BLOG ABOUT ALL POSTS LPIC-2 LPIC-3 RHCE RHCA

← Setting up a Kerberised NFS Server on RHEL 7

Setting up an OpenSSH Server with SELinux on RHEL 7 →

Setting up a Samba Server with SELinux on RHEL 7

Posted on 07/06/2016 by Tomas

We are going to set up a Samba server and configure a network share suitable for group collaboration.

The Lab

We have two RHEL 7.0 servers available in our lab:

srv1.rhce.local (10.8.8.71) – will be configured as a Samba server
srv2.rhce.local (10.8.8.72) – will be configured as a Samba client

Both servers have SELinux set to enforcing mode.

Samba Server

All commands in this section are run on the server **srv1**.

The **samba** package version used in the article is **4.1.1**.

Packages, Services and Firewall

The **samba-client** package contains the **smbpasswd** command.

```
# yum install -y samba samba-client
# systemctl enable smb nmb
# firewall-cmd --permanent --add-service=samba
# firewall-cmd --reload
```

Prepare Shared Directories

We are going to create two different shares as explained below:

`/srv/samba_pub` – a public Samba share with r/w for all,
`/srv/samba_group` – a Samba share for group collaboration.

Create directories:

```
# mkdir /srv/{samba_pub,samba_group}
```

Change permissions for the public Samba share:

```
# chmod 0777 /srv/samba_pub
```

Configure collaboration for the group share:

```
# groupadd devops
# chgrp devops /srv/samba_group
# chmod 2775 /srv/samba_group
```

We want to give read-only privileges for all users who are not members of the **devops** group.

When a user authenticates to the Samba server, a Samba user account is used, but the Samba user account is mapped to a Linux user account, and that user account needs access permissions.

Note that users with no write permissions on the Linux file system will not have write permissions on a share. If a share is set to writable, all users with write permissions on the Linux file system have write access to the share.

Create a couple of Samba users, **dev1** and **dev2**, where **dev1** is a member of the **devops** Linux group:

```
# useradd -s /sbin/nologin -G devops dev1
# useradd -s /sbin/nologin dev2
# smbpasswd -a dev1
# smbpasswd -a dev2
```

Check Samba users' database:

```
# pdbedit -L
```

Apply SELinux Context

Let us check the default SELinux context:

```
# ls -dZ /srv/samba_*
drwxrwsr-x. root devops unconfined_u:object_r:var_t:s0   /srv/samba_g
drwxrwxrwx. root root   unconfined_u:object_r:var_t:s0   /srv/samba_p
```

Apply the **samba_share_t** context type to the group share:

```
# semanage fcontext -a -t samba_share_t "/srv/samba_group(/.*)?"
```

Note that if the shared directory will only be accessed through Samba, then it should be labeled **samba_share_t**, which gives Samba read and write access.

Samba can also serve files labeled with the SELinux types **public_content_t** (readonly) and **public_content_rw_t** (read-write). For the public share, we are going to use the public_content_rw_t type.

Note that files labeled with the **public_content_t** type allow them to be read by FTP, Apache, Samba and rsync. Files labeled with the **public_content_rw_t** type require booleans to be set before services can write to files labeled with the public_content_rw_t type.

The boolean that’s require in Samba’s case is **smbd_anon_write**.

```
# setsebool -P smbd_anon_write=1
# semanage fcontext -a -t public_content_rw_t "/srv/samba_pub(/.*)?"
```

Don’t forget to restore SELinux context:

```
# restorecon -Rv /srv/samba_*
```

Other SELinux Booleans Worth Mentioning

If we wanted to share any standard directory read-only, we would set the boolean **samba_export_all_ro**:

```
# setsebool -P samba_export_all_ro=1
```

The boolean above would allow Samba to read every file on the system. It is off by default.

Similarly, if we wanted to share all files and directories read/write via Samba, we would set the **samba_export_all_rw**:

```
# setsebool -P samba_export_all_rw=1
```

This boolean would allow Samba to read and write every file on the system. It’s a bad idea in general, as compromised Samba server would become extremelly dangerous. It is off by default.

