TECHNICAL WHITE PAPER

Using FlashArray//C File Services as a Veeam Repository

Veeam compatibility with FlashArray//C™ File Services.

Contents

Introduction

Backup repositories using FlashArray™ storage have proven to be an excellent choice for Veeam Backup & Replication (VBR) due to its ease of management, speed, and data reduction capability. Using the Pure Storage® FlashArray plug-in for Veeam Universal API for hardware snapshot management improves backup performance even further by enabling backup from storage snapshot mode for faster backups and less impact on ESXi. With 99.9999% availability from FlashArray, recovery point objectives (RPO) can also be lowered by enabling snapshot-only copies for critical applications.

Building upon this success, the objective of this white paper is to evaluate how FlashArray//C File gateway is a viable network-attached storage (NAS) repository target for VBR. One of the advantages of NAS is that it is easily shared between backup proxies, providing horizontal scalability.

For this paper, you will need to be using Veeam Backup & Replication. This is not a step-by-step guide; rather we will walk you through some of the setup steps when necessary and point out relevant configuration settings. Though measuring performance is not the intended purpose, we have used ingest rate and backup duration as a way to compare the different settings and gauge the impact they have on the overall environment. The goal is to establish data from which we can start more detailed best practices guidelines.

FlashArray File Services

FlashArray File Services adds native, multi-protocol file storage and access to FlashArray components for a unified block and file platform that eliminates the trouble and expense of running two environments. With FlashArray File Services, both your file-based data and block-based data can benefit from a unified data layer,, a shared pool of capacity, and global data reduction for file-based data as for block-based data.

Currently, FlashArray File Services is available only with pre-approval from Pure Storage product management. To get started, open a case with Pure Support. A support engineer will then walk you through prerequisites, approval, and file services enablement.

FlashArray File Services use virtual interfaces (VIFs) for client connectivity to share and export. VIFs should be set up with two physical interfaces from each controller for redundancy. A minimum of one VIF is required; however, it is possible to have more than one for tasks like communicating with FlashArray File Services over multiple IP subnets or improving performance in certain scenarios. During our testing, only a single VIF was used.

Clients connect using the VIF and the export or file share name. For each protocol, exports or share names must be unique for the entire FlashArray system. However, an NFS export and a server message block (SMB) share can be made available to clients using the same name. For example, if you have a directory that you would like to make available to both NFS and SMB clients, then you can use a single export or share name in both protocol configurations. However, you can't use that very same name to share or export additional directories, and you can't re-share or re-export a directory that has already been assigned an export or share name.

File Services Export Policies Configuration

Export policies on FlashArray govern how the managed directories can be accessed, and the policies can be reused to create multiple exports. Modifying a policy (adding or removing rules) affects the exports for all directories where the policy is used. Use export policies to define:

- NAS protocol; either NFS or SMB

- Rules outlining which client systems are allowed access to exports/shares, what kinds of access they are allowed (no-root-squash for NFS, and anonymous access for SMB), and what permissions they have (read-only, or read/write)

- Whether the policy is enabled or temporarily disabled

Whether it's an NFS or SMB export, the process is very similar. An export policy is created per protocol.

To create an export/share policy:

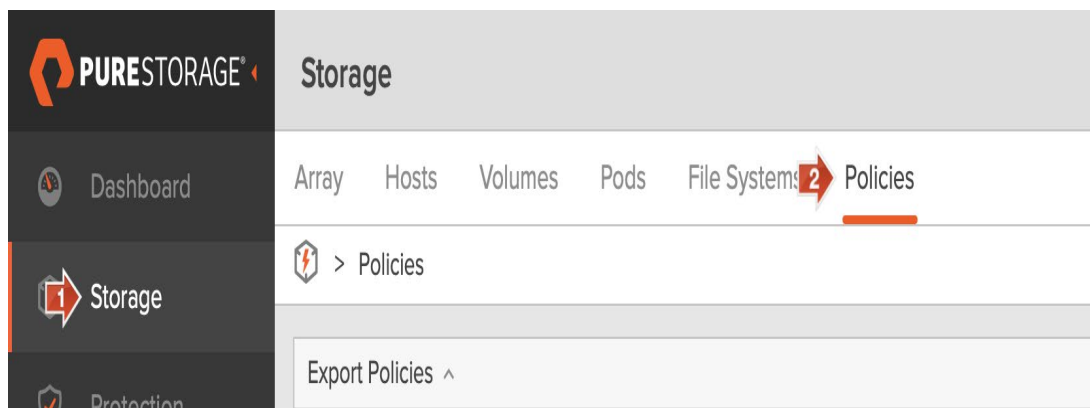1. Navigate to the FlashArray Web UI and log in.

