PURESTORAGE®

TECHNICAL WHITE PAPER

Rapid Restore for MySQL and MariaDB

Deliver fast recovery of mission-critical applications on open-source databases.

Contents

Executive Summary **3**

Introduction **3**

Pure Storage FlashBlade **3**

How to Use This Guide **6**

Solution Architecture **6**

Test Environment Overview **7**

System Configuration Details 7

FlashArray Configuration 7

FlashBlade Configuration 8

MySQL Test Cases and Results **8**

Test Methodology 8

MySQL Backup and Restore (Physical Servers) **9**

MySQL Backup 9

Virtualized MySQL Backup and Restore **14**

Best Practices **15**

Conclusion **15**

Appendix A: MariaDB Test Results **16**

MariaDB Backup and Restore (Physical Servers) 16

Impact of Mariabackup on Read/Write Thread Count and Backup Performance 17

Virtualized MariaDB Backup and Restore 19

Appendix B: Hero Numbers **20**

Appendix C: Impact of nconnect **21**

About the Author **23**

Executive Summary

Open-source database usage is growing rapidly. MySQL and its variants like MariaDB are used extensively by many customers, including global software-as-a-service (SaaS) providers and social media giants. As digital transformation accelerates, these organizations are laser-focused on eliminating downtime for mission-critical applications to get the most value out of their data. Fast recovery technology is required in order to drive this additional value. Pure Storage® FlashBlade® with its all-flash, scale-out architecture is the ideal solution for the rapid restoration of mission-critical applications.

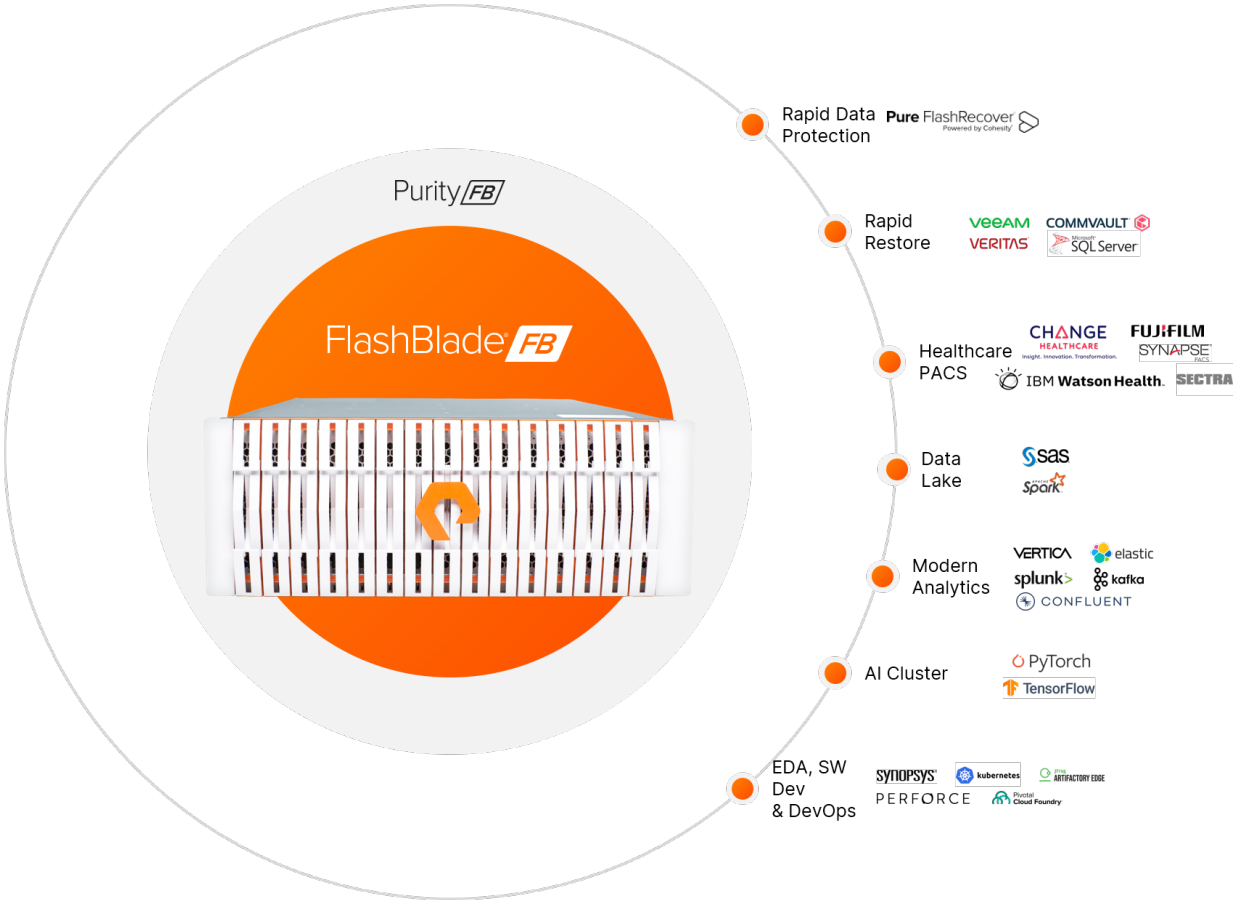
This white paper will show how FlashBlade delivers on the promise of fast recovery for mission-critical applications based on open-source databases like MySQL and its variants.

Introduction

Open-source databases such as MySQL and MariaDB have been on the scene for a number of years as rapidly-deployable databases used in enterprise testing environments. More recently, these database management systems have been widely adopted in mission-critical production environments for modern web applications and other types of applications. Based on their application/service design, the number of MySQL or MariaDB database instances can run into hundreds or thousands. This makes the job of a database administrator (DBA) very complex as there is less time available to perform the essential tasks of backing up data and routine backup testing validation. In the case of SaaS applications, individual customers can initiate backup and recovery operations themselves, further complicating the planning and scheduling. While traditional backup appliances are designed to efficiently store data, they are notoriously slow at rehydrating and recovering data. The data restore process is used more than ever before as DBAs and software developers iterate on their code with up-to-date databases to accelerate the software development process. Pure Storage FlashBlade helps reduce the time for both backup and restore operations by providing high-bandwidth performance and simple management. Legacy backup solutions place businesses at risk of missing recovery and access service level agreements (SLAs) for their critical production databases. Pure Storage offers a modern rapid recovery architecture, tapping the power of scale-out, all-flash storage to restore data rapidly, meet SLAs, and save money.

Pure Storage FlashBlade

Pure Storage FlashBlade is a scale-out, all-flash storage system powered by a distributed file system that is purpose-built for massive concurrency across all data types. FlashBlade is a unified fast file and object (UFFO) storage platform that can simultaneously host multiple file systems and multi-tenant object stores for thousands of clients. FlashBlade can scale up to multi-petabyte capacity with linear-scale performance, simply by adding a single blade at a time, up to 150 blades. Because of its native scale-out architecture and ability to drive performance for any type of workload, it enables enterprises to consolidate a range of workloads, from backup to analytics to AI, on a single platform.

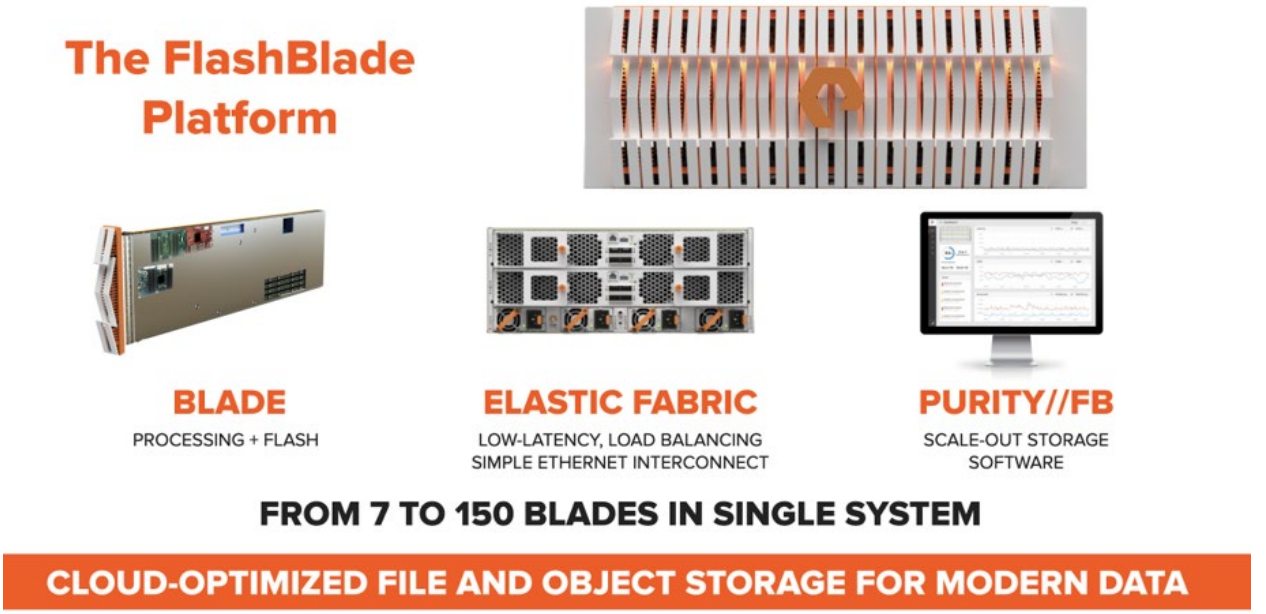


Many customers build their data protection strategy with FlashBlade, enjoying rapid backup and restore performance while investing in a platform that enables them to consolidate data lakes and other data silos.

