2021

Immutable Backup Repository OVA

A Virtual Appliance designed for mounting iSCSI LUN as Veeam Immutable Backup Repository

JOHN BORHEK

VMsources Group, Inc.
www.vmsources.com
1-866-644-7764

Contents

VMsources Group, Inc.
www.vmsources.com
1-866-644-7764

Introduction

Veeam Backup and Replication 11 is a true game-changer with its Immutable Backup feature. Moving forward, Veeam Immutable Backups will free organizations from the hassle of physical tape libraries as well as eliminating the need for expensive "virtual tape libraries," all while providing easily manageable backup chains.

Veeam Immutable Backups are the equivalent of "air gapped," tape, or WORM backups! Immutable Backups are accomplished through the use of a Linux repository server, the XFS filesystem (to support Fast Clone), one-time-use credentials, and the immutability flag which is a property of just about any Linux filesystem.

What about validation? Does it actually work, you ask? Yes! And it is tested and proven by Cohasset Associates meeting the requirements for non-rewritable, non-erasable storage as specified by SEC 17a-4(f), FINRA 4511(c) and CFTC 1.31(c)-(d).

What an Immutable Repository is designed to do:

- Protect your backups from unintentional encryption by Ransomware
- Prevent backup file deletion by unauthorized administrators.

What an Immutable Repository cannot do:

- Prevent deletion of the SAN LUN by users who have SAN administrator credentials
- Prevent formatting of the volume by users who have login credentials for the Immutable Repository VM.

As a Best Practice, Veeam recommends that Immutable Repositories be deployed on a physical server with lots of disk space, such as an HPE Apollo. While that's an admirable goal, there are lots of us with Virtualized and Cloud Infrastructures, and we rely on SAN systems to provide block storage.

VMsources' goal has been to create a Virtual Appliance in OVA format, following Veeam Best Practices, making it easy for anybody to mount an iSCSI LUN as a Veeam Immutable Repository. Not everybody is fond of Linux or the CLI, so we have created an Ubuntu 20.04LTS server with all of the requirements baked-in as an OVA Virtual Appliance, and then documented the step-by-step instructions to mounting a SAN LUN, formatting it as XFS, and then creating your Veeam Immutable Repository.

PLEASE NOTE: **This Virtual Appliance is provided without warranty of any kind.** You must change the passwords to very strong, disable SSH, and the VM must run on secure, firewalled networks in order for any of this to amount to a hill of beans.

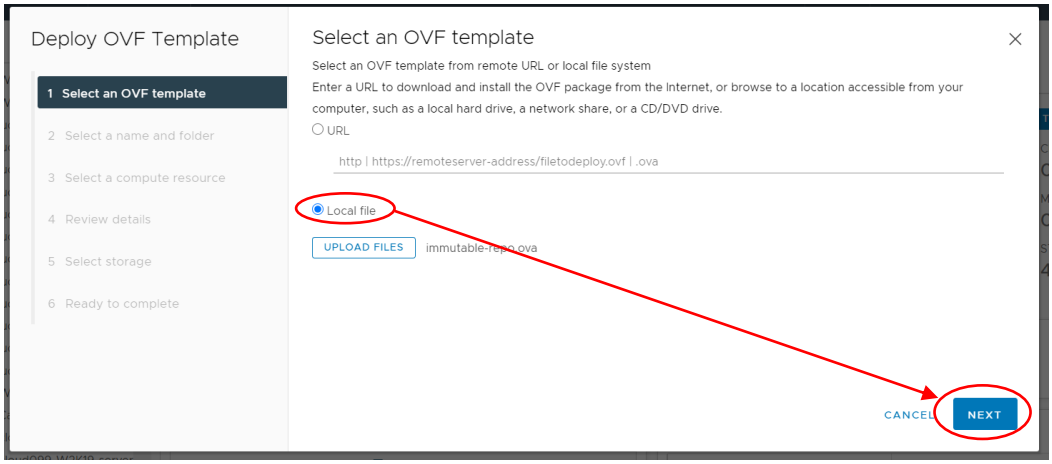
Download the appliance: https://www.johnborhek.com/download/immutable-repository-for-veeam-backup-and-replication-11/

Creating a LUN on Your iSCSI SAN

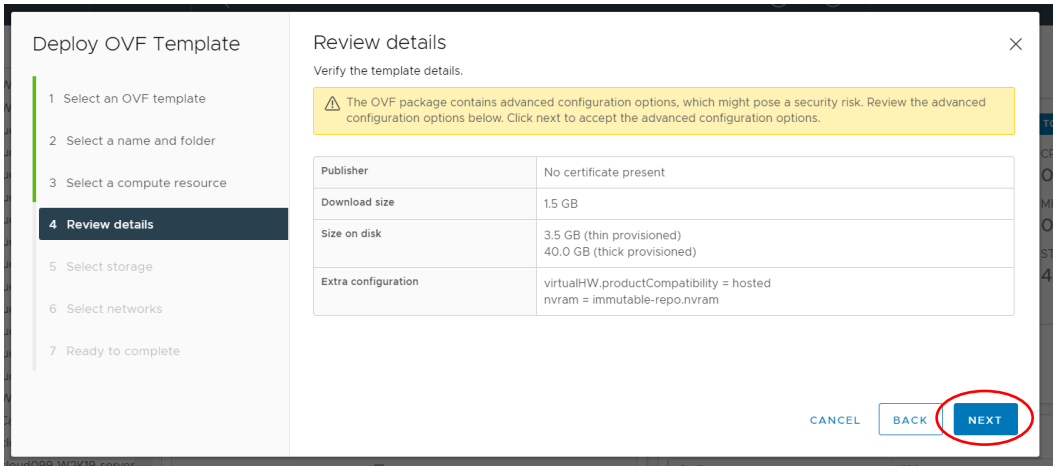
1. Create a LUN on your iSCSI SAN
    a. Most likely, your iSCSI SAN will require you to map the IQN of the initiator (the Immutable Repository) to the LUN which you have carved out. The Immutable Repository IQN is: iqn.2008-05.com.vmsources.immutable-repo:01
    b. The IQN can be configured by editing: /etc/iscsi/initiatorname.iscsi

Import the OVA

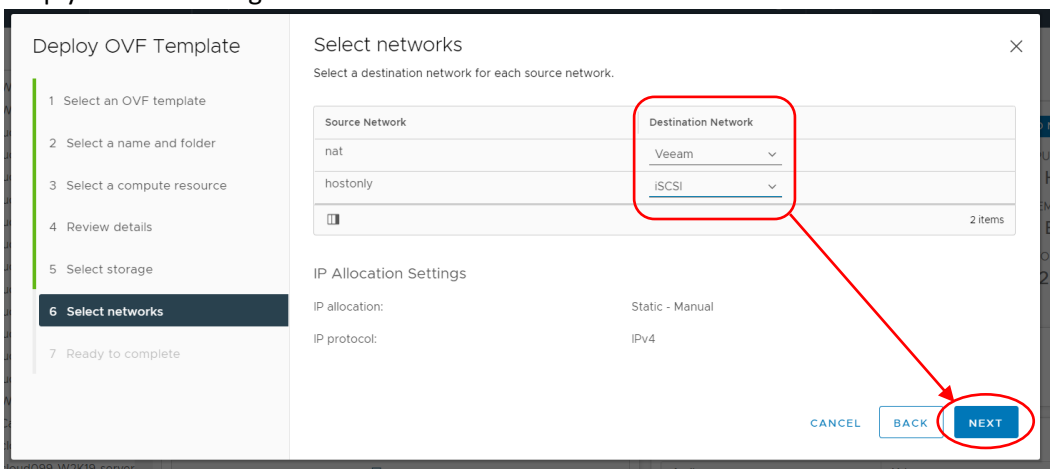
1. Import the Immutable Repository OVA to your VMware vSphere environment.
    a. Locate the OVA package and click Next.



    b. This is expected when importing an OVA. Click Next.



c. Map your networking and click Next.



