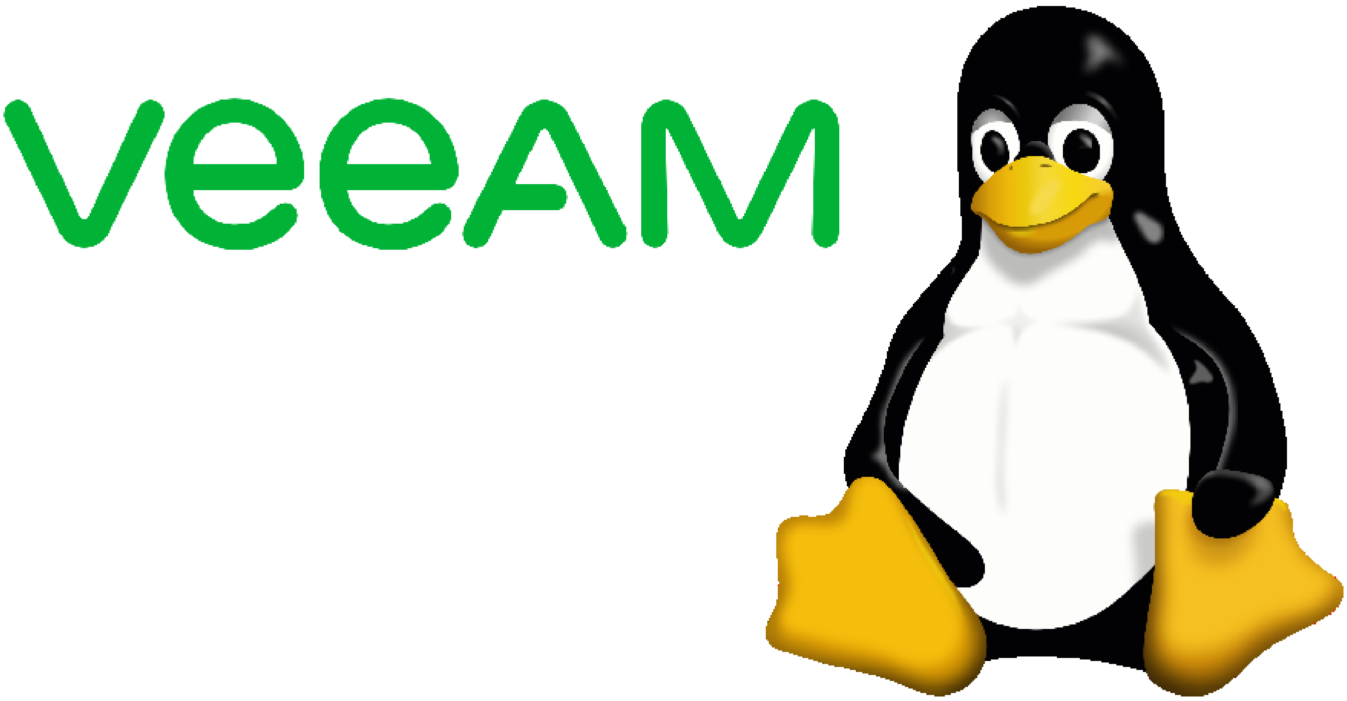
Build an immutable backup repository for Veeam Backup & Replication. Part 9



This guide will show you, step by step, how to create and implement a disk-based immutable Veeam backup repository from scratch. In this part: Maintenance and deactivation/reactivation of MFA/2FA.

Introduction

Purpose of these articles

You are a Windows administrator running Veeam Backup & Replication and wish to raise protection against malware attacks and hackers without reverting to shuffle or rotate physical media.

This you can accomplish by *immutable backups* stored on a physical server running Linux. However, you have no Linux servers running and don't want to.

But, like it or not, that is your only option, as the XFS file system is the only one capable of immutability, and XFS only runs under Linux.

Thus, a Linux server is a must. When you have accepted this fact, then what? Where to start?

Like me, you have about zero experience with Linux and, therefore, hesitate to set up a Linux server, indeed in a production environment.

If so, this guide is for you. Here, nothing about Linux is taken for granted.

Sections

The guide has been split in eight parts. This allows you to skip parts you are either familiar with or wish to implement later if at all.

Requirements

You are familiar with:

- the usual tasks administering at least a small network with one Windows Server
- *Veeam Backup & Replication* and have it installed and running
- the command line - from PowerShell, Command Prompt, or even DOS

> *Veeam Backup & Replication* is assumed to be of *version 11* or later. It can be a licensed trial or paid version or even the free *Community Edition*.

Part 9. Maintenance and deactivation/reactivation of MFA/2FA

In the previous section, we tightened the security on the server. However, in certain cases, it will be convenient (or even a demand) to turn off 2FA/MFA temporarily.

One example is, when it is time to update the *Veeam Linux Transport* running on the Linux server. This will be the case, if *Veeam Backup & Replication* has been updated and requires a matching (updated) transport service installed on the Linux server.

The route to follow isn't convoluted:

- Turn off MFA/2FA authentication
- Perform the maintenance required
- Turn on MFA/2FA authentication

In the following, we will first show how to turn off and on MFA/2FA authentication, and then - having MFA/2FA turned off - how to update the *Veeam Linux Transport* on the Linux server. It is not difficult, only not completely straight-forward.

Deactivate MFA/2FA temporarily

MFA/2FA can be turned off with a few commands.

First, open the *pam* configuration file in the editor with this command:

```
sudo nano /etc/pam.d/sshd
```

Locate the line (probably the last line):

```
auth required pam_google_authenticator.so
```

Comment out that line by prefixing it with an *octothorpe* (#) like this:

```
# auth required pam_google_authenticator.so
```

Press *Ctrl+O* and *Enter*, and exit the editor with *Ctrl+X*.

Next, open the *sshd* configuration file, again using the *nano* editor:

```
sudo nano /etc/ssh/sshd_config
```

In this file, locate the line:

```
ChallengeResponseAuthentication yes
```

and change its status from "yes" to "no". This will instruct SSH to not ask for an authentication code whenever someone attempts to log in to the system.

Press *Ctrl+O* and *Enter* to save the file, then *Ctrl+X* to exit.

Finally, restart the service:

```
sudo systemctl restart sshd.service
```

You can now log in via SSH without a verification code. No reboot is required.

Reactivate MFA/2FA

If you wish to turn on again the MFA/2FA authentication, all that is needed is to reverse the changes in the two configuration files you made above.

First, open the *pam* configuration file with the command we used previously:

```
sudo nano /etc/pam.d/sshd
```

This time, remove the octothorpe (#) from the line:

```
# auth required pam_google_authenticator.so
```

to make it read:

```
auth required pam_google_authenticator.so
```

Press *Ctrl+O* and *Enter* to save the file, and exit with *Ctrl+X*.

Next, open the *sshd* configuration file, again using the *nano* editor:

```
sudo nano /etc/ssh/sshd_config
```

In this file, again locate the line:

```
ChallengeResponseAuthentication no
```

and change its status from "no" to "yes". This will instruct SSH to ask for an authentication code whenever someone attempts to log in to the system.

Press *Ctrl+O* and *Enter* to save the file, then *Ctrl+X* to exit.

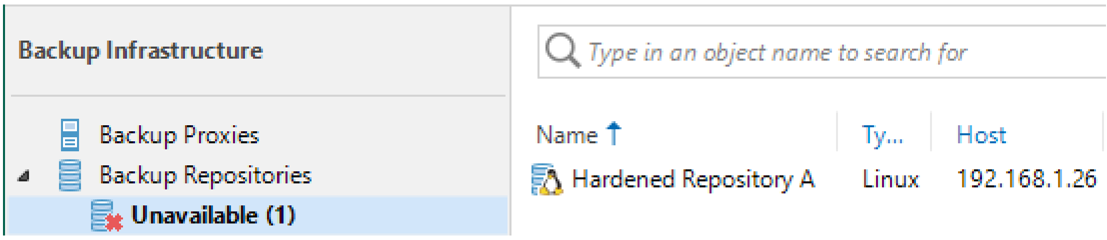
Finally, restart the service:

```
sudo systemctl restart sshd.service
```

MFA/2FA has again been activated for remote connections to the Linux server.