A FlashBlade system's ability to scale performance and capacity is based on six key innovations:

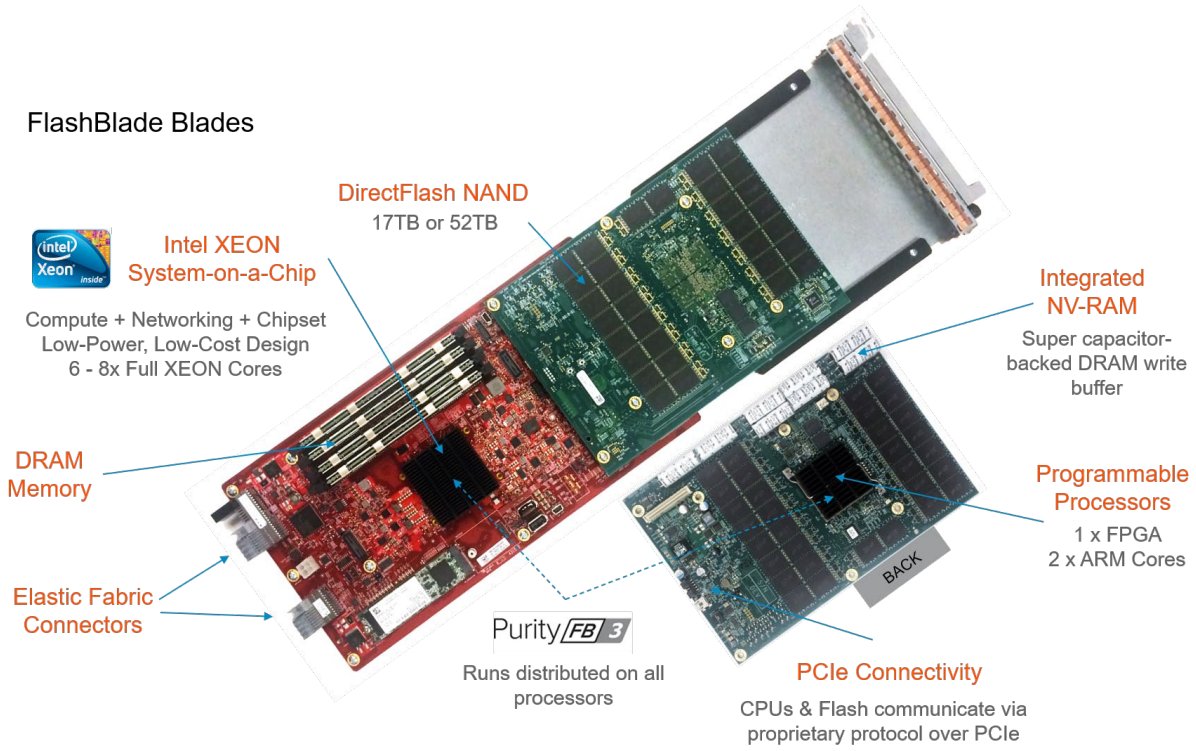
- **High-performance storage**: FlashBlade maximizes the advantages of an all-flash architecture by storing data in storage units instead of crippling, high-latency storage media such as traditional spinning disks and conventional solid-state drives. Integrated, scalable NVRAM helps scale performance and capacity proportionally when new blades are added to a system.
- **Unified network**: FlashBlade consolidates communication traffic between clients and internal administrative hosts into a single, reliable high-performing network that supports both IPv4 and IPv6 client access over Ethernet links at speeds of up to 100Gb/s.
- **Purity//FB storage operating system**: A symmetrical operating system running on the FlashBlade system's fabric modules, Purity//FB minimizes workload balancing problems by distributing all client operation requests among the blades on FlashBlade.
- **Common media architectural design for files and objects**: The single underlying media architecture of FlashBlade natively supports concurrent access to files via a variety of protocols such as NFSv4.1, NFS over HTTP, and SMB and objects via S3 across the entire FlashBlade configuration.

- **High-performance replication of objects and files**: The latest version of Purity storage software delivers replication of FlashBlade files and objects to enable disaster recovery either to a secondary site or the public cloud.
- **Simple usability**: Purity//FB on FlashBlade alleviates system management headaches because it simplifies storage operations by performing routine administrative tasks autonomously. With a robust operating system, FlashBlade is capable of self-tuning and providing system alerts when components fail.



A full FlashBlade system configuration consists of up to 10 self-contained, rack-mounted chassis interconnected by high-speed links to up to four external fabric modules (XFM). At the rear of each chassis, two on-board fabric modules provide high-speed Ethernet interconnects to the blades, other chassis, and clients using TCP/IP. Fabric modules are interconnected, and each contains a control processor and Ethernet switch ASIC. For reliability, each chassis is equipped with redundant power supplies and cooling fans.

The front of each chassis holds up to 15 blades for processing data operations and storage. Each blade assembly is a self-contained compute module equipped with processors, communication interfaces, and either 17TB or 52TB of flash memory for persistent data storage. The current FlashBlade system can support over 1.5 million NFSv3 *getattr* operations per second, or >17GiB/s of 512KiB reads or >8GiB/s of 512KiB overwrites on a 3:1 compressible dataset in a single 4U chassis with 15 blades. FlashBlade can scale both compute and performance up to a maximum configuration of 10 x 4U chassis with 150 blades.

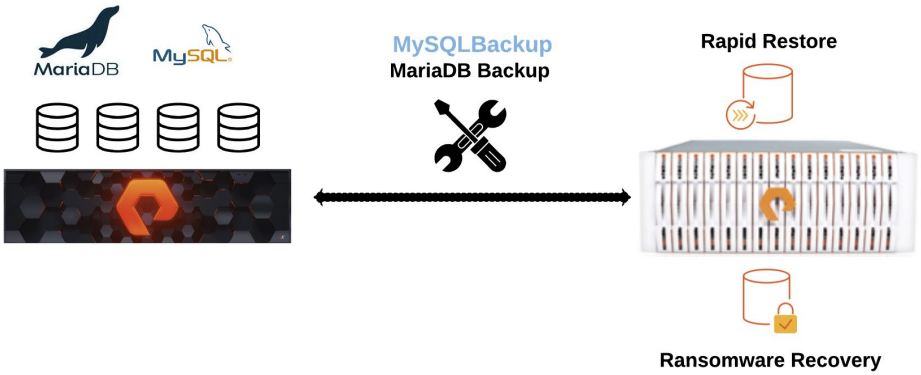


How to Use This Guide

This technical white paper is for IT, storage, and DBA specialists using open-source database's native backup and recovery features. It provides guidelines and techniques to optimize backup and restore performance with MySQL, MariaDB, and FlashBlade.

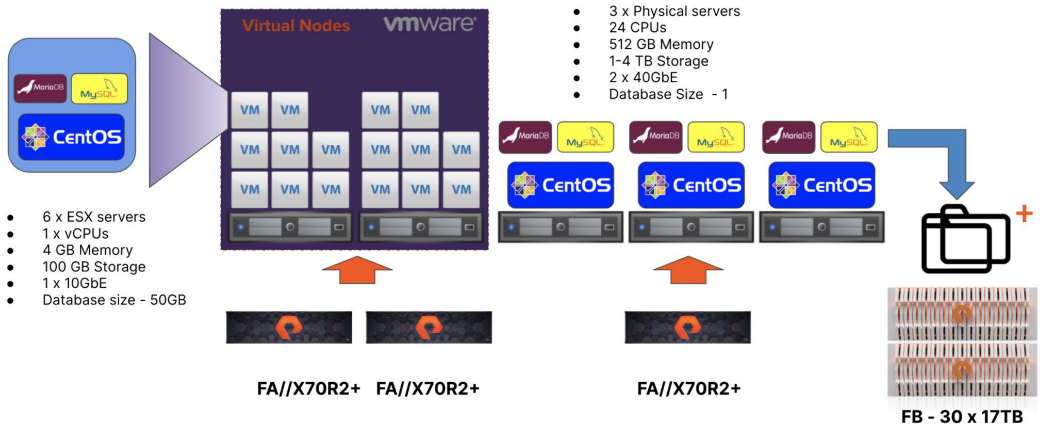
Solution Architecture

The following diagram represents the high-level architecture with FlashBlade and Rapid Restore. The primary database is deployed on Pure Storage FlashArray in all our testing. The best practices we outline in this document are also applicable to MySQL and MariaDB databases protected by FlashBlade and running on any type of storage, including local drives.



Test Environment Overview

The diagrams below (Figures 5 and 6) illustrate the FlashBlade test environment used for this white paper. We designed it to test the performance and scalability of FlashBlade and to determine the best practices for MySQL and MariaDB Server backup deployment. In this environment, the FlashBlade system is configured to store MySQL and MariaDB server backups in NFS volumes. The native backup and recovery tools were used to initiate all backup and restore operations. In multi-server testing, virtualized instances of the databases were used. The source databases were provisioned out of FlashArrays in all our testing. The details of deploying MySQL on FlashArray is beyond the scope of this document. Please refer to the best practices guide for more information.



