13BECS501

Information Storage and Management **L T P C**
3 0 0 3

Course Objectives:

- Evaluate storage architectures and key data center elements in classic, virtualized and cloud environments
- Explain physical and logical components of a storage infrastructure including storage subsystems, RAID and intelligent storage systems
- Describe storage networking technologies such as FC-SAN, IP-SAN, FCoE, NAS and object-based, and unified storage
- Understand and articulate business continuity solutions – backup and replications, along with archive for managing fixed content Explain key characteristics, services, deployment models, and infrastructure components for a cloud computing

Learning Outcomes:

- Describe and apply storage technologies
- Identify leading storage technologies that provide cost-effective IT solutions for medium to large scale businesses and data centers
- Describe important storage technologies‟ features such as availability, replication, scalability and performance
- Work in project teams to install, administer and upgrade popular storage solutions
- Identify and install current storage virtualization technologies
- Manage virtual servers and storage between remote locations
- Design, analyze and manage clusters of resources

UNIT I Storage System **9**

Introduction to information storage, Virtualization and cloud computing, Key data center elements, Compute, application, and storage virtualization, Disk dive & flash drive components and performance, RAID, Intelligent storage system and storage provisioning (including virtual provisioning)

UNIT II Storage Networking Technologies and Virtualization **9**

Fibre Channel SAN components, FC protocol and operations, Block level storage virtualization, iSCL and FCIP as an IP-SAN solutions, Converged networking option – FcoE, Network Attached Storage (NAS) – components, protocol and operations, File level storage virtualization, Object based storage and unified storage platform.

UNIT III Backup, Archive and Replication **9**

Business continuity terminologies, planning and solutions, Clustering and multipathing to avoid single points of failure, Backup and recovery – methods, targets and topologies, data deduplication and backup in virtualized environment, fixed content and data archive, Local replication in classic and virtual environments, Remote replication in classic and virtual environments, Three-site remote replication and continuous data protection.

UNIT IV Cloud Computing **9**

Characteristics and benefits**,** Services and deployment models, Cloud infrastructure components, Cloud migration considerations.

UNIT V Securing and Managing **9**

Storage Infrastructure Security threats, and countermeasures in various domains, Security solutions for FC-SAN, IP-SAN and NAS environments, Security in virtualized and cloud environments, Monitoring and managing various information infrastructure components in classic and virtual environments, Information lifecycle Management (ILM) and storage tiering.

Total Hours:45

Text Books:

1. Information Storage and Management: Storing, Managing and Protecting Digital Information in classic, Virtualized and Cloud Environments, 2nd Edition, EMC Educations Services, Wiley, May 2012.

References:

1. Ulf Troppens, Rainer Erkens, Wolfgang Mueller-Friedt, Rainer Wolafka, Nils Haustein ,"Storage Networks Explained: Basics and Application of Fibre Channel SAN, NAS, iSCSI, InfiniBand and FCoE, 2nd Edition, Wiley, July 2009
2. Information Storage and Management: Storing, Managing, and Protecting Digital Information, EMC Education Services, Wiley, January 2010

KARPAGAM ACADEMY OF HIGHER EDUCATION
FACULTY OF ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

13BECS501 INFORMATION STORAGE MANAGEMENT LECTURE PLAN

S.NO	DESCRIPTION OF PORTION TO BE COVERED	HOURS	Reference Book & Page Nos. Used for teaching	TEACHING AIDS
1	Introduction to Information Storage and Management	1	T[1] Page no1-3	PPT
2	Basic Terminologies in ISM	1	T[1] Page no 4-7	PPT
3	Discussion on Course Plan, Course Objective and Expected Outcomes	1	Syllabus	PPT
UNIT I Storage System				**9**
4	Introduction to information storage, Virtualization and cloud computing	1	T[1]-Page no 8	BB
5	Key data center elements Compute, application	1	T[1]-Page no 9	PPT
6	storage virtualization	1	T[1]-Page no 10	BB
7	Disk dive & flash drive components and performance	1	T[1]-Page no 14	BB
8	RAID	1	T[1]-Page no 14	BB
9	Intelligent storage system	1	T[1]-Page no 14	BB
10	storage provisioning	1	T[1]-Page no 16	BB
11	storage provisioning (including virtual provisioning)	1	T[1]-Page no 17	BB
12	TUTORIAL	1		
TOTAL HOURS FOR UNIT-I		**9**		
UNIT II Storage Networking Technologies and Virtualization				
13	Fibre Channel SAN components	1	T[1]-Page no 21	BB
14	FC protocol and operations	1	T[1]-Page no 38	PPT

15	Block level storage virtualization	1	T[1]-Page no 30	BB
16	iSCL and FCIP as an IP-SAN solutions	1	T[1]-Page no 41	PPT
17	Converged networking option – FcoE	1	T[1] Page no 42-43	PPT
18	Network Attached Storage (NAS)components	1	T[1]-Page no 45	BB
19	protocol and operations	1	T[1] Page no 51	BB
20	File level storage virtualization	1	T[1]-Page no 57	PPT
21	Object based storage and unified storage platform	1	T[1]-Page no 63	PPT
TOTAL HOURS FOR UNIT-II		**9**		
UNIT III Backup, Archive and Replication				**9**
22	Business continuity terminologies, planning and solutions	1	T[1]-Page no 104	BB
23	Clustering and multipathing to avoid single points of failure	1	T[1]-Page no 107	BB
24	Backup and recovery – methods, targets and topologies	1	T[1]-Page no 117	BB
25	data deduplication and backup in virtualized environment	1	T[1]-Page no 117	PPT
26	fixed content and data archive	1	T[1]-Page no 149	BB
27	Local replication in classic and virtual environments,	1	T[1]-Page no 171	BB
28	Remote replication in classic and virtual environments	1	T[1]-Page no 186	PPT
29	Three-site remote replication	1	T[1]-Page no 189	PPT
30	continuous data protection.	1	T[1]-Page no 205	BB
TOTAL HOURS FOR UNIT-III		**9**		
UNIT IV Cloud Computing .				
31	Characteristics and benefits	1	T[1]-Page no 260	BB
32	Characteristics and benefits	1	T[1]-Page no 260	BB
33	Services and deployment models	1		BB
34	Services and deployment models	1	T[1]-Page no 267	BB
35	Cloud infrastructure components,	1	T[1]-Page no 275	BB

36	Cloud infrastructure components,	1	T[1]-Page no 275	PPT
37	Cloud migration considerations	1	T[1]-Page no 309	PPT
38	Cloud migration considerations	1	T[1]-Page no 309	BB
39	TUTORIAL	1		
TOTAL HOURS FOR UNIT-IV		**9**		
UNIT V Securing and Managing				
40	Storage Infrastructure Security threats, and	1	T[1]-Page no 335	BB
41	countermeasures in various domains,	1	T[1]-Page no 337	BB
42	Security solutions for FC-SAN	1	T[1]-Page no 337	BB
43	IP-SAN and NAS environments	1	T[1]-Page no 338	BB
44	Security in virtualized and cloud environments	1	T[1]-Page no 363	BB
45	Monitoring and managing various information infrastructure components	1	T[1]-Page no 366	BB
46	classic and virtual environments	1	T[1]-Page no 367	PPT
47	Information lifecycle Management (ILM)	1	T[1]-Page no 364	BB
48	and storage tiering.	1	T[1]-Page no 375	BB
49	**Discussion on Previous University question papers**			
TOTAL HOURS FOR UNIT-V		**9**		
TOTAL HOURS		**49**		

Text Books:

1. Information Storage and Management: Storing, Managing and Protecting Digital Information in classic, Virtualized and Cloud Environments, 2nd Edition, EMC Educations Services, Wiley, May 2012.

References:

2. Ulf Troppens, Rainer Erkens, Wolfgang Mueller-Friedt, Rainer Wolafka, Nils Haustein ,"Storage Networks Explained: Basics and Application of Fibre Channel SAN, NAS, iSCSI, InfiniBand and FCoE, 2nd Edition, Wiley, July 2009
3. Information Storage and Management: Storing, Managing, and Protecting Digital Information, EMC Education Services, Wiley, January 2010

UNIT-1

INTRODUCTION TO STORAGE TECHNOLOGY

1.1Introduction:

Information is increasingly important in our daily lives. We have become Information dependents of the twenty-first century, living in an on-command, on-demand world that means we need information when and where it is required. We access the Internet every day to perform searches, participate in social networking, send and receive e-mails, share pictures and videos, and scores of other applications. Equipped with a growing number of content-generating devices, more information is being created by individuals than by businesses. Information created by individual gains value when shared with others. When created, information resides locally on devices such as cell phones, cameras, and laptops. To share this information, it needs to be uploaded via networks to data centers. It is interesting to note that while the majority of information is created by individuals, it is stored and managed by a relatively small number of organizations.

The importance, dependency, and volume of information for the business world also continue to grow at astounding rates. Businesses depend on fast and reliable access to information critical to their success. Some of the business applications that process information include airline reservations, telephone billing systems, e-commerce, ATMs, product designs, inventory management, e-mail archives, Web portals, patient records, credit cards, life sciences, and global capital markets.

The increasing criticality of information to the businesses has amplified the challenges in protecting and managing the data. The volume of data that business must manage has driven strategies to classify data according to its value and create rules for the treatment of this data over its lifecycle. These strategies not only provide financial and regulatory benefits at the business level, but also manageability benefits at operational levels to the organization. Data centers now view information storage as one of their core elements, along with applications, databases, operating systems, and networks. Storage technology continues to evolve with technical advancements offering increasingly higher levels of availability, security, scalability, performance, integrity, capacity, and manageability.

1.2Information Storage

Businesses use data to derive information that is critical to their day-to-day Operations. Storage is a repository that enables users to store and retrieve this digital data.

1.2.1Data

Data is a collection of raw facts from which conclusions may be drawn.
Handwritten letters, a printed book, a family photograph, a movie on videotape, printed and duly signed copies of mortgage papers, a bank's ledgers, and an account holder's passbooks are all examples of data.

Before the advent of computers, the procedures and methods adopted for data creation and sharing were limited to fewer forms, such as paper and film. Today, the same data can be con-

verted into more convenient forms such as an e-mail message, an e-book, a bitmapped image, or a digital movie. Data in this form is called digital data and is accessible by the user only after it is processed by a computer.

With the advancement of computer and communication technologies, the rate of data generation and sharing has increased exponentially.

The following is a list of some of the factors that have contributed to the growth of digital data:

■ **Increase in data processing capabilities:** Modern-day computers provide a significant increase in processing and storage capabilities. This enables the conversion of various types of content and media from conventional forms to digital formats.

■ **Lower cost of digital storage:** Technological advances and decrease in the cost of storage devices have provided low-cost solutions and encouraged the development of less expensive data storage devices. This cost benefit has increased the rate at which data is being generated and stored.

■**Affordable and faster communication technology:** The rate of sharing digital data is now much faster than traditional approaches. A handwritten letter may take a week to reach its destination, whereas it only takes a few seconds for an e-mail message to reach its recipient.

Inexpensive and easier ways to create, collect, and store all types of data, coupled with increasing individual and business needs, have led to accelerated data growth, popularly termed the *data explosion*. Data has different purposes and criticality, so both individuals and businesses have contributed in varied proportions to this data explosion.

The importance and the criticality of data vary with time. Most of the data created holds significance in the short-term but becomes less valuable over time.

This governs the type of data storage solutions used. Individuals store data on a variety of storage devices, such as hard disks, CDs, DVDs, or Universal Serial
Bus (USB) flash drives.

Businesses generate vast amounts of data and then extract meaningful information from this data to derive economic benefits. Therefore, businesses need to maintain data and ensure its availability over a longer period.

Furthermore, the data can vary in criticality and may require special handling. For example, legal and regulatory requirements mandate that banks maintain account information for their customers accurately and securely.

Some businesses handle data for millions of customers, and ensure the security and integrity of data over a long period of time. This requires high capacity storage devices with enhanced security features that can retain data for a long period.

1.2.2 Types of Data

Data can be classified as structured or unstructured (see Figure 1-3) based on how it is stored and managed. Structured data is organized in rows and columns in a rigidly defined format so that applications can retrieve and process it efficiently. Structured data is typically stored using a database management system (DBMS).

Data is unstructured if its elements cannot be stored in rows and columns, and is therefore difficult to query and retrieve by business applications. For example, customer contacts may be stored in various forms such as sticky notes, e-mail messages, business cards, or even digital format files such as .doc, .txt, and .pdf . Due its unstructured nature, it is difficult to retrieve using a customer relationship management application. Unstructured data may not have the required components to identify itself uniquely for any type of processing or interpretation. Business are primarily concerned with managing unstructured data because over 80 percent of enterprise data is unstructured and requires significant storage space and effort to manage.

1.2.3 Information

Data, whether structured or unstructured, does not fulfill any purpose for individuals or businesses unless it is presented in a meaningful form. Businesses need to analyze data for it to be of value. *Information* is the intelligence and knowledge derived from data.

Businesses analyze raw data in order to identify meaningful trends. On the basis of these trends, a company can plan or modify its strategy. For example, a retailer identifies customers' preferred products and brand names by analyzing their purchase patterns and maintaining an inventory of those products.

Effective data analysis not only extends its benefits to existing businesses, but also creates the potential for new business opportunities by using the information in creative ways. Job portal is an example. In order to reach a wider set of prospective employers, job seekers post their résumés on various websites offering job search facilities. These websites collect the résumés and post them on centrally accessible locations for prospective employers. In addition, companies post available positions on job search sites. Job-matching software matches keywords from résumés to keywords in job postings. In this manner, the job search engine uses data and turns it into information for employers and job seekers.

Because information is critical to the success of a business, there is an ever present concern about its availability and protection. Legal, regulatory, and contractual obligations regarding the availability and protection of data only add to these concerns. Outages in key industries, such as financial services, telecommunications, manufacturing, retail, and energy cost millions of U.S. dollars per hour.

1.2.4 Storage

Data created by individuals or businesses must be stored so that it is easily accessible for further processing. In a computing environment, devices designed for storing data are termed *storage devices* or simply *storage*. The type of storage used varies based on the type of data and the rate at which it is created and used.

Devices such as memory in a cell phone or digital camera, DVDs, CD-ROMs, and hard disks in personal computers are examples of storage devices.

Businesses have several options available for storing data including internal hard disks, external disk arrays and tapes.

1.2.5 Evolution of Storage Technology and Architecture

Historically, organizations had centralized computers (mainframe) and information storage devices (tape reels and disk packs) in their data center. The evolution of open systems and the affordability and ease of deployment that they offer made it possible for business units/departments to have their own servers and storage. In earlier implementations of open systems, the storage was typically internal to the server.

The proliferation of departmental servers in an enterprise resulted in unprotected, unmanaged, fragmented islands of information and increased operating cost. Originally, there were very limited policies and processes for managing these servers and the data created. To overcome these challenges, storage technology evolved from non-intelligent internal storage to intelligent networked storage Highlights of this technology evolution include:

■ **Redundant Array of Independent Disks (RAID):** This technology was developed to address the cost, performance, and availability requirements of data. It continues to evolve today and is used in all storage architectures such as DAS, SAN, and so on.

■**Direct-attached storage (DAS):** This type of storage connects directly to a server (host) or a group of servers in a cluster. Storage can be either internal or external to the server. External DAS alleviated the challenges of limited internal storage capacity.

■ **Storage area network (SAN):** This is a dedicated, high-performance *Fiber Channel (FC)* network to facilitate *block-level* communication between servers and storage. Storage is partitioned and assigned to a server for accessing its data. SAN offers scalability, availability, performance, and cost benefits compared to DAS.

■**Network-attached storage (NAS):** This is dedicated storage for *file serving* applications. Unlike a SAN, it connects to an existing communication network
(LAN) and provides file access to heterogeneous clients. Because it is purposely built for providing storage to file server applications, it offers higher scalability, availability, performance, and cost benefits compared to general purpose file servers.

■ **Internet Protocol SAN (IP-SAN):** One of the latest evolutions in storage architecture, IP-SAN is a convergence of technologies used in SAN and NAS. IP-SAN provides block-level communication across a local or wide area network (LAN or WAN), resulting in greater consolidation and availability of data.

Storage technology and architecture continues to evolve, which enables organizations to consolidate, protect, optimize, and leverage their data to achieve the highest return on information assets.

1.3 Data Center Infrastructure

Organizations maintain data centers to provide centralized data processing capabilities across the enterprise. Data centers store and manage large amounts of mission-critical data. The data center infrastructure includes computers, storage systems, network devices, dedicated power backups, and environmental controls (such as air conditioning and fire suppression).

Large organizations often maintain more than one data center to distribute data processing workloads and provide backups in the event of a disaster. The storage requirements of a data center are met by a combination of various storage architectures.

1.3.1Core Elements of data center infrastructure

Five core elements are essential for the basic functionality of a data center:

■**Application:** An application is a computer program that provides the logic for computing operations. Applications, such as an order processing system, can be layered on a database, which in turn uses operating system services to perform read/write operations to storage devices.

■**Database:** More commonly, a database management system (DBMS) provides a structured way to store data in logically organized tables that are interrelated. A DBMS optimizes the storage and retrieval of data.

■ **Server and operating system:** A computing platform that runs applications and databases.

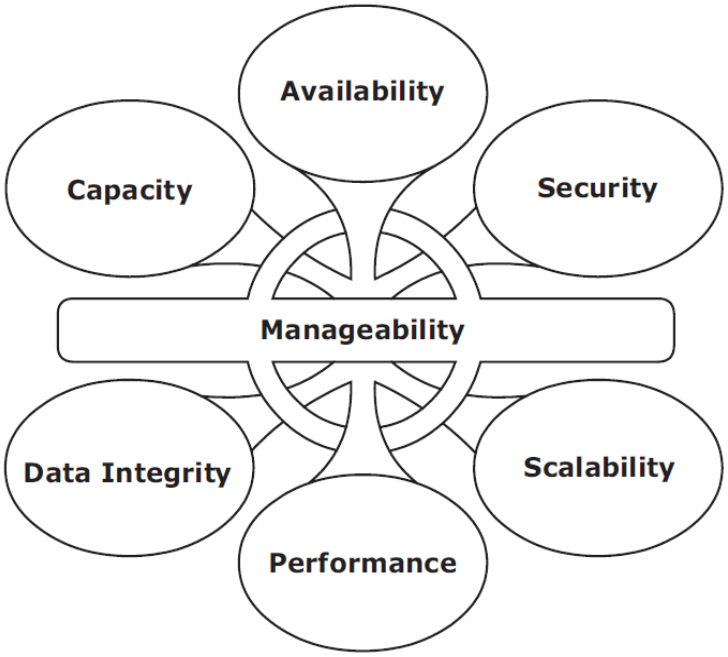
■**Network:** A data path that facilitates communication between clients and servers or between servers and storage.

■ **Storage array:** A device that stores data persistently for subsequent use.

These core elements are typically viewed and managed as separate entities, but all the elements must work together to address data processing requirements.

1.3.2 Key Requirements for Data Center Elements

Uninterrupted operation of data centers is critical to the survival and success of a business. It is necessary to have a reliable infrastructure that ensures data is accessible at all times. While the requirements, are applicable to all elements of the data center infrastructure, our focus here is on storage systems.



■**Availability:** All data center elements should be designed to ensure accessibility.
The inability of users to access data can have a significant negative impact on a business.

■ **Security:** Polices, procedures, and proper integration of the data center core elements that will prevent unauthorized access to information must be established. In addition to the security measures for client access, specific mechanisms must enable servers to access only their allocated resources on storage arrays.

■ **Scalability:** Data center operations should be able to allocate additional processing capabilities or storage on demand, without interrupting business operations. Business growth often requires deploying more servers, new applications, and additional databases. The storage solution should be able to grow with the business.

■ **Performance:** All the core elements of the data center should be able to provide optimal performance and service all processing requests at high speed. The infrastructure should be able to support performance requirements.

■**Data integrity:** Data integrity refers to mechanisms such as error correction codes or parity bits which ensure that data is written to disk exactly as it was received. Any variation in data during its retrieval implies corruption, which may affect the operations of the organization.

■ **Capacity:** Data center operations require adequate resources to store and process large amounts of data efficiently. When capacity requirements increase, the data center must be able to provide additional capacity without interrupting availability, or, at the very least, with minimal disruption.
Capacity may be managed by reallocation of existing resources, rather than by adding new resources.

■**Manageability:** A data center should perform all operations and activities in the most efficient manner. Manageability can be achieved through automation and the reduction of human (manual) intervention in common tasks.

1.3.3 Managing Storage Infrastructure

Managing a modern, complex data center involves many tasks. Key management activities include:

■*Monitoring* is the continuous collection of information and the review of the entire data center infrastructure. The aspects of a data center that are monitored include security, performance, accessibility, and capacity.

■ *Reporting* is done periodically on resource performance, capacity, and utilization. Reporting tasks help to establish business justifications and chargeback of costs associated with data center operations.

■ *Provisioning* is the process of providing the hardware, software, and other resources needed to run a data center. Provisioning activities include capacity and resource planning. *Capacity planning* ensures that the user's and the application's future needs will be addressed in the most cost-effective and controlled manner. *Resource planning* is the process of evaluating and identifying required resources, such as personnel, the facility (site), and the technology. Resource planning ensures that adequate resources are available to meet user and application requirements.

1.3.4 Key Challenges in Managing Information

In order to frame an effective information management policy, businesses need to consider the following key challenges of information management:

■ **Exploding digital universe:** The rate of information growth is increasing exponentially. Duplication of data to ensure high availability and repurposing has also contributed to the multifold increase of information growth.

■ **Increasing dependency on information:** The strategic use of information plays an important role in determining the success of a business and provides competitive advantages in the marketplace.

■ **Changing value of information:** Information that is valuable today may become less important tomorrow. The value of information often changes over time.

UNIT-2
STORAGE SYSTEM ARCHITECTURE

2.1 Host

Users store and retrieve data through applications. The computers on which these applications run are referred to as *hosts*. Hosts can range from simple laptops to complex clusters of servers. A host consists of physical components (hardware devices) that communicate with one another using logical component (software and protocols). Access to data and the overall performance of the storage system environment depend on both the physical and logical components of a host.

2.1.1 Physical Components

A host has three key physical components:

■ Central processing unit (CPU)

■ Storage, such as internal memory and disk devices

■ Input / Output (I/O) devices

The physical components communicate with one another by using a communication pathway called a *bus*. A bus connects the CPU to other components, such as storage and I/O devices.

CPU

The CPU consists of four main components:

■**Arithmetic Logic Unit (ALU):** This is the fundamental building block of the CPU. It performs arithmetical and logical operations such as addition, subtraction, and Boolean functions (AND, OR, and NOT). ■ **Control Unit:** A digital circuit that controls CPU operations and coordinates the functionality of the CPU.

■ **Register:** A collection of high-speed storage locations. The registers store intermediate data that is required by the CPU to execute an instruction and provide fast access because of their proximity to the ALU. CPUs typically have a small number of registers.

■ **Level 1 (L1) cache:** Found on modern day CPUs, it holds data and program instructions that are likely to be needed by the CPU in the near future. The L1 cache is slower than registers, but provides more storage space.

Storage

Memory and storage media are used to store data, either persistently or temporarily.

Memory modules are implemented using semiconductor chips, whereas storage devices use either magnetic or optical media. Memory modules enable data access at a higher speed than the storage media. Generally, there are two types of memory on a host:

■ **Random Access Memory (RAM):** This allows direct access to any memory location and can have data written into it or read from it. RAM is volatile; this type of memory requires a constant supply of power to maintain memory cell content. Data is erased when the system's power is turned off or interrupted.

■ **Read-Only Memory (ROM):** Non-volatile and only allows data to be read from it. ROM holds data for execution of internal routines, such as system startup.

Storage devices are less expensive than semiconductor memory. Examples of storage devices are as follows:

■Hard disk (magnetic)

■ CD-ROM or DVD-ROM (optical)

■ Floppy disk (magnetic)

■ Tape drive (magnetic)

I/O Devices

I/O devices enable sending and receiving data to and from a host. This communication may be one of the following types:

■**User to host communications:** Handled by basic I/O devices, such as the keyboard, mouse, and monitor. These devices enable users to enter data and view the results of operations.

■**Host to host communications:** Enabled using devices such as a Network Interface Card (NIC) or modem.

■**Host to storage device communications:** Handled by a *Host Bus Adaptor (HBA)*. HBA is an application-specific integrated circuit (ASIC) board that performs I/O interface functions between the host and the storage, relieving the CPU from additional I/O processing workload. HBAs also provide connectivity outlets known as *ports* to connect the host to the storage device. A host may have multiple HBAs.

2.1.2 Logical Components of the Host

The logical components of a host consist of the software applications and protocols that enable data communication with the user as well as the physical components. Following are the logical components of a host:

■Operating system

■Device drivers

■ Volume manager

■ File system

■Application

Operating System

An *operating system* controls all aspects of the computing environment. It works between the application and physical components of the computer system. One of the services it provides to the application is data access. The operating system also monitors and responds to user actions and the environment. It organizes and controls hardware components and manages the allocation of hardware resources. It provides basic security for the access and usage of all managed resources. An operating system also performs basic storage management tasks while managing other underlying components, such as the file system, volume manager, and device drivers.

Device Driver

A *device driver* is special software that permits the operating system to interact with a specific device, such as a printer, a mouse, or a hard drive. A device driver enables the operating system to recognize the device and to use a standard interface (provided as an *application programming interface*, or *API*) to access and control devices. Device drivers are hardware dependent and operating system specific.

Volume Manager

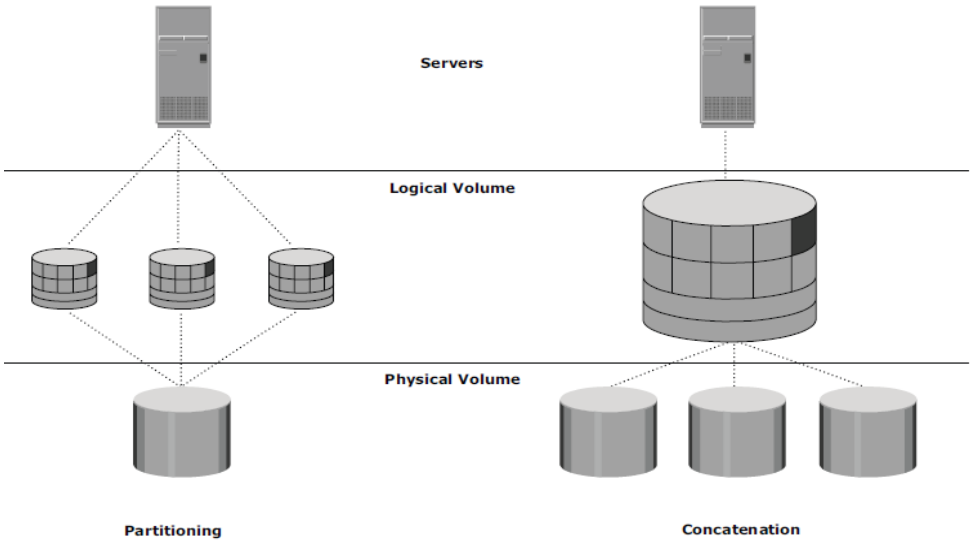
In the early days, an HDD appeared to the operating system as a number of continuous disk blocks. The entire HDD would be allocated for the file system or other data entity used by the operating system or application. The disadvantage was lack of flexibility: As an HDD ran out of space, there was no easy way to extend the file system's size. As the storage capacity of the HDD increased, allocating the entire HDD for the file system often resulted in underutilization of storage capacity.

Disk partitioning was introduced to improve the flexibility and utilization of HDDs. In partitioning, an HDD is divided into logical containers called *logical volumes (LVs)*.For example, a large physical drive can be partitioned into multiple LVs to maintain data according to the file system's and applications' requirements. The partitions are created from groups of contiguous cylinders when the hard disk is initially set up on the host. The host's file system accesses the partitions without any knowledge of partitioning and the physical structure of the disk.

Concatenation is the process of grouping several smaller physical drives and presenting them to the host as one logical drive.

The evolution of *Logical Volume Managers (LVMs)* enabled the dynamic extension of file system capacity and efficient storage management. LVM is software that runs on the host computer and manages the logical and physical storage.

LVM is an optional, intermediate layer between the file system and the physical disk. It can aggregate several smaller disks to form a larger virtual disk or to partition a larger-capacity disk into virtual, smaller-capacity disks, which are then presented to applications. The LVM provides optimized storage access and simplifies storage resource management. It hides details about the physical disk and the location of data on the disk; and it enables administrators to change the storage allocation without changing the hardware, even when the application is running.



The basic LVM components are the physical volumes, volume groups, and logical volumes. In LVM terminology, each physical disk connected to the host system is a physical volume (PV). LVM converts the physical storage provided by the physical volumes to a logical view of storage, which is then used by the operating system and applications. A volume group is created by grouping together one or more physical volumes. A unique physical volume identifier (PVID) is assigned to each physical volume when it is initialized for use by the LVM. Physical volumes can be added or removed from a volume group dynamically. They cannot be shared between volume groups; the entire physical volume becomes part of a volume group. Each physical volume is partitioned into equal-sized data blocks called physical extents when the volume group is created.

Logical volumes are created within a given volume group. A logical volume *can be thought of as a virtual disk partition, while the volume group itself can* be thought of as a disk. A volume group can have a number of logical volumes.

The size of a logical volume is based on a multiple of the physical extents. The logical volume appears as a physical device to the operating system. A logical volume can be made up of noncontiguous physical partitions and can span multiple physical volumes. A file system can be created on a logical volume; and logical volumes can be configured for optimal performance to the application and can be mirrored to provide enhanced data availability.

File System

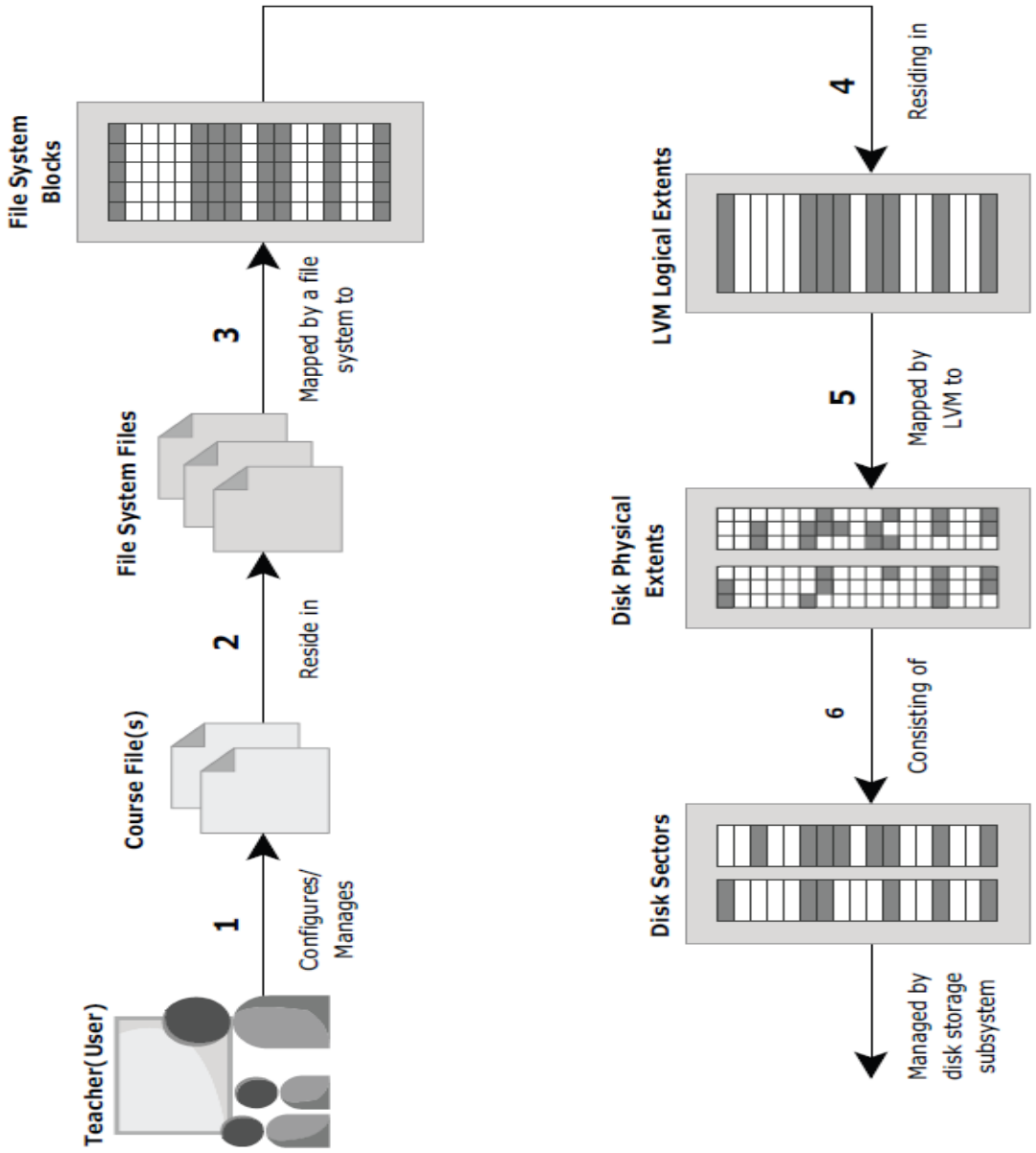
A file is a collection of related records or data stored as a unit with a name. A file system is a hierarchical structure of files. File systems enable easy access to data files residing within a disk drive, a disk partition, or a logical volume. A file system needs host-based logical structures and software routines that control access to files. It provides users with the functionality to create, modify, delete, and access files. Access to the files on the disks is controlled by the permissions given to the file by the owner, which are also maintained by the file system.

A file system organizes data in a structured hierarchical manner via the use of directories, which are containers for storing pointers to multiple files. All file systems maintain a pointer map to the directories, subdirectories, and files that are part of the file system. Some of the common file systems are as follows:

■ FAT 32 (File Allocation Table) for Microsoft Windows
■NT File System (NTFS) for Microsoft Windows
■UNIX File System (UFS) for UNIX

■ Extended File System (EXT2/3) for Linux

Apart from the files and directories, the file system also includes a number of other related records, which are collectively called the metadata. For example, metadata in a UNIX environment consists of the superblock, the nodes, and the list of data blocks free and in use. The metadata of a file system has to be consistent in order for the file system to be considered healthy. A superblock contains important information about the file system, such as the file system type, creation and modification dates, size and layout, the count of available resources (such as number of free blocks, nodes, etc.), and a flag indicating the mount status of the file system. A node is associated with every file and directory and contains information about file length, ownership, access privileges, time of last access/modification, number of links, and the addresses for finding the location on the physical disk where the actual data is stored.



A file system *block* is the smallest "container" of physical disk space allocated for data. Each file system block is a contiguous area on the physical disk. The block size of a file system is fixed at the time of its creation. File system size depends on block size and the total number of blocks of data stored. A file can span multiple file system blocks because most files are larger than the predefined block size of the file system. File system blocks cease to be contiguous

(i.e., become fragmented) when new blocks are added or deleted. Over time, as files grow larger, the file system becomes increasingly fragmented.
Process of mapping user files to the disk storage subsystem with an LVM:
1. Files are created and managed by users and applications.
2. These files reside in the file systems.
3. The file systems are then mapped to units of data, or file system blocks.
4. The file system blocks are mapped to logical extents.
5. These in turn are mapped to disk physical extents either by the operating system or by the LVM.
6. These physical extents are mapped to the disk storage subsystem.
If there is no LVM, then there are no logical extents. Without LVM, file system blocks are directly mapped to disk sectors.

The file system tree starts with the root directory. The root directory has a number of subdirectories. A file system should be mounted before it can be used.

A file system can be either a journaling file system or a non-journaling file system. Non-journaling file systems create a potential for lost files because they may use many separate writes to update their data and metadata. If the system crashes during the write process, metadata or data may be lost or corrupted.

When the system reboots, the file system attempts to update the metadata structures by examining and repairing them. This operation takes a long time on large file systems. If there is insufficient information to recreate the desired or original structure, files may be misplaced or lost, resulting in corrupted file systems.

A journaling file system uses a separate area called a log, or journal. This journal may contain all the data to be written (physical journal), or it may contain only the metadata to be updated (logical journal). Before changes are made to the file system, they are written to this separate area. Once the journal has been updated, the operation on the file system can be performed. If the system crashes during the operation, there is enough information in the log to "replay" the log record and complete the operation. Journaling results in a very quick file system check because it only looks at the active, most recently accessed parts of a large file system. In addition, because information about the pending operation is saved, the risk of files being lost is reduced.

Block-Level Access

Block-level access is the basic mechanism for disk access. In this type of access, data is stored and retrieved from disks by specifying the logical block address.

The block address is derived based on the geometric configuration of the disks. Block size defines the basic unit of data storage and retrieval by an application. Databases, such as Oracle and SQL Server, define the block size for data access and the location of the data on the disk in terms of the logical block address when an I/O operation is performed.

File-Level Access

File-level access is an abstraction of block-level access. File-level access to data is provided by specifying the name and path of the file. It uses the underlying block-level access to storage and hides the complexities of logical block addressing (LBA) from the application and the DBMS.

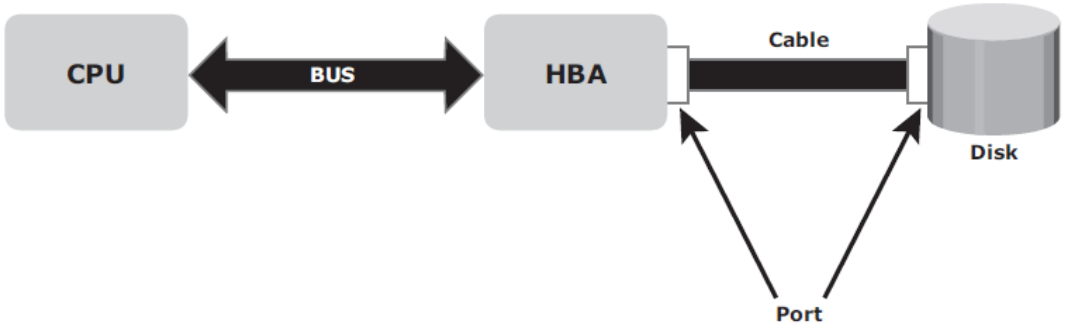
2.2 Connectivity

Connectivity refers to the interconnection between hosts or between a host and any other peripheral devices, such as printers or storage devices. The discussion here focuses on the connectivity between the host and the storage device. The components of connectivity in a storage system environment can be classified as physical and logical. The *physical components* are the hardware

elements that connect the host to storage and the *logical components* of connectivity are the protocols used for communication between the host and storage.