Figure 1. FlashArray Web UI, Policies tab

2. In the left navigation, , click on **Storage**. The Storage page opens on the right.

3. Click on the **Policies** tab.

4. In the **Export Policies** rile, click the plus sign (**+**). A dialog boxopens for creating export policies.
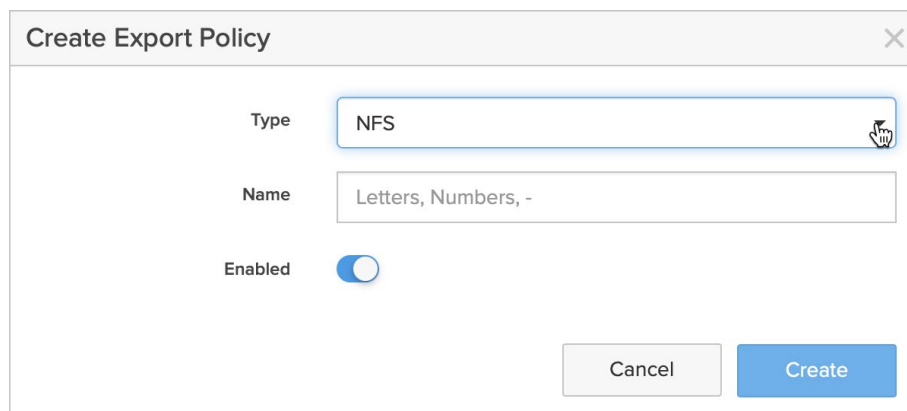

Figure 2. Create Export Policy dialog box (NFS)

5. By default, the **Type** field in the dialog box is set to createan NFS policy. To create an SMB policy, you can change the type, as shown in figure 3, by clicking on the **Type** selection-list icon ▼ .

Figure 3. Create Export Policy dialog box (SMB)

6. Set the rest of the policy in the dialog box. It is worth noting that, when creating an SMB policy, the Access Based Enumeration (ABE) option is available for even more granular permissions. When you are done, click on **Create**.

Once policies are created, they will be listed in the **Export Policies** tile.

Figure 4. Available export policies

Export policies are created with default values and no access rules. Often, it's necessary to edit export policies to update their rules, such as limiting access to export/share to specific hosts, disabling anonymous (guest) access on an SMB share, or enabling no_root_squash on an NFS share. To edit the policy from these default settings, click on the policy name. Below is an example of our **smb-simple** export policy.

Figure 5. SMB export policy example

Access based on host IP address is available for both NFS and SMB. You can use this feature to restrict VBR NAS repository access to only the VBR server and proxies.
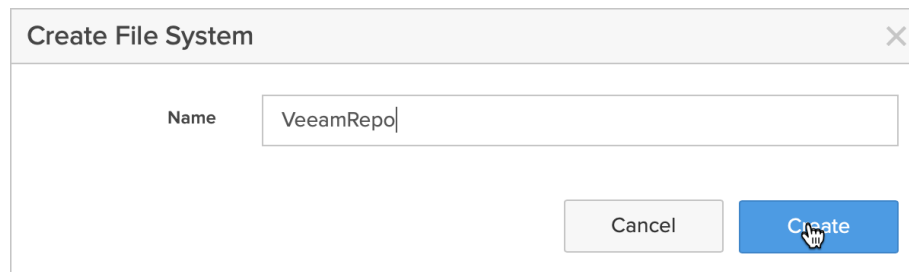
File System Configuration

Creating directory exports or shares starts with creating a file system. FlashArray File Systems introduces the concept of managed directories. A *managed directory* is a directory that allows attaching exports or shares, and you can apply quotas and snapshot policies on it. When an administrator creates a file system, a root (/) managed directory is created, under which other managed subdirectories can be created. It's important to point out that managed directories can't be created by NFS or SMB clients: They are created using the FlashArray Web UI or CLI. For more details, refer to the FlashArray File Services overview in the Supporting Material section.

To create a file system:

1. Navigate to the FlashArray Web UI and log in.
2. In the left navigation, click on **Storage**. The Storage page opens on the right.
3. Click on **File Systems.**
4. In the **File Systems** tile, click the plus sign (**+**). The **Create File System** dialog box opens.
5. Enter the name for the new file system, and then click **Create.**

Figure 6: Create a file system window

As shown below, a new file system will appear in the **File Systems** and **Directories** tiles.

Figure 7. FlashArray Web UI, tiles on File Systems page

As stated earlier, when a new file system is created, the root directory is automatically created. While you can create and export managed directories within the root directory, for the purpose of this paper, we will create NFS exports and SMB shares from the root directory.

Export/Shares Configuration
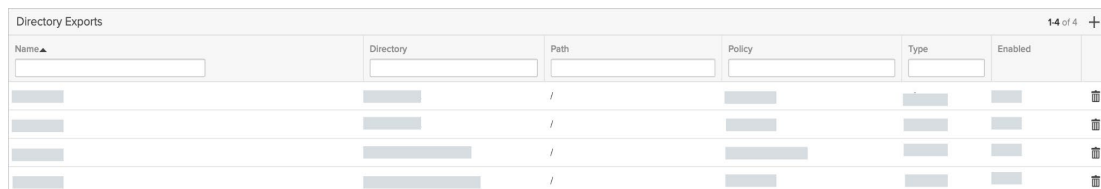
FlashArray supports the following versions of NFS and SMB:

- SMB: versions 1.0 / 2.0 / 2.1 / 3.0 / 3.02 / 3.11

- NFS: version 3

   NOTE: By default, clients will negotiate protocol versions starting from highest to lowest. SMB version 1.0 is deprecated and disabled by default on FlashArray because it's no longer considered secure as it lacks encryption and security.