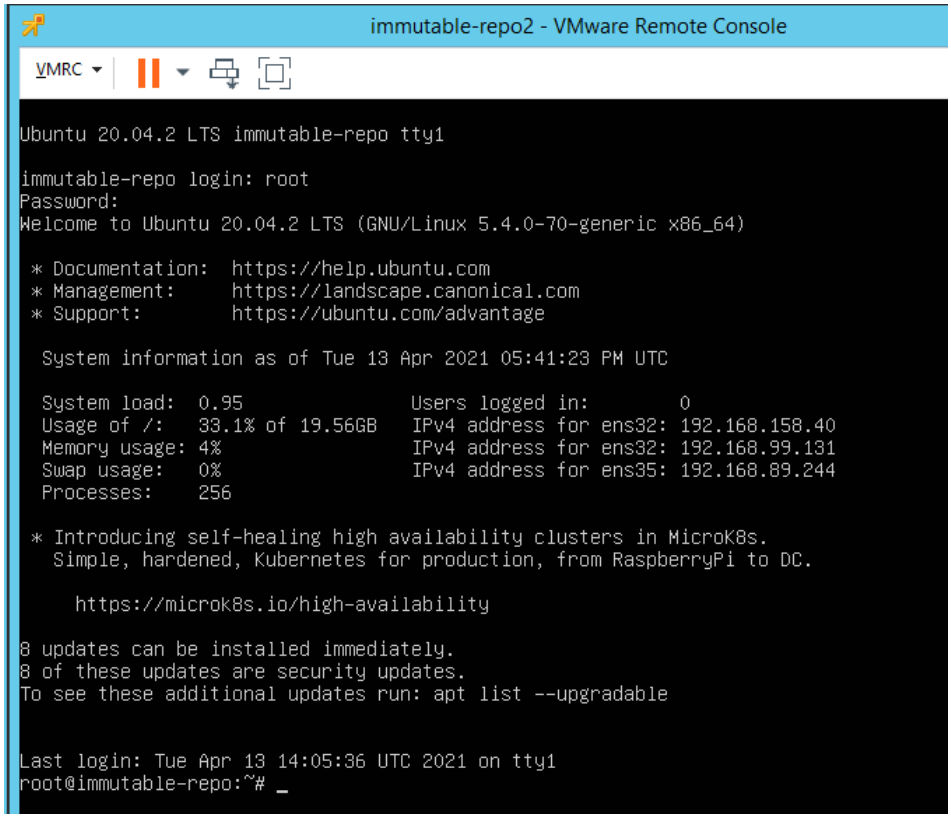
i. Place the Source Network 'nat' on the same network as your Veeam Servers
ii. Place the Source Network 'hostonly' on the same network as your iSCSI target ports

Powering on and Configuring the Immutable Repository VM

The initial password for both root and administrator is: P@ssw0rd (That's a zero in P@ssw0rd)

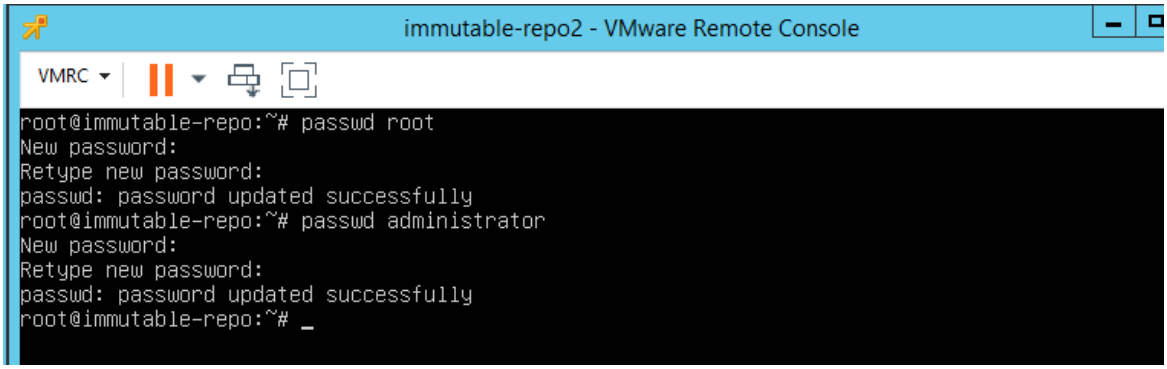
1. Power on immutable-repo VM

2. Open VMRC to immutable-repo VM and log on as root (possible because you're using the console).



3. Change root and vbruser password (HINT: use a randomized long password that is not used on any other system and store this password securely and offline). NOTE: the username “vbruser” is arbitrary and can be changed as needed. There is only a requirement for an un-privileged user that has ownership rights to the Immutable Repository folder (Later)

```
passwd root
passwd vbruser
```



4. Change the Netplan network config to suit your environment. You can use the /etc/netplan/50-immutable-init.yaml as a template and modify the variables (in white). Be careful, indentation is important in this file, follow the example.

```
vi /etc/netplan/50-immutable-init.yaml
```

   i. ens32 is your Veeam/Management Network. It will require an IP address, gateway and nameservers. Modify the values in white.
   ii. ens35 is your iSCSI network. In most cases a gateway is not required or even desirable on your iSCSI/SAN network. Modify the values in white.
   iii. If Jumbo Frames are not wanted, remove the MTU line

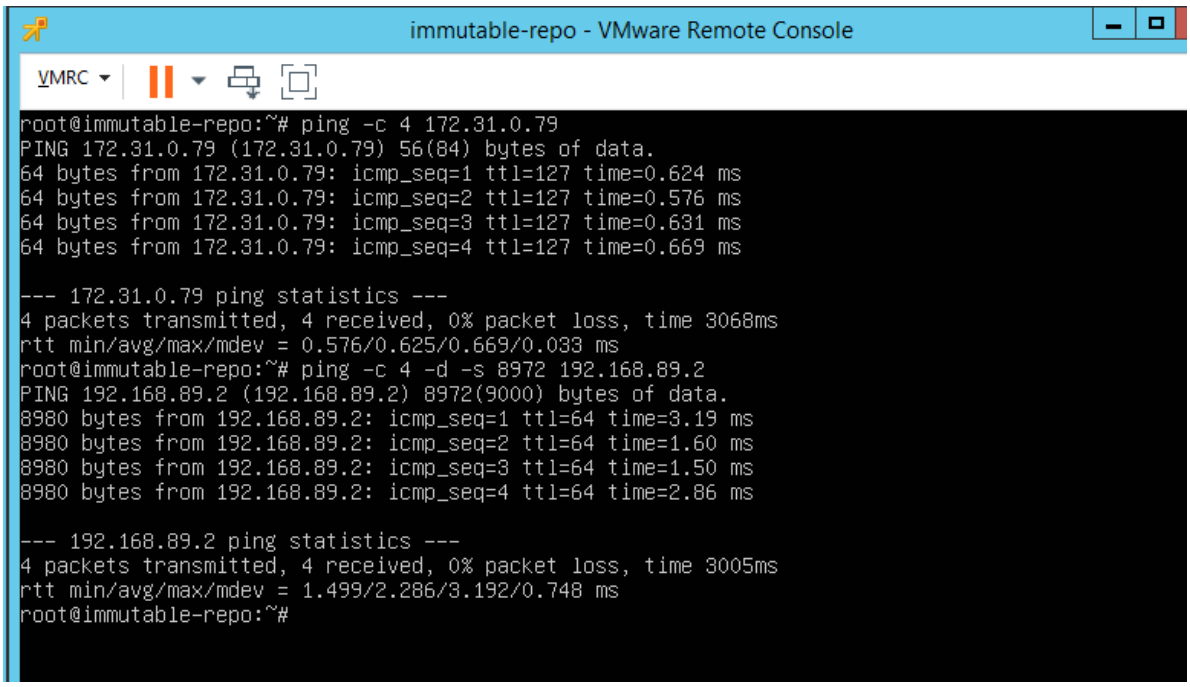
```
# This is the network config written by 'subiquity'
network:
  ethernets:
    ens32:
            addresses:
              - 192.168.158.40/24
            gateway4: 192.168.158.2
            nameservers:
                    search: [mydomain, otherdomain]
                    addresses: [8.8.8.8, 8.8.4.4]
            dhcp4: true
    ens35:
      addresses:
              - 192.168.89.244/24
      mtu: 9000
  version: 2
~
~
~
~
```

b. Run the command:

```
netplan apply
```

i. If it returns an error, most likely indentation is off. Go back and look for mistakes.

