PURESTORAGE®

TECHNICAL WHITE PAPER

Enhancing Veeam with FlashArray//C

Achieve faster backup and restore without complexity.

Contents

Introduction

We can all agree that backup isn't glamorous. It's easily overshadowed by the many exciting developments around containers, AI, and myriad other new and shiny technologies. But it's still a critical underpinning to business success. With ransomware and data breaches growing more common all the time—Bitdefender measured a 715% increase in attacks in 2020—it's more important than ever to have a backup and recovery solution that can ensure your data is protected quickly and get you back online faster in the event of a compromise.

Enter Pure Storage® FlashArray//C and Veeam. FlashArray™ enhances the simple protection and recovery experience of Veeam Backup & Replication with the speed of flash and efficient data reduction to give you a powerful, scalable backup platform that's easy to deploy and manage. FlashArray SafeMode™ snapshots mitigate against attacks on your backup data by ransomware, rogue admins, and other bad actors. And it's delivered at QLC economics, with the future-proofing of Evergreen™ Storage.

This white paper is intended as a how-to and best practices guide to assist with the design and implementation of Pure Storage FlashArray//C into Veeam Backup & Replication environments. The target audience for this document includes, but is not limited to, system architects, systems engineers, IT managers, and storage administrators.

Solving Backup and Recovery Challenges

Backup and recovery have never been more critical to businesses. As data continues to grow exponentially, accelerated by the COVID-19 pandemic, and ransomware attacks grow ever more common and sophisticated, the ability to recover data and applications quickly can mean the difference between a temporary disruption and major hit, or even bankruptcy. As the statistics for cyberattacks continue to rise in 2021, it's clear you need to ensure you can not only restore your services but do it quickly.

Veeam V11 and FlashArray//C can help you get more of your critical workloads back online immediately by taking advantage of the all-flash FlashArray architecture to accelerate the performance of Instant VM Recovery. That enables you to run those critical services closer to production speeds while you're getting them back on the Tier 1 systems where they usually run.

Traditional backup and recovery also benefit. FlashArray//C can help you shrink your backup window and reduce recovery time objective (RTO) for streaming restores. If you have your production VMs on FlashArray, you can use the Pure plug-in for Veeam to incorporate hardware snapshots into your backups, lessening the "stun" effect on your production VMs and hosts.

As IT budgets continue to be stretched, reducing the total cost of ownership (TCO) is essential for enabling all your technology goals. Veeam and FlashArray//C are designed to be easy to deploy and manage. Combining simplicity, disaggregated compute

and storage, flash throughput, and QLC density makes you more agile and able to react to changes and threats without breaking the bank.

FlashArray//C

FlashArray//C builds upon the industry-leading success of the FlashArray platform, extending flash performance to workloads that need high capacity more than the lowest latency. FlashArray//C optimizes data density through features such as DirectFlash® managed QLC, inline data reduction, and global deduplication, making it a perfect fit for high-performance backup storage. With integrated SMB and NFS file services, FlashArray//C can serve multiple use cases in a very small footprint. Pure Evergreen Storage ensures your environment stays simple and modern for as long as you own it. Non-disruptive upgrades of controllers and storage mean the end of forklift refreshes and migrations.

Figure 1: FlashArray//C

Veeam

Veeam Backup & Replication (VBR) is an industry-leading data protection software for small to medium businesses to large enterprises. VBR is known for easy deployment, broad compatibility, and high performance. Veeam continues to evolve the platform with improvements to performance and scale, enhanced Instant VM Recovery, and hardening for ransomware protection.

Pure and Veeam: Better Together

When you deploy VBR with FlashArray//C as its storage, you get a winning combination. FlashArray//C amplifies the performance of VBR, giving you backup and, importantly, recovery at the speed of flash. Always-on data reduction—across all your Veeam repositories on FlashArray//C —enhances the efficiency of Veeam backups, driving high storage density in a small footprint and reducing TCO without sacrificing performance. SafeMode Snapshots provide peace of mind against ransomware attacks destroying your backups. And you can deploy confidently knowing FlashArray//C is Veeam Ready qualified.

Pairing VBR with FlashArray//X for primary storage makes the story even better. VMware datastores become extremely fast without complexity. The Veeam Universal Storage API seamlessly incorporates hardware snapshots into backups to prevent VM stun. Veeam Explorers let you quickly restore data from FlashArray snapshots for lower recovery time objective (RTO).

Solution Architecture

Figure 2 illustrates the logical architecture of the solution. The source VMware vSphere VMs reside on VMFS datastores on FlashArray//X. Veeam proxies read VM data, using one of the Veeam transport modes, and transfer it to the Veeam backup repository server, which stores the data on FlashArray//C volumes. You can add more proxies as needed.

Veeam can use the backup data and snapshots for traditional restore and Instant VM Recovery operations, as well as an on-demand sandbox and SureBackup verification.

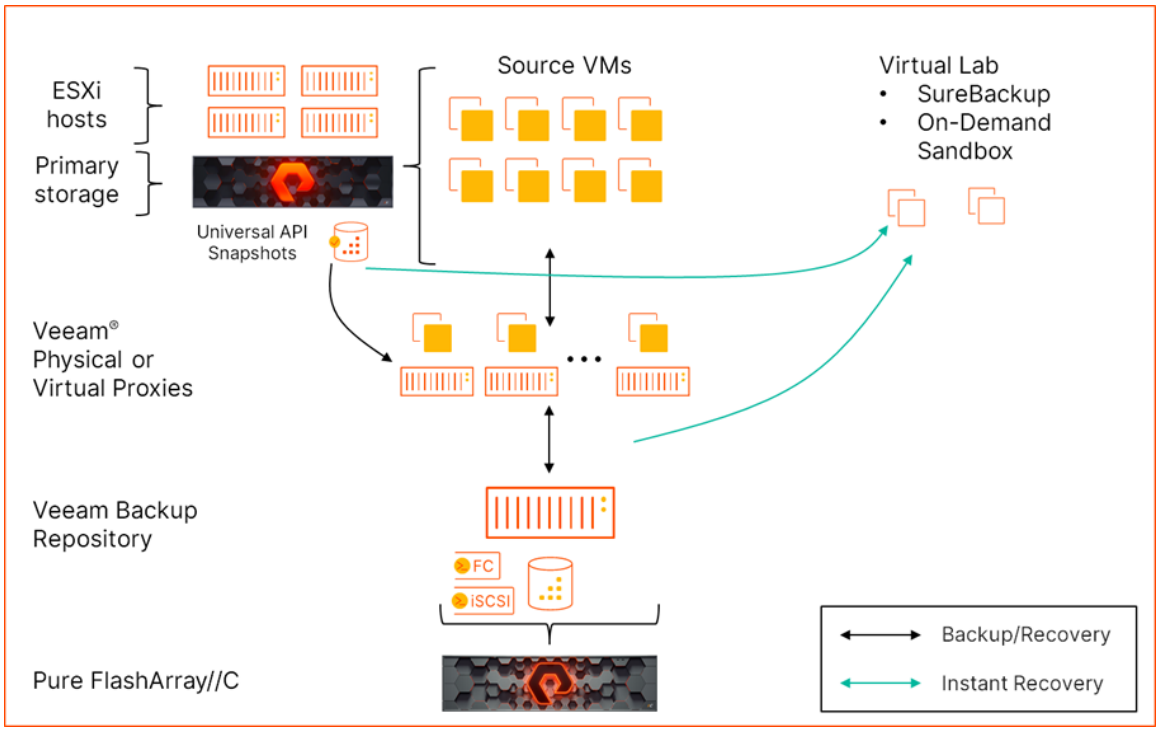


Figure 2: Solution logical architecture

Figure 3 shows the physical architecture in which all systems interconnect through an IP network. FlashArray//X serves vSphere datastores over iSCSI or Fibre Channel (FC). One or more Veeam repository servers connect to block volumes on FlashArray//C using iSCSI or FC. Proxies also access the datastores and FlashArray snapshots over iSCSI or FC by using the direct storage access transport, through vSphere datastores with the virtual appliance transport (for proxies on VMs), or through the vSphere hosts by using the NBD transport.

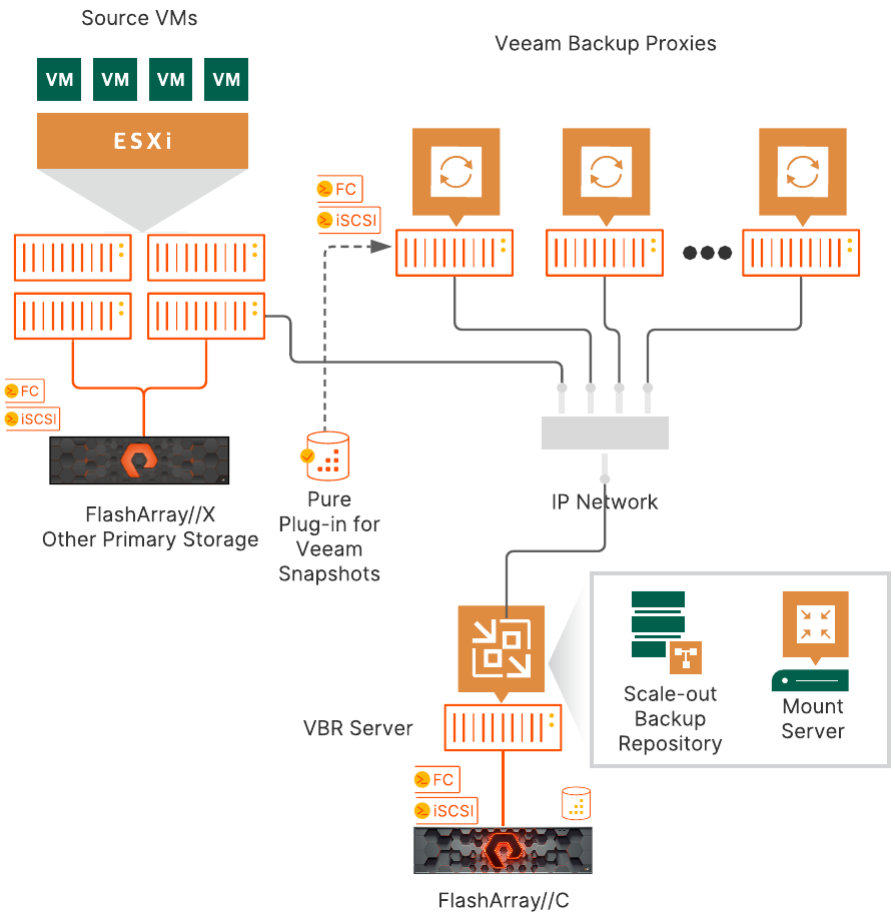


Figure 3: Solution physical architecture

Scaling

A single backup repository and mount server on the VBR server will deliver enough performance to meet backup and recovery needs for many VMware environments. One or more backup proxies will manage the data movement for backup and recovery, and you can add proxies based on your requirements. We recommend deploying FlashArray//C as a scale-out backup repository (SOBR), which you can easily grow to multiple repository volumes and servers if you need more capacity or processing power. You can also add mount servers to improve Instant VM Recovery performance.

The Veeam Help Center has detailed information about supported deployment scenarios.