System Configuration Details

MySQL/MariaDB Server

Each database server includes the following components:

Compute:

- 2x Intel Xeon E5-2697 v2 @ 2.70GHz (24 physical cores total), HyperThreading enabled
- 512GB RAM

Network:

- 2 x Mellanox ConnectX-3 family network adapter (40Gbps)

Software:

- MySQL Community Server 8.0.25
- MariaDB 10.5.9
- CentOS 8.3

FlashArray Configuration

The following table shows the configuration of the two Pure Storage FlashArray systems used for hosting the MySQL and MariaDB databases in this solution testing:

Component	Description
FlashArray	//X70R2
Capacity	174.45TB raw (base chassis)
Connectivity	4x 16Gb/s redundant Fibre Channel ports 4x 10Gb/s Ethernet ports 1Gb/s redundant Ethernet (management port)
Software	Purity//FA 6.0.3

Table 1. FlashArray configuration

FlashBlade Configuration

FlashBlade acts as a backup target for MySQL/MariaDB backups, using the NFS protocol to receive the backup data. The FlashBlade system used in testing is configured as shown in Table 2.

Component	Description
FlashBlade	30x 17TB blades (multi-chassis system)
Capacity	527.77TB raw
Connectivity	4x 100Gbps Ethernet (data) 2x 1Gb/s redundant Ethernet (management port)
Physical	10U
Software	Purity//FB 3.2.x

Table 2. FlashBlade configuration

The biggest advantage of FlashBlade is that all its performance can be harnessed with fewer numbers of volumes and network configurations. In this testing, a single volume was used for hosting all the backups of physical and virtual MySQL/MariaDB backups. All servers involved accessed their backup volume using just one IP address (Data VIP) on the FlashBlade system.

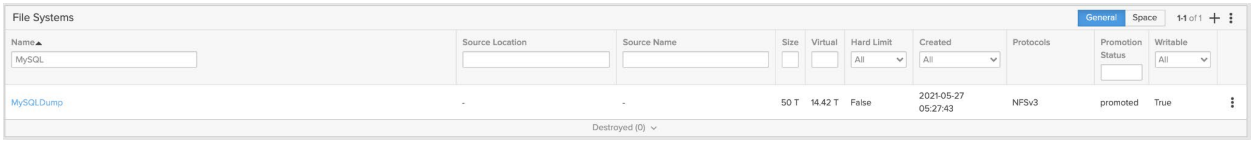


Figure 6. FlashBlade SMB volume configuration

MySQL Test Cases and Results

Test Methodology

This section discusses the test results from MySQL testing. Please refer to Appendix A for results from MariaDB testing. This paper mainly focuses on full database backup and restore operations because they happen to be very strenuous and resource consuming. The other backup types like individual database backup, partial database backup, and incremental backup are not

covered in this document. The FlashBlade system should provide excellent performance improvement irrespective of the backup/recovery type chosen by the customer. Some of the backup and recovery parameters are tuned to achieve better backup and recovery rates, as documented in the paper. The testing also includes database consolidation use cases. It is common for enterprise customers to have 10s to 100s of MySQL databases.

MySQL Backup and Restore (Physical Servers)

This section describes the test results from backup and recovery of large MySQL instances deployed on physical servers. As indicated earlier, all our testing is focused on full database backup and recovery.

MySQL Backup

The MySQL database provides two major choices for backup. Each comes with its strengths and weaknesses.

- *Mysqldump:* A logical backup of the database
- *Mysqlbackup*: An enterprise backup tool with a rich set of features

Mysqldump

The mysqldump application is a backup program that is generally used to dump a database or a collection of databases for backup purposes. The dump file typically contains SQL statements in plain text to create tables, indices, and other database metadata objects and populate them. Database administrators can also use the dump file to transfer data between MySQL databases or its variant databases or into non-MySQL databases like Postgres. This capability is seen as a major advantage of the mysqldump tool by some administrators as it gives them an ability to port their data across databases. The mysqldump application can also be used to generate files in CSV, other delimited text, or XML format. A detailed description of mysqldump tools and features and their description is beyond the scope of this document. The tool is simply used to take a full backup of a complete database instance (all databases) in our testing by using the following command:

```
mysqldump -uroot -p --all-databases > /mnt/MySQLDump/Test/$HOST_NAME/mdp.sql
```

Mysqlbackup

The mysqlbackup client is an enterprise edition of a backup and recovery tool for MySQL databases with lots of optimizations built in. The tool can be used to back up an entire database or parts of it into a backup directory or even to a single file stream. It also provides options to parallelize backup and recovery operations. This paper focuses on full database backup and recovery. Use the following command:

```
mysqlbackup --user=mysqlbackup --password=YourPassword --host=127.0.0.1 --backup-
dir=/mnt/MySQLDump/Test/`hostname` --read-threads=8 --write-threads=8 backup
```

Mysqlbackup performs more than 10 times faster than mysqldump when backing up a single large 820GB MySQL database, as seen in the chart below.

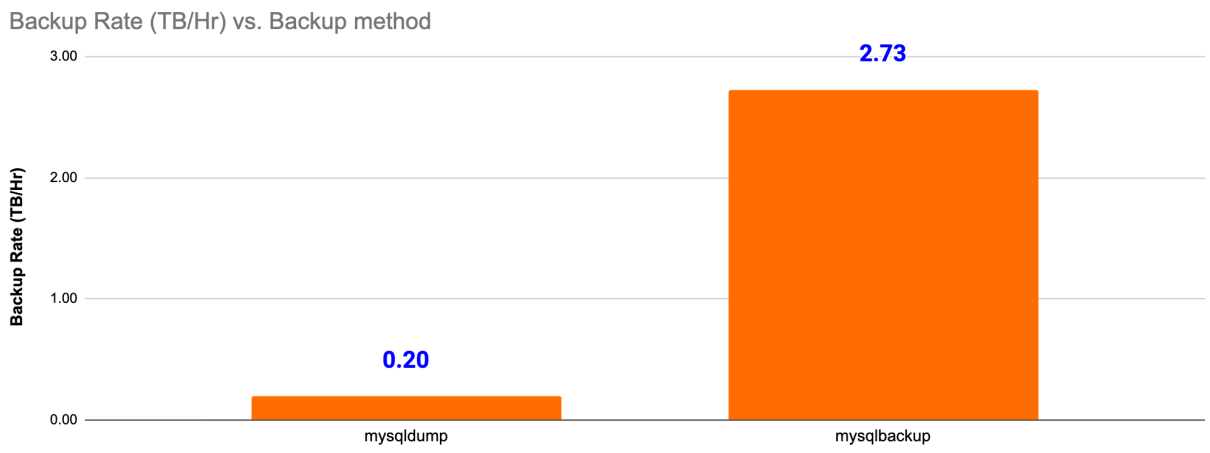


Figure 7. Backup performance comparison between mysqldump and mysqlbackup

Impact of Mysqlbackup on Read/Write Thread Count and Backup Performance

The mysqlbackup tool has some options to optimize for performance and storage capacity. Database servers with more CPU cores can take advantage of the options by allocating more threads for data reading, writing, and processing (like compression and encryption). Our testing indicates that while leveraging multiple threads has some advantages, the mysqlbackup tool still only processes one database file at a time. The following chart shows the impact of changing read/write thread counts. In the tests below, the read and write thread counts were changed while leaving the process thread count at its default value.

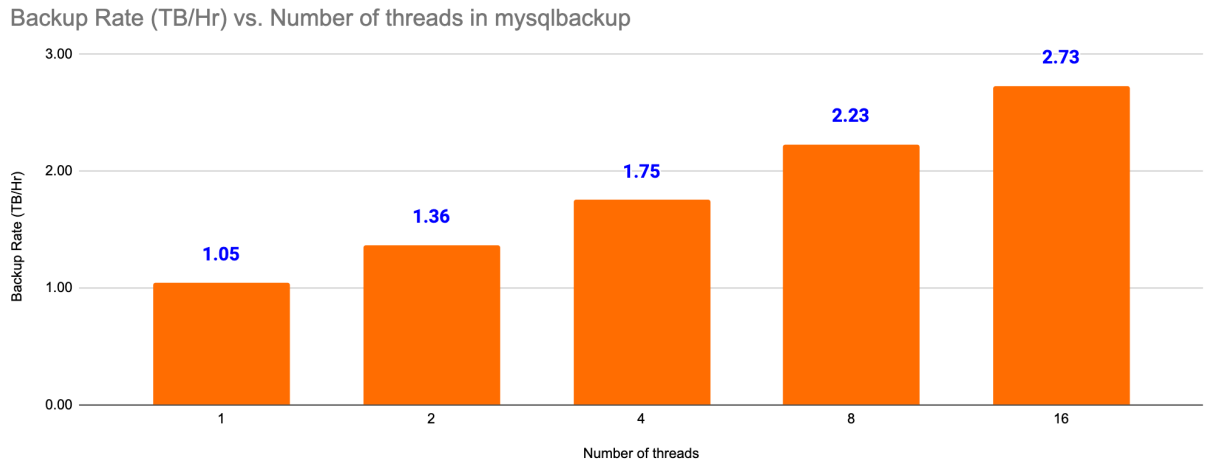


Figure 8. Impact of changing read/write thread count on database backup rate

Impact of Mysqlbackup on Backup Compression and Backup Performance

The mysqlbackup tool can compress backups before storing them on the backup target. This may reduce storage capacity required for backups and, more importantly, the network bandwidth at the cost of host CPU cycles. It is important to realize that the FlashBlade system always stores data compressed natively to optimize the raw flash consumed. Native compression gets most of the capacity savings irrespective of whether the compression option is used during backup or not. So, it is up to the user to choose a backup option depending on which resource they want to optimize for. Our testing indicates that by choosing backup level compression LZ4, the backup consumed 40% less storage and finished 30% faster at the expense of host CPU cycles.

