

# Implementation Guide for IBM Elastic Storage System 3500

Phillip Gerrard

Monika Balichetty

Luis Bolinches

Puneet Chaudhary

Pidad D'Souza

Mika Heino

Wesley Jones

Stieg Klein

John Lewars

Laxmi Rajmane

Van Smith

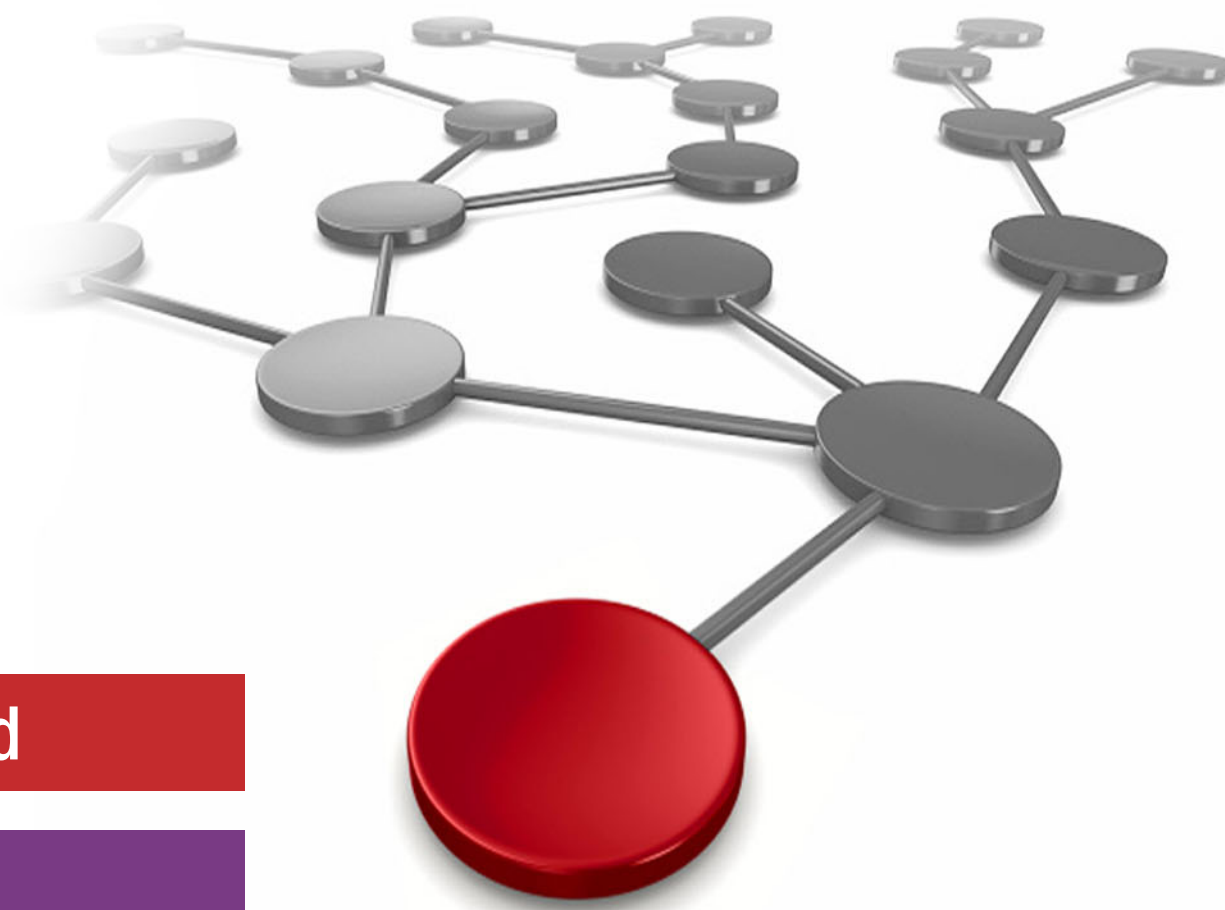
Ratan Swami

Jay Vaddi

Olaf Weiser

Farida Yaragatti

Ricardo Zamora



 **Cloud**

**Storage**





IBM Redbooks

**Implementation Guide for IBM Elastic Storage System  
3500**

June 2023

**Note:** Before using this information and the product it supports, read the information in “Notices” on page ix.

**First Edition (June 2023)**

This edition applies to IBM Elastic Storage System 3500.

**© Copyright International Business Machines Corporation 2023.**

Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.





# Contents

<b>Notices</b> .....	ix
Trademarks .....	x
<b>Preface</b> .....	xi
Authors .....	xi
Now you can become a published author, too! .....	xv
Comments welcome .....	xvi
Stay connected to IBM Redbooks .....	xvi
<b>Chapter 1. Introduction</b> .....	1
1.1 IBM Spectrum Scale RAID .....	2
1.1.1 Product history and IBM ESS options .....	2
1.1.2 Distinguishing features .....	2
1.2 IBM Elastic Storage System .....	5
1.3 IBM Elastic Storage System 3500 .....	5
1.3.1 Features of the IBM ESS 3500 .....	5
1.3.2 Additional features of the IBM ESS 3500 .....	7
1.4 License considerations .....	8
<b>Chapter 2. IBM Elastic Storage System 3500 architecture and overview</b> .....	9
2.1 Platform .....	10
2.1.1 Canisters and servers .....	10
2.1.2 Peripheral component interconnect express (PCIe) .....	12
2.2 GUI overview .....	12
2.2.1 GUI users .....	13
2.2.2 System setup wizard .....	14
2.2.3 Using the GUI .....	20
2.2.4 Monitoring of IBM ESS 3500 hardware .....	22
2.2.5 Storage .....	25
2.2.6 Replacing broken disks .....	25
2.2.7 Health events .....	27
2.2.8 Event notification .....	27
2.2.9 Dashboards .....	29
2.2.10 More information .....	30
2.3 Software enhancements .....	30
2.3.1 Containerized deployment .....	30
2.3.2 Red Hat Ansible .....	31
2.3.3 The mmvdisk command .....	32
2.3.4 The mmhealth command .....	32
2.4 RAS enhancements .....	36
2.4.1 RAS features .....	36
2.4.2 Enclosure overview .....	37
2.4.3 Machine type model and warranty .....	39
2.4.4 Components: FRU and CRU .....	39
2.4.5 Maintenance and service procedures .....	41
2.4.6 Software-related RAS enhancements .....	42
2.4.7 Call home functions .....	42
2.5 Performance .....	43
2.5.1 Networks .....	43

2.5.2 Non-volatile memory express drives . . . . .	45
2.5.3 Shared recovery group . . . . .	46
2.5.4 Tuning . . . . .	47
2.6 IBM Spectrum Scale Multi-Rail over TCP and RDMA over Converged Ethernet . . . . .	52
<b>Chapter 3. Planning considerations . . . . .</b>	<b>59</b>
3.1 Planning . . . . .	60
3.1.1 Technical and delivery assessment . . . . .	60
3.1.2 Hardware planning . . . . .	60
3.1.3 Software planning . . . . .	64
3.1.4 Network planning . . . . .	65
3.1.5 ESS Management Server considerations: . . . . .	67
3.1.6 Skills and services . . . . .	67
3.2 Standalone environment . . . . .	67
3.3 Mixed environment . . . . .	68
3.3.1 Adding the IBM ESS 3500 to an existing ESS cluster . . . . .	68
3.3.2 Scenario 1: Using IBM ESS 3500 for metadata network shared disks . . . . .	72
3.3.3 Scenario 2: Using IBM ESS 3500 to create a file system . . . . .	73
<b>Chapter 4. Providing your own IBM Elastic Storage System Management Server. . . . .</b>	<b>75</b>
4.1 Requirements . . . . .	76
4.1.1 Host requirements . . . . .	76
4.1.2 Networking . . . . .	77
4.1.3 Other information . . . . .	78
4.1.4 EMSVM hosts with more than one quad port card . . . . .	78
4.2 EMS VM deployment . . . . .	79
4.2.1 EMSVM deployment flow . . . . .	80
4.2.2 Reviewing the current documentation . . . . .	80
4.2.3 The emsvm tool . . . . .	81
4.2.4 Validating the host . . . . .	83
4.2.5 Downloading the EMSVM disk image from IBM FixCentral . . . . .	85
4.2.6 Starting the EMSVM . . . . .	85
4.2.7 Configuring IP addresses . . . . .	88
4.3 Other considerations . . . . .	91
4.3.1 Backing up and restoring EMSVM . . . . .	91
4.3.2 Stopping EMSVM . . . . .	92
4.3.3 Monitoring host HW using an EMSVM . . . . .	92
4.3.4 Accessing the host . . . . .	92
4.3.5 Updating host OS and firmware . . . . .	92
4.3.6 Changing the port type of NVIDIA ConnectX VPI adapter . . . . .	93
<b>Chapter 5. Use cases . . . . .</b>	<b>95</b>
5.1 Introducing performance storage use cases . . . . .	96
5.1.1 IBM ESS 3500 as part of a larger storage for data and AI infrastructure . . . . .	97
5.1.2 Typical IBM ESS 3500 performance storage use cases . . . . .	98
5.2 Metadata and high-speed data tier . . . . .	98
5.3 Data feed to GPUs for massive AI data acceleration . . . . .	99
5.4 Other use cases . . . . .	99
5.4.1 IBM Spectrum Scale with big data and analytics solutions . . . . .	99
5.4.2 Genomics medicine workloads in IBM Spectrum Scale . . . . .	100
<b>Appendix A. Configuring the 48 ports top of the rack management network switch</b>	<b>103</b>
Configuring the switches . . . . .	104



<b>Appendix B. Configuring two 8831-T48 switches as top of the rack switches . . . . .</b>	<b>109</b>
Logical overview. . . . .	110
<b>Related publications . . . . .</b>	<b>115</b>
IBM Redbooks . . . . .	115
Online resources . . . . .	115
Help from IBM . . . . .	116



# Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing, IBM Corporation, North Castle Drive, MD-NC119, Armonk, NY 10504-1785, US*

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.


## COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

# Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at “Copyright and trademark information” at <http://www.ibm.com/legal/copytrade.shtml>

The following terms are trademarks or registered trademarks of International Business Machines Corporation, and might also be trademarks or registered trademarks in other countries.

AIX®	IBM Research®	POWER8®
IBM®	IBM Security®	POWER9™
IBM Cloud®	IBM Spectrum®	Redbooks®
IBM Elastic Storage®	IBM Spectrum Fusion™	Redbooks (logo)  ®
IBM Garage™	POWER®	Spectrum Fusion™

Intel, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

The registered trademark Linux® is used pursuant to a sublicense from the Linux Foundation, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis.

Java, and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Ansible, Red Hat, are trademarks or registered trademarks of Red Hat, Inc. or its subsidiaries in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

# Preface

This IBM Redbooks publication introduces and describes the IBM Elastic Storage System 3500 (IBM ESS 3500) as a scalable, high-performance data and file management solution. The solution is built on proven IBM Spectrum Scale technology.

The IBM ESS 3500 is a hybrid platform able to provide both very high data throughput for high demand AI and ML workloads using an NVMe based drive configuration, or provide large amounts of storage when attached to additional disk based ESS HDD storage units.

This book provides a technical overview of the ESS 3500 solution and helps to plan the installation of the environment. We also explain the use cases where we believe it fits best.

Our goal is to position this book as the starting point document for customers that would use the IBM ESS 3500 as part of their IBM Spectrum Scale setups.

This book is targeted toward technical professionals (consultants, technical support staff, IT Architects, and IT Specialists) who are responsible for delivering cost-effective storage solutions with the IBM ESS 3500.

## Authors

This book was produced by a team of IBM product specialists from around the world.



**Phillip Gerrard** is a Project Leader for the International Technical Support Organization working out of Beaverton, Oregon. As part of IBM® for over 15 years he has authored and contributed to hundreds of technical documents published to IBM.com and worked directly with IBM's largest customers to resolve critical situations. As a team lead and Subject Matter Expert for the IBM Spectrum Protect support team, he is experienced in leading and growing international teams of talented IBMers, developing and implementing team processes, creating and delivering education. Phillip holds a degree in computer science and business administration from Oregon State University.



**Monika Balichetty** is an information developer who works with the IBM Spectrum Scale® and ESS solutions and has 12 years of experience developing documents for various audiences. She is a detail-oriented author with proven success crafting impactful content aligned with organizational needs and style guidelines. Monika graduated from S.V. University with a bachelor's degree in electrical and electronic engineering.



**Luis Bolinches** is part of the IBM Storage Scale development team. Has been working with Scale since version 3.4, and ESS prior to GA. With a background of Networking, Power systems and Linux, he is regularly involved with large customer engagements, development of ESS deployment solutions and involved in customer facing events.



**Puneet Chaudhary** is a Technical Solutions Architect who works with the IBM ESS and IBM Spectrum Scale solutions. He has worked with IBM GPFS, now IBM Spectrum Scale, for many years.



**Pidad Gasfar D'Souza** is a System Architect who specializes in performance engineering of IBM Spectrum Scale and IBM POWER® systems. He has been with IBM for more than 17 years. He led the system-performance engineering of the GPU-accelerated systems designed for applications in the fields of AI and HPC. He also led development teams in the areas of IBM AIX® base-libraries and JVM. He has presented extensively at several international conferences, and delivered customer workshops and lab sessions.



**Mika Heino** is a senior IT Management Consultant with IBM Lab Services working in Finland for local and international IBM accounts. He has a degree in Telecommunications and Computer Science from Turku University of Applied Sciences. Mika has 25 years experience with Linux, IBM AIX and IBM i, and server virtualization for both Intel and IBM POWER. He is Master Certified Technical Specialist for Storage Systems with more than 15 years of experience with storage area networks (SANs), IBM Storage Systems, and storage virtualization.



**Wesley Jones** serves as the test-team lead for IBM Spectrum Scale Native RAID. He also serves as one of the principle deployment architects for IBM ESS. His focus areas include IBM Power servers, IBM Spectrum Scale (GPFS), cluster software (xCAT), Red Hat Linux, Networking (especially InfiniBand and Gigabit Ethernet), storage solutions, automation, and Python.



**Stieg Klein** is a Senior Storage Technical Specialist in IBM Advanced Technology Group (ATG). He joined IBM in 2012 and is focused on IBM software-defined storage (SDS), including IBM Elastic Storage® System (IBM ESS) and IBM Storage Scale. He holds a Bachelor of Arts degree in Computer Science from the University of California, Berkeley.



**John Lewars** is a Senior Technical Staff Member who leads performance engineering work in the IBM Spectrum Scale development team. He has been with IBM for over 20 years, working first on some of IBM's largest high-performance computing systems, and later on the IBM Spectrum Scale (formerly GPFS) development team. John's work on the IBM Spectrum Scale team includes working with large customer deployments and improving network resiliency, along with co-leading development of the team's first public cloud and container-support deliverables.



**Laxmi Rajmane** is an information developer with 13 years of experience in software documentation creation and language review. Since 2019 she has been involved with IBM Spectrum Scale and IBM Elastic Storage System documentation. She holds a master's degree in English language and literature.



**Van Smith** works in the Storage Development organization focusing on reliability, availability, and serviceability (RAS) across various platforms. Previously, he was the Content Manager for Technical Training for IBM Systems Storage in the IBM Garage™ for Systems organization. He is a Certified Reliability Engineer. He has over 20 years with IBM, and has served in subject matter expert (SME), program management, and managerial roles.



**Ratan Swami** is a Java Development Lead who works with the IBM Spectrum Scale and ESS solutions as a GUI Architect. He has 14 years of experience involving analysis, design, development, integration, deployment of enterprise application software in Web-based, Multi-Tiered Architecture with object-oriented Technology and experience in service industry. He also carries strong exposure in developing Portal based solutions in client/server with n-tier architecture using J2EE Technologies and container-based technologies like Docker, Kubernetes etc. Ratan holds a master's degree in Computer Application specialization in Computer programming with various languages C, C++, Java and Python etc.



**Jay Vaddi** is a Storage Performance Engineer at IBM Tucson, AZ. He has been with IBM and the performance team for over five years. His focus is primarily on performance analysis and evaluations of IBM Spectrum Scale and IBM ESS products.



**Olaf Weiser** joined IBM as an experienced professional more than ten years ago and worked in the DACH TSS team delivering Power-based solutions to enterprise and HPC customers. He developed deep skills in IBM Spectrum Scale (previously IBM GPFS) and has a worldwide reputation as the performance-optimization specialist for IBM GPFS. At the IBM European Storage Competence Center (ESCC), Olaf works on Advanced Technical Support (ATS) and Lab Services and Skill Enablement tasks that are required to grow the IBM Spectrum Scale business in EMEA. For the past two years, he worked as a performance engineer for RDMA and RoCE in IBM Research® and Development GmbH.





**Farida Yaragatti** is a Senior Software Engineer at IBM India. She has a BE, Electronics and Communication from Karnataka University, India and has 12 years of experience in the Software Testing field. She is part of manual and automation testing for IBM Spectrum Scale and IBM Elastic Storage System (ESS) deployment as a Senior Tester. Farida worked at IBM for over five years and previously held roles within the IBM Platform Computing and IBM Smart analytics system (ISAS) testing teams. She has strong engineering professional skills in Software deployment testing, including automation using various scripting technologies, such as Python, shell scripting, Robot framework, Ansible, and Linux.



**Ricardo D. Zamora Ruvalcaba** is a Software Engineer at IBM Guadalajara, Mexico and holds a bachelor's degree in Mechatronics. He started his career in IBM Spectrum Scale as a Test Automation Engineer and eventually moved to the ESS deployment development team where he constantly implements new tools and technologies. As the focal point for Manufacturing and a Technical Advocate for IBM ESS customers around the globe, he has strong automation engineering skills in various scripting technologies, such as Python, Ansible, shell scripting, Jenkins, the Robot framework, and Linux.

Additional thanks to the following people for their contributions to this project:

Chiahong Chen and Chris Maestas  
**IBM Storage Systems**

## Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an IBM Redbooks® residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:  
[ibm.com/redbooks/residencies.html](https://ibm.com/redbooks/residencies.html)

## Comments welcome

Your comments are important to us!

We want our books to be as helpful as possible. Send us your comments about this book or other IBM Redbooks publications in one of the following ways:

- ▶ Use the online **Contact us** review Redbooks form found at:

[ibm.com/redbooks](http://ibm.com/redbooks)

- ▶ Send your comments in an email to:

[redbooks@us.ibm.com](mailto:redbooks@us.ibm.com)

- ▶ Mail your comments to:

IBM Corporation, IBM Redbooks  
Dept. HYTD Mail Station P099  
2455 South Road  
Poughkeepsie, NY 12601-5400

## Stay connected to IBM Redbooks

- ▶ Look for us on LinkedIn:

<http://www.linkedin.com/groups?home=&gid=2130806>

- ▶ Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

<https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm>

- ▶ Stay current on recent Redbooks publications with RSS Feeds:

<http://www.redbooks.ibm.com/rss.html>



# Introduction

This chapter introduces the IBM Elastic Storage System 3500 (IBM ESS 3500) solution, describes the software characteristics of IBM Spectrum Scale RAID software that runs on the IBM ESS 3500 and provides an overview of the IBM ESS 3500. This chapter includes the following topics:

- ▶ 1.1, “IBM Spectrum Scale RAID” on page 2
- ▶ 1.2, “IBM Elastic Storage System” on page 5
- ▶ 1.3, “IBM Elastic Storage System 3500” on page 5
- ▶ 1.4, “License considerations” on page 8

IBM ESS 3500 is a high-performance, NVMe flash-storage based member *of* the IBM Spectrum Scale and IBM Elastic Storage System family. This storage solution was designed for high performance, high-scalability Data, and AI applications. For an overview of how IBM ESS 3500 fits into this overall family, see [IBM Elastic Storage System Introduction Guide, REDP-5323](#).

# 1.1 IBM Spectrum Scale RAID

This section provides a high-level technical overview of the IBM Spectrum Scale RAID that is used in all IBM ESS models including IBM ESS 3500. IBM Spectrum Scale RAID on IBM ESS 3500 provides significant storage cost reduction while simultaneously providing enterprise-class reliability, performance, and serviceability.

The IBM Spectrum Scale RAID software in IBM ESS 3500 uses local NVMe drives. Because the software manages the RAID functions, IBM ESS 3500 does not require an external RAID controller or acceleration hardware.

IBM Spectrum Scale RAID in IBM ESS 3500 supports two and three fault-tolerant RAID codes. The two-fault tolerant codes include 8-data plus 2-parity, 4-data plus 2-parity, and 3-way replication. The three-fault tolerant codes include 8-data plus 3-parity, 4-data plus 3-parity, and 4-way replication. Figure 1-1 shows example RAID tracks consisting of data and parity strips.

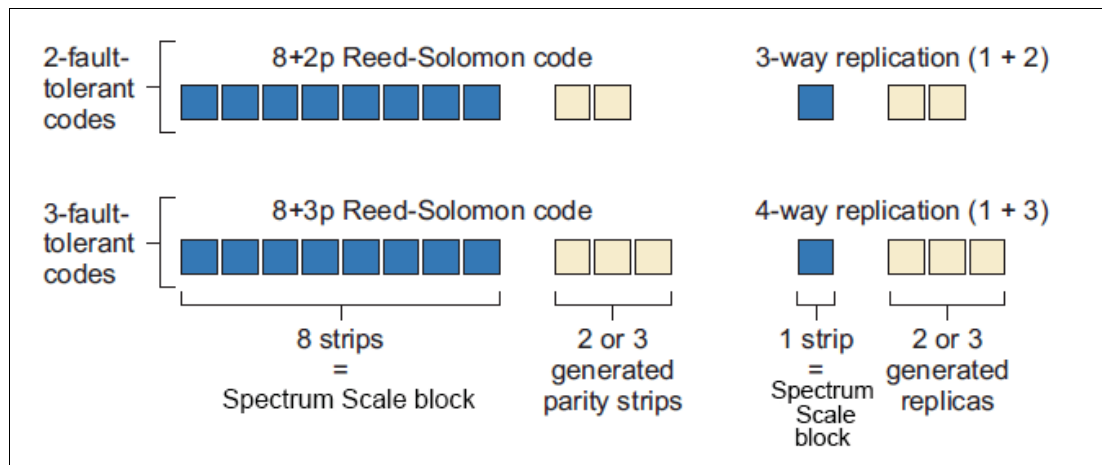


Figure 1-1 RAID tracks

## 1.1.1 Product history and IBM ESS options

For more information about the history of IBM Elastic Storage Systems, where the IBM ESS 3500 fits into the range of storage offerings, and the related components of an IBM Spectrum Scale Environment see [IBM Elastic Storage System Introduction Guide, REDP-5253](#).

## 1.1.2 Distinguishing features

IBM Spectrum Scale RAID distributes data and parity information across node failure domains to tolerate unavailability or failure of all physical disks in a node. It also distributes spare capacity across nodes to maximize parallelism in rebuild operations.

IBM Spectrum Scale RAID uses end-to-end checksums and not layered or serialized checksums that terminate in the path from host to host. IBM Spectrum Scale RAID implements data versions to detect and correct the data integrity problems of traditional RAID. Data is verified by using end-to-end checksums for both read and write operations on the device.

Figure 1-2 shows a simple example of declustered RAID. The left side shows a traditional RAID layout that consists of three 2-way mirrored RAID volumes and a dedicated spare disk that uses seven drives. The right side shows the equivalent declustered layout, which still uses seven drives. Here, the blocks of the three RAID volumes and the spare capacity are scattered over the seven disks.

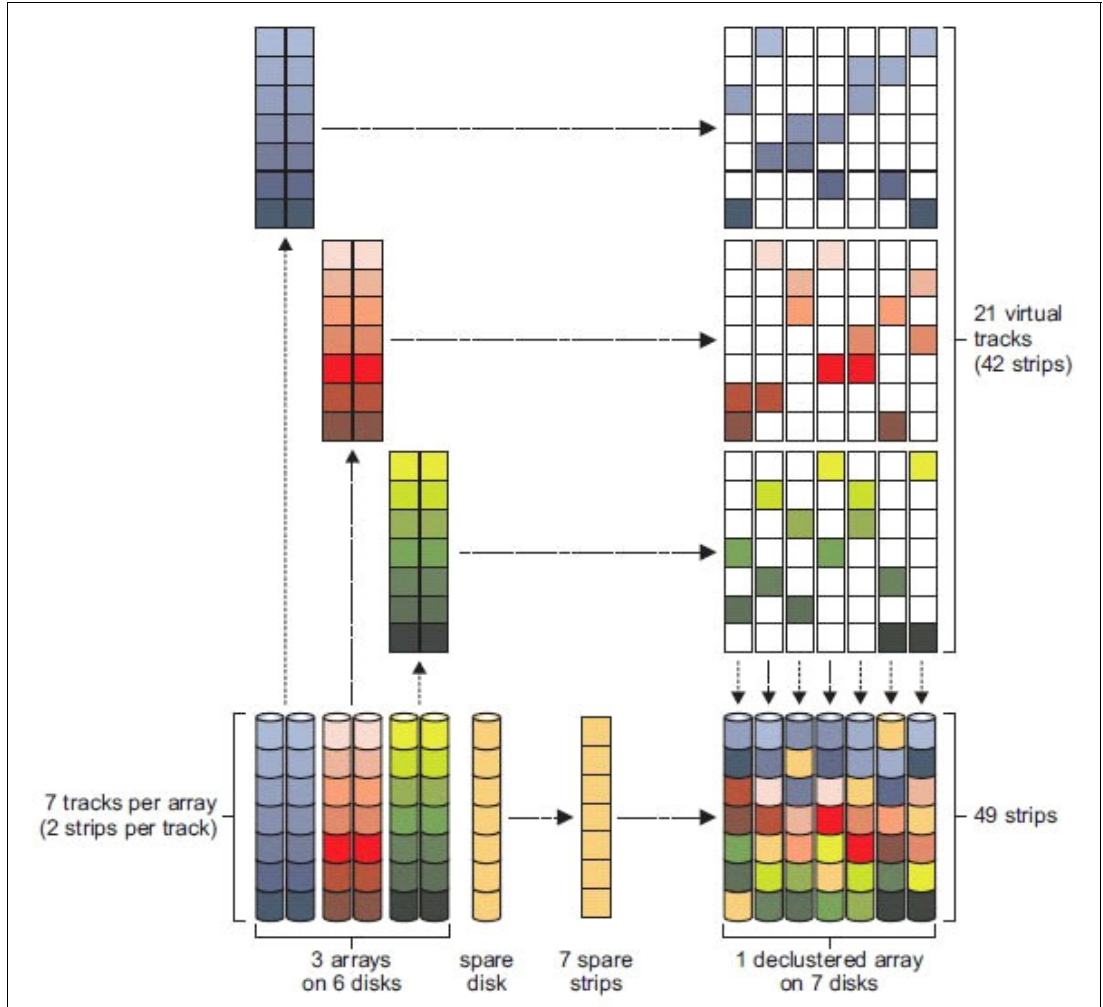


Figure 1-2 Declustered array versus 1+1 array

The declustered RAID layout provides the following advantages over the traditional RAID layout:

- ▶ Figure 1-3 shows a significant advantage of the declustered RAID layout over the traditional RAID layout after a drive failure. With the traditional RAID layout on the left side of Figure 1-3, the system must copy the surviving replica of the failed drive to the spare drive, reading only from one drive and writing only to one drive.

However, with the declustered layout that is shown on the right side of Figure 1-3, the affected replicas and the spares are distributed across all six surviving disks. This configuration rebuilds reads from all surviving disks and writes to all surviving disks, which greatly increases rebuild parallelism.

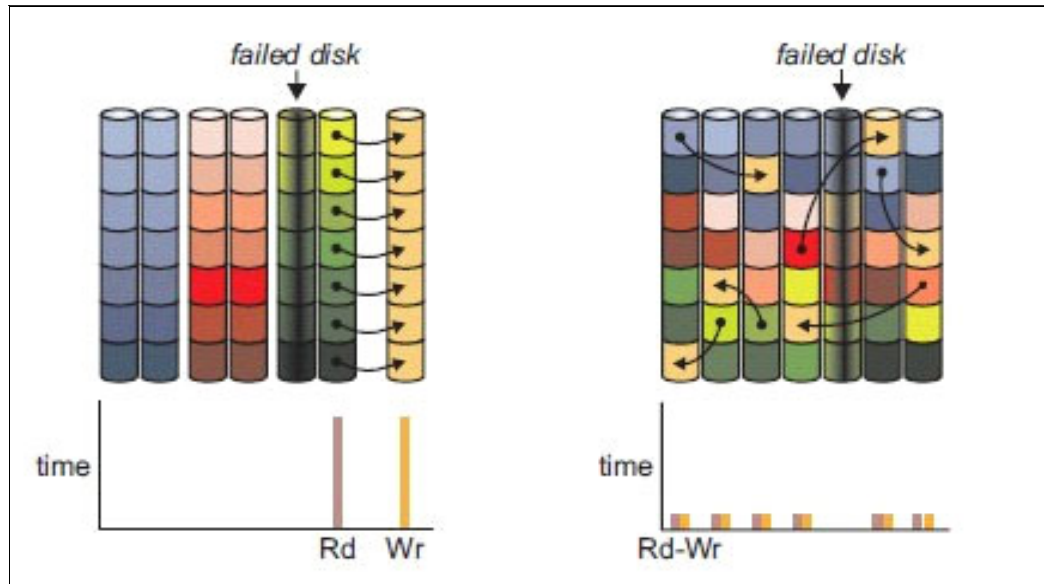


Figure 1-3 Array rebuild operation

- ▶ Another advantage of the declustered RAID technology that is used by IBM ESS 3500 is that it minimizes the worst-case number of critical RAID tracks in the presence of multiple disk failures. IBM ESS 3500 handles restoring protection to critical RAID tracks as a high priority and assigns a lower priority to RAID tracks that are not critical.

For example, consider an 8+3p RAID code on an array of 100 PDisks. In the traditional layout and declustered layout, the probability that a specific RAID track is critical is  $11/100 \times 10/99 \times 9/98$  (0.1%). However, when a track is critical in the traditional RAID array, all tracks in the volume are critical, whereas with declustered RAID, only 0.1% of the tracks are critical. By prioritizing the rebuild of more critical tracks over less critical tracks, IBM ESS 3500 quickly gets out of critical rebuild and then can tolerate another failure.

IBM ESS 3500 adapts these priorities dynamically. If a *noncritical* RAID track is used and more drives fail, this RAID track's rebuild priority can be escalated to *critical*.

- ▶ A third advantage of declustered RAID is that it makes it possible to support any number of drives in the array and to dynamically add and remove drives from the array. Adding a drive in a traditional RAID layout (except when adding a spare) requires significant data reorganization and restriping. However, only targeted data movement is needed to rebalance the array to include the added drive in a declustered array.

## 1.2 IBM Elastic Storage System

IBM Elastic Storage System (IBM ESS) is based on IBM Spectrum Scale running with validated IBM hardware. IBM Spectrum Scale Native RAID provides physical-disk protection; is tightly integrated with IBM Spectrum Scale; and provides file system access over the network to all of the IBM Spectrum Scale clients. Other protocols can be used to access the IBM Spectrum Scale file system. IBM Spectrum Scale RAID on IBM Spectrum Scale is optimized to use the underlying hardware and software stack to improve and simplify the ability to monitor, access, and maintain the storage components.

Details about the ways to access an IBM Spectrum Scale file system falls outside of the scope of this publication. Documentation for accessing an IBM Spectrum Scale file can be found in [IBM Spectrum Scale documentation](#) and [IBM Elastic Storage System Introduction Guide, REDP-5253](#).

## 1.3 IBM Elastic Storage System 3500

IBM ESS 3500 is designed to address the challenge of managing data for analytics with high-performance storage for AI workloads. The IBM ESS 3500 is a hybrid offering and is available in multiple configurations that can be customized to suit many customer needs, examples of which are provided in the following list:

- ▶ Reduce the time-to-value cycle for artificial intelligence (AI)
- ▶ Deep learning
- ▶ High-performance computing workloads by using all-NVMe storage
- ▶ Expandability to handle larger storage needs by adding up to 8@4U ESS expansion units

The hardware and software design of IBM ESS 3500 provides performance that helps decrease processor wait times. The IBM ESS 3500 is compatible with all IBM Elastic Storage System models.

The IBM ESS 3500 can be ordered either half populated (12) or fully populated (24) NVMe drives.

For more information about the IBM ESS 3500, see [IBM Elastic Storage System 3500](#).

### 1.3.1 Features of the IBM ESS 3500

The IBM ESS 3500 is part of the third generation of IBM ESS. IBM ESS 3500 was announced in May 2022 and includes the following features:

- ▶ Options for the IBM ESS 3500 deployment are provided in the following list:
  - Performance - All NVMe based storage drives for demanding workload
  - Capacity - HDD-based storage, with up to eight expansion storage enclosures
  - Hybrid - NVMe storage and HDD storage in a single configuration
- ▶ Design that is based on the IBM ESS 3200 but with faster x86 processors and various design improvements that are focused on serviceability
- ▶ Increased expansion, which was announced in 4Q2022 [to support up to eight additional storage enclosures](#) that enables support for up to 16 PB of raw HDD capacity

- ▶ The IBM ESS 3500 supports enterprise class NVMe drives:
  - You can order an IBM ESS 3500 populated with 12 NVMe drives or 24 NVMe drives in capacities of 3.84, 7.68, 15.36 or 30.72 TB, which allows up to 736 TB raw capacity in a 2U form factor.
  - When ordered with 12 drives, an additional 12 NVMe drives can be added nondisruptively to the IBM ESS 3500.

**Important:** The additional drives must be of the same size as the first 12 drives.

- ▶ Design of the IBM ESS 3500 provides the following benefits:
  - High performance, NVMe flash storage with up to 91GBps read throughput per 2U building block.
  - Edge capability and global data access solution, which can be deployed in either data centers or at the edge, receiving and processing data that then uses IBM Spectrum Scale Active File Management to share data globally.
- ▶ Protocol node deployment that can be moved to the IBM ESS 3500 and run as a virtual machine on each canister, reducing initial deployment costs

The third-generation IBM ESS 3500 is designed to meet the challenges of managing the high-demand workloads of today and tomorrow. The IBM ESS 3500 provides high-performance, software-defined flash storage and couples it with the storage technology of IBM Spectrum Scale NVME to offer industry-leading file management capabilities. The IBM ESS 3500 builds on and extends a track record of meeting the many needs of an organization that requires large amounts of high-performance storage. The IBM ESS 3500 is up to 12% faster than previous generations of IBM ESS NVMe storage<sup>1</sup>.

---

<sup>1</sup> <https://www.ibm.com/downloads/cas/R0Q1DV1X>



Figure 1-4 Shows the core IBM ESS 3500 2U unit and the H4 model, which adds additional attached storage units.