If wanted to allow samba to create new home directories, we would need to turn on the **samba_create_home_dirs** boolean:

```
# setsebool -P samba_create_home_dirs=1
```

By default SELinux policy turns off SELinux sharing of home directories (the **[homes]** section defines a special file share which is enabled by default). If we were to set up a VM as a Samba server and wanted to share users home directories, we would need to set the **samba_enable_home_dirs** boolean:

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

The above needs to be enabled for **[homes]** to work.

Note that Samba SELinux policy will not allow any confined applications to access remote samba shares mounted on the server. If we want to use a remote Samba server for the home directories on the server, we must set the **use_samba_home_dirs** boolean:

```
# setsebool -P use_samba_home_dirs=1
```

The above allows remote Samba file shares to be mounted and used as local Linux home directories.

Another important boolean is **samba_share_nfs**. By default, SELinux prevents Samba daemons from reading and writing NFS shares. If we were using Samba to share NFS file systems, we would need to turn the **samba_share_nfs** boolean on:

```
# setsebool -P samba_share_nfs=1
```

Failure to do so will cause a *permission denied* mount error, but nothing will be logged in to the log file `/var/log/audit/audit.log`, what makes it hard to troubleshoot.

Configure Samba

Open the file `/etc/samba/smb.conf` for editing and add the following:

```
[global]
;       Most Windows systems default to WORKGROUP
        workgroup = MYGROUP
        server string = Samba Server Version %v
;       netbios name = MYSERVER

        interfaces = lo 10.8.8.0/24
        hosts allow = 127. 10.8.8.
        hostname lookups = yes

        log file = /var/log/samba/log.%m
        max log size = 50

        security = user
        passdb backend = tdbsam
        map to guest = bad user
        guest account = nobody
        load printers = no

[public]
```

```
        comment = Public Share
        path = /srv/samba_pub
;       public = yes
        writable = yes
        browseable = yes
        printable = no
        guest ok = yes

[group]
        comment = Group Share
        path = /srv/samba_group
        writable = no
        browseable = yes
        printable = no
        guest ok = no
        write list = @devops
        read list = dev2
        valid users = @devops, dev2
```

Note the **hosts allow** parameter, if it's specified in the **[global]** section, then it will apply to all shares regardless of whether each share has a different setting. Hosts can be specified by a host name or by a source IP address. Host names are checked by reverse-resolving the IP address of the incoming connection attempt. The default **name resolve order** for name resolution is to use the LMHOSTS file, followed by standard Unix name resolution methods (some combination of /etc/hosts, DNS and NIS), then query a WINS server and finally use broadcasting to determine the address of a NetBIOS name. Be advised that **hostname lookups** must to be enabled for reverse-resolving to work.

If a share is set as read-only (**read only = yes**, or inverted synonym **writable = no**), which is the default, users that are listed in the **write list** still have read-write access to the share. So for the group share, all users who are members of the **devops** group have read-write access. However, user **dev2** can mount the share, but has read-only access.

On the other hand, if a share is writeable (**read only = no**), users in the **read list** will not be given write access, no matter what the read only option is set to.

Note that a printable service (**printable = yes**) will always allow writing to the directory (user privileges permitting), but only via spooling operations. The default is **printable = no**.

The **valid users** parameter specifies a list of users who are allowed to access the share. Users not on the list are not allowed to access the share. Note that leaving the list blank, which is the default, allows all users to access the share.

Please note that **guest ok** is a synonym for **public**.

To summarise, these are the defaults, and can be omitted, unless a change is required:

```
hosts allow = # none (all hosts permitted access)
```

```
read only = yes
writable = no
printable = no
browseable = yes
valid users = # no valid users list (anyone can login)
guest ok = no
```

