BROCADE
A Broadcom Company

SAN Design and Best Practices
Design Guide

Table of Contents

Chapter 1: Preface

1.1 Introduction

This document is a high-level storage area networking (SAN) design and best-practices guide based on Brocade products and features, focusing on Fibre Channel SAN design. The storage landscape continues to modernize, and multiple choices must be made to design the right Fibre Channel architecture. Covered topics include the early planning phase, understanding possible operational challenges, and monitoring and improving an existing SAN infrastructure.

The guidelines in this document do not apply to every environment, but they will help guide you through the decisions that you must make for successful SAN design. Consult your Brocade representative or refer to the documents in Appendix D for details about the hardware and software products.

NOTE: This is a “living” document that is continuously being expanded, so be sure to frequently check Broadcom.com for the latest updates to this and other best-practice documents.

1.2 Audience and Scope

This guide is for technical IT architects who are directly or indirectly responsible for SAN design based on Brocade Fibre Channel SAN platforms. It describes many of the challenges that face SAN designers today in both greenfield and legacy storage environments. While not intended as a definitive design document, this guide introduces concepts and guidelines to help you avoid potential issues that can result from poor design practices. This document describes best-practice guidelines in the following areas:

- Modernizing the storage landscape
- Architecting a SAN
- SAN topologies
- Data flows
- Predeployment infrastructure testing
- Device connections
- Scalability and performance
- Supportability
- Monitoring
- Troubleshooting
- FC routing
- FC and IP Extension
- Intelligent services
- NPIV and Access Gateway devices
- Workloads
- SAN management
- Security
- Automation

NOTE: A solid understanding of SAN concepts and Brocade Fibre Channel technology is assumed. Please see Appendix D for recommended additional publications.

1.3 Approach

Although some advanced features and specialized SAN applications are discussed, these topics are covered in greater detail in separate documents. The primary objective of this guide is to provide a solid foundation to facilitate successful SAN designs—designs that effectively meet current and future requirements. This document addresses basic administration and maintenance, including capabilities to identify early warning signs for end-device (initiator or target) latency, which can cause congestion in the SAN fabric. However, you should consult product documentation and documents in Appendix D for more details. Comprehensive discussions of SAN fabric administration, storage network cabling, and Fibre Channel security best practices are covered in separate documents.

1.4 Overview

Although Brocade SAN fabrics are plug-and-play and can function properly even if left in a default state, Fibre Channel networks clearly benefit from a well-thought-out design and deployment strategy. In order to provide reliable and efficient delivery of data, your SAN topology should follow best-practice guidelines based on SAN industry standards and considerations specific to Brocade.

This document does not consider physical environment factors such as power, cooling, and rack layout. Rather, the focus is on network connectivity edge devices to the fabric and any inter-switch links (ISLs) and software configurations.

NOTE: The scope of this document is switch-centric and does not discuss end-device setup, configuration, or maintenance. Some fabric monitoring, management, diagnostics, cabling, and migration are covered, but if you want full details, please refer to other separate documents.

Chapter 2: Storage Landscape

2.1 The Storage Landscape

In the IT infrastructure world, storage is a critical element. It is where the data resides. It is where secure copies of the data exist. It is the foundation of the overall performance of the IT application base. After all, no matter how many CPU cores and how much memory a server has, the server waits for data at the same speed as every other machine on the planet. Consequently the scope of this environment included early disk drives, tape for securely and cost effectively backing up the data and software implementations that provided access, performance and security. Furthermore, the responsibilities of the storage admin include the need to have secure copies of the data whether through RAID configurations or site-to-site replication solutions. The mantra of storage admins is “a single copy of any data set is a single point of failure waiting for a disaster to occur.” Data loss is never an acceptable option from the application point of view.

2.2 Tipping Point—The All-Flash Data Center

Over the years the state of the storage environments in IT has changed dramatically. A quick review of the changes takes you from the early tape systems through the evolution of the hard disk drive (HDD) into the development of RAID systems and enterprise arrays. One of the things that had traditionally been true was that storage, based on HDD building blocks, did not evolve rapidly. The changes from 5400 RPM disk drives to 7200 RPM disk drives as an element of performance (more data under the head per second) took some 10 years to fully populate the data center from its inception. Other developments included the density of the magnetic signature on the drive “platter” as well as the number of “platters” and “heads” per drive. This retrospective is only useful in that it denotes that the storage environment in IT did not progress as rapidly as say CPU development or memory performance and capacity. Those developments were in silicon, and the drive development was mechanical in nature. As a result, Moore’s Law applied to CPU and memory but not to storage. With the advent of solid-state drives (SSDs), this has begun to change.

Initially the progress was moderate. In a brilliant market-enabling move, the drive vendors basically made the SSD platform in the same shape/canister as existing HDDs with the same SCSI, SAS, SATA connectors. This meant “plug compatibility” on the back end of the array for the new technology. However, early on the enterprise array controllers and, in the case of embedded disk drives in servers, the OS driver stack did not do much to take advantage of the change in performance and other drive characteristics. This was part of the reason that the early “hybrid arrays” (which used a mix of traditional HDD and SSD drives on the back end) were less performant than the expectations that many storage administrators had. This had an impact on the rate of adoption but also meant that the IT organization did not see as much of an issue with the existing storage network infrastructures as one might have expected. For over 40 years in IT, we have been moving bottlenecks in CPU performance, memory speed/scale, storage capacity/performance, and network speeds. Removing one bottleneck simply lets you find the next one, not unlike how widening one segment of a major highway pushes the traffic backup to the next narrow section of the highway. The result was that many customers felt that the performance gain versus the technology cost only fit for their very highest demand applications.

However, with the advent of the all-flash array environment and some of the inline features that came along with the software (compression, encryption, deduplication), the cost per terabyte of the environment became much more enticing. Additionally, the newer enterprise array controllers began to be designed for all-flash performance characteristics and gained some significant increases in both IO per second (IOPS) and latency. Added to that was the density of the storage, which allowed them to collapse multiple racks of HDD platforms into a partial rack of SSD, as well as the benefits in power and cooling reduction. What was the consequence? A much more rapid adoption rate of all-flash arrays, which are currently more than 70% of the shipping environment.

That shift leads to a set of demands on the design of storage area networks. This is true regardless of the technology being used. If the expectation is to be able to utilize the capacity and performance of these platforms, then serious consideration has to be given to the design.

Dedicated storage network infrastructure that provides lossless, low-latency, deterministic, scalable, and performant storage services to the applications becomes critical. Storage admins will talk about “fan-in” or “fan-out” ratios for storage platforms, which are the number of servers/applications in the network that are using a particular array or array port for their storage access. Depending upon the types of applications and their performance needs, that ratio may range anywhere from low single digits to 40 to 50 servers (hundreds of virtual machines). As with any provisioning scenario, the storage admin is dealing with projections of how much capacity and performance any server/application is going to use. But application performance is variable; time of day, week, month, and seasonal or event-driven events cause spikes or drops in application demand.

Another consideration here is that the entire application base does not refresh at the same time. And in fact, the more common scenario is that multiple generations of performance will exist in the environment simultaneously. This is driven by the ability to take a service window to re-platform existing environments to new server and storage acquisitions as well as the fact that some legacy applications may not have an environment that runs in current operating system (OS) releases or application versions. One may have 10-year-old or older operating systems running connected with a host bus adapter (HBA) that is two or more older generations of technology behind. How then do you balance that still critical application against the needs and performance of the newer machines?

The answer to this is a combination of topology, balanced provisioning, granular monitoring, and automated mitigation.

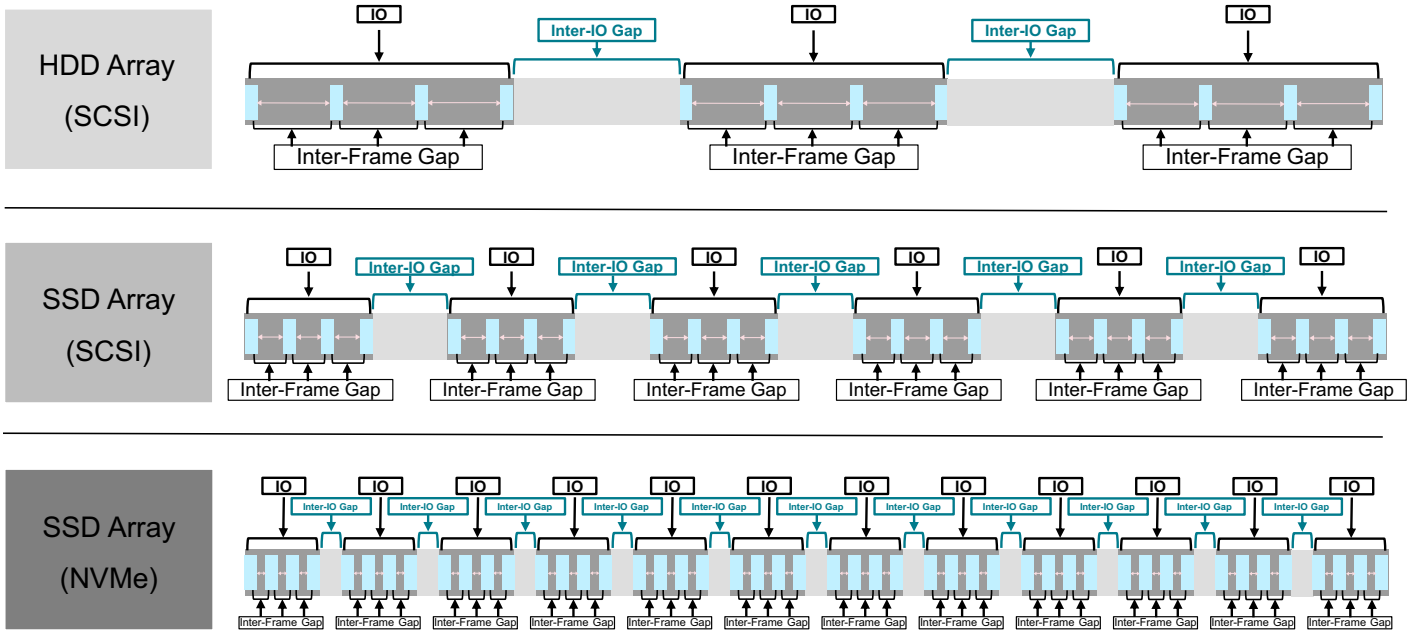
The most adaptable configuration is a core-edge topology. This design allows a significant level of scale while keeping the number of “hops” (the number of times that the data transfers from one platform to another) low. Locality of storage connectivity for high-performance applications is a consideration that drives some storage admins to place storage ports on the same blade, switch, or port group as the application server. This design, however, does impact the flexibility of IT to move the application from one server platform to another (not a trivial consideration in a highly virtualized environment where hypervisor platforms may frequently migrate applications between server platforms).

Another alternative topology is a full-mesh design. Basically, every switch or chassis has a direct ISL to every other switch or chassis in the fabric. This is more problematic for switch platforms (except for a small number of switches) since it consumes ports for inter-switch links (ISLs) that might otherwise be used to connect servers and storage. On the chassis platforms, however, the inter-chassis link (ICL) ports allow exceptional bandwidth and scale between chassis without consuming any of the blade ports that would normally be used for storage and server connectivity while ensuring that no device is more than one fabric hop away from any other device.

From a performance perspective, it is important to note that the advent of the all-flash data center also means that new storage technologies, performance based or capacity based or both, will be arriving at 18 to 24 month intervals. This does not mean a wholesale replacement of the existing platforms, but rather that your storage network must be able to accommodate roughly two of these iterations per 4 to 5 year capital depreciation cycle. One of the advantages of a Fibre Channel SAN in that regard is the dual redundant hardware isolated nature of SAN fabric design. The intent of this design is that no single human event or activity should be able to take the storage connectivity down completely, yet it also allows for more seamless upgrades of technology. This is true both as regards the storage network elements themselves and the server and storage platforms connected to them.

One of the additional changes required by the all-flash data center is the need for improved monitoring. This is due in part to the reduction in latency and the amount of data in flight per second in the storage network.

Figure 1: NVMe Implies Less Idle Time on the Network



As illustrated in Figure 1, one can readily see that the amount of idle time in the network is continuing to reduce. Consequently, the “event window” of a problem can be very brief and traditional monitoring systems based on “sample rate” (inspecting perhaps one packet in 8000), while sufficient for modeling, may not provide the rapid root-cause analysis that the modern SAN requires. This situation is further exacerbated by the scale/capacity of modern all-flash storage. Some of these platforms scale above a petabyte (PB) or 1000 terabytes (TB) of data capacity in a 2U rack height. The consideration here is that such a platform could be hosting between 6000 and 10,000 virtual machines or applications and any issue or outage that affects that kind of footprint becomes intolerable. This drives a need for more granular monitoring and for a self-optimizing and self-healing capability in the storage network. The net of the performance curve is that humans are no longer fast enough or responsive enough to problems in the all-flash data center.

2.3 NVMe

The next consideration in terms of why the storage network must be rethought and re-architected is Non-Volatile Memory Express (NVMe). At a device level, there are some characteristics to be aware of:

- Density – Current NVMe devices have 8 to 10 times the density of DRAM.
- Latency – Current NVMe devices have a sub-20-microsecond latency.
- Bandwidth – Current NVMe devices consume 4 PCIe Gen 3 lanes (32Gb/s).
- Streamlined software – Current NVMe software has 13 required and 25 optional commands.

Why are these characteristics important to consider? Because the density of the storage will continue to increase on a rough “Moore's Law” schedule. The latency of devices will continue to be reduced over time and is already significantly lower than either HDD or traditional SSD devices. Current network environments below 32Gb/s are potentially choke points for the fan-in/fan-out ratios. And the biggest issue is that for the first time in over three decades we are changing the language that we use to talk to the storage.

Taking advantage of the new technology and especially its performance aspects requires attention to the storage network. Legacy environments (whether Ethernet or Fibre Channel) will not be able to take advantage of NVMe performance.

> “Speed is the new currency of business” - Marc Benioff, CEO of Salesforce

The ability to scale to massive IOPS, to drive extreme consolidation to the new storage density, and to reduce the overhead on application servers with a streamlined software stack depends upon having the right infrastructure. An NVMe over Fibre Channel fabric is a production-ready NVMe environment.

But this will be a slide conversion. It will take time for all of the servers and applications to migrate to NVMe from SCSI. With the proper design and implementation of a Fibre Channel SAN, both NVMe and SCSI can be run concurrently on the *same* HBA, on the *same* FC switch, to the *same* NVMe-based storage. And applications can potentially be migrated from SCSI to NVMe in a nondisruptive manner.

However, this new environment will need to be self-learning, self-optimizing, and self-healing simply because it will be both too critical and too performant to wait for human intervention to solve the problems.

Chapter 3: Architecting a SAN

The SAN planning process is similar to any type of project planning and includes the following phases:

- Phase I: Gathering requirements
- Phase II: Developing technical specifications
- Phase III: Estimating project costs
- Phase IV: Analyzing Return on Investment (ROI) or Total Cost of Ownership (TCO) (if necessary)
- Phase V: Creating a detailed SAN design and implementation plan

When selecting which criteria to meet, you should engage users, server and storage subject matter experts (SMEs), and other relevant experts to understand the role of the fabric. Since most SANs tend to operate for a long time before they are renewed, you should take future growth into account as SANs are difficult to re-architect. Deploying new SANs or expanding existing ones to meet additional workloads in the fabrics requires a critical assessment of business and technology requirements. Proper focus on planning will ensure that the SAN, once deployed, meets all current and future business objectives, including availability, deployment simplicity, performance, future business growth, and cost. Tables in Appendix B are provided as a reference for documenting assets and metrics for SAN projects.

A critical aspect for successful implementation that is often overlooked is the ongoing management of the fabric. Identifying systems-level SMEs for all components that make up the SAN, as well as adequate and up-to-date training on those components, is critical for efficient design and operational management of the fabric. When designing a new SAN or expanding an existing SAN, you should take into account the following parameters.

Application Virtualization

- Which applications will run under a virtual machine (VM) environment?
- How many VMs will run on a physical server?
- Under what conditions will the VMs be migrated (business and nonbusiness hours; is additional CPU or memory needed to maintain response times)?
- Is there a need for solid-state storage media to improve read response times?

Homogeneous/Heterogeneous Server and Storage Platforms

- Are blade servers or rack servers used?
- Is auto-tiering in place?
- Which Brocade Fabric OS® (FOS) versions are supported in a multivendor storage environment?
- What is the planned refresh cycle of servers and storage platforms (2 years/3 years)?

Scalability

- How many user ports are needed now?
- How many devices will connect through an access gateway?
- How many inter-switch links (ISLs)/Brocade UltraScale inter-chassis links (ICLs) are required to minimize congestion in the fabric?
- What distances for ISL/ICL connections need to be supported?
- Does the fabric scale out at the edge or the core?

Backup and Disaster Tolerance

- Is there a centralized backup? (This will determine the number of ISLs needed to minimize congestion at peak loads.)
- What is the impact of backup on latency-sensitive applications?
- Is the disaster solution based on a metro Fibre Channel (FC) or Fibre Channel over IP (FCIP) solution?

Diagnostics and Manageability

- What is the primary management interface to the SAN (command-line interface [CLI], Brocade SANnav™, or third-party tool)?
- How often will Brocade FOS and SANnav be updated?
- How is cable and optics integrity validated?

Investment Protection

- Is support needed for adding Gen 7 switches into a Gen 6 fabric?
- Is support needed for storage technologies like NVMe over Fabrics?
- What device interoperability support is required?
- Is interoperability required for other technologies such as UCS?

3.1 Operational Considerations

Even though Brocade fabrics scale in terms of port density and performance, the design goal should be to ensure simplicity for easier management, future expansion, and serviceability. Examples of this simplicity may include using a two-tier core-edge topology, avoiding the use of both inter-fabric routing (IFR) and Virtual Fabrics (VF) where not required, and turning on port monitoring parameters for critical applications.

NOTE: Refer to the *Brocade SAN Scalability Guidelines* for currently tested and supported scalability limits. Any requirements beyond the tested scalability limits should be pretested in a nonproduction environment, or system resources like CPU and memory utilization should be actively monitored to minimize fabric anomalies.

3.2 Be the Pilot

Whether building a new SAN or connecting to an existing SAN, prestaging and validating a fabric or application before putting it into production ensures that there are baseline metrics in terms of rated throughput, latency, and expected cyclic redundancy check (CRC) errors based on patch panel and physical cable infrastructure.

3.3 Predeployment Cabling and Optics Validation

SANs built with Brocade Gen 5, Gen 6, and Gen 7 Fibre Channel switches that are equipped with 16-Gb or higher optics have the ability to run Brocade ClearLink® Diagnostics. This enables the use of predeployment testing to validate the integrity of the physical network infrastructure before operational deployment. Part of Brocade Fabric Vision® technology, Brocade ClearLink Diagnostic Port (D_Port) mode allows you to convert a Fibre Channel port into a diagnostic port for testing traffic and running electrical and optical loopback tests. The test results can be very useful in diagnosing a variety of port and link problems. ClearLink is an offline diagnostics tool that allows users to perform an automated battery of tests to measure and validate maximum throughput speeds as well as latency and distance across fabric links. ClearLink Diagnostics can also be used to verify the health and integrity of all 16-Gb and 32-Gb transceivers in the fabric on a one-by-one basis. Diagnostics should be conducted before deployment when a fabric could potentially have CRC errors caused by physical-layer issues.

A ClearLink Diagnostics port (D_Port) requires that only the individual ports attached to the tested link go offline, allowing the remainder of the ports to stay online in isolation from the link. D_Port can also be used to test links to a new fabric switch without allowing the new switch to join or even be aware of the current fabric, providing an opportunity to measure and test ISLs before they are put into production. This fabric-based physical-layer validation enables the following:

- Transceiver health check
- Transceiver uptime

- Local and long-distance measurements (5-meter granularity for 16Gb/s and 32Gb/s small form-factor pluggable [SFP] optics and 50 meters for 10Gb/s SFP optics)
- Link latency measurements between D_Ports
- Link power (dB) loss
- Link performance

Refer to the *Brocade Fabric OS Troubleshooting and Diagnostics User Guide* for a more detailed discussion of diagnostic port usage.

Refer to “Appendix A: ClearLink Diagnostics” in the *SAN Fabric Resiliency and Administration Best Practices User Guide* for details on enhancements in each FOS release.

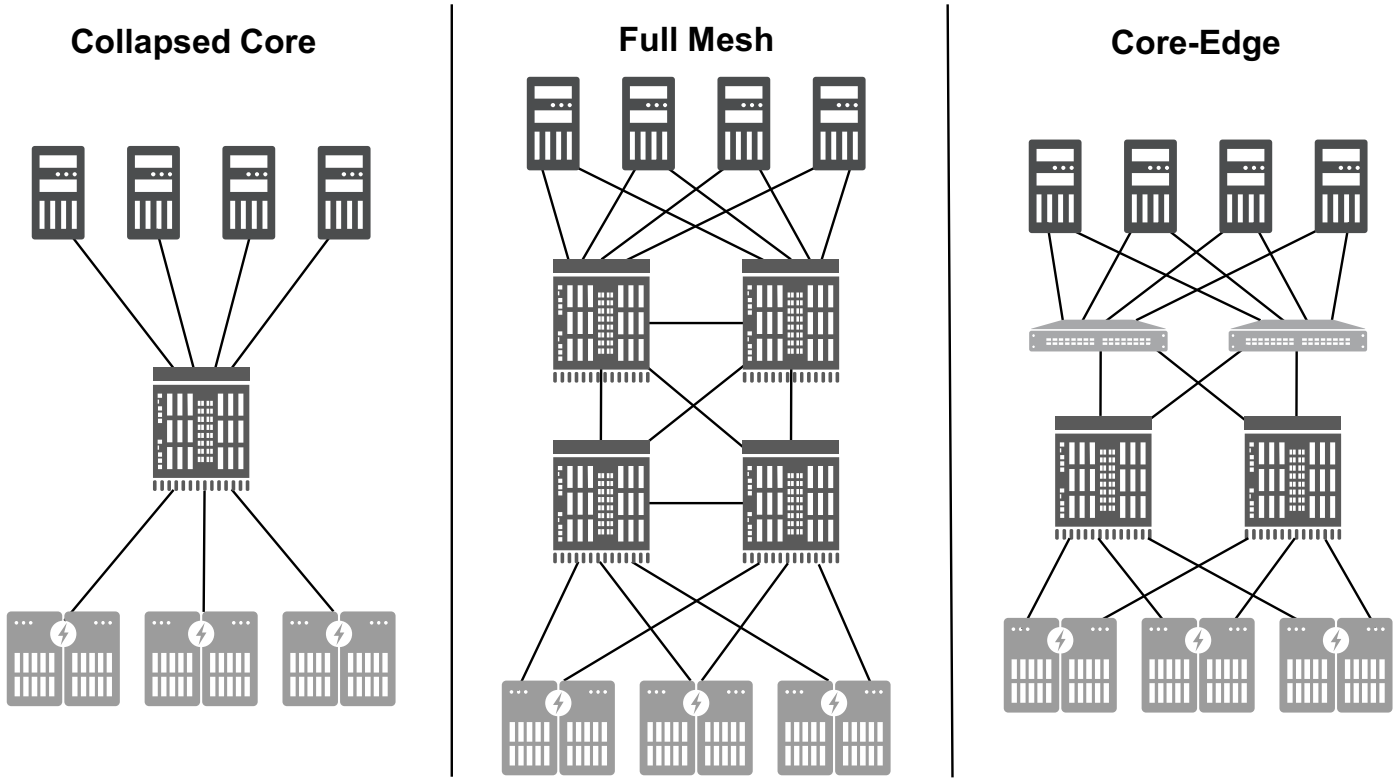
Chapter 4: SAN Design Basics

This chapter provides high-level guidelines for installing a typical SAN. The focus is on best practices for collapsed core, core-edge, or mesh fabrics. The discussion starts at the highest level, the data center, and works down to the port level, providing recommendations at each point along the way.

4.1 Topologies

A typical SAN design comprises devices on the edge of the network, switches in the core of the network, and the cabling that connects all the devices together. Topology is usually described in terms of how the switches are interconnected, such as collapsed core, core-edge, and fully meshed. The recommended SAN topology to optimize performance, management, and scalability is a tiered, core-edge topology. This approach provides good performance without unnecessary interconnections. At a high level, the tiered topology has a large number of edge switches used for device connectivity and a smaller number of core switches used for routing traffic between the edge switches, as shown in Figure 2.

Figure 2: Three Scenarios of Tiered Network Topologies



The difference between these three scenarios is device placement (where devices are attached to the network) and the associated traffic flow.

- Scenario A, collapsed core, has localized traffic, which could sit on the same ASIC, on the same blade, or between blades. Collapsed Core can have small performance advantages for performance-optimized workloads but does not provide ease of scalability and depending on how many fabrics you would need, would determine the impact on manageability.
- Scenario B, core-edge, separates the storage and servers, thus providing ease of management and higher scalability. An core-edge topology has only one hop from server to storage, providing similar performance benefits as full mesh while allowing higher scalability.
- Scenario C is a full-mesh topology, and server to storage is no more than one hop. Designing fabrics with UltraScale ICLs is an efficient way to save front-end ports, and users can easily build a large (for example, 3456 ports or larger) fabric with minimal SAN design considerations.

4.1.1 Collapsed Core

The collapsed core topology (Figure 2) places initiators (servers) and storage (targets) on the same ASIC, line card or same chassis. This has a number of benefits depending on the size of the environment. This is used a lot when customers are migrating from multiple switches to a single dual core architecture where they can fit all imitators and storage into the same chassis. If future design requirements are needed going with an core-edge could be more beneficial in the long run when it comes to scale and ease of management.

4.1.2 Core-Edge

The core-edge topology (Figure 2) places initiators (servers) on the edge tier and storage (targets) on the core tier. Since the servers and storage are on different switches, this topology provides ease of management as well as good performance with minimal fabric latency, with most traffic traversing only one hop from the edge to the core. (Storage-to-storage traffic is two hops if the second storage is on another core switch, but the two cores can be connected if fabrics are redundant.) The disadvantage to this design is that the storage and core connections are in contention for expansion as they scale; however, using modular platforms allows for flexibility while allowing the use of ICL ports for intraswitch connectivity to free up device ports.

4.1.3 Full Mesh

A full-mesh topology (Figure 2) allows you to place servers and storage anywhere, since communication between source and destination is no more than one hop. Using director-class switches with UltraScale ICL ports for interconnectivity is essential to this design in order to ensure maximum device port availability and utilization. Design this architecture with a minimum of two switches up to nine switches in a full mesh.

4.2 High-Performance Workloads

Over the last few years, enterprises have come to leverage low-latency, high-throughput flash arrays for demanding, performance-sensitive workloads. Brocade’s Gen 7 Fibre Channel is perfectly suited to these types of workloads due to the submicrosecond latency through the switch and the increased bandwidth offered by 32/64-Gb throughput speeds while providing accurate I/O latency instrumentation. Performance testing has shown that 32-Gb and even 8-Gb all-flash arrays can realize dramatic benefits by connecting to a Gen 7 SAN and host adapter, offering gains up to 2x over Gen 5 and Gen 6 SANs. The Gen 6 and Gen 7 standard includes the use of forward error correction (FEC) to ensure transmission reliability and a highly deterministic data flow. FEC corrects up to 140 corrupt bits per 5280-bit frame at the receiving end of the link, avoiding the need to retransmit frames when bit errors are detected.

For these demanding workloads, a no-hop fabric connection through a single ASIC switch like the Brocade G720 or locally switched on a single director port blade will minimize SAN fabric latency to submicrosecond speeds. Local switching is the ability to switch data traffic through a single ASIC by using both ingress and egress switching ports in a common port group. When using switches that contain multiple switching ASICs like the X6 /X7 director, configuring host and target connections on the ports that share a common ASIC will minimize latency by avoiding the need to move data across multiple ASICs/port groups or across a director backplane to a different blade. To find details on port groups and local switching configuration, refer to the *Brocade Fabric OS Administration Guide* and the hardware installation guide for the appropriate product.

Because Gen 7 FC is backwards compatible with Gen 6 networking, a Gen 7 edge switch or director can be added to an existing Gen 6 fabric. This allows for local all-flash connectivity to a Gen 7 switch to gain the performance advantages of Gen 7 while preserving the investment in Gen 6 networks.

4.3 Redundancy and Resiliency

An important aspect of SAN topology is the resiliency and redundancy of the fabric. The main objective is to remove any single point of failure. Resiliency is the ability of the network to continue to function and/or recover from a failure, whereas redundancy describes duplication of components, even an entire fabric, to eliminate a single point of failure in the network. Brocade fabrics have resiliency built into Brocade Fabric OS (FOS), the software that runs on all Brocade B-Series switches, that can quickly “repair” the network to overcome most failures. For example, when a link between switches fails, FSPF quickly recalculates all traffic flows. Of course, this assumes that there is a second route, which is when redundancy in the fabric becomes important.

The key to high availability and enterprise-class installation is redundancy. By eliminating a single point of failure, business continuance can be provided through most foreseeable and even unforeseeable events. At the highest level of fabric design, the complete network should be redundant, with two completely separate fabrics that do not share any network equipment (routers or switches).

Servers and storage devices should be connected to both networks utilizing some form of multipath I/O (MPIO) solution, such that data can flow across both networks seamlessly in either an active/active or active/passive mode. MPIO ensures that if one path fails, an alternative path is readily available. Ideally, the networks would be identical, but at a minimum they should be based on the same switch architecture to ensure consistency of performance. In some cases, these networks are in the same location. However, in order to provide for disaster recovery (DR), two separate locations are often used, either for each complete network or for sections of each network.

Regardless of the physical geography, there are two separate networks for complete redundancy.

