

**Course Outcomes**

- CO1:** Identify key challenges in managing information and analyze different storage networking technologies and virtualization
- CO2:** Explain components and the implementation of NAS
- CO3:** Describe CAS architecture and types of archives and forms of virtualization
- CO 4:** Illustrate the storage infrastructure and management activities

**Institution Vision**

To produce top-quality engineers who are groomed for attaining excellence in their profession and competitive enough to help in the growth of nation and global society.

**Institution Mission**

- M1:** To offer affordable high-quality graduate program in engineering with value education and make the students socially responsible.
- M2:** To support and enhance the institutional environment to attain research excellence in both faculty and students and to inspire them to push the boundaries of knowledge base.
- M3:** To identify the common areas of interest amongst the individuals for the effective industry- institute partnership in a sustainable way by systematically working together.
- M4:** To promote the entrepreneurial attitude and inculcate innovative ideas among the engineering professionals.

**Department Vision**

To be a center of excellence in Information Science & Engineering education, research and training to meet the growing needs of the industry and society.

**Department Mission**

- M1:** To impart theoretical and practical knowledge through the concepts and technologies in Information Science and Engineering
- M2:** To foster research, collaboration and higher education with premier institutions and industries.
- M3:** Promote innovation and entrepreneurship to fulfill the needs of the society and industry

**Program Educational Objectives**

- PEO1:** Analyse, design and implement solutions to the real-world problems in the field of Information Science and Engineering with multidisciplinary setup
- PEO2:** Keep abreast with the technology, innovation and pursue higher education with high standards of social and professional ethics
- PEO3:** Develop professional and entrepreneurship skills to work effectively as an individual and in a team to meet the ever-changing goals of the organization

**Program Outcomes**

- PO1: Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems.
- PO2: Problem Analysis:** Identify, formulate, review research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural science and engineering sciences.
- PO3: Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal and environmental considerations.
- PO4: Conduct investigations of complex problems:** Use research based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
- PO5: Modern tool usage:** Create, select and apply appropriate techniques, resources and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations
- PO6: The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice
- PO7: Environment sustainability:** Understand the impact of the professional engineering solutions in the societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
- PO8: Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
- PO9: Individual and team work:** Function effectively as an individual and as a member or leader in diverse teams, and in multidisciplinary settings.
- PO10: Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions
- PO11: Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to ones own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
- PO12: Lifelong learning:** Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broader context of technological change.

**Program Specific Outcome**

- PSO1:** Design, implement and maintain the information systems that fulfill the current needs of the industry and society
- PSO2:** Apply computational theory, storage and networking concepts to solve the day to day problems of the world

---

## Local Replication

In today's business environment, it is imperative for an organization to protect mission critical data and minimize the risk of business disruption. If a local outage or disaster occurs, fast data restore and restart is essential to ensure business continuity (BC). Replication is one of the ways to ensure BC. It is the process to create an exact copy (replica) of data. These replica copies are used for restore and restart operations if data loss occurs. These replicas can also be assigned to other hosts to perform various business operations, such as backup, reporting, and testing.

Replication can be classified into two major categories: local and remote. Local replication refers to replicating data within the same array or the same data center. Remote replication refers to replicating data at a remote site.

### Replication Terminology

The common terms used to represent various entities and operations in a replication environment are listed here:

**Source:** A host accessing the production data from one or more LUNs on the storage array is called a *production host*, and these LUNs are known as source LUNs (devices/volumes), production LUNs, or simply the *source*.

**Target:** A LUN (or LUNs) on which the production data is replicated, is called the target LUN or simply the *target* or replica.

**Point-in-Time (PIT) and continuous replica:** Replicas can be either a PIT or a continuous copy. The PIT replica is an identical image of the source at some specific timestamp. For example, if a replica of a file system is created at 4:00 p.m. on Monday, this replica is the Monday 4:00 p.m. PIT copy. On the other hand, the continuous replica is in-sync with the production data at all times.