5. Now, let's ping our Veeam server and iSCSI Target to make sure networking is correct.
   a. Run the command: `ping -c 4 <IP of Veeam>`
   b. Run the command: `ping -c 4 -d -s 8972 <IP of ISCSI Target>`
   c. If all pings are returned, you are configured correctly.
   d. If the iSCSI ping does not return, remove: mtu: 9000 from Netplan
      i. Run: `netplan apply again`.



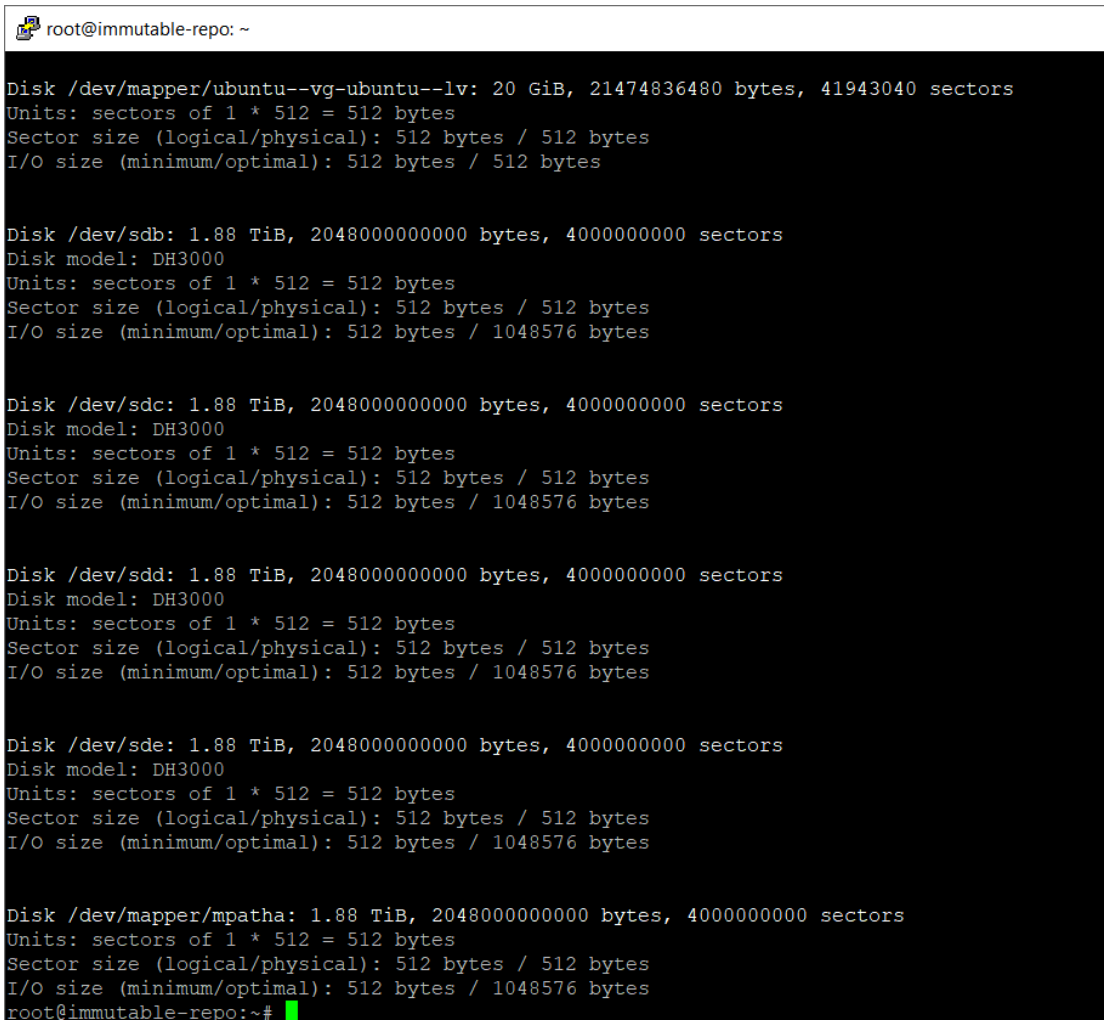
6. At this point, you can log on using ssh or continue to use the VMRC console. ssh is convenient as it will allow you to copy and paste the required commands to get started. To login with ssh:
   a. Start by logging on as the user: administrator
      i. Use the password you just assigned to the administrator user
   b. Then execute: su –
      i. Use the password you assigned to root

7. Now, configure your iSCSI Target by running the commands, one line at a time:

```
iscsiadm -m discovery -t sendtargets -p <iSCSI Target IP>
iscsiadm -m node -o show
iscsiadm -m node --op=update -n node.conn[0].startup -v automatic
iscsiadm -m node --op=update -n node.startup -v automatic
iscsiadm -m node --loginall=automatic (requires time)
systemctl enable open-iscsi
systemctl enable iscsid
systemctl restart iscsid.service
iscsiadm -m session -o show
```

8. Find your volume by running the command:

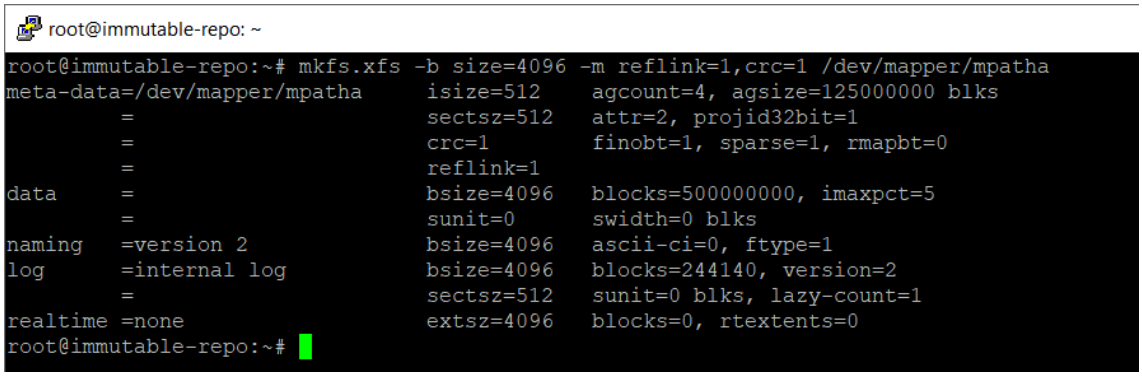
```
fdisk -l
```



Your device will either be identified by: /dev/sdb (if there is only one target with one path, or by: /dev/mapper/mpatha (if it is a multipath iSCSI SAN). You will want to use multipath (mpath) if available.

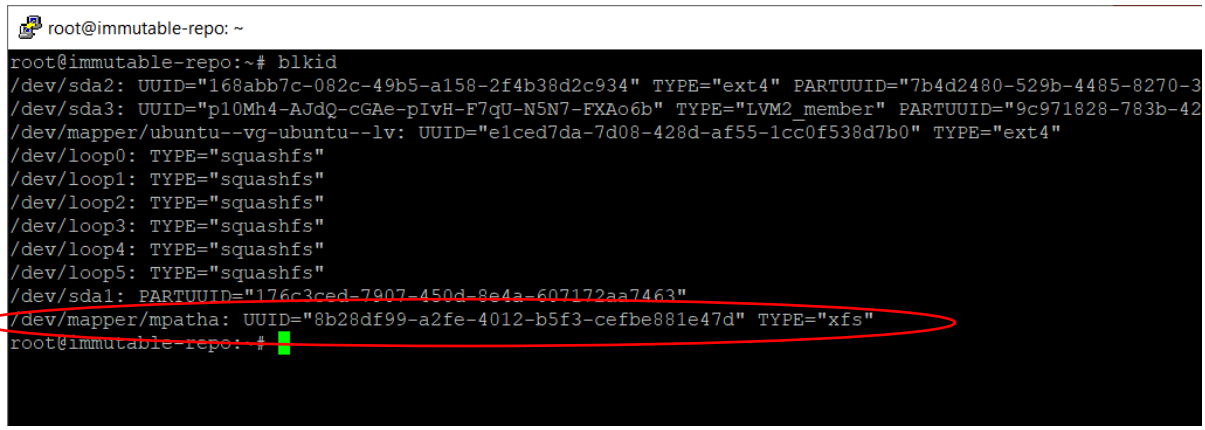
9. Once you have identified your volume, format it with this command with /dev/sdb or /dev/mapper/mpatha as appropriate (takes a bit):

```
mkfs.xfs -b size=4096 -m reflink=1,crc=1 /dev/mapper/mpatha
```



