versions, it is highly recommended to set `server min protocol = SMB2_02` to protect yourself from ransomware attacks. In Samba 4.11 and newer, SMB2 is the default min protocol, so no changes are required there.

Append `server min protocol` and `server max protocol` in `/etc/samba/smb.conf` to force usage of a minimum and maximum protocol:

/etc/samba/smb.conf

```
[global]
  server min protocol = SMB2_02
  ; server max protocol = SMB3
```

See `server max protocol` in **smb.conf(5) (https://man.archlinux.org/man/smb.conf.5)** for an overview of supported protocols.

For compatibility with older clients and/or servers, you might need to set `client min protocol = CORE` or `server min protocol = CORE`, but please note that this makes you vulnerable to exploits in SMB1 including ransomware attacks.

Tip: Use `server min protocol = SMB3_00` when clients should only connect using the latest SMB3 protocol, e.g. on clients running Windows 8 and later.

Clients using `mount.cifs` may need to specify the correct `vers=*`, e.g.:

```
# mount -t cifs //SERVER/sharename /mnt/mountpoint -o username=username,password=password,iocharset=utf8,
```

```
vers=3.1.1
```

See **mount.cifs(8) (https://man.archlinux.org/man/mount.cifs.8)** for more information.

1.3.6 Use native SMB transport encryption

Native SMB transport encryption is available in SMB version 3.0 or newer. Clients supporting this type of encryption include Windows 8 and newer, Windows server 2012 and newer, and smbclient of Samba 4.1 and newer.

To use native SMB transport encryption by default, set the `server smb encrypt` parameter globally and/or by share. Possible values are `off`, `enabled` (default value), `desired`, or `required`:

/etc/samba/smb.conf

```
[global]
  server smb encrypt = desired
```

To configure encryption for on the client side, use the option `client smb encrypt`.

See **smb.conf(5) (https://man.archlinux.org/man/smb.conf.5)** for more information, especially the paragraphs *Effects for SMB1* and *Effects for SMB2*.

Tip: When **mounting** a share, specify the `seal` mount option to force usage of encryption.

1.3.7 Disable printer sharing

By default Samba shares printers configured using **CUPS**.

If you do not want printers to be shared, use the following settings:

/etc/samba/smb.conf

```
[global]
  load printers = no
  printing = bsd
  printcap name = /dev/null
  disable spoolss = yes
  show add printer wizard = no
```

1.3.8 Block certain file extensions on Samba share

Note: Setting this parameter will affect the performance of Samba, as it will be forced to check all files and directories for a match as they are scanned.

Samba offers an option to block files with certain patterns, like file extensions. This option can be used to prevent dissemination of viruses or to dissuade users from wasting space with certain files. More information about this option can be found in **smb.conf(5) (https://man.archlinux.org/man/smb.conf.5)**.

/etc/samba/smb.conf

```
...
[myshare]
  comment = Private
  path = /mnt/data
  read only = no
  veto files = /*.exe/*.com/*.dll/*.bat/*.vbs/*.tmp/*.mp3/*.avi/*.mp4/*.wmv/*.wma/
```

1.3.9 Improve throughput

Warning: Beware this may lead to corruption/connection issues and potentially cripple your TCP/IP stack.

The default settings should be sufficient for most users. However setting the 'socket options' correct can improve performance, but getting them wrong can degrade it by just as much. Test the effect before making any large changes.

Read the **smb.conf(5) (https://man.archlinux.org/man/smb.conf.5)** man page before applying any of the options listed below.

The following settings should be **appended** to the `[global]` section of `/etc/samba/smb.conf`.

Setting a deadtime is useful to stop a server's resources from being exhausted by a large number of inactive connections:

```
deadtime = 30
```

The usage of sendfile may make more efficient use of the system CPU's and cause Samba to be faster:

```
use sendfile = yes
```

Setting min receivefile size allows zero-copy writes directly from network socket buffers into the filesystem buffer cache (if available). It may improve performance but user testing is recommended:

```
min receivefile size = 16384
```

Increasing the receive/send buffers size and socket optimize flags might be useful to improve throughput. It is recommended to test each flag separately as it may cause issues on some networks:

```
socket options = IPTOS_LOWDELAY TCP_NODELAY IPTOS_THROUGHPUT SO_RCVBUF=131072 SO_SNDBUF=131072
```

Note: Network-interface adjustments may be needed for some options to work, see **Sysctl#Networking**.

1.3.10 Enable access for old clients/devices

Latest versions of Samba no longer offer older authentication methods and protocols which are still used by some older clients (IP cameras, etc). These devices usually require Samba server to allow NTMLv1 authentication and NT1 version of the protocol, known as CIFS. For these devices to work with latest Samba, you need to add these two configuration parameters into `[global]` section:

```
server min protocol = NT1
ntlm auth = yes
```

Anonymous/guest access to a share requires just the first parameter. If the old device will access with username and password, you also need the add the second line too.