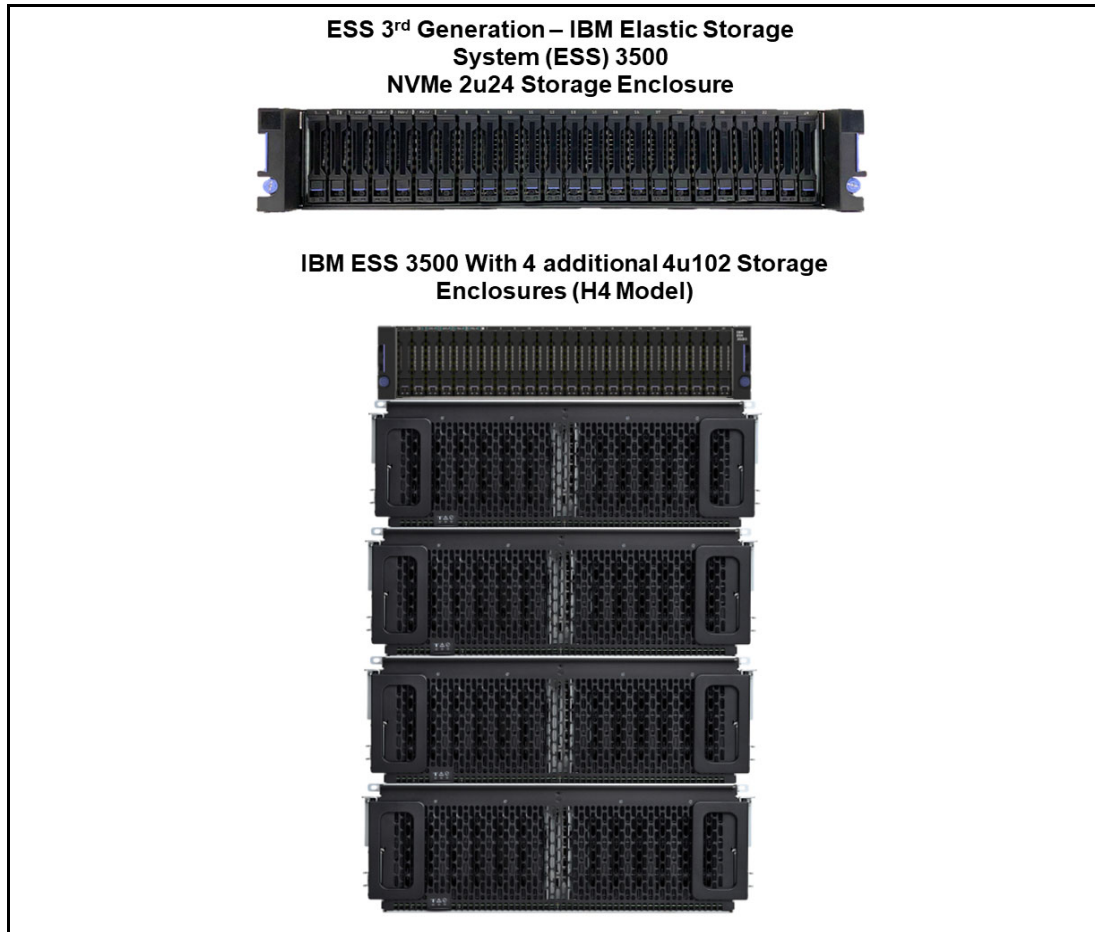


Figure 1-4 IBM ESS 3500 and IBM ESS 3500 H4 configurations

### 1.3.2 Additional features of the IBM ESS 3500

The IBM ESS 3500 provides a high-performance tier of IBM Spectrum Scale file storage for a broad variety of AI, analytics, and Big Data applications. The IBM ESS 3500 is designed to ensure that GPUs in AI workloads are running at peak performance. Like all models of IBM ESS, the IBM ESS 3500 runs the IBM Spectrum Scale RAID erasure coding, which provides superior consistent high performance and mitigation of storage hardware failures. It also provides intelligent monitoring, management, and dynamic tuning of all IBM Elastic Storage Systems for IBM Spectrum Scale data through the IBM ESS management server node.

#### Operational efficiency

The demands on IT staff time and expertise can be reduced by installing the containerized software and by using a powerful management GUI. The ability to add and manage more data with dense storage units allows for a smaller data center footprint.

#### Reliability

The software-defined erasure coding assures data recovery and uses less space than data replication. Data restoration can take minutes, rather than hours or days, and can be run without disrupting operations.

## Deployment flexibility

The IBM ESS 3500 is available in a wide range of capacities, from tens to hundreds of terabytes per 2U enclosure. The IBM ESS 3500 is deployable as either a stand-alone system or as an edge storage system. It is scalable to suit various high-performance workloads with other IBM ESS 3500 systems or handle larger storage capacity needs with the addition of up to 8 storage enclosures per storage rack. The added ability to run embedded virtual protocol nodes allows a single hybrid platform to be configured to handle the vast majority of workloads.

## 1.4 License considerations

The IBM ESS 3500 follows the same license model as the other IBM ESS products. The two currently available options for IBM ESS are IBM Spectrum Scale for ESS *Data Access Edition* and IBM Spectrum Scale for *ESS Data Management Edition*.

ESS uses capacity-based licensing, which means that a customer can connect as many clients as needed without extra license costs. For other types of configurations, contact IBM or your IBM Business Partner for license details.

For more information about licensing on IBM Spectrum Scale and IBM ESS, see [The FAQ page for IBM Spectrum Scale and IBM ESS Licensing](#) and [IBM Spectrum Scale IBM Elastic Storage Licensing Information](#).



# IBM Elastic Storage System 3500 architecture and overview

This chapter describes the architecture and provides an overview of IBM Elastic Storage System 3500 (IBM ESS 3500). It covers the following topics:

- ▶ 2.1, “Platform” on page 10
- ▶ 2.2, “GUI overview” on page 12
- ▶ 2.3, “Software enhancements” on page 30
- ▶ 2.4, “RAS enhancements” on page 36
- ▶ 2.5, “Performance” on page 43
- ▶ 2.6, “IBM Spectrum Scale Multi-Rail over TCP and RDMA over Converged Ethernet” on page 52

## 2.1 Platform

This section provides an overview of the IBM ESS 3500 platform. An IBM ESS 3500 enclosure contains a set of Non-Volatile Memory Express (NVMe)-attached SSD drives and a pair of server canisters. The IBM ESS 3500 is an all-Flash array platform and uses NVMe-attached SSD drives to provide significant performance improvements when compared to SAS-attached flash drives.

### 2.1.1 Canisters and servers

This section describes the CPU, memory, and networking components of the IBM ESS 3500 (Model 5141-FN2) system.

#### CPU

The IBM ESS 3500 system uses a single socket AMD EPYC Rome 7642 48-core processor per I/O canister node for a total of two CPUs per enclosure. Figure 2-1 shows a CPU in a canister:

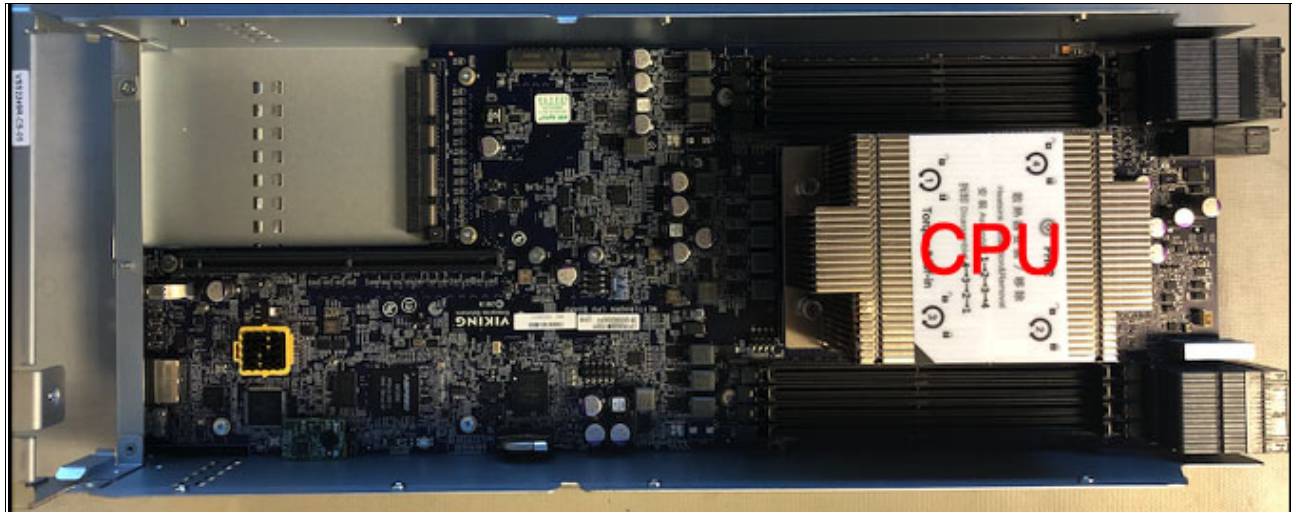


Figure 2-1 Internal view of single CPU in a canister

#### Memory

The memory of each canister and enclosure is depicted in Figure 2-1.

Table 2-1 Memory configuration

Number of DIMMs per server canister	Total memory per server canister	Number of DIMMs per server enclosure	Total memory per server enclosure
8 (64 GB DIMM only)	512 GB	16	1024 TB
8 (128 GB DIMM)	1024 GB	16	2048 TB

## Networking

The IBM ESS 3500 includes four Gen4 x16 Peripheral component interconnect express (PCIe) slots per canister. Adapter options are listed in Table 2-2.

Table 2-2 IBM ESS 3500 adapter options

Feature Code	PCIe Form Factor	InfiniBand	Ethernet
AJZL	CX-6 VPI and InfiniBand	HDR200 200 Gb, HDR100 100 Gb, EDR 100 Gb	100 GbE, 200 GbE
AJZN	CX-6 DX		100 GbE
AJP1	CX-5 VPI and InfiniBand	EDR 100 Gb	100 GbE

You can configure IBM ESS 3500 in a mixed cluster that contains IBM ESS 5000, IBM ESS 3500, IBM ESS 3200, and IBM ESS 3000 systems

Figure 2-2 shows the rear panel of each IBM ESS 3500 canister that includes the PCIe slots and other port locations.

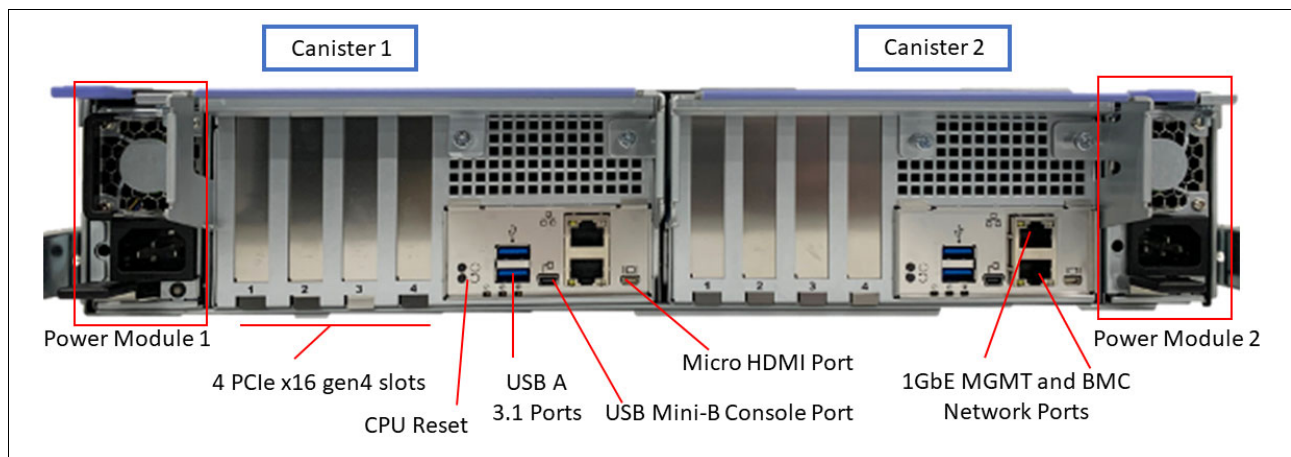


Figure 2-2 4 Gen4 PCIe slots per server canister are included with the IBM ESS 3500.

## 2.1.2 Peripheral component interconnect express (PCIe)

The following list describes the PCIe lanes:

- ▶ 128 x Gen4 PCIe from each CPU (switchless)
  - NVMe drives - 86 [48 used (24 x 2)]
  - HBA PCIe adapters - 64 (4 x 16)
- ▶ Nontransparent bridge-to-peer canister - x8 Gen4
- ▶ Boot Drive - 8 (2x4 G3)

Figure 2-3 shows the PCIe lane details.

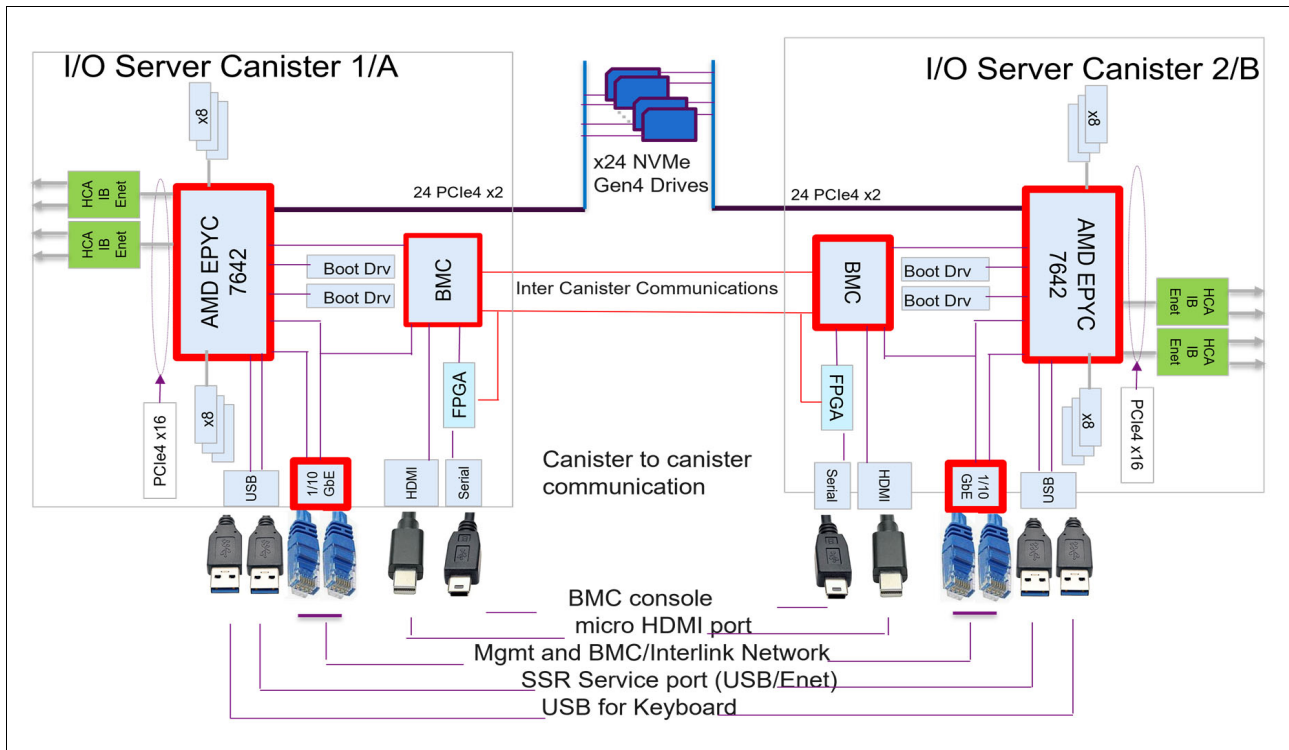


Figure 2-3 PCIe lanes in IBM ESS 3500

## 2.2 GUI overview

A graphical user interface (GUI) service runs on the IBM ESS 3500 management server (EMS). It can be used to monitor the health of the IBM ESS 3500 and to perform management tasks. This section provides an overview of the GUI, but is not comprehensive.

The `systemctl` command can be run on the EMS to start or stop the GUI. Table 2-3 shows the `systemctl` command options.

Table 2-3 The `systemctl` command options

Command	Description
Start the GUI service	<code>systemctl start gpfsGUI</code>
Check the status of the GUI service	<code>systemctl status gpfsGUI</code>
Stop the GUI service	<code>systemctl stop gpfsGUI</code>

To access the GUI, enter the IP address or hostname of the EMS in a web browser by using the secure https mode:

`https://<IP or hostname of EMS>`

## 2.2.1 GUI users

GUI users must be created before the GUI can be used. To grant special rights, roles are assigned to the users.

When the GUI is used for the first time, an initial user must be created:

```
/usr/lpp/mmfs/gui/cli/mkuser <username> -g SecurityAdmin
```

After the initial user is created, you can log in to the GUI with the newly created user. To create more users, select **Services** → **GUI** → **Users** and enter the user information. By default, users are stored in an internal user repository. Alternatively, an external user repository can also be used. An external user repository can be configured by selecting **Services** → **GUI** → **Additional Authentication** and entering the user information.

Administrators can also configure multi-factor authentication, which requires GUI users to be registered in an IBM Security® Verify repository. After multi-factor authentication is enabled, additional login information will be sent to the registered email or phone of the user. The user must provide the additional login information for two factor authentication (2FA) before they are able to successfully log in to the GUI. For more information, see [Configuring GUI details in IBM Security Verify for multi-factor authentication](#).

## 2.2.2 System setup wizard

After you log in to the GUI for the first time, the system-setup wizard is started. The following steps are a high-level overview of what to expect when using the system-setup wizard:

1. After the welcome page, the Verify Storage page queries important system information and performs several checks to verify whether the system is ready for use. See Figure 2-4.

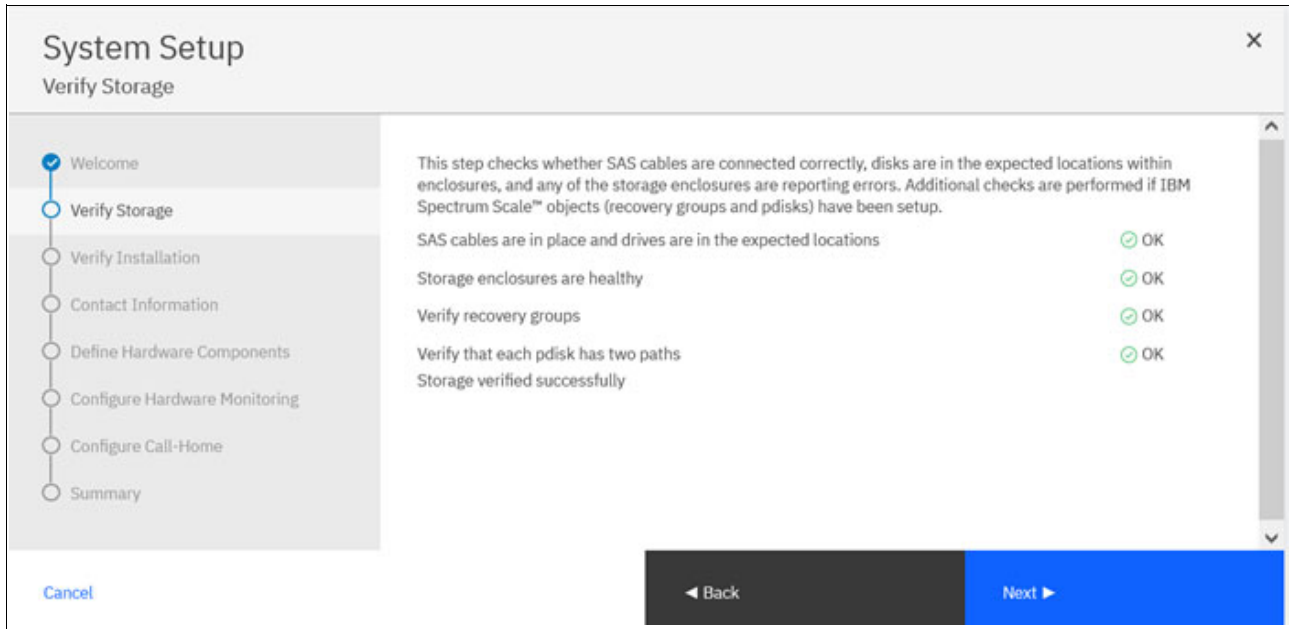


Figure 2-4 The System Setup wizard



2. After passing the storage verification check, the next Verify Installation page will check for the following other prerequisite conditions as shown in Figure 2-5:
  - Does the `mmhealth` utility report that the cluster file system is working correctly on the EMS and building block systems?
  - Is there at least one recovery group configured?
  - Is the system able to successfully look up the servers and enclosures, and add them to the component database?

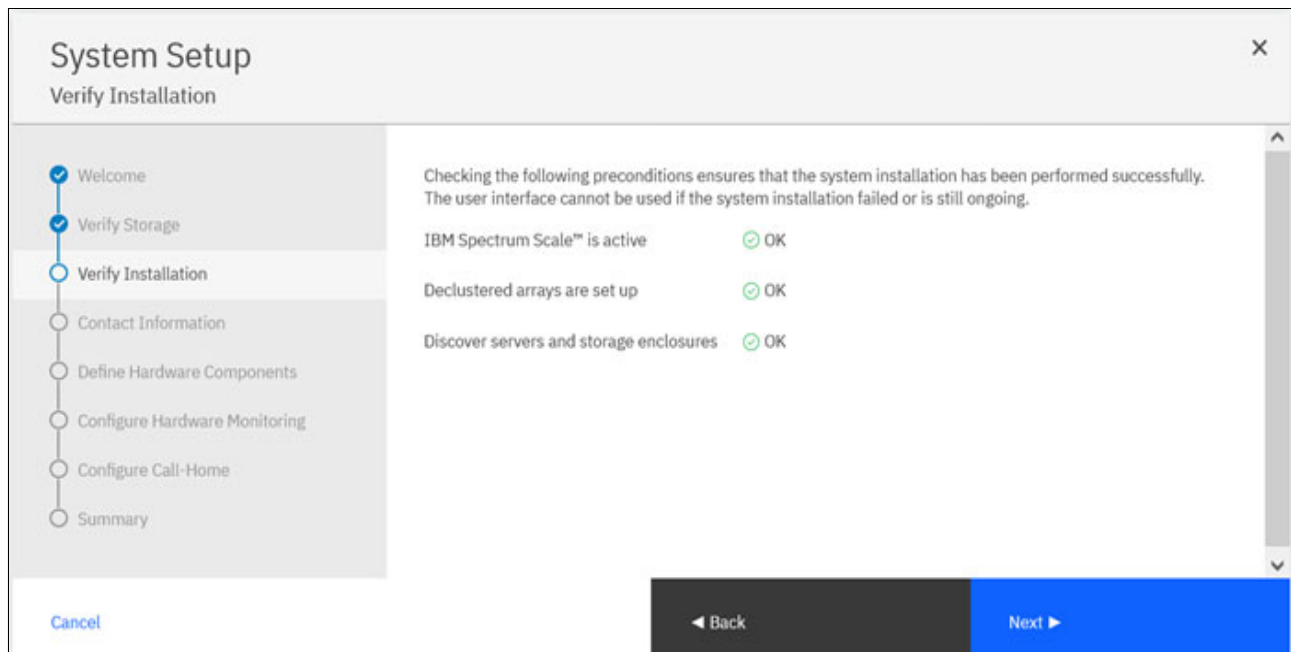


Figure 2-5 GUI Verify Installation page

3. After the installation is verified, in the Contact Information page enter the information to enable the optional *hardware callhome support* feature. Provide the company information, system information, primary contact, and optional secondary contact information for the environment. See Figure 2-6, Figure 2-7, and Figure 2-8 on page 17.

The screenshot shows the 'System Setup' window with the 'Contact Information' tab selected. A progress bar on the left indicates that 'Welcome', 'Verify Storage', and 'Verify Installation' are completed, while 'Contact Information' is the current step. The main content area contains a heading 'Contact Information' followed by a paragraph: 'Provide your company information. In the event of any issue, the details collected here will be used to provide hardware callhome support. Inaccurate or incomplete information entered might delay the support procedures.' Below this are two sections: 'Company Information' and 'System Information'. The 'Company Information' section has fields for 'Company name:' (containing 'IBM'), 'E-mail:' (containing 'john.doe@ibm.com'), and 'Country:' (a dropdown menu showing 'United States'). The 'System Information' section has a 'Location/Site Country:' dropdown menu showing 'United States'. At the bottom, there are 'Cancel', 'Back', and 'Next' buttons.

Figure 2-6 Contact information pt.1

The screenshot shows the 'System Setup' window with the 'Contact Information' tab selected. The progress bar on the left shows 'Contact Information' as the current step. The main content area contains a heading 'System Information' followed by several fields: 'Location/Site Country:' (dropdown showing 'United States'), 'Country:' (dropdown showing 'United States'), 'Address:' (text field containing '1 Orchard Rd, Armonk'), 'City:' (text field containing 'Armonk'), 'State/Province:' (text field containing 'NY'), and 'ZIP:' (text field containing '10504'). Below these is the 'Primary contact:' section with 'Contact name:' and 'Phone number:' fields. At the bottom, there are 'Cancel', 'Back', and 'Next' buttons.

Figure 2-7 Contact information pt.2

**System Setup**  
Contact Information

Welcome  
 Verify Storage  
 Verify Installation  
 **Contact Information**  
 Define Hardware Components  
 Configure Call-Home  
 Summary

**Primary contact:**

Contact name:  Phone number:

Secondary contact(optional)

Contact name:  E-mail:

Phone number:

[Cancel](#)
[◀ Back](#)
[Next ▶](#)

Figure 2-8 Contact information pt.3

- After all checks pass, in the Racks page the user can define where the IBM ESS 3500 systems are installed as shown in Figure 2-9. The user can either choose a predefined rack type or choose **Add new specification** if none of the rack types match the rack the IBM ESS 3500 was installed in. It is important that the selected rack-type has the same number of height units. A meaningful name can then be associated with each of the racks that are defined to help differentiate them.

**System Setup**  
Racks

Verify Storage  
 Verify Installation  
 Contact Information  
 **Racks**  
 Define Hardware Components  
 Building Blocks  
 Rack Locations  
 Configure Call-Home

Part Number	Name	Height	Description
1410HEA	Rack 1	42	IBM Intelligent Cluster 42U 1200mm Deep Expansion Rack
1410HPA	Rack 2	42	IBM Intelligent Cluster 42U 1200mm Deep Primary Rack

[Add new specification...](#)

[Cancel](#)
[◀ Back](#)
[Next ▶](#)

Figure 2-9 Specifying the racks

- The Building Blocks page displays one row for each IBM ESS 3500 or other IBM ESS models in the environment. The user can assign names to each building block or go with the default. See Figure 2-10.

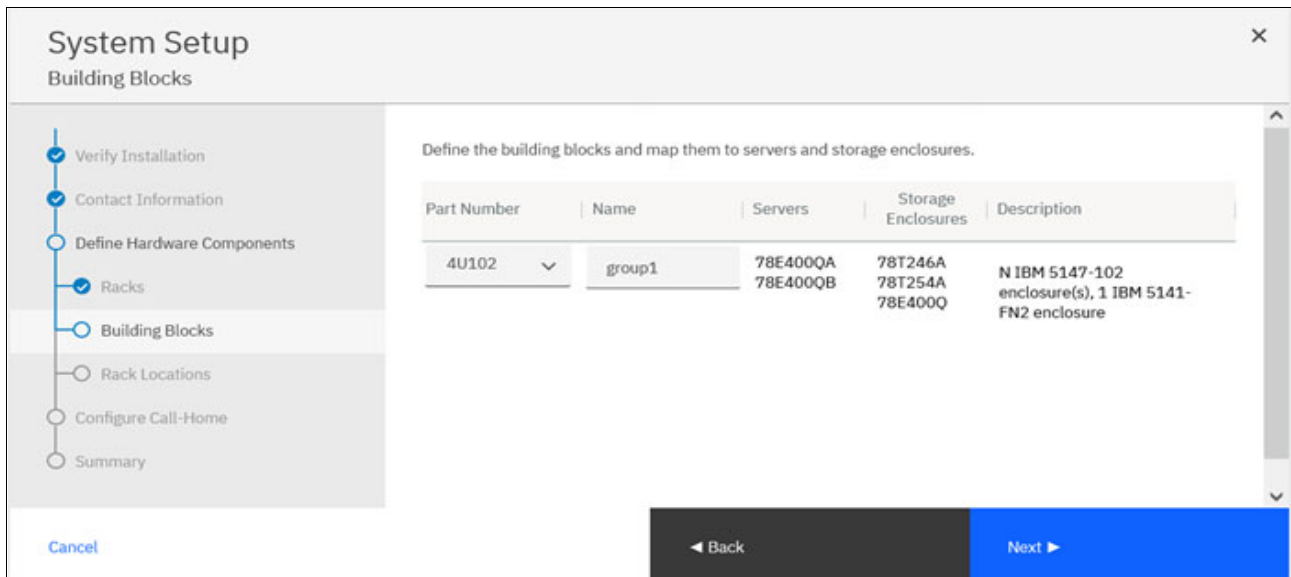


Figure 2-10 Defining building blocks

- The IBM ESS 3500 systems are assigned to the rack locations in which they are mounted. See Figure 2-11.

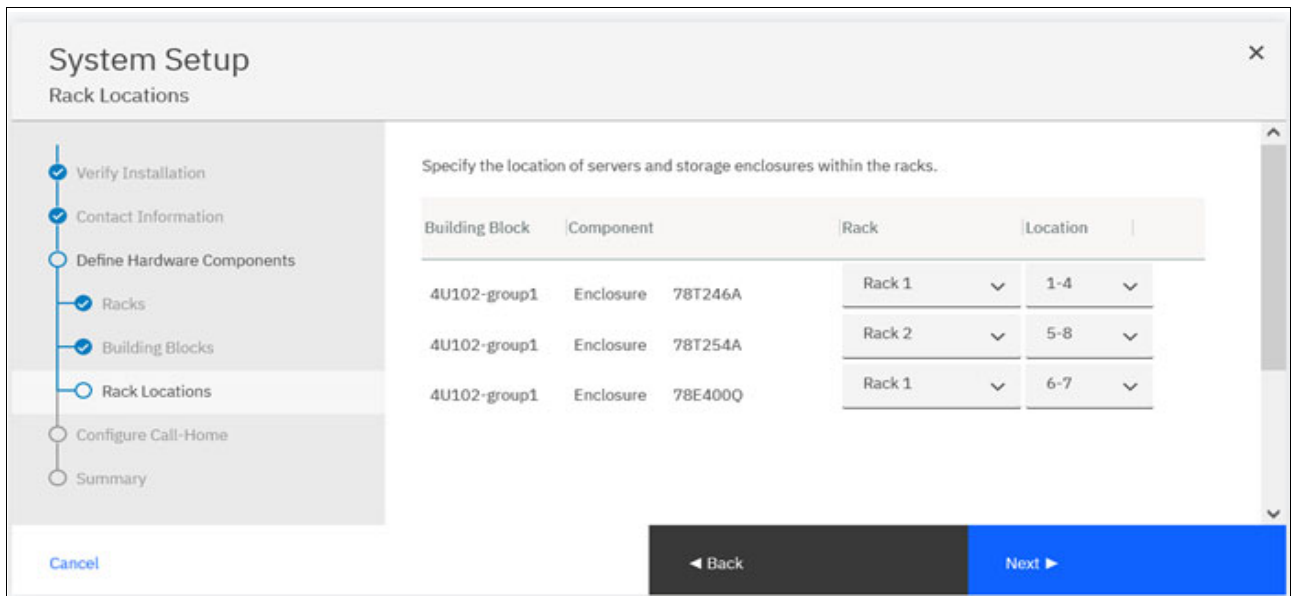


Figure 2-11 Assigning rack locations

7. After all component configuration steps are complete, provide the details to configure Call-Home:
  - a. The Select Nodes page asks for the details that are needed to automatically create a support case with IBM Support when issues are reported as shown in Figure 2-12.

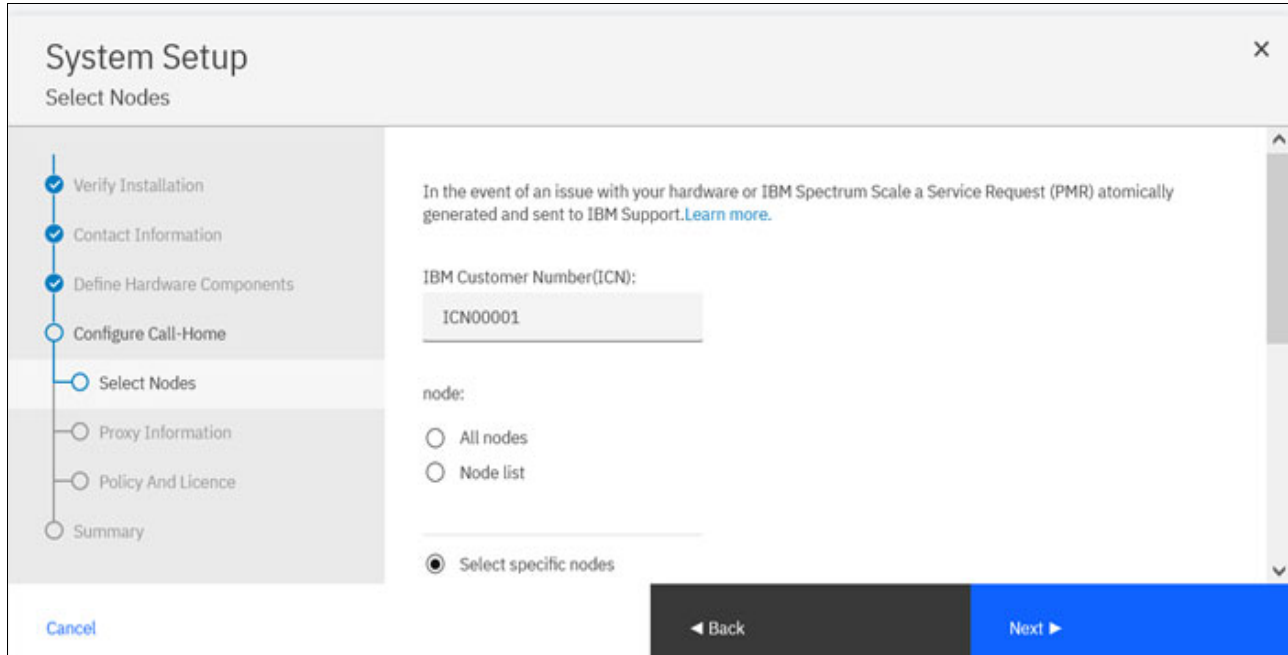


Figure 2-12 Call-Home configuration: Select Nodes

- b. The Proxy Information page is used to configure the network proxy settings to allow communication for Call-Home to contact IBM Support as shown in Figure 2-13. Configuring the proxy information is optional.

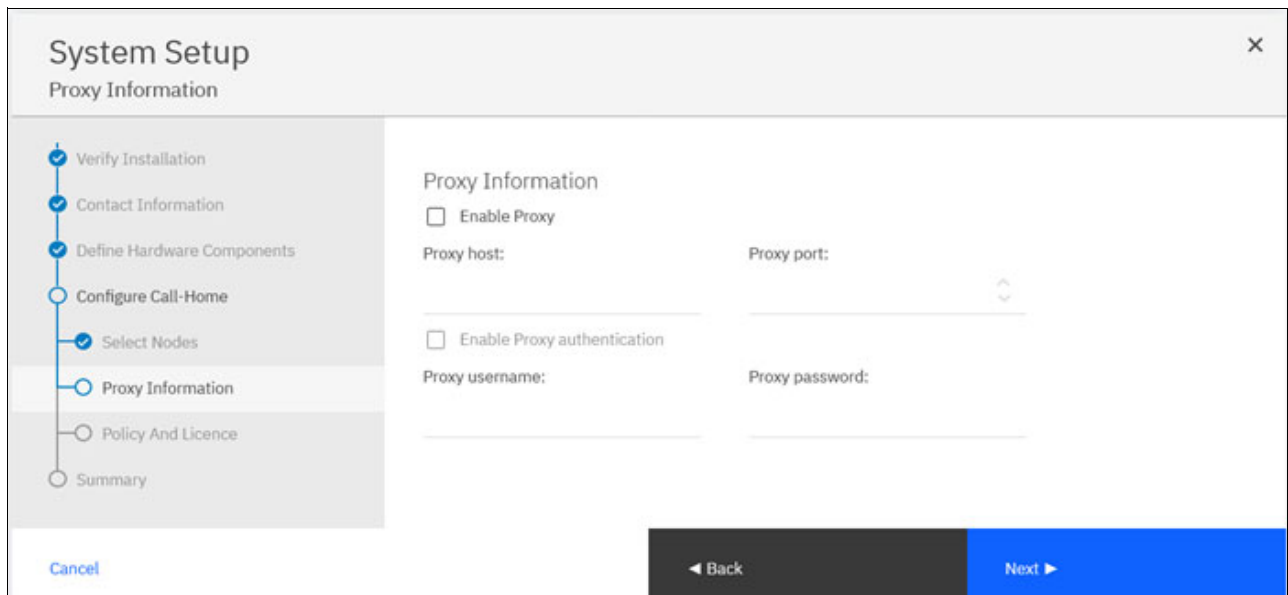


Figure 2-13 Configuring Call-Home: Proxy Information

- c. The Policy and Licence section is the final page to enable call home. Review and accept the IBM privacy policy to enable call home as shown in Figure 2-14.