2.2.1 Physical Components of Connectivity

The three physical components of connectivity between the host and storage are Bus, Port, and Cable.



The bus is the collection of paths that facilitates data transmission from one part of a computer to another, such as from the CPU to the memory. The port is a specialized outlet that enables connectivity between the host and external devices. Cables connect hosts to internal or external devices using copper or fiber optic media.

Physical components communicate across a bus by sending bits (control, data, and address) of data between devices. These bits are transmitted through the bus in either of the following ways:

■**Serially:** Bits are transmitted sequentially along a single path. This transmission can be unidirectional or bidirectional.

■**In parallel:** Bits are transmitted along multiple paths simultaneously.

Parallel can also be bidirectional.

The size of a bus, known as its width, determines the amount of data that can be transmitted through the bus at one time. The width of a bus can be compared to the number of lanes on a highway. For example, a 32-bit bus can transmit 32 bits of data and a 64-bit bus can transmit 64 bits of data simultaneously. Every bus has a clock speed measured in MHz (megahertz). These represent the data transfer rate between the end points of the bus. A fast bus allows faster transfer of data, which enables applications to run faster.

Buses, as conduits of data transfer on the computer system, can be classified as follows:

■ **System bus:** The bus that carries data from the processor to memory.

■ **Local or I/O bus:** A high-speed pathway that connects directly to the processor and carries data between the peripheral devices, such as storage devices and the processor.

2.2.2 Logical Components of Connectivity (Protocols)

The popular interface protocol used for the local bus to connect to a peripheral device is peripheral component interconnect (PCI). The interface protocols that connect to disk systems are Integrated Device Electronics/Advanced Technology
Attachment (IDE/ATA) and Small Computer System Interface (SCSI).

PCI

PCI is a specification that standardizes how PCI expansion cards, such as network cards or modems, exchange information with the CPU. PCI provides the interconnection between the CPU and attached devices. The plug-and-play functionality of PCI enables the host to easily recognize and configure new cards and devices. The width of a PCI bus can be 32 bits or 64 bits. A 32-bit

PCI bus can provide a throughput of 133 MB/s. PCI Express is an enhanced version of PCI bus with considerably higher throughput and clock speed.

IDE/ATA

IDE/ATA is the most popular interface protocol used on modern disks. This protocol offers excellent performance at relatively low cost.

An Integrated Device Electronics/Advanced Technology Attachment (IDE/ATA) disk supports the IDE protocol. The term IDE/ATA conveys the dual-naming conventions for various generations and variants of this interface. The IDE component in IDE/ATA provides the specification for the controllers connected to the computer's motherboard for communicating with the device attached.

The ATA component is the interface for connecting storage devices, such as CD-ROMs, floppy disk drives, and HDDs, to the motherboard.

IDE/ATA has a variety of standards and names, such as ATA, ATA/ATAPI, EIDE, ATA-2, Fast ATA, ATA-3, Ultra ATA, and Ultra DMA. The latest version of ATA—Ultra DMA/133—supports a throughput of 133 MB per second.

In a master-slave configuration, an ATA interface supports two storage devices per connector. However, if the performance of the drive is important, sharing a port between two devices is not recommended.

A 40-pin connector is used to connect ATA disks to the motherboard, and a
34-pin connector is used to connect floppy disk drives to the motherboard.

An IDE/ATA disk offers excellent performance at low cost, making it a popular and commonly used hard disk.

A SATA (Serial ATA) is a serial version of the IDE/ATA specification. SATA is a disk-interface technology that was developed by a group of the industry's leading vendors with the aim of replacing parallel ATA.

A SATA provides point-to-point connectivity up to a distance of one meter and enables data transfer at a speed of 150 MB/s. Enhancements to the SATA have increased the data transfer speed up to 600 MB/s.

A SATA bus directly connects each storage device to the host through a dedicated link, making use of *low-voltage differential signaling (LVDS)*. LVDS is an electrical signaling system that can provide high-speed connectivity over low-cost, twisted-pair copper cables. For data transfer, a SATA bus uses LVDS with a voltage of 250 mV.

A SATA bus uses a small 7-pin connector and a thin cable for connectivity.

A SATA port uses 4 signal pins, which improves its pin efficiency compared to the parallel ATA that uses 26 signal pins, for connecting an 80-conductor ribbon cable to a 40-pin header connector.

SATA devices are *hot-pluggable,* which means that they can be connected or removed while the host is up and running. A SATA port permits single-device connectivity. Connecting multiple SATA drives to a host requires multiple ports to be present on the host. Single-device connectivity enforced in SATA , eliminates the performance problems caused by cable or port sharing in IDE/ATA.

SCSI

SCSI has emerged as a preferred protocol in high-end computers. This interface is far less commonly used than IDE/ATA on personal computers due to its higher cost. SCSI was initially used

as a parallel interface, enabling the connection of devices to a host. SCSI has been enhanced and now includes a wide variety of related technologies and standards.

FC protocol

The FC architecture forms the fundamental construct of the SAN infrastructure. *Fiber Channel* is a high-speed network technology that runs on high-speed optical fiber cables (preferred for front-end SAN connectivity) and serial copper cables (preferred for back-end disk connectivity). The FC technology was created to meet the demand for increased speeds of data transfer among computers, servers, and mass storage subsystems.

Higher data transmission speeds are an important feature of the FC networking technology. The initial implementation offered throughput of 100 MB/s (equivalent to raw bit rate of 1Gb/s i.e. 1062.5 Mb/s in Fiber Channel), which was greater than the speeds of Ultra SCSI (20 MB/s) commonly used in DAS environments. FC in full-duplex mode could sustain throughput of 200 MB/s. In comparison with Ultra-SCSI, FC is a significant leap in storage networking technology. Latest FC implementations of 8 GFC (Fiber Channel) offers throughput of 1600 MB/s (raw bit rates of 8.5 GB/s), whereas Ultra320 SCSI is available with a throughput of 320 MB/s. The FC architecture is highly scalable and theoretically a single FC network can accommodate approximately 15 million nodes.

Internet protocol

IP offers easier management and better interoperability. When block I/O is run over IP, the existing network infrastructure can be leveraged, which is more economical than investing in new SAN hardware and software. Many long-distance, disaster recovery (DR) solutions are already leveraging IP-based networks. In addition, many robust and mature security options are now available for IP networks. With the advent of block storage technology that leverages
IP networks (the result is often referred to as IP SAN), organizations can extend the geographical reach of their storage infrastructure.

2.3 Storage

The storage device is the most important component in the storage system environment. A storage device uses magnetic or solid state media. Disks, tapes, and diskettes use magnetic media. CD-ROM is an example of a storage device that uses optical media, and removable flash memory card is an example of solid state media.

Tapes are a popular storage media used for backup because of their relatively low cost. In the past, data centers hosted a large number of tape drives and processed several thousand reels of tape. However, tape has the following limitations:

■Data is stored on the tape linearly along the length of the tape. Search and retrieval of data is done sequentially, invariably taking several seconds to access the data. As a result, random data access is slow and time consuming.

This limits tapes as a viable option for applications that require real-time, rapid access to data.

■In a shared computing environment, data stored on tape cannot be accessed by multiple applications simultaneously, restricting its use to one application at a time.

■On a tape drive, the read/write head touches the tape surface, so the tapedegrades or wears out after repeated use.

■The storage and retrieval requirements of data from tape and the overhead associated with managing tape media are significant.

In spite of its limitations, tape is widely deployed for its cost effectiveness and mobility. Continued development of tape technology is resulting in high capacity medias and high speed drives. Modern tape libraries come with additional memory (cache) and / or disk drives to increase data

throughput. With these and added intelligence, today's tapes are part of an end-to-end data management solution, especially as a low-cost solution for storing infrequently accessed data and as long-term data storage.

Optical disk storage is popular in small, single-user computing environments.

It is frequently used by individuals to store photos or as a backup medium on personal/laptop computers. It is also used as a distribution medium for single applications, such as games, or as a means of transferring small amounts of data from one self-contained system to another. Optical disks have limited capacity and speed, which limits the use of optical media as a business data storage solution.

The capability to write once and read many (WORM) is one advantage of optical disk storage. A CD-ROM is an example of a WORM device. Optical disks, to some degree, guarantee that the content has not been altered, so they can be used as low-cost alternatives for long-term storage of relatively small amounts of fixed content that will not change after it is created. Collections of optical disks in an array, called *jukeboxes*, are still used as a fixed-content storage solution.

Other forms of optical disks include CD-RW and variations of DVD.

Disk drives are the most popular storage medium used in modern computers for storing and accessing data for performance-intensive, online applications.

Disks support rapid access to random data locations. This means that data can be written or retrieved quickly for a large number of simultaneous users or applications.

In addition, disks have a large capacity. Disk storage arrays are configured with multiple disks to provide increased capacity and enhanced performance.

2.4 RAID

RAID is an enabling technology that leverages multiple disks as part of a set, which provides data protection against HDD failures. In general, RAID implementations also improve the I/O performance of storage systems by storing data across multiple HDDs.

2.4.1 Implementation of RAID

There are two types of RAID implementation, hardware and software. Both have their merits and demerits and are discussed in this section.

Software RAID

Software RAID uses host-based software to provide RAID functions. It is implemented at the operating-system level and does not use a dedicated hardware controller to manage the RAID array.

Software RAID implementations offer cost and simplicity benefits when compared with hardware RAID. However, they have the following limitations:

■ **Performance:** Software RAID affects overall system performance. This is due to the additional CPU cycles required to perform RAID calculations.

The performance impact is more pronounced for complex implementations of RAID

■ **Supported features:** Software RAID does not support all RAID levels.

■**Operating system compatibility:** Software RAID is tied to the host operating system hence upgrades to software RAID or to the operating system should be validated for compatibility. This leads to inflexibility in the data processing environment.

Hardware RAID

In *hardware RAID* implementations, a specialized hardware controller is implemented either on the host or on the array. These implementations vary in the way the storage array interacts with the host.

Controller card RAID is host-based hardware RAID implementation in which a specialized RAID controller is installed in the host and HDDs are connected to it. The RAID Controller interacts with the hard disks using a PCI bus.

Manufacturers also integrate RAID controllers on motherboards. This integration reduces the overall cost of the system, but does not provide the flexibility required for high-end storage systems.

The external RAID controller is an array-based hardware RAID. It acts as an interface between the host and disks. It presents storage volumes to the host,which manage the drives using the supported protocol. Key functions of RAID controllers are:

■Management and control of disk aggregations

■ Translation of I/O requests between logical disks and physical disks

■Data regeneration in the event of disk failures

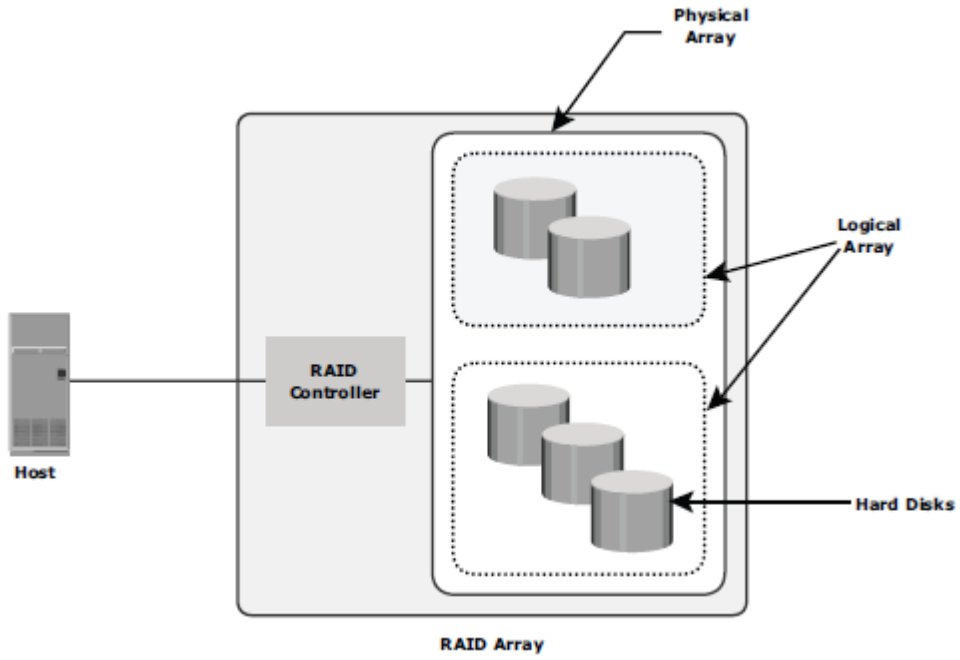
2.4.2 RAID Array Components

A RAID array is an enclosure that contains a number of HDDs and the supporting hardware and software to implement RAID. HDDs inside a RAID array are usually contained in smaller sub-enclosures. These sub-enclosures, or physical arrays, hold a fixed number of HDDs, and may also include other supporting hardware, such as power supplies. A subset of disks within a RAID array can be grouped to form logical associations called logical arrays, also known as a RAID set or a RAID group.

Logical arrays are comprised of logical volumes (LV). The operating system recognizes the LVs as if they are physical HDDs managed by the RAID controller.

The number of HDDs in a logical array depends on the RAID level used.

Configurations could have a logical array with multiple physical arrays or a physical array with multiple logical arrays.



2.4.3 RAID Technologies

RAID levels are defined on the basis of striping, mirroring, and parity techniques. These techniques determine the data availability and performance characteristics of an array. Some RAID arrays use one technique, whereas others use a combination of techniques. Application performance and data availability requirements determine the RAID level selection.

Striping

A RAID set is a group of disks. Within each disk, a predefined number of contiguously addressable disk blocks are defined as *strips*. The set of aligned strips that spans across all the disks within the RAID set is called a *stripe*. *Strip size* (also called *stripe depth*) describes the number of blocks in a *strip*, and is the maximum amount of data that can be written to or read from a single

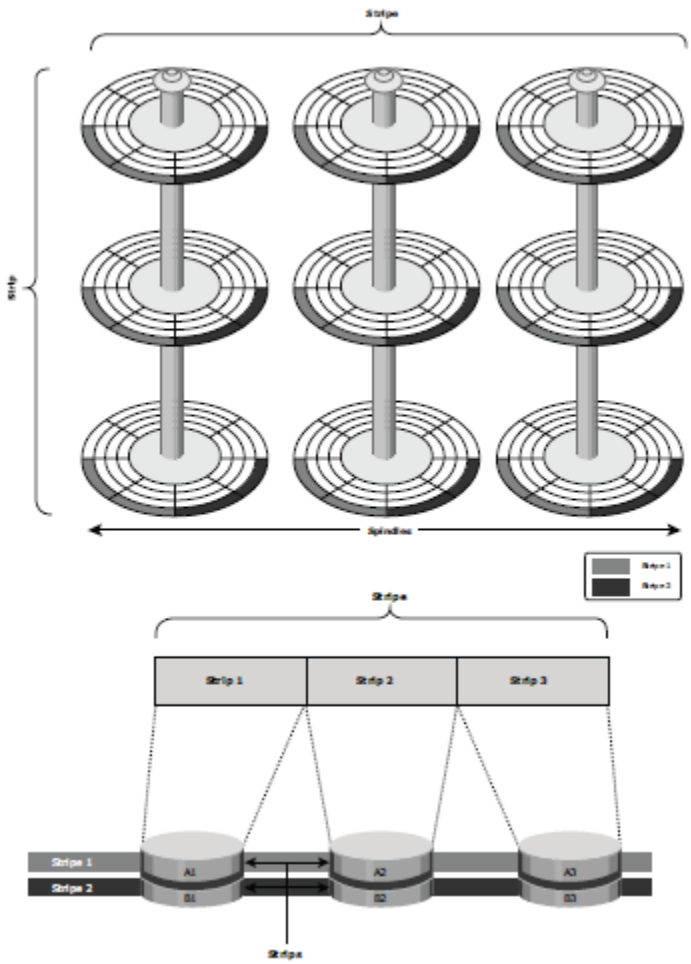
HDD in the set before the next HDD is accessed, assuming that the accessed data starts at the beginning of the strip. Note that all strips in a stripe have the same number of blocks, and decreasing strip size means that data is broken into smaller pieces when spread across the disks.

Stripe size is a multiple of strip size by the number of HDDs in the RAID set.

Stripe width refers to the number of data strips in a stripe.

Striped RAID does not protect data unless parity or mirroring is used.

However, striping may significantly improve I/O performance. Depending on the type of RAID implementation, the RAID controller can be configured to access data across multiple HDDs simultaneously.

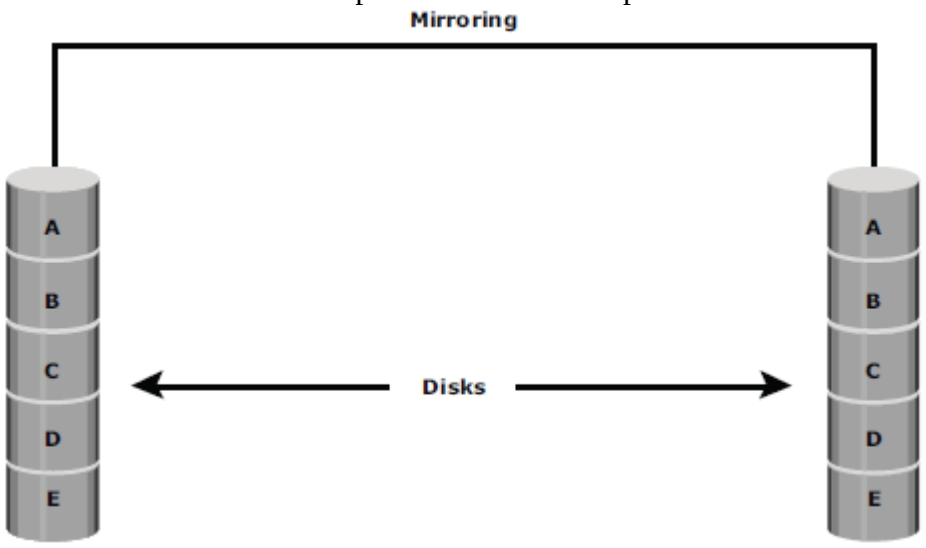


Mirroring

Mirroring is a technique whereby data is stored on two different HDDs, yielding two copies of data. In the event of one HDD failure, the data is intact on the surviving HDD (see Figure 3-3) and the controller continues to service the host's data requests from the surviving disk of a mirrored pair.

Mirroring involves duplication of data — the amount of storage capacity needed is twice the amount of data being stored. Therefore, mirroring is considered expensive and is preferred for

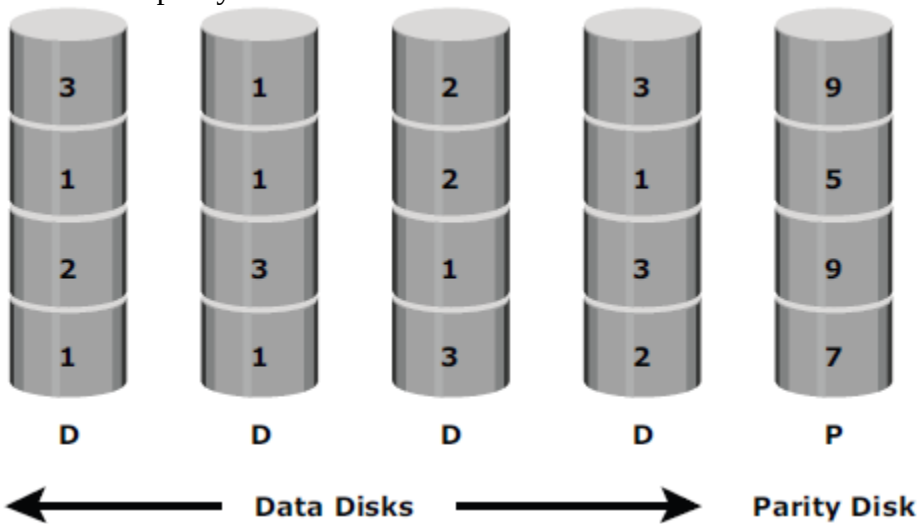
mission-critical applications that cannot afford data loss. Mirroring improves read performance because read requests can be serviced by both disks. However, write performance deteriorates, as each write request manifests as two writes on the HDDs. In other words, mirroring does not deliver the same levels of write performance as a striped RAID.



Parity

Parity is a method of protecting striped data from HDD failure without the cost of mirroring. An additional HDD is added to the stripe width to hold parity, a mathematical construct that allows re-creation of the missing data. Parity is a redundancy check that ensures full protection of data without maintaining a full set of duplicate data. Parity information can be stored on separate, dedicated HDDs or distributed across all the drives in a RAID set. Figure 3-4 shows a parity RAID. The first four disks, labeled *D*, contain the data. The fifth disk, labeled *P*, stores the parity information, which in this case is the sum of the elements in each row. Now, if one of the *D*s fails, the missing value can be calculated by subtracting the sum of the rest of the elements from the parity value. the computation of parity is represented as a simple arithmetic operation on the data. However, parity calculation is a *bitwise XOR* operation.

Calculation of parity is a function of the RAID controller.



Compared to mirroring, parity implementation considerably reduces the cost associated with data protection. Consider a RAID configuration with five disks. Four of these disks hold data, and the fifth holds parity information.

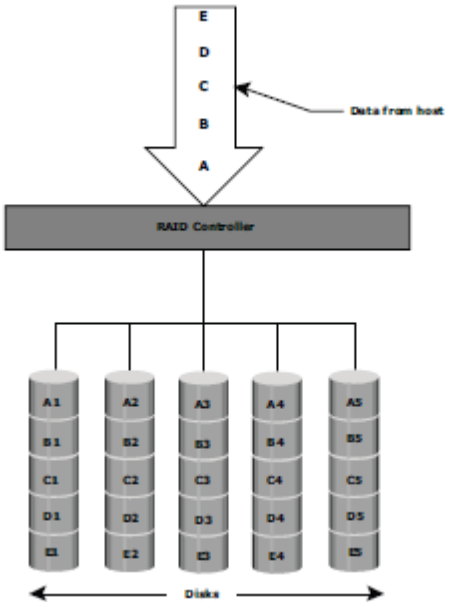
Parity requires 25 percent extra disk space compared to mirroring, which requires 100 percent extra disk space. However, there are some disadvantages of using parity. Parity information is generated from data on the data disk. Therefore, parity is recalculated every time there is a change in data. This recalculation is time-consuming and affects the performance of the RAID controller.

2.4.4 RAID Levels

RAID 0 Striped array with no fault tolerance
RAID 1 Disk mirroring
RAID 3 Parallel access array with dedicated parity disk
RAID 4 Striped array with independent disks and a dedicated parity disk
RAID 5 Striped array with independent disks and distributed parity
RAID 6 Striped array with independent disks and dual distributed parity
Nested Combinations of RAID levels. Example: RAID 1 + RAID 0

RAID 0

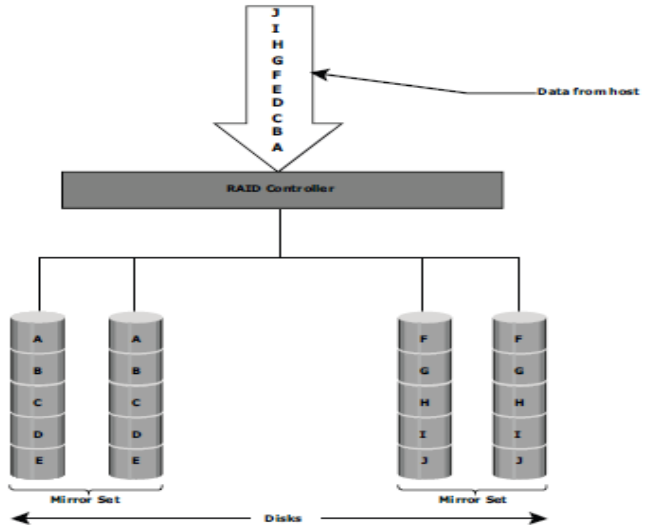
In a RAID 0 configuration, data is striped across the HDDs in a RAID set. It utilizes the full storage capacity by distributing strips of data over multiple HDDs in a RAID set. To read data, all the strips are put back together by the controller.
The stripe size is specified at a host level for software RAID and is vendor specific for hardware RAID. Figure 3-5 shows RAID 0 on a storage array in which data is striped across 5 disks. When the number of drives in the array increases, performance improves because more data can be read or written simultaneously.
RAID 0 is used in applications that need high I/O throughput. However, if these applications require high availability, RAID 0 does not provide data protection and availability in the event of drive failures.



RAID 1

In a RAID 1 configuration, data is mirrored to improve fault tolerance. A RAID 1 group consists of at least two HDDs. As explained in mirroring, every write is written to both disks, which is transparent to the host in a hardware RAID implementation. In the event of disk failure, the im-

pact on data recovery is the least among all RAID implementations. This is because the RAID controller uses the mirror drive for data recovery and continuous operation. RAID 1 is suitable for applications that require high availability.



Nested RAID

Most data centers require data redundancy and performance from their RAID arrays. RAID 0+1 and RAID 1+0 combine the performance benefits of RAID 0 with the redundancy benefits of RAID 1. They use striping and mirroring techniques and combine their benefits. These types of RAID require an even number of disks, the minimum being four (see Figure 3-7).

RAID 1+0 is also known as RAID 10 (Ten) or RAID 1/0. Similarly, RAID 0+1 is also known as RAID 01 or RAID 0/1. RAID 1+0 performs well for workloads that use small, random, write-intensive I/O. Some applications that benefit from RAID 1+0 include the following:

■High transaction rate Online Transaction Processing (OLTP)

■Large messaging installations

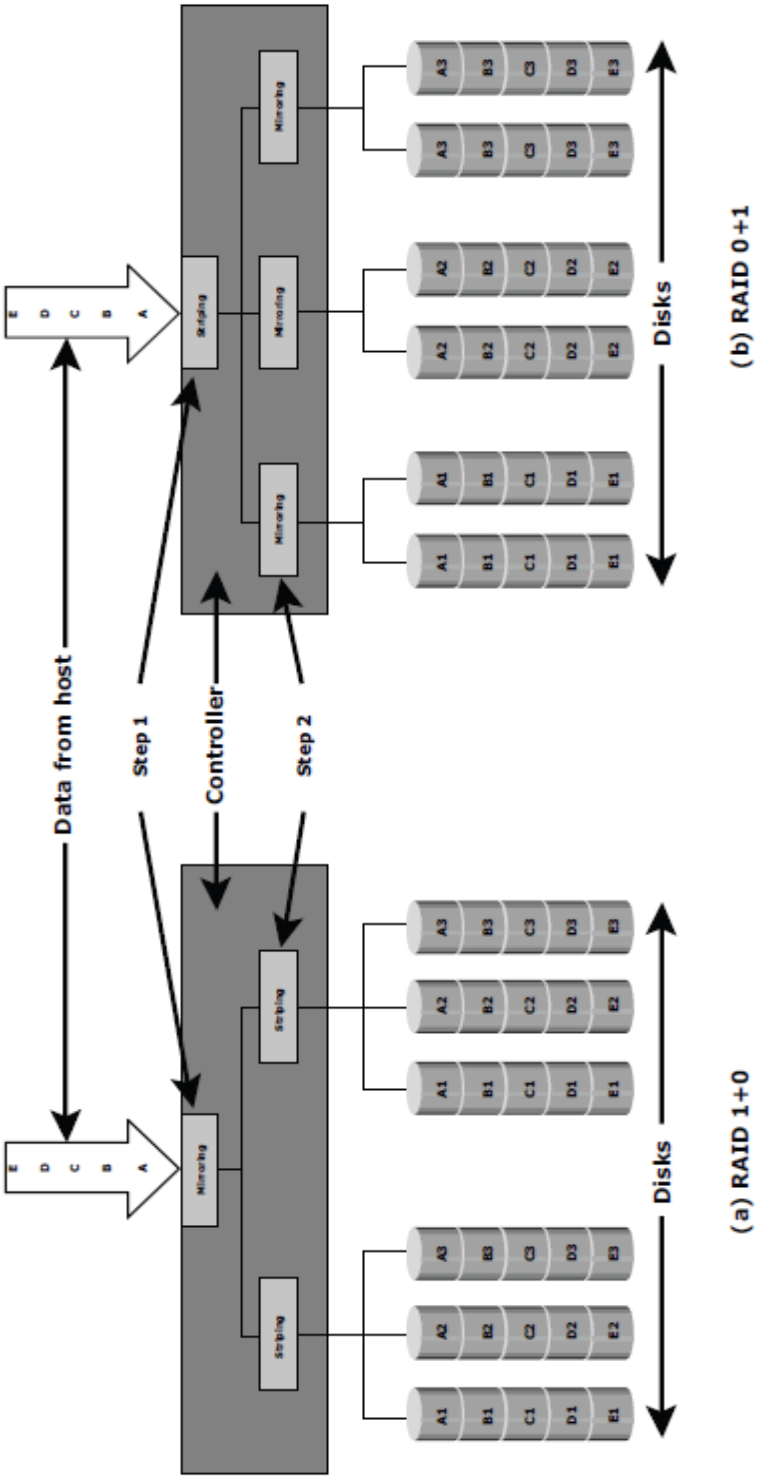
■Database applications that require high I/O rate, random access, and high availability

A common misconception is that RAID 1+0 and RAID 0+1 are the same. Under normal conditions, RAID levels 1+0 and 0+1 offer identical benefits. However, rebuild operations in the case of disk failure differ between the two.

RAID 1+0 is also called *striped mirror*. The basic element of RAID 1+0 is a mirrored pair, which means that data is first mirrored and then both copies of data are striped across multiple HDDs in a RAID set. When replacing a failed drive, only the mirror is rebuilt. In other words, the disk array controller uses the surviving drive in the mirrored pair for data recovery and continuous operation.

Data from the surviving disk is copied to the replacement disk.

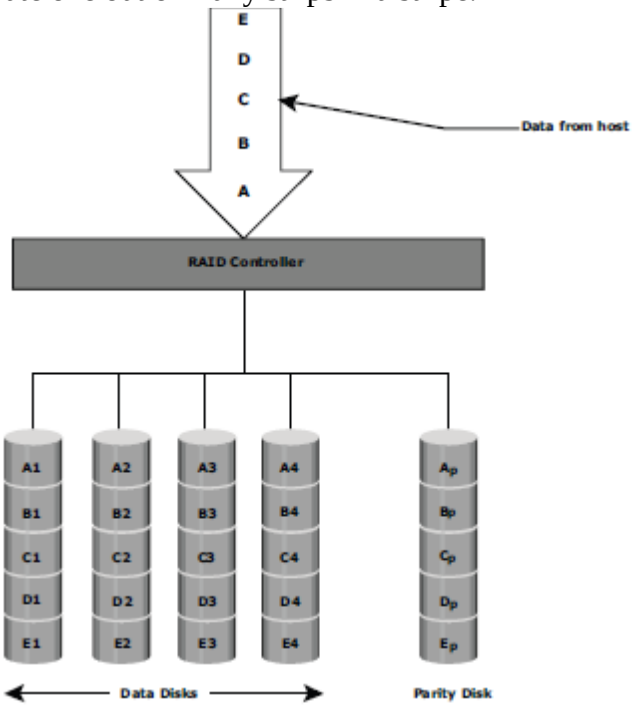
RAID 0+1 is also called *mirrored stripe*. The basic element of RAID 0+1 is a stripe. This means that the process of striping data across HDDs is performed initially and then the entire stripe is mirrored. If one drive fails, then the entire stripe is faulted. A rebuild operation copies the entire stripe, copying data from each disk in the healthy stripe to an equivalent disk in the failed stripe. This causes increased and unnecessary I/O load on the surviving disks and makes the RAID set more vulnerable to a second disk failure.



RAID 3

RAID 3 stripes data for high performance and uses parity for improved fault tolerance. Parity information is stored on a dedicated drive so that data can be reconstructed if a drive fails. For

example, of five disks, four are used for data and one is used for parity. Therefore, the total disk space required is 1.25 times the size of the data disks. RAID 3 always reads and writes complete stripes of data across all disks, as the drives operate in parallel. There are no partial writes that update one out of many strips in a stripe.



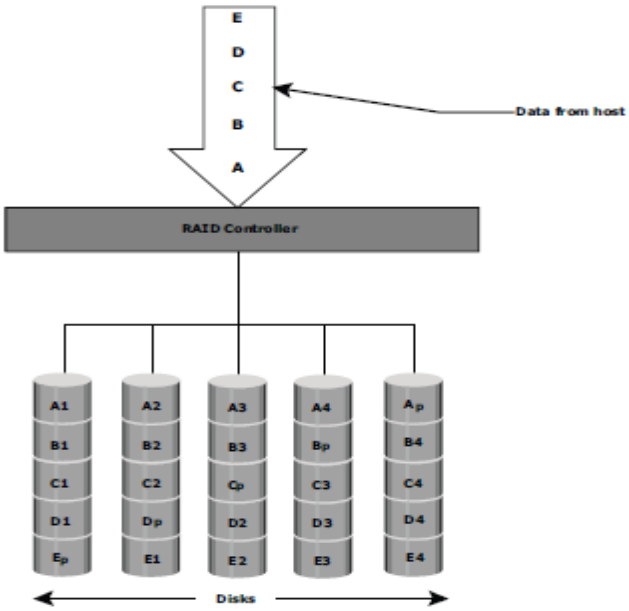
RAID 3 provides good bandwidth for the transfer of large volumes of data.
RAID 3 is used in applications that involve large sequential data access, such as video streaming.

RAID 4

Similar to RAID 3, RAID 4 stripes data for high performance and uses parity for improved fault tolerance (refer to Figure 3-8). Data is striped across all disks except the parity disk in the array. Parity information is stored on a dedicated disk so that the data can be rebuilt if a drive fails. Striping is done at the block level.
Unlike RAID 3, data disks in RAID 4 can be accessed independently so that specific data elements can be read or written on single disk without read or write of an entire stripe. RAID 4 provides good read throughput and reasonable write throughput.

RAID 5

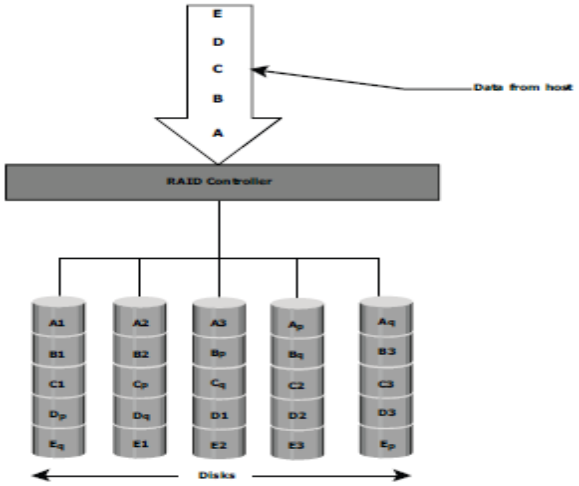
RAID 5 is a very versatile RAID implementation. It is similar to RAID 4 because it uses striping and the drives (strips) are independently accessible. The difference between RAID 4 and RAID 5 is the parity location. In RAID 4, parity is written to a dedicated drive, creating a write bottleneck for the parity disk. In RAID 5, parity is distributed across all disks. The distribution of parity in RAID 5 overcomes the write bottleneck.



RAID 5 is preferred for messaging, data mining, medium-performance media serving, and relational database management system (RDBMS) implementations in which database administrators (DBAs) optimize data access.

RAID 6

RAID 6 works the same way as RAID 5 except that RAID 6 includes a second parity element to enable survival in the event of the failure of two disks in a RAID group. Therefore, a RAID 6 implementation requires at least four disks. RAID 6 distributes the parity across all the disks. The write penalty in RAID 6 is more than that in RAID 5; therefore, RAID 5 writes perform better than RAID 6. The rebuild operation in RAID 6 may take longer than that in RAID 5 due to the presence of two parity sets.



UNIT-3
INTRODUCTION NETWORK STORAGE

3.1 Evolution of Network storage

Organizations are experiencing an explosive growth in information. This information needs to be stored, protected, optimized, and managed efficiently. Data center managers are burdened with the challenging task of providing low-cost, high-performance information management solutions. An effective information management solution must provide the following:

■ **Just-in-time information to business users:**
Information must be available to business users when they need it. The explosive growth in online storage, proliferation of new servers and applications, spread of mission-critical data throughout enterprises, and demand for 24 × 7 data availability are some of the challenges that need to be addressed.

■ **Integration of information infrastructure with business processes:** The storage infrastructure should be integrated with various business processes without compromising its security and integrity.

■ **Flexible and resilient storage architecture:** The storage infrastructure must provide flexibility and resilience that aligns with changing business requirements. Storage should scale without compromising performance requirements of the applications and, at the same time, the total cost of managing information must be low.

3.1.1 Direct-attached storage(DAS)

Direct-attached storage (DAS) is often referred to as a stove piped storage environment. Hosts "own" the storage and it is difficult to manage and share resources on these isolated storage devices. Efforts to organize this dispersed data led to the emergence of the storage area network (SAN). SAN is a high speed, dedicated network of servers and shared storage devices. Traditionally connected over Fiber Channel (FC) networks, a SAN forms a single-storage pool and facilitates data centralization and consolidation. SAN meets the storage demands efficiently with better economies of scale. A SAN also provides effective maintenance and protection of data. This chapter provides detailed insight into the FC technology on which a SAN is deployed and also reviews SAN design and management fundamentals.