Lab Setup

Server Details

For the lab testing, we used a four-node ESXi cluster, a physical VBR server, two physical backup proxies, and two virtual proxies. Table 1 shows the hardware and configuration details.

Server Role	CPU	RAM	Networking	Storage	Operating System
ESXi Host (x4)	2x Intel Xeon E5-2697 v2 @ 2.70GHz, 24 cores total HyperThreading enabled	512GB	2 x Mellanox MT27500 family network adapter @ 40Gbps	4 datastores from 3 FlashArray//M70 arrays	VMware ESXi 6.7.0
Veeam Backup Server	2x Intel Xeon E5-2697 v2 @ 2.70GHz, 24 cores total HyperThreading enabled	512GB	2 x Mellanox MT27500 family network adapter @ 40Gbps, in LACP team	256GB Toshiba SSD on 6Gb SATA iSCSI LUNs from FlashArray//C iSCSI connection to FlashArray//M70	Windows Server 2019 data center build 1809
Veeam Proxy (Physical) (x2)	2x Intel Xeon E5-2697 v2 @ 2.70GHz, 24 cores total HyperThreading enabled	512GB	2 x Mellanox MT27500 family network adapter @ 40Gbps, in LACP team	256GB Toshiba SSD on 6Gb SATA	Windows Server 2019 data center build 1809
Veeam Proxy (Virtual) (x2)	8 vCPU	20GB	vmxnet3 virtual adapter	40GB VMDK iSCSI connection to FlashArray//M70	Windows Server 2016 data center

Table 1: Server configuration details

Storage Details

A FlashArray//C60R3 hosted the Veeam backup repository. Source VMs were hosted on FlashArray//M70 arrays. All arrays were connected using 4x10GbE iSCSI.

Storage Role	Array Model	Purity Releases	Physical Storage	iSCSI Sessions
Backup Repository	FlashArray//C60R3	6.1.4; 6.1.3 (Instant VM Recovery)	195.15TB (usable)	1 per target
Data Source (x3)	FlashArray//M70	6.0.3	21TB (usable)	8 per target per ESXi host

Table 2: Storage configuration details

Source Data Details

We built 96 source VMs with the configuration shown in Table 3. The VMs were evenly distributed across the ESXi hosts and FlashArray//M70 datastores, with 24 VMs per host and 32 VMs per datastore. Each test used some part of the VM set, distributed as evenly as possible across the hosts and datastores.

VM Role	CPU	RAM	Networking	Storage	Operating System
ESXi Host (x4)	2 vCPU	4GB	vmxnet3 virtual adapter	1x100GB VMDK thin provisioned	Windows 10

Table 3: Test VM details

The test data set on each VM was a blend of unique data generated on the VM and shared data copied across the VMs, with variable compressibility between 0% and 40% compressible. Between backups, we added randomly selected data from a shared pool to generate data changes with partial overlap across the VMs. Average simulated daily change rate was 5%.

Adding FlashArray//C Volumes as Veeam Backup Repositories

To set up the backup repositories, we followed the below procedure.

Create the Repository Volume on FlashArray//C

1. Define the repository host on FlashArray//C (Figure 4).

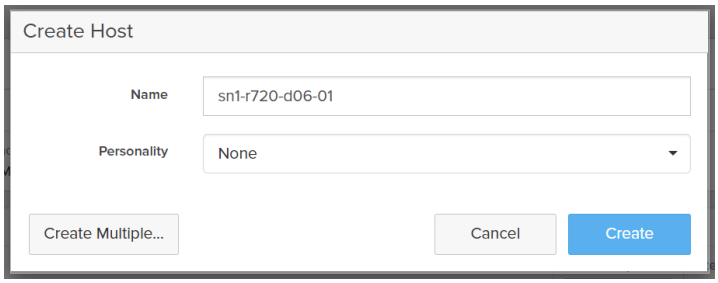


Figure 4: Creating the repository host to FlashArray//C

2. Add the host ports to the host. In our lab, we added the iSCSI iQN (Figure 5).

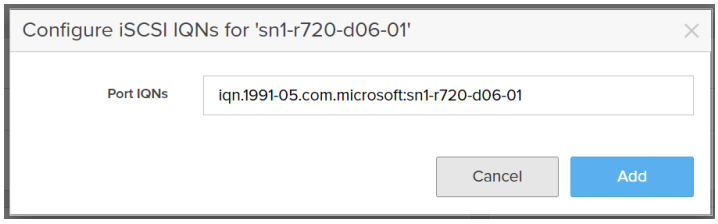


Figure 5: Adding iQN to repository host on FlashArray//C

3. Create a 200TB volume for the repository (Figure 6).

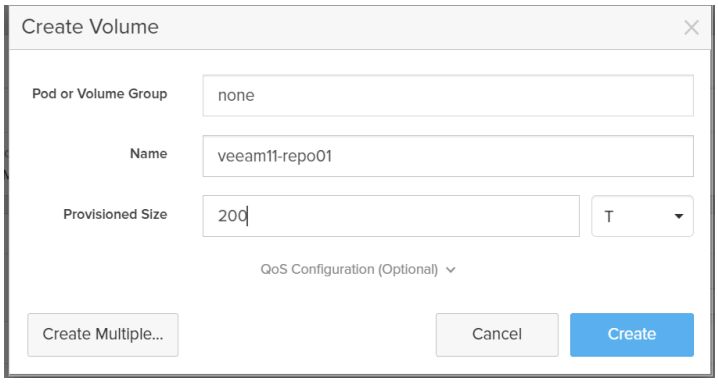


Figure 6: Creating the repository volume on FlashArray//C

Connect the volume to the repository host (Figure 7).

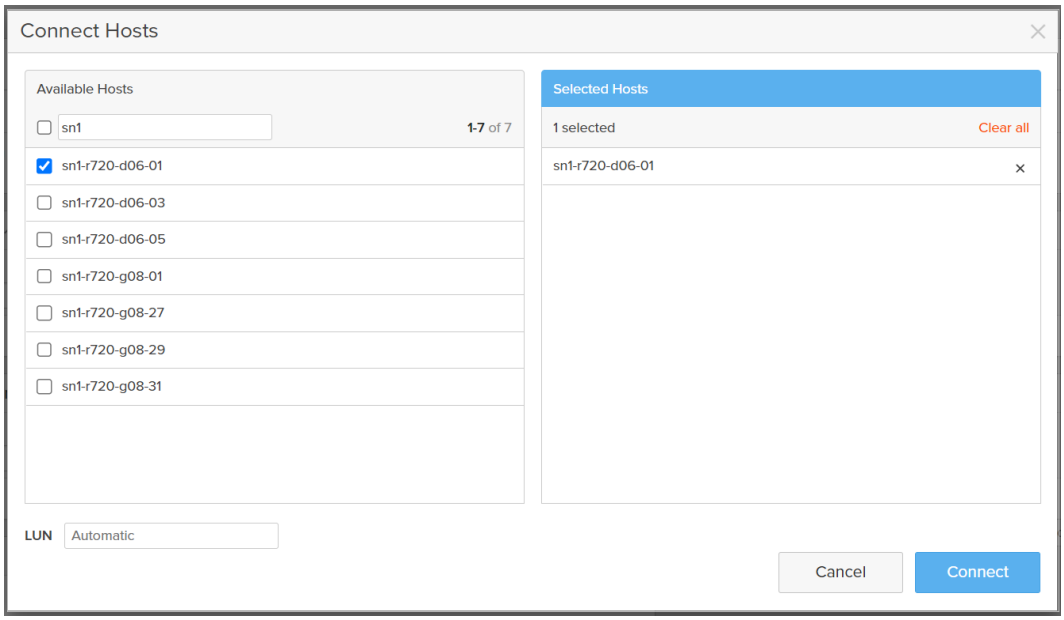


Figure 7: Connecting repository volume to host on FlashArray//C

Mount and Format Repository Volume

1. Connect the repository host to the FlashArray//C using iSCSI or Fibre Channel. For iSCSI, connect 1 session for each iSCSI target IP address on FlashArray//C (Figures 8 and 9).

   NOTE: FlashArray//C will not allow a host iSCSI access unless it is configured with at least one connected volume.

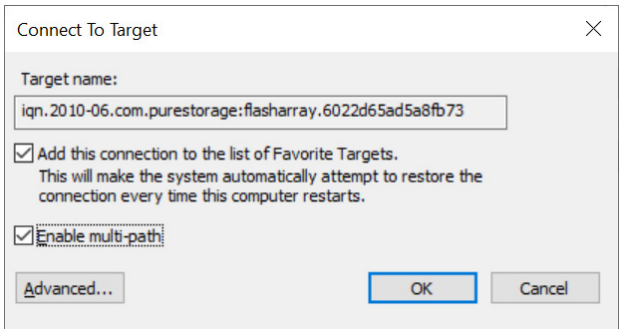


Figure 8: Enabling multiple iSCSI sessions

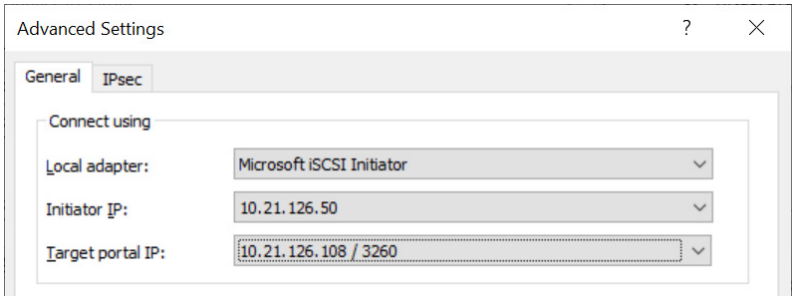


Figure 9: Defining iSCSI connection endpoints

In our lab, we ended up with four iSCSI sessions (Figure 10).

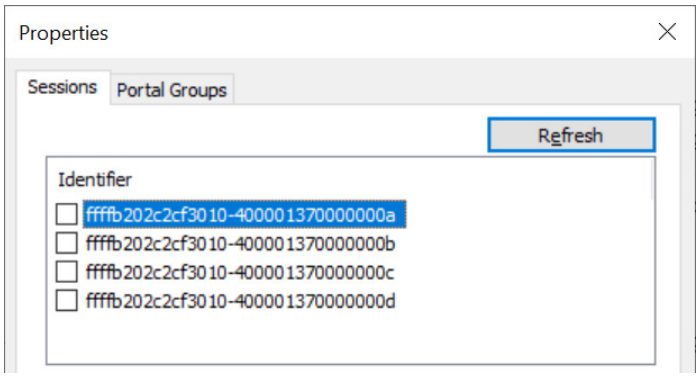


Figure 10: iSCSI sessions after setup

2. Rescan for new storage, then bring the new disk online. Create a new ReFS volume with a 64K allocation unit size. Mount the volume to a drive letter (Figure 11).

   NOTE: We recommend using Windows Server 2019, build 1809 or newer, for ReFS repositories. Previous releases were not aggressive about reclaiming capacity. You may also use NTFS, with a 64K allocation unit size, but we do not recommend running synthetic full backups on NTFS repositories. See the Storage Consumption section below for more information.