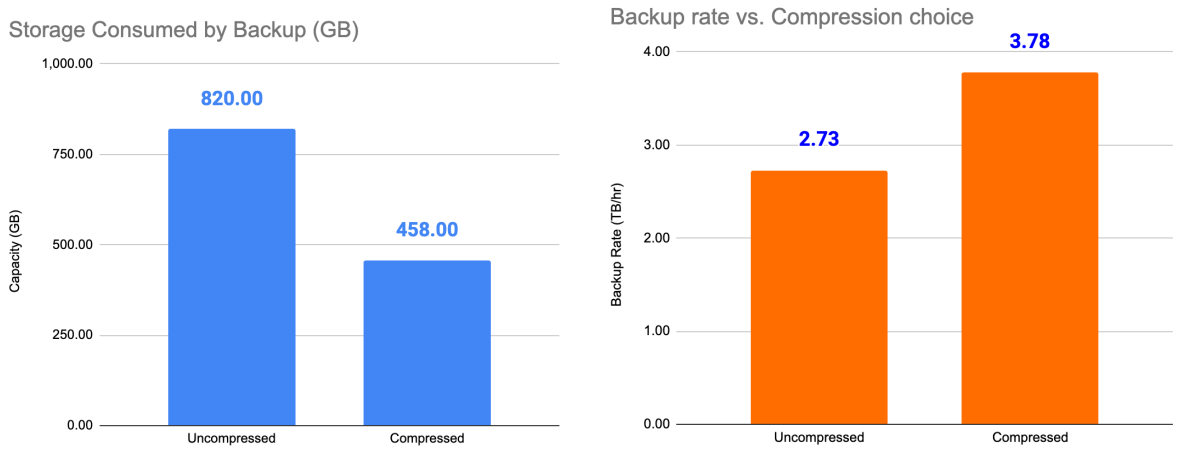


Figure 9. Impact of backup compression

Mysqlbackup: FlashBlade Scalability for Backups

The FlashBlade system's multi-dimensional scalability allows customers to scale their capacity and performance needs as they grow. The chart below shows the near-linear scalability of FlashBlade as we scale mysql physical instances. The backup rate scales linearly with more database servers

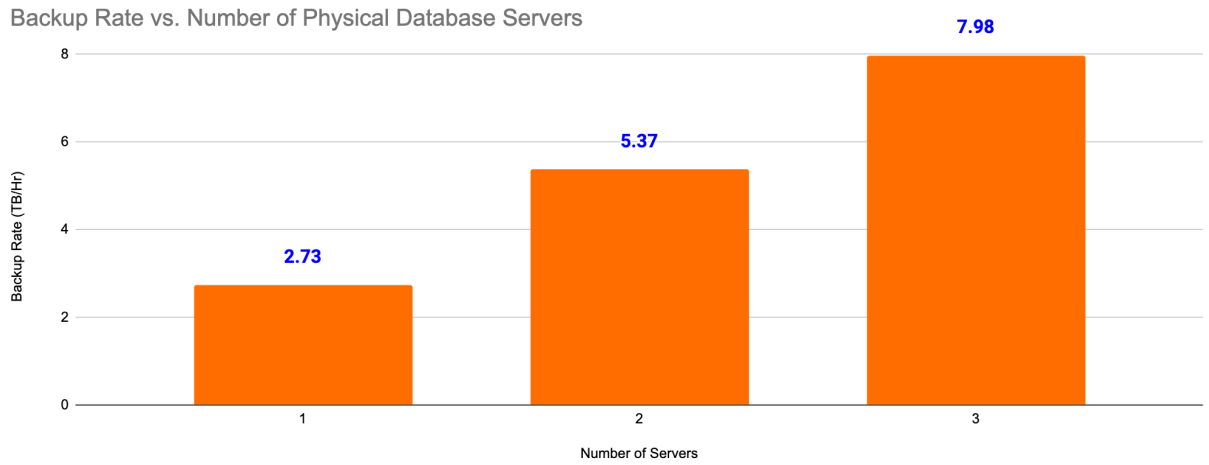


Figure 10. FlashBlade scalability with number of MySQL Instance concurrent backup

MySQL Restore

The MySQL backups created by the mysqldump tool can be restored by simply using "mysql" client and executing the data definition language (DDL) and data manipulation language (DML) statements in the dump file. This tool tends to be very slow and has few options to parallelize. Alternatively, backups done with the mysqlbackup enterprise tool can be restored using the mysqldump tool and provides a rich set of options to accelerate the restore rate. Note that the restore performance is also a function of the storage system hosting the database files. While the backup targets like FlashBlade can be faster, the restore performance is often dictated by the source database storage performance.

Mysql

The backups created by the mysqldump client are nothing but a collection of SQL statements to recreate those database objects and load data. The dump file, if in SQL format, can be executed directly from the mysql command

```
mysql -uroot -p < /mnt/MySQLDump/Test/$HOST_NAME/mdp.sql
```

Mysqlbackup

The same mysqlbackup client can be used for restoring from backups made by using the command line shown below. The mysqlbackup tool is extremely fast compared to restoring from a MySQL dump file created by the mysqldump tool. Full database backup and restore operations were performed in all our testing. So, there are some maintenance tasks required before and after running the restore command, like shutting down the MySQL instance and changing ownership of restored files. Please refer to the mysqlbackup manual for more details.

```
mysqlbackup --user=mysqlbackup --password=Yourpassword --host=127.0.0.1 --backup-
dir=/mnt/MySQLDump/Test/`hostname`/mb --read-threads=4 --write-threads=4 copy-back-and-apply-log
```

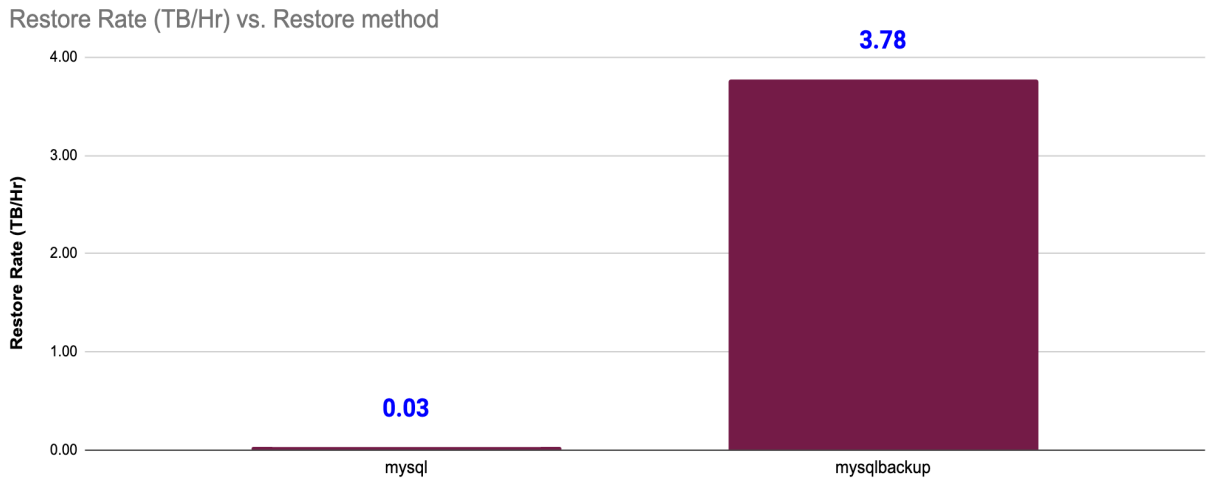


Figure 11. Restore performance comparison between mysql (from mysqldump) and mysqlbackup

Impact of Mysqlbackup on Read/Write Thread Count and Restore Performance

Changing the read and write threads has some impact on the restore performance, but more threads does not always mean more performance, as indicated by the chart below. Users are recommended to calibrate their system to determine the right number of thread counts for their database environment.