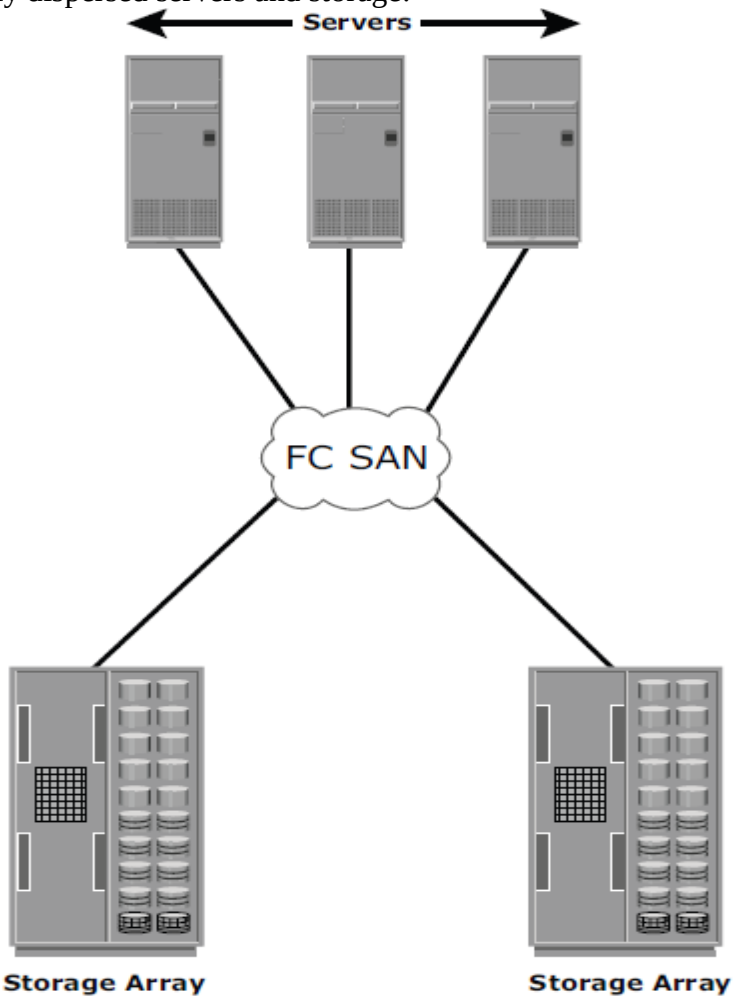
3.1.2 Fiber Channel (FC)

The FC architecture forms the fundamental construct of the SAN infrastructure. Fiber Channel is a high-speed network technology that runs on high-speed optical fiber cables (preferred for front-end SAN connectivity) and serial copper cables (preferred for back-end disk connectivity). The FC technology was created to meet the demand for increased speeds of data transfer among computers, servers, and mass storage subsystems. Although FC networking was introduced in 1988, the FC standardization process began when the American National Standards Institute (ANSI) chartered the Fiber Channel Working Group (FCWG). By 1994, the new high-speed computer interconnection standard was developed and the Fiber Channel Association (FCA) was founded with 70 charter member companies. Technical Committee T11, which is the committee within INCITS (International Committee for Information Technology Standards), is responsible for Fiber Channel interfaces. T11 (previously known as X3T9.3) has been producing interface standards for high performance and mass storage applications since the 1970s. Higher data transmission speeds are an important feature of the FC networking technology. The initial implementation offered throughput of 100 MB/s (equivalent to raw bit rate of 1Gb/s i.e. 1062.5 Mb/s in Fiber Channel), which was greater than the speeds of Ultra SCSI (20 MB/s) commonly used in DAS environments. FC in full-duplex mode could sustain throughput of 200 MB/s. In comparison with Ultra-SCSI, FC is a significant leap in storage networking technology.
Latest FC implementations of 8 GFC (Fiber Channel) offers throughput of 1600 MB/s (raw bit rates of 8.5 GB/s), whereas Ultra320 SCSI is available with a throughput of 320 MB/s. The FC

architecture is highly scalable and theoretically a single FC network can accommodate approximately 15 million nodes

3.1.3 The SAN and Its Evolution

A *storage area network (SAN)* carries data between servers (also known as *hosts*) and storage devices through fiber channel switches (see Figure 6-1). A SAN enables storage consolidation and allows storage to be shared across multiple servers. It enables organizations to connect geographically dispersed servers and storage.



SAN implementation

A SAN provides the physical communication infrastructure and enables secure and robust communication between host and storage devices. The SAN management interface organizes connections and manages storage elements and hosts.

In its earliest implementation, the SAN was a simple grouping of hosts and the associated storage that was connected to a network using a hub as a connectivity device. This configuration of a SAN is known as a *Fiber Channel Arbitrated Loop (FC-AL)*.Use of hubs resulted in isolated FC-AL SAN islands because hubs provide limited connectivity and bandwidth.

The inherent limitations associated with hubs gave way to high-performance FC *switches*. The switched fabric topologies improved connectivity and performance, which enabled SANs to be highly scalable. This enhanced data accessibility to applications across the enterprise. FC-AL has been abandoned for SANs due to its limitations, but still survives as a disk-drive interface.

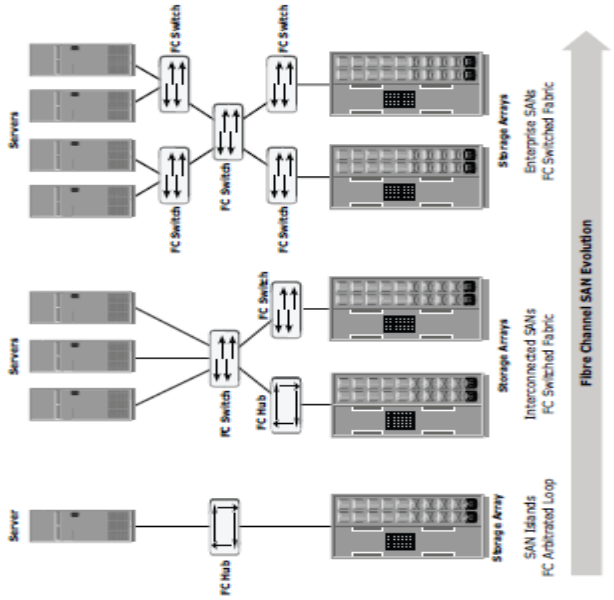
Today, Internet Protocol (IP) has become an option to interconnect geographically separated SANs. Two popular protocols that extend block-level access to applications over IP are iSCSI and Fiber Channel over IP (FCIP).

Components of SAN

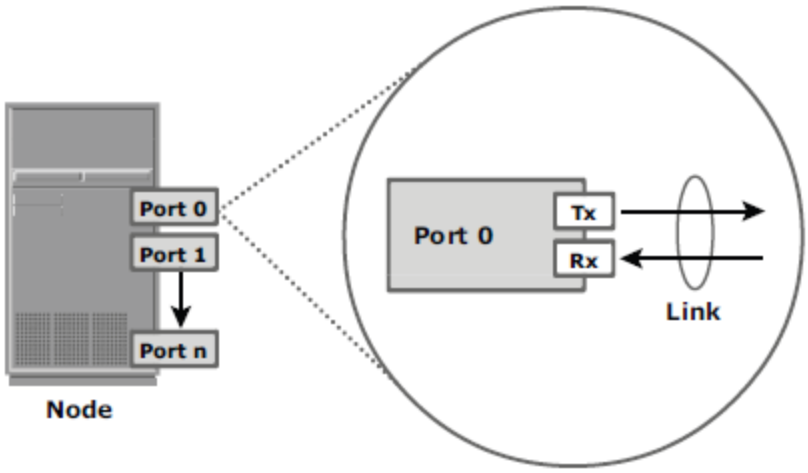
A SAN consists of three basic components: servers, network infrastructure, and storage. These components can be further broken down into the following key elements: node ports, cabling, interconnecting devices (such as FC switches or hubs), storage arrays, and SAN management software.

Node Ports

In fiber channel, devices such as hosts, storage and tape libraries are all referred to as *nodes.* Each node is a source or destination of information for one or more nodes. Each node requires one or more ports to provide a physical interface for communicating with other nodes. These ports are integral components of an HBA and the storage front-end adapters. A port operates in full-duplex data transmission mode with a *transmit (Tx)* link and a *receive (Rx)* link.



FC SAN evolution



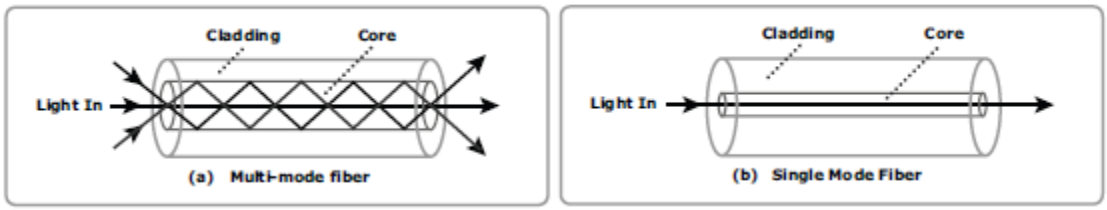
Nodes ,ports and links

Cabling

SAN implementations use optical fiber cabling. Copper can be used for shorter distances for back-end connectivity, as it provides a better signal-to-noise ratio for distances up to 30 meters. Optical fiber cables carry data in the form of light. There are two types of optical cables, multi-mode and single-mode.

Multi-mode fiber (MMF) cable carries multiple beams of light projected at different angles simultaneously onto the core of the cable. Based on the bandwidth, multi-mode fibers are classified as OM1 (62.5µm), OM2 (50µm) and laser optimized OM3 (50µm). In an MMF transmission, multiple light beams traveling inside the cable tend to disperse and collide. This collision weakens the signal strength after it travels a certain distance — a process known as *modal dispersion*. An MMF cable is usually used for distances of up to 500 meters because of signal degradation (attenuation) due to modal dispersion.

Single-mode fiber (SMF) carries a single ray of light projected at the center of the core. These cables are available in diameters of 7–11 microns; the most common size is 9 microns. In an SMF transmission, a single light beam travels in a straight line through the core of the fiber. The small core and the single light wave limits modal dispersion. Among all types of fiber cables, single-mode provides minimum signal attenuation over maximum distance (up to 10 km). A single-mode cable is used for long-distance cable runs, limited only by the power of the laser at the transmitter and sensitivity of the receiver



Multi-mode fiber and single mode fiber

MMFs are generally used within data centers for shorter distance runs, while SMFs are used for longer distances. MMF transceivers are less expensive as compared to SMF transceivers. A Standard connector (SC) (see Figure 6-5 (a)) and a Lucent connector (LC) are two commonly used connectors for fiber optic cables. An SC is used for data transmission speeds up to 1 GB/s, whereas an LC is used for speeds up to 4 GB/s. Figure 6-6 depicts a Lucent connector and a Standard connector.

A *Straight Tip (ST)* is a fiber optic connector with a plug and a socket that is locked with a half-twisted bayonet lock (see Figure 6-5 (c)). In the early days of FC deployment, fiber optic cabling predominantly used ST connectors. This connector is often used with Fiber Channel patch panels.

(a) Standard Connector

(b) Lucent connector

(c) Straight Tip Connector

SC,LC and LT connector

The Small Form-factor Pluggable (SFP) is an optical transceiver used in optical communication. The standard SFP+ transceivers support data rates up to 10 GB/s.

Interconnect Devices

Hubs, switches, and directors are the interconnect devices commonly used in SAN. *Hubs* are used as communication devices in FC-AL implementations. Hubs physically connect nodes in a logical loop or a physical star topology. All the nodes must share the bandwidth because data travels through all the connection points. Because of availability of low cost and high performance switches, hubs are no longer used in SANs.

Switches are more intelligent than hubs and directly route data from one physical port to another. Therefore, nodes do not share the bandwidth. Instead, each node has a dedicated communication path, resulting in bandwidth aggregation

Directors are larger than switches and are deployed for data center implementations. The function of directors is similar to that of FC switches, but directors have higher port count and fault tolerance capabilities.

Storage Arrays

The fundamental purpose of a SAN is to provide host access to storage resources. The large storage capacities offered by modern storage arrays have been exploited in SAN environments for storage consolidation and centralization.

SAN implementations complement the standard features of storage arrays by providing high availability and redundancy, improved performance, business continuity, and multiple host connectivity.

SAN Management Software

SAN management software manages the interfaces between hosts, interconnect devices, and storage arrays. The software provides a view of the SAN environment and enables management of various resources from one central console. It provides key management functions, including mapping of storage devices, switches, and servers, monitoring and generating alerts for discovered devices, and logical partitioning of the SAN, called *zoning*. In addition, the software provides management of typical SAN components such as HBAs, storage components, and interconnecting devices.

3.2 FC Topologies

Fabric design follows standard topologies to connect devices. Core-edge fabric is one of the popular topology designs. Variations of core-edge fabric and mesh topologies are most commonly deployed in SAN implementations.

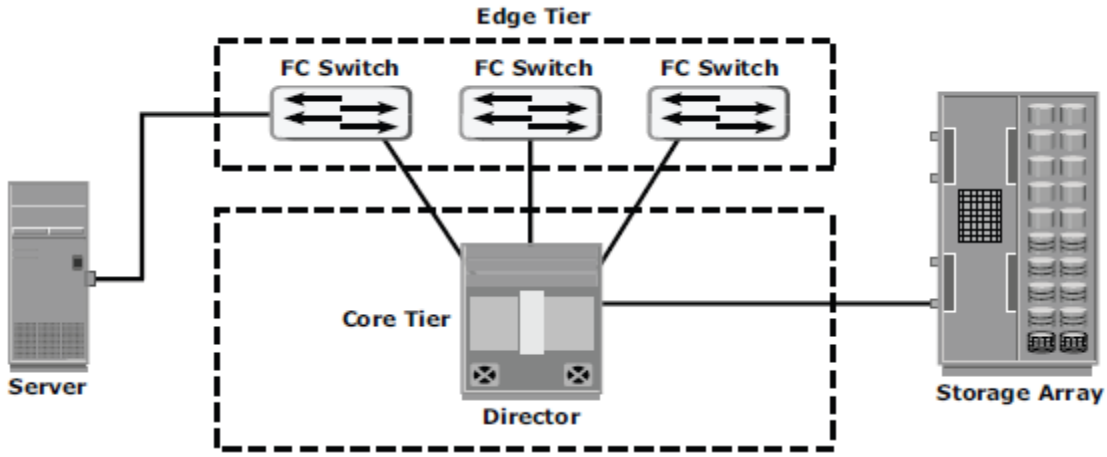
3.2.1 Core-Edge Fabric

In the *core-edge fabric* topology, there are two types of switch tiers in this fabric. The *edge tier* usually comprises switches and offers an inexpensive approach to adding more hosts in a fabric. The tier at the edge fans out from the tier at the core. The nodes on the edge can communicate with each other.

The *core tier* usually comprises enterprise directors that ensure high fabric availability. Additionally all traffic has to either traverse through or terminate at this tier. In a two-tier configuration, all storage devices are connected to the core tier, facilitating fan-out. The host-to-storage traffic has to traverse one and two ISLs in a two-tier and three-tier configuration, respectively. Hosts used for mission-critical applications can be connected directly to the core tier and consequently avoid traveling through the ISLs to process I/O requests from these hosts.

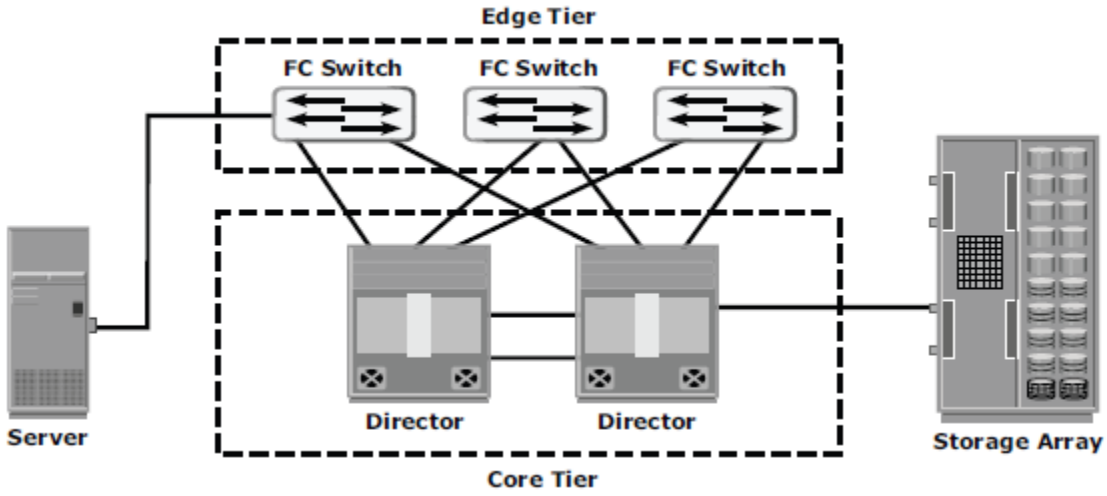
The core-edge fabric topology increases connectivity within the SAN while conserving overall port utilization. If expansion is required, an additional edge switch can be connected to the core. This topology can have different variations.

In a *single-core topology*, all hosts are connected to the edge tier and all storage is connected to the core tier. Figure 6-21 depicts the core and edge switches in a single-core topology.



Single core topology

A *dual-core topology* can be expanded to include more core switches. However, to maintain the topology, it is essential that new ISLs are created to connect each edge switch to the new core switch that is added. Figure 6-22 illustrates the core and edge switches in a dual-core topology.



Dual-core topology

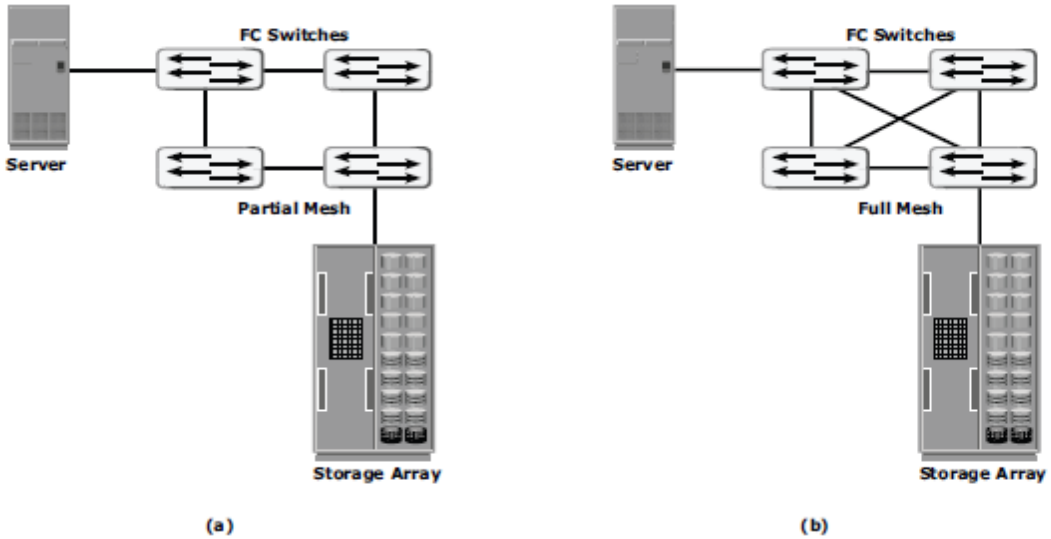
3.2.2 Mesh Topology

In a *mesh topology*, each switch is directly connected to other switches by using ISLs. This topology promotes enhanced connectivity within the SAN. When the number of ports on a network increases, the number of nodes that can participate and communicate also increases.

A mesh topology may be one of the two types: full mesh or partial mesh. In a *full mesh*, every switch is connected to every other switch in the topology. Full mesh topology may be appropriate when the number of switches involved is small. A typical deployment would involve up to four switches or directors, with each of them servicing highly localized host-to-storage traffic. In a full mesh topology, a maximum of one ISL or hop is required for host-to-storage traffic.

In a *partial mesh* topology, several hops or ISLs may be required for the traffic to reach its destination. Hosts and storage can be located anywhere in the fabric, and storage can be localized to a director or a switch in both mesh topologies.

A full mesh topology with a symmetric design results in an even number o switches, whereas a partial mesh has an asymmetric design and may result in an odd number of switches.



Partial mesh and full mesh topologies

3.3 Fiber Channel Architecture

The FC architecture represents true channel/network integration with standard interconnecting devices. Connections in a SAN are accomplished using FC. Traditionally, transmissions from host to storage devices are carried out over channel connections such as a parallel bus. Channel technologies provide high levels of performance with low protocol overheads. Such performance is due to the static nature of channels and the high level of hardware and software integration provided by the channel technologies. However, these technologies suffer from inherent limitations in terms of the number of devices that can be connected and the distance between these devices. *Fiber Channel Protocol (FCP)* is the implementation of serial SCSI-3 over an FC network. In the FCP architecture, all external and remote storage devices attached to the SAN appear as local devices to the host operating system. The key advantages of FCP are as follows:

■ Sustained transmission bandwidth over long distances.

■Support for a larger number of addressable devices over a network. Theoretically, FC can support over 15 million device addresses on a network.

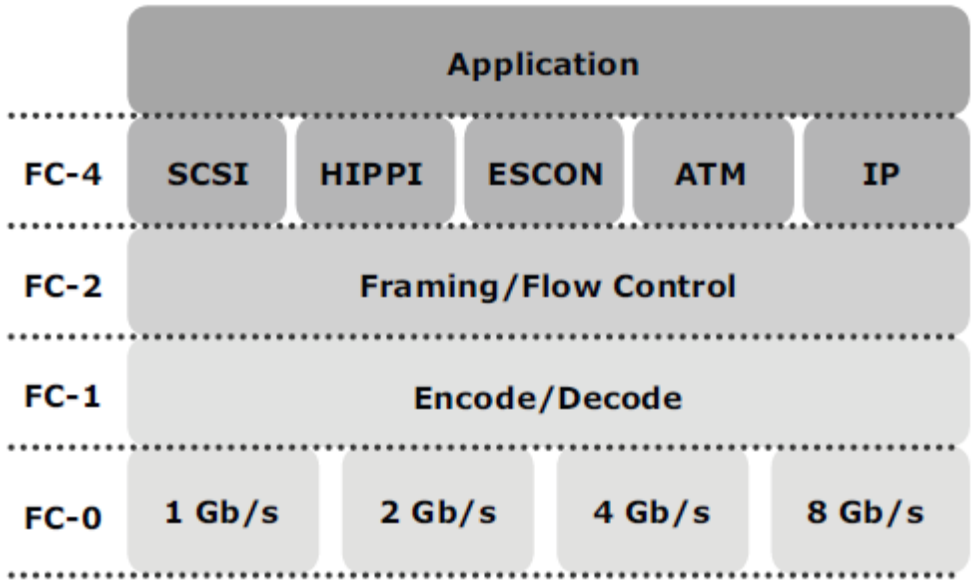
■ Exhibits the characteristics of channel transport and provides speeds up to 8.5 GB/s (8 GFC).

FCP is specified by standards produced by T10; FCP-3 is the last issued standard, and FCP-4 is under development. FCP defines a Fibre Channel mapping layer (FC-4) that uses the services defined by ANS X3.230-199X, Fiber Channel-Physical and Signaling Interface (FC-PH) to transmit SCSI commands, data, and status information between SCSI initiator and SCSI target. FCP defines Fiber Channel information units in accordance with the SCSI architecture model. FCP also defines how the Fibre Channel services are used to perform the services defined by the SCSI architecture model.

The FC standard enables mapping several existing *Upper Layer Protocols (ULPs*) to FC frames for transmission, including SCSI, IP, High Performance Parallel Interface (HIPPI), Enterprise System Connection (ESCON), and Asynchronous Transfer Mode (ATM).

3.4 Fiber Channel Protocol Stack

It is easier to understand a communication protocol by viewing it as a structure of independent layers. FCP defines the communication protocol in five layers: FC-0 through FC-4 (except FC-3 layer, which is not implemented). In a layered communication model, the peer layers on each node talk to each other through defined protocols.



FC-4 Upper Layer Protocol

FC-4 is the uppermost layer in the FCP stack. This layer defines the application interfaces and the way Upper Layer Protocols (ULPs) are mapped to the lower FC layers. The FC standard defines several protocols that can operate on the FC-4 layer (see Figure 6-7). Some of the protocols include SCSI, HIPPI Framing Protocol, Enterprise Storage Connectivity (ESCON), ATM, and IP.

FC-2 Transport Layer

The FC-2 is the transport layer that contains the payload, addresses of the source and destination ports, and link control information. The FC-2 layer provides Fiber Channel addressing, structure, and organization of data (frames, sequences, and exchanges). It also defines fabric services, classes of service, flow control, and routing.

FC-1 Transmission Protocol

This layer defines the transmission protocol that includes serial encoding and decoding rules, special characters used, and error control. At the transmitter node, an 8-bit character is encoded into a 10-bit transmissions character. This character is then transmitted to the receiver node. At the receiver node, the 10-bit character is passed to the FC-1 layer, which decodes the 10-bit character into the original 8-bit character.

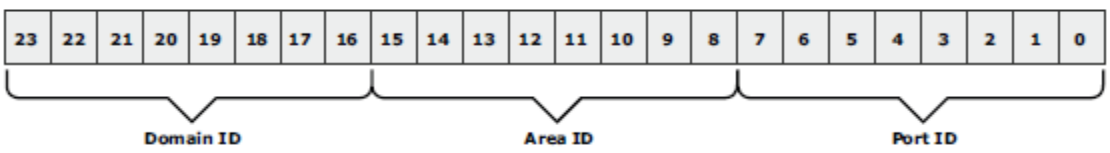
FC-0 Physical Interface

FC-0 is the lowest layer in the FCP stack. This layer defines the physical interface, media, and transmission of raw bits. The FC-0 specification includes cables, connectors, and optical and electrical parameters for a variety of data rates. The FC transmission can use both electrical and optical media.

Fiber Channel Addressing

An FC address is dynamically assigned when a port logs on to the fabric. The FC address has a distinct format that varies according to the type of node port in the fabric. These ports can be an N_port and an NL_port in a public loop, or an NL_port in a private loop.

The first field of the FC address of an N_port contains the domain ID of the switch. This is an 8-bit field. Out of the possible 256 domain IDs, 239 are available for use; the remaining 17 addresses are reserved for specific services. For example, FFFFFC is reserved for the name server, and FFFFFE is reserved for the fabric login service. The maximum possible number of N_ports in a switched fabric is calculated as 239 domains □□256 areas □□256 ports =15,663,104 Fiber Channel addresses.



The area ID is used to identify a group of F_ports. An example of a group of F_ports would be a card on the switch with more than one port on it. The last field in the FC address identifies the F_port within the group.

3.5 World Wide Names

Each device in the FC environment is assigned a 64-bit unique identifier called the *World Wide Name* (WWN). The Fiber Channel environment uses two types of WWNs: World Wide Node Name (WWNN) and World Wide Port Name (WWPN). Unlike an FC address, which is assigned dynamically, a WWN is a static name for each device on an FC network. WWNs are similar to the Media Access Control (MAC) addresses used in IP networking. WWNs are *burned* into the hardware or assigned through software. Several configuration definitions in a SAN use WWN for identifying storage devices and HBAs. The name server in an FC environment keeps the association of WWNs to the dynamically created FC addresses for nodes.

3.6 Structure and Organization of FC Data

In an FC network, data transport is analogous to a conversation between two people, whereby a frame represents a word, a sequence represents a sentence, and an exchange represents a conversation.

■ **Exchange operation:** An exchange operation enables two N_ports to identify and manage a set of information units. This unit maps to a sequence.

Sequences can be both unidirectional and bidirectional depending upon the type of data sequence exchanged between the initiator and the target.

■ **Sequence:** A sequence refers to a contiguous set of frames that are sent from one port to another. A sequence corresponds to an information unit, as defined by the ULP.

■ **Frame:** A frame is the fundamental unit of data transfer at Layer 2. Each frame can contain up to 2,112 bytes of payload.

3.6.1 Flow Control

Flow control defines the pace of the flow of data frames during data transmission. FC technology uses two flow-control mechanisms: buffer-to-buffer credit (BB_Credit) and end-to-end credit (EE_Credit).

BB_Credit

FC uses the *BB_Credit* mechanism for hardware-based flow control. BB_Credit controls the maximum number of frames that can be present over the link at any given point in time. In a switched fabric, BB_Credit management may take place between any two FC ports. The transmitting port maintains a count of free receiver buffers and continues to send frames if the count is greater than 0. The BB_Credit mechanism provides frame acknowledgment through the *Receiver Ready (R_RDY)* primitive.

EE_Credit

The function of end-to-end credit, known as EE_Credit, is similar to that of BB_ Credit. When an initiator and a target establish themselves as nodes communicating with each other, they exchange the EE_Credit parameters (part of Port Login). The EE_Credit mechanism affects the flow control for class 1 and class 2 traffic only.

3.6.2 Classes of Service

The FC standards define different classes of service to meet the requirements of a wide range of applications. The table below shows three classes of services

and their features (Table 6-1).

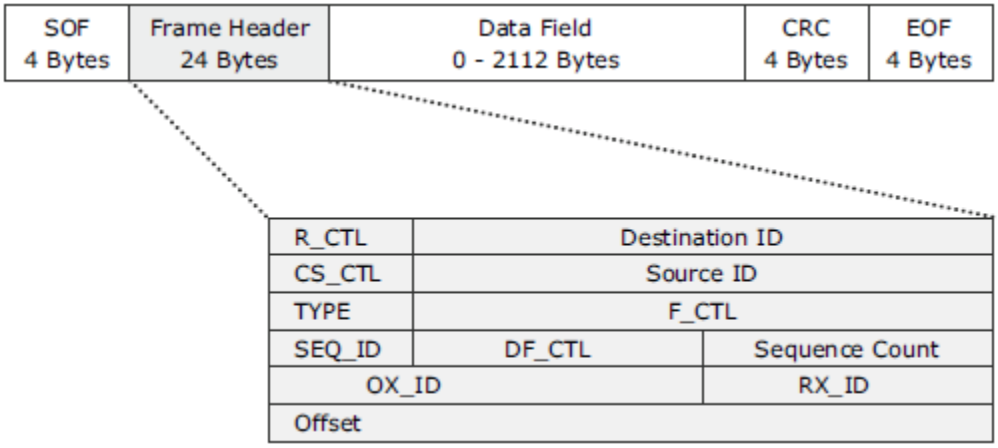
Table 6-1: FC Class of Services

	CLASS 1	CLASS 2	CLASS 3
Communication type	Dedicated connection	Nondedicated connection	Nondedicated connection
Flow control	End-to-end credit	End-to-end credit B-to-B credit	B-to-B credit
Frame delivery	In order delivery	Order not guaranteed	Order not guaranteed
Frame acknowl-edgement	Acknowledged	Acknowledged	Not acknowledged
Multiplexing	No	Yes	Yes
Bandwidth utilization	Poor	Moderate	High

Another class of services is *class F*, which is intended for use by the switches communicating through ISLs. Class F is similar to Class 2, and it provides notification of non delivery of frames. Other defined Classes 4, 5, and 6 are used for specific applications. Currently, these services are not in common use.

3.7 FC Frame

An FC frame consists of five parts: *start of frame (SOF)*, *frame header*, *data field*, *cyclic redundancy check (CRC)*, and *end of frame (EOF)*. The SOF and EOF act as delimiters. In addition to this role, the SOF is a flag that indicates whether the frame is the first frame in a sequence of frames. The frame header is 24 bytes long and contains addressing information for the frame. It includes the following information: Source ID (S_ID), Destination ID (D_ID), Sequence ID (SEQ_ID), Sequence Count (SEQ_CNT), Originating
Exchange ID (OX_ID), and Responder Exchange ID (RX_ID), in addition to some control fields.



FC frame

The S_ID and D_ID are standard FC addresses for the source port and the destination port, respectively. The SEQ_ID and OX_ID identify the frame as a component of a specific sequence and exchange, respectively. The frame header also defines the following fields:

■ **Routing Control (R_CTL)**: This field denotes whether the frame is a link control frame or a data frame. Link control frames are non data frames that do not carry any payload. These frames are used for setup and messaging. In contrast, data frames carry the payload and are used for data transmission.

■ **Class Specific Control (CS_CTL)**: This field specifies link speeds for class 1 and class 4 data transmission.

■ **TYPE**: This field describes the upper layer protocol (ULP) to be carried on the frame if it is a data frame. However, if it is a link control frame, this field is used to signal an event such as "fabric busy." For example, if the TYPE is 08, and the frame is a data frame, it means that the SCSI will be carried on an FC.

■**Data Field Control (DF_CTL)**: A 1-byte field that indicates the existence of any optional headers at the beginning of the data payload. It is a mechanism to extend header information into the payload.

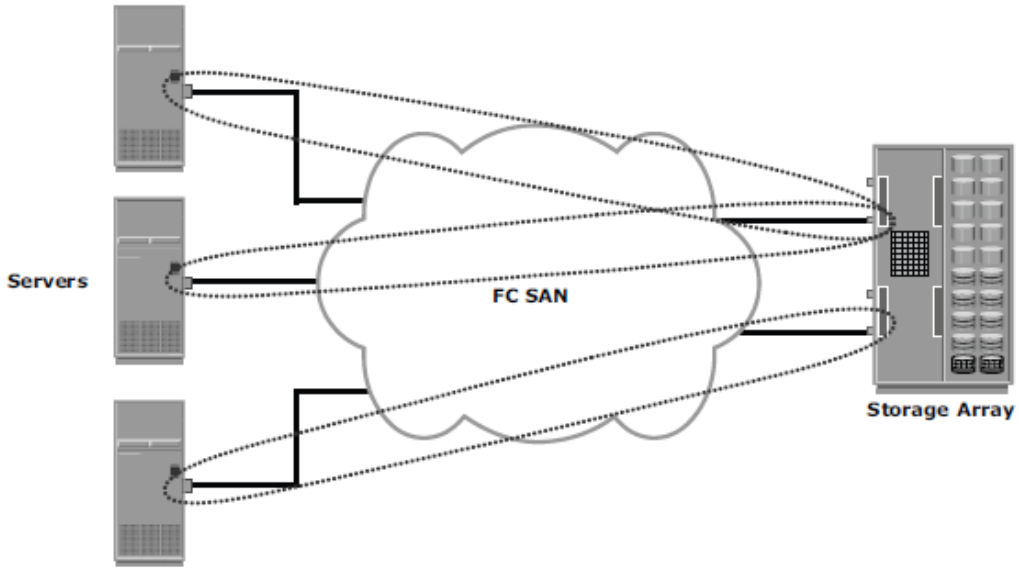
■ **Frame Control (F_CTL)**: A 3-byte field that contains control information related to frame content. For example, one of the bits in this field indicates whether this is the first sequence of the exchange.

The data field in an FC frame contains the data payload, up to 2,112 bytes of original data — in most cases, SCSI data. The biggest possible payload an FC frame can deliver is 2,112 bytes of data with 36 bytes of fixed overhead. A link control frame, by definition, has a payload of 0 bytes. Only data frames carry a payload.

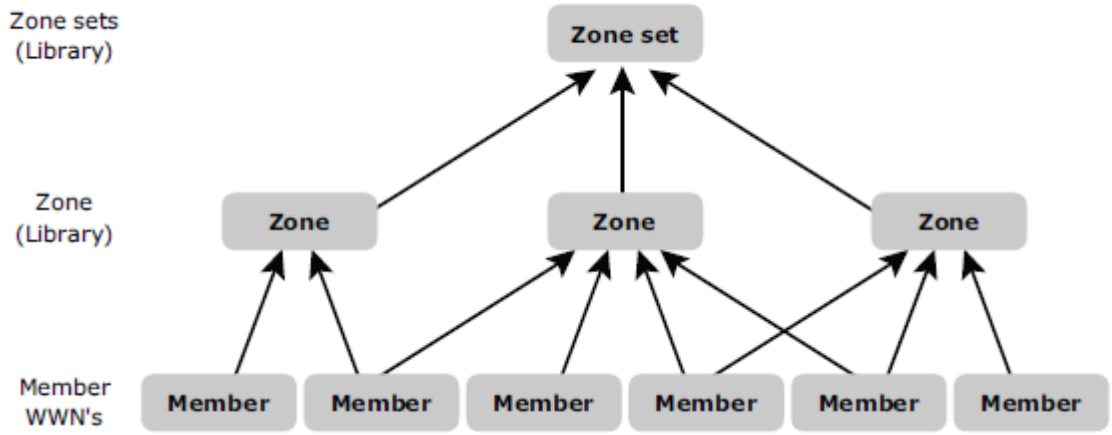
The CRC checksum facilitates error detection for the content of the frame. This checksum verifies data integrity by checking whether the content of the frames was received correctly. The CRC checksum is calculated by the sender before encoding at the FC-1 layer. Similarly, it is calculated by the receiver after decoding at the FC-1 layer.

3.8 Zoning

Zoning is an FC switch function that enables nodes within the fabric to be logically segmented into groups that can communicate with each other. When a device (host or storage array) logs onto a fabric, it is registered with the name server. When a port logs onto the fabric, it goes through a device discovery process with other devices registered in the name server. The zoning function controls this process by allowing only the members in the same zone to establish these link-level services.



Multiple zone sets may be defined in a fabric, but only one zone set can be active at a time. A zone set is a set of zones and a zone is a set of members. A member may be in multiple zones. Members, zones, and zone sets form the hierarchy defined in the zoning process. *Members* are nodes within the SAN that can be included in a zone. *Zones* comprise a set of members that have access to one another. A port or a node can be a member of multiple zones. *Zone sets* comprise group of zones that can be activated or deactivated as a single entity in a fabric. Only one zone set per fabric can be active at a time. Zone sets are also referred to as *zone configurations*.



3.8.1 Types of Zoning

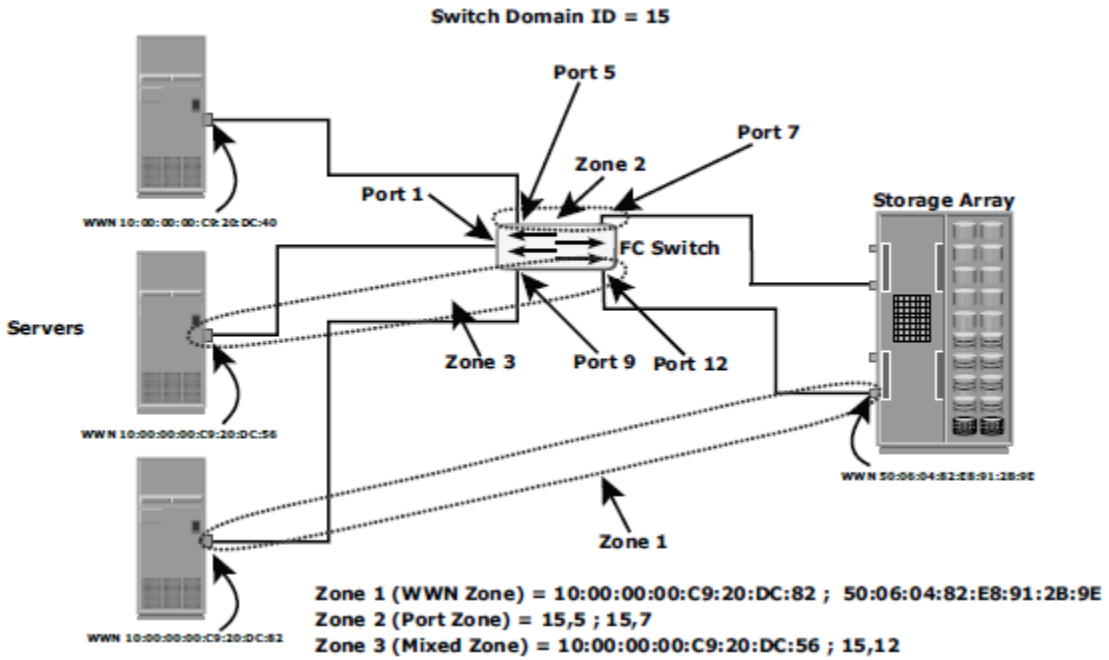
Zoning can be categorized into three types:

■ **Port zoning:** It uses the FC addresses of the physical ports to define zones.

