Using SUSE Enterprise Storage for Veeam Backup and Restore

Architecture & Implementation for the RBD Protocol

Written by:

David Byte, SUSE

Contents

Introduction

The objective of this document is to explain how to implement SUSE Enterprise Storage as a backup target for Veeam. For best results, please read this entire document and the supplemental appendix information before attempting the process. The end result should provide a working Veeam backup environment that leverages block storage provided by SUSE Enterprise Storage via RBD. This document will discuss the architecture used for the Veeam Ready testing and the associated tuning data.

Target Audience

The target audience for this document is IT staff or consultants responsible for implementing SUSE Enterprise Storage as a backup target for Veeam. Reasonable administrative knowledge of Linux and SUSE Enterprise Storage and operational knowledge of Veeam is assumed.

Solution Description

The solution outlined in this guide enables a customer to deploy a disk-to-disk target that is orchestrated through Veeam. SUSE Enterprise Storage can be utilized as a backup target via a Veeam proxy over a common network, and the result is a high-performing and flexible backup target with exabyte scalability.

SUSE Enterprise Storage delivers savings for a Veeam disk-to-disk backup solution with:

- Commodity hardware for minimal hardware cost
- Open source software, for minimal software cost and maximum flexibility
- A self-managing, self-healing architecture for minimal management cost
- A flexible, cluster-based design for graceful and inexpensive upgrade and innovative licensing model that avoids per-gigabyte storage charges, so you won't owe more for saving more data.
- With minimal acquisition cost, management cost, and upgrade cost, SUSE Enterprise Storage is the lowest-price solution for enterprise archive and backup implementations

Requirements

Enterprise backup environments require stability, long-term data durability, scalability and simple management. Establishing these requirements enables the business to consistently meet its data-protection needs and comply with data-recovery requirements often mandated by operational needs or governmental regulations.

Functional requirements

The solution has the following requirements:

- Simple to setup and deploy and meets the documented guidelines for system hardware, networking and environmental prerequisites.
- Adaptable to the physical and logical constraints needed by the business, both initially and as needed over time for performance, security, and scalability concerns.
- Resilient to changes in physical infrastructure components, caused by failure or required maintenance.
- Capable of providing optimized object and block services to client access nodes, either directly or through gateway services.
- Data protection configurable to meet the customer's individual needs at a granular level.
- Meet the Veeam Ready performance guidelines.

Architectural overview

Solution Architecture - RBD

SUSE Enterprise Storage can be used as a target for backups via a Veeam Linux Repository. The architecture and settings described below were used as part of the Veeam Ready testing to meet the performance requirements specified by Veeam. Older components were utilized in an effort to define a minimum set of hardware resources required to achieve the needed performance metrics.

The architecture used to achieve compliance with the Veeam Ready program utilizes a RADOS Block Device on a Veeam Linux Repository. This paper will also briefly discuss CephFS as a potential storage mechanism, though it was not utilized for testing.

Testing Environment Specifications

This architecture assumes an understanding of SUSE Enterprise Storage as presented in the SUSE Enterprise Storage Technical Overview document available online, which presents the concepts behind software-defined storage and Ceph. It is also assumed that you're familiar with the concepts and architecture of Veeam and cloud storage as described in the Veeam documentation

Software Basline

- SUSE Enterprise Storage 5.5
- Windows 2016 Backup Server virtual guest
- ESXi 6.7
- SLES 12 SP3 virtual guest
- Veeam Backup & Replication 9.5.0.1922

Hardware/VM Configuration

ESX Server

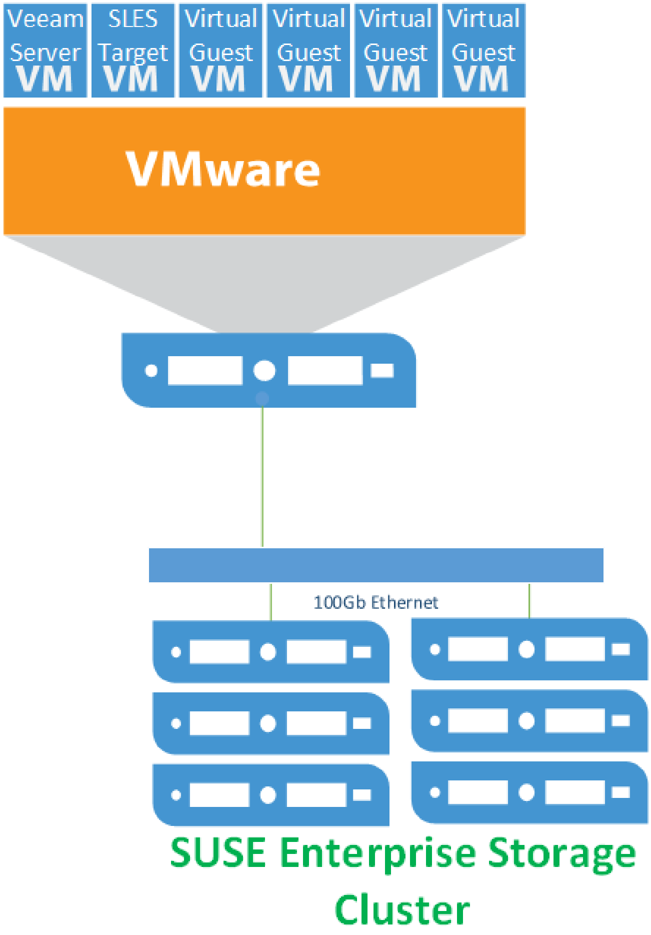
- 1x Intel Xeon Silver 4110 CPUs
- 64GB RAM
- Mellanox ConnectX-4 100Gb
- 4 Micron 960GB SATA SSD