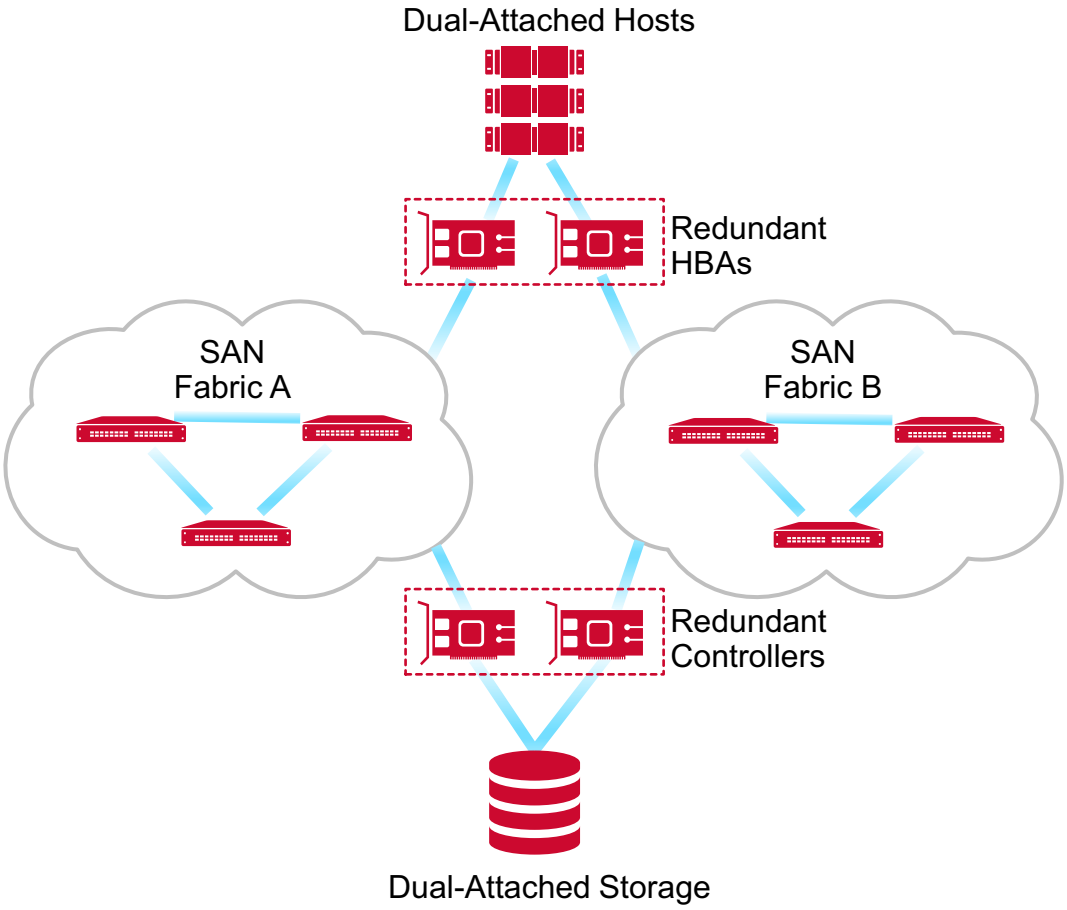
In summary, best practices for SAN design are to ensure application availability and resiliency via the following:

- Redundancy built into fabrics to avoid a single point of failure
- Servers connected to storage via redundant fabrics
- MPIO-based failover from server to storage
- Redundant fabrics based on similar architectures
- Redundant ISLs/ICLs for interswitch connectivity
- Separate storage and server tiers for independent expansion
- Core switches of equal or higher performance compared to the edge switches
- The highest performance switch in the fabric defined to be the principal switch

4.4 Switch Interconnections

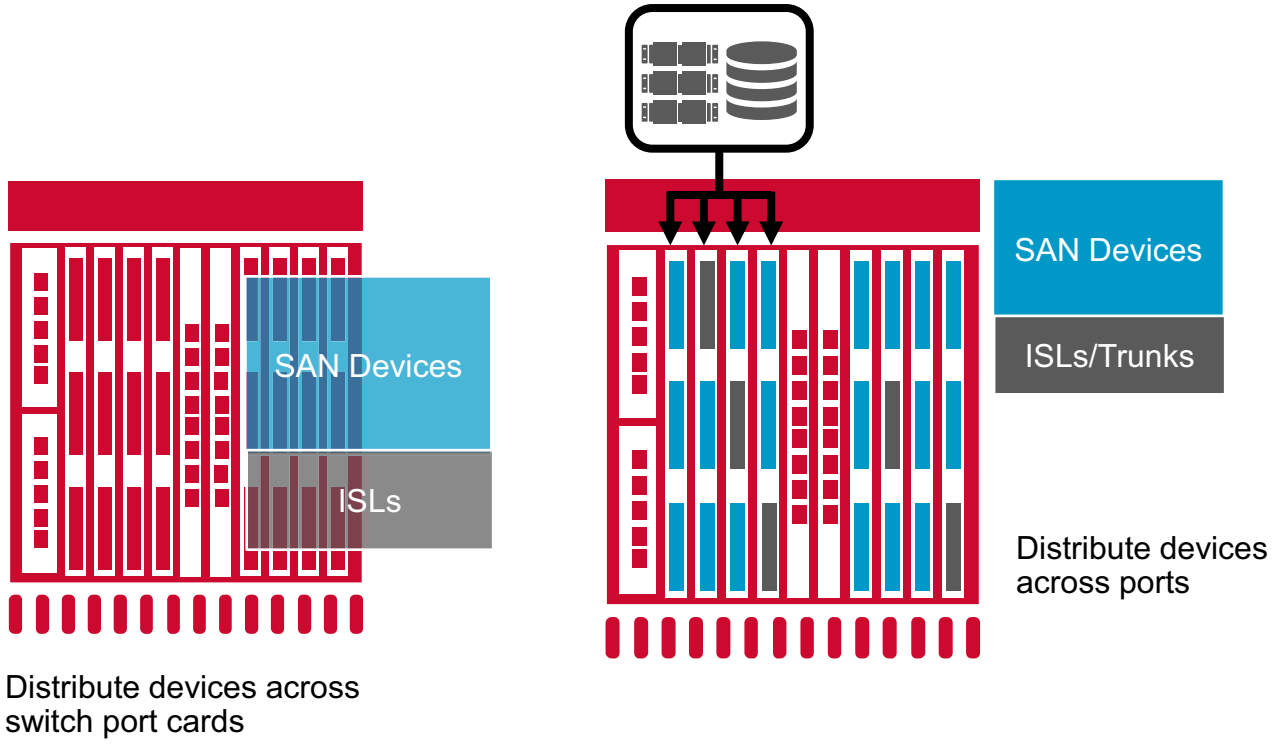
As mentioned previously, there should be at least two of every element in the SAN to provide redundancy and improve resiliency. The number of available ports and device locality (server/storage tiered design) determine the number of ISLs needed to meet performance requirements. This means that there should be a minimum of two trunks, with at least two ISLs per trunk. Each source switch should be connected to at least two other switches, and so on. In Figure 3, each of the connection lines represents at least two physical cable connections.

Figure 3: Connecting Devices through Redundant Fabrics



In addition to redundant fabrics, redundant links should be placed on different blades, different ASICs, or at least different port groups whenever possible, as shown in Figure 4. (See the appropriate hardware manual to determine trunk groups for the various port blades. For more details, refer to the *Brocade Fabric OS Administration Guide*.) Whatever method is used, it is important to be consistent across the fabric. For example, do not place ISLs on lower port numbers in one chassis (as shown in the left diagram in Figure 4) and stagger them in another chassis (as shown in the right diagram in Figure 4). Doing so would be mismatched ISL placement.

Figure 4: Examples of Distributed ISL Placement for Redundancy



NOTE: In Figure 4, ISL trunks are placed on separate ASICs or port groups. It is important to match ISL placement between devices and across fabrics to ensure simplicity in design and assist in problem determination.

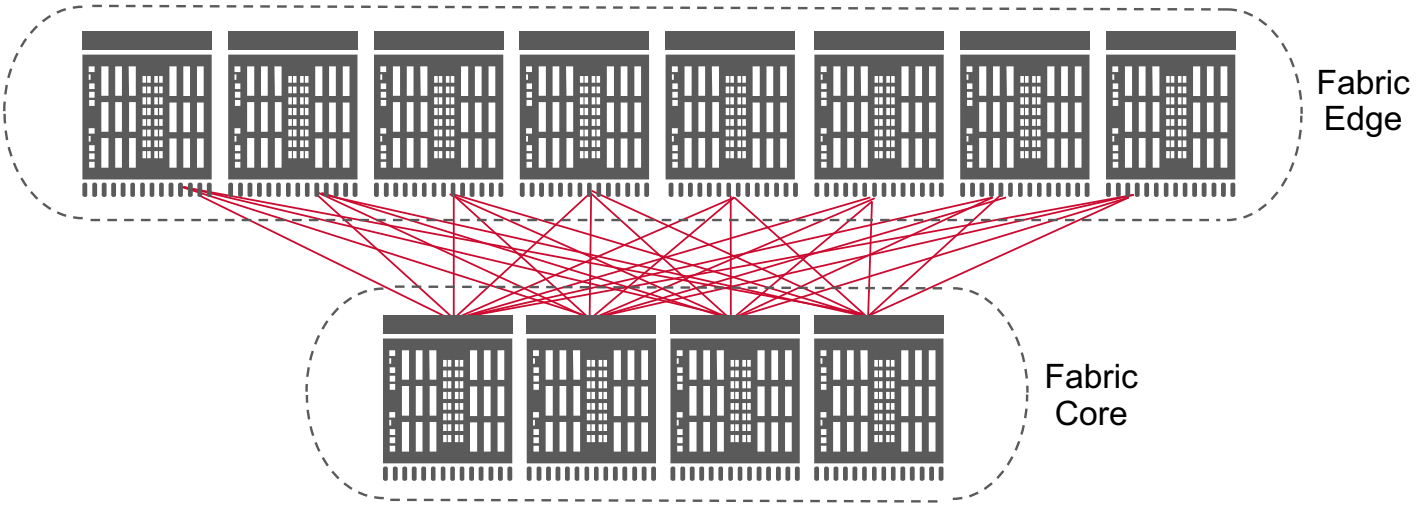
4.4.1 UltraScale ICL Connectivity for Gen 5 Brocade DCX® 8510-8/8510-4, Gen 6 Brocade X6-8/X6-4, and Gen 7 Brocade X7-8/X7-4

The Brocade DCX® 8510, X6, and X7 platforms use second-generation UltraScale ICL technology from Brocade with optical QSFPs. The Brocade DCX 8510-8, X6-8, and X7-8 support up to 32 QSFP ports per chassis (Figure 5), and the Brocade DCX 8510-4, X6-4, and X7-4 support up to 16 QSFP ports to help preserve director ports for connections to end devices. Each QSFP port actually has four independent links, each of which terminates on a different ASIC within the core blade.

NOTE: X6 ICL ports can also connect to DCX 8510 ICL ports at 16-Gb speeds with the port speed on the X6 ICL configured to 16 Gb and use of the 16-Gb QSFP on the attached X6 port.

NOTE: X7 ICL ports can also connect to DCX 8510 ICL ports at 16-Gb speeds with the port speed on the X7 ICL configured to 16 Gb and use of the 16-Gb QSFP on the attached X7 port.

Figure 5: 12-Chassis UltraScale ICL-Based Core-Edge Design

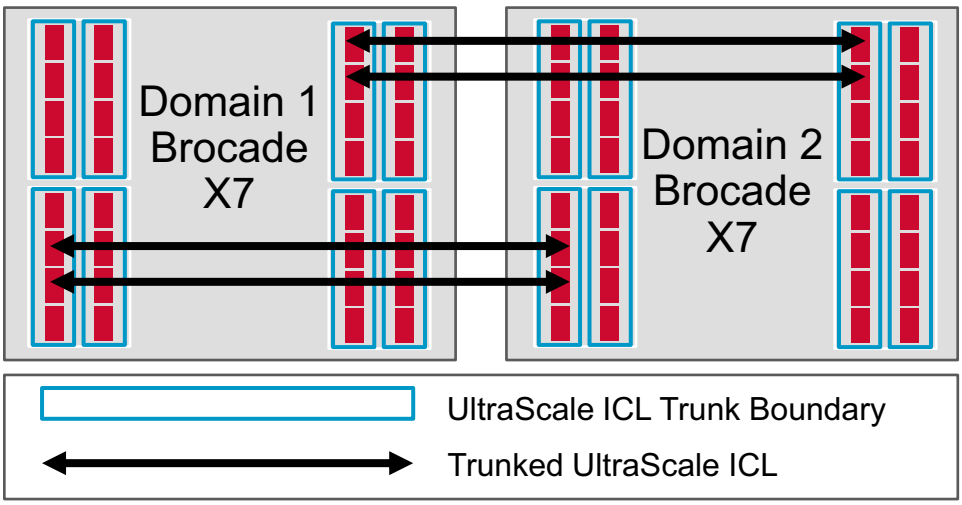


4.5 Best Practices for Brocade UltraScale ICL Connections

Each core blade in a chassis must be connected to each of the two core blades in the destination chassis to achieve full redundancy (Figure 6).

NOTE: For redundancy, use at least one pair of links between two core blades.

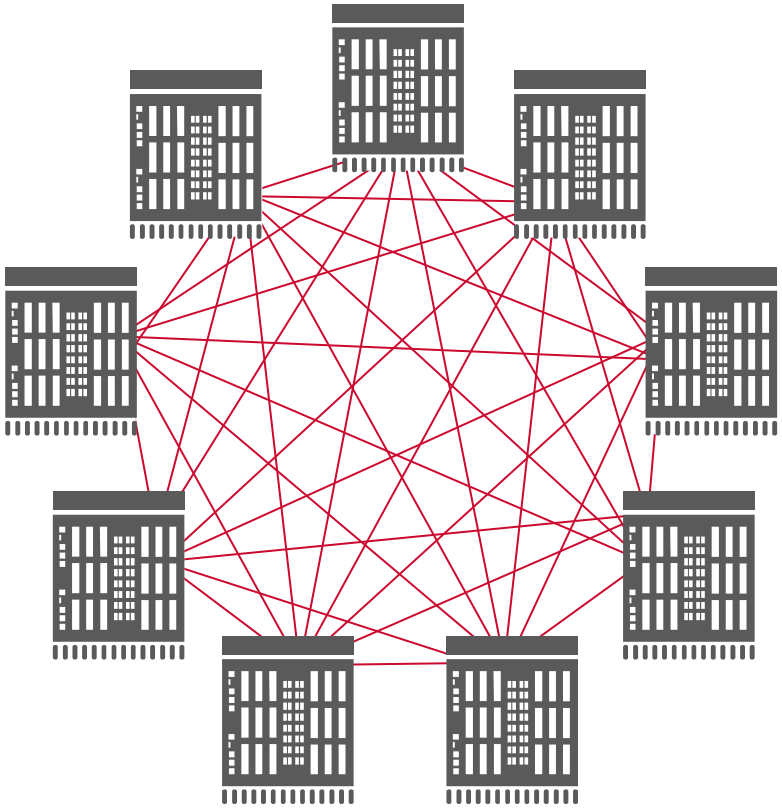
Figure 6: Minimum ICL Connections Needed between Brocade X7 Chassis



4.6 Mesh Topology

A mesh design provides a single hop between source and destination. Beginning with FOS 7.3.0x, Brocade supports a 9-chassis mesh design with up to 100-meter distances using select QSFPs and OM4 fiber. In the configuration shown in Figure 7, up to 1152 16Gb/s ports are supported using UltraScale ICLs with a 12:1 oversubscription. As more UltraScale ICLs are added, oversubscription can be reduced to 3:1.

Figure 7: 9-Chassis UltraScale ICL-Based Full-Mesh Topology



NOTE: Refer to the *Scale-Out Architecture with Brocade UltraScale Inter-Chassis Links Design Guide* for details. UltraScale ICL connections are considered a “hop of no concern” in a FICON fabric.

When using core-edge SAN design methodologies, edge switches should connect to at least two core switches with trunks of at least two ISLs each. Each of those trunks should be attached to a different blade/port group. In order to be completely redundant, there would be a completely mirrored second fabric and devices must be connected to both fabrics using MPIO.

Recommendations for switch ISL/UltraScale ICL connectivity follow:

- There should be at least two core switches.
- Every edge switch should have at least two trunks to each core switch.
- Select small trunk groups (keep trunks to two ISLs) unless you anticipate very high traffic volumes. This ensures that you can lose a trunk member without losing ISL connectivity.
- Place redundant links on separate blades.
- Trunks should be in a port group (ports within an ASIC boundary).
- Allow no more than 30m in cable difference for optimal performance for ISL trunks.

- Use the same cable length for all UltraScale ICL connections.
- Use either ISL or UltraScale ICL connectivity into the same domain. Mixing the two types of connections is not supported.
- Use the same type of optics on both sides of the trunks: Short Wavelength (SWL), Long Wavelength (LWL), or Extended Long Wavelength (ELWL).

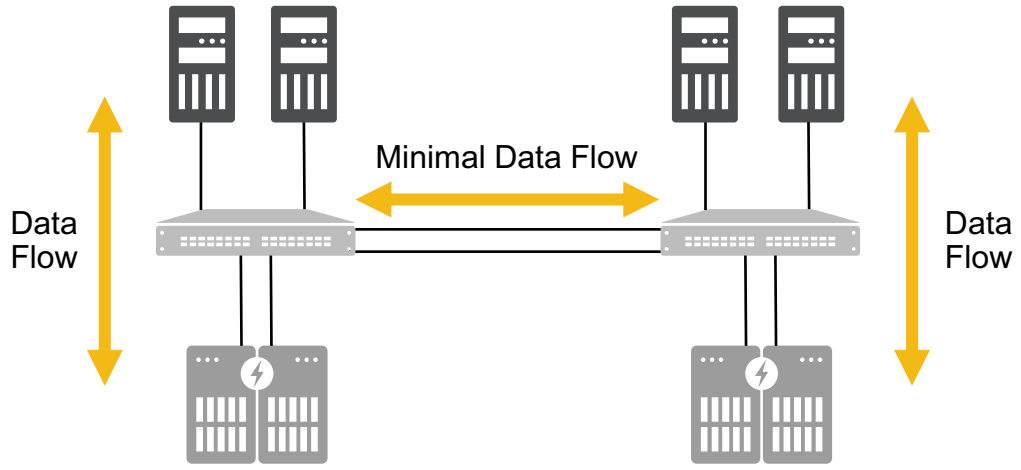
4.7 Device Placement

Device placement is a balance between traffic isolation, scalability, manageability, and serviceability. With the growth of virtualization and multinode clustering on the UNIX platform, frame congestion can become a serious concern in the fabric if there are interoperability issues with the end devices.

4.7.1 Traffic Locality

Designing device connectivity depends a great deal on the expected data flow between devices. For simplicity, communicating hosts and targets can be attached to the same switch (Figure 8).

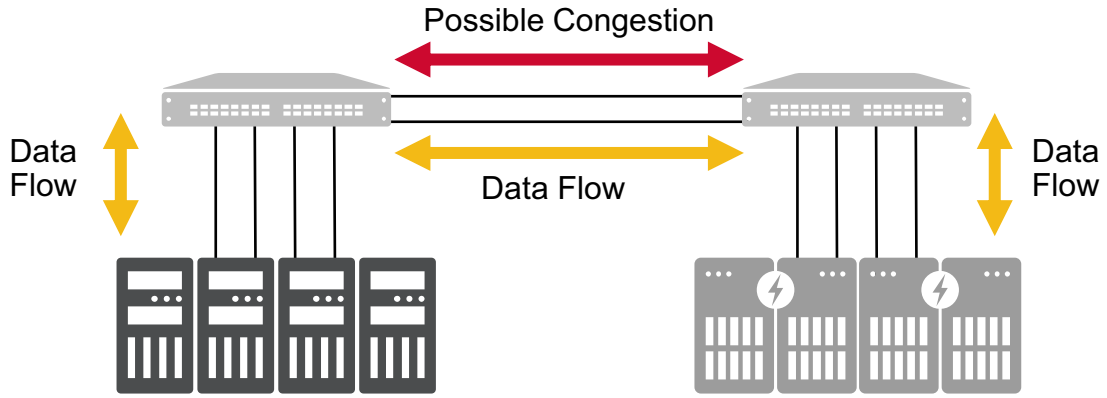
Figure 8: Hosts and Targets Attached to the Same Switch to Maximize Locality of Data Flow



However, this approach does not scale well. Given the high-speed, low-latency nature of Fibre Channel, attaching these host-target pairs on different switches does not mean that performance is adversely impacted for common workloads.

Although traffic congestion is possible, it can be mitigated with proper provisioning of ISLs/UltraScale ICLs. With current generation switches, locality is not required for performance or to reduce latencies. For mission-critical applications that depend on extremely fast response times, architects may want to localize the traffic when using flash storage or in very exceptional cases, particularly if the number of ISLs available is restricted or there is a concern for resiliency in a multihop environment (Figure 9).

Figure 9: Hosts and Targets Attached to Different Switches for Ease of Management and Expansion



One common scheme for scaling a core-edge topology is dividing the edge switches into a storage tier and a host/initiator tier. This approach lends itself to ease of management as well as ease of expansion. In addition, host and storage devices generally have different performance requirements, cost structures, and other factors that can be readily accommodated by placing initiators and targets in different tiers.

Chapter 5: Data Flow Considerations

5.1 Fan-In Ratios and Oversubscription

A critical aspect of data flow is the *fan-in ratio* or *oversubscription* in terms of source ports to target ports and devices to ISLs. This is also referred to as the *fan-out ratio* if viewed from the storage array perspective. The ratio is the number of device ports that share a single port, whether ISL, UltraScale ICL, or target. This is always expressed from the single entity point of view, such as 7:1 for seven hosts utilizing a single ISL or storage port.

What is the optimum number of hosts that should connect to a storage port? This seems like a fairly simple question. However, once you take into consideration clustered hosts, VMs, workload characteristics, and the number of logical unit numbers (LUNs) (storage) per server, the situation can quickly become much more complex. Determining how many hosts to connect to a particular storage port can be narrowed down to three considerations: port queue depth, I/O per second (IOPS), and throughput. Of these three, throughput is the only network component. Thus, a simple calculation is to add up the expected peak bandwidth usage for each host accessing the storage port.

In practice, however, it is highly unlikely that all hosts perform at their maximum level at any one time. With a traditional application-per-server deployment, the host bus adapter (HBA) bandwidth is overprovisioned. However, with virtual servers (KVM, Xen, Hyper-V, proprietary UNIX OSs, and VMware), the game can change radically. Network oversubscription is built into the virtual server concept. To the extent that servers leverage virtualization technologies, you should reduce network-based oversubscription proportionally. It may therefore be prudent to oversubscribe ports to ensure a balance between cost and performance.

Another method is to assign host ports to storage ports based on the I/O capacity requirements of the host servers. The intended result is a small number of high-capacity servers or a larger number of low-capacity virtual servers assigned to each storage port, thus distributing the load across multiple storage ports.

Regardless of the method used to determine the fan-in/fan-out ratios, port monitoring should be used to determine actual utilization and what adjustments, if any, should be made. In addition, ongoing monitoring provides useful heuristic data for effective expansion and efficient assignment of existing storage ports. For determining the device-to-ISL fan-in ratio, a simple calculation method works best: the storage port should not be oversubscribed into the core (for example, a 32Gb/s storage port should have a 32Gb/s pipe into the core).

Chapter 6: Scalability and Performance

Brocade products are designed with scalability in mind, knowing that most installations will continue to expand and that growth is supported with very few restrictions. However, following the same basic principles outlined in previous sections as the network grows will ensure that the levels of performance and availability will continue.

Evaluate the impact on topology, data flow, workload, performance, and perhaps most importantly, redundancy and resiliency of the entire fabric any time one of the following actions is performed:

- Adding or removing initiators:
  - Changes in workload
  - Changes in provisioning
- Adding or removing storage:
  - Changes in provisioning
  - Changes in storage media type (for example, increased deployment of flash-based storage)
- Adding or removing switches
- Adding or removing ISLs and ICLs
- Change in virtualization (workload and storage) strategies and traffic flow pattern

If these design best practices are followed when the network is deployed, then small incremental changes should not adversely impact the availability and performance of the network. However, if changes are ongoing and the fabric is not properly evaluated and updated, then performance and availability can be jeopardized. Some key points to cover when looking at the current status of a production FC network include:

Reviewing redundancy and resiliency:

- Are there at least two physically independent paths between each source and destination pair?
- Are there two redundant fabrics?
- Does each host connect to two different edge switches?
- Are edge switches connected to at least two different core switches?
- Are interswitch connections composed of two trunks of at least two ISLs?
- Does each storage device connect to at least two different edge switches or separate port blades?
- Are storage ports provisioned such that every host has at least two ports through which it can access LUNs?
- Are redundant power supplies attached to different power sources?
- Are zoning and security policies configured to allow for patch/device failover?

Reviewing performance requirements:

- Host-to-storage port fan-in/out ratios
- Oversubscription ratios:
  - Host to ISL
  - Edge switch to core switch
  - Storage to ISL
- Size of trunks
- Routing policy and currently assigned routes (evaluate actual utilization for potential imbalances)
- Use of FEC for all ISLs and connections to Gen 5 (if supported) and Gen 6 devices

Watching for latencies because of the following:

- Poor storage performance
- Overloaded hosts or applications
- Distance issues over constrained long-distance links resulting from changes in usage, such as adding mirroring or too many workloads
- Deteriorating optics resulting in declining signal strength and increased error rate

In Gen 6 and Gen 7 networks, storage response latency can be baselined and monitored continuously using IO Insight in conjunction with MAPS. Deal with latencies immediately; they can have a profound impact on the fabric.

In summary, although Brocade SANs are designed to allow for any-to-any connectivity and they support provision-anywhere implementations, these practices can have an adverse impact on the performance and availability of the SAN if left unchecked. As detailed above, the network needs to be monitored for changes and routinely evaluated for how well it meets desired redundancy and resiliency requirements.

Chapter 7: Supportability

Supportability is a critical part of deploying a SAN. Follow the guidelines in this chapter to ensure that the data needed to diagnose fabric behavior or problems has been collected.

- Configure Brocade MAPS monitoring: Leverage Brocade MAPS to implement proactive monitoring of errors and warnings such as CRC errors, loss of synchronization, and high-bandwidth utilization.
- Configure syslog forwarding: By keeping historical log messages and having all switch messages sent to one centralized syslog server, troubleshooting can be expedited and simplified. Forwarding switch error messages to one centralized syslog server and keeping historical log messages enables faster and more effective troubleshooting and provides simple monitoring functionality.
- Create a switch configuration template in SANnav to avoid configuration drift from occurring over time. You can easily adopt existing configurations as a template for deploying new switches in the fabric, ensuring consistency across the data center.
- Follow Brocade best practices in the LAN infrastructure for management interfaces: Brocade best practice in the LAN infrastructure is to set up different physical LAN broadcast segments, for example, by placing IP routers between segments or configuring different VLANs for the management interfaces of two fabric switches.
- Enable audit functionality: To provide audit functionality for the SAN, keep track of which administrator made which changes, usage of multiple user accounts (or RADIUS), and configuration of change tracking or audit functionality (along with use of errorlog/syslog forwarding).
- Configure multiple user accounts (LDAP/OpenLDAP or RADIUS): Make mandatory the use of personalized user accounts as part of the IT/SAN security policy, so that user actions can be tracked. Also, restrict access by assigning specific user roles to individual users.
- Establish a test bed: Set up a test bed to test new applications, firmware upgrades, driver functionality, and scripts to avoid missteps in a production environment. Validate functionality and stability with rigorous testing in a test environment before deploying into the production environment.
- Implement a serial console server: Implement serial remote access so that switches can be managed even when there are network issues or problems during switch boot or firmware upgrades.
- Use aliases: Using aliases to give switch ports and devices meaningful names can lead to faster troubleshooting.
- Configure `supportftp`: Configure `supportftp` for automatic file transfers. The parameters set by this command are used by `supportSave` and `traceDump`.
- Configure an NTP server: To keep a consistent and accurate date and time on all the switches, configure switches to use an external time server.

7.1 Firmware Upgrade Considerations

Both fixed-port and modular switches support hot code load for firmware upgrades.

- Disruptive versus nondisruptive upgrades:
  - Simultaneous upgrades on neighboring switches
  - Standard FC ports versus application and special-feature ports
- Review the *Brocade Fabric OS Release Notes* for the following:
  - Upgrade path
  - Changes to feature support
  - Changes to backward compatibility
  - Known issues and defects
- Consider a separate AG firmware upgrade strategy. Brocade Access Gateways have no fundamental requirement to be at the same firmware release level as Brocade FOS. Upgrading only directors and switches minimizes the infrastructure changes required during an upgrade cycle.

Chapter 8: Monitoring

8.1 Brocade Fabric Vision Technology

Organizations face a constant struggle to both manage data growth and deliver actionable intelligence from raw data—all while meeting SLAs. As a result, even well-managed IT organizations must often make difficult choices about resource allocation, weighing the benefits of focusing more resources on monitoring, for instance, and fewer resources on planning or optimizing. With Brocade Fabric Vision technology, organizations can achieve unprecedented insight and visibility across the storage network through critical monitoring and diagnostic capabilities.

8.1.1 Monitoring and Alerting Policy Suite

Monitoring and Alerting Policy Suite (MAPS) is a health and threshold monitoring tool that allows for autonomous self-monitoring of directors and switches in the fabric. It helps detect potential and active problems, automatically alerting users to those problems long before they become costly outages. MAPS is a part of the Brocade Fabric Vision feature set and is available with FOS 7.2.0 and above.

MAPS tracks a variety of SAN fabric health categories and events. Monitoring fabric-wide events, ports, bit errors, and environmental parameters enables early fault detection and isolation as well as a means to measure performance. All health monitoring categories are customizable, providing flexibility around how and what users want to monitor. Create your own monitoring groups, assign custom thresholds, and with FOS 9.0 and above gain the same monitoring capabilities at a flow level. This means users can now threshold monitor application flows for abnormal completion times to stay on top of SLAs. Users can also easily integrate MAPS with enterprise system management solutions.

MAPS also provides predefined monitoring policies for users to start off with. These policies provide thresholds derived from 20 years of best practices and customer experiences. Based on how closely users want to monitor their SAN environment, they can select from conservative, moderate, or aggressive policies. If the default policies do not meet your monitoring needs, simply customize the thresholds and actions of interest and activate the custom policy. MAPS provides notifications before problems arise, such as reporting on overutilized ports approaching the specified bandwidth limit, potentially leading to congestion. These insights enable SAN administrators to perform pre-emptive network maintenance, such as trunking or zoning, to avoid potential network failures.

MAPS also lets you define how often switches and fabric elements are measured while specifying notification thresholds. Whenever fabric elements exceed these thresholds, MAPS can automatically take actions. These actions include administrative notification using email messages, SNMP traps, RASLog entries, and automated actions, such as slow drain device quarantine and unquarantine and FPIN notifications.

8.1.1.1 MAPS Recommendations

Brocade MAPS is a recommended optional feature that provides threshold monitoring of multiple switch elements. For instance, MAPS monitors port groups based on the port type, allowing for different thresholds to be set within those port groups and for the port groups to be monitored simultaneously. Different port types (for example, F_Ports, E_Ports, N_Ports) tend to differ in characteristics. MAPS provides the flexibility in monitoring and alerting capabilities to address a wide variety of cases.

MAPS also allows for the monitoring and alerting of IO Insight flow metrics, providing SAN admins with notifications of performance degradation and an early alert into developing congestion issues that may impact storage response times. When support for VM Insight is enabled, potential issues can be identified end to end from the individual virtual machine to the LUN to which it is communicating.

8.1.1.2 Tips on Getting Started with MAPS

Are you new to Brocade Monitoring and Alerting Policy Suite and looking to get started with the autonomous monitoring of your SAN environments? MAPS will provide deep insights into the health of your SAN and its performance with a single click. The following are some quick tips on the initial use of MAPS. Note that in order to take advantage of all monitoring capabilities, a Fabric Vision license is required, which enables over 300 additional rules.

When starting off with MAPS, SAN admins should begin monitoring their fabric with one of the three predefined policies (Conservative, Moderate, Aggressive), ideally the dflt_conservative_policy. This policy will allow SAN admins to gain a better understanding of what MAPS monitors for, the severity of thresholds set, and the alerts being generated. If the conservative policy is not meeting your monitoring needs, change to the Moderate and/or Aggressive policies. SAN admins can then begin personalizing any of the default policies with their own observed thresholds and desired actions to better fit their SAN environments.

SAN admins can achieve policy customization and management through Brocade SANnav Management Portal or the CLI. The following are some examples of customization.

- Clone predefined policies for customizations of individual thresholds and rules.
- Create custom monitoring groups (for example, ports, SFPs, application flows).
- Distribute policies across the SAN for uniform fabric monitoring.
- Configure MAPS actions and take advantage of automated problem mitigation.
- Create custom MAPS monitoring dashboards through SANnav Management Portal.

8.1.2 Fabric Performance Impact Monitoring

Fabric Performance Impact (FPI) monitoring leverages predefined MAPS policies to automatically detect and alert administrators to the severity of latency and identify slow drain devices that could impact network performance. This feature enables the detection of various latency severity levels, pinpointing exactly which devices are causing back pressure in the fabric and/or are being impacted by a bottleneck port. MAPS and FPI then work together to automatically quarantine the slow drain devices, preventing buffer credit starvation.

8.1.3 Slow Drain Device Quarantine/Unquarantine Explained

When a device is experiencing congestion due to lost credits, credit stall, or oversubscription, users can make use of the automated Slow Drain Device Quarantine (SDDQ) and Unquarantine (UNQUAR) actions available through MAPS in order to mitigate any back pressure in the fabric, which can potentially affect neighboring traffic flows and in turn degrade performance.

SDDQ works together with MAPS and FPI monitoring to detect different congestion types and isolate problematic devices into their own low-priority virtual channel (VC) that has its own resources. Traffic in a Brocade fabric typically runs by default in normal-priority VCs. The isolation of the slow drain device happens automatically and nondisruptively based on performance impacts, frame loss, and oversubscription being experienced in the fabric. Note that MAPS actions can be individually enabled for any of these conditions. Once the problematic traffic flow is isolated, the back pressure in the fabric is relieved, freeing up buffer credits and link bandwidth for flows in the normal-priority VCs.

FPI Monitoring is also constantly checking for when the congestion conditions are cleared or are no longer being experienced on the flagged devices; in turn, allowing MAPS to take the automatic unquarantine action. The MAPS unquarantine action is able to move the traffic flow, which now has cleared congestion states, back into the normal-priority VCs. This process, similar to the quarantine action, is also nondisruptive to all flows.