**Recoverability and restartability:** Recoverability enables restoration of data from the replicas to the source if data loss or corruption occurs. Restartability enables restarting business operations using the replicas. The replica must be consistent with the source so that it is usable for both recovery and restart operations.

**Replicas are immediately accessible by the applications, but the backup copy must be restored by backup software to make it accessible to applications. Backup is always a point in time copy, but a replica can be a point-in-time copy or continuous.**

**Backup is typically used for operational or disaster recovery but replicas can be used for recovery and restart, and also for other business operations, such as backup, reporting, and testing. Replicas typically provide faster RTO compared to recovery from backup.**

### Uses of Local Replicas

One or more local replicas of the source data may be created for various purposes, including the following:

**Alternative source for backup:** Under normal backup operations, data is read from the production volumes (LUNs) and written to the backup device. This places an additional burden on the production infrastructure because production LUNs are simultaneously involved in production operations and servicing data for backup operations. The local replica contains an exact point-in-time (PIT) copy of the source data, and therefore can be used as a source to perform backup operations. This alleviates the backup I/O workload on the production volumes. Another benefit of using local replicas for backup is that it reduces the *backup window* to zero.

**Fast recovery:** If data loss or data corruption occurs on the source, a local replica might be used to recover the lost or corrupted data. If a complete failure of the source occurs, some replication solutions enable a replica to be used to restore data onto a different set of source devices, or production can be restarted on the replica. In either case, this method provides faster recovery and minimal RTO compared to traditional recovery from tape backups. In many instances, business operations can be started using the source device before the data is completely copied from the replica.

**Decision-support activities, such as reporting or data warehousing:** Running the reports using the data on the replicas greatly reduces the I/O burden placed on the production device. Local replicas are also used for data-warehousing applications. The data-warehouse application may be populated by the data on the replica and thus avoid the impact on the production environment.

**Testing platform:** Local replicas are also used for testing new applications or upgrades. For example, an organization may use the replica to test the production application upgrade; if the test is successful, the upgrade may be implemented on the production environment.

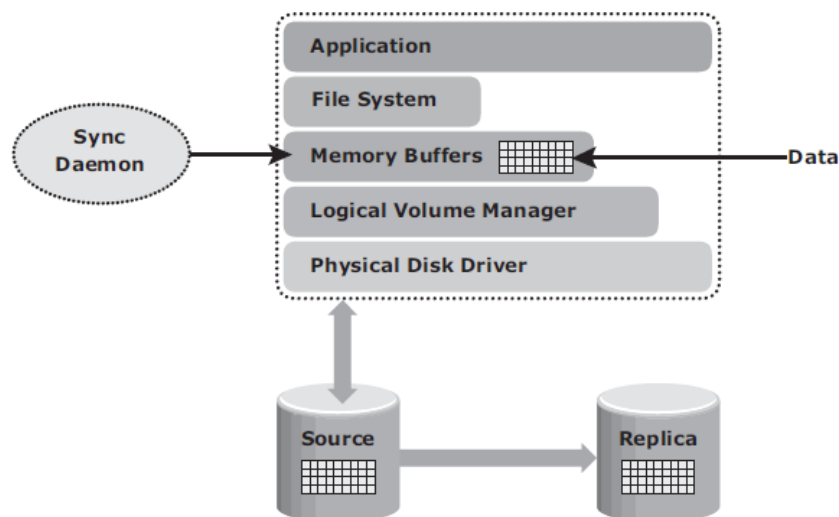
**Data migration:** Another use for a local replica is data migration. Data migrations are performed for various reasons, such as migrating from a smaller capacity LUN to one of a larger capacity for newer versions of the application.

### Replica Consistency

Most file systems and databases buffer the data in the host before writing it to the disk. A consistent replica ensures that the data buffered in the host is captured on the disk when the replica is created. The data staged in the cache and not yet committed to the disk should be flushed before taking the replica. The storage array operating environment takes care of flushing its cache before the replication operation is initiated. Consistency ensures the usability of a replica and is a primary requirement for all the replication technologies.