6x SUSE Enterprise Storage OSD Nodes

- 1x Intel Xeon Silver 4116 CPU
- 96 GB RAM
- 1x 960GB SSD (Boot/OS)
- 10x Micron 960GB SSDs
- 1x Mellanox ConnectX-4 100Gb

SUSE Linux Veeam Repository Virtual Machine

- 8 vCPU (1 socket, 8 core configuration)
- 16GB RAM (set for fully reserved)
- 1 VMXNet3 vNICs on 100Gb



Veeam Backup Server & VMware Proxy

- 10 vCPU (non shared setting under advanced settings and 8GHz reserved)
- 24 GB RAM (set for fully reserved)
- 1x VMXNet3 vNICs
- 1x extra virtual HDD on paravirtual SCSI controller

Network

- 1x 100GbE switch

Network Configuration

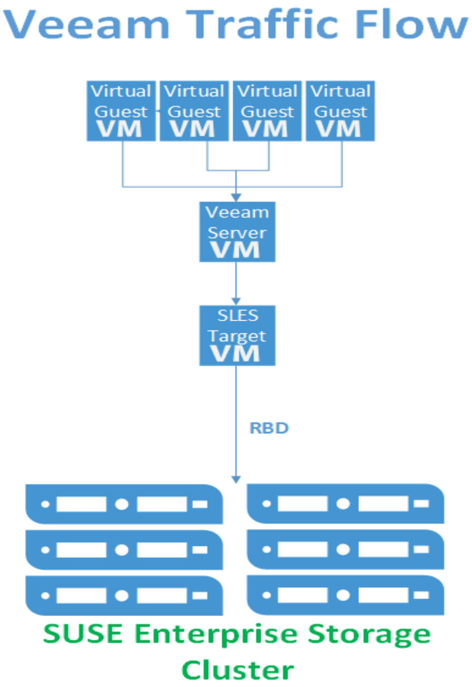
The network uses a default configuration with an MTU of 1500 and no special connections in place. The Linux hosts were not tuned except where noted.

Architectural Notes & Discussion

In this setup, both the proxy and the Linux Repository server are local to the ESX server, which offers several benefits:

1. The Veeam Proxy server (located on the backup server) was utilized to directly mount the VMware snapshot images, resulting in the highest possible streaming read performance for the backups. The figure to the right illustrates the traffic flow of a backup process.
2. Network communication from the Veeam Proxy to the Linux Repository server flowed across the ESX server without traversing the physical network infrastructure. This resuted in very high network performance between these two critical pieces of infrastructure for the Veeam Backup and Restore environment.

An optimal configuration for a large site would include Veeam Proxy and targets on each ESX host. This being not feasible for many customers, it is recommended that at least a Veeam Proxy be present where it is able to perform a vmware native mount of each system being backed up.



Pool Configuration

When configuring the SUSE Enterprise Storage cluster for use as a backup target, the data protection scheme is an important consideration. There are two main options for data protection, each with advantages and disadvantages.

The first is replication. It works by replicating each data chunk on each of the specified number of unique devices. The default is three. If the failure domain is assumed to be at the storage host level, this means the cluster could survive the loss of two storage servers without data loss. The downside of replication is the space overhead, which is 200% or two-thirds of the total cluster capacity.

The second scheme is erasure coding (EC). It works by splitting the data into the specified number of chunks (k) and then performing a mathematical calculation to create the requested number of EC chunks (m). Again, assuming the failure domain is at the host level, a system using an EC scheme of k=6, m=3 has an overhead of only 50%, or one-third of the total cluster capacity. Because EC actually writes less data, it is sometimes faster than replication for writes, but slower on the reads due to the requirement to reassemble the data from multiple nodes.

Another aspect to consider is the total cluster size. In general, it is not recommended to use EC with a cluster of fewer than seven storage nodes. When using EC with SUSE Enterprise Storage, it is recommended that the data chunks + (2x erasure coding chunks) is less than or equal to the cluster node count. Expressed in a formula:

```
data chunks [k] + (coding chunks [m] * 2) <= cluster node count
```

A cluster size of seven would thus allow for *3 data chunks* + *2 erasure coding chunks* + *2 spare nodes* to allow for device failures. In a larger cluster, EC profiles of 8+3, 6+4, 9+3 and the like are not uncommon.