The SDDQ and UNQUAR MAPS actions are supported on devices that are sitting on the local switch and the remote switch (attached via ISLs), as well as on devices attached through Brocade Access Gateways.

8.1.4 Flow Vision

Flow Vision was designed as a diagnostics tool and is supported on all Brocade SAN platforms running Fabric OS 7.2 and later. Flow Vision provides the SAN administrator with visibility into fabric traffic flows and with the ability to copy traffic flows for later analysis. Flow Vision also allows for test-flow generation at line-rate speeds to prevalidate SAN hardware performance and connectivity. Use the flow generation capability before operational deployment where possible to confirm optimal health and the ability to support spikes in throughput.

For the most mission-critical applications, consider running Flow Vision constantly to keep a historical record of application performance profiles and intermittent irregularities. For frequent callers such as critical application owners, run Flow Vision on a regular basis when time permits to verify good fabric health and look deeper into issues.

8.1.5 IO Insight

IO Insight, also known as Flow Monitor, is a capability supported by Brocade's Gen 6 and Gen 7 Fibre Channel switching products that provides even deeper flow-level IO statistics. These statistics include storage device latency and IOPS metrics such as first IO response time, IO completion time, and number of pending IOs for a specific host and target or target and LUN, providing IO workload monitoring and early detection of storage performance degradation.

These IO Insight metrics should be added into MAPS policies and dashboards for notification of storage response time and performance degradation. This reporting is of tremendous value for performance-sensitive workloads, enabling administrators to meet their critical SLAs. IO Insight should be monitored for storage devices that support those critical apps to provide feedback to application and storage administrators on performance over time on device reliability and performance optimization. For example, pending IOs will measure the current queue depth of an HBA and can be used to fine-tune the server queue depth configuration.

Beginning with Fabric OS 9.0, IO Insight autonomously learns all flows that are traversing a switch with no user configuration required as it is enabled by default. Once switches are discovered via SANnav Management Portal, telemetry data is automatically propagated to the management platform, which can then be utilized for flow-level and application-level investigation.

For configuration and usage details on Flow Vision and IO Insight, refer to the *Brocade Fabric OS Flow Vision User Guide*.

8.1.6 VM Insight

The VM Insight feature essentially provides the same IO and performance-level metrics that IO Insight provides but for individual virtual machines. This feature can logically distinguish individual VM flows all the way down to the LUN to which they are communicating even if other VMs are sharing the same LUN. This allows for unprecedented visibility for monitoring the health and performance of applications that are running on virtual machines.

VM Insight also integrates with MAPS, allowing users to threshold monitor and alert on VM-level flow-performance metric deviations, similar to IO Insight/Flow Monitor.

This feature is available on Gen 6 platforms that are running Fabric OS 8.1 and later as well as on all Gen 7 platforms.

8.2 SANnav Management Portal Monitoring Overview

SANnav Management Portal is Brocade's latest management platform, which tightly integrates with all Fabric OS feature sets; from the configuration of features down to the analysis of gathered telemetry data and events, providing actionable insights to SAN admins.

Brocade takes full advantage of Fabric OS features in SANnav to provide device health details, identify various types of congestion, collect flow telemetry data for detailed investigation and troubleshooting, and customize dashboards to monitor entire SAN environments. See Section 16.1.2, SANnav Management Portal, in this document for more detail around the management platform monitoring capabilities.

8.3 Troubleshooting

8.3.1 ClearLink Diagnostics (D_Port) — Predeployment Cabling and Optics Validation

For SANs built with Brocade Gen 5, Gen 6, or Gen 7 Fibre Channel switches equipped with 16-Gb or higher optics, Brocade ClearLink Diagnostics enables the use of predeployment testing to validate the integrity of the physical network infrastructure before operational deployment. Part of Brocade Fabric Vision technology, ClearLink is an offline diagnostics tool that allows users to perform an automated suite of tests to measure and validate maximum throughput speeds as well as latency and distance across fabric links. ClearLink Diagnostics can also be used to verify the health and integrity of all 16Gb/s and 32Gb/s SFP+ transceivers as well as 32Gb/s QSFP transceivers in the fabric on a one-by-one basis. Diagnostics should be conducted before deployment or when there is an excessive count of CRC errors that may be caused by physical-layer issues.

A ClearLink Diagnostics port (D_Port) requires that only the individual ports attached to the tested link go offline, allowing the remainder of the ports to stay online in isolation from the link. ClearLink can also be used to test links to a new fabric switch without allowing the new switch to join or even be aware of the current fabric, providing an opportunity to measure and test ISLs before they are put into production. This fabric-based, physical-layer validation enables the following:

- Transceiver health check
- Transceiver uptime
- Local and long-distance measurements (5-meter granularity for 16Gb/s and 32Gb/s Small Form-factor Pluggable [SFP] or 32Gb/s Quad Small Form-factor Pluggable [QSFP] optics and 50-meter granularity for 10Gb/s SFP optics)
- Link latency measurements between D_Ports
- Link power (dB) loss
- Link performance

8.3.2 Recommendation: D_Port On-Demand

When an on-demand D_Port-capable switch or chassis comes online, the switch checks if the other end of the connection supports dynamic D_Port mode. If dynamic D_Port is supported on the other end, the switch changes the remote port to D_Port mode and then triggers diagnostic tests automatically. The D_Ports change to normal port mode after successful completion of the tests.

For Brocade ClearLink Diagnostics guidelines and restrictions, refer to the *Brocade Fabric OS Troubleshooting and Diagnostics User Guide* for a more detailed discussion of diagnostic port usage.

8.3.3 Forward Error Correction

Forward error correction (FEC) provides a data transmission error control method by including redundant data (error correcting code) to ensure error-free transmission on a specified port or port range. FEC supports the following data transmissions:

- When 10/16G FEC is enabled, it can correct one burst of up to 11-bit errors in every 2112-bit transmission, whether the error is in a frame or a primitive.
- When 32G FEC is enabled, it can correct up to 7 symbols in every 5280-bit transmission. A symbol consists of 10 bits, so there are 528 symbols in every 5280-bit transmission.
- When 64G FEC is enabled, it can correct up to 15 symbols in every 5440-bit transmission. A symbol consists of 10 bits, so there are 544 symbols in every 5440-bit transmission.

Because FEC is optional at 10Gb/s and 16Gb/s speeds, the Transmitter Training Signal (TTS) was extended to include a means to negotiate FEC capabilities. FEC is negotiated and activated when both sides of the link have FEC enabled. The FEC active indicator in Fabric OS indicates whether FEC was successfully negotiated. FEC uses unused bits within the signaling protocol to generate an error correcting code (ECC) and correct bits as needed.

For FEC configuration options and limitations, refer to the *Brocade Fabric OS Administration Guide*.

8.3.4 Buffer Credit Loss Detection and Recovery

Enable both credit loss detection and recovery on your Brocade platforms since it is disabled by default. This enables Brocade hardware to detect and automatically recover any lost credits that occur on backend ports with no user interaction.

Buffer credit recovery allows links to recover after buffer credits are lost when the buffer credit recovery logic is enabled. The buffer credit recovery feature also maintains performance. If a credit is lost, a recovery attempt is initiated. During link reset, the frame and credit loss counters are reset without performance degradation.

Credit recovery is supported on E_Ports, F_Ports, and EX_Ports. Buffer credit recovery is enabled automatically across any long-distance connection for which the E_Port, F_Port, or EX_Port buffer credit recovery mechanism is supported.

8.3.5 RASLog Messages

RASLog messages report significant system events (failure, error, or critical conditions) or information and are also used to show the status of high-level user-initiated actions. RASLog messages are forwarded to the console, to the configured syslog servers, and to the SNMP management station through Simple Network Management Protocol (SNMP) traps or informs. For instance, SANnav Management Portal can also be used a receiver.

The following are the severity levels for messages and their descriptions:

- 1 = CRITICAL

  Critical-level messages indicate that the software has detected serious problems that will cause a partial or complete failure of a subsystem if not corrected immediately; for example, a power supply failure or a rise in temperature must receive immediate attention.
- 2 = ERROR

  Error-level messages represent an error condition that does not impact overall system functionality significantly. For example, error-level messages might indicate timeouts on certain operations, failures of certain operations after retries, invalid parameters, or failure to perform a requested operation.

- 3 = WARNING

  Warning-level messages highlight a current operating condition that should be checked or it may lead to a failure in the future. For example, a power supply failure in a redundant system relays a warning that the system is no longer operating in redundant mode unless the failed power supply is replaced or fixed.
- 4 = INFO

  Info-level messages report the current nonerror status of the system components; for example, detecting the online and offline status of a fabric port.

8.3.6 Audit Log Messages

Event auditing is designed to support post-event audits and problem determination based on high-frequency events of certain types such as security violations, zoning configuration changes, firmware downloads, and certain types of fabric events. Audit messages that are flagged only as AUDIT are not saved in the switch error logs. The switch can be configured to stream audit messages to the switch console and to forward the messages to specified syslog servers. The audit log messages are not forwarded to an SNMP management station. There is no limit to the number of audit events.

For any given event, audit messages capture the following information:

- User Name: The name of the user who triggered the action.
- User Role: The access level of the user, such as root or admin.
- Event Name: The name of the event that occurred.
- Event Information: Information about the event.

8.4 Monitoring the Switches

You should seriously consider implementing some form of monitoring of each switch. Issues often start out relatively benign and gradually degrade into more serious problems. Monitoring the logs for warning, critical, and error severity messages will go a long way in avoiding many problems.

- Plan for a centralized collection of RAS log and perhaps Audit log via syslog. You can optionally filter these messages relatively easily through some simple scripting programs, or you can perform advanced correlation using an event management engine.
- Brocade platforms are capable of generating SNMP traps for most error conditions. Consider implementing some sort of alerting mechanism via SNMP or email notifications.

8.5 Latencies

Latencies have many causes:

- Slow devices such as disk-based storage arrays
- Oversubscribed devices
- Long-distance links
- Servers that are not responding rapidly enough to I/O requests that they have previously made
- Degraded cables and SFPs that cause many retried I/Os

Very little can be done in the fabric to accommodate end-device latencies; they typically must be addressed through other means. Array latencies can be dealt with by array or LUN reconfiguration or data migration. Long-distance problems might require more long-distance bandwidth or reconfiguration of the distance setting on the switch. Applications might require tuning to improve their performance, and failing links and SFPs must be identified and replaced. At best, the fabric can help identify the source of the problem. Brocade has been working hard to enhance RAS features in Brocade FOS in line with changing customer requirements. Some of these features are described briefly in the sections that follow.

8.6 Misbehaving Devices

All fabrics, regardless of the equipment vendor, are vulnerable to the effects of badly behaving devices, that is, a server or storage device that for some reason stops functioning or starts flooding the fabric with data or control frames. The effects of such behavior can be very severe, causing other applications to fail over or even stop completely. The fabric can do nothing to anticipate this behavior. Brocade has implemented several new features that are designed to rapidly detect a misbehaving device and isolate it from the rest of the fabric.

Isolating a single server has much less impact on applications than disabling a storage array port. Typically, a storage port services many applications, and the loss of that storage can severely impact all the applications connected to it. One of the advantages of a core-edge design is that it is very simple to isolate servers from their storage and ensure that any action applied to a host port for a given behavior can be very different than the action applied to a storage port for the same behavior.

Detailed guidance on monitoring for misbehaving devices and configuring fabrics to respond to developing issues can be found in the *SAN Fabric Resiliency and Administration Best Practices User Guide*.

8.7 Design Guidelines

- **Transaction-based systems**: Make sure that ISL/UltraScale ICLs that are traversed by transaction-based systems to access their storage do not contain too many flows. The fan-in from the hosts/initiators should not exceed a ratio of 10 to 1. Also ensure that there is as little interference from other applications as possible so that latencies and congestion from other sources do not affect the overall performance of the applications.
- **I/O-intensive applications**: Bandwidth is the most common constraint for I/O-intensive applications. Modern fabrics typically provide more bandwidth than is needed except for the most powerful hosts. Take care to ensure that these high-performing systems do not interfere with other applications, particularly if utilization spikes at specific times or if batch runs are scheduled. When in doubt, add more paths (ISLs or trunks) through the fabric.
- **Clusters**: Clusters often have behavioral side effects that must be considered. This is particularly true during storage provisioning. It is possible, for example, for a cluster to inundate the fabric and storage arrays with LUN status queries and other short frame requests. This behavior can cause frame congestion in the fabric and can stress the control processors of the arrays. Make sure that you spread out the LUNs accessed by the hosts in the cluster across as many arrays as possible.
- **Congestion**: Traffic congestion (total link capacity regularly consumed) is remedied by adding more links or more members to a trunk. Frame congestion is typically addressed by dealing with the nodes that are causing the congestion.
- **Misbehaving devices**: As stated earlier, there is little that can be done in the fabric to mitigate the effects of a badly behaving device other than to remove it from the fabric. Brocade supports a Brocade FOS capability called Port Fencing, which is designed to isolate rogue devices from the network. Port Fencing works with Brocade MAPS to disable a port when a specific threshold has been reached. Port Fencing, in combination with FPI monitoring, can be used for detecting and isolating high-latency devices from impacting the rest of the devices in the fabric.
- **Initiator and targets**: If possible, isolate host and storage ports on separate switches for much greater control over the types of controls that you can apply to misbehaving and high-latency devices. The effect on applications is typically much less severe if a host is disabled versus disabling a storage port, which may be servicing flows from many servers.

Chapter 9: FC Routing

9.1 Overview and Purpose

A large SAN may have thousands of end devices, which could inundate or exceed fabric service scalability, convergence timeliness, and user manageability. FC Routing (FCR) constrains fabric services to within the edge fabric and the backbone fabric. Fabric services do not span the entire fabric. FCR enables end devices to communicate across multiple autonomous edge fabrics without merging them. Fabric services, for example the name server, are self-contained within each edge fabric or backbone.

Limiting fabric services to within an edge fabric is done for various reasons:

- The overall SAN can scale to a much larger relative size compared to the scalability of the fabric services within an edge fabric.
- Within an edge fabric, FCR reduces switch domains and managed zones.
- Edge-fabric disturbances and reconfigurations affect only the local fabric services, thereby providing fault isolation.
- FCR increases security because end devices cannot communicate outside of an edge fabric unless explicitly zoned to do so.

9.2 Edge Fabrics

Edge fabrics are traditional fabrics except they are connected to backbone EX_Ports. Edge fabrics contain end devices and may be connected to other edge fabrics by a backbone fabric. An edge fabric can be a combination edge fabric and backbone.

Generally, edge fabrics follow the core-edge or collapsed-core best practice design, the same as traditional fabrics. Unique to FCR is edge-fabric interconnectivity to a backbone.

9.3 Inter-Fabric Links

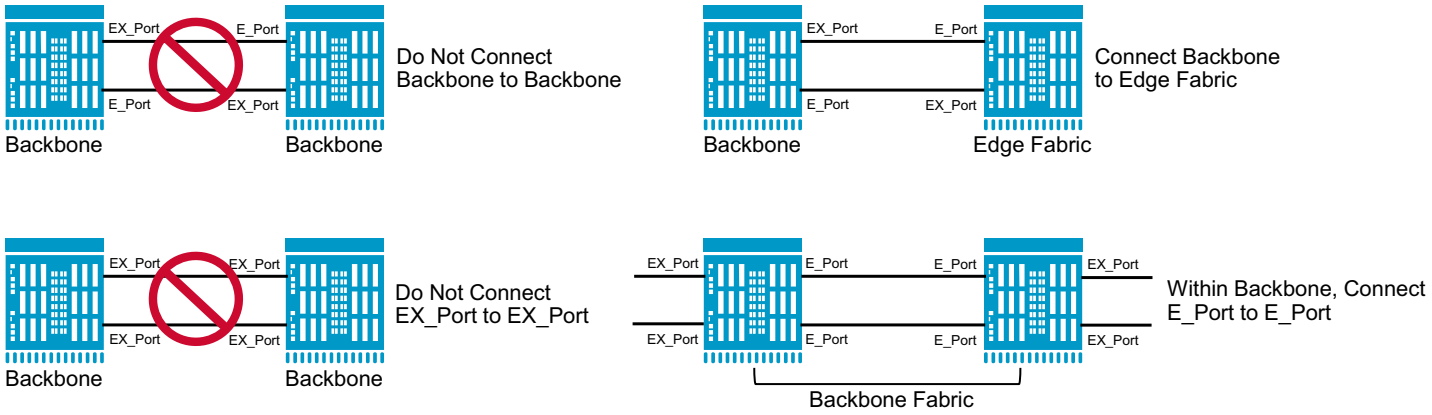
An inter-fabric link (IFL) connects an EX_Port to an E_Port; it is a type of inter-switch link (ISL) that spans from an edge fabric to a backbone or to a combination backbone/edge fabric.

Provision enough IFLs between each edge fabric and the backbone to accommodate the projected peak traffic load.

9.4 Backbone Fabrics

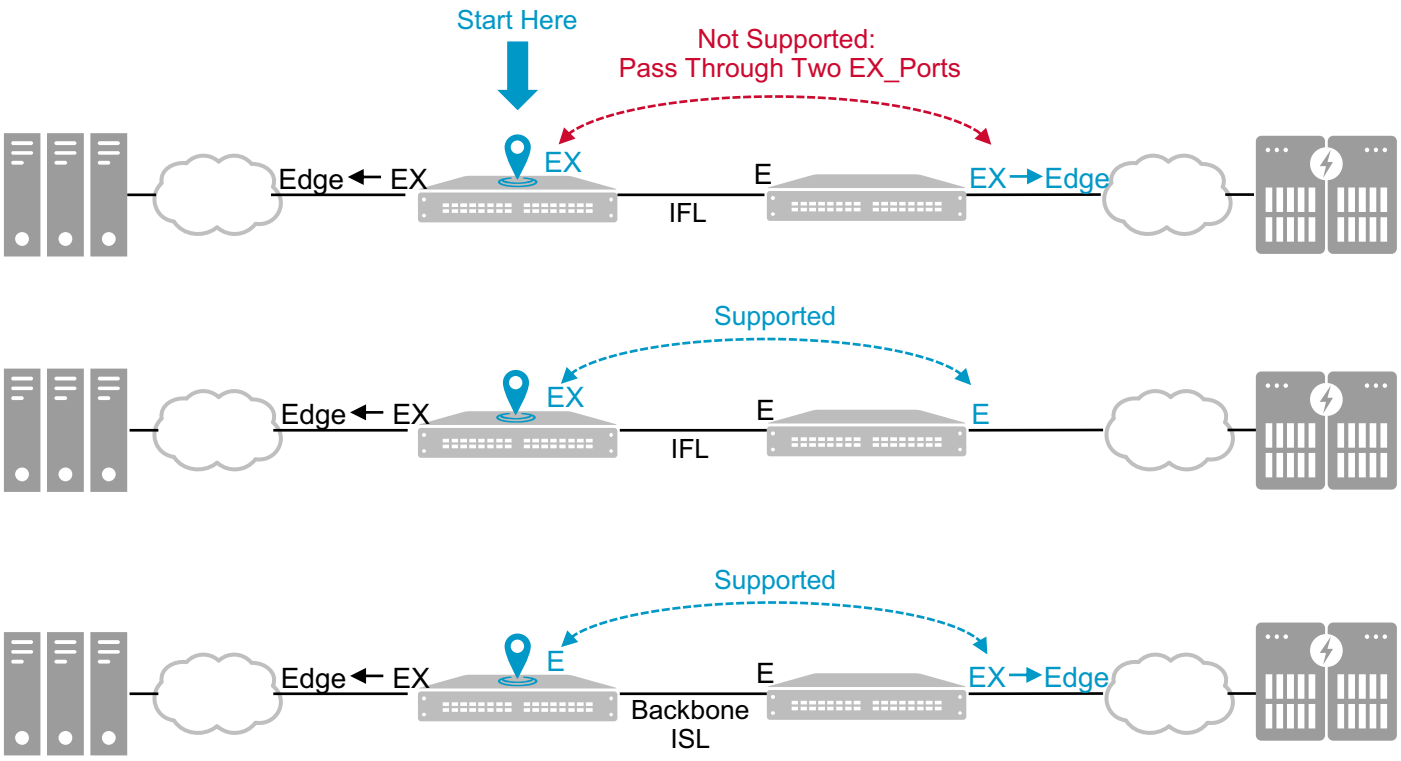
Backbone fabrics contain EX_Ports and are the demarcation for fabric services. A fabric may contain one backbone. A backbone can be dedicated to being a backbone with no connected end devices, or it may have connected end devices—both architectures are supported (Figure 10). Do not connect a backbone to a backbone, which is not supported; see Figure 10. Additionally, the backbone fabric may or may not contain Extension (FCIP) links, see Section 9.8, FCR and Extension.

Figure 10: Supported Backbone Architectures



EX_Ports are fabric-service demarcation points. Fabric services do not pass beyond an EX_Port. Two EX_Ports cannot be connected together. EX_Ports connect only to E_Ports. Topology supportability is determined by starting within an FC router and moving toward the end device; traffic cannot pass more than one EX_Port along the path to the end device. If more than one EX_Port is passed, the architecture is unsupported. (See Figure 11.)

Figure 11: Supported FCR Architectures



There are many factors to consider when designing backbone fabrics. Backbone fabrics vary based on size, requirements for redundancy, and distance between edge fabrics. Generally, SAN architecture recommendations apply equally to backbone fabrics. There should be redundant fabrics, and each fabric should have redundant paths to every edge fabric. Consider the following factors when identifying the best switch platforms and backbone topology, including interconnections. The number of edge fabrics can impact the backbone topology and the manner in which they are attached. Brocade FCR can be enabled on standard FC ports; in some cases a license may be required.

Composition of edge fabrics:

- **Scale and interoperability**: Ensure that director and switch platforms are capable of supporting the scale and interoperability needed.
- **Legacy SAN platforms**: Anywhere in the SAN, earlier directors/switches or firmware may impact supported features, manageability, and interoperability.
- **Advanced SAN applications and features**: Some advanced SAN applications and features may not be compatible with FCR or a particular platform type.

Projected inter-fabric traffic patterns:

- **Quantity (bandwidth utilization)**: Provision enough ISLs within the backbone to accommodate projected peak traffic loads that will traverse the backbone.
- **Bursty versus peak traffic**: Bursty traffic is a sudden spike that often dissipates rapidly. It is not the same as peak traffic, which may not be bursty. If infrequent, bursty traffic can be forgiving. Traffic bursts can cause temporary increases in response times due to oversubscription and congestion. Buffer credits may be withheld until the burst subsides. Such congestion would be less likely during continuous traffic patterns.
- **Small versus large frame size**: Fibre Channel is a high-speed, low-latency protocol. It relies on buffer-to-buffer credit (BBC) flow control. This mechanism is a fundamental part of FC and provides lossless data communications. A sequence of small frames uses the same number of BBCs as a series of large frames. On the other hand, large frames use more bandwidth. In other words, a large amount of small-frame traffic can fully utilize available buffers while consuming only a very small amount of bandwidth. Therefore, consider not only bandwidth but also the typical frame size. For instance, FC compression creates primarily smaller FC frames. If the bulk of frames is expected to be smaller, additional buffers should be allocated to the paths that are handling those I/O patterns. Pay extra attention to this type of congestion, because congested backbones adversely impact the performance of all connected edge fabrics. When in doubt, overprovision IFLs.
- **Distance (location of fabrics)**: Long-distance IFLs require adequate bandwidth and BBCs to prevent data transmission congestion and droop respectively. Consider all potential traffic flows that may traverse the long-distance links. Long-distance solutions have increased latency (simple physics of time = distance/rate); therefore, it is important that long-distance links be overprovisioned to prevent oversubscription, such that unexpected bursts do not adversely impact data flow and potentially the entire fabric.
- **Virtual Fabrics (VF)**: All EX_Ports must reside in the base switch. The base switch does not support ISL R_RDY mode. If a logical switch has XISL enabled, you cannot connect an EX_Port to that logical switch. The base switch is similar to a backbone switch, and a base fabric is like a backbone fabric. All switches in a backbone fabric must have the same backbone fabric ID, which must be unique relative to any of the edge fabrics.

Potential growth:

- **Number of fabrics**: If the number of fabrics is likely to increase, then deploy backbone fabrics such that they can readily accommodate additional edge fabrics and additional traffic loads.
- **Size of fabrics**: If the size of edge fabrics is likely to grow, and the inter-fabric traffic is expected to grow accordingly, provision additional IFLs and ISLs such that the capacity of available paths stays well ahead of current usage. That way, incremental growth on the edge can be accommodated without the need to immediately upgrade the backbone.
- **Amount of traffic between fabrics**: If the inter-fabric traffic is expected to grow even without growth in the individual edge fabrics, then provision additional IFLs and ISLs such that the capacity of available paths stays ahead of current usage. That way, incremental increases in data flow across the backbone can be accommodated without the need to immediately upgrade the backbone. Make sure that you allow for plenty of room for backbone expansion.

NOTE: Refer to the *Brocade SAN Scalability Guidelines* for FCR scalability limits.

Consider using FCR under the following conditions:

- There are requirements for added scalability.
- There are benefits to compartmentalize manageability.
- Enhanced security is required.
- There is a limited number of initiator-target pairs shared between edge fabrics.
- There is a limited number of LUNs shared between edge fabrics.
- Archiving devices, such as tape libraries, must be shared.

The implementation and configuration of inter-fabric links (IFLs in the case of FCR) should be based on the expected data flow between the switches and/or fabrics in question and the desired level of redundancy between edge switches and across the routed SAN. Some architectural examples of FCR topologies follow.

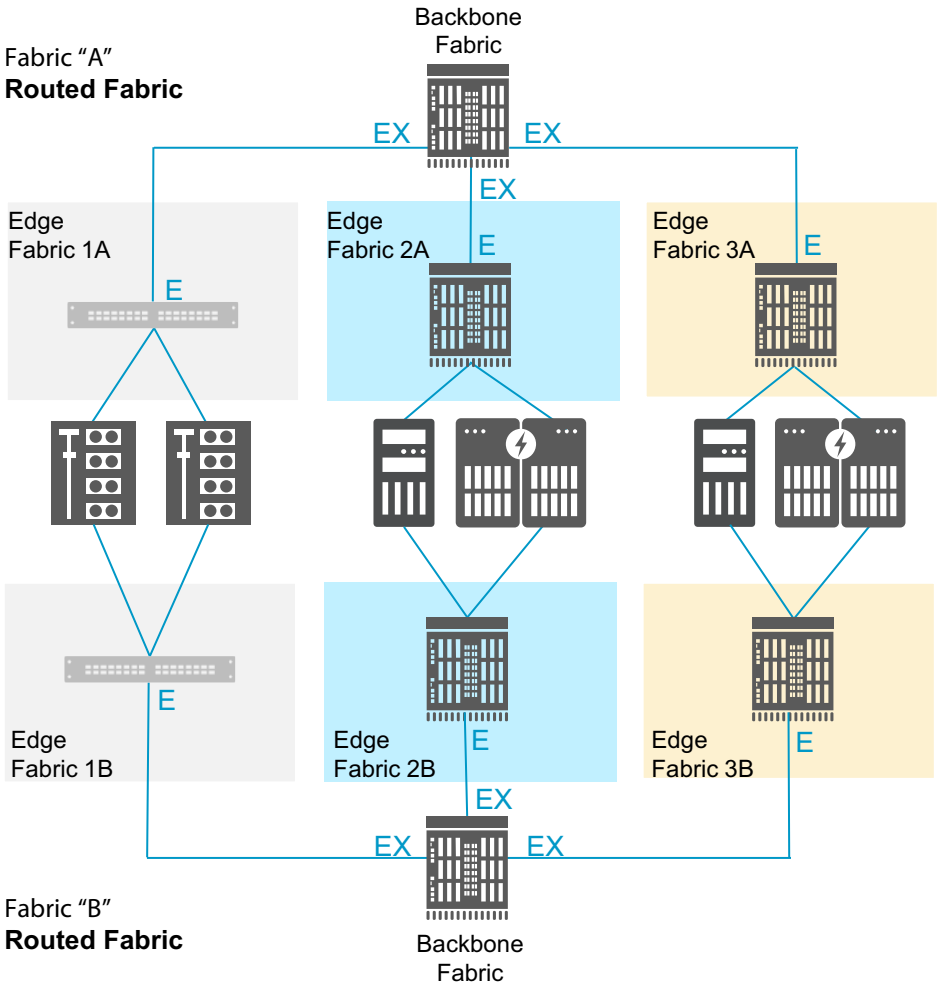
Except in the case of tape, which often has only a single pathway, there should always be A and B fabrics with IFL redundancy. A routed SAN environment consists of multiple edge fabrics interconnected by one or more backbone fabrics. Multiple backbone fabrics would be in parallel and belong to only either the A or B fabric, not both. A backbone fabric can simply be a single switch or a core-edge topology. This is true for the edge fabrics as well.

In Figure 12, the architecture consists of three edge fabrics and a backbone fabric. A and B fabrics are shown. The “A” backbone connects to each edge fabric via EX_Ports. EX_Ports in the backbone connect to E_Ports in the edge fabric to form IFLs. Each backbone must have a unique backbone fabric ID (BBFID), and all switches within that backbone must have that same BBFID. The default is 128, and when a single backbone is deployed, as in Figure 12, no BBFID needs to be configured because the default will suffice. An alias can be assigned to BBFIDs.

Each edge fabric must have a unique edge-fabric ID (EFID), and all EX_Port connections to that edge fabric must use that EFID. Each EX_Port is configured with the EFID that belongs to the edge fabric to which it is connecting. E_Ports are not configured with any additional parameters when connecting to EX_Ports.

This is a relatively simple and straightforward FCR architecture.

Figure 12: Routed A and B Fabric Collapsed-Core Architecture



In Figure 13, a separate backbone fabric has not been deployed. Instead, the middle fabric has been assigned a dual role of backbone and edge fabric, which is fully supported. There are three edge fabrics with their own self-contained fabric services.

Not having a separate backbone fabric limits the topology from being an interconnected full mesh. There are only two connections coming out from the center edge fabric, and there is no connection between the left and right edge fabrics. Such a design violates the previously mentioned supported FCR architectures by creating a situation in which more than one EX_Port may be traversed from inside an FC router to the ultimate destination device.

This design may be used when the cost of an additional fabric for the backbone is prohibitive.