10. Identify the UUID of the volume:

```
blkid
```



11. Copy the UUID of the xfs volume to a text file, but remove the quotations for use later:

```
UUID=8b28df99-a2fe-4012-b5f3-cefbe881e47d
```

12. Your Immutable Repository mount point is pre-configured and ready-to-go with the correct ownership and user permissions pre-assigned as:

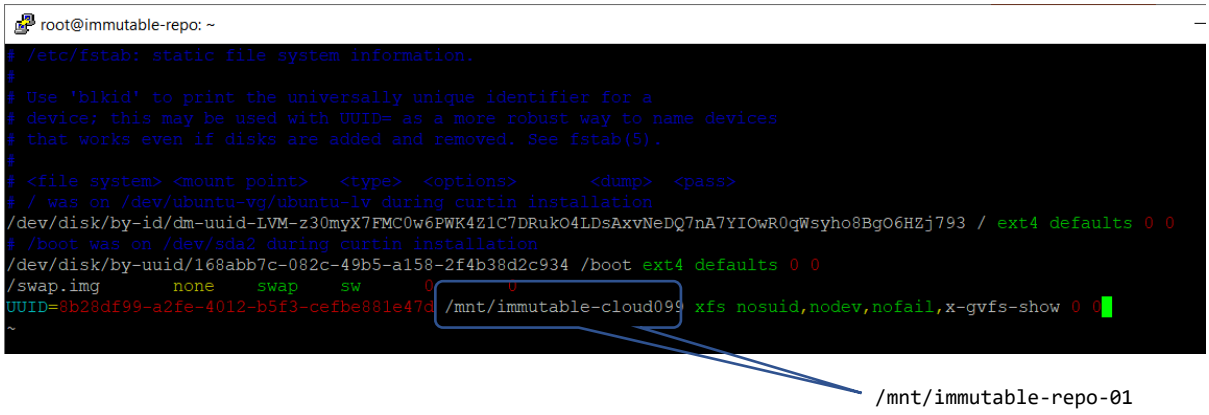
```
/mnt/immutable-repo-01
```

13. OPTIONAL: Step 13 is entirely optional as there is a pre-configured user and mount-point. If you wish to create additional vbrusers and repositories using this server, add additional users and mount points using these commands:

```
adduser newvbrusername
usermod -aG sudo newvbrusername
mkdir /mnt/newvbrreponame
chown -R newvbrusername: newvbrusername /mnt/newvbrreponame
chmod -R 700 /mnt/newvbrreponame
```

14. Edit /etc/fstab and substitute your UID and mount point (pre-configured mount point is /mnt/immutable-repo-01) for the values shown in the example:

```
vi /etc/fstab
```



a. Add this line, **substituting your UUID** :

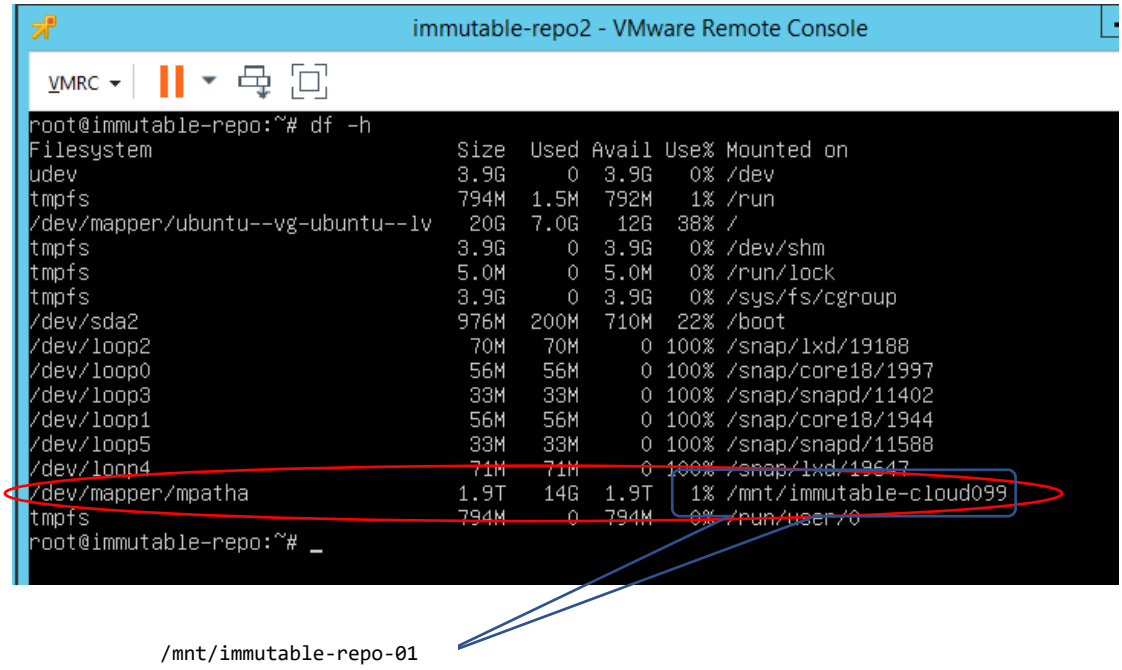
```
UUID=8b28df99-a2fe-4012-b5f3-cefbe881e47d /mnt/immutable-repo-01 xfs
nosuid,nodev,nofail,x-gvfs-show 0 0
```