### Consistency of a Replicated File System

File systems buffer the data in the host memory to improve the application response time. The buffered data is periodically written to the disk. In UNIX operating systems, *sync daemon* is the process that flushes the buffers to the disk at set intervals. In some cases, the replica is created between the set intervals, which might result in the creation of an inconsistent replica. Therefore, host memory buffers must be flushed to ensure data consistency on the replica, prior to its creation. Figure 11-1 illustrates how the file system buffer is flushed to the source device before replication. If the host memory buffers are not flushed, the data on the replica will not contain the information that was buffered in the host. If the file system is unmounted before creating the replica, the buffers will be automatically flushed and the data will be consistent on the replica.



**Figure 11-1:** Flushing the file system buffer

If a mounted file system is replicated, some level of recovery, such as *fsck* or *log replay*, is required on the replicated file system. When the file system replication and check process are completed, the replica file system can be mounted for operational use.

### Consistency of a Replicated Database

A database may be spread over numerous files, file systems, and devices. All of these must be replicated consistently to ensure that the replica is restorable and restartable. Replication is performed with the database offline or online. If the database is offline during the creation of the replica, it is not available for I/O operations. Because no updates occur on the source, the replica is consistent. If the database is online, it is available for I/O operations, and transactions to the database update the data continuously. When a database is replicated while it is online, changes made to the database at this time must be applied to the replica to make it consistent. A consistent replica of an online database is created by using the dependent write I/O principle or by holding I/Os momentarily to the source before creating the replica.

A *dependent write I/O* principle is inherent in many applications and database management systems (DBMS) to ensure consistency. According to this principle, a write I/O is not issued by an application until a prior related write I/O has completed. For example, a data write is dependent on the successful completion of the prior log write.

For a transaction to be deemed complete, databases require a series of writes to have occurred in a particular order. These writes will be recorded on the various devices or file systems. Figure 11-2, illustrates the process of flushing the buffer from the host to the source; I/Os 1 to 4 must complete for the transaction to be considered complete. I/O 4 is dependent on I/O 3 and occurs only if I/O 3 is complete. I/O 3 is dependent on I/O 2, which in turn depends on I/O 1. Each I/O completes only after completion of the previous I/O(s).

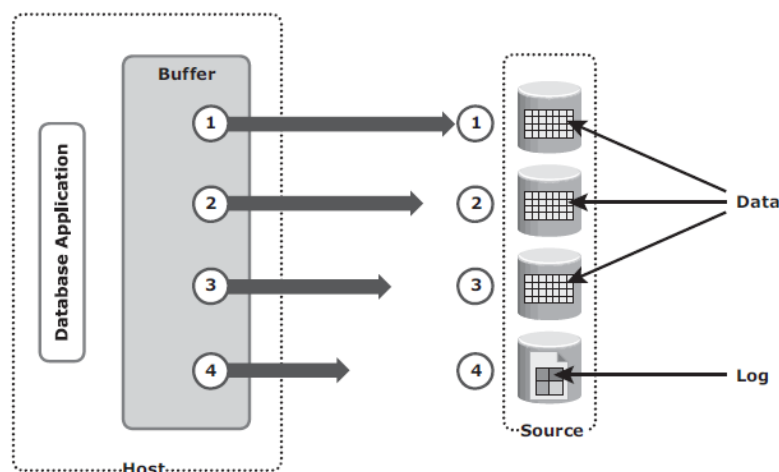


Figure 11-2: Dependent write consistency on sources

When the replica is created, all the writes to the source devices must be captured on the replica devices to ensure data consistency. Figure 11-3 illustrates the process of replication from the source to the replica. I/O transactions 1 to 4 must be carried out for the data to be consistent on the replica.

It is possible that I/O transactions 3 and 4 were copied to the replica devices, but I/O transactions 1 and 2 were not copied. Figure 11-4 shows this situation. In this case, the data on the replica is inconsistent with the data on the source. If a restart were to be performed on the replica devices, I/O 4, which is available on the replica, might indicate that a particular transaction is complete, but all the data associated with the transaction will be unavailable on the replica, making the replica inconsistent.