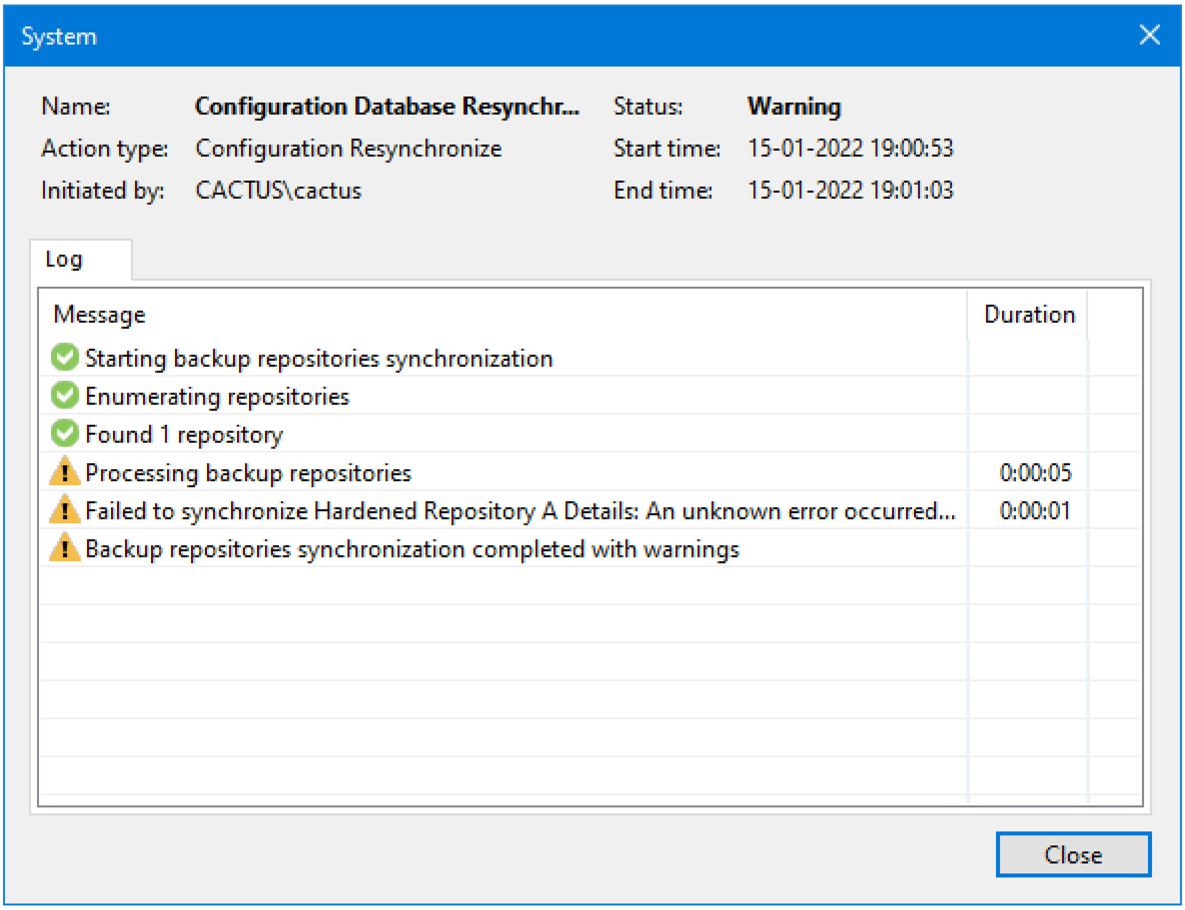
Updating the Veeam Linux Transport

After an upgrade of *Veeam Backup & Replication*, it will be out of sync with the version of the *Veeam Linux Transport* installed on the Linux server. In the VBR console, you may see the repository on the Linux server listed as **Unavailable**:



Name	Ty...	Host
Hardened Repository A	Linux	192.168.1.26

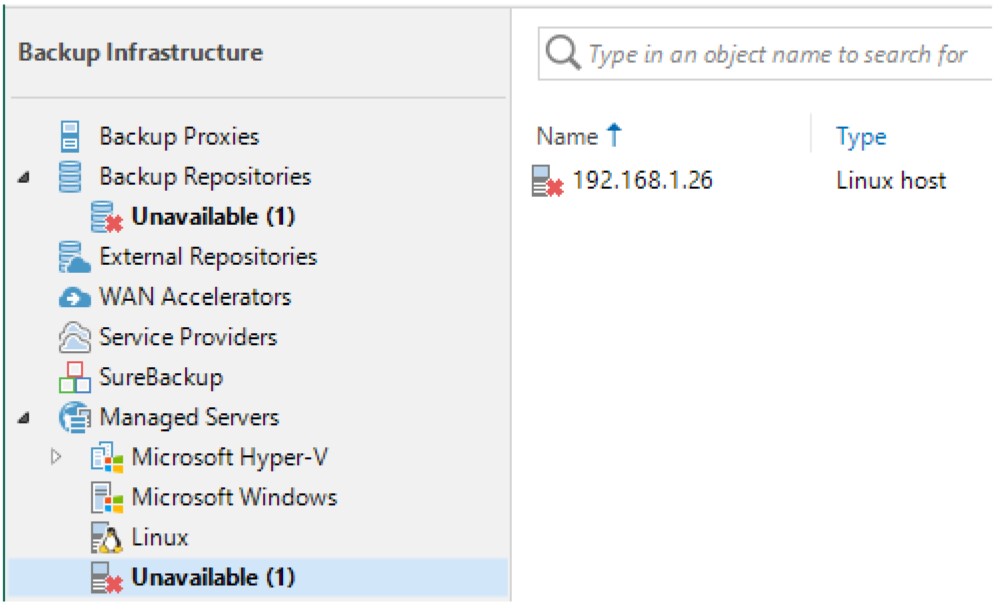
If you right-click the repository and select *Rescan*, this will complete with a status of *Warning*:



Name: **Configuration Database Resynchr...**
Status: **Warning**
Action type: Configuration Resynchronize
Start time: 15-01-2022 19:00:53
Initiated by: CACTUS\cactus
End time: 15-01-2022 19:01:03

