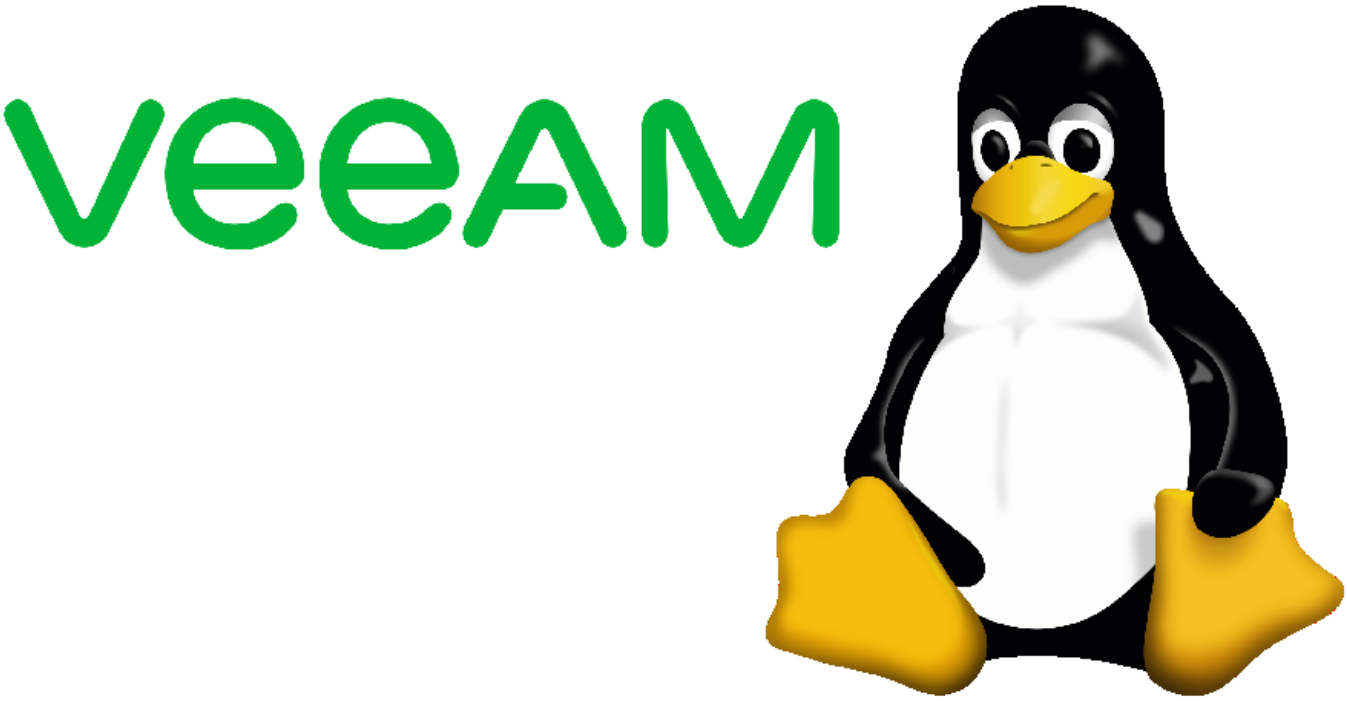
Build an immutable backup repository for Veeam Backup & Replication. Part 4



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: Create the immutable Veeam backup repository

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.

1. Prepare the install of Linux
2. Install Linux on the server
3. Prepare the Linux server for Veeam
4. Create the immutable Veeam backup repository
5. Prepare for backup of the Linux server itself
6. Backup of the Linux server itself
7. Bare Metal Recovery of the Linux server
8. Tighten security on the Linux server (MFA/2FA)
9. Maintenance and deactivation/reactivation of MFA/2FA

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*.

XFS and the virtual air gap

The XFS file system was introduced by SGI in 1993 for its IRIX 5.0 operation system which was based on UNIX System V Release 4.

XFS was ported to Linux in 2001. As SGI ceased operations in 2009, Linux is today the only operating system supporting XFS.

Why is this important? Because XFS is the only file system offering immutability:

> Once the file is set immutable, this file is impervious to change for any user. Even the root cannot modify, remove, overwrite, move or rename the file. You will need to unset the immutable attribute before you can tamper with the file again.

For the details about handling this, study *Dan Nanni*'s blog on Xmodulo: How to make a file immutable on Linux.

Applying immutability to your backup files hosted on a physical server introduces a virtual *air gap* in your backup chain, protecting the backup files from anything else than direct physical access. This way, the backup files will be protected from any attack caused by advanced malware or possible hackers.

The effect is the same as if you back up to tape or DVD and, when done, remove the media from its drive.

Part 4. Create the immutable Veeam backup repository

In this section we will create the immutable backup repository for Veeam on the Linux server we prepared for this in Part 3.

Elevating the Veeam user account

Veeam Backup & Replication (which always runs on a Windows server) will install the *Veeam Linux Agent* on the Linux server to be able to communicate with this.

As the installation of the Veeam agent on the Linux server requires administrator rights, and we for security reasons do not wish to store the password (even when encrypted, as it would be) for a Linux administrator in the Veeam database (on the Windows server), we temporarily must elevate the *veeamuser* account we created in Part 2.

So, open PowerShell and connect via SSH to the Linux server using this command where *hostname* is the hostname or the IP-address of the Linux server:

```
ssh linuxadmin@hostname
```

Then call this command to assign our user *administrator rights* by signing the user in to the *sudo* group:

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

As a result, for this session or until cancelled, user *veeamuser* has been assigned administrator rights.

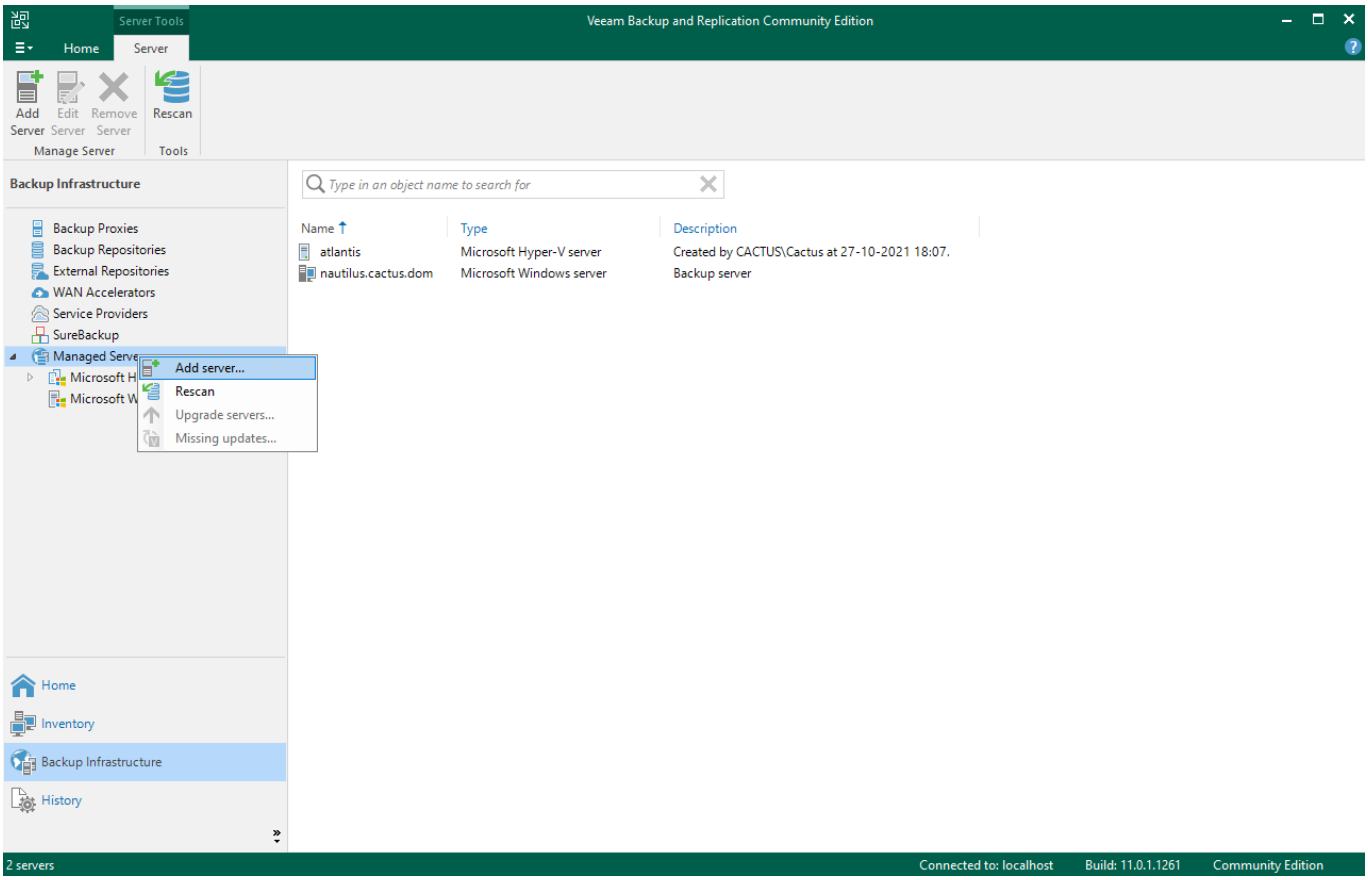
Installing the Veeam Linux Transport

Now is the time to match Veeam and the Linux server.

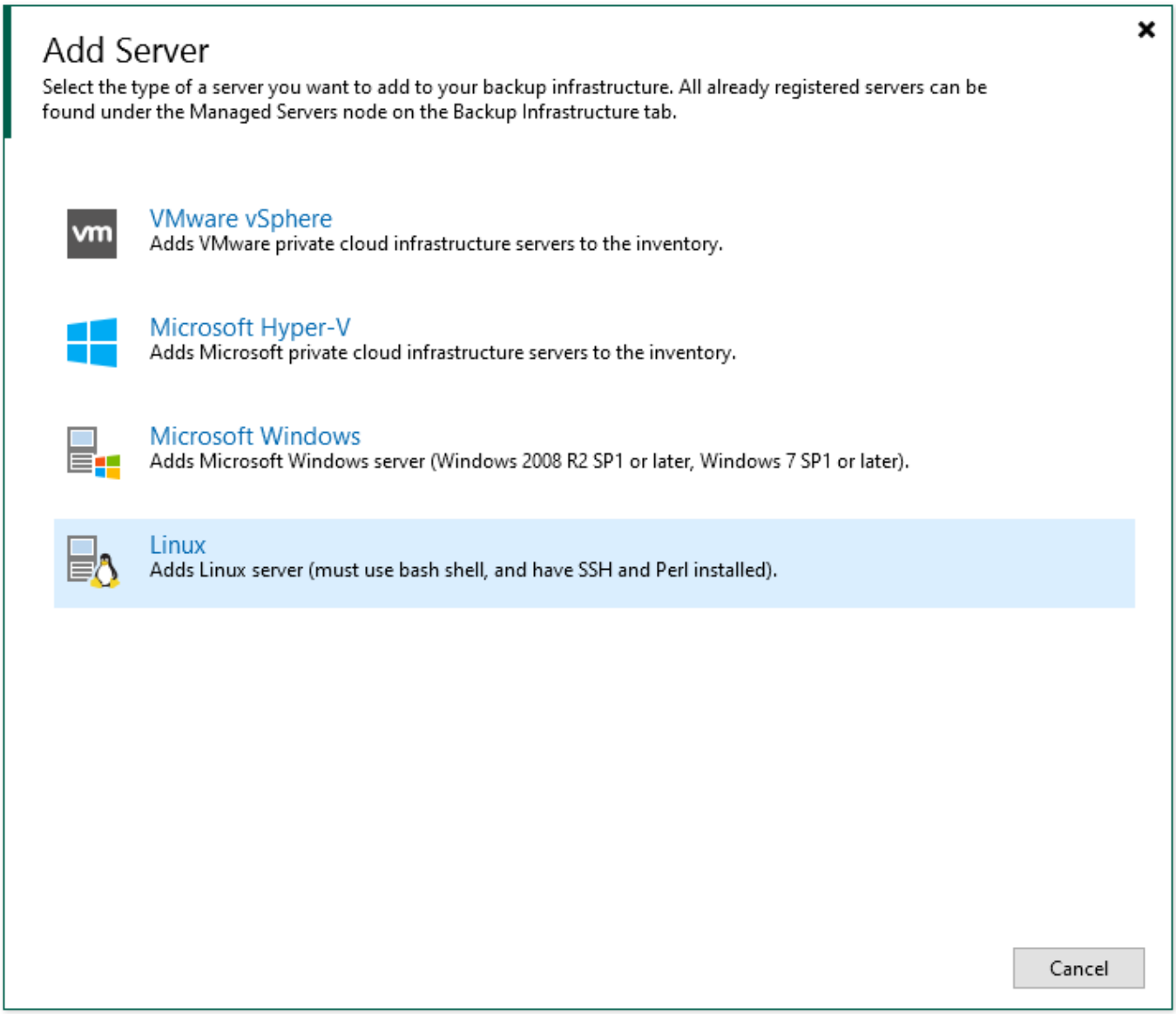
Open your *Veeam Backup & Replication Console* and navigate to:

- Backup Infrastructure
    - Managed Servers

and click *Add server ...*:

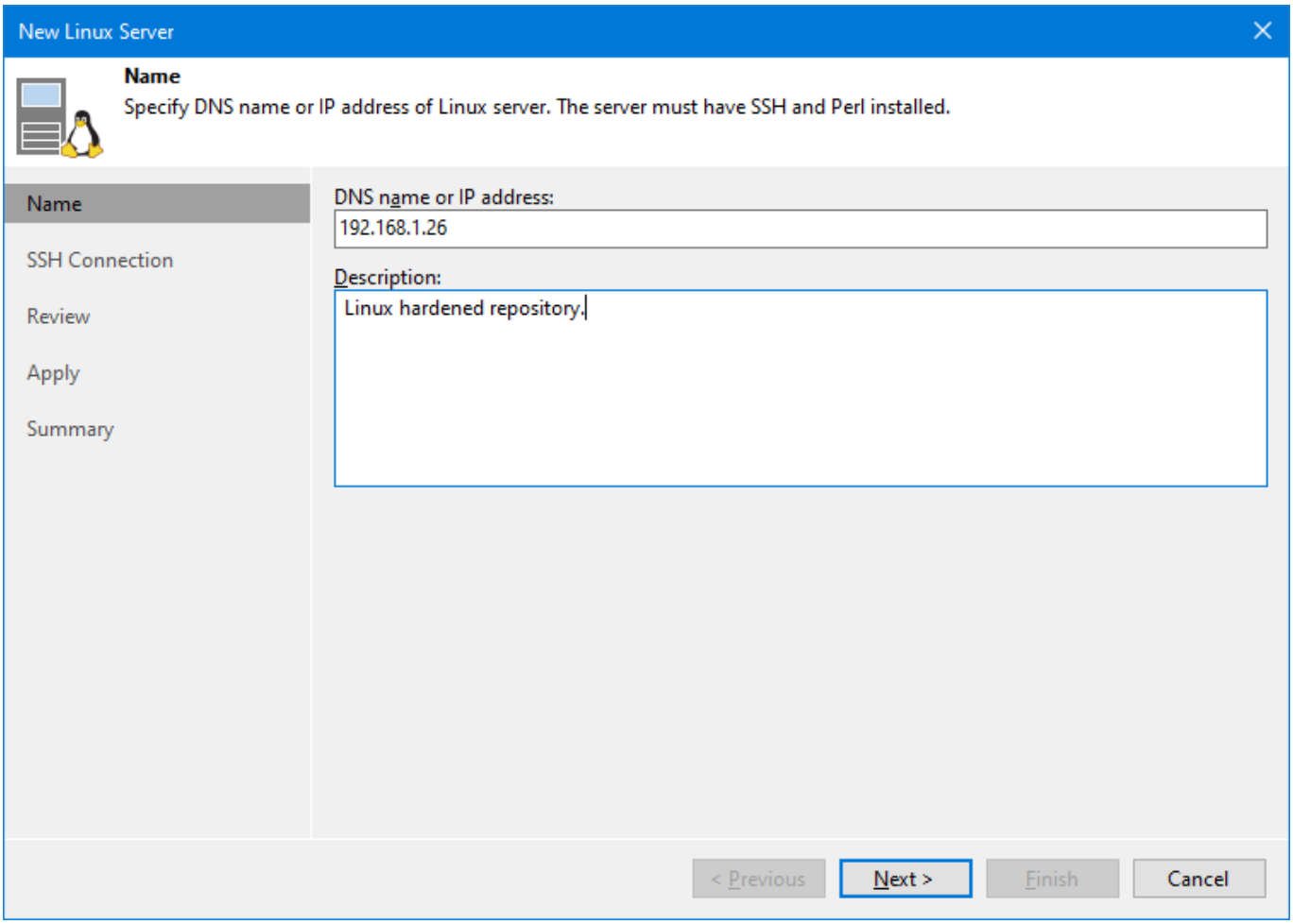


Select *Linux*:



The *New Linux Server* wizard opens.