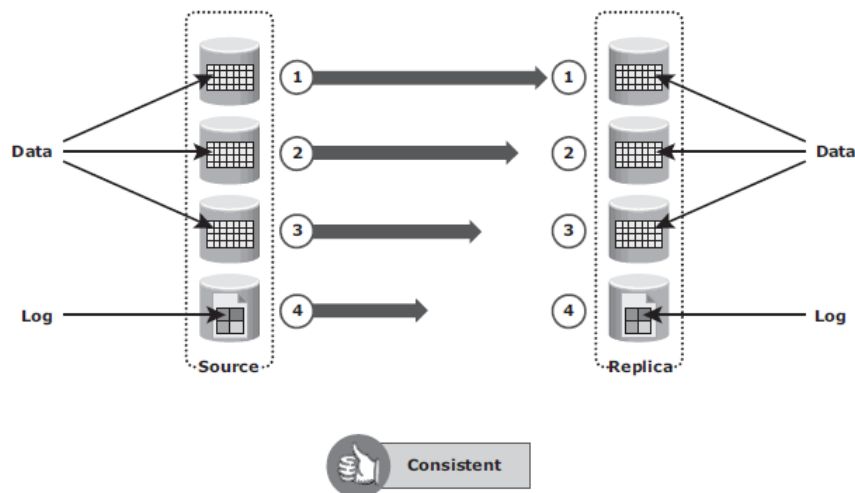


Figure 11-3: Dependent write consistency on replica

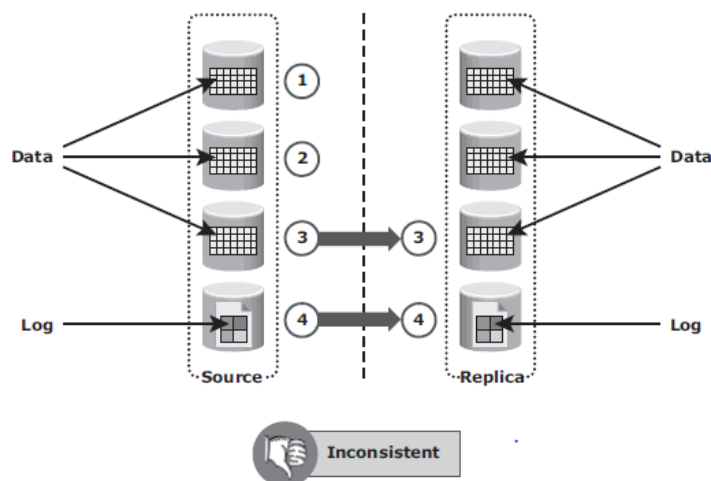


Figure 11-4: Inconsistent database replica

Another way to ensure consistency is to make sure that the write I/O to all source devices is held for the duration of creating the replica. This creates a consistent image on the replica. However, databases and applications might time out if the I/O is held for too long.

## Local Replication Technologies

Host-based, storage array-based, and network-based replications are the major technologies used for local replication. File system replication and LVM-based replication are examples of host-based local replication. Storage array-based replication can be implemented with distinct solutions, namely, full-volume mirroring, pointer-based full-volume replication, and pointer based virtual replication. Continuous data protection (CDP) is an example of network based replication.

### Host-Based Local Replication

LVM-based replication and file system (FS) snapshot are two common methods of host-based local replication.

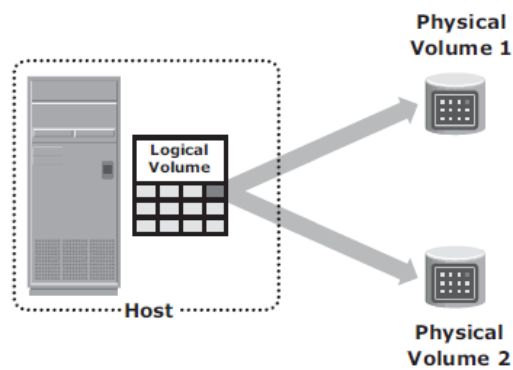
#### *LVM-Based Replication*

In *LVM-based replication*, the logical volume manager is responsible for creating and controlling the host-level logical volumes. An LVM has three components: physical volumes (physical disk), volume groups, and logical volumes. A *volume group* is created by grouping 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 block* in a logical volume is mapped to two physical blocks on two different physical volumes, as shown in Figure 11-5. An application write to a logical volume is written to the two physical volumes 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.