In port zoning, access to data is determined by the physical switch port to which a node is connected. The FC address is dynamically assigned when the port logs on to the fabric. Therefore, any change in the fabric configuration affects zoning. Port zoning is also called *hard zoning*. Although this method is secure, it requires updating of zoning configuration information in the event of fabric reconfiguration.

■**WWN zoning:** It uses World Wide Names to define zones. WWN zoning is also referred to as *soft zoning*. A major advantage of WWN zoning is its flexibility. It allows the SAN to be recabled without reconfiguring the zone information. This is possible because the WWN is static to the node port.

■**Mixed zoning:** It combines the qualities of both WWN zoning and port zoning. Using mixed zoning enables a specific port to be tied to the WWN of a node.



Zoning is used in conjunction with LUN masking for controlling server access to storage. However, these are two different activities. Zoning takes place at the fabric level and LUN masking is done at the array level.

3.9 Network- Attached Storage

Network-attached storage (NAS) is an IP-based file-sharing device attached to a local area network. NAS provides the advantages of server consolidation by eliminating the need for multiple file servers. It provides storage consolidation through file-level data access and sharing. NAS is a preferred storage solution that enables clients to share files quickly and directly with minimum storage management overhead. NAS also helps to eliminate bottlenecks that users face when accessing files from a general-purpose server. NAS uses network and file-sharing protocols to perform filing and storage functions. These protocols include TCP/IP for data transfer and CIFS and NFS for remote file service.

NAS enables both UNIX and Microsoft Windows users to share the same data seamlessly. To enable data sharing, NAS typically uses NFS for UNIX, CIFS for Windows, and File Transfer Protocol (FTP) and other protocols for both environments. Recent advancements in networking technology have enabled NAS to scale up to enterprise requirements for improved performance and reliability in accessing data. A NAS device is a dedicated, high-performance, high-speed, single-purpose file serving and storage system. NAS serves a mix of clients and servers over an IP network. Most NAS devices support multiple interfaces and networks.

A NAS device uses its own operating system and integrated hardware, software components to meet specific file service needs. Its operating system is optimized for file I/O and, therefore, performs file I/O better than a general purpose server. As a result, a NAS device can serve more clients than traditional file servers, providing the benefit of server consolidation.

3.9.1Benefits of NAS

NAS offers the following benefits:

■ **Supports comprehensive access to information:** Enables efficient file sharing and supports many-to-one and one-to-many configurations. The many-to-one configuration enables a NAS

device to serve many clients simultaneously. The one-to-many configuration enables one client to connect with many NAS devices simultaneously.

■ **Improved efficiency:** Eliminates bottlenecks that occur during file access from a general purpose file server because NAS uses an operating system specialized for file serving. It improves the utilization of general-purpose servers by relieving them of file-server operations.

■ **Improved flexibility:** Compatible for clients on both UNIX and Windows platforms using industry-standard protocols. NAS is flexible and can serve requests from different types of clients from the same source.

■ **Centralized storage:** Centralizes data storage to minimize data duplication on client workstations, simplify data management, and ensures greater data protection.

■ **Simplified management:** Provides a centralized console that makes it possible to manage file systems efficiently.

■ **Scalability:** Scales well in accordance with different utilization profiles and types of business applications because of the high performance and low-latency design.

■**High availability:** Offers efficient replication and recovery options, enabling high data availability. NAS uses redundant networking components that provide maximum connectivity options. A NAS device can use clustering technology for failover.

■ **Security:** Ensures security, user authentication, and file locking in conjunction with industry-standard security schemas.

3.9.2 Components of NAS

A NAS device has the following components

■NAS head (CPU and Memory)

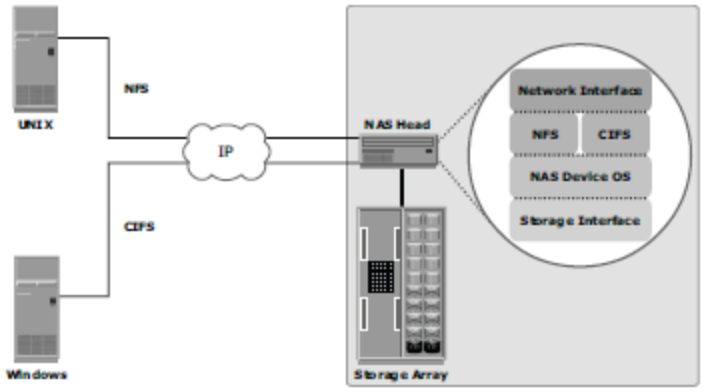
■One or more network interface cards (NICs), which provide connectivity to the network. Examples of NICs include Gigabit Ethernet, Fast Ethernet,
ATM, and Fiber Distributed Data Interface (FDDI).

■An optimized operating system for managing NAS functionality

■NFS and CIFS protocols for file sharing

■ Industry-standard storage protocols to connect and manage physical disk resources, such as ATA, SCSI, or FC.

The NAS environment includes clients accessing a NAS device over an IP network using standard protocols.



3.9.3 NAS Implementations

There are two types of NAS implementations: integrated and gateway. The integrated NAS device has all of its components and storage system in a single enclosure. In gateway Implementation, NAS head shares its storage with SAN environment.

Integrated NAS

An integrated NAS device has all the components of NAS, such as the NAS head and storage, in a single enclosure, or frame. This makes the integrated NAS a self-contained environment. The NAS head connects to the IP network to provide connectivity to the clients and service the file I/O requests. The storage consists of a number of disks that can range from low-cost ATA to high throughput FC disk drives. Management software manages the NAS head and storage configurations. An integrated NAS solution ranges from a low-end device, which is a single enclosure, to a high-end solution that can have an externally connected storage array.

A low-end appliance-type NAS solution is suitable for applications that a small department may use, where the primary need is consolidation of storage, rather than high performance or advanced features such as disaster recovery and business continuity. This solution is fixed in capacity and might not be upgradable beyond its original configuration. To expand the capacity, the solution must be scaled by deploying additional units, a task that increases management overhead because multiple devices have to be administered. In a high-end NAS solution, external and dedicated storage can be used. This enables independent scaling of the capacity in terms of NAS heads or storage. However, there is a limit to scalability of this solution.

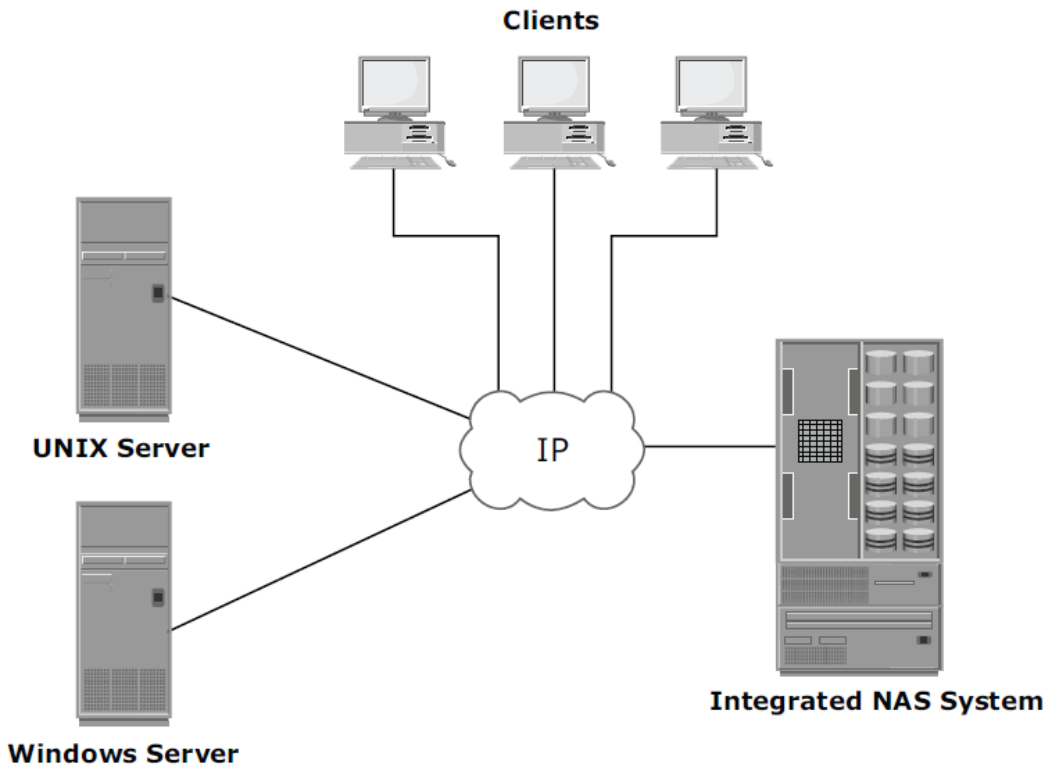
Gateway NAS

A gateway NAS device consists of an independent NAS head and one or more storage arrays. The NAS head performs the same functions that it does in the integrated solution; while the storage is shared with other applications that require block-level I/O. Management functions in this type of solution are more complex than those in an integrated environment because there are separate administrative tasks for the NAS head and the storage. In addition to the components that are explicitly tied to the NAS solution, a gateway solution can also utilize the FC infrastructure, such as switches, directors, or direct-attached storage arrays.

The gateway NAS is the most scalable because NAS heads and storage arrays can be I dependently scaled up when required. Adding processing capacity to the NAS gateway is an example of scaling. When the storage limit is reached, it can scale up, adding capacity on the SAN independently of the NAS head. Administrators can increase performance and I/O processing capabilities for their environments without purchasing additional interconnect devices and storage. Gateway NAS enables high utilization of storage capacity by sharing it with SAN environment.

Integrated NAS Connectivity

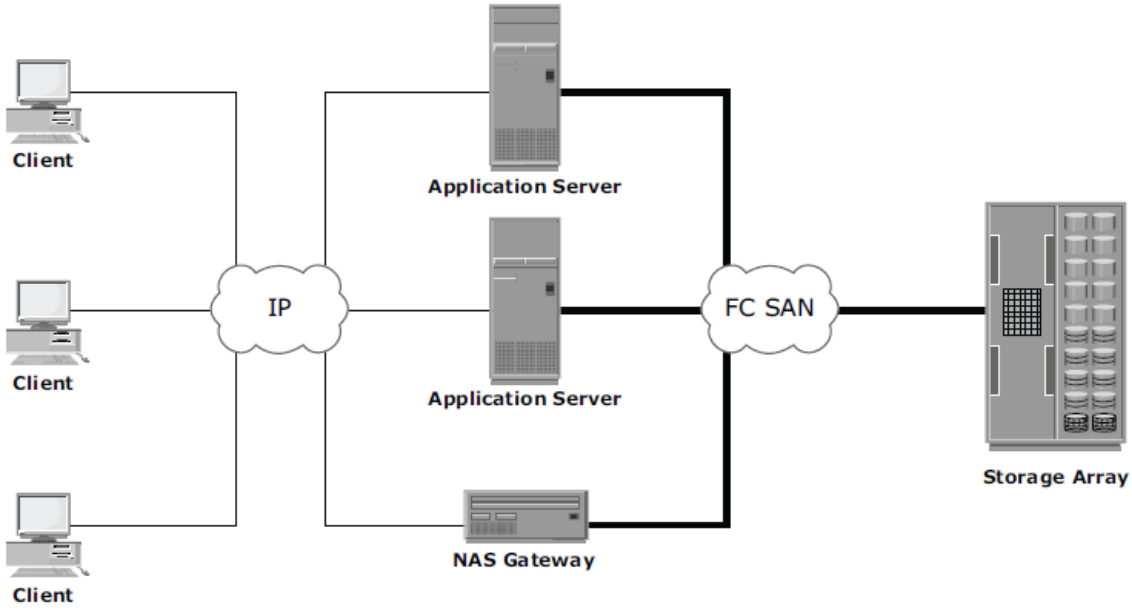
An integrated solution is self-contained and can connect into a standard IP network. Although the specifics of how devices are connected within a NAS implementation vary by vendor and model. In some cases, storage is embedded within a NAS device and is connected to the NAS head through internal connections, such as ATA or SCSI controllers. In others, the storage may be external but connected by using SCSI controllers. In a high-end integrated NAS model, external storage can be directly connected by FC HBAs or by dedicated FC switches. In the case of a low-end integrated NAS model, backup traffic is shared on the same public IP network along with the regular client access traffic. In the case of a high-end integrated NAS model, an isolated backup network can be used to segment the traffic from impeding client access. More complex solutions may include an intelligent storage subsystem, enabling faster backup and larger capacities while simultaneously enhancing performance.



Gateway NAS Connectivity

In a gateway solution, front-end connectivity is similar to that in an integrated solution. An integrated environment has a fixed number of NAS heads, making it relatively easy to determine IP networking requirements. In contrast, networking requirements in a gateway environment are complex to determine due to scalability options. Adding more NAS heads may require additional networking connectivity and bandwidth.

Communication between the NAS gateway and the storage system in a gateway solution is achieved through a traditional FC SAN. To deploy a stable NAS solution, factors such as multiple paths for data, redundant fabrics, and load distribution must be considered.



Implementation of a NAS gateway solution requires analysis of current SAN environment. This analysis is required to determine the feasibility of introducing a NAS workload to the existing SAN. Analyze the SAN to determine whether the workload is primarily read or write, or random or sequential. Determine the predominant I/O size in use. In general, sequential workloads have large I/Os. Typically, NAS workloads are random with small I/O size. Introducing sequential workload with random workloads can be disruptive to the sequential workload. Therefore, it is recommended to separate the NAS and SAN disks. Also, determine whether the NAS workload performs adequately with the configured cache in the storage subsystem.

3.9.4 NAS File-Sharing Protocols

Most NAS devices support multiple file service protocols to handle file I/O requests to a remote file system. As mentioned earlier, NFS and CIFS are the common protocols for file sharing. NFS is predominantly used in UNIX-based operating environments; CIFS is used in Microsoft Windows–based operating environments.

These file sharing protocols enable users to share file data across different operating environments and provide a means for users to migrate transparently from one operating system to another.

NFS

NFS is a client/server protocol for file sharing that is most commonly used on UNIX systems. NFS was originally based on the connectionless User Datagram Protocol (UDP). It uses a machine-independent model to represent user data. It also uses Remote Procedure Call (RPC) as a method of inter process communication between two computers. The NFS protocol provides a set of RPCs to access a remote file system for the following operations:

- Searching files and directories
- Opening, reading, writing to, and closing a file
- Changing file attributes
- Modifying file links and directories

NFS uses the mount protocol to create a connection between the client and the remote system to transfer data. NFS (NFSv3 and earlier) is a *stateless* protocol, which means that it does not maintain any kind of table to store information about open files and associated pointers. Therefore, each call provides a full set of arguments to access files on the server. These arguments include a file name and a location, a particular position to read or write, and the versions of NFS. Currently, three versions of NFS are in use:

- **NFS version 2 (NFSv2):** Uses UDP to provide a stateless network connection between a client and a server. Features such as locking are handled outside the protocol.
- **NFS version 3 (NFSv3):** The most commonly used version, it uses UDP or TCP, and is based on the stateless protocol design. It includes some new features, such as a 64-bit file size, asynchronous writes, and additional file attributes to reduce re-fetching.
- **NFS version 4 (NFSv4):** This version uses TCP and is based on a stateful protocol design. It offers enhanced security.

CIFS

CIFS is a client/server application protocol that enables client programs to make requests for files and services on remote computers over TCP/IP. It is a public, or open, variation of Server Message Block (SMB) protocol. The CIFS protocol enables remote clients to gain access to files that are on a server. CIFS enables file sharing with other clients by using special locks. File names in CIFS are encoded using unicode characters. CIFS provides the following features to ensure data integrity:

- It uses file and record locking to prevent users from overwriting the work of another user on a file or a record.

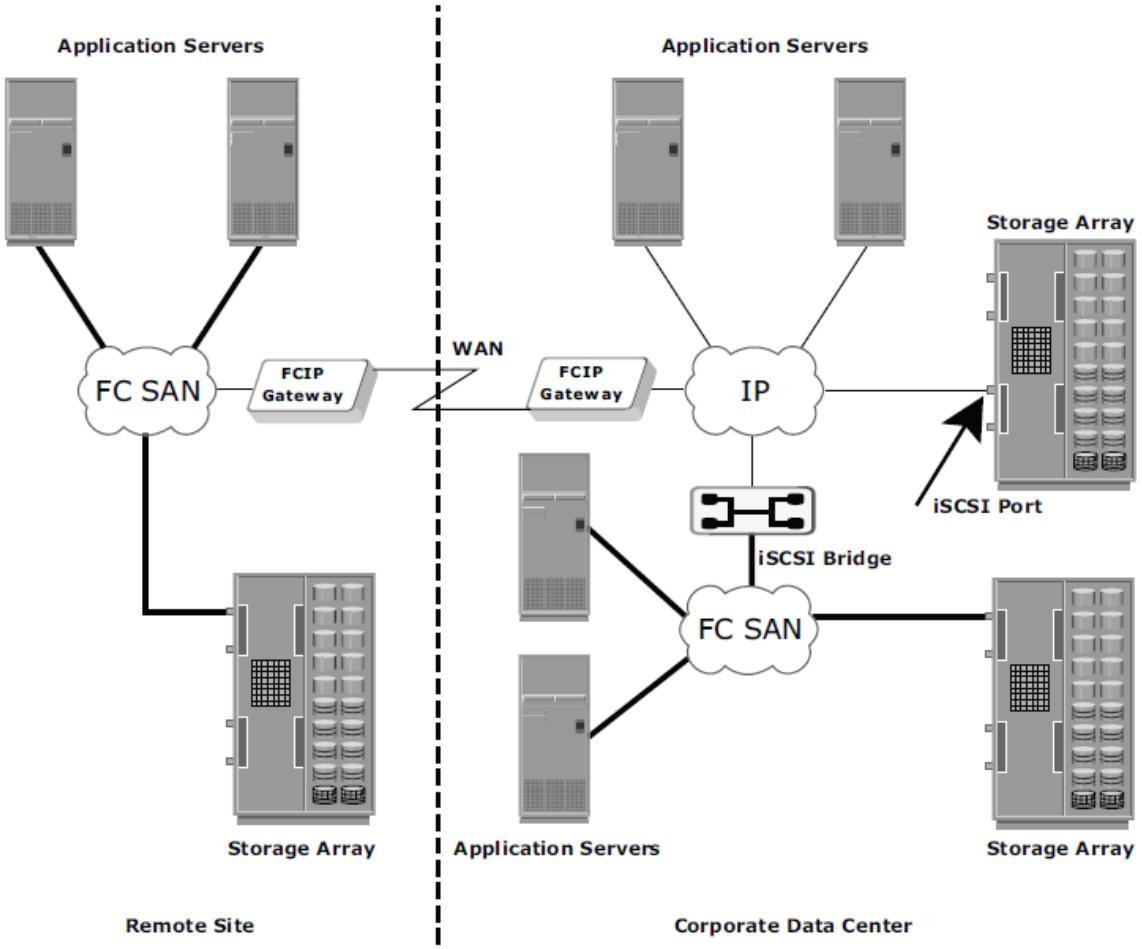
■It runs over TCP.
■It supports fault tolerance and can automatically restore connections and reopen files that were open prior to interruption. The fault tolerance features of CIFS depend on whether an application is written to take advantage of these features. Moreover, CIFS is a stateful protocol because the CIFS server maintains connection information regarding every connected client. In the event of a network failure or CIFS server failure, the client receives a disconnection notification. User disruption is minimized if the application has the embedded intelligence to restore the connection. However, if the embedded intelligence is missing, the user has to take steps to reestablish the CIFS connection.

3.10 IPSAN

Traditional SAN environments allow block I/O over Fibre Channel, whereas NAS environments allow file I/O over IP-based networks. Organizations need the performance and scalability of SAN plus the ease of use and lower TCO of NAS solutions. The emergence of IP technology that supports block I/O over IP has positioned IP for storage solutions. IP offers easier management and better interoperability. When block I/O is run over IP, the existing network infrastructure can be leveraged, which is more economical than investing in new SAN hardware and software. Many long-distance, disaster recovery (DR) solutions are already leveraging IP-based networks. In addition, many robust and mature security options are now available for IP networks. With the advent of block storage technology that leverages IP networks (the result is often referred to as IP SAN), organizations can extend the geographical reach of their storage infrastructure.

IP SAN technologies can be used in a variety of situations. Disaster recovery solutions can also be implemented using both of these technologies.

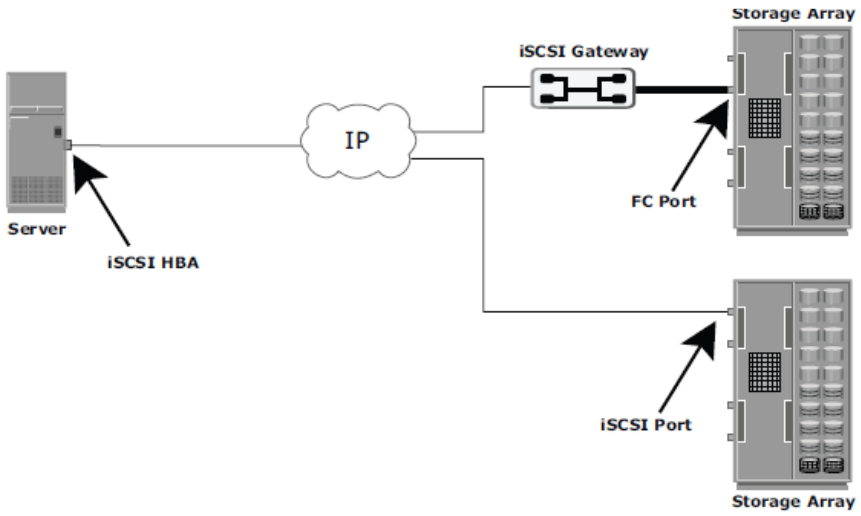
Two primary protocols that leverage IP as the transport mechanism are iSCSI and Fibre Channel over IP (FCIP).



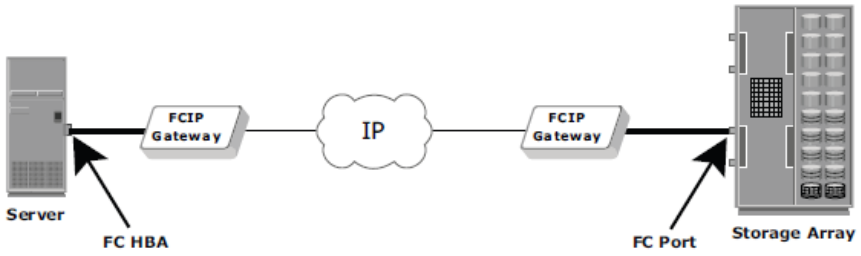
iSCSI is the host-based encapsulation of SCSI I/O over IP using an Ethernet NIC card or an iSCSI HBA in the host. As illustrated in Figure 8-2 (a), IP traffic is routed over a network either to a gateway device that extracts the SCSI I/O from the IP packets or to an iSCSI storage array. The gateway can then send the SCSI I/O to an FC-based external storage array, whereas an iSCSI storage array can handle the extraction and I/O natively.

FCIP uses a pair of bridges (FCIP gateways) communicating over TCP/IP as the transport protocol. FCIP is used to extend FC networks over distances and/or an existing IP-based infrastructure.

Today, iSCSI is widely adopted for connecting servers to storage because it is relatively inexpensive and easy to implement, especially in environments where an FC SAN does not exist. FCIP is extensively used in disaster-recovery implementations, where data is duplicated on disk or tape to an alternate site.



(a) iSCSI Implementation



3.10.1iSCSI

iSCSI is an IP-based protocol that establishes and manages connections between storage, hosts, and bridging devices over IP. iSCSI carries block-level data over IP-based networks, including Ethernet networks and the Internet. iSCSI is built on the SCSI protocol by encapsulating SCSI commands and data in order to allow these encapsulated commands and data blocks to be transported using TCP/IP packets.

Components of iSCSI

Host (initiators), targets, and an IP-based network are the principal iSCSI components. The simplest iSCSI implementation does not require any FC components. If an iSCSI-capable storage array is deployed, a host itself can act as an iSCSI initiator, and directly communicate with the storage over an IP network. However, in complex implementations that use an existing FC array for iSCSI connectivity, iSCSI gateways or routers are used to connect the existing FC SAN. These devices perform protocol translation from IP packets to FC packets and vice-versa, thereby bridging connectivity between the IP and FC environments.

iSCSI Host Connectivity

iSCSI host connectivity requires a hardware component, such as a NIC with a software component (iSCSI initiator) or an iSCSI HBA. In order to use the iSCSI protocol, a software initiator or a translator must be installed to route the SCSI commands to the TCP/IP stack.

A standard NIC, a TCP/IP offload engine (TOE) NIC card, and an iSCSI HBA are the three physical iSCSI connectivity options. A standard NIC is the simplest and least expensive connectivity option. It is easy to implement because most servers come with at least one, and in many cases two, embedded NICs. It requires only a software initiator for iSCSI functionality. However, the NIC provides no external processing power, which places additional overhead on the host CPU because it is required to perform all the TCP/IP and iSCSI processing. If a standard NIC is

used in heavy I/O load situations, the host CPU may become a bottleneck. *TOE NIC* help alleviate this burden. A TOE NIC offloads the TCP management functions from the host and leaves iSCSI functionality to the host processor. The host passes the iSCSI information to the TOE card and the TOE card sends the information to the destination using TCP/IP. Although this solution improves performance, the iSCSI functionality is still handled by a software initiator, requiring host CPU cycles. An *iSCSI HBA* is capable of providing performance benefits, as it offloads the entire iSCSI and TCP/IP protocol stack from the host processor. Use of an iSCSI HBA is also the simplest way for implementing a boot from SAN environment via iSCSI. If there is no iSCSI HBA, modifications have to be made to the basic operating system to boot a host from the storage devices because the NIC needs to obtain an IP address before the operating system loads. The functionality of an iSCSI HBA is very similar to the functionality of an FC HBA, but it is the most expensive option.

A fault-tolerant host connectivity solution can be implemented using host based multipathing software (e.g., EMC PowerPath) regardless of the type of physical connectivity. Multiple NICs can also be combined via link aggregation technologies to provide failover or load balancing. Complex solutions may also include the use of vendor-specific storage-array software that enables the iSCSI host to connect to multiple ports on the array with multiple NICs or HBAs.

Topologies for iSCSI Connectivity

The topologies used to implement iSCSI can be categorized into two classes: native and bridged.

Native topologies do not have any FC components; they perform all communication over IP. The initiators may be either directly attached to targets or connected using standard IP routers and switches.

Bridged topologies enable the co-existence of FC with IP by providing iSCSI-to-FC bridging functionality. For example, the initiators can exist in an IP environment while the storage remains in an FC SAN.

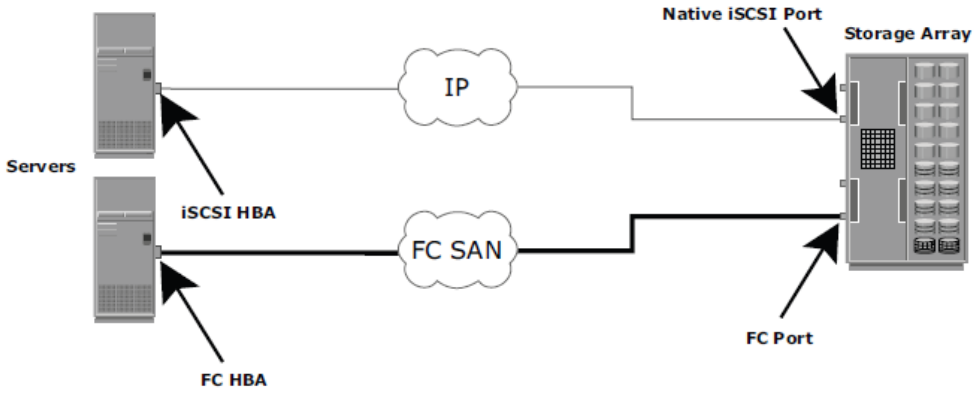
Native iSCSI Connectivity

If an iSCSI-enabled array is deployed, FC components are not needed for iSCSI connectivity in the native topology. The array has one or more Ethernet NICs that are connected to a standard Ethernet switch and configured with an IP address and listening port. Once a client/ initiator is configured with the appropriate target information, it connects to the array and requests a list of available LUNs. A single array port can service multiple hosts or initiators as long as the array can handle the amount of storage traffic that the hosts generate. Many arrays provide more than one interface so that they can be configured in a highly available design or have multiple targets configured on the initiator. Some NAS devices are also capable of functioning as iSCSI targets, enabling file-level and block-level access to centralized storage. This offers additional storage options for environments with integrated NAS devices or environments that don’t have an iSCSI/FC bridge.

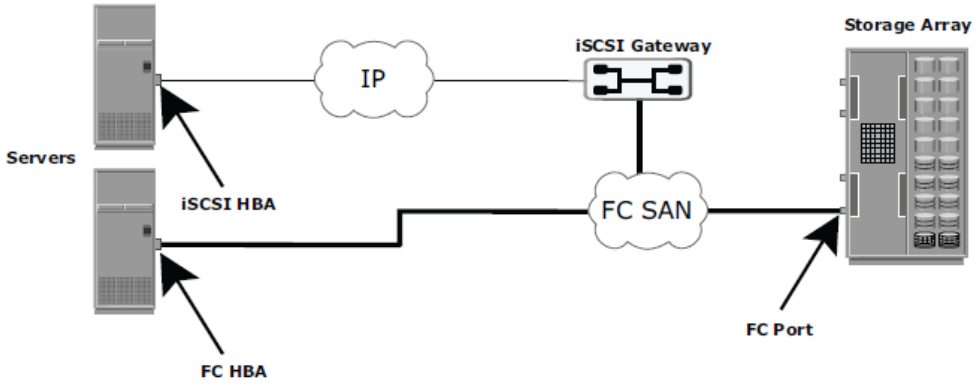
Bridged iSCSI Connectivity

A bridged iSCSI implementation includes FC components in its configuration. The array does not have any native iSCSI capabilities—that is, it does not have any Ethernet ports. Therefore, an external device, called a bridge, router, gateway, or a multi-protocol router, must be used to bridge the communication from the IP network to the FC SAN. These devices can be a standalone unit, or in many cases are integrated with an existing FC switch. In this configuration, the

bridge device has Ethernet ports connected to the IP network, and FC ports connected to the storage. These ports are assigned IP addresses, similar to the ports on an iSCSI-enabled array. The iSCSI initiator/host is configured with the bridge's IP address as its target destination. The bridge is also configured with an FC initiator or multiple initiators. These are called *virtual initiators* because there is no physical device, such as an HBA, to generate the initiator record.



(a) Native iSCSI Connectivity



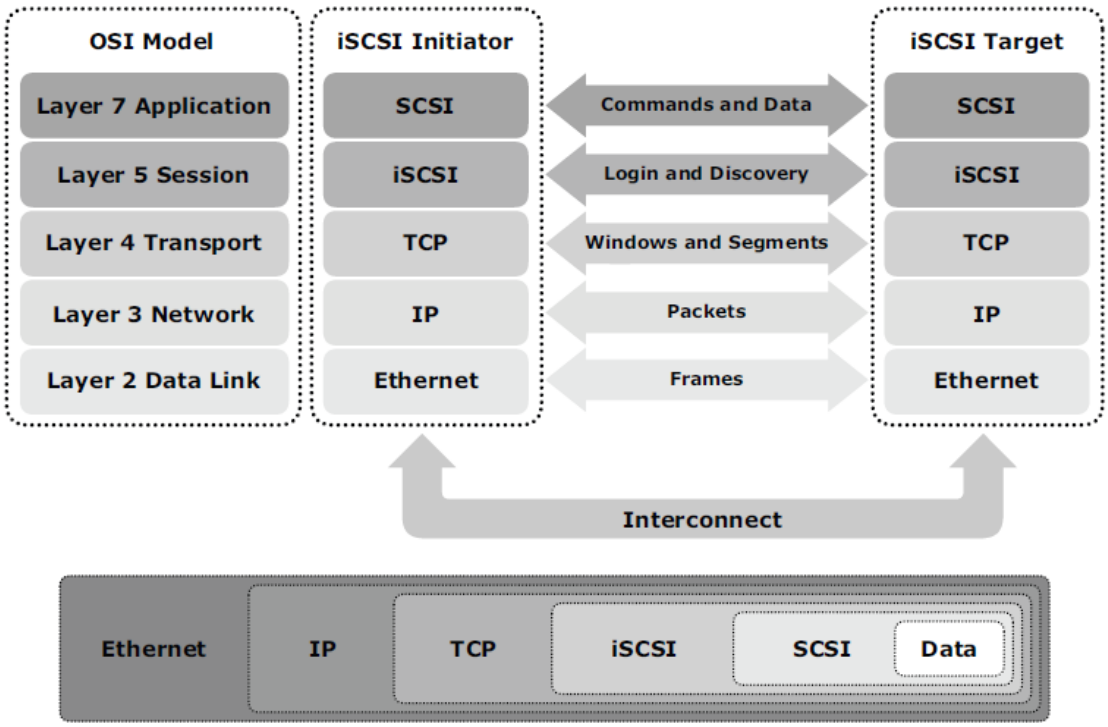
(b) Bridged iSCSI Connectivity

Combining FCP and Native iSCSI Connectivity

A combination topology can also be implemented. In this case, a storage array capable of connecting the FC and iSCSI hosts without the need for external bridging devices is needed. These solutions reduce complexity, as they remove the need for configuring bridges. However, additional processing requirements are placed on the storage array because it has to accommodate the iSCSI traffic along with the standard FC traffic.

iSCSI Protocol Stack

The architecture of iSCSI is based on the client/server model.



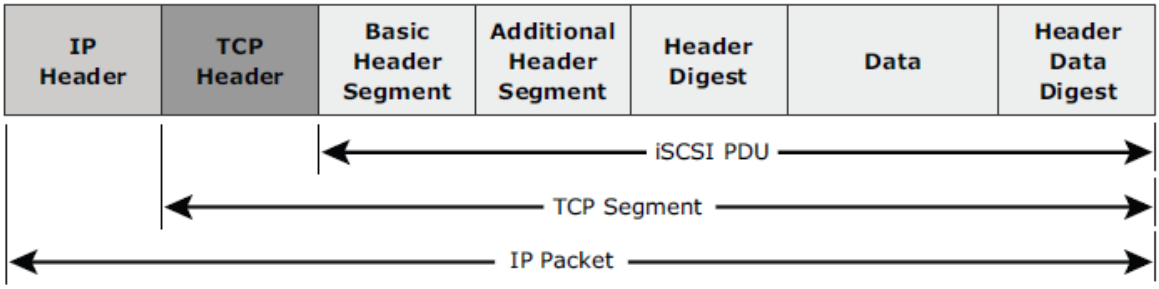
SCSI is the command protocol that works at the application layer of the OSI model. The initiators and targets use SCSI commands and responses to talk to each other. The SCSI command descriptor blocks, data, and status messages are encapsulated into TCP/IP and transmitted across the network between initiators and targets. iSCSI is the session-layer protocol that initiates a reliable session between a device that recognizes SCSI commands and TCP/IP. The iSCSI session-layer interface is responsible for handling login, authentication, target discovery, and session management. TCP is used with iSCSI at the transport layer to provide reliable service.

TCP is used to control message flow, windowing, error recovery, and retransmission. It relies upon the network layer of the OSI model to provide global addressing and connectivity. The layer-2 protocols at the data link layer of this model enable node-to-node communication for each hop through a separate physical network.

iSCSI PDU

iSCSI initiators and targets communicate using iSCSI Protocol Data Units (PDUs). All iSCSI PDUs contain one or more header segments followed by zero or more data segments. The PDU is then encapsulated into an IP packet to facilitate the transport.

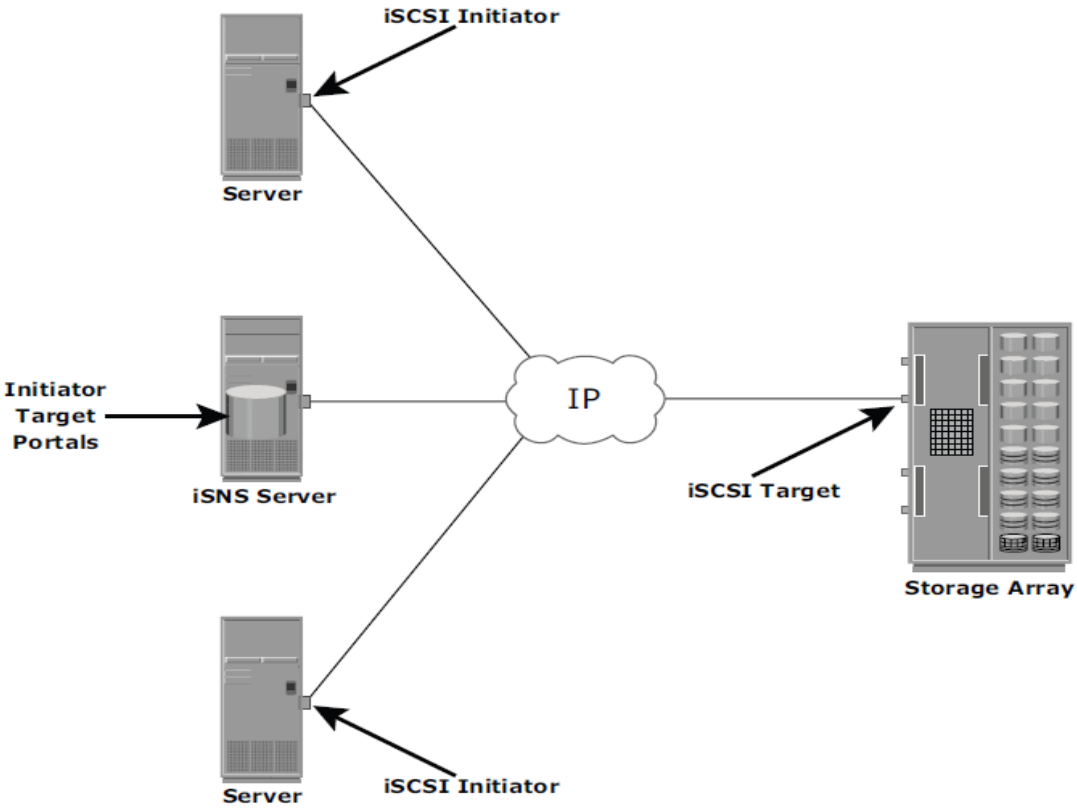
The IP header provides packet-routing information that is used to move the packet across a network. The TCP header contains the information needed to guarantee the packet's delivery to the target. The iSCSI header describes how to extract SCSI commands and data for the target. iSCSI adds an optional CRC, known as the *digest*, beyond the TCP checksum and Ethernet CRC to ensure datagram integrity. The header and the data digests are optionally used in the PDU to validate integrity, data placement, and correct operation.



iSCSI Discovery

An initiator must discover the location of the target on a network, and the names of the targets available to it before it can establish a session. This discovery can take place in two ways: *SendTargets discovery* and *internet Storage Name Service (iSNS)*. In SendTargets discovery, the initiator is manually configured with the target's network portal, which it uses to establish a discovery session with the iSCSI service on the target. The initiator issues the SendTargets command, and the target responds with the names and addresses of the targets available to the host.

iSNS enables the automatic discovery of iSCSI devices on an IP network. The initiators and targets can be configured to automatically register themselves with the iSNS server. Whenever an initiator wants to know the targets that it can access, it can query the iSNS server for a list of available targets.