Creating NFS exports or SMB shares can be achieved via the Web UI or CLI. The following steps will walk through the process using the Web UI.
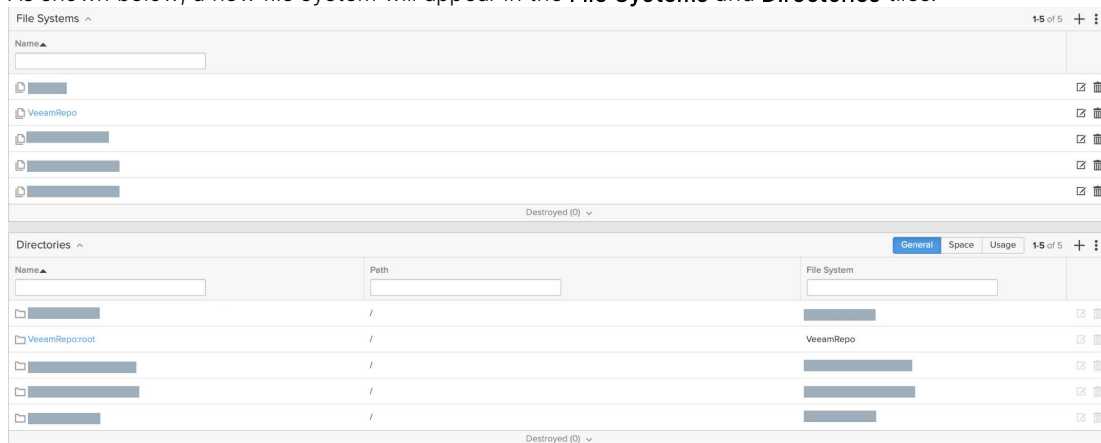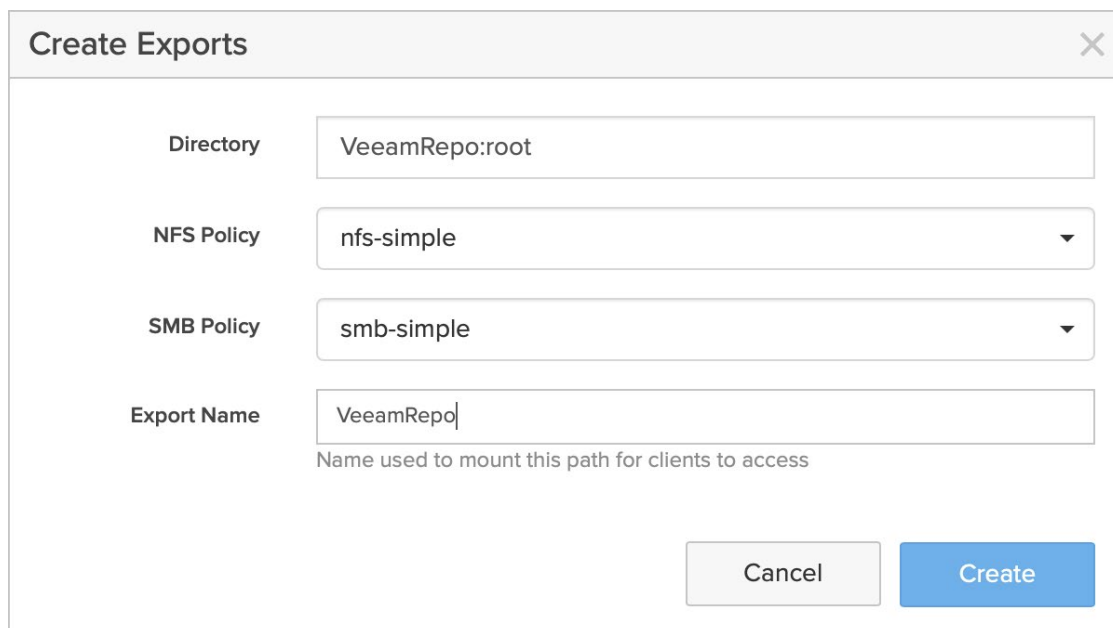
1. Navigate to the FlashArray Web UI and log in.

2. In the left navigation, click on **Storage**. The Storage page opens on the right.

Figure 8. Directory Exports tile

3. Click on **File Systems.**

4. Depending on the screen resolution, you might have to scroll down to the **Directory Exports** tile. Click on the plus sign (**+**) on the top-right corner of the tile.

5. The **Create Exports** dialog box l opens/ Choose the file system/managed directory to be exported, along with enabling either NFS, SMB, or both. In this example, both are being enabled.

6. Click on **Create**.

Figure 9. Create NFS and SMB export on VeeamRepo root directory

Active Directory Setup

SMB on FlashArray requires Active Directory (AD) or LDAP for authentication and authorization. We used Active Directory, as it's the most used.

Prerequisites to successfully set up Active Directory:

- Consistent time and date with the AD domain controller. Ideally, all hosts and arrays should sync to the same network time protocol (NTP) source.
- If DNS is a separate service, the AD server(s) should be defined on the DNS server.
- A user account with privileges to join a computer to the AD domain.