Figure 13: Common-Backbone, Dual Collapsed Core Architecture

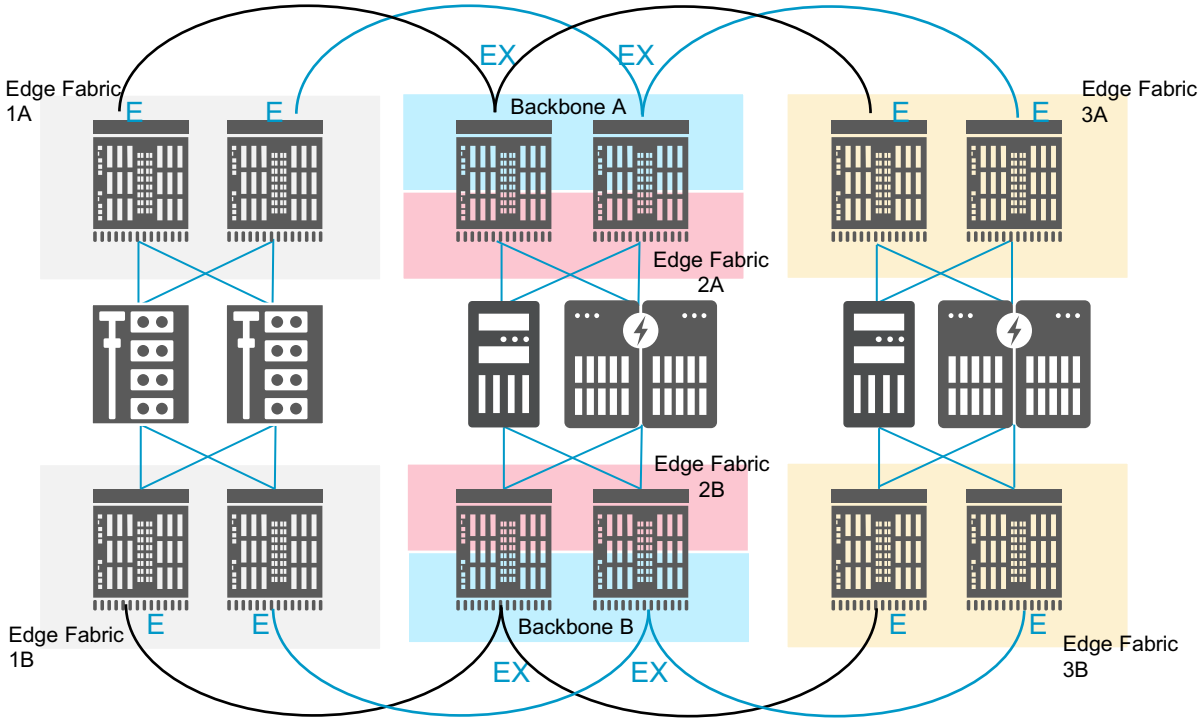
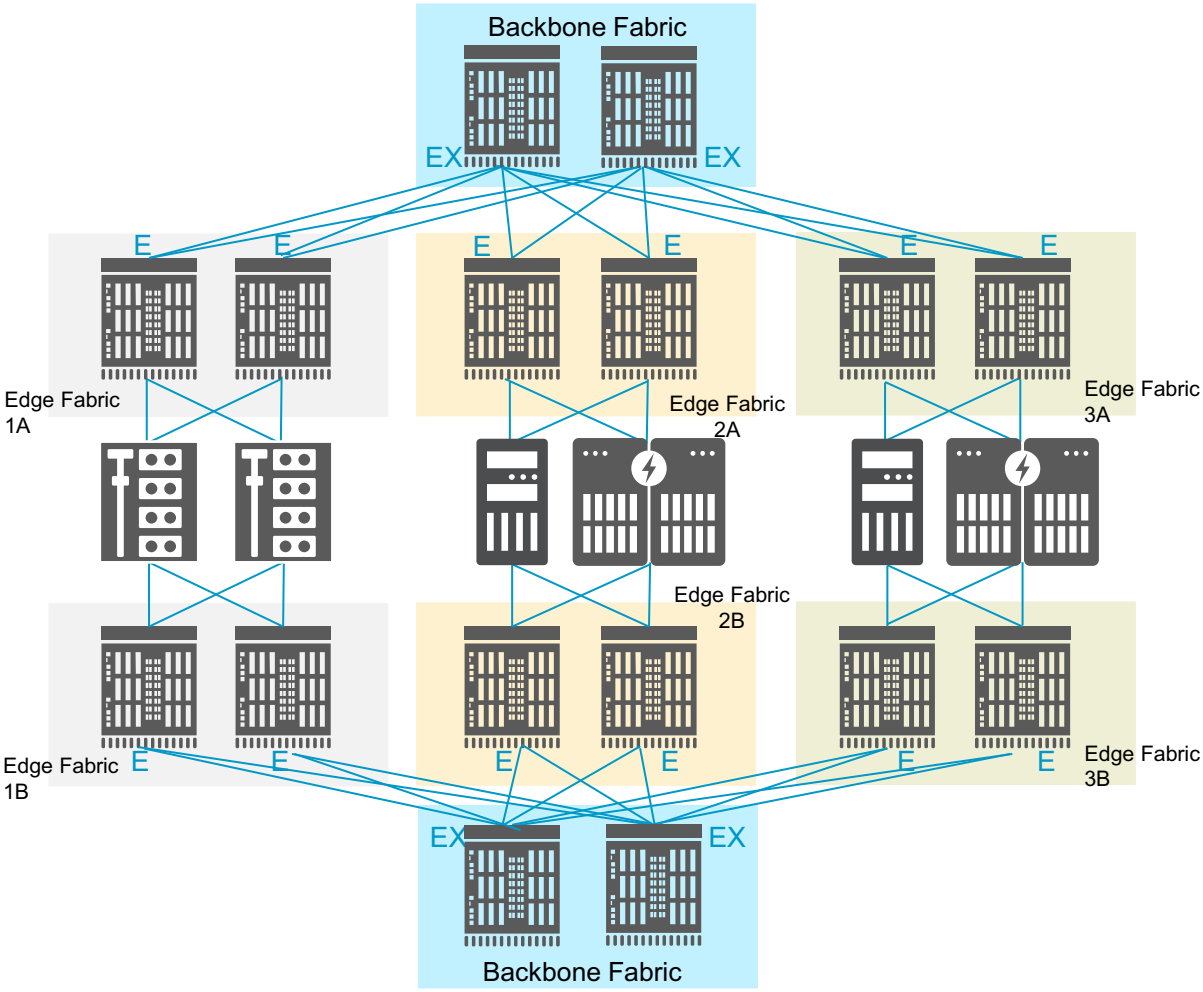


Figure 14 shows a routed SAN with A and B fabrics each having a dual core backbone and a unique BBFID. The EX_Ports are exclusively in the backbone, and fabric services do not pass beyond the EX_Ports. There are three edge fabrics, each with its own EFID. There are multiple IFLs to each edge fabric. The Dual Core Backbone architecture is a highly redundant/ resilient and scalable architecture for critical enterprise applications that demand zero downtime. Considering the scale of a dual core backbone FCR SAN, it is easy to manage.

Figure 14: Dual Core Backbone Routed Fabric



9.5 Redundancy

FCR SAN redundancy is achieved by:

- Using best practices within the edge fabrics (core-edge or collapsed core architectures).
- Using best practices within the backbone fabric(s) (core-edge or collapsed core architectures).
- Deploying dual backbone fabrics for each fabric (A and B). The need for redundancy versus cost and operations must be considered. Ask yourself what the purpose of the routed SAN is. What happens if routing between edge fabrics goes offline, yet the edge fabrics themselves remain online?
- Parallel IFLs between the backbone and edge fabrics. This includes ports, optics, and cable redundancy.

9.6 Avoiding Congestion

As with a flat Layer 2 fabric, a routed SAN must be evaluated for traffic bandwidth and potential bandwidth utilization between all endpoints. For routed topologies, this means calculating the traffic flowing in and out of every edge fabric and providing enough links into and across the backbone to accommodate that traffic. The same ISL guidelines and best practices apply when connecting edge fabrics via IFLs for improved utilization and resiliency. A higher performance edge fabric versus an

underperforming backbone may result in an oversubscribed backbone, which during peak loads can lead to congestion, higher latency, and longer storage response times. If the edge fabric has 64Gb/s FC ISLs, the backbone fabric must have 64Gb/s FC ISLs as well. Prior to upgrading an edge fabric, upgrade the backbone as well to avoid congestion and oversubscription issues.

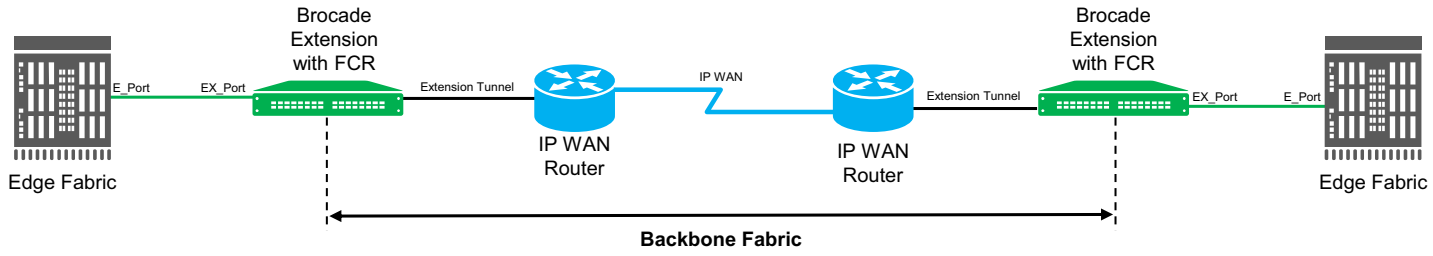
9.7 Available Paths

An optimal approach is to have multiple Brocade trunking paths between edge fabrics so that traffic can be spread across available resources. Never attach both A and B fabrics to the same backbone device. Connecting A and B edge fabrics to the same backbone device destroys the air gap between A and B and is *not* considered a redundant architecture and best practice. From the perspective of FC, you should adhere to the concept of an “air gap” all the way from host to storage. A common device connected to both A and B fabrics can cause a SAN-wide outage. If an air gap is implemented, faults on one fabric cannot affect the other fabric. These faults can manifest from defects in hosts, drivers, the fabric operating system, the fabric hardware, the storage hardware, the storage software, and human error. It is not relevant that FCR keeps fabric services separate because faults within one large routed fabric can transcend FCR, causing the entire SAN to fail.

9.8 FCR and Extension

FCR can be used within a single data center or across campus data centers, as well as between edge fabrics connected by FCIP over a metropolitan area network (MAN) or wide-area network (WAN), as shown in Figure 15. A Brocade Extension tunnel is an ISL (VE_Port to VE_Port). A VE_Port is an E_Port that is an endpoint of an Extension tunnel. Each Extension platform becomes part of the backbone fabric. EX_Ports on the Extension platforms connect out to the edge fabrics via one or more IFLs.

Figure 15: Fibre Channel Routed Fabric over Extension



More information about Extension can be found in Chapter 12.

9.9 FCR Design Guidelines and Constraints

The following are some of the key metrics and best practices for routed SAN topologies:

- Keep A and B fabrics separated all the way from host to storage from an FC perspective. This is referred to as an *air gap*. This does not include FCIP over an IP network because Ethernet switches and IP routers will not merge FC fabrics. Extension VE_Port endpoints should never connect A and B fabrics, which is the same as connecting a traditional ISL, resulting in the cross-connection of the A and B fabrics.
- Localize traffic within an edge fabric to the greatest extent possible.
- Have a predefined schema for assigning domains within the SAN. For example, edge cores and switches, EFIDs, translate domains, and BBFIDs should be within a certain range to avoid domain overlap.

- Consider upgrading backbone fabrics prior to upgrading edge fabrics to avoid oversubscription and congestion.
- Have no more than one long-distance ISL or Extension between source and destination during normal operations. An additional hop may be used for high availability during an outage. For example, in a triangle architecture in which the primary link has gone down, the remaining two legs of the triangle can be used as a backup path, although, latency and response times will likely be longer.
- Long-distance links are within the backbone and not between an edge fabric and the backbone. Edge fabrics will be isolated from disruption because fabric services are not extended beyond the EX_Port. Most often, long-distance links are the primary cause of instability.
- Logical SAN (LSAN) zones are only for end devices that communicate from edge fabric to edge fabric across a backbone. In other words, do not make zones within edge-fabric LSAN zones.
- Redundant backbone fabrics improve resiliency. This means redundancy for each fabric; therefore, fabric A would be resilient, as would fabric B. The entire fabric A (both backbones) would have to fail before relying solely on fabric B to maintain critical operations across edge fabrics.
- Separate redundant backbone fabrics that share connections to the same edge fabrics must have unique BBFIDs. This refers to the case in which there are multiple backbone fabrics for A and multiple backbone fabrics for B. This does not refer to cross-connections between the A and B fabrics, nor does it refer to cross-connections between the parallel backbones within fabric A or within fabric B.

Chapter 10: Virtual Fabrics Topologies

Virtual Fabrics (VF) is an architecture to virtualize hardware boundaries within a SAN platform. Traditionally, SAN design and management are done at the granularity of a physical switch. Virtual Fabrics allows SAN design and management to be done at the granularity of a port.

Virtual Fabrics is a suite of related features that can be customized based on your needs. The Virtual Fabrics suite consists of the following specific features:

- Logical switch
- Logical fabric
- Device sharing

Hardware-level fabric isolation is accomplished through the concept of a logical switch, which provides the ability to partition physical switch ports into one or more “logical” switches. Logical switches are then connected to form logical fabrics. As the number of available ports on a switch continues to grow, partitioning switches gives storage administrators the ability to take advantage of high-port-count switches by dividing physical switches into different logical switches. Without VF, an FC switch is limited to 512 ports. A storage administrator can then connect logical switches through various types of ISLs to create one or more logical fabrics.

There are three ways to connect logical switches: via a traditional ISL, via an IFL (EX_Port used by FCR), or via an extended ISL (XISL). An ISL can be used only for normal L2 traffic between the connected logical switches, carrying only data traffic within the logical fabric of which the ISL is a member. One advantage of Virtual Fabrics is that logical switches can share a common physical connection, and each logical switch does not require a dedicated ISL. In order for multiple logical switches, in multiple logical fabrics, to share an ISL, Virtual Fabrics supports an XISL connection, which is a physical connection between two base switches. Base switches are a special type of logical switch that are specifically intended for intrafabric and interfabric communication. As mentioned, base switches are connected via XISLs and form the base fabric.

Once a base fabric is formed, the virtual fabric determines all of the logical switches and logical fabrics that are physically associated via the base fabric, as well as the possible routes between them. For each local logical switch, a logical ISL (LISL) is created for every destination logical switch in the same virtual fabric that is reachable via the base fabric. Thus, an XISL comprises the physical link between base switches and all of the virtual connections associated with that link. In addition to XISL support, the base fabric also supports IFLs via EX_Port connections for communication between virtual fabrics. Base switches also interoperate with FC router switches, either in the base fabric or in separate backbone fabrics.

10.1 Use Case: FICON and Open Systems (Intermix)

Virtual Fabrics enables customers to share FICON and FCP traffic on the same physical platform. As chassis densities increase, this is a viable option for improved hardware utilization while maintaining director-class availability. The primary reasons for moving to an Intermix environment are the following:

- Array-to-array RDR of FICON volumes (uses FCP)
- ESCON-FICON migration
- Sharing of infrastructure in a nonproduction environment
- Reduced TCO
- Growth of zLinux on the mainframe

From a SAN design perspective, consider the following guidelines when considering FICON Intermix:

- Connect devices across port blades (connectivity from the same device should be spread over multiple blades).
- A one-hop count still applies (there are “Hops of No Concern” in some cases).

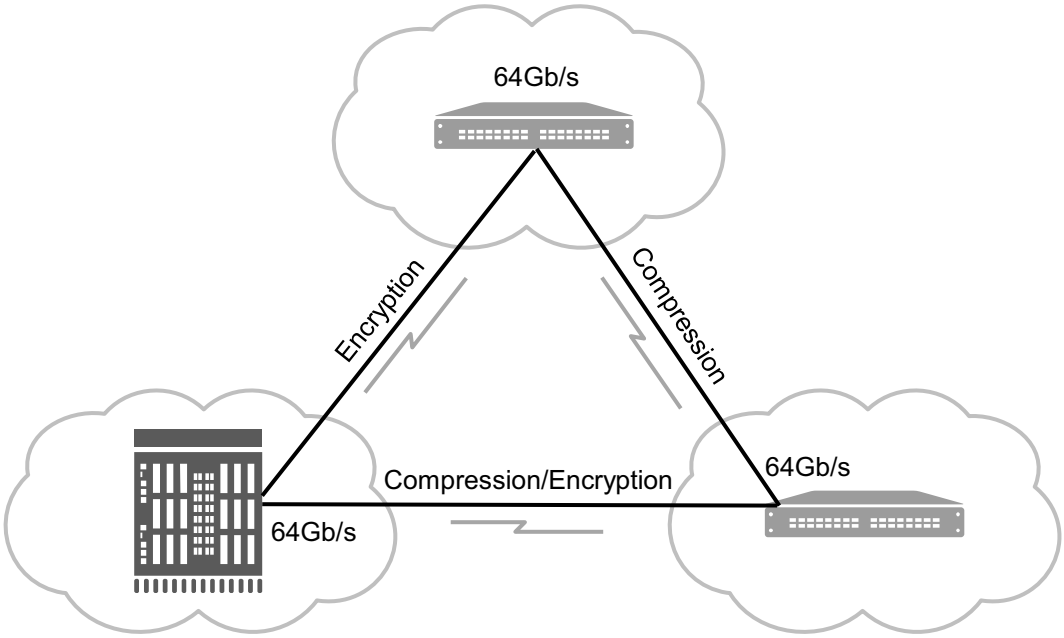
For details, refer to the *Brocade FICON/FCP Intermix Best Practices Guide*.

Chapter 11: Fibre Channel Intelligent Services

11.1 In-Flight Encryption and Compression

Brocade Gen 6 and Gen 7 Fibre Channel platforms support both in-flight compression and/or encryption at a port level for both local and long-distance ISL links (see Figure 16). In-flight data compression is a useful tool for saving money when either bandwidth caps or bandwidth usage charges are in place for transferring data between fabrics. Similarly, in-flight encryption enables a further layer of security with no key management overhead when transferring data between local and long-distance data centers besides the initial setup.

Figure 16: Latency for Encryption and Compression



Enabling in-flight ISL data compression and/or encryption increases the latency as the ASIC processes the frame compression and/or encryption. Approximate latency at each stage (including encryption, compression, and local switching) is 6.2 microseconds (see Figure 16).

11.1.1 Virtual Fabric Considerations: Encryption and Compression

The E_Ports in the user-created logical switch, base switch, or default switch can support encryption and compression. Both encryption and compression are supported on XISL ports, but they are not supported on LISL ports. If encryption or compression is enabled and ports are being moved from one LS to another LS, it must be disabled prior to moving from one LS to another LS.

11.1.2 Guidelines: In-Flight Encryption and Compression

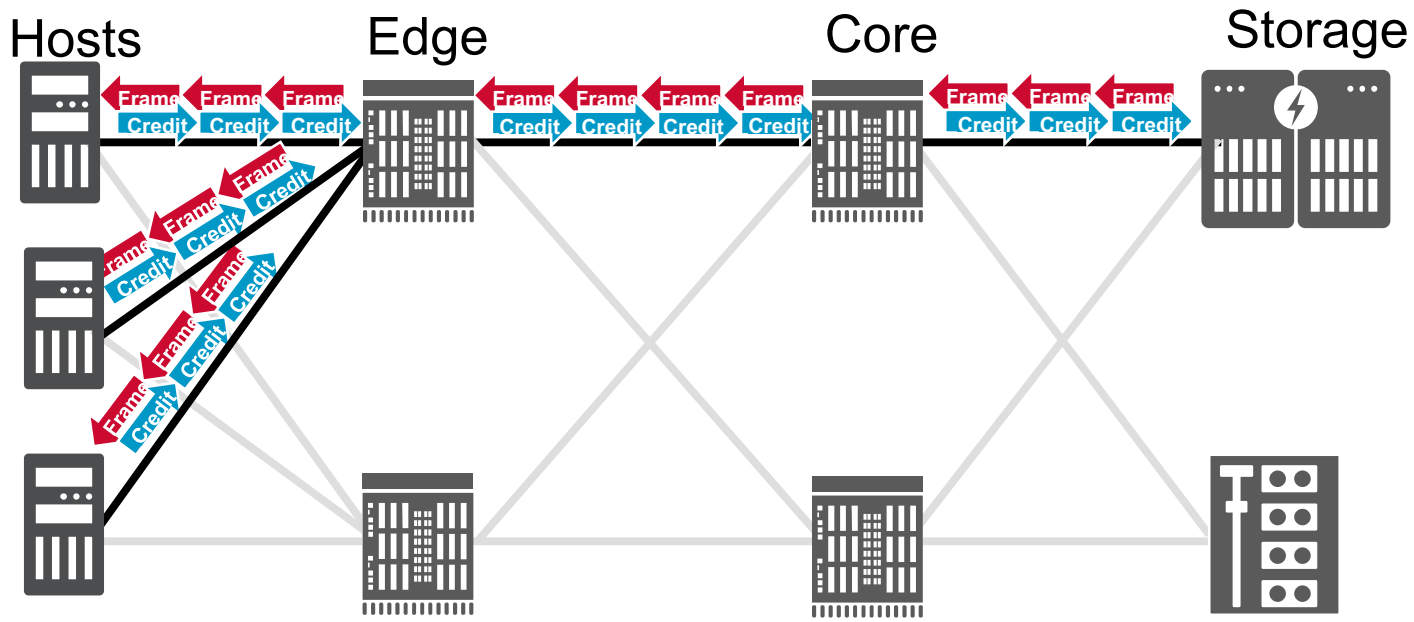
Refer to the *Brocade Fabric OS Administration Guide* for the latest information.

11.2 Fabric Notifications

Fibre Channel networks can be elusive to troubleshoot because flows are difficult to visualize and the affected devices do not likely correspond with the cause of the problem. Fibre Channel uses a credit-based flow-control mechanism (see Figure 17), which has inherent congestion characteristics due to head of line blocking. Brocade introduces a hardware, software, and management solution for achieving congestion reduction and elimination that is called Fabric Notifications.

By identifying useful data from various sources, data can be collected, evaluated, and disseminated to interested devices, allowing for faster and sometimes automatic problem resolution. End devices can employ basic response and recovery mechanisms. Fabric information is useful for end devices, and end devices have useful information for the fabric and peer end devices. Fabric Notifications plays a key role in collecting and disseminating information among interested and related devices.

Figure 17: Freely Moving Lossless Credit-Based Flow-Control FC Network



Fabric Notifications addresses four issues: congestion (oversubscription and credit stall), link integrity, and SCSI command delivery failure.

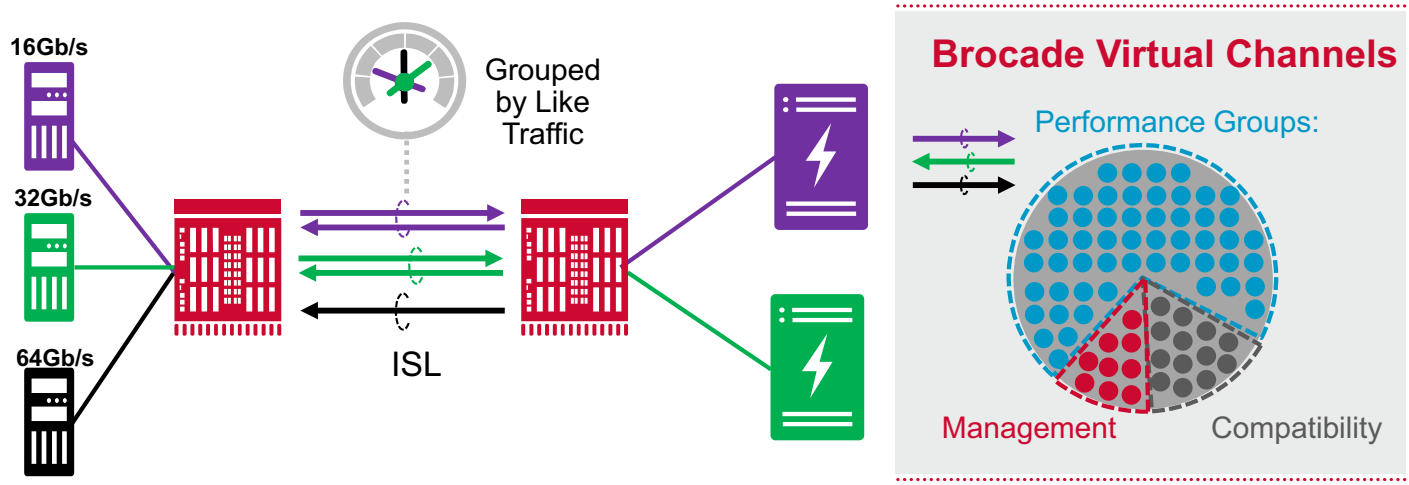
11.3 Traffic Optimizer

For years, Brocade virtual channels (VCs) worked well at slower rates of 1, 2, 4, 8, and 16 gigabit by leveraging multiple logical independent paths. To a degree, this mitigated interference of slower flows that were impeding upon faster flows. Plus, optionally faster flows could be manually assigned to a high QoS VC, and slower flows could be manually assigned to a low QoS VC to prevent interference.

Technology evolves, and Brocade has optimized VC efficiency by enhancing effectiveness to targeted flow characteristics. Demands on an enterprise SAN have never been greater due to other complimentary technology evolutions such as NVMe, AFA, and host virtualization. Flows are competing, and head of line blocking is not an option and must be efficaciously dealt with. This is where Traffic Optimizer will help.

Brocade Traffic Optimizer technology takes VCs to the next level (see Figure 18). Traffic Optimizer organizes and manages traffic flows using performance groups (PGs), and fabric resources are allocated based on performance groups. Flows are assigned to a VC based on the destination port speed or protocol (for example, NVMe). Brocade fabrics know the destination port speed and protocol for every flow.

Figure 18: Host to Storage via Traffic Optimizer VCs



Brocade Gen 7 hardware added more VCs. E_Ports and EX_Ports use 16 new Traffic Optimization VCs, which are assigned as follows: Four VCs per speed (<16, 32, 64) for a total of 12 VCs, plus four VCs dedicated to NVMe. Benefiting from an NVMe storage investment requires NVMe dedicated resources in the SAN.

QoS VCs have not changed; there are 2 for low, 4 for medium (default), and 5 for high. QoS VCs are used when traffic has been designated to a high or low priority. Slow Drain Device Quarantine (SDDQ) uses the QoS low VCs.

Chapter 12: Extension

12.1 Extension Common Principles

12.1.1 Ethernet RJ-45, SFP, SFP+

Depending on the Extension platform, a variety of Ethernet interfaces are available. The Brocade 7840 Extension Switch and SX6 Extension Blade have two 40GE interfaces and sixteen GE/10GE interfaces. Certainly, the interfaces selected must be capable of carrying the bandwidth required by the storage application and configured for the circuit. For example, you cannot put a 2Gb/s circuit on a GE interface.

Additionally, GE/10GE interfaces are in port groups, and within a group, the speed must be consistent. The following is a list of ports that block each other when their speed is not set the same. The best practice is to use 2, 3, 4, 5, 10, 11, 12, and 13 first and then move on to doubling up within a port group. Please check the *Brocade Fabric OS Extension User Guide* for additional information.

Ethernet interfaces in the same group:

- 2 & 6
- 3 & 7
- 4 & 8
- 5 & 9
- 10 & 14
- 11 & 15
- 12 & 16
- 13 & 17

Keep in mind that the Ethernet interfaces do not perform any type of speed negotiation between 1Gb/s and 10Gb/s. The speed is user-configurable only. An SFP (1Gb/s) and an SFP+ (10Gb/s) are different optics, and neither can change to the speed of the other. If you need 1Gb/s connections, you must order GE optics; likewise, if you need 10Gb/s connections, you must order 10GE optics.

The Brocade 7810 has GE and 10GE interfaces. Two of the interfaces (GE0 and GE1) are RJ-45 copper ports (1Gb/s only). Copper enabled is the default. These copper ports are mutually exclusive with optical ports GE0 and GE1. Either copper is enabled or the optics are enabled, but not both. There is no disadvantage to using one set of interfaces over another.

12.1.2 Brocade Extension Trunking

Brocade Extension Trunking (BET) is an exclusive Brocade feature that offers the following benefits:

- In-order delivery
- Remediation of data lost in flight
- Bandwidth aggregation
- Granular load balancing
- Lossless failover/failback

A tunnel that contains more than one circuit is a Brocade Extension Trunk. A tunnel endpoint is defined by a VE_Port. BET forms a single ISL between two VE_Ports. BET circuits terminate at the VE_Port on each side; therefore, only a single tunnel is load-balancing across the circuits. A circuit is a connection that is defined by a source and destination IP address and other configuration parameters such as QoS, min/max rate limits, and KATOV.

NOTE: Compression and IPsec are at the tunnel level, not the individual circuit level.

A circuit is assigned to an Ethernet interface via the IP address (ipif). Often, each circuit is assigned to its own dedicated Ethernet interface; however, this depends on the speed of the interface, the cumulative max-comm-rates of the circuits, and the port redundancy required. Ethernet interfaces physically connect to a DC LAN switch or WAN router. Connect one interface to switch “A” and the other interface to switch “B.” Circuits load-balance with a high degree of granularity. BET will prevent transmission data loss when a network device fails or goes offline or when optic instability, cable damage, a human error, or a service-provider disruption occurs.

Circuits may have varying characteristics, for instance, circuits may experience different latency, take different paths (like DC LAN switches/WAN routers), and belong to different service providers. Circuits that belong to a VE_Port can have bandwidth differences up to 4x the lowest bandwidth circuit. For example, if the cir0 min-comm-rate is 1Gb/s, the min-comm-rate on cir1 can be no more than 4Gb/s.

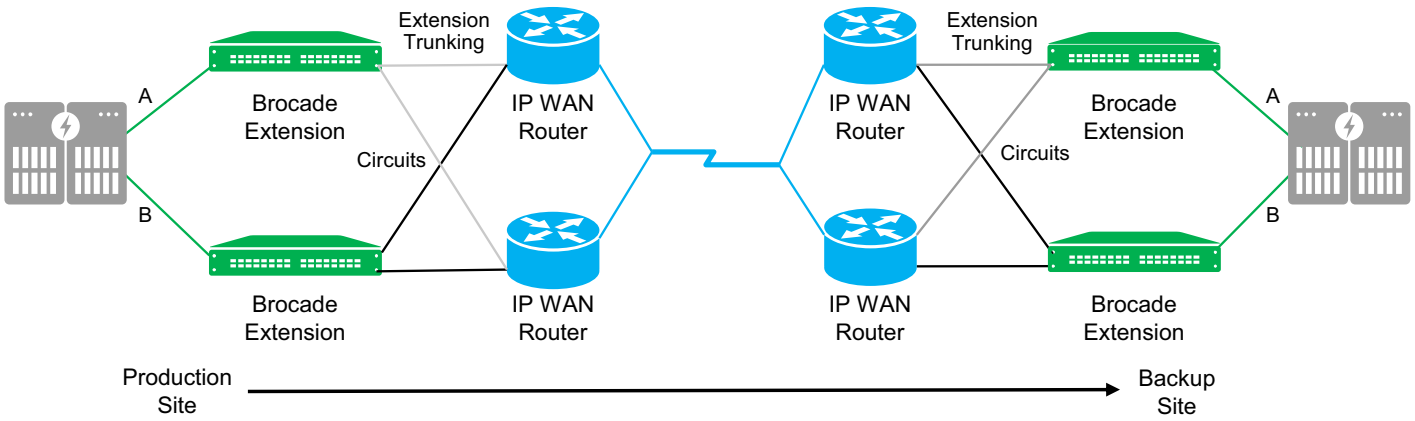
Using BET is considered best practice versus using multiple VE_Ports. Multiple VE_Ports between the same two domains will use one of the following methods to route traffic across the VE_Ports:

- EBR (Exchange-Base Routing: default)
- DBR (Device-Based Routing)
- PBR (Port-Based Routing)

The architecture shown in Figure 19 is a dedicated high-availability replication SAN with no connections to the production SAN. Array replication ports are nearly always dedicated; therefore, these ports should not connect to the production fabric.

About BET, there is an extension trunk for the “A” path (grey - top to top Extension boxes) and an extension trunk for the “B” path (black - bottom to bottom Extension boxes). The IP network merely connects these point-to-point circuits to each associated endpoint.

Figure 19: Four-Box High-Availability Replication SAN



All data centers have redundant routers/switches. Connecting each Brocade Extension platform to each router/switch for redundancy and resiliency is best practice. Without BET this design would require two VE_Ports per Extension box. Available VE_Ports are not the issue; performance, resiliency, and redundancy are of paramount importance. It is best to use one VE_Port with BET to form a trunk that consists of two circuits between the domains.

Typically, tape can take only a single SAN path, but there are exceptions. Brocade Extension Trunking (BET) is logically a single path. Take advantage of BET to implement redundant network paths with each circuit. Tape must transparently fail over / fail back paths without data loss while maintaining in-order delivery; otherwise, tape jobs fail.

For disk and tape protocol optimization (FastWrite for disk and Open Systems Tape Pipelining for tape), a single tunnel between domains is required. Use multiple circuits for redundancy, resiliency, and failover protection. If multiple tunnels are required, Virtual Fabrics Logical Switches (VF LS) must be configured to ensure that all sequences from every exchange traverse the same VE_Port in both directions. This means one VE_Port per LS.

12.1.3 IPsec

As a best practice, use Brocade IPsec to protect data end-to-end. When Brocade Extension is implemented, it is always prudent to enable IPsec. All data that leaves the secure confines of a data center into a vulnerable infrastructure guarantees no security, and no service provider guarantees data security in flight. Links must be authenticated and data must be encrypted to prevent attacks and eavesdropping. Brocade IPsec is easy and practical to deploy. Would your company operate WiFi with no encryption? Of course not!