Let us test the configuration:

```
# testparm -s
Load smb config files from /etc/samba/smb.conf
rlimit_max: increasing rlimit_max (1024) to minimum Windows limit (1
Processing section "[public]"
Processing section "[group]"
Loaded services file OK.
Server role: ROLE_STANDALONE
[global]
        workgroup = MYGROUP
        server string = Samba Server Version %v
        interfaces = lo, 10.8.8.0/24
        map to guest = Bad User
        log file = /var/log/samba/log.%m
        max log size = 50
        load printers = No
        idmap config * : backend = tdb
        hosts allow = 127., 10.8.8.

[public]
        comment = Public Share
        path = /srv/samba_pub
        read only = No
        guest ok = Yes

[group]
        comment = Group Share
        path = /srv/samba_group
        valid users = @devops, dev2
        read list = dev2
        write list = @devops
```

Start the services:

```
# systemctl start smb nmb
```

Test access locally:

```
# smbclient //localhost/public -U guest%
```

Samba Client

All commands in this section are run on the server **srv2**.

Install Packages

```
# yum install -y samba-client cifs-utils
```

Mount Samba Shares

Create mountpoints:

```
# mkdir /mnt/{samba_pub,samba_group}
```

Mount Samba shares:

```
# mount -o username=dev1 //srv1.rhce.local/group /mnt/samba_group
# mount -o username=guest,password= //srv1.rhce.local/public /mnt/sa
```

Add the following to the file /etc/fstab to mount on boot:

```
//srv1.rhce.local/group   /mnt/samba_group cifs username=dev1,passw
//srv1.rhce.local/public  /mnt/samba_pub   cifs username=guest,passw
```

We can also use the **credentials** parameter to pass the user details that are stored in a file, for example:

```
//srv1.rhce.local/group   /mnt/samba_group cifs credentials=/root/c
```

Where the content of the /root/creds.txt file is this:

```
username=dev1
password=pass
```

The file should be read by the root user only.

Sander van Vugt recommends that all remote file systems that need to be mounted through /etc/fstab include the **_netdev** and the **x-systemd.automount** mount options.

The **_netdev** mount option ensures that the mount is delayed until the network is fully available. The **x-systemd.automount** option ensures optimal integration with systemd and will ensure that the mount is made a lot faster.

If we now try to write to the group share, it should work as the user **dev1** is a member of the **devops** group. However, if we remount the group share using the user's **dev2** credentials, we'll get read-only access and won't be able to create any files.

On the Samba server **srv1**, we can check current connections:

```
# smbstatus
Samba version 4.1.1
PID     Username      Group         Machine
-------------------------------------------------------------------
2790      dev1          dev1          10.8.8.72    (ipv4:10.8.8.72:
2790      nobody        nobody        10.8.8.72    (ipv4:10.8.8.72:

Service      pid     machine       Connected at
```

```
-------------------------------------------------------
IPC$         2790   10.8.8.72     Tue Jun  7 19:44:27 2016
group        2790   10.8.8.72     Tue Jun  7 19:44:27 2016
public       2790   10.8.8.72     Tue Jun  7 19:40:53 2016
IPC$         2790   10.8.8.72     Tue Jun  7 19:40:53 2016
```

SMB/CIFS resources can also be accessed with **smbclient**:

```
# smbclient -L srv1.rhce.local -N
Domain=[MYGROUP] OS=[Unix] Server=[Samba 4.1.1]

        Sharename       Type      Comment
        ---------       ----      -------
        public          Disk      Public Share
        group           Disk      Group Share
        IPC$            IPC       IPC Service (Samba Server Version
Domain=[MYGROUP] OS=[Unix] Server=[Samba 4.1.1]

        Server               Comment
        ---------            -------

        Workgroup            Master
        ---------            -------
```

Multiuser Samba Mount

In RHEL 7 we can use the **multiuser** mount option to create a multiuser Samba mount.

We mount the share with a user who has minimal permissions on the share. Regular users can then add their own SMB username and password in their current session to elevate their permissions to their own permission level.

Mount the share as a multiuser mount:

```
# mount -o username=dev2,multiuser,sec=ntlmssp //server1.rhce.local/
```

Note that by default the protocol that's used to authenticate users is NTLM v2 password hashing encapsulated in raw NTLMSSP messages (**sec=ntlmssp**). It's for compatibility with Microsoft Windows.