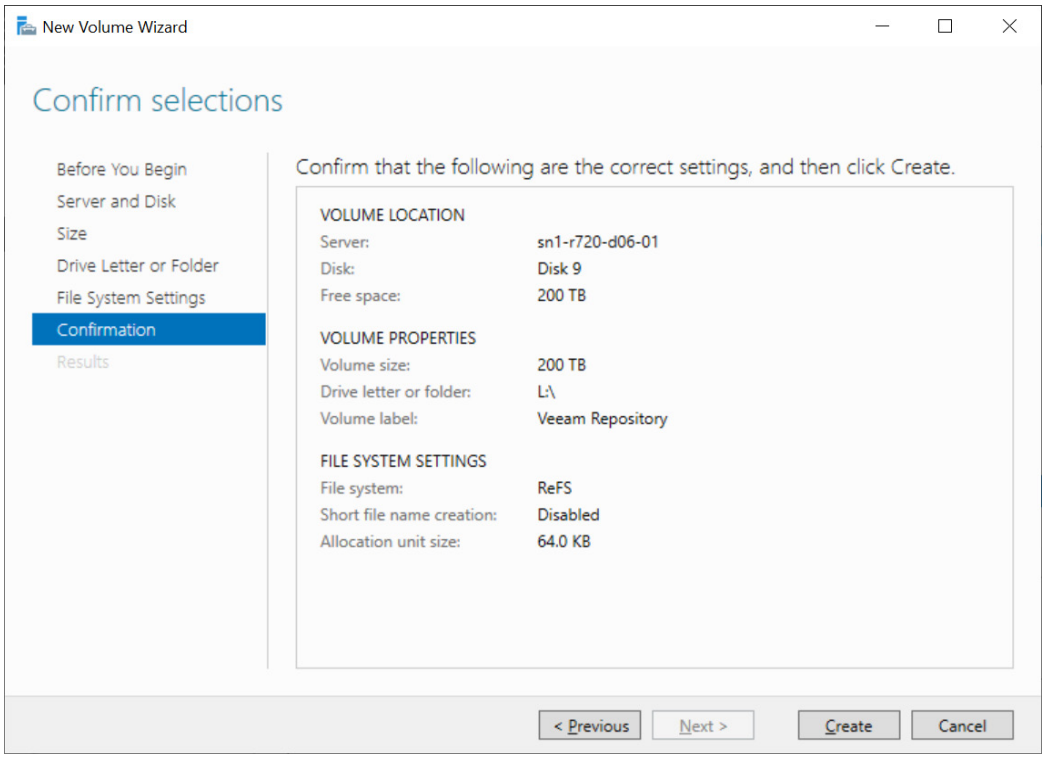


Figure 11: Creating the ReFS repository volume in Windows

Create a Veeam Scale-out Backup Repository on the Volume

1. Add a backup repository, selecting the options for direct-attached storage and Microsoft Windows. Enter a repository name and select the appropriate repository server.
2. For the repository location, enter a folder path on the new repository volume. Limit the maximum tasks to the number of physical CPU cores (Figure 12).

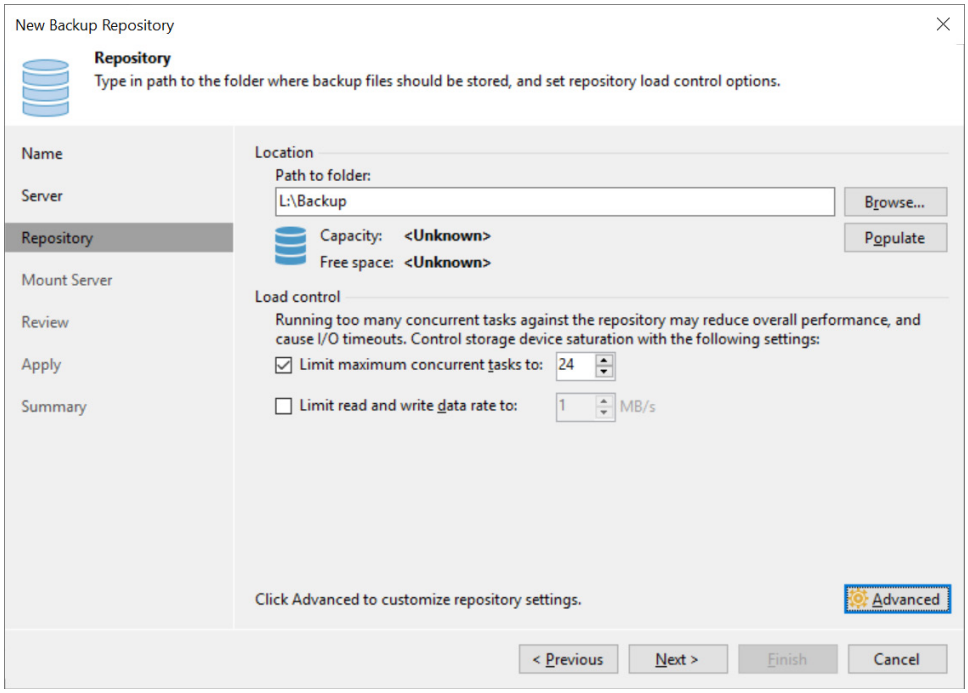


Figure 12: Limiting Veeam backup repository concurrent tasks

3. Click **Advanced** to configure the repository settings for FlashArray//C. Enable the **Align backup file data blocks**, **Decompress backup file data blocks before storing**, and **Use per-machine backup files** options, then click OK (Figure 13).

NOTE: While the SOBR will override the per-machine backup files behavior, it is good practice to set it on all FlashArray//C repositories.

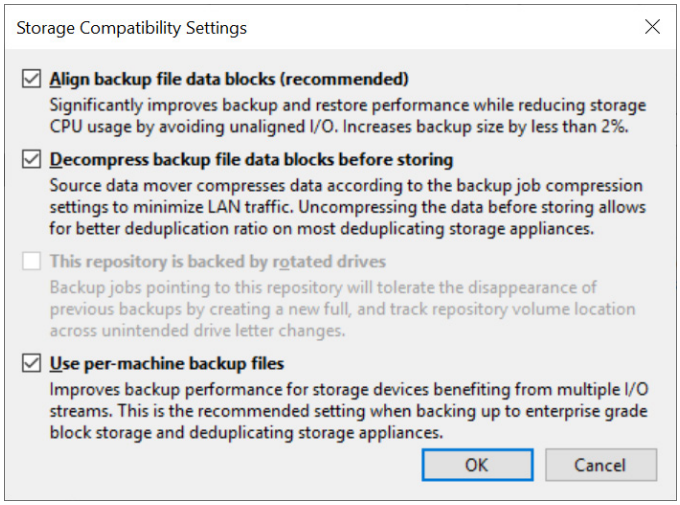


Figure 13: Advanced Veeam backup repository options

4. Set the mount server options, then review and apply the settings (Figure 14).

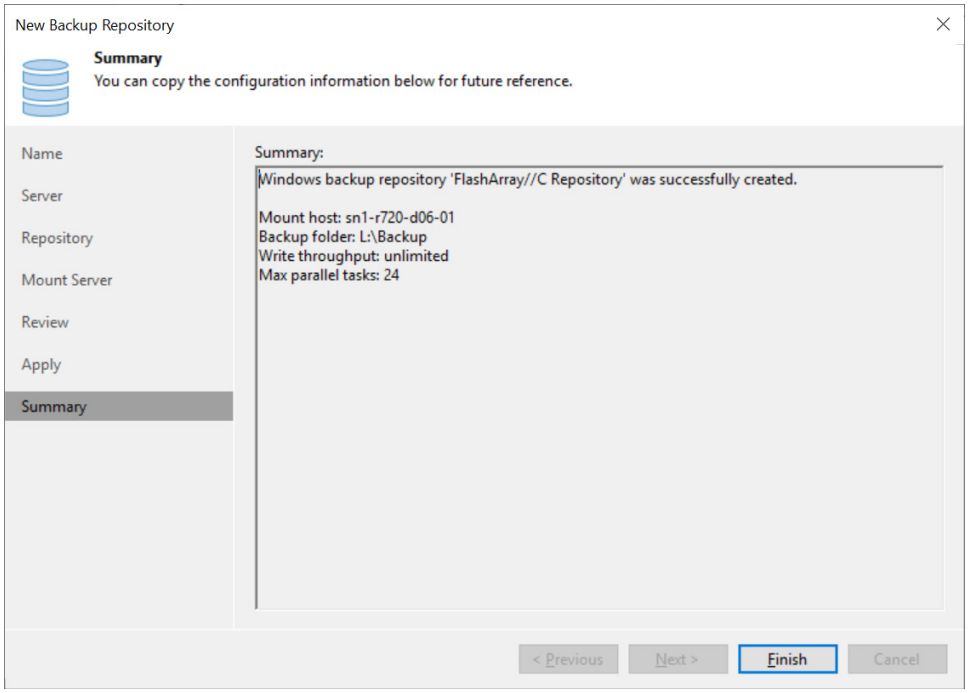


Figure 14: Finishing Veeam backup repository setup

5. Add a scale-out repository. After entering a name and description, add the newly created repository as an extent (Figure 15). Click on **Advanced** to open the advanced settings.

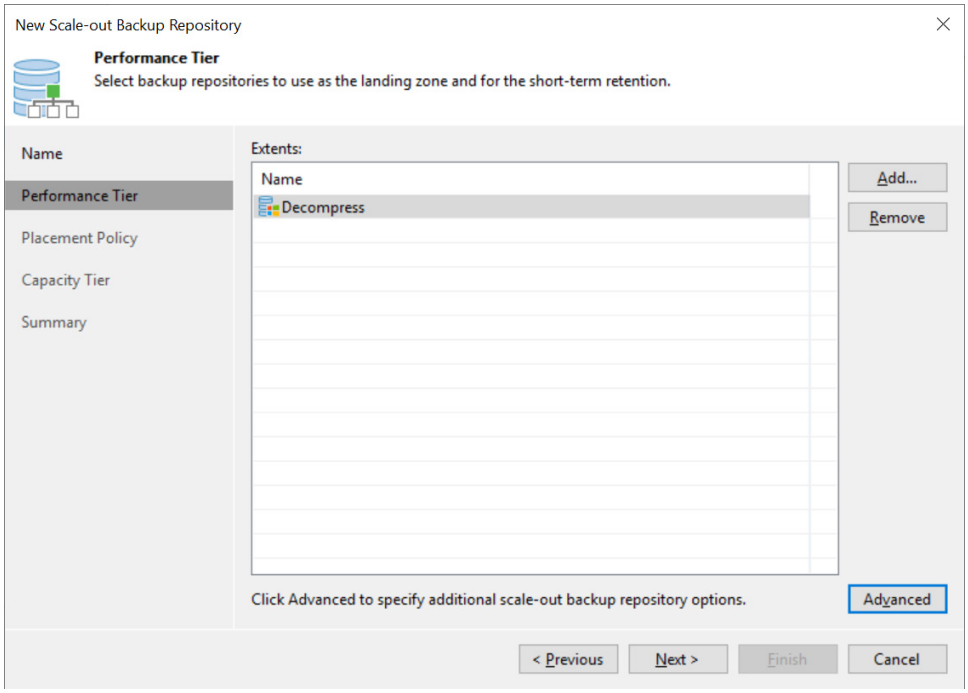


Figure 15: Adding a SOBR extent

6. Enable the **Use per-machine backup files** option if not already selected (Figure 16). Click OK, then click Next to continue. If prompted to automatically update jobs and backups to use the new repository, click Yes.