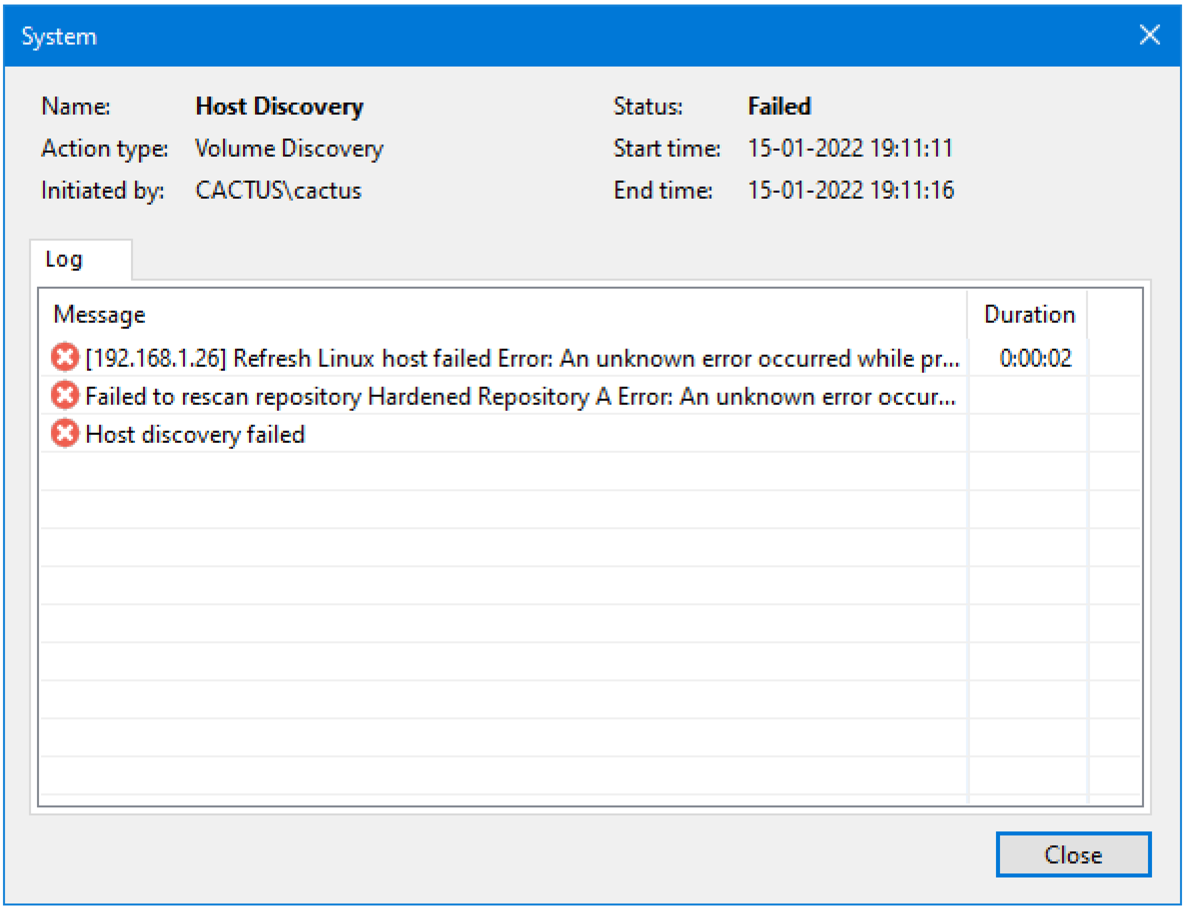
Message	Duration
✓ Starting backup repositories synchronization	
✓ Enumerating repositories	
✓ Found 1 repository	
⚠ Processing backup repositories	0:00:05
⚠ Failed to synchronize Hardened Repository A Details: An unknown error occurred...	0:00:01
⚠ Backup repositories synchronization completed with warnings	

Likewise will the Linux server itself be listed as **Unavailable**:



Name ↑	Type
192.168.1.26	Linux host

If you right-click the server and select *Rescan*, this will report a status of *Failed*:



Name: Host Discovery
Action type: Volume Discovery
Initiated by: CACTUS\cactus
Status: Failed
Start time: 15-01-2022 19:11:11
End time: 15-01-2022 19:11:16

Message	Duration
✕ [192.168.1.26] Refresh Linux host failed Error: An unknown error occurred while pr...	0:00:02
✕ Failed to rescan repository Hardened Repository A Error: An unknown error occur...	
✕ Host discovery failed	

Simply put: The *Veeam Linux Transport* is out of date, and the immutable repository is inaccessible.

To resolve this, we must take the steps as described in detail in *Part 4*, which enabled *Veeam backup & Replication* to actively install the transport on the Linux server.

Update the Linux system

Open a SSH connection to the Linux server for remote operation (or go to the physical keyboard and monitor of the Linux server).

As an initial step, make sure that the Ubuntu installation is current, as this may be required for the new version of the transport to be installed, and because the updates may improve security in general.

Use this command - as described in detail in *Part 3* of this series - to retrieve and install the updates:

```
sudo apt-get upgrade
```

When prompted, enter *Y* to install the updates. This may take some minutes.

You now have a firm starting point for the update of the *Veeam Linux Transport*.

Deactivate MFA/2FA

If MFA/2FA authentication is enabled on the server, the first step is to *deactivate MFA/2FA* as described above. When done, continue.

Elevate user veeamuser

As you recall, the Linux account (in this series named *veeamuser*) used by *Veeam Backup & Replication* has extremely limited capabilities, thus can't be used for installing anything.

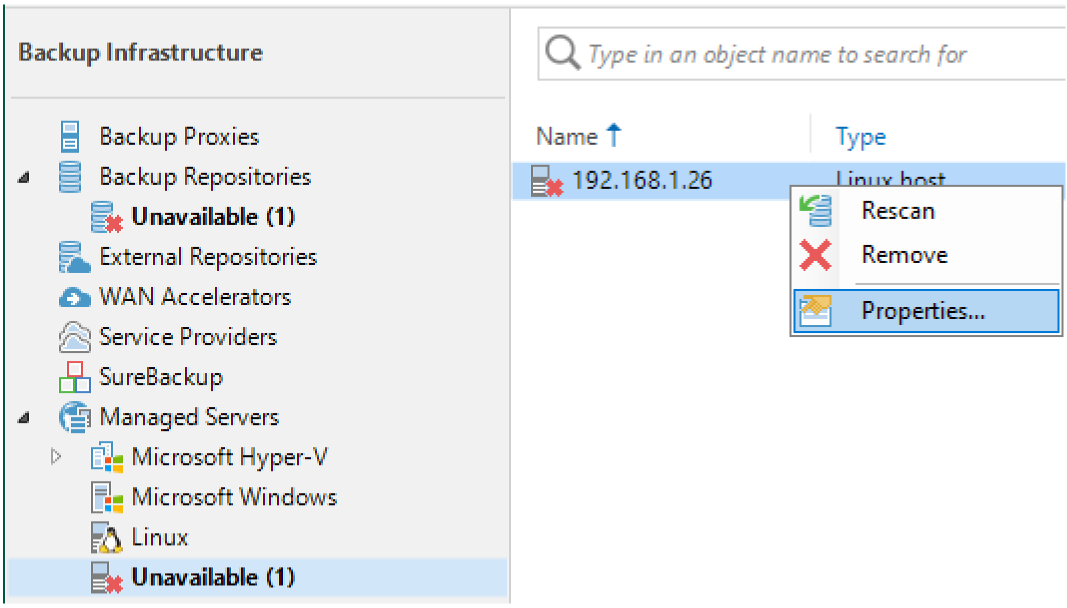
Temporarily, however, we wish to use this account to install the updated transport. To be able to do that, call this command to assign our Veeam user *administrator rights* by signing the account in to the *sudo* group - as explained in detail in *Part 4* of this series:

```
sudo usermod -a -G sudo veeamuser
```