Brocade IPsec is an hardware implementation that can operate at line rate. IPsec is included in all base Extension platforms. There are no additional licenses or costs for IPsec. Encryption adds an insignificant propagation delay (5 µs). Pre-shared key (PSK) configuration is simple.

Brocade IPsec is Suite B compliant and implements the latest encryption technologies, such as AES 256, SHA-512 HMAC, IKEv2, and Diffie-Hellman. Rekeying occurs in the background approximately every 2 billion frames or every 4 hours, and the process is nondisruptive.

Firewalls are *not* considered best practice for the following reasons:

- Brocade IPsec can operate up to 20Gb/s per data processor (DP), which is WAN-side full line rate, and 40Gb/s if implementing both DPs. Most firewalls cannot meet this throughput requirement.
- Brocade requires minimal propagation delay because it is storage traffic.
- Brocade IPsec encrypts data closer to the source and destination of the data, which is considered best practice.
- Firewalls and WAN optimization that proxy with TCP proxy sessions may result in remote emergent TCP segments not being identical. This is not supported. Such WAN optimization devices are also not supported for this same reason.

12.1.4 Adaptive Rate Limiting

Adaptive Rate Limiting (ARL) is an integral part of most Extension designs. When there is more than one circuit feeding into the same WAN link or when the WAN is shared with other traffic, ARL is an essential component. ARL is used to manage data sent into the IP network based on min and max rate settings and the available WAN bandwidth between that range. There may be a single WAN connection or multiple WAN connections. The cumulative bandwidth of the circuits assigned to a particular WAN link is what matters. Multiple WAN connections are evaluated independently.

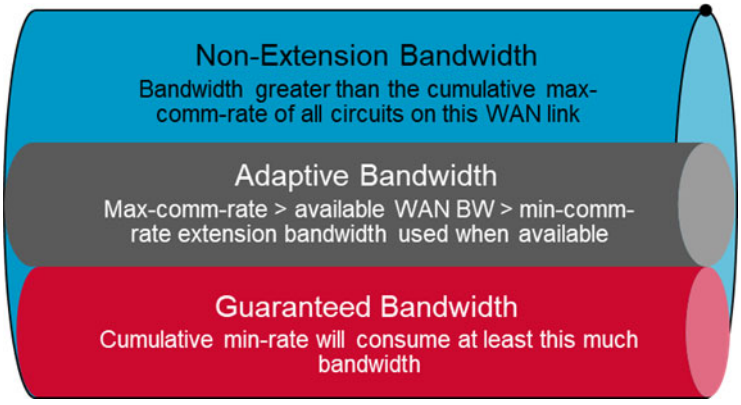
See Figure 20 and assume a 1Gb/s WAN. The ARL max-comm-rate is set to either the GE interface line rate or the maximum available WAN bandwidth, whichever is lowest. In this case, the max-comm-rate is set to 1Gb/s.

Each circuit is configured with a floor (min-comm-rate) and ceiling (max-comm-rate) bandwidth value in b/s (bits per second). Assuming adequate bandwidth demand from the storage application, the minimum circuit bandwidth will never be less than the min-comm-rate and will never be more than the max-comm-rate. The available bandwidth to the circuit will adjust automatically between the min and max based on IP network conditions. A congestion event causes the rate limit to readjust toward the minimum. An absence of congestion events causes it to rise toward the maximum. If the current rate is not at the maximum, ARL will periodically attempt to adjust upward; however, if another congestion event is detected, the rate will remain stable.

The ARL min-comm-rate is set to the max-comm-rate divided by the number of circuits that are feeding the WAN connection. In this example, 1Gb/s ÷ 4 = 250Mb/s. The min-comm-rate is set to 250Mb/s. When all the circuits are up, each will run at 250Mb/s. In an extreme case in which three circuits have gone offline, the remaining circuit will run at 1Gb/s. At 1Gb/s all the WAN bandwidth continues to be consumed and the replication application remains satisfied.

When more than one circuit is feeding a WAN link, the two circuits equalize and utilize the available bandwidth as shown in Figure 20. When an interface or the entire platform goes offline, ARL will re-adjust to utilize bandwidth that is no longer being used by the offline circuits. This maintains the utilization of WAN bandwidth during periods of maintenance or failures.

Figure 20: ARL Min and Max Comm Rates



In a shared WAN, consider bandwidth as separated into three distinct areas:

- Guaranteed Bandwidth (0 → min)
- Adaptive Bandwidth (min → max)
- Non-Extension Bandwidth (max → WAN bandwidth)

ARL manages the bandwidth in these areas. Red is the minimum bandwidth used by Extension on the WAN link. This is the amount of bandwidth reserved exclusively for Extension. It is important to note that the minimum bandwidth used will be the aggregate of *all* circuit minimums on the WAN link. Blue is reserved exclusively for other traffic that is sharing the WAN link. The bandwidth where the blue section starts is the aggregate of *all* maximum circuit values on the WAN link. Gray is the area between the top of the red section and the bottom of the blue section. Extension circuits may use this bandwidth when available; other applications that share the WAN are not currently using it. There are many ways in which ARL can be used.

12.2 WAN Side

Brocade Extension can conceptually be considered as having three sides: FC/FICON, WAN, and LAN. This section is specific to the WAN side, which is the side that faces the WAN and forms the tunnel and circuits that traverse the WAN. There are only VE_Ports and non-LAN Ethernet interfaces on the WAN side.

12.2.1 LLDP

A best practice is to use Link Layer Discovery Protocol (LLDP) as an indicator of Ethernet connectivity. LLDP is used by both the Extension platform administrator and the network administrator. LLDP information is logged on both ends. Network administrators see what is connected to their ports. Storage sees what it is connected to. For troubleshooting and adds/moves/changes, LLDP is a handy and convenient tool.

LLDP is enabled by default and operates on all data-bound GE interfaces. It does not operate on the management interface. Most DC LAN Ethernet switches support and by default have LLDP enabled.

There are certain TLV (type, length, value) that should be enabled and disabled when connecting Extension to a DC LAN switch:

- Enable
  - Chassis ID
  - System Name
  - System Capabilities
  - System Description
  - Port ID
  - Port Description
  - Management Address
- Disable
  - dcbx
  - fcoe-app
  - fcoe-lls
  - dot1
  - dot3

12.3 FCIP

12.3.1 Compression

Use compression with Remote Data Replication (RDR) applications, including RDR/S. Commonly, tape traffic is already compressed, and compressing data again is not helpful.

NOTE: Broadcom makes no guarantees, warranties, or claims as to the actual compression ratio that will be achieved with customer-specific data.

There are three modes of compression besides disabled: Fast-Deflate, Deflate, and Aggressive-Deflate.

12.3.1.1 Fast-Deflate

Fast-Deflate typically gets about a 2:1 compression ratio. Fast-Deflate is a hardware-implemented (FPGA) compression algorithm suitable for synchronous applications and adds a mere 10 µs of propagation delay and accommodates large bandwidth rates. In hybrid mode on the Brocade 7840 and SX6 Blade, a 2:1 fast-deflate compression ratio can accommodate 20Gb/s per DP of FC traffic. In FCIP mode, the Brocade 7840 and SX6 Blade can accommodate 40Gb/s per DP of FC traffic.

12.3.1.2 Deflate

Deflate typically gets about a 3:1 compression ratio. Deflate accommodates up to 16Gb/s ingress from the FC side per DP. Deflate has been designed to work efficiently with circuits up to 5Gb/s per DP. Deflate is a software-processed algorithm with hardware assist. Software-based algorithms take longer to process and may not be suitable for synchronous applications.

12.3.1.3 Aggressive-Deflate

Aggressive-Deflate takes the trade-off between the compression ratio and the compression rate further. Aggressive-Deflate typically gets about a 4:1 compression ratio. The maximum rate per DP is 10Gb/s ingress from the FC side. Aggressive-Deflate has been designed to work efficiently with circuits up to 2.5Gb/s per DP. Aggressive-Deflate is a software-based algorithm and is not suitable for synchronous applications.

12.3.2 FCIP Architectures

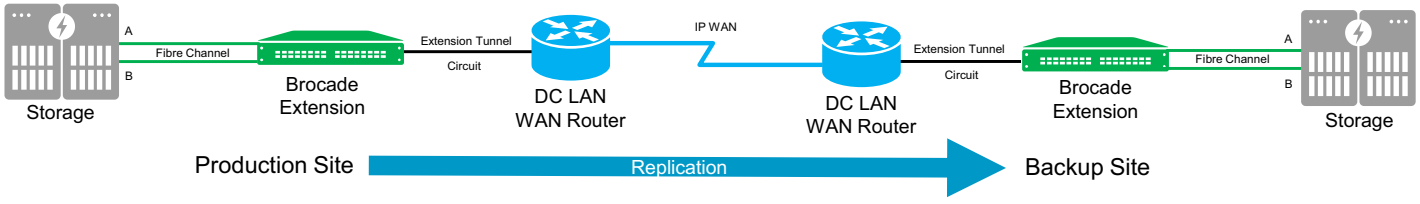
Extension is most commonly used for business continuance via disaster recovery. Leveraging Remote Data Replication (RDR) and remote tape applications to transport critical data across a significant enough distance outside a catastrophic event preserves data. Data preservation permits an organization to recover operations.

RDR is typically disk-array–to–array communications. The local storage array at the production site sends data to the array at the backup site. RDR can be done via native FC if the backup site is within a practical distance and there is WDM or dark fiber between the sites. However, a cost-sensitive ubiquitous IP infrastructure, not native FC connectivity, is most commonly available.

Brocade Extension is high speed and adds only about 50 µs of propagation delay per pass through an Extension platform (four passes round trip = 0.2 ms). It is appropriate for both asynchronous RDR and synchronous RDR applications. Best-practice deployment is to connect array N_Ports directly to Extension F_Ports, and not connect through a production fabric. Storage replication ports are dedicated to RDR and should have no host traffic. Nevertheless, there remain valid reasons to connect via the production fabric, such as tape applications and too many array ports for the Extension platform.

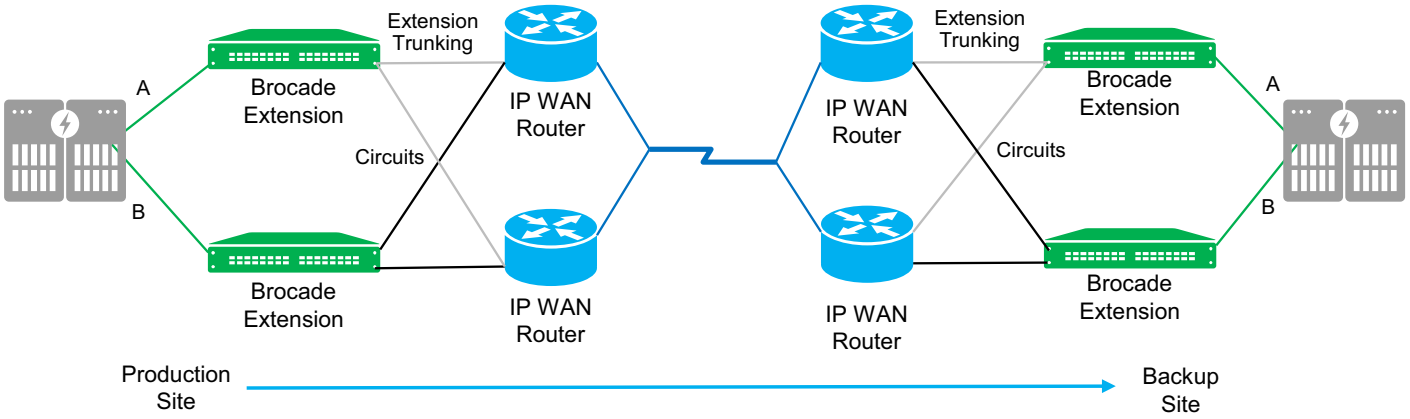
A single Extension platform can be directly connected to both “A” and “B” storage controllers. This is referred to as a two-box solution, one at each DC as shown in Figure 21.

Figure 21: Basic Non-Redundant Extension Architecture



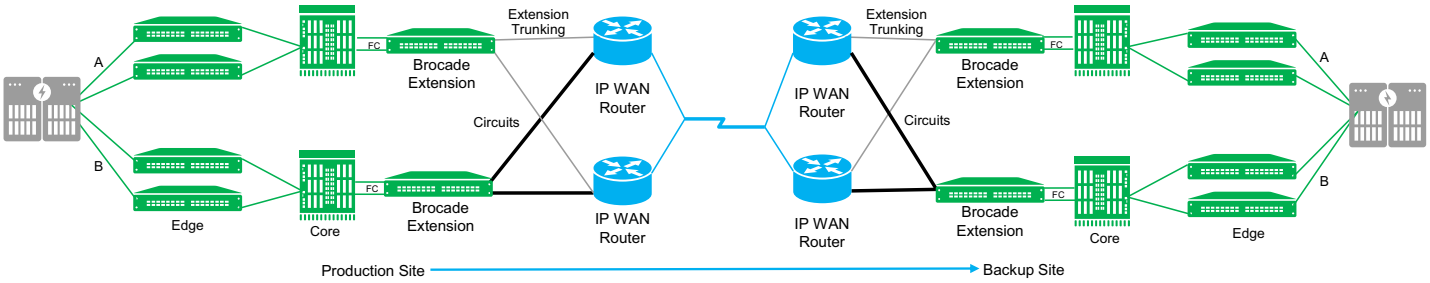
Alternatively, when the Extension platform is dedicated to the “A” fabric or controller and a physically different Extension platform is dedicated to the “B” fabric or controller, this is referred to as a four-box solution, as shown in Figure 22. A single WAN link for both paths may be used, or different service providers may be used. This depends on the requirements, the cost, and the recovery tolerance.

Figure 22: FCIP Architecture with Dedicated Extension for Each Controller



A production fabric can be extended using Extension as shown in Figure 23, but do not do so unless there is a compelling reason. The most common reason is distributed systems tape to gain connectivity to many devices then pipelining the tape traffic back to a DR DC.

Figure 23: Extension of a Routed SAN

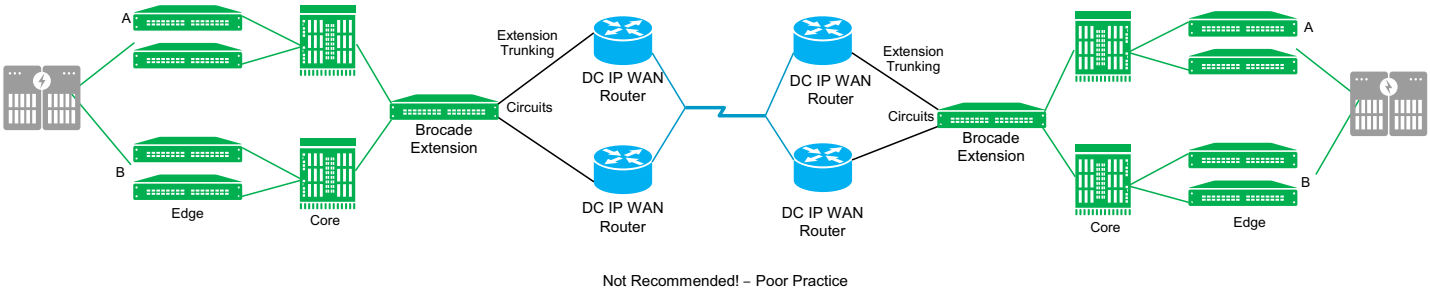


In environments that require Extension attached to a production SAN, it is not best practice to interconnect the same Extension platform to both the “A” and “B” fabrics. A best practice is to have two separate and redundant fabrics in a production environment, especially if the organization could suffer financial losses during a SAN outage. Even momentary SAN outages can “blue screen” or hang servers, forcing a reboot, which in most situations takes significant time.

For maximum availability, it is best practice to divide a SAN into “A” and “B” fabrics, which implies that there is an “air gap” between the two autonomous fabrics from the server to the storage. There are no physical links between the two fabrics. Servers, storage, and VMs are equipped with drivers that monitor pathways and send data accordingly. When a path is detected as down, the driver fails over the traffic to a remaining path.

Extension via FCR to a production fabric cannot connect to both the “A” and “B” fabrics, as shown in Figure 24. Do not use this type of architecture. Without FCR, the fabrics would merge into one big fabric, which destroys any notion of redundant autonomous fabrics. If FCR is used, the fabrics do not merge; however, there is still a common Linux kernel device running that is attached to both fabrics. If maximum availability is the goal, this architecture is not acceptable and is considered poor practice due to high risk. Additionally, an architecture with a common device to A and B fabrics is also susceptible to human error, which can bring down an entire SAN.

Figure 24: Two-Box Solution Connected to Both Production Fabrics: Poor Practice



When connecting Extension to production fabrics, each production fabric should be designed using best-practice traditional core-edge concepts. Since storage connects directly to the core in a core-edge design, Extension switches connect to the core or an Extension blade is placed in a core director. Standalone Extension platforms should be connected to a fabric with at least two inter-switch links (ISLs) for redundancy.

In a four-box solution, it is inappropriate to make ISL cross-connections between the two Extension platforms and the “A” and “B” fabrics because of the same reasons discussed above, a common Linux kernel and human error.

Cross-connecting circuits from a tunnel to various Ethernet DC LAN switches or IP network devices is encouraged. Circuits that traverse the IP network are point-to-point and can take alternate resilient and redundant paths without merging the A and B fabrics.

12.4 IP Extension

IP Extension accelerates, secures, and is used to manage supported IP storage applications by leveraging Extension technology. Much of the technology is common across the FCIP and IP Extension protocols, and they share the same tunnel. Enemies to TCP/IP-based replication are latency and packet loss. IP Extension overcomes performance degradation caused by these inherent characteristics; plus it provides encryption.

This section covers points that are unique to IP Extension design, best practice, and architectures.

12.4.1 LAN Side

As mentioned previously, Brocade Extension can be represented by having three sides. One of those sides is the LAN side. The LAN side is specific to IP Extension, and it is used to connect IP storage Ethernet ports via the connected LAN.

IP Extension supports connectivity of multiple DC LANs via VLAN tagging (802.1Q) on the Ethernet links between the Extension platform and the DC LAN switches. An IP Extension gateway (`ipif lan.dp#`) must be configured and specified for each VLAN.

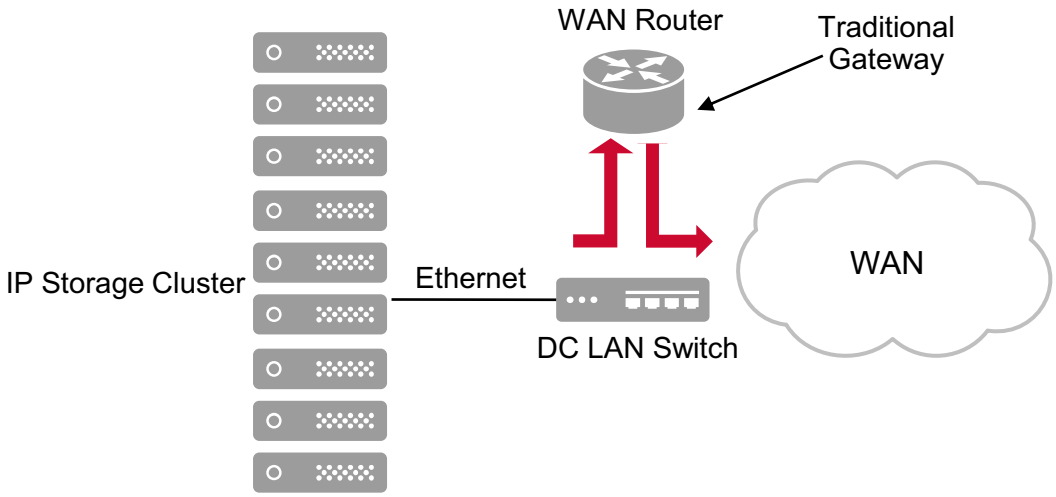
NOTE: IP subnets used for LAN-side end devices on each end of the IP Extension tunnel cannot be the same subnet.

12.4.1.1 IP Extension Gateway

IP Extension is the gateway for traffic that is meant to cross the Extension tunnel. If IP storage traffic is not forwarded to the IP Extension gateway, it will not utilize IP Extension.

In Figure 25, traffic comes from the IP storage cluster into the DC LAN switch. The DC LAN switch forwards traffic to the traditional router gateway. This router could be an inherent part of the DC LAN switch, or not. The router sends the traffic toward the destination.

Figure 25: Traditional DC Gateway

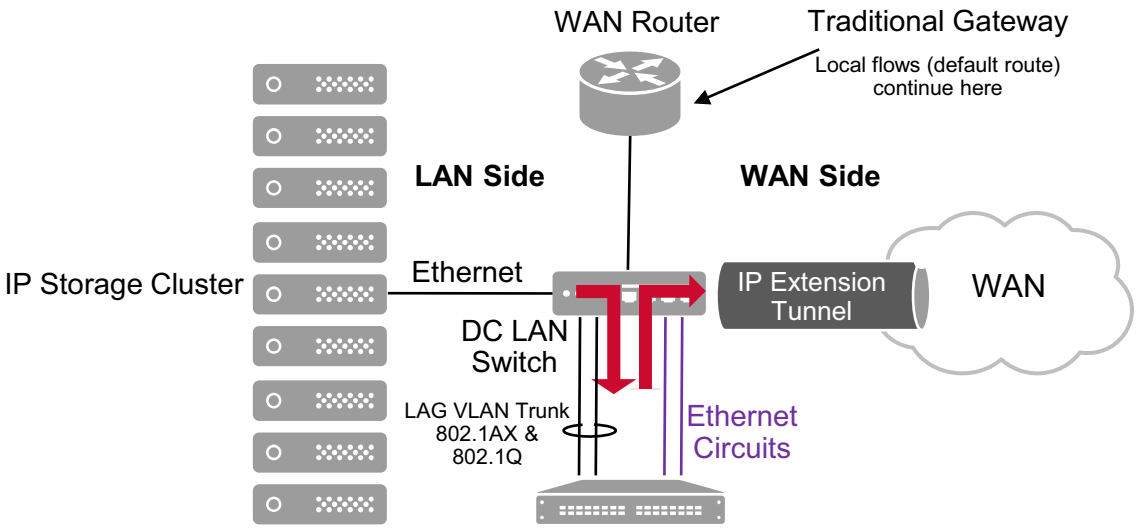


With IP Extension, it is required that the end device have a static route, or some route that is more specific than the default route, that forwards remote subnet traffic to the IP Extension gateway. The remote end device is on the remote subnet. The default route stays pointed to the traditional router gateway and is used only when a more specific route does not exist.

It is important to keep in mind that putting IP Extension in the path and removing it from the path merely involves activating or deactivating the static routes on the end devices. With static routes, the traffic goes to the IP Extension gateway (IP Extension in the path). Without static routes, the traffic goes to the traditional router (IP Extension out of the path).

In Figure 26, traffic from the remote IP storage cluster is forwarded to the IP Extension gateway where it is first evaluated by the TCL (only once when the initial TCP connection is opened). If the traffic matches a rule allowing the traffic to enter the tunnel, it is sent into the specified tunnel (target). The IP Extension traffic is now on the WAN side.

Figure 26: IP Extension Gateway



NOTE: End devices must have a static route:
For remote subnet, use IP extension gateway

12.4.1.2 GE Interfaces

GE interfaces (including interfaces set at 10Gb/s) are either WAN or LAN facing. GE interfaces cannot do both and must be configured for one or the other. The default is WAN. LAN-side connectivity can be made from GE/10GE interfaces on all platforms. The Brocade 7840 and SX6 Blade must be in app-mode “hybrid” (supports both FCIP and IP Extension) before a GE interface can be configured to be LAN facing with a maximum of 8 of 16 interfaces. The Brocade 7810 has no app-mode setting; it is only in hybrid mode; four of the six active interfaces can be configured as LAN facing.

The Brocade 7810 has two copper (RJ-45) ports, which can only do 1Gb/s. There is no advantage or disadvantage to using the copper ports (GE0 and GE1) over the reciprocal optical ports (GE0 and GE1). Speed is the only limitation on the copper ports.

NOTE: In-band management is not supported on data GE/10GE interfaces on any Extension platform.

12.4.2 Compression

Compression operates in the tunnel scope. Compression cannot be configured per circuit. Compression must be configured identically on both ends of the tunnel; asymmetrical compression is not supported. IP Extension compression is limited to Deflate and Aggressive-Deflate. Fast-Deflate is not available for IP Extension on the Brocade 7840 and SX6 Blade. The Brocade 7810 does not have Fast-Deflate because the platform does not support the capacity that the Fast-Deflate hardware accommodates.

Compression can be configured specifically for each protocol (FCIP and IP Extension). For example, on a Brocade 7840, configure Fast-Deflate for FCIP and configure Deflate for IP Extension. This is considered best practice in a hybrid environment because the Fast-Deflate and Deflate compression engines are different. Fast-Deflate uses a hardware engine, and Deflate uses a processor. 20Gb/s of FC ingress to Fast-Deflate does not consume the IP Extension capacity available for Deflate or Aggressive-Deflate.

Table 1: WAN-Side Egress Rate: Which Compression Algorithm to Use for IP Extension?

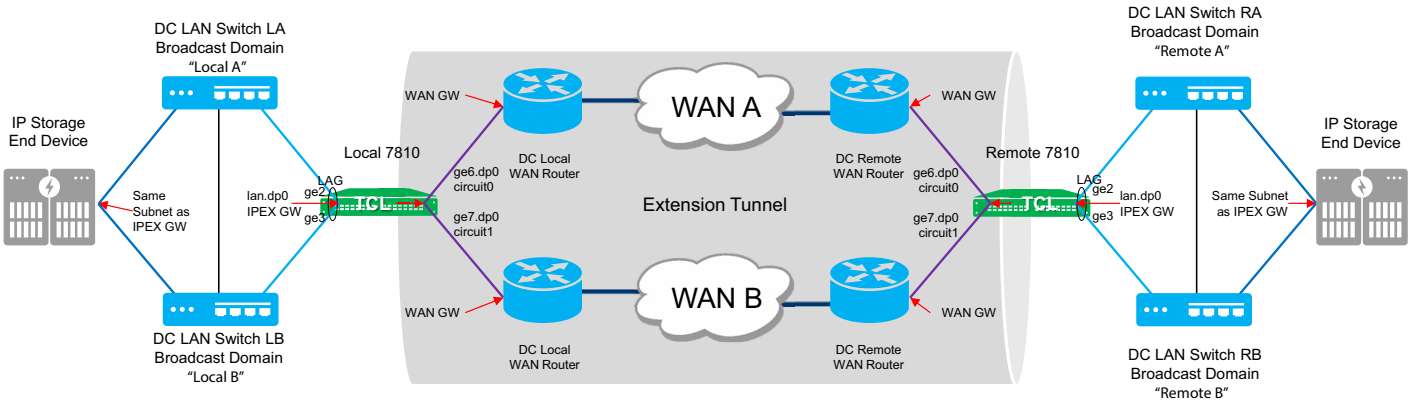
Hybrid Mode	Brocade 7810	Brocade 7840	Brocade SX6 Blade
Disabled (1:1)	2.5Gb/s (DP max egress)	20Gb/s (DP max egress)	20Gb/s (DP max egress)
Fast-Deflate (2:1)	N/A (not on the platform)	N/A (no IP Extension)	N/A (no IP Extension)
Deflate (3:1)	1.5Gb/s to 2.5Gb/s	10Gb/s to 15Gb/s	10Gb/s to 15Gb/s
Aggressive-Deflate (4:1)	20Mb/s to 1.5Gb/s	20Mb/s to 10Gb/s	20Mb/s to 10Gb/s

NOTE: Compression ratios are approximate. Broadcom makes no warranties, guarantees, or claims as to the actual compression ratio achieved with customer-specific data.

12.4.3 IP Extension Architectures

Data flow through an IP Extension architecture starts at the originating end device. There is a static route on that device that says that if the destination is subnet “x,” use the IP Extension gateway. The IP Extension gateway is on the same L2 network, and the end device forwards the traffic to the gateway. Traffic that matches a traffic control list is put in a tunnel and forwarded to the remote side via the IP WAN network. On the remote side, the traffic is removed from the tunnel and forwarded to the destination end device. See Figure 27.

Figure 27: Data Flow through IP Extension Architecture

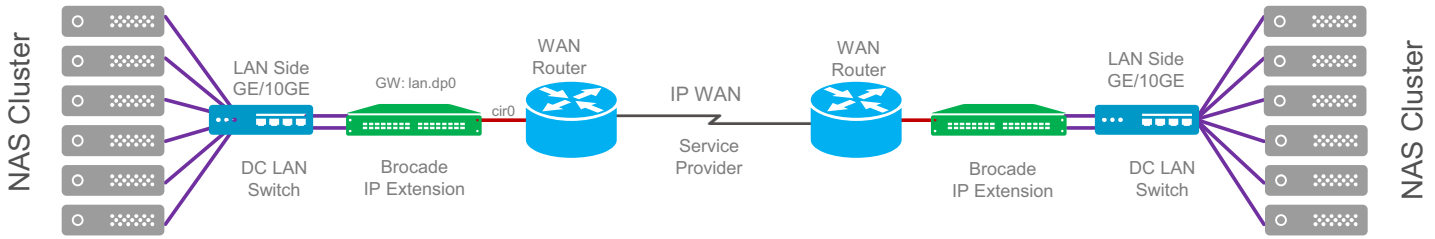


Various IP Extension architectures can be built. These range from simplistic and cost-effective to more complex, higher availability, and higher capacity.

12.4.3.1 Two-Box Solutions

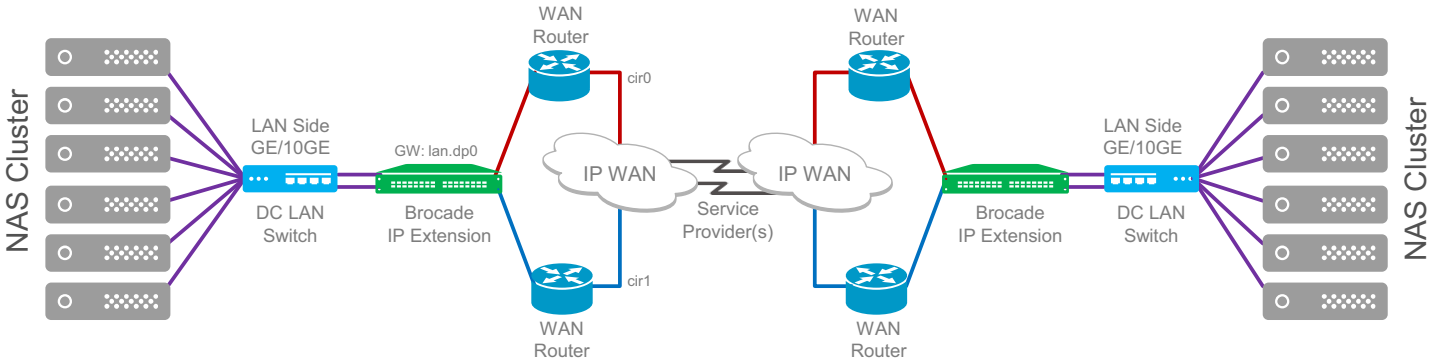
The two-box solution that is not using Brocade Extension Trunking (BET) is typically used with the base Brocade 7810. The base Brocade 7810 is the most cost-effective Extension platform. It does not support BET; however, BET can be enabled with an upgrade license. Enabling BET is required to create more than one circuit per tunnel; therefore, the architecture in Figure 28 shows only a single circuit.