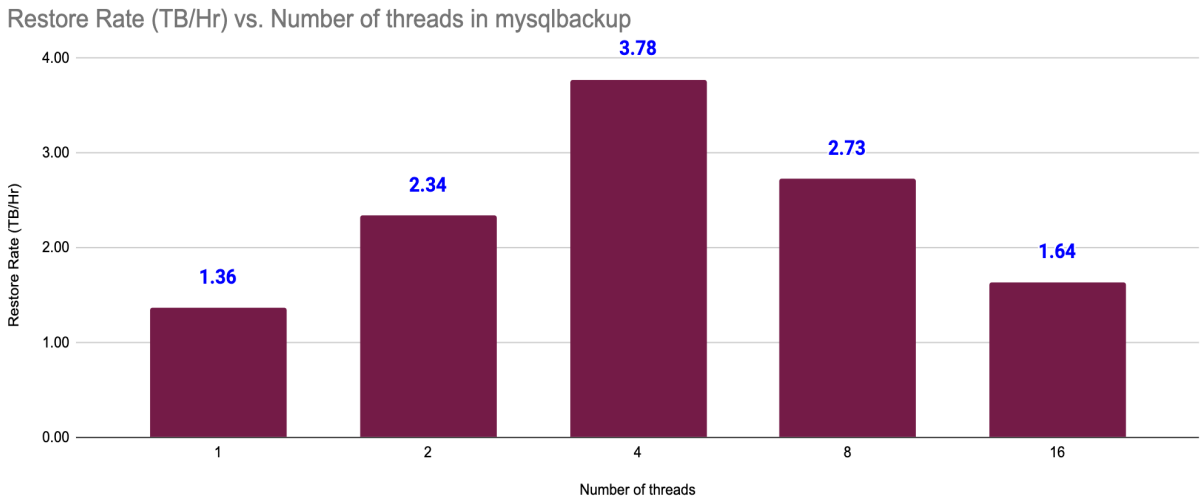


Figure 12. Impact of changing read/write thread count on database restore rate

Impact of Mysqlbackup on Compression and Restore Performance

Restoring a pre-compressed backup done with the mysqlbackup toolwill mean fetching less data from the backup target, but the system must spend CPU cycles on the host to decompress the data before restoring it. The chart below compares the restore rate for a compressed and uncompressed backup. It is important to note that the restore performance also depends on the performance of the source database storage.

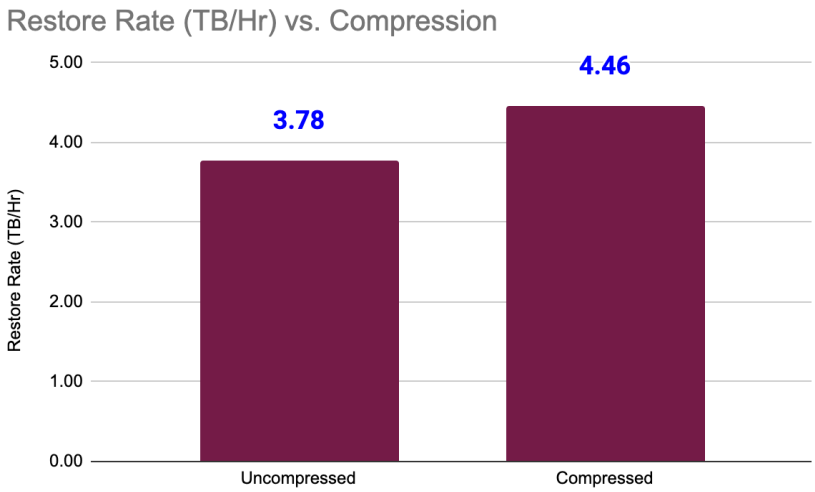


Figure 13. Impact of backup compression on restore rate

FlashBlade Scalability for Restores with Mysqlbackup

In this test, the MySQL instances were scaled and, as shown in the chart below, the performance of FlashBlade linearly scales with more physical servers. The performance observed is limited by server and network connectivity to FlashBlade, and it is nowhere near the performance capabilities of FlashBlade. The restore performance will scale higher with more concurrent restore operations.

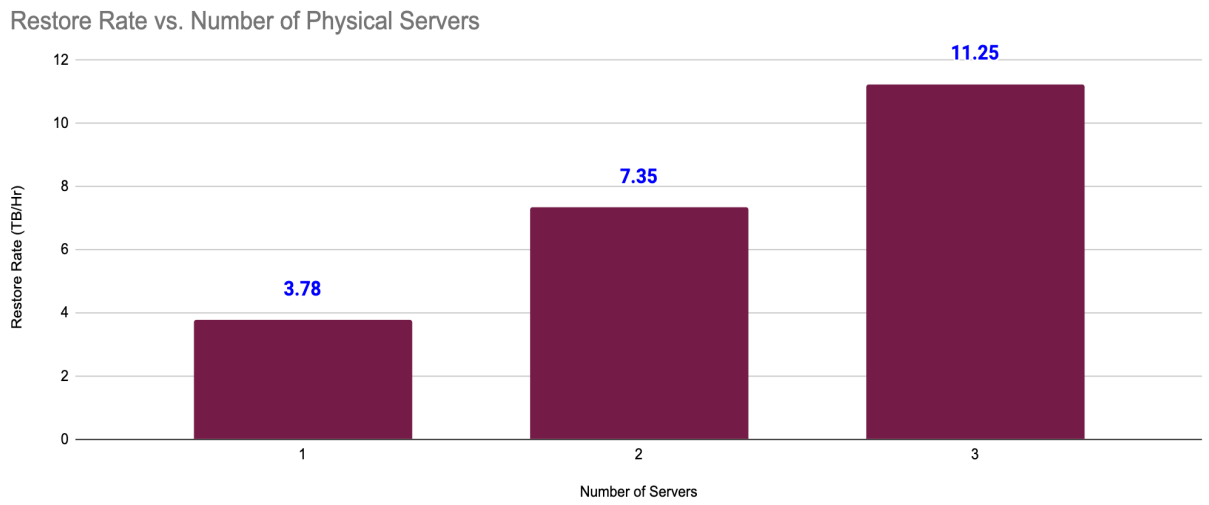


Figure 14. FlashBlade Scalability with number of MySQL Instance concurrent restore

Virtualized MySQL Backup and Restore

Most of the implementations deploy a handful of MySQL databases per tenant or user, so there can be several hundreds to thousands of databases in a SaaS situation. These transactional databases are critical and the cornerstones for the services they offer, so it is important to protect these databases. In this section, we demonstrate backing up and restoring up to 48 50GB MySQL databases concurrently. There were six servers virtualized using VMware ESXi, each running eight Linux VMs. There was a 50GB MySQL instance in each of these virtualized Linux servers. The following charts show the single ESXi server scalability with VMs scaling from one VM to eight VMs on a single server.

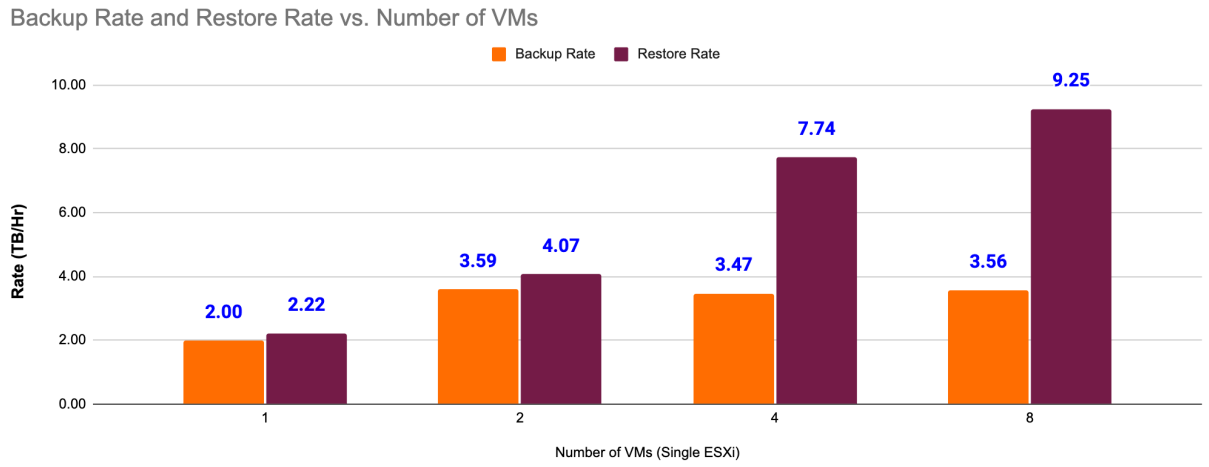


Figure 15. Virtualized MySQL backup and recovery scalability on single ESXi server with VMs

The chart below shows the backup and restore performance scalability as we scale the number of ESXi servers, each running eight VMs hosting one MySQL instance per host. As you can see, the backup performance linearly scales, whereas the restore performance is limited by the source array write performance hosting MySQL databases. The FlashBlade system can support far more concurrent restores if we can add more source arrays to absorb the heavy ingest rate. In this testing, only two FlashArray//X70 sysetms were used to host those 48 databases.