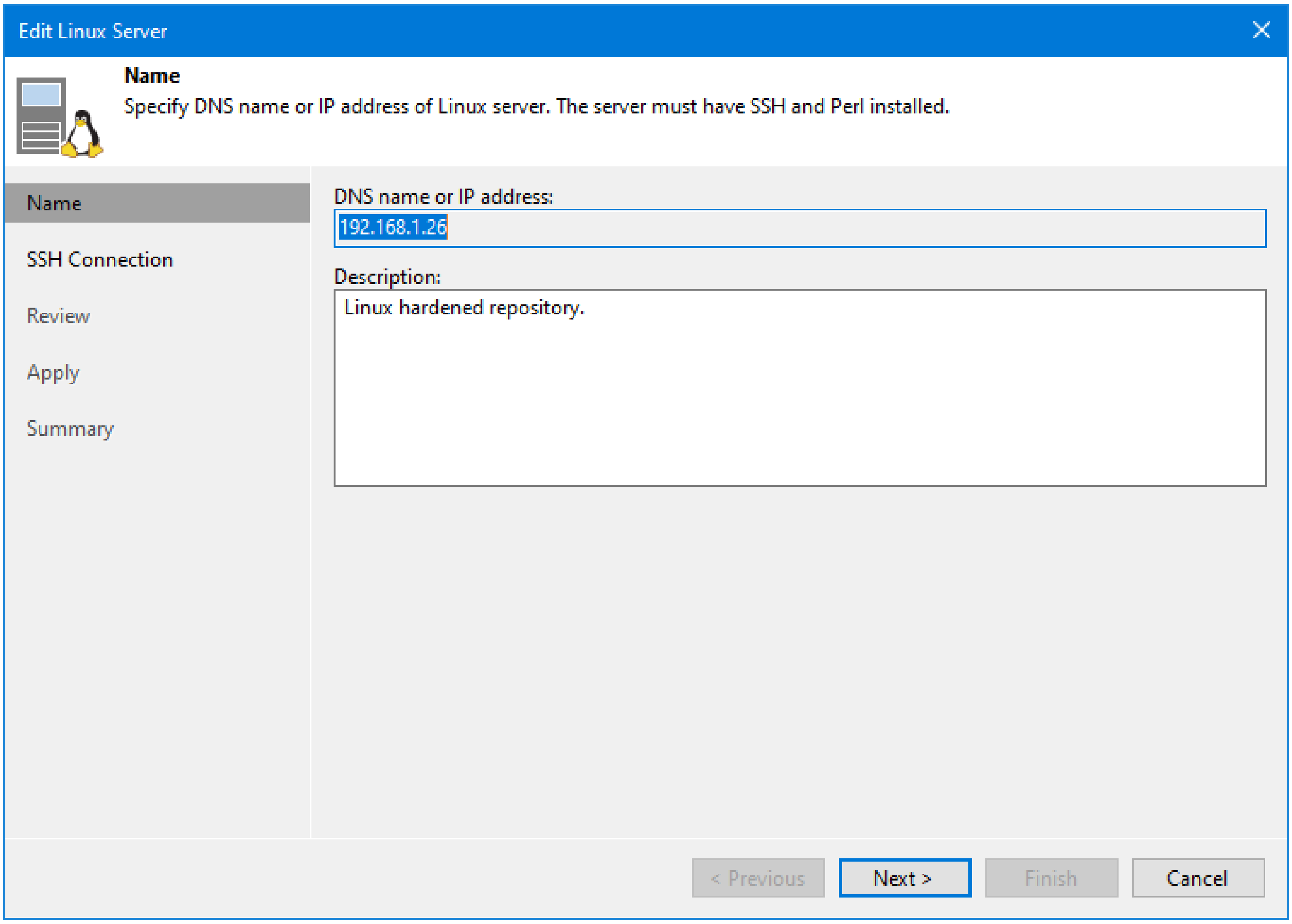
In the command, *veeamuser* is the name of the Linux user account we have set up to be used by VBR.

Update the transport

Now, return to the *Veeam Backup & Replication* console and *right-click* on the Linux server and select *Properties* to view these:



The *Properties* will display with the option to edit these:



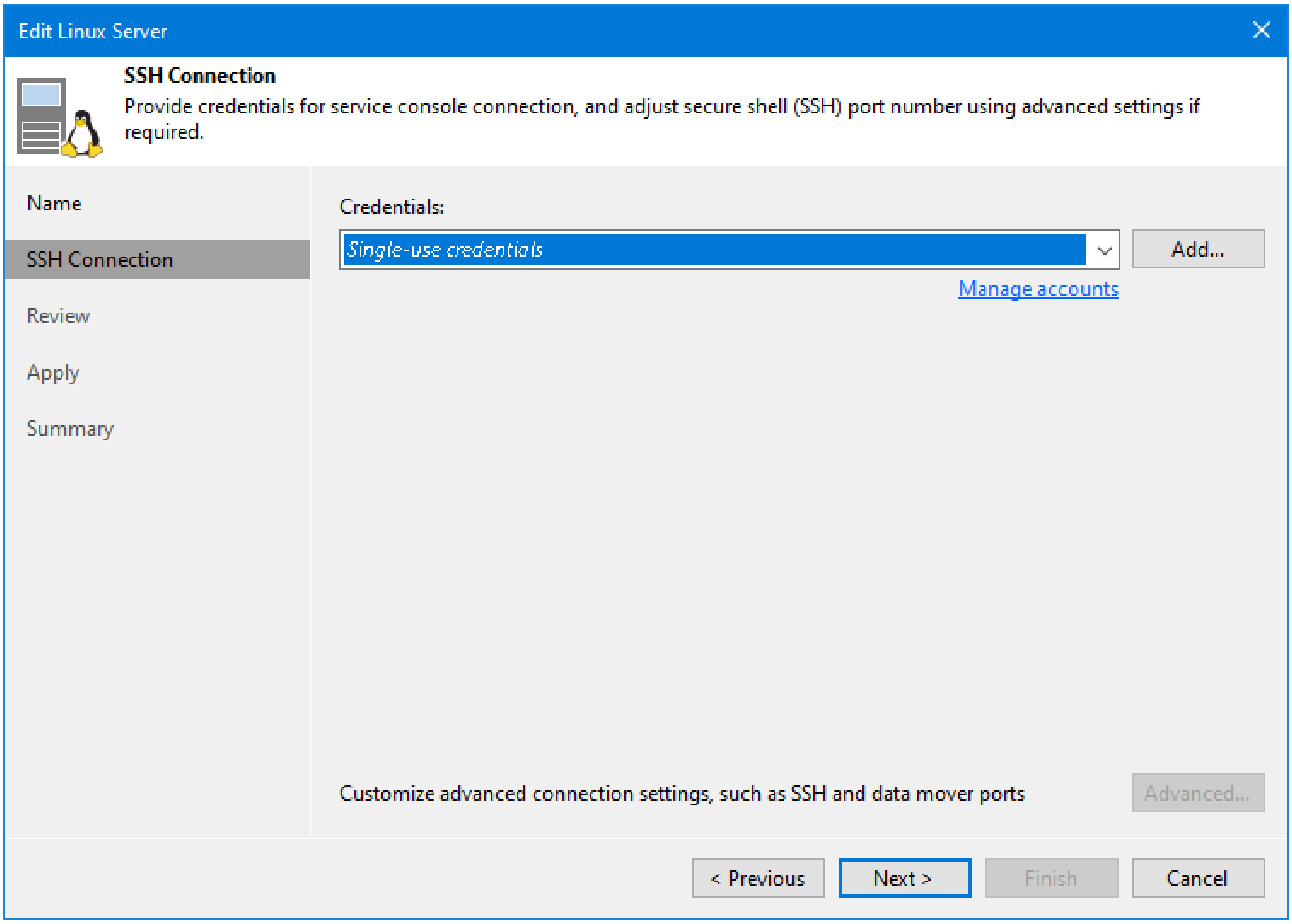
Name

Specify DNS name or IP address of Linux server. The server must have SSH and Perl installed.