#### *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.



**Figure 11-5:** LVM-based mirroring

### ***Limitations of LVM-Based Replication***

Every write generated by an application translates into two writes on the disk, and thus, an additional burden is placed on the host CPU. This can degrade application performance. Presenting an LVM-based local replica to another 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 resynchronization operations is also a challenge because 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 that the LVM mirroring provides is unnecessary. This solution does not scale to provide replicas of federated databases and applications. Both the replica and source are stored within the same volume group. Therefore, the replica might become unavailable if there is an error in the volume group. If the server fails, both the source and replica are unavailable until the server is brought back online.

***A federated database is a collection of databases that work together as a single entity. Each individual database in a federated database is self-contained and fully functional.***

**When a federated database receives a query, it forwards the request to the database entity that contains the requested data. A federated database appears as a unified database to an application. This eliminates the need to send queries to multiple databases and combine the results.**

### *File System Snapshot*

A file system (FS) snapshot is a pointer-based replica that requires a fraction of the space used by the production FS. This snapshot can be implemented by either FS or by LVM. It uses the Copy on First Write (CoFW) principle to create snapshots.

When a snapshot is created, a bitmap and blockmap 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 the snap creation. The blockmap is used to indicate the exact address from which the data is to be read when the data is accessed from the Snap FS. Immediately after the creation of the FS snapshot, all reads from the snapshot are actually served by reading the production FS. In a CoFW mechanism, if a write I/O is issued to the production FS for the first time after the creation of a snapshot, the I/O is held and the original data of production FS corresponding to that location is moved to the Snap FS. Then, the write is allowed to the production FS. The bitmap and blockmap are updated accordingly. Subsequent writes to the same location do not initiate the CoFW activity. 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 blockmap, and the data is read from that address on the Snap FS. Read requests from the production FS work as normal.

Figure 11-6 illustrates the write operations to the production file system. For example, a write data “C” occurs on block 3 at the production FS, which currently holds data “c” The snapshot application holds the I/O to the production FS and first copies the old data “c” to an available data block on the Snap FS. The bitmap and blockmap values for block 3 in the production FS are changed in the snap metadata. The bitmap of block 3 is changed to 1, indicating that this block has changed on the production FS. The block map of block 3 is changed and indicates the block number where the data is written in Snap FS, (in this case block 2). After this is done, the I/Os to the production FS are allowed to complete. Any subsequent writes to block 3 on the production FS occur as normal, and it does not initiate the CoFW operation. Similarly, if an I/O is issued to block 4 on the production FS to change the value of data “d” to “D,” the snapshot application holds the I/O to the production FS and copies the old data to an available data block on the Snap FS. Then it changes the bitmap of block 4 to 1, indicating that the data block has changed on the production FS. The blockmap for block 4 indicates the block number where the data can be found on the Snap FS, in this case, data block 1 of the Snap FS. After this is done, the I/O to the production FS is allowed to complete.