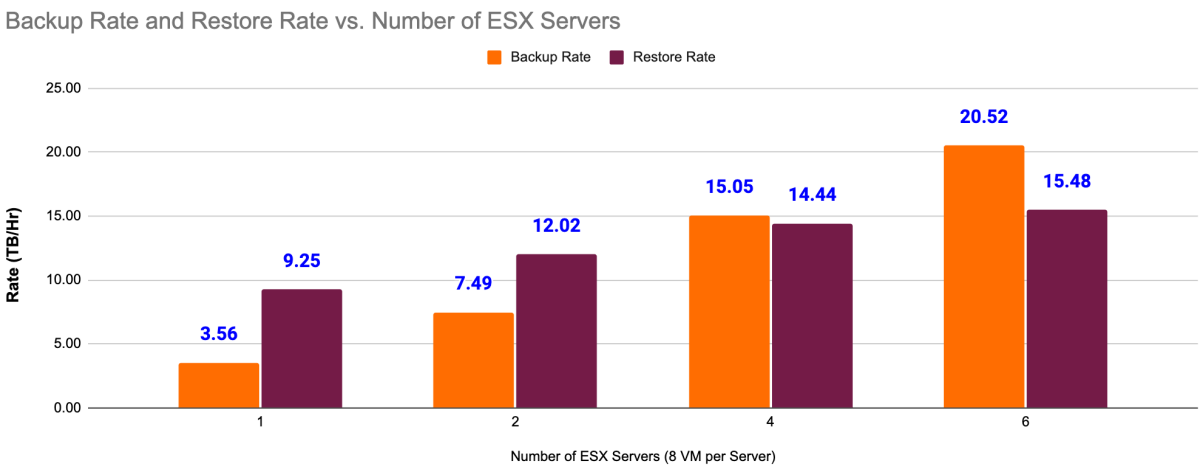


Figure 16. Virtualized MySQL backup and recovery scalability with VMs on ESXi servers (eight VMs per ESXi server)

Best Practices

FlashBlade is built on the philosophy of simplicity. Achieving the best possible performance on FlashBlade does not require any special tuning for MySQL backup and restore. Based on the completed testing, there is guidance below on how to achieve optimum performance; however, it is strongly recommended that you perform calibration tests to figure out the ideal parameters for your deployment.

1. For MySQL, , tuning the reader and writer threads gave some improvements. We recommend setting the following parameters while using mysqlbackup (Enterprise) tool:
   a. Set reader-threads and writer-threads to 16 for database backup.
   b. Set reader-threads and writer-threads to 4 for database restore.
2. Enabling backup compression provides betterr backup and restore performance at the expense of host CPU cycles. We recommend making your selections based on the resource you want to optimize.
3. We recommend host Ethernet interface bonding/teaming to simplify network configuration. It is also a good idea to isolate the backup network from the management network to keep the configuration clean.
4. There is no need to perform any special configuration of data VIPs (Virtual IPs) on FlashBlade. A single IP address should be able to achieve maximum possible performance.

Conclusion

Organizations today are in an era in which data availability drives innovation, provides a competitive advantage, and is essential to business operations. For these modern enterprises, Rapid Restore with FlashBlade is a compelling alternative to a legacy data protection architecture. The native backup and recovery features built into MySQL and MariaDB databases, combined with the highly performant FlashBlade system, provide customers with a very good choice for backup and recovery needs. This combination gives customers the best possible performance while keeping the architecture simple and straightforward. The Flashblade unified fast file and object (UFFO) storage platform can be used not only for massive backup workload consolidation, but also to consolidate many other workloads, including file serving, analytics, and AI, onto a single scalable platform.

Appendix A: MariaDB Test Results

MariaDB is a variant of MySQL that has evolved into a fully functional database with its own unique features. MariaDB Enterprise Server is an enhanced, hardened, and secured version of MariaDB Community Server that provides customers with enterprise reliability, stability, and long-term support as well as greater operational efficiency when it comes to managing large database deployments for business- and mission-critical applications. The goal of this testing is not to compare MySQL vs. MariaDB. The results we observed in our lab are being presented here.

MariaDB Backup and Restore (Physical Servers)

This section describes the test results from backup and recovery of large MariaDB instances deployed on physical servers. As indicated earlier, all our testing is focused on full database backup and recovery.

The MariaDB Enterprise Server provides two major choices for backup. Each comes with their strengths and weaknesses.

- *Mysqldump*: A logical backup of the database
- *Mariabackup*: An enterprise backup tool with rich set of features

Mysqldump

The mysqldump application is a backup program, as described before, that has been available for a while and is also part of the MariaDB distribution. The tool is simply used to take a full backup of a complete database instance (all databases) in our testing using the command:

```
mysqldump --all-databases > /mnt/MySQLDump/Test/YourBackupDir/mdp.sql
```

Mariabackup

The mariabackup client is an enterprise edition of a backup and recovery tool for MariaDB databases with many optimizations built in. The tool can be used to back up an entire database or parts of it into a backup directory or even to a single file stream. It also provides options to parallelize backup and recovery operations. This paper focuses on full database backup and recovery.

```
mariabackup --backup --target-dir=/mnt/MySQLDump/Test/YourBackupDir --user=mariabackup --
password=yourpassword --parallel=4  2> /tmp/mdb.out
```

There is a difference of more than ten times in performance between mysqldump and mariabackup for backing up a single large 820GB MariaDB database, as shown in the chart below.

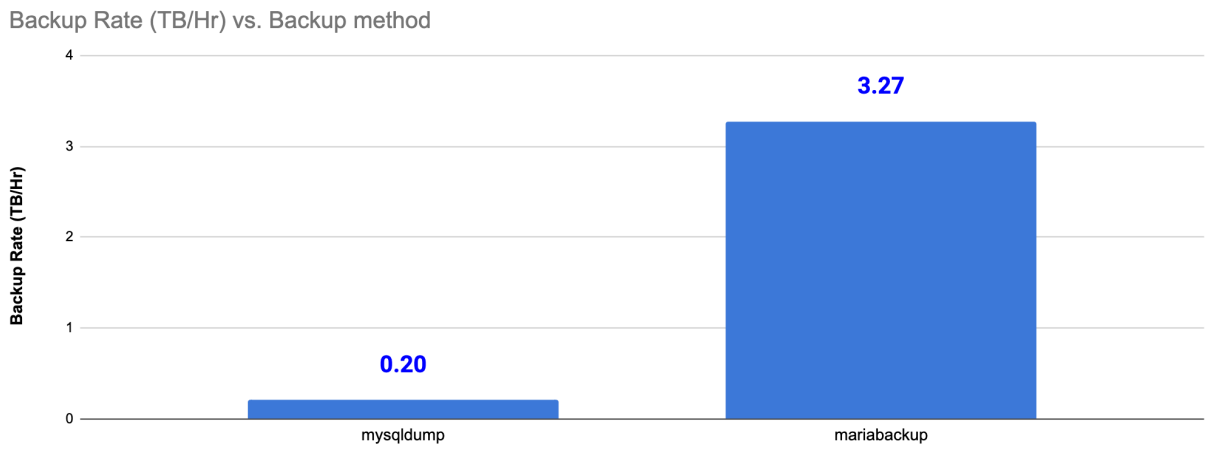


Figure 17. Backup performance comparison between mysqldump and mariabackup

Impact of Mariabackup on Read/Write Thread Count and Backup Performance

The mariabackup tool has some options to optimize for performance and storage capacity. Database servers with more CPU cores can take advantage of them by allocating more threads for reading, writing, and processing of data (like compression and encryption). Our testing indicates that while leveraging multiple threads has some advantages; the following chart shows the impact of changing thread count. MariaDB was able to achieve maximum performance even with smaller thread counts.

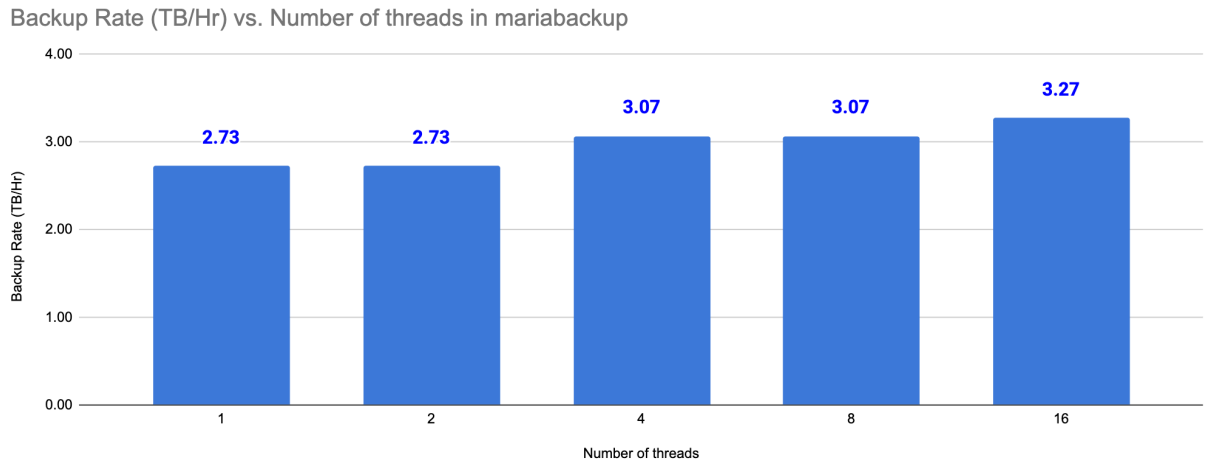


Figure 18. Impact of changing thread count on database backup rate of mariabackup tool

FlashBlade Scalability for Backups with Mariabackup

The FlashBlade multi-dimensional scalability allows customers to scale their capacity and performance needs as they grow. The chart below shows the near-linear scalability of FlashBlade as we scale mariaDB physical instances. The backup rate scales linearly with more database servers.