Enter the hostname or IP address of the Linux server and a meaningful description:

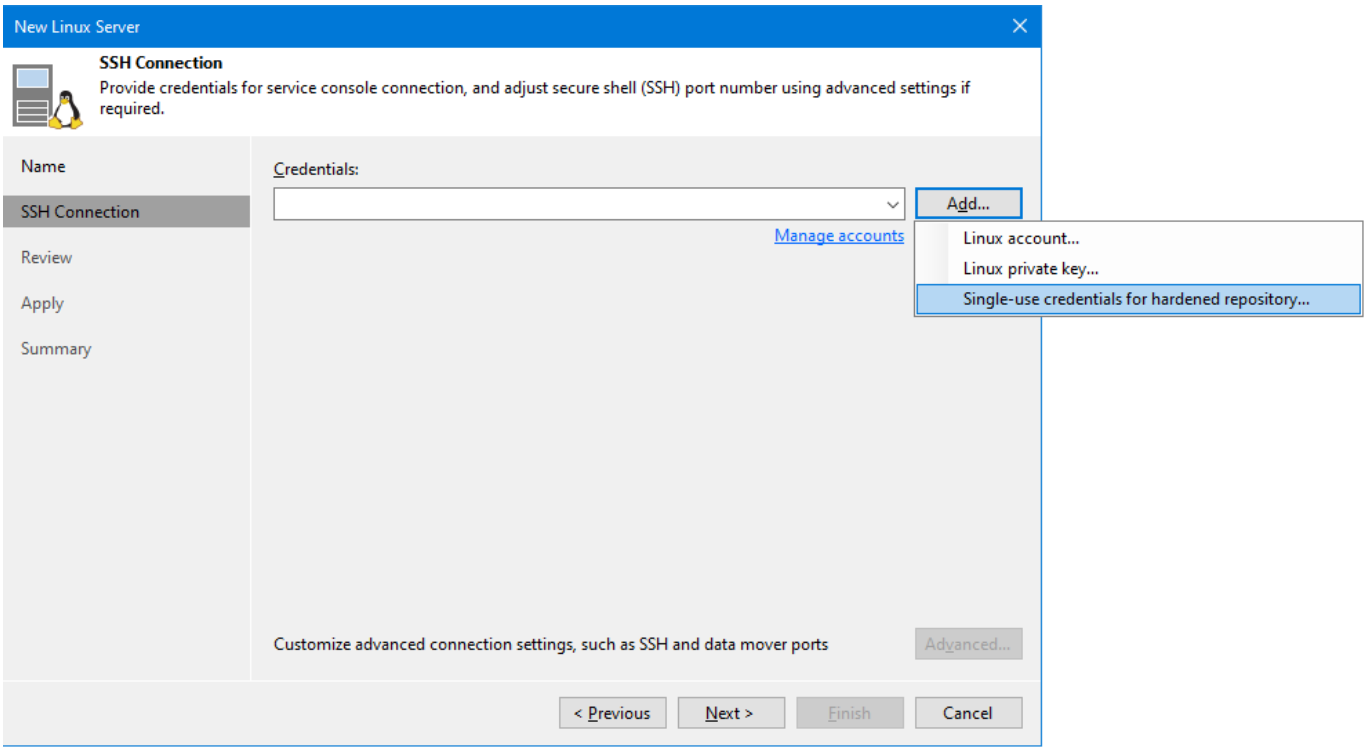


Click *Next*.

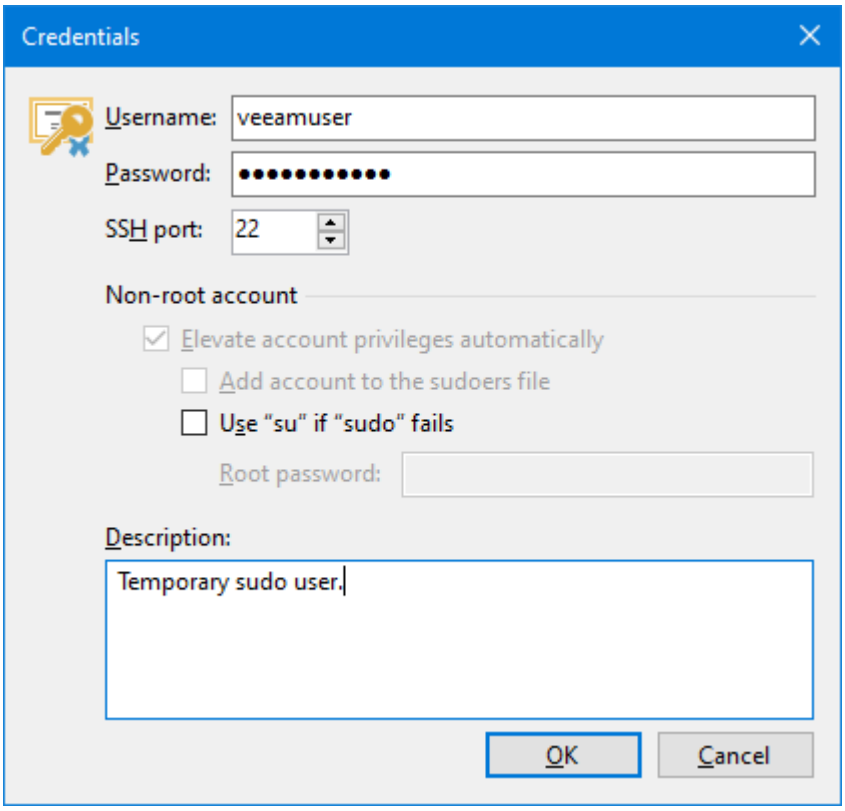
Now comes an important step. Veeam can *use a set if credentials for this session only, without storing it anywhere*. This means, that even if a hacker or some malware should gain access to the Veeam database and break the encryption, nothing will be found about admission to the hardened Linux repository.

So, click *Add* .. and then select:

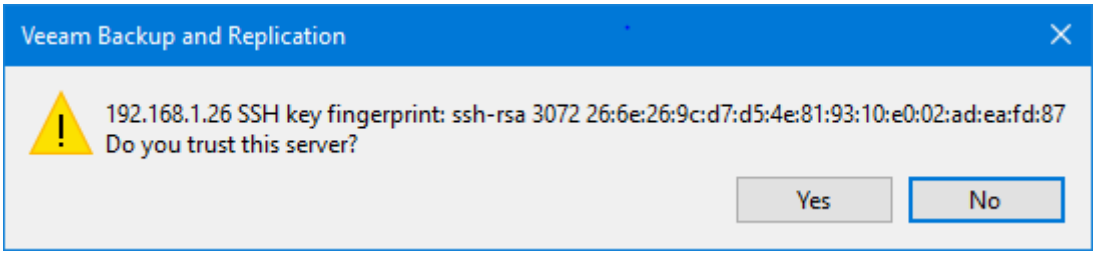
- Single-use credentials for hardened repository ...



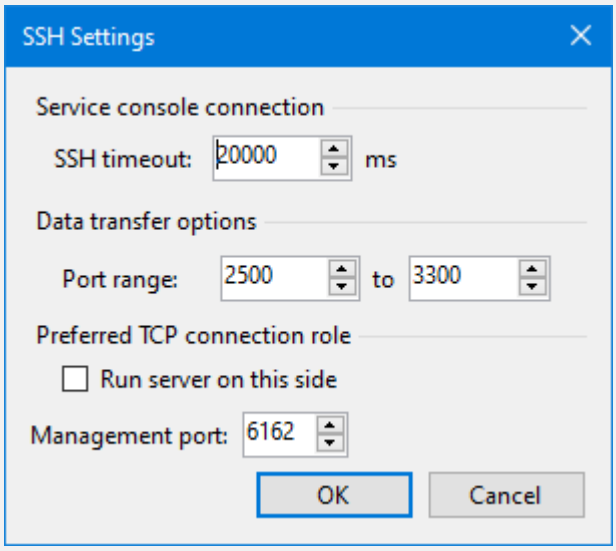
Then enter the credentials for our *veeamuser* account:



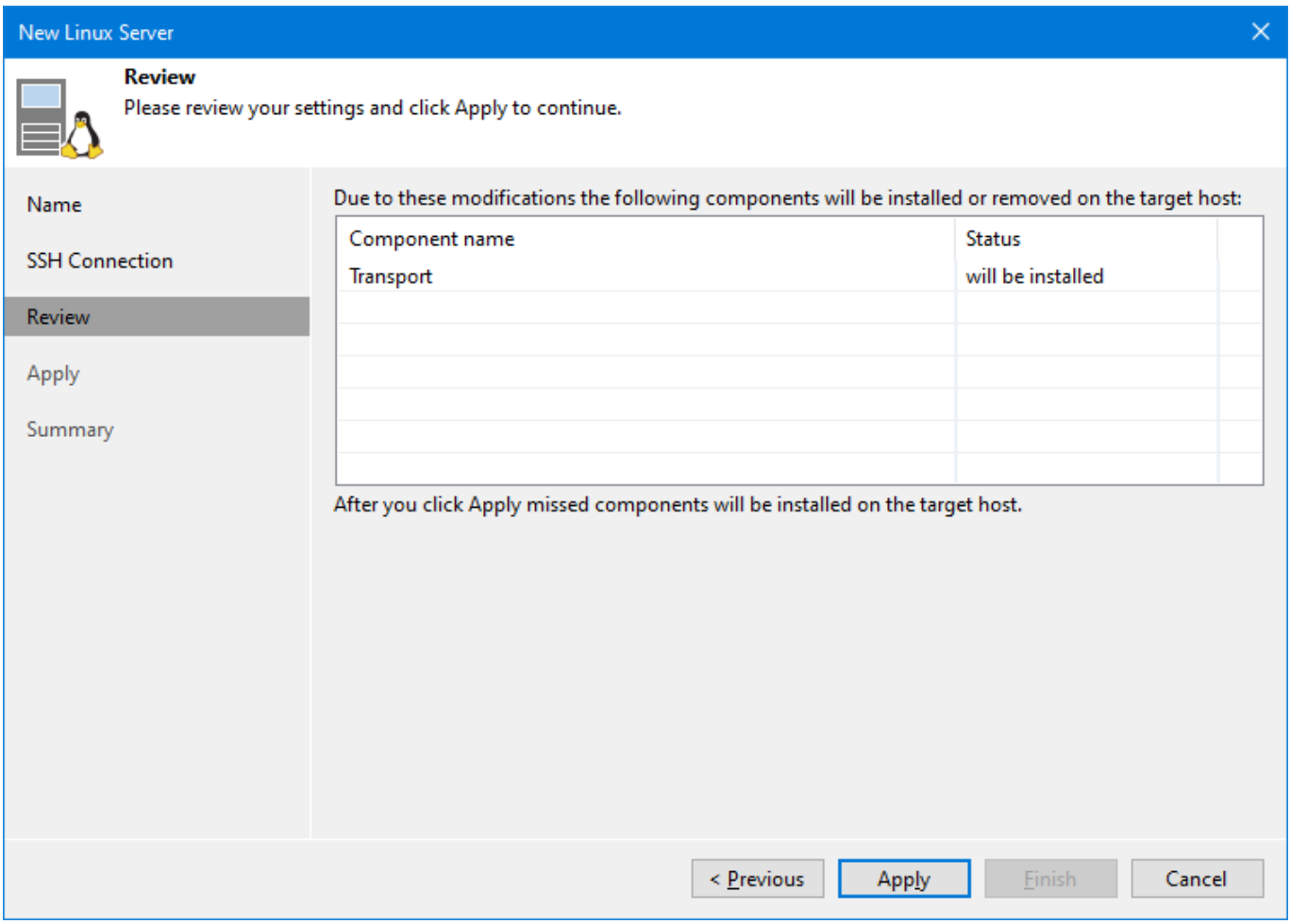
Click *OK*, and you will be prompted for a confirmation of the connection:



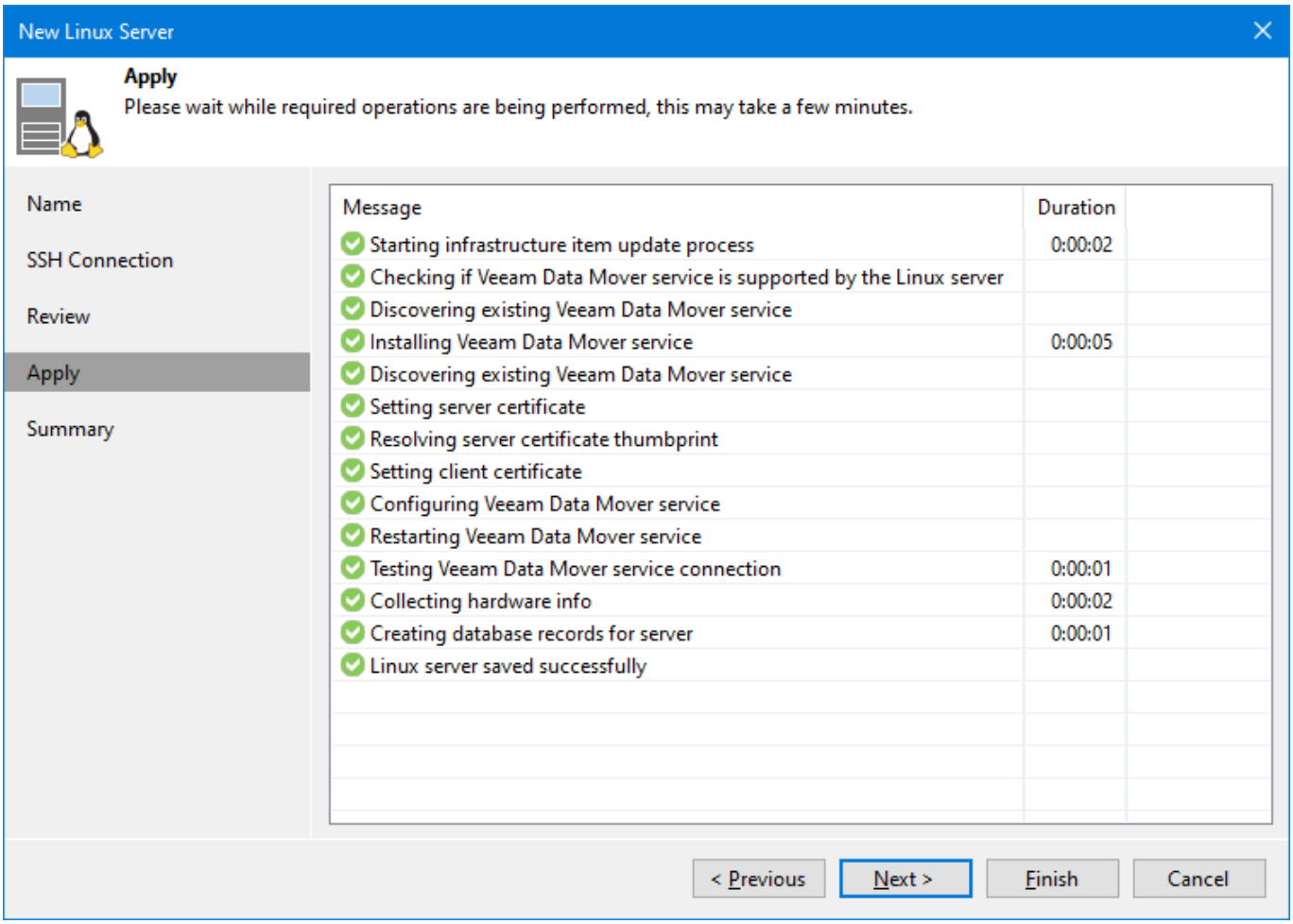
Click *Yes*

> At this point, click *Advanced* if you wish to adjust the *SSH Settings*. Most likely, the defaults will be fine:
>
> 
>