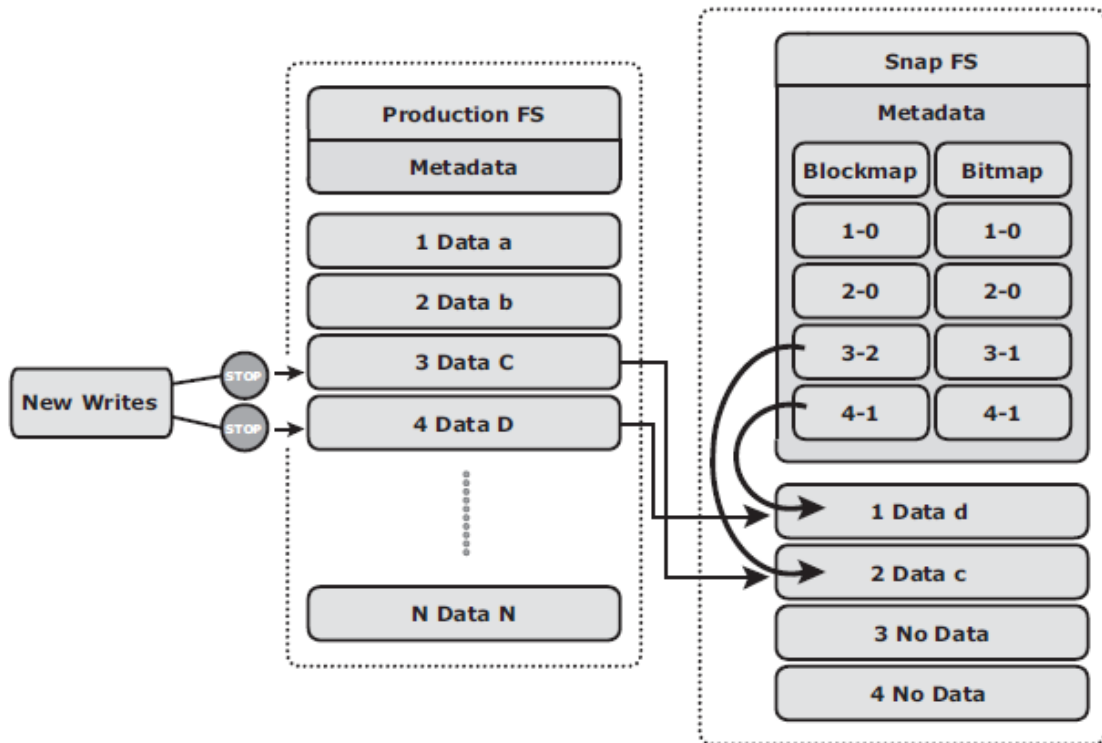


Figure 11-6: Write to production FS

**Storage Array-Based Local Replication**

In *storage array-based local replication*, the array-operating environment performs the local replication process. The host resources, such as the 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 alternative host for other business operations.

In this replication, the required number of replica devices should be selected on the same array and then data should be replicated between the source-replica pairs. Figure 11-7 shows a storage array-based local replication, where the source and target are in the same array and accessed by different hosts.