15. Reboot!

16. Login as “root” using the VMRC.

17. Verify that your volume automounted to the correct path with the command

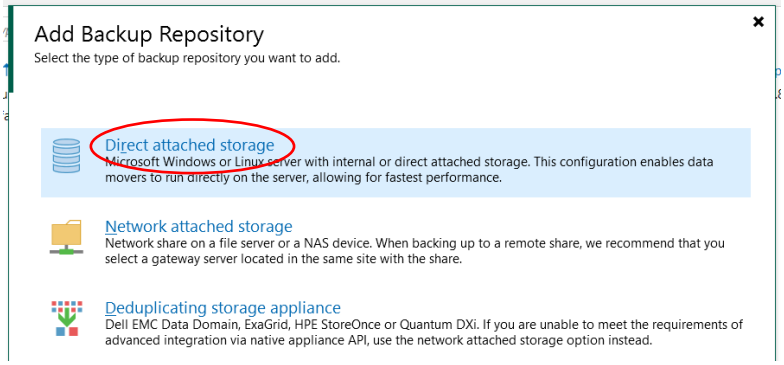
```
df -h
```



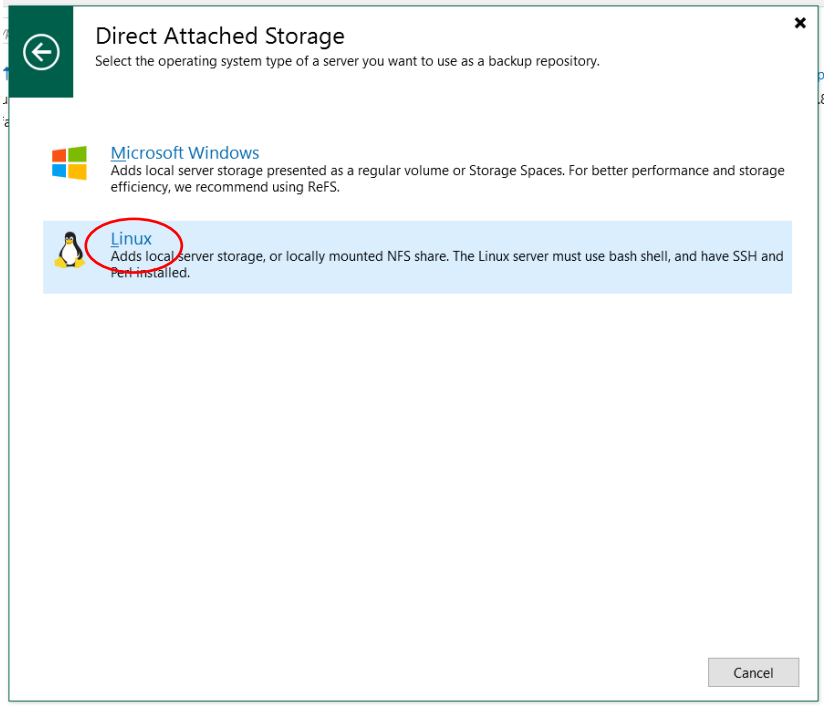
/mnt/immutable-repo-01

Configuring Your Immutable Repository in Veeam Backup and Replication 11

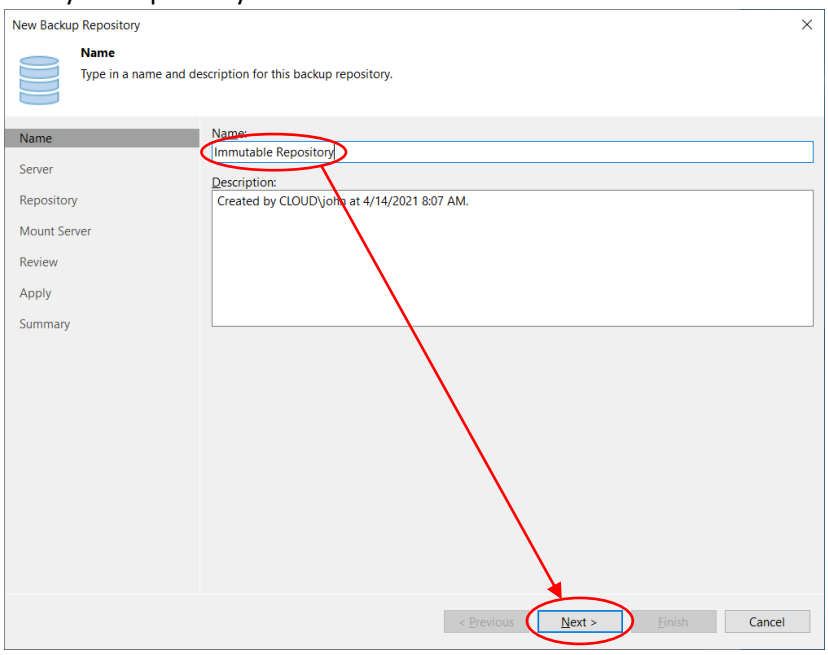
1. Now configure your Veeam Immutable Repository in Veeam Backup and Replication 11.



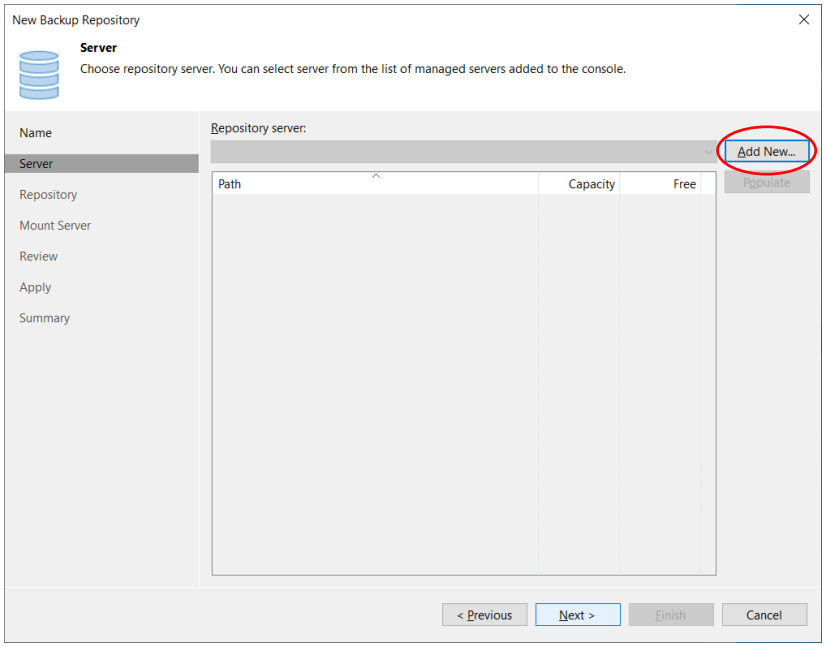
2. Now choose Linux.



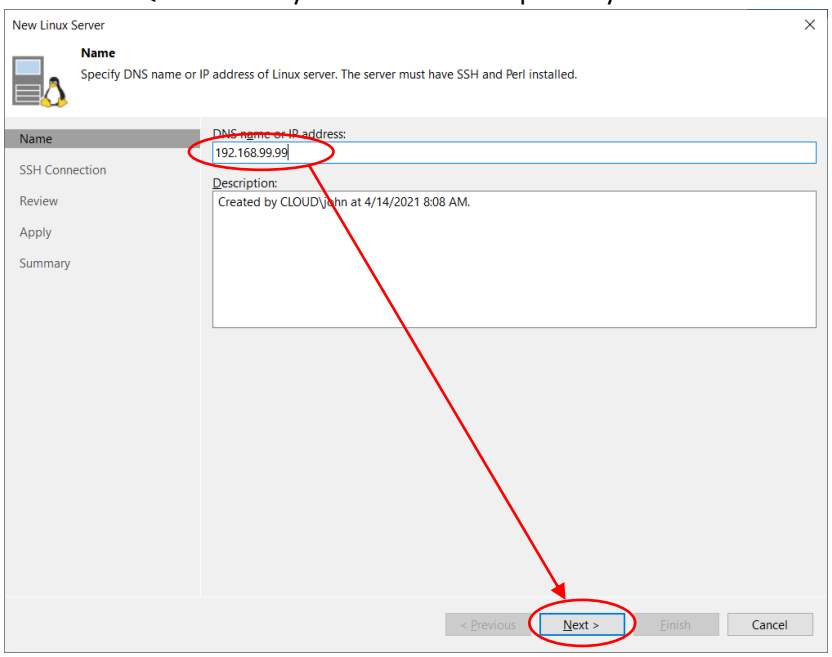
3. Give your repository and name and click Next.



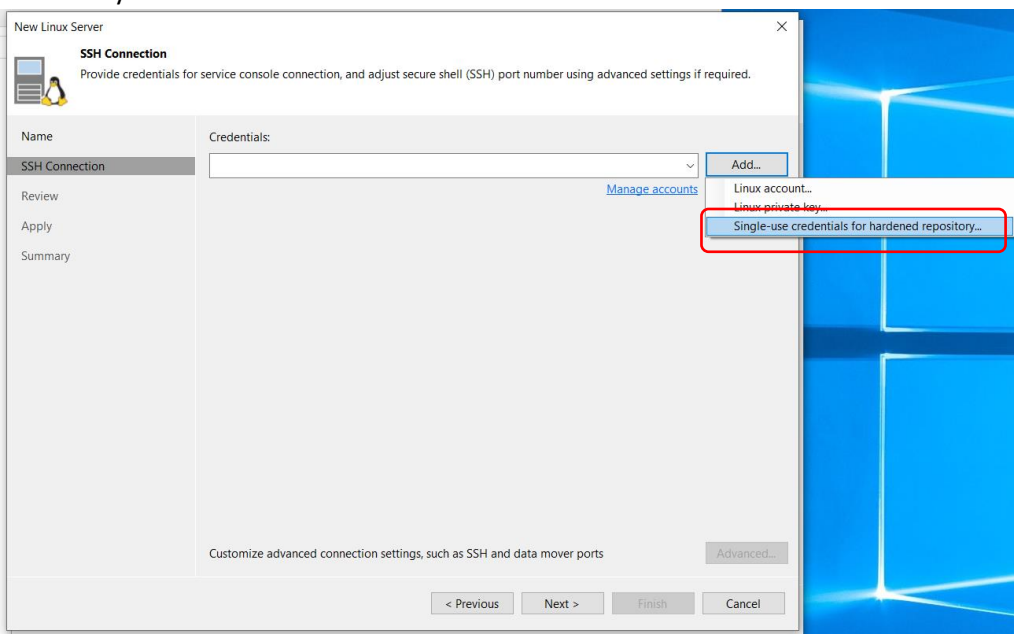
4. Now you need to click on Add New so you can assign a repository.



a. Enter the FQDN or IP of your Immutable Repository VM and click Next.