An additional consideration is the availability of hardware accelerators for erasure coding. Intel CPUs provide such an accelerator, which is specified with the plugin option when creating the erasure coding profile for the pool.

```
# ceph osd erasure-code-profile set veeam_ec plugin=isa k=8 m=3
```

For the tested environment, an EC profile configured with the ISA plugin and specifying k=4, m=2 was utilized, though a production environment would need more nodes to safely accommodate this configuration.

Ceph protocol – RBD

The RBD protocol is the native block protocol for Ceph. Clients leveraging RBD could be termed “intelligent” because they are able to leverage the CRUSH algorithm to determine where data will be placed and thus communicate directly to each individual storage device. The result is performance that scales horizontally with the cluster.

As a client protocol, RBD has numerous tuning options that can be controlled on each client, or for the cluster as a whole. These include things like caching type, size, etc. For this effort, some tuning was performed for the caching parameters to optimize performance for the I/O patterns being tested. These are outlined in the deployment section below.

The Veeam Linux Repository maps the RBD device created as a block device and then a file system is placed on it. This allows for tuning that can be applied to the particular filesystem you plan to use and to accelerate performance.

Ceph protocol – CephFS

While not explicitly tested for Veeam Ready performance testing, CephFS, the distributed file system, is available for use with Veeam as well. Anecdotal testing indicates performance of nearly the same level as RBD. An advantage of this particular protocol choice is that multiple repositories can be hosted on the same massively scaleable distributed file system. This also means that if a backup server disappears or fails, it is quite simple to add the repository to another server.

Deployment

This section outlines the steps required to deploy an environment similar in architecture to the tested environment.

Deploy SES

- Build and deploy a SUSE Enterprise Storage Cluster as described in the SUSE Enterprise Storage Deployment Guide (https://www.suse.com/documentation/suse-enterprise-storage-5/book_storage_deployment/data/book_storage_deployment.html)
- Create an EC profile from command line on the admin node:

```
# ceph osd erasure-code-profile set veeam_ec plugin=isa k=4 m=2
```

- Create a pool

```
# ceph osd pool create ecpool 512 512 erasure veeam_ec
```

- Create the RBD

```
# rbd create reppool/veeam -size 5T -data-pool ecpool
```

Create virtual machines on ESX

- Configure resource reservations

Configure Linux Repository virtual machines

- Perform base linux install
  - Select KVM Host install pattern
  - Unselect KVM Host from Software Selection on Summary Screen
- Disable spectre/meltdown
  - Information regarding diabling Spectre and Meltdown mitigations can be found here: https://www.suse.com/support/kb/doc/?id=7023480
  - Disabling these mitigations on the storage nodes and the Linux target results in enhanced performance.
- Enable Multi-queue block IO
  - Do this on the Ceph OSD nodes and the Linux target VM(s)
  - Information on blk-mq on enabling it can be found here: https://www.suse.com/documentation/sles-12/book_sle_tuning/data/cha_tuning_io_scsimq.html
- Set network tuning parameters in /etc/sysctl.conf' for SUSE Enterprise Storage nodes and Linux target

```
net.ipv4.ip_forward = 0
net.ipv6.conf.all.forwarding = 0
net.core.netdev_max_backlog = 10000
net.core.netdev_budget = 300
net.core.somaxconn = 128
net.core.busy_poll = 50
net.core.busy_read = 50
net.core.rmem_max = 125829120
net.core.wmem_max = 125829120
net.core.rmem_default = 125829120
net.core.wmem_default = 125829120
net.ipv4.tcp_fastopen = 1
net.ipv4.tcp_low_latency = 1
net.ipv4.tcp_sack = 1
net.ipv4.tcp_rmem = 10240 87380 125829120
net.ipv4.tcp_wmem = 10240 87380 125829120
net.ipv4.ip_local_port_range = 1024 64999
net.ipv4.tcp_max_syn_backlog = 1024
net.ipv4.tcp_tw_reuse = 0
net.ipv4.tcp_tw_recycle = 0
net.ipv4.tcp_timestamps = 0
net.ipv4.tcp_syn_retries = 5
```

- Add repositories and packages for veeam
  - Veeam requires perl modules be present for the Linux repository to function. These are detailed in: https://www.veeam.com/kb2216

- Modify /etc/ssh/sshd_config to enable Veeam service to work correctly
  - https://www.veeam.com/kb1512
  - Find the PasswordAuthentication parameter and set the value to yes
  - Save and restart the sshd daemon

```
# systemctl restart sshd.service
```

- To add perl-SOAP-Lite the SDK repos will need to be added.
  - For SLES12SP3

```
# SUSEConnect -p sle-sdk/12.3/x86_64
# zypper in perl-SOAP-Lite
```