We should get the permission denied error trying to write to the share as the user **dev2** doesn't have write privileges:

```
# touch /mnt/samba_group/test
touch: cannot touch '/mnt/samba_group/test': Permission denied
```

On the server **srv2**, create a local user **dev1**:

```
# useradd dev1
```

Change to the newly created user and check the Samba mount:

```
# su - dev1
```

```
$ ls -l /mnt/
ls: cannot access /mnt/samba_group: Permission denied
total 12
dr-xr-xr-x. 10 root root 4096 May  7  2014 rhel7dvd
d??????????  ? ?    ?       ?            ? samba_group
drwxr-xr-x.  2 root root 4096 Jun  7 19:55 samba_pub
```

We can use **cifscreds** command to add authentication credentials to the current session (keyring) of a user:

```
$ cifscreds add srv1
Password:
```

Check the Samba mount again:

```
$ ls -l /mnt/
total 12
dr-xr-xr-x. 10 root root 4096 May  7  2014 rhel7dvd
drwxrwsr-x.  2 root dev1    0 Jun  7 19:57 samba_group
drwxr-xr-x.  2 root root 4096 Jun  7 19:55 samba_pub
```

We should be able to write now:

```
$ touch /mnt/samba_group/test
```

```
$ ls -l /mnt/samba_group/test
-rw-r--r--. 1 dev1 dev1 0 Jun  7 19:58 /mnt/samba_group/test
```

And if we check on the Samba server **srv1** with **smbstatus**, we should see active connections for both users **dev1** and **dev2**.

References

https://www.samba.org/samba/docs/using_samba/appb.html

This entry was posted in Linux, Samba/NFS and tagged CentOS, cifs, EX300, RHCE, RHEL, Samba. Bookmark the permalink. If you notice any errors, please contact us.

← Setting up a Kerberised NFS Server on RHEL 7

Setting up an OpenSSH Server with SELinux on RHEL 7 →

58 thoughts on “Setting up a Samba Server with SELinux on RHEL 7”

Martin Chamambo says:
31/07/2016 at 10:24 am

Hie Tomas

This line gives me an error on centos 7.2

//srv1.rhce.local/public /mnt/samba_pub cifs
username=guest,password= 0 0

I am still checking to see if there are other ways of mounting the share using guest access

Reply

Tomas says:
31/07/2016 at 11:20 am

It works fine for me on RHEL 7.2

Aleks says:
27/03/2018 at 3:11 pm

try with mount.cifs or mount -t cifs

Martin Chamambo says:
31/07/2016 at 3:57 pm

Let me keep checking ,somehow its giving me a mount error : permission denied error while the other share samba_group is working perfectly with credentials.

Reply

Tomas says:
31/07/2016 at 4:31 pm

I'm sure you'll figure it out.

Martin Chamambo says:
31/07/2016 at 5:32 pm

had missed this line below and including it fixed my problem

```
map to guest = bad user
```

Reply

Tomas says:
31/07/2016 at 6:17 pm

Thought so.

Martin Chamambo says:
21/08/2016 at 11:59 am

@tomas ,i am trying the multiuser option and dont really know what i am missing.

on the samba server i have this

```
[multi]
comment = Multi Share
path = /srv/samba_multi
writable = no
browseable = yes
printable = no
guest ok = no
write list = @devops
read list = dev2
valid users = @devops, dev2
```

and on the client i have this

```
//rhce.example.com/multi /mnt/samba_multi cifs
username=dev2,multiuser,sec=ntlmssp 0 0
```

and i created a dev1 local user on the client
the cifscreds add rhce is not giving me permissions and when i reboot the client ,the multiuser mount option asks me for the dev2 password ,is this normal

Reply

Tomas says:
21/08/2016 at 12:14 pm

Read the article carefully, it’s all explained.

Martin Chamambo says:
23/08/2016 at 12:56 pm

Somehow my client doesnt want to mount using any user who isnt in the devops group , will try again with a fresh install and see how it goes.Any user in the devops group is able to mount it with the multiuser option without any issues

Reply

Tomas says:
23/08/2016 at 1:00 pm

Let us know once you manage to fix this, it may help others.