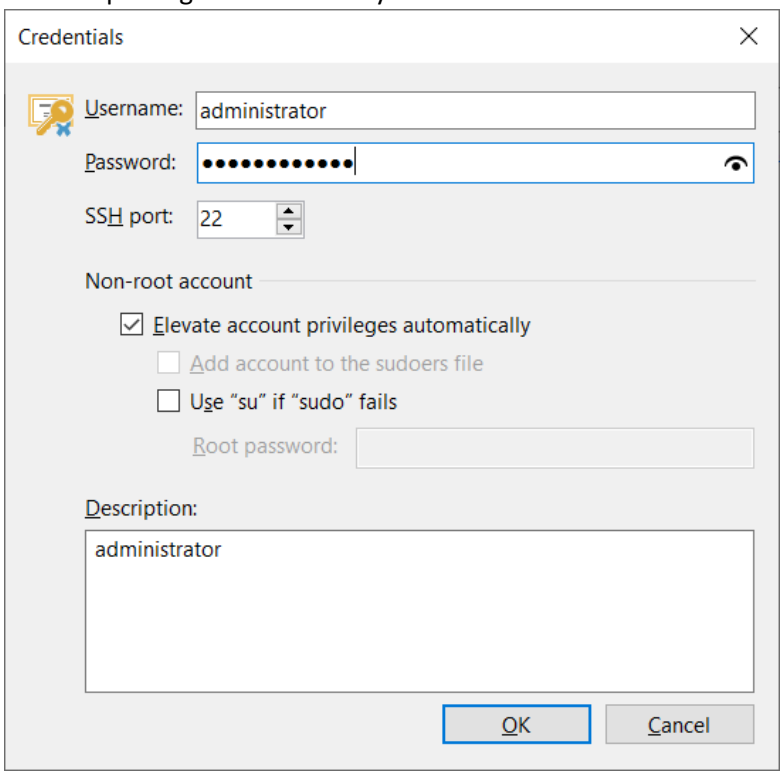


b. Now add your SSH connection.



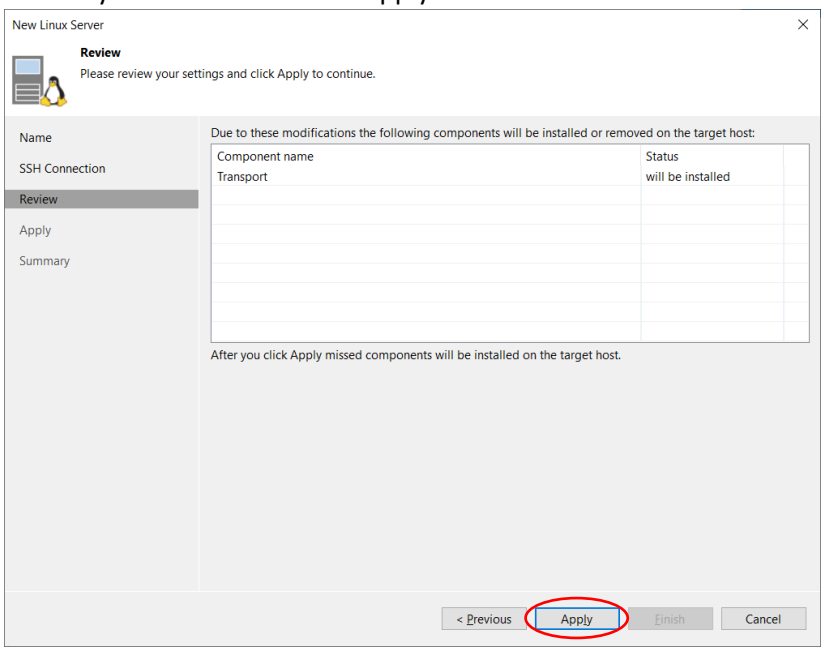
c. You will use the ‘vbruser’ user credentials in Veeam. At this point, you still have root access and could ‘Use “su” if “sudo” fails’, however you shouldn’t have to choose this. Choose only: Elevate

account privileges automatically.

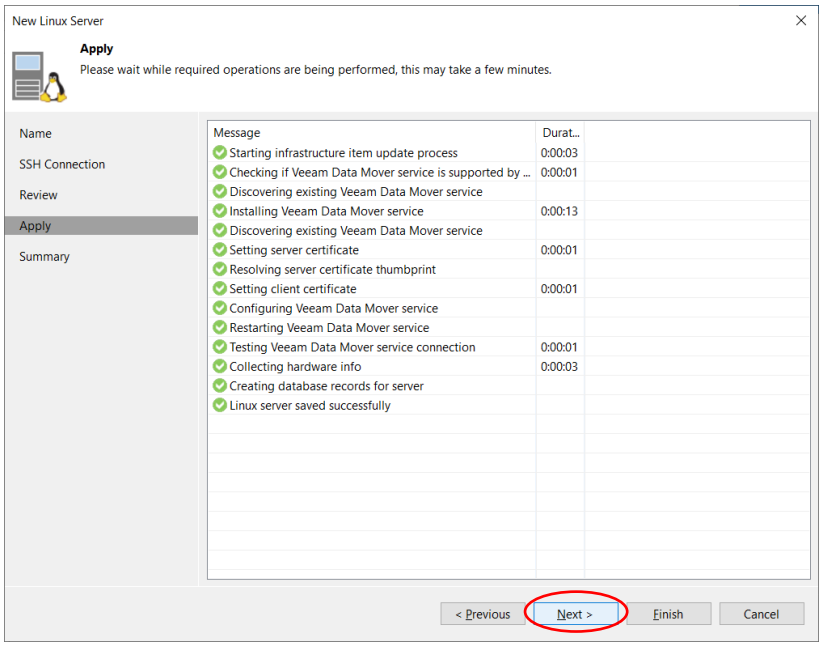


d. We just created it. Click Yes.

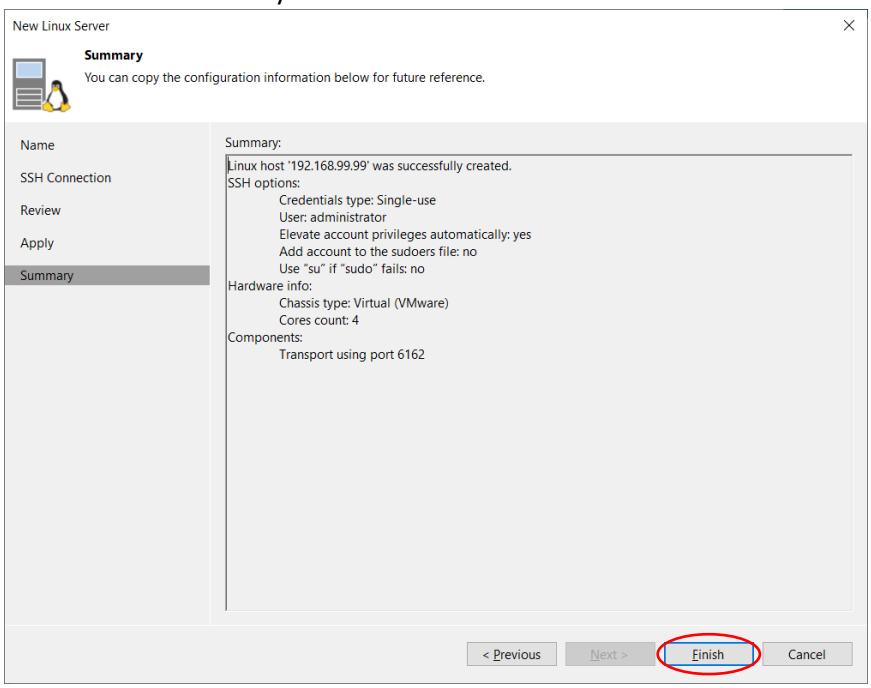
e. Review your choices and click Apply.