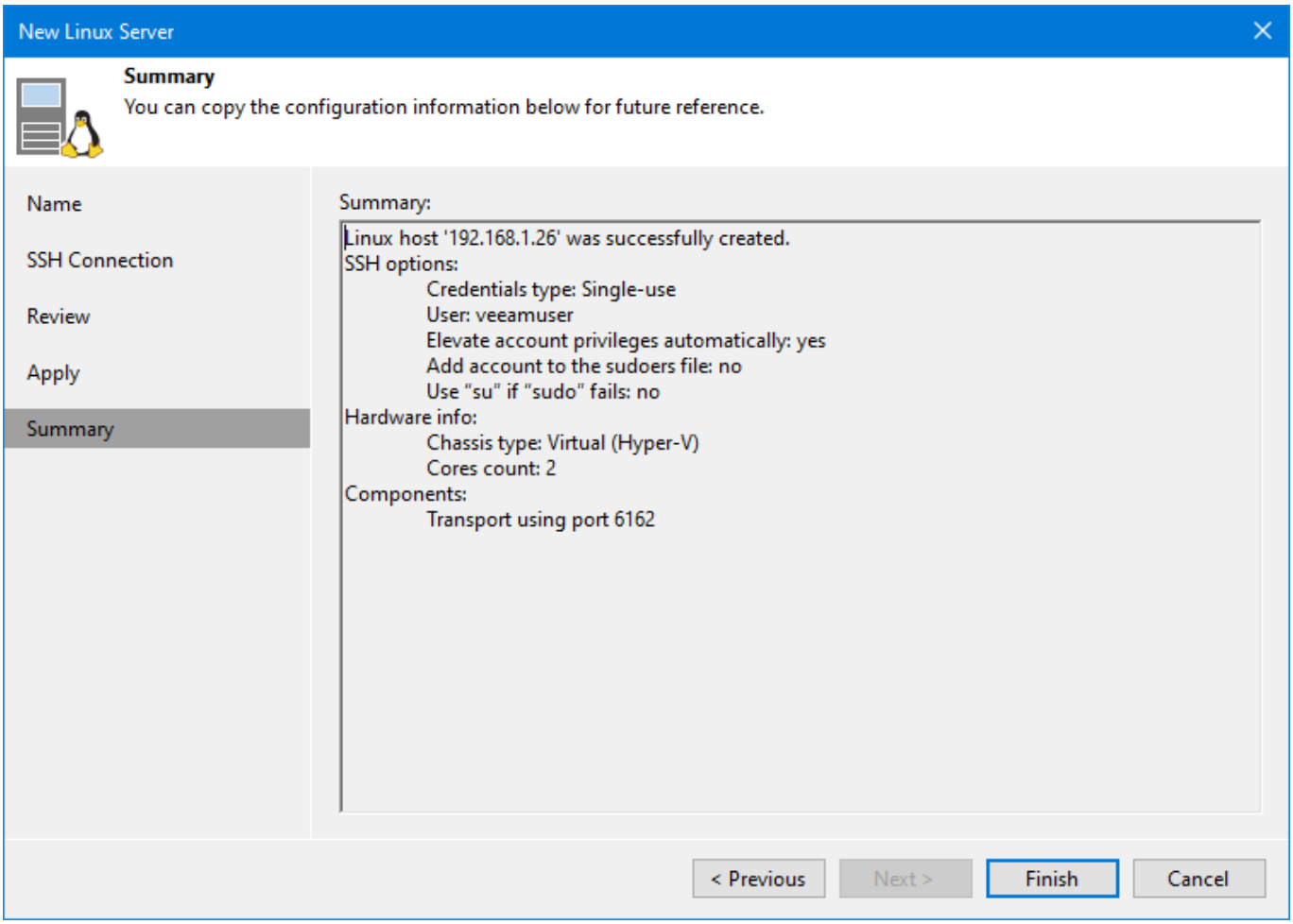
In the wizard, click *Next* and browse the *Review* pane:



Click *Apply* and, after a little while, the Linux will have had the Veeam components installed:



Click *Next* to view the *Summary*:



Click *Finish* to close the wizard.

The Linux server will now be listed under:

- Managed Servers
  - Linux

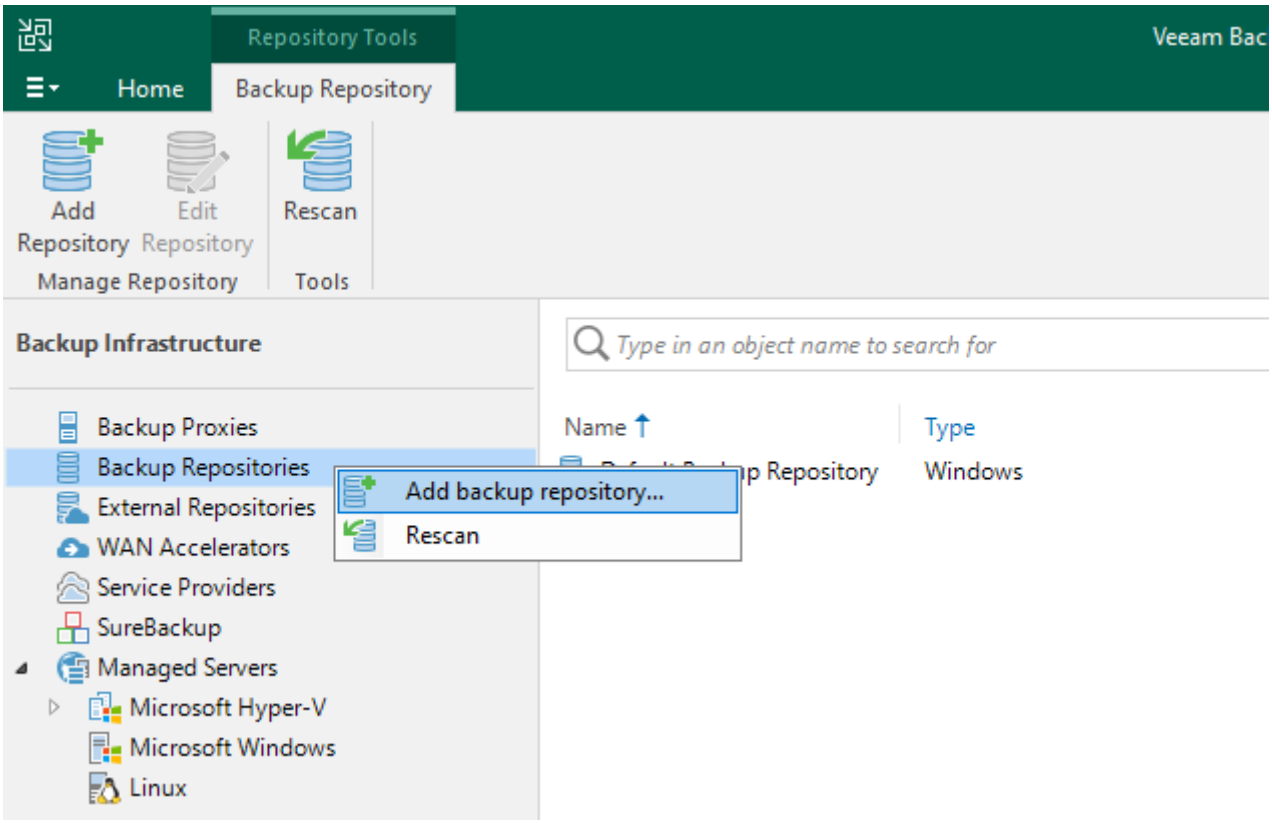
Set up the hardened repository

Veeam Backup & Replication is now aware of the Linux server, which makes it possible to add a hardened repository hosted on this server to the backup infrastructure.

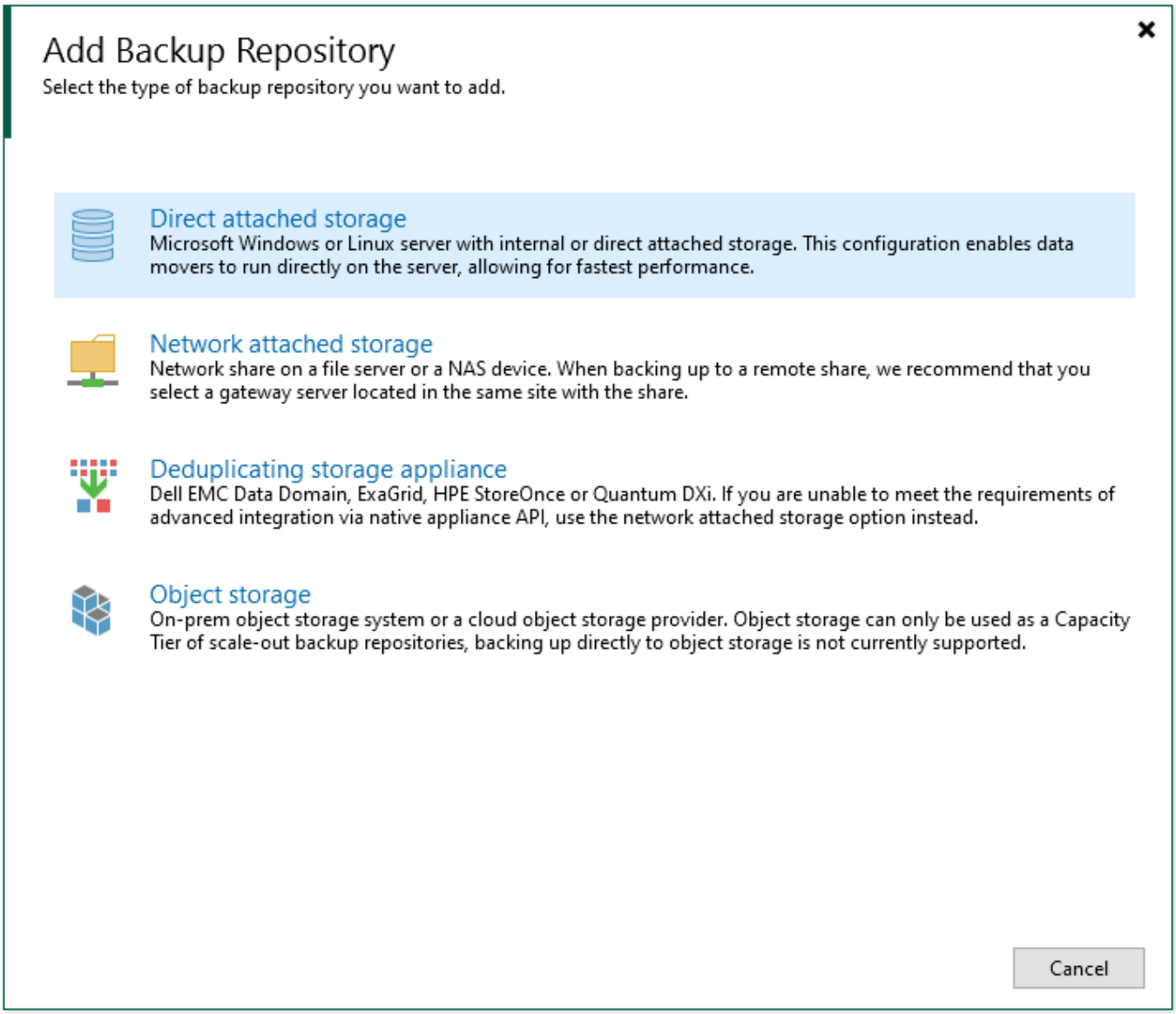
Navigate to:

- Backup Infrastructure
  - Backup Repositories

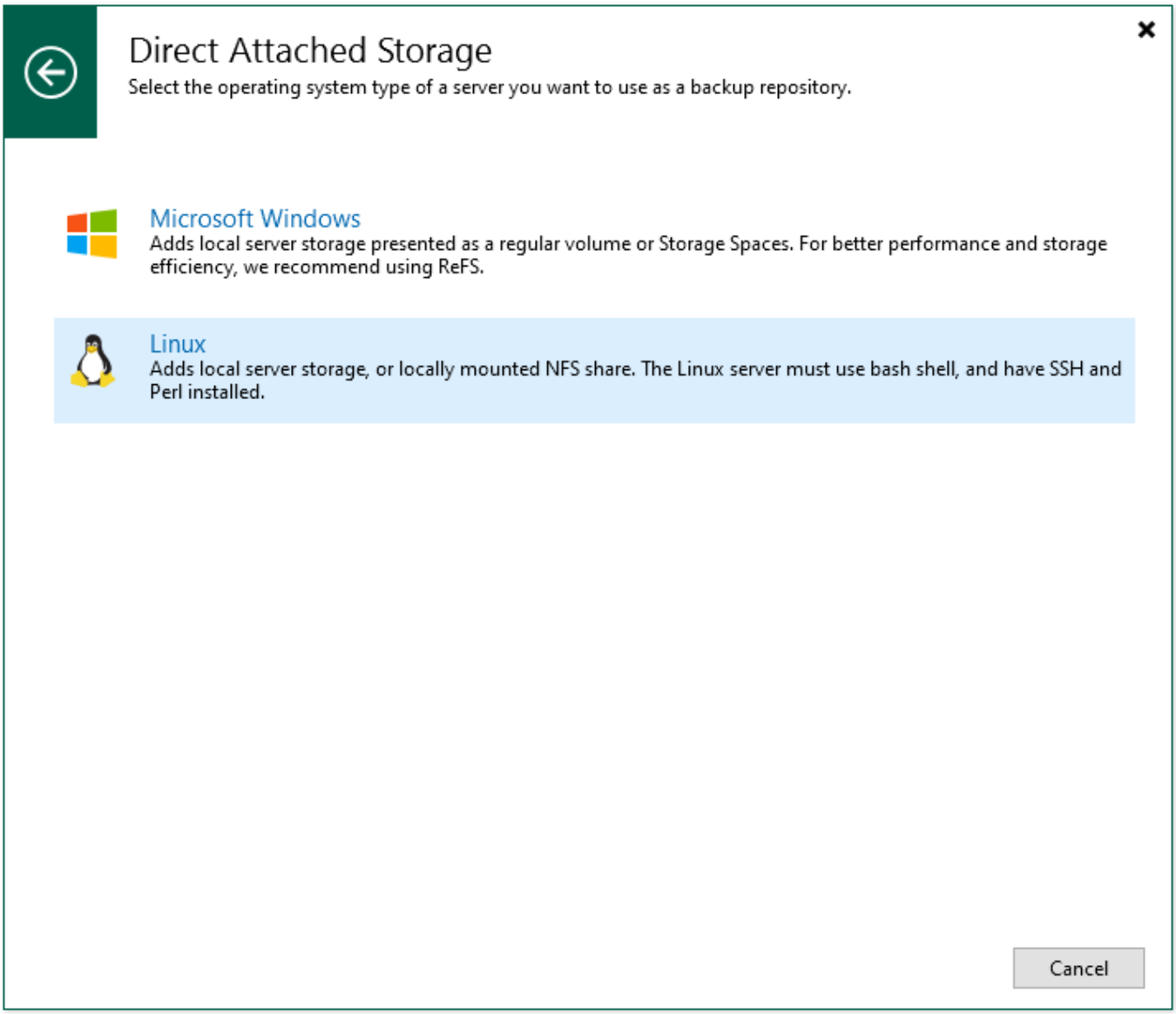
Right-click *Backup Repositories* and select *Add backup repository ...*:



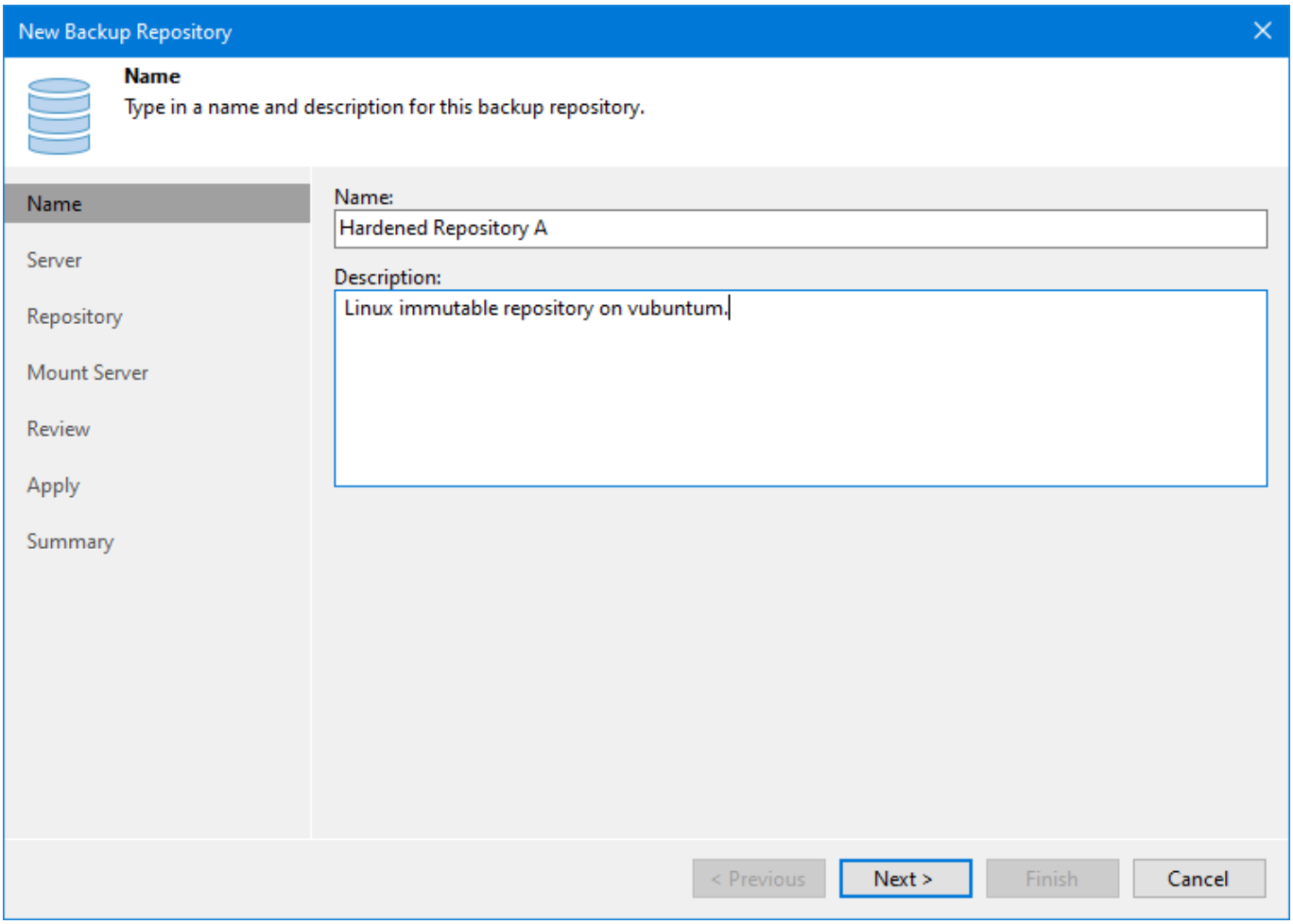
Add Backup Repository opens:



Click *Direct attached storage*, and you can select the server type:

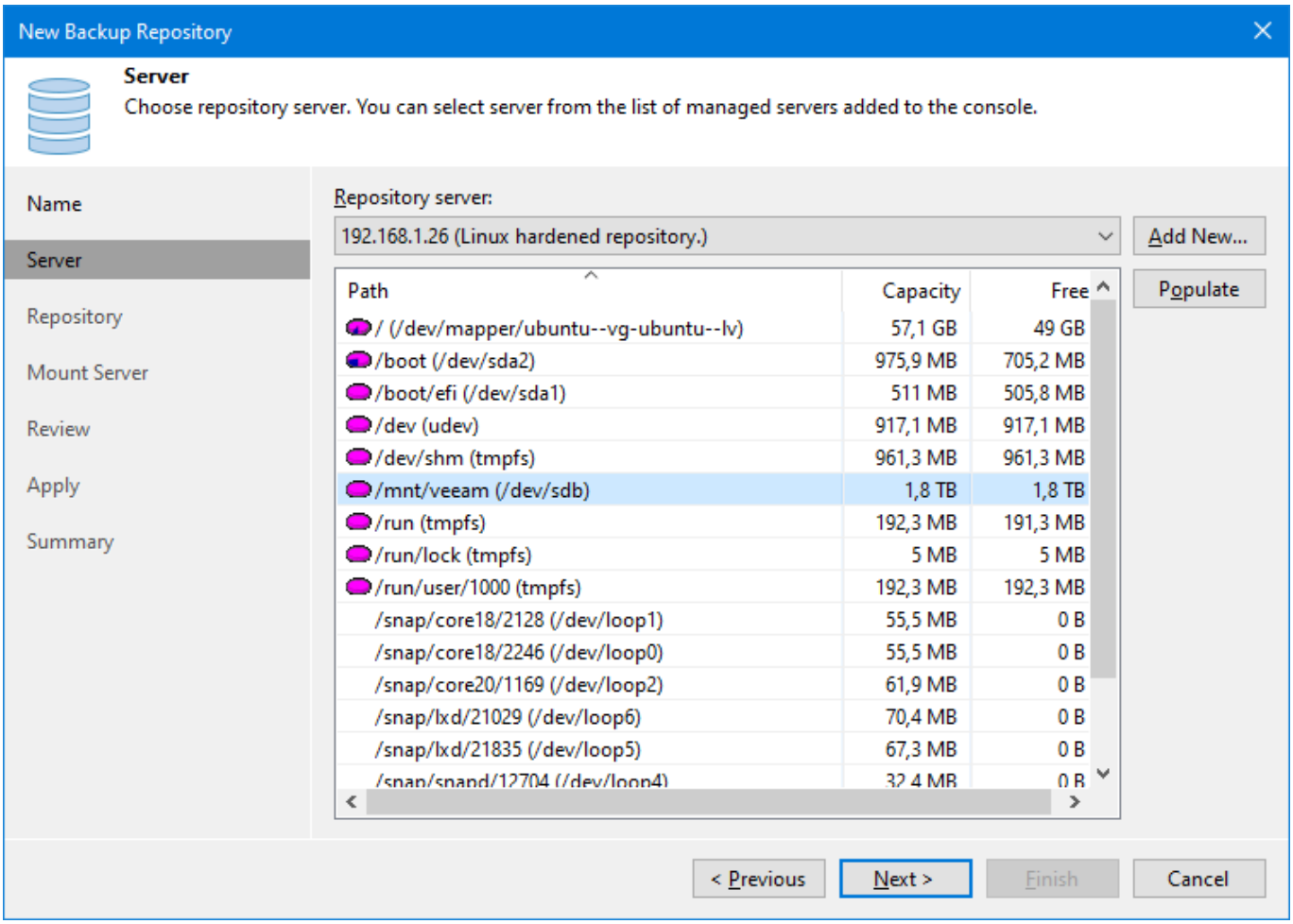


Click *Linux* to open the wizard. Assign a meaningful *Name* and *Description* to the new backup repository:



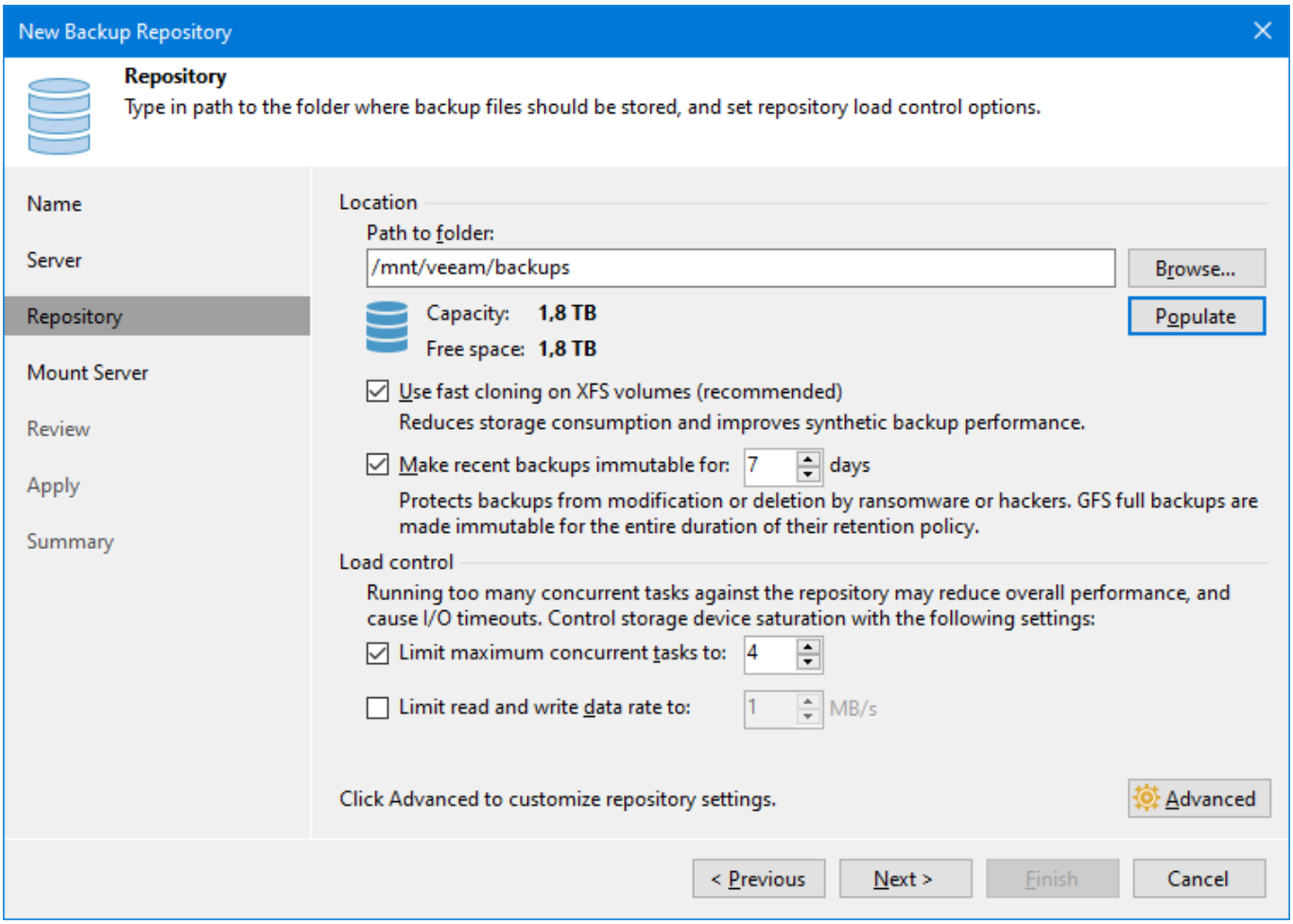
Click *Next* and, for the *Repository server*, select the server we added above.

Then click *Populate* to list the available paths at the server, and mark */mnt/veeam*:



Click *Next* and the server will be connected using the */mnt/veeam* mounting point. By default, Veeam here creates a folder */backups*.

Click *Populate*, and the values for *Capacity* and *Free space* should change from to the current values, here **1.8 TB**:



Important

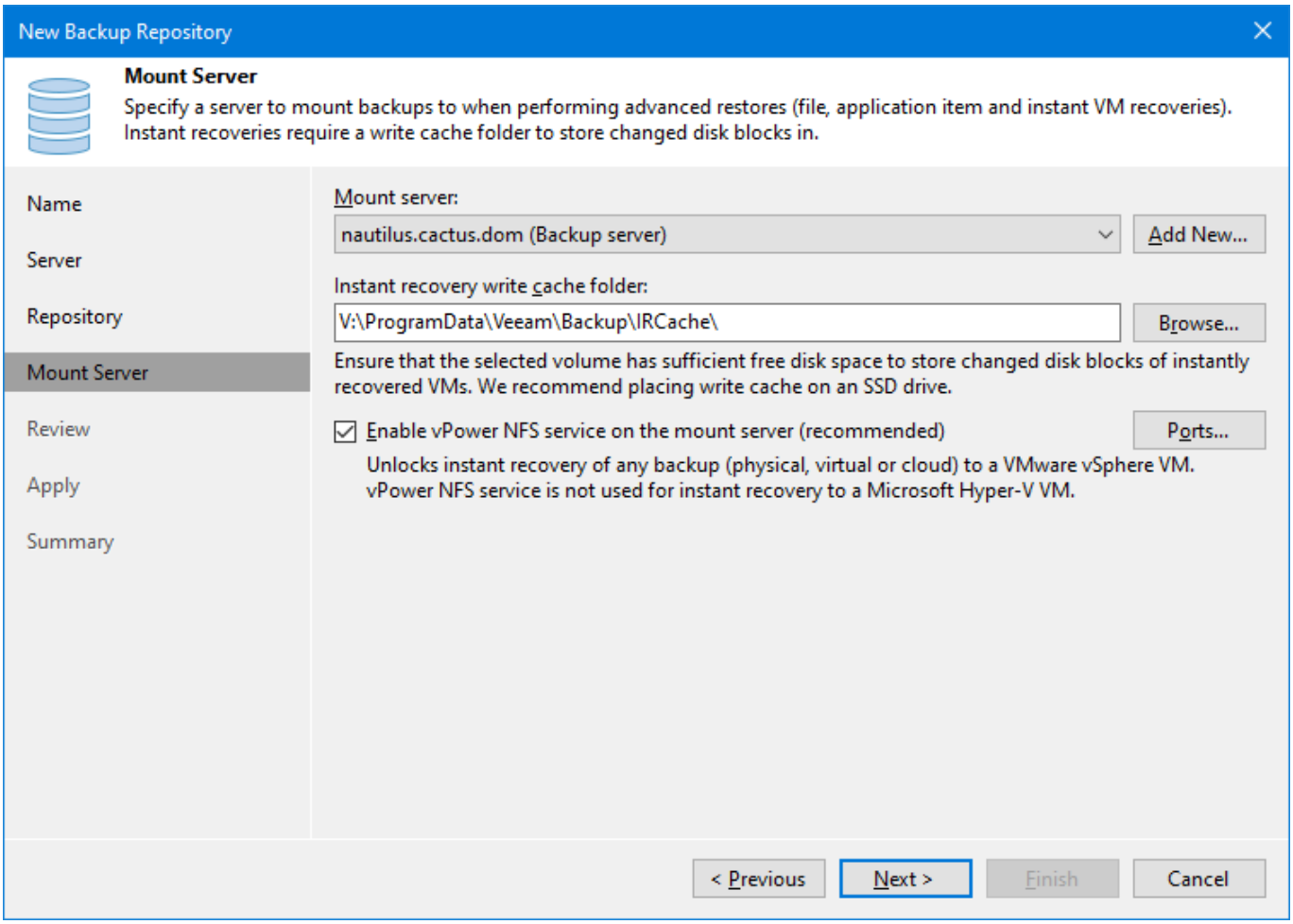
Do mark the two options (as shown above):

- Use fast cloning on XFS volumes (recommended)
- Make recent backups immutable for 7 days

Here, **7** is the default and the minimum value. You may wish to use a longer period, though that may consume a lot of space.

The settings under *Advanced* should be left as is.