1.3.11 Enable Spotlight searching

Spotlight allows supporting clients (e.g. MacOS Finder) to quickly search shared files.

Install and start/enable OpenSearch. Install **fs2es-indexer (https://aur.archlinux.org/packages/fs2es-indexer/)**[AUR], configure the directories you want to index in `/etc/fs2es-indexer/config.yml`, and start/enable `fs2es-indexer.service` for periodic indexing.

Edit `smb.conf` as described in the Samba wiki (https://wiki.samba.org/index.php/Spotlight_with_Elasticsearch_Backend#Samba) to enable Spotlight per share, and restart `smb.service` to apply the changes.

2 Client

Install **smbclient (https://archlinux.org/packages/?name=smbclient)** for an `ftp`-like command line interface. See **smbclient(1) (https://man.archlinux.org/man/smbclient.1)** for commonly used commands.

For a lightweight alternative (without support for listing public shares, etc.), install **cifs-utils (https://archlinux.org/packages/?name=cifs-utils)** that provides `/usr/bin/mount.cifs`.

Depending on the desktop environment, GUI methods may be available. See #File manager configuration for use with a file manager.

Note:

- **smbclient (https://archlinux.org/packages/?name=smbclient)** requires a `/etc/samba/smb.conf` file (see #Installation), which you can create as an empty file using the `touch` utility.
- After installing **cifs-utils (https://archlinux.org/packages/?name=cifs-utils)** or **smbclient (https://archlinux.org/packages/?name=smbclient)**, load the `cifs` kernel module or reboot to prevent mount fails.

2.1 List public shares

The following command lists public shares on a server:

```
$ smbclient -L hostname -U%
```

Alternatively, running `$ smbtree -N` will show a tree diagram of all the shares. It uses broadcast queries and is therefore not advisable on a network with a lot of computers, but can be helpful for diagnosing if you have the correct sharename. The `-N` (`-no-pass`) option suppresses the password

prompt.

Note: `smbtree` uses SMB1 and NetBIOS, which means they must be enabled on the servers and you need to set `client min protocol = NT1` in `smb.conf` on the client. Otherwise, `smbtree` will show empty output.

2.2 NetBIOS/WINS host names

Samba clients handle NetBIOS host names automatically by default (the behavior is controlled by the `name resolve order` option in `smb.conf`). Other programs (including `mount.cifs`) typically use Name Service Switch, which does not handle NetBIOS by default.

The **`smbclient (https://archlinux.org/packages/?name=smbclient)`** package provides a libnss driver to resolve NetBIOS host names. To use it, install it along with the **`samba (https://archlinux.org/packages/?name=samba)`** package (which provides the *winbindd* daemon), start/enable `winbind.service` and add `wins` to the `hosts` line in **`nsswitch.conf(5) (https://man.archlinux.org/man/nsswitch.conf.5)`**:

/etc/nsswitch.conf

```
...
hosts: mymachines resolve [!UNAVAIL=return] files myhostname dns wins
...
```

Note: Due to a current mistake in `winbind.service`, you may have to modify the unit file as described in this **bug-report (https://bugs.launchpad.net/ubuntu/+source/samba/+bug/1789097)**

Now, during host resolving (e.g. when using `mount.cifs` or just `ping` *`netbios-name`*), *winbindd* will resolve the host name by sending queries using NetBIOS Name Service (NBNS, also known as WINS) protocol.

By default it sends a broadcast query to your local network. If you have a WINS server, you can add `wins server =` *`wins-server-ip`* to `smb.conf` and restart `winbind.service`, then *winbindd* and other Samba clients will send unicast queries to the specified IP.

If you want to resolve your local host name (specified in the `netbios name` option in `smb.conf`), start/enable `nmb.service`, which will handle incoming queries.

You can test WINS resolution with `nmblookup`. By default it sends broadcast queries to your local network regardless of the `wins server` option.

Note that WINS resolution requires incoming traffic originating from port 137.

2.2.1 Disable NetBIOS/WINS support

When not using NetBIOS/WINS host name resolution, it may be preferred to disable this protocol:

/etc/samba/smb.conf

```
[global]
  disable netbios = yes
  dns proxy = no
```

Finally **disable**/**stop** `winbind.service`.

2.3 Manual mounting

Mount the share using `mount.cifs` as `type`. Not all the options listed below are needed or desirable:

```
# mount --mkdir -t cifs //SERVER/sharename /mnt/mountpoint -o username=username,password=password,workgro
up=workgroup,iocharset=utf8,uid=username,gid=group
```

The options `uid` and `gid` corresponds to the local (e.g. client) **user**/**user group** to have read/write access on the given path.

