



How to install and configure samba on RHEL 8 / CentOS 8

👤 Egidio Docile 📁 Redhat / CentOS / AlmaLinux

🕒 22 September 2019

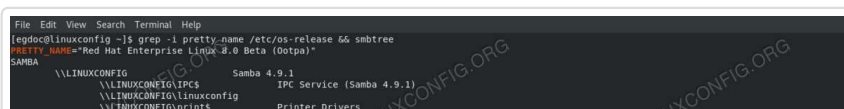
Contents

1. Software Requirements and Conventions Used (<https://linuxconfig.org/install-samba-on-redhat-8#h1-software-requirements-and-conventions-used>)
2. Introducing Samba (<https://linuxconfig.org/install-samba-on-redhat-8#h2-introducing-samba>)
3. Installation (<https://linuxconfig.org/install-samba-on-redhat-8#h3-installation>)
4. Configuring the firewall (<https://linuxconfig.org/install-samba-on-redhat-8#h4-configuring-the-firewall>)
5. Configuring a shared directory accessible by guests (<https://linuxconfig.org/install-samba-on-redhat-8#h5-configuring-a-shared-directory-accessible-by-guests>)
6. Configure a share accessible only by registered users (<https://linuxconfig.org/install-samba-on-redhat-8#h6-configure-a-share-accessible-only-by-registered-users>)
7. Setup SELinux for samba (<https://linuxconfig.org/install-samba-on-redhat-8#h7-setup-selinux-for-samba>)
8. Conclusions (<https://linuxconfig.org/install-samba-on-redhat-8#h8-conclusions>)

Samba provides server and client software to allow file sharing between Linux and Windows machines. Installing and configuring it on [RHEL 8 \(/how-to-install-rhel-8\)](#) / CentOS 8, is quite easy. Keep reading to learn how to share a directory with samba, and how to apply the appropriate SELinux context to it.

In this tutorial you will learn:

- How to install samba on RHEL8
- How to enable and start the smb and nmb daemons
- How to create a samba share
- How to setup the firewall to allow samba share to be accessed
- How to setup the correct SELinux context for samba to work correctly



([/images/smbtree-rhel8.png](#))

Samba shares on RHEL 8 / CentOS 8

Software Requirements and Conventions Used

Software Requirements and Linux Command Line Conventions

Category	Requirements, Conventions or Software Version Used
System	Red Hat Enterprise Linux 8
Software	Samba, coreutils and policycoreutils-python-utils packages
Other	Permission to run command with root privileges.
Conventions	# - requires given linux commands (/linux-commands) to be executed with root privileges either directly as a root user or by use of <code>sudo</code> command \$ - requires given linux commands (/linux-commands) to be executed as a regular non-privileged user

Introducing Samba

Samba, as stated in the homepage of the project, is an open source software, released under the `GPL` license, which allow us to share files and print services using the `SMB/CIFS` protocol.

The project provides both server and client software to allow interoperation with Windows machines, representing the ideal solution in mixed environments. In this tutorial we will see how to install Samba on Red Hat Enterprise Linux 8, how to setup a samba share, how the setup the firewall to allow access to shared resources, and how to apply the appropriate SELinux context.

SUBSCRIBE TO NEWSLETTER

Subscribe to our NEWSLETTER (<https://bit.ly/2X5D30q>) and receive latest Linux news, jobs, career advice and tutorials.

Installation

The first thing we have to do is to install samba on our machine. The package and the needed libraries are available in the official RHEL 8 / CentOS 8 repositories, therefore we can install them just by using yum or dnf. In this version of RHEL/CentOS, the first command it's just a "link" to the second:

```
$ sudo dnf install samba samba-client
```

The samba-client package is not strictly needed, but the utilities provided by it can be useful. Once the packages are installed, we have to start and enable the `smb` and the `nmb` daemons at boot. The first is the daemon which takes care of performing the actual transfers and the sharing operations, while the second performs the `NetBIOS` name resolutions, allowing the resources to

appear when browsing the network on Windows. We can now enable and start both systemd services (/redhat-8-start-service-on-boot) with just one command:

```
$ sudo systemctl enable --now {smb,nmb}
```

Configuring the firewall

The next step is the firewall configuration. We must open the appropriate ports, so that the samba-shared resources can be accessible from other machines. The default firewall management software on RHEL 8 / CentOS 8, is firewalld (<https://linuxconfig.org/introduction-to-firewalld-and-firewall-cmd-command-on-linux>).