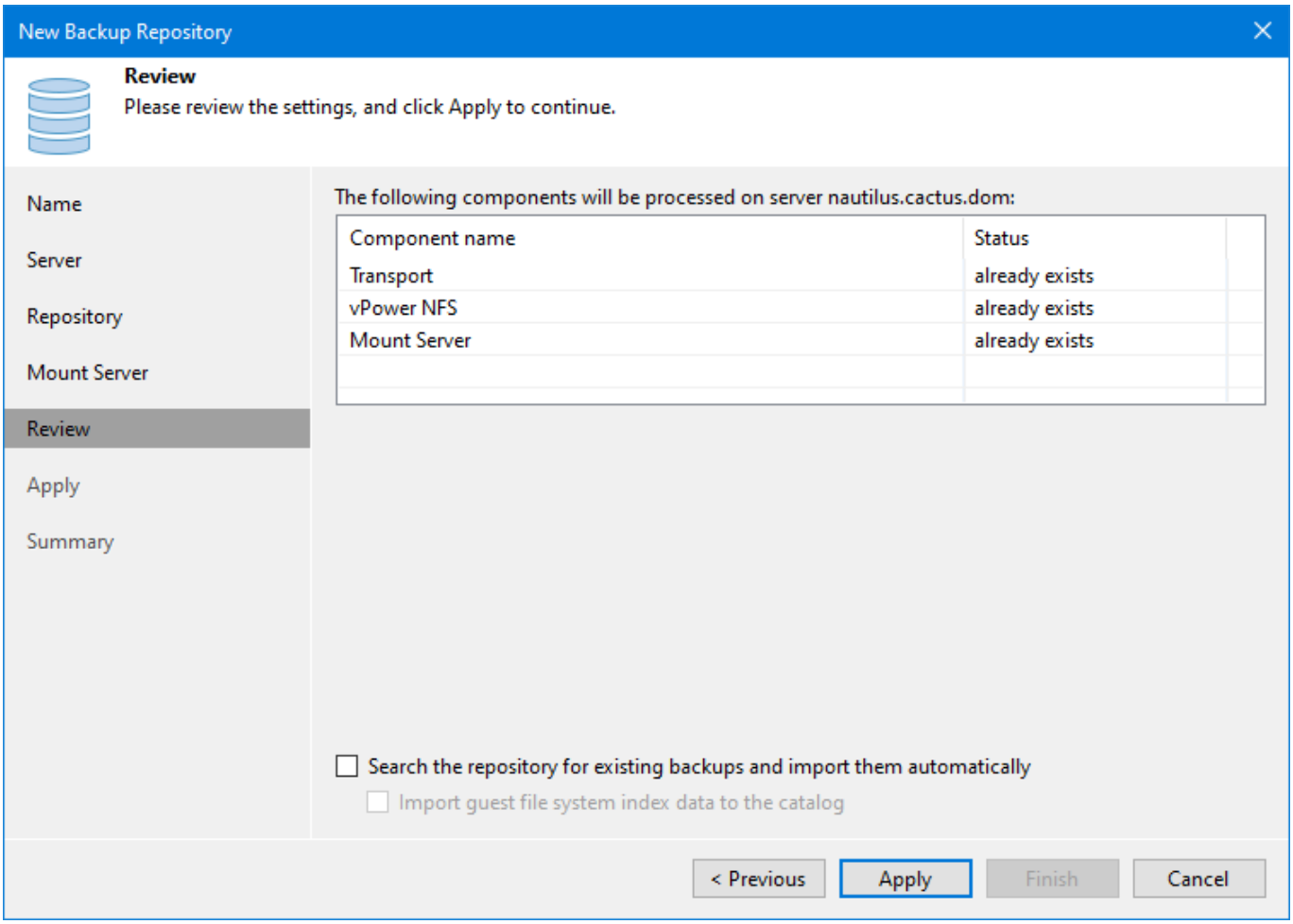
Click *Next*, and Veeam will check if the XFS conditions are met. They should be, and the wizard will ask for *a server to mount backups to when performing advanced restores*:



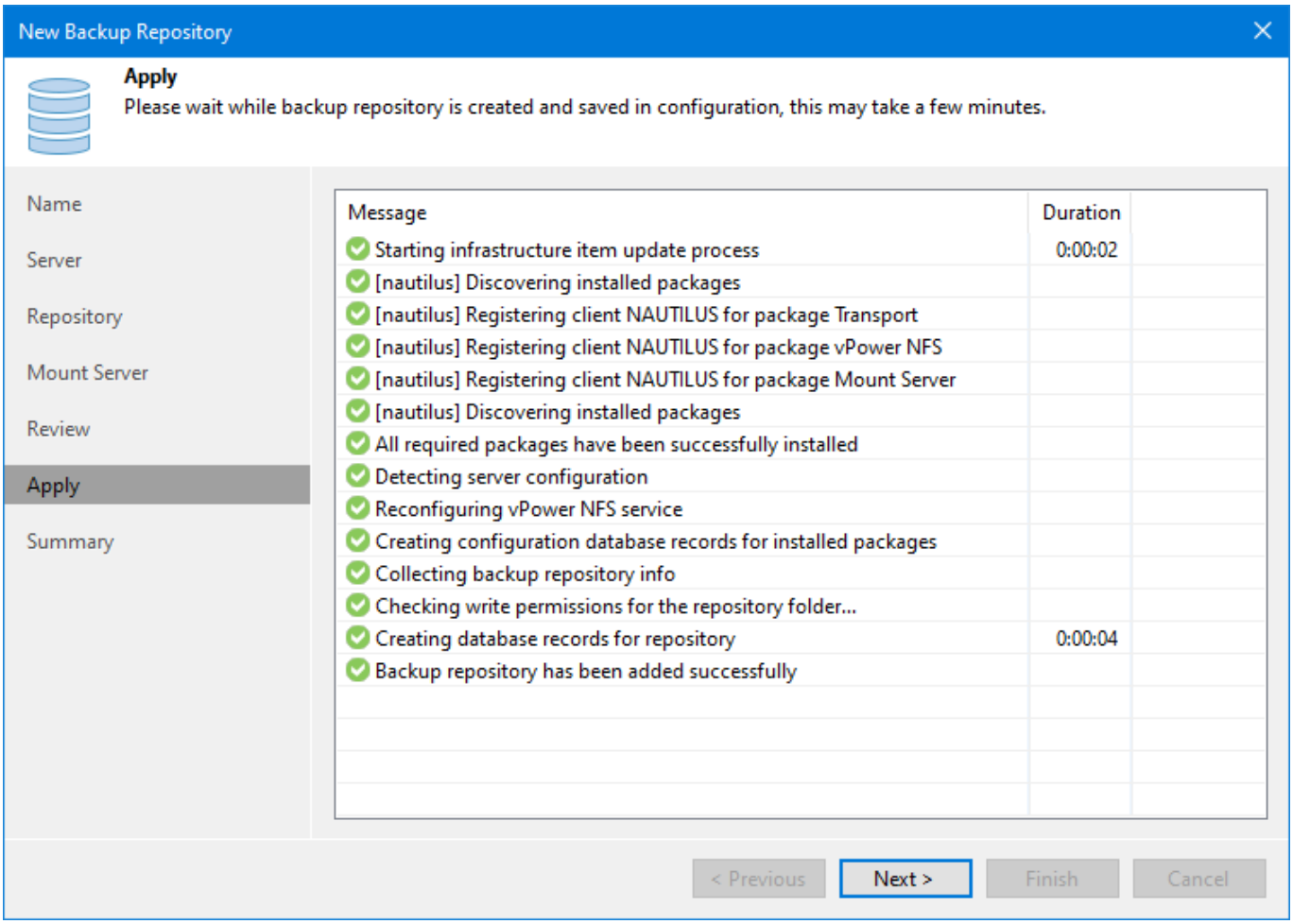
Leave the default values.

> If you don't have VMware running, you can clear:
>
> *Enable vPower NFS service on the mount server*.

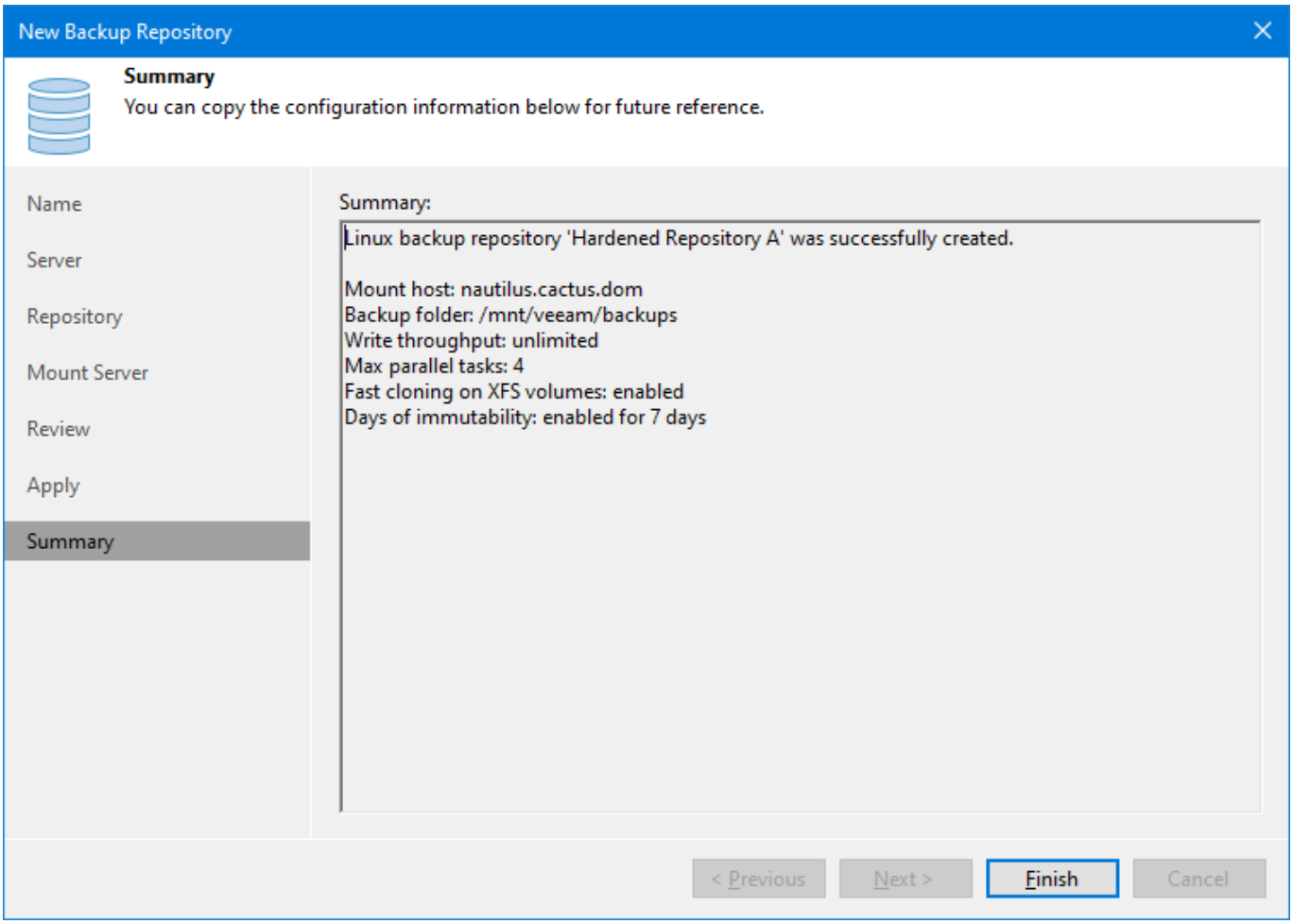
Click *Next* to go to the *Review* pane listing the components needed:



Click *Apply*, and the install will run and finish shortly after:

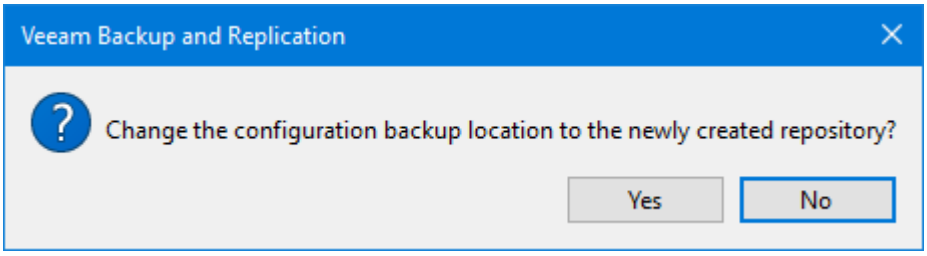


Click *Next* to view the summary:



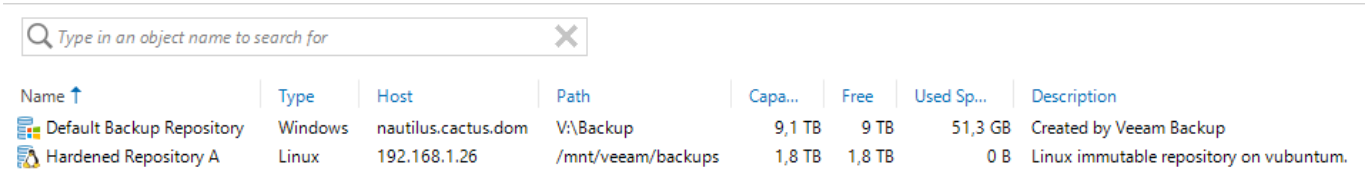
Click *Finish* to close the wizard.

A message box may pop up:



Click *No*.

Verify, that the hardened repository is listed among the other backup repositories you may have:



The hardened repository is now ready.

Demote the Veeam user account

This is an important step.

Remember, that we initially elevated the Veeam user account to allow it to install the Veeam components on the Linux server?

As these now have been installed, it is time to demote the account. Use this command to carry that out by removing the user account from the sudo group (*veeamuser* is the name of the Veeam user account):

```
sudo deluser veeamuser sudo
```

The command may indicate, that it will delete the account, but that is not the case; it will only *remove* it from the sudo group.

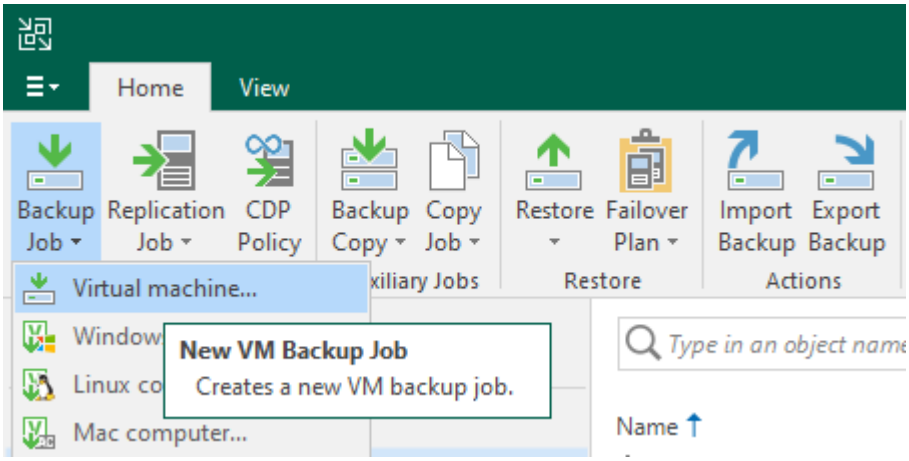
The response will be:

```
Removing user `veeamuser' from group `sudo' ...
Done.
```

Check immutability

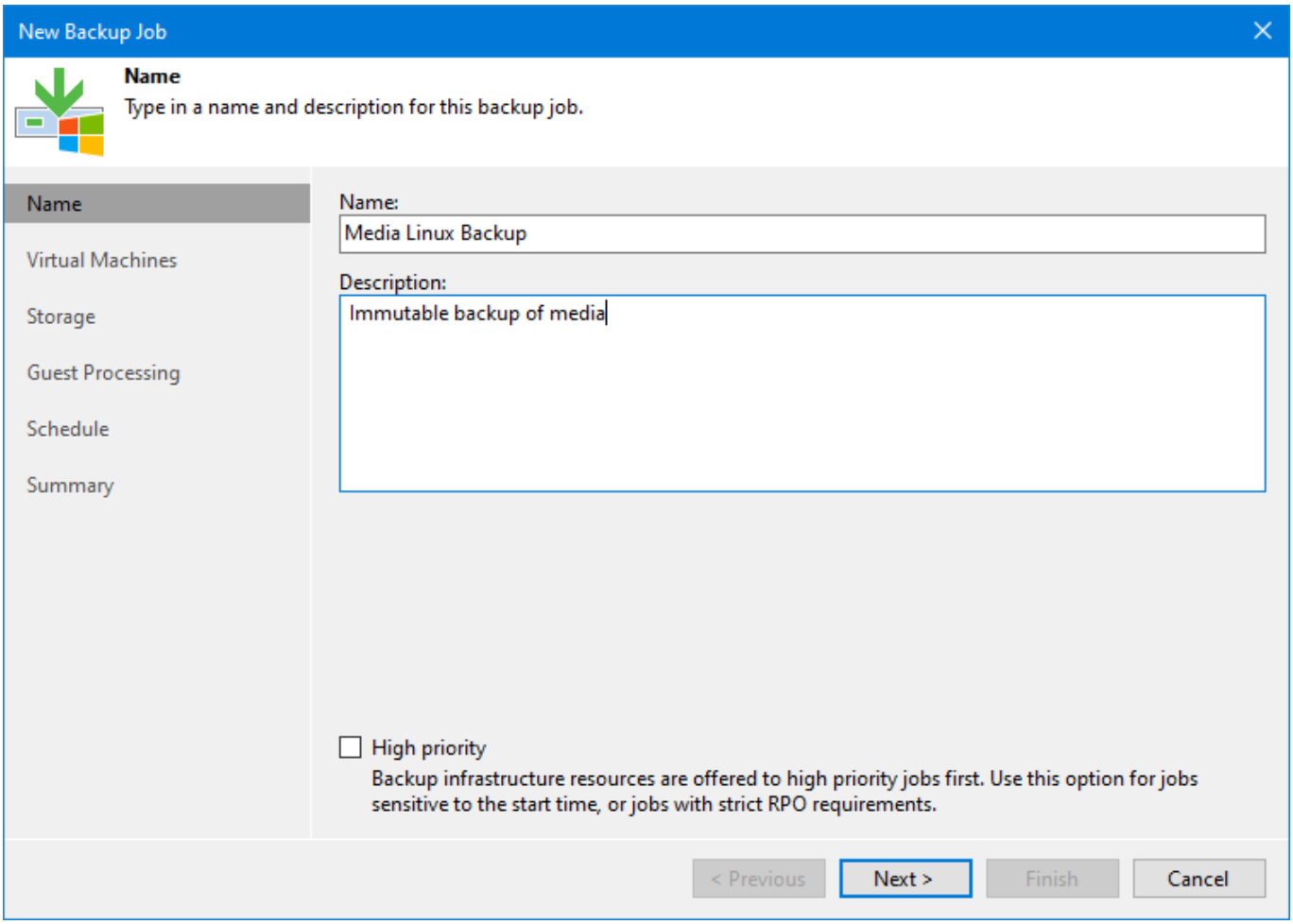
Let's create a small backup job to demonstrate the immutability.