Discovery can also take place by using Service Location Protocol (SLP). However, this is less commonly used than SendTargets discovery and iSNS.

iSCSI Names

A unique worldwide iSCSI identifier, known as an *iSCSI name*, is used to name the initiators and targets within an iSCSI network to facilitate communication. The unique identifier can be a

combination of department, application, manufacturer name, serial number, asset number, or a tag that can be used to recognize and manage a storage resource. There are two types of iSCSI names:

■ **iSCSI Qualified Name (IQN):** An organization must own a registered domain name in order to generate iSCSI Qualified Names. This domain name does not have to be active or resolve to an address. It just needs to be reserved to prevent other organizations from using the same domain name to generate iSCSI names. A date is included in the name to avoid potential conflicts caused by transfer of domain names; the organization is required to have owned the domain name on that date. An example of an IQN is iqn.2008-02.com.example:optional_string The optional_string provides a serial number, an asset number, or any of the storage device identifiers.

■ **Extended Unique Identifier (EUI):** An EUI is a globally unique identifier based on the IEEE EUI-64 naming standard. An EUI comprises the eui prefix followed by a 16-character hexadecimal name, such as eui.0300732A32598D26. The 16-character part of the name includes 24 bits for the company name assigned by IEEE and 40 bits for a unique ID, such as a serial number. This allows for a more streamlined, although less user-friendly, name string because the resulting iSCSI name is simply eui followed by the hexadecimal WWN. In either format, the allowed special characters are dots, dashes, and blank spaces. The iSCSI Qualified Name enables storage administrators to assign meaningful names to storage devices, and therefore manage those devices more easily.

Network Address Authority (NAA) is an additional iSCSI node name type to enable worldwide naming format as defined by the Inter National Committee for Information Technology Standards (INCITS) T11 - Fiber Channel (FC) protocol sand used by Serial Attached SCSI (SAS). This format enables SCSI storage devices containing both iSCSI ports and SAS ports to use the same NAA-based SCSI device name. This format is defined by RFC3980, “T11 Network Address Authority (NAA) Naming Format for iSCSI Node Names.”

iSCSI Session

An iSCSI session is established between an initiator and a target. A session ID (SSID), which includes an initiator ID (ISID) and a target ID (TSID), identifies a session. The session can be intended for one of the following:

■Discovery of available targets to the initiator and the location of a specific target on a network

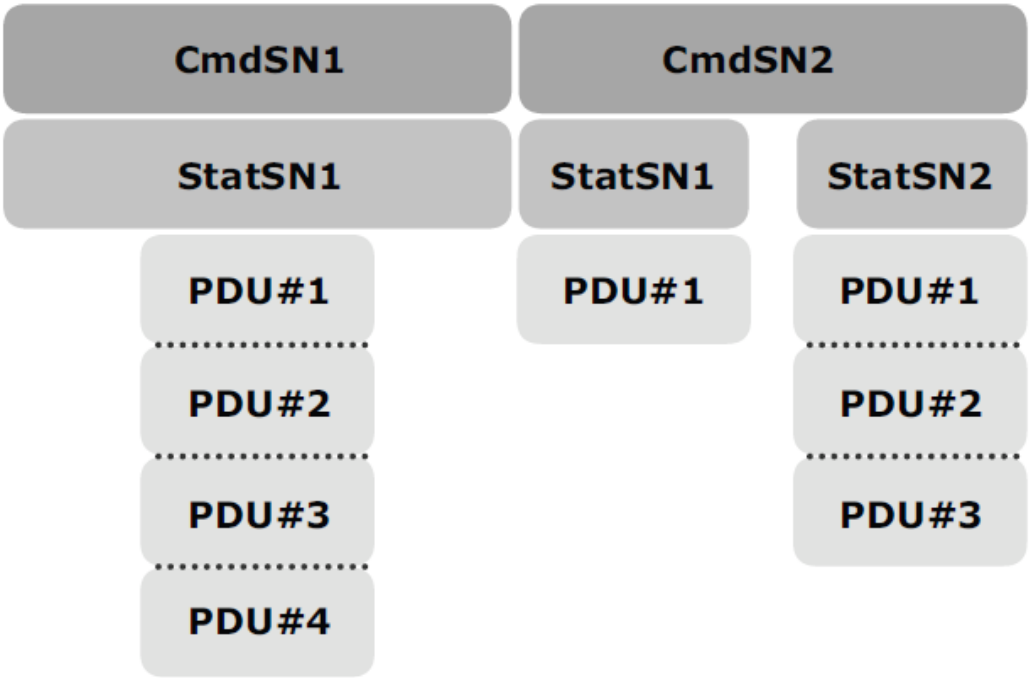
■Normal operation of iSCSI (transferring data between initiators and targets)

TCP connections may be added and removed within a session. Each iSCSI connection within the session has a unique connection ID (CID).

Ordering and Numbering

iSCSI communication between initiators and targets is based on the request response command sequences. A command sequence may generate multiple PDUs. A *command sequence number (CmdSN)* within an iSCSI session is used to number all initiator-to-target command PDUs belonging to the session. This number is used to ensure that every command is delivered in the same order in which it is transmitted, regardless of the TCP connection that carries the command in the session.

Command sequencing begins with the first login command and the CmdSN is incremented by one for each subsequent command. The iSCSI target layer is responsible for delivering the commands to the SCSI layer in the order of their CmdSN. This ensures the correct order of data and commands at a target even when there are multiple TCP connections between an initiator and the target using portal groups. Similar to command numbering, a *status sequence number (StatSN)* is used to sequentially number status responses. These unique numbers are established at the level of the TCP connection.



A target sends the *request-to-transfer (R2T)* PDUs to the initiator when it is ready to accept data. *Data sequence number (DataSN)* is used to ensure in-order delivery of data within the same command. The DataSN and R2T sequence numbers are used to sequence data PDUs and R2Ts, respectively. Each of these sequence numbers is stored locally as an unsigned 32-bit integer counter defined by iSCSI. These numbers are communicated between the initiator and target in the appropriate iSCSI PDU fields during command, status, and data exchanges.

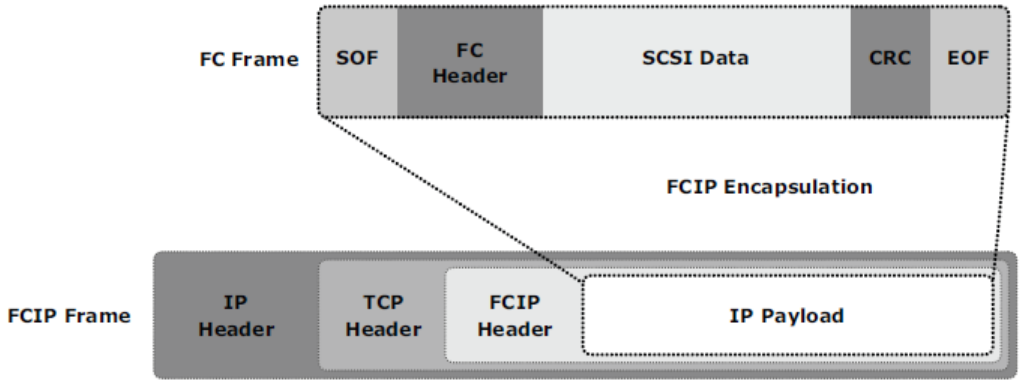
In the case of read operations, the DataSN begins at zero and is incremented by one for each subsequent data PDU in that command sequence. In the case of a write operation, the first unsolicited data PDU or the first data PDU in response to an R2T begins with a DataSN of zero and increments by one for each subsequent data PDU. R2TSN is set to zero at the initiation of the command and incremented by one for each subsequent R2T sent by the target for that command.

3.10.2 FCIP

Organizations are now looking for new ways to transport data throughout the enterprise, locally over the SAN as well as over longer distances, to ensure that data reaches all the users who need it. One of the best ways to achieve this goal is to interconnect geographically dispersed SANs through reliable, high-speed links. This approach involves transporting FC block data over the existing IP infrastructure used throughout the enterprise.

The FCIP standard has rapidly gained acceptance as a manageable, cost effective way to blend the best of two worlds: FC block-data storage and the proven, widely deployed IP infrastructure. FCIP is a tunneling protocol that enables distributed FC SAN islands to be transparently interconnected over existing IP-based local, metropolitan, and wide-area networks. As a result, organizations now have a better way to protect, store, and move their data while leveraging investments in existing technology.

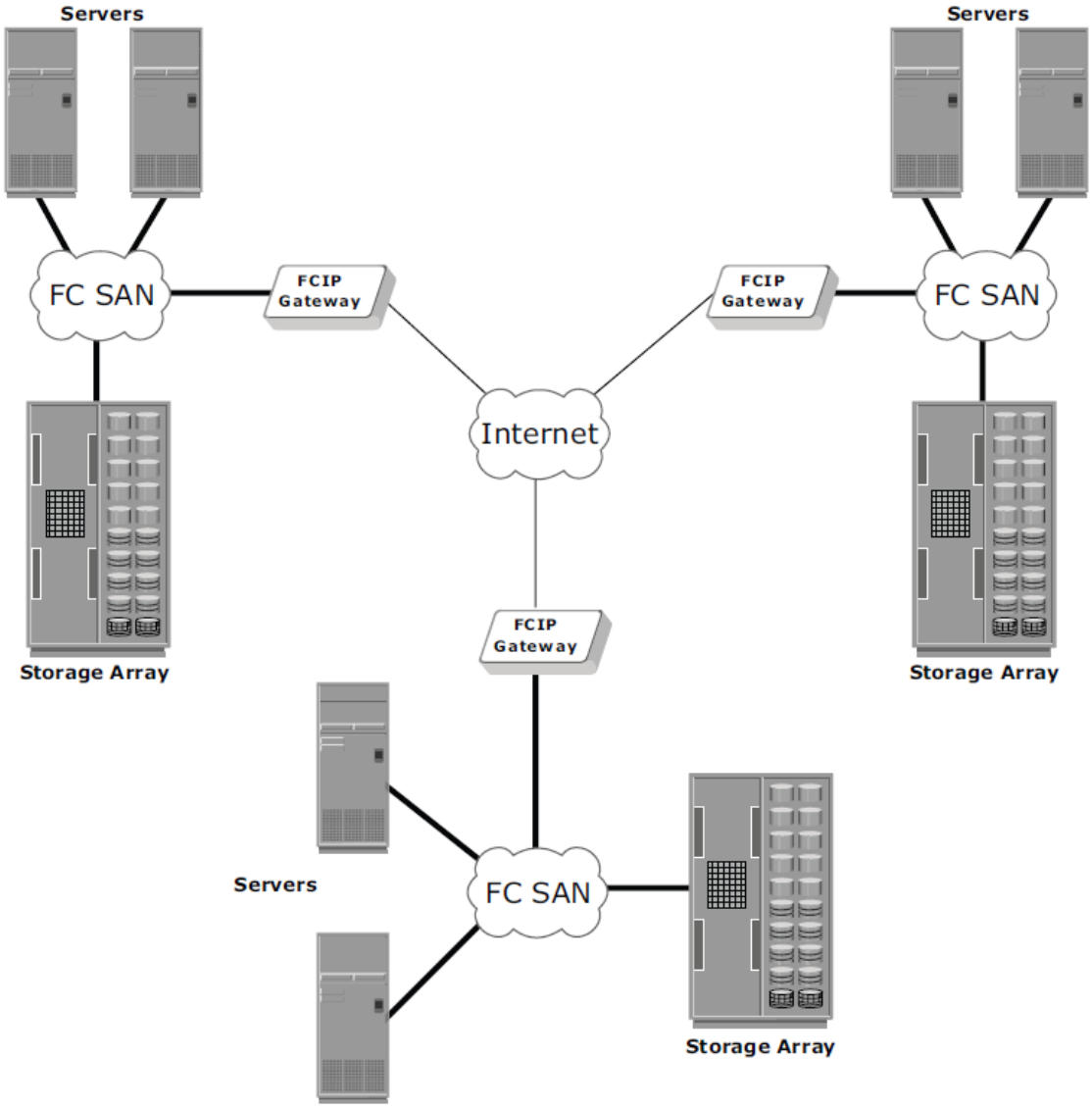
FCIP uses TCP/IP as its underlying protocol. In FCIP, the FC frames are encapsulated onto the IP payload, as shown in Figure 8-9. FCIP does not manipulate FC frames (translating FC IDs for transmission). When SAN islands are connected using FCIP, each interconnection is called an *FCIP link*. A successful FCIP link between two SAN islands results in a fully merged FC fabric.



FCIP Topology

An FCIP environment functions as if it is a single cohesive SAN environment. Before geographically dispersed SANs are merged, a fully functional layer 2 network exists on the SANs. This layer 2 network is a standard SAN fabric. These physically independent fabrics are merged into a single fabric with an IP link between them.

An FCIP gateway router is connected to each fabric via a standard FC connection.The fabric treats these routers like layer 2 fabric switches. The other port on the router is connected to an IP network and an IP address is assigned to that port. This is similar to the method of assigning an IP address to an iSCSI port on a gateway. Once IP connectivity is established, the two independent fabrics are merged into a single fabric. When merging the two fabrics, all the switches and routers must have unique domain IDs, and the fabrics must contain unique zone set names. Failure to ensure these requirements will result in a segmented fabric. The FC addresses on each side of the link are exposed to the other side, and zoning or masking can be done to any entity in the new environment.



FCIP Performance and Security

Performance, reliability, and security should always be taken into consideration when implementing storage solutions. The implementation of FCIP is also subject to the same consideration. From the perspective of performance, multiple paths to multiple FCIP gateways from different switches in the layer 2 fabrics eliminates single points of failure and provides increased bandwidth. In a scenario of extended distance, the IP network may be a bottleneck if sufficient bandwidth is not available. In addition, because FCIP creates a unified fabric, disruption in the underlying IP network can cause instabilities in the SAN environment. These include a segmented fabric, excessive RSCNs, and host timeouts. The vendors of FC switches have recognized some of the drawbacks related to FCIP and have implemented features to provide additional stability, such as the capability to segregate FCIP traffic into a separate virtual fabric.

Security is also a consideration in an FCIP solution because the data is transmitted over public IP channels. Various security options are available to protect the data based on the router's support. IPSec is one such security measure that can be implemented in the FCIP environment.

3.11 Content addressed storage (CAS)

In the life cycle of information, data is actively created, accessed, edited, and changed. As data ages, it becomes less likely to change and eventually becomes "fixed" but continues to be ac-

cessed by multiple applications and users. This data is called *fixed content*. Traditionally, fixed content was not treated as a specialized form of data and was stored using a variety of storage media, ranging from optical disks to tapes to magnetic disks. While these traditional technologies store content, none of them provide all of the unique requirements for storing and accessing fixed content.

Accumulations of fixed content such as documents, e-mail messages, web pages, and digital media throughout an organization have resulted in an unprecedented growth in the amount of data. It has also introduced the challenge of managing fixed content. Furthermore, users demand assurance that stored content has not changed and require an immediate online access to fixed content. These requirements resulted in the development of Content-Addressed Storage (CAS).

CAS is an *object-based system* that has been purposely built for storing fixed content data. It is designed for secure online storage and retrieval of fixed content. Unlike file-level and block-level data access that use file names and the physical location of data for storage and retrieval, CAS stores user data and its attributes as separate objects. The stored object is assigned a globally unique address known as a *content address (CA)*. This address is derived from the object's binary representation. CAS provides an optimized and centrally managed storage solution that can support *single-instance storage (SiS)* to eliminate multiple copies of the same data.

3.11.1 Features and Benefits of CAS

CAS has emerged as an alternative to tape and optical solutions because it overcomes many of their obvious deficiencies. CAS also meets the demand to improve data accessibility and to properly protect, dispose of, and ensure service-level agreements for archived data. The features and benefits of CAS include the following:

■ **Content authenticity:** It assures the genuineness of stored content. This is achieved by generating a unique content address and automating the process of continuously checking and recalculating the content address for stored objects. Content authenticity is assured because the address assigned to each piece of fixed content is as unique as a fingerprint. Every time an object is read, CAS uses a hashing algorithm to recalculate the object's content address as a validation step and compares the result to its original content address. If the object fails validation, it is rebuilt from its mirrored copy.

■ **Content integrity:** Refers to the assurance that the stored content has not been altered. Use of hashing algorithm for content authenticity also ensures content integrity in CAS. If the fixed content is altered, CAS assigns a new address to the altered content, rather than overwrite the original fixed content, providing an audit trail and maintaining the fixed content in its original state. As an integral part of maintaining data integrity and audit trail capabilities, CAS supports parity RAID protection in addition to mirroring. Every object in a CAS system is systematically checked in the background. Over time, every object is tested, guaranteeing content integrity even in the case of hardware failure, random error, or attempts to alter the content with malicious intent.

■ **Location independence:** CAS uses a unique identifier that applications can leverage to retrieve data rather than a centralized directory, path names, or URLs. Using a content address to access fixed content makes the physical location of the data irrelevant to the application requesting the data. Therefore the location from which the data is accessed is transparent to the application. This yields complete content mobility to applications across locations.

■ **Single-instance storage (SiS):** The unique signature is used to guarantee the storage of only a single instance of an object. This signature is derived from the binary representation of the object. At write time, the CAS system is polled to see if it already has an object with the same signature. If the object is already on the system, it is not stored, rather only a pointer to that object is

created. SiS simplifies storage resource management tasks, especially when handling hundreds of terabytes of fixed content.

■ **Retention enforcement:** Protecting and retaining data objects is a core requirement of an archive system. CAS creates two immutable components: a data object and a meta-object for every object stored. The meta object stores object's attributes and data handling policies. For systems that support object-retention capabilities, the retention policies are enforced until the policies expire.

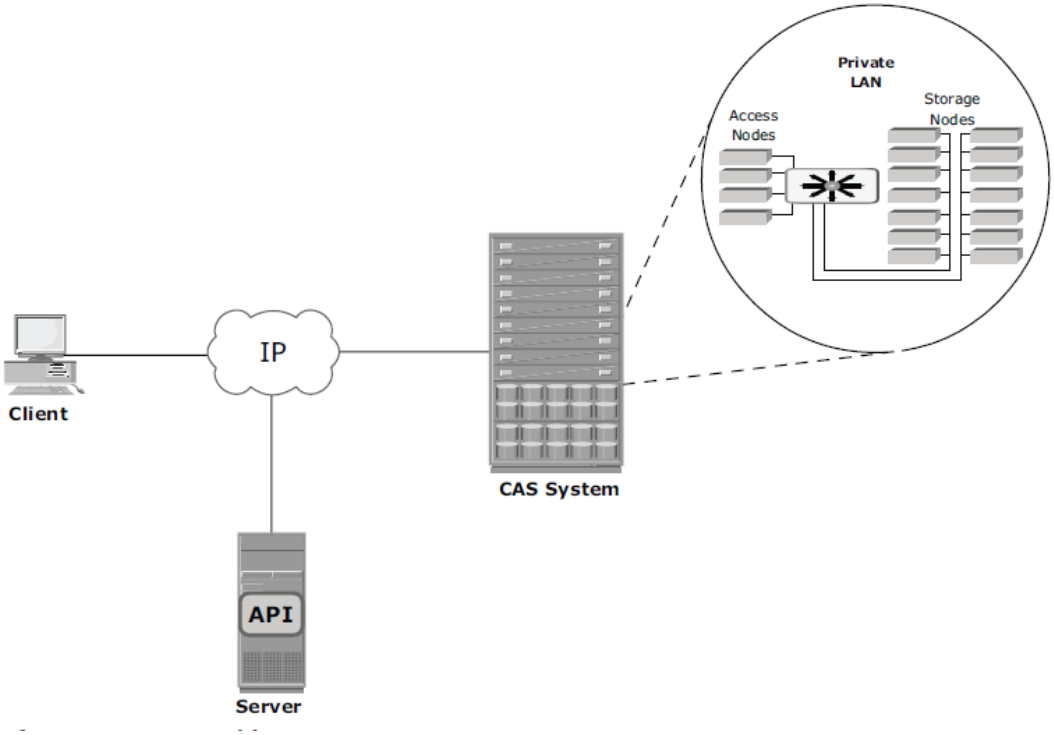
■ **Record-level protection and disposition:** All fixed content is stored in CAS once and is backed up with a protection scheme. The array is composed of one or more storage clusters. Some CAS architectures provide an extra level of protection by replicating the content onto arrays located at a different location. The disposition of records also follows the stringent guidelines established by regulators for shredding and disposing of data in electronic formats.

■ **Technology independence:** The CAS system interface is impervious to technology changes. As long as the application server is able to map the original content address the data remains accessible. Although hardware changes are inevitable, the goal of CAS hardware vendors is to ensure compatibility across platforms.

■ **Fast record retrieval:** CAS maintains all content on disks that provide sub second "time to first byte" (200 ms–400 ms) in a single cluster. Random disk access in CAS enables fast record retrieval.

3.11.2 CAS Architecture

A client accesses the CAS-Based storage over a LAN through the server that runs the CAS API (application programming interface). The CAS API is responsible for performing functions that enable an application to store and retrieve the data. CAS architecture is a *Redundant Array of Independent Nodes (RAIN)*. It contains storage nodes and access nodes networked as a cluster by using a private LAN that is internal to it. The internal LAN can be reconfigured automatically to detect the configuration changes such as the addition of storage or access nodes. Clients access the CAS on a separate LAN, which is used for interconnecting clients and servers to the CAS. The nodes are configured with low-cost, high-capacity ATA HDDs. These nodes run an operating system with special software that implements the features and functionality required in a CAS system.



When the cluster is installed, the nodes are configured with a "role" defining the functionality they provide to the cluster. A node can be configured as a storage node, an access node, or a dual-role node. *Storage nodes* store and protect data objects. They are sometimes referred to as *back-end nodes*. *Access nodes* provide connectivity to application servers through the customer's LAN. They establish connectivity through a private LAN to the storage nodes in the cluster. The number of access nodes is determined by the amount of user required throughput from the cluster. If a node is configured solely as an "access node," its disk space cannot be used to store data objects. This configuration is generally found in older installations of CAS. Storage and retrieval requests are sent to the access node via the customer's LAN. *Dual-role nodes* provide both storage and access node capabilities. This node configuration is more typical than a pure access node configuration. Almost all CAS products have the same features and options. Some may be implemented differently, but the following features are an essential part of any CAS solution:

■ **Integrity checking:** It ensures that the content of the file matches the digital signature (hashed output or CA). The integrity checks can be done on every read or by using a background process. If problems are identified in any of the objects the nodes automatically repair or regenerate the object.

■**Data protection and node resilience:** This ensures that the content stored on the CAS system is available in the event of disk or node failure. Some CAS systems provide local replication or mirrors that copy a data object to another node in the same cluster. This decreases the total available capacity by 50 percent. Parity protection is another way to protect CAS data. It uses less capacity to store data, but takes longer to regenerate the data if corrupted. Remote replication copies data objects to a secondary storage device in a remote location. Remote replication is used as a disaster recovery solution or for backup.

■ **Load balancing:** Distributes data objects on multiple nodes to provide maximum throughput, availability, and capacity utilization.

■ **Scalability:** Adding more nodes to the cluster without any interruption to data access and with minimum administrative overhead.

■ **Self diagnosis and repair:** Automatically detects and repairs corrupted objects and alert the administrator of any potential problem. These failures can be at an object level or a node level. They are transparent to the users who access the archive. CAS systems can be configured to alert remote support teams who diagnose and make repairs remotely.

■ **Report generation and event notification:** Provides on-demand reporting and event notification. A command-line interface (CLI) or a graphical user interface (GUI) can produce various types of reports. Any event notification can be communicated to the administrator through syslog, SNMP, SMTP, or e-mail.

■ **Fault tolerance:** Ensures data availability if a component of the CAS system fails, through the use of redundant components and data protection schemes. If remote replication of CAS is implemented, failover to the remote CAS system occurs when the primary CAS system is unavailable.

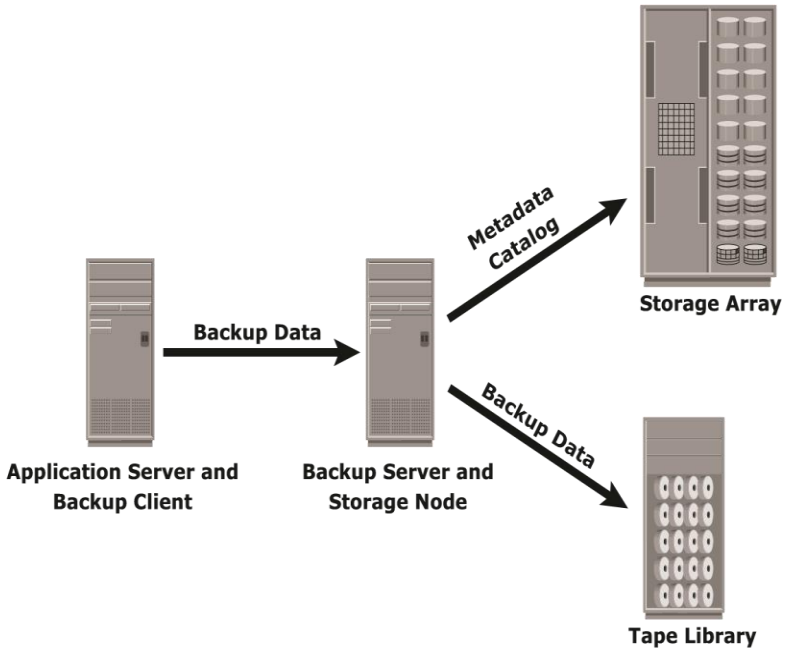
■**Audit trails:** Enable documentation of management activity and any access and disposition of data. Audit trails are mandated by compliance requirements.

UNIT-4

INFORMATION AVAILABILITY MONITORING AND MANAGING DATA CENTER

4.1 Backup Architecture

A backup system uses client/server architecture with a backup server and multiple backup clients. The backup server manages the backup operations and maintains the backup catalog, which contains information about the backup process and backup metadata. The backup server depends on backup clients to gather the data to be backed up.



Backup architecture and process

The backup clients can be local to the server or they can reside on another server, presumably to back up the data visible to that server. The backup server receives backup metadata from the backup clients to perform its activities.

The storage node is responsible for writing data to the backup device (in a backup environment, a storage node is a host that controls backup devices). Typically, the storage node is integrated with the backup server and both are hosted on the same physical platform. A backup device is attached directly to the storage node's host platform. Some backup architecture refers to the storage node as the media server because it connects to the storage device. Storage nodes play an important role in backup planning because they can be used to consolidate backup servers.

The backup process is based on the policies defined on the backup server, such as the time of day or completion of an event. The backup server then initiates the process by sending a request to a backup client (backups can also be initiated by a client). This request instructs the backup client to send its metadata to the backup server, and the data to be backed up to the appropriate storage node. On receiving this request, the backup client sends the metadata to the backup serv-

er. The backup server writes this metadata on its metadata catalog. The backup client also sends the data to the storage node, and the storage node writes the data to the storage device.

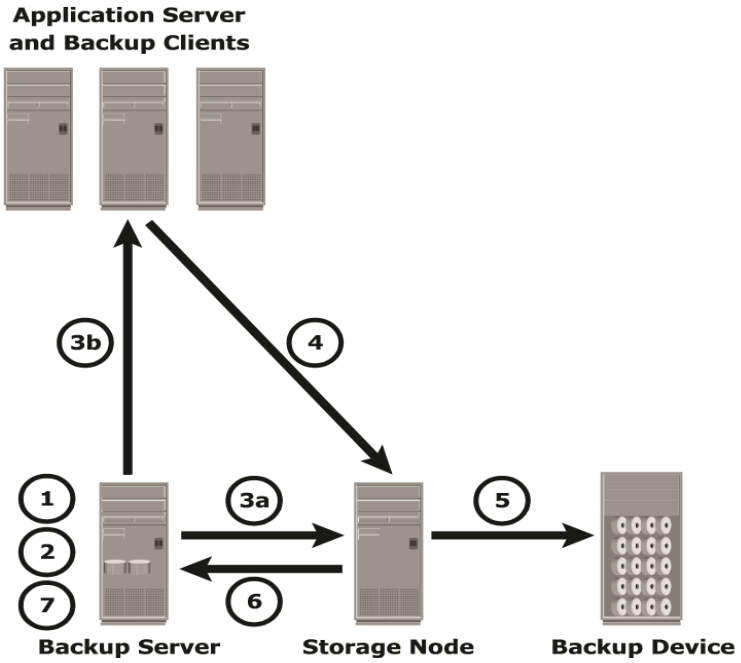
After all the data is backed up, the storage node closes the connection to the backup device. The backup server writes backup completion status to the metadata catalog.

Backup software also provides extensive reporting capabilities based on the backup catalog and the log files. These reports can include information such as the amount of data backed up, the number of completed backups, the number of incomplete backups, and the types of errors that may have occurred. Reports can be customized depending on the specific backup software used.

4.1.1 Backup and Restore Operations

When a backup process is initiated, significant network communication takes place between the different components of a backup infrastructure. The backup server initiates the backup process for different clients based on the backup schedule configured for them. For example, the backup process for a group of clients may be scheduled to start at 3:00 am every day.

The backup server coordinates the backup process with all the components in a backup configuration. The backup server maintains the information about backup clients to be contacted and storage nodes to be used in a backup operation. The backup server retrieves the backup-related information from the backup catalog and, based on this information, instructs the storage node to load the appropriate backup media into the backup devices. Simultaneously, it instructs the backup clients to start scanning the data, package it, and send it over the network to the assigned storage node. The storage node, in turn, sends metadata to the backup server to keep it updated about the media being used in the backup process. The backup server continuously updates the backup catalog with this information.

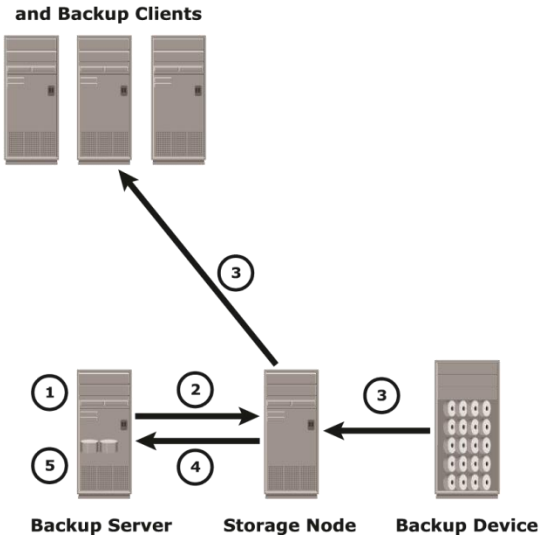


Backup operation

After the data is backed up, it can be restored when required. A restore process must be manually initiated. Some backup software has a separate application for restore operations. These restore applications are accessible only to the administrators.

The administrator first selects the data to be restored and initiates the restore process. The backup server, using the appropriate storage node, then identifies the backup media that needs to be mounted on the backup devices. Data is then read and sent to the client that has been identified to receive the restored data.

Some restorations are successfully accomplished by recovering only the requested production data. For example, the recovery process of a spreadsheet is completed when the specific file is restored. In database restorations, additional data such as log files and production data must be restored. This ensures application consistency for the restored data. In these cases, the RTO is extended due to the additional steps in the restoration process.



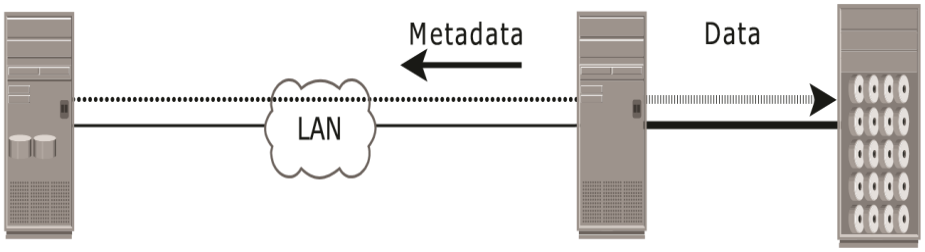
Restore operation

Upon receiving a restore request, an administrator opens the restore application to view the list of clients that have been backed up. While selecting the client for which a restore request has been made, the administrator also needs to identify the client that will receive the restored data. Data can be restored on the same client for whom the restore request has been made or on any other client. The administrator then selects the data to be restored and the specified point in time to which the data has to be restored based on the RPO. Note that because all of this information comes from the backup catalog, the restore application must also communicate to the backup server.

4.2 Backup Topologies

Three basic topologies are used in a backup environment: direct attached backup, LAN based backup, and SAN based backup. A mixed topology is also used by combining LAN based and SAN based topologies.

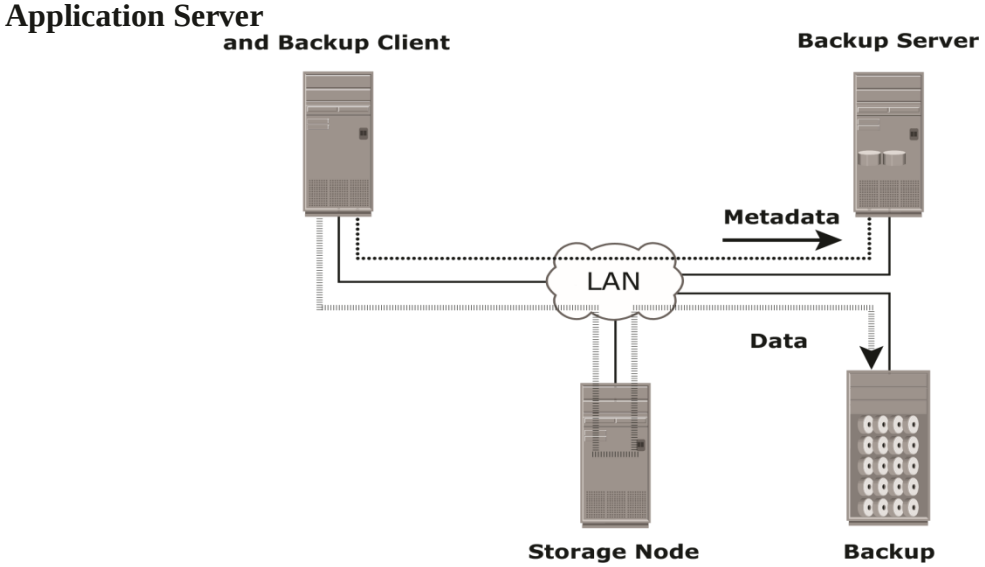
In a Direct-attached backup, a backup device is attached directly to the client. Only the metadata is sent to the backup server through the LAN. This configuration frees the LAN from backup traffic. The example below diagram depicts use of a backup device that is not shared. As the environment grows, however, there will be a need for central management of all backup de-



vices and to share the resources to optimize costs. An appropriate solution is to share the backup devices among multiple servers. In this example, the client also acts as a storage node that writes data on the backup device to the LAN and all storage devices are directly attached to the storage

node . The data to be backed up is transferred from the backup client (source), to the backup device (destination) over the LAN, which may affect network performance. Streaming across the LAN also affects network performance of all systems connected to the same segment as the backup server. Network resources are severely constrained when multiple clients access and share the same tape library unit (TLU).

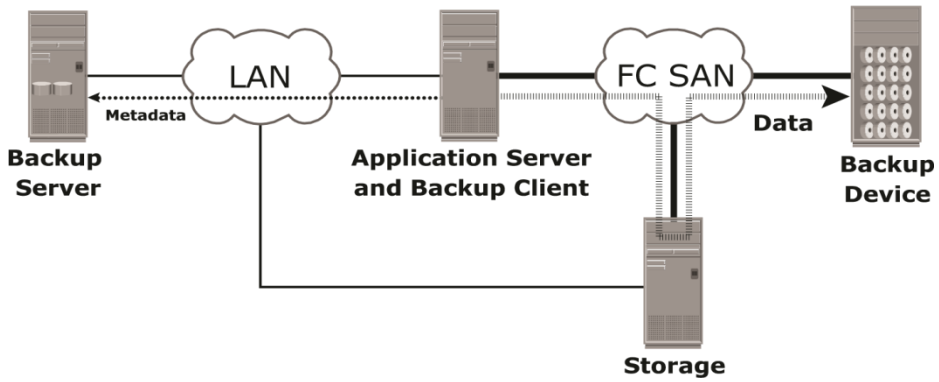
This impact can be minimized by adopting a number of measures, such as configuring separate networks for backup and installing dedicated storage nodes for some application servers.



LAN-based backup topology

The SAN-based backup is also known as the LAN-free backup. SAN-based backup. The SAN-based backup topology is the most appropriate solution when a backup device needs to be shared among the clients. In this case the backup device and clients are attached to the SAN.

In this example, clients read the data from the mail servers in the SAN and write to the SAN attached backup device. The backup data traffic is restricted to the SAN, and backup metadata is transported over the LAN. However, the volume of metadata is insignificant when compared to production data. LAN performance is not degraded in this configuration.