Note:

- If the `uid` and `gid` being used does not match the user of the server, the `forceuid` and `forcegid` options may be helpful. However note permissions assigned to a file when `forceuid` or `forcegid` are in effect may not reflect the real (server) permissions. See the *File And Directory Ownership And Permissions* section in **mount.cifs(8) § FILE AND DIRECTORY OWNERSHIP AND PERMISSIONS (https://man.archlinux.org/man/mount.cifs.8#FILE_AND_DIRECTORY_OWNERSHIP_AND_PERMISSIONS)** for more information.
- To mount a Windows share without authentication, use `"username=*"`.

Warning: Using `uid` and/or `gid` as mount options may cause I/O errors, it is recommended to set/check correct **File permissions and attributes** instead.

- *SERVER* — The server name.
- *sharename* — The shared directory.
- *mountpoint* — The local directory where the share will be mounted.
- `[-o` *options*`]` — See **mount.cifs(8) (https://man.archlinux.org/man/mount.cifs.8)** for more information.

Note:

- Abstain from using a trailing `/`. `//SERVER/sharename/` will not work.
- If your mount does not work stable, stutters or freezes, try to enable different SMB protocol version with `vers=` option. For example, `vers=2.0` for Windows Vista mount.
- If having timeouts on a mounted network share with cifs on a shutdown, see **wpa_supplicant#Problem with mounted network shares (cifs) and shutdown**.

2.3.1 Storing share passwords

Storing passwords in a world readable file is not recommended. A safer method is to use a credentials file instead, e.g. inside `/etc/samba/credentials`:

```
/etc/samba/credentials/share
```

```
username=myuser
```

```
password=mypass
```

For the mount command replace `username=myuser,password=mypass` with `credentials=/etc/samba/credentials/share`.

The credential file should explicitly readable/writeable to root:

```
# chown root:root /etc/samba/credentials
# chmod 700 /etc/samba/credentials
# chmod 600 /etc/samba/credentials/share
```

2.4 Automatic mounting

Note: You may need to **enable** `systemd-networkd-wait-online.service` or `NetworkManager-wait-online.service` (depending on your setup) to proper enable booting on start-up.

2.4.1 Using NetworkManager and GIO/gvfs

NetworkManager can be configured to run a script on network status change. This script uses the *gio* command so that it mounts the Samba shares automatically, the same way your file manager does, as explained **below**. The script also safely unmounts the Samba shares before the relevant network connection is disabled by listening for the `pre-down` and `vpn-pre-down` events. Make the script is **executable** after creating it.

/etc/NetworkManager/dispatcher.d/30-samba.sh

```
#!/bin/sh

# Find the connection UUID with "nmcli con show" in terminal.
# All NetworkManager connection types are supported: wireless, VPN, wired...
WANTED_CON_UUID="CHANGE-ME-NOW-9c7eff15-010a-4b1c-a786-9b4efa218ba9"

# The user the share will be mounted under
USER="yourusername"
# The path that appears in your file manager when you manually mount the share you want
SMB_URL="smb://servername/share"

# Get runtime user directory. If it does not exist, do nothing and just exit
XDG_RUNTIME_DIR=$(loginctl show-user --property=RuntimePath --value "$USER") || exit 0

if [ "$CONNECTION_UUID" = "$WANTED_CON_UUID" ]; then

    # Script parameter $1: network interface name, not used
    # Script parameter $2: dispatched event

    case "$2" in
        "up"|"vpn-up")
            su $USER -c "DBUS_SESSION_BUS_ADDRESS=unix:path=$XDG_RUNTIME_DIR/bus gio mount $SMB_URL"
            ;;
        "pre-down"|"vpn-pre-down")
            su $USER -c "DBUS_SESSION_BUS_ADDRESS=unix:path=$XDG_RUNTIME_DIR/bus gio mount -uf $SMB_URL"
            ;;
    esac
fi
```

Create a symlink inside `/etc/NetworkManager/dispatcher.d/pre-down` to catch the `pre-down` events:

```
# ln -s /etc/NetworkManager/dispatcher.d/30-samba.sh /etc/NetworkManager/dispatcher.d/pre-down.d/30-samba.sh
```

Note: Since this script uses the user bus, it will only work if the user has active sessions. This means that the share will not mount automatically after boot if the connection is established before you are logged in.

2.4.2 As mount entry

This is a simple example of a `cifs` **mount entry** that requires authentication:

/etc/fstab

```
//SERVER/sharename /mnt/mountpoint cifs _netdev,nofail,username=myuser,password=mypass 0 0
```

Note:

- Spaces in sharename should be replaced by `\040` (ASCII code for space in octal). For example, `//SERVER/share name` on the command line should be `//SERVER/share\040name` in `/etc/fstab`.
- To allow users to mount it as long as the mount point resides in a directory controllable by the user; i.e. the user's home, append the `users` mount option. The option is user**s** (plural). For other filesystem types handled by mount, this option is usually *user*; sans the "**s**".