7. In addition, the FlashArray DNS settings must be configured with the IP address and domain of the AD server.

1. From the FlashArray Web UI, click **Settings > Network.**

2. Scroll down to the **DNS Settings** tile, and then click on the checkmark ( ) icon.

3. In the **Edit DNS** dialog box, add the domain and IP address of the AD server.

Figure 9. FlashArray Web UI, Edit DNS dialog box

To configure Active Directory:

1. Navigate to the FlashArray Web UI and log in.

2. In the left navigaton, click **Settings**. The Settings page opens.

3. Click **Access**.
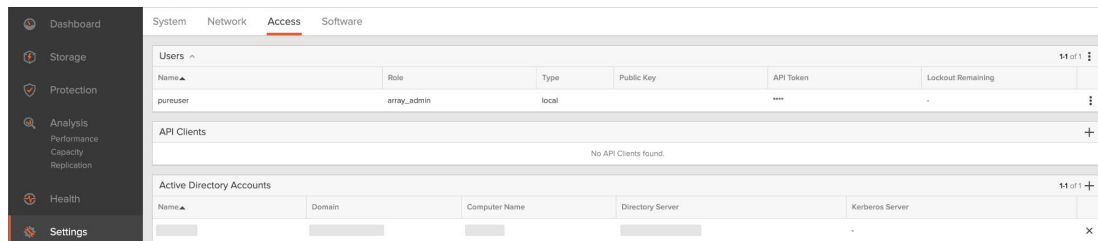
Figure 10. FlashArray Web UI, Settings page, Access tab

4. In the **Active Directory Accounts** tile, click on the plus sign (✚) on the right side.

Figure 11. FlashArray Web UI,,Create Active Directory Account dialog box

5. In the dialog box that opens, fill in the following fields for the Active Directory account:

    a. **Name**: Arbitrary name internal to FlashArray to identify the specific AD configuration. .

    b. **Domain:** Fully qualified domain name (FQDN) for the Active Directory domain.

    c. **Computer Name:** SMB share name that will be used on the network.

    d. **Directory Server:** Active Directory domain controller DNS name or IP address. (This is optional if the domain controller is properly registered with DNS.)

    e. **User:** Active Directory domain user with domain join privileges.

6. Click **Create.**

7. Once you take these steps, you will see the **Active Directory Account t**ile populated with the AD account information. On the Active Directory side, you will be able to see a computer account matching the computer name in the "Computers" container in Active Directory.

VBR Environment Settings

In this section, we will walk through the settings we've used during testing. We will cover the backup proxy server settings and move on to the NAS repositories.

Backup Proxy Settings (VMware Proxies)

Proxies in VBR are the workhorse components. They run the Veeam Data Mover component that is responsible for transferring VM data from production storage to VBR repositories. We can use several physical or virtual backup proxy servers to optimize the performance of multiple concurrent backup jobs, and VBR will distribute the backup workload among the available proxy servers.

For optimal performance, it's important to review [Veeam's best practices guide on vSphere proxies](#).

To view or edit the settings for backup proxy servers:

1. From the left navigation menu on the VBR console, click on **Backup Infrastructure**, and then click on **Backup Proxies**. A list of backup proxy servers will be displayed in the right pane.

2. Select the desired server and then right-click on **Properties.**
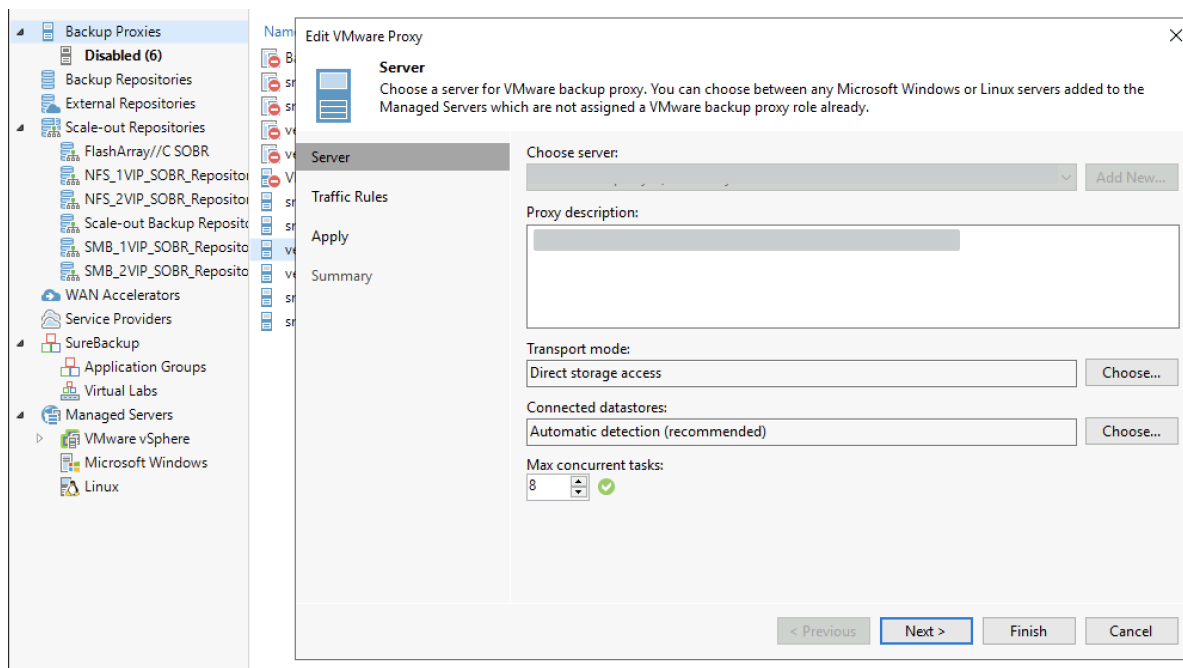
Figure 12. VBR console, viewing the VMware proxy server settings.

The VMware proxy server settings are discussed below.