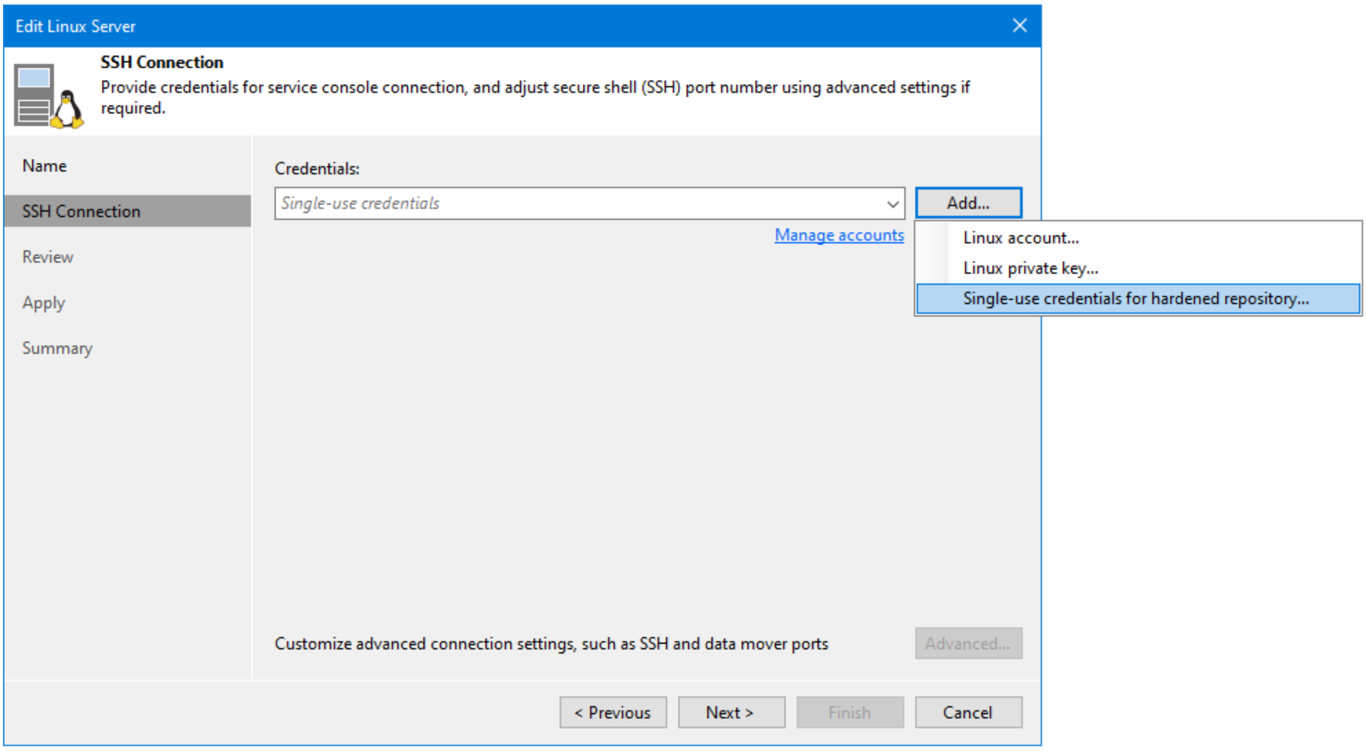
DNS name or IP address: 192.168.1.26

Description: Linux hardened repository.

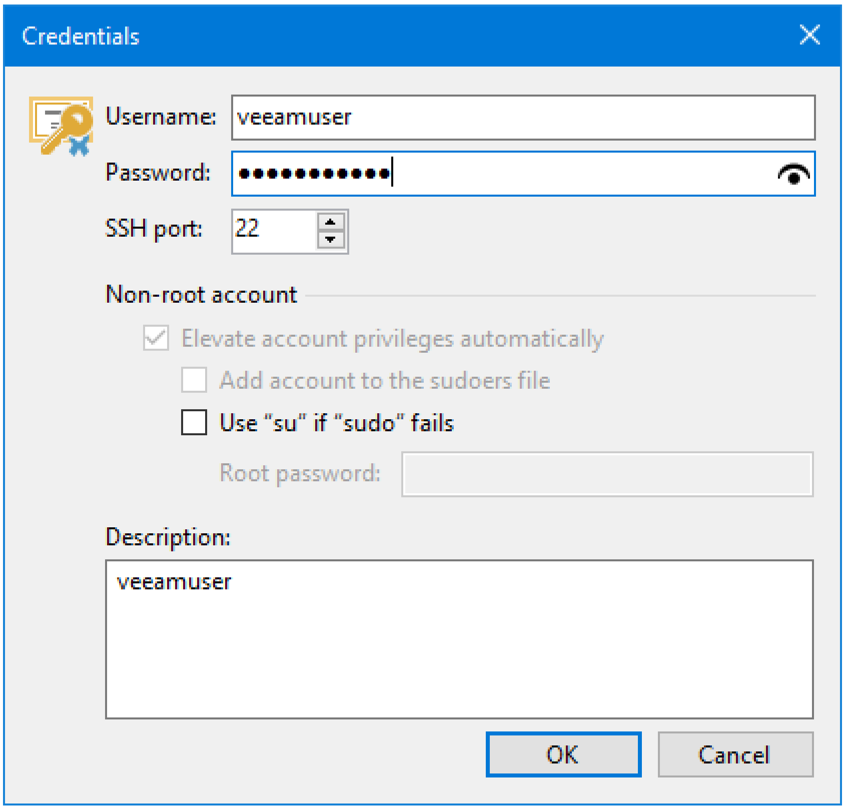
Click *Next* to reach the *SSH Connection* settings:


Edit Linux Server
SSH Connection
Provide credentials for service console connection, and adjust secure shell (SSH) port number using advanced settings if required.
Name
SSH Connection
Review
Apply
Summary
Credentials:
Single-use credentials
Add...
Manage accounts
Customize advanced connection settings, such as SSH and data mover ports
Advanced...
< Previous
Next >
Finish
Cancel

Click *Add* and select *Single-use credentials for hardened repository ...* :


Edit Linux Server
SSH Connection
Provide credentials for service console connection, and adjust secure shell (SSH) port number using advanced settings if required.
Name
SSH Connection
Review
Apply
Summary
Credentials:
Single-use credentials
Add...
Manage accounts
Linux account...
Linux private key...
Single-use credentials for hardened repository...
Customize advanced connection settings, such as SSH and data mover ports
Advanced...
< Previous
Next >
Finish
Cancel

to open the *Credentials* window:



Credentials

Username: veeamuser

Password: ●●●●●●●●●●●

SSH port: 22

Non-root account

☑ Elevate account privileges automatically

□ Add account to the sudoers file

□ Use “su” if “sudo” fails

Root password:

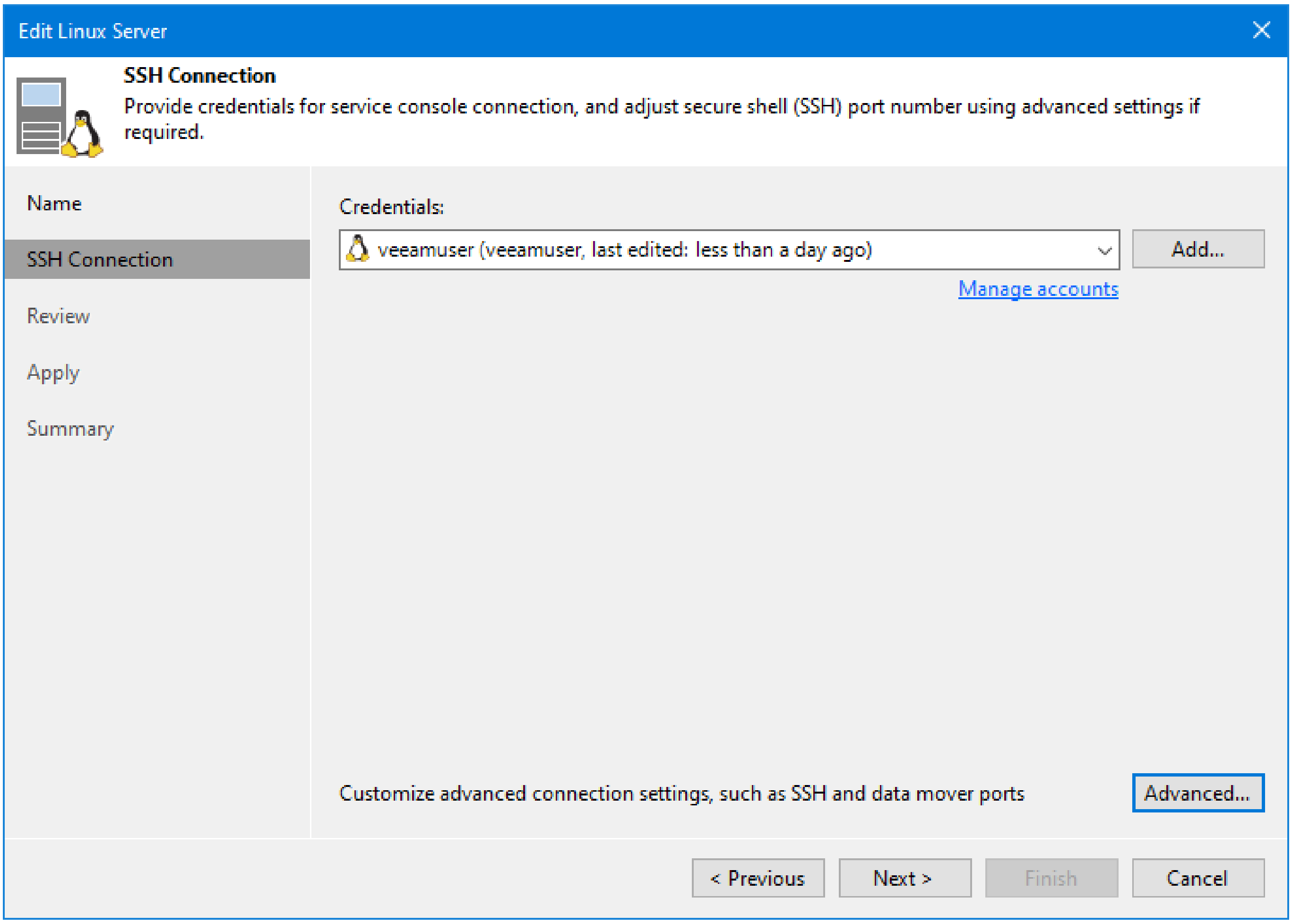
Description:

veeamuser

Type in the username and password as shown.

> Don't just select the previously entered *veeamuser* account. Retyping it ensures, that Veeam will attempt to use the account as an administrator account - as indicated by the checkmark in:
>
> *Elevate account privileges automatically*

Click *OK* to use these credentials:



Edit Linux Server

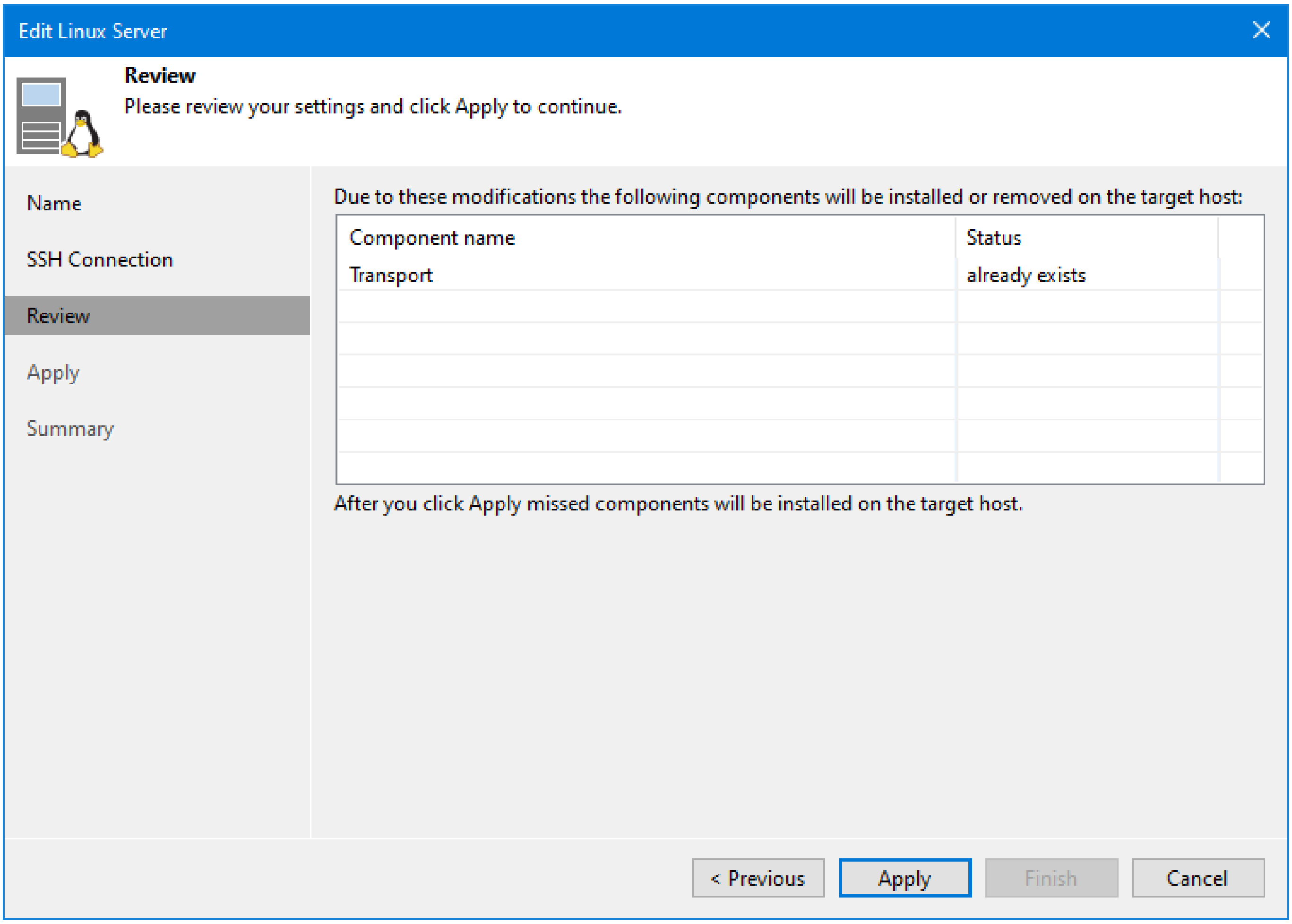
SSH Connection

Provide credentials for service console connection, and adjust secure shell (SSH) port number using advanced settings if required.

Credentials: veeamuser (veeamuser, last edited: less than a day ago)

Customize advanced connection settings, such as SSH and data mover ports

Click *Next* to review and be ready for the next step:



Review

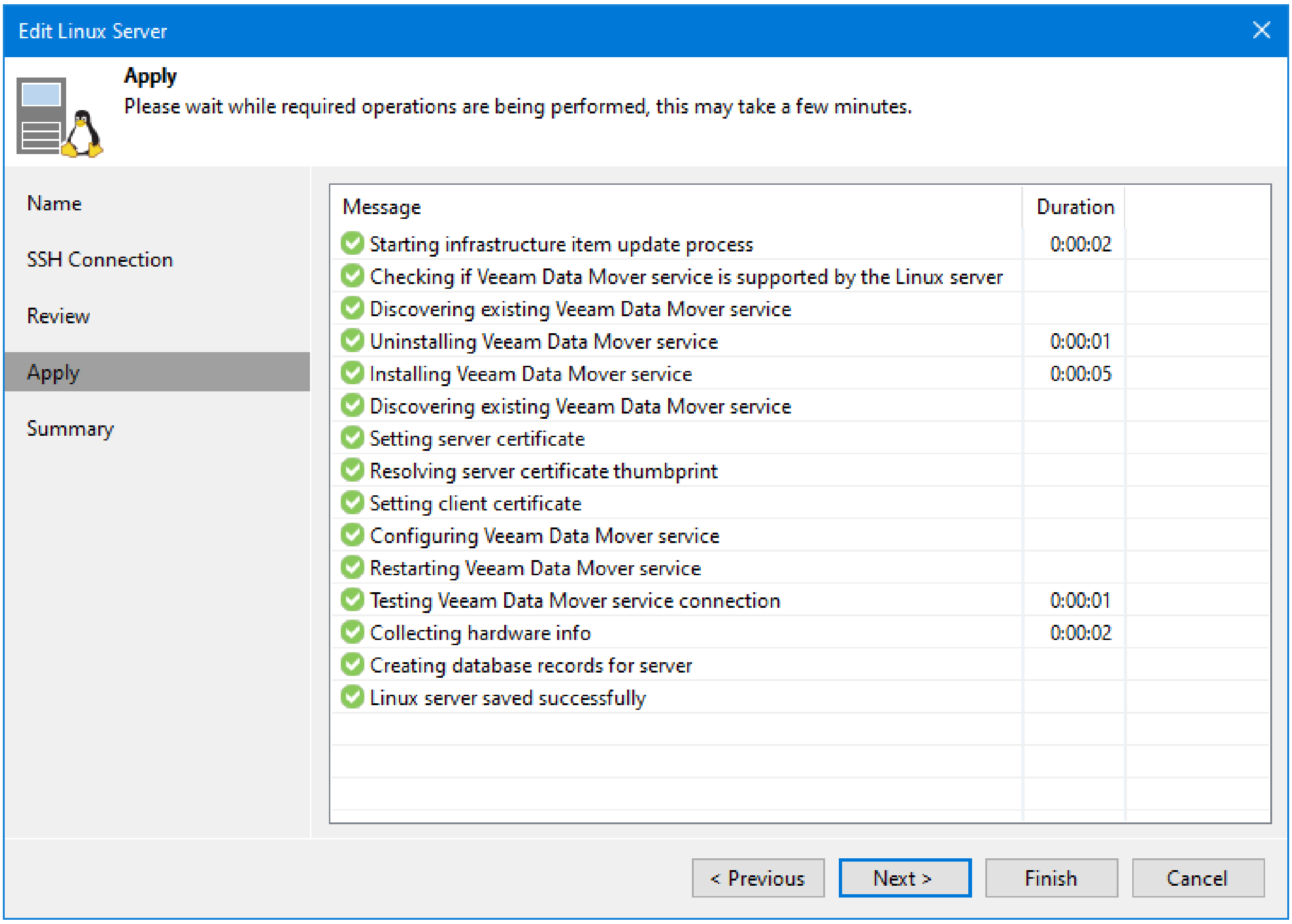
Please review your settings and click Apply to continue.