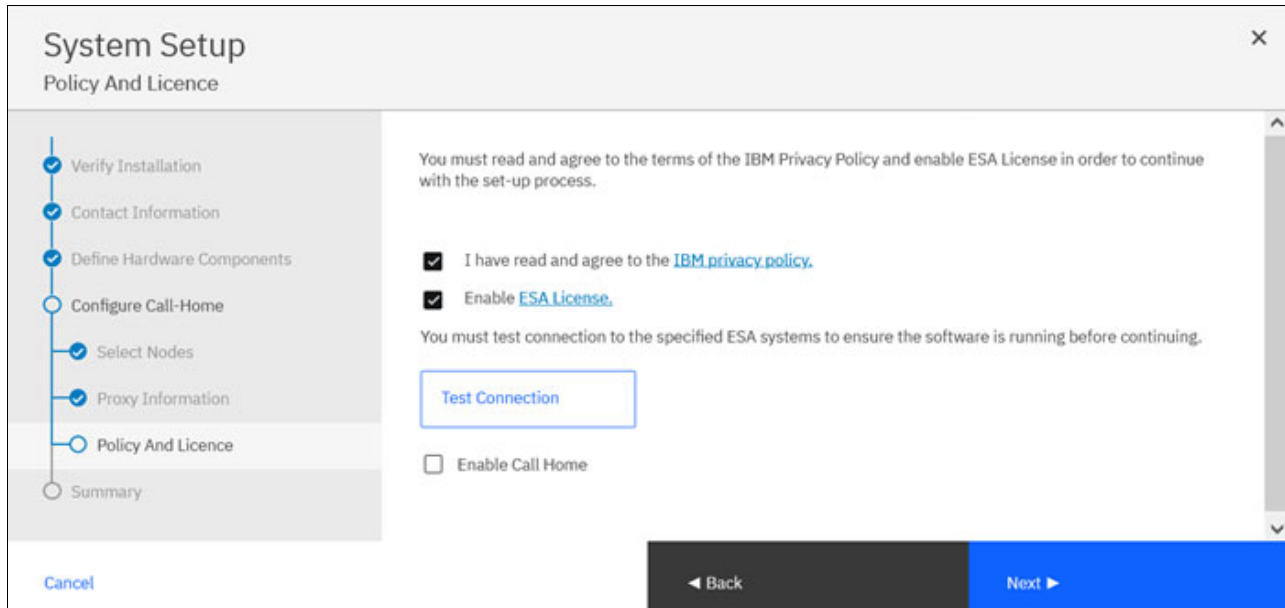


Figure 2-14 Configuring Call-Home: Policy and Licence

### 2.2.3 Using the GUI

After logging in to the GUI, the Overview page will be displayed. This page provides a view of all components in the environment and their health states. Clicking the numbers or links displays a more detailed view. See Figure 2-15.

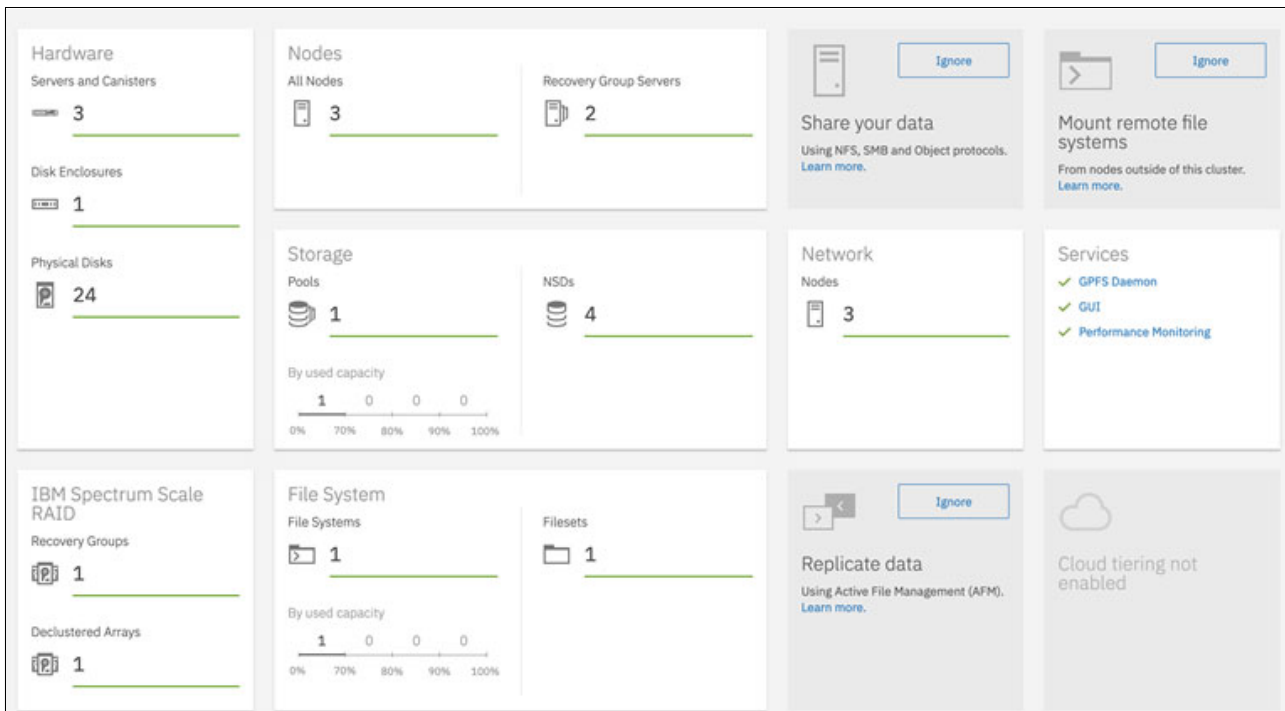


Figure 2-15 The Overview page

The header area of the GUI provides a quick view of the current health problems and tips for addressing them, if applicable. Also, some links to help resources are presented on the page.

Use the navigation menu on the left panel of the GUI to select other GUI pages as shown in Figure 2-16. Each GUI page has a unique URL, which allows the user to bookmark and directly access specific pages and log in and load the GUI directly to a specific section.

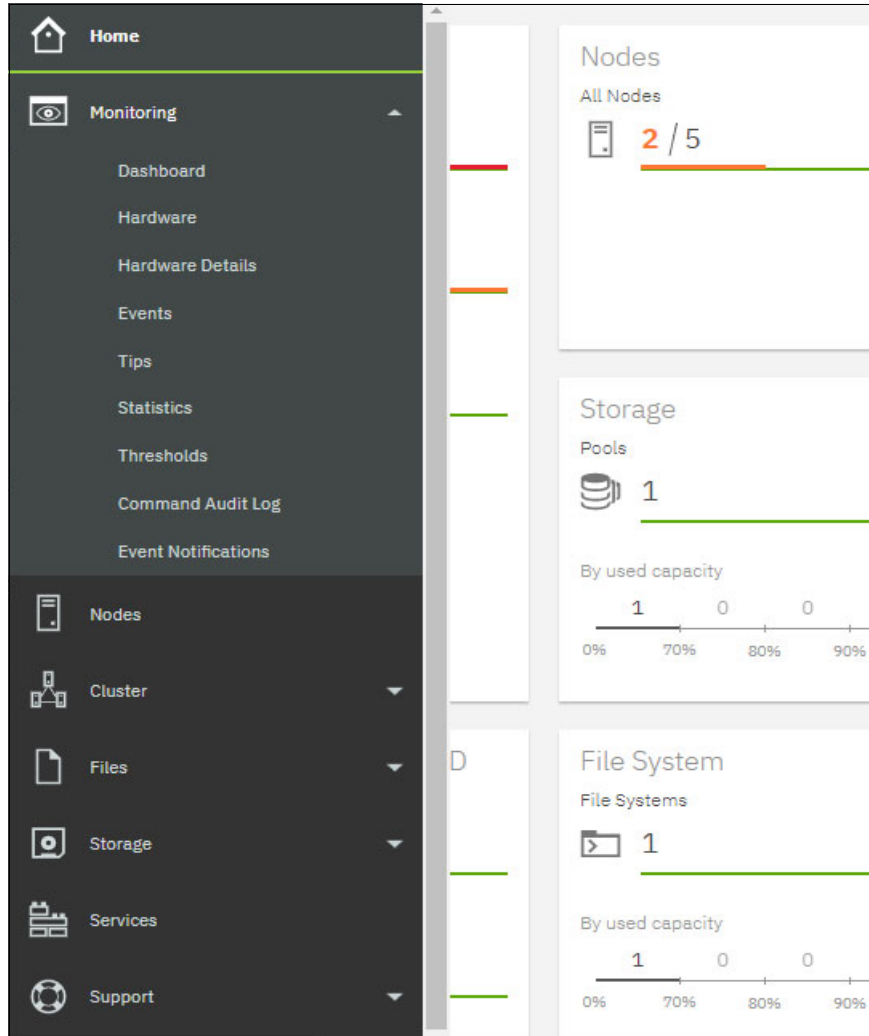


Figure 2-16 Navigation panel of the GUI

Some menus, such as Protocols, are only displayed when the related features, such as NFS, SMB, or AFM are enabled.

Most tables that are shown in the GUI have columns that are hidden by default. Right-click the table header as shown in Figure 2-17, and select the columns to display the columns.

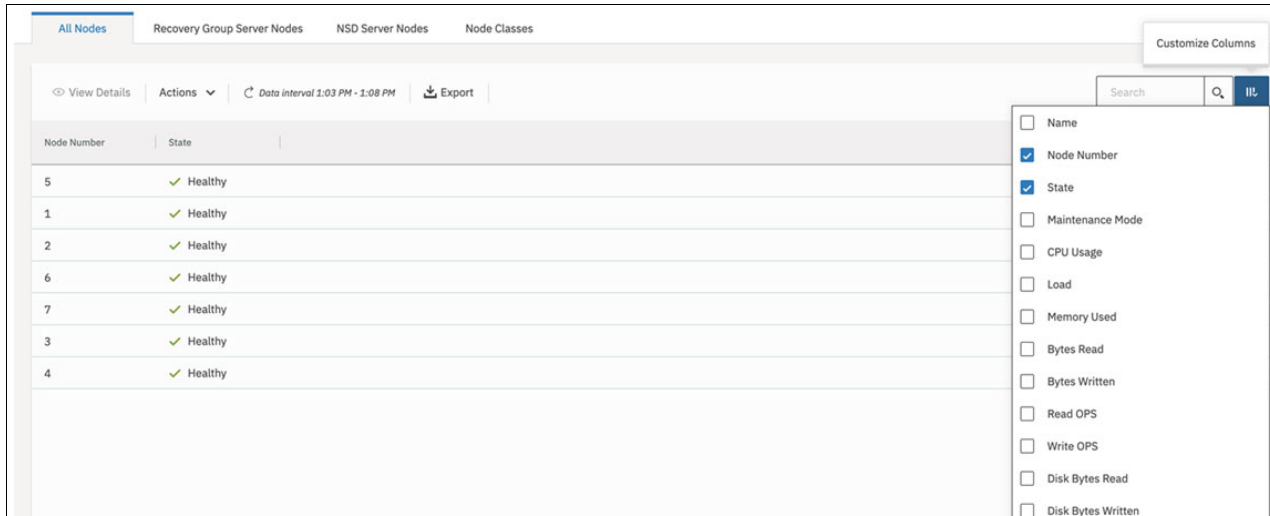


Figure 2-17 Showing and hiding table columns

The table values can be sorted by clicking one of the column headers. The arrow in the table header indicates which column is being used for sorting.

Double-click a table row to open a more detailed view of the selected item.

## 2.2.4 Monitoring of IBM ESS 3500 hardware

Select **Monitoring** → **Hardware** to list the IBM ESS 3500 enclosures within the racks. A table lists all enclosures and the related canisters. See Figure 2-18.

Name	Serial Number	State	Building Block	Type
5141-FN2-78E400Q	78E400Q	✓ Healthy	group1	4U102 Enclosure
5147-102-78T246A	78T246A	✓ Healthy	group1	4U102 Enclosure
ess3500rw6b-hs.test...	78E400QB	✓ Healthy	group1	Canister/Server
ess3500rw6a-hs.test...	78E400QA	✓ Healthy	group1	Canister/Server
ems9-hs.test.net	78A35BA	✓ Healthy		Management Server
5147-102-78T254a	78T254A	✓ Healthy	group1	4U102 Enclosure

Figure 2-18 Hardware page with two IBM ESS 3500 systems

Select **Edit Rack Components** when IBM ESS enclosures or servers are added or removed, or if their rack location changes.

Select **Replace Broken Disks** to start a guided procedure to replace any failed disks. Also, the IBM ESS 3500 can be configured to replace broken disks by using [commandless disk replacement](#).



Click the IBM ESS 3500 in the virtual rack to see more information about the IBM ESS 3500, including the physical disks and the two canisters as shown in Figure 2-19. Move the mouse over the components, such as drives and power supplies, to see more information. Clicking components opens a page with more detailed information on the specific component. Failed disks are indicated with the color red. Right-click the component to open a context menu that enables the user to replace the selected failed disk.

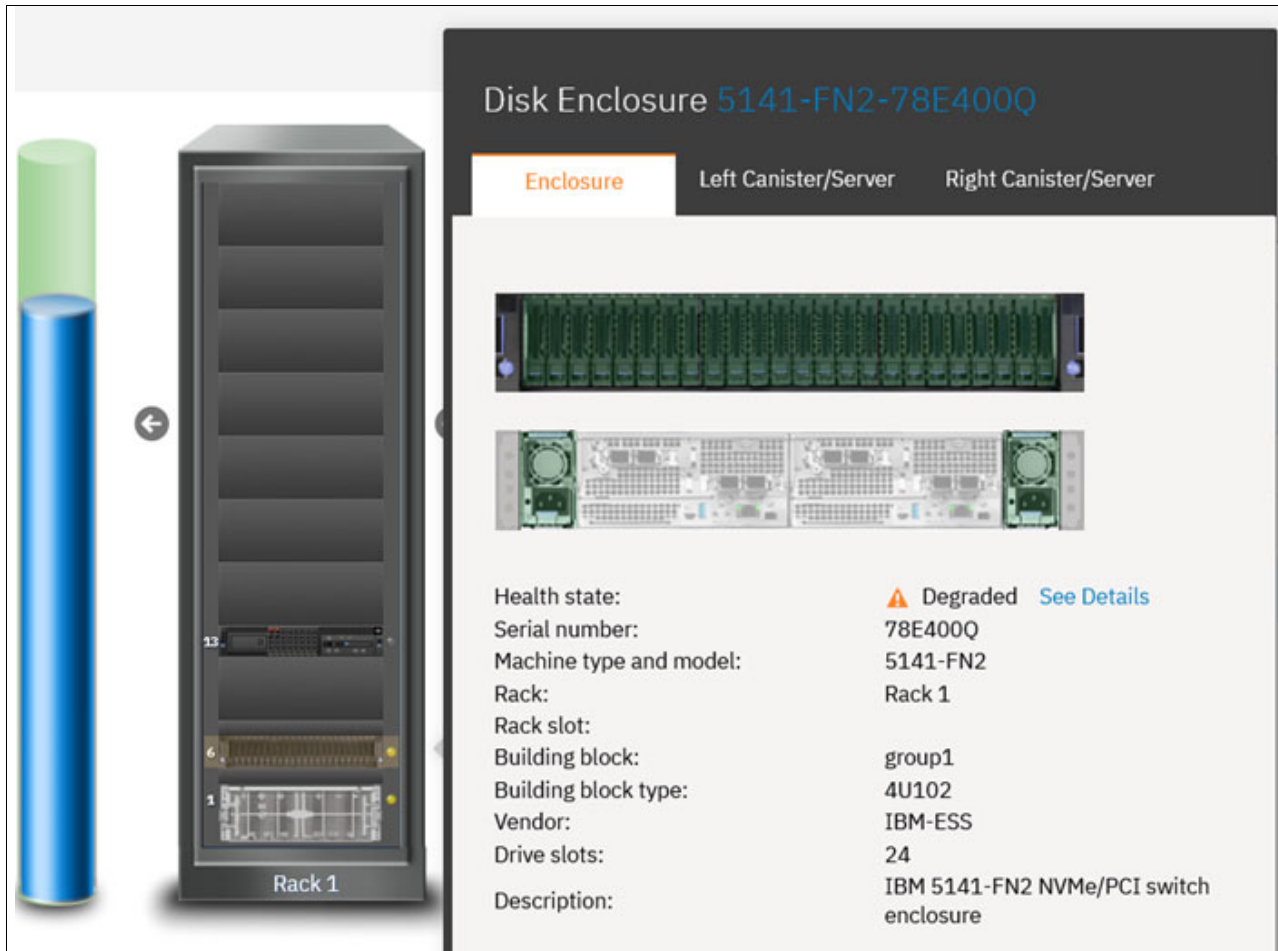


Figure 2-19 IBM ESS 3500 details in the Hardware page

If there is more than one rack, click the arrows that are displayed on the left and the right side of the rack to switch to another rack.

Select **Monitoring** → **Hardware Details** to display a more detailed information and the health states of the IBM ESS 3500 and its internal components. See Figure 2-20.

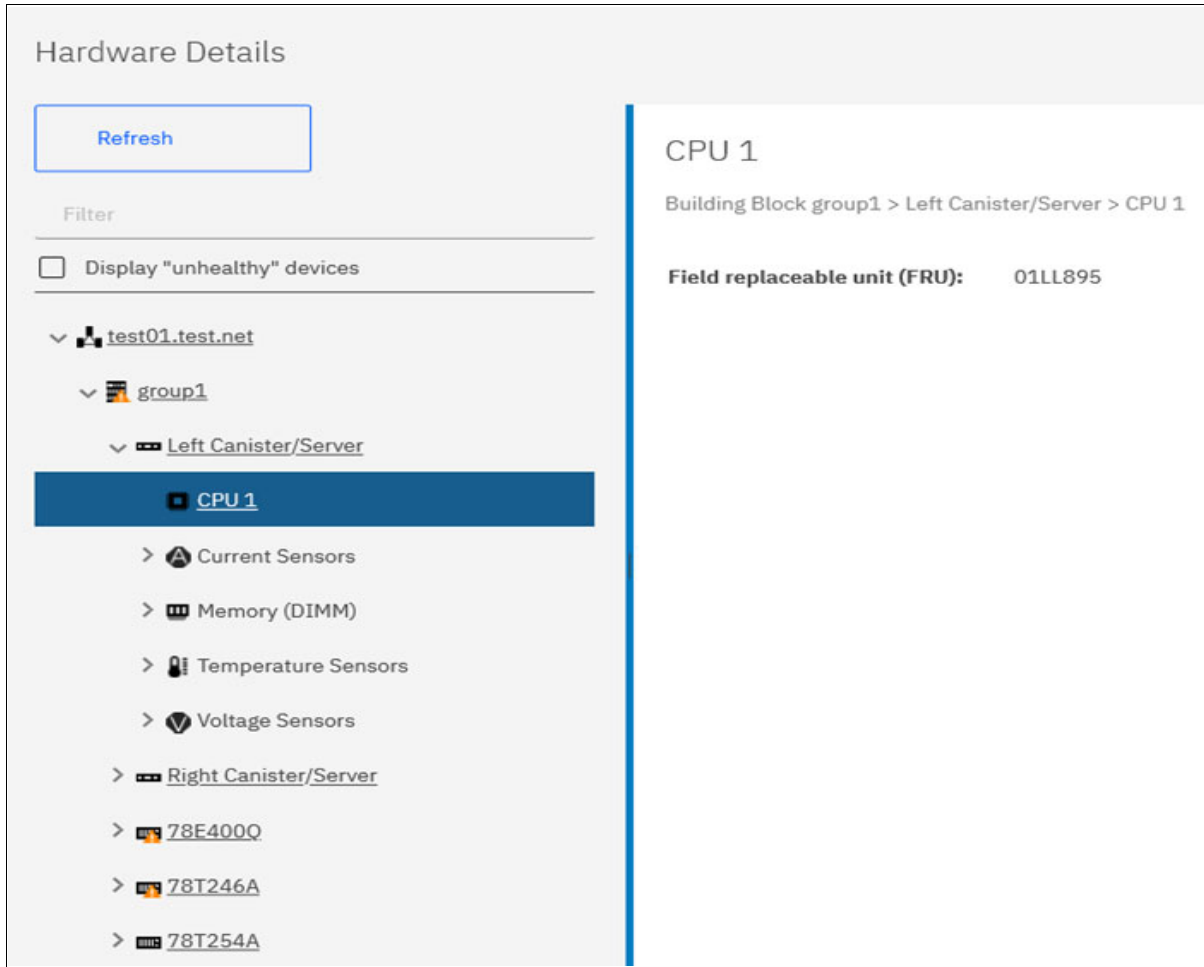


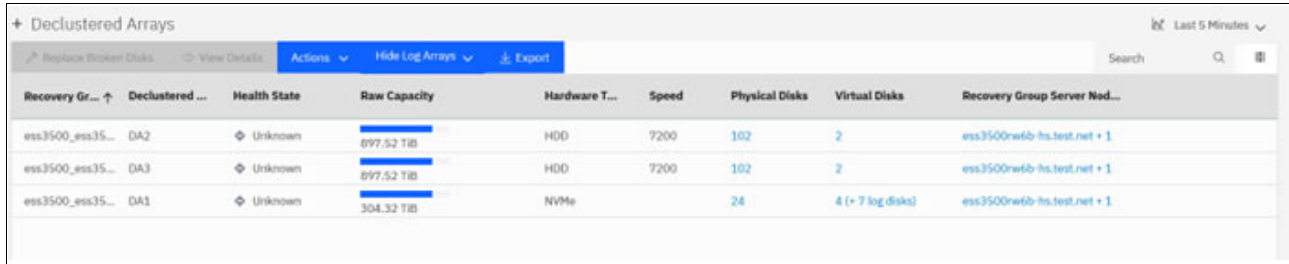
Figure 2-20 The Hardware Details page

The Hardware Details page allows the user to search for components by text and filter the results to display only unhealthy hardware.

Click the > icon on the tree nodes to display subsequent children. For example, to view all current sensors, the user can click **Current Sensors** of the canister in Figure 2-20.

## 2.2.5 Storage

The Storage menu provides views into the different storage components, such as the physical disks, declustered arrays, recovery groups, virtual disks, network shared disks (NSDs), and storage pools. The list of declustered arrays is shown in Figure 2-21.



Recovery Gr...	Declustered ...	Health State	Raw Capacity	Hardware T...	Speed	Physical Disks	Virtual Disks	Recovery Group Server Nod...
ess3500_ess35...	DA2	Unknown	897.52 TiB	HDD	7200	102	2	ess3500n66b-fts.test.net + 1
ess3500_ess35...	DA3	Unknown	897.52 TiB	HDD	7200	102	2	ess3500n66b-fts.test.net + 1
ess3500_ess35...	DA1	Unknown	304.32 TiB	NVMe		24	4 (+ 7 log disks)	ess3500n66b-fts.test.net + 1

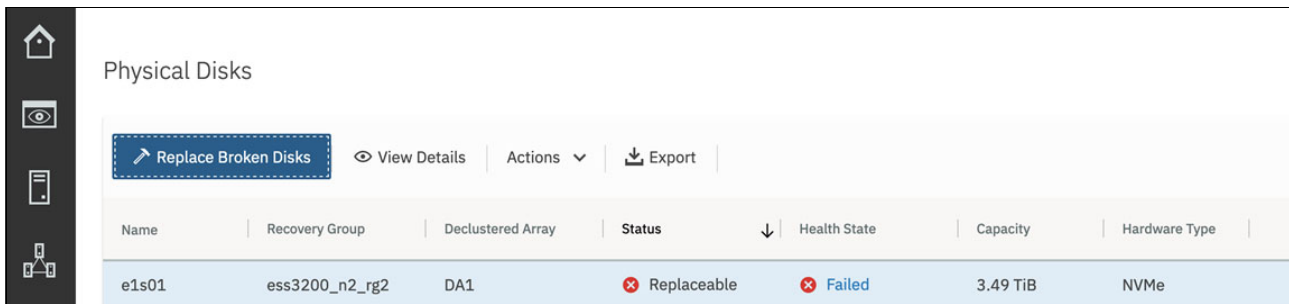
Figure 2-21 Declustered Arrays view

## 2.2.6 Replacing broken disks

The GUI provides a guided procedure that can be used to replace broken disks. Verify that the replacement disks have the same field-replaceable unit (FRU) numbers as the disks that are going to be replaced.

**Note:** If commandless disk replacement is enabled, the guided disk replacement is unavailable in the GUI.

This procedure can be started from multiple places within the GUI where disk notifications can be seen. For example, to check for broken disks, you can select **Storage** → **Physical Disks** as shown in Figure 2-22.



Name	Recovery Group	Declustered Array	Status	Health State	Capacity	Hardware Type
e1s01	ess3200_n2_rg2	DA1	Replaceable	Failed	3.49 TiB	NVMe

Figure 2-22 Physical Disks page

Select **Replace Broken Disks** to see a list of all broken disks that can then be selected for replacement. Select an individual disk from the table and select **Replace Disk** to replace the selected disk. In both cases, a fix procedure guides you through replacing the disks. See Figure 2-23.

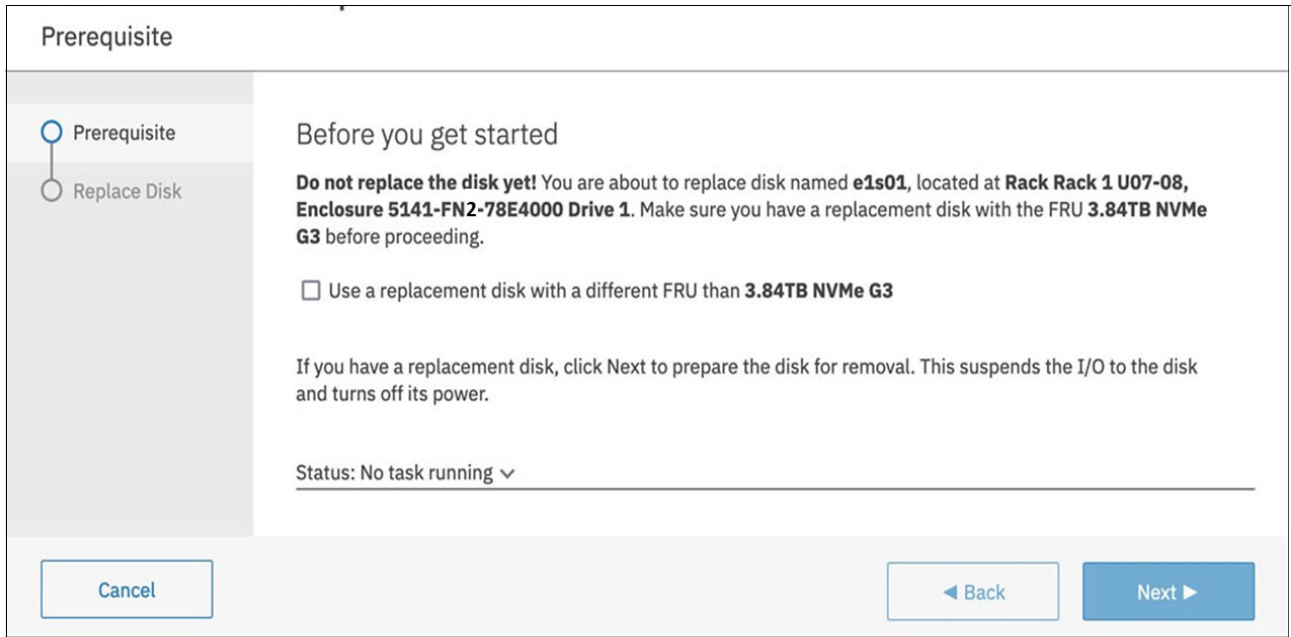


Figure 2-23 Fix procedure for replacing disks

## 2.2.7 Health events

Select **Monitoring** → **Events** to review the entire set of events that are reported in the IBM ESS system. Under the Event Groups tab, all individual events with the same name are grouped into single rows, which can be useful when many events are reported. The Individual Events tab lists all the events, including multiple occurrences of the same event name. Events are assigned to a component, such as canister, enclosure, or file system. The user can click any of the components in the bar chart above the grid to filter for events of that selected component. See Figure 2-24.

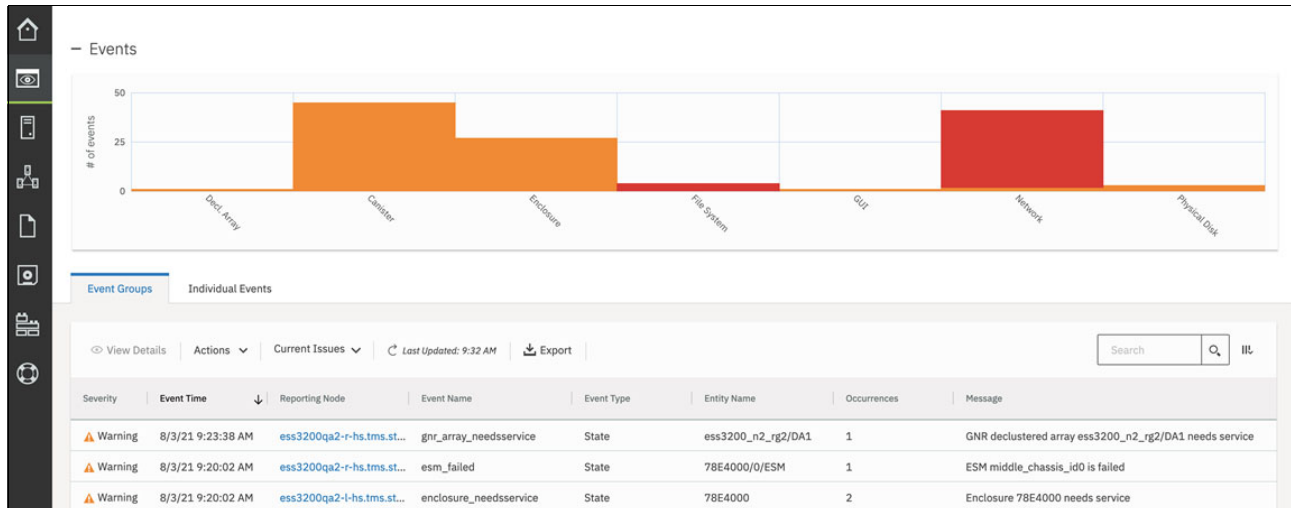


Figure 2-24 The Events page

The following filter options by event type are available as a drop-down list in the Events page shown in Figure 2-24:

- ▶ **Current Issues** displays all unfixed errors and warnings.
- ▶ **Notices** displays all transient messages of type “notice” that were not marked as read. While active state events disappear when the related problem is solved, the notices stay forever until they are marked as read.
- ▶ **Current State** displays all events that define the current state of the entities, and excludes notices and historic events.
- ▶ **All Events** displays all messages, even historic messages, and messages that are marked as read. This filter is not available in the Event Groups view because of performance implications.

The user can mark events of type Notices as read to change the status of the event in the Events view. The status icons become gray if an error or warning is fixed, or if it is marked as read.

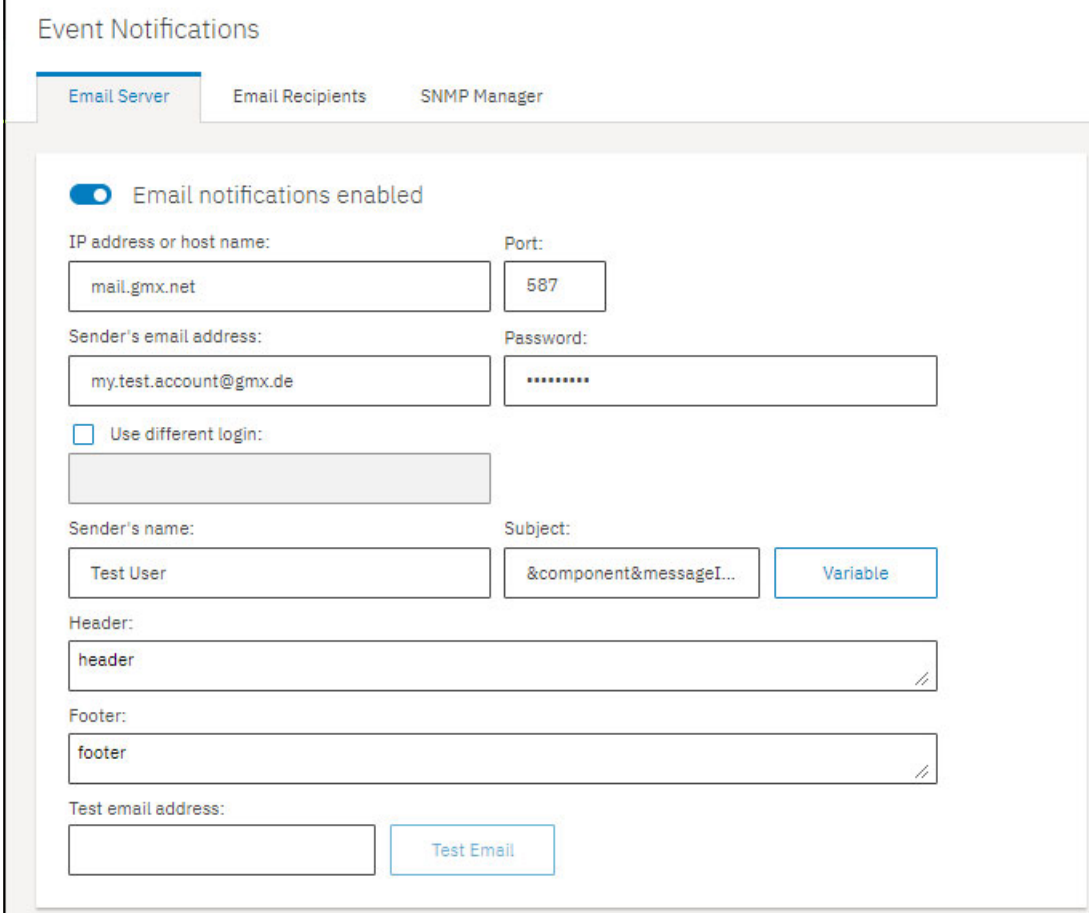
Some issues can be resolved by selecting **Run Fix Procedure**, which is available on select events. Right-click an event in the Events table to see this option.

## 2.2.8 Event notification

The system can send emails and Simple Network Management Protocol (SNMP) notifications when new health events appear. Any combination of these notification methods can be used simultaneously. Select **Monitoring** → **Event Notifications** to configure event notifications.

## Sending emails

Select **Monitoring** → **Event Notifications** → **Email Server** to configure the email server where the emails are sent. In addition to the email server, an email subject and the senders name can also be configured. Select **Test Email** to send a test-email to an email address. See Figure 2-25.



The screenshot shows the 'Event Notifications' configuration interface with the 'Email Server' tab selected. The 'Email notifications enabled' toggle is turned on. The configuration fields are as follows:

- IP address or host name:** mail.gmx.net
- Port:** 587
- Sender's email address:** my.test.account@gmx.de
- Password:** [masked with dots]
- Use different login: [empty field]
- Sender's name:** Test User
- Subject:** &component&messageI... (with a 'Variable' button next to it)
- Header:** header
- Footer:** footer
- Test email address:** [empty field] (with a 'Test Email' button next to it)

Figure 2-25 Configuring the email server

You can define multiple email recipients, selecting **Monitoring** → **Event Notifications** → **Email Recipients**. For each recipient, the user can select the components for which to receive emails, and the **For minimum severity level** (Tip, Info, Warning, or Error). Optionally, instead of sending a separate email per event, a daily summary email can be sent. You can also elect to receive a **Daily Quota report**. See Figure 2-26.

**Create Email Recipient**

Name:

Email address:

Select the type of content and level of detail that should be send to this recipient.

Event notifications by component:  ▼

For minimum security level:  ▼

Daily event summary by component:  ▼

Daily Quota reports:  ▼

Figure 2-26 Creating email recipient

## Sending SNMP notifications

Select **Monitoring** → **Event Notifications** → **SNMP Manager** to define one or more SNMP managers that receive an SNMP notification for each new event. Unlike with email notification, filters cannot be applied to SNMP notification, and an SNMP notification is sent for any health event that occurs in the system. For more information on configuring SNMP, see [Configuring SNMP manager](#) or use the GUI help topic for event notifications.

## 2.2.9 Dashboards

Select **Monitoring** → **Dashboard** to view an easy-to-read, single-page, real-time user interface that provides an overview of the system performance.

Some default dashboards are included with the product. Users can further modify, delete the default dashboards, and create new dashboards to suit their requirements. The same dashboards are available to all GUI users, so any modifications are visible to all users.