Luckily for us, there is no need to open ports manually: all we have to do is to add the "samba" service to our zone. A "service" is just an abstraction which let us allow traffic through all the ports needed by a service, by referencing the service name, instead of having to setup (and remember) each port used by it. In case we want to gather information about a "service", we can run:

```
$ sudo firewall-cmd --info-service samba
samba
ports: 137/udp 138/udp 139/tcp 445/tcp
protocols:
source-ports:
modules: netbios-ns
destination:
```

From the output of the command we see that the service will allow traffic through ports 173/udp, 138/udp, 139/tcp and 445/tcp. To permanently add the service to the default zone, we can run:

```
$ sudo firewall-cmd --permanent --add-service=samba
```

When running the command we used the `--permanent` switch in order to make our change persistent. We also assumed the `default` zone to be in use. If we wanted to

specify another zone for the action to be applied on, we would have used the `--zone` option, and provided the zone name as argument (e.g. `--zone=external`). Since our change is set to be permanent, in order for it to become effective, we must reload the firewall configuration:

```
$ sudo firewall-cmd --reload
```

We can verify that the "samba" service is now part of our zone, by running:

```
$ sudo firewall-cmd --list-services  
cockpit dhcpv6-client http samba ssh
```

Again, if no zone is specified, the command is applied to the default zone.

Configuring a shared directory accessible by guests

Let's say we want to share a directory via samba, and we want to let free access to this directory to guest users, without them having to provide a password. To obtain the desired result, we must make some changes to the `/etc/samba/smb.conf` file, and add a "stanza" for our share. Open the file with your favorite editor, and in the `[global]` section, add the highlighted text:

```
[global]  
    workgroup = SAMBA  
    security = user  
  
    passdb backend = tdbsam  
  
    printing = cups  
    printcap name = cups  
    load printers = yes  
    cups options = raw  
    map to guest = bad user
```

The `map to guest = bad user` instruction, will map login attempts with bad usernames to the default guest user, which, by default is `nobody`. This is needed to allow anonymous access without having to provide a password.

After this change, we must append a new stanza dedicated to our share at the end of the file. We will name the shared resource "linuxconfig":

```
[linuxconfig]
    path = /mnt/shared
    guest only = yes
```

With the above setup we declared that we want to share the content of the `/mnt/shared` directory, without the need of user authentication. This setup is obviously risky, and is here reported only as an example: in a real world scenario, you may want to at least deny write access to guests (you can do it by adding the `writeable = no` instruction). To make the changes effective, we must restart the daemons:

```
$ sudo systemctl restart {smb,nmb}
```

Configure a share accessible only by registered users

To protect access to a resource with a login prompt, when samba is running as a standalone server, we must add an existing user to the samba database. The credentials of this user will be necessary to access the shared directory. For security reasons, it's a good practice to create a dedicated user for the task, omitting the creation of his home directory, and assigning him a fake shell:

```
sudo adduser -M sambauser -s /sbin/nologin
```

The `-M` option passed to the command is the short form for `--no-create-home`, which is quite self-explanatory; the `-s` option, instead, let us specify a shell, in this case an invalid one on purpose: `/sbin/nologin`. At this point the user doesn't even need to have a password set on our system.