In Veeam B&R Console, navigate to *Home* and click *Backup Job* in the band:

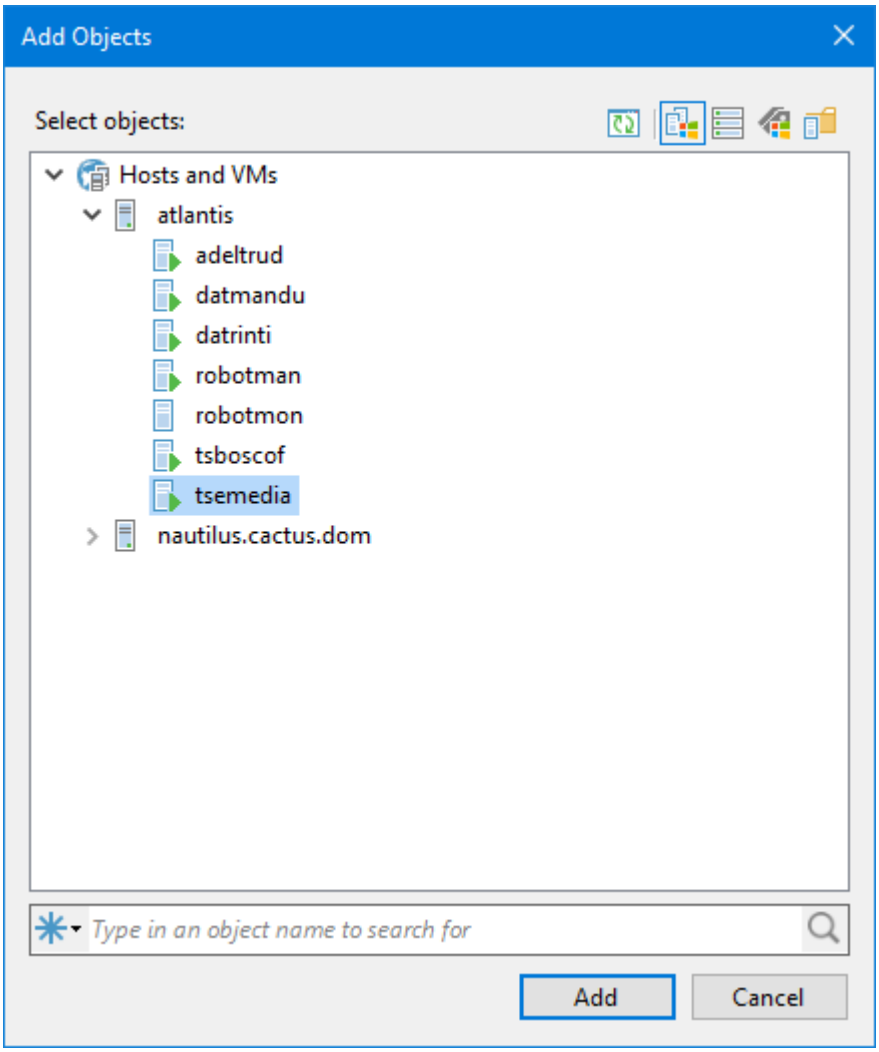


Select and click *Virtual machine ...* to open the wizard that will help you to configure the backup job.

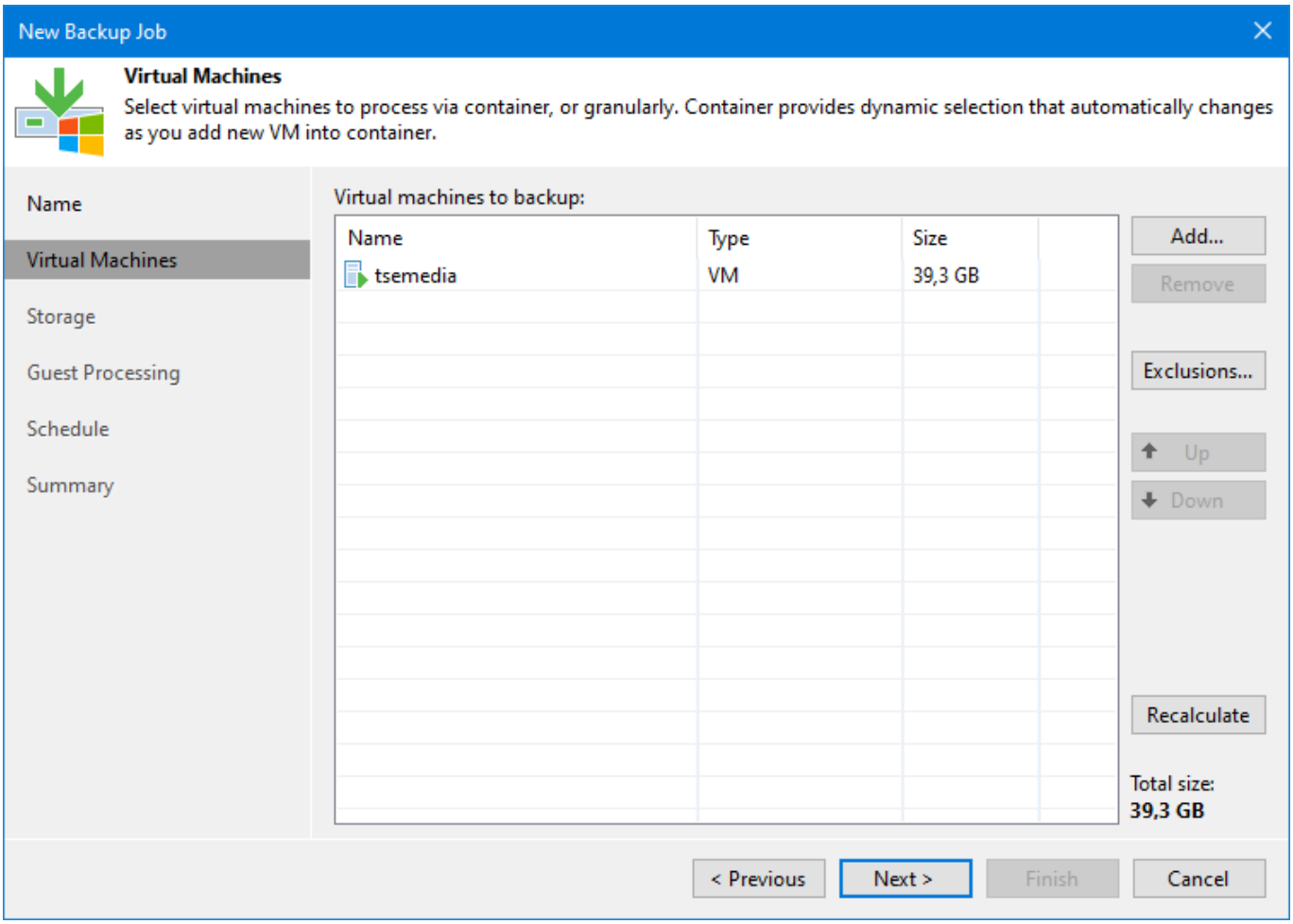
Enter a meaningful *Name* and *Description* for the job:



Click *Next* and, in the *Virtual Machines* pane, click *Add* to select a virtual machine to backup:

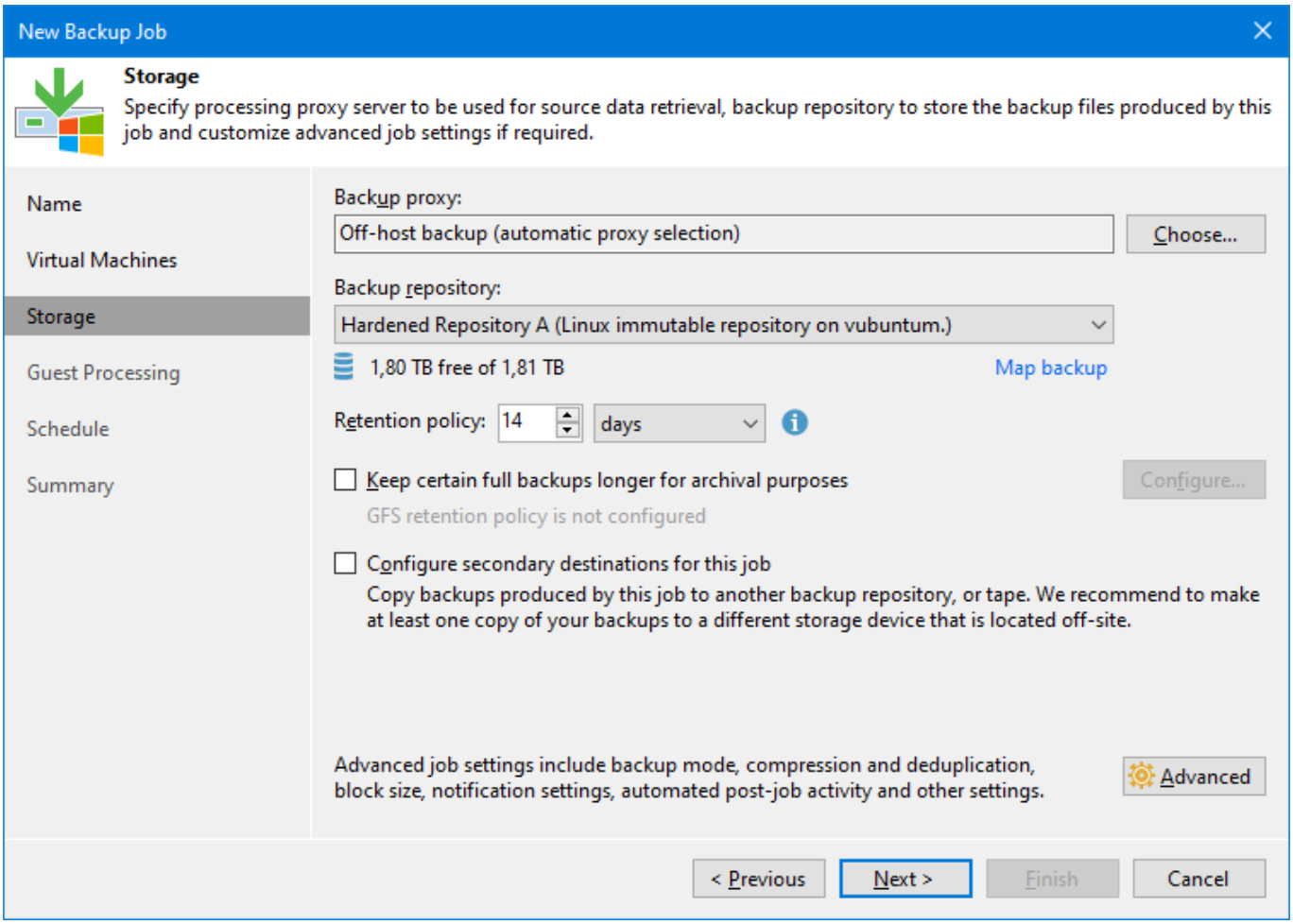


Select a machine and click *Add*. The machine will be listed for backup:



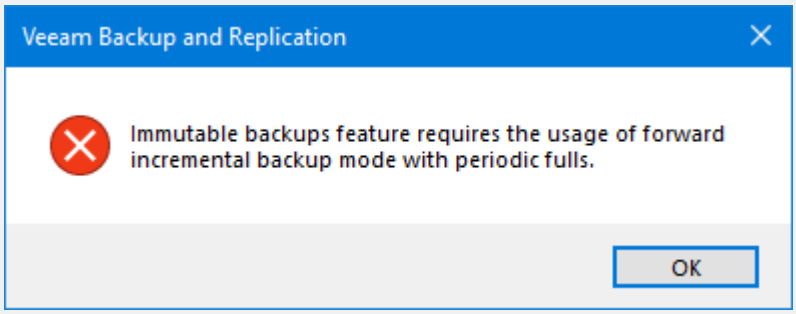
Click *Next* to proceed to the *Storage* pane.

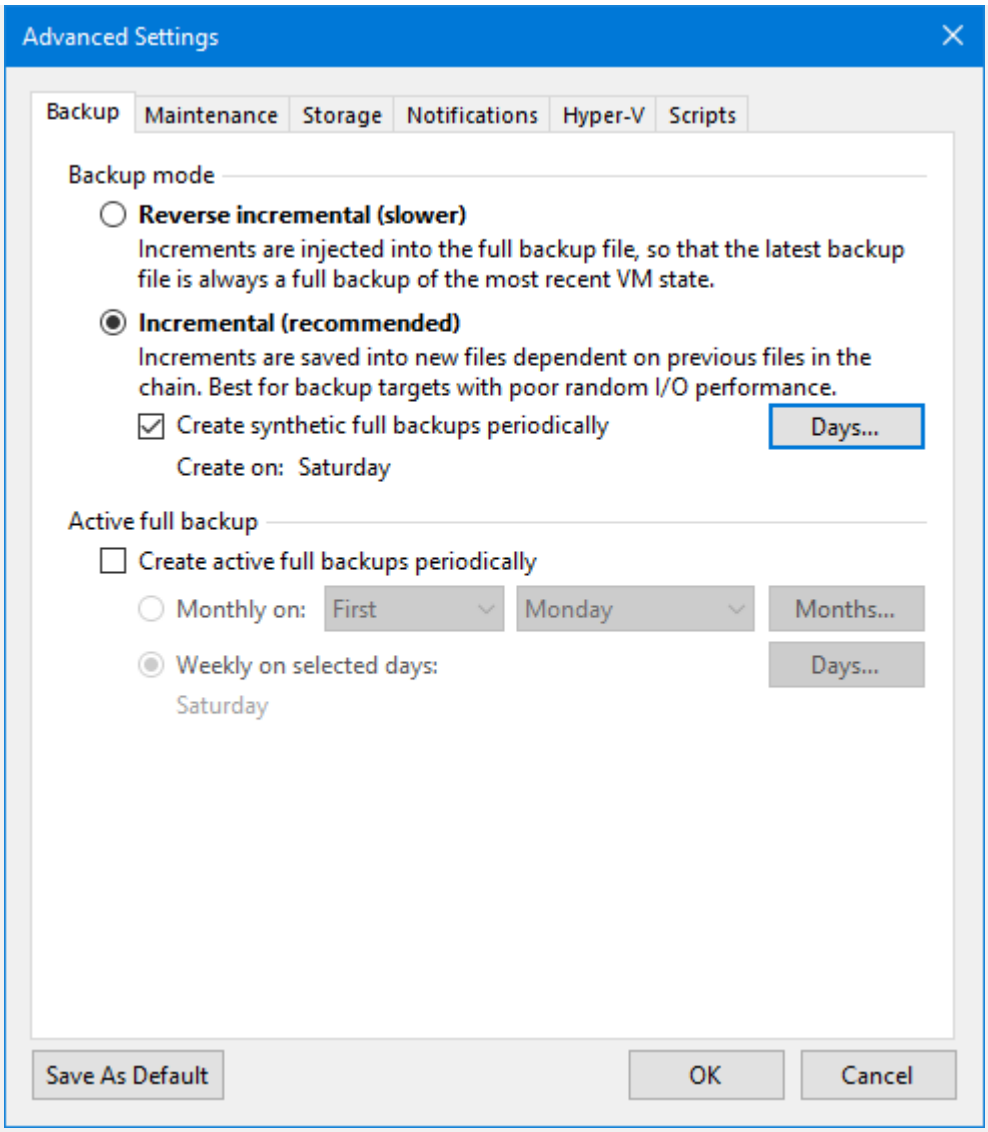
For *Backup repository*, select the hardened repository:



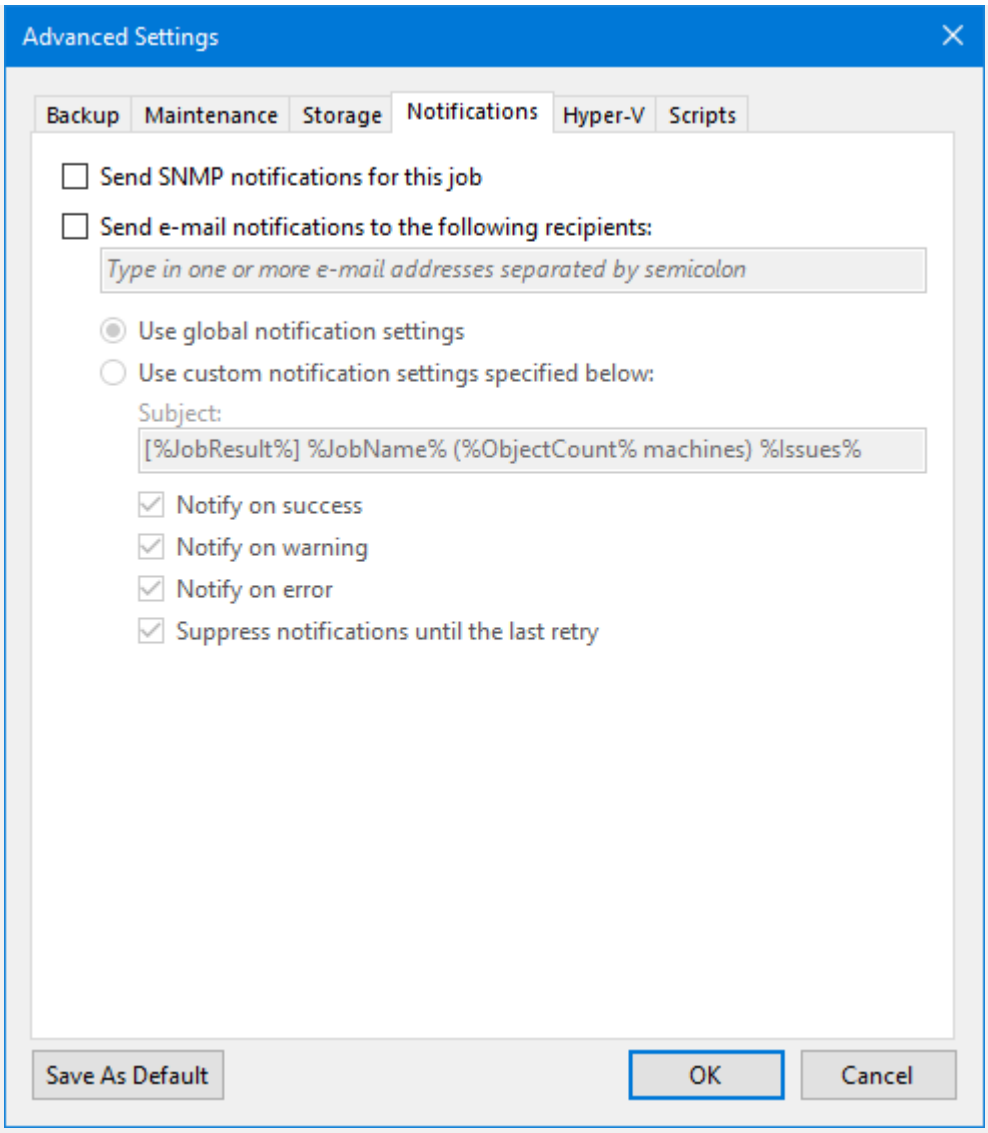
For testing, set *Retention policy* to a few days (not 14 as shown).

When ready, click *Next*.

> If this message pops up:
>
> 
>
>
> then click *OK* and then *Advanced* to open the settings for *Backup* and select **Incremental**:

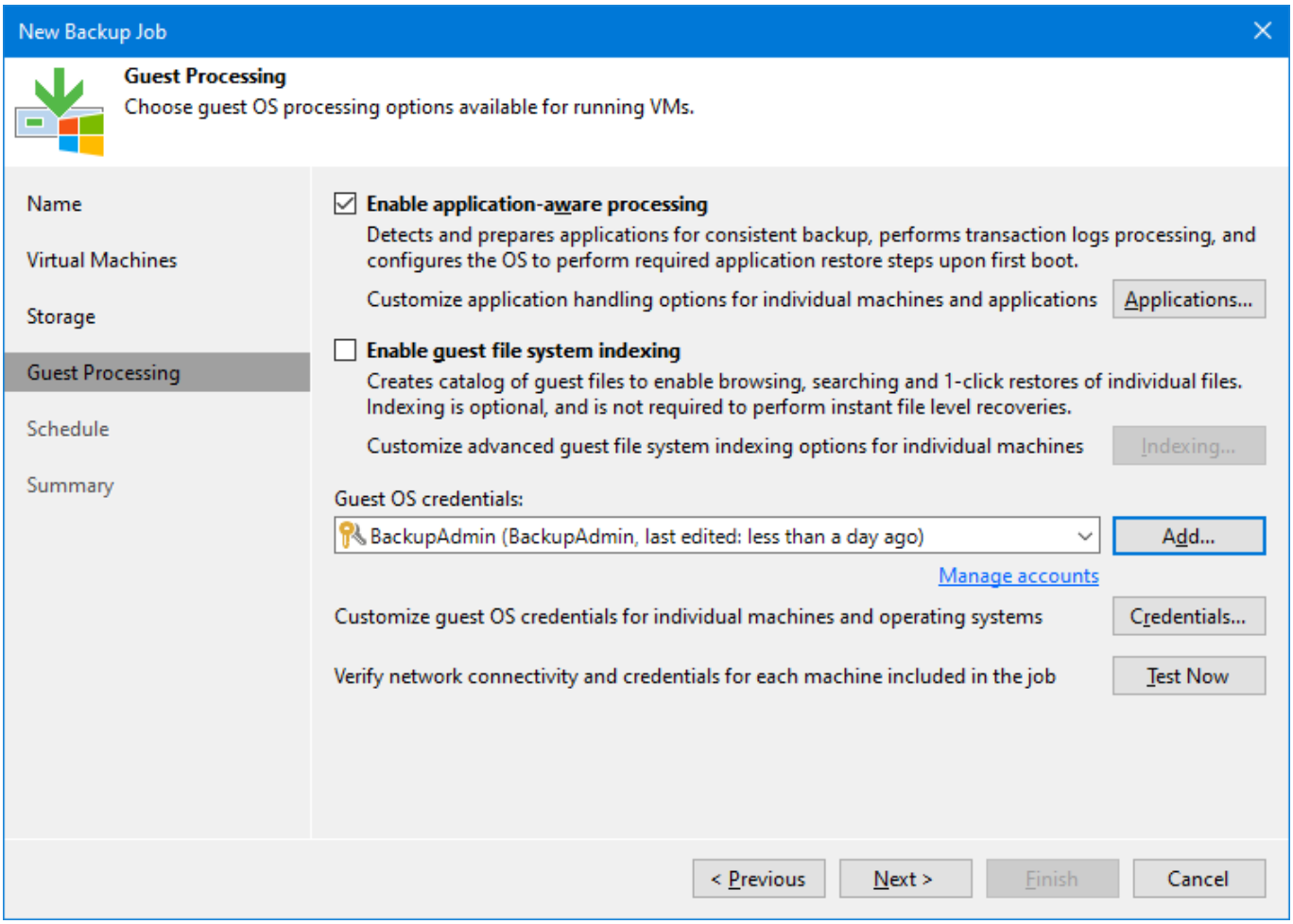


You may also adjust *Notifications* (if you don't, the default notification method will be used):



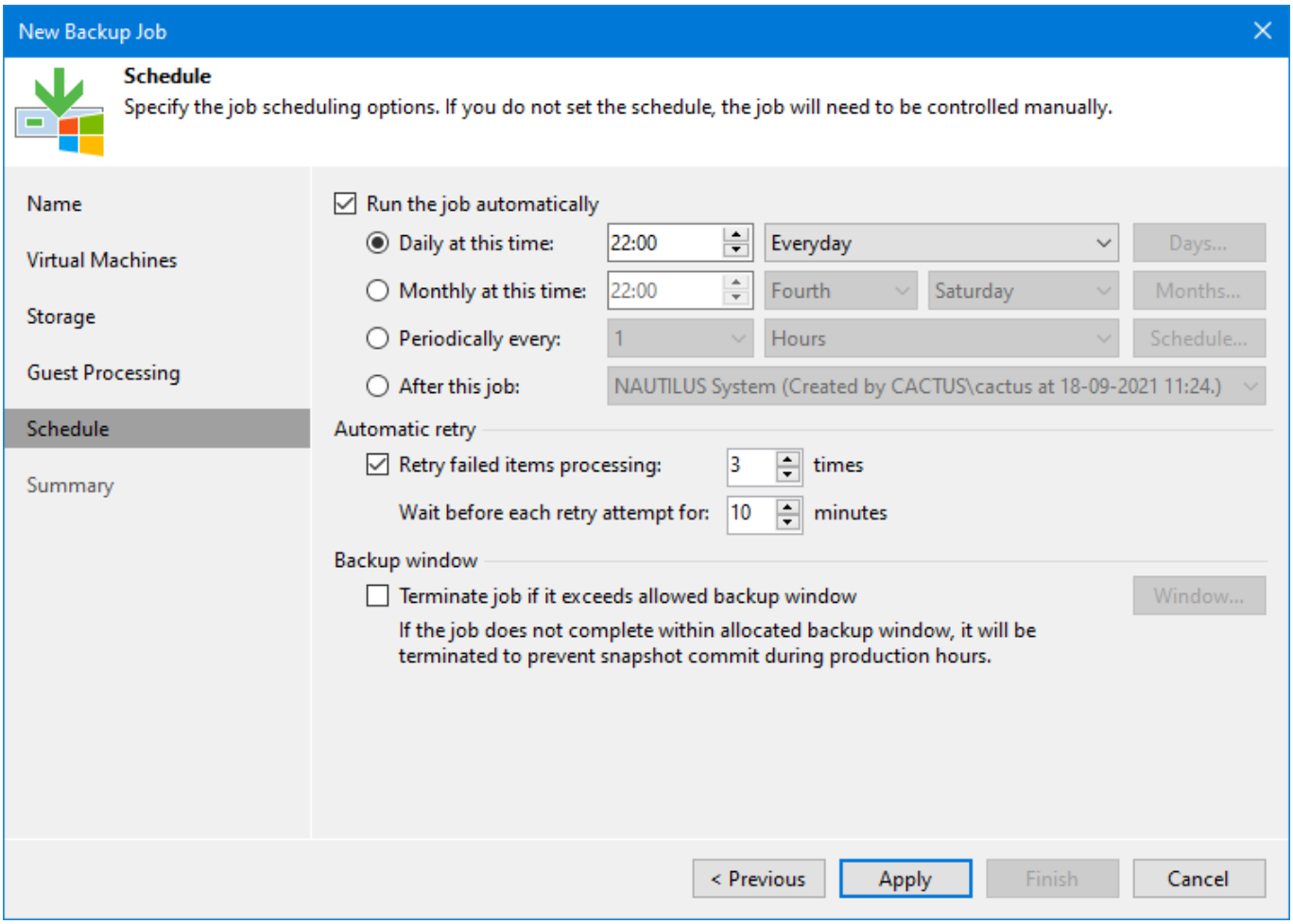
Click *OK* when done.

Click *Next* to view *Guest Processing*:

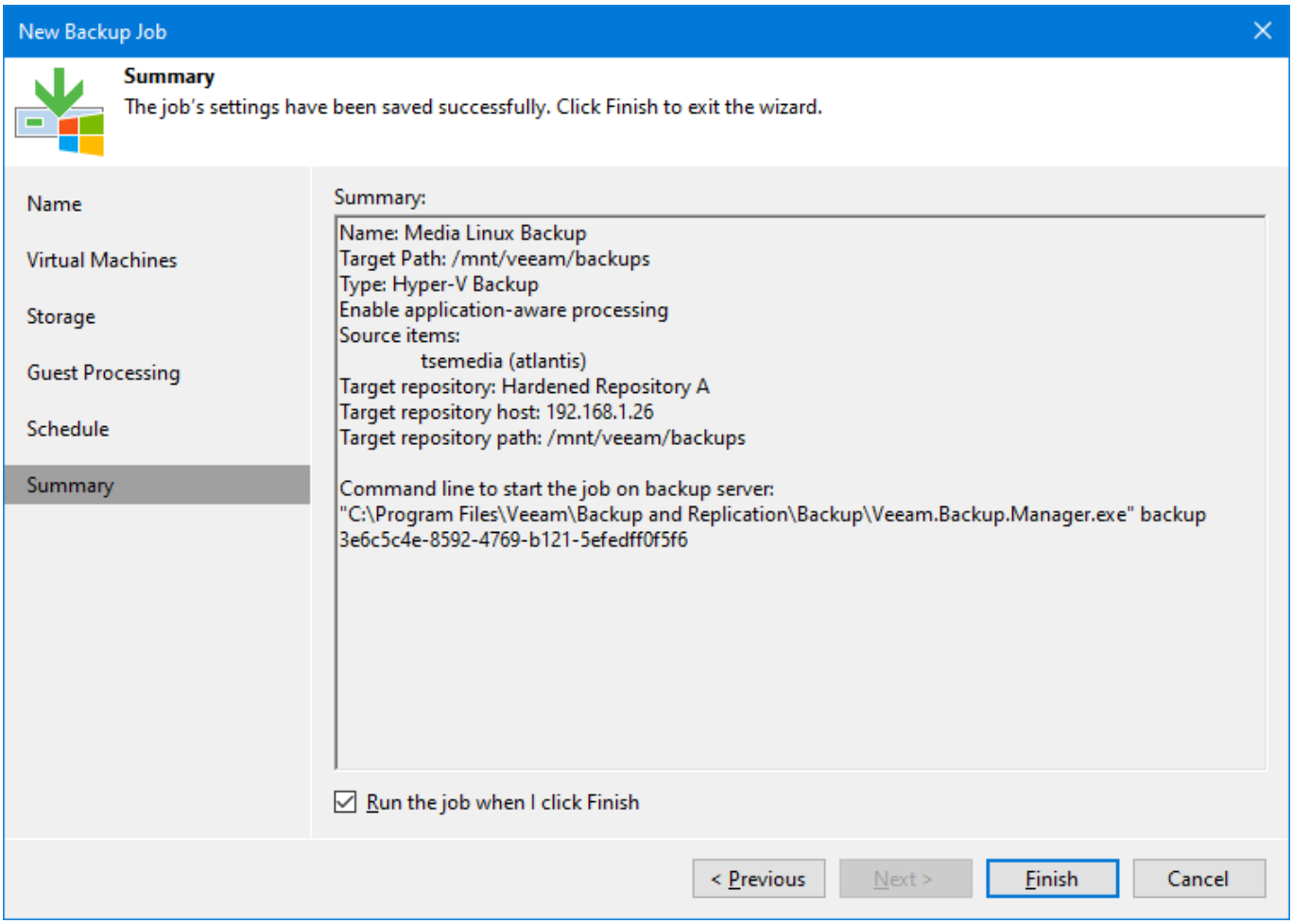


If you mark *Enable application-aware processing*, an account must be selected.