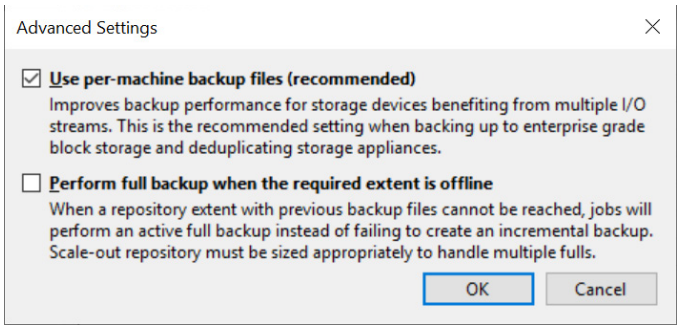


Figure 16: SOBR advanced settings

7. Select the **Data locality** placement policy (Figure 17). With multiple extents in a SOBR, the Performance policy will make synthetic full backups behave more like NTFS repositories, losing much of the efficiency of Fast Clone.

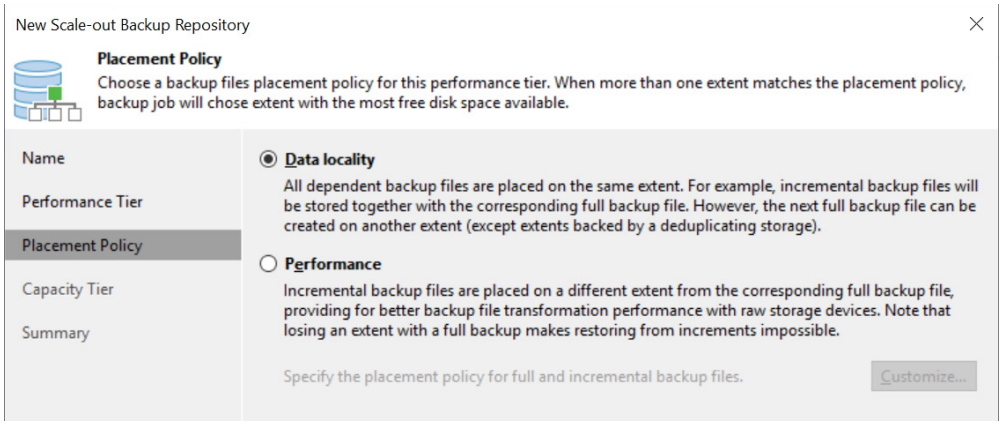


Figure 17: SOBR placement policy

8. Complete the repository creation without any additional options.

   NOTE: While we did not enable the capacity tier, there are no issues with doing so if you wish to use the public cloud or another archive storage option.

Exclude Veeam Processes from AV scanning

For maximum performance, ensure the VeeamAgent.exe and VeeamDeploymentSvc.exe processes are excluded from AV scanning. In our lab, we used the following PowerShell commands to exclude them from Windows Defender monitoring.

```
Add-MpPreference -ExclusionProcess 'C:\Program Files (x86)\Veeam\Backup
Transport\x86\VeeamAgent.exe'
Add-MpPreference -ExclusionProcess 'C:\Program Files (x86)\Veeam\Backup
Transport\x64\VeeamAgent.exe'
Add-MpPreference -ExclusionProcess 'C:\Windows\Veeam\Backup\VeeamDeploymentSvc.exe'
```

Testing Details

We set out to measure several factors with Veeam V11 deployed on FlashArray//C. We wanted to see a performance difference between physical and virtual backup proxies, whether NTFS or Resilient File System (ReFS) repositories were better, how the Veeam compression and deduplication settings affect performance, and how FlashArray//C data reduction shrinks Veeam backups.

We ran a series of operations, varying the number of VMs, proxy type, and file system. For all variations, we backed up sets of VMs, starting with an active full backup, followed by five simulated days of 4GB data change and incremental backups, plus a synthetic full backup. We then ran full VM restores for all the VMs in the backup set. We also ran instant recovery and migrations to measure performance and scaling. We measured data reduction and storage consumption at the host and array. We deleted all backup data and allowed garbage collection to complete before beginning the next test.

To limit the influence of ESXi servers, VM load, and networking on throughput, all backups used FlashArray storage snapshots orchestrated through the Pure plug-in. We performed all backups from snapshots with Direct SAN mode, where snapshots were attached directly to the VM proxies. All full VM restores also used Direct SAN mode. For more information on transport modes, see the Veeam Help Center.

Test Results

Physical and Virtual Backup Proxies

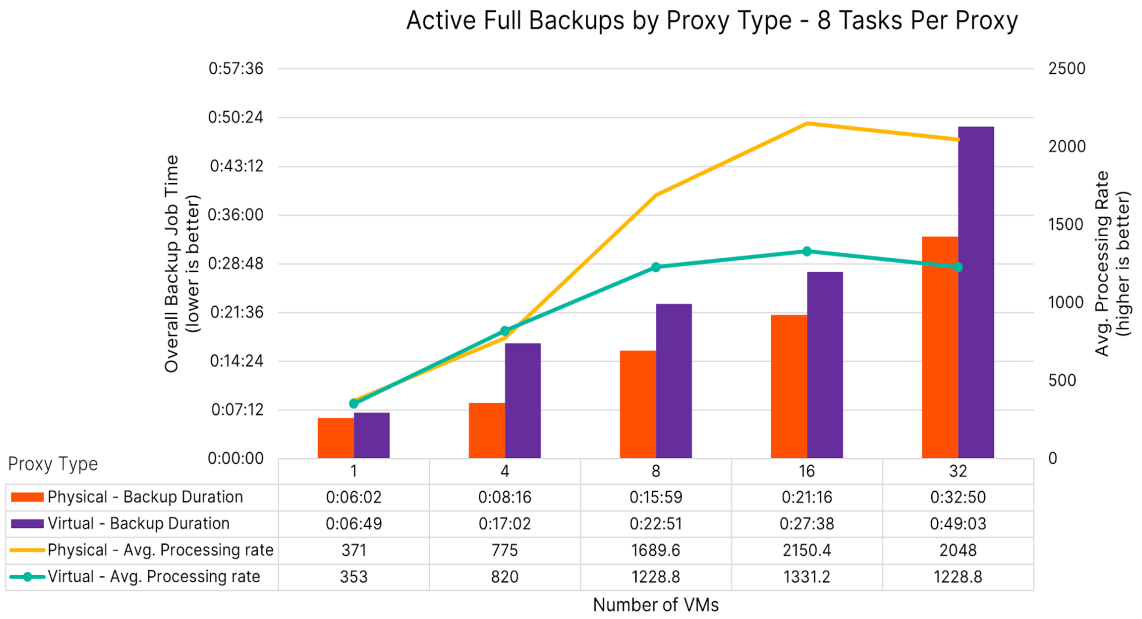
First, we looked at the difference between physical and virtual backup proxies. To ensure as fair a comparison as possible, we limited the physical proxies to eight concurrent tasks even though can handle many more. Veeam does not recommend more than eight vCPUs in a virtual proxy.

It's important to note that the backup and restore statistics that follow include more than just data movement. Creating and deleting VMware snapshots and manipulating storage adds some time to backup jobs. That time isn't factored in the average processing rate Veeam reports, so our tests' processing rates and durations will not match up. Peak processing rates were consistently higher than the averages.

When restoring VMs, Veeam likewise does not include VM creation in its processing rate. The restore process also managed VMs in individual jobs, so there were no overall statistics to use. The numbers in our tests are based on the earliest start time and latest completion time, with processing rates averaged from the entire amount of data divided by the overall duration.

Backup

As shown in Figure 18, performance between physical and virtual proxies was significantly different. Although the CPU types are identical, networking was the most significant factor determining performance: iSCSI throughput on the virtual proxies limited by the hypervisor. The physical proxies reached throughputs up to 67% higher than the virtual proxies. In our lab environment, it would take more virtual proxies to equal the processing power of the physical proxies. Other environments might see a narrower performance gap or no gap at all.



Proxy Type	1	4	8	16	32
Physical - Backup Duration	0:06:02	0:08:16	0:15:59	0:21:16	0:32:50
Virtual - Backup Duration	0:06:49	0:17:02	0:22:51	0:27:38	0:49:03
Physical - Avg. Processing rate	371	775	1689.6	2150.4	2048
Virtual - Avg. Processing rate	353	820	1228.8	1331.2	1228.8

Figure 18: Backup performance by proxy type

Performance scaled better than linearly as we backed up more VMs. Each doubling of VM counts increased backup times by less than 80%. This trend is most noticeable below the maximum tasks available on all proxies, where there is headroom to run more concurrent backups. Crossing that threshold, moving from 16 to 32 VMs, still sees less than an 80% increase, mainly because the overhead of quiescing VMs, creating snapshots, and attaching them to the proxies stays fairly consistent as more VMs are backed up.

On the physical proxies, we also measured the effect of increasing the number of tasks per proxy from eight to 24 to take advantage of the 24 physical CPU cores (Figure 19). At eight tasks per proxy, backups can process 16 parallel VMs, so we didn't see any difference until we reached 24 VMs. As the number of VMs being processed increased, we saw more efficiency as the proxies spent proportionally more time transferring data. However, the repository was already close to its maximum write speed with 16 concurrent VMs, so there wasn't much room for improvement. We saw an average backup processing rate, including all preparation tasks such as VM snapshots, of 9.3TiB/hr.

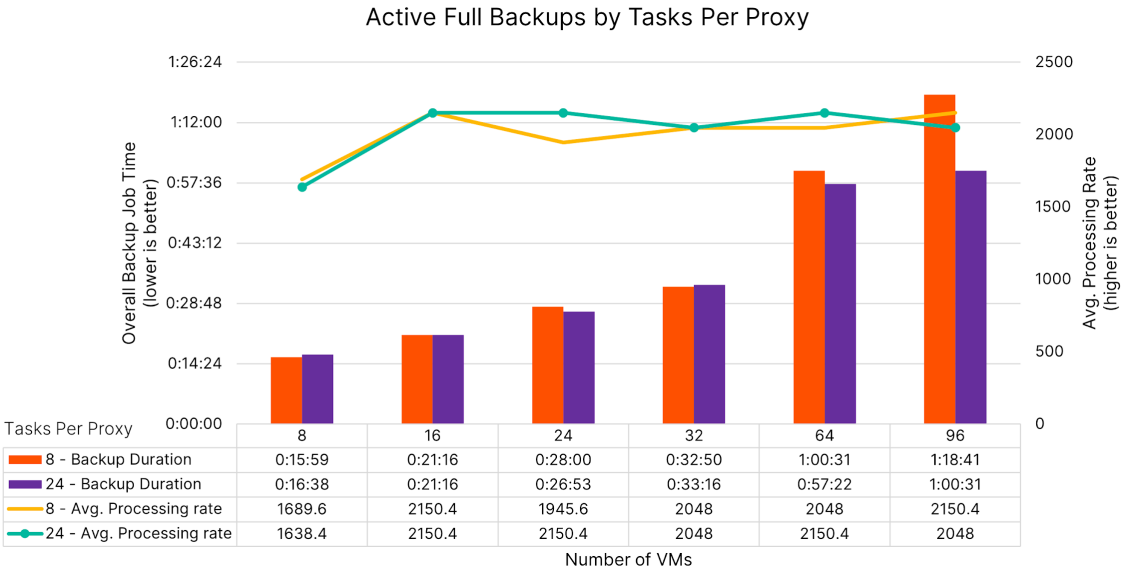


Figure 19: Backup performance scaling by number of tasks

Restore

Since our source VMs use thin-provisioned disks, Veeam defaults to network (NBD) mode on restore. To keep the comparison between physical and virtual proxies equal and keep the hypervisor out of the mix as much as possible, we chose to have Veeam convert the virtual disks during restore to thick provisioned eager zero format and force Direct SAN mode. We used the same backup job as the source for all the restore tests so the restored data would be consistent.