Figure 28: Brocade 7810 Base Unit Architecture



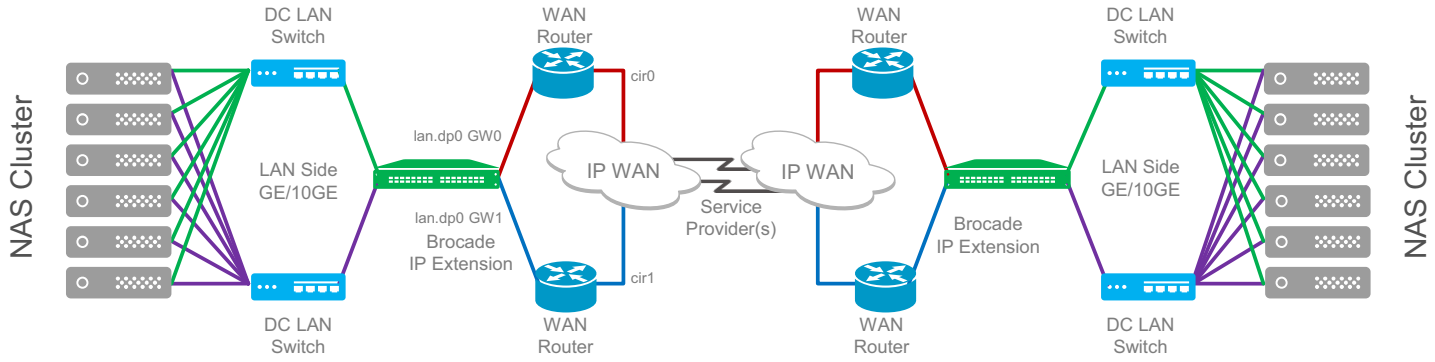
The two-box solution using BET is a more common architecture than without BET. Connecting the WAN side to A and B switches/routers gains higher availability; see Figure 29. The tunnel remains up and undisturbed when a switch/router/optic/cable/WAN connection goes offline. All traffic will be delivered and delivered in order, although a portion of the bandwidth may no longer be available. In this architecture, the DC LAN switches and Brocade Extension platform remain single points of failure.

Figure 29: Two-Box Solution Using BET: Single DC LAN Switch



The two-box solution using BET with dual DC LAN switches eliminates the DC LAN switches as single points of failure. This architecture requires that the IP storage devices be capable of multiple gateways specific to their Ethernet interfaces. As shown in Figure 30, each end device has dual NICs, each connected to a different DC LAN switch. The DC LAN switch, in turn, connects to the Extension platform. The same IP Extension gateway cannot accommodate the two separate DC LAN switches. The green path requires its own VLAN and IP Extension gateway, as does the purple path. The Brocade Extension platform is a single point of failure in this architecture.

Figure 30: Two-Box Solution Using BET: Dual DC LAN Switches



12.4.3.2 Four-Box Solutions

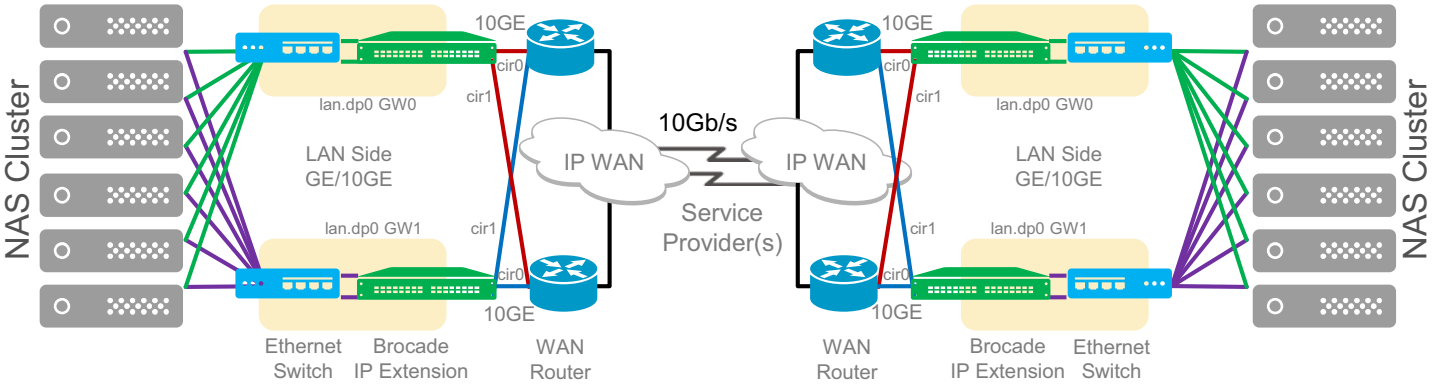
Avoiding a single point of failure may be paramount in some environments. A four-box Brocade Extension solution is possible when the end device supports the ability to accommodate more than one gateway. Deploying more than one local IP Extension platform requires more than one gateway. Brocade IP Extension on the LAN side does not offer Virtual Router Redundancy Protocol (VRRP) or Hot Standby Router Protocol (HSRP), which provide a single virtual gateway. Therefore, the end device must be capable of the following:

- A different subnet, static route, or gateway per Ethernet interface or set of Ethernet interfaces
- The same subnet with the ability to configure a unique static route per Ethernet interface or set of Ethernet interfaces

The DC LAN switches in this architecture are not configured to form a single logical Ethernet switch; see Figure 31. The two DC LAN switches are autonomous. The NAS cluster can configure multiple unique gateways, not including the default gateway. The IP Extension gateway is not the default gateway. When sending replication traffic to the remote NAS cluster, the green connections forward traffic to lan.dp0 GW0 (top), and the purple connections forward traffic to lan.dp0 GW1 (bottom).

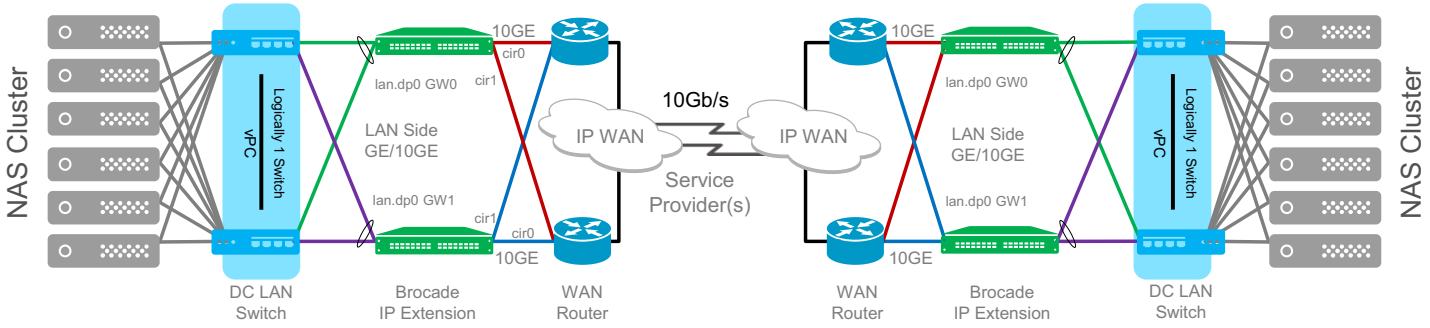
On the WAN side, there are two Brocade extension trunks (red and blue) with two circuits each. Each circuit connects to a different WAN switch/router. While most data centers implement VRRP, it is still necessary to connect to different WAN switches in the event that a switch goes down, an optic fails, or a cable is damaged. Most often, the gateway used on the WAN side is the virtual gateway created by VRRP and is accessible from both WAN switches. The Brocade Extension platforms are not single points of failure in this architecture.

Figure 31: Dual Connected High-Availability Architecture



The DC LAN switches in Figure 32 are logically a single Ethernet switch. There is a virtual port channel (vPC) connection that allows the switches to join and appear as one to the network. Brocade Extension can form port channels not only with each individual DC LAN switch but across both. The DC LAN switches must be configured as a single logical switch to do so. The purpose of this architecture is to gain redundant DC LAN switches. When one of the switches goes offline, the other switch continues to forward traffic to the IP Extension gateway. The Brocade Extension platforms are not single points of failure in this architecture.

Figure 32: Four-Box Solution with Single Logical DC LAN Switch



Chapter 13: SAN Design for Critical Workloads

With all-flash arrays (AFAs) being the standard in enterprise data centers today and the transition to FC-NVMe AFAs beginning, critical business applications are increasingly dependent on consistent low-latency, high-throughput storage performance for demanding, performance-sensitive workloads. When designing the SAN, it is important to consider the placement of critical workloads (in relation to the placement of storage), the fan-in ratio to storage ports, and the fan-in ratio to ISLs/trunks.

Whereas Brocade SAN technology provides a suite of built-in measures to avoid interference of workloads that expose any kind of congestion behavior, such as Traffic Optimizer, FPIN, MAPS, and SDDQ, protecting critical workloads is crucial. Ideally the most demanding and critical workloads have dedicated storage array target ports (or even dedicated arrays) and the shortest possible path through the SAN. The purpose for this is to avoid any interference of other less critical workloads that may expose behavior that results in congestion or back pressure, which could interfere and adversely impact the performance of critical workloads.

13.1 Placement of Servers with Business-Critical Workloads

With core-edge SAN designs, it is often advantageous to connect servers with critical workloads directly on the core together with the storage array ports. This works well when the number of business-critical workloads is easily defined to a specific subset of servers in the SAN and there is an adequate number of ports available on the core to connect both storage and business-critical servers.

In the case where the number of business-critical servers exceeds the number of available ports on the core for server connections, it is necessary to connect the business-critical servers on edge switches. The most common model is then to use dedicated edge switches for business-critical servers and decrease the fan-in ratio of servers to ISLs.

An alternative model is to attempt to evenly distribute business-critical servers across edge switches with the assumption that the workload pressure evens out with the less critical and demanding workloads on other servers. Although this may seem to be a logical approach, experience has shown that throughout the lifetime of applications and the SAN, it is operationally more complex to guarantee optimal performance for business-critical workloads with this model.

13.2 Business-Critical VMs

In today's data centers, it is not uncommon for some or all business-critical workloads to be running on VMs. Combining this with the digital business environment, the value (business criticality) and performance requirements for a given application often change throughout the life cycle of the application. This means that it can be difficult (or impossible) to predict the requirements for an application and plan placement accordingly from the beginning—luckily hypervisors provide the ability to move VMs between hosts without downtime and to migrate storage when necessary. To apply the same principle of server placement of (bare metal installed) business-critical applications to VMs, deploy dedicated hypervisor clusters connected on the core or on high-performance edge switches and use these hypervisor clusters only for business-critical and performance-sensitive VM workloads.

Visibility into each VM's workload on the same data store (backed by LUN or NSID) can be achieved with VM Insight. VM Insight enables storage administrators to monitor VM-level application performance and set baseline workload behavior. This information can be used to quickly determine whether a storage fabric is the source of performance anomalies for applications running on VMs. Based on VM Insight, storage administrators can provision and plan placement in the SAN on application requirements and can fine-tune the infrastructure to meet service-level objectives.

Chapter 14: Access Gateway and NPIV

In this chapter we go over SAN design considerations when using Access Gateway (AG) and N_Port ID Virtualization (NPIV). These considerations are mainly related to increased density and scale. In addition we describe AG default port mapping and best practices on how to design port mapping for specific SAN designs (if your environment so requires) while ensuring balancing and failover. For detailed information on deploying and configuring Access Gateway, refer to the *Brocade Fabric OS Access Gateway User Guide*.

Standards-based NPIV can be used to connect multiple F_Ports to the fabric on a single N_Port. These F_Ports can be virtual initiator ports from a single physical HBA port connecting to the fabric such as from a hypervisor with virtual HBAs or a storage device with multiple virtual target ports on the same physical HBA port.

Another use case for NPIV is where a switch is configured in NPIV mode and thereby connects to the SAN fabric as an Access Gateway (AG). An AG does not participate in the fabric as a switch; rather it extends the number of ports that are connected to the fabric without increasing the number of domains in the fabric. Placing a switch in NPIV mode is also a way to connect a switch from a different vendor to the fabric. Brocade fabrics support connecting switches, in NPIV mode, from other vendors that follow standards-based NPIV.

Common NPIV use cases include the following:

- Using Brocade Access Gateway to increase port count without increasing the number of domains, often deployed in a POD architecture design.
- Connecting embedded server blade chassis switches.
- Connecting switches from other vendors (including UCS-FIs).
- Storage arrays with a virtual storage controller architecture presenting separate virtual target ports *behind* the same physical target port.
- Hypervisors with virtual HBAs provisioned to VMs for Raw Device Mapping (RDM) allocation of storage.

When connecting a switch in AG mode, the F_Ports connect to the fabric as N_Ports rather than as E_Ports, as illustrated in the following two figures. Figure 33 shows a switch in native mode with all devices connecting to F_Ports and switch-to-switch connections as E_Ports (ISLs).

Figure 33: Switch Functioning in Native Mode

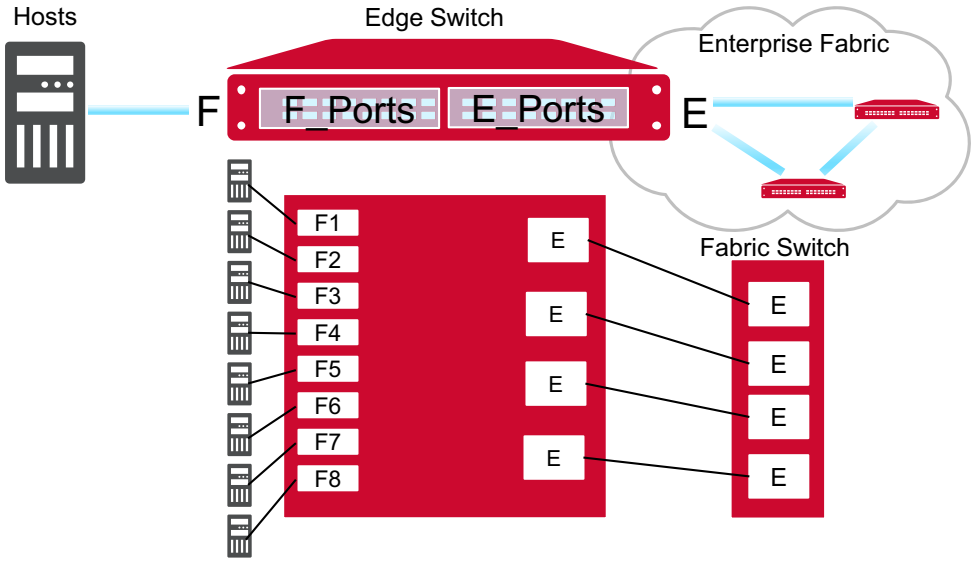
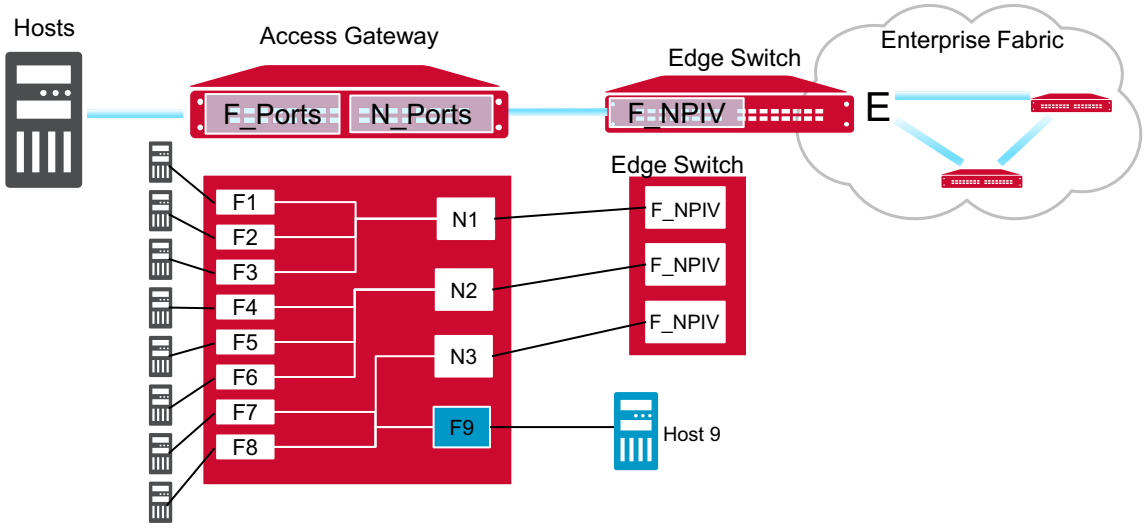


Figure 34 shows a switch in Access Gateway mode with all devices connecting to F_Ports, which then map to N_Ports, providing the AG to switch connectivity.

Figure 34: Switch Functioning in Access Gateway Mode



Switches in AG mode are logically transparent to the host and the fabric. Therefore, you can increase the number of hosts that have access to the fabric without increasing the number of switch domains. AG mode simplifies configuration and management in a large fabric by reducing the number of domain IDs and ports. In turn, a fabric-specific configuration is not available on the AG but is *inherited* from the fabric.

The main reason for using AG mode is to achieve scalability with a large number of small switches. In an environment with many blade servers, the embedded switches can easily start to encroach on the limits of total domain count in a fabric. Placing these switches in AG mode means that they will not consume a domain ID in the fabric.

The main downside to AG mode in previous generations of Fabric OS was the functionality (or feature set) available. With current levels of Fabric OS, AG functionality is enriched, although there are still some scenarios where switch functionality is more advanced. Therefore, deciding to use AG in a SAN design involves an evaluation of the SAN environment and the desired functionality to determine if AG is a design option for the environment. In a fabric with many legacy devices, identifying and isolating misbehaving devices is easier to do in a full-fabric environment.

Also, for configurations with hosts and targets on the same AG, the traffic does need to go through the fabric switch, but it is handled within the local switch and does not need to traverse another switch in the fabric and then back again. The theoretical domain limit in a single fabric is 239, but most fabrics are typically limited to a much smaller number (56 is the maximum number of domains supported in Brocade fabrics). The domain count limit typically comes into play when a large number of small-port-count switches are deployed. Large-bladed server deployments, for example, can easily push the domain count over recommended limits when embedded blade switches are part of the implementation. FC switches in blade server enclosures typically represent fewer than 32 ports.

NPIV was originally developed to provide access to Fibre Channel devices from IBM mainframes and to improve the efficiency of mainframe I/O for virtualized environments.

14.1 Benefits of the Brocade Access Gateway

- **Scalability**: You can add many Access Gateways to a fabric without increasing the domain count. A major scalability constraint is avoided when small-port-count switches or embedded switches are part of an infrastructure. Registered state change notifications (RSCNs) are also greatly reduced; only those that are related to the initiators on the downstream Access Gateway ports are passed on through to the fabric. Since it is essentially a device, the Access Gateway can connect to more than one fabric from its upstream ports. Brocade Access Gateways can be cascaded to reduce the number of fabric connections required to support a given workload or traffic level from the attached hosts.
- **Error isolation and management**: Most initiator errors are not propagated through to the fabric. Disconnecting an upstream port, for example, does not cause a fabric rebuild. Most management activities on the Brocade Access Gateway are also isolated from the fabric. One possible scenario is server administrators managing the Access Gateways and storage administrators simply providing LUNs and zoning support for the servers using NPIV.
- **Increased resiliency**: The Brocade Access Gateway supports F_Port trunking, which increases the resiliency of connections into the fabric. Losing a trunk member simply reduces the bandwidth of the upstream trunk. Although a few frames may be lost, no host connections are affected.
- **Other**: Hosts or HBAs can be configured to automatically fail over to another upstream link should the one they are using fail. The Brocade Access Gateway also implements many advanced features such as adaptive networking services, trunking, hot code load, Brocade MAPS, Brocade ClearLink, credit recovery, and forward error correction.

14.2 Constraints

The advantages of the Brocade Access Gateway are compelling, but there are constraints:

- Although benefits are much more obvious for servers, the Brocade Access Gateway supports storage devices, but the traffic must flow through the fabric, which has its own limitations.
- The maximum number of NPIV connections per upstream port is 254.
- The number of Brocade Access Gateways per switch is limited only by what the fabric switches can support.

The primary factors to consider follow:

- The total number of devices that attach to the fabric through the Access Gateways
- The number of devices per Access Gateway N_Port
- The total number of devices attached to the switch and fabric

Refer to the *Brocade SAN Scalability Guidelines* for details.

The number of fabrics to which a single Brocade Access Gateway can be connected is limited to the number of N_Ports on that Access Gateway. In general, most deployments require a single Access Gateway connection to only one or two fabrics. Note that the ability to connect different upstream ports to different fabrics does not reduce the requirement for redundancy. All attached servers should have dual paths to their storage through different fabrics via separate Access Gateways.

14.3 Design Guidelines

Use the Brocade Access Gateway when you deploy blade servers or have many low-port-count switches and need to connect different servers in different fabrics from a single bladed enclosure. The Access Gateway can be very valuable when you want to separate the management of blade enclosures so that the enclosure is completely managed by server administrators and the fabric is handled by storage administrators. Management separation is provided through the NPIV connection, which allows the Access Gateway to be managed separately by, for example, integrated blade server enclosure management tools, without any adverse effects on the fabric.

14.4 Monitoring

Brocade Access Gateway has been enhanced to include many features found in the standard version of Brocade FOS, such as Port Fencing, device security policies, FPI monitoring, and SDDQ, although monitoring and troubleshooting NPIV flows is generally less feature rich than traditional flows.

14.5 Maintenance

There is usually no need to keep the Brocade Access Gateway firmware levels synchronized with the firmware levels deployed in the fabrics to which it is connected (and Brocade supports connections from other vendors' NPIV-enabled devices, where firmware synchronization is impossible). This can be significant for very large fabrics with many devices, including many Access Gateways. The version of Brocade FOS running on fabric switches can be upgraded at one time and the Access Gateways at another time, which greatly reduces the amount of change required to the infrastructure during a single maintenance window.

See the *Brocade Fabric OS Release Notes* to determine if a synchronized Brocade FOS upgrade of Brocade Access Gateway devices is required.

14.6 Access Gateway Mapping

When a switch operates in AG mode, you must specify the routes that the AG device uses to direct traffic from the devices on its F_Ports to the fabric ports connected to its N_Ports. The routes must be preprovisioned. The process of preprovisioning routes in AG mode is called mapping. (By comparison, a switch operating in Native mode determines the best routing path between its F_Ports.)

You can create two types of maps: port maps and device maps. Port maps are required. Device maps are optional and assign device WWNs to N_Ports and N_Port groups. Port mapping and device mapping operate as follows.

14.6.1 Port Mapping

Port mapping ensures that all traffic from a specific F_Port always goes through the same N_Port. A single F_Port is mapped to a single N_Port or to an N_Port group. To map an F_Port to an N_Port group, map the F_Port to an N_Port that belongs to that port group. All F_Ports mapped to an N_Port group are part of that N_Port group.

14.6.2 Device Mapping

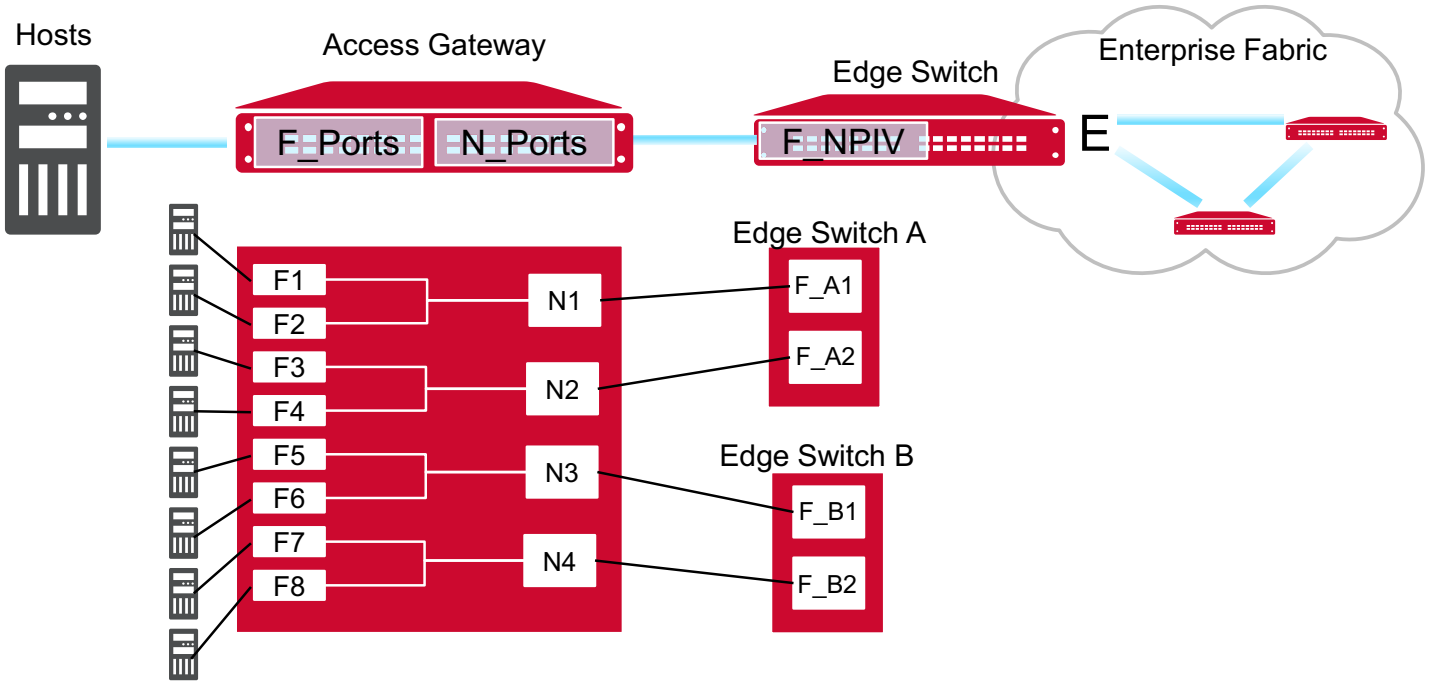
Device mapping is optional. Port maps must exist before you can create device maps. Device mapping allows a virtual port to access its destination device regardless of the F_Port where the device resides. Device mapping also allows multiple virtual ports on a single physical machine to access multiple destinations residing in different fabrics.

The preferred method is to map a device WWN to an N_Port group. When a device WWN is mapped to an N_Port, and a failover N_Port is also specified, the device can reach the fabric through the primary or secondary N_Port only. However, when a device WWN is mapped to a port group, it can log in to the fabric until the last N_Port in the particular port group remains online.

You can map a device to multiple groups. Alternatively, you can map a device to a specific N_Port.

F_Ports must be mapped to N_Ports before the F_Ports can come online. Figure 35 shows an example in which eight F_Ports are mapped evenly to four N_Ports on a switch in AG mode. The N_Ports connect to the same fabric through different edge switches.

Figure 35: Port Mapping Example



The following table shows the port mapping illustrated in the figure. F_Ports F1 and F2 map to N_Port N1; F_Ports F3 and F4 map to N_Port N2, and so forth.

Table 2: Description of Port Mapping

Access Gateway		Fabric	
F_Port	N_Port	Edge Switch	F_Port
F1, F2	N1	Switch A	F_A1
F3, F4	N2	Switch A	F_A2
F5, F6	N3	Switch B	F_B1
F7, F8	N4	Switch B	F_B2

14.6.3 Default Port Mapping

When you enable AG mode on a switch, a default mapping is used for the F_Ports and N_Ports.

The following table describes the default port mapping for a G720. Refer to the *Brocade Fabric OS Access Gateway User Guide* for default mappings on all supported hardware platforms.

Table 3: Access Gateway Default Port Mapping for a G720

Brocade Platform	Total Ports	F_Ports	N_Ports	Default Port Mapping
G720	64	0-39, 48-63	40-47	0-6 mapped to 40 7-13 mapped to 41 14-20 mapped to 42 21-27 mapped to 43 28-34 mapped to 44 35-39, 48-49 mapped to 45 50-56 mapped to 46 57-63 mapped to 47

NOTE: By default, failover and failback policies are enabled on all N_Ports.

The default mapping can be changed to meet specific requirements for your environment. For more information, refer to the *Brocade Fabric OS Access Gateway User Guide*.

Chapter 15: Security

There are many components to SAN security in relation to SAN design, and the decision to use them is greatly dependent on installation requirements rather than network functionality or performance.

When you design your SAN security policy, you do not need to implement and enable every available security feature. Some security features add performance overhead, others may affect administrator productivity, and still others may have associated implementation costs. A balance must be struck between the features and the value of the assets being protected and the probability that the system vulnerability will actually be exploited.

One clear exception is the zoning feature used to control device communication. The proper use of zoning is key to fabric functionality, performance, and stability, especially in larger networks. Other security-related features are largely mechanisms for limiting access and preventing attacks on the network (often mandated by regulatory requirements), while not required for SAN fabric operation.

This chapter covers best practices for secure communication within the SAN and secure access and protection of the SAN infrastructure.

15.1 Zoning: Controlling Device Communication

The SAN is primarily responsible for the flow of data between devices. Managing this device communication is of utmost importance for the effective, efficient, and secure use of the storage network. Brocade zoning plays a key role in the management of device communication. Zoning is used to specify the devices in the fabric that should be allowed to communicate with each other. If zoning is enforced, devices that are not in the same zone cannot communicate.

In addition, zoning provides protection from disruption in the fabric. Changes in the fabric result in notifications (RSCNs) being sent to switches and devices in the fabric. Zoning puts bounds on the scope of RSCN delivery by limiting their delivery to devices when there is a change within their zone (this also reduces the processing overhead on the switch by reducing the number of RSCNs being delivered); and it limits the impact in the rare cases where a faulty adapter creates “noise.” Thus, only devices in the zones impacted by the change are disrupted. Based on this fact, the best practice is to create zones with one initiator and one target with which it communicates (“single-initiator zoning”), so that changes to initiators do not impact other initiators or other targets and disruptions are minimized (one initiator and one target device per zone), as illustrated in Figure 36. In addition, the default zone setting (what happens when zoning is disabled) should be set to No Access, which means that devices are isolated when zoning is disabled.

Zones can be defined by either switch port or device World Wide Name (WWN). Although it takes a bit more effort to use WWNs in zones, it provides greater flexibility; if necessary, a device can be moved to anywhere in the fabric and maintain valid zone membership.

15.1.1 Peer Zoning

As the number of zones increases, it may become difficult to configure and maintain the zones using the single-initiator zoning practice. In addition, the storage requirements of defining unique zones for each host and target may exceed the zone database size limits. “One-to-many zoning” defines a zone that has one target and other members as initiators. This approach has the advantages of being easier to manage and avoiding reaching the zone database size limits on the switches in a larger fabric; but zoning all the initiators together in this manner results in less effective use of hardware resources and

greater RSCN traffic. Prior to the availability of peer zoning, it was not uncommon to zone multiple initiators of the same kind (OS) together with a single or even multiple targets to achieve operational efficiency when provisioning, although this is not in line with best practices. In today's SANs, using peer zoning provides both operational efficiency and effective single-initiator zoning while reducing the zone database usage.

Peer zoning allows one or more *principal* devices to communicate with the rest of the devices (*nonprincipal* devices) in the zone as if they were a single-initiator zone. Nonprincipal devices in the zone can communicate with the principal devices only, but they cannot communicate with each other—nor can principal devices communicate with each other. This approach establishes zoning connections that are effectuated as single-initiator zoning with the operational simplicity of one-to-many zoning and reduced zone database usage.

A peer zone can have one or multiple principals. In general storage ports are assigned as principals. Multiple principal members in a peer zone are used when all the nonprincipals (initiators) in the zone are to share the same target (storage) ports.

The peer zone members can be defined as WWNs and aliases specifying WWNs or "domain, port" and aliases specifying "domain, port" ("domain, index"), but you cannot mix WWNs and "domain, port" or corresponding aliases when defining peer zoning.

15.1.2 Target-Driven Zoning

Target-driven zoning is a variant of peer zoning. Where a regular peer zone is defined by a user-specified configuration, a target-driven peer zone is defined by the principal device. This device is usually a storage array (but does not have to be).

Target-driven zoning manages the zoning using a third-party management interface to manage the device and the switch interactions. To permit a target-driven peer zone, target-driven zoning must be enabled on the F_Port that connects the principal device to the fabric. Refer to the manual (for example, from the storage array vendor) for the device used as the principal device to determine the supported commands and options to construct a target-driven peer zone. Refer to the latest *Brocade Fabric OS Administration Guide* and your storage vendor for additional limitations and considerations.