Tip: Use `x-systemd.automount` if you want them to be mounted only upon access. See **Fstab#Remote file system** for details.

2.4.3 As systemd unit

Create a new `.mount` file inside `/etc/systemd/system`, e.g. `mnt-myshare.mount`. See **systemd.mount(5) (https://man.archlinux.org/man/systemd.mount.5)** for details.

Note: Make sure the filename corresponds to the mountpoint you want to use. E.g. the unit name `mnt-myshare.mount` can only be used if are going to mount the share under `/mnt/myshare`. Otherwise the following error might occur:
`systemd[1]: mnt-myshare.mount: Where= setting does not match unit name. Refusing.`
.

`What=` path to share

`Where=` path to mount the share

`Options=` share mounting options

Note:

- Network mount units automatically acquire `After` dependencies on `remote-fs-pre.target`, `network.target` and `network-online.target`, and

gain a `Before` dependency on `remote-fs.target` unless `nofail` mount option is set. Towards the latter a `Wants` unit is added as well.
- **Append** `noauto` to `Options` preventing automatically mount during boot (unless it is pulled in by some other unit).
- If you want to use a hostname for the server you want to share (instead of an IP address), add `nss-lookup.target` to `After`. This might avoid mount errors at boot time that do not arise when testing the unit.

`/etc/systemd/system/mnt-myshare.mount`

```
[Unit]
Description=Mount Share at boot

[Mount]
What=//server/share
Where=/mnt/myshare
Options=_netdev,credentials=/etc/samba/credentials/myshare,iocharset=utf8,rw
Type=cifs
TimeoutSec=30

[Install]
WantedBy=multi-user.target
```

Tip:

- In case of an unreachable system, **append** `ForceUnmount=true` to `[Mount]`, allowing the share to be (force-)unmounted.
- If your share has groups with read-only access, **append** `uid=`*`username`* or `gid=`*`group`* to `Options=`, to specify your user / group allowing writing to the share.

To use `mnt-myshare.mount`, **start** the unit and **enable** it to run on system boot.

2.4.3.1 automount

To automatically mount a share (when accessed, like autofs), one may use the following automount unit:

`/etc/systemd/system/mnt-myshare.automount`

```
[Unit]
Description=Automount myshare

[Automount]
Where=/mnt/myshare

[Install]
WantedBy=multi-user.target
```

Disable/**stop** the `mnt-myshare.mount` unit, and **enable**/**start** `mnt-myshare.automount` to automount the share when the mount path is being accessed.

Tip: Append `TimeoutIdleSec` to enable auto unmount. See **systemd.automount(5) (https://man.archlinux.org/man/systemd.automount.5)** for details.

2.4.4 smbnetfs

Note: smbnetfs needs an intact Samba server setup. See above on how to do that.

First, check if you can see all the shares you are interested in mounting:

```
$ smbtree -U remote_user
```

If that does not work, find and modify the following line in `/etc/samba/smb.conf` accordingly:

```
domain master = auto
```

Now restart `smb.service` and `nmb.service`.

If everything works as expected, install **smbnetfs (https://archlinux.org/packages/?name=smbnetfs)**.

Then, add the following line to `/etc/fuse.conf`:

```
user_allow_other
```

Now copy the directory `/etc/smbnetfs/.smb` to your home directory:

```
$ cp -a /etc/smbnetfs/.smb ~
```

Then create a link to `smb.conf`:

```
$ ln -sf /etc/samba/smb.conf ~/.smb/smb.conf
```

If a username and a password are required to access some of the shared folders, edit `~/.smb/smbnetfs.auth` to include one or more entries like this:

~/.smb/smbnetfs.auth

```
auth                    "hostname" "username" "password"
```

It is also possible to add entries for specific hosts to be mounted by smbnetfs, if necessary. More details can be found in `~/.smb/smbnetfs.conf`.

If you are using the Dolphin or GNOME Files, you may want to add the following to `~/.smb/smbnetfs.conf` to avoid "Disk full" errors as smbnetfs by default will report 0 bytes of free space:

~/.smb/smbnetfs.conf

```
free_space_size 1073741824
```

When you are done with the configuration, you need to run

```
$ chmod 600 ~/.smb/smbnetfs.*
```

Otherwise, smbnetfs complains about 'insecure config file permissions'.

Finally, to mount your Samba network neighbourhood to a directory of your choice, call

```
$ smbnetfs mount_point
```

2.4.4.1 Daemon

The Arch Linux package also maintains an additional system-wide operation mode for smbnetfs. To enable it, you need to make the said modifications in the directory `/etc/smbnetfs/.smb`.

Then, you can start and/or enable the `smbnetfs` **daemon** as usual. The system-wide mount point is at `/mnt/smbnet/`.

2.4.5 autofs

See **Autofs** for information on the kernel-based automounter for Linux.