A dashboard consists of several dashboard widgets that can be displayed within a chosen layout.

Widgets are available to display the following items, as shown in Figure 2-27:

- ▶ Performance metrics
- ▶ System health events
- ▶ File system capacity by file set
- ▶ File sets with the largest growth rate in the last week
- ▶ Timelines that correlate performance charts with health events

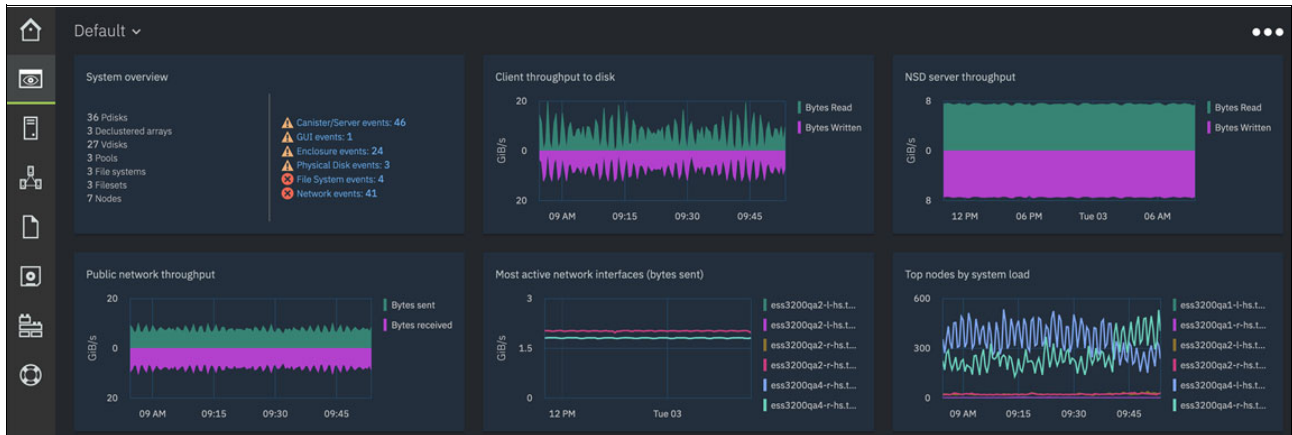


Figure 2-27 The dashboard

## 2.2.10 More information

The previous sections provided a rough overview of the GUI. For more detailed information on the GUI, read the [Monitoring and Managing the IBM Elastic Storage Server Using the GUI, REDP-5471](#) and use the online help pages that are included within the GUI. Additional information is also available in [IBM Spectrum Scale Version 5.1.1](#).

## 2.3 Software enhancements

In this section, the software enhancements in IBM ESS 3500 are described.

### 2.3.1 Containerized deployment

The IBM ESS installation and management software includes but is not limited to the following items:

- ▶ ESS-specific documentation for installation and upgrade scripts
- ▶ A container-based deployment model that focuses on ease of use
- ▶ Other tools for the IBM SSR to use for installing IBM ESS, such as essutils

Third-generation IBM ESS systems deploy a container-oriented management software stack in the IBM ESS Management Server that includes Ansible Playbooks for installation and orchestration.

IBM preinstalls this complete, integrated, and tested ESS solution stack on the ESS servers during manufacturing.



The ESS solution-stack levels are released as version, release, modification, and fix pack level.

For more information about the release levels of the ESS software solution and the levels of the software components for that ESS release level, see [ESS software deployment preparation](#).

For more information about Containerized deployment, see [ESS Quick Deployment Guide](#).

The IBM ESS solution-stack components are periodically updated, tested, and released as a new level of IBM ESS solution software. IBM recommends that clients plan to upgrade their IBM ESS to the current level of IBM ESS solution software stack at least once a year.

### 2.3.2 Red Hat Ansible

In the IBM ESS 3500 container, the Ansible library is included, which helps to orchestrate a set of commands (also called *tasks*). With this capability, you can automate the deployment process into a few commands.

Figure 2-28 shows the tree of the `ansible` directory that is included in the container.

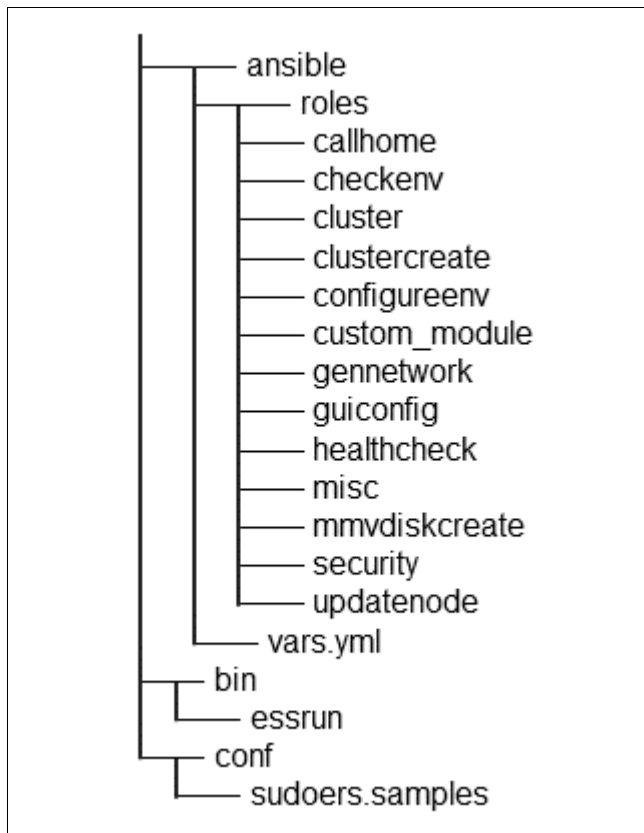


Figure 2-28 Ansible directory tree included in the container

The `roles` directory contains a set of folders that contain the various tasks that can be run, for example, `configureenv`, contains the **essrun config load** set of tasks.

If you want to import or use the roles within your own Ansible Playbook, you can import the roles and the `vars.yml` file because it contains several variables that are used within each role.

Example 2-1 shows how to import an ESS role into your own Ansible Playbook.

*Example 2-1 How to import an ESS role into an Ansible Playbook*

---

```
---
- name: ESS config load
  hosts: all
  remote_user: root

  vars_files:
    - /opt/ibm/ess/deploy/ansible/vars.yml

  # importing roles
  tasks:
    - include_role:
        name:
          /opt/ibm/ess/deploy/ansible/roles/configureenv
```

---

### 2.3.3 The `mmvdisk` command

The `mmvdisk` command is an integrated command suite for IBM Spectrum Scale RAID. It can simplify IBM Spectrum Scale RAID administration and encourages and enforces consistent best practices for server, recovery group, VDisk NSD, and file system configuration.

The `mmvdisk` command can be used to manage IBM Spectrum Scale RAID installations. If you are integrating IBM ESS 3500 with an existing installation of ESS systems that are not `mmvdisk` recovery groups, those systems must be converted into `mmvdisk` recovery groups before you add the IBM ESS 3500 into the same cluster.

For more information about the `mmvdisk` command, see [Managing IBM Spectrum Scale RAID with the `mmvdisk` command](#).

### 2.3.4 The `mmhealth` command

The `mmhealth` command monitors and displays the health status of services that are hosted on nodes and the health status of an IBM Spectrum Scale cluster. Although a cluster might be made up of many different types of components, `mmhealth` provides a health status of the varied components in the system, including any important cluster issues.

For any type of cluster, the status includes the status of the IBM General Parallel File System (GPFS) daemons, the node software status, tracking of events that happened to the cluster, and the file system health status.

The depth of the details for one node depends on a few factors:

- ▶ If the node is a software-only node, where IBM Spectrum Scale formats only external block devices to the cluster
- ▶ When the IBM ESS 3500, running with IBM Spectrum Scale RAID or running with IBM Spectrum Fusion™, is using `mmhealth` to monitor and report on the following, inexhaustive, list of items:
  - Hardware items which are the same as other IBM ESS hardware solutions:
    - Temperature of different sensors of the enclosure
    - Power supply hardware status
    - Fan speeds and status
    - Voltage sensors data
    - Firmware levels reporting and monitoring
    - Boot drive status and monitoring
  - IBM Spectrum Scale RAID specific items:
    - Recovery Groups status and monitoring
    - Declustered Array status and monitoring
    - Physical drives status and monitoring
    - VDisks status and monitoring
  - IBM Spectrum Scale software-related items:
    - NSD status and monitoring
    - Network communication status and monitoring
    - GUI status and monitoring (of the GUI nodes)
    - Status and monitoring [protocol support](#)
    - File system status and monitoring
    - Pool status and monitoring
    - NSD protocol and statistics
    - Statistics of other protocols when applicable

The `mmhealth` command provides IBM Spectrum Scale software-related checks across all node and device types present in the cluster. Software RAID checks are present across all IBM Spectrum Scale RAID offerings (such as IBM ESS 5000, IBM ESS 3500, and ECE). Devices (such as IBM ESS 3500) that are integrated with IBM Spectrum Scale hardware, also get hardware checks and monitoring.

For more information about how to use the `mmhealth` command, see [mmhealth command](#).

## Monitoring IBM Spectrum Scale RAID

This section provides links to additional documentation:

- ▶ [Commands to Monitor IBM Spectrum Scale RAID](#)
- ▶ [IBM Spectrum Scale Erasure Code Edition: Planning and Implementation Guide](#) which describes the `mmhealth` command in the following sections:
  - Section 7.7 shows general command usage.
  - Section 7.8 shows example use case scenarios.
- ▶ The `mmhealth` command with the complete command usage, features, and examples

## Updates to mmhealth command

Several significant changes were made to `mmhealth` command for the release of the IBM ESS 3500. The `mmhealth` command was updated to support monitoring these new features. The following list includes some of these differences:

- ▶ Architecture change to x86\_64 from IBM POWER
- ▶ Support for NVME drives
- ▶ External storage enclosures support
- ▶ Dual canister design within single building block

The `mmhealth` command includes several changes to support the IBM ESS 3000, the IBM ESS 3200, and the IBM ESS 3500.

- ▶ The “Canister events” category was included to support many of the differences between legacy IBM ESS systems and IBM ESS 3500.
- ▶ The “Server” category was also adjusted.

Both of these adjustments and other changes to `mmhealth` are included in the following sections.

## Canister events

These events are new and specifically added to support the new canister-based building-block configuration of the IBM ESS 3000, IBM ESS 3200, and IBM ESS 3500. For more information about, for example, events that are related to the boot drive, temperature, CPU, and memory, see [Canister events](#) in IBM Documentation.

A new command, `ess3kpl1t`, was created by GNR to provide CPU and memory health information to `mmhealth` with the following command path:

```
/opt/ibm/gss/tools/bin/ess3kpl1t
```

The `ess3kpl1t` command uses the following parameters:

```
usage: ess3kpl1t [-h] [-t SELECTION] [-Y] [-v] [--local]
```

Optional arguments:

<b>-h, --help</b>	Show this help message and exit.
<b>-t SELECTION</b>	Provide selection keyword [ <code>memory cpula11</code> ].
<b>-Y</b>	Select report listing.
<b>-v</b>	Enable additional output.
<b>--local</b>	Select localhost option.

The `ess3kplt` command can be used to inspect memory or CPU resources. Example 2-2 shows a sample output.

*Example 2-2 Sample ess3kplt output*

---

```
ESS3K Mem Inspection:
  InspectionPassed:      True
  Total Available Slots: 8 (expected 8)
  Total Installed Slots: 8 (expected 0 or 8)
  DIMM Capacity Errors: 0 (Number of DIMMs with a size different from ['64 GB'])
  DIMM Speed Errors:    0 (Number of DIMMs with a speed of neither 3200 MT/s nor 3200
MT/s MT/s)
  Inspection DateTime:   2021-08-31 14:02:28.338486

ESS3K Cpu Inspection:
  InspectionPassed:      True
  Total CPU Sockets:    1 (expected 1)
  Total Populated Sockets: 1 (expected 1)
  Total Enabled CPU Sockets: 1 (expected 1)
  Total Cores:          48 (expected [48])
  Total Enabled Cores:  48 (expected [48])
  Online CPUs:          ---
  Total Threads:        96 (expected 96)
  CPU Speed Errors :    0 (Number of CPUs with a speed different from ['3300 MHz'] MHz)
  Inspection DateTime:   2021-08-31 14:02:28.562756
```

---

Example 2-3 shows a sample verbose output.

*Example 2-3 Sample ess3kplt verbose output*

---

```
ess3kplt:memory:HEADER:version:reserved:reserved:location:size:speedMTs:
ess3kplt:memorySummary:HEADER:version:reserved:reserved:availableSlots:installedSlots:capacityEr
ror:speedError:inspectionPassed:
ess3kplt:memory:0:1::PO_CHANNEL_D:passed:passed:
ess3kplt:memory:0:1::PO_CHANNEL_C:passed:passed:
ess3kplt:memory:0:1::PO_CHANNEL_B:passed:passed:
ess3kplt:memory:0:1::PO_CHANNEL_A:passed:passed:
ess3kplt:memory:0:1::PO_CHANNEL_E:passed:passed:
ess3kplt:memory:0:1::PO_CHANNEL_F:passed:passed:
ess3kplt:memory:0:1::PO_CHANNEL_G:passed:passed:
ess3kplt:memory:0:1::PO_CHANNEL_H:passed:passed:
ess3kplt:memorySummary:0:1::8:8:0:0:true:
ess3kplt:cpu:HEADER:version:reserved:reserved:location:speedMHz:status:status2:numCores:numCores
Enabled:numThreads:
ess3kplt:cpuSummary:HEADER:version:reserved:reserved:totalSockets:populatedSockets:enabledSocket
s:totalCores:enabledCores:totalThreads:speedErrors:inspectionPassed:
ess3kplt:cpu:0:1::CPU0:passed:ok:ok:48:48:96:
ess3kplt:cpuSummary:0:1::1:1:1:48:48:96:0:true:
```

---

CPU and DIMM-related events reported by the `mmhealth` command rely on the `ess3kplt` command in the IBM ESS 3000, IBM ESS 3200, and IBM ESS 3500 environments.

## 2.4 RAS enhancements

IBM ESS 3500 is the next generation of the IBM ESS product family that is built on a high availability and performance storage server platform.

The IBM ESS 3500 system consists of the server portion, MTM 5141-FN2, and the storage enclosure, MTM 5147-102 (1 – 8 JBOD drawers are supported). For the purposes of this document, the focus is on the 2U form factor server portion.

IBM ESS 3500 is designed to provide an improved customer experience compared to previous ESS releases. The IBM ESS 3500 has improvements in the following areas:

- ▶ Ordering
- ▶ Installing
- ▶ Upgrading
- ▶ Using
- ▶ Servicing

The following list includes the key components of the server portion, 5141-FN2:

- ▶ IBM storage enclosure with commercial NVMe drives
- ▶ Red Hat Enterprise Linux (RHEL) 8.4 with NVMe support
- ▶ IBM Spectrum Scale 5.1.3.x software features and functions
- ▶ IBM Spectrum Scale Software RAID

IBM ESS 3500 is an IBM installed product with a combination of customer-replaceable units (CRUs) and FRUs.

### 2.4.1 RAS features

IBM ESS 3500 was designed with improvements to reduce the frequency of failures, minimize workload interruptions, and easily detect, identify, and report problems to decrease service-repair time. It also incorporates redundancy into its design so that component replacements do not interfere or affect system operations.

IBM ESS 3500 is designed to offer high system and data availability with the following features:

- ▶ Dual-active, intelligent node canisters with mirrored cache
- ▶ Erasure coding for data durability that uses the RAID component of IBM Spectrum Scale for ESS, supporting a combination of 3-way, 4-way, 8+2P, and 8+3P erasure codes
- ▶ Checksums and versions of data blocks with *always on* checks for data validity
- ▶ Rapid rebuild of failed drives or data blocks with detected errors
- ▶ Dual-port flash drives with automatic drive failure detection and RAID rebuild
- ▶ Redundant hardware, including power supplies and fans
- ▶ Hot-swappable and client replaceable components
- ▶ Automated path failover support for the data path between the server and the drives
- ▶ Embedded BMC and remote control capability
- ▶ Monitoring
  - Hardware components
  - Firmware levels

- GNR and IBM Spectrum Scale components
- ▶ Event notification
- ▶ Call home
- ▶ IBM Spectrum Scale Healthchecker
- ▶ First-time data capture (FTDC)

To maintain high levels of system availability, multiple methods and services are used to monitor the various system components. System status changes are reported by notifications that provide detailed event information, which also includes user-action information for next-steps or required actions to address the issue. If callhome is enabled, IBM ESS 3500 can generate a call home for inventory updates and for various failures that can occur on the system. This automated service processes and sends the proper diagnostic information to IBM Support servers to assist with debug and troubleshooting.

IBM ESS 3500 RAS features consist of the following items:

- ▶ Monitoring
  - Hardware components
  - Firmware levels
  - GNR and IBM Spectrum Scale components
- ▶ Event notification
- ▶ Call home
- ▶ IBM Spectrum Scale Healthchecker
- ▶ First-time data capture (FTDC)

## 2.4.2 Enclosure overview

IBM ESS 3500 includes node-to-node communication through an internal ethernet private network and nontransparent bridge (NTB) for peer node diagnostic and control. Remote console through serial over LAN (SOL) using baseboard management controller (BMC) intelligent platform management interface (IPMI) is available for monitoring and controlling the IBM ESS 3500 enclosure and to assist with deployment and installation.

Several methods of power control for the canister and drive slots are available on the system to assist with system recovery and troubleshooting, which helps to decrease component downtime. LED power indicators are in place on the front of the enclosure, drive carrier, fan, power module, and the canister as a visible sign that the hardware is receiving power. LED status indicators are used for the drives, fans, canisters, power modules, and enclosures that help point out if any components might be experiencing issues.

IBM ESS 3500 offers an NVMe drive with the following capacity with either 12-drive or 24-drive installation options:

- ▶ 3.8 TB
- ▶ 7.6 TB
- ▶ 15.3 TB
- ▶ 30.72 TB

The drives are accessible from the front for service without having to extend the drawer to a service position. Figure 2-29 shows the front view of the IBM ESS 3500 enclosure.

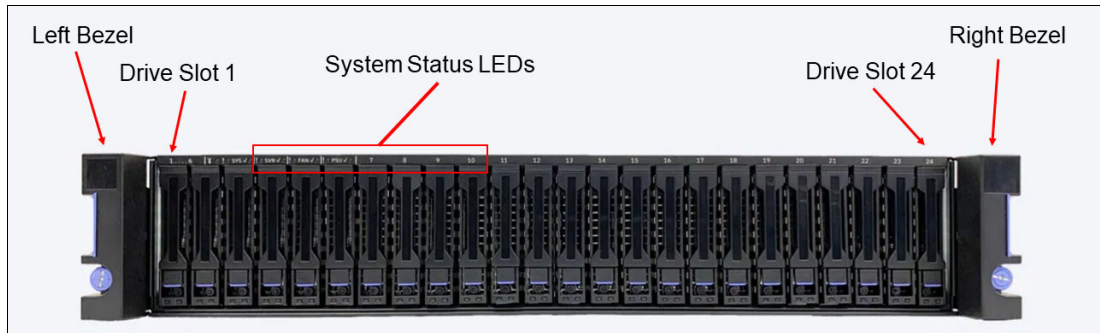


Figure 2-29 Front view of the IBM ESS 3500 enclosure

The canister FRUs and the power modules are accessible from the rear without extending the drawer into a service position. Figure 2-30 shows the rear view of the IBM ESS 3500 enclosure (the 5141-FN2 portion).

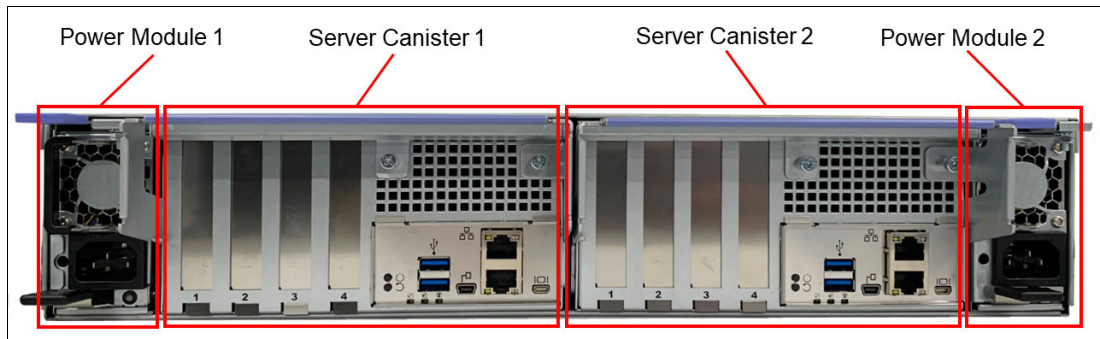


Figure 2-30 Rear view of the IBM ESS 3500 enclosure

Cables are connected at the rear of the enclosure. Figure 2-31 shows the ports and cable connections for IBM ESS 3500 enclosure (the 5141-FN2 portion).

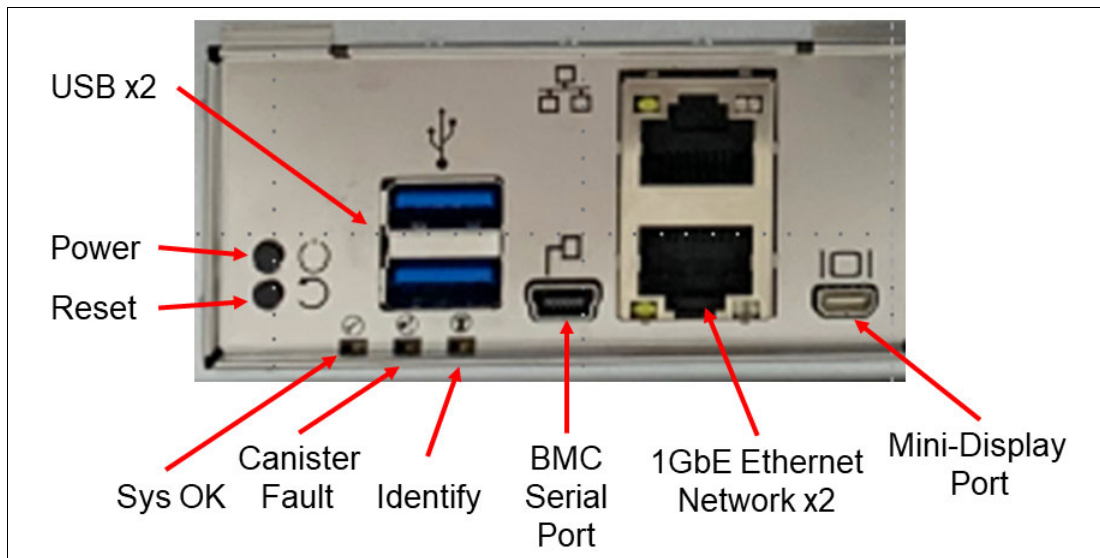


Figure 2-31 Port map of the 5141-FN2 (IBM ESS 3500)



## Memory configuration details

The IBM ESS 3500 canisters each contain 8 DIMM slots and are available in two memory configuration options:

- ▶ 512 GB with 8 × 64 GB 64GB DDR4 3200 MT/s Memory DIMMs (#AJZV)
- ▶ 1 TB with 8 × 128 GB 128 GB DDR4 Memory DIMMs (#AJPW)

Memory DIMMs cannot be mixed.

## Disk details

IBM ESS 3500 uses a mirrored set of 960 GB M.2 NVMe SSD drives as the boot disks. The M.2 SSD includes the Power Loss Protection (PLP) feature. Given the IBM Spectrum Scale RAID design and given the M.2 PLP feature to ensure that the data is persistent for IBM Spectrum Scale RAID log files that are maintained in the boot disks, IBM ESS 3500 does not require a Battery Backup Unit (BBU).

## Networking details

When planning to install a 100G adapter, the following adapters are available:

- ▶ (#AJP1) - PCIe4 LP 2-Port VPI 100 Gb IB-EDR / Ethernet adapter
- ▶ (#AJZL) - CX-6 InfiniBand / VPI in PCIe form factor (InfiniBand and Ethernet, no IPsec)
- ▶ (#AJZN) - CX-6 DX in PCIe form factor (Ethernet only, IPsec)

## 2.4.3 Machine type model and warranty

IBM ESS 3500 server has a single MTM value: 5141-FN2. It includes a 3-year warranty.

IBM ESS 3500 also offers same-day service upgrade options, and optional priced services that include lab-based services (LBS) installation.

## 2.4.4 Components: FRU and CRU

The components within the IBM ESS 3500 enclosure are similar to the previous IBM ESS 3200:

- ▶ Dual-canister architecture with two hot swappable I/O canister nodes
- ▶ Design for easier access to the system by using rear access for replaceable parts and a simple pull-down lever mechanism.
- ▶ Six Fan units (5+1 redundant) at the front of the enclosure from the top, which allows for removal and replacement without interrupting system functionality.

The IBM ESS 3500 contains some notable improvements compared to previous ESS versions:

- ▶ The canisters are modified with a focus on increased serviceability over IBM ESS 3200.
  - Each canister has two additional adapter slots, allowing for added connectivity and support for attaching additional storage enclosures.
  - No removable screws when accessing FRU parts inside a canister.

- A redesigned single canister lid, with no attached riser, improves serviceability and accessibility. All adapters now attach to the riser on the canister system board.
- Service Port (SSR port) added. See Figure 2-32.

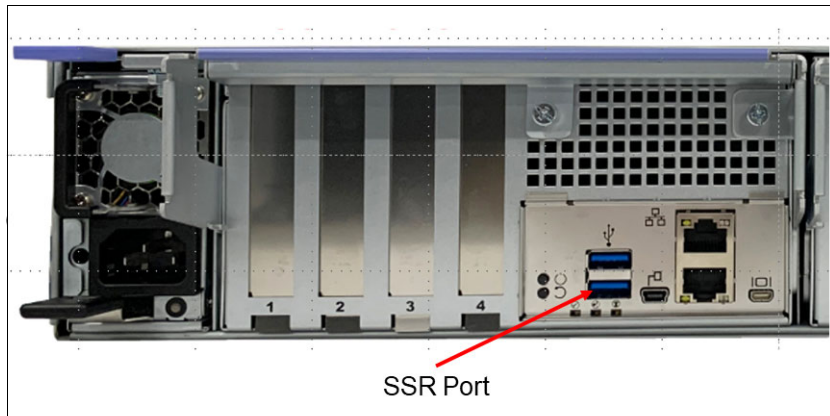


Figure 2-32 SSR port on rear of the device

The fans can be accessed for service by extending the drawer into a service position as shown in Figure 2-33.

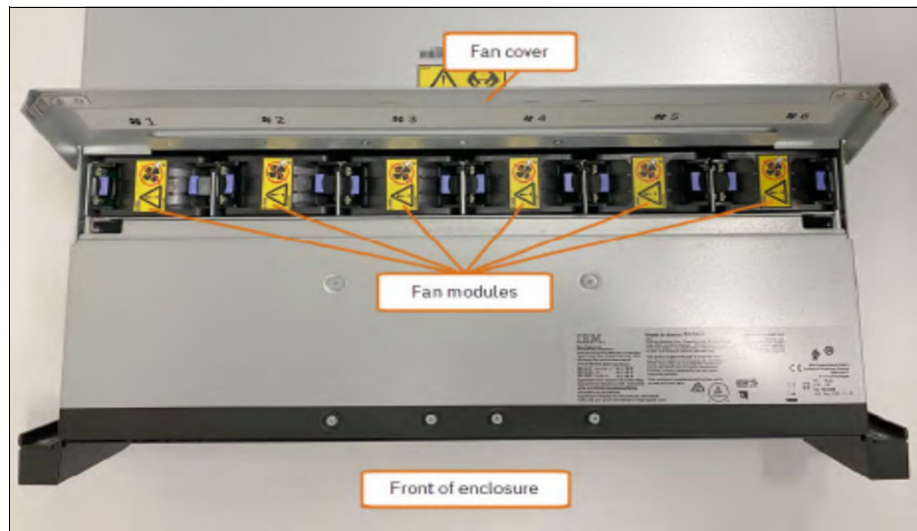


Figure 2-33 Fan cover access to IBM ESS 3500

Replacing the enclosure and the rails also requires extending the drawer to the service position.

In general, IBM ESS 3500 service strategy takes the following approach:

- ▶ Any hardware component that can be accessed only by opening the cover of the canister is considered to be a FRU. A FRU requires IBM service personnel to perform any needed service or repair to those components that are inside the canister cover.
  - FRUs
    - Fan module (from top)
    - Server Module Canister FRU kit (from rear) – FRU
    - Canister top lid FRU kit - FRU
    - Additional FRUs (inside canister)
      - Boot Drive
      - DIMM Memory
      - Coin Cell Battery
      - TPM
      - PCIe network adapter
      - Enclosure Chassis, including midplanes, which cannot be replaced separately
      - Cable Management Arm (CMA)
- ▶ Any hardware component that can be accessed for maintenance without removing the canister is typically a CRU. You can perform the repair by using assistance and information from the ESS GUI or related IBM Documentation.
  - CRUs
    - NVMe drives at the front
    - Power Supply Module at the rear
    - Enclosure bezel kit
    - Drives: 3.84 TB, 7.68 TB, 15.36 TB, 30.72 TB, Drive Filler

## 2.4.5 Maintenance and service procedures

The IBM ESS 3500 platform can identify a failed component for repair based on the monitoring and failure isolation capability of the RAS software. Guided maintenance and service procedures are available in the [IBM ESS 3500 Service Guide](#).

Because IBM Documentation is a single point of reference that provides information about IBM systems hardware, operating systems, and server software, it is recommended that users search for IBM ESS 3500 to get the most recent updates. The online IBM documentation is actively maintained and updated with the latest information.

When an issue is identified, a *Service Guide* option is listed that brings up the appropriate service procedure. The ESS Service Guide can also be downloaded by using the link in the navigation.

Concurrent maintenance repair and updates are available on the IBM ESS 3500 for servicing and replacing of FRUs and CRUs in the system. This allows for maintenance to be performed on the system while it is used in normal operations. In addition, CRUs are hot-swappable components in the IBM ESS 3500 that allow for replacement without powering down the server. Components such as fans and power modules are supported as concurrently removable and replaceable parts of the IBM ESS 3500.

## 2.4.6 Software-related RAS enhancements

Managing and monitoring software components of the IBM ESS 3500 is critical to obtain the necessary detail from the command-line interface to properly debug a failing component. To increase availability to physical disks, there is physical disk redundancy in the IBM ESS 3500, where each server has a path to the data physical disks, but the paths from both servers are always active. In addition, to achieve optimal performance on the ESS, a shared recovery group layout is available that allows both servers in an IBM ESS 3500 building-block to concurrently access all the available drives and their bandwidth. Additional information on recovery groups is provided in the next section.

### FRU and location

The `mmlsenclosure` command displays the environmental status of IBM Spectrum Scale RAID disk enclosures. The enclosure status reported by the `mmlsenclosure` command (and consequently, the GUI) displays the FRU number and the location of that FRU within the enclosure to help diagnose failures. Information about multiple components, including fans and power modules, is reported with an indication for service, if needed. For more information, see the [mmlsenclosure command](#).

### Enclosure components

Several component statuses are reported by the enclosure as shown in Table 2-4.

Table 2-4 Component status reported by the `mmlsenclosure` command

Component	Command output
canister	Status of left and right node canister
cpu	Status of each CPU in each canisters
dimmm	Status of each memory module associated with a canister
fan	Status of each fan in canisters and power supplies

## 2.4.7 Call home functions

Concurrent with the release of the IBM ESS 3500 is a new type of call home. In this document, the term “legacy call home” refers to what has been in place for previous ESS solutions. ESS legacy call home applies to IBM ESS 5000, IBM ESS 3000, and IBM ESS 3200. It is also referred to as hardware and software call home. ESS legacy call home uses the Electronic Service Agent (ESA) on the EMS. It uses legacy IBM infrastructure.

IBM ESS 3500 uses a new version of the call home software that is called *Call Home Connect Cloud*. *Call Home Connect Cloud* applies to the EMS plus IBM ESS 3500 for all clusters that contain at least one IBM ESS 3500.

*Call Home Connect Cloud* is a more holistic or unified approach to addressing service needs for the hardware. The new code in the command `mmsysmon` runs on the EMS and IO nodes and is part of the IBM Spectrum Scale code base. Now, as part of the IBM Spectrum Scale code, software-based call home channels all of the needed information to IBM by using cloud-based infrastructure to process the data.

Daily call home still uploads files directly to ECuRep.

The main advantage of the new call home is that it addresses fragmentation of hardware and software data. No longer are there two different systems or sets of data, “software call home” and “hardware call home”.

The list provides some other advantages of using *Call Home Connect Cloud*:

- ▶ Enables easier integration of internal tools inside IBM (Health Checker, quality databases, customer portals).
- ▶ Call home messages are not triggered solely for support cases. Call home events can trigger additional events for monitoring.
- ▶ Allows two ways communication with customer systems (IBM Spectrum Scale back channel).

For more information about how call home works, see the [IBM Elastic Storage System Deployment Guide](#).

For a complete background and overview of ESA, see [IBM Electronic Service Agent](#).

## 2.5 Performance

Measurements in the IBM lab on a freshly installed and fully populated IBM ESS 3500 achieved a sequential-read performance of up to 91 GBps and a sequential write-performance of up to 58 GBps when using an InfiniBand network with Remote Direct Memory Access (RDMA) enabled.

**Note:** The performance measurements that are referenced in this document were made by using standard benchmarks in a controlled environment. The actual performance might vary depending on several factors such as the interconnection network, the configuration of client nodes, and the workload characteristics.

Some factors related to IBM ESS 3500 performance are listed in the following sections.

### 2.5.1 Networks

Network components play a key role in the overall performance of IO operations. This section provides details of types of network hardware, associated configuration, and utilities in assessing the network device throughput.

#### High-speed network type

The first choice that must be made is to decide between configuring IBM Spectrum Scale to use Ethernet or InfiniBand.

One of the desirable features of InfiniBand is its RDMA capability. With RDMA enabled, the communication between servers can bypass the operating system kernel, so the applications have lower latency and CPU utilization.

With an Ethernet network that uses the TCP/IP protocol, the communications must go through the kernel stack, resulting in higher latencies than RDMA and reduced read and write bandwidths.

RDMA is available on standard Ethernet-based networks by using the RDMA over Converged Ethernet (RoCE) interface. For more information on how to set up RoCE, see [Highly Efficient Data Access with RoCE on IBM Elastic Storage Systems and IBM Spectrum Scale](#).

After the IBM ESS 3500 is installed, the network type can be changed, if required, by using the `mmchnode` command. For more information, see the `mmchnode` command.

When using RDMA, verify the following settings by using the `mm1sconfig` command. These settings can be modified as needed by using the `mmchconfig` command:

- ▶ The `verbsRdma` option controls whether RDMA (instead of TCP) is used for NSD data transfers. Valid values are `enable` and `disable`.
- ▶ The `verbsRdmaSend` option controls whether RDMA is used instead of TCP and is also used for most nondata IBM Spectrum Scale daemon-to-daemon communication. Valid values are `yes` and `no`.
- ▶ The `verbsPorts` option specifies the device names and port numbers that are used for RDMA transfers between IBM Spectrum Scale client and server nodes. You must enable `verbsRdma` to enable `verbsPorts`.

## Bandwidth optimization when using TCP/IP

IBM Spectrum Scale achieves high IO throughput by establishing multiple connections between source and destination. This section describes the related configuration for optimal IBM ESS 3500 network device bandwidth.

### *Multiple connections over TCP*

Multiple connections over TCP (MCOT) feature establishes multiple TCP connections between nodes (with the same daemon IP address that is used on each end) to optimize the use of network bandwidth. The number of connections is controlled through the `maxTcpConnsPerNodeConn` parameter, which can be changed by using the `mmchconfig` command. Valid values are 1-8, with the default of 2.

The value that is assigned to the `maxTcpConnsPerNodeConn` parameter must be defined after you consider the following factors:

- ▶ The overall bandwidth of the cluster network
- ▶ The number of nodes in the cluster
- ▶ The value that is configured for the `maxReceiverThreads` parameter
- ▶ Memory resource implications of setting a higher value for the `maxTcpConnsPerNodeConn` parameter. For more information about Configuring MCOT, see [Recommendations for tuning maxTCPConnsPerNodeConn parameter](#) in IBM Documentation.