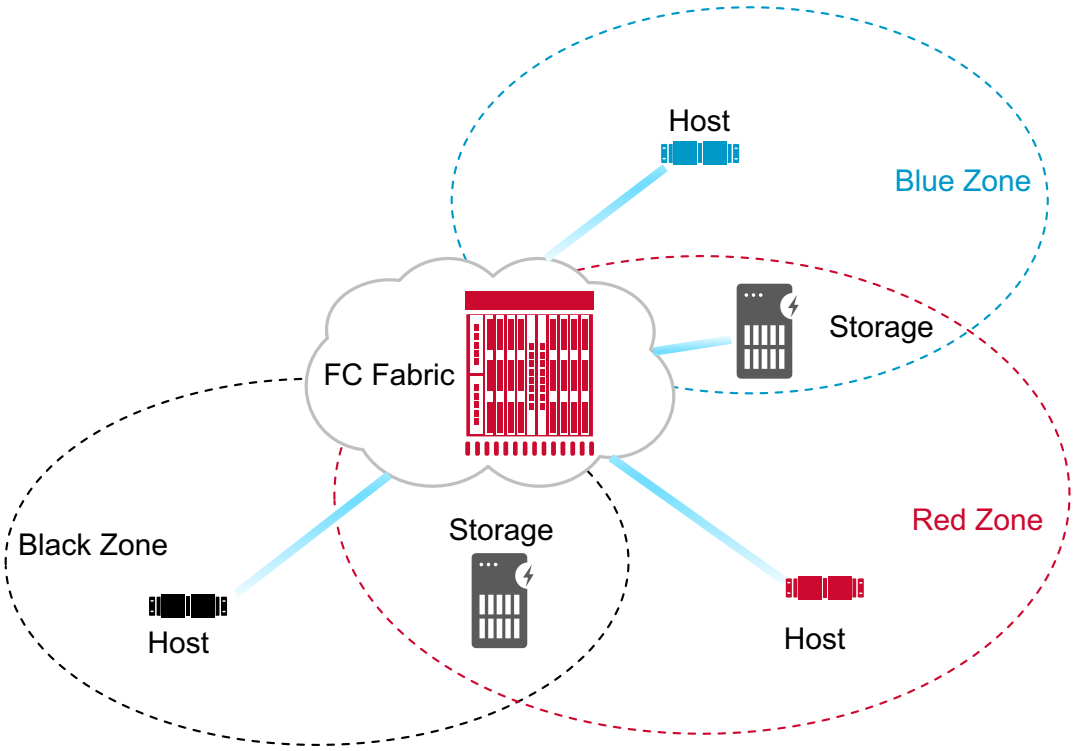
15.1.3 Zone Management: Duplicate WWNs

In a virtual environment like VMware or HP's Virtual Connect, it is possible to encounter duplicate World Wide Names (WWNs) in the fabric, most often as a transient condition. This impacts the switch response to fabric service requests like "get port WWN," resulting in unpredictable behavior, and it represents a security risk by enabling spoofing of the intended target. The fabric's handling of duplicate WWNs is not meant to be an intrusion detection tool but a recovery mechanism. Prior to Brocade FOS 7.0, when a duplicate entry is detected, a warning message is sent to the RAS log, but no effort is made to prevent the login of the second entry.

Available since Brocade FOS 7.0, the handling of duplicate WWNs is as follows:

- Same switch: The choice of which device stays in the fabric is configurable (the default is to retain the existing device).
- Local and remote switches: Remove both entries.
- Zoning recommendations include the following:
  - Always enable zoning.
  - Use peer zoning or single-initiator zoning.
  - Define zones using device World Wide Port Names (WWPNs).
  - Set default zoning to No Access.
- Follow vendor guidelines for preventing the generation of duplicate WWNs in a virtual environment.

Figure 36: Example of Single-Initiator to Single-Target Zones



15.2 Securing the SAN Infrastructure

One of the operational advantages of a Brocade SAN is the ability to easily add a new switch into the fabric. A SAN administrator need only connect a new switch to an available port on an existing switch in a fabric using an ISL and then power up the new switch. Automatically, a unique domain ID is assigned, and the configuration files are downloaded to the new switch. From a security perspective, however, this time-saving administrative ease-of-use capability also means that anyone with a switch and physical access could potentially connect to an existing fabric and gain control of the fabric. If an attacker with admin or root access on the rogue switch were to use this technique, the attacker would now have admin and root privileges for the entire fabric.

There are several layers of defense available to secure and protect the SAN. The following list (in order of ease of deployment) describes best-practice configurations (layers) to secure the SAN. At a minimum, the first four layers should be deployed in any environment, and the remaining layers should be deployed depending on the security requirements in your organization.

- Persistently disable unused ports.
- Prevent switch ports from becoming E_Ports.
- Configure monitoring, alerting, and logging.
- Use a strict fabric-wide consistency policy where possible.
- Use the SCC policy to restrict switch connections to the fabric.
- Use an FCS policy to further restrict security configuration changes.
- Use DCC policies to restrict device access by WWN and/or by a physical port on a switch.
- For more sensitive environments, use DH-CHAP to authenticate devices that join a fabric.

The first and simplest line of defense is to persistently disable all unused ports, preventing someone without management privileges to the fabric to connect a switch or other devices to the fabric. It is important to use the persistent disable option to ensure that disabled ports remain disabled after a reboot or power-off cycle. Otherwise, an attacker could simply cause a power failure on a switch to enable the unused ports.

The second line of defense is to prevent ports from becoming E_Ports. In the event that an unused port is enabled, a switch would still be unable to join the fabric since the port will not be allowed to become an E_Port.

The third line of defense is to configure monitoring, alerting, and logging to ensure visibility into any unexpected switch events and attacks.

The fourth line of defense is to use a fabric-wide consistency policy to ensure that all switches in all fabrics consistently avoid a “weak-link” that could be exploited in an attack.

The subsequent layers are to use an access control list defense, described in more detail in the following section.

Clearly, it is not necessary to implement every one of these lines of defense to prevent FC device access to a fabric. The number of layers of security that an organization decides to implement will depend on their business requirements, the sensitivity of the environment, and the amount of risk accepted as tolerable. In reality, very few organizations implement all of these proposed levels of security. It is up to each organization to establish the risk and decide which features should be implemented.

15.3 Access Control Lists

Access control lists (ACLs) are used to provide network security via policy sets. Brocade FOS provides several ACL policies, including a Switch Connection Control (SCC) policy, a Device Connection Control (DCC) policy, a Fabric Configuration Server (FCS) policy, an IP Filter policy, and others. The following subsections briefly describe each policy and provide basic guidelines. A more in-depth discussion of ACLs can be found in the *Brocade Fabric OS Administration Guide*.

15.3.1 SCC Policy

The SCC policy restricts the fabric elements (FC switches) that can join the fabric. Only switches that are specified in the policy are allowed to join the fabric. All other switches will fail authentication if they attempt to connect to the fabric, resulting in the respective E_Ports being segmented due to the security violation.

Use the SCC policy in environments where there is a need for strict control of fabric members. Since the SCC policy can prevent switches from participating in a fabric, it is important to regularly review and properly maintain the SCC ACL.

15.3.2 FCS Policy

Use the FCS policy to restrict the source of fabric-wide settings to one FC switch. The policy contains the WWN of one or more switches, and the first WWN (that is online) in the list is the primary FCS. If the FCS policy is active, only the primary FCS is allowed to make and/or propagate fabric-wide parameters. These parameters include zoning, security (ACL) policy databases, and other settings.

Use the FCS policy in environments where there is a need for strict control of fabric settings. As with other ACL policies, it is important to regularly review and properly maintain the FCS policy.

15.3.3 DCC Policy

The DCC policy restricts the devices that can attach to a single FC port. The policy specifies the FC port and one or more WWNs that are allowed to connect to the port. The DCC policy set comprises all of the DCC policies that are defined for individual FC ports. (Note that not every FC port must have a DCC policy, and only ports with a DCC policy in the active policy set enforce access controls.) A port that is present in the active DCC policy set will allow only WWNs in its respective DCC policy to connect and join the fabric. All other devices will fail authentication when attempting to connect to the fabric, resulting in the respective F_Ports being disabled due to the security violation.

Use the DCC policy in environments where there is a need for strict control of fabric members. Since the DCC policy can prevent devices from participating in a fabric, it is important to regularly review and properly maintain the DCC policy set.

15.3.4 Policy Database Distribution

Security policy database distribution provides a mechanism for controlling the distribution of each policy on a per-switch basis. Switches can use individually configured policies to either accept or reject a policy distribution from another switch in the fabric. In addition, a fabric-wide distribution policy can be defined for the SCC and DCC policies with support for strict, tolerant, and absent modes. These modes can be used to enforce whether the SCC and/or DCC policy must be consistent throughout the fabric.

- Strict mode: All updated and new policies of the type specified (SCC, DCC, or both) must be distributed to all switches in the fabric, and all switches must accept the policy distribution.
- Tolerant mode: All updated and new policies of the type specified (SCC, DCC, or both) are distributed to all switches (Brocade FOS v6.2.0 or later) in the fabric, but the policy does not need to be accepted.
- Absent mode: Updated and new policies of the type specified (SCC, DCC, or both) are not automatically distributed to the other switches in the fabric; policies can still be manually distributed.

Together, the policy distribution and fabric-wide consistency settings provide a range of control on the security policies from little or no control to very strict control.

Clearly, it is not necessary to implement every one of these lines of defense to prevent FC device access to a fabric. The number of layers of security that an organization decides to implement will depend on their business requirements, the sensitivity of the environment, and the amount of risk accepted as tolerable. In reality, very few organizations implement all of these proposed levels of security. It is up to each organization to establish the risk and decide which features should be implemented.

For a detailed discussion of SAN security concepts and issues, refer to the *Brocade Fibre Channel Security Best Practices*.

15.3.5 Authentication Protocols

Brocade FOS supports both Fibre Channel Authentication Protocols (FCAPs) and Diffie-Hellman Challenge Handshake Authentication Protocols (DH-CHAPs) on E_Ports and F_Ports. Authentication protocols provide additional security during link initialization by assuring that only the desired device/device type connects to a given port.

15.4 Secure SAN Management

User account and privilege management is cardinal to secure SAN management, with strong policies for accounts and passwords in combination with separation of duties and assigned privileges on a need-to basis only.

The following list outlines best practices for secure SAN management:

- Allow only secure protocols to connect the switches (SSH, HTTPS, and SNMPv3).
- Use unique user accounts with proper roles and privileges (RBAC).
- Change default passwords on *all* default accounts, and ideally do not use default accounts.
- Create and enforce password policies (strength, history, expiration, and lockout).
- Use a centralized account and password management method such as RADIUS, TACACS+, or LDAP.

15.4.1 Role-Based Access Controls

One way to provide limited accessibility to the fabric is through user roles. Brocade FOS has predefined user roles, each of which has access to a subset of CLI commands. These are known as Role-Based Access Controls (RBACs), and they are associated with the user login credentials. Using RBAC, separation of duties is easily implemented to restrict the privileges granted to a user in alignment with assigned function and authority. A role could be read-only, allowing a user to only view information but not modify or delete it. At the other end of the spectrum is a role that grants full admin privileges; with other roles somewhere in between. Typically, these roles are customized for specific types of functions, such as an operator or a security administrator.

15.5 Securing Management Interfaces

Management interfaces are one of the most vulnerable points in any IT infrastructure. Therefore, protecting the management interfaces should always be of high priority and is fairly straightforward to do. The following list outlines the measures to protect the management interfaces:

- Use a separate VLAN (or private VLANs) for the management network.
- Use secure protocols to access management interfaces (SSH, HTTPS, and SNMPv3).
- Disable the equivalent unsecure protocols (Telnet, HTTP, and SNMPv1).
- Limit the points of entry for management with an IP Filter policy and use an FCS policy if necessary.

One of the simplest techniques for protecting management interfaces is to use a separate subnet, or VLAN, to isolate the management network from the production network. This limits access to the management network to SAN administrators only, not to the company at large. Since insiders can be a significant threat, it is always good practice to use only secure protocols to encrypt the communications between management workstations and the devices being managed. This can be done using protocols such as SSH, HTTPS, and SNMPv3 and disabling the equivalent unsecure Telnet, HTTP, and SNMPv1 protocols.

15.5.1 IP Filter

The IP Filter policy is a set of rules applied to the IP management interfaces as a packet-filtering firewall. The firewall permits or denies the traffic going through the IP management interfaces according to the policy rules.

Fabric OS supports multiple IP Filter policies to be defined at the same time. Each IP Filter policy is identified by a name and has an associated type. Two IP Filter policy types, IPv4 and IPv6, exist to provide separate packet filtering for IPv4 and IPv6. You cannot specify an IPv6 address in the IPv4 filter, nor can you specify an IPv4 address in the IPv6 filter. There can be up to six different IP Filter policies defined for both types. Only one IP Filter policy for each IP type can be activated on the affected management IP interfaces.

The IP Filter policy is used to restrict access through the Ethernet management ports of a switch. Only the IP addresses listed in the IP Filter policy are permitted to connect to the specified TCP/UDP port via the switch management ports.

The IP Filter policy should be used only to allow secure protocols. In addition, for additional security in environments where there is a need for strict control of fabric management access, the source addresses or subnet from which SAN administration is done should be specified as the only IP addresses with access permitted. As with other ACL policies, it is important to regularly review and properly maintain the IP Filter policy.

Chapter 16: Brocade Management Platform and Data Gathering

16.1 SANnav™ Management Suite

16.1.1 SANnav Global View

To address the management needs of very large-scale SAN environments or those that are distributed globally, SANnav supports a hierarchical management model, where a higher-level “global” application view provides comprehensive visibility, summarization, and seamless navigation across multiple instances of SANnav Management Portal. SANnav Global View allows a user to drill down into any individual instance and perform detailed monitoring, investigation, and troubleshooting based on events being rolled up into the global view.

16.1.2 SANnav Management Portal

SANnav allows you to efficiently manage your SAN infrastructure through various easy-to-use functions. SANnav implements a highly scalable client-server architecture for SAN management. With a modern browser-based UI, SANnav eliminates the need for a Java-based thick client. The SANnav user interface is designed based on real-world use cases and user workflows, providing a highly intuitive user experience. SANnav uses a micro-services architecture based on Docker container technology that allows it to scale to meet the management needs of both small and large SAN environments and those that may change over time. This scalable architecture also allows SANnav to support new functionality in the future without causing degradation to the performance of the application.

SANnav Management Portal allows management of one or more SAN fabrics that are in the same or different geographical locations, and it supports a maximum of 15,000 physical SAN ports. For environments that are larger than15,000 ports, you can deploy multiple SANnav Management Portal instances.

SANnav Management Portal does not replace Brocade Web Tools or the Fabric OS command line interface.

16.1.3 SANnav Deployment: Requirements and Scalability

SANnav is available as an installable on Linux or on a virtual appliance (OVA for VMware vSphere deployments). Specified in the following table are the server and virtual-machine requirements for deploying SANnav Management Portal.

Table 4: Requirements for SANnav Installation on a Bare Metal Server or VM

Product/Edition	Max. Switch Ports/Instances under Management	Operating System	Host Type	CPU	Memory	Hard Disk
Brocade SANnav Management Portal Base Edition (Manages switches only, no directors)	600	RHEL 7.7, 8.0, & 8.1 CentOS 7.7, 8.0, & 8.1	Bare Metal, ESXi VM	16 Cores	48 GB	600 GB
Brocade SANnav Management Portal Enterprise Edition (Required to manage directors)	Up to 3,000	RHEL 7.7, 8.0, & 8.1 CentOS 7.7, 8.0, & 8.1	Bare Metal, ESXi VM	16 Cores	48 GB	600 GB
	Between 3,000 and 15,000	RHEL 7.7, 8.0, & 8.1 CentOS 7.7, 8.0, & 8.1	Bare Metal, ESXi VM	24 Cores	96 GB	1.2 TB

Table 4: Requirements for SANnav Installation on a Bare Metal Server or VM

Product/Edition	Max. Switch Ports/Instances under Management	Operating System	Host Type	CPU	Memory	Hard Disk
Brocade SANnav Global View	Up to 20 SANnav Management Portal instances	RHEL 7.7, 8.0, & 8.1 CentOS 7.7, 8.0, & 8.1	Bare Metal, ESXi VM	16 Cores	32 GB	450 GB

The following table shows the virtual machine specifications required for the OVA deployment of SANnav Management Portal.

Table 5: Requirements for SANnav Virtual Appliance (OVA) Installation

Requirement	Base or Enterprise License with up to 3,000 Ports, Included with the SANnav OVA Package	Enterprise License with up to 15,000 Ports
Operating System	CentOS 8.0	CentOS 8.0
Server Package	VMware ESXi / vCenter 6.7 or higher	VMware ESXi / vCenter 6.7 or higher
CPU	16 Cores	24 Cores
CPU Sockets	2	2
Memory (RAM)	48 GB	96 GB

16.2 Getting Started with Brocade SANnav Management Portal

16.2.1 Fabric Discovery

Discover your fabrics easily through SANnav Management Portal to begin managing and monitoring all entities that belong to such fabrics. When devices running Fabric OS 9.0 and later are discovered, streaming of telemetry data to the SANnav server is automatically initiated for your monitoring and troubleshooting needs with no additional configuration required.

16.2.2 User Management

Access to SANnav Management Portal is controlled by authentication and authorization of users. Authentication is the process of validating user names and passwords. Authorization is the process of validating the roles and areas of responsibility (AORs) for each user.

Roles and AORs

In SANnav, SAN admins should set appropriate roles and fabric permissions for the intended users. This will help minimize any potential configuration issues in SAN environments and will permit changes to be made only by those users with sufficient SAN admin rights. SANnav offers the required flexibility to fully customize user roles and areas of responsibility to fit a wide range of customer needs.

Third-Party Authentication and Authorization Integration (LDAP, RADIUS, TACACS+)

Customers can also configure SANnav to use an external server to authenticate user names and passwords. Optionally, external servers can also be used for user authorization involving roles and AORs. The SANnav Management Suite supports external server authentication and authorization via LDAP, RADIUS, and TACACS+.

16.2.3 Device Enclosure Validation

SANnav introduced support for the automatic creation of host and storage enclosures of devices, given that such devices support FDMI. Moreover, if there are devices present in the SAN that do not support FDMI, SANnav provides users with the ability to manually create or edit the desired host and storage enclosures.

Device enclosures provide users with an effective way to keep track of device ports and what physical devices they belong to. Enclosures also aid SANnav in properly assessing the health of such devices and in forming accurate topology views.

16.2.4 Monitoring and Alerting Policy Suite

With SANnav, SAN admins can enable, configure, and distribute monitoring policies across switches and fabrics. This makes it easy for users to obtain consistent monitoring across their SAN environment or monitor specific switches of interest more or less aggressively than others.

If you are already familiar with MAPS and have existing policies that have been tailored over time, SANnav also supports the import, customization, and distribution of such policies.

16.2.5 Dashboard Monitoring

SANnav Management Portal offers a set of predefined monitoring dashboards aimed to satisfy the monitoring needs of most customers:

- **Health Summary Dashboard**
  Monitors the health of all SAN entities (hosts, storage, switches, and fabrics) through established Brocade best practices, MAPS violations, and hardware statuses. SANnav then provides insights and recommended actions to further assess the issue at hand.
- **Network Port Traffic Conditions Dashboard**
  Monitors all ports in the fabric for congestion while visually tracking its severity overtime. This dashboard also allows users to take a deeper look into the causes of such congestion through port and flow-level investigation.
- **Extension Dashboard**
  Focuses on monitoring extension-related statistics, providing insights into tunnel and circuit performance.

In addition, users can create custom monitoring dashboards from the 80+ status and performance widgets available; for example, a MAPS Database Dashboard that monitors for all violations occurring in your SAN while providing easy access to troubleshooting actions.

16.2.6 Reporting

SANnav reporting, like dashboards, is fully customizable by the user, offering an extended set of widgets to choose from. Reports can be scheduled, run on demand, viewed in SANnav, and downloaded in various formats (HTML, PDF, CSV).

Reports can be used to keep track of SAN device inventory, zoning configurations, top talkers, congested devices, events of interest, and more. SANnav also allows custom filters to be applied to reporting templates, providing users with the flexibility of reporting on only what is of interest.

16.2.7 FOS Version Management

Make use of the SANnav built-in Fabric OS Repository, where users can upload and manage FOS versions along with release notes. This enables users to perform single or multiple FOS upgrades across multiple switches simultaneously, leading to substantial time savings.

16.2.8 Event Management

Using Event Management features within SANnav allows trap configurations, SNMP traps registrations, syslog events, and other information from switches. Users can also view, search, and filter event logs. SNMP traps and syslog messages can also be configured and forwarded to external destinations.

The following are some of the actions that you can configure for events when triggered:

- Generate email alerts.
- Trigger SupportSave.
- Enable maintenance mode to suppress events in the event log.

You can also perform powerful event analysis by filtering events using custom network scopes, date ranges, and stackable filters. Users can perform searches within the event log and generate event reports on demand.

In addition, Event Management provides several event-managing widgets that you can add to dashboards, such as the Top N Events, Events Summary, and Health Violations widgets.

16.2.9 Northbound Telemetry Streaming Support

Northbound streaming provides support to securely stream performance and flow metrics from the switch to an external Kafka cluster (the northbound server). Streaming gives you access to a large set of data from one or more managed fabrics that can be used to build customized reports or applications.

Raw SNMP metrics from Performance Monitor data and raw flow metrics received from switch streaming will be streamed to the northbound server.

Configuration of the streaming interface is controlled using the REST API.

For more information around northbound streaming and its configuration, refer to the *Brocade SANnav Management Portal REST API and Northbound Streaming Reference Manual*.

16.2.10 SANnav Backup and Restore

SANnav allows you to back up the SANnav server data and restore it as required, such as in scenarios where the data is deleted or corrupted. Also, you can use the backup when you want to bring up a new SANnav server. Creating a backup helps to protect the server’s data and configuration in the event of a disaster, such as a server failure. You have options to exclude large data sets, which can save restoration time in some situations. You can perform a scheduled daily or weekly backup or an on-demand backup.

16.2.11 Backup Recommendations

The recommendations for backup follow:

- Perform a full backup on a weekly or monthly basis as daily backups may slow down the server.
- Ensure that your backup location has enough disk space before you back up your data.

- Ensure that your backup location is different from the location where SANnav is installed.
- For scheduled backups, occasionally check if the backup data size has abnormal patterns (files being too large or too small).
- Periodically verify that there is enough disk space for successful scheduled backups (SANnav does not purge older backups).

16.3 Brocade SAN Health

Brocade SAN Health is a free tool that allows SAN administrators to securely capture, analyze, and report comprehensive information about Brocade fabrics with switches that are running Brocade Fabric OS and Cisco MDS fabrics that are running SANOS/NXOS. SAN Health can perform tasks such as the following:

- Taking inventory of devices, switches, firmware versions, and SAN fabrics.
- Capturing and displaying historical performance data.
- Comparing zoning and switch configurations against best practices.
- Assessing performance statistics and error conditions.
- Alerting the administrator of relevant technical support bulletins (TSBs).
- Producing detailed reports (in Microsoft Excel) and diagrams (in Microsoft Visio).

Download Brocade SAN Health and find details and instructions on how to use it at www.broadcom.com/sanhealth.

Chapter 17: Brocade Support Link

Brocade Support Link (BSL) is available on all switches running Fabric OS 8.2.1c or later. In this chapter, we go over the main features and considerations for enabling and deploying Brocade Support Link.

Brocade Support Link depends on enabling Active Support Connectivity (ASC) on all switches in the SAN. When ASC is enabled and configured, every switch in the fabric periodically collects configuration settings, log messages, and port metrics and sends the data to the Support Link Server (SLS) at Broadcom. This is comparable to the data sets collected when using SAN Health or a subset of Support Save. The collected data is exclusively event logs, configuration data, diagnostics data, and metadata and does not include any data that is being transported in the SAN.

The transfer of the collected data set can be performed directly from the switches to the SLS or by using the Brocade Active Support Connectivity Gateway (ASC-G), which is best practice.

Deploying with ASC-G also provides additional functions and offerings such as Data Collection Assistant and Automated Case Creation, which are not available when switches send data directly to the SLS.

17.1 BSL Features

Brocade Support Link provides a tiered model of offerings based on the support contract with Brocade or the OEM. For more detail, contact you sales representative.

- Configuration, Performance, and Inventory (CPI) reports

  The CPI report provides a comprehensive package of all configuration and inventory data together with performance data for the past 24 hours from the date of CPI report generation. This report is valuable for any inventory-related tasks such as documentation, planning, and SAN operations.
- Best Practice Assessment (BPA) reports

  The BPA report provides a best practice assessment of the SAN across hundreds of indicators in the areas of security, monitoring, health, utilization, performance, scalability, and comparison of configuration consistency across fabrics. The assessment includes score cards for easy tracking and a findings list with a recommended priority and explanation of any correction. For all recommendations, the exact syntax for executing the recommended action is provided, enabling SAN operators to follow best practices based on Brocade experience from more than two decades. In addition, BPA will identify whether any known defects apply to the SAN and will call out a proposed action.
- Fabric Analytics (FA)

  Fabric Analytics continually processes all telemetry data from the SAN in an analytics engine to identify and root-cause current or potential performance issues related to congestion and/or physical-layer issues in the SAN. The Proactive Health Summary report provides a comprehensive view of all current SAN performance-impacting issues in the past 24 hours, it compares this to a baseline of the past week, and it provides trending to proactively identify potential issues before they impact performance. For each identified issue, an in-depth analysis is provided that identifies culprits and adversely impacted “victims” in the SAN.
- FCIP Reports

  FCIP reports are provided as an add-on to Fabric Analytics and deliver comprehensive analysis and identification of deviation in performance on SAN Extension links with FCIP.

- Automated Case Creation (ACC)

  Automated Case Creation integrates SAN monitoring with support. When critical events are identified in the SAN, a support case is automatically created and the Technical Assistance Center (TAC) can immediately start troubleshooting and resolving the case without any human interaction in the process from the occurrence of the critical event to the creation of the case. ACC requires deployment of ASC-Gateway.
- Data Collection Assistant (DCA)

  Data Collection Assistant is enabled with ASC-G as a centralized point for Support Save triggering or scheduling by the end user across all switches in the SAN. In addition, DCA can be combined with ACC; when a critical SAN event occurs, ACC creates a support case with TAC and secondly triggers Support Save from the switches necessary to root-cause the critical event.

17.2 Deployment Options

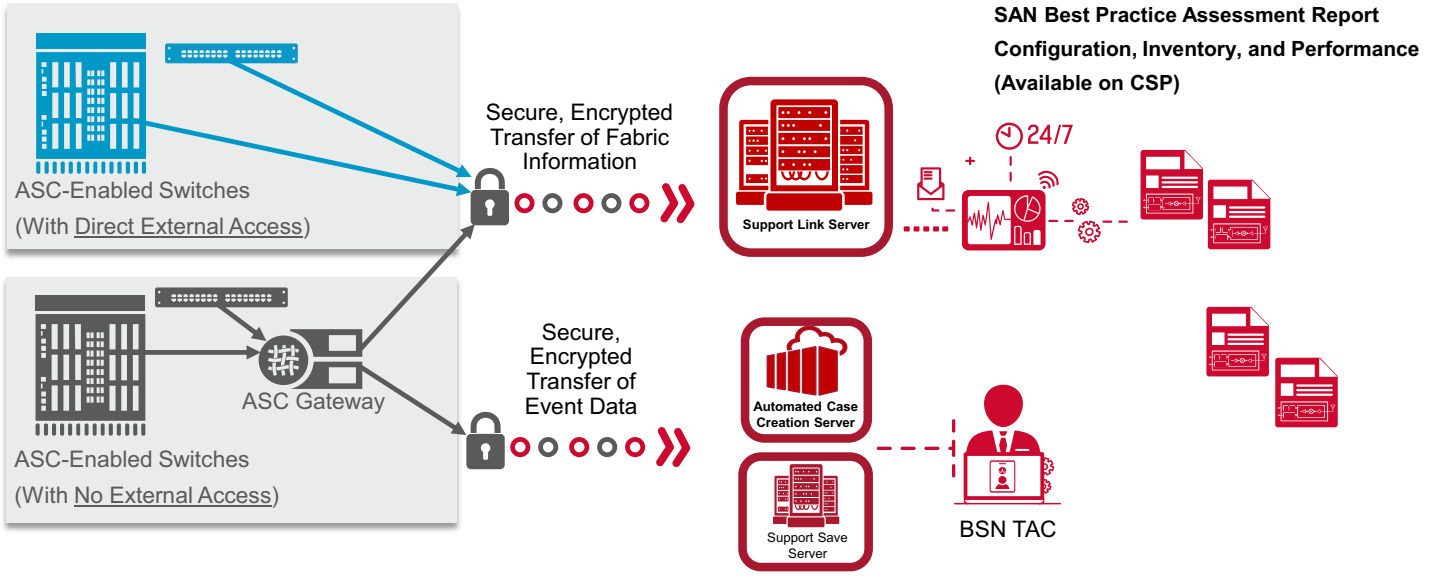
When deploying BSL, the first decision is whether to enable the switches to use the ASC-G to provide transport to the Support Link Server or to enable the switches to send directly to the SLS, as illustrated in Figure 37.

The general recommendation is to deploy with ASC-G unless it is not possible to provide the infrastructure services necessary for the ASC-G(s) or if the SAN environment is very small (one to two switches).

With ASC-G deployment, only the ASC-Gs (best practice is to deploy two ASC-Gs for redundancy) need access through the firewall. In addition, features such as Data Collection Assistant and Automated Case Creation are available only in deployments with the ASC-G.

Illustrated in Figure 37 are the two different ASC connectivity methods for transport of BSL data to Broadcom.

Figure 37: Active Support Connectivity Architecture



17.3 BSL Deployment with ASC-G

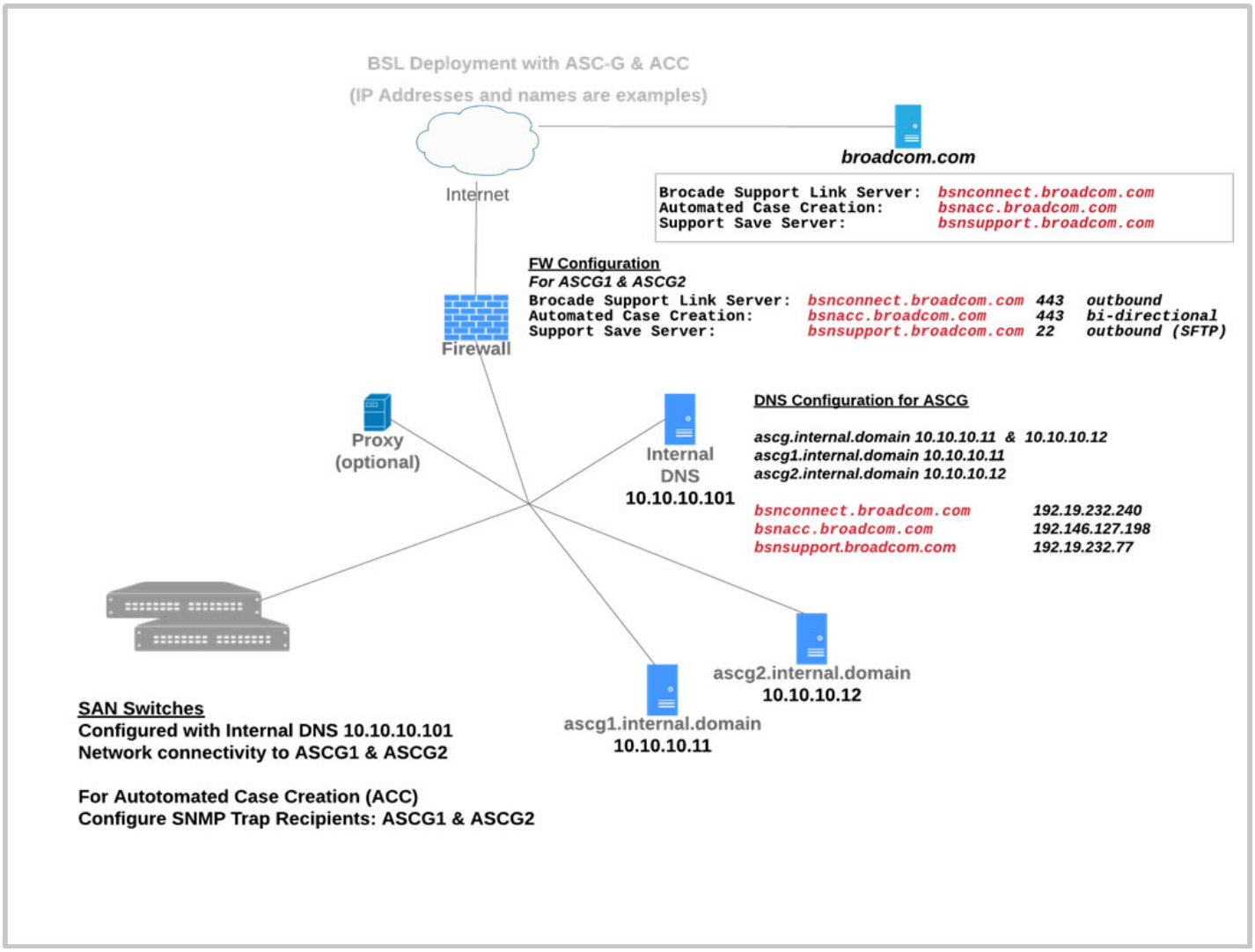
Deployment of BSL with the ASC Gateway is illustrated in Figure 38 with each of the infrastructure requirements described in more detail. In general, deploy two or more ASC-Gs per site for redundancy purposes. ASC-Gs are sized to handle 200 switches per instance.