  - For SLES15

```
# SUSEConnect -p PackageHub/15/x86_64
# zypper in perl-SOAP-Lite
```

- The following script can be used to validate that all packages/perl modules are installed

```
#!/bin/bash
for i in constant Carp Cwd Data::Dumper Encode Encode::Alias Encode::Config
Encode::Encoding Encode::MIME::Name Exporter Exporter::Heavy File::Path
File::Spec File::Spec::Unix File::Temp List::Util Scalar::Util SOAP::Lite
Socket Storable threads
do
   echo "Checking for perl $i;..."
   perldoc -lm $i >/dev/null
   perlpkgfound=$?
   if [ ! $perlpkgfound -eq 1 ]
   then
      echo Installed
   fi
done
```

- Add ceph-common to the Linux target

```
# zypper in ceph-common
```

- Add client key and ceph.conf to /etc/ceph
  - From the admin node:

```
# scp /etc/* root@vtarget:/etc/ceph/
```

- Edit /etc/ceph/rbdmap and add the rbd

```
# RbdDevice Parameters
# poolname/imagename   id=client,keyring=/etc/ceph/ceph.client.keyring
reppool/veeam    id=admin,keyring=/etc/ceph/ceph.client.admin.keyring
```

- Enable and started system rbdmap service

```
 # systemctl enable rbdmap
 # systemctl start rbdmap
```

- Mkfs.xfs the target

```
# mkfs.xfs /dev/rbd0
```

- Add mount point

```
# mkdir /veeam
```

- Add entry to fstab (include any tuning desired)

```
# /dev/rbd0 /veeam xfs _netdev 1 1
```

- Mount the filesystem

```
# mount -a
```

- Verify it mounted

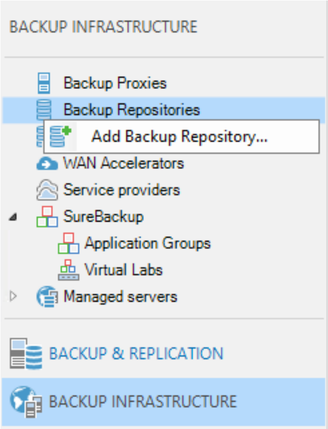
```
# mount
```

Output should be:

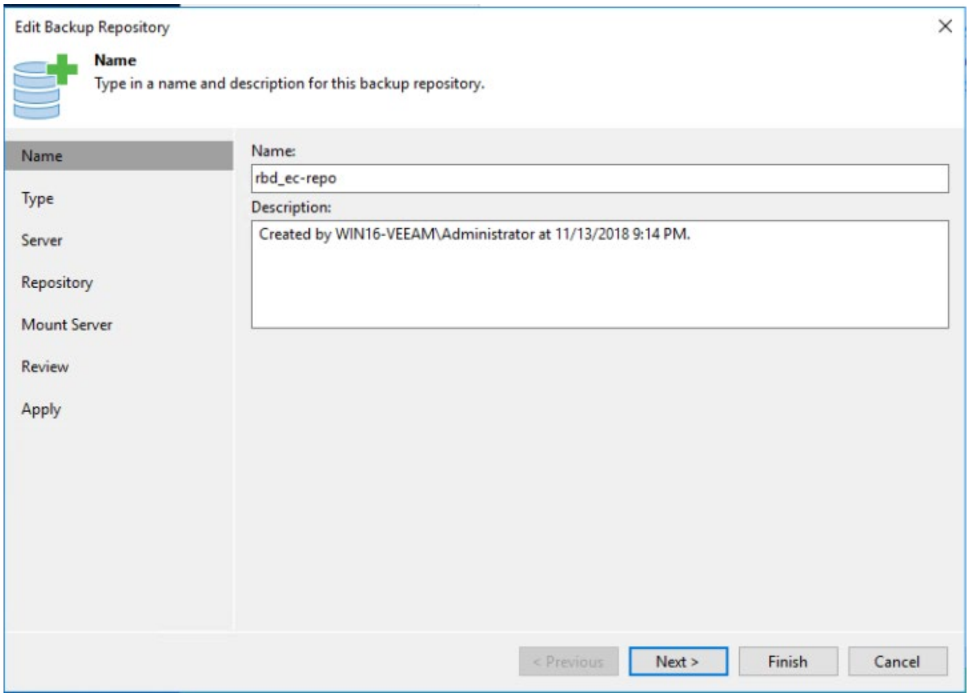
```
/dev/rbd0 on /veeam type xfs
(rw,relatime,attr2,inode64,sunit=8192,swidth=8192,noquota,_netdev)
```

Add Veeam Linux Repository

1. Within the Veeam Console, click **Backup Infrastructure** on the left-hand menu bar. Right-click on **Backup Repositories** followed by **Add Backup Repository**