Once the user is created, we need to add it to the samba database: we can perform the operation by using the `smbpasswd` command:

```
$ sudo smbpasswd -a sambauser
New SMB password:
Retype new SMB password:
Added user sambauser.
```

After running the command, we are prompted to assign a password for the user, and also to confirm it: this password will only be valid in the samba context, doesn't have anything to do with the user account on our system. To restrict the share we previously created, we have to make a little change in the dedicated section:

```
[linuxconfig]
    path = /mnt/shared
    guest ok = no
```

We can verify that our setup is valid, by using the `testparm` command:

SUBSCRIBE TO NEWSLETTER

*Subscribe to our NEWSLETTER (<https://bit.ly/2X5D30q>).
and receive latest Linux news, jobs, career advice and
tutorials.*

```
$ testparm
Load smb config files from
/etc/samba/smb.conf
rlimit_max: increasing rlimit_max (1024) to
minimum Windows limit (16384)
Processing section "[homes]"
Processing section "[printers]"
Processing section "[print$]"
Processing section "[linuxconfig]"
Loaded services file OK.
Server role: ROLE_STANDALONE
```

Again, we must restart the `smb` and `nmb` daemons for our changes to be effective. The shared directory will now be accessible only after providing the correct credentials.

Setup SELinux for samba

SELinux is often saw like a complication, and disabled right away. This is not necessary: we only must learn how to configure it, and take advantage of the enhancement in security it provides. In order for our samba share to work when SELinux is in "enforcing" mode, we must assign the appropriate context to our shared directory and files:

```
$ sudo chcon -R -t samba_share_t /mnt/shared
```

In the example above, we used the `chcon` command with the `-t` option, to change the `TYPE` section of the SELinux context to `samba_share_t`. We also used the `-R` switch to make the command recursive. This change will survive a reboot, but not a relabeling of the system if a default policy exists for our directory and files, because in such case the default setup would be re-applied.

If we want our change to survive a relabeling event, we must add our rule to the policy. We can do this by using the `semanage` command:

```
$ sudo semanage fcontext -a -t samba_share_t "/mnt/
```


Since we used the `(/.*)?` regex, the rule will be applied to all the content of the "shared" directory, and to the directory itself. We can verify that our rule has been added to the policy by listing the labels in use on our system:

```
$ sudo semanage fcontext -l | grep
/mnt/shared
/mnt/shared(/.*)?
all files
system_u:object_r:samba_share_t:s0
```

We should now have a working samba setup. To further tweak samba configuration we may need to also manipulate SELinux booleans. For example, when adding existent users to samba, to let their home directories be shared, we must enable the dedicated SELinux boolean. To list all booleans related to samba, we can run:

```
sudo semanage boolean -l|grep samba
samba_create_home_dirs      (off ,  off)
Allow samba to create home dirs
samba_domain_controller     (off ,  off)
Allow samba to domain controller
samba_enable_home_dirs      (off ,  off)
Allow samba to enable home dirs
samba_export_all_ro         (off ,  off)
Allow samba to export all ro
samba_export_all_rw         (off ,  off)
Allow samba to export all rw
samba_load_libgfapi         (off ,  off)
Allow samba to load libgfapi
samba_portmapper            (off ,  off)
Allow samba to portmapper
samba_run_unconfined        (off ,  off)
Allow samba to run unconfined
samba_share_fusefs          (off ,  off)
Allow samba to share fusefs
samba_share_nfs             (off ,  off)
Allow samba to share nfs
sanlock_use_samba           (off ,  off)
Allow sanlock to use samba
tmpreaper_use_samba         (off ,  off)
Allow tmpreaper to use samba
use_samba_home_dirs         (off ,  off)
Allow use to samba home dirs
virt_use_samba              (off ,  off)
Allow virt to use samba
```

In the output above, the second column indicates the current value of the boolean, while the third the default one (both are off, in this case). The boolean we want to activate, to enable the sharing of home directories, is `samba_enable_home_dirs`. We can perform the operation by using the `setsebool` command:

```
$ sudo setsebool samba_enable_home_dirs=1
```

Conclusions

In this tutorial we saw how to install samba on a RHEL 8 / CentOS 8 system. We also saw how to share a directory, allowing access to guests or restricting it to authenticated users. We also saw how to configure the firewall in order for the share to be accessible from other machines on the network.