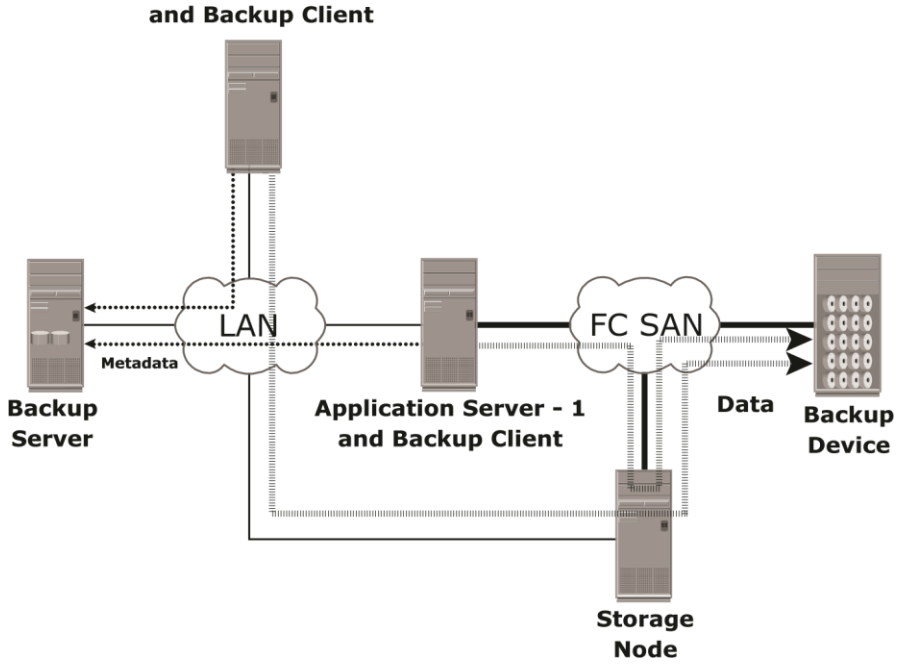


By removing the network bottleneck, the SAN improves backup to tape performance because it frees the LAN from backup traffic. At the same time, LAN free backups may affect the host and the application, as they consume host I/O bandwidth, memory, and CPU resources.

SAN Based Topology

The emergence of low-cost disks as a backup medium has enabled disk arrays to be attached to the SAN and used as backup devices. A tape backup of these data backups on the disks can be created and shipped offsite for disaster recovery and long-term retention.

The mixed topology uses both the LAN-based and SAN-based topologies. This topology might be implemented for several reasons, including cost, server location, reduction in administrative overhead, and performance considerations.



Mixed backup topology

Server less Backup

Server less backup is a LAN-free backup methodology that does not involve a backup server to copy data. The copy may be created by a network-attached controller, utilizing a SCSI extended

copy or an appliance within the SAN. These backups are called server less because they use SAN resources instead of host resources to transport backup data from its source to the backup device, reducing the impact on the application server.

Another widely used method for performing server less backup is to leverage local and remote replication technologies. In this case, a consistent copy of the production data is replicated within the same array or the remote array, which can be moved to the backup device through the use of a storage node.

4.3Replication Topologies

4.3.1 Local Replication Topology

Host-based and storage-based replications are the two major technologies adopted for local replication. File system replication and LVM-based replication are examples of host-based local replication technology. Storage array–based replication can be implemented with distinct solutions namely, full-volume mirroring, pointer based full-volume replication, and pointer-based virtual replication.

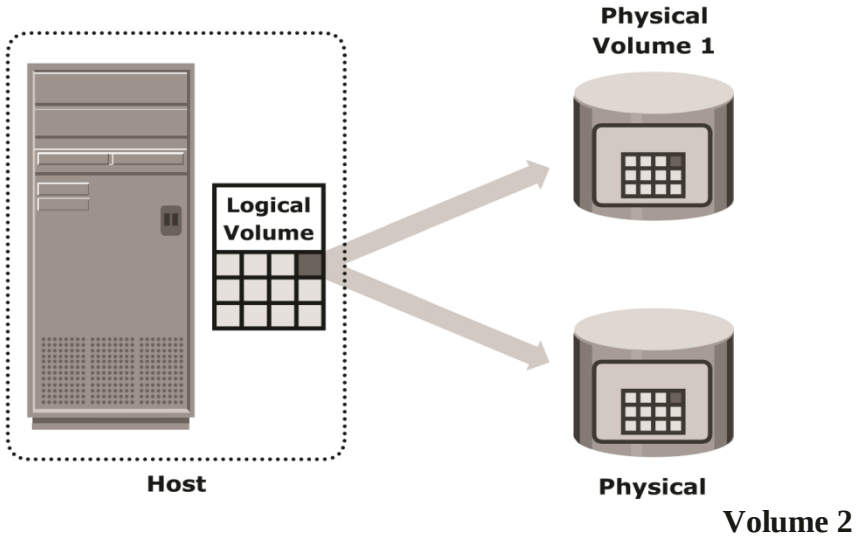
Host-Based Local Replication

In host-based replication, logical volume managers (LVMs) or the file systems perform the local replication process. LVM-based replication and file system (FS) snapshot are examples of host-based local replication.

LVM-Based Replication

In LVM-based replication, logical volume manager is responsible for creating and controlling the host-level logical volume. An LVM has three components: physical volumes (physical disk), volume groups, and logical volumes. A volume groups is created by grouping together one or more physical volumes. Logical volumes are created within a given volume group. A volume group can have multiple logical volumes.

In LVM-based replication, each logical partition in a logical volume is mapped to two physical partitions on two different physical volumes. An application write to a logical partition is written to the two physical partitions by the LVM device driver. This is also known as LVM mirroring. Mirrors can be split and the data contained therein can be independently accessed. LVM mirrors can be added or removed dynamically.



LVM-based mirroring

Advantages of LVM-Based Replication

The LVM-based replication technology is not dependent on a vendor-specific storage system. Typically, LVM is part of the operating system and no additional license is required to deploy LVM mirroring.

Limitations of LVM-Based Replication

As every write generated by an application translates into two writes on the disk, an additional burden is placed on the host CPU. This can degrade application performance. Presenting an LVM-based local replica to a second host is usually not possible because the replica will still be part of the volume group, which is usually accessed by one host at any given time.

Tracking changes to the mirrors and performing incremental synchronization operations is also a challenge as all LVMs do not support incremental resynchronization. If the devices are already protected by some level of RAID on the array, then the additional protection provided by mirroring is unnecessary. This solution does not scale to provide replicas of federated databases and applications. Both the replica and the source are stored within the same volume group. Therefore, the replica itself may become unavailable if there is an error in the volume group. If the server fails, both source and replica are unavailable until the server is brought back online.

File System Snapshot

File system (FS) snapshot is a pointer-based replica that requires a fraction of the space used by the original FS. This snapshot can be implemented by either FS itself or by LVM. It uses Copy on First Write (CoFW) principle. CoFW mechanism is discussed later in the chapter.

When the snapshot is created, a bitmap and a block map are created in the metadata of the Snap FS. The bitmap is used to keep track of blocks that are changed on the production FS after creation of the snap. The block map is used to indicate the exact address from which data is to be

read when the data is accessed from the Snap FS. Immediately after creation of the snapshot all reads from the snapshot will actually be served by reading the production FS.

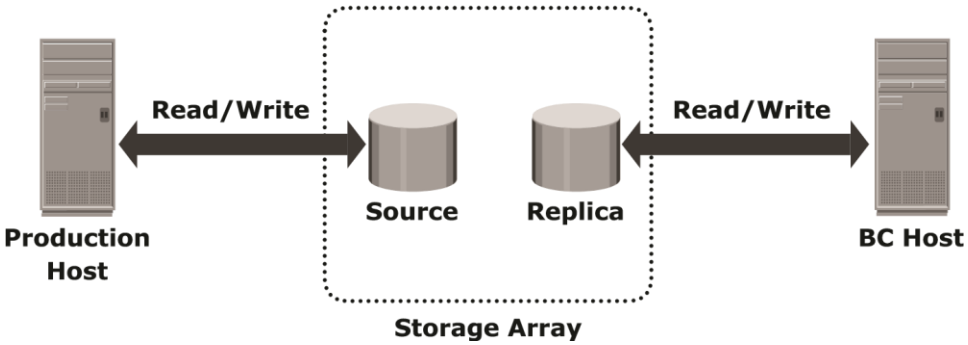
To read from the Snap FS, the bitmap is consulted. If the bit is 0, then the read is directed to the production FS. If the bit is 1, then the block address is obtained from the block map and data is read from that address.Reads from the production FS work as normal.

4.3.2 Storage Array–Based Replication

In Storage array-based local replication, the array operating environment performs the local replication process. The host resources such as CPU and memory are not used in the replication process. Consequently, the host is not burdened by the replication operations. The replica can be accessed by an alternate host for any business operations.

In this replication, the required number of replica devices should be selected on the same array and then data is replicated between source-replica pairs. A database could be laid out over multiple physical volumes and in that case all the devices must be replicated for a consistent PIT copy of the database.

Below diagram shows storage array based local replication, where source and target are in the same array and accessed by different hosts.



Storage array-based replication

Storage array-based local replication can be further categorized as full-volume mirroring, pointer-based full-volume replication, and pointer-based virtual replication. Replica devices are also referred as target devices, accessible by business continuity host.

Full-Volume Mirroring

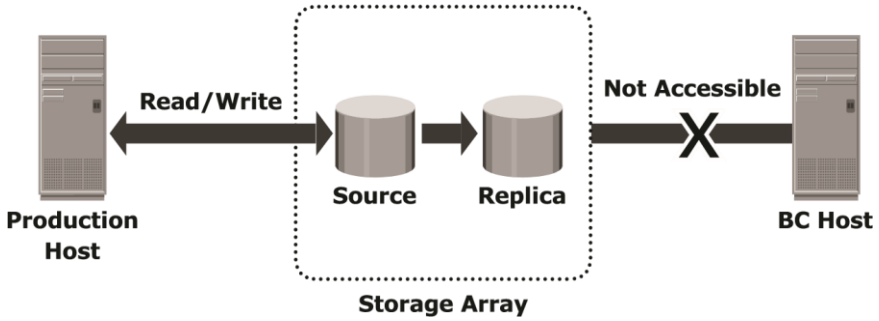
In full-volume mirroring, the target is attached to the source and established as a mirror of the source. Existing data on the source is copied to the target. New updates to the source are also up-

dated on the target. After all the data is copied and both the source and the target contain identical data, the target can be considered a mirror of the source.

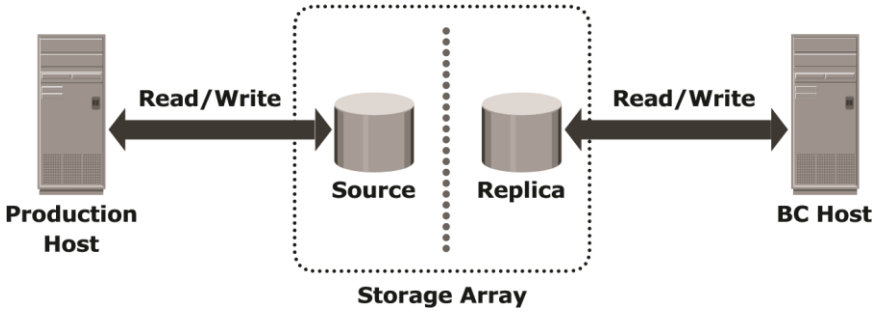
While the target is attached to the source and the synchronization is taking place, the target remains unavailable to any other host. However, the production host can access the source.

After synchronization is complete, the target can be detached from the source and is made available for BC operations.

Full Volume Mirroring



Notice that both the source and the target can be accessed for read and write operations by the



production hosts.

After the split from the source, the target becomes a PIT copy of the source. The point-in-time of a replica is determined by the time when the source is detached from the target. For example, if the time of detachment is 4:00 pm, the PIT for the target is 4:00 pm

After detachment, changes made to both source and replica can be tracked at some predefined granularity. This enables incremental resynchronization (source to target) or incremental restore (target to source). The granularity of the data change can range from 512 byte blocks to 64 KB blocks. Changes are typically tracked using bitmaps, with one bit assigned for each block. If any updates occur to a particular block, the whole block is marked as changed, regardless of the size of the actual update. However, for resynchronization (or restore), only the changed blocks have to be copied, eliminating the need for a full synchronization (or restore) operation. This method reduces the time required for these operations considerably.

In full-volume mirroring, the target is inaccessible for the duration of the synchronization process, until detachment from the source. For large databases, this can take a long time.

Pointer-Based, Full-Volume Replication

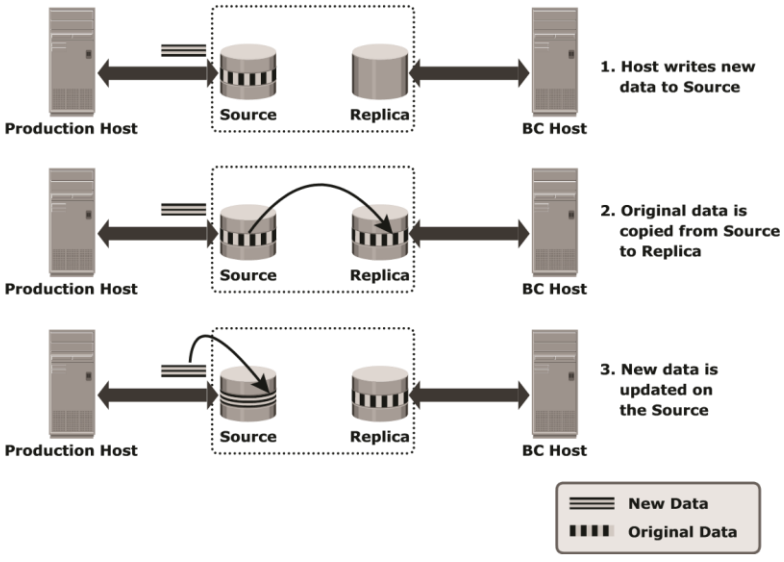
An alternative to full-volume mirroring is pointer-based full-volume replication. Like full-volume mirroring, this technology can provide full copies of the source data on the targets. Unlike full-volume mirroring, the target is made immediately available at the activation of the replication session. Hence, one need not wait for data synchronization to, and detachment of, the target in order to access it. The time of activation defines the PIT copy of source.

Pointer-based, full-volume replication can be activated in either Copy on First Access (CoFA) mode or Full Copy mode. In either case, at the time of activation, a protection bitmap is created for all data on the source devices. Pointers are initialized to map the (currently) empty data blocks on the target to the corresponding original data blocks on the source. The granularity can range from 512 byte blocks to 64 KB blocks or higher. Data is then copied from the source to the target, based on the mode of activation.

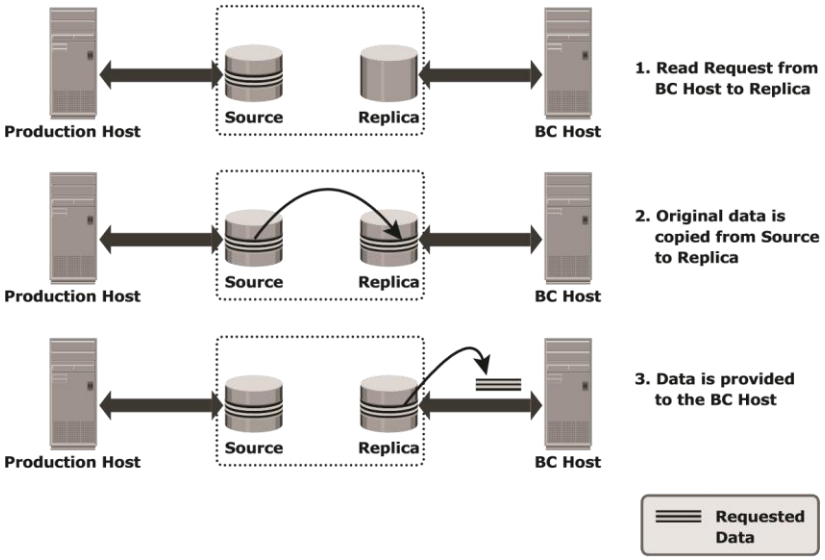
In CoFA, after the replication session is initiated, data is copied from the source to the target when the following occurs:

- A write operation is issued to a specific address on the source for the first time.
- A read or write operation is issued to a specific address on the target for the first time.

When a write is issued to the source for the first time after session activation, original data at that address is copied to the target. After this operation, the new data is updated on the source. This ensures that original data at the point-in-time of activation is preserved on the target.



When a read is issued to the target for the first time after session activation, the original data is



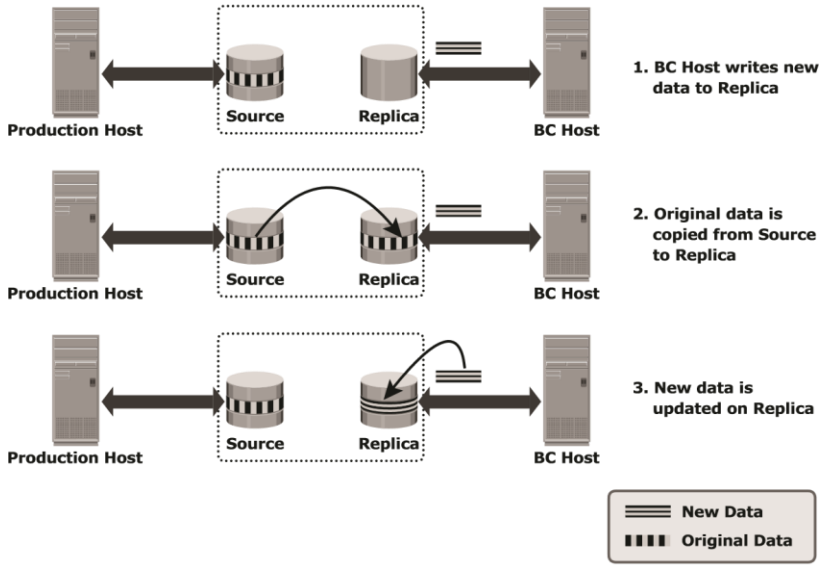
copied from the source to the target and is made available to the host.

When a write is issued to the target for the first time after session activation, the original data is copied from the source to the target. After this, the new data is updated on the target.

In all cases, the protection bit for that block is reset to indicate that the original data has been copied over to the target. The pointer to the source data can now be discarded. Subsequent writes to the same data block on the source, and reads or writes to the same data blocks on the target, do not trigger a copy operation (and hence are termed Copy on First Access).

If the replication session is terminated, then the target device only has the data that was accessed until the termination, not the entire contents of the source at the point-in-time. In this case, the data on the target cannot be used for a restore, as it is not a full replica of the source.

In Full Copy mode, all data from the source is copied to the target in the background. Data is copied regardless of access. If access to a block that has not yet been copied is required, this block is preferentially copied to the target. In a complete cycle of the Full Copy mode, all data from the source is copied to the target. If the replication session is terminated now, the target will contain all the original data from the source at the point-in-time of activation. This makes the target a viable copy for recovery, restore, or other business continuity operations.



The key difference between pointer-based, Full Copy mode and full-volume mirroring is that the target is immediately accessible on session activation in Full Copy mode. In contrast, one has to wait for synchronization and detachment to access the target in full-volume mirroring.

Both the full-volume mirroring and pointer-based full-volume replication technologies require the target devices to be at least as large as the source devices. In addition, full-volume mirroring and pointer-based full-volume replication in Full Copy mode can provide incremental resynchronization or restore capability.

Pointer-Based Virtual Replication

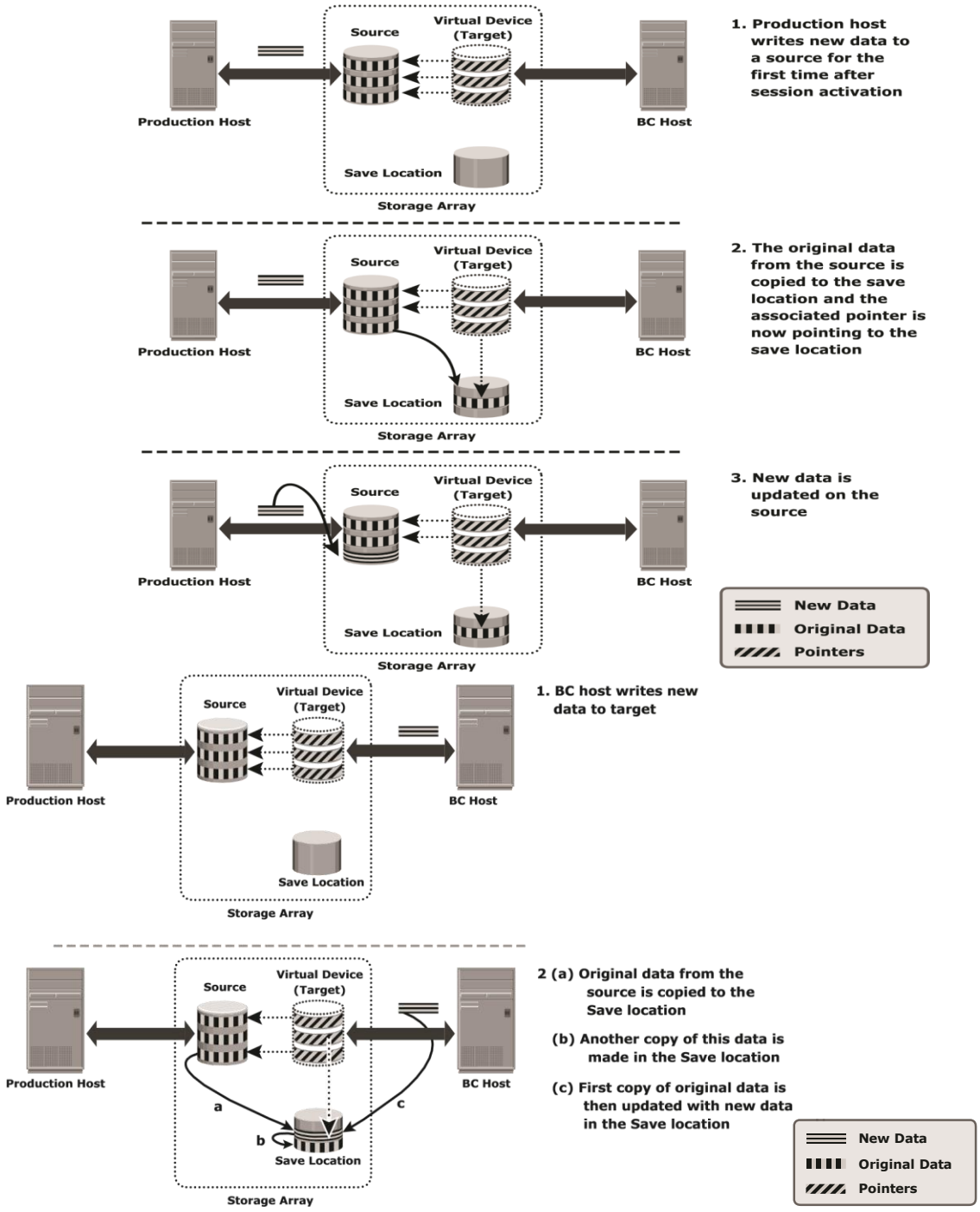
In pointer-based virtual replication, at the time of session activation, the target contains pointers to the location of data on the source. The target does not contain data, at any time. Hence, the target is known as a virtual replica. Similar to pointer-based full-volume replication, a protection bitmap is created for all data on the source device, and the target is immediately accessible. Granularity can range from 512 byte blocks to 64 KB blocks or greater.

When a write is issued to the source for the first time after session activation, original data at that address is copied to a predefined area in the array. This area is generally termed the save location. The pointer in the target is updated to point to this data address in the save location. After this, the new write is updated on the source.

When a write is issued to the target for the first time after session activation, original data is copied from the source to the save location and similarly the pointer is updated to data in save location. Another copy of the original data is created in the save location before the new write is updated on the save location.

When reads are issued to the target, unchanged data blocks since session activation are read from the source. Original data blocks that have changed are read from the save location.

Pointer-based virtual replication uses CoFW technology. Subsequent writes to the same data block on the source or the target do not trigger a copy operation.

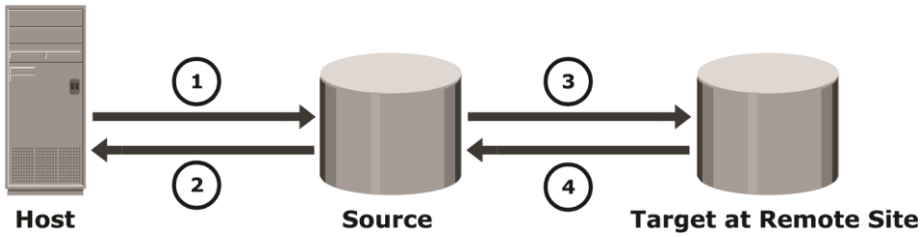


Pointer-based virtual replication — write to target

Data on the target is a combined view of unchanged data on the source and data on the save location. Unavailability of the source device invalidates the data on the target. As the target only contains pointers to data, the physical capacity required for the target is a fraction of the source device. The capacity required for the save location depends on the amount of expected data change.

4.4 Remote Replication Technologies

Remote replication of data can be handled by the hosts or by the storage arrays. Other options include specialized appliances to replicate data over the LAN or the SAN, as well as replication between storage arrays over the SAN.



Asynchronous replication

4.4.1 Host-Based Remote Replication

Host-based remote replication uses one or more components of the host to perform and manage the replication operation. There are two basic approaches to host-based remote replication: LVM-based replication and database replication via log shipping.

LVM-Based Remote Replication

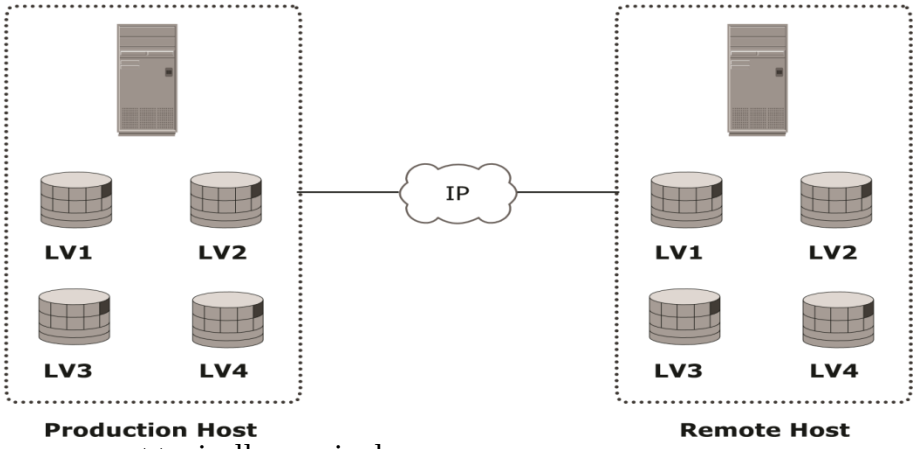
LVM-based replication is performed and managed at the volume group level. Writes to the source volumes are transmitted to the remote host by the LVM. The LVM on the remote host receives the writes and commits them to the remote volume group.

Prior to the start of replication, identical volume groups, logical volumes, and file systems are created at the source and target sites. Initial synchronization of data between the source and the replica can be performed in a number of ways. One method is to backup the source data to tape and restore the data to the remote replica. Alternatively, it can be performed by replicating over the IP network. Until completion of initial synchronization, production work on the source volumes is typically halted. After initial synchronization, production work can be started on the source volumes and replication of data can be performed over an existing standard IP network.

LVM-based remote replication supports both synchronous and asynchronous modes of data transfer. In asynchronous mode, writes are queued in a log file at the source and sent to the re-

mote host in the order in which they were received. The size of the log file determines the RPO at the remote site. In the event of a network failure, writes continue to accumulate in the log file. If the log file fills up before the failure is resolved, then a full resynchronization is required upon network availability. In the event of a failure at the source site, applications can be restarted on the remote host, using the data on the remote replicas.

LVM-based remote replication eliminates the need for a dedicated SAN infrastructure. LVM-based remote replication is independent of the storage arrays and types of disks at the source and remote sites. Most operating systems are shipped with LVMs, so additional licenses and special-



ized hardware are not typically required.

The replication process adds overhead on the host CPUs. CPU resources on the source host LVM-based remote replication are shared between replication tasks and applications, which may cause performance degradation of the application.

As the remote host is also involved in the replication process, it has to be continuously up and available. LVM-based remote replication does not scale well, particularly in the case of applications using federated databases.

Host-Based Log Shipping

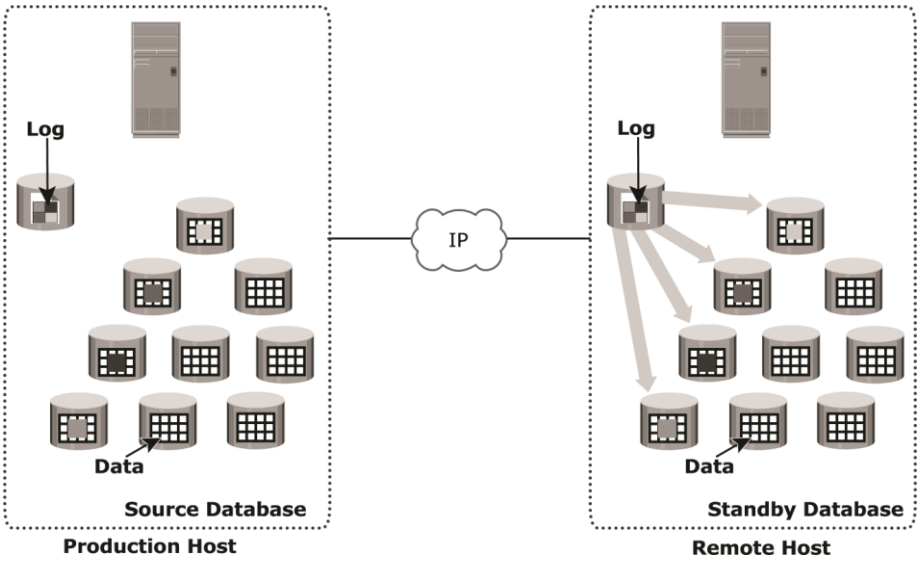
Database replication via log shipping is a host-based replication technology supported by most databases. Transactions to the source database are captured in logs, which are periodically transmitted by the source host to the remote host . The remote host receives the logs and applies them to the remote database.

Prior to starting production work and replication of log files, all relevant components of the source database are replicated to the remote site. This is done while the source database is shut down.

After this step, production work is started on the source database. The remote database is started in a standby mode. Typically, in standby mode, the database is not available for transactions. Some implementations allow reads and writes from the standby database.

All DBMSs switch log files at preconfigured time intervals, or when a log file is full. The current log file is closed at the time of log switching and a new log file is opened. When a log switch occurs, the closed log is transmitted by the source host to the remote host. The remote host receives the log and updates the standby database.

This process ensures that the standby database is consistent up to the last committed log. RPO at the remote site is finite and depends on the size of the log and the frequency of log switching. Available network bandwidth, latency, and rate of updates to the source database, as well as the frequency of log switching, should be considered when determining the optimal size of the log file.



Host-based log shipping

Because the source host does not transmit every update and buffer them, this alleviates the burden on the source host CPU. Similar to LVM-based remote replication, the existing standard IP network can be used for replicating log files. Host-based log shipping does not scale well, particularly in the case of applications using federated databases.

4.4.2 Storage Array-Based Remote Replication

In storage array-based remote replication, the array operating environment and resources perform and manage data replication. This relieves the burden on the host CPUs, which can be better utilized for running an application. A source and its replica device reside on different storage arrays. In other implementations, the storage controller is used for both the host and replication workload. Data can be transmitted from the source storage array to the target storage array over a shared or a dedicated network.

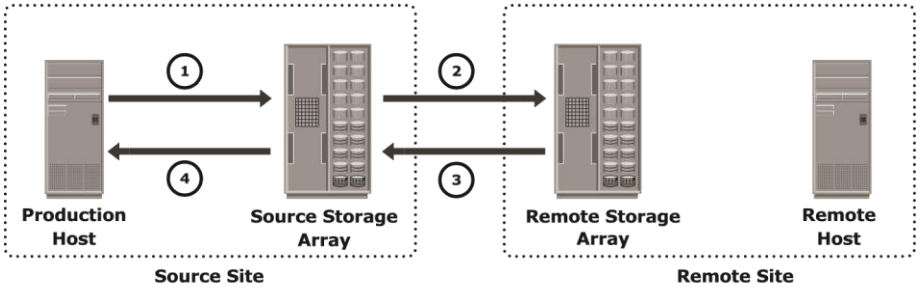
Replication between arrays may be performed in synchronous, asynchronous, or disk-buffered modes. Three-site remote replication can be implemented using a combination of synchronous mode and asynchronous mode, as well as a combination of synchronous mode and disk-buffered mode.

Synchronous Replication Mode

In array based synchronous remote replication, writes must be committed to the source and the target prior to acknowledging "write complete" to the host. Additional writes on that source cannot occur until each preceding write has been completed and acknowledged. The array-based synchronous replication process.

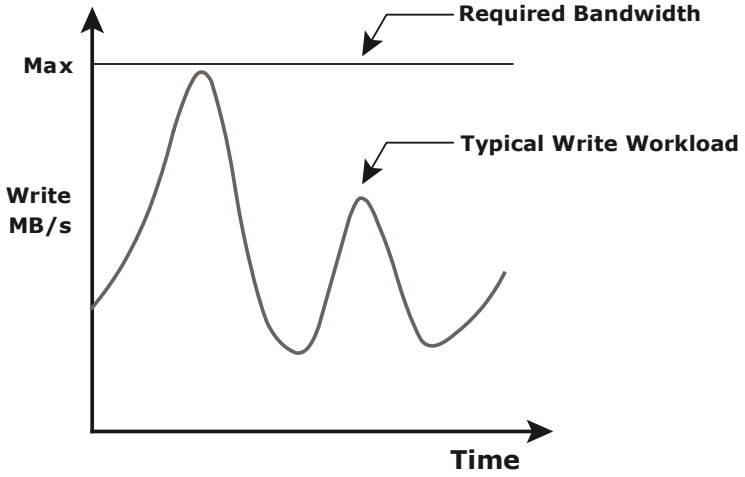
In the case of synchronous replication, to optimize the replication process and to minimize the impact on application response time, the write is placed on cache of the two arrays. The intelligent storage arrays can de-stage these writes to the appropriate disks later.

If the network links fail, replication is suspended; however, production work can continue uninterrupted on the source storage array. The array operating environment can keep track of the



writes that are not transmitted to the remote storage array. When the network links are restored, the accumulated data can be transmitted to the remote storage array. During the time of network link outage, if there is a failure at the source site, some data will be lost and the RPO at the target will not be zero.

For synchronous remote replication, network bandwidth equal to or greater than the maximum write workload between the two sites should be provided at all times. Below graph illustrates the write workload (expressed in MB/s) over time. The "Max" line indicated in Below graph represents the required bandwidth that must be provisioned for synchronous replication. Bandwidths lower than the maximum write workload results in an unacceptable increase in application response time.

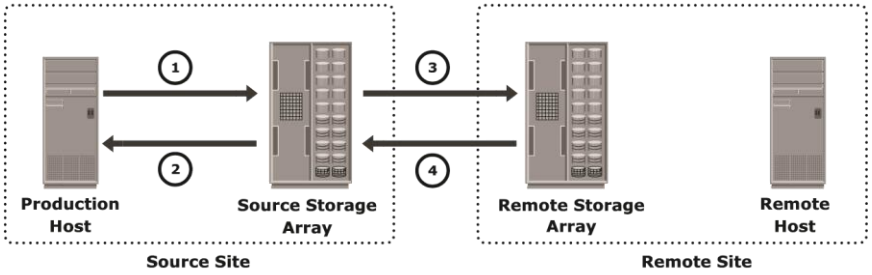


Network bandwidth requirement for synchronous replication

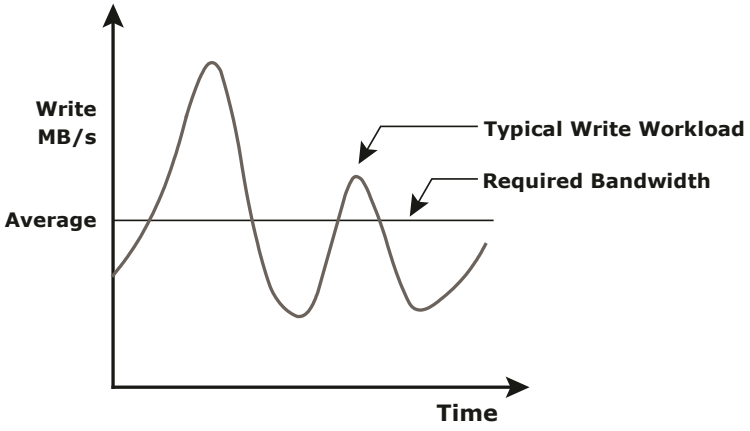
Asynchronous Replication Mode

In array-based asynchronous remote replication mode, a write is committed to the source and immediately acknowledged to the host. Data is buffered at the source and transmitted to the remote site later. The source and the target devices do not contain identical data at all times. The data on the target device is behind that of the source, so the RPO in this case is not zero.

Similar to synchronous replication, asynchronous replication writes are placed in cache on the two arrays and are later de-staged to the appropriate disks.



Some implementations of asynchronous remote replication maintain write ordering. A time stamp and sequence number are attached to each write when it is received by the source. Writes are then transmitted to the remote array, where they are committed to the remote replica in the exact order in which they were buffered at the source. This implicitly guarantees consistency of data on the remote replicas. Other implementations ensure consistency by leveraging the dependent write principle inherent to most DBMSs. The writes are buffered for a predefined period of time. At the end of this duration, the buffer is closed, and a new buffer is opened for subsequent writes. All writes in the closed buffer are transmitted together and committed to the remote replica.



Network bandwidth requirement for asynchronous replication

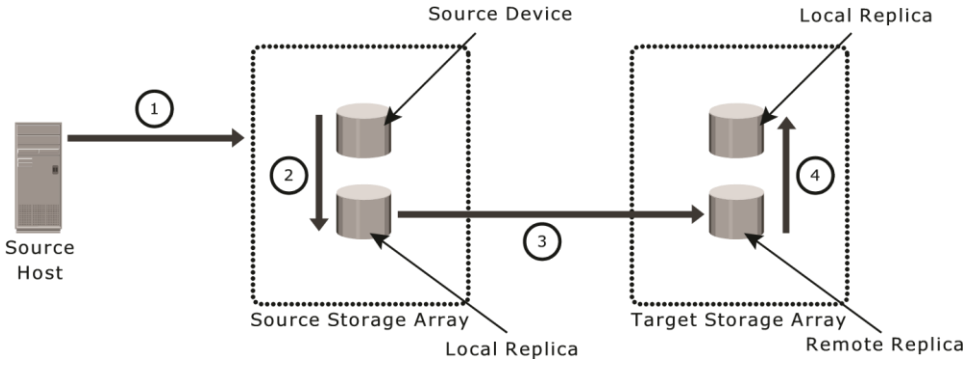
Asynchronous remote replication provides network bandwidth cost savings, as only bandwidth equal to or greater than the average write workload is needed, as represented by the “Average”

line . During times when the write workload exceeds the average bandwidth, sufficient buffer space has to be configured on the source storage array to hold these writes.