In Figure 38, ascg1.internal.domain and ascg2.internal.domain are the two instances of the ASC Gateway to be deployed. The common name for the two instances, ascg.internal.com, is used as a common DNS record for configuration on the SAN switches. All IP addresses on the internal network are examples.

The internal DNS server is the DNS server that the switches are configured to use to resolve the ASC Gateway address. When the same DNS server is used by the ASC Gateway instances, the DNS server must also resolve for the Brocade Support Link Server (SLS) bsnconnect.broadcom.com and, when ACC and DCA are implemented, bsnacc.broadcom.com and bsnsupport.broadcom.com.

The firewall must be configured to allow HTTPS outbound from the ASC-G instances to the SLS and ACC servers. For upload of Support Saves (with DCA), the firewall must allow SFTP outbound from the ASC-G instances to the Support Save server. If a proxy is used for Internet access, the proxy configuration must be performed on the ASC-G instances.

Figure 38: BSL Deployment with the ASC Gateway



17.4 DNS Registration of ASC-G Instances

The ASC Gateways must be registered on the internal DNS server for the switches that use the ASC-G to perform name resolution. The common name in this example, ascg.internal.domain, is used as a common DNS record for the ASC Gateways in addition to individual records for each instance in order to use DNS round robin as a load-balancing mechanism. Use of a load balancer is also supported, in which case the common DNS record must point to the load balancer VIP for the ASC-Gs.

Example DNS records for ASC Gateways:

ascg.internal.domain	10.10.10.11 & 10.10.10.12
ascg1.internal.domain	10.10.10.11
ascg1.internal.domain	10.10.10.12

In addition, the ASC-G instances must be able to resolve for bsnconnect.broadcom.com; for Automated Case Creation, bsnacc.broadcom.com; and for Data Collection Assistant, bsnsupport.broadcom.com.

17.5 Certificate Authority to Sign SSL Certificates

In order for the switches to trust the ASC-Gs for HTTPS transfer, a signed SSL certificate is installed on the ASC-Gs during configuration. The certificate can be signed by a public CA or your own enterprise CA; self-signed certificates from Fabric OS 8.2.2 are also supported. The same SSL certificate (and private key) using the common name for the ASC-G instances in this example, ascg.internal.domain, is used for a pair of ASC-G instances.

NOTE: When using an enterprise CA or self-signed certificates, the CA certificate must be installed on the switches in order for the switches to validate the signed certificate on the ASC Gateways.

When configuring for ACC, a certificate must also be created on the switches to be trusted by the ASC-G instances. This procedure is performed during the onboarding process for ACC. For more information, see the *Brocade Active Support Connectivity Gateway User Guide*.

17.6 Hypervisor/VM/Server on Which to Install ASC Gateways

The ASC Gateway is available both as an OVA and an installable (package). Depending on the organizational policies for infrastructure components in your environment, either deploy the OVA version in a VMware vSphere environment or install the installable version on a server/VM provided by your infrastructure team. RHEL, Oracle Linux, and CentOS are supported. For either option, refer to the *Brocade Active Support Connectivity Gateway User Guide* for specific information.

17.7 Enabling ASC on the Switches

ASC is enabled and configured on the switches using the `supportlink` command.

Regardless of whether the switches send directly to the Brocade Support Link Server or use the ASC Gateway, it is mandatory to specify a user name.

The user name is an email address. Subsequently, users from the same email domain with access to the Broadcom Customer Support Portal (portal.broadcom.com) can be authorized to request and receive BPA reports. Use a common user name, for example operators@domain.com, although it can simply be an admin's email address, for example admin@domain.com, for all switches from the same organization.

In addition, use the User Group Tag to group switches that you want to appear in the same report. This can be based on location or other identifier for one or more SAN fabrics.

NOTE: You can always request a BPA report, which includes *all* switches configured with the same user name.

Part of enabling ASC on the switches includes configuring the frequency and time of day for ASC data collection. In general, the recommended settings are to send data daily between 1 a.m. and 4 a.m. or another time considered off hours.

For additional configuration details, refer to the *Brocade Active Support Connectivity Gateway User Guide*.

With the ASC configuration complete, the switch will upload ASC data daily to the Brocade Support Link Server. In addition, you can manually trigger ad hoc data collection and upload on demand.

Chapter 18: Automation

18.1 Overview and Purpose

Most, if not all, IT administrators have first-hand experience in managing the growing complexity of enterprise infrastructure, including SANs. According to a report from the Enterprise Strategy Group, “The cost and complexity of protecting and storing data is increasing, and IT leaders are responding with attempts to better optimize and automate storage—but they need better tools.”

Brocade is in the unique position to spot and understand the impact of automation, helping organizations get more from their SAN infrastructure.

Brocade offers a combination of SAN automation with RESTful APIs and a SAN management platform to help organizations drive greater efficiency from their SANs. This is accomplished through a variety of means:

- Brocade SAN automation, provided with a multilayer architecture.
- RESTful API support on switches and management tools.
- Brocade’s Ansible management framework, designed to eliminate repetitive tasks, simplify management, and orchestrate across the full SAN infrastructure.

18.2 Motivation to Automate

The following are five reasons why organizations should embrace SAN automation:

- Reducing human error and streamlining operational processes have never been more crucial. As organizations move to digitize and adapt to new workloads, data availability, processing time, and agility in provisioning applications on-demand become the life-blood of the business. This demands a more efficient and expedient infrastructure management approach, leaving no room for human error. As a result, storage administrators need to be freed from repetitive manual tasks such as configuration management, reporting, documenting inventory, and troubleshooting. Instead, IT organizations need SAN automation to help them automate and orchestrate repetitive tasks, significantly improve efficiency, and decrease the risk of operational mistakes.
- Demand for more accurate and more frequent infrastructure reports is on the rise. It is not just IT managers who crave information on storage performance, utilization, and forecasting; business stakeholders are also asking for and expecting this data on demand. No one wants to wait for a slot when storage administrators can allocate time to produce a report. This information should be available as frequently as business demands dictate—all at the click of a button. Automation not only provides this kind of responsiveness that traditional manual storage management processes cannot deliver, but it can also be customized so that all stakeholders get more accurate data aligned to their responsibility.
- SAN configuration management must be streamlined. With more enterprise applications demanding access to more data and with more virtual machines, deploying and configuring servers, storage, and the network has become more time-consuming and more complex than ever. By streamlining SAN operations through automation, application provisioning workflows are simplified across hypervisor, network, and storage, delivering agility and responsiveness to meet dynamic business demands.
- IT service delivery is not always as responsive as the business demands. As organizations become increasingly reliant on having world-class IT services available to them for proactive, agile business decision-making, it is essential that bottlenecks to IT service delivery be identified and eliminated immediately. This has to be done without hiring more storage administrators or boosting SAN-related CapEx spending, making SAN automation the only viable option to drive increased IT agility and more tightly align IT service delivery with fast-changing business needs.

- Consistent configuration validation is a must. Manual configuration changes are occurring with greater frequency as enterprises diversify their IT architectures in general and their storage architectures specifically. SAN automation ensures validation of consistent configuration parameters across the different SAN fabrics in order to facilitate troubleshooting of frequent alerts without reliance on manual intervention.

Brocade automation solutions leverage RESTful APIs to facilitate solutions architecture, share best practices, and get to production status faster.

The combination of SAN automation and management solutions is a major reason why Brocade was awarded a bronze medal by TechTarget's Storage magazine and SearchStorage's 2018 Products of the Year storage management tools category. One judge for the category wrote: "Brocade's SAN automation brings automation to the switch level that helps reduce the cost and complexity of managing storage systems."

18.3 Overview of the REST API

The FOS REST API is a programmable web service interface for Brocade Fabric OS that can manage Brocade SAN switches across a fabric. This API uses standard HTTP methods to perform Create, Read, Update, and Delete (CRUD) operations on the fabric configuration data, and it provides an interface for provisioning, status, and validation operations using the YANG data model described in the YANG 1.1 RFC, but not the data store managed with NETCONF. An Apache web server embedded in Fabric OS is used to serve the API.

The RESTful API approach lets you think of a network device as a web server. By using standard web-based tools, automation can send and receive transactions to or from a network device just as it would send transactions to and from a website. This means that transactions happen over a secure socket using HTTP rules to handle the exchange. The data appears in the form of XML or JSON depending on the RESTful API services implemented on the networking device.

To interact with a SAN (or other) device, you need to consult its RESTful API reference to learn, among other things, what "uniform resource identifiers" (URIs) you need to use. (Simply put, URIs are identifiers that can be used as part of a web address.) According to the documentation, the URI for accessing a listing of the zones in the active configuration is as follows:

```
GET <base_URI>/running/zoning/defined-configuration/
```

The model that is used to represent state and configuration information is expressed in a modeling language called Yang. Yang describes the structure of the different elements inside the model and is used to describe whether each element is read-only or read-write. It describes the type of data that the element can hold, such as string or integer, and it shows the relationship among various elements, the other nested elements they contain, their peer elements, and the parent elements that contain them. Here is a segment of the description of a zone in Yang:

```
list zone {
   key "zone-name";
   description
      "List of the members in the zone. The members can only be identified as a WWN,
      domain, index, or a zone alias.";
   leaf zone-name {
      type zoning-name-type;
      description
         "The zone name.";
      }
```

```
      leaf zone-type {
         type zone-type-type;
         description
            "The zone type. Not that target zone types cannot be created or modified (only
            deleted).";
         }
      container member-entry {
      description
         "The zone member.";
      leaf-list entry-name {
         type zone-member-type;
         min-elements 1;
         description
            "List of the members in the zone. The members can only be identified as a WWN,
            domain, index, or zone alias.";
         }
      leaf-list principal-entry-name {
         when "../..zone-type=1 or ../../zone-type=2";
         type zone-member-type;
         min-elements 1;
         description
            "List of the principal members in the peer zone. The members can only be
            identified as a WWN, domain,index, or zone alias.";
         }
      }
   }
```

Ordinarily more information goes into a Yang module, such as revisioning and governance information; this listing omits them for brevity. Thus, the Yang description is complete, but it is also wordy. Although this precision is necessary when interacting with the model programmatically, it is sometimes useful to get a global view of the abstraction provided by the model to see how the data is structured.

An open source tool called pyang can parse the Yang model and produce a tree that represents the elements in the model. The listing includes information about each element, such as whether it is read-only or read write, a list, optional, or nested. Here is the representation of the zoning model in tree form:

```
   module: brocade-zone
     +--rw brocade-zone
        +--rw defined-configuration
        |  +--rw cfg* [cfg-name]
        |  |  +--rw cfg-name       zoning-name-type
        |  |  +--rw member-zone
        |  |     +--rw zone-name*   zoning-name-type
        |  +--rw zone* [zone-name]
        |  |  +--rw zone-name       zoning-name-type
        |  |  +--rw zone-type?      zone-type-type
        |  |  +--rw member-entry
        |  |     +--rw entry-name*             zone-member-type
        |  |     +--rw principal-entry-name*   zone-member-type
```

```
|  +--rw alias* [alias-name]
|     +--rw alias-name      zoning-name-type
|     +--rw member-entry
|        +--rw alias-entry-name*   union
+--rw effective-configuration
   +--rw cfg-name?                   zoning-name-type
   +--rw checksum?                   string
   +--rw cfg-action?                 uint8
   +--rw default-zone-access?        uint8
   +--ro db-max?                     uint32
   +--ro db-avail?                   uint32
   +--ro db-committed?               uint32
   +--ro db-transaction?             uint32
   +--ro transaction-token?          uint32
   +--ro db-chassis-wide-committed?  uint32
   +--ro enabled-zone* [zone-name]
      +--ro zone-name       zoning-name-type
      +--ro zone-type?      zone-type-type
      +--ro member-entry
         +--ro entry-name*             union
         +--ro principal-entry-name*   union
```

18.4 Simple Automation Example

In this example, the <base_URI> is http://<our device IP address>/rest. Begin by creating a login session with a switch in the fabric by executing the following command (which you type as a single line):

```
curl -X POST -v -u admin:password http://10.18.254.37/rest/login
```

- `curl` is the name of the command.
- `-X POST` specifies the POST HTTP method (instead of GET).
- `-v` specifies verbose output to access the authorization string in the header of the response used in the next step.
- `-u admin:password` specifies the credentials to use.

The last parameter is the uniform resource identifier (URI) for `curl` to use to log in. (The URI value is described in the RESTful API reference.)

This command establishes the session used for the following commands. The following is a trace of its execution:

```
*  Trying 10.18.254.37...
*  Connected to 10.18.254.37 (10.18.254.37) port 80
(#0)
*  Server auth using Basic with user 'admin'
>  POST /rest/login HTTP/1.1
>  Host: 10.18.254.37
>  Authorization: Basic YWRtaW46cGFzc3dvcmQ=
>  User-Agent: curl/7.47.0
>  Accept: */* >
<  HTTP/1.1 200 OK
<  Date: Wed, 31 Jan 2018 16:01:24 GMT
<  Server: Apache
```

```
<  Authorization: Custom_Basic YWRtaW46eHh4OjNkYTllZmM3NzMxYjk4OGU2ODg1YzZkMGRjNWJlM
zMyNjBhZDYxZThkOWQ2MWMxNzNiMGVlMjU3YmM2OTcyYjA=
< Cache-Control: no-cache
< X-Frame-Options: DENY
< Content-Secure-Policy: default-src 'self'
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Connection: close
< Transfer-Encoding: chunked
< Content-Type: application/yang-data+xml
```

Next, you perform a GET of the URI to return the current configuration using the Custom Basic value returned from the login for authentication:

```
curl -v -H "Authorization: Custom_Basic
YWRtaW46eHh4OjNkYTllZmM3NzMxYjk4OGU2ODg1YzZkMGRjNWJlM
zMyNjBhZDYxZThkOWQ2MWMxNzNiMGVlMjU3YmM2OTcyYjA=="
http://10.18.254.37/rest/running/zoning/defined-configuration
```

By default, `curl` uses the GET method, so you do not need to specify it. `-H "Authorization: Custom_Basic YWR...jA=="` is the authentication and session identifying string returned in the previous command. `-H` places the string into the GET request header as seen in the following trace:

```
*  Trying 10.18.254.37...
*  Connected to 10.18.254.37 (10.18.254.37) port 80
(#0)
>  GET /rest/running/zoning/defined-configuration
HTTP/1.1
>  Host: 10.18.254.37
>  User-Agent: curl/7.47.0
>  Accept: */*
>  Authorization: Custom_Basic YWRtaW46eHh4OjNkYTllZmM3NzMxYjk4OGU2ODg1YzZkMGRjNWJlM
zMyNjBhZDYxZThkOWQ2MWMxNzNiMGVlMjU3YmM2OTcyYjA=
>
<  HTTP/1.1 200 OK
<  Date: Wed, 31 Jan 2018 16:09:39 GMT
<  Server: Apache
<  Cache-Control: no-cache
<  X-Frame-Options: DENY
<  Content-Secure-Policy: default-src 'self'
<  X-Content-Type-Options: nosniff
<  X-XSS-Protection: 1; mode=block
<  Connection: close
<  Transfer-Encoding: chunked
<  Content-Type: application/yang-data+xml <
<?xml version="1.0"?>
<Response>
<defined-configuration>
<cfg>
<cfg-name>CFG_FABRIC_A</cfg-name> <member-zone> <zone-name>CLUSTER1</zone-name>
```

```
<zone-name>Z_AIXHOST_FCS2_VMAX01_SN1234_9F0 </zone-name>
...
<alias> <alias-name>esx66_5d3d00</alias-name> <member-entry> <alias-entry-
name>10:00:8c:7c:ff:5d:3d:00 </alias-entry-name>
</member-entry>
</alias>
</defined-configuration>
</Response>
* Closing connection 0
```

The results appear as an XML data segment structured according to the description in the Yang model, and so it is important to have access to that model along with the RESTful API manual. The models can be found on GitHub as a repository among those in Brocade's repositories at http://github.com/brocade/yang. The RESTful API manual can be retrieved from the Broadcom website. Having retrieved the zoning information from the fabric, you should close the session using the CLI command (the results are omitted to save space):

```
curl -v -H "Authorization: Custom_Basic
YWRtaW46eHh4OjNkYTllZmM3NzMxYjk4OGU2ODg1YzZkMGRjNWJlM
zMyNjBhZDYxZThkOWQ2MWMxNzNiMGVlMjU3YmM2OTcyYjA=" http://10.18.254.37/rest/logout
```

In FOS version 8.2.2 or later, the REST API session-less operation allows you to provide authentication credentials directly for each GET request. Essentially, FOS REST API session-less operation completes the login, GET operation, and logout as one complete request. You can use only basic authentication formats for REST API session-less operation, which includes plain text or Base64.

The following example shows a GET request using plain-text authentication:

```
curl -u admin:password http://10.155.2.190/rest/running/brocade-media/media-rdp>"
```

18.5 Ansible as an Alternative

The previous section shows an example of an approach that uses a procedural methodology. The workflow starts at the beginning, executes a series of steps, and then terminates. Most traditional programs work this way.

Ansible takes a declarative approach. Rather than provide sequential steps, Ansible describes each of the hosts in an inventory. The description appears in a document called a *playbook*. For example, rather than provide steps to install a particular application, Ansible describes a host state where the application is already installed. When you run the playbook, Ansible takes no action if the application is already installed. If the application is not installed, Ansible calls installation routines so that the host is brought into the desired state without requiring the administrator to write any specific steps.

In the realm of storage networks, the use of a declarative language means that you can describe switches and fabrics where, for example, a zone is already configured with the proper hosts and storage arrays. When you run the Ansible playbook, those zones are defined as needed, and the hosts and storage arrays are added to them if necessary.

With some other declarative automation utilities, it is necessary to install an agent on each host that the utility manages. This agent retrieves the commands from a command center and runs them on the local host. Ansible is unique in that it does not require agents. In order to make switch state changes, Ansible establishes a secure shell session to a proxy and sends it a small Python script. The script performs the necessary operations using the switch API and removes itself from the host.

You need two different skill sets to successfully implement an Ansible solution. First, you must understand the most common playbook operations. These operations are coded and installed for use by the playbooks. As vendors announce support for Ansible, they also provide script libraries for the most common tasks. If there is a task that is required but is not available in the official Ansible distribution, the open source community may provide code for that task in publicly available repositories.

Second, you must understand your business needs to provide ongoing playbook development. The person who maintains the playbooks does not need to be a programmer and does not need to know how remote system operations occur. That person needs to know only the desired outcomes and should be able to construct playbooks in YAML, the markup language used by Ansible. The following is an example of an Ansible playbook:

```
---
- hosts: edgeSwitches
  vars_files:
    -../fos_passwords.yml
  gather_facts: False

  tasks:

  - name: run fos commands
    brocade_fos_command:
      switch_login: “{{switch_admin_account}}”
      switch_password: “{{switch_password}}”
      switch_address: “{{switch_ip_address}}”
      command_set:
        - command: alicreate "SampleAlias1", "10:23:45:67:76:54:32:10"
```

The three dashes at the beginning are part of the YAML specification. The `hosts` section identifies automation target switches. You can keep sensitive information in a separate file, as demonstrated by the `fos_passwords.yml` line. The name of a variable in double braces such as `{{switch_password}}` indicates variable substitution. The variable file specified in `vars_files` tells where to find external variables.

18.6 SANnav’s REST API

SANnav Management Portal and SANnav Global View provide end-to-end visibility into enterprise SANs. These tools detect, analyze, and take action based on SAN behavior and performance, helping administrators get to the root of problems faster and remediate them fully. Storage administrators can troubleshoot across the storage fabric in as little as 30 seconds. This is unprecedented with any other shared storage infrastructure architectures.

The SANnav REST API provides functionality that complements that found in the Fabric OS REST APIs. The SANnav REST API includes support for the following SANnav features:

- List fabrics managed by the SANnav server.
- List the seed and principal switch data for each fabric.
- List switches (members) in the fabric.
- Display FCR topology information for routing topologies such as edge-to-edge, backbone-to-edge, and edge-to-backbone.
- Configure event forwarding.
- Acknowledge or unacknowledge events.
- Display a list of filtered events.
- Search inventory.

18.7 Conclusion

SAN automation is a critical element in IT modernization and digital transformation because it helps organizations handle storage-related processes more efficiently without hiring more administrators or adding to the storage infrastructure CapEx budget. SAN automation is a high-leverage approach to turning network storage into a strategic asset.

Brocade’s commitment to SAN automation, combined with its long-standing leadership in storage fabrics and technical innovation, makes it an ideal candidate for your IT infrastructure automation strategy.

Appendix A: Optical Cables

Table 6: Supported Distances Based on Cable Type and Data Rates

Speed Name	OM1 Link Distance 62.5-µm Core and 200 MHz*km	OM2 Link Distance 50-µm Core and 500 MHz*km	OM3 Link Distance 50-µm Core and 2000 MHz*km	OM4 Link Distance 50-µm Core and 4700 MHz*km	OS1 Link Distance 9-µm Core and ~infinite MHz*km
1GFC	300	500	860	*	10,000
2GFC	150	300	500	*	10,000
4GFC	50	150	380	400	10,000
8GFC	21	50	150	190	10,000
10GFC	33	82	300	*	10,000
16GFC	15	35	100	125	10,000
32GFC	—	20	70	100	10,000

Table 7: LWL Optics Support (SFP+)

Transceiver Data Rate (Gb/s)	Distance (km)
4	4, 10, & 30
8	10, 25
10	10
16	10
32	10

Appendix B: Fabric Details

This appendix provides example checklists and tables that you can use to identify dominant factors, including facilities that will have an impact on the SAN design.

Table 8: Current Fabrics

SAN/Fabric	No. of Switches	Type of Switches	Total Ports	Domains	No. of Servers	No. of Storage Devices	Location	Notes
Fabric 1								
Fabric 2								
Fabric 3								
Fabric 4								
Fabric 5								

Table 9: Individual Fabric Details

SAN/Fabric	Domain Number	Serial Number	Model	Speed	WWN	IP Addresses	Brocade FOS/M-EOS Version	Notes
Switch 1								
Switch 2								
Switch 3								
Switch 4								
Switch 5								

Table 10: Device Details

Servers & Storage	Vendor	Model	WWN	Alias	Zone	OS Version	Application	Fabric/ Switches	Notes
Server 1									
Server 2									
Server 3									
Storage 1									
Storage 2									
Storage 3									

Table 11: Metrics and Impact on SAN Design and Performance

Metric	Source	Impact
Servers in the SAN	Estimate/Brocade SAN Health	Normal operations
Host Level Mirroring	Estimate	Distance, ISL congestion, traffic levels
Clusters (MSFT, HACMP, NetApp) Average number of nodes Workload level	Estimate Estimate: High/Med/Low	In-band heartbeat, frame congestion, host fan-in, traffic isolation
Virtualization: VIO Server No. of servers Consolidation ratio	Estimate Estimate Estimate	Frame congestion, edge traffic increase per port, server fan-in on target ports, device latencies
Virtualization: VMware No. of VMware servers Consolidated ratio? Shared VMFS? DRS? RDM? I/O intensive?	Estimate Estimate Yes/No Yes (%)/No Yes (%)/No High/Med/Low Yes/No	Frame congestion, device latencies, and SCSI2 reservations

Table 12: Consolidated SAN Snapshot

SAN Requirements Data (Complete for Each SAN)	
Fabric Information	
Target number of user ports per fabric	
Target number of total ports per fabric	
Target number of switches per fabric (number of switches/switch type, total switches)	
Number of fabrics	
Number of sites in environment	
Topology (core-edge, ring, mesh, other)	
Maximum hop count	
Expected growth rate (port count)	
Fabric licenses	
SAN Device Information	
Number/types of hosts and OS platforms	
Number/types of storage devices	
Number/types of tapes	
Number/types of HBAs	
Other devices (VTL/deduplication appliance)	
Total number of SAN devices per fabric	
Customer requirement for failover/redundancy, reliability of SAN (multipathing software utilized)	

Table 12: Consolidated SAN Snapshot (Continued)

SAN Requirements Data (Complete for Each SAN)	
Application Details	
SAN Application (Storage Consolidation, Backup and Restore, Business Continuance)	
Fabric management application(s)	
Performance	
Maximum latency (ms)	
Targeted ISL oversubscription ratio (3:1, 7:1, 15:1, other)	

Table 13: Application-Specific Details

Backup/Restore Infrastructure		
Servers		
System	OS Version, Patch Level	HBA Driver Version
Server 1/HBA		
Server 2/HBA		
Server 3/HBA		
Backup Software		
Vendor	Version	Patch
FC Switch		
Vendor	Model	Firmware
Brocade		
Storage		
Vendor	Model	Firmware
Array 1		
Array 2		
Tape Library		
Vendor	Model	Firmware
Library		

NOTE: Keep a similar table for each application.

Appendix C: Terminology

Table 14: Quantitative Analysis: Radar Maps

SAN/Storage Admin Concerns	Rank (1 is Low, 10 is High)	Notes
ISL utilization	8	Is traffic balanced across ISLs during peaks?
Switch outage	1	Have there been switch outages? If so, what was the cause?
Zoning policy	6	Is the zoning policy defined?
Number of switches in the fabric	10	Is the current number of switches a concern for manageability?
Scalability	6	Can the existing design scale to support additional switches, servers, and storage?
Redundancy	10	Is the existing SAN redundant for supporting a phased migration or firmware update?
Server: high availability	10	Does the cluster software fail over reliably?
Storage: high availability	10	Do the LUNs fail over reliably?
Available disk pool	6	Is there sufficient disk pool to support additional apps?
Management tools for SAN	4	Are the right tools used for SAN management?
Application response	7	Have there been any instances of slow application response but no outage?

Term	Brief Description
base switch	Base switch of an enabled virtual fabric mode switch, providing a common communication infrastructure that aggregates traffic from logical switches in the physical switch in a common base fabric.
ClearLink Diagnostics	Diagnostics tool that allows users to automate a battery of tests to verify the integrity of optical cables and 16Gb/s transceivers in the fabric.
D_Port	A fabric port configured in ClearLink Diagnostics testing mode for cable and optics integrity testing.
default switch	Default logical switch of an enabled virtual fabric mode switch, automatically created when Virtual Fabrics is enabled on a VF-capable switch.
E_Port	A standard Fibre Channel mechanism that enables switches to network with each other.
EX_Port	A type of E_Port that connects a Fibre Channel router to an edge fabric.
F_Port	A fabric port to which an N_Port is attached.
FCIP	Fibre Channel over IP, which enables Fibre Channel traffic to flow over an IP link.
FCR	Fibre Channel Routing, which enables multiple fabrics to share devices without having to merge the fabrics.
IFL	Inter-fabric link, a link between fabrics in a routed topology.
ISL	Inter-switch link, used for connecting fixed-port and modular switches.
logical switch	Logical switch of an enabled virtual fabric mode switch, managed in the same way as physical switches and configurable in any mode.
oversubscription	A condition in which more devices might need to access a resource than that resource can fully support.
port group	A set of defined sequential ports (for example, ports 0-3).
QoS	A traffic shaping feature that allows the prioritization of data traffic based on the SID/DID of each frame.
redundancy	Duplication of components, including an entire fabric, to avoid a single point of failure in the network (fabrics A and B are identical).
resiliency	The ability of a fabric to recover from failure; could be in a degraded state but functional (for example, an ISL failure in a trunk group).

TI zone	Traffic isolation zone, which controls the flow of interswitch traffic by creating a dedicated path for traffic flowing from a specific set of source ports.
trunk	Trunking that allows a group of ISLs to merge into a single logical link, enabling traffic to be distributed dynamically at the frame level.
UltraScale ICL	UltraScale inter-chassis link, used for connecting director chassis (Gen 6 and Gen 7 without using front-end device ports.
VC	Virtual channel, which creates multiple logical data paths across a single physical link or connection.
VF	Virtual Fabrics, a suite of related features that enable customers to create a logical switch or logical fabric or to share devices in a Brocade Fibre Channel SAN.

Appendix D: References

D.1 Software and Hardware Product Documentation

FOS documentation is located in two different locations within Broadcom for Brocade products and software. The publicly facing content is located on Broadcom.com. The nonpublic documentation is located on the Broadcom Customer Support Portal (CSP). Please refer to the documentation for your particular Fabric OS release.

Broadcom.com (https://www.broadcom.com/products/fibre-channel-networking)

Contains all user guides, reference manuals, white papers, eBooks, product briefs, administrative guides, compatibility guides, and case studies.

- *Brocade Fabric OS Administration Guide*
- *Brocade Fabric OS Command Reference Manual*
- *Brocade Fabric OS MAPS User Guide*
- *Brocade Fabric OS Flow Vision User Guide*
- *Brocade Fabric OS Access Gateway User Guide*
- *Brocade Fabric OS Extension User Guide*
- *Brocade X6-4 Director Hardware Installation Guide*
- *Brocade X6-8 Director Hardware Installation Guide*
- *Brocade X7-4 Director Hardware Installation Guide*
- *Brocade X7-8 Director Hardware Installation Guide*
- *SAN Fabric Resiliency and Administration Best Practices User Guide*
- *Brocade SANnav Flow Management User Guide*
- *Brocade SANnav Management Portal Installation and Migration Guide*
- *Brocade SANnav Management Portal REST API and Northbound Streaming Reference Manual*
- *Brocade SANnav Management Portal User Guide*
- *Brocade SANnav Global View User Guide*

Broadcom Customer Support Portal (CSP) (https://portal.broadcom.com/group/support/docsafe/downloads)

Contains supported Fabric OS (FOS) software, supported SANnav software, Target Path selection guides, and release notes.

This content *requires a valid registered account*. It is available only to approved Brocade Direct Support (BDS and BSS) customers and authorized Brocade OEMs with valid entitlement on their products.

- *Brocade Fabric OS Release Notes*
- *Brocade Fabric OS Upgrade Guide*
- *Brocade Fabric OS Troubleshooting and Diagnostics Guide*

D.2 Compatibility, Scalability, and Target Path

Broadcom.com

- *Brocade Fabric OS 8.x Open Systems Compatibility Matrix*

Broadcom Customer Support Portal (CSP)

- *Brocade SAN Scalability Guidelines: Brocade Fabric OS v9.X*
- *Brocade Fabric OS Target Path Selection Guide*

D.3 Brocade SAN Health

- www.broadcom.com/sanhealth

D.4 Brocade Bookshelf

- *NVMe over Fibre Channel for Dummies*
- *Networking Next-Gen Storage for Dummies*
- *Brocade Mainframe Connectivity Solutions*
- *SAN Automation for Dummies*

D.5 Other

- *The SNIA Dictionary*
- *SAN System Design and Deployment Guide*

Revision History

53-1004781-02; September 1, 2020

- Added updates for FOS 9.x and Gen 7.
- Added the “Automation” chapter.

53-1004781-01; November 23, 2016

- Initial release.