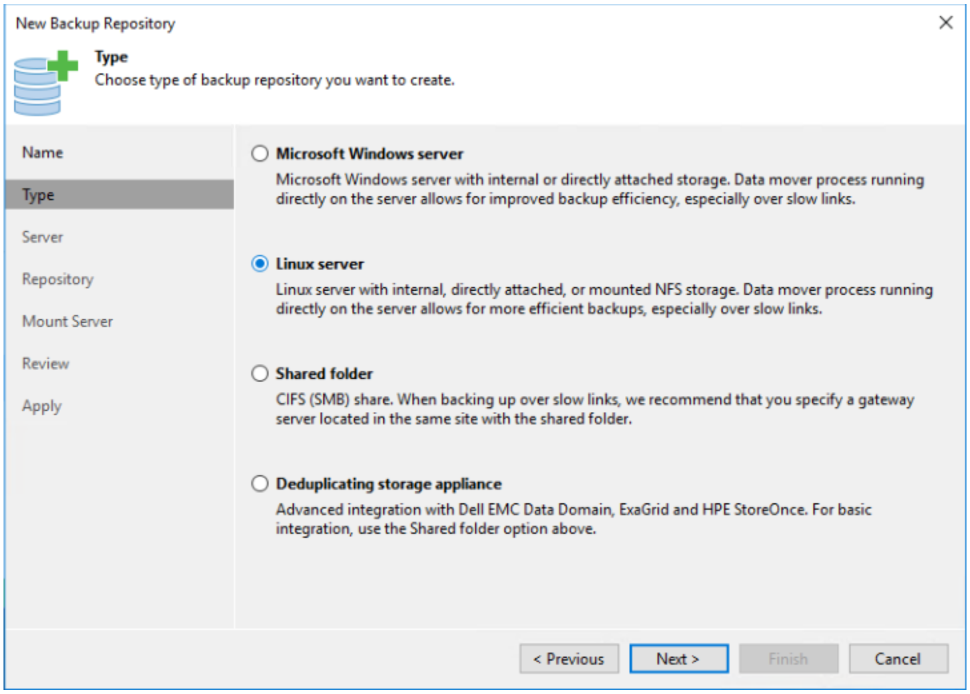
2. Provide a friendly **name** to distinguish the multiple repositories.



Name: rbd_ec-repo

Description: Created by WIN16-VEEAM\Administrator at 11/13/2018 9:14 PM.

3. Choose a **repository** type and click next.


New Backup Repository

Type
Choose type of backup repository you want to create.

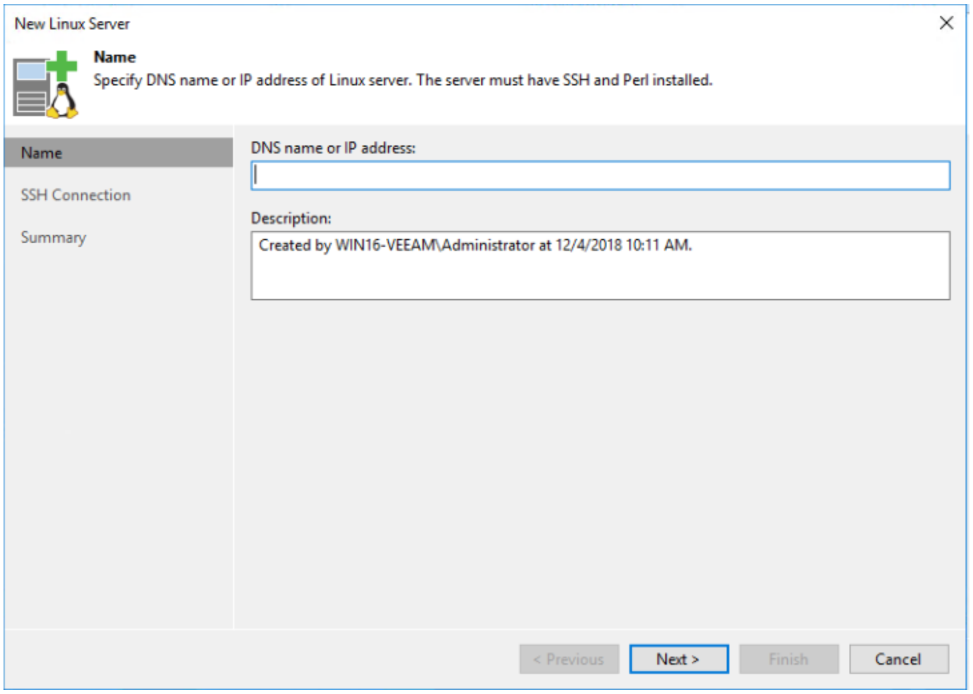
○ **Microsoft Windows server**
Microsoft Windows server with internal or directly attached storage. Data mover process running directly on the server allows for improved backup efficiency, especially over slow links.