Due to these modifications the following components will be installed or removed on the target host:

Component name	Status
Transport	already exists

After you click Apply missed components will be installed on the target host.

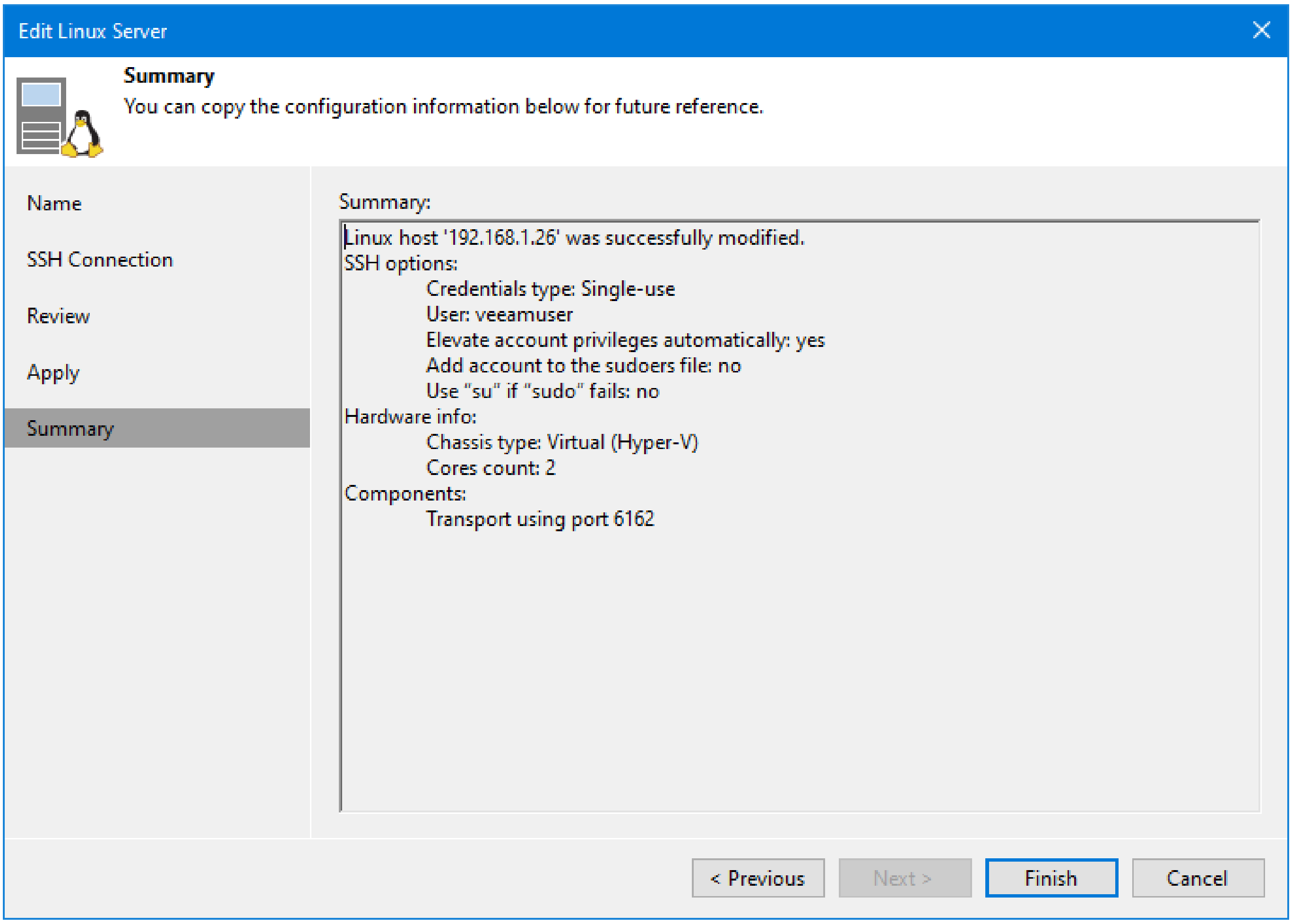
Click *Apply*, and the transport service will be installed while displaying the progress:



Apply
Please wait while required operations are being performed, this may take a few minutes.

Message	Duration
✓ Starting infrastructure item update process	0:00:02
✓ Checking if Veeam Data Mover service is supported by the Linux server	
✓ Discovering existing Veeam Data Mover service	
✓ Uninstalling Veeam Data Mover service	0:00:01
✓ Installing Veeam Data Mover service	0:00:05
✓ Discovering existing Veeam Data Mover service	
✓ Setting server certificate	
✓ Resolving server certificate thumbprint	
✓ Setting client certificate	
✓ Configuring Veeam Data Mover service	
✓ Restarting Veeam Data Mover service	
✓ Testing Veeam Data Mover service connection	0:00:01
✓ Collecting hardware info	0:00:02
✓ Creating database records for server	
✓ Linux server saved successfully	

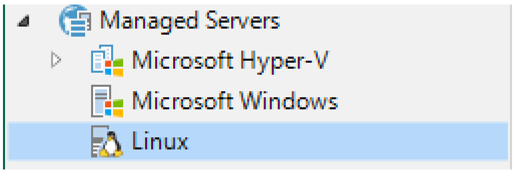
Click *Next* to view the *Summary*:



```
Linux host '192.168.1.26' was successfully modified.
SSH options:
        Credentials type: Single-use
        User: veeamuser
        Elevate account privileges automatically: yes
        Add account to the sudoers file: no
        Use "su" if "sudo" fails: no
Hardware info:
        Chassis type: Virtual (Hyper-V)
        Cores count: 2
Components:
        Transport using port 6162
```

Click *Finish* to exit the wizard.

You'll notice, that under *Managed Servers*, the Linux server is no longer listed under **Unavailable**:



Demote user veeamuser

For a moment, return to the terminal window, because now is the time to demote the veeamuser user account. Again, as explained in detail in *Part 4*, use this command to remove the Veeam user account from the *sudo* group:

```
sudo deluser veeamuser sudo
```

As previously noted, here *veeamuser* is the name of the Linux user account we have set up to be used by VBR. When executed, the command will have reset the user account to only have the limited rights necessary to access the immutable repository on the Linux server.