## Link aggregation

The bonding or link aggregation can be an important factor for Ethernet TCP/IP performance. Most deployments use the Link Aggregation Control Protocol (LACP) standard based on IEEE 802.3ad as the aggregation mode. The LACP aggregation determines the interface to use based on the hash of the source and destination information of a packet.

With multiple connections over TCP (MCOT), multiple TCP port numbers are used. By using LACP `xmit_hash_policy=1` or by using `layer3+4` in which the hash is generated by using the IP and Port information of the source and destination, a better chance exists of using multiple interfaces between a particular pair of nodes. The load-balancing algorithm on the switch is also important to ensure better balancing across links from switch to destination.

For Multi-Rail Over TCP support, refer to 2.6, “IBM Spectrum Scale Multi-Rail over TCP and RDMA over Converged Ethernet” on page 52.

### Assessing network bandwidth

The network bandwidth can be assessed by using a tool like **nsdperf**. For an overview and usage instructions about this tool, see [IBM Spectrum Scale and IBM Spectrum Scale and IBM Elastic Storage system Network Guide, REDP-5484](#).

## 2.5.2 Non-volatile memory express drives

Non-volatile memory express (NVMe) is an interface by which non-volatile storage media can be accessed through a PCIe bus. As a result of the efficiency of the protocol, NVMe generally provides better performance over alternatives, such as Serial Advanced Technology Attachment (SATA), when comparing devices that share the same underlying technology.

**Note:** A NAND flash NVMe drive has the potential for improved performance over a NAND flash SATA drive because of its more efficient bus connection and protocol improvements. For example, NVMe allows for longer command queues.

### NVMe drives I/O completion

The time that is taken by NVMe drives to complete an I/O request accounts for only a portion of the overall time that it takes for IBM Spectrum Scale to complete the I/O request. To get a more detailed list of time spent, you can run the following command on the IBM ESS 3500 and client nodes that are processing I/O requests.

```
/usr/lpp/mmfs/bin/mmdiag --iohist
```

At the lowest layer is the physical disk (Pdisk) I/O times, obtained by running this command on an IBM ESS 3500 server and looking at the NVMe drive I/O latencies. In Example 2-4, 192 (512 byte) sectors were read in 125 microseconds.

*Example 2-4 192 (512 byte) sectors were read in 125 microseconds*

I/O start time	RW	Buf type	disk:sectorNum	nSec	time ms	tag1	tag2	Disk UID typ	NSD node	context	thread [...]
19:07:22.558042	R	data	19:7219193624	192	0.125	83733675	103	C0A85216:61002B87 pd		Pdisk	NSDThread [...]

To look at the I/O latencies of requests at the NSD layer on the IBM ESS 3500 server, look for *srv* layer I/O times. These times show I/O latencies that account for disk I/O times and NSD processing on the server.

On the IBM ESS 3500, disk I/Os are typically faster than on the non-NVMe based ESS models. This means that for the IBM ESS 3500, the ratio of time that is spent in remote procedure calls (RPCs) relative to the actual disk I/O times tends to be higher. For this reason, systems that support RDMA should enable the **verbsRdmaSend** option, so that RPCs can be handled through low latency RDMA operations.

Example 2-5 shows a 195-microsecond network shared disk (NSD) *srv* layer I/O on an IBM ESS 3500, which corresponds to the previously shown 125-microsecond Pdisk I/O. Example 2-5 also shows 128 sectors that are shown in Example 2-4, as Pdisk layer I/O accounts for additional sectors read for checksum validation.

*Example 2-5 195-microsecond NSD srv layer I/O on an IBM Elastic Storage System 3500*

I/O start time	RW	Buf type	disk:sectorNum	nSec	time ms	tag1	tag2	Disk UID typ	NSD node	context	thread [...]
19:07:22.558003	R	data	2:207962112	128	0.195	83733675	103	C0A85216:61002E71 srv	100.168.85.111	NSDWorker	NSDThread [...]

If an I/O is satisfied from the GNR cache, there is not a corresponding Pdisk level I/O as seen in the Example 2-4 on page 45. Even though the drive accesses on the IBM ESS 3500 are more efficient than comparable drive accesses on non-NVMe devices, the relative benefit of data stored in the GNR disk cache will be lower. However, an improved performance is expected as the elements can be efficiently swapped in and out of the GNR cache.

To see the latency of I/O requests from the client's perspective, look for *cli* I/O times on the client-node in the output of `mmdiag --iohist`. (These times include network processing time and the time that requests wait for processing on the server.) Example 2-6 shows that for the previously shown 128 sector I/O, it took about 575 microseconds from the client's perspective.

Example 2-6 `mmdiag --iohist` output on the client node

I/O start time	RW	Buf type	disk:sectorNum	nSec	time ms	tag1	tag2	Disk UID typ	NSD node	context	thread	[..]
19:07:22.557911	R	data	2:207964928	128	0.575	83733675	103	COA85216:61031045 cli	100.168.82.21	MBHandler	DioHandlerThread	[..]

### IBM Spectrum Scale RAID TRIM support

Optimal *write* performance is achieved when the IBM ESS 3500 NVMe drives are new, or after a purge (NVMe format). Depending upon usage, the write performance can degrade over time as the available free space for internal-drive garbage collection decreases. **TRIM** commands must be issued to an NVMe drive, so that the deleted space is designated as available for garbage collection.

Using the **TRIM** command of IBM Spectrum Scale enables optimal performance by enabling storage controllers to reclaim the free space. The IBM Spectrum Scale `mmreclaimspace` command can be used to send **TRIM** commands to the storage media. The **TRIM** feature must be enabled at the physical disk level, within a declustered array, and at the file system NSD level.

The amount of time for the `mmreclaimspace` command to complete space reclamation depends on the total space to be reclaimed. The amount of time that is needed increases linearly as the amount of space to reclaim increases. After IBM Spectrum Scale RAID issues the **TRIM** command to the NVMe devices, the actual discard operation on the NVMe devices can continue running asynchronously in the background even after `mmreclaimspace` command returns a result. The write performance is fully restored after the NVMe device background activity is completed.

Automatic background space reclamation can be set through the `mmchconfig` attribute `backgroundSpaceReclaimThreshold` to specify the percentage of reclaimable blocks that must occur in an allocation space for devices capable of space reclaim. If set to a smaller value, the free space is reclaimed frequently.

The space reclamation, background or manual, affects write workloads that allocate new blocks from the file system. If such workloads are the primary use case or occur at unscheduled times, then the automatic background file system **TRIM** might not be the best fit for such an environment.

For more information, see [Managing TRIM support for storage space reclamation](#).

### 2.5.3 Shared recovery group

The IBM ESS 3500 uses a shared recovery group layout, where a single recovery group is defined and shared by both canisters. The NVMe drives, in performance and hybrid models, are concurrently accessed by both canisters as was the case with previous shared recovery group-based models, the IBM ESS 3200 and IBM ESS 3000.



However, the HDD enclosures in capacity and hybrid models are divided into a pair of declustered arrays (DAs) within the single recovery group, with each canister taking exclusive primary responsibility for one DA of the pair.

For more information, see [Recovery Group Issues](#).

## 2.5.4 Tuning

IBM ESS 3500 configuration parameters are set automatically for optimal performance during installation. This section describes the key configuration parameters on the I/O servers and client nodes that can be further modified to achieve optimal performance. Modifications are based on the nature of the I/O activities of the application.

### I/O server tuning

Choosing the right file system block size influences the subblock sizes that would be set for both data and metadata blocks. This section explains block size settings and the procedures to verify the IBM ESS 3500 configuration parameters.

#### ***Larger data block sizes***

Larger data block sizes traditionally help large sequential streaming I/O workloads. However, storage systems that use erasure encoding might experience a write-amplification effect when the amount of data that is written is smaller than the file system block size and when those writes are not coalesced. This can have a negative performance impact. On such storage systems, workloads for which small write-performance is an important component might see a performance improvement if the file system block-size is optimized to minimize write-amplification.

#### ***Subblock sizes***

IBM Spectrum Scale Version 5 introduced variable subblock sizes, making space allocations for smaller files more efficient with larger block sizes, and improving file creation and block allocation times. With variable subblock sizes, it is advised to avoid using different block sizes for data and metadata within the same file system. Setting metadata block size that is smaller than the data block size results in a larger subblock for user storage pools. This causes block-allocation time to become longer when compared to the case where the block size for metadata and data blocks is the same.

See the descriptions of the `-B BlockSize` parameter and the `-metadata-block-size MetaDataBlockSize` option in the help topic `mmcrfs` command. For more information, see [Block Size](#).

### Verification of server configuration

IBM ESS performance also depends on the correct IBM Spectrum Scale RAID configuration, operating system, and network tuning. The IBM ESS tuning parameters are automatically configured during the file system creation by using the `essrun` command or by directly running the `mmvdisk server configure` command before creating IBM Spectrum Scale RAID recovery group. The `essrun` command is also used for deployment and cluster creation. When debugging performance issues, verify that the correct and intended configuration parameters are in place.

The tuning can be verified by using the following methods:

- ▶ The `essinstallcheck` command checks various aspects of the installation along with the IBM Spectrum Scale RAID configuration settings and tuned profile. For more information

about how to run this command, see the [essinstallcheck command](#). Review the output carefully to address any issues.

- ▶ The IBM Spectrum Scale RAID configuration values can also be checked by using **mmvdisk server configure --verify** option.

The **--verify** option checks whether the IBM Spectrum Scale RAID configuration attributes for the node class are set to the expected values by checking the real memory and server disk topology for each of the nodes in the node class. The **--verify** option can also be used to check whether the IBM Spectrum Scale RAID has newer, best-practice configuration values applied.

The **mmvdisk server configure --update** command can be used to apply newer, best-practice configuration values or reset the node class to the intended default configuration values.

Example 2-7 shows an example of the **mmvdisk server configure** command with the **--verify** option and its output.

*Example 2-7 The mmvdisk server configure command run with the --verify option*

---

```
# mmvdisk server configure --verify --node-class
ess3500_mmvdisk_ess3500a4_hs_ess3500b4_hs
mmvdisk: Checking resources for specified nodes.
mmvdisk: Node class 'ess3500_mmvdisk_ess3500a4_hs_ess3500b4_hs' has a shared
recovery group disk topology.
mmvdisk: Node class 'ess3500_mmvdisk_ess3500a4_hs_ess3500b4_hs' has server disk
topology 'ESS 3500 FN2 24 NVMe'.
mmvdisk: Node class 'ess3500_mmvdisk_ess3500a4_hs_ess3500b4_hs' uses
'ess3500.shared' recovery group configuration.
```

daemon configuration attribute	expected value	configured value
pagepool	162029432832	as expected
nsdRAIDTracks	128K	as expected
nsdRAIDBufferPoolSizePct	80	as expected
nsdRAIDNonStealableBufPct	50	as expected
pagepoolMaxPhysMemPct	90	as expected
nspdBufferMemPerQueue	24m	as expected
nspdQueues	120	as expected
nspdThreadsPerQueue	2	as expected
nsdRAIDBlockDeviceMaxSectorsKB	0	as expected
nsdRAIDBlockDeviceNrRequests	0	as expected
nsdRAIDBlockDeviceQueueDepth	0	as expected
nsdRAIDBlockDeviceScheduler	off	as expected
nsdRAIDDefaultGeneratedFD	no	as expected
nsdRAIDDiskDiagTimeout	130	as expected
nsdRAIDEnableRGCMRebalanceWeight	yes	as expected
nsdRAIDEventLogToConsole	all	as expected
nsdRAIDMasterBufferPoolSize	2G	as expected
nsdRAIDReconstructAggressiveness	0	as expected
nsdRAIDSmallThreadRatio	2	as expected
nsdRAIDSSDPerformanceShortTimeConstant	2500000	as expected
nsdRAIDThreadsPerQueue	16	as expected
ignorePrefetchLUNCount	yes	as expected
maxFilesToCache	128k	as expected
maxMBpS	50000	as expected
maxStatCache	128k	as expected

nsdMaxWorkerThreads	3842	as expected
nsdMinWorkerThreads	3842	as expected
nsdSmallThreadRatio	1	as expected
numaMemoryInterleave	yes	as expected
panicOnIOHang	yes	as expected
pitWorkerThreadsPerNode	32	as expected
prefetchPct	50	as expected
workerThreads	1024	as expected

mmvdisk: All configuration attribute values are as expected or customized.

---

- ▶ The tuned profile should be set to *scale* automatically after ESS deployment. Check the current active profile by using the **tuned-adm active** command, as shown in Example 2-8.

*Example 2-8 The tuned-adm active command*

---

```
# tuned-adm active
Current active profile: scale
```

---

- ▶ If tuned profile is not set to *scale*, modify using **tuned-adm profile** as shown in Example 2-9.

*Example 2-9 The tuned-adm profile command*

---

```
# tuned-adm profile scale
```

---

- ▶ The system settings can be verified against current profile by using the **tuned-adm verify** command, as shown in Example 2-10.

*Example 2-10 The tuned-adm verify command*

---

```
# tuned-adm verify
Verification succeeded, current system settings match the preset profile. See
tuned log file ('/var/log/tuned/tuned.log') for details.
```

---

## Client tuning

After the client cluster is created in the installation phase, extra networking and performance settings can be applied. The **gssClientConfig.sh** script can be used to apply basic best-practice settings for your client NSD cluster. Running this script with the **-D** option shows the configuration settings that it intends to set without setting them.

**Note:** The `/usr/lpp/mmfs/samples/gss/gssClientConfig.sh` script is part of `gpfs.gnr` installation package. Typically, the `gpfs.gnr` package is not installed on clients. Hence, you must manually copy the script to the client cluster.

Also, this script attempts to configure the client nodes for RDMA access by setting the following **mmchconfig** parameter values if applicable:

- ▶ **verbsRdma** *enable*
- ▶ **verbsRdmaSend** *yes*
- ▶ **verbsPorts** *<Infiniband ports>*

## Pagepool

Client-side **pagepool** configuration influences the end-to-end performance of applications.

Pagepool defines the amount of memory that is used for caching file system data and metadata. Pagepool is also used in some non-caching operations, such as buffers allocated for encryption and DMA transfers for DIO data.

A *pagepool* is a pinned memory region that cannot be swapped out to disk, that is, IBM Spectrum Scale always uses at least the value of the **pagepool** attribute in the system memory. Users need to consider the memory requirements of other applications that are running on the node when determining a value for the **pagepool** attribute.

For best sequential performance, tune the **pagepool** attribute. Increasing **pagepool** beyond this value is most beneficial for workloads (non-direct I/O) that re-read the same data because more data can be cached in the pagepool.

Use the `-P` option of the `gssClientConfig.sh` script to set the `pagepool` value:

```
#gssClientConfig.sh -P <size in MiB> <node names>
```

## Information lifecycle management

The IBM Spectrum Scale Information Lifecycle Management (ILM) feature uses the powerful IBM Spectrum Scale policy engine to achieve efficient, policy-driven, automated tiered storage management.

In an IBM ESS 3500 hybrid deployment model, the overall performance of a particular workload depends on the amount of data that is stored on NVMe disk and the amount of data stored on hard disk. The ILM toolkit can help you establish efficient data placement that achieves optimal use of the storage devices available on your system. Using the ILM toolkit, IBM Spectrum Scale can automatically determine where to physically store the data, regardless of its placement in the logical directory structure. Improved price-performance ratio is achieved by determining the cost of the storage compared to the preferred response time of the storage.

The ILM based tools allow the creation of storage pools, which define tiers of storage, grouping storage devices based on performance, locality, or reliability characteristics.

Establishing tiers allows for the optimal price-performance ratio in the usage of your storage by considering the cost of storage and the performance requirements of the storage.

Optimal price-performance ratio is obtained by considering the following items:

- ▶ Ensuring that premium storage is prioritized for performance sensitive data
- ▶ Assigning slower, less costly, storage to data that is less performance sensitive
- ▶ Optionally integrating external storage pools into the storage lifecycle management flow, for example, storage assigned for backup or archiving purposes through an external application such as IBM Spectrum Protect

For more information, see [Information lifecycle management](#).

## Quality of service for I/O operations

With the IBM Spectrum Scale Quality of Service (QoS) feature, you can limit the bandwidth and IOPS of various workloads, so you can minimize the potential for different workloads interfering with each other.

Here are example scenarios to illustrate the benefits of employing QoS to limit the interference of workloads to applications:

- ▶ QoS can be used to limit potential interference effects of I/O-intensive IBM Spectrum Scale maintenance commands on application system I/O performance. For example, the administrator can use the QoS feature to limit the performance impact of a long running **mmrestripe rebalance** operation on user workloads.
- ▶ By assigning work on a project and imposing I/O limits on specific file sets, you can use QoS to limit the interference between different projects. For example, if project work is assigned to a file set, FS01, then a read/write operations per second (IOPS) limit or bandwidth limit (MBps) can be placed on I/O to the FS01 file set, which limits the interference that users of project FS01 can have on other users of the file system.

You can display regular or fine-grained statistics of I/O accesses by processes over time. The **mmqos** command provides all the features that were introduced with the **mmchqos** and **mm1sqos** commands, supports QoS user classes, and has an easy-to-use command syntax.

For more information, see [Setting QoS](#).

## 2.6 IBM Spectrum Scale Multi-Rail over TCP and RDMA over Converged Ethernet

Multi-Rail Over TCP (MROT) enables the concurrent use of multiple subnets to communicate with a specified destination, and now allows the concurrent use of multiple physical network interfaces without requiring bonding to be configured.

### MROT features

Starting with IBM Spectrum Scale 5.1.1, it is now possible to establish communications to a target with multiple TCP connections from the source that have the same daemon IP address. IBM Spectrum Scale 5.1.5 introduces the MROT feature that enables the concurrent use of multiple subnets to communicate with a specified destination and now allows the concurrent use of multiple physical network interfaces without requiring bonding to be configured. The number of connections is controlled through the **maxTcpConnsPerNodeConn** parameter, which can be changed by using the **mmchconfig** command.

With MROT, IEE Link aggregation is not required. MROT provides fault tolerance, recovery, and load balancing between all network interfaces of the communicating nodes.

Also, MROT provides benefits to IBM Spectrum Scale clusters when Remote Direct Memory Access (RDMA) is enabled. RDMA can now be configured over Ethernet (through RDMA over Converged Ethernet (RoCE)) and with MROT, it is no longer mandatory to assign IP addresses over different subnets as it was with IBM Spectrum Scale 5.1.4 and earlier.

All IP interfaces can now be configured with the same subnet.

At the time of writing, information documented in [Highly Efficient Data Access with RoCE on IBM Elastic Storage Systems and IBM Spectrum Scale, REDP-5658](#) is not 100% accurate and will be updated in the future. Previous releases did not support more than one IP interface on the subnet in which the IP address of the **mmfs** daemon is configured. Any other high-speed interface with an IP address, for example, RoCE or InfiniBand Connected Mode, had to be in another subnet different from the **mmfsd**. This is no longer a requirement.

## Network topology

The term “host” in this section is used for an endpoint in the network. It can be an IBM Spectrum Scale client machine or an ESS and NSD server machine.

Configuration of a host depends on the number of network ports needed. The number of network ports a node should use to connect with the network depends on the expected bandwidth; the nominal speed and bandwidth of the network; and the number of different networks it needs to access. High availability requirements can also generate the need for more than one network interface per node.

Today in TCP environments, bandwidth scaling and high availability are commonly implemented by bonding network ports, which is known as link aggregation. However, the effects of bonding might require additional configuration of the network, switches, and so on. Using bonds for scale out over multiple network adapters can make a deployment complex.

The need to use IEEE link aggregation to scale over more than one network interface is no longer required with IBM Spectrum Scale. GPFS can be configured to use multiple interfaces, without the need to have bonding or link aggregation configured.

Introducing MROT to IBM Spectrum Scale automatically simplifies any other IP address configuration and also allows RDMA configuration scenarios to benefit from MROT.

Figure 2-34 shows an example of best practice for network topology

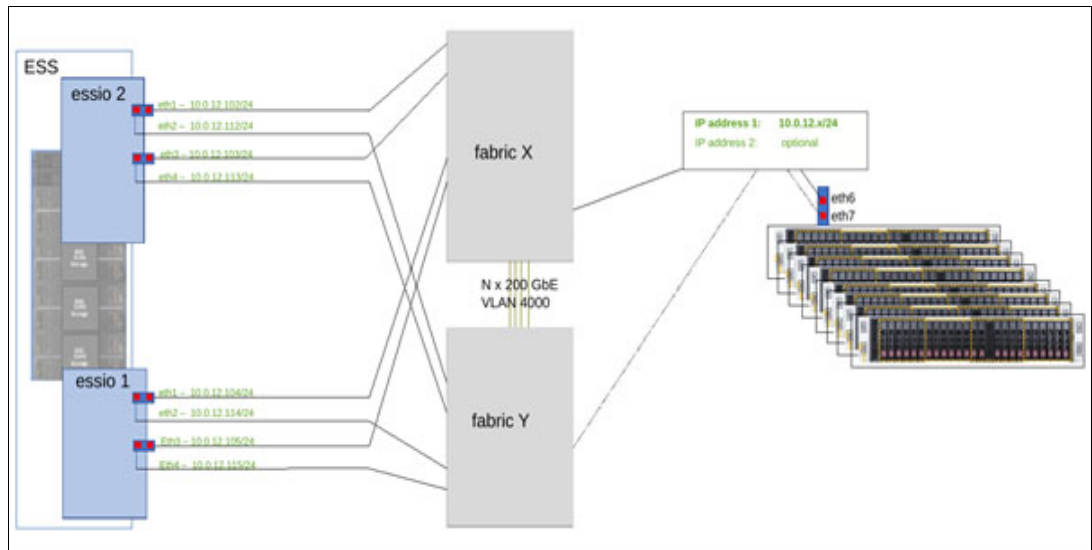


Figure 2-34 Network Topology with MROT for mmfsd communication

In cases where bonding and configuring link aggregation is needed, it is still supported with IBM GPFS.

The use of MROT can help to simplify deployments by using MROT features of load balancing and HA protection for connectivity.

**Note:** The adapters in a RoCE enabled environment can run TCP IP traffic and RDMA traffic simultaneously.

### Connecting more than one IP interface to the same subnet

A limitation of the Linux kernel is that configuring multiple interfaces to the same subnets is a challenge. Because of ARP requests, selecting rules for outgoing IP traffic when you have more than one interface per subnet becomes mandatory. More than one interface pointing into the same subnet creates a need to define a *source based routing* configuration for those interfaces.

The `mmchconfig subnets` attribute establishes fault tolerance or automatic failover. All the IP addresses that are defined in the `subnets` attribute are used to establish connections with the other IBM Spectrum Scale nodes and helps establish N:N or M:N connection models between source and destination.

Figure 2-35 shows configuring the N:N connection model (`maxTcpConnsPerNodeConn = 2`, `subnets = "192.168.1.0 192.168.2.0"`).

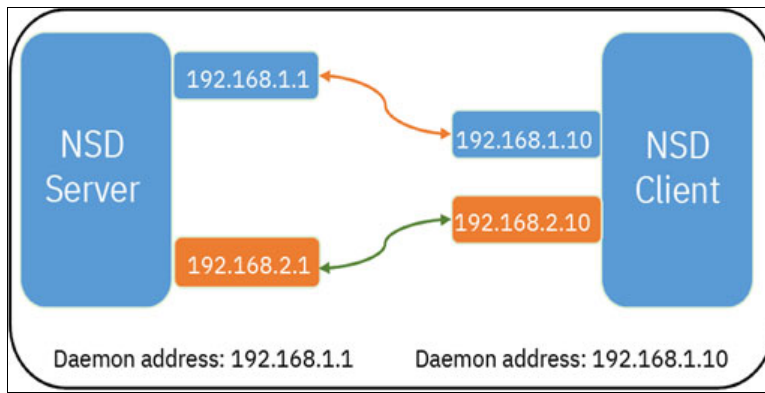


Figure 2-35 Example of N:N connection model configuration

Figure 2-36 shows configuring the M:N connection model (`maxTcpConnsPerNodeConn = 4`, `subnets = "192.168.1.0"`).

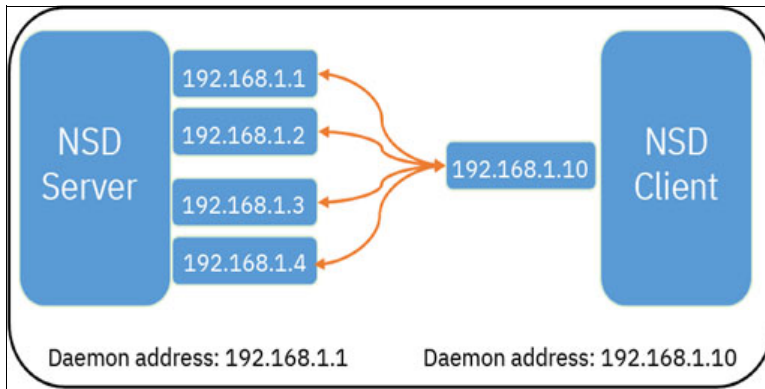


Figure 2-36 Example of M:N connection model configuration



The number of connections are controlled through the `mmchconfig` parameter `maxTcpConnsPerNodeConn`. Valid values are 1 through 16, with the default of 2.

Consider the following factors when assigning a value to `maxTcpConnsPerNodeConn`:

- ▶ The overall bandwidth of the cluster network
- ▶ The number of nodes in the cluster
- ▶ The value that is configured for the `maxReceiverThreads` parameter
- ▶ Memory resource implications of setting a higher value for the `maxTcpConnsPerNodeConn` parameter

For more information, see [Configuring MROT](#).

Figure 2-34 on page 53 depicts configuring source based routing for all four interfaces. The following example shows commands that define a boot resilient configuration. These steps must be repeated for each node.

*Example 2-11 Commands to define a boot resilient configuration*

---

```
nmcli con add type 802-3-ethernet ifname eth1 connection.interface-name eth1
connection.id eth1
nmcli con mod eth1 ipv4.addresses 10.0.12.102/24
nmcli con mod eth1 ipv4.method static
nmcli con mod eth1 connection.autoconnect yes
nmcli con mod eth1 802-3-ethernet.mtu 9000
nmcli con mod eth1 ipv6.addr-gen-mode eui64
ifup eth1
```

```
ifdown eth2
nmcli con del eth2
nmcli con add type 802-3-ethernet ifname eth2 connection.interface-name eth2
connection.id eth2
nmcli con mod eth2 ipv4.addresses 10.0.12.112/24,192.168.12.102/24
nmcli con mod eth2 ipv4.method static
nmcli con mod eth2 connection.autoconnect yes
nmcli con mod eth2 802-3-ethernet.mtu 9000
nmcli con mod eth2 ipv6.addr-gen-mode eui64
ifup eth2
```

```
ifdown eth3
nmcli con del eth3
nmcli con add type 802-3-ethernet ifname eth3 connection.interface-name eth3
connection.id eth3
nmcli con mod eth3 ipv4.addresses 10.0.12.103/24
nmcli con mod eth3 ipv4.method static
nmcli con mod eth3 connection.autoconnect yes
nmcli con mod eth3 802-3-ethernet.mtu 9000
nmcli con mod eth3 ipv6.addr-gen-mode eui64
ifup eth3
```

```
ifdown eth4
nmcli con del eth4
nmcli con add type 802-3-ethernet ifname eth4 connection.interface-name eth4
connection.id eth4
nmcli con mod eth4 ipv4.addresses 10.0.12.113/24
nmcli con mod eth4 ipv4.method static
```

```

nmcli con mod eth4 connection.autoconnect yes
nmcli con mod eth4 802-3-ethernet.mtu 9000
nmcli con mod eth4 ipv6.addr-gen-mode eui64
ifup eth4

#echo "101 t1" >> /etc/iproute2/rt_tables
#echo "102 t2" >> /etc/iproute2/rt_tables
#echo "103 t3" >> /etc/iproute2/rt_tables
#echo "104 t4" >> /etc/iproute2/rt_tables

nmcli con modify eth1 +ipv4.routes "0.0.0.0/1 10.0.12.254 table=101, 128.0.0.0/1
10.0.12.254 table=101"
nmcli con modify eth1 +ipv4.routes "10.0.12.0/24 table=101 src=10.0.12.102"
nmcli con modify eth1 +ipv4.routing-rules "priority 32761 from 10.0.12.102 table
101"
ifdown eth1; ifup eth1

nmcli con modify eth2 +ipv4.routes "0.0.0.0/1 10.0.12.254 table=102, 128.0.0.0/1
10.0.12.254 table=102"
nmcli con modify eth2 +ipv4.routes "10.0.12.0/24 table=102 src=10.0.12.112"
nmcli con modify eth2 +ipv4.routing-rules "priority 32761 from 10.0.12.112 table
102"
ifdown eth2; ifup eth2

#
nmcli con modify eth3 +ipv4.routes "0.0.0.0/1 10.0.12.254 table=103, 128.0.0.0/1
10.0.12.254 table=103"
nmcli con modify eth3 +ipv4.routes "10.0.12.0/24 table=103 src=10.0.12.103"
nmcli con modify eth3 +ipv4.routing-rules "priority 32761 from 10.0.12.103 table
103"
ifdown eth3; ifup eth3

nmcli con modify eth4 +ipv4.routes "0.0.0.0/1 10.0.12.254 table=104, 128.0.0.0/1
10.0.12.254 table=104"
nmcli con modify eth4 +ipv4.routes "10.0.12.0/24 table=104 src=10.0.12.113"
nmcli con modify eth4 +ipv4.routing-rules "priority 32761 from 10.0.12.113 table
104"
ifdown eth4; ifup eth4

```

---

Additionally, some kernel parameters will need to be set by using `sysctl`.

*Example 2-12 Setting additional kernel parameters*

---

```

sysctl -w net.ipv4.conf.all.arp_ignore=2
sysctl -w net.ipv4.conf.default.arp_ignore=2
sysctl -w net.ipv4.conf.all.arp_announce=1
sysctl -w net.ipv4.conf.default.arp_announce=1
sysctl -w net.ipv4.conf.all.rp_filter=2
sysctl -w net.ipv4.conf.default.rp_filter=2
sysctl -w net.ipv4.conf.default.arp_filter=1
sysctl -w net.ipv4.conf.all.arp_filter=1

```

---

**Note:** Additional entries are required in your customized *tuned.conf* file, or other mechanisms must be used to make the settings boot resilient.

## Additional steps, when using RoCE

In addition to the multiple rail and multiple socket support, where RDMA is used it is possible to generate even greater performance with lower system utilization. RoCE is fully supported by IBM Spectrum Scale and commonly used as part of a best practice deployment.

As a rule, the system utilization for handling a 10 Gbps data stream by using TCP/IP rather than using RDMA requires roughly, the additional use of 5 - 8 CPU cores.

When using RDMA with IBM Spectrum Scale, the following items are mandatory to configure on the machines.

- ▶ The Ethernet fabric must support ECN1 and PFC2
- ▶ The host's interfaces for ECN and PFC must be enabled.

### *Example 2-13 Configuring Mellanox switch PFC*

---

```
mlnx_qos -i [network-Interface] --trust dscp  
mlnx_qos -i [network-Interface] --pfc 0,0,0,1,0,0,0,0c  
ma_roce_tos -d [mlx5_x] -t 106
```

---

Configure IBM Spectrum Scale to use RDMA as shown in Example 2-14.

### *Example 2-14 Configuring IBM Spectrum Scale to use RDMA*

---

```
verbsRdmaCm enable  
verbsNumaAffinity enable  
verbsPorts mlx5_0/1/3 mlx5_1/1/4 mlx5_2/1/3 mlx5_3/1/4  
verbsRdmaSend yes  
verbsRdma enable
```

---

These configuration steps are documented in [Highly Efficient Data Access with RoCE on IBM Elastic Storage Systems and IBM Spectrum Scale, REDP-5658](#).

For more information, see [Configuring Multi-Rail over TCP \(MROT\)](#).





## Planning considerations

This chapter provides planning information specific to deploying the IBM Elastic Storage System (ESS) 3500 hardware, software, networking, and ESS Management Server (EMS). Guidance is also provided on required skills and recommendations for services that you might want to consider. It covers the following topics:

- ▶ 3.1, “Planning” on page 60
- ▶ 3.2, “Standalone environment” on page 67
- ▶ 3.3, “Mixed environment” on page 68

## 3.1 Planning

This section provides planning information specific to deploying the IBM ESS 3500 hardware, software, networking, and EMS. Guidance is also provided on required skills and recommendations for services that you might want to consider.

### 3.1.1 Technical and delivery assessment

When you order an IBM ESS 3500, certain functional and non-functional requirements need to be fulfilled before the configuration can be created and the order can be entered.

A technical and delivery assessment (TDA) is an internal IBM process that includes a technical inspection of a completed solution design. This process assures customer satisfaction and ensures a smooth and timely installation. Technical subject matter experts (SMEs) who were not involved in the solution design participate to answer the following questions:

- ▶ Will the IBM ESS 3500 solution work?
- ▶ Is the implementation and plan sound?
- ▶ Will it meet customer requirements and expectations?

TDAs are necessary for every ESS order. For the systems service representative (SSR) to begin installation, the TDA procedure must be performed to complete the installation worksheet. The worksheet outlines the items that must be implemented by the SSR during setup.

Include the following information on the TDA worksheet:

- ▶ Management IP address and netmask
- ▶ Baseboard Management Control (BMC) IP address and netmask
- ▶ VLAN tag
- ▶ Root password

The two TDA processes are described in the following list:

1. The pre-sales TDA. This is done by IBMers or IBM Business Partners by using the [file and object solution design engine \(FOSde\)](#) tool.
2. The pre-install TDA. SMEs also evaluate the customer's readiness to install, implement, and support the proposed solution. This can be done with the [IMPACT tool](#).

The two TDA processes have assessment questions, but also baseline benchmarks that need to be performed before the order can be fulfilled. Those tools are run by IBM sales or resellers, so they can help and direct you regarding this process.

### 3.1.2 Hardware planning

These requirements include the hardware solution components that are mandatory but are not included in the IBM ESS 3500 building block (2U). Two types of these requirements are described in the following list:

1. The requirements that must be IBM-provided, such as the management switch
2. The requirements that can be either customer-provided or IBM-provided, such as the HS network or rack

## Rack solution

The IBM ESS 3500 comes with at least one rack from IBM; it can come with more if multiple building blocks (BB) are ordered. The rack also holds the EMS and the management switch. If the HS switches are ordered from IBM, those are also included in the rack.

Although the preferred option is the rack version of the IBM ESS 3500 solution, it is possible to order the IBM ESS 3500 without the rack. If you choose to follow this path, you must verify that the rack can hold the weight of the solution, and that the power distribution units (PDUs) on the rack are the right ones for the IBM ESS 3500 solution. In addition, you must contact IBM to configure the management switch that comes with the solution.

## Management switch

The IBM ESS 3500 first building block on a site includes a 1GbE management switch from IBM (8831-S52). This switch is part of the IBM ESS 3500 solution, and it is not an independent part that can be replaced with equivalent hardware by the customer.

The deployment configuration of the 1 GbE management switches have a specific configuration where ports 1 - 12 are “ESS 3500” ports as shown in Figure 3-1.

Figure 3-1 is an example of the 1GbE Management Switch deployment. Deployment specifics can change from release to release. Always check the ESS Quick Deployment Guide for the information for your ESS implementation. At the time of this writing, you can consult the IBM ESS Version 6.1.5 [ESS Deployment Guide](#).