Martin Chamambo says:
29/08/2016 at 3:24 pm

@everyone ,i managed to figure this one out after reading Micheal Jang. The multiuser option will work as explained on this blog post but the only caveat for me ,was the multi user mount was refusing to work if i used dev2 ,which isnt part of the devops group.so the only way it worked was for the dev2 user to have r and execute access to the /srv/samba_multiuser share folder via setfacl ………..

Reply

Tomas says:
29/08/2016 at 4:12 pm

I don’t mean to sound rude Martin, but it’s all explained in the blog post.

When a user authenticates to the Samba server, a Samba user account is used, but the Samba user account is mapped to a Linux user account, and **that user account needs access permissions**.

Your dev2 user needs access permissions. You can do it with setfacl if you wish, or you can do it as in this article:

```
# chmod 2775 /srv/samba_group
```

Santosh says:
08/02/2018 at 1:39 pm

Hi Tomas, I gave the separate permissions with setfacl but still I'm having issue. I don't have any problem to mount. I get the permission deny even though cifscreds add. I tried with both permissions. Cifscreds add system1 -u user1.user1 has full permissions Please let me know where I missed?

Martin Chamambo says:
29/08/2016 at 5:03 pm

its okay @tomas ,its my bad ,i guess i am used to the 2770 group permission where everything is restricted to the users and the groups only but as you explained it above.that works

Reply

Yash says:
17/03/2017 at 1:31 am

It is good Tomas has emphasized to read clearly how access is granted based on mapped linux user.
I tested this way:
on srv1 I added extra two user – bob, lisa – in the same group devops
then on second machine srv2:
# useradd bob1;su – bob
$ cifscreds add -u lisa srv1
$ touch file1 /mnt/samba_group/

check its ownership on srv1. It is not bob, even you think you did su – bob, so file should be created with bob as owner, but cifscred mapped user lisa
-rw-r–r–. 1 lisa1 devops 0 Mar 16 20:55 file1

so now I understand why we should share and mount with least access and let user elevate their access based on need
—
another note. It is really important not to miss all three words when you mount :credentails=whateverfile.txt,multiuser,sec=ntlmssp . I once

forgot multiuser and wasted 15 minutes on troubleshooting.

Alex says:
23/11/2016 at 8:57 pm

Hello Tomas,
I am confused with the booleans.
If I wanna share a standard directory, say, homedirs, I should enable samba_export_all_rw and samba_enable_home_dirs.
If I wanna share a non-standard directory only via Samba, I use samba_share_t
If I wanna share a non-standard directory via Samba and NFS, I use samba_public_content_t or samba_public_content_rw_t.
Is everything correct?

Reply

Tomas says:
23/11/2016 at 9:27 pm

What do you mean by saying “a standard directory”? You need **samba_enable_home_dirs=**1 if you want to share users home directories.

If you have a directory that you want to be accessed through Samba, use **samba_share_t**. If you need that directory to be also accessed through FTP, Apache and rsync, use either **public_content_t** or **public_content_rw_t**.

Alex says:
23/11/2016 at 10:34 pm

https://selinuxproject.org/page/SambaRecipes – one of the sources where “standard” directories are mentioned (bold-text paragraph).
Thanks for the confirmation about the contexts.

Reply

Tomas says:
23/11/2016 at 10:48 pm

Ah, I see, basically the ones that come with an OS. Thanks.

thegeekaid says:
01/01/2017 at 9:10 am

I have problem with guest share, when i try to mount the share folder i get an error error(13): Permission denied.But if i use Nautilus and try to browse to the samba share it work fine,
Any help appreciated

smb.conf configuration

```
[public]
comment = public
path = /public
browseable = yes
writeable = yes
guest ok = yes
```

————————————————————

permission of the public folder
drwxrwxrwx. 3 root root 18 Jan 1 03:49 public

————————————————————

```
mount //ldap/public /mnt -o username=guest,password=
mount error(13): Permission denied
Refer to the mount.cifs(8) manual page (e.g. man mount.cifs)
```

Reply

Tomas says:
03/01/2017 at 10:29 pm

Do you have bad users mapped to guest? This looks like a problem to me.

hunter86_bg says:
17/04/2017 at 4:19 pm

Hi friends,

I have a question that came up my mind after I've learned that Samba supports Unix ACLs (and windows ofc).I feel more comfortable using them, but if the samba server is evaluated by a Windows machine – I'm not sure how well it will work out.
What do you think about using ACLs? If the Folder is "writable = yes" and the "inherit acls = yes" then all depends on the file/folder permissions on the Samba Server.