2.5 File manager configuration

2.5.1 GNOME Files, Nemo, Caja, Thunar and PCManFM

In order to access samba shares through GNOME Files, Nemo, Caja, Thunar or PCManFM, install the **gvfs-smb (https://archlinux.org/packages/?name=gvfs-smb)** package.

Press `Ctrl+l` and enter `smb://servername/share` in the location bar to access your share.

The mounted share is likely to be present at `/run/user/your_UID/gvfs` or `~/.gvfs` in the filesystem.

2.5.2 KDE

KDE applications (like Dolphin) has the ability to browse Samba shares built in. Use the path `smb://servername/share` to browse the files. If you want to access files from on non-KDE application, you can install **kio-fuse (https://archlinux.org/packages/?name=kio-fuse)**.

To use a GUI in the KDE System Settings, you will need to install the **kdenetwork-filesharing (https://archlinux.org/packages/?name=kdenetwork-filesharing)** package.

2.5.3 Other graphical environments

There are a number of useful programs, but they may need to have packages created for them. This can be done with the Arch package build system. The good thing about these others is that they do not require a particular environment to be installed to support them, and so they bring along less baggage.

- pyneighborhood (https://aur.archlinux.org/packages/pyneighborhood/) AUR
- LinNeighborhood, RUmba, xffm-samba plugin for Xffm are not available in the official repositories or the AUR. As they are not officially (or even unofficially supported), they may be obsolete and may not work at all.

3 Tips and tricks

3.1 Discovering network shares

If nothing is known about other systems on the local network, and automated tools such as smbnetfs are not available, you can manually probe for Samba shares.

First, install the nmap (https://archlinux.org/packages/?name=nmap) and smbclient (https://archlinux.org/packages/?name=smbclient) packages.

Use nmap to scan your local network to find systems with TCP port 445 open, which is the port used by the SMB protocol. Note that you may need to use `-Pn` or set a custom ping scan type (e.g. `-PS445`) because Windows systems are usually firewalled.

```
$ nmap -p 445 "192.168.1.*"
```

```
Starting Nmap 7.92 ( https://nmap.org ) at 2022-03-13 12:00 UTC
Nmap scan report for 192.168.1.1
Host is up (0.0011s latency).

PORT    STATE  SERVICE
445/tcp open  microsoft-ds

Nmap scan report for 192.168.1.2
Host is up (0.00011s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds

Nmap done: 256 IP addresses (2 hosts up) scanned in 2.45 seconds
```

The first result is another system; the second happens to be the client from where this scan was performed.

Now you can connect to their IP addresses directly, but if you want to use NetBIOS host names, you can use nmblookup(1) (https://man.archlinux.org/man/nmblookup.1) to check for NetBIOS names. Note that this will not work if NetBIOS is disabled on the server.

```
$ nmblookup -A 192.168.1.1
```

```
Looking up status of 192.168.1.1
        PUTER           <00> -         B <ACTIVE>
        HOMENET         <00> - <GROUP> B <ACTIVE>
        PUTER           <03> -         B <ACTIVE>
        PUTER           <20> -         B <ACTIVE>
        HOMENET         <1e> - <GROUP> B <ACTIVE>
        USERNAME        <03> -         B <ACTIVE>
        HOMENET         <1d> -         B <ACTIVE>
        MSBROWSE        <01> - <GROUP> B <ACTIVE>
```

Regardless of the output, look for **<20>**, which shows the host with open services.

Use `smbclient(1)` (https://man.archlinux.org/man/smbclient.1) to list which services are shared on these systems. You can use NetBIOS host name (`PUTER` in this example) instead of IP when available. If prompted for a password, pressing enter should still display the list:

```
$ smbclient -L \\192.168.1.1
```

```
Sharename       Type      Comment
---------       ----      -------
MY_MUSIC        Disk
SHAREDDOCS      Disk
PRINTER$        Disk
PRINTER         Printer
IPC$            IPC       Remote Inter Process Communication

Server               Comment
---------            -------
PUTER

Workgroup            Master
---------            -------
HOMENET               PUTER
```

3.2 Remote control of Windows computer

Samba offers a set of tools for communication with Windows. These can be handy if access to a Windows computer through remote desktop is not an option, as shown by some examples.

Send shutdown command with a comment:

```
$ net rpc shutdown -C "comment" -I IPADDRESS -U USERNAME%PASSWORD
```

A forced shutdown instead can be invoked by changing -C with comment to a single -f. For a restart, only add -r, followed by a -C or -f.