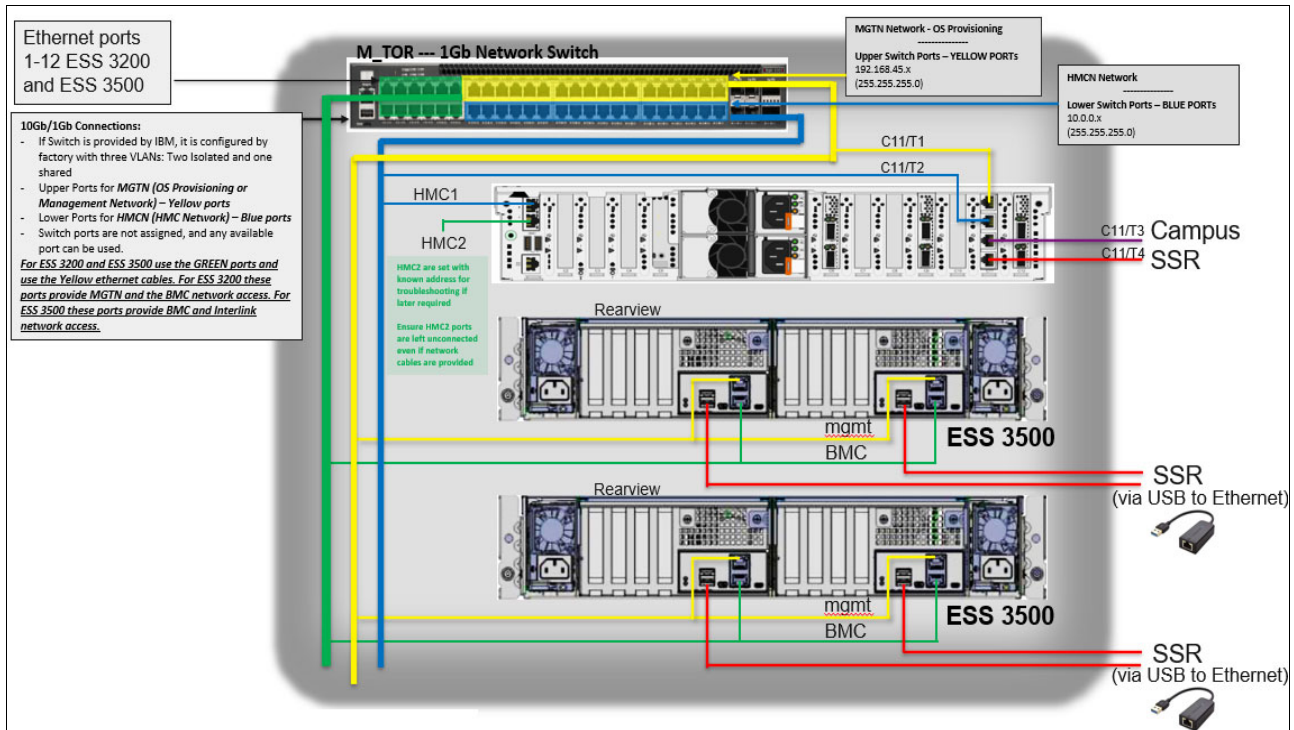


Figure 3-1 Overview of the management network and ports

All the IBM ESS 3500 server canisters must be connected to ports 1 - 12 only. In case you have already a management switch that was ordered before you ordered the IBM ESS 3500 and does not have the new management switch configuration (v2), you can ask IBM TSS to convert the switch to be usable by IBM ESS 3500 systems. You can also convert the switch yourself with the instructions in Appendix A, “Configuring the 48 ports top of the rack management network switch” on page 103.

## Dual 24 port (48 ports) management switch

When the management TOR is included in an order, you will receive two of the 24 port management switches. The reason to deliver two instead of one is to keep similar number of ports available as with the 48 ports switch option. The switches appear as seen in Figure 3-2.

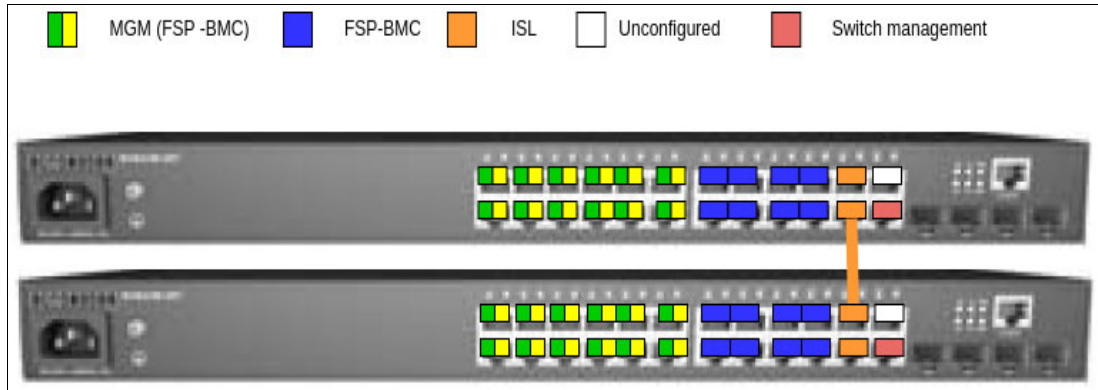


Figure 3-2 Dual 24 port management switch

The orange-colored cable shown in Figure 3-2 must be connected between port 22 of the upper switch and 21 of the lower switch as part of the configuration. That cable works as inter-switch link (ISL) between the two switches.

## High-speed network

As with any other IBM Spectrum Scale and IBM ESS configurations, the IBM ESS 3500 requires a high-speed (HS) network to be used as the data-storage cluster network. In some product documentation, this network is referred to as a clustering network. The hardware for the HS network can be provided by IBM or by the customer. If the hardware is provided by the customer, it must be compatible with the network interfaces that the IBM ESS 3500 supports. See 2.1.1, “Canisters and servers” to see the available network options on IBM ESS 3500.

## ESS Management Server

The IBM ESS 3500 requires an POWER9 ESS Management Server (EMS), IBM machine type 5105-22E. The EMS is required on standalone installs. If the IBM ESS 3500 is added to an existing IBM ESS IBM Spectrum Scale cluster that already has a POWER9 EMS, then the existing EMS can support the IBM ESS 3500. The minimum release level on the IBM ESS 3500 canisters is ESS 6.1.5.

For more details about the EMS server, see [IBM Documentation for 5102-22E](#).

**Notes:** If your previously-installed IBM Spectrum Scale or ESS configuration uses a previous-generation POWER8 EMS, you must add a POWER9 EMS to support the IBM ESS 3500.

IBM does not support re-purposing an existing server or a VM or LPAR to be used as the EMS.

The IBM or IBM Business Partner team uses the FOSde tool and IBM eConfig for Cloud to configure the EMS. IBM eConfig for Cloud configures the EMS with the appropriate network cards such that the EMS can connect to the same HS networks that are configured on the IBM ESS 3500.



The default IBM ESS Management Server memory size is enough for most IBM ESS installations. If many IBM ESSs are used in your IBM Spectrum Scale IBM ESS configuration, check with your IBM representative to see whether larger IBM ESS Management Server memory sizes might be required for your installation. More EMS memory can be specified at order time or added later as a field Miscellaneous Equipment Specification (MES).

### **Best practices for deploying and optimizing a stand-alone IBM ESS 3500**

A standalone IBM ESS 3500 unit, which is known as a building block, must minimally consist of the following components:

- ▶ One EMS node in a 2U form factor
- ▶ One IBM ESS 3500 node in 2U form factor
- ▶ 1 GbE Network switch for management network (1U)
- ▶ 100 or 200 Gb high-speed IB or Ethernet network for internode communication (1U)

The EMS node acts as the administrative end point for your IBM ESS 3500 environment. It performs the following functions:

- ▶ Hosts the IBM Spectrum Scale GUI
- ▶ Hosts Call Home services
- ▶ Hosts system health and monitoring tools.
- ▶ Manages cluster configuration, file system creation, and software updates
- ▶ Acts as a cluster quorum node

The IBM ESS 3500 features a new container-based deployment model that focuses on ease-of-use. The container runs on the EMS node. All of the configurations tasks that were performed by the `essutils` utility in legacy ESS are now implemented as Ansible Playbooks that are run inside of the container. These playbooks are accessed using the `essrun` command.

The `essrun` tool handles almost the entire deployment process, and is used to install software, apply updates, and deploy the cluster and file system. Only minimum initial user input is required, and most of that is provided during the TDA process before setting up the system. The `essrun` tool automatically defines system parameters to optimize performance of the single IBM ESS 3500 system. File system parameters and IBM Spectrum Scale RAID Erasure code selection can be customized from their defaults before file system creation.

For more information about deployment customization, see the [ESS Quick Deployment Guide](#).

These are some of the best practices:

- ▶ Refrain from running admin commands directly on the IBM ESS 3500 I/O canisters. Use the EMS node instead.
- ▶ Do not mount the file system on the IBM ESS 3500 I/O canisters because this consumes additional resources. The file system must be mounted on the EMS node for the GUI to function properly.

- ▶ To access the file system managed by the IBM ESS 3500 building block, you must use external GPFS client nodes or protocol nodes.
- ▶ On a single building block deployment, the I/O canister nodes are specified as GPFS cluster or file system manager nodes while the EMS node is not. Although the EMS node is considered the building block's primary management server, avoid specifying the EMS node as a manager node. The GPFS management role is an internal designation that was previously the manager of the cluster and the file system and does not directly affect the function of the EMS node.

### 3.1.3 Software planning

The IBM ESS 3500 provides an integrated, tested ESS solution software stack that includes the following features:

- ▶ Embedded Red Hat Enterprise Linux license
- ▶ Firmware drivers for the Mellanox network cards
- ▶ IBM Spectrum Scale
- ▶ All necessary supporting software

IBM supports the ESS software stack as a solution.

When installing ESS software updates, each installation cannot be done in the same way. Each installation has different operational and non-operational requirements that can impact what is possible to achieve and when and how often is possible to do software updates.

Ideally, you would update systems at least once a year, but for some installations, annual updates are not possible because of legal certification reasons, operational, or other reasons. Consequently, updates might occur once every three or more years.

IBM strongly recommends that the following key points are followed when doing software currency on ESS-related environments:

- ▶ Never do more than N-3 jump of an ESS-software update. Do intermediate jumps, if needed, to maintain this rule.
- ▶ Always update the EMS first.
- ▶ Perform offline updates when possible. If online update is a requirement, explore the `-serial` option to limit the risk exposure in case some nodes experience problems during the update.
- ▶ If you encounter a problem, contact IBM Support. Resolving the problem without the help of support might fix the problem in the near term, but could create future issues due to the automation expecting the configuration to be a certain way. So, we recommend that you stabilize the environment and then contact IBM Support.
- ▶ Always keep the ESS cluster in the same level. You can update different systems separately, but all should be running the same version after the updates complete. If that is not possible, consider partitioning your backend cluster to achieve this rule.
- ▶ Use defaults, unless you have a specific reason to not do so.

## 3.1.4 Network planning

The ESS system includes certain network names. To avoid confusion, those networks are defined in this section.

### Management network

The *management network* is a non-routable private network. It connects the EMS PCI card slot 11 (C11) port 4 (T1), which acts as a DHCP server to all I/O nodes on C11 T1.

Through this network, EMS, and containers on the EMS, manage the OS of the I/O nodes. This network cannot use VLAN tagging of any kind, so it must be configured as an access VLAN on the switch. You can choose any netblock that fits your needs, but as a best practice use a /24 block. If you have no preference, use the 192.168.45.0/24 block because it is the one that is used in most of the documentation examples.

### Flexible service processor network

The *flexible service processor (FSP) network* is a non-routable private network. It connects the EMS C11 T2 with each out-of-band management ports of the I/O nodes that are labeled as "HMC 1". That includes the EMS that has a connection to the HMC1 from this network and any other ESS node running on an IBM POWER platform in the cluster managed by the same EMS.

The EMS and the containers running on the EMS use this network to do FSP operations and BMC operations on the physical servers, which include powering-on and powering-off the servers. This network uses VLAN tagging at the IBM ESS 3500 ports and no tagging on the rest of ports. You can choose any netblock that fits your needs, but as a best practice use a /24 block. If you do not have a preference, use 172.16.0.0/24 for the same reasons as described for the management network.

### Campus network

The campus network is also termed as public or external network that connects to C11-T3 on the EMS node. This connection serves as a way to access the GUI or the ESA agent (call home) from outside of the management network. The container creates a bridge to the management network, thus having a campus connection is highly advised.

POWER9 EMS campus connection must be set prior to deployment (C11-T3). This allows remote access to the EMS and ensures you will not lose a connection when starting the container. Optionally, space is also allocated to set a campus connection on the HMC2 port. This will allow remote access to the FSP which aids the recovery of the node (console/power control) in case of an outage.

### High-speed network

The HS data network is where the IBM Spectrum Scale daemon and admin networks should be configured. It is a customer-provided and customer-managed network.

Network design for a parallel file system can be complex. The HS network design and implementation is usually the deciding factor on what the overall performance your system delivers. Unfortunately, there is no single design that fits every use case.

Here are some design ideas that must be considered:

- ▶ The IBM Spectrum Scale admin network has the following characteristics:
  - Used for the running of administrative commands
  - Requires TCP/IP

- Can be the same network as the IBM Spectrum Scale daemon network or a different one
- Establishes the reliability of IBM Spectrum Scale
- ▶ The IBM Spectrum Scale daemon network has the following characteristics:
  - Used for communication between the mmfsd daemon of all nodes.
  - Requires TCP/IP.
  - In addition to TCP/IP, IBM Spectrum Scale can be optionally configured to use Remote Direct Memory Access (RDMA) for daemon communication. TCP/IP is still required if RDMA is enabled for daemon communication.
  - Establishes the performance of IBM Spectrum Scale, as determined by its bandwidth, latency, and reliability of the IBM Spectrum Scale daemon network.

In cases where the HS data network is Ethernet based, it is a best practice to place the daemon and admin network on the HS Ethernet network.

If you have InfiniBand networks, you can use Ethernet adapters on the HS network if they are available, or you can use an IP over InfiniBand encapsulation.

The Management or FSP network should not be part of the IBM Spectrum Scale cluster as a management or daemon network.

### ***Sizing the network***

With the networking information described in this section, perform the following sizing exercise by considering two parameters:

1. The expected client-required throughput performance and number of client-ports with their aggregated performance
2. The number of IBM ESS 3500 or other ESS I/O nodes or canister aggregated performance.

Include any inter switch links (ISLs) that are in place as well as PCIe speeds and feeds for each system. As an example, consider a simple two HS IB 200 Gbit ports scenario, where each IBM ESS 3500 includes eight ports connected. Assuming there are PCI3 Gen 4 x16 lines on the clients and 200 Gbit IB ports connected (high dynamic range (HDR)), it is an ideal scenario to have up to eight of those clients and four ISLs between switches. If you increase the demand on the network beyond that, then some network oversubscriptions might occur, which could have a negative impact on your workload.

### **IBM SSR network port**

The IBM System Services Representative (SSR) network port on the IBM EMS is on C11 port T4. This port should never be cabled or connected to any switch as it is only for IBM field engineers to use. This port is configured on the 10.111.222.100/30 block.

The ESS canisters do not have direct Ethernet connectivity through the SSR port. The IBM SSR accesses the device through a serial cable on each canister.

**Note:** The management network, flexible processor network and high-speed network are required to be reachable by all nodes in the cluster including the EMS servers. If the systems are located in different racks, rooms or even data centers (called a stretch cluster). All nodes in the stretch cluster must be able to reach these networks from all racks, rooms and data centers to be a supported configuration. It is not supported for a node or EMS in one site to only have access to the nodes of its own site and vice versa.

### 3.1.5 ESS Management Server considerations:

The IBM ESS 3500 requires a POWER9 ESS Management Server (EMS), IBM machine type 5105-22E. The EMS is required on standalone installations. If the IBM ESS 3500 is added to an existing IBM Spectrum Scale or ESS cluster that already has a POWER9 EMS, the existing EMS can support the IBM ESS 3500.

**Note:** If a previously-installed IBM Spectrum Scale or ESS configuration uses a previous-generation IBM POWER8® EMS, you must add an IBM POWER9™ EMS to support the IBM ESS 3500. IBM does not support re-purposing an existing server, VM or LPAR to be used as the EMS.

For more details about the EMS server, see the [Documentation for 5105-22E](#).

IBM or the IBM Business Partner team uses the FOSde tool and IBM eConfig for Cloud to configure the EMS. eConfig configures the EMS with the appropriate network cards such that the EMS can participate in the same high-speed networks that are configured on the IBM ESS 3500.

The default IBM EMS memory size is sufficient for most IBM ESS installations. If a large number of IBM ESS enclosures are used in your IBM Spectrum Scale IBM ESS configuration, then check with your IBM representative to see whether an IBM EMS with more memory might be required for your environment. More EMS memory can be specified at order time or added later, as a field miscellaneous equipment specification (MES).

### 3.1.6 Skills and services

Installation and support of the IBM ESS 3500 requires several skills:

- ▶ Red Hat Enterprise Linux
- ▶ TCP/IP and high-speed networking
- ▶ IBM Spectrum Scale

IBM and IBM Business Partners can provide education courses and services to teach or improve these skills.

Customers and IBM Business Partners also have the ability to engage IBM Systems Lab Services, which are available and recommended, to provide customized help in integrating IBM ESS 3500 into a new or existing client environment.

## 3.2 Standalone environment

This section describes best practices for deploying and optimizing a standalone IBM ESS 3500.

A standalone IBM ESS 3500 unit, which is known as a *building block*, must minimally consist of the following components:

- ▶ One EMS node in a 2U form factor
- ▶ One IBM ESS 3500 node in 2U form factor
- ▶ 1 GbE Network switch for management network (1U)
- ▶ 100 or 200 Gb high-speed IB or Ethernet network for internode communication (1U)

The EMS node acts as the administrative end point for the IBM ESS 3500 environment. It performs the following functions:

- ▶ Hosts the IBM Spectrum Scale GUI
- ▶ Hosts Call-Home services
- ▶ Hosts system health and monitoring tools.
- ▶ Manages cluster configuration, file system creation, and software updates
- ▶ Acts as a cluster quorum node

The IBM ESS 3500 features a container-based deployment model that focuses on ease-of-use. The container runs on the EMS node. All of the configurations tasks that were performed by the `gssutils` utility in legacy ESS are now implemented as Ansible Playbooks that are run inside of the container. These playbooks are accessed using the `essrun` command.

The `essrun` tool handles the majority of the deployment process, and is used to install software, apply updates, and deploy the cluster and file system. Only minimum initial user input is required, most of which is information covered by the TDA process which is required to be completed before setting up the system. The `essrun` tool automatically configures performance-related parameters to get the most out of a single IBM ESS 3500 system. File system parameters and IBM Spectrum Scale RAID Erasure code selection can be customized from their defaults before file system creation.

For more information about deployment customization, see the [Quick Deployment Guide](#).

The following items are considered best practices:

- ▶ Refrain from running admin commands directly on the IBM ESS 3500 I/O canisters. Use the EMS node instead.
- ▶ The file system must be mounted on the EMS node for the GUI to function properly. Do not mount the file system on the IBM ESS 3500 I/O canisters as this consumes additional resources.
- ▶ To access the file system managed by the IBM ESS 3500 building block, you must use external GPFS client nodes or protocol nodes.
- ▶ On a single building block deployment, the I/O canister nodes are specified as GPFS cluster and file system manager nodes while the EMS node is not. Although the EMS node is considered the building block's primary management server, avoid specifying the EMS node as a manager node. The GPFS-management role is an internal designation that, previously, was the manager of the cluster and the file system and does not directly affect the function of the EMS node.

## 3.3 Mixed environment

This section provides information about integrating the IBM ESS 3500 into an existing ESS environment. This includes additional considerations when integrating into a mixed-vendor environment for migration purposes, or for using IBM ESS 3200 as the HS storage tier.

### 3.3.1 Adding the IBM ESS 3500 to an existing ESS cluster

The following guidance is for adding a Standalone IBM ESS 3500 building block into an existing ESS cluster or into an existing IBM ESS 3000, IBM 3200 or IBM ESS 5000.

## Prerequisites and assumptions

There are prerequisites and assumptions for adding an IBM ESS 3500 to an existing ESS cluster:

- ▶ Existing IBM ESS 3000 cluster is connected or reachable to the same HS network block.
- ▶ Existing IBM ESS 3000, IBM ESS 5000, IBM ESS 3200 or IBM ESS 3500 is connected or reachable to the same management low-speed network block.
- ▶ IBM ESS 3000 is configured with a POWER8 or POWER9 EMS node and running Podman container.
- ▶ IBM ESS 5000 contains a POWER9 EMS node and it is running Podman container.
- ▶ IBM ESS 3200 contains a POWER9 EMS node and it is running Podman container.
- ▶ IBM ESS 3500 nodes were added to `/etc/hosts` which contains the same information on POWER8 EMS and POWER9 EMS:
  - Low-speed names: fully qualified domain names (FQDNs), short names, and IP addresses
  - High-speed names: FQDNs, short names, and IP addresses (add suffix of low-speed names)
- ▶ Host name and domain is set in POWER9 EMS.
- ▶ Latest code for IBM ESS 3000 and IBM ESS 5000 stored in `/home/dep1oy` on POWER8 and POWER9 EMS.
- ▶ Linux root password is common across all of the nodes (Legacy, IBM ESS 3000, IBM ESS 5000, IBM ESS3200 and IBM ESS 3500).

## Adding IBM ESS 3500 to an ESS Legacy cluster

Run the `config load` command within an IBM ESS 3500 container that is running in the POWER9 EMS to fix the SSH keys across all of the nodes. See Example 3-1.

*Example 3-1 Run config load with ESS Legacy*

---

```
ESS UNIFIED v6.1.3.1 CONTAINER [root@cems0 /]# essrun -N
ESS3500Node1,ESS3500Node2,GSSNode1,GSSNode2,ESS3500EMSNode,GSSEMSNode config load
-p RootPassword
```

---

Create bonds in IBM ESS 3500 building block within IBM ESS 3500 container that is running in the POWER9 EMS. See Example 3-2.

*Example 3-2 Create network bonds*

---

```
ESS UNIFIED v6.1.3.1 CONTAINER [root@cems0 /]# essrun -N
ESS3500Node1,ESS3500Node2,ESS3500EMSNode network --suffix=Suffix
```

---

Add IBM ESS 3500 I/O nodes to the existing cluster from ESS3200Node1. See Example 3-3.

*Example 3-3 Add IBM ESS 3500 nodes to ESS Legacy cluster*

---

```
[root@ESS3200Node1~]# essaddnode -N ESS3500Node1,ESS3500Node2 --cluster-node
GSSEMSNode --nodetype ess3500 --suffix=Suffix --accept-license --no-fw-update
```

---

Add ESS EMS node (Example 3-4) to the existing cluster from ESS3200Node1. See Example 3-4.

*Example 3-4 Add POWER9 EMS to ESS Legacy cluster*

---

```
[root@ESS3200Node1~]# essaddnode -N <POWER9 EMS> --cluster-node ESSLegacyNode1  
--nodetype ems --suffix=Suffix --accept-license --no-fw-update
```

---

### **Adding IBM ESS 3500 to an IBM ESS 5000 cluster**

Run the **config load** command within the ESS container that is running in the POWER9 EMS to fix the SSH keys across all of the nodes. See Example 3-5.

*Example 3-5 Run config load with IBM ESS 3500*

---

```
ESS UNIFIED v6.1.3.1 CONTAINER [root@cems0 /]# essrun -N  
ESS3500Node1,ESS3500Node2,ESS5000EMSNode,ESS5000Node1,ESS5000Node2 config load -p  
RootPassword
```

---

Create bonds in IBM ESS 3500 building block within ESS container that is running in the POWER9 EMS. See Example 3-6.

*Example 3-6 Create network bonds*

---

```
ESS UNIFIED v6.1.3.1 [root@cems0 /]# essrun -N ESS3500Node1,ESS3500Node2 network  
--suffix=Suffix
```

---

Add IBM ESS 3500 I/O nodes to the existing IBM ESS 5000 cluster from within the ESS container that is running in the POWER9 EMS. See Example 3-7.

*Example 3-7 Add IBM ESS 3500 nodes to ESS 5000 cluster*

---

```
ESS UNIFIED v6.1.3.1 CONTAINER [root@cems0 /]# essrun -N ESS5000Node1 cluster  
--add-nodes ESS3500Node1,ESS3500Node2 --suffix=Suffix
```

---

### **Adding IBM ESS 3500 to an ESS 3000 cluster**

Run the **config load** command within ESS Unified container that is running in the POWER9 EMS to fix the SSH keys across all of the nodes. See Example 3-8.

*Example 3-8 Run config load with ESS 3000*

---

```
ESS UNIFIED v6.1.3.1 CONTAINER [root@cems0 /]# essrun -N  
ESS3500Node1,ESS3500Node2,ESS3500EMSNode,ESS3000Node1,ESS3000Node2,ESSP8EMSNode  
config load -p RootPassword
```

---

Create bonds in IBM ESS 3500 building block within ESS unified container that is running in the POWER9 EMS. See Example 3-9.

*Example 3-9 Create network bonds*

---

```
UNIFIED v6.1.3.1 CONTAINER [root@cems0 /]# essrun -N  
ESS3500Node1,ESS3500Node2,ESS3500EMSNode network --suffix=Suffix
```

---



Add IBM ESS 3500 I/O nodes to existing the ESS 3000 cluster from within ESS 3000 container that is running in the POWER9 EMS. See Example 3-10.

*Example 3-10 Add IBM ESS 3500 nodes to IBM ESS 3000 cluster*

---

```
UNIFIED v6.1.3.1 CONTAINER [root@cems0 /]# essrun -N ESS3000Node1 cluster
--add-nodes ESS3500Node1,ESS3500Node2 --suffix=Suffix
```

---

Add IBM ESS 3500 EMS node to existing IBM ESS 3000 cluster from within IBM ESS 3000 container that is running in the POWER9 EMS. See Example 3-11.

*Example 3-11 Add IBM ESS 3500 EMS to ESS 3000 cluster*

---

```
ESS UNIFIED v6.1.3.1 CONTAINER [root@cems0 /]# essrun -N ESS3000Node1 cluster
--add-ems ESS3500EMSNode --suffix=Suffix
```

---

### **Adding IBM ESS 3500 to a mixed ESS Legacy cluster (IBM ESS 3000, IBM ESS 3200 and IBM ESS 5000)**

Run the **config load** command within IBM ESS 3500 container that is running in the POWER9 EMS to fix the SSH keys across all of the nodes. See Example 3-12.

*Example 3-12 Running config load with ESS Legacy + ESS 3000 + ESS 5000*

---

```
ESS UNIFIED v6.1.3.1 CONTAINER [root@cems0 /]# essrun -N
ESS3500Node1,ESS3500Node2,ESS3500EMSNode,ESS5000Node1,ESS5000Node2,ESS3000Node1,
ESS3000Node2,GSSNode1,GSSNode2,GSEMSNode,ESS3200Node1,ESS3200Node2,ESS3200EMSNode,
ESS3500Node1,ESS3500Node2
config load -p RootPassword
```

---

Create bonds in IBM ESS 3500 building block within the IBM ESS 3500 container that is running in the POWER9 EMS. See Example 3-13.

*Example 3-13 Creating network bonds*

---

```
ESS UNIFIED v6.1.3.1 CONTAINER [root@cems0 /]# essrun -N
ESS3500Node1,ESS3500Node2,ESS3500EMSNode network --suffix=Suffix
```

---

Add IBM ESS 3500 I/O nodes to existing ESS Legacy of IBM ESS 3000 cluster, IBM ESS 5000 and IBM ESS 3200 from within IBM ESS 3200 container that is running in the POWER9 EMS. See Example 3-14.

*Example 3-14 Adding IBM ESS 3500 nodes to IBM ESS 5000 cluster*

---

```
ESS UNIFIED v6.1.3.1 CONTAINER [root@cems0 /]# essrun -N
ESS3500Node1,ESS3500Node2,ESS5000EMSNode,ESS5000Node1,ESS5000Node2 config load -p
RootPassword
```

---

### **Adding IBM ESS 3500 to an IBM ESS 3200 cluster**

Run the **config load** command within the IBM ESS 3500 container that is running in the POWER9 EMS to fix the SSH keys across all of the nodes. See Example 3-15.

*Example 3-15 Defining SSH keys across the nodes*

---

```
ESS UNIFIED v6.1.3.1 CONTAINER [root@cems0 /]# essrun -N
ESS3500Node1,ESS3500Node2,ESS5000EMSNode,ESS5000Node1,ESS5000Node2 config load -p
RootPassword
```

---

Create bonds in IBM ESS 3500 building block within IBM ESS 3500 container that is running in the POWER9 EMS. See Example 3-16.

*Example 3-16 Creating network bonds*

---

```
ESS UNIFIED v6.1.3.1 [root@cems0 /]# essrun -N ESS3500Node1,ESS3500Node2 network
--suffix=Suffix
```

---

Add IBM ESS 3500 I/O nodes to the existing IBM ESS 3200 cluster from within an ESS container that is running in the POWER9 EMS. See Example 3-17.

*Example 3-17 Adding IBM ESS 3500 I/O nodes to an existing IBM ESS 3200 Cluster*

---

```
ESS UNIFIED v6.1.3.1 CONTAINER [root@cems0 /]# essrun -N ESS3200Node1 cluster
--add-nodes ESS3500Node1,ESS3500Node2 --suffix=Suffix
```

---

### 3.3.2 Scenario 1: Using IBM ESS 3500 for metadata network shared disks

This section describes how to use IBM ESS 3500 for metadata network shared disks (NSDs) with the existing file system.

This scenario contains an existing IBM ESS 5000 cluster and file system deployed from a POWER9 EMS.

The steps for the high-level plan are as follows:

1. Deploy the IBM ESS 3500 container into the POWER9 EMS.
2. Add the IBM ESS 3500 Building Block to the cluster.
3. Create the IBM ESS 3500 VDisk set as metadataOnly.
4. Add the IBM ESS 3500 VDisk set to the existing IBM ESS 5000 file system.

The following steps provide guidance to set up the IBM ESS 3500 for metadata NSDs for the existing file system:

1. Deploy the IBM ESS 3500 container into the POWER9 EMS: Log in to the POWER9 EMS and completing the [ESS common installation instructions](#) from the Quick Deployment Guide.

2. Add the IBM ESS 3500 Building Block to the cluster: From step 1, within the container, run the Ansible `essrun` command to add the new IBM ESS 3500 to the IBM ESS 5000 cluster. The parameter “`essio1`” is an example name of an existing ESS I/O node in the cluster:

```
root@cems0:/ # essrun -N essio1 cluster --add-nodes CommaSeparatedNodesList
--suffix=-hs
```

3. Create the IBM ESS 3500 VDisk set as metadataOnly:

Within the container, run the Ansible `essrun` command to create the new ESS VDisk set (using 16M as block size, because IBM ESS 5000 file system is the default one):

```
root@cems0:/ # essrun -N ess32001a,ess32001b vdisk --name newVdisk --bs 16M
--suffix=-hs --extra-vars "--nsd-usage metadataOnly --storage-pool system"
```

4. Add the IBM ESS 3500 VDisk set to the existing IBM ESS 5000 file system:

From the POWER9 EMS node, use `mmvdisk` command to add the new VDisk to the existing file system:

```
[root@p9ems ~]# mmvdisk filesystem add --file-system filesystemName --vdisk-set
vs_newVdisk
```

### 3.3.3 Scenario 2: Using IBM ESS 3500 to create a file system

To create a file system with IBM ESS 3500, the IBM ESS 3500 container must be running.

After the container is running and the cluster and recovery groups are created, create the file system by running the `essrun` command in which “`essio1`” and “`essio1`” are example names of existing IO nodes:

```
$ essrun -N essio1,essio2 filesystem --suffix=-hs
```

**Note:** This command creates vdisk sets, NSDs, and file systems by using `mmvdisk`. The defaults are 4M blocksize, 80% set size, and 8+2p RAID code. These values can be customized by using additional flags.

For CES deployment, the IBM ESS 3500 system should have a CES file system. To create the CES file system, run the following `essrun` command:

```
$ essrun -N essio1,essio2 filesystem --suffix=-hs --name cesSharedRoot --ces
```

**Note:** A CES and other file systems can coexist on the same IBM ESS cluster.





# Providing your own IBM Elastic Storage System Management Server

This chapter discusses deployment of the IBM Elastic Storage System (ESS) 3500 with outside ESS Management Server (EMS). It includes the following topics:

- ▶ 4.1, “Requirements” on page 76
- ▶ 4.2, “EMS VM deployment” on page 79
- ▶ 4.3, “Other considerations” on page 91

Beginning with Version 6.1.5, for new deployments of the IBM ESS 3500 you can choose to manage the IBM ESS 3500 by ordering the POWER9 EMS hardware from IBM. Alternatively, you can provide your own hardware and run the EMS for the cluster as a virtual machine (VM). This VM is referred to as the EMSVM and this type of EMS is referred to as bring your own EMS (BYOE).

When you are deciding which EMS option to use, consider the differences between your own hardware and the hardware that is validated, supported, and monitored by IBM. The biggest difference between POWER9 EMS hardware and EMS hardware that is provided by the customer is that there is no hardware monitoring by the ESS solution on the host that is running the EMSVM. There are no call-home related events to hardware, no firmware updates on the hardware that hosts the EMSVM, and no operating system (OS) updates on the base OS hosting the EMSVM. Call home monitoring, firmware updates, and OS updates are the responsibility of the customer. IBM takes no ownership nor support on such activities.

**Note:** The OS of the EMSVM updates as though it were an IBM POWER9 EMS. All the hardware of the BYOE solution (I/O nodes and IBM Protocol Nodes) is monitored and upgraded by using standards and contracts that are related to the EMSVM but not related to IBM POWER9 EMS.

## 4.1 Requirements

The following requirements are current at the time of writing. Always check the latest available IBM Documentation for current information and updates.

**Note:** These requirements can be complex because of the requirements for BIOS, system board, hardware, wiring, and cards. It is the responsibility of the customer to ensure that the requirements are fully met. To meet the requirements of the BYOE hardware, customers might need assistance from the hardware vendor. IBM ESS support does not provide support on how to meet these requirements on BYOE hardware.

### 4.1.1 Host requirements

For hosts that are not IBM hosts, specific hardware requirements must be met for running the EMSVM to be considered running in a supported environment.

The requirements cover two different configurations: *small* and *standard*. Small configurations can be used to manage no more than two ESS units. Standard configurations can be used to manage up to the same number ESS as physical EMS systems from IBM in a one to one ratio.

Supported BYOE EMS hosts must have the following characteristics:

- ▶ AMD EPYC processor, single or dual socket. (AMD EPYC Naples not supported).
- ▶ VT-x enabled.
- ▶ KVM enabled and installed at OS.
- ▶ A PCI pass-through capable system.
- ▶ Minimum of 600 GB free on `/ems vm`.
- ▶ `/ems vm` must be mounted on a local file system on a local internal device (not iSCSI, FC, NFS, or IBM Spectrum Scale).
- ▶ One quad port Ethernet card for low-speed network (LSN), with all ports identical.
- ▶ Red Hat Enterprise Linux 8.6 running in the host with a valid subscription.
- ▶ SUSE Linux Enterprise Server disabled on host.
- ▶ Host fully compliant with tuned virtual-host profile.
- ▶ No other VM running in the host.
- ▶ No other workload running in the host.
- ▶ IBM Spectrum Scale software is not installed on the host.
- ▶ At least one ConnectX-6 VPI card for a high-speed network (HSN) connection.
- ▶ No more than the three HSN NICs.
- ▶ At least two HSN ports.
- ▶ No more than six HSN ports.

#### Small setups

For small setups, the host must have a minimum of 16 cores and 128 GB RAM with at least PCIe Version 3.0 x 16 lanes for HSN slots.

## Standard setups

For standard setups, the host must have a minimum of 32 cores and 256 GB RAM with at least PCIe Version 4.0 x 16 lanes for HSN slots.

**Note:** It is important to check latest available IBM Documentation for the most recent list of requirements. Some of these requirements might change in future updates.

IBM created a tool to assess the readiness of a node to run IBM Spectrum Scale EMS as a VM. Customers can [download the tool from GitHub](#) and certify their hosts before you make any purchase from IBM.

## Additional EMS host recommendations

Although not mandatory for a supported configuration, it is a best practice to have the following additional configuration items in your environment. Some might become mandatory on future releases:

- ▶ Mirrored host OS drive.
- ▶ A way to have out of band management of the host.
- ▶ An Ethernet port that is not part of the quad card or HSN cards that are mapped to the EMSVM to allow remote access to the host OS
- ▶ Redundant power supplies.

## 4.1.2 Networking

The EMSVM has two types of networks, low-speed network (LSN) connections provided by a quad port Ethernet card with all identical ports, and 1 - 3 other cards for high-speed network (HSN) connections each with 2 - 6 available ports. Figure 4-1, maps the ports of that quad card with their logical use.