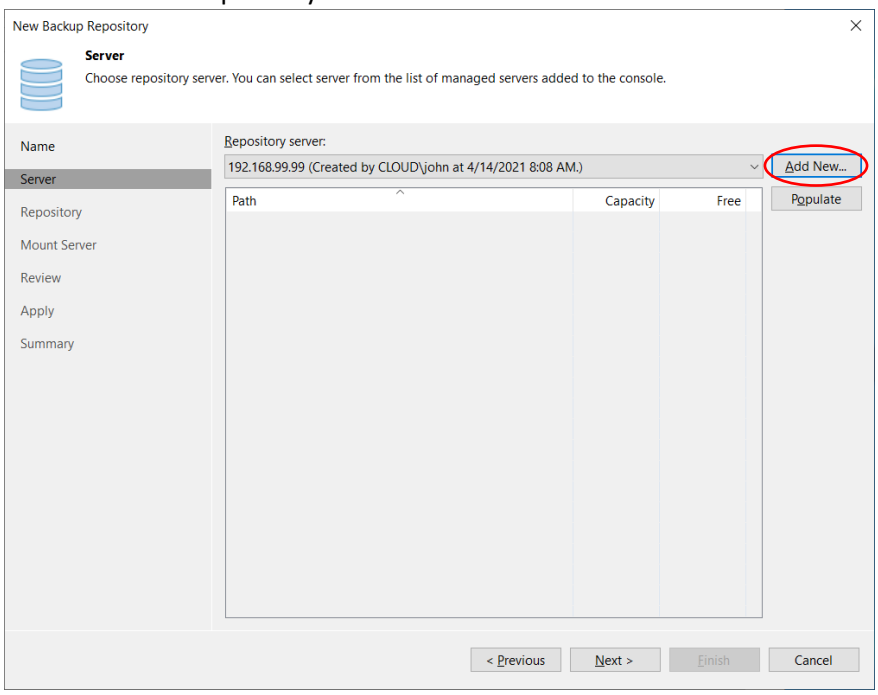
f. After it's finished, click on Next.



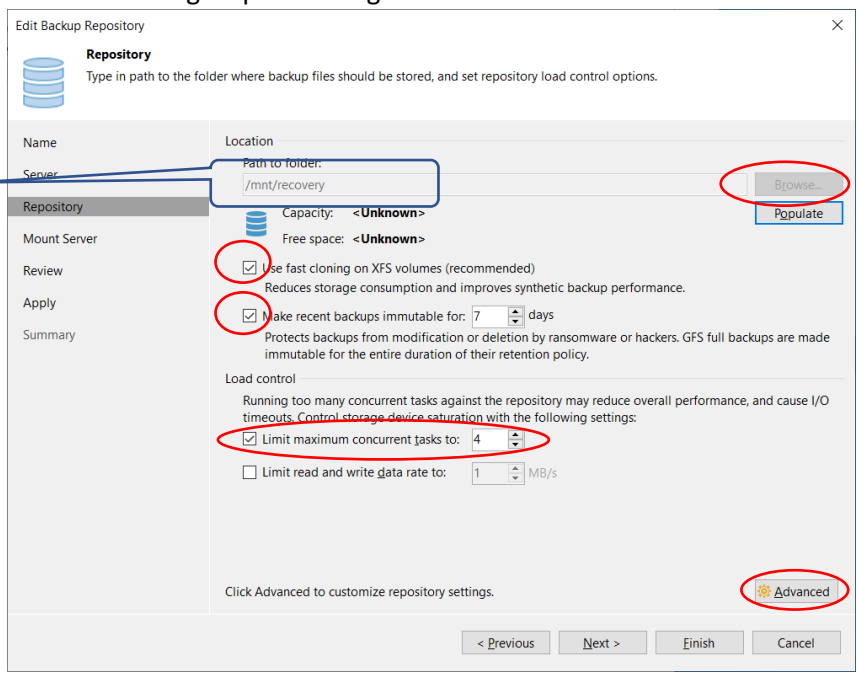
g. Check out the summary and then click Finish.



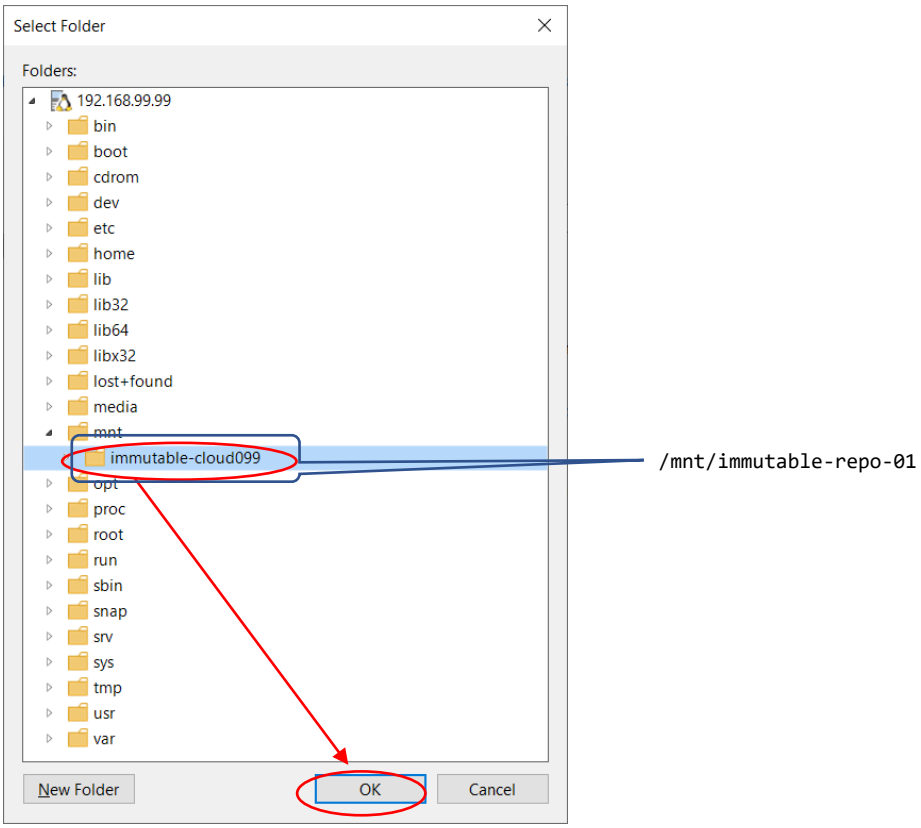
h. Now we need a repository server. Click on Add New.



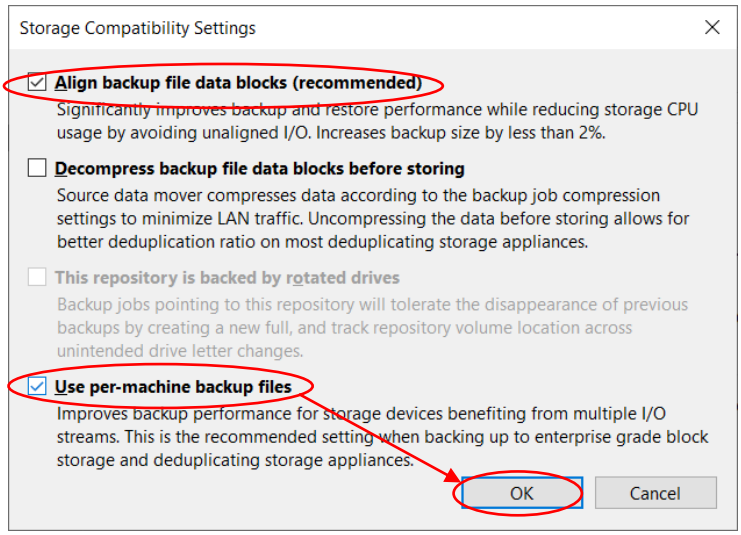
i. Do the following steps to configure:



i. Click Browse, and Locate the /mnt/immutable-repo-01 folder (or folder you created in Step 13) and click OK.



ii. Click on Advanced:
    1. Align backup file data blocks
    2. Use per-machine backup files