● **Linux server**
Linux server with internal, directly attached, or mounted NFS storage. Data mover process running directly on the server allows for more efficient backups, especially over slow links.

○ **Shared folder**
CIFS (SMB) share. When backing up over slow links, we recommend that you specify a gateway server located in the same site with the shared folder.

○ **Deduplicating storage appliance**
Advanced integration with Dell EMC Data Domain, ExaGrid and HPE StoreOnce. For basic integration, use the Shared folder option above.

4. Click Add New and enter the details and click next.

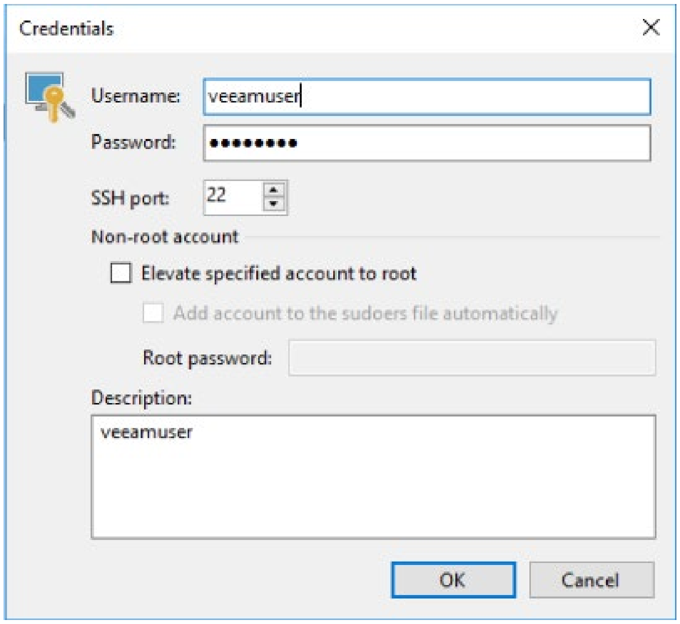

New Linux Server

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

DNS name or IP address:

Description:
Created by WIN16-VEEAM\Administrator at 12/4/2018 10:11 AM.

5. Click Add to add credentials that have Read, Write, and Execute permissions to the mounted storage location and the ability to execute Perl code, and click OK and then Finish.



Username: veeamuser

Password: ●●●●●●●●

SSH port: 22

Non-root account

☐ Elevate specified account to root

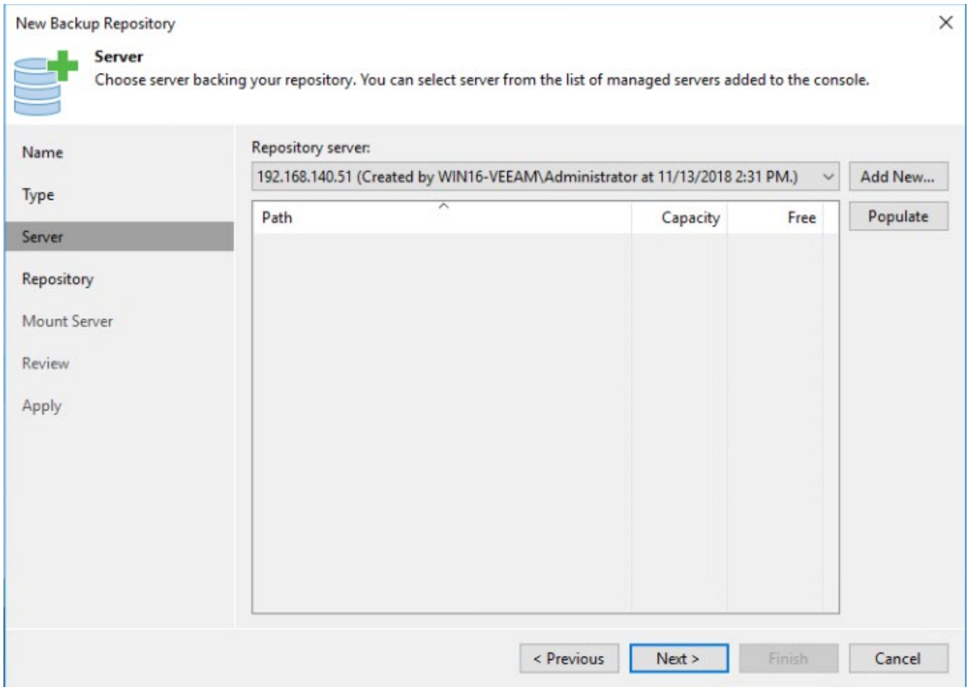
☐ Add account to the sudoers file automatically

Root password:

Description:

veeamuser

6. Ensure the credentials are selected and click next.



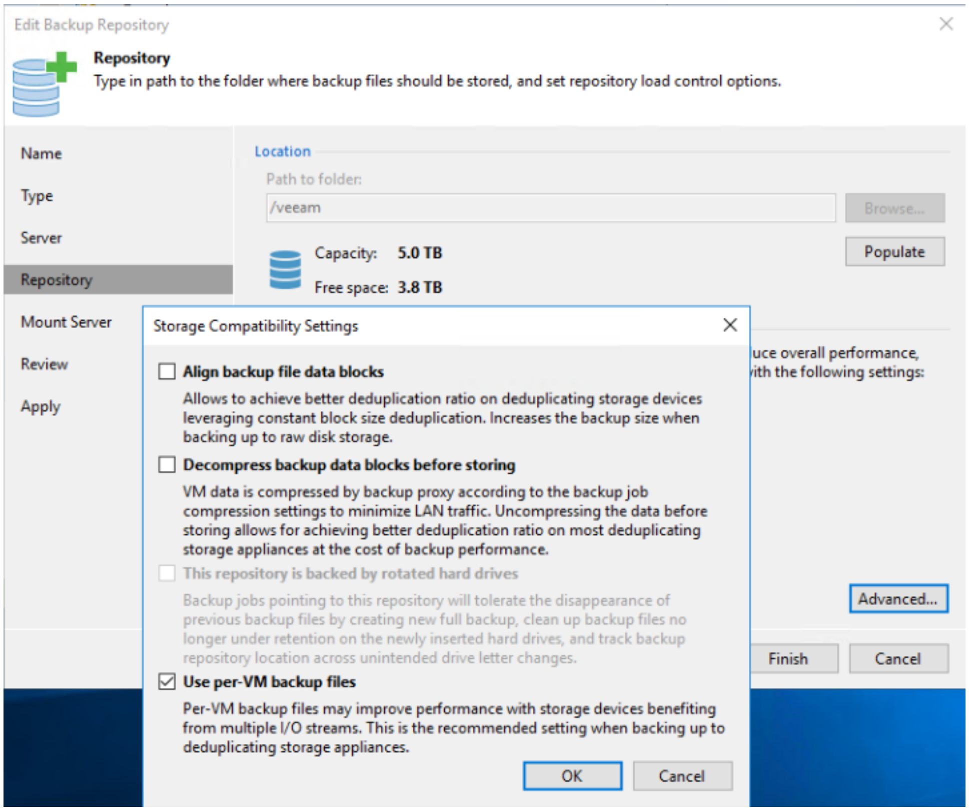
Server

Choose server backing your repository. You can select server from the list of managed servers added to the console.

Repository server:

192.168.140.51 (Created by WIN16-VEEAM\Administrator at 11/13/2018 2:31 PM.)

7. Click Browse and select the path to the mounted RBD with the XFS filesystem, and then click Advanced to select Use per-VM backup files.



Repository

Type in path to the folder where backup files should be stored, and set repository load control options.

Path to folder: /veeam

Capacity: **5.0 TB**

Free space: **3.8 TB**

uce overall performance,
ith the following settings:

☐ **Align backup file data blocks**

Allows to achieve better deduplication ratio on deduplicating storage devices leveraging constant block size deduplication. Increases the backup size when backing up to raw disk storage.

☐ **Decompress backup data blocks before storing**

VM data is compressed by backup proxy according to the backup job compression settings to minimize LAN traffic. Uncompressing the data before storing allows for achieving better deduplication ratio on most deduplicating storage appliances at the cost of backup performance.

☐ This repository is backed by rotated hard drives

Backup jobs pointing to this repository will tolerate the disappearance of previous backup files by creating new full backup, clean up backup files no longer under retention on the newly inserted hard drives, and track backup repository location across unintended drive letter changes.

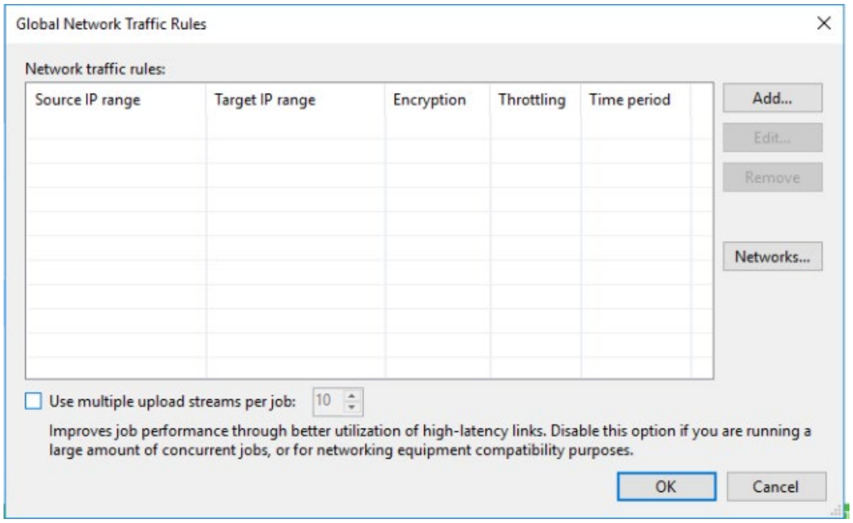
☑ **Use per-VM backup files**

Per-VM backup files may improve performance with storage devices benefiting from multiple I/O streams. This is the recommended setting when backing up to deduplicating storage appliances.