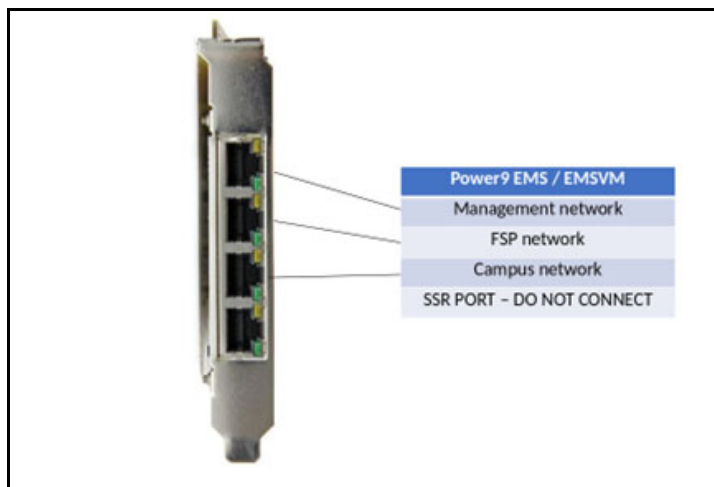


Figure 4-1 LSN NIC port details

For more information about each network, see 3.1.4, “Network planning” on page 65 and the IBM ESS documentation.

### Low-speed network

The LSN in the EMSVM is comparable to the quad port card on POWER9 EMS systems. The LSN in the EMSVM is used to connect to the ESS Management network, the ESS FSP network, and the customer network. The lower port is IBM reserved; do not use the lower port.

### High-speed network

The high-speed network (HSN) used on the EMSVM is identical to what would be on a POWER9 EMS node, used to connect to the IBM Spectrum Scale daemon and admin networks. For more information, see 3.1.4, “Network planning” on page 65 and IBM Documentation for IBM ESS.

## 4.1.3 Other information

To run the EMSVM management tool that is available from the public GitHub, the host must fulfill the following requirements:

- ▶ Python 3.6+ installed
- ▶ Federal Information Processing Standards disabled on host (needed for MD5 sums)
- ▶ The following Red Hat Package Manager (RPM) packages must be installed:
  - dmidecode
  - pciutils
  - coreutils
  - numactl-libs
  - tuned
  - virt-manager
  - libvirt-daemon
  - libvirt-client
  - libvirt
  - libguestfs-tools
  - virt-install
  - qemu-img
  - python3-libvirt
  - python3-requests
  - python3-netifaces
  - python3-ethtool
  - xz
  - xz-libs

## 4.1.4 EMSVM hosts with more than one quad port card

If the EMSVM host has more than one quad port Ethernet card with identical ports, then map the lowest PCI addresses only to the EMSVM. Do not map any other card to the EMSVM. See Example 4-1.



*Example 4-1 Ethernet PCI address output*

---

```
# grep PCI_SLOT_NAME /sys/class/net/*/device/uevent
/sys/class/net/eth0/device/uevent:PCI_SLOT_NAME=0000:45:00.0
/sys/class/net/eth1/device/uevent:PCI_SLOT_NAME=0000:45:00.1
/sys/class/net/eth2/device/uevent:PCI_SLOT_NAME=0000:45:00.2
/sys/class/net/eth3/device/uevent:PCI_SLOT_NAME=0000:45:00.3
/sys/class/net/eth4/device/uevent:PCI_SLOT_NAME=0000:c3:00.0
/sys/class/net/eth5/device/uevent:PCI_SLOT_NAME=0000:c3:00.1
/sys/class/net/eth6/device/uevent:PCI_SLOT_NAME=0000:c3:00.2
/sys/class/net/eth7/device/uevent:PCI_SLOT_NAME=0000:c3:00.3
/sys/class/net/ib0/device/uevent:PCI_SLOT_NAME=0000:27:00.0
/sys/class/net/ib1/device/uevent:PCI_SLOT_NAME=0000:27:00.1
```

---

The card on 000:45:00 PCI address (lowest in hexadecimal) is the one that would be passed to the EMSVM.

**Note:** The PCI address is a concern only if the EMSVM host has more than one quad port Ethernet card with identical ports. If any other type of card is used for host access, this can be ignored.

## 4.2 EMS VM deployment

For a successful deployment of the EMS as a virtual machine, ensure that before you start deployment, the chosen system passes validation, and ensure that the host fulfills the requirements that are listed in 4.1.1, “Host requirements” on page 76.

The following additional items are required to complete the initial configuration of the EMSV:

- ▶ Access to EMSVM repository on public [rGitHub for the emsvm tool download](#)
- ▶ IBMid account that has access to IBM FixCentral for downloading the EMSVM disk image.
- ▶ IP address information that is used during the initial configuration of the EMSVM
- ▶ Campus network information:
  - IP address
  - Netmask
  - Gateway
- ▶ ESS management network information:
  - IP address
  - Netmask
- ▶ OS FSP network information:
  - IP address
  - Netmask

**Note:** As a best practice, obtain an additional IP address that can be configured for the OS of the system hosting the EMSVM. This IP address must not be configured on the 4-port Ethernet adapter that is assigned to the EMSVM.

## 4.2.1 EMSVM deployment flow

Successful EMSVM deployment requires that the defined deployment flow is being followed.

Refer to the latest version of IBM Documentation for EMSVM deployment as the information that is presented in this book might have an update after this book is published.

Figure 4-2 shows the deployment for the EMSVM.

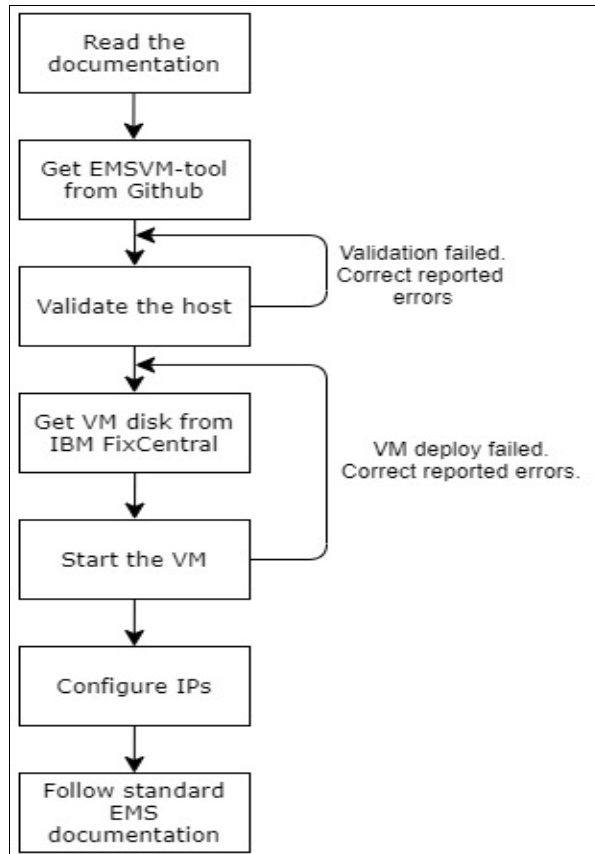


Figure 4-2 EMSVM deployment flow

## 4.2.2 Reviewing the current documentation

Read the latest version of IBM Documentation available for the EMSVM deployment as the information that is presented in this book might be updated after the book is published.

## 4.2.3 The emsvm tool

EMSVM deployment is automated and controlled by the **emsvm** tool, which is written by IBM. The **emsvm** tool is used for the initial deployment of EMSVM and for the management operations of the EMSVM. The current version of the **emsvm** tool can be downloaded from the [GitHub repository](#).

The **emsvm** tool must be copied to a file system that is local to the host that will be running the EMSVM.

### emsvm tool options

The **emsvm** tool provides several options. To list details for each option that can be used by the tool, run the tool without additional options or with **-h** as shown in Example 4-2.

*Example 4-2 emsvm tool option details*

---

```
[root@x86node EMSVM]# ./emsvm -h
usage: EMS VM [-h] (--check-host | --connect-EMS | --start-EMS | --check-EMS)
                [-d] [-v]
```

EMS VM host check and VM run

optional arguments:

- h, --help show this help message and exit
- check-host Assess the readiness of the host to run EMS VM
- connect-EMS Connect to the EMS VM console information
- start-EMS Start EMS VM in this host
- check-EMS Check if EMS VM is UP
- d, --debug Print verbose messages also to shell
- v, --version show program's version number and exit

```
[root@x86node EMSVM]#
```

---

### --check-host

Running the **emsvm** tool with **--check-host** option checks the host system to verify that all prerequisites to run the EMSVM on this host are met. Reviewing this output for any errors or warnings that are reported can be used to help identify issues preventing the successful start of the EMSVM. Also, you can use this option to determine whether a selected host can run EMSVM before you decide whether to run the EMSVM or use an IBM provided EMS.

**Note:** When using the tool only to check the host before an actual start of the EMSVM, the EMSVM disk image from FixCentral is not required.

## --connect-EMS

This option connects the current user session to the console of the currently running EMSVM on the host.

The **connect-EMS** option must be used for the initial start of a new EMSVM to access the virtual machine and perform the *IBM SSR setup* task. This is required as the EMSVM does not have any default IP addresses configured.

The *IBM SSR setup* task configures the following information on EMSVM according to user-supplied information:

- ▶ Password for user root
- ▶ SSH keys and passwordless SSH
- ▶ Network IP addresses:
  - Campus network
  - ESS Management network
  - OS FSP network

**Note:** Do not configure IP addresses manually by using OS provided tools. The *IBM SSR setup* tool must be used for IP address configuration of the EMSVM.

## --start-EMS

The **start-EMS** option of **emsvm** tool is used to start the EMSVM. A manual startup of the EMSVM is required the first time that the virtual machine is deployed and if the virtual machine was manually shutdown.

After initial deployment, when the host boots, the EMSVM is set to autostart with the host. There is no need to use the **start-EMS** option every time the host boots.

## --check-EMS

This **check-EMS** option of **emsvm** tool can be used to validate the status of EMSVM and reports the EMSVM domain status on the host.

The **check-EMS** option returns the status of EMSVM domain. In addition to human-readable output, the command returns “0”(zero) if the EMSVM domain is found active and “1” (one) in any other state.

This information can be used to validate the power state of EMSVM with an external monitoring tool that is running in the host. A sample output is shown in Example 4-3.

*Example 4-3 emsvm tool --check-EMS option output*

```
[root@x86node EMSVM]# ./emsvm --check-EMS

2022-11-16 08:39:37,269 INFO:    Welcome to EMS VM version 0.53
2022-11-16 08:39:37,269 INFO:    Please visit https://github.com/IBM/EMSVM for
issues and new versions
2022-11-16 08:39:37,269 INFO:    Log file with details for this run is saved on:
/var/log/emsvm/emsvm_check EMS_2022_11_16_08_39_37.log
2022-11-16 08:39:37,279 INFO:    This system is AMD EPYC based
2022-11-16 08:39:37,279 INFO:    This host does not have AMD EPYC 1st Generation
(Naples)
2022-11-16 08:39:37,388 INFO:    This host CPU[s] are KVM capable and has
extensions enabled
```

```

2022-11-16 08:39:37,388 INFO:    Going to check if /dev/kvm exists
2022-11-16 08:39:37,389 INFO:    The device /dev/kvm exists
2022-11-16 08:39:38,417 INFO:    Red Hat Enterprise Linux 8.6 is a supported OS
2022-11-16 08:39:38,428 INFO:    Checking RPM packages status
2022-11-16 08:39:38,581 INFO:    All required RPM packages have the expected
installation status
2022-11-16 08:39:38,665 INFO:    Could not get current version from repository,
please be sure that you are running latest version from
https://github.com/IBM/EMSVM
2022-11-16 08:39:43,712 INFO:    EMSVM KVM domain exists
2022-11-16 08:39:43,712 INFO:    KVM VM domain EMSVM is already active
[root@x86node EMSVM]#

```

---

## 4.2.4 Validating the host

The host system running the virtualization layer for the EMSVM must fulfill the previously stated hardware and software requirements. The `emsvm` tool is used to validate the host platform every time the tool is run on the EMSVM, like deployment or starting the EMSVM.

The `emsvm` tool writes a log file on the host in the `/var/log/emsvm` directory each time the tool is run. These logs can be used to audit the activity taken by the EMSVM when using the `emsvm` tool and to debug issues during deployment and startup of the EMSVM.

Validate the host before proceeding to deploy the EMSVM by checking the summary results at the end of the tool output. A sample output can be found in Example 4-4.

*Example 4-4 emsvm tool --check-host option output*

---

```

[root@x86node EMSVM]# ./emsvm --check-host
2022-11-14 06:06:51,400 INFO:    Welcome to EMS VM version 0.53
2022-11-14 06:06:51,400 INFO:    Please visit https://github.com/IBM/EMSVM for
issues and new versions
2022-11-14 06:06:51,400 INFO:    Log file with details for this run is saved on:
/var/log/emsvm/emsvm_check_host_2022_11_14_06_06_51.log
2022-11-14 06:06:51,409 INFO:    This system is AMD EPYC based
2022-11-14 06:06:51,409 INFO:    This host does not have AMD EPYC 1st Generation
(Naples)
2022-11-14 06:06:51,516 INFO:    This host CPU[s] are KVM capable and has
extensions enabled
2022-11-14 06:06:51,516 INFO:    Going to check if /dev/kvm exists
2022-11-14 06:06:51,516 INFO:    The device /dev/kvm exists
2022-11-14 06:06:52,534 INFO:    Red Hat Enterprise Linux 8.6 is a supported OS
2022-11-14 06:06:52,544 INFO:    Checking RPM packages status
2022-11-14 06:06:52,697 INFO:    All required RPM packages have the expected
installation status
2022-11-14 06:06:52,779 INFO:    Could not get current version from repository,
please be sure that you are running latest version from
https://github.com/IBM/EMSVM
2022-11-14 06:06:57,785 INFO:    Looking that we have at least one NIC for High
Speed Network (HSN)
2022-11-14 06:06:57,803 INFO:    The NIC 'Infiniband controller: Mellanox
Technologies MT28800 Family [ConnectX-6 Ex]' is OK as HSN NIC. Going to append PCI
address '0000:27:00.0' as HSN NIC

```

```

2022-11-14 06:06:57,803 INFO: The NIC 'Infiniband controller: Mellanox
Technologies MT28800 Family [ConnectX-6 Ex]' is OK as HSN NIC. Going to append PCI
address '0000:27:00.1' as HSN NIC
2022-11-14 06:06:57,823 INFO: Found at least two HSN ports
2022-11-14 06:06:57,823 INFO: Looking that we have at least one quad port
Ethernet NIC for Low Speed Network (LSN)
2022-11-14 06:06:57,832 INFO: The NIC 'Ethernet controller: Intel Corporation
I350 Gigabit Network Connection (rev 01)' is OK as LSN NIC. Going to append PCI
address '0000:45:00.' for ports 0, 1, 2 and 3
2022-11-14 06:06:57,832 INFO: Host has 32 core[s] which complies with 32 cores
required
2022-11-14 06:06:57,832 INFO: Total memory is 251.38 GB, which is more than the
required 247 GB
2022-11-14 06:07:00,181 INFO: Current tune profile is 'virtual-host'
2022-11-14 06:07:00,358 INFO: OS settings match the running profile
2022-11-14 06:07:00,399 INFO: KVM storage pool EMSVM already exists
2022-11-14 06:07:00,449 INFO: The path /emsvm has 1181.29 GB free excluding
reserved space which is more than the required 600 GB
2022-11-14 06:07:00,455 INFO: KVM executable file /usr/libexec/qemu-kvm exists
2022-11-14 06:07:00,455 INFO:
2022-11-14 06:07:00,455 INFO: -----
2022-11-14 06:07:00,455 INFO: | Summary of this run |
2022-11-14 06:07:00,455 INFO: -----
2022-11-14 06:07:00,455 INFO: Log file with details for this run is saved on:
/var/log/emsvm/emsvm_check_host_2022_11_14_06_06_51.log
2022-11-14 06:07:00,456 INFO: Start of the checks on: 2022/11/14 06:06:51
2022-11-14 06:07:00,456 INFO: End of the checks on: 2022/11/14 06:07:00
2022-11-14 06:07:00,456 INFO: All tests were passed passed on x86node
2022-11-14 06:07:00,456 INFO: The path /emsvm has 1181.29 GB free excluding
reserved space which is more than the required 600 GB
2022-11-14 06:07:00,456 INFO: CPU model is AMD EPYC 7302 16-Core Processor
2022-11-14 06:07:00,456 INFO: This host has 2 socket(s)
2022-11-14 06:07:00,456 INFO: The VM CPU cores would be 30 core[s]
2022-11-14 06:07:00,456 INFO: The VM memory would be 239 GB
2022-11-14 06:07:00,456 INFO: The VM HSN PCI address[es] would be
['0000:27:00.0', '0000:27:00.1']
2022-11-14 06:07:00,456 INFO: The VM LSN PCI addresseses would be
['0000:45:00.0', '0000:45:00.1', '0000:45:00.2', '0000:45:00.3']
2022-11-14 06:07:00,456 INFO: OS is running and matching tune profile
virtual-host

```

OK: All tests passed, you can run EMS VM on x86node host

```
[root@x86node EMSVM]#
```

---

**Note for small setups:** Even when the validation checks pass, you receive a “WARNING” status message in the summary instead of “OK” as a reminder that the configuration is limited to managing two ESS.

## 4.2.5 Downloading the EMSVM disk image from IBM FixCentral

An IBMid with a valid subscription is required to download the EMSVM disk image from [IBM Fix Central for vHMC](#).

The downloaded disk image file must be transferred to the `/emsvm` file system of the host running the EMSVM.

**Note:** EMSVM is delivered as a compressed qcow2 disk image file. Do not decompress or rename the file.

*Example 4-5 EMSVM.qcow2.xz file must be placed in the /emsvm directory unchanged*

---

```
[root@x86node emsvm]# pwd
/emsvm
[root@x86node emsvm]# ls -la
drwxr-xr-x. 2 root root      4096 Nov 14 06:42 .
dr-xr-xr-x. 23 root root     4096 Oct 20 06:15 ..
-rw-r--r-- 1 root root 1094970356 Nov 14 06:42 EMSVM.qcow2.xz
```

---

## 4.2.6 Starting the EMSVM

A manual start by using the `emsvm` tool is required only on first start of a new EMSVM or if the EMSVM is shutdown manually. During the first manual start, the EMSVM is set to autostart with the host system. Example 4-6 shows the output if no compressed disk file exists.

*Example 4-6 Error during start the EMSVM with no virtual disk file*

---

```
[root@x86node EMSVM]# ./emsvm --start-EMS
...
2022-11-14 06:41:06,611 WARNING:      EMSVM KVM domain does not exist. Going to
create it
2022-11-14 06:41:06,638 INFO:       KVM VM has been created
2022-11-14 06:41:06,638 INFO:       KVM VM domain EMSVM is set to autostart
2022-11-14 06:41:06,638 ERROR:     EMS VM compressed disk file /emsvm/EMSVM.qcow2.xz
does not exist. Download the file from IBM and place it on the expected directory
[root@x86node EMSVM]#
```

---

Example 4-7 shows the output if the quad-port adapter has an IP address configured when the EMSVM is started.

*Example 4-7 Error during start of the EMSVM with an IP address configured on the quad-port adapter*

---

```
[root@x86node EMSVM]# ./emsvm --start-EMS
...
2022-11-14 06:45:00,240 INFO:       OS device name: 'eth2' PCI address:
'0000:45:00.0' Description: 'Ethernet controller: Intel Corporation I350 Gigabit
Network Connection (rev 01)
2022-11-14 06:45:00,240 INFO:       OS device name: 'eth5' PCI address:
'0000:45:00.1' Description: 'Ethernet controller: Intel Corporation I350 Gigabit
Network Connection (rev 01)
2022-11-14 06:45:00,240 INFO:       OS device name: 'eth6' PCI address:
'0000:45:00.2' Description: 'Ethernet controller: Intel Corporation I350 Gigabit
Network Connection (rev 01)'
```

```

2022-11-14 06:45:00,240 INFO: OS device name: 'eth7' PCI address:
'0000:45:00.3' Description: 'Ethernet controller: Intel Corporation I350 Gigabit
Network Connection (rev 01)'
2022-11-14 06:45:00,243 INFO: Interface eth2 has IP 192.168.20.231
2022-11-14 06:45:00,244 ERROR: We found IP address[es] on one or more interfaces
that are going to be passthrough to the VM
[root@x86node EMSVM]#

```

---

If no errors are encountered during the prerequisites check on startup, a confirmation to start the EMSVM on the host is displayed as shown in Example 4-8.

*Example 4-8 Successful start of EMSVM*

---

```

[root@x86node EMSVM]# ./emsvm --start-EMS
2022-11-14 06:48:28,934 INFO: Welcome to VM version 0.53
2022-11-14 06:48:28,934 INFO: Please visit https://github.com/IBM/EMSVM for
issues and new versions
2022-11-14 06:48:28,934 INFO: Log file with details for this run is saved on:
/var/log/emsvm/emsvm_start EMS_2022_11_14_06_48_28.log
2022-11-14 06:48:28,941 INFO: This system is AMD EPYC based
2022-11-14 06:48:28,942 INFO: This host does not have AMD EPYC 1st Generation
(Naples)
2022-11-14 06:48:29,052 INFO: This host CPU[s] are KVM capable and has
extensions enabled
2022-11-14 06:48:29,052 INFO: Going to check if /dev/kvm exists
2022-11-14 06:48:29,052 INFO: The device /dev/kvm exists
2022-11-14 06:48:30,068 INFO: Red Hat Enterprise Linux 8.6 is a supported OS
2022-11-14 06:48:30,078 INFO: Checking RPM packages status
2022-11-14 06:48:30,232 INFO: All required RPM packages have the expected
installation status
2022-11-14 06:48:30,316 INFO: Could not get current version from repository,
please be sure that you are running latest version from
https://github.com/IBM/EMSVM
2022-11-14 06:48:35,321 INFO: Looking that we have at least one NIC for High
Speed Network (HSN)
2022-11-14 06:48:35,338 INFO: The NIC 'Infiniband controller: Mellanox
Technologies MT28800 Family [ConnectX-6 Ex]' is OK as HSN NIC. Going to append PCI
address '0000:27:00.0' as HSN NIC
2022-11-14 06:48:35,338 INFO: The NIC 'Infiniband controller: Mellanox
Technologies MT28800 Family [ConnectX-6 Ex]' is OK as HSN NIC. Going to append PCI
address '0000:27:00.1' as HSN NIC
2022-11-14 06:48:35,357 INFO: Found at least two HSN ports
2022-11-14 06:48:35,357 INFO: Looking that we have at least one quad port
Ethernet NIC for Low Speed Network (LSN)
2022-11-14 06:48:35,365 INFO: The NIC 'Ethernet controller: Intel Corporation
I350 Gigabit Network Connection (rev 01)' is OK as LSN NIC. Going to append PCI
address '0000:45:00.' for ports 0, 1, 2 and 3
2022-11-14 06:48:35,366 INFO: Host has 32 core[s] which complies with 32 cores
required
2022-11-14 06:48:35,366 INFO: Total memory is 251.38 GB, which is more than the
required 247 GB
2022-11-14 06:48:38,097 INFO: Current tune profile is 'virtual-host'
2022-11-14 06:48:38,273 INFO: OS settings match the running profile
2022-11-14 06:48:38,314 INFO: KVM storage pool EMSVM already exists

```



```
2022-11-14 06:48:38,362 INFO:    The path /emsvm has 1181.29 GB free excluding
reserved space which is more than the required 600 GB
2022-11-14 06:48:38,367 INFO:    KVM executable file /usr/libexec/qemu-kvm exists
2022-11-14 06:48:38,367 INFO:    All tests passed, starting EMS VM
2022-11-14 06:48:38,368 WARNING:    EMSVM KVM domain does not exist. Going to
create it
2022-11-14 06:48:38,394 INFO:    KVM VM has been created
2022-11-14 06:48:38,394 INFO:    KVM VM domain EMSVM is set to autostart
2022-11-14 06:48:38,395 INFO:
2022-11-14 06:48:38,395 INFO:    You have requested to start the EMS VM in this
host
2022-11-14 06:48:38,395 INFO:    The following PCI devices are going to be
assigned to the VM and are not going to be accessible from the host
2022-11-14 06:48:38,395 INFO:    If you are using any of them to connect to the
host, you are going to lose connectivity to this host
2022-11-14 06:48:38,395 INFO:    Please be sure that you are not using any of the
following devices before continuing the start up of the EMS VM
2022-11-14 06:48:38,432 INFO:    PCI address: '0000:27:00.0' Description:
'Infiniband controller: Mellanox Technologies MT28800 Family [ConnectX-6 Ex]'
2022-11-14 06:48:38,433 INFO:    PCI address: '0000:27:00.1' Description:
'Infiniband controller: Mellanox Technologies MT28800 Family [ConnectX-6 Ex]'
2022-11-14 06:48:38,433 INFO:    OS device name: 'eth2' PCI address:
'0000:45:00.0' Description: 'Ethernet controller: Intel Corporation I350 Gigabit
Network Connection (rev 01)'
2022-11-14 06:48:38,433 INFO:    OS device name: 'eth5' PCI address:
'0000:45:00.1' Description: 'Ethernet controller: Intel Corporation I350 Gigabit
Network Connection (rev 01)'
2022-11-14 06:48:38,433 INFO:    OS device name: 'eth6' PCI address:
'0000:45:00.2' Description: 'Ethernet controller: Intel Corporation I350 Gigabit
Network Connection (rev 01)'
2022-11-14 06:48:38,433 INFO:    OS device name: 'eth7' PCI address:
'0000:45:00.3' Description: 'Ethernet controller: Intel Corporation I350 Gigabit
Network Connection (rev 01)'
2022-11-14 06:48:38,435 INFO:    No IP addresses found on the interfaces to be
passthrough to the VM
2022-11-14 06:48:38,435 INFO:    Do you want to continue with the start EMS VM?
(y/n):
y
2022-11-14 06:49:47,807 INFO:    It can take several minutes to start the VM,
please wait ...
2022-11-14 06:50:32,206 INFO:    EMS VM is started now ...
[root@x86node EMSVM]#
```

---

## 4.2.7 Configuring IP addresses

The downloaded EMSVM disk image does not contain pre-configured IP addresses for any network interface. During initial startup of the EMSVM, the IP addresses must be defined to allow inbound and outbound communication of EMSVM.

To configure IP addresses for EMSVM, use the `emsvm --connect-EMS` option to connect to EMSVM console from the host system as shown in Example 4-9.

*Example 4-9 Connect to the EMSVM console*

---

```
[root@x86node EMSVM]# ./emsvm --connect-EMS
2022-11-14 06:54:50,900 INFO: Welcome to EMS VM version 0.53
2022-11-14 06:54:50,900 INFO: Please visit https://github.com/IBM/EMSVM for
issues and new versions
2022-11-14 06:54:50,900 INFO: Log file with details for this run is saved on:
/var/log/emsvm/emsvm_connect EMS_2022_11_14_06_54_50.log
2022-11-14 06:54:50,909 INFO: This system is AMD EPYC based
2022-11-14 06:54:50,910 INFO: This host does not have AMD EPYC 1st Generation
(Naples)
2022-11-14 06:54:51,018 INFO: This host CPU[s] are KVM capable and has
extensions enabled
2022-11-14 06:54:51,018 INFO: Going to check if /dev/kvm exists
2022-11-14 06:54:51,018 INFO: The device /dev/kvm exists
2022-11-14 06:54:52,038 INFO: Red Hat Enterprise Linux 8.6 is a supported OS
2022-11-14 06:54:52,048 INFO: Checking RPM packages status
2022-11-14 06:54:52,201 INFO: All required RPM packages have the expected
installation status
2022-11-14 06:54:52,282 INFO: Could not get current version from repository,
please be sure that you are running latest version from
https://github.com/IBM/EMSVM
2022-11-14 06:54:57,331 INFO: EMSVM KVM domain exists
2022-11-14 06:54:57,331 INFO: KVM VM domain EMSVM is already active

Connected to domain 'EMSVM'
Escape character is ^] (Ctrl + ])
```

Red Hat Enterprise Linux 8.6 (Ootpa)  
Kernel 4.18.0-372.9.1.el8.x86\_64 on an x86\_64

```
localhost login: root
Password:
You are required to change your password immediately (administrator enforced)
Current password:
New password:
Retype new password:
Last login: Tue Nov 1 03:50:33 on ttyS0
[root@localhost ~]#
```

---

The message “Connected to domain ‘EMSVM’” is displayed. Press the Enter key to proceed to the login prompt.

**Note:** The EMSVM has a default password `ibmesscluster` configured for user `root`. The default password must be changed on first login.

IP address configuration of EMSVM must be performed that uses the supplied `ess_ssr_setup` tool that is provided with the EMSVM disk image as shown in Example 4-10.

Information from the technical and delivery assessment (TDA) tool, containing a table that is populated with IP addresses, can be used to ensure a correct configuration.

The `ess_ssr_setup` tool includes the option to configure the EMSVM as a new deployment or to add the IBM EMS to an existing environment. Regardless of the selected option, the tool includes an option to test the management network connectivity to an existing device on that network. This test can be skipped if the new deployment option is selected.

*Example 4-10 IP address configuration using `ess_ssr_setup` tool*

---

```
[root@localhost ~]# ess_ssr_setup
2022-11-14 11:58:18,381 INFO: Welcome to IBM ESS 'code 20' helper tool
2022-11-14 11:58:18,381 INFO: You are going to need the TDA table information
to continue. If you do not have that information, you CANNOT continue.
2022-11-14 11:58:18,381 INFO: For debug output or issues, be sure to include
the debug file on /opt/ibm/ess/tools/bin/code20_debug.log

Do you want to continue? (y/n): y

2022-11-14 11:58:39,291 INFO: We strongly recommend setting a campus network in
the EMS.

Do you want set up the campus network in this node? (y/n): y
Please type the campus IP of this node (i.e. 192.168.80.10): 192.168.64.231
Please type the campus netmask of this node (i.e. 255.255.255.0): 255.255.255.0
Please type the campus gateway IP (i.e. 192.168.80.1): 192.168.64.1
Please type the management IP of this node (i.e. 192.168.10.10): 192.168.21.231
Please type the management netmask of this node (i.e. 255.255.255.0):
255.255.255.0
Please type the OS FSP IP (C11-T2 for P9 EMS) of this EMS (i.e. 192.168.20.9):
192.168.20.232
Please type the OS FSP netmask of this node (i.e. 255.255.255.0): 255.255.255.0
2022-11-14 12:00:29,966 INFO: Going to ask for the node password. All the nodes
must have the same password for code 20
2022-11-14 12:00:29,966 INFO: Be aware that the password will not be prompted
into the screen and you will not see it on it

Password:

2022-11-14 12:00:34,965 INFO: Please type the same password again

Password:

Is this a new deployment or adding a block to a running cluster? (new/add): new

Is there already any node in this environment that we can test ping connectivity
with? (y/n): n

2022-11-14 12:01:26,158 WARNING: This is a new setup and we have no peers
to check with, no ping tests will happen

Please review the entered information before continuing.
No changes in the system had been performed at this point.
```

The entered IP in CIDR format for CAMPUS is 192.168.64.231/24  
The entered IP in CIDR format for MANAGEMENT is 192.168.21.231/24  
The entered IP in CIDR format for EMS OS FSP (C11-T2 for P9 EMS) is  
192.168.20.232/24  
No ping tests to be performed.  
The entered root password to set is: cluster

Do you want to continue and perform changes and tests in this node? (y/n): y  
2022-11-14 12:02:25,072 INFO: Going to set the root user password of this node  
to the password typed before  
2022-11-14 12:02:25,349 INFO: Run 'Root\_password\_set' completed successfully  
2022-11-14 12:02:25,436 WARNING: Passwordless root SSH to localhost test  
is passed. We are going to try to fix it. If a password is asked type same you  
where entered during this run.  
The authenticity of host 'localhost (::1)' can't be established.  
ECDSA key fingerprint is SHA256:eT/DjXNMy61MxM8z4J+3eD9INi7u0Cwcb/aEZpfq0A.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
2022-11-14 12:02:34,811 INFO: Self fix of root SSH to localhost worked, moving  
on  
2022-11-14 12:02:34,814 INFO: Run 'Passwordless root SSH localhost' completed  
successfully  
2022-11-14 12:02:34,814 INFO: Going to set Management Network IP on this node  
2022-11-14 12:02:34,866 WARNING: The command sudo nmcli con show mgmt  
returned code 10  
2022-11-14 12:02:34,867 INFO: The output of running the command sudo nmcli con  
show mgmt was:

Error: mgmt - no such connection profile.

0 0 0 0

2022-11-14 12:02:35,051 INFO: Run 'Set MGMT IP' completed successfully  
2022-11-14 12:02:35,051 INFO: Going to set Campus Network IP on this node  
2022-11-14 12:02:35,102 WARNING: The command sudo nmcli con show campus  
returned code 10  
2022-11-14 12:02:35,102 INFO: The output of running the command sudo nmcli con  
show campus was:  
Error: campus - no such connection profile.

2022-11-14 12:02:35,276 INFO: Run 'Set Campus IP' completed successfully  
2022-11-14 12:02:35,276 INFO: Going to set OS FSP Network IP on this node  
2022-11-14 12:02:35,325 WARNING: The command sudo nmcli con show fsp  
returned code 10  
2022-11-14 12:02:35,325 INFO: The output of running the command sudo nmcli con  
show fsp was:

Error: fsp - no such connection profile.

2022-11-14 12:02:35,502 INFO: Run 'Set OS FSP IP' completed successfully  
2022-11-14 12:02:35,502 WARNING: This is the first node of a new setup, no  
ping tests are going to be performed

```
2022-11-14 12:02:35,506 INFO: Run 'Ping not tested' completed successfully
```

The following tasks were run in this node

```
TASK: DB_init was successfully run on 2022-11-14T11:58:18.141767
TASK: Root_password_set was successfully run on 2022-11-14T12:02:25.346899
TASK: Passwordless root SSH localhost was successfully run on
2022-11-14T12:02:34.812026
TASK: Set MGMT IP was successfully run on 2022-11-14T12:02:35.048829
TASK: Set Campus IP was successfully run on 2022-11-14T12:02:35.273364
TASK: Set OS FSP IP was successfully run on 2022-11-14T12:02:35.497673
TASK: Ping not tested was successfully run on 2022-11-14T12:02:35.502590
2022-11-14 12:02:35,508 INFO: All run tasks were successful
2022-11-14 12:02:35,509 INFO: All required tests were completed successfully
2022-11-14 12:02:35,509 INFO: CODE 20 completed.
```

```
[root@localhost ~]#
```

---

A successful execution of `ess_ssr_setup` tool returns a message `CODE 20 completed`. Successful execution with a return code of *Code 20* is a mandatory requirement to allow the EMSVM to be part of an IBM Spectrum Scale cluster.

After a successful defining of the IP addresses using the `ess_ssr_setup` tool, you can access the EMSVM by using an SSH client.

**Note:** To disconnect from the console session of EMSVM on Linux systems, enter the key combination `^]`. For windows systems, enter the key combination `Ctrl + ^`.

At this point, the EMSVM is ready for use. To continue with the ESS setup, follow the deployment process that is described in the [IESS Quick Deployment Guide](#).

## 4.3 Other considerations

The following section contains information that is related to additional tasks that are commonly performed on the EMSVM after initial configuration is complete. These tasks are related to the general administration of the host that is running EMSVM, which is a customer responsibility.

### 4.3.1 Backing up and restoring EMSVM

There might be situations in which the host goes into an unrecoverable state. The quickest way to recover from that situation would be to reinstall the OS on the same or different host, ensure that host also passes the requirements check, then start your EMSVM from this system.

To be able to reinstall the OS, maintain a copy of the `/emsvm/ESSVM.qcow2` file outside of the host; this copy must be consistent. A best practice is to copy the file while the EMSVM is shutdown. Any type of quiescence of the EMSVM or the underlying file system in the host, creates the possibility for corruption of the EMSVM OS or a quorum loss at the IBM Spectrum Scale cluster that the EMSVM is part of.

When shutting down the EMSVM for a backup and before making the copy, be sure to verify that your cluster is not in a quorum loss situation.

To perform a recovery after a host failure, if there is a good remote copy of the EMSVM virtual disk, get the EMSVM GitHub tool to re-create the EMSVM on the host. The host can be the same host or a different host. Copy the virtual disk into the `/emsvm` directory and then start the EMSVM with the GitHub tool.