On the physical backup proxies, with eight tasks per proxy, restores were faster than backups (Figure 20). Part of the difference is the snapshot processing during the backups, but the average processing rate was up to 200MiB/s higher. The lower available bandwidth on the virtual proxies held them back, limiting them to an overall processing rate of under 750MiB/s, or around 2.5TiB/hr.

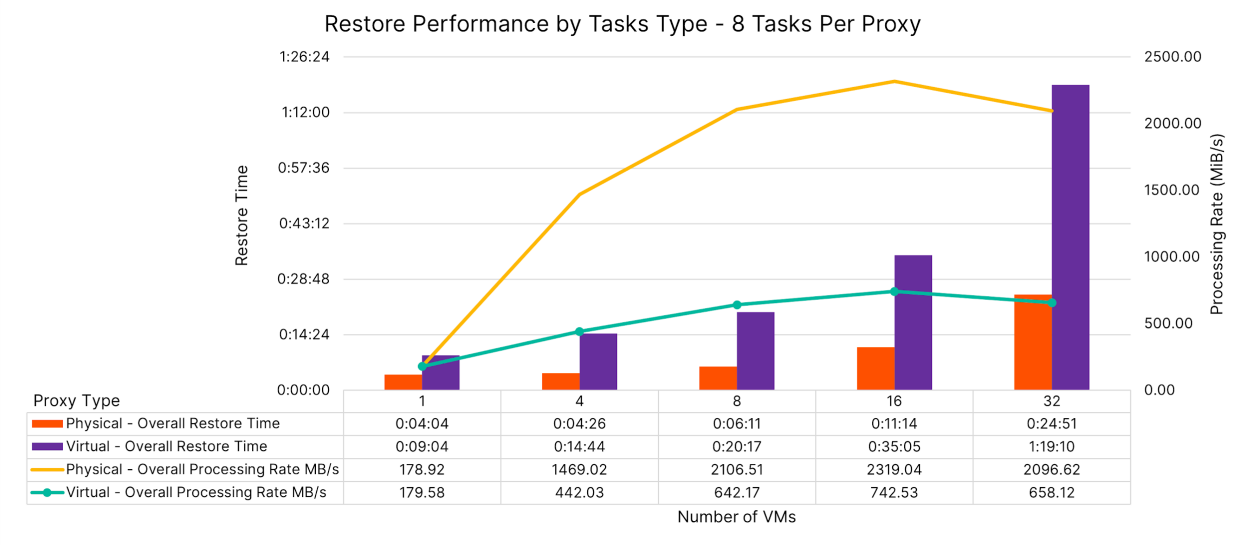
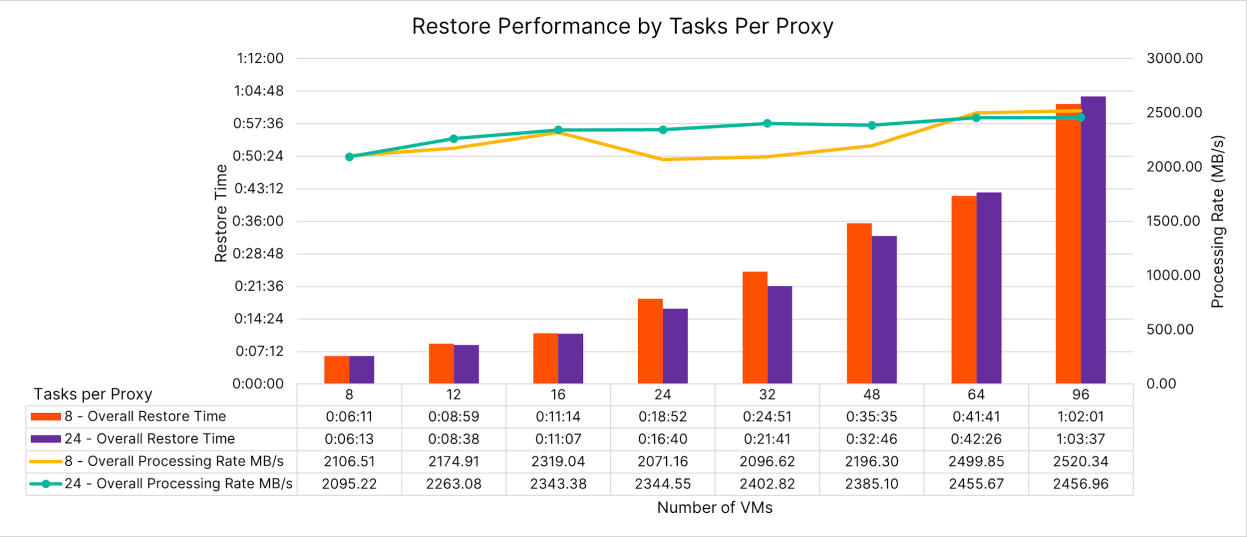


Figure 20: Restore performance by proxy type

We also compared restore times with eight and 24 tasks per proxy (Figure 21). For 16 concurrent VMs and below, the restore times are virtually the same since there are not enough total tasks to consume more than eight tasks per proxy. From 24 to 48 tasks, the extra available tasks did improve throughput, but by less than 15%. As with backups, Veeam is already pushing the FlashArray//C near its limit at 16 tasks, and the per-VM overhead equalizes the overall performance at higher VM counts. With the VMs sized at 100GiB, the average restore throughput works out to a little over 9TiB/hr.



Tasks per Proxy	8	12	16	24	32	48	64	96
8 - Overall Restore Time	0:06:11	0:08:59	0:11:14	0:18:52	0:24:51	0:35:35	0:41:41	1:02:01
24 - Overall Restore Time	0:06:13	0:08:38	0:11:07	0:16:40	0:21:41	0:32:46	0:42:26	1:03:37
8 - Overall Processing Rate MB/s	2106.51	2174.91	2319.04	2071.16	2096.62	2196.30	2499.85	2520.34
24 - Overall Processing Rate MB/s	2095.22	2263.08	2343.38	2344.55	2402.82	2385.10	2455.67	2456.96

Figure 21: Restore time by the number of tasks per proxy

FlashArray//C removed the traditional restore penalty with physical proxies, with data protection and recovery rates close to 10TiB/hr. This means less downtime cost due to ransomware or other data events.

Repository File System

Veeam V11 supports both NTFS and ReFS repositories on Windows servers. Veeam recommends ReFS to take advantage of fast cloning for faster synthetic full backups, but some customers choose to use NTFS with SAN storage such as FlashArray to gain TRIM/UNMAP support. While TRIM/UNMAP is not supported on ReFS, with Windows Server 2019 it is more aggressive than NTFS at releasing unused space.

We compared full, incremental, and synthetic full backup and restore performance and storage consumption with both file system types, using sets of four, eight, 16, and 32 VMs. All tests used physical proxy servers limited to 24 tasks.

Backup

On active full and incremental backups, backup times were comparable between NTFS and ReFS. As the number of VMs increased, active full backups to ReFS repositories pulled ahead of NTFS. As Figure 22 shows, ReFS has a clear advantage over NTFS when it comes to synthetic full backups. Whereas Veeam must read and write back the entire data set for the backup job on an NTFS repository, ReFS fast cloning cuts the synthetic full time by more than half, and the difference increases as the data grows.

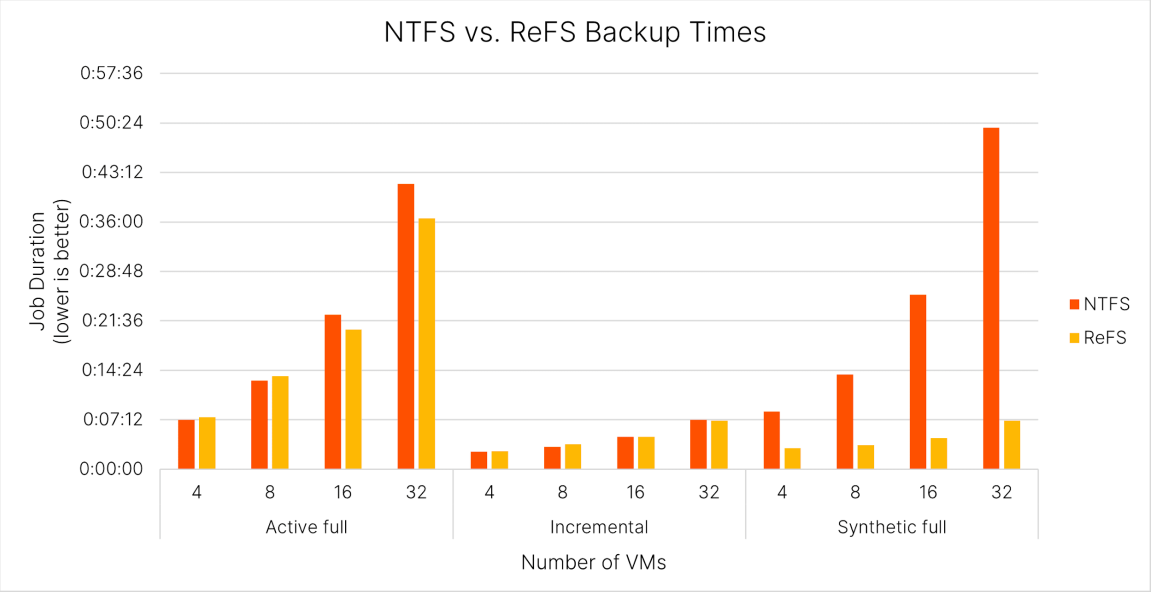


Figure 22: NTFS and ReFS backup times

Note that the job duration in Figure 22 is not limited to creating the synthetic full backup file. Veeam performs an incremental backup as part of the synthetic full backup session, so the time includes that. As Figure 23 shows, for a large set of VMs, the synthetic full can be a small part of the overall time; ReFS fast clone let Veeam create a synthetic full backup of almost 9TB of source data in under five minutes. In a large environment, ReFS can save many hours of backup time per week.