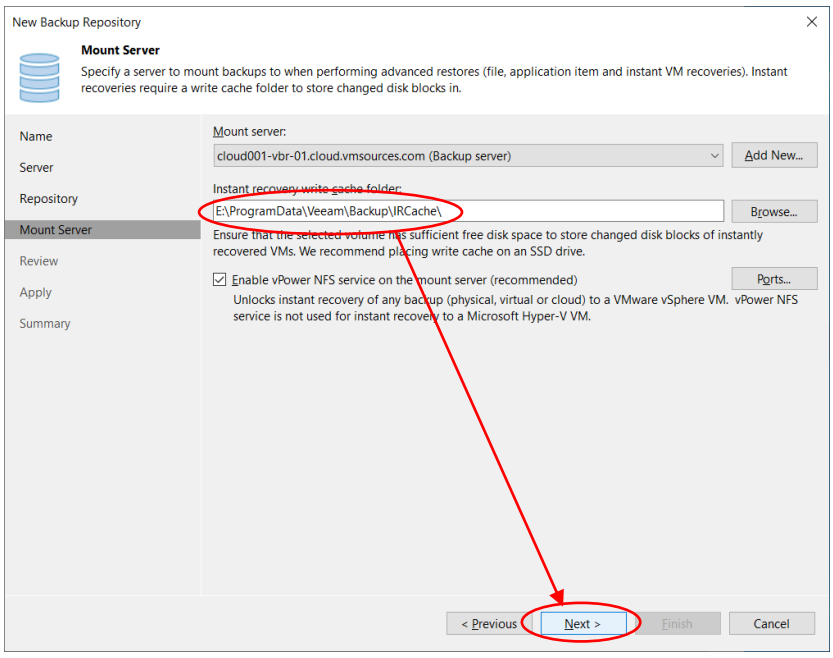


iii. Use Fast cloning on XFS….

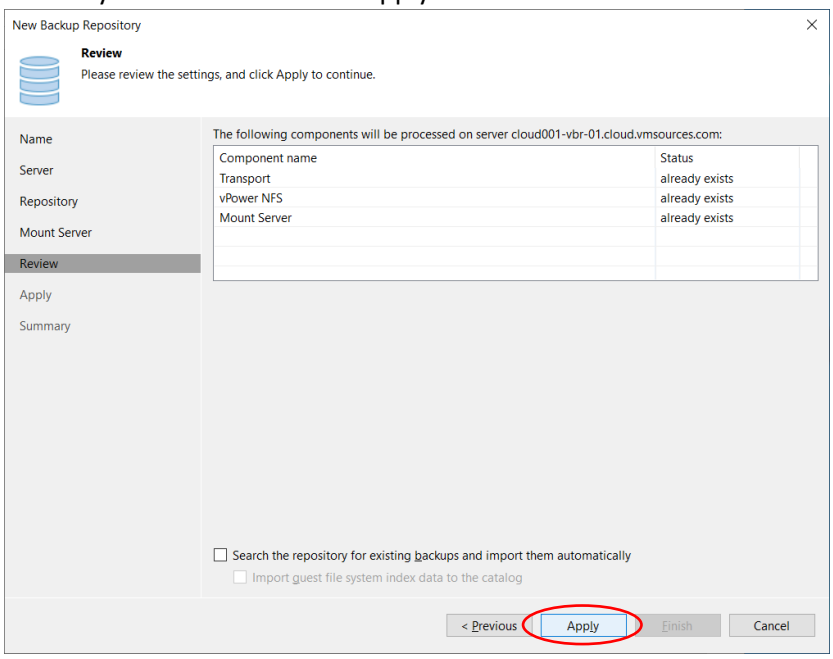
iv. Make recent backups immutable for…. (set your desired number)

v. Set “Limit maximum concurrent tasks to” equal to the number of vCPUs which you assigned the repository VM

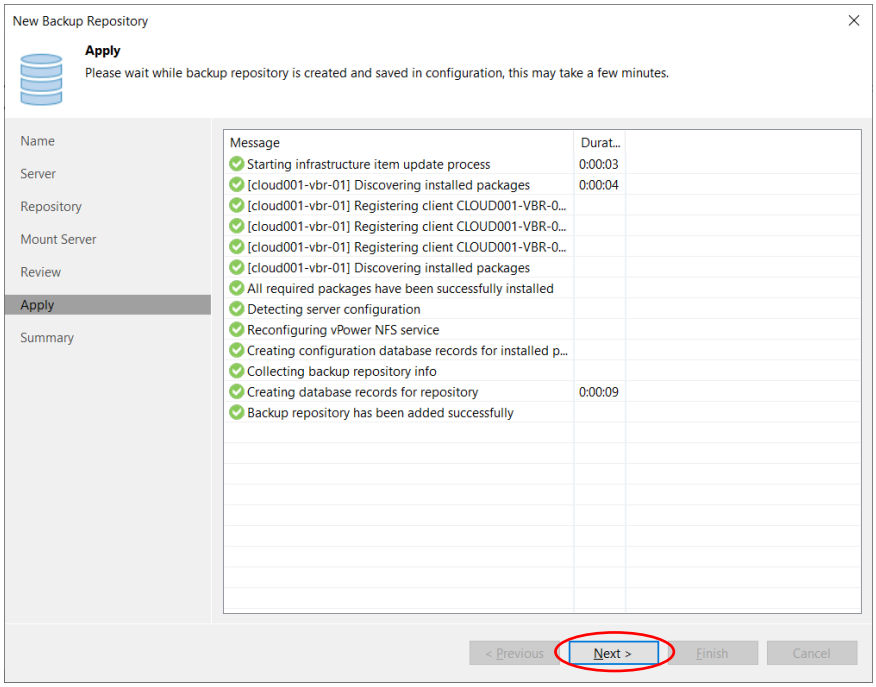
j. Choose a write cache folder and click Next.



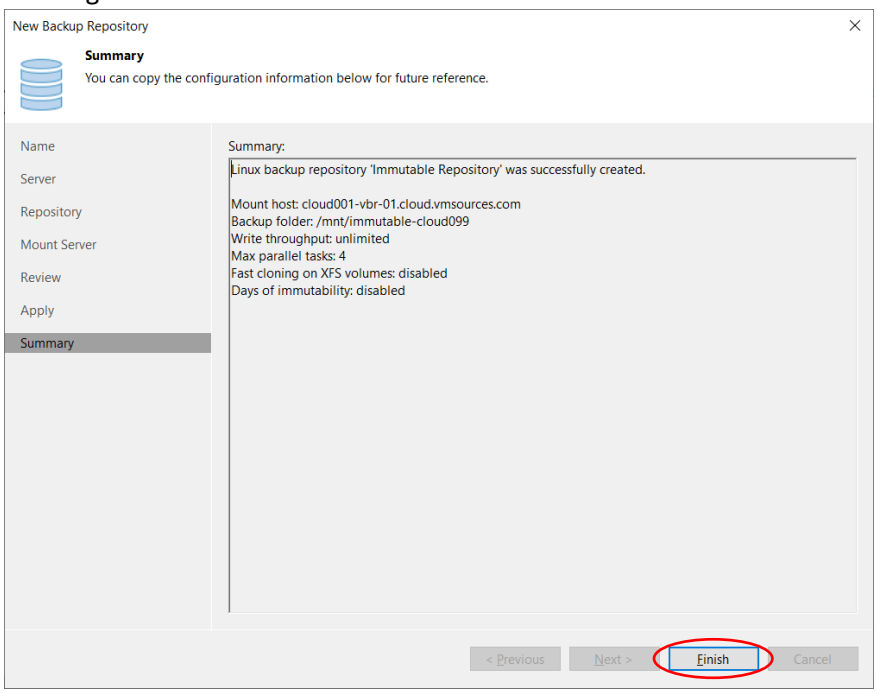
k. Review your choices and click Apply.



l. After it's finished doing its thing, click Next.



m. Nothing left to do but click Finish!



Additional Measures

Configure SSH to use two-factor authentication

https://ubuntu.com/tutorials/configure-ssh-2fa#1-overview

References

https://ubuntu.com/server/docs/service-iscsi

https://netplan.io/examples/

https://www.veeam.com/blog/v11-immutable-backup-storage.html