- **Transport mode: Direct storage access** was chosen during testing because the VMFS datastores are located on the FlashArray//M70s, which are also visible to the VBR proxy servers on the SAN. Since the FlashArray plug-in for Veeam is installed, VBR will orchestrate hardware snapshots on the FlashArray//M70s to create copies of the VMFS datastores, and then proxies will leverage the SAN to process backup data from the snapshots directly, bypassing the EXSi hosts.

- **Connected datastore**: **Automatic detection** was selected because proxy servers have direct SAN connections to datastores and will have the ability to detect datastores that the VMs are running on.

- **Max concurrent tasks:** Based on available compute resources, this setting helps to balance the workload across the backup infrastructure and avoid performance bottlenecks. In our test environment, it was configured based on best practices and Veeam support recommendations.

NAS Repository

The procedure for setting up a NAS repository follows the same steps as for all other types of repositories, as described in Veeam's online user guide on creating backup repositories.

1. In the VBR console, click on **Backup Infrastructure** in the left navigation menu.

2. Right-click on **Backup Repositories > Add Backup Repository**. The Add Backup Repository wizard will open.

3. Click on **Network Attached Storage** to begin.
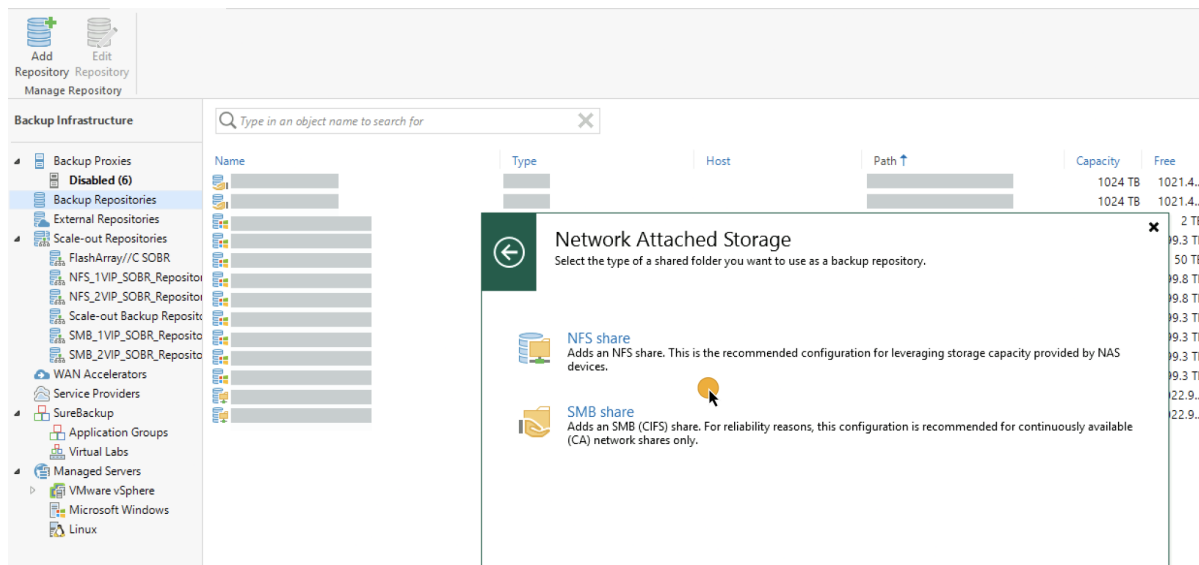
Figure 13. VBR console, adding NAS repository

From this point on, fill out the necessary information as it pertains to each NAS protocol.

NFS and SMB Repositories Common Settings

For greater flexibility and performance, Veeam's current recommendation for new repositories is to use scale-out backup repositories (SOBR). We will focus on the settings that impact the performance of NAS repositories. The following settings are common between NFS and SMB repositories that should be reviewed and adjusted properly if needed.

- **Gateway Server Selection**: Traditionally, with direct-attached storage, Veeam Data Mover (the component responsible for processing backup data) is installed on the same server as the disk. With NAS backup repositories, Veeam Data Mover cannot be deployed on FlashArray//C; instead, Veeam deploys a gateway service component where the Data Mover is installed to facilitate communication between proxy servers and the backup repository on the FlashArray unit.

A gateway server is not necessarily an added physical infrastructure component. Any Veeam proxy server—physical or virtual—can act as a gateway server if it's a Microsoft Windows server for SMB protocol, and it can use either Linux or Microsoft Windows for NFS protocol.

Two choices are available for assigning a gateway server to a repository. You can choose a specific server from the list of available proxy servers, or you can let VBR automatically select one (**Automatic Selection**). Selecting a specific gateway server means all VMs in the backup job will have to use the selected gateway server. This is a benefit when you want to manually manage backup workload distribution. Most of the time though, **Automatic Selection** is preferred, especially when enabling **Use per-machine backup files**, which we will cover below.

- **Load Control Settings:** Under repository properties, the Load Control setting in the repository section governs the load on the backup repository and prevents possible timeouts of repository storage I/O operations due to resources contention.

  - **Limit maximum concurrent tasks:** This option at the backup repository level limits the number of concurrent tasks. VBR will not start a new task until one of the current tasks completes. During testing, we left this option unchecked (disabled). This was done because we configured **Limit maximum concurrent tasks** in the properties settings of VMware proxy servers. We also enabled **Use per-machine backup files** on the repository itself. With all other components properly tuned, leaving this option disabled allows us to focus on the FlashArray File Services performance. This is not a best practice recommendation.