Stop and start services:

```
$ net rpc service stop SERVICENAME -I IPADDRESS -U USERNAME%PASSWORD
```

To see all possible net rpc command:

```
$ net rpc
```

4 Troubleshooting

4.1 Failed to start Samba SMB/CIFS server

Possible solutions:

- Check `smb.conf` on syntactic errors with `testparm(1)` (https://man.archlinux.org/man/testparm.1).
- Set correct permissions for `/var/cache/samba/` and restart `smb.service`:

```
# chmod 0755 /var/cache/samba/msg
```

4.2 Permission issues on SELinux

SELinux not allow samba to access user home directories by default, to solve this, run:

```
# setsebool -P samba_enable_home_dirs 1
```

Similarly, `samba_export_all_ro` and `samba_export_all_rw` make Samba has the ability to read or "read and write" all files.

4.3 Permission issues on AppArmor

If using a share path located outside of a home or usershares directory, whitelist it in `/etc/apparmor.d/local/usr.sbin.smbd`. E.g.:

```
/etc/apparmor.d/local/usr.sbin.smbd
```

```
"/data/" rk,
"/data/**" lrwk,
```

4.4 No dialect specified on mount

The client is using an unsupported SMB/CIFS version that is required by the server.

See #Restrict protocols for better security for more information.

4.5 Unable to overwrite files, permissions errors

Possible solutions:

- Append the mount option `nodfs` to the `/etc/fstab` entry.
- Add `msdfs root = no` to the `[global]` section of the server's `/etc/samba/smb.conf`.

4.6 Windows clients keep asking for password even if Samba shares are created with guest permissions

Set `map to guest` inside the `global` section of `/etc/samba/smb.conf`:

```
map to guest = Bad Password
```

If you are still using Samba < 4.10.10, use `Bad User` instead of `Bad Password`.

4.7 Windows 10 1709 and up connectivity problems - "Windows cannot access" 0x80004005

This error affects some machines running Windows 10 version 1709 and later. It is not related to SMB1 being disabled in this version but to the fact that Microsoft disabled insecure logons for guests on this version for some, but not others.

To fix, open Group Policy Editor (`gpedit.msc`). Navigate to *Computer configuration\administrative templates\network\Lanman Workstation > Enable insecure guest logons* and enable it. Alternatively,change the following value in the registry:

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanWorkstation\Parameters]
"AllowInsecureGuestAuth"=dword:1
```

4.8 Error: Failed to retrieve printer list: NT_STATUS_UNSUCCESSFUL

If you are a home user and using samba purely for file sharing from a server or NAS, you are probably not interested in sharing printers through it. If so, you can prevent this error from occurring by adding the following lines to your `/etc/samba/smb.conf`:

/etc/samba/smb.conf

```
[global]
  load printers = No
  printing = bsd
  printcap name = /dev/null
  disable spoolss = Yes
```

Restart the samba service, `smb.service`, and then check your logs:

```
# cat /var/log/samba/smbd.log
```

and the error should now no longer be appearing.

4.9 Sharing a folder fails

It means that while you are sharing a folder from *Dolphin* (file manager) and everything seems ok at first, after restarting *Dolphin* the share icon is gone from the shared folder, and also some output like this in terminal (*Konsole*) output:

```
'net usershare' returned error 255: net usershare: usershares are currently disabled
```

To fix it, enable usershare as described in **#Enable Usershares**.

4.10 "Browsing" network fails with "Failed to retrieve share list from server"

And you are using a firewall (iptables) because you do not trust your local (school, university, hotel) network. This may be due to the following: When the smbclient is browsing the local network it sends out a broadcast request on udp port 137. The servers on the network then reply to your client but as the source address of this reply is different from the destination address iptables saw when sending the request for the listing out, iptables will not recognize the reply as being "ESTABLISHED" or "RELATED", and hence the packet is dropped. A possible solution is to add:

```
iptables -t raw -A OUTPUT -p udp -m udp --dport 137 -j CT --helper netbios-ns
```

to your iptables setup.

For **Uncomplicated Firewall**, you need to add `nf_conntrack_netbios_ns` to the end of the following line in `/etc/default/ufw`

```
IPT_MODULES="nf_conntrack_ftp nf_nat_ftp nf_conntrack_irc nf_nat_irc"
```

and then run the following commands as root:

```
echo 1 > /proc/sys/net/netfilter/nf_conntrack_helper
ufw allow CIFS
ufw reload
```

To make this change persistent across reboots, add the following line at the end of `/etc/ufw/sysctl.conf`:

```
net.netfilter.nf_conntrack_helper=1
```

4.11 Protocol negotiation failed: NT_STATUS_INVALID_NETWORK_RESPONSE

The client probably does not have access to shares. Make sure clients' IP address is in `hosts allow =` line in `/etc/samba/smb.conf`.

Another problem could be, that the client uses an invalid protocol version. To check this try to connect with the `smbclient` where you specify the maximum protocol version manually:

```
$ smbclient -U <user name> -L //<server name> -m <protocol version: e. g. SMB2> -W <domain name>
```

If the command was successful then create a configuration file:

```
~/.smb/smb.conf
```

```
[global]
  workgroup = <domain name>
  client max protocol = SMB2
```

4.12 Connection to SERVER failed: (Error NT_STATUS_UNSUCCESSFUL)

You are probably passing a wrong server name to `smbclient`. To find out the server name, run `hostnamectl` on the server and look at "Transient hostname" line

4.13 Connection to SERVER failed: (Error NT_STATUS_CONNECTION_REFUSED)

Make sure that the server has started. The shared directories should exist and be accessible.

4.14 Protocol negotiation failed: NT_STATUS_CONNECTION_RESET

Probably the server is configured not to accept protocol SMB1. Add option `client max protocol = SMB2` in `/etc/samba/smb.conf`. Or just pass argument `-m SMB2` to `smbclient`.

4.15 Password Error when correct credentials are given (error 1326)

Samba 4.5 (https://www.samba.org/samba/history/samba-4.5.0.html) has NTLMv1 authentication disabled by default. It is recommend to install the latest available upgrades on clients and deny access for unsupported clients.

If you still need support for very old clients without NTLMv2 support (e.g. Windows XP), it is possible force enable NTLMv1, although this is **not recommend** for security reasons:

```
/etc/samba/smb.conf
```

```
[global]
  lanman auth = yes
  ntlm auth = yes
```

If NTLMv2 clients are unable to authenticate when NTLMv1 has been enabled, create the following file on the client:

```
/home/user/.smb/smb.conf
```

```
[global]
  sec = ntlmv2
  client ntlmv2 auth = yes
```

This change also affects samba shares mounted with **mount.cifs**. If after upgrade to Samba 4.5 your mount fails, add the **sec=ntlmssp** option to your mount command, e.g.

```
mount.cifs //server/share /mnt/point -o sec=ntlmssp,...
```

See the `mount.cifs(8)` (https://man.archlinux.org/man/mount.cifs.8) man page: **ntlmssp** - Use NTLMv2 password hashing encapsulated in Raw NTLMSSP message. The default in mainline kernel versions prior to v3.8 was **sec=ntlm**. In v3.8, the default was changed to **sec=ntlmssp**.

4.16 Mapping reserved Windows characters

Starting with kernel 3.18, the cifs module uses the "mapposix" option by default (https://git.kernel.org/cgit/linux/kernel/git/torvalds/linux.git/commit/?id=2baa2682531ff02928e2d3904800696d9e7193db). When mounting a share using unix extensions and a default Samba configuration, files and directories containing one of the seven reserved Windows characters `: \ * < > ?` are listed but cannot be accessed.

Possible solutions are:

- Use the undocumented `nomapposix` mount option for cifs

```
# mount.cifs //server/share /mnt/point -o nomapposix
```

- Configure Samba to remap `mapposix` ("SFM", Services for Mac) style characters to the correct native ones using fruit (https://www.mankier.com/8/vfs_fruit)

```
/etc/samba/smb.conf
```

```
[global]
  vfs objects = catia fruit
  fruit:encoding = native
```

- Manually remap forbidden characters using catia (https://www.mankier.com/8/vfs_catia)

```
/etc/samba/smb.conf
```

```
[global]
  vfs objects = catia
  catia:mappings = 0x22:0xf022, 0x2a:0xf02a, 0x2f:0xf02f, 0x3a:0xf03a, 0x3c:0xf03c, 0x3e:0xf03e, 0x3f:0xf
03f, 0x5c:0xf05c, 0x7c:0xf07c, 0x20:0xf020
```

The latter approach (using catia or fruit) has the drawback of filtering files with unprintable characters.

4.17 Folder shared inside graphical environment is not available to guests

This section presupposes:

1. Usershares are configured following previous section
2. A shared folder has been created as a non-root user from GUI
3. Guests access has been set to shared folder during creation
4. Samba service has been restarted at least once since last `/etc/samba/smb.conf` file modification

For clarification purpose only, in the following sub-sections is assumed:

- Shared folder is located inside user home directory path (`/home/yourUser/Shared`)
- Shared folder name is *MySharedFiles*
- Guest access is read-only.
- Windows users will access shared folder content without login prompt

4.17.1 Verify correct samba configuration

Run the following command from a terminal to test configuration file correctness:

```
$ testparm
```

4.17.2 Verify correct shared folder creation

Run the following commands from a terminal:

```
$ cd /var/lib/samba/usershares
$ ls
```

If everything is fine, you will notice a file named `mysharedfiles`

Read the file contents using the following command:

```
$ cat mysharedfiles
```

The terminal output should display something like this:

```
/var/lib/samba/usershares/mysharedfiles
```

```
path=/home/yourUser/Shared
comment=
usershare_acl=S-1-1-0:r
guest_ok=y
sharename=MySharedFiles
```

4.17.3 Verify folder access by guest

Run the following command from a terminal. If prompted for a password, just press Enter:

```
$ smbclient -L localhost
```

If everything is fine, MySharedFiles should be displayed under `Sharename` column

Run the following command in order to access the shared folder as guest (anonymous login)

```
$ smbclient -N //localhost/MySharedFiles
```

If everything is fine samba client prompt will be displayed:

```
smb: \>
```

From samba prompt verify guest can list directory contents:

```
smb: \> ls
```

If the `NTFS_STATUS_ACCESS_DENIED` error is displayed, the issue is likely to be with Unix directory permissions. Ensure that your samba user has access to the folder and all parent folders. You can test this by sudoing to the user and attempting to list the mount directory, and all of its parents.

4.18 Mount error: Host is down

This error might be seen when mounting shares of Synology NAS servers. Use the mount option `vers=1.0` to solve it.

> **Note:** SMB version 1 is known to have security vulnerabilities and was used in successful ransomware attacks.

4.19 Software caused connection abort

File managers that utilizes **gvfs-smb (https://archlinux.org/packages/?name=gvfs-smb)** can show the error `Software caused connection abort` when writing a file to a share/server. This may be due to the server running SMB/CIFS version 1, which many routers use for USB drive sharing (e.g. Belkin routers). To write to these shares specify the CIFS version with the option `vers=1.0`. E.g.:

/etc/fstab

```
//SERVER/sharename /mnt/mountpoint cifs _netdev,guest,file_mode=0777,dir_mode=0777,vers=1.0 0 0
```

This can also happen after updating Samba to version 4.11, which deactivates SMB1 as default, and accessing any Samba share. You can reenable it by adding

/etc/samba/smb.conf

```
[global]
client min protocol = CORE
```

4.20 Connection problem (due to authentification error)

Be sure that you do not leave any space characters before your username in Samba client configuration file as follows:

~/.samba

```
username= user
password=pass
```

The correct format is:

~/.samba

```
username=user
password=pass
```

4.21 Windows 1709 or up does not discover the samba server in Network view

With Windows 10 version 1511, support for SMBv1 and thus NetBIOS device discovery was disabled by default. Depending on the actual edition, later versions of Windows starting from version 1709 ("Fall Creators Update") do not allow the installation of the SMBv1 client anymore. This causes hosts running Samba not to be listed in the Explorer's "Network (Neighborhood)" views. While there is no connectivity problem and Samba will still run fine, users might want to have their Samba hosts to be listed by Windows automatically. **wsdd (https://aur.archlinux.org/packages/wsdd/)**[AUR] implements a Web Service Discovery host daemon. This enables (Samba) hosts, like your local NAS device, to be found by Web Service Discovery Clients like Windows. The default settings should work for most installations, all you need to do is start enable `wsdd.service`.

If the default configuration (advertise itself as the machine hostname in group "WORKGROUP") should be all you need in most cases. If you need, you can change configuration options by passing additional arguments to wsdd by adding them in `/etc/conf.d/wsdd` (see the manual page for wsdd for details).

wsdd2 (https://aur.archlinux.org/packages/wsdd2/)[AUR] does the same thing, but is written in C instead of Python. By default, it will look for the `netbios name` and `workgroup` values in `smb.conf`.

4.22 IOS Files can no longer copy-to Samba share on Arch Linux beginning with IOS 14.5

Beginning with IOS 14.5 attempting to transfer from a device running IOS using the "Files" app to a samba share on Arch Linux will result in the error:

```
The operation couldn't be completed
Operation canceled
```

To correct this problem, add add the following to the global section of your `smb.conf` and **restart** `smb.service`. Comment optional:

```
## addition for IOS Files transfer-to server
vfs object = fruit streams_xattr
```

See **https://apple.stackexchange.com/q/424681** Apple.Stackexchange.com - "The operation couldn't be completed"/"Operation canceled" error message when saving to a Samba share via Files app.

4.23 Slow initial connections from certain clients without other performance problems

Some SMB clients, such as Solid Explorer for Android, take significantly longer to connect to Samba if they fail to resolve the NetBIOS name. Enabling `nmb.service` will greatly speed up initial connections if this is the case. Since this is a bug in the client software, please report such cases to the authors of conflicting software.

5 See also

- **Official website (https://www.samba.org/)**
- **Samba: An Introduction (https://www.samba.org/samba/docs/SambaIntro.html)**
- **Samba 3.2.x HOWTO and Reference Guide (https://www.samba.org/samba/docs/Samba-HOWTO-Collection.pdf)** (outdated but still most extensive documentation)
- **Wikipedia**
- **Gentoo:Samba/Guide**
- **Debian:Samba/ServerSimple**
- **KSMBD (https://docs.kernel.org/filesystems/smb/ksmbd.html)** - A linux kernel server which implements SMB3 protocol in kernel space for sharing files over network.