8. Finish the process by selecting a Mount server (Veeam Backup Server or proxy) and enabling a vPower NFS service as desired and selecting Finish

Disable Multiple Streams

Multiple streams are designed to enhance performance for higher latency environments. It may be desireable to disable this for the local deployment. This can be done when defining the job, by setting it for the proxy, or globally. In all cases, it involves selecting the Network Traffic Rules and de-selecting Multiple Streams.



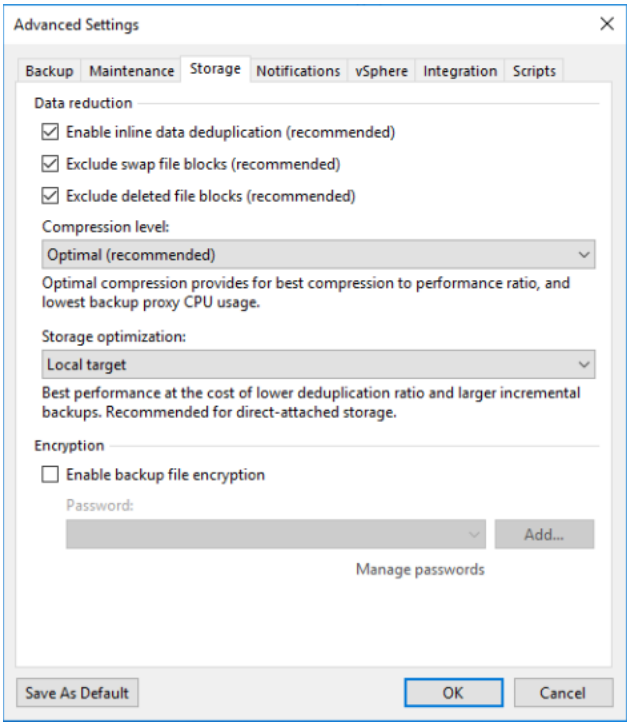
Source IP range	Target IP range	Encryption	Throttling	Time period

☐ Use multiple upload streams per job: 10

Improves job performance through better utilization of high-latency links. Disable this option if you are running a large amount of concurrent jobs, or for networking equipment compatibility purposes.

Define a Backup Job

Create a backup job. When on the Storage setting tab, select the correct proxy and repository. While on the Storage screen, select Advanced. On the Storage tab, set the appropriate rules for your environment. Veeam Ready testing was performed using the information shown at the right.



Data reduction

☑ Enable inline data deduplication (recommended)

☑ Exclude swap file blocks (recommended)

☑ Exclude deleted file blocks (recommended)

Compression level: Optimal (recommended)

Optimal compression provides for best compression to performance ratio, and lowest backup proxy CPU usage.

Storage optimization: Local target

Best performance at the cost of lower deduplication ratio and larger incremental backups. Recommended for direct-attached storage.

Encryption

□ Enable backup file encryption

Password:

Special Notes

If performing numerous tests, it may be necessary to go to the Linux target host via SSH and run `fstrim` for the mounted filesystem to force garbage collection instead of allowing the default actions of lazy garbage collection.

In the Veeam Ready Testing, a performance issue was encountered when using SSDs directly on the ESX server as a restoration target. In this case, it was necessary to disable VAAI for VMware to perform optimally.

- This is to resolve a problem that may occur with VMware writing to images hosted on VMFS. The symptom that indicates this may be needed is when iostat -xmt 1 on OSD nodes and on vtarget indicate very low utilization during a Veeam restore operation. https://kb.vmware.com/s/article/1033665
- In the Veeam Ready testing performed, adjusting this parameter enabled the restore tests to reach about 100MB/s per guest being restored. The limitation in that case was the device being written to.

Additional Resources

- Veeam KB – SUSE KB Articles
  https://www.veeam.com/kb_search_results.html?product=Backup_Replication&kb-search-type=&search=suse
- Veeam Documentation
  https://www.veeam.com/documentation-guides-datasheets.html
- SUSE Enterprise Sftorage Technical Overview
  https://www.suse.com/docrep/documents/1mdg7eq2kz/suse_enterprise_storage_technical_overview_wp.pdf
- SUSE Enterprise Storage v5 – Deployment Guide
  https://www.suse.com/documentation/suse-enterprise-storage-5/pdfdoc/book_storage_deployment/book_storage_deployment.pdf
- SUSE Enterprise Storage v5 – Administration Guide
  https://www.suse.com/documentation/suse-enterprise-storage-5/pdfdoc/book_storage_admin/book_storage_admin.pdf