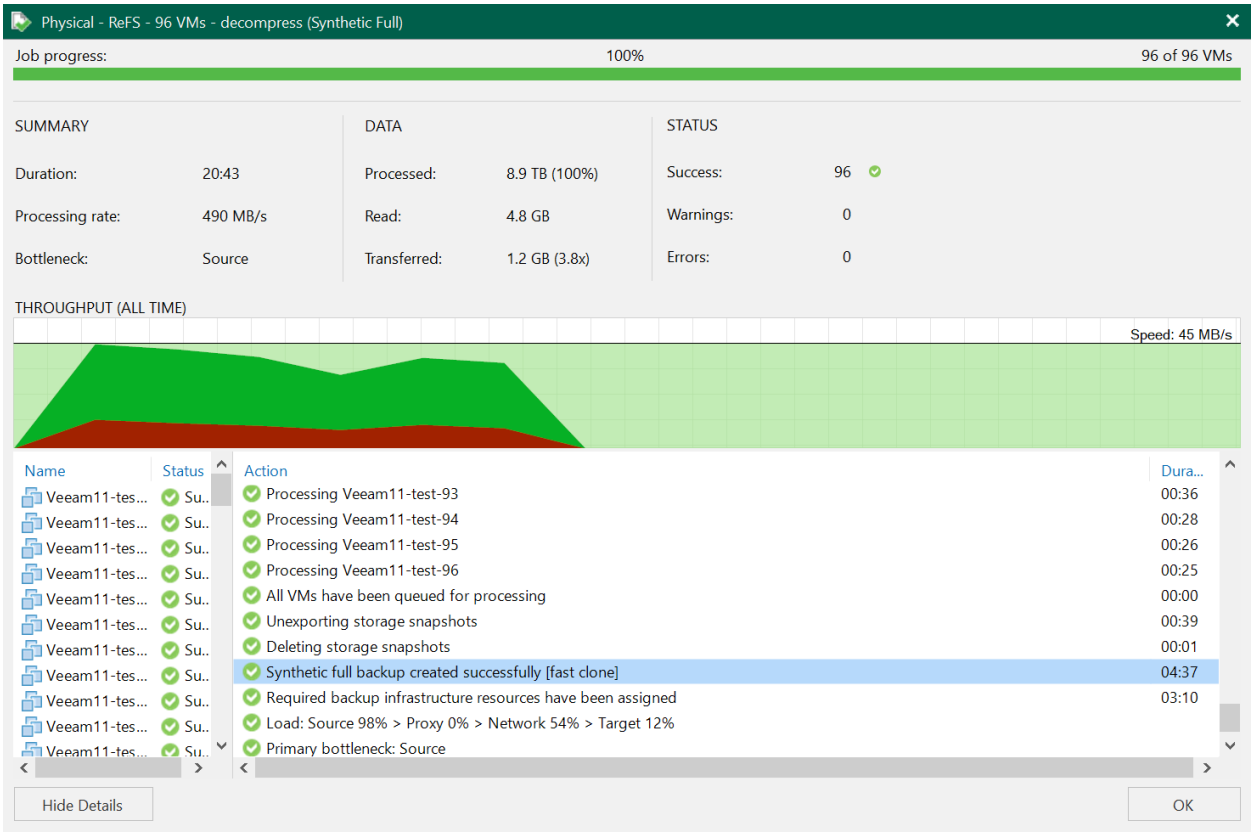


Figure 23: ReFS synthetic full portion of the backup session

Restore

We measured NTFS was about 10% faster than ReFS for full VM restore (Figure 24). ReFS managed to average over 8.8TiB/hr., with NTFS achieving 9.7TiB/hr. Both exceeded the performance of active full backups by a wide margin. Factoring in the benefits of Fast Clone, however, ReFS is the better option overall.

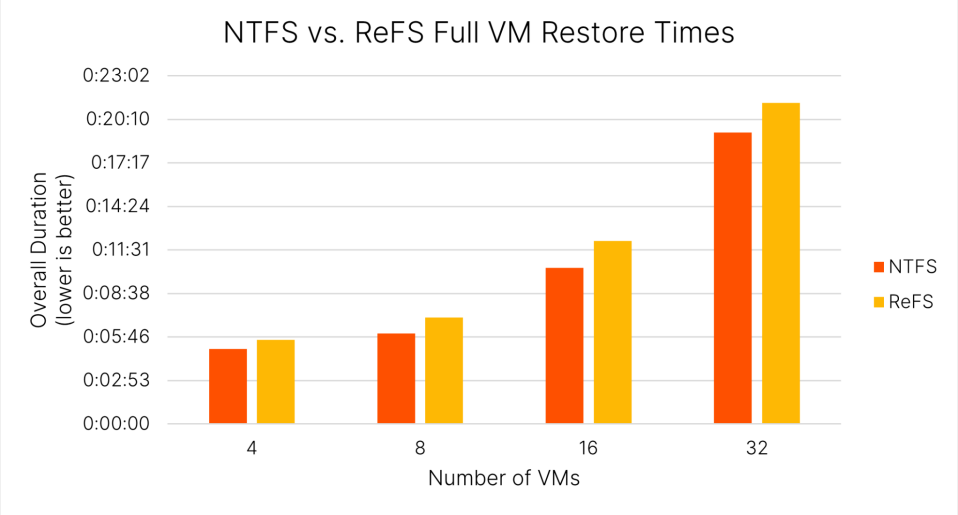


Figure 24: NTFS and ReFS restore times

Storage Consumption

With each backup cycle, we measured the peak and post-reduction consumption on the FlashArray//C. As shown in Figure 25, peak consumption was much lower on ReFS than on NTFS, but both file systems ended up using almost the same amount of space. The efficiency of fast cloning in ReFS makes synthetic full backups effectively zero-footprint, while NTFS must read and rewrite nearly the entire set of backup data. FlashArray//C removes the duplicate data from NTFS anyway, making it effectively wasted storage, I/O, and time. We recommend against running synthetic full backups with NTFS repositories.

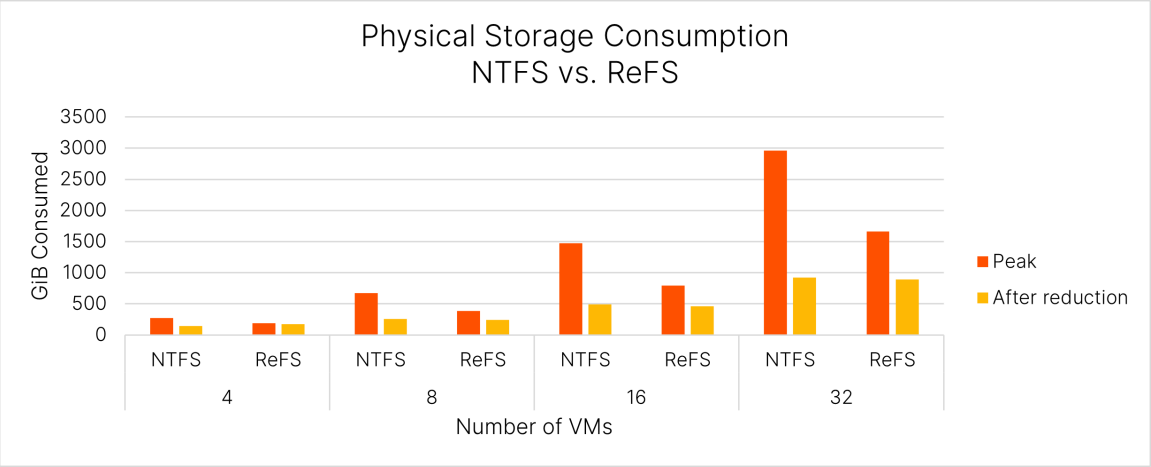


Figure 25: NTFS and ReFS storage consumption

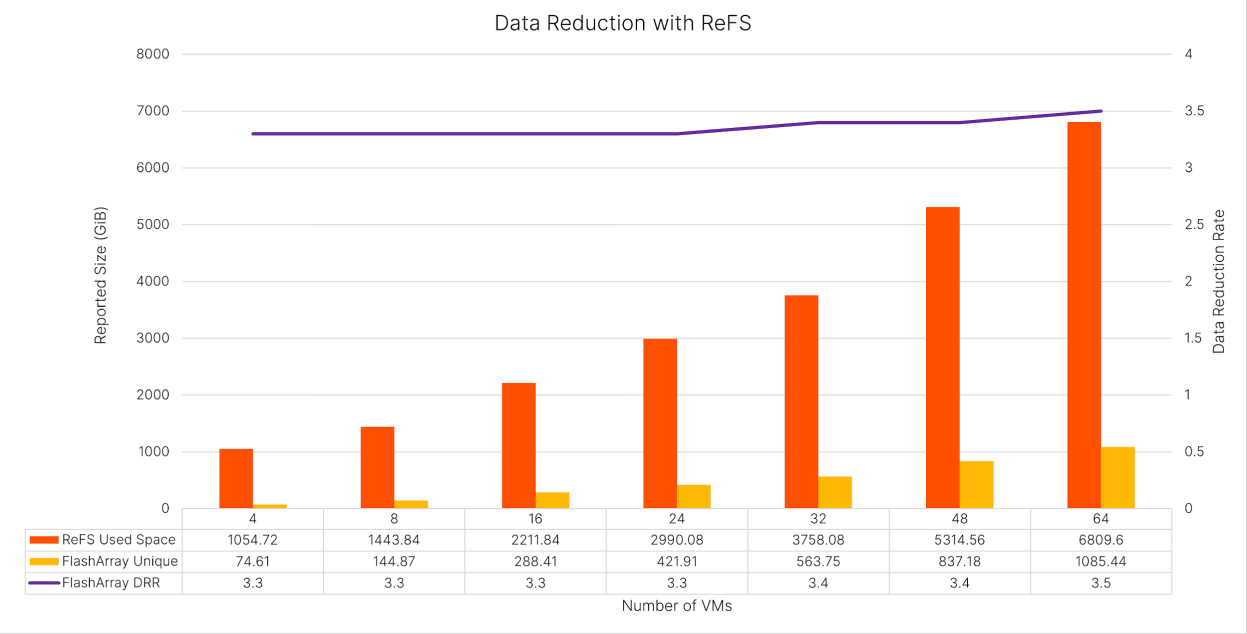
Data Reduction

FlashArray//C compresses and deduplicates data across the entire array, so it's able to remove commonalities both within and between Veeam backup files. This adds significant data reduction to a Veeam backup environment. Before each backup, we

deleted all the data from the Veeam repository and waited for ReFS and the array to finish freeing unused storage. This resulted in a consistent starting point of 671GiB used on the repository volume, as reported by Windows, which represented under 5GiB unique data on FlashArray. We collected statistics for each backup run after the array had reached a steady state to measure data reduction. We looked at the reported storage consumption on the ReFS repository volume and the FlashArray volume it resides on, plus the data reduction rate (DRR) FlashArray reported for the volume.

As Figure 25 shows, accurately measuring overall data reduction requires looking at several statistics. The ReFS volume consistently showed over six times as much used storage as the amount of unique data reported on FlashArray. On top of that, due to Fast Clone with ReFS, the repository folder's logical size was as much as 60% higher than the ReFS volume reported as used. Since ReFS-cloned blocks aren't exposed to FlashArray, the array reported a relatively consistent, lower DRR of 3.3:1, which increased slightly to 3.5:1 as the backup size grew. With active full backups on ReFS, or with a repository on NTFS, the FlashArray would report a DRR several times higher over the backup lifecycle because it would have to process nearly 100% data redundancy across backups. Synthetic full backups on ReFS completely hide that redundancy from the FlashArray and lead to the different DRR for the same physical footprint.

> **Note:** Data reduction rates are included only as an example. DRR is specific to the data set and can vary significantly between environments.