Disk-Buffered Replication Mode

Disk-buffered replication is a combination of local and remote replication technologies. A consistent PIT local replica of the source device is first created. This is then replicated to a remote replica on the target array.

At the beginning of the cycle, the network links between the two arrays are suspended and there is no transmission of data. While production application is running on the source device, a consistent PIT local replica of the source device is created. The network links are enabled, and data on the local replica in the source array is transmitted to its remote replica in the target array. After synchronization of this pair, the network link is suspended and the next local replica of the source is created. Optionally, a local PIT replica of the remote device on the target array can be created. The frequency of this cycle of operations depends on available link bandwidth and the data change rate on the source device.



Disk-buffered remote replication

Array-based replication technologies can track changes made to the source and target devices. Hence, all resynchronization operations can be done incrementally.

For example, a local replica of the source device is created at 10:00 am and this data is transmitted to the remote replica, which takes one hour to complete. Changes made to the source device after 10:00 am are tracked. Another replica of the source device is created at 11:00 am by applying track changes between the source and local replica (10:00 am copy). During the next cycle of transmission (11:00 am data), the source data has moved to 12:00 pm The local replica in the remote array has the 10:00 am data until the 11:00 am data is successfully transmitted to the remote replica. If there is a failure at the source site prior to the completion of transmission, then the worst-case RPO at the remote site would be two hours (as the remote site has 10:00 am data).

Three-Site Replication

In synchronous and asynchronous replication, under normal conditions the workload is running at the source site. Operations at the source site will not be disrupted by any failure to the target

site or to the network used for replication. The replication process resumes as soon as the link or target site issues are resolved. The source site continues to operate without any remote protection. If failure occurs at the source site during this time, RPO will be extended.

In synchronous replication, source and target sites are usually within 200 KM (125 miles) of each other. Hence, in the event of a regional disaster, both the source and the target sites could become unavailable. This will lead to extended RPO and RTO because the last known good copy of data would have to come from another source, such as offsite tape library.

A regional disaster will not affect the target site in asynchronous replication, as the sites are typically several hundred or several thousand kilometers apart. If the source site fails, production can be shifted to the target site, but there will be no remote protection until the failure is resolved.

Three-site replication is used to mitigate the risks identified in two-site replication. In a three-site replication, data from the source site is replicated to two remote data centers. Replication can be synchronous to one of the two data centers, providing a zero-RPO solution. It can be asynchronous or disk buffered to the other remote data center, providing a finite RPO. Three-site remote replication can be implemented as a cascade/multi-hop or a triangle/multi-target solution.

Three-Site Replication—Cascade/Multi-hop

In the cascade/multi-hop form of replication, data flows from the source to the intermediate storage array, known as a bunker, in the first hop and then from a bunker to a storage array at a remote site in the second hop. Replication between the source and the bunker occurs synchronously, but replication between the bunker and the remote site can be achieved in two ways: disk-buffered mode or asynchronous mode.

Synchronous + Asynchronous

This method employs a combination of synchronous and asynchronous remote replication technologies. Synchronous replication occurs between the source and the bunker. Asynchronous replication occurs between the bunker and the remote site. The remote replica in the bunker acts as the source for the asynchronous replication to create a remote replica at the remote site.

RPO at the remote site is usually on the order of minutes in this implementation. In this method, a minimum of three storage devices are required (including the source) to replicate one storage device. The devices containing a synchronous remote replica at the bunker and the asynchronous replica at the remote are the other two devices.

If there is a disaster at the source, operations are failed over to the bunker site with zero or near-zero data loss. But unlike the synchronous two-site situation, there is still remote protection at the third site. The RPO between the bunker and third site could be on the order of minutes.

If there is a disaster at the bunker site or if there is a network link failure between the source and bunker sites, the source site will continue to operate as normal but without any remote replication. This situation is very similar to two-site replication when a failure/disaster occurs at the

target site. The updates to the remote site cannot occur due to the failure in the bunker site. Hence, the data at the remote site keeps falling behind, but the advantage here is that if the source fails during this time, operations can be resumed at the remote site. RPO at the remote site depends on the time difference between the bunker site failure and source site failure.

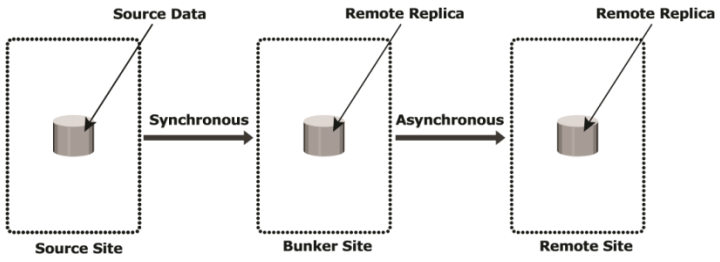
A regional disaster in three-site cascade/multihop replication is very similar to a source site failure in two-site asynchronous replication. Operations will failover to the remote site with an RPO on the order of minutes. There is no remote protection until the regional disaster is resolved. Local replication technologies could be used at the remote site during this time.

If a disaster occurs at the remote site, or if the network links between the bunker and the remote site fail, the source site continues to work as normal with disaster recovery protection provided at the bunker site.

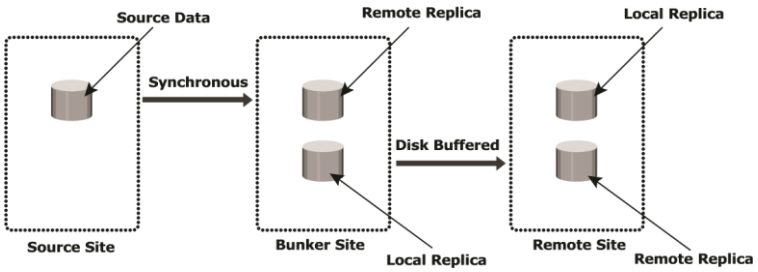
Synchronous + Disk Buffered

This method employs a combination of local and remote replication technologies. Synchronous replication occurs between the source and the bunker: A consistent PIT local replica is created at the bunker. Data is transmitted from the local replica at the bunker to the remote replica at the remote site. Optionally, a local replica can be created at the remote site after data is received from the bunker.

In this method, a minimum of four storage devices are required (including the source) to replicate one storage device. The other three devices are the synchronous remote replica at the bunker, a consistent PIT local replica at the bunker, and the replica at the remote site. RPO at the remote site is usually in the order of hours in this implementation. For example, if a local replica is created at 10:00 am at the bunker and it takes an hour to transmit this data to the remote site, changes made to the remote replica at the bunker since 10:00 am are tracked. Hence only one hour's worth of data has to be resynchronized between the bunker and the remote site during the next cycle. RPO in this case will also be two hours, similar to disk-buffered replication.



(a) **Synchronous + asynchronous**



(b) **Synchronous + disk buffered**

Three-site replication

The process of creating the consistent PIT copy at the bunker and incrementally updating the remote replica and the local replica at the remote site occurs continuously in a cycle. This process can be automated and controlled from the source.

Three-Site Replication—Triangle/Multi-target

In the three-site triangle/multi-target replication, data at the source storage array is concurrently replicated to two different arrays. The source-to-bunker site (target 1) replication is synchronous, with a near-zero RPO. The source-to-remote site (target 2) replication is asynchronous, with an RPO of minutes. The distance between the source and the remote site could be thousands of miles. This configuration does not depend on the bunker site for updating data on the remote site, because data is asynchronously copied to the remote site directly from the source.

The key benefit of three-site triangle/multi-target replication is the ability to failover to either of the two remote sites in the case of source site failure, with disaster recovery (asynchronous) protection between them. Resynchronization between the two surviving target sites is incremental. Disaster recovery protection is always available in the event of any onesite failure.

During normal operations all three sites are available and the workload is at the source site. At any given instant, the data at the bunker and the source is identical. The data at the remote site is behind the data at the source and the bunker. The replication network links between the bunker and remote sites will be in place but not in use. Thus, during normal operations there is no data movement between the bunker and remote arrays. The difference in the data between the bunker and remote sites is tracked, so that in the event of a source site disaster, operations can be resumed at the bunker or the remote sites with incremental resynchronization between the sites.

4.4.3 SAN-Based Remote Replication

SAN-based remote replication enables the replication of data between heterogeneous storage arrays. Data is moved from one array to the other over the SAN/ WAN. This technology is application and server operating system independent, because the replication operations are performed by one of the storage arrays (the control array). There is no impact on production servers (because replication is done by the array) or the LAN (because data is moved over the SAN).

SAN-based remote replication is a point-in-time replication technology. Uses of SAN-based remote replication include data mobility, remote vaulting, and data migration. Data mobility en-

ables incrementally copying multiple volumes over extended distances, as well as implementing a tiered storage strategy. Data vaulting is the practice of storing a set of point-in-time copies on heterogeneous remote arrays to guard against a failure of the source site. Data migration refers to moving data to new storage arrays and consolidating data from multiple heterogeneous storage arrays onto a single storage array.

The array performing the replication operations is called the control array. Data can be moved to/from devices in the control array to/from a remote array. The devices in the control array that are part of the replication session are called control devices. For every control device there is a counterpart, a remote device, on the remote array.

The terms “control” or “remote” do not indicate the direction of data flow, they only indicate the array that is performing the replication operation. Data movement could be from the control array to the remote array or vice versa. The direction of data movement is determined by the replication operation.

The front-end ports of the control array must be zoned to the front-end ports of the remote array. LUN masking should be performed on the remote array to allow access to the remote devices to the front-end port of the control array. In effect, the front-end ports of the control array act as an HBA, initiating data transfer to/from the remote array.

SAN-based replication uses two types of operations: push and pull. These terms are defined from the perspective of the control array. In the push operation, data is transmitted from the control storage array to the remote storage array. The control device, therefore, acts like the source, while the remote device is the target. The data that needs to be replicated would be on devices in the control array.

In the pull operation, data is transmitted from the remote storage array to the control storage array. The remote device is the source and the control device is the target. The data that needs to be replicated would be on devices in the remote array.

When a push or pull operation is initiated, the control array creates a protection bitmap to track the replication process. Each bit in the protection bitmap represents a data chunk on the control device. Chunk size may vary with technology implementations. When the replication operation is initiated, all the bits are set to one, indicating that all the contents of the source device need to be copied to the target device. As the replication process copies data, the bits are changed to zero, indicating that a particular chunk has been copied. At the end of the replication process, all the bits become zero.

During the push and pull operations, host access to the remote device is not allowed because the control storage array has no control over the remote storage array and cannot track any change on the remote device. Data integrity cannot be guaranteed if changes are made to the remote device during the push and pull operations. Therefore, for all SAN-based remote replications, the remote devices should not be in use during the replication process in order to ensure data integrity and consistency.

The push/pull operations can be either hot or cold. These terms apply to the control devices only. In a cold operation, the control device is inaccessible to the host during replication. Cold operations guarantee data consistency because both the control and the remote devices are offline to every host operation. In a hot operation, the control device is online for host operations. With hot operations, changes can be made to the control device during push/pull because the control array can keep track of all changes, and thus ensures data integrity.

When the hot push operation is initiated, applications can be up and running on the control devices. I/O to the control devices is held while the protection bitmap is created. This ensures a consistent PIT image of the data. The protection bitmap is referred prior to any write to the control devices. If the bit is zero, the write is allowed. If the bit is one, the replication process holds the write, copies the required chunk to the remote device, and then allows the write to complete.

In the hot pull operation, the hosts can access control devices after starting the pull operation. The protection bitmap is referenced for every read or write operation. If the bit is zero, a read or write occurs. If the bit is one, the read or write is held, and the replication process copies the required chunk from the remote device. When the chunk is copied, the read or write is completed. The control devices can be used after the pull operation is initiated and as soon as the protection bitmap is created.

In SAN-based replication, the control array can keep track of changes made to the control devices after the replication session is activated. This is allowed in the incremental push operation only. A second bitmap, called a resynchronization bitmap, is created. All the bits in the resynchronization bitmap are set to zero when a push is initiated, as shown in (a). As changes are made to the control device, the bits are flipped from zero to one, indicating that changes have occurred, as shown in (b). When resynchronization is required, the push is reinitiated and the resynchronization bitmap becomes the new protection bitmap, as shown in (c), and only the modified chunks are transmitted to the remote devices. If changes are made to the remote device, the SAN-based replication operation is unaware of these changes, therefore, data integrity cannot be ensured if an incremental push is performed.

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

(a) Resynchronization bitmap when push is initiated

0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0

(b) Resynchronization bitmap when data chunks are updated

0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0

(c) Resynchronization bitmap becomes the protection bitmap

Bitmap status in SAN-based replication

UNIT-5

SECURING STORAGE AND STORAGE VIRTUALIZATION

5.1 Information Storage

The basic security framework is built around the four primary services of security: accountability, confidentiality, integrity,and availability. This framework incorporates all security measures required to mitigate threats to these four primary security attributes:

■ **Accountability service:** Refers to accounting for all the events and operations that takes place in data center infrastructure. The accountability service maintains a log of events that can be audited or traced later for the purpose of security.

■ **Confidentiality service:** Provides the required secrecy of information and ensures that only authorized users have access to data. This service authenticates users who need to access information and typically covers both data in transit (data transmitted over cables), or data at rest (data on a backup media or in the archives).

Data in transit and at rest can be encrypted to maintain its confidentiality. In addition to restricting unauthorized users from accessing information, confidentiality services also implement traffic flow protection measures as part of the security protocol. These protection measures generally

include hiding source and destination addresses, frequency of data being sent, and amount of data sent.

■ **Integrity service:** Ensures that the information is unaltered. The objective of the service is to detect and protect against unauthorized alteration or deletion of information. Similar to confidentiality services, integrity services work in collaboration with accountability services to identify and authenticate the users. Integrity services stipulate measures for both in-transit data and at-rest data.

■**Availability service:** This ensures that authorized users have reliable and timely access to data. These services enable users to access the required computer systems, data, and applications residing on these systems. Availability services are also implemented on communication systems used to transmit information among computers that may reside at different locations. This ensures availability of information if a failure in one particular location occurs. These services must be implemented for both electronic data and physical data.

5.2 Security Attributes For Information

Risk triad defines the risk in terms of threats, assets, and vulnerabilities. Risk arises when a threat agent (an attacker) seeks to access assets by exploiting an existing vulnerability.

To manage risks, organizations primarily focus on vulnerabilities because they cannot eliminate threat agents that may appear in various forms and sources to its assets. Organizations can install countermeasures to reduce the impact of an attack by a threat agent, thereby reducing vulnerability.

Risk assessment is the first step in determining the extent of potential threats and risks in an IT infrastructure. The process assesses risk and helps to identify appropriate controls to mitigate or eliminate risks. To determine the probability of an adverse event occurring, threats to an IT system must be analyzed in conjunction with the potential vulnerabilities and the existing security controls.

The severity of an adverse event is estimated by the impact that it may have on critical business activities. Based on this analysis, a relative value of criticality and sensitivity can be assigned to IT assets and resources. For example, a particular IT system component may be assigned a high-criticality value if an attack on this particular component can cause a complete termination of mission-critical services.

The following sections examine the three key elements of the risk triad. Assets, threats, and vulnerability are considered from the perspective of risk identification and control analysis.

5.2.1 Assets

Information is one of the most important *assets* for any organization. Other assets include hardware, software, and the network infrastructure required to access this information. To protect these assets, organizations must develop a set of parameters to ensure the availability of the resources to authorized users and trusted networks. These parameters apply to storage resources, the network infrastructure, and organizational policies.

Several factors need to be considered when planning for asset security. Security methods have two objectives. First objective is to ensure that the network is easily accessible to authorized users. It should also be reliable and stable under disparate environmental conditions and volumes of usage. Second objective is to make it very difficult for potential attackers to access and compromise the system. These methods should provide adequate protection against unauthorized access to resources, viruses, worms, Trojans and other malicious software programs. Security measures should also encrypt critical data and disable unused services to minimize the number of potential security gaps. The security method must ensure that updates to the operating system and other software are installed regularly. At the same time, it must provide adequate redundancy in the form of replication and mirroring of the production data to prevent catastrophic data loss if there is an unexpected malfunction. In order for the security system to function smoothly, it is important to ensure that all users are informed of the policies governing use of the network.

The effectiveness of a storage security methodology can be measured by two criteria. One, the cost of implementing the system should only be a small fraction of the value of the protected data. Two, it should cost a potential attacker more, in terms of money and time, to compromise the system than the protected data is worth.

5.2.2 Threats

Threats are the potential attacks that can be carried out on an IT infrastructure. These attacks can be classified as active or passive. *Passive* attacks are attempts to gain unauthorized access into the system. They pose threats to confidentiality of information. *Active* attacks include data modification, Denial of Service (DoS), and repudiation attacks. They pose threats to data integrity and availability.

In a *modification* attack, the unauthorized user attempts to modify information for malicious purposes. A modification attack can target data at rest or data in transit. These attacks pose a threat to data integrity.

Denial of Service (DoS) attacks denies the use of resources to legitimate users. These attacks generally do not involve access to or modification of information on the computer system. Instead, they pose a threat to data availability. The intentional flooding of a network or website to prevent legitimate access to authorized users is one example of a DoS attack.

Repudiation is an attack against the accountability of the information. It attempts to provide false information by either impersonating someone or denying that an event or a transaction has taken place.

5.2.3 Vulnerability

The paths that provide access to information are the most vulnerable to potential attacks. Each of these paths may contain various access points, each of which provides different levels of access to the storage resources. It is very important to implement adequate security controls at *all* the access points on an access path. Implementing security controls at each access point of every access path is termed as *defense in depth*.

Defense in depth recommends protecting all access points within an environment. This reduces vulnerability to an attacker who can gain access to storage resources by bypassing inadequate security controls implemented at the vulnerable single point of access. Such an attack can jeopardize the security of information assets. For example, a failure to properly authenticate a user may put the confidentiality of information at risk. Similarly, a DoS attack against a storage device can jeopardize information availability.

Attack surface, *attack vector*, and *work factor* are the three factors to consider when assessing the extent to which an environment is vulnerable to security threats. *Attack surface* refers to the various entry points that an attacker can use to launch an attack. Each component of a storage network is a source of potential vulnerability. All of the external interfaces supported by that component, such as the hardware interfaces, the supported protocols, and the management and administrative interfaces, can be used by an attacker to execute various attacks. These interfaces form the attack surface for the attacker. Even unused network services, if enabled, can become a part of the attack surface.

An *attack vector* is a step or a series of steps necessary to complete an attack. For example, an attacker might exploit a bug in the management interface to execute a snoop attack whereby the attacker can modify the configuration of the storage device to allow the traffic to be accessed from one more host. This redirected traffic can be used to snoop the data in transit.

Work factor refers to the amount of time and effort required to exploit an attack vector. For example, if attackers attempt to retrieve sensitive information, they consider the time and effort that would be required for executing an attack on a database. This may include determining privileged accounts, determining the database schema, and writing SQL queries. Instead, based on the work factor, they consider a less effort-intensive way to exploit the storage array by attaching to it directly and reading from the raw disk blocks.

Having assessed the vulnerability of the network environment to security threats, organizations can plan and deploy specific control measures directed at reducing vulnerability by minimizing attack surfaces and maximizing the work factor. These controls can be technical or nontechnical. Technical controls are usually implemented through computer systems, whereas nontechnical controls are implemented through administrative and physical controls. Administrative controls include security and personnel policies or standard procedures to direct the safe execution of var-

ious operations. Physical controls include setting up physical barriers, such as security guards, fences, or locks.

Based on the roles they play, controls can be categorized as preventive, detective, corrective, recovering, or compensating. The discussion here focuses on preventive, corrective, and detective controls only. The preventive control attempts to prevent an attack; the detective control detects whether an attack is in progress; and after an attack is discovered, the corrective controls are implemented. *Preventive* controls avert the vulnerabilities from being exploited and prevent an attack or reduce its impact. *Corrective* controls reduce the effect of an attack, while *detective* controls discover attacks and trigger preventive or corrective controls. For example, an Intrusion Detection/Intrusion Prevention System (IDS/IPS) is a detective control that determines whether an attack is underway and then attempts to stop it by terminating a network connection or invoking a firewall rule to block traffic.

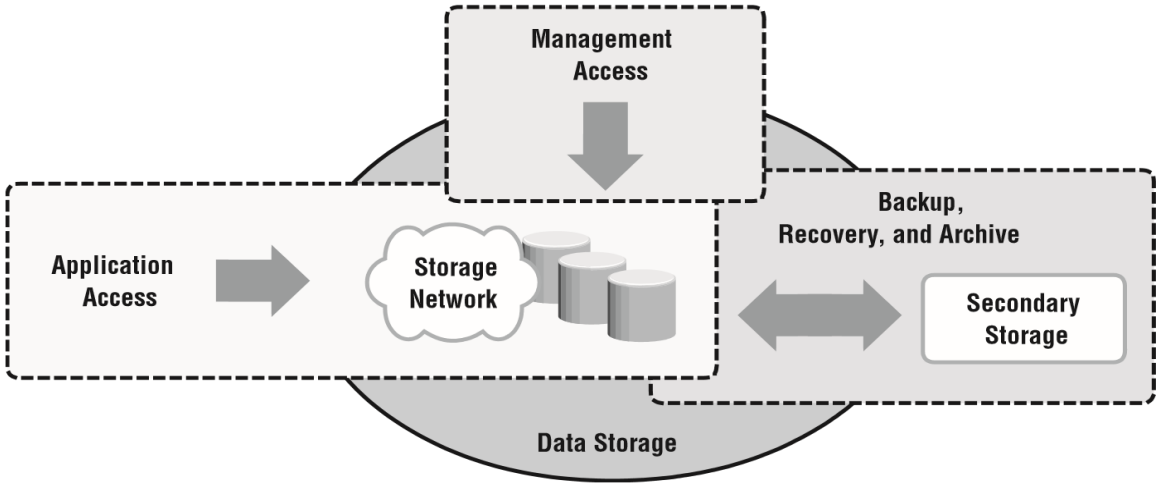
5.3 Storage Security Domains

Storage devices that are not connected to a storage network are less vulnerable because they are not exposed to security threats via networks. However, with increasing use of networking in storage environments, storage devices are becoming highly exposed to security threats from a variety of sources. Specific controls must be implemented to secure a storage networking environment. This requires a closer look at storage networking security and a clear understanding of the access paths leading to storage resources. If a particular path is unauthorized and needs to be prohibited by technical controls, one must ensure that these controls are not compromised. If each component within the storage network is considered a potential access point, one must analyze the attack surface that each of these access points provides and identify the associated vulnerability.

In order to identify the threats that apply to a storage network, access paths to data storage can be categorized into three security domains: *application access*, *management access*, and *BURA (backup, recovery, and archive)*. Figure 15-1 depicts the three security domains of a storage system environment.

The first security domain involves application access to the stored data through the storage network. The second security domain includes management access to storage and interconnect devices and to the data residing on those devices. This domain is primarily accessed by storage administrators who configure and manage the environment. The third domain consists of BURA access. Along with the access points in the other two domains, backup media also needs to be secured.

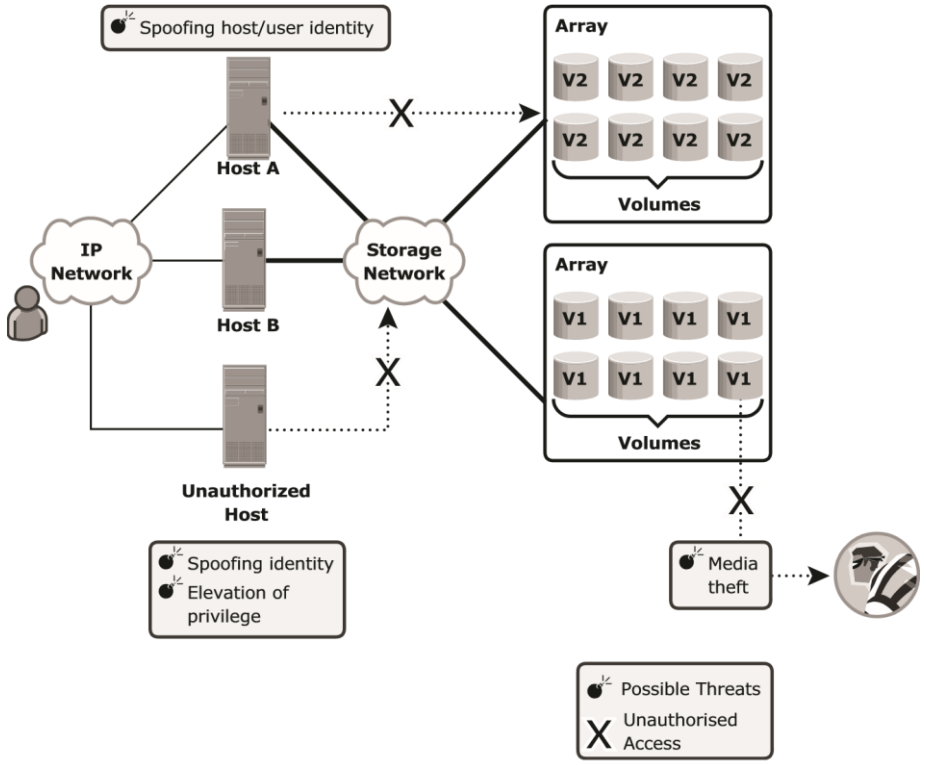
To secure the storage networking environment, identify the existing threats within each of the security domains and classify the threats based on the type of security services—availability, confidentiality, integrity, and accountability. The next step is to select and implement various controls as countermeasures to the threats.



5.3.1 Securing the Application Access Domain

The application access domain may include only those applications that access the data through the file system or a database interface.

Host A can access all V1 volumes; host B can access all V2 volumes. These volumes are classified according to access level, such as confidential, restricted, and public. Some of the possible threat in this scenario could be host A spoofing the identity or elevating the privileges of host B to gain access to host B's resources. Another threat could be an unauthorized host gain access to the network; the attacker on this host may try to spoof the identity of another host and tamper with data, snoop the network, or execute a DoS attack. Also any form of media theft could also compromise security. These threats can pose several serious challenges to the network security, hence they need to be addressed.



An important step for securing the application access domain is to identify the core functions that can prevent these threats from being exploited and to identify the appropriate controls that should be applied. Implementing physical security is also an important consideration to prevent media theft.

Controlling User Access to Data

Access control services regulate user access to data. These services mitigate the threats of spoofing host identity and elevating host privileges. Both of these threats affect data integrity and confidentiality.

Technical control in the form of user authentication and administrative control in the form of user authorization are the two access control mechanisms used in application access control. These mechanisms may lie outside the boundaries of the storage network and require various systems to interconnect with other enterprise identity management and authentication systems—for example, systems that provide strong authentication and authorization to secure user identities against spoofing. NAS devices support the creation of *access control lists* that are used to regulate user access to specific files. The Enterprise Content Management application enforces access to data by using Information Rights Management (IRM) that specify which users have what rights to a document. Restricting access at the host level starts with authenticating a node when it tries to connect to a network. Different storage networking technologies, such as iSCSI, FC, and IP-based storage, use various authentication mechanisms, such as Challenge-Handshake Authentication Protocol (CHAP), Fibre Channel Security Protocol (FC-SP), and IPSec, respectively, to authenticate host access.

After a host has been authenticated, the next step is to specify security controls for the storage resources, such as ports, volumes, or storage pools, that the host is authorized to access. *Zoning* is a control mechanism on the switches that segments the network into specific paths to be used for data traffic; *LUN masking* determines which hosts can access which storage devices. Some devices support mapping of a host's WWN to a particular FC port, and from there to a particular LUN. This binding of the WWN to a physical port is the most secure.

Finally, it is very important to ensure that administrative controls, such as defined policies and standards, are implemented. Regular auditing is required to ensure proper functioning of administrative controls. This is enabled by logging significant events on all participating devices. Event logging must be protected from unauthorized access because it may fail to achieve its goals if the logged content is exposed to unwanted modifications by an attacker.

Protecting the Storage Infrastructure

Securing the storage infrastructure from unauthorized access involves protecting all the elements of the infrastructure. Security controls for protecting the storage infrastructure address the threats of unauthorized tampering of data in transit that leads to a loss of data integrity, denial of service that compromises availability, and network snooping that may result in a loss of confidentiality.

The security controls for protecting the network fall into two general categories: *connectivity infrastructure integrity* and *storage network encryption*. Controls for ensuring the infrastructure integrity include a fabric switch function to ensure fabric integrity. This is achieved by preventing a host from being added to the SAN fabric without proper authorization. Storage network encryption methods include the use of IPSec, for protecting IP-based storage networks, and FC-SP, for protecting FC networks.

In secure storage environments, root or administrator privileges for a specific device are not granted to any individual. Instead, *role-based access control (RBAC)* is deployed to assign necessary privileges to users, enabling them to perform their roles. It is also advisable to consider administrative controls, such as "separation of duties," when defining data center procedures. Clear separation of duties ensures that no single individual is able to both specify an action and carry it out. For example, the person who authorizes the creation of administrative accounts should not be the person who uses those accounts. Securing management access is covered in detail in the next section.

Management networks for storage systems should be logically separate from other enterprise networks. This segmentation is critical to facilitate ease of management and increase security by allowing access only to the components existing within the same segment. For example, IP network segmentation is enforced with the deployment of filters at layer 3 by using routers and firewalls, as well as at layer 2 by using VLANs and port-level security on Ethernet switches.

Finally, physical access to the device console and the cabling of FC switches must be controlled to ensure protection of the storage infrastructure. All other established security measures fail if a device is physically accessed by an unauthorized user; the mere fact of this access may render the device unreliable.

Data Encryption

The most important aspect of securing data is protecting data held inside the storage arrays. Threats at this level include tampering with data, which violates data integrity, and media theft, which compromises data availability and confidentiality. To protect against these threats, encrypt the data held on the storage media or encrypt the data prior to being transferred to the disk. It is also critical to decide upon a method for ensuring that data deleted at the end of its lifecycle has been completely erased from the disks and cannot be reconstructed for malicious purposes.

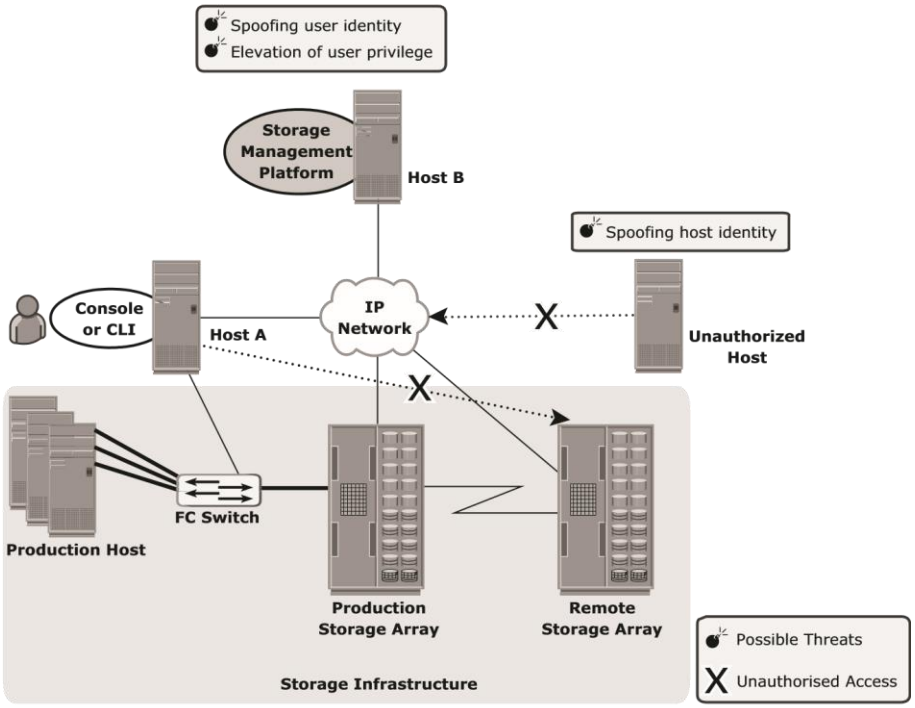
Data should be encrypted as close to its origin as possible. If it is not possible to perform encryption on the host device, an encryption appliance can be used for encrypting data at the point of entry into the storage network. Encryption devices can be implemented on the fabric that encrypts data between the host and the storage media. These mechanisms can protect both the data at rest on the destination device and data in transit.

On NAS devices, adding antivirus checks and file extension controls can further enhance data integrity. In the case of CAS, use of MD5 or SHA-256 cryptographic algorithms guarantee data integrity by detecting any change in content bit patterns. In addition, the CAS data erasure service ensures that the data has been completely scrubbed from the disk before the disk is discarded. An organization's data classification policy determines whether the disk should actually be scrubbed prior to discarding it as well as the level of erasure needed based on regulatory requirements.

5.3.2 Securing the Management Access Domain

Management access, whether monitoring, provisioning, or managing storage resources, is associated with every device within the storage network. Most management software supports some form of CLI, system management console, or a web-based interface. It is very important to implement appropriate controls for securing storage management applications because the damage that can be caused to the storage system by using these applications can be far more extensive than that caused by vulnerability in a server.

Further, this configuration has a storage management platform on Host B and a monitoring console on Host A. All these hosts are interconnected through an IP network. Some of the possible threats in this system are, unauthorized host may spoof the user or host identity to manage the storage arrays or network. For example, Host A may gain management access to array B. Remote console support for the management software also increases the attack surface. Using remote console support, several other systems in the network may also be used to execute an attack.



Providing management access through an external network increases the potential for an unauthorized host or switch to connect to that network. In such circumstances, implementing appropriate security measures prevents certain types of remote communication from occurring. Using secure communication channels, such as Secure Shell (SSH) or Secure Sockets Layer (SSL)/ Transport Layer Security (TLS), provides effective protection against these threats. Event log monitoring helps to identify unauthorized access and unauthorized changes to the infrastructure.

The storage management platform must be validated for available security controls and ensures that these controls are adequate to secure the overall storage environment. The administrator's identity and role should be secured against any spoofing attempts so an attacker cannot manipulate the entire storage array and cause intolerable data loss by reformatting storage media or making data resources unavailable.

Controlling Administrative Access

Controlling administrative access to storage aims to safeguard against the threats of an attacker spoofing an administrator's identity or elevating another user's identity and privileges to gain administrative access. Both of these threats affect the integrity of data and devices. To protect against these threats, administrative access regulation and various auditing techniques are used to enforce accountability. Every storage component should provide access control. In some storage environments, it may be necessary to integrate storage devices with third-party authentication directories, such as *Lightweight Directory Access Protocol (LDAP)* or Active Directory.

Security best practices stipulate that no single user should have ultimate control over all aspects of the system. If an administrative user is a necessity, the number of activities requiring administrative privileges should be minimized. Instead, it is better to assign various administra-

tive functions by using RBAC. Auditing logged events is a critical control measure to track the activities of an administrator. However, access to administrative log files as well as their content must be protected. Deploying a reliable *Network Time Protocol* on each system that can be synchronized to a common time is another important requirement to ensure that activities across systems can be consistently tracked. In addition, having a Security Information Management (SIM) solution supports effective analysis of the event log files.

Protecting the Management Infrastructure

Protecting the management network infrastructure is also necessary. Controls to protect the management network infrastructure include encrypting management traffic, enforcing management access controls, and applying IP network security best practices. These best practices include the use of IP routers and Ethernet switches to restrict traffic to certain devices and management protocols. At the IP network layer, restricting network activity and access to a limited set of hosts minimizes the threat of an unauthorized device attaching to the network and gaining access to the management interfaces of all devices within the storage network. Access controls need to be enforced at the storage-array level to specify which host has management access to which array. Some storage devices and switches can restrict management access to particular hosts and limit commands that can be issued from each host.

A separate private management network must be created for management traffic. If possible, management traffic should not be mixed with either production data traffic or other LAN traffic used in the enterprise. Restricting traffic makes it easy for IDS to determine whether there is unauthorized traffic on the network segment. Unused network services must be disabled on every device within the storage network. This decreases the attack surface for that device by minimizing the number of interfaces through which the device can be accessed.

To summarize, security enforcement must focus on the management communications between devices, confidentiality and integrity of management data, and availability of management networks and devices.

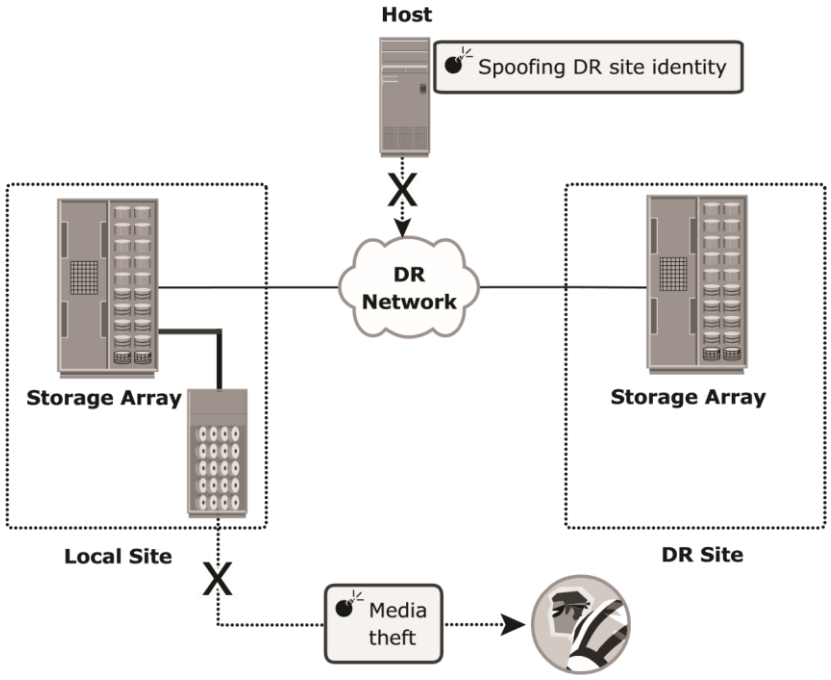
5.3.3 Securing Backup, Recovery, and Archive (BURA)

BURA is the third domain that needs to be secured against attack. A backup involves copying the data from a storage array to backup media, such as tapes or disks. Securing BURA is complex and is based on the BURA software accessing the storage arrays. It also depends on the configuration of the storage environments at the primary and secondary sites, especially with remote backup solutions performed directly on a remote tape device or using array-based remote replication.