Finally, we saw how to perform the needed changes in order to have a working samba setup with SELinux in "enforcing" mode. If you are interested SELinux, you can also read our [article on the subject](https://linuxconfig.org/introduction-to-selinux-concepts-and-management) (<https://linuxconfig.org/introduction-to-selinux-concepts-and-management>).

◀ [Prev \(/how-to-install-ruby-on-redhat-8\)](/how-to-install-ruby-on-redhat-8)

[Next](/redhat-8-increase-swap-size) ▶ [\(/redhat-8-increase-swap-size\)](/redhat-8-increase-swap-size)

FIND LATEST LINUX JOBS
(<https://www.linuxcareers.com/>) on
LinuxCareers.com

Submit your RESUME
(<https://www.linuxcareers.com/candidate/register/>),
create a JOB ALERT
(<https://www.linuxcareers.com/jobs/alert/>).

***LINUX CAREER
NEWSLETTER***
Subscribe to
NEWSLETTER

***DO YOU NEED
ADDITIONAL HELP?***
Get extra help by visiting
our LINUX FORUM

(<https://bit.ly/2X5D30q>).
and receive latest news,
jobs, career advice and
tutorials.

(<https://forum.linuxconfig.org>).
or simply use comments
below.

MORE ON LINUXCONFIG.ORG:

- [How to start service on boot with RHEL 8 / CentOS 8 Linux \(/redhat-8-start-service-on-boot\)](#).
 - [How to install asterisk on RHEL 8 / CentOS 8 Linux \(/how-to-install-asterisk-on-redhat-8\)](#).
 - [How to install and setup an example service with xinetd on RHEL 8 / CentOS 8 Linux \(/how-to-install-xinetd-service-in-redhat-8\)](#).
 - [How to install vsftpd on RHEL 8 / CentOS 8 \(/how-to-install-vsftpd-on-redhat-8\)](#).
-
-

Comments and Discussions

LINUXCONFIG.ORG
F O R U M

1 reply



hortimech

Mar '19

Hi, you may want to read 'man smb.conf', your example shares are not going to do what you think



Continue Discussion

LINUXCONFIG

NEWSLETTER

Subscribe to [Linux Career Newsletter](https://bit.ly/2X5D30q)
(<https://bit.ly/2X5D30q>) to receive latest news, jobs,
career advice and featured configuration tutorials.

WRITE FOR US

LinuxConfig is looking for a technical writer(s) geared towards GNU/Linux and FLOSS technologies. Your articles will feature various GNU/Linux configuration tutorials and FLOSS technologies used in combination with GNU/Linux operating system.

When writing your articles you will be expected to be able to keep up with a technological advancement regarding the above mentioned technical area of expertise. You will work independently and be able to produce at minimum 2 technical articles a month.

APPLY NOW

(<https://www.linuxcareers.com/jobs/floss-technical-writer-new-york-city-new-york/1-1/>).

CONTACT

✉ [web \(at \) linuxconfig \(dot \) org](mailto:web@linuxconfig.org)

FEATURED LINUX TUTORIALS

- [How To enable the EPEL Repository on RHEL 8 / CentOS 8 Linux \(/redhat-8-epel-install-guide\)](#)
- [Bash scripting Tutorial \(/bash-scripting-tutorial\)](#)
- [How to install VMware Tools on RHEL 8 / CentOS 8 \(/how-to-install-vmware-tools-on-rhel-8-centos-8\)](#)
- [Howto mount USB drive in Linux \(/howto-mount-usb-drive-in-linux\)](#)
- [How to install the NVIDIA drivers on Ubuntu 18.04 Bionic Beaver Linux \(/how-to-install-the-nvidia-drivers-on-ubuntu-18-04-bionic-beaver-linux\)](#)

[on-ubuntu-18-04-bionic-beaver-linux](#)).