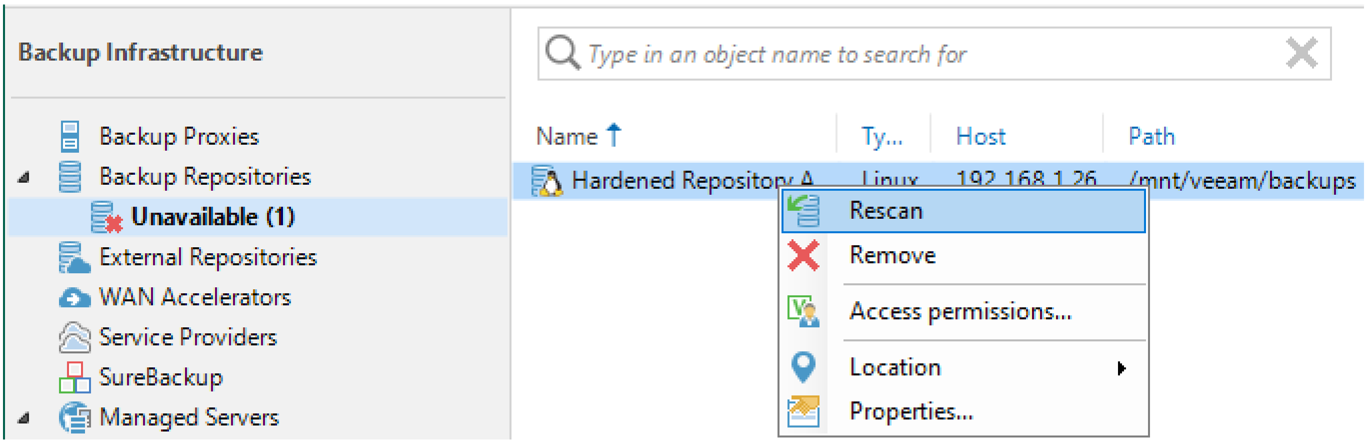
Reactivate MFA/2FA

Also, not to forget if you had MFA/2FA authentication activated, *reactivate* this as described above.

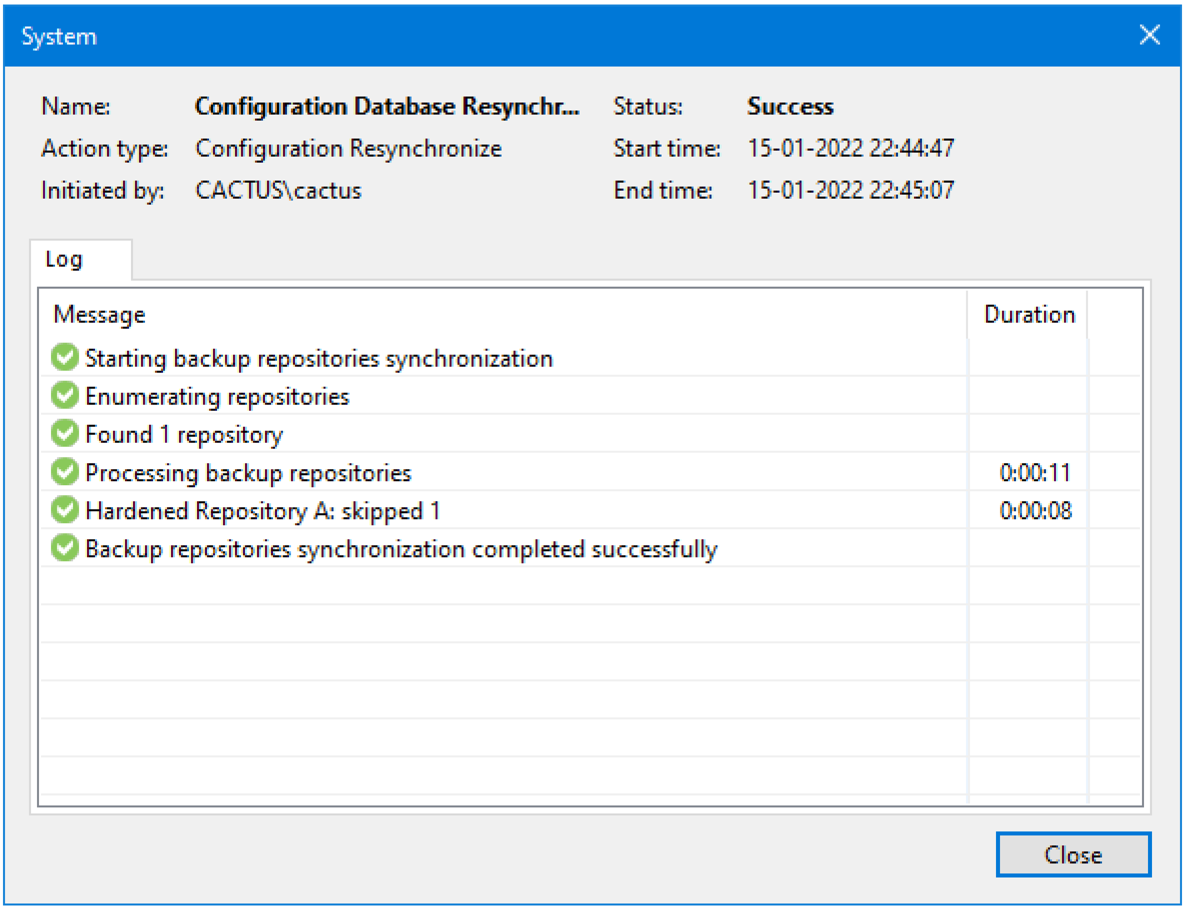
When done, you can log out from the Linux server.

Reactivate the immutable repository

Finally, go to the VBR console and navigate to the Linux server's repository, *right-click* on it, and select *Rescan*:



The connection to the repository will now be verified and resynchronised, and - after a little while where you can follow the progress - status will report *Success*:

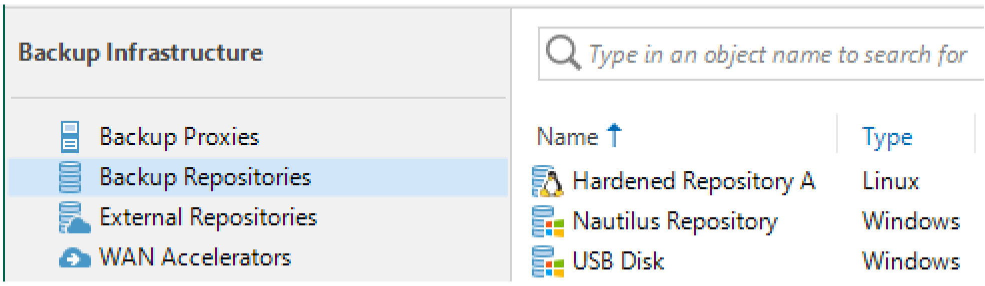


Name: **Configuration Database Resynchr...** Status: **Success**
Action type: Configuration Resynchronize Start time: 15-01-2022 22:44:47
Initiated by: CACTUS\cactus End time: 15-01-2022 22:45:07

Message	Duration
✓ Starting backup repositories synchronization	
✓ Enumerating repositories	
✓ Found 1 repository	
✓ Processing backup repositories	0:00:11
✓ Hardened Repository A: skipped 1	0:00:08
✓ Backup repositories synchronization completed successfully	

Click *Close* to close the window.

In the console window, check that the hardened repository is no longer listed as **Unavailable**:



Name ↑	Type
Hardened Repository A	Linux
Nautilus Repository	Windows
USB Disk	Windows

Conclusion

In this section, it has been shown how to deactivate and reactivate MFA/2FA authentication, for example to enable special maintenance. Also, the steps to update the *Veeam Linux Transport* has been documented in detail. Mastering these tasks should ensure, that you can keep your Linux server highly secure, while still being able to perform the traditional tasks like updating the services running on the server.

This is the last part of the series of articles that started with:

Build an immutable backup repository for Veeam Backup & Replication. Part 1