Organizations must ensure that the DR site maintains the same level of security for the backed up data. Protecting the BURA infrastructure requires addressing several threats, including spoofing the legitimate identity of a DR site, tampering with data, network snooping, DoS attacks, and media theft. Such threats represent potential violations of integrity, confidentiality, and availability. In a remote backup solution where the storage components are separated by a network, the threats at the transmission layer need to be countered. Otherwise, an attacker can spoof the iden-

tity of the backup server and request the host to send its data. The unauthorized host claims to be the backup server at the DR site, which may lead to a remote backup being performed to an unauthorized and unknown site. In addition, attackers can use the connection to the DR network to tamper with data, snoop the network for authentication data, and create a DoS attack against the storage devices.

The physical threat of a backup tape being lost, stolen, or misplaced, especially if the tapes contain highly confidential information, is another type of threat. Backup-to-tape applications are vulnerable to severe security implications if they do not encrypt data while backing it up.



5.4 Security Implementations in Storage Networking

The following discussion details some of the basic security implementations in SAN, NAS, and IP-SAN environments.

5.4.1 SAN

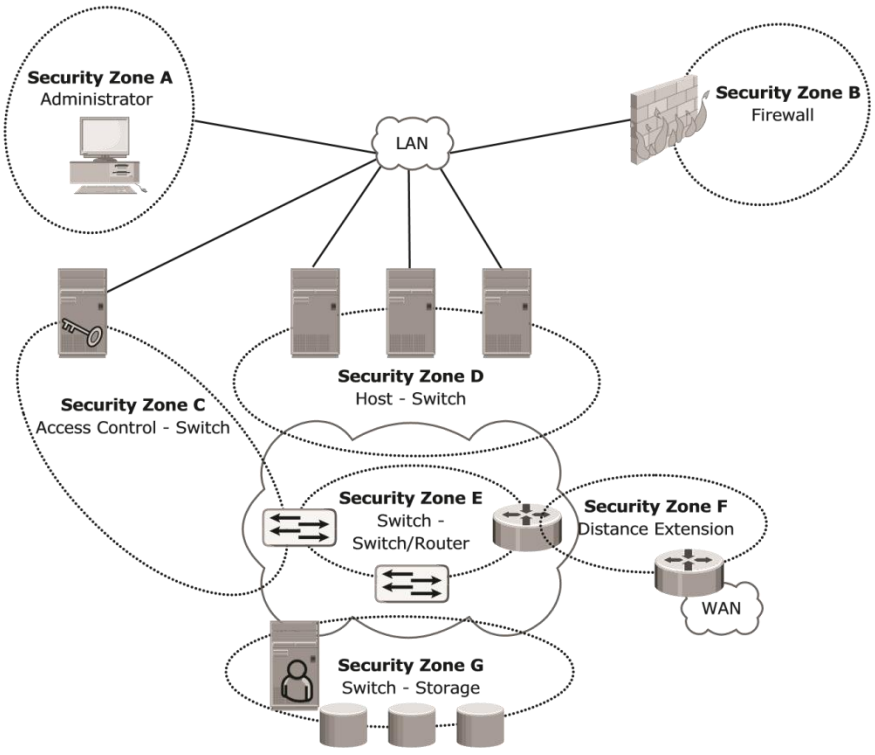
Traditional FC SANs enjoy a natural security advantage over IP-based networks. An FC SAN is configured as an isolated private environment with fewer nodes than an IP network. Consequently, FC SANs impose fewer security threats. However, this scenario has changed with storage consolidation, driving rapid growth and necessitating designs for large, complex SANs that span multiple sites across the enterprise. Today, no single comprehensive security solution is available

for SANs. Many SAN security mechanisms have evolved from their counterpart in IP networking, thereby bringing in mature security solutions.

FC-SP (Fibre Channel Security Protocol) standards (T11 standards), published in 2006, align security mechanisms and algorithms between IP and FC interconnects. These standards describe protocols used to implement security measures in an FC fabric, among fabric elements and N_Ports within the fabric. They also include guidelines for authenticating FC entities, setting up session keys, negotiating the parameters required to ensure frame-by-frame integrity and confidentiality, and establishing and distributing policies across an FC fabric. The current version of the FC-SP standard is referred to as FC-SP-1.

SAN Security Architecture

Storage networking environments are a potential target for unauthorized access, theft, and misuse because of the vastness and complexity of these environments. Therefore, security strategies are based on the *defense in depth* concept, which recommends multiple integrated layers of security. This ensures that the failure of one security control will not compromise the assets under protection. Figure 15-5 illustrates various levels (zones) of a storage networking environment that must be secured and the security measures that can be deployed.



SANs not only suffer from certain risks and vulnerabilities that are unique, but also share common security problems associated with physical security and remote administrative access. In addition to implementing SAN-specific security measures, organizations must simultaneously leverage other security implementations in the enterprise. Table provides a comprehensive list of protection strategies that must be implemented in various security zones. Note that some of the

security mechanisms listed in Table are not specific to SAN, but are commonly used data center techniques. For example, two-factor authentication is implemented widely; in a simple implementation it requires the use of a user name/password and an additional security component such as a smart card for authentication.

Table: Security Zones and Protection Strategies

SeCurITyZoneS	proTeCTIonSTraTegIeS
Zone a (Authentication at the Management Console)	(a) Restrict management LAN access to authorized users (lock down MAC addresses) (b) Implement VPN tunneling for secure remote access to the management LAN (c) Use two-factor authentication for network access
Zone b (Firewall)	Block inappropriate or dangerous traffic by: (a) Filtering out addresses that should not be allowed on your LAN (b) Screening for allowable protocols—block wellknown ports that are not in use
Zone C (Access Control Switch)	Authenticate users/administrators of FC switches using RADIUS (Remote Authentication Dial In User Service), DH-CHAP (Diffie-Hellman Challenge Handshake Authentication Protocol), etc.
Zone d(ACL and Zoning)	Restrict FC access to legitimate hosts by: (a) Implementing ACLs: Known HBAs can connect on specific switch ports only (b) Implementing a secure zoning method such as port zoning (also known as hard zoning)
Zone e (Switch to Switch/ Switch to Router)	Protect traffic on your fabric by: (a) Using E_Port authentication (b) Encrypting the traffic in transit (c) Implementing FC switch controls and port controls
Zone f (Distance Extension)	Implement encryption for in-flight data: (a) FCsec for long-distance FC extension (b) IPSec for SAN extension via FCIP
Zone g (Switch-Storage)	Protect the storage arrays on your SAN via: (a) WWPN-based LUN masking (b) S_ID locking: Masking based on source FCID (Fibre Channel ID/Address)

Basic SAN Security Mechanisms

LUN masking and zoning, switch-wide and fabric-wide access control, RBAC, and logical partitioning of a fabric (Virtual SAN) are the most commonly used SAN security methods.

LUN Masking and Zoning

LUN masking and zoning are the basic SAN security mechanisms used to protect against unauthorized access to storage. Standard implementations of storage arrays mask the LUNs that are presented to a front-end storage port, based on the WWPNs of the source HBAs. A stronger vari-

ant of LUN masking may sometimes be offered whereby masking can be done on the basis of source FCIDs. Note that the FCID typically changes if the HBA is relocated across ports in the fabric. To avoid this problem, major switch vendors offer a mechanism to lock down the FCID of a given node port regardless of its location.

Hard zoning or *port zoning* is the mechanism of choice in security-conscious environments. Unlike soft zoning or WWPN zoning, it actually filters frames to ensure that only authorized zone members can communicate. However, it lacks one significant advantage of WWPN zoning: The zoning configuration must change if the source or the target is relocated across ports in the fabric. There is a trade-off between ease of management and the security provided by WWPN zoning and port zoning.

Apart from zoning and LUN masking, additional security mechanisms such as port binding, port lockdown, port lockout, and persistent port disable can be implemented on switch ports. *Port binding* limits the number of devices that can attach to a particular switch port and allows only the corresponding switch port to connect to a node for fabric access. Port binding mitigates but does not eliminate WWPN spoofing. *Port lockdown* and *port lockout*restrict a switch port's type of initialization. Typical variants of port lockout ensure that the switch port cannot function as an E_Port and cannot be used to create an ISL, such as a rogue switch. Some variants ensure that the port role is restricted to only FL_Port, F_Port, E_Port, or a combination of these. *Persistent port disable*prevents a switch port from being enabled even after a switch reboot.

Switch-wide and Fabric-wide Access Control

As organizations grow their SANs locally or over longer distances there is a greater need to effectively manage SAN security. Network security can be configured on the FC switch by using *access control lists (ACLs)* and on the fabric by using fabric binding.

ACLs incorporate the device connection control and switch connection control policies. The device connection control policy specifies which HBAs and storage ports can be a part of the fabric, preventing unauthorized devices (identified by WWPNs) from accessing it. Similarly, the switch connection control policy specifies which switches are allowed to be part of the fabric, preventing unauthorized switches (identified by WWNs) from joining it.

Fabric binding prevents an unauthorized switch from joining any existing switch in the fabric. It ensures that authorized membership data exists on every switch and that any attempt to connect two switches by using an ISL causes the fabric to segment.

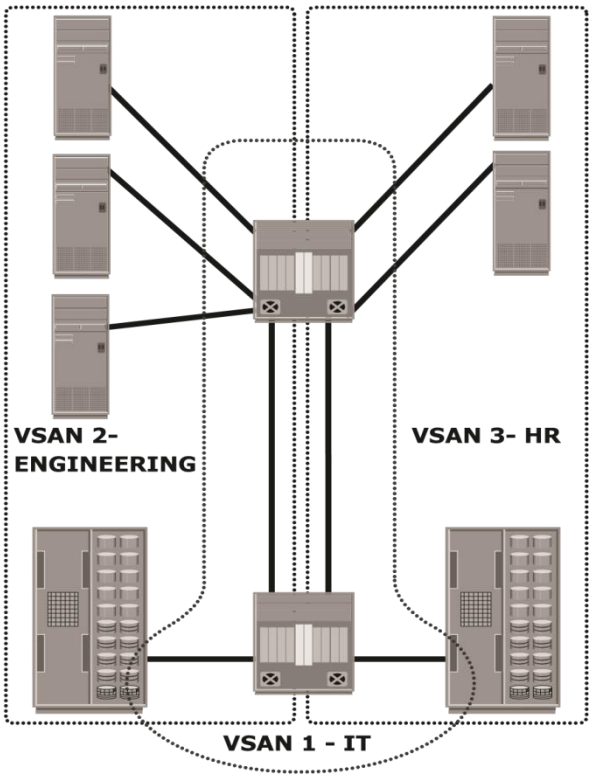
Role-based access control provides additional security to a SAN by preventing unauthorized management activity on the fabric for management operations. It enables the security administrator to assign roles to users that explicitly specify privileges or access rights after logging into the fabric. For example, the *zone admin* role is able to modify the zones on the fabric, whereas a basic user may only be able to view fabric-related information, such as port types and logged-in nodes.

Logical Partitioning of a Fabric: Virtual SAN

VSANs enable the creation of multiple logical SANs over a common physical SAN. They provide the capability to build larger consolidated fabrics and still maintain the required security and isolation between them.

Zoning should be done for each VSAN to secure the entire physical SAN. Each managed VSAN can have only one active zone set at a time. As depicted in the figure, VSAN 1 is the active zone set. The SAN administrator can create distinct VSANs other than VSAN 1 and populate each of them with switch ports. In the example, the switch ports are distributed over three VSANs: 1, 2, and 3—for the IT, Engineering, and HR divisions, respectively. A zone set is defined for each VSAN, providing connectivity for HBAs and storage ports logged into the VSAN. Therefore, each of the three divisions—Engineering, IT, and HR—has its own logical fabric. Although they share physical switching gear with other divisions, they can be managed individually as stand-alone fabrics.

VSANs minimize the impact of fabric wide disruptive events because management and control traffic on the SAN—which may include RSCNs, zone set activation events, and more—does not traverse VSAN boundaries. Therefore, VSANs are a cost-effective alternative for building isolated physical fabrics. They contribute to information availability and security by isolating fabric events and providing a finer degree of authorization control within a single fabric.



Securing SAN with VSAN

5.4.2 NAS

NAS is open to multiple exploits, including viruses, worms, unauthorized access, snooping, and data tampering. Various security mechanisms are implemented in NAS to secure data and the storage networking infrastructure. Permissions and ACLs form the first level of protection to NAS resources by restricting accessibility and sharing. These permissions are deployed over and above the default behaviors and attributes associated with files and folders. In addition, various other authentication and authorization mechanisms, such as Kerberos and directory services, are implemented to verify the identity of network users and define their privileges. Similarly, firewalls are used to protect the storage infrastructure from unauthorized access and malicious attacks.

NAS File Sharing: Windows ACLs

Windows supports two types of ACLs: *discretionary access control lists (DACLs)* and *system access control lists (SACLs)*. The DACL, commonly referred to as the ACL, is used to determine access control. The SACL determines what accesses need to be audited if auditing is enabled.

In addition to these ACLs, Windows also supports the concept of object ownership. The owner of an object has hard-coded rights to that object, and these rights do not have to be explicitly granted in the SACL. The owner, SACL, and DACL are all statically held as an attribute of each object. Windows also offers the functionality to inherit permissions, which allows the child objects existing within a parent object to automatically inherit the ACLs of the parent object.

ACLs are also applied to directory objects known as SIDs. These are automatically generated by a Windows server or domain when a user or group is created, and they are abstracted from the user. In this way, though a user may identify his or her login ID as "User1," it is simply a textual representation of the true SID, which is used by the underlying operating system. ACLs are set by using the standard Windows Explorer GUI, but can also be configured with CLI commands or other third-party tools.

NAS File Sharing: UNIX Permissions

For the UNIX operating system, a *user* is an abstraction that denotes a logical entity for assignment of ownership and operation privileges for the system. A user can be either a person or a system operation. A UNIX system is only aware of the privileges of the user to perform specific operations on the system, and identifies each user by a user ID (UID) and a user name, regardless of whether it is a person, a system operation, or a device.

In UNIX, a user can be organized into one or more groups. The concept of groups serves the purpose of assigning sets of privileges for a given resource and sharing them among many users that need them. For example, a group of people working on one project may need the same permissions for a set of files.

UNIX permissions specify the operations that can be performed by any ownership relation with respect to a file. In simpler terms, these permissions specify what the owner can do, what the owner group can do, and what everyone else can do with the file. For any given ownership relation, three bits are used to specify access permissions. The first bit denotes read (r) access,

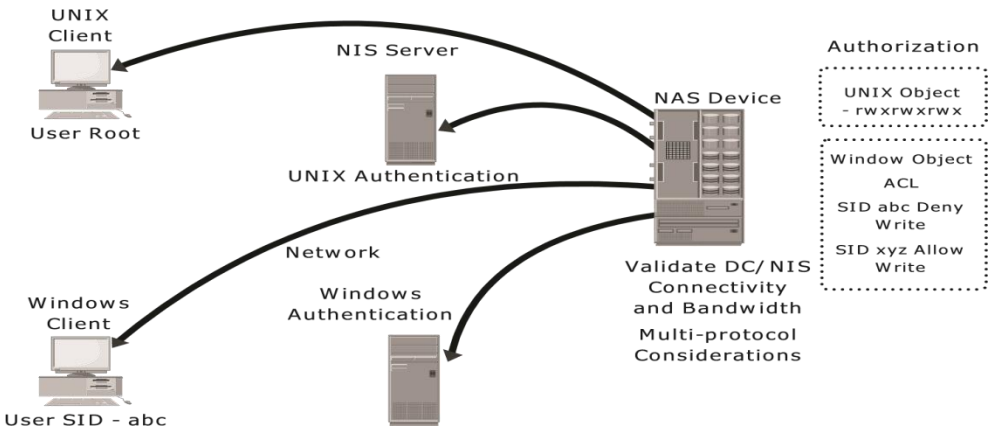
the second bit denotes write (w) access, and the third bit denotes execute (x) access. As UNIX defines three ownership relations, (Owner, Group, and All) a triplet (defining the access permission) is required for each ownership relationship, resulting in nine bits. Each bit can be either set or clear. When displayed, a set bit is marked by its corresponding operation letter (r, w, or x), a clear bit is denoted by a dash (-), and all are put in a row, such as rwxr-xr-x. In this example, the owner can do anything with the file, but group owners and the rest of the world can only read or execute.

When displayed, a character denoting the mode of the file may precede this ninebit pattern. For example, if the file is a directory, it is denoted as "d"; and if it is a link, it is denoted as "l."

Authentication and Authorization

In a file-sharing environment, NAS devices use standard file-sharing protocols, NFS and CIFS. Therefore, authentication and authorization are implemented and supported on NAS devices in the same way as in a UNIX or Windows file sharing environment.

Authentication requires verifying the identity of a network user and therefore involves a login credential lookup on a Network Information System (NIS) server in a UNIX environment. Similarly, a Windows client is authenticated by a Windows domain controller that houses the Active Directory. The Active Directory uses LDAP to access information about network objects in the directory, and Kerberos for network security. NAS devices use the same authentication techniques to validate network user credentials.



Securing user access in a NAS environment

Authorization defines user privileges in a network. The authorization techniques for UNIX users and Windows users are quite different. UNIX files use mode bits to define access rights granted to owners, groups, and other users, whereas Windows uses an ACL to allow or deny specific rights to a particular user for a particular file.

Although NAS devices support both of these methodologies for UNIX and Windows users, complexities arise when UNIX and Windows users access and share the same data. If the NAS device supports multiple protocols, the integrity of both permission methodologies must be

maintained. NAS device vendors provide a method of mapping UNIX permissions to Windows and vice versa, so a multiprotocol environment can be supported. However, it is important to examine these complexities of multiprotocol support when designing a NAS solution. At the same time, it is important to validate the domain controller and/ or NIS server connectivity and bandwidth. If multiprotocol access is required, specific vendor access policy implementations need to be considered. Additional care should be taken to understand the resulting access rights for data being accessed by NFS and CIFS because the access techniques for Windows and UNIX are quite different.

Kerberos

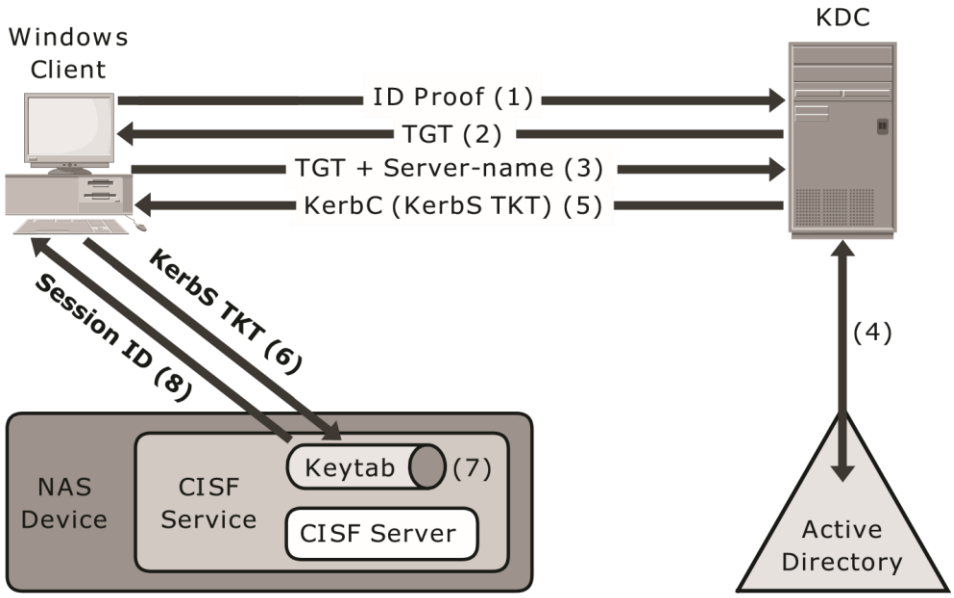
Kerberos is a network authentication protocol. It is designed to provide strong authentication for client/server applications by using secret-key cryptography. It uses cryptography so that a client and server can prove their identity to each other across an insecure network connection. After the client and server have proven their identity, they can choose to encrypt all of their communications to ensure privacy and data integrity.

In Kerberos, all authentications occur between clients and servers. The client gets a ticket for a service, and the server decrypts this ticket by using its secret key. Any entity, user, or host that gets a service ticket for a Kerberos service is called a *Kerberos client*. The term *Kerberos server* generally refers to the Key Distribution Center (KDC). The KDC implements the Authentication Service (AS) and the Ticket Granting Service (TGS). The KDC has a copy of every password associated with every principal, so it is absolutely vital that the KDC remain secure.

In a NAS environment, Kerberos is primarily used when authenticating against a Microsoft Active Directory domain although it can be used to execute security functions in UNIX environments. The Kerberos authorization process shown in Figure 15-8 includes the following steps:

1. The user logs on to the workstation in the Active Directory domain (or forest) using an ID and a password. The client computer sends a request to the AS running on the KDC for a Kerberos ticket. The KDC verifies the user's login information from Active Directory. (Note that this step is not explicitly shown in Figure 15-8.)
2. The KDC responds with a TGT (TKT is a key used for identification and has limited validity period). It contains two parts, one decryptable by the client and the other by the KDC.
3. When the client requests a service from a server, it sends a request, consist of the previously generated TGT and the resource information, to the KDC.
4. The KDC checks the permissions in Active Directory and ensures that the user is authorized to use that service.
5. The KDC returns a service ticket to the client. This service ticket contains fields addressed to the client and to the server that is hosting the service.
6. The client then sends the service ticket to the server that houses the desired resources.
7. The server, in this case the NAS device, decrypts the server portion of the ticket and stores the information in a keytab file. As long as the client's Kerberos ticket is valid, this authorization process does not need to be repeated. The server automatically allows the client to access the appropriate resources.

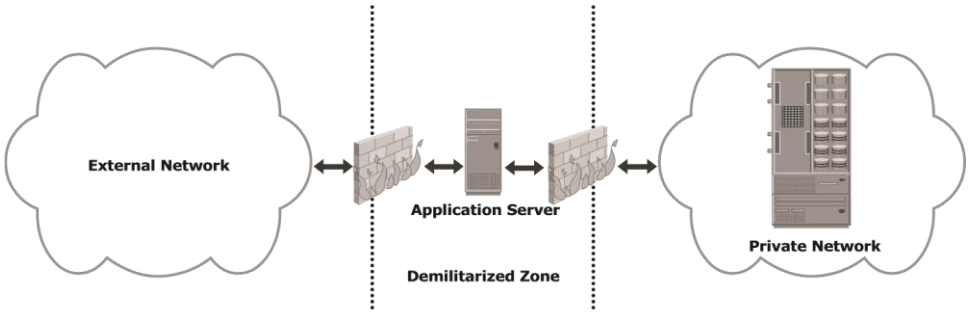
8. A client/server session is now established. The server returns a session ID to the client, which is used to track client activity, such as file locking, as long as the session is active.



Network-Layer Firewalls

Because NAS devices utilize the IP protocol stack, they are vulnerable to various attacks initiated through the public IP network. Network layer firewalls are implemented in NAS environments to protect the NAS devices from these security threats. These network-layer firewalls are capable of examining network packets and comparing them to a set of configured security rules. Packets that are not authorized by a security rule are dropped and not allowed to continue to the requested destination. Rules can be established based on a source address (network or host), a destination address (network or host), a port, or a combination of those factors (source IP, destination IP, and port number). The effectiveness of a firewall depends on how robust and extensive the security rules are. A loosely defined rule set can increase the probability of a security breach.

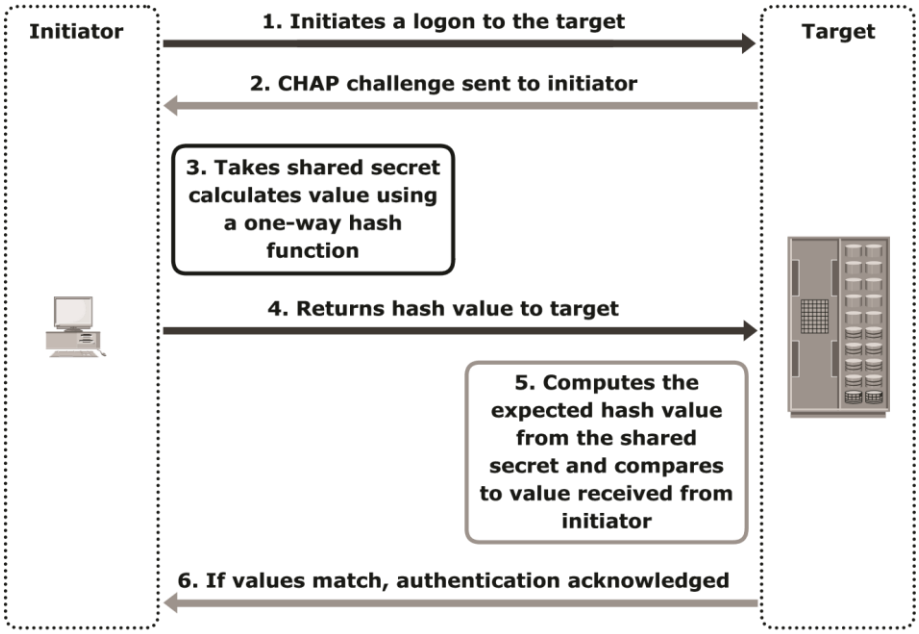
Figure 15-9 depicts a typical firewall implementation. Demilitarized zone (DMZ) is commonly used in networking environments. A DMZ provides a means of securing internal assets while allowing Internet-based access to various resources. In a DMZ environment, servers that need to be accessed through the Internet are placed between two sets of firewalls. Application-specific ports, such as HTTP or FTP, are allowed through the firewall to the DMZ servers. However, no Internet-based traffic is allowed to penetrate the second set of firewalls and gain access to the internal network.



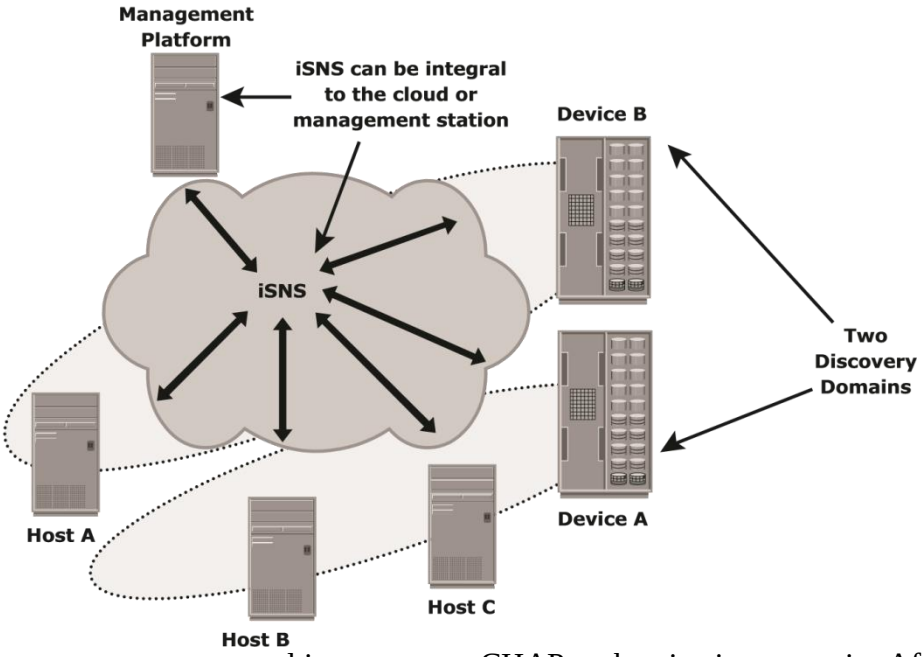
The servers in the DMZ may or may not be allowed to communicate with internal resources. In such a setup, the server in the DMZ is an Internet-facing Web application that is accessing data stored on a NAS device, which may be located on the internal private network. A secure design would only serve data to internal and external applications through the DMZ.

5.4.3 IP SAN

This section describes some of the basic security mechanisms of IP SAN environments. The *Challenge-Handshake Authentication Protocol (CHAP)* is a basic authentication mechanism that has been widely adopted by network devices and hosts. CHAP provides a method for initiators and targets to authenticate each other by utilizing a secret code or password. CHAP secrets are usually random secrets of 12 to 128 characters. The secret is never exchanged directly over the wire; rather, a one-way hash function converts it into a hash value, which is then exchanged. A hash function, using the MD5 algorithm, transforms data in such a way that the result is unique and cannot be changed back to its original form. Figure 15-10 depicts the CHAP authentication process.



If the initiator requires reverse CHAP authentication, the initiator authenticates the target by using the same procedure. The CHAP secret must be configured on the initiator and the target. A CHAP entry, comprising the name of a node and the secret associated with the node, is maintained by the target and the initiator.



The same steps are executed in a two-way CHAP authentication scenario. After these steps are completed, the initiator authenticates the target. If both authentication steps succeed, then data access is allowed. CHAP is often used because it is a fairly simple protocol to implement and can be implemented across a number of disparate systems. *iSNS discovery domains* function in the same way as FC zones. Discovery domains provide functional groupings of devices in an IP-SAN. In order for devices to communicate with one another, they must be configured in the same discovery domain. State change notifications (SCNs) tell the iSNS server when devices are added or removed from a discovery domain. Figure 15-11 depicts the discovery domains in iSNS.

INFORMATION STORAGE AND MANAGEMENT- ONLINE QUESTION BANK

A Subprofile can reference other subprofiles

A. TRUE
B. FALSE
C. neither true nor false
D. either true or false

Clients use which protocol to discover SMI Agents on Storage Area Network?

A. SLP (Service Location Protocol)
B. AGP(Agent Discovery Protocol)
C. SMIP (SMI Protocol)
D. Interface Protocol

Encoding mechanism of CIM Data as XML Elements ?

A. CIM-XML
B. xmlCIM
C. SGML
D. XML

Can a vendor implement additional Classes or support additional Properties than are defined in a Profile and still be considered conformant?

A. TRUE
B. FALSE
C. neither true nor false
D. either true or false

CIM and WBEM has been defined by ?

A. DMTF
B. SMI Standards Organization
C. Open Standards Body
D. None

_____ instruments one or more aspects of the CIM Schema

A. Distributor
B. Provider
C. Manager
D. None

The server can operate directly on the underlying system by calling the system's commands, services, and library functions

A. TRUE
B. FALSE
C. neither true nor false
D. either true or false

Transport protocol used for XMLCIM is

A. UDP
B. HTTP
C. SNMP
D. None

A single CIM based management application can manage storage arrays from multiple vendors.

A. TRUE
B. FALSE
C. neither true nor false
D. either true or false

Collects responses from providers and returns to the client

A. xmlCIM
B. CIMOM
C. DMTF
D. None

A "Logical Volume Manager" helps in

A. Virtualizing storage
B. provide direct access to the underlying storage
C. Manage disk space efficiently without having to know the actual hardware details.
D. Both a & c

Physical Volumes are

A. The space on a physical storage that represent a logical volume
B. Disk or disk partitions used to construct logical volumes
C. A bunch of disks put together that can be made into a logical volume
D. None of the above

Which of the following are true. Logical Volumes

A. Can span across multiple volume groups
B. Can span across multiple physical volumes
C. Can be constructed only using a single physical disk.
D. None of the above

A logical Extent(LE) and Physical extent(PE) are related as follows

A. PE resides on a disk, whereas LE resides on a logical volume
B. LE is larger in size than a PE
C. LE's are unique whereas PE's are not
D. Every LE maps to a one and only one PE

Which of the following statements are true

A. LVM is storage independent whereas a RAID system is limited to the storage subsystem
B. LVM provides snapshot feature
C. With LVM we can grow volumes to any size
D. RAID system can provide more storage space than a LVM

LVM is independent of device IDs because

A. LVM uses it's own device naming to identify a physical disk
B. LVM stores the volume management information on the disks that helps it reconstructVolumes
C. LVM is an abstraction layer over physical devices and does not need device ids
D. Device ids are used only by the device drivers

Concatenation is the technique of

A. Adding physical volumes together to make a volume group
B. Filling up a physical volume completely before writing to the next one in a logical volume
C. writing a block of data onto one disk and then a block onto another disk in an alternate fashion
D. Increasing the size of a volume by adding more disks

Which of the following is not a feature of LVM

A. Independent of disk location
B. Concatenation and striping of storage systems
C. protection against disk failures
D. snapshot capability

LVM does not incur much performance overheads because

A. The writes/reads happen only to logical devices
B. The mapping of logical to physical storage is kept in RAM
C. The time lost in mapping devices is gained by writing to disks in parallel
D. None of the above

VGDA represents

A. The data stored on the logical volume
B. The data stored on physical volume

C. LVM configuration data stored on each physical volume
D. None of the above

_____ is a collection of raw facts from which conclusions may be drawn.

A. Data
B. Storage
C. Information
D. Network

What are the types of data?

A. Structured data
B. Unstructured data
C. Both a and b
D. Modified data

_____ is the intelligence and knowledge derived from data

A. Network
B. Storage
C. Information
D. Data

_____ is a collection of raw facts from which conclusions may be drawn.

A. Data
B. Letters
C. Photograph
D. Book

This data can be generated using a computer and stored in strings of 0s and 1s, in this form is called _____and is accessible by the user only after it is processed by a computer

A. bitmapped data
B. digital data
C. string data
D. raw data

Inexpensive and easier ways to create, collect, and store all types of data,coupled with increasing individual and business needs, have led to accelerated data growth, popularly termed the _____

A. data explosion.
B. Data integrity
C. Data mirroring
D. Data stripping

. _____ Involves collecting data related to various sources and parameters of earthquakes, and other relevant data that needs to be processed to derive meaningful information

A. Data mining
B. Warehousing
C. Seismology
D. Data abstraction

_____ Includes data related to various aspects of a product,such as inventory, description, pricing, availability, and sales

A. Process data
B. Product data
C. Customer data
D. Medical data

_____A combination of data related to a company's customers,such as order details, shipping addresses, and purchase history.

A. Process data
B. Product data
C. Customer data
D. Medical data

._____Data related to the health care industry, such as patient history,radiological images, details of medication and other treatment, and insurance information

A. Customer data
B. Product data
C. Process data
D. Medical data

Which level of RAID refers to disk mirroring with block striping?

A. RAID level 1
B. RAID level 2
C. RAID level 0
D. RAID level 3

Optical disk technology uses

A. Helical scanning
B. DAT
C. a laser beam
D. RAID

With multiple disks, we can improve the transfer rate as well by _____ data across multiple disks.

A. Striping
B. Dividing
C. Mirroring
D. Dividing

Which one of the following is a Stripping technique

A. Byte level stripping
B. Raid level stripping
C. Disk level stripping
D. Block level stripping

The RAID level which mirroring is done along with stripping is

A. RAID 1+0
B. RAID 0
C. RAID 2
D. Both a and b

Where performance and reliability are both important, RAID level ____ is used.

A. 0
B. 1
C. 2
D. 0+1

_____ partitions data and parity among all N+1 disks, instead of storing data in N-disks and parity in one disk.

A. Block interleaved parity
B. Block interleaved distributed parity
C. Bit parity
D. Bit interleaved parity

Hardware RAID implementations permit _____; that is, faulty disks can be removed and replaced by new ones without turning power off.

A. Scrapping
B. Swapping
C. Hot swapping
D. None of the mentioned

. _____ is popular for applications such as storage of log files in a database system, since it offers the best write performance

A. RAID level 1
B. RAID level 2
C. RAID level 0
D. RAID level 3

_____ which increases the number of I/O operations needed to write a single logical block, pays a significant time penalty in terms of write performance

A. RAID level 1
B. RAID level 2
C. RAID level 5
D. RAID level 3

Tertiary storage is built with

A. a lot of money
B. unremovable media
C. removable media
D. secondary storage

Operating system is responsible for

A. disk initialization
B. booting from disk
C. bad-bock recovery
D. all of the mentioned

A magneto-optic disk is

A. primary storage
B. secondary storage
C. tertiary storage
D. removable disk

Which of the following is the process of selecting the data storage and data access characteristics of the database?
A. Logical database design
B. Physical database design
C. Testing and performance tuning
D. Evaluation and selecting

The replacement of a bad block generally is not totally automatic because
A. data in bad block can not be replaced
B. data in bad block is usually lost
C. bad block does not contain any data
D. none of the mentioned

Which of the following is the oldest database model?
A. Relational
B. Hierarchical
C. Physical
D. Network

The surface area of a tape is _____ the surface area of a disk
A. much lesser than
B. much larger than
C. equal to
D. None of these

Which one of the following is not a secondary storage
A. magnetic disks
B. magnetic tapes
C. RAM
D. none of the mentioned

In magnetic disk _____ stores information on a sector magnetically as reversals of the direction of magnetization of the magnetic material
A. Read–write head
B. Read-assemble head
C. Head–disk assemblies
D. Disk arm

. A _____ is the smallest unit of information that can be read from or written to the disk.
A. Track
B. Spindle
C. Sector
D. Platter

The disk platters mounted on a spindle and the heads mounted on a disk arm are together known as _____.
A. Read-disk assemblies
B. Head–disk assemblies
C. Head-write assemblies
D. Read-read assemblies

The disk controller uses _____ at each sector to ensure that the data is not corrupted on data retrieval
A. Checksum
B. Unit drive
C. Read disk
D. Readsum

_____ is the time from when a read or write request is issued to when data transfer begins.
A. Access time
B. Average seek time
C. Seek time
D. Rotational latency time

The time for repositioning the arm is called the _____, and it increases with the distance that the arm must move

A. Access time
B. Average seek time
C. Seek time
D. Rotational latency time

_____ is around one-half of the maximum seek time
A. Access time
B. Average seek time
C. Seek time
D. Rotational latency time

Once the head has reached the desired track, the time spent waiting for the sector to be accessed to appear under the head is called the _____.
A. Access time
B. Average seek time
C. Seek time
D. Rotational latency time

In Flash memory, the erase operation can be performed on a number of pages, called an _____, at once, and takes about 1 to 2 milliseconds
A. Delete block
B. Erase block
C. Flash block
D. Read block

Hybrid disk drives are hard-disk systems that combine magnetic storage with a smaller amount of flash memory, which is used as a cache for frequently accessed data.
A. Hybrid drivers
B. Disk drivers
C. Hybrid disk drivers
D. All of the mentioned

Which of the following is a physical storage media ?
A. Tape Storage
B. Optical Storage
C. Flash memory
D. All of the mentioned

Transport protocol used for XMLCIM is
A. UDP
B. HTTP
C. SNMP
D. None