	4	8	16	24	32	48	64
ReFS Used Space	1054.72	1443.84	2211.84	2990.08	3758.08	5314.56	6809.6
FlashArray Unique	74.61	144.87	288.41	421.91	563.75	837.18	1085.44
FlashArray DRR	3.3	3.3	3.3	3.3	3.4	3.4	3.5

Figure 26: Data reduction rates and post-reduction storage (ReFS)

Instant VM Recovery

We tested Veeam Instant VM Recovery using a backup cycle of all 96 VMs as the source for all recoveries. Using the same 96-VM backup job as a source, we recovered sets of four, eight, 16, 24, 32, 48, 64, and 96 VMs, balancing the VMs evenly across the ESXi hosts and datastores. For each VM, we redirected writes to a separate datastore on the same FlashArray where we migrated the VM. We used Storage vMotion for all migrations. We used a round-robin algorithm when selecting VMs to migrate, balancing migrations across the ESXi hosts and datastores.

When VBR mounts a VM for Instant VM Recovery, there is a minimum processing time to create the NFS sessions and set up the temporary datastore through vCenter. Efficiency improves when multiple instantly recovered VMs share an NFS session. Veeam controls the session allocation automatically. As Figure 27 shows, increasing from four to eight VMs only added about 10 seconds to the overall mount time, but the average time per VM dropped almost in half. Once we reached 32 VMs, the average mount time per VM became steady, and overall times grew linearly.

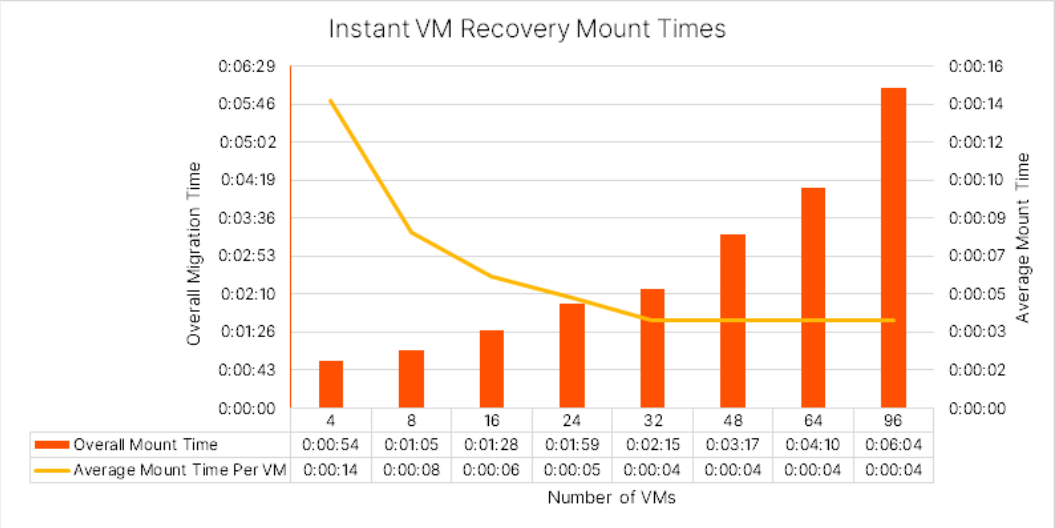


Figure 27: Instant VM Recovery mount times

VMware limits come into play quickly for instant recovery migrations. According to VMware's published limits, a single ESXi host can perform two concurrent Storage vMotion migrations, and a single datastore can support up to eight concurrent migrations. Therefore, the four hosts in our test environment should be limited to a maximum of eight concurrent migrations, regardless of how many datastores we are using.

The migration results bear this out (Figure 28). It took almost the same amount of time to migrate four VMs as it did to migrate eight since eight migrations could run in parallel. Beyond eight, the migration times were close to linear, with average migration times steady at around two minutes.

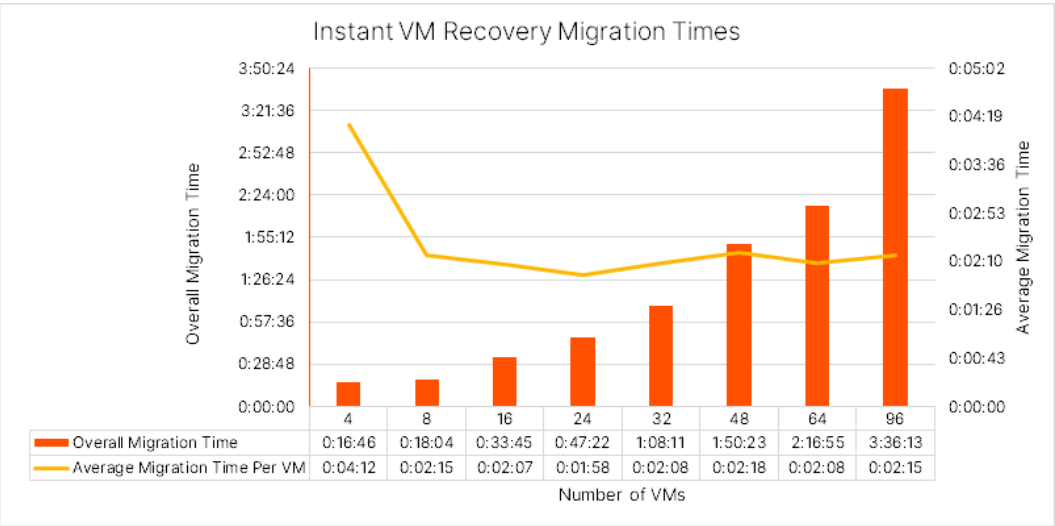


Figure 28: Instant VM Recovery migration times

In our lab, the Instant VM Recovery migrations ran at slightly under 3TiB per hour. While this is significantly slower than full VM restore, being able to resume multiple critical workloads and run them while they move to production storage, at several TiB per hour, can save a lot of downtime and its associated costs. Note that ESXi host performance and VMkernel throughput figure heavily in Storage vMotion migration speed, so your results may vary.

Best Practices for Veeam V11 with FlashArray//C

System Design

Follow Veeam's system requirements when choosing infrastructure servers.

Ensure your proxies and repositories have enough network bandwidth to transfer the compressed backup data within your backup window. We recommend having multiple 10Gbps or higher Ethernet adapters for redundancy and performance. A single 10Gbps adapter is unlikely to drive enough throughput to utilize FlashArray//C fully. Physical Veeam proxies are recommended, especially if your network throughput is limited within virtual machines.

Enable multipath I/O (MPIO) on repository servers for redundancy and performance. Use the least queue depth (LQD) MPIO policy. Ensure your repository servers have enough bandwidth available to the FlashArray//C to support the workload. Using higher bandwidth interfaces, such as 25Gbps or 40Gbps Ethernet adapters or 32Gbps Fibre Channel HBAs, will give the best throughput to and from the repository.

Disable antimalware (AV) scanning of Veeam processes on all Veeam infrastructure. AV scanning can degrade backup performance by 20% or more, even on proxy systems that aren't writing data directly to storage. It is important not to exclude the repository paths so malware can't be written there by other processes.

Use Direct SAN mode for best backup performance. For VMs running on VMFS datastores on Pure FlashArray storage, leverage the Pure plug-in to reduce production impact and enable recovery from snapshots.

Use a scale-out repository with extents of 200TB each or less to allow easy, seamless scaling as your data grows.

Use Windows Server 2019, build 1809 or later, for repository servers, especially if you plan to use ReFS. We have seen issues with space reclamation on older versions of Windows that can lead to capacity issues. You should format repositories as ReFS using 64K clusters, or XFS on Linux, using 4K blocks, to take advantage of synthetic full backups and Fast Clone. You can save hours of weekly backup time and reduce the load on production systems. For XFS, use a Linux distribution supported by Veeam. If you choose NTFS instead of ReFS, run periodic active full backups instead of synthetic full backups to save unnecessary I/O and backup time. You will need enough available space to hold the additional backup data until you can de-duplicate it.

Veeam Proxy Configuration

Physical and virtual proxies should be restricted to no more than one task for each single CPU core and 2GB RAM. For example, a physical proxy with 12 cores and 64GB of RAM should be limited to 12 tasks even though the RAM would support more. Virtual proxies should be restricted to no more than eight tasks, even if they have sufficient resources allocated to support more.

Veeam Repository Configuration

Use a scale-out repository, starting with a single extent. Use the **data locality** placement policy. Configure the SOBR advanced settings as follows (Figure 29):

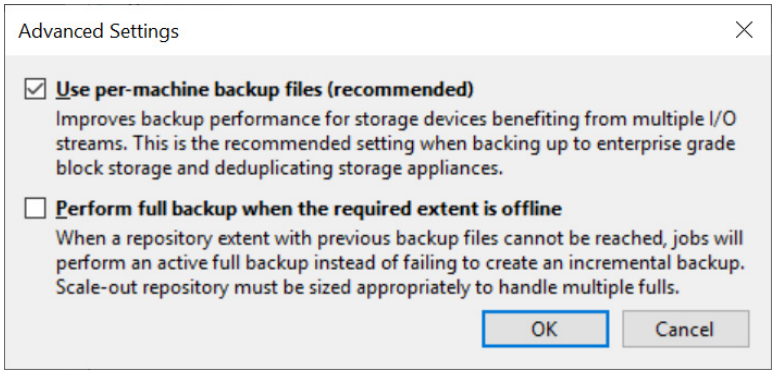


Figure 29: SOBR advanced settings

- Enable the **Use per-machine backup files** option for the best performance when backing up and restoring multiple VMs.
- Disable the **Perform full backup when the required extent is offline** option.

Set the following repository advanced settings on each SOBR extent (Figure 30):

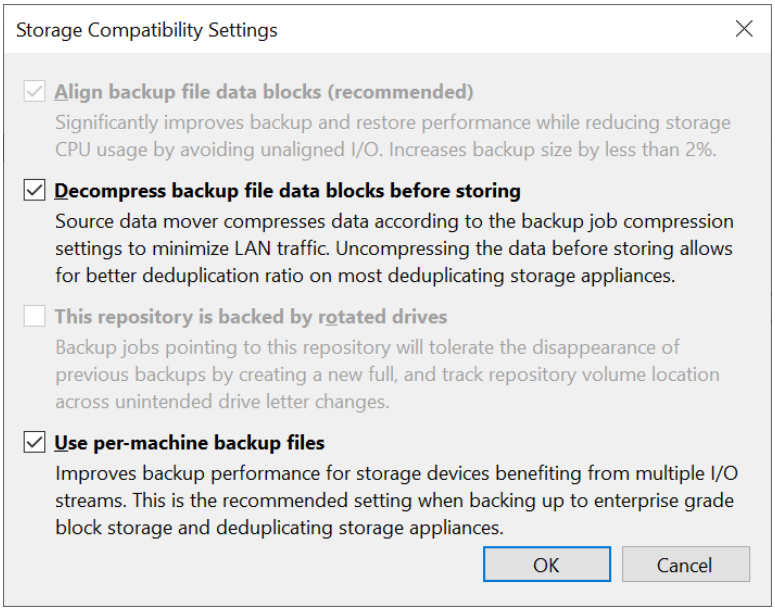


Figure 30: Advanced repository settings

- Enable the **Align backup file data blocks** option to improve efficiency and performance.
- Enable the **Decompress backup file data blocks before storing** option to maximize DRR on FlashArray//C.
- Enable the **Use per-machine backup files** option for the best performance when backing up and restoring multiple VMs. The SOBR will also enforce this option.