Reply

Tomas says:
17/04/2017 at 6:06 pm

I don't use Windows in this context so cannot really tell much.

hunter86_bg says:
18/04/2017 at 8:24 am

Sadly I found an issue , which turns ACLs useless unless AD/LDAP is used. In order ACLs to work – both the user on the Samba server and on the client machine should have the same UID /GID for groups/ as in ALC mode we are rely on File System permissions only.
I guess for the exam both methods will do the trick. Either Samba controls permissions or the File System of the share.

Reply

Tomas says:
18/04/2017 at 12:16 pm

Thanks.

Michael says:
12/05/2017 at 8:24 pm

There's a few things missing from the examples that will cause permission denied errors. The [global] section needs the following line to allow host restrictions to work.

hostname lookups = yes

Reply

Tomas says:
19/05/2017 at 1:49 pm

This directive isn't required unless you use hostname lookups with hosts deny and hosts allow. These weren't used in the examples, therefore I'm not sure on what permission denied errors you refer to. Unless you put a DNS name and not an IP address as per example, you shouldn't have any issues.

Mirec says:
01/09/2017 at 8:50 am

Hi Tomas,

I have some improvement suggestions for your group share samba example.
The "public" option is a synonym for "guest ok", so listing both options with the same value (used in your public share example) is useless and listing both options with different values (like in your group share example) is (at least) confusing. I suggest removing the "public = yes" statement in your example, as it contradicts to "guest ok = no".
Another possible improvement area (depending on, if files placed in your group share should be writable by the group per default) in your example might be to consider setting an explicit mask for the group share files with the following options:
create mask = 0660
force create mode = 0660
The "create mask" and "force create mode" options ensure, that, when a user in group1 creates a new file, the permissions will be set to 0660. By default, files were created with 0744, which prevents other members of the group from writing to the files, unless the user creating the file manually assigns write permissions for the group.
After setting those options, this would be done automatically.

Reply

Tomas says:
01/09/2017 at 1:10 pm

Hi Mirec, thanks for your feedback, these are really good points! I'll update the article making a note that **public** is also called **guest ok**.

Setting masks is optional in my opinion and depends on your set up.

olive says:
24/09/2017 at 6:45 pm

Hi,

I have a problem with cifscreds but it does not seem to work. I mounted a share with multiuser option with a user that has rw so I am trying to test a second user with ro permissions but I can t get the credentials for that user. Am I missing something. I thought that you can switch from different users and inherent the permissions

Cheers,
olive

Reply

Scott says:
30/11/2017 at 9:39 pm

I have encountered the same issue using RHEL 7.0. Maybe the issue was resolved in a subsequent release? It appears that the elevated permissions persist from one user to the next and not removed when the user session is terminated.

Bjarni says:
01/11/2017 at 9:44 pm

Hi Tomas.
I'm having some trouble when I want to share a directory that is not located on the "/" filesystem.
For example I have a sambashare on /sambashare and another on /data/sambadev
I've configured SELinux for both locations, f.x
semanage fcontext -a -t samba_share_t "/srv/samba_dev(/.*)?"
restorecon -R /srv/samba_dev

Here is my smb.cof
[sambashare]
comment = /sambashare
path = /sambashare

```
browseable = yes
writeable = no
public = no
write list = @sambagroup
valid users = @sambagroup
force group = +sambagroup
[Sambadev]
comment = /data/sambadev
path = /data/sambadev
browseable = yes
writeable = no
printable = no
write list = @devops
valid users = @devops
public = no
```

The share named “sambashare” works perfectly fine but the when I try to mount the “Sambadev” share I get the following “error mount error(6): No such device or address”

I’ve disabled SELinux and the firewall and I still get this error. Could give me some input on what I am doing wrong.