Click *Next* to adjust the *Schedule* for the job:

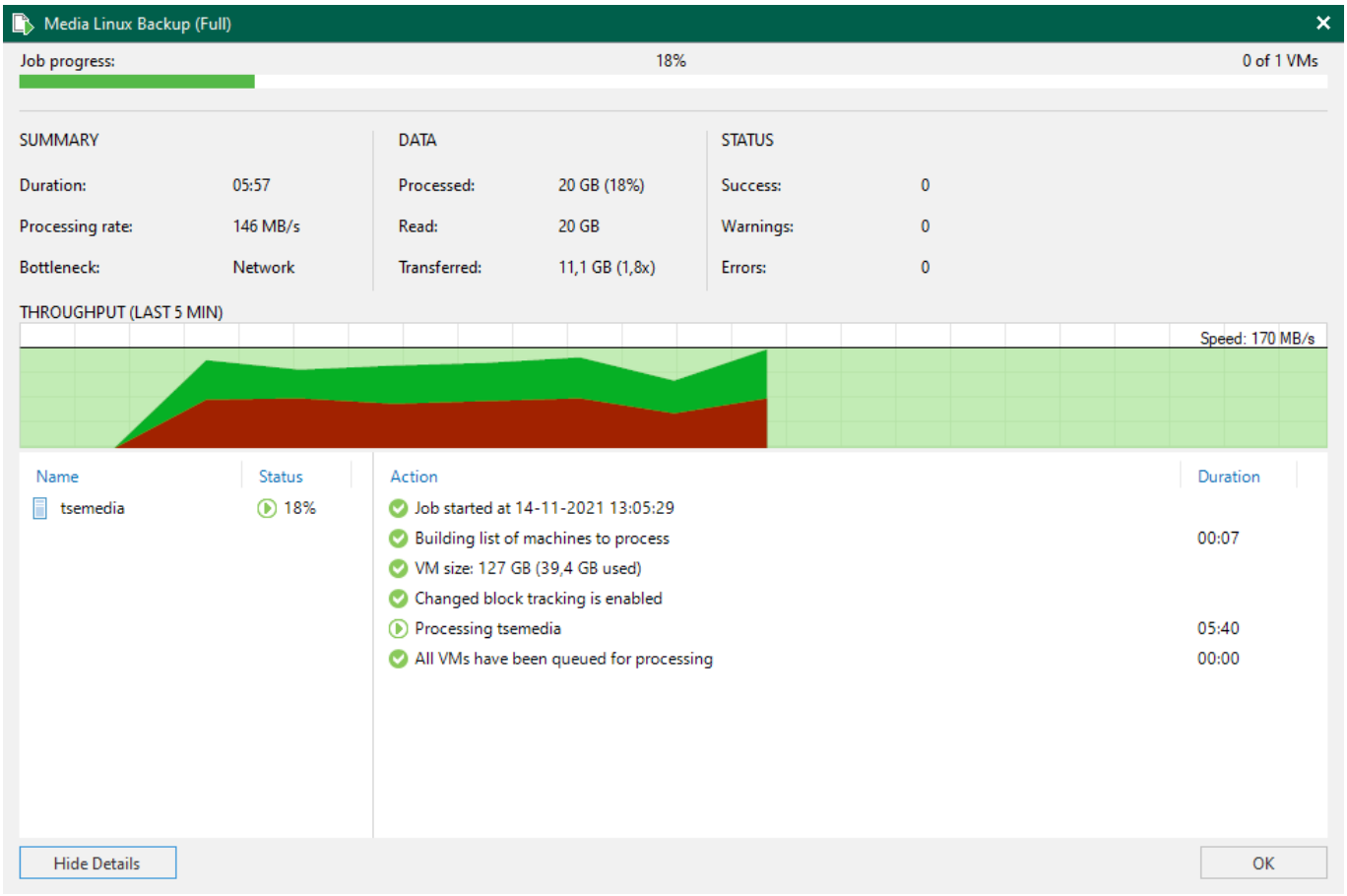


When ready, click *Apply* and the *Summary* will be displayed:

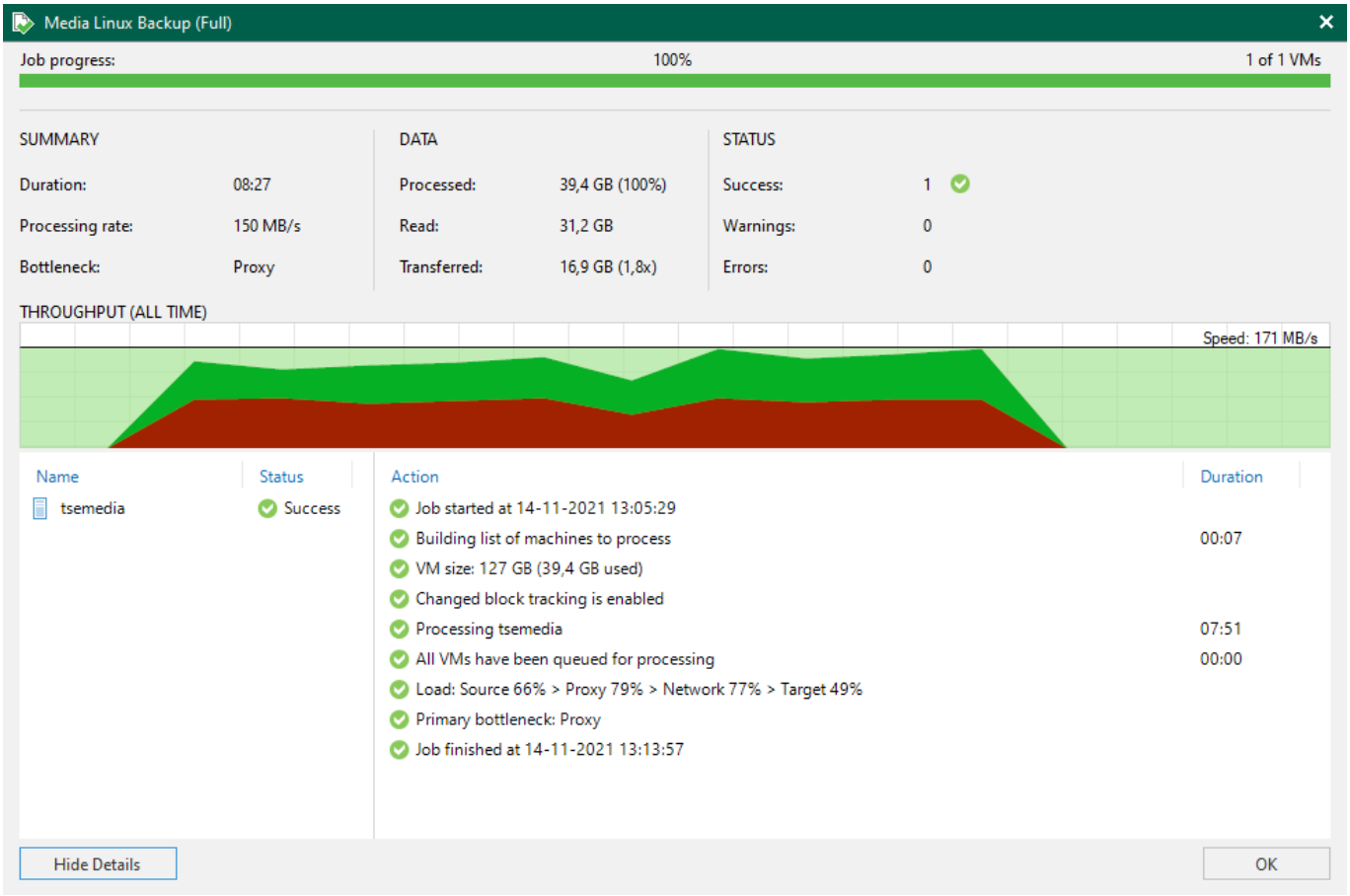


Mark *Run the job when I click Finish*.

Click *Finish*, and the job will start:



and finish:

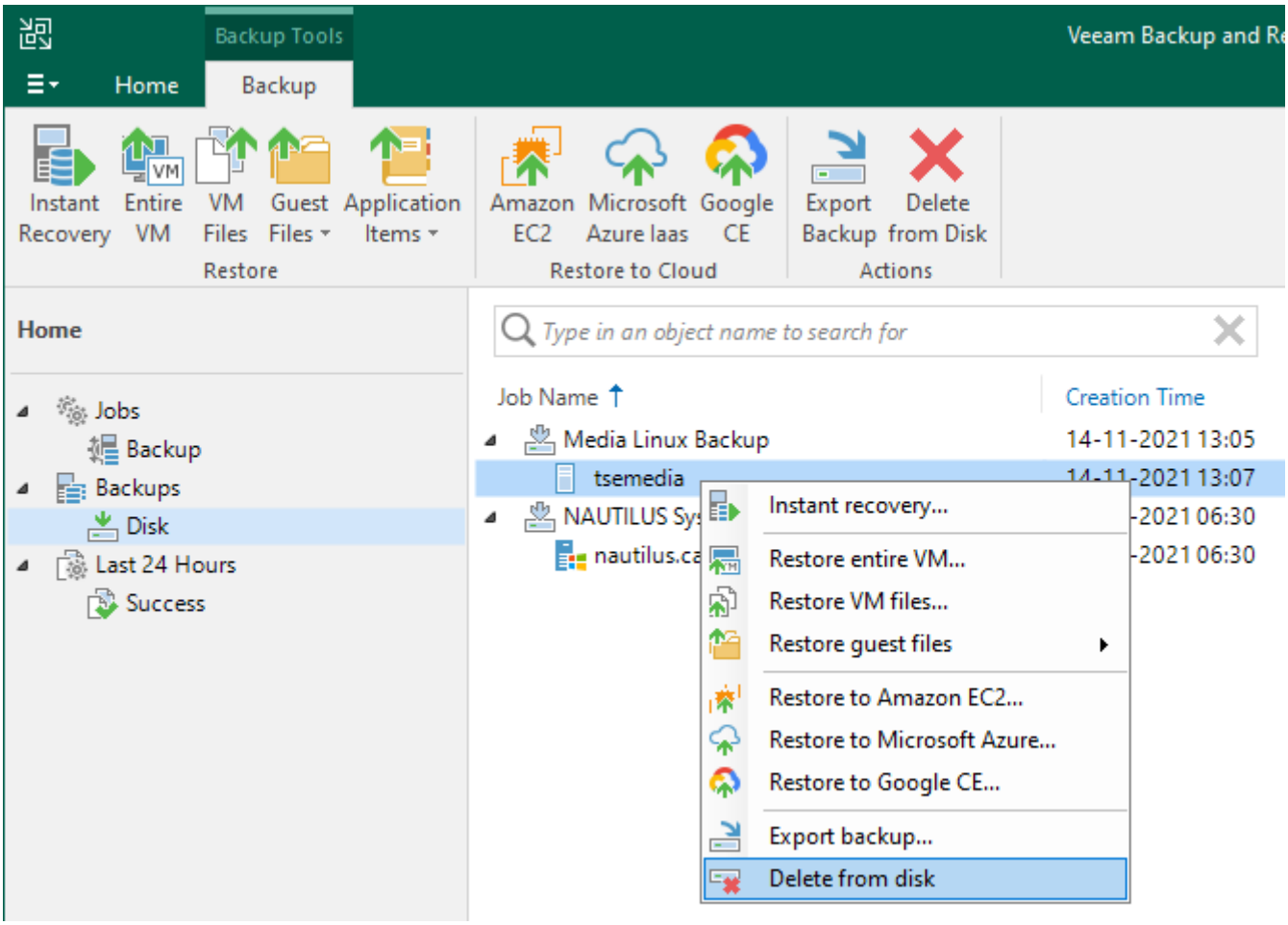


When *Status* reports *Success*, navigate to:

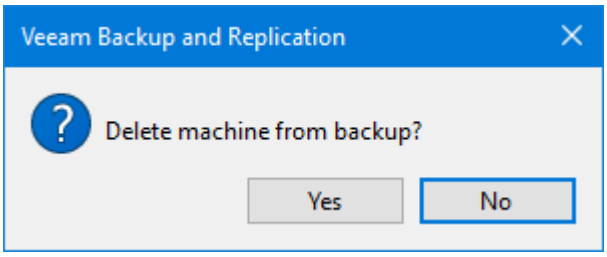
- Home
    - Backups
        - Disk

In the right pane, unfold the backup job, right-click the backup, and from the popup menu select:

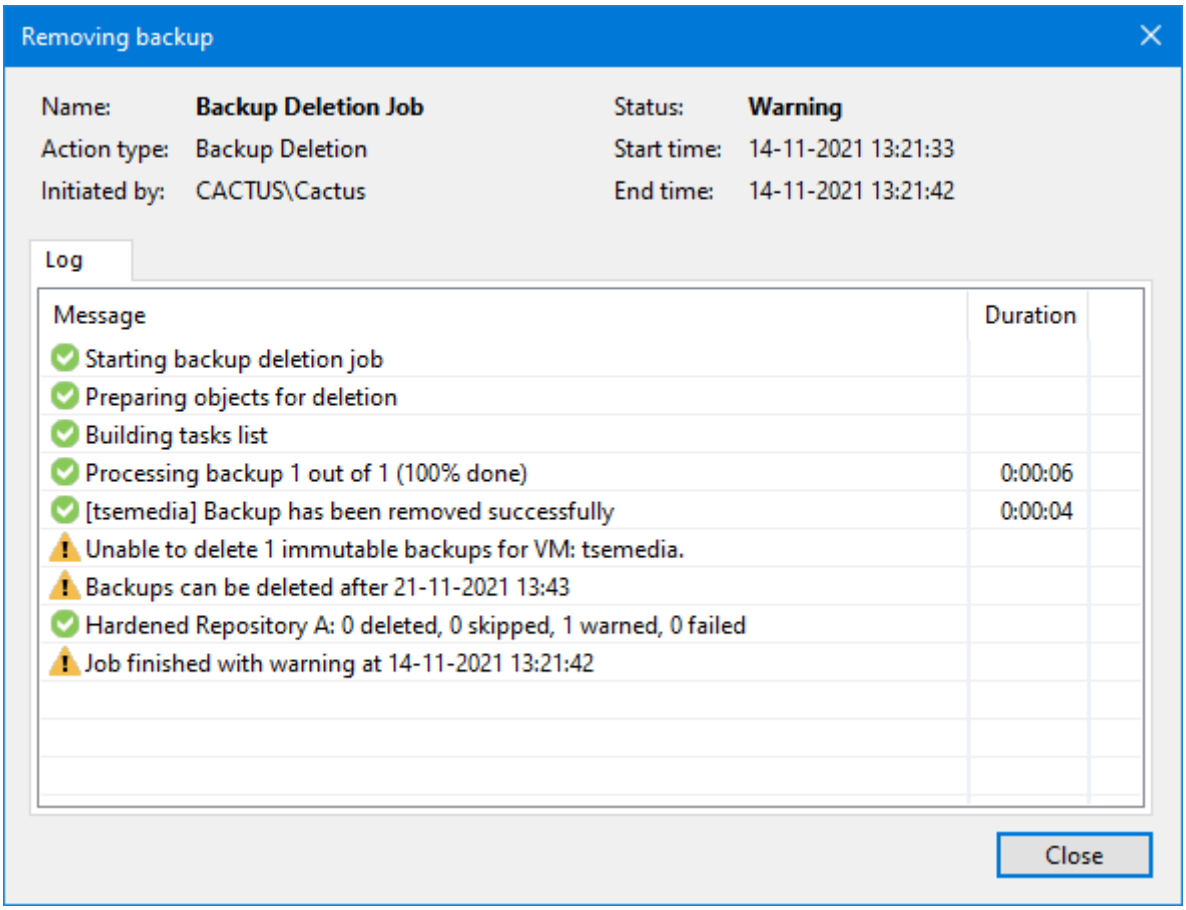
- Delete from disk



You will be prompted:



Click *Yes* and watch the progress of the deletion job and the warnings:



As you will see, the Linux server refused to delete the backup file at this moment. It also tells, at which time it will be possible to delete the file.

The hardened repository is ready.

Conclusion

The server has now proved its ability to host a backup and keep it immutable for a certain amount of days.

The only option to delete the backup file is to have physical access to the machine or to have the credentials for administrator.

However, you can tighten the security further by implementing MFA/2FA authentication making it impossible to login to the server having only the credentials. This we will establish in **Part 8** of this series:

Build an immutable backup repository for Veeam Backup & Replication. Part 8

Next step, however, is to establish *backup of the Linux server* itself. This we will arrange for in **Part 5** of this series:

Build an immutable backup repository for Veeam Backup & Replication. Part 5