- This option is enabled to increase repository utilization, and to avoid potential resource contention on the gateway server. This option is under the Advanced button → Storage Compatibility Settings.
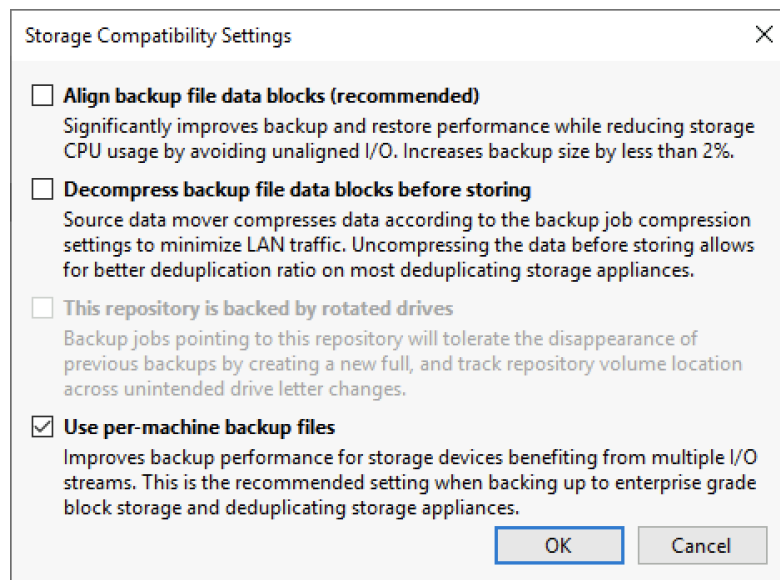
Figure 14. VBR, Repository, Advanced Settings button

In scale-out backup repositories (SOBR), this setting is enabled by default; however, if it's not, it can be enabled: under the **Performance Tier** section, click on the **Advanced** button. The setting will apply to all extents under the SOBR.

The increased utilization of the backup repository comes from using separate write streams for each VM in the backup job. However, with higher utilization, each VM's data is stored in a separate backup file, preventing VBR from performing data

reduction between VMs on the same job. For more details, please refer to the topic in the Veeam user guide about limitations of per-machine backup files.

- **Decompress backup file data blocks before storing:** Generally, enabling compression for a backup job is desirable, as it improves network utilization and backup speeds. However, it's not free: The cost is a less efficient data reduction rate (DRR) on the FlashArray unit. To achieve a higher DRR, VBR offers the **Decompress backup data blocks before storing** setting. When data compression is enabled for a job, VBR will compress VM data on the source side, transport it to the target side, decompress VM data on the target side, and then write raw VM data to the repository NAS target.

  Since our proxy servers are using Direct SAN transport mode, compression/decompression happens on the proxy servers, so the true cost is higher CPU utilization and a little lower transfer rate, but the benefit is a better data reduction rate on the FlashArray unit.

Though we ran tests while this option was enabled, on later tests we left this option disabled. This allowed us to use a little more CPU to drive FlashArray//C file gateway to ingest rate performance.

NFS Repository

As stated earlier, FlashArray File Services supports NFSv3; however, it does not implement network lock manager (NLM). NLM is a separate service that allows a client to request a file be locked by the NFS server. The server must maintain a record of all locks from all clients.

VBR on Linux uses the native NFS client, which defaults to NLM locking enabled, resulting in failed requests for exclusive backup file locks to the FlashArray//C file gateway (NFS Server). However, NFS clients bundled with VBR on Windows do not default to NLM locking, so they do not have this issue.

For that reason, backups running on Linux VBR proxy servers will timeout waiting for exclusive locks from the FlashArray//C file.

Veeam provides a Windows registry setting to override NFS client default options. The registry affects only Linux NFS clients. (Using this setting with Windows gateways does not have any effects). The setting should be applied on the VBR server using:

```
Registry Entry: "NFSLinuxMountOptions"
Registry Type: "REG_MULTI_SZ"
Example REG Data: "fashare01:/extent01;rw,soft,timeo=600,retrans=5,intr,nolock"
```

As this is an override setting, all options recommended by Veeam, in addition to the "nolock" option (which will disable NLM), should be included in the NFS client to map the share correctly.

Once the NFS repository registry on VBR is configured, backup jobs with NFS targets will use the relevant NFS export options to map the desired NFS targets on the gateway servers.

> NOTE: Multiple NFS shares can be added with one NFS share per line.

> NOTE: For Windows Gateways you can only use NfsShareVersions which allows the NFS Version 3 or 4.1 to be set.

SMB Repository

As stated earlier, FlashArray supports multiple versions of SMB. During testing, we only focused on SMBv3.1, as it's already included with Microsoft Windows Server 2019 and 2016, and we wanted to make use of the SMBv3 WriteThrough feature, which was introduced with Windows Server 2019.

The WriteThrough feature is interesting because, by default, SMB clients use caching when connecting to file shares. This is for better performance by reducing network traffic. However, with backup software, it is not a good idea for a couple of reasons. First, the backup files are typically new, and they are large, so caching them is an unnecessary overhead. Second, pending cached writes delay detecting intermittent underlying network issues (see delayed cached Write Failed error 50 message). In a backup environment, we need a guaranteed acknowledgment that data has been written to storage, so if connectivity is unreliable, we have to stop writing. For additional information, please refer to the Veeam technical blog on when to use the SMB WriteThrough feature.