Reply

Bjarni says:
01/11/2017 at 9:59 pm

Well, I feel stupid…. I was trying to mount the full path instead of the section name..

Tomas says:
02/11/2017 at 10:05 am

Ah, I see the confusion, for the first share you used the same name for the share as well as the path, but it was different in the second case where you tried using the path to mount it. I’m glad you got that sorted.

exot says:
09/11/2017 at 9:53 pm

I’ve encountered a problem following these configurations when it comes to the public share.

On the Samba Server, 777 permissions have been set to /publicshare yet guests are not able to write to it on the Samba client.

Guests are only able to read, not write.

Anyone know what the solution may be? I've followed this article to the T and still keep getting this same issue.

Reply

Tomas says:
10/11/2017 at 9:00 am

Are you mapping bad users to guests?

Is the public share writable `writable = yes`?

Ryan says:
09/12/2017 at 1:14 pm

Tomas, thanks for Great resources.Question 12 doesn't ask to mount .Mount is needed on srv2.rhce.local or not?how about entry in fstab?

Reply

LinuxDS says:
30/12/2017 at 7:27 am

thanks for this amazing post Tomas. I have been following your RHCE blogs.Would like to ask samba specific Q here. What does it mean when its asked to create samba share with access to domain users only/ accessible to subdomY.domainX.com ONLY ?
Thanks.

Reply

Tomas says:
31/12/2017 at 2:14 pm

It means that only users from that domain should be able to access the share.

LinuxDS says:
01/01/2018 at 5:29 am

and how do we achieve that ??

Tomas says:
01/01/2018 at 3:12 pm

I see that you've figured it out already.

LinuxDS says:
01/01/2018 at 5:31 am

Meaning is there a "valid users" or some other directive that we need to define in smb.conf? or a firewall-cmd rich rule ??

Tomas says:
01/01/2018 at 3:14 pm

There is a "valid users" option, yes. It lists the users allowed to access the share.

LinuxDS says:
01/01/2018 at 6:00 am

Found it in this page itself.
I guess by defining the "hosts allow", the access to ap particular domain can be achieved if we know the subnets.
hosts allow = 127. 10.8.8.

Reply

Ramesh says:
09/02/2018 at 2:07 am

Tomas,please help me on this where I missed?
Samba server:

```
[multi]
path = /paas
writable = yes
browseable = yes
valid users = brian bina
write list = brian
```

fstab entry in client :

//192.168.10.2/paas /mnt/multi cifs credentials=/root
/bina,multiuse,sec=ntlmssp 0 0
I'm able to mount /pass under /mnt/multi but having permission issues on cifscreds.brian has rwx and bina has rx permission on /paas (with setfacl) .I have created bob local user on the client. When I tried to add cifscreds for bob : I did : su – bob
bob@ ,,,cifscreds add 192.168.10.2 -u brian entered brian password and #cd /mnt/multi then #touch ll.It says permission deny even though brian is getting rwx permission.Also this user is in write list in smb.conf file. Same thing is with bina user which I think is right but why I'm getting permission issue with brian?

note: /paas is getting public_content_rw_t selinux type . Thanks

Reply

Barbie says:
10/02/2018 at 4:04 pm

Hi Tomas,Can you please suggest to me how to fix the following error.
ipv4 network is 172.25.1.0.Server ip is 172.25.1.1 and client ip is 172.25.1.2 and I added hosts allow = 127. 172.25.1.
all other configuration is correct in the smb.conf file.when I do:
smbclient -L //localhost
enter
enter
protocol negotiation failed:
NT_STATUS_INVALID_NETWORK_RESPONSE : on Cerver
mount -o username=user1 //172.25.1.1/data /mnt/multi
protocol negotiation failed:
NT_STATUS_INVALID_NETWORK_RESPONSE on Client

Reply

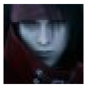
Tomas says:
12/02/2018 at 12:48 pm

The error suggest that the client is being denied access by hosts allow parameter in `/etc/samba/smb.conf`. Please verify.