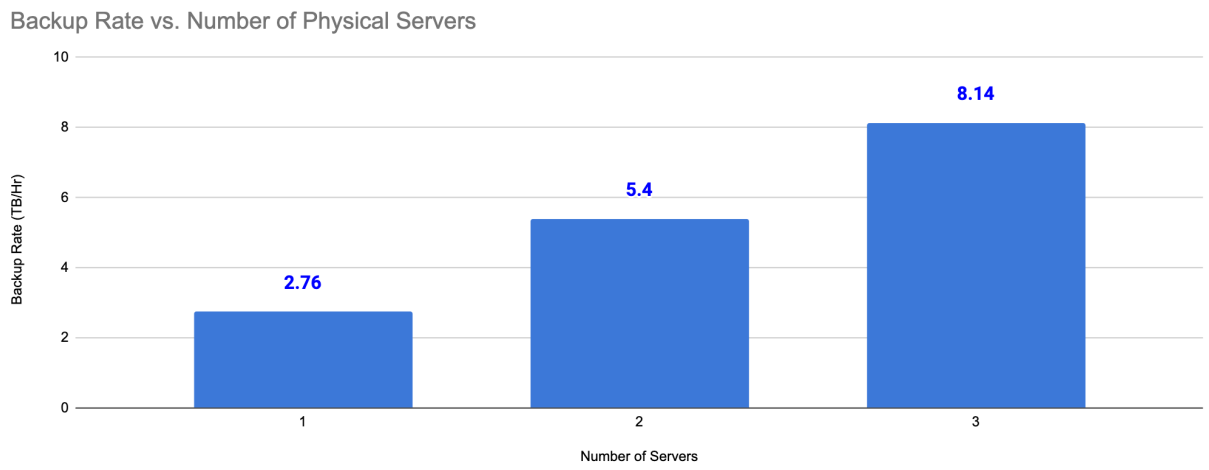


Figure 19. Backup rate vs. number of servers with Mariabackup.

MariaDB Restore

The MariaDB backups created by the mysqldump tool can be restored by simply using the mysql client and executing the DDL and DML statements in the dump file. This tool tends to be very slow and has few options to parallelize. Alternatively, backups done with the mariabackup enterprise tool can be restored using the same tool and provides a rich set of options to accelerate the restore rate. Note that the restore performance is also a function of the storage system hosting the database files. While the backup targets like FlashBlade can be faster, the restore performance is often dictated by the source database storage performance.

Mysql

The backups created by the mysqldump client are nothing but a collection of SQL statements to re-create those database objects and load data. The dump file, if in SQL format, can be executed directly from the mysql command `mysql < /Yourdumpfilelocation /Yourfumpfile.sql.`

Mariabackup

The same mariabackup client used for backup of the database can be used to restore from backups made using that tool by using the command line shown below. The mariabackup tool is extremely fast compared to restoring from a MariaDB dump file created by the mysqldump tool. Full database backup and restore operations were performed in all our testing. So, there are some maintenance tasks required before and after running the restore command, like shutting down the MySQL instance and changing ownership of restored files. Please refer to the mariabackup manual for all the details.

```
mariabackup --copy-back  --target-dir=/Yourdumpfileslocation --user=mariabackup --
password=mypassword --parallel=4  2> /tmp/mdb.out
```

Our tests indicate that the MariaDB restore operation was single threaded, so the bandwidth was limited to 1Gb/s. The scalability tests with multiple threads for restore did not yield any better performance.

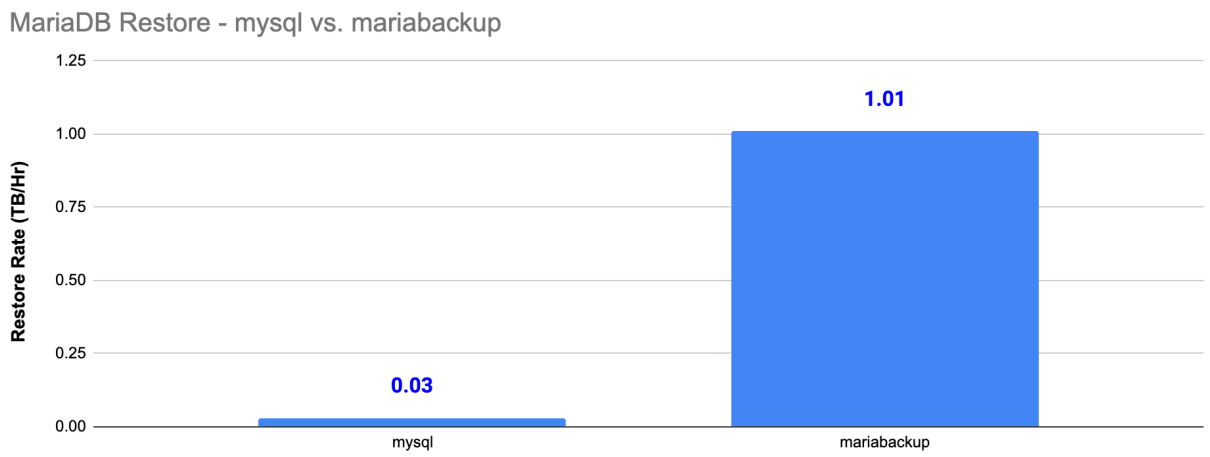


Figure 20. Restore performance comparison between mysql (from mysqldump) and mariabackup

Virtualized MariaDB Backup and Restore

MariaDB and its variants are commonly used by web applications, where there are hundreds to thousands of databases, each representing an application or a tenant in a SaaS service situation. These small transactional databases are the cornerstone for the service they offer. Therefore, it is important and becoming common to protect these databases on a periodic basis. In this section, we demonstrate backing up and restoring up to 48 databases simultaneously MariaDB instances which are 50GB each. There are a total of six servers virtualized using VMware ESXi each running eight Linux VMs. There is a 50GB MariaDB instance in each of these virtualized Linux servers. The following charts below show the single ESXi server scalability with VMs scaling from 1 to 8 VMs on a single server.

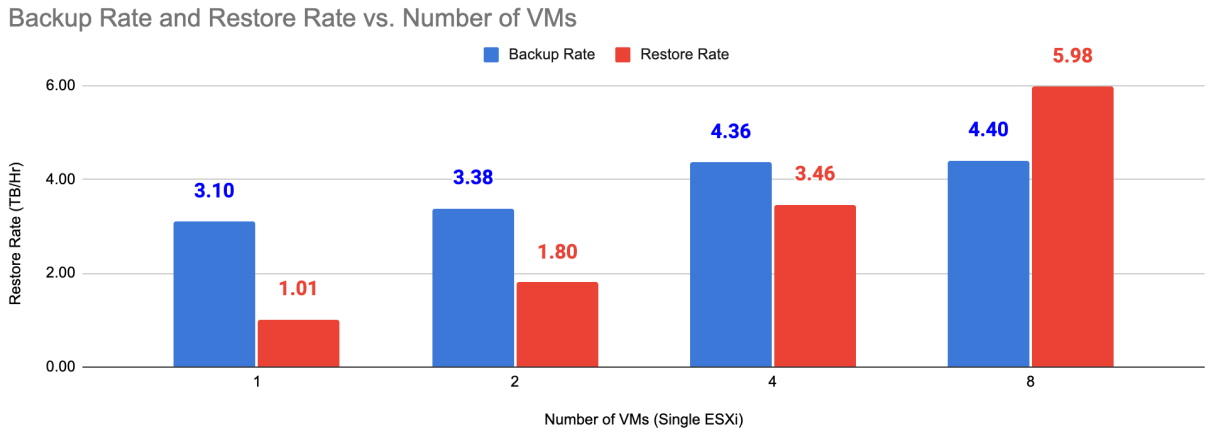


Figure 22. Virtualized MariaDB backup and recovery scalability on single ESXi server with VMs

The chart below shows the backup and restore performance scalability. As we scale the number of ESXi servers each running 8 VMs hosting one MariaDB instance per host. As you can see, the backup performance linearly scales, whereas the restore performance is limited by the source array write performance hosting MariaDB databases. The FlashBlade system can support far more concurrent restores if we can add more source arrays to absorb the heavy ingest rate. In this test, only two FlashArray//X70 systems were used to host those 48 databases.