Veeam provides a Windows registry setting that disables SMB client default behavior. The registry setting should be applied to the VBR server (Master):

```
Registry Entry: "UseUnbufferedAccess"
Registry Type: "DWORD"
Value Data : 1
Base: Hexadecimal
```

When applying the registry setting, we noticed a slight uptick in performance.

Backup Job Setup

Having the FlashArray plug-in for Veeam installed enables Direct SAN access transport mode. VBR will leverage vSphere Storage APIs for Data Protection (VADP) snapshots created for each VM, directly from the SAN, by creating hardware snapshots of the VMFS datastores, and then mounting those snapshots on the available proxy servers. The result is reduced load on the ESXi hosts and much faster data transfer. For detailed information, please refer to the Veeam Direct SAN Access documentation.

Clicking the **Advanced** button under the **Storage** section of the job settings will open the **Advanced Settings** dialog box. Under the **Integration** tab, the **Enable backup from storage snapshots** checkbox (selected by default) will enable VBR to perform reading the source data from snapshots of the datastores mounted on VBR proxy servers.

form reading the source data from snapshots of the datastores mounted on VBR proxy servers.
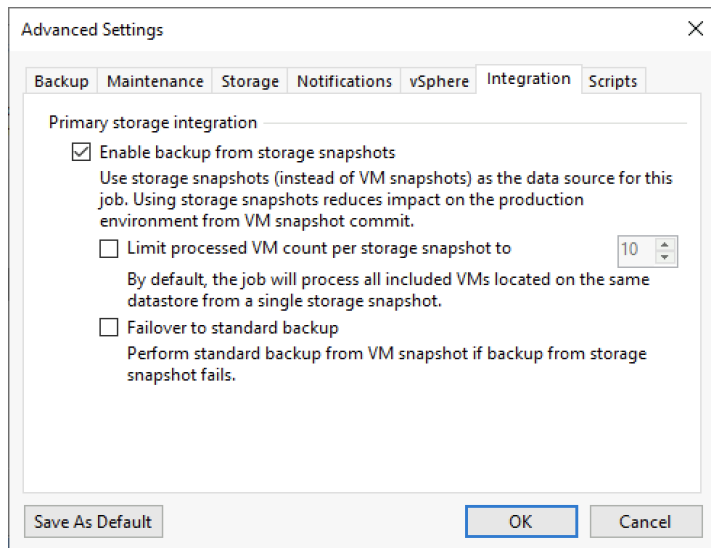
Figure 15. VBR, viewing storage integration settings, in the backup Job settings

Examining VBR and the FlashArray File Repository

We used the following criteria to assess FlashArray File Services as a Veeam NAS repository:

- Number of proxy servers

- Type of proxy servers (physical, virtual, mix)

- Operating system running on proxy servers

- "Decompress backup file data blocks before storing" option

- Repository level, maximum concurrent tasks

- Direct SAN transport

- All backup jobs are Active Full

The data points we collected were:

- Backup duration

- Processing rate as reported by Veeam

- Calculated average actual write activities over the backup duration

Recall, we are not comparing the FlashArray//C File Services to any other solution, we're simply exploring how the FlashArray File Services will behave under what is considered a typical workload, and by tweaking features already available to end-users.

Test Environment Details

We used the following for our tests:

- Four clustered VMware ESXi 6.7 hosts. Each ESXi host features two redundant 40GbE NICs carrying both VM traffic and iSCSI connections.

- Three Pure Storage FlashArray//M70s (FA//M70) storage arrays serving as the primary storage. They were visible to all four ESXi hosts in the cluster. They ran Purity//FA version 6.1.4. Each FA//M70 was uplinked to the network with two redundant 10Gbps iSCSI ports from each controller (four total iSCSI ports).

- 32 Microsoft Windows 10 Enterprise VM instances, which served as the source data. Each VM contained 100GB of data (10GB for the OS and 90GB for random data, 15% compressible), distributed across the three FA//M70 arrays as illustrated in the table below. Due to this environment not being dedicated to this test, the busiest FA//M70 was assigned fewer VMs, while the least busy FA//M70 was assigned more VMs.

FA//M70 #1	FA//M70 #2	FA//M70 #3
8 VMs (~800GB)	11 VMs (~1.1TB)	13 VMs (~1.3TB)

Table 1: VM distribution on FA//M70s

- Veeam Backup & Replication V.11.0.0.837 hosts. The following table lays out the specification for each host:

	VBR	Veeam Proxy #1	Veeam Proxy #2	Veeam Proxy #3	Veeam Proxy #4	Veeam Proxy #5	Veeam Proxy #6
Type	PowerEdge R720	PowerEdge R720	PowerEdge R720	PowerEdge R720	PowerEdge R720	Virtual	Virtual
OS	Microsoft Windows Server (MSFT) 2019	Microsoft Windows Server (MSFT) 2019	Microsoft Windows Server (MSFT) 2019	RedHat Enterprise Linux (RHEL) 8.4	RedHat Enterprise Linux (RHEL) 8.4	Microsoft Windows Server (MSFT) 2016	Microsoft Windows Server (MSFT) 2016
CPU Cores	24	24	24	24	24	8	8
Memory	512GB	512GB	512GB	512GB	512GB	20GB	20GB
Storage Interface (iSCSI)	2 bonded 40Gbps	2 bonded 40Gbps	2 bonded 40Gbps	2 bonded 40Gbps	2 bonded 40Gbps	VMXNET3 10Gbps	VMXNET3 10Gbps