Veeam Backup Job Advanced Settings

Configure these advanced settings in Veeam backup jobs for the best backup performance.

On the **Backup** tab (Figure 31):

- Select the **Incremental** option.
- Enable the **Create synthetic full backups periodically** option when using ReFS or XFS repositories.

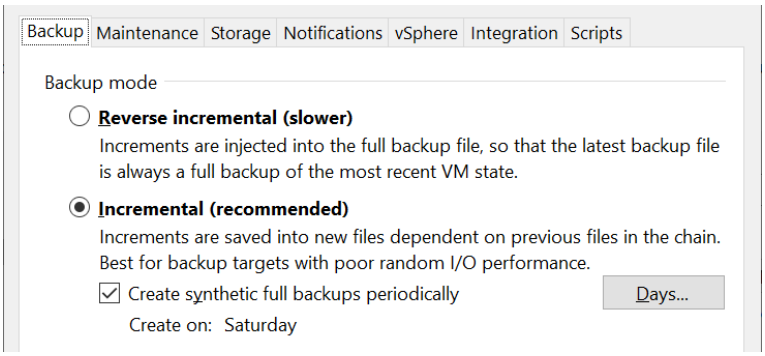


Figure 31: Advanced job options: Backup tab

On the Storage tab (Figure 32):

- Enable the **Enable inline data deduplication** option.
- Enable the **Exclude swap file blocks** option.
- Enable the **Exclude deleted file blocks** option.
- Select the **Optimal compression** level.
- Select the **Local target storage optimization** option.

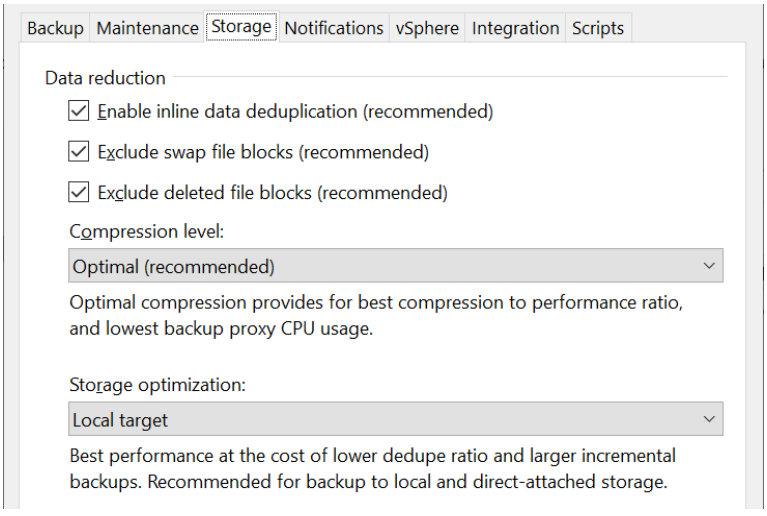


Figure 32: Advanced job options: Storage tab

NOTE: Setting a different compression level or completely disabling compression will affect network utilization between the backup proxy and repository servers if they are separate. A proxy can transmit data 20%—or more—faster than the link speed when compression is enabled.

If you use FlashArray for your primary storage and the Pure plug-in, on the Integration tab (Figure 33):

- Enable the **Enable backup from storage snapshots** option.
- If you have many VMs per datastore, enable the **Limit processed VM count per storage snapshot to** option and set a value between 10 and 20. Note that this will create more snapshots on FlashArray during the backup but can reduce the backup time for large datastores.
- Enable the **Failover to standard backup** option to allow backups to succeed if snapshots fail on the primary FlashArray. If you enable this option, you should connect the source datastores to the backup proxies to allow SAN mode to work without snapshots.

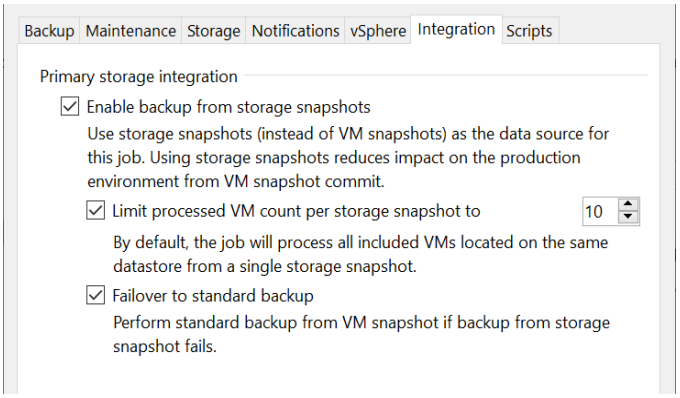


Figure 33: Advanced job options - Integration tab

Recovery Settings

For best performance during Instant VM Recovery, enable the **Redirect write cache** option (Figure 34). Select a datastore on fast storage such as FlashArray//X. Do not select the same datastore where you will migrate the VM, as this will force Veeam to use Quick Migration instead of Storage vMotion, causing a service disruption.

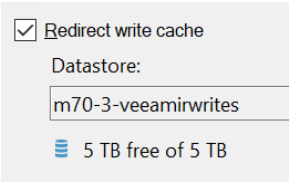


Figure 34: Redirect write cache options

To force Veeam to use Direct SAN mode to restore VMs with thin-provisioned disks, change the disk type to **Thick (eager zeroed)** (Figure 35). **Thick (lazy zeroed)** will also force Direct SAN mode, but lazy zeroing is not recommended on FlashArray. If you do not change the disk type, thin-provisioned VMs will use NBD mode , or Virtual Appliance (HotAdd) mode on virtual proxies, for restores.

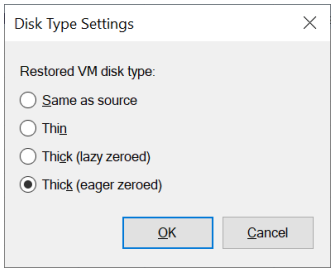


Figure 35: Virtual disk conversion options

Notes on ReFS

While ReFS is the recommended repository file system for Windows for efficient synthetic full backups, there are caveats. ReFS and NTFS behave differently with regard to deleted space and self-maintenance. Where NTFS does not perform self-maintenance and requires chkdsk to detect and repair corruption, and TRIM or UNMAP commands to release unneeded space, ReFS performs ongoing checks and corrections. As part of its normal operations on Windows Server 2019, ReFS will zero out deleted blocks, reclaiming the space on the FlashArray. It isn't necessarily fast, but during lab testing space was consistently reclaimed within several hours, even when deleting over 90TiB at once (Figure 36).

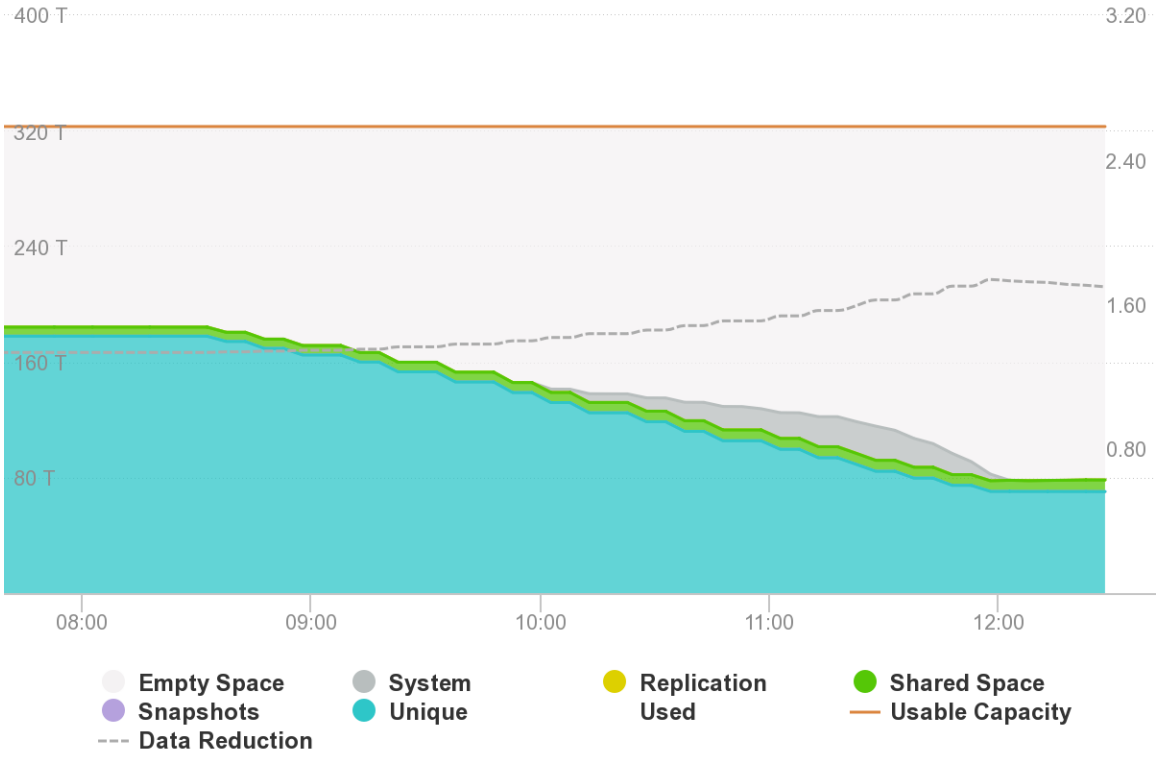


Figure 36: ReFS space reclamation on FlashArray//C

If your repositories are busy throughout the day, you may see longer cleanup times after deleting large amounts of backup data. If you need to reclaim space manually, we recommend using Microsoft's SDelete utility to overwrite unused blocks on your repositories.

Conclusion

Veeam and FlashArray//C make a great combination for fast, efficient backup and recovery of VMware environments. The solution is simple to deploy and manage for lower TCO. FlashArray//C adds global data reduction to Veeam backups, and the shared storage platform eliminates inefficient data silos. The performance and density of QLC flash let you protect petabytes of data fast, in as little as 3U, without the traditional "restore tax," with economics that rival disk-based solutions. FlashArray SafeMode Snapshots add an extra layer of protection against malicious and accidental destruction of your backup data, helping you get back online faster after a ransomware attack.

When you're ready to see how FlashArray//C and Veeam V11 can improve data protection for you, visit Pure's Veeam solutions page, and reach out to your Pure account team.

Additional Resources

Next Steps

- Learn more about FlashArray//C.
- See how Veeam V11 can improve availability in your environment.
- Learn how to use SafeMode Snapshots on FlashArray//C to protect your Veeam backup data against ransomware attacks.

Supporting Information

- FlashArray//C Data Sheet
- Veeam Backup & Replication 11 User Guide for VMware vSphere
- Resilient File System (ReFS) Overview

About the Author

Roy Child is a Senior Solution Architect with Pure Storage, responsible for defining data recovery solutions and reference architectures for primary workloads such as Oracle, SQL, and VMware. Roy has worked in and with the data protection industry for more than 20 years, from end-user to IT architect with multiple backup and recovery products, followed by product management with Commvault. Roy joined Pure Storage in April 2019.

Pure Storage, Inc.
650 Castro Street, #400
Mountain View, CA 94041