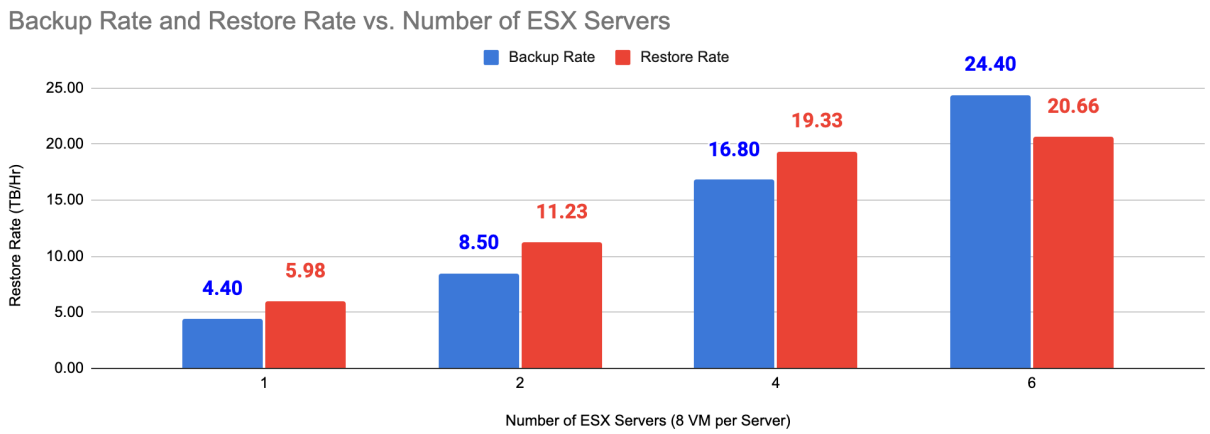


Figure 22. Virtualized MariaDB backup and recovery scalability with VMs on ESXi servers (8 VMs per ESXi server)

Appendix B: Hero Numbers

The main goal of this test was to see how far the backup and restore performance can be pushed with maximum servers and storage servers, based on what is available in our test environment. The test setup was all virtualized, as shown by the diagram below. There were nine servers and three FlashArray units for hosting the source databases. Each of the servers had 12 VMs totaling up to 108 virtual machines running a MySQL instance on each. A 50GB database was created under each instance. The databases were concurrently backed up and restored on to a 30-blade FlashBlade system.

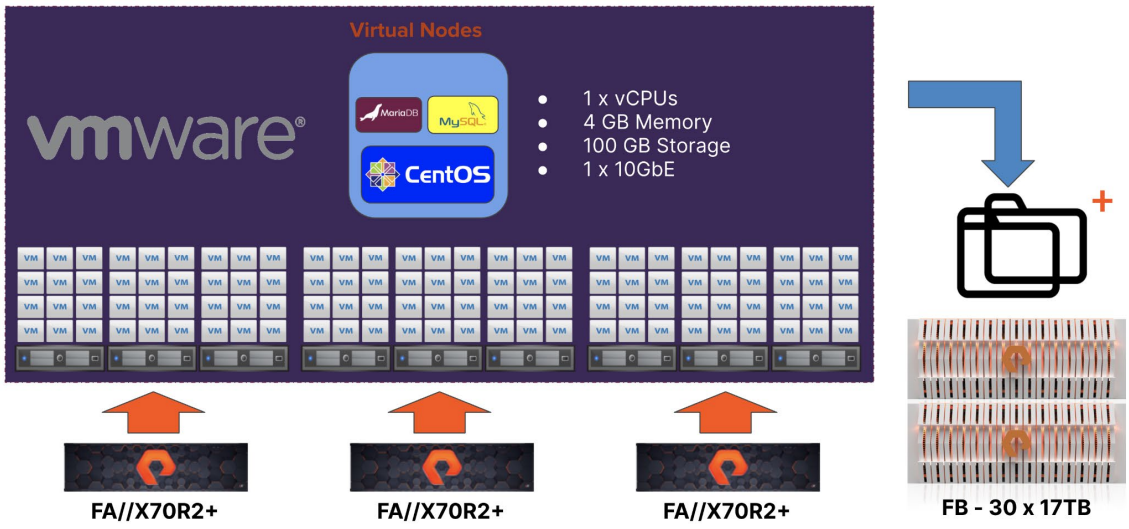


Figure 23. Test setup for performing Hero Number tests

The chart below shows the maximum performance achieved with the system under test. The extreme performance becomes an important factor when recovering from a ransomware attack. We performed tests to protect snapshots of backup directories using FlashBlade SafeMode™ and performed our recovery tests after simulating a ransomware attack.

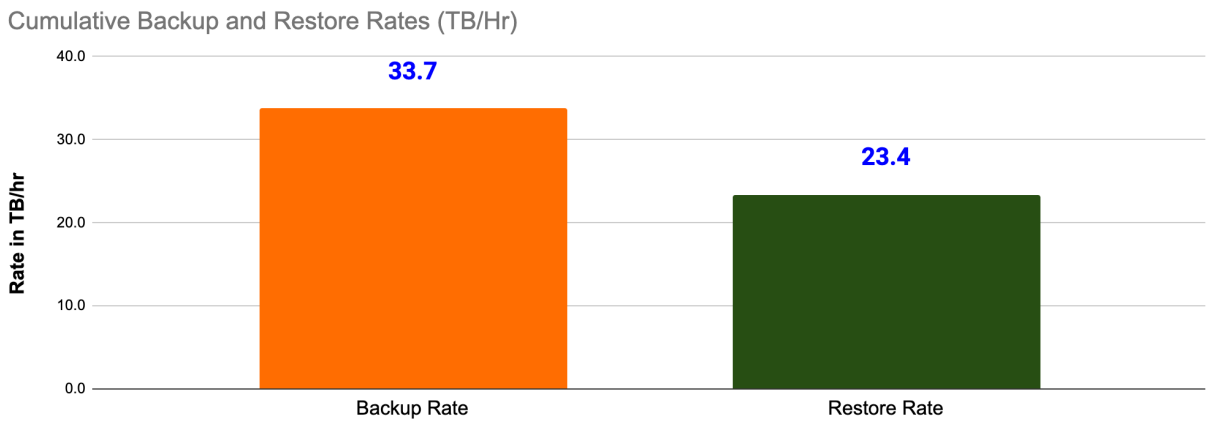


Figure 24.: Cumulative backup and restore rates from 108 VMs

Appendix C: Impact of nconnect

> CAUTION: This is an experimental feature and only certain versions of Linux kernels support it. Please consult with Pure Storage Support before enabling it.

A new NFS client mount option, *nconnect*, has been introduced to leverage massively scalable storage systems like FlashBlade; however, this feature is only available with the latest Linux kernels. Typically, a single session is established by an NFS client to the NFS server per mount point. This can limit performance to a maximum of 1GB/s for that session with FlashBlade. That is usually sufficient for most applications. By using nconnect, the NFS client establishes more concurrent sessions to the NFS server to drive more I/O. In some cases, where the application can take advantage of more bandwidth and parallelism, customers can achieve higher performance with the nconnect feature. Please note that not every application can take advantage of this feature.

The nconnect feature, if supported by your Linux kernel, can be enabled by setting the mount option mount to

```
-t nfs -o nconnect=16 NFSServer:/YourVolume /mnt/LocalMountPoint/
```

The chart below shows the impact of the nconnect mount option on backup performance of MariaDB. The recovery operation is more serialized, and so it could not take advantage of the nconnect option.

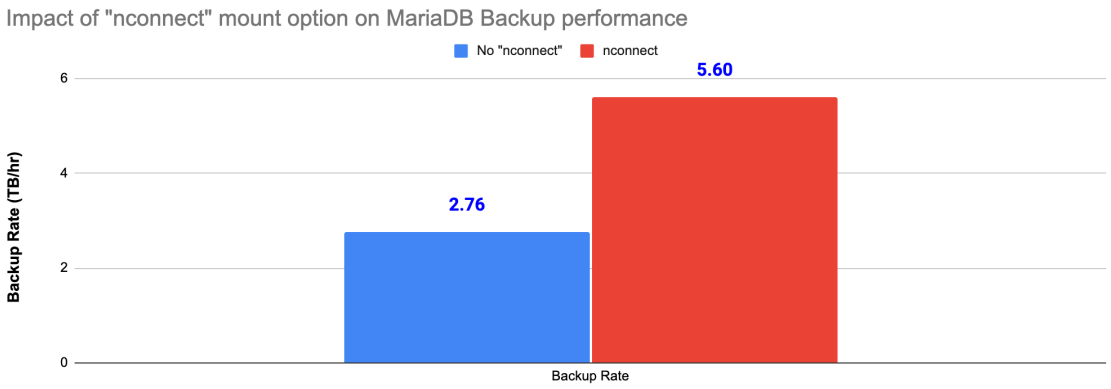


Figure 25. Impact of “nconnect” mount parameter on backup performance

About the Author

Radha Manga is a senior director at Pure Storage working on solution strategy. He is responsible for identifying new solution areas and expanding storage TAM across Pure Storage product portfolio. Radha has over 28 years of experience working in various roles at different companies ranging from working as a solutions architect to leading large solution architect teams.

©2020 Pure Storage, the Pure P Logo, and the marks on the Pure Trademark List at https://www.purestorage.com/legal/productenduserinfo.html are trademarks of Pure Storage, Inc. Other names are trademarks of their respective owners. Use of Pure Storage Products and Programs are covered by End User Agreements, IP, and other terms, available at: https://www.purestorage.com/legal/productenduserinfo.html and https://www.purestorage.com/patents

The Pure Storage products and programs described in this documentation are distributed under a license agreement restricting the use, copying, distribution, and decompilation/reverse engineering of the products. No part of this documentation may be reproduced in any form by any means without prior written authorization from Pure Storage, Inc. and its licensors, if any. Pure Storage may make improvements and/or changes in the Pure Storage products and/or the programs described in this documentation at any time without notice.

THIS DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID. PURE STORAGE SHALL NOT BE LIABLE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS DOCUMENTATION. THE INFORMATION CONTAINED IN THIS DOCUMENTATION IS SUBJECT TO CHANGE WITHOUT NOTICE.

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