Barbie says:
13/02/2018 at 2:06 am

For my scenario I tried both of the following in
hosts allow = 127. 172.25.1. but did not worked.Can you please suggest to me for correct order.I don't know where I missed.I also tried 127. example.com Any specific rule needed for this case?

Reply



Tomas says:
13/02/2018 at 10:03 pm

No specific rules should be required.

Jank says:
17/02/2018 at 9:22 pm

Nice explanation on Samba here.
In regards cifscreds, is there a way to do this in a permanent way, so that when you issue a
cifscreds add srv1
it will survive a reboot?

Reply

Sven says:
23/10/2018 at 9:08 pm

I'm currently studying for the RHCE exam. A big thank you for your work.

Some additional notes:
I ran into trouble with the samba public share. Accessing the public share as root works without problems. But when you try to access the share as a non-root user, you can create a file (0777), but you can't write into the file. Therefore you have to use the "noperm" mount option to avoid this:

```
//samba-server/public /mnt/public cifs defaults,noperm,username=guest,password=,_netdev 0 0
```

And another thing:
According to the Manpage x-systemd.automount is the Systemd replacement for autofs. There's no need to use this for every net drive.

Reply

Tomas says:
28/10/2018 at 9:35 am

Thanks for your feedback, this is helpful.

omipenguin says:
31/10/2018 at 7:51 am

Hi Tomas,

I created two shares /srv/samba-pub and /srv/samba-grp

on /public i set the Selinux context and Permissions like this

```
drwxrwsr-x. root devops unconfined_u:object_r:samba_share_t:s0 /srv/samba-grp
drwxrwxrwx. nobody nobody unconfined_u:object_r:public_content_rw_t:s0 /srv/samba-pub
```

And in smb.conf i included this options for both shares

```
[public]
comment = Public Stuff
path = /public
browsable =yes
writable = yes
guest ok = yes
read only = no
force user = nobody
```

```
[group]
comment = Samba Group Share
path = /srv/samba-grp
public = no
valid users = @devops, dev2
read list = dev2
write list = @devops
```

From windows machine i can access the /srv/samba-grp share without any problem, but group share gives permission denied.

In Hosts allow i set
"hosts allow = 127. 10.8.8."
and also included
guest account = nobody

security = user

Any idea what im doing wrong

Reply

Panos says:
31/12/2018 at 8:25 pm

In my case to make group collaboration to work nicely in the latest Red Hat release (7.6), the mount command needs vers=1.0.

Without to specify the version in the mount command the default mount options are: uid=0,noforceuid,gid=0,noforcegid

Reply

Tomas says:
01/01/2019 at 9:49 am

Thanks!

dobo says:
13/02/2019 at 5:35 pm

Perfect step-by-step explanation, thank You.
Regarding SElinux things – there is a long comment of the samba/selinux things at the begining of smb.conf file. I always used this to assign proper label for the samba shares.

Reply

Tomas says:
13/02/2019 at 9:07 pm

No worries, thanks.

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

Post Comment

LOOKUP

Search …

ARCHIVES

Archives

Select Month

CATEGORIES

AWS (14)

Database (8)

DNS (5)

Exchange Server (6)

FTP (3)

High Availability (23)

LDAP/Kerberos (9)

Linux (184)

Mac OS X (2)

- Mail/SMTP (6)
- Monitoring (29)
- Networking (11)
- Notes (8)
- OpenVPN (3)
- Proxy (3)
- Python (4)
- Raspberry Pi (4)
- Samba/NFS (7)
- Security (17)
- SSH (2)
- Virtualisation (8)
- VoIP (4)
- Webserver (7)
- Windows (24)

RECENT COMMENTS

- Tomas on Setting up a Samba Server with SELinux on RHEL 7
- Tomas on Simple Python Script to Start and Stop Amazon AWS Instances
- dobo on Setting up a Samba Server with SELinux on RHEL 7
- tzah on Simple Python Script to Start and Stop

Amazon AWS Instances

Tomas on GRUB2 Rescue Mode "error: unknown filesystem"