- [How to update Kali Linux \(/how-to-update-kali-linux\)](#)
- [Ubuntu 20.04 Download \(/ubuntu-20-04-download\)](#)
- [How To Upgrade Ubuntu To 20.04 LTS Focal Fossa \(/how-to-upgrade-ubuntu-to-20-04-lts-focal-fossa\)](#)
- [How to install node.js on RHEL 8 / CentOS 8 Linux \(/how-to-install-node-js-on-redhat-8-linux\)](#)
- [How to check CentOS version \(/how-to-check-centos-version\)](#)
- [How to Parse Data From JSON Into Python \(/how-to-parse-data-from-json-into-python\)](#)
- [Check what Debian version you are running on your Linux system \(/check-what-debian-version-you-are-running-on-your-linux-system\)](#)
- [Bash Scripting Tutorial for Beginners \(/bash-scripting-tutorial-for-beginners\)](#)
- [Ubuntu 20.04 Guide \(/ubuntu-20-04-guide\)](#)
- [How to stop/start firewall on RHEL 8 / CentOS 8 \(/redhat-8-stop-start-firewall\)](#)
- [Install gnome on RHEL 8 / CentOS 8 \(/install-gnome-on-redhat-8\)](#)
- [Linux Download \(/linux-download\)](#)
- [How To Upgrade from Ubuntu 18.04 and 19.10 To Ubuntu 20.04 LTS Focal Fossa \(/how-to-upgrade-ubuntu-to-20-04-lts-focal-fossa\)](#)
- [Enable SSH root login on Debian Linux Server \(/enable-ssh-root-login-on-debian-linux-server\)](#)

LATEST ARTICLES

- [Basic Linux Kernel module administration commands \(/basic-linux-module-administration-commands\)](#)
- [Configure Apache VirtualHost on Fedora \(/configure-apache-virtualhost-on-fedora\)](#)
- [Backup permissions on Linux \(/backup-permissions-in-linux\)](#)
- [How to configure shared VirtualBox folder on Linux \(/share-folders-virtualbox-quick-linux-setup\)](#)
- [Turn Off directory browsing on Apache \(/turn-off-directory-browsing-on-apache\)](#)
- [Linux Backup Restore Destroy and Install MBR - Master Boot Record \(/linux backup restore destroy and install mbr\)](#)

- [Master Boot Record \(/linux-backup-restore-destroy-and-install-mbr-master-boot-record\)](#)
- [How to Create an Integrated Application Launcher For an Appimage file in Ubuntu \(/how-to-create-an-integrated-application-launcher-for-an-appimage-file-in-ubuntu\)](#)
- [Data recovery of deleted files from the FAT filesystem \(/data-recovery-of-deleted-files-from-the-fat-filesystem\)](#)
- [USB stick encryption using Linux \(/usb-stick-encryption-using-linux\)](#)
- [Getting to know the hardware of your Linux box \(/getting-know-a-hardware-on-your-linux-box\)](#)
- [How to retrieve and change partition's UUID Universally Unique Identifier on linux \(/how-to-retrieve-and-change-partitions-universally-unique-identifier-uuid-on-linux\)](#)
- [Introductory tutorial to Git on Linux \(/introductory-tutorial-to-git-on-linux\)](#)
- [C++ function to calculate Fibonacci number sequence \(/c-function-to-calculate-fibonacci-number\)](#)
- [How to get CUDA Cores count on Linux \(/how-to-get-cuda-cores-count-on-linux\)](#)
- [How to install NVIDIA CUDA on Debian 10 \(buster\) \(/how-to-install-nvidia-cuda-on-debian-10-buster\)](#)

© 2007 - 2021 [LinuxConfig.org](#)

[Privacy \(/privacy\)](#)

[Twitter \(https://twitter.com/linuxconfig\)](#)