### 4.3.2 Stopping EMSVM

The IBM GitHub EMSVM tool does not have a stop EMSVM function. This is by design as shutting down the EMSVM is always done from within the EMSVM. Connect to the host with either SSH or with the EMSVM tool's `connect-EMS` option.

Verify that shutting down the EMSVM does not cause a problem in your IBM Spectrum Scale cluster. Shutdown the EMSVM with standard OS commands.

If the EMSVM OS is non-responsive, you can also restart the host as a way to restart the EMSVM.

**Note:** Remember, after initial configuration the EMSVM is set to auto start when the host boots and does not need to be started manually.

As a last resort, it is possible to use the standard `virsh` commands from the host to stop the EMSVM. However, this is not recommended.

### 4.3.3 Monitoring host HW using an EMSVM

When using an EMSVM, it is the responsibility of the customer to monitor the host HW. Follow the best practices of the hardware vendor for monitoring the host HW to collect, monitor, and address any issues.

### 4.3.4 Accessing the host

While not required for an EMSVM, it is a best practice to define extra Ethernet ports on the host for remote access and out of band management access to the host. Do this according to the best practices of the host hardware vendors.

### 4.3.5 Updating host OS and firmware

Keeping the host OS and firmware current and compatible with the EMSVM is the responsibility of the customer. IBM does not include support that is related to those tasks with ESS or IBM Spectrum Scale.

When doing an OS upgrade, be sure that the host OS remains at a version that is supported by the EMSVM.

When doing firmware upgrades, especially with the HSN cards, verify the version that is being installed is compatible with the Mellanox OFED version that EMSVM uses. This is documented in the IBM Documentation.

Also, before rebooting the host, confirm that the EMSVM is down and that there are no issues with the IBM Spectrum Scale cluster.

### 4.3.6 Changing the port type of NVIDIA ConnectX VPI adapter

If a change of the VPI port type from Ethernet to InfiniBand or the opposite is required, it will need to be done from the host and never from the EMSVM. Refer to the documentation of your system provider on how to make this change. In most cases this includes the use of the `mst` command.

Example steps to complete this change:

- ▶ Shut down the EMSVM.
- ▶ Verify that there are no issues with the quorum of the IBM Spectrum Scale cluster.
- ▶ Change the needed ports to the required type.
- ▶ Restart the host. EMSVM starts automatically.
- ▶ Start using the ports from within the EMSVM.

Alternatively, this can be set before deploying the EMSVM on the host.







## Use cases

This chapter discusses IBM Elastic Storage System (ESS) 3500 use cases. It includes the following topics:

- ▶ 5.1, “Introducing performance storage use cases” on page 96
- ▶ 5.2, “Metadata and high-speed data tier” on page 98
- ▶ 5.3, “Data feed to GPUs for massive AI data acceleration” on page 99
- ▶ 5.4, “Other use cases” on page 99

## 5.1 Introducing performance storage use cases

Across industries and locations around the world, high-performance storage use cases can appear to vary significantly. However, upon looking closely at today's storage data and AI applications, a generalized view exists of where high-performance storage fits into today's storage data and AI infrastructure. See Figure 5-1.

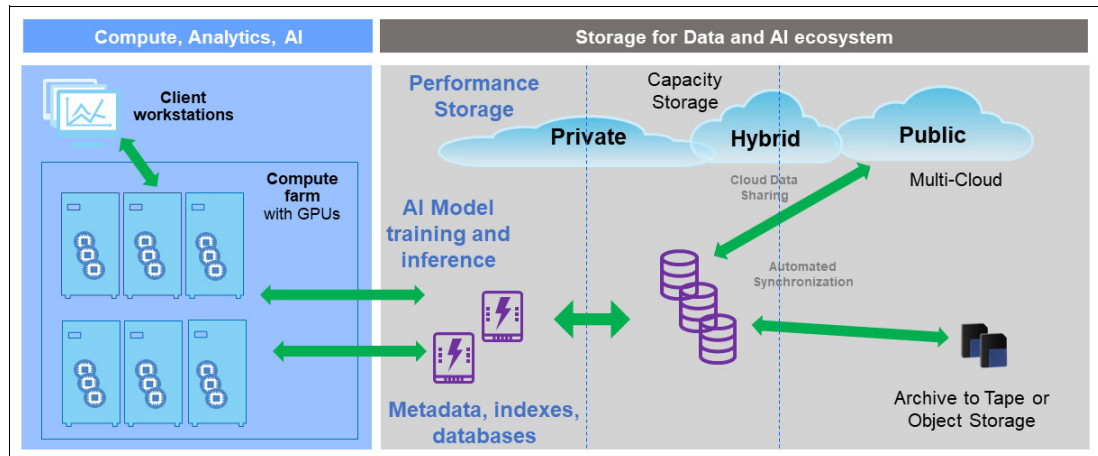


Figure 5-1 Performance storage use case positioning

Storage data and AI use cases require high-performance storage and some other features:

- ▶ An infrastructure of dynamic, scalable, reliable, high-performance storage
- ▶ A storage tier that delivers GBps to TBps of throughput to provide storage for graphics processing units (GPUs) and modern computers.
- ▶ Performance storage that must also seamlessly integrate as part of an enterprise data fabric that also has capacity tiers for other uses of the storage:
  - Enterprise data repositories
  - Scalable flexible hybrid cloud tiers
  - Cost-effective archive tape and object tiers

The following sections describe how the IBM ESS 3500, as performance storage, addresses essential storage data and AI application requirements for AI model training, inference, metadata, indexes, and databases. The IBM ESS 3500 is also described as a seamless, integrated data component, in a larger data and AI infrastructure based on IBM Spectrum Scale.

## 5.1.1 IBM ESS 3500 as part of a larger storage for data and AI infrastructure

Further examination of the storage infrastructure of a performance-storage scenario, shows the IBM ESS 3500 is positioned as a high-performance storage system within the performance tier.

Also, the IBM ESS 3500 is part of a larger family of IBM Storage solutions that comprehensively covers all aspects of the storage for data and AI configuration, as shown in Figure 5-2.

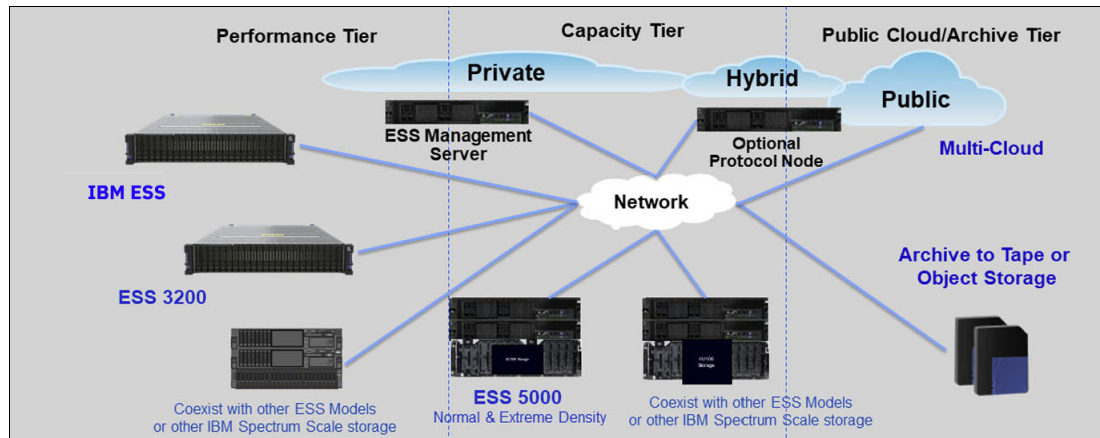


Figure 5-2 ESS positioning within the data and AI storage infrastructure

The IBM ESS 3500 is part of a larger storage infrastructure. Within the larger storage infrastructure, you can use IBM Spectrum Scale to nondisruptively add, expand, and modify the storage data and AI infrastructure, as needed. The IBM ESS 3500 serves as storage for multiple types of configurations:

- ▶ You can install the IBM ESS 3500 as a stand-alone, single, high-performance system.
- ▶ You can add additional IBM ESS 3500 enclosures to expand the performance tier.
- ▶ You can add one or multiple IBM ESS 5000 enclosures as HDD high-capacity tier.
- ▶ You can add flexibility by adding other IBM and third-party storage components to the storage infrastructure, including hybrid cloud capacity, and archive capacity to tape or object storage.

The diversity and flexibility is because the IBM ESS 3500 is part of a set of storage data devices and AI devices that provides an end-to-end enterprise data-fabric and data-management storage solution. This is all powered by the IBM Spectrum Scale single global namespace that spans all storage tiers.

## 5.1.2 Typical IBM ESS 3500 performance storage use cases

The IBM ESS 3500 is designed to provide scalable, reliable, dense, fast storage for the performance storage tier. Typical use cases for the IBM ESS 3500 include specific performance tier high-performance computing (HPC), AI, analytics, or other high-performance workloads:

- ▶ AI applications that require high-performance data effectively using GPU technology at high resource usage
- ▶ Acceleration of scale-out applications with dense NVMe Flash technology
- ▶ Information Lifecycle Management and data-tiering management of data in new or existing IBM Spectrum Scale environments
- ▶ Metadata acceleration, indexes, database acceleration
- ▶ High-performance storage at the edge

The following sections of this chapter explore some of these use cases.

## 5.2 Metadata and high-speed data tier

In an IBM Spectrum Scale cluster, performance of the entire cluster can be accelerated by placing IBM Spectrum Scale and other metadata on the IBM ESS 3500. Thus, in a high-performance computing environment, a predominant use for the IBM ESS 3500 is to provide the high-performance metadata storage for IBM Spectrum Scale by using the high throughput of NVMe flash storage.

Metadata generally refers to *data about data*, and in the context of IBM Spectrum Scale *metadata* refers to various on-disk data structures that are necessary to manage user data. Directory entries and inodes are defined as metadata, but at times the distinction between data and metadata might not be obvious.

For example, examine a configuration with a 4 KB inode. Although the inode might contain user data, the inode is still classified as IBM Spectrum Scale metadata. The inode is placed in a metadata pool if data and metadata are separated. Another example is the case of directory blocks, which are classified as metadata but also contain user file and directory names.

In many high-performance use cases, performance improvements might be obtained by assigning IBM Spectrum Scale metadata to a fast tier, which can be accomplished by placing faster IBM ESS 3500 storage in its own storage pool. For details about how to implement this technique, see 3.3.2, “Scenario 1: Using IBM ESS 3500 for metadata network shared disks” on page 72.

This approach to metadata tiering can be adopted when optimizing the performance of metadata operations, such as listing directories and making `stat()` calls on files. For more information, see IBM Documentation for IBM Spectrum Scale on [User Storage Pools](#).

Alternatively, instead of tiering data based on a data or metadata classification, you can use the IBM Spectrum Scale File Heat function to migrate data between storage pools based on how frequently data is accessed. For more information about this approach, see IBM Documentation for IBM Spectrum Scale [File Heat: Tracking File Access Temperature](#).

## 5.3 Data feed to GPUs for massive AI data acceleration

Another predominant use for the IBM ESS 3500 is to provide the high-performance storage throughput that is necessary to feed modern GPU data acceleration systems for real-time AI and machine learning (ML). NVIDIA and IBM created a reference architecture for NVIDIA DGX and IBM ESS 3500 working together on AI and ML workloads. Together, NVIDIA and IBM provide an integrated, individually scalable compute and storage solution with end-to-end parallel throughput from flash to GPU for accelerated DL training, and inference. For more information, see [IBM Storage and SDI solutions Reference Architecture](#) and [IBM Storage Reference Architecture with NVIDIA DGX A100 Systems](#).

These reference architectures provide a blueprint for enterprise leaders, solution architects, and others who want to learn how the IBM Spectrum Storage for AI with NVIDIA DGX systems simplifies and accelerates AI. The scalable infrastructure solution integrates the NVIDIA DGX systems with IBM Spectrum Scale GPU Direct file storage software, which powers the IBM ESS family of storage systems that includes the IBM ESS 3500.

The reference architecture document describes how the read throughput of the DGX A100 system increases linearly with the addition of an IBM ESS system. For example, in the document, a single IBM ESS system provides a read throughput of 48 GBps. Adding a second IBM ESS system increases the read throughput to 94 GBps, which is almost twice the throughput of a single IBM ESS system.

AI and ML workload can benefit from the outstanding performance capabilities of the IBM ESS 3500 system. For more information about using the IBM ESS 3500 with high-performance GPU, see [IBM, NVIDIA Team on Supercomputing Scalability for AI](#).

For more information about the IBM Spectrum Scale GPUDirect Storage (GDS) Technical Preview, see [IBM Spectrum Scale: GDS is available as a Technical Preview in Spectrum Scale 5.1.1](#).

## 5.4 Other use cases

The IBM ESS 3500 runs IBM Spectrum Scale as its file system, so some use cases and planning that apply to other members of the IBM Spectrum Scale family also apply to the IBM ESS 3500.

### 5.4.1 IBM Spectrum Scale with big data and analytics solutions

IBM Spectrum Scale provides flexible and scalable software-defined file storage for analytics workloads. Enterprises around the globe deploy IBM Spectrum Scale to form large data repositories to perform high-performance computing (HPC) and analytics workloads. IBM Spectrum Scale is known to scale performance and capacity without bottlenecks.

Cloudera is a leader in Hadoop and Spark distributions. Cloudera addresses the needs of data-at-rest, powers real-time customer applications, and delivers robust analytics that accelerate decision-making and innovation. IBM Spectrum Scale solves the challenge of explosive growth of unstructured data against a flat IT budget. IBM Spectrum Scale provides unified file and software-defined object storage for high-performance, large-scale workloads, and it can be deployed on-premises or in the cloud. Refer to [Cloudera Data Platform Private Cloud Base with IBM Spectrum Scale, REDP-5608](#).

IBM Spectrum Scale is compatible with Portable Operating System Interface (POSIX), so it supports various applications and workloads. By using IBM Spectrum Scale Hadoop Distributed File System (HDFS) Transparency Hadoop connector, you can analyze file data and object data in place, without data transfer or data movement. Traditional systems and analytics systems use and share data that is hosted on IBM Spectrum Scale file systems.

Hadoop and Spark services can use a storage system to save IT costs because special-purpose storage is not required to run the analytics. IBM Spectrum Scale features a rich set of enterprise-level data management and protection features. These features include snapshots, information lifecycle management (ILM), compression, and encryption, all of which can provide more value than traditional analytic systems do. For more information, see [IBM Spectrum Scale: Big Data and Analytics, REDP-5397](#).

## 5.4.2 Genomics medicine workloads in IBM Spectrum Scale

IT administrators, physicians, data scientists, researchers, bioinformaticians, and other professionals who are involved in the genomics workflow need the right foundation to achieve their research objectives efficiently. At the same time, they want to improve patient care and outcomes. Thus, it is important to understand the different stages of the genomics workload and the key characteristics of it.

Advanced genomics medicine customers are outgrowing network-attached storage (NAS). The move from a traditional NAS system or a modern scale-out NAS system to a parallel file system like IBM Spectrum Scale requires a new set of skills. For basic background information and for information about optional professional services, see [IBM Spectrum Scale Best Practices for Genomics Medicine Workloads, REDP-5479](#).

You can combine the IBM ESS 3500 in an IBM Spectrum Scale cluster, with Cloud Object Storage through the IBM Spectrum Scale Active File Management (AFM)-to-Cloud Object Storage feature. This function enables copies of files or objects in an IBM ESS 3500 or IBM Spectrum Scale cluster to be written to, or retrieved from, an external Cloud Object Storage. The same functions can also be used to read or write data to or from other external NFS data sources.

This integration provides the IBM ESS 3500 with the ability to seamlessly integrate and accelerate IBM Spectrum Scale data access to and from external NFS storage and to object storage such as Amazon S3 and IBM Cloud® Object Storage.

The IBM ESS 3500 integrates external NFS storage and object storage into a common data repository with enterprise-high scalability, data availability, security, and performance. The AFM-to-cloud object storage associates an IBM Spectrum Scale file set with a Cloud Object Storage bucket. Figure 5-3 shows an example of this configuration.

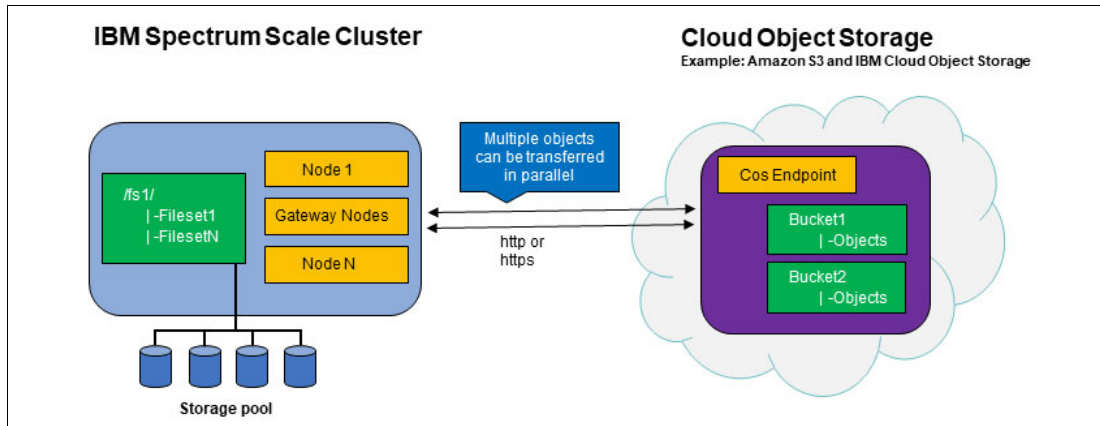


Figure 5-3 IBM Spectrum Scale Active File Management connected to Cloud Object Storage

Using this function, IBM Spectrum Scale file sets and Cloud Object Storage buckets become extensions of each other. Files and objects that are required for applications such as AI and big data analytics can be shared, downloaded, worked upon, and uploaded between the IBM ESS 3500, IBM Spectrum Scale, and the Cloud Object Storage. These use cases are shown in Figure 5-4.

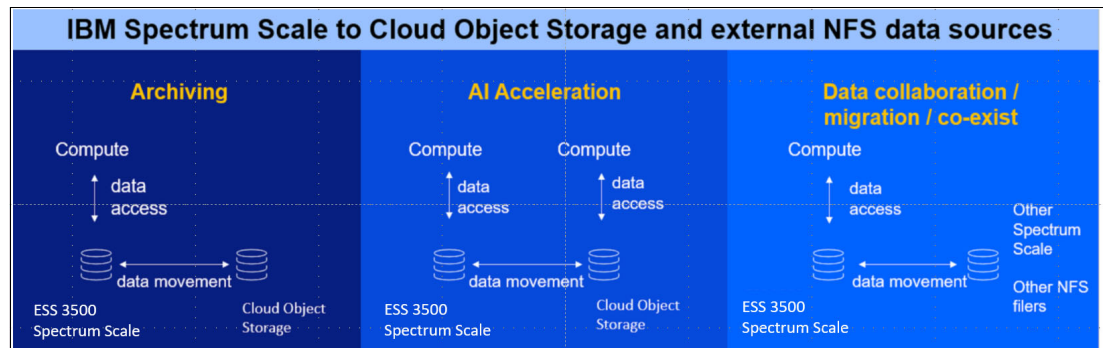


Figure 5-4 IBM Spectrum Scale to Cloud Object Storage and external NFS data sources

The following use cases are a subset of possible use cases:

- ▶ Archiving data to and from external object storage data sources
- ▶ Optimizing data movement and data access, speeding time to data value
- ▶ Connecting and migrating data to consolidate, or co-exist with, external NFS NAS data sources

The workloads and workflows that might benefit from these use cases include (but are not limited to) mobile applications, backup and restore, enterprise applications, big data analytics, and file servers.

The AFM-to-Cloud Object Storage feature also allows data center administrators to free IBM ESS 3500 and IBM Spectrum Scale storage capacity by policy management. Data is moved to lower-cost or off-premise cloud storage, which reduces capital and operational expenditures. The data movement can be done automatically through the AFM-based cache eviction feature or through policy. The data movement can be used to automate and optimize data placement between IBM ESS 3500 and other storage within the IBM Spectrum Scale storage infrastructure.





# A

## **Configuring the 48 ports top of the rack management network switch**

This appendix also provides you with the information to consider when not using the switches that IBM provides and supports for the management network. It describes how to configure the management network switch to achieve the same functionality with your own network devices.

## Configuring the switches

As shown in 3.1.2, “Hardware planning” on page 60, the management switch includes ports 1 - 12 as “ESS 3500” ports. Those ports are different from Version 1 because both management flexible service processor (FSP) networks are configured in the same port.

The process to platform the switch is not changed from Version 1. The configuration-content of the file is used to platform the switch. The same two VLANs that were used on Version 1 are used in Version 2. New VLANs are not added from Version 1.

Follow these steps to configure a Version 2 switch:

1. Connect through *serial*. You lose access to the switch if you apply it through IP access because the default configuration does not have any IP address configured.

You must have the cumulus user ID and password. It is likely the user ID of *cumulus* and a password of *CumulusLinux!*, but it might be the serial number. The serial configuration is:

- Baud rate: 115200
- Parity: None
- Stop bits: 1
- Data bits: 8
- Flow control: None

You need the configuration file that is detailed at the end of this appendix in Example A-1 on page 105.

2. Log in to the switch by using the *cumulus* user ID.
3. Enter the command `sudo su -`.
4. As *root*, put the contents of the configuration file into `/etc/network/interfaces`. (You can copy and paste.) Previous contents of the configuration file must be discarded.

**Important:** Connect through serial, or you lose access in step 5.

5. Apply the configuration with the command `ifreload -a`.  
At this point, the new configuration is applied.
6. Confirm that the configuration change was applied by using `ifquery -a`.
7. A best practice is to set a static IP for remote log in on the switch, for example, given these parameters: network is 192.168.44.0/24; IP switch 192.168.44.20; gateway 192.168.44.1:
  - `net add interface eth0 IP address 192.168.44.20/24`
  - `net add interface eth0 IP gateway 192.168.44.1`
  - `net pending`
  - `net commit`

**Note:** When you convert a switch that was not previously configured for an IBM ESS 3500, if ports 1 - 12 are not already being used, then go directly to the steps shown in the following example. If any port from 1 - 12 is being used on the switch, then these ports must be reconfigured.

Move the cables on the upper ports 1 -12, to any free upper port that is outside the 1-12 port range. Any lower cable plugged to a port in the range 1-12 also needs to be moved to any lower port not in the 1-12 port range.

Move one cable at a time and wait until the link LED on the destination port becomes lit. After all the ports in the range 1-12 are no longer cabled, continue and apply the steps that are listed in Example A-1.

The file with the configuration must contain the data that is shown in Example A-1.

*Example A-1 Data that is required for the configuration file*

---

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).
source /etc/network/interfaces.d/*.intf
# The loopback network interface auto
lo
iface lo inet loopback
# The primary network interface auto
eth0
iface eth0 inet dhcp
# EVEN Ports/Lower ports PVID 101 for FSP network auto
swp14
iface swp14
bridge-access 101
auto swp16
iface swp16
bridge-access 101
auto swp18
iface swp18
bridge-access 101
auto swp20
iface swp20
bridge-access 101
auto swp22
iface swp22
bridge-access 101
auto swp24
iface swp24
bridge-access 101
auto swp26
iface swp26
bridge-access 101
auto swp28
iface swp28
bridge-access 101
auto swp30
iface swp30
bridge-access 101
auto swp32
```

```
iface swp32
bridge-access 101
auto swp34
iface swp34
bridge-access 101
auto swp36
iface swp36
bridge-access 101
auto swp38
iface swp38
bridge-access 101
auto swp40
iface swp40
bridge-access 101
auto swp42
iface swp42
bridge-access 101
auto swp44
iface swp44
bridge-access 101
auto swp46
iface swp46
bridge-access 101
auto swp48
iface swp48
bridge-access 101
```

```
# ODD Ports/Upper ports PVID 102 for xCAT network auto
swp13
iface swp13
bridge-access 102
auto swp15
iface swp15
bridge-access 102
auto swp17
iface swp17
bridge-access 102
auto swp19
iface swp19
bridge-access 102
auto swp21
iface swp21
bridge-access 102
auto swp23
iface swp23
bridge-access 102
auto swp25
iface swp25
bridge-access 102
auto swp27
iface swp27
bridge-access 102
auto swp29
iface swp29
```

```
bridge-access 102
auto swp31
iface swp31
bridge-access 102
auto swp33
iface swp33
bridge-access 102
auto swp35
iface swp35
bridge-access 102
auto swp37
iface swp37
bridge-access 102
auto swp39
iface swp39
bridge-access 102
auto swp41
iface swp41
bridge-access 102
auto swp43
iface swp43
bridge-access 102
auto swp45
iface swp45
bridge-access 102
auto swp47
iface swp47
bridge-access 102
```

```
# ESS 3500 ports (1 to 12) FSP + OS on single physical port
auto swp1
iface swp1
bridge-pvid 102
bridge-vids 101
auto swp2
iface swp2
bridge-pvid 102
bridge-vids 101
auto swp3
iface swp3
bridge-pvid 102
bridge-vids 101
auto swp4
iface swp4
bridge-pvid 102
bridge-vids 101
auto swp5
iface swp5
bridge-pvid 102
bridge-vids 101
auto swp6
iface swp6
bridge-pvid 102
bridge-vids 101
```

```
auto swp7
iface swp7
bridge-pvid 102
bridge-vids 101
auto swp8
iface swp8
bridge-pvid 102
bridge-vids 101
auto swp9
iface swp9
bridge-pvid 102
bridge-vids 101
auto swp10
iface swp10
bridge-pvid 102
bridge-vids 101
auto swp11
iface swp11
bridge-pvid 102
bridge-vids 101
auto swp12
iface swp12
bridge-pvid 102
bridge-vids 101

# Bridge setup
auto bridge iface
bridge
bridge-vlan-aware yes
bridge-ports glob swp1-48
bridge-pvid 101
bridge-pvid 102
bridge-stp
off
```

---



# Configuring two 8831-T48 switches as top of the rack switches

This document explains how to configure a pair of ECS4100-28T switches to be the Elastic Storage System (ESS) top of the rack (TOR) switches.

This process is used by IBM manufacturing, but can be also used by field engineers when needed.

When the management TOR is part of an order, IBM delivers two switches as part of the order. Using two switches allows a similar number of ports to be available, as when a 48-port switch option is in use.

## Logical overview

From a logical perspective, the switch configuration follows the example in Figure B-1.

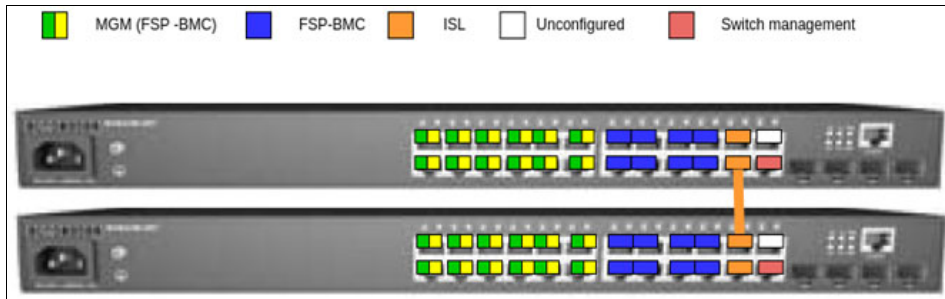


Figure B-1 Logical switch configuration

This requires that a cable, which is shown in Figure B-1 in orange, must be connected between port 22 of the upper switch and 21 of the lower switch as part of the configuration. This cable works as an interswitch link (ISL) between the two switches.

### Port definitions

Ports 1 - 12, named as *MGM (FSP-BMC)* in the green and yellow colors, are used for management connections to EMS and I/O nodes. These ports are used for all ESS models, regardless if they have dual MAC ports as in the IBM ESS 3200 and 3500 or not as in the IBM ESS 5000. This configuration is different from the previous IBM provided TOR rack in which each logical color network had dedicated ports. Green and yellow ports share the same physical ports (1 - 12).

Ports 13 - 20, named *FSP-BMC* and colored blue in Figure B-1, are to be used for systems that have dedicated FSP-BMC connections. On the EMS IBM POWER9, the ports are C11-T2 and HMC1 ports, and on the IBM ESS 5000 IO nodes the ports are HMC1 ports.

Ports 21 and 22, named *ISL* and colored orange in Figure B-1, are used only for ISL between switches. These ports allow for extending the setup to include more than two switches on a line topology. The first and last switches use one ISL connection. The first switch connects to the next switch; the last switch connects to the previous switch. The switches that are not on the edge of that line topology use both ISL ports, one to the previous switch and one to the next switch in line.

Port 23, named *Unconfigured* colored white in Figure B-1, is not used and is shutdown. This port might be used in the future.

Port 24, named *Switch management* colored red in Figure B-1, is to be used to access the management functions of the switch. It is intended for a customer switch management network and it is set up to get an IP address through the DHCP protocol.

The *Switch management* port is by default set as VLAN 1305 access port. It works on any setup that provides an access port that is connected to it. If the field setup requires a different VLAN name, change the line as shown in Example B-1.

#### Example B-1 Changing the VLAN name

---

```
VLAN 1305 name CUSTOMER media ethernet
```

---

If a change to the VLAN address definition is required, use the command that is shown in Example B-2.



### Example B-2 Changing the VLAN address definition

---

```
interface vlan 1305
 ip address dhcp
 exit
```

---

If you need to set a static IP address on the “Switch management” port, use the command in Example B-3, and replace the *ip\_address* variable with a specific IP address and *netmask* with a specific netmask.

### Example B-3 Command to define a static IP address

---

```
interface vlan 1305
 ip address ip_address netmask
 exit
```

---

In Example B-4, the IP address is set to 192.168.44.22 and netmask set to 255.255.255.0.

### Example B-4 Defining a static IP address

---

```
interface vlan 1305
 ip address 192.168.44.22 255.255.255.0
 exit
```

---

## Switch customization

Each switch requires the following customization before use. This document describes how to configure one switch. This process must be repeated on the second switch and any subsequent switches that are added to the environment.

In these examples, the “Switch management” port is not used. The connection to the switch is made with a serial connection. For the serial connection, use of an RJ-45 to DB-9 cable, which comes with the switch. You might need extra adapters and converters to connect the switch to the computer if the switch does not have a DB-9 connection on it. This is beyond the scope of this document.

The serial port (RJ-45) is on the top right of the switch. Define the serial configuration settings:

- ▶ 15200 bps
- ▶ 8 character
- ▶ no parity
- ▶ one stop bit
- ▶ no flow control.

If the switch is not configured, the default user *admin* and the default password of *admin* can be used to log in to the switch.

If further details about the initial serial connection are needed, refer to the [Quick Start Guide](#) provided by the switch manufacturer.

After successfully logging in to the switch, run the commands that are shown in Example B-5 to begin the configuration of the switch.

*Example B-5 Commands to configure the switch*

---

```
ip ssh crypto host-key generate
configure
ip ssh server
vlan database
VLAN 100 name ESS_MNG media ethernet
VLAN 101 name ESS_BMC media ethernet
VLAN 1305 name CUSTOMER media ethernet
exit
loopback-detection action none
no loopback-detection

interface ethernet 1/1-12
  switchport mode hybrid
  switchport native vlan 100
  switchport allowed vlan add 100 untagged
  switchport allowed vlan add 101 tagged
  switchport allowed vlan remove 1
  no spanning-tree loopback-detection
  no shutdown
  exit

interface ethernet 1/13-20
  switchport mode access
  switchport allowed vlan add 101 untagged
  switchport native vlan 101
  switchport allowed vlan remove 1
  no shutdown
  exit

interface ethernet 1/21-22
  switchport mode hybrid
  switchport native vlan 100
  switchport allowed vlan add 100 untagged
  switchport allowed vlan add 101 tagged
  spanning-tree spanning-disabled
  switchport allowed vlan remove 1
  no shutdown
  exit

interface ethernet 1/23
  shutdown
  no loopback-detection
  spanning-tree spanning-disabled
  exit

interface ethernet 1/24
  switchport allowed vlan add 1305 untagged
  switchport mode access
  switchport native vlan 1305
  switchport allowed vlan remove 1
  no shutdown
```

```
exit

interface vlan 1305
 ip address dhcp
 exit

exit
copy running-config startup-config
```

---

Until now, the configuration is the same for every switch. The next steps describe settings that are unique to each individual switch in the environment. Before the switch is shipped, the password for the switch is set to the serial number. The serial number of the switch is needed to continue the configuration. To retrieve the serial number, run the **show version** command, an example of which is shown in Example B-6.

*Example B-6 Example output of the show version command*

---

```
Vty-1#show version
Unit 1
Serial Number       : EC2028001435
Hardware Version    : R02A
Number of Ports     : 28
Main Power Status   : Up
Role                : Master
Loader Version      : 1.0.1.9
Linux Kernel Version : 2.6.19-g496f2361-di
Operation Code Version : 1.2.71.203
```

---

Example B-7 uses the serial number EC2028001435. Use the serial number of each switch with the commands.

*Example B-7 Example switch configuration commands*

---

```
configure
username guest password 0 EC2028001435
username admin password 0 EC2028001435
exit
copy running-config startup-config
exit
```

---

Now, you are disconnected from the switch and the switch is configured for ESS usage.



# Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this book.

## IBM Redbooks

The following IBM Redbooks publications provide additional information about the topic in this document. Note that some publications referenced in this list might be available in softcopy only.

- ▶ *Highly Efficient Data Access with RoCE on IBM Elastic Storage Systems and IBM Spectrum Scale*, REDP-5658
- ▶ *IBM Elastic Storage System Introduction Guide*, REDP-5253
- ▶ *Implementation Guide for IBM Elastic Storage System 3000*, SG24-8443
- ▶ *Implementation Guide for IBM Elastic Storage System 5000*, SG24-8498
- ▶ *Monitoring and Managing the IBM Elastic Storage Server Using the GUI*, REDP-5471

You can search for, view, download or order these documents and other Redbooks, Redpapers, Web Docs, draft and additional materials, at the following website:

[ibm.com/redbooks](http://ibm.com/redbooks)

## Online resources

These websites are also relevant as further information sources:

- ▶ IBM Documentation - IBM Elastic Storage System 3500:  
[https://www.ibm.com/docs/en/ess/6.1.5\\_lts](https://www.ibm.com/docs/en/ess/6.1.5_lts)
- ▶ IBM Spectrum Scale V 5.1.6 Planning:  
<https://www.ibm.com/docs/en/spectrum-scale/5.1.6?topic=planning>
- ▶ Licensing on IBM Spectrum Scale  
<https://www.ibm.com/docs/en/spectrum-scale/5.1.6?topic=overview-capacity-based-licensing>
- ▶ mmvdisk Command Reference:  
<https://www.ibm.com/docs/en/spectrum-scale-ece/5.1.6?topic=commands-mmvdisk-command>
- ▶ Using IBM Cloud Object Storage with IBM Spectrum Scale:  
<https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=WUS12361USEN>

## Help from IBM

- ▶ IBM Support and Downloads  
[ibm.com/support](https://ibm.com/support)
- ▶ IBM Global Services  
[ibm.com/support](https://ibm.com/support)



**Redbooks**

# Implementation Guide for IBM Elastic Storage System

SG24-8538-00

ISBN 0738460176

(1.5" spine)  
1.5" <-> 1.998"  
789 <-> 1051 pages



**Redbooks**

# Implementation Guide for IBM Elastic Storage System 3500

SG24-8538-00

ISBN 0738460176

(1.0" spine)  
0.875" <-> 1.498"  
460 <-> 788 pages



**Redbooks**

# Implementation Guide for IBM Elastic Storage System 3500

SG24-8538-00

ISBN 0738460176

(0.5" spine)  
0.475" <-> 0.873"  
250 <-> 459 pages



**Redbooks**

# Implementation Guide for IBM Elastic Storage System 3500

(0.2" spine)

0.17" <-> 0.473"

90 <-> 249 pages

(0.1" spine)

0.1" <-> 0.169"

53 <-> 89 pages



# Implementation Guide for IBM Elastic Storage System

SG24-85538-00

ISBN 0738460176

(2.5" spine)  
2.5" <-> nnn.n"  
1315 <-> nnnn pages



# Implementation Guide for IBM Elastic Storage System 3500

SG24-85538-00

ISBN 0738460176

(2.0" spine)  
2.0" <-> 2.498"  
1052 <-> 1314 pages









SG24-8538-00

ISBN 0738461210

Printed in U.S.A.

Get connected