Table 2. Hardware specifications

- Max Concurrent Tasks settings. The following table describes the setting for each VMware proxy used during testing:

	VBR	Veeam Proxy #1	Veeam Proxy #2	Veeam Proxy #3	Veeam Proxy #4	Veeam Proxy #5	Veeam Proxy #6
Type	Physical	Physical	Physical	Physical	Physical	Virtual	Virtual
Cores	24	24	24	24	24	8	8
Max concurrent tasks	22	22	22	22	22	8	8

Table 3. Max Concurrent tasks per VBR proxy

Both NFS and SMB repositories were configured as SOBR, with a single extent.

NFS Observations

The following table summarizes five test runs with three permutations. Each row represents a test run in which a single variable is changed.

Protocol	Number of proxies	Decompress before storing to repo	Repository maximum concurrent task	Duration (hh:mm:ss)	Veeam reported processing rate (GB/s)	Calculated average throughput (GB/s)	Veeam reported bottleneck (%)
NFS	2	Enabled	22	00:41:19	1.5	1.19	target(99)
NFS	6	Enabled	22	00:37:21	1.6	1.34	source(85)
NFS	6	Disabled	22	00:33:46	1.8	1.51	source(88)
NFS	6	Disabled	Unlimited	00:30:28	2.2	1.7	source(92)

Table 4. VBR backup jobs, for NFS repository

As expected, increasing the number of proxy servers improved backup throughput and duration.

Enabling **Decompressing backup files before storing** to the repository had a slight negative impact on backup duration and throughput. Disabling this option showed that the FlashArray//C File Services could do more processing. Leaving both **Decompress backup data blocks before storing** and **Limit maximum concurrent tasks** disabled helped highlight the FlashArray//C File Services capabilities.

SMB Observations

As with NFS tests, the same five tests were performed against the SMB repository.

Protocol	Number of proxies	Decompress before storing to repo	Repository maximum concurrent task	Duration (hh:mm:ss)	Veeam reported processing rate (GB/s)	Calculated average throughput (GB/s)	Veeam reported bottleneck (%)
SMB	2	Enabled	22	00:39:37	1.7	1.25	source(78)
SMB	6	Enabled	22	00:39:16	1.7	1.26	target(94)
SMB	6	Disabled	22	00:37:06	1.7	1.35	target(91)
SMB	6	Disabled	Unlimited	00:26:18	2.7	2.03	target(95)

Table 5:. VBR backup jobs, for SMB repository

The results were:

- SMB was slightly faster than NFS, with just two VBR proxies.

- Adding more proxies did not reduce backup duration significantly, as was the case with NFS. Upon looking into job details of the top two test runs, the second test run with more VBR proxies drove more I/O to the SMB share, where the top test with only two proxies was busier reading data from the source snapshots than the other.

- Having **Decompress backup data blocks before storing** disabled only slightly improved backup throughput and duration.

- Leaving both **Decompress backup data blocks before storing** and **Limit maximum concurrent tasks** disabled was the best setting to exercise the FlashArray//C File Services, which is impressive.

NAS Repository Protection

Typically, we would want to protect the NAS repository itself from unintentional deletion or ransomware.

FlashArray//C file snapshots are the recommended mechanism to protect the VBR repository itself from accidental deletion or any malicious activities. Recovering the NAS repository involves mounting the read-only snapshot, copying VBR backup files to a read/write enabled share, and then scanning that share to reimport the backups.

Uncomplicate Data Storage, Forever

Conclusion

In summary, FlashArray//C File Services performed very well as a VBR NAS repository. Setup and configuration were very simple. With more and better resources, even better performance should be expected. In our environment, we were able to achieve backups at rates of ~8-10TB/h. This is comparable to the extensive testing documented in the VBR performance on FlashArray//C Block white paper. We didn't extensively test restore performance, but restore activities lagged a little behind at ~7TB/h. The difference is because the VMs are thin-provisioned, and VBR will default to network mode.

For the most part, we stuck with Veeam's best practices. We deviated from the best practice of limiting maximum concurrent tasks in order to measure how much we can push the performance of FlashArray//C File Services. In produciton, setting this to "unlimited" can harm the overall environmental performance.

As to whether to use NFSv3 or SMBv3.1, we expect that, with proper sizing and planning, both protocols will have reasonable performance, so you may choose the protocol that you are most familiar with.

Supporting Material

- FlashArray File Services Overview YouTube Video

- Veeam 11 User Guide for VMware vSphere

- Veeam Backup & Replication Best Practices

- Enhancing Veeam with FlashArray//C

- Veeam & Pure Storage Integrated Deployment Guide

- Setup iSCSI on Windows Server 2012, 2012 R2, 2016 and 2019 on FlashArray

- Pure Storage Blog: Why You Should Be Excited about FlashArray File Services

- Microsoft Blog: Controlling write-through behaviors in SMB

- Veeam Blog: When to use SMB WriteThrough in Windows Server 2019

- NFSv3 and NLM Documentation on Red Hat Customer Portal