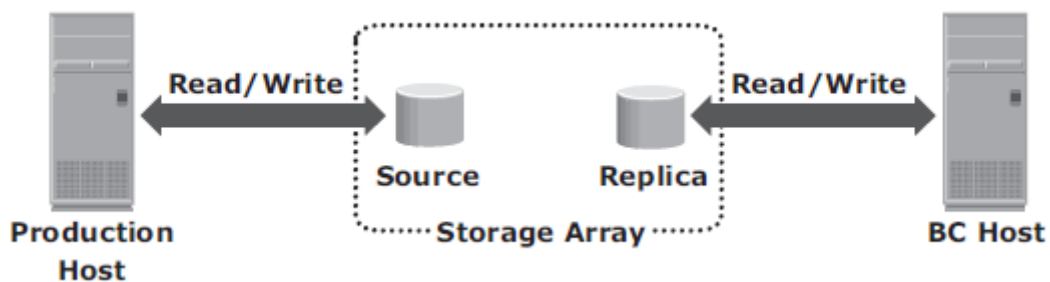


Figure 11-7: Storage array-based local replication

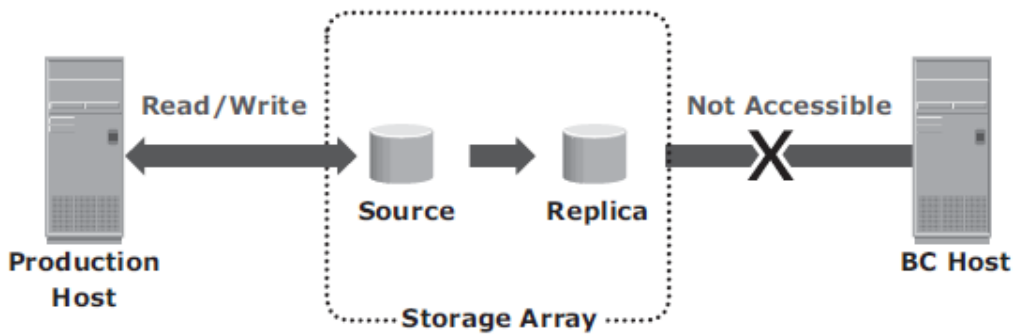
Storage array-based local replication is commonly implemented in three ways:

1. Full-volume mirroring
2. Pointer-based full-volume replication, and
3. Pointer-based virtual replication.

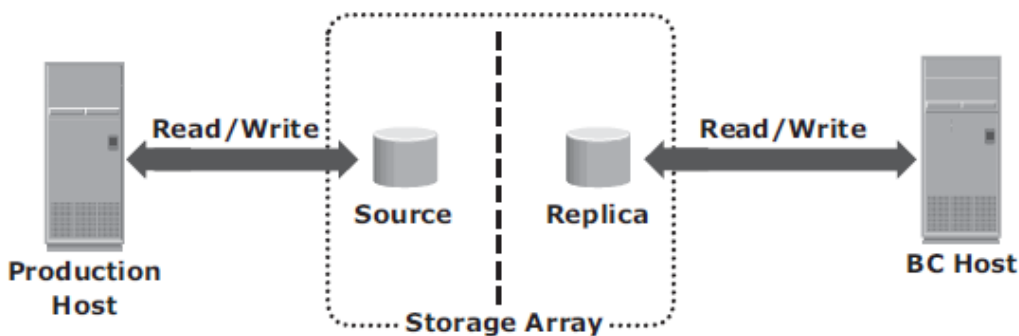
Replica devices are also referred as target devices, accessible by other hosts.

### ***Full-Volume Mirroring***

In *full-volume mirroring*, the target is attached to the source and established as a mirror of the source (Figure 11-8 [a]). The data on the source is copied to the target. New updates to the source are also updated on the target. After all the data is copied and both the source and the target contain identical data, the target can be considered as a mirror of the source.



(a) Full Volume Mirroring with Source Attached to Replica



(b) Full Volume Mirroring with Source Detached from Replica

### **Figure 11-8: Full-volume mirroring**

While the target is attached to the source it remains unavailable to any other host. However, the production host continues to access the source. After the synchronization is complete, the target can be detached from the source and made available for other business operations. Figure 11-8 (b) shows full-volume mirroring when the target is detached from the source. Both the

source and the target can be accessed for read and write operations by the production and business continuity hosts respectively.

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

After detachment, changes made to both the 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 or higher.

### ***Pointer-Based, Full-Volume Replication***

Another method of array-based local replication is *pointer-based full-volume replication*. Similar to full-volume mirroring, this technology can provide full copies of the source data on the targets. Unlike full-volume mirroring, the target is immediately accessible by the BC host after the replication session is activated. Therefore, data synchronization and detachment of the target is not required to access it. Here, the time of replication session activation defines the PIT copy of the 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. The protection bitmap keeps track of the changes at the source device. The pointers on the target are initialized to map the corresponding data blocks on the source. The data is then copied from the source to the target based on the mode of activation.

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

- A write I/O is issued to a specific address on the source for the first time.
- A read or write I/O 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 replication session activation, the original data at that address is copied to the target. After this operation, the new data is updated on the source. This ensures that the original data at the point-in-time of activation is preserved on the target (see Figure 11-9).

When a read is issued to the target for the first time after replication session activation, the original data is copied from the source to the target and is made available to the BC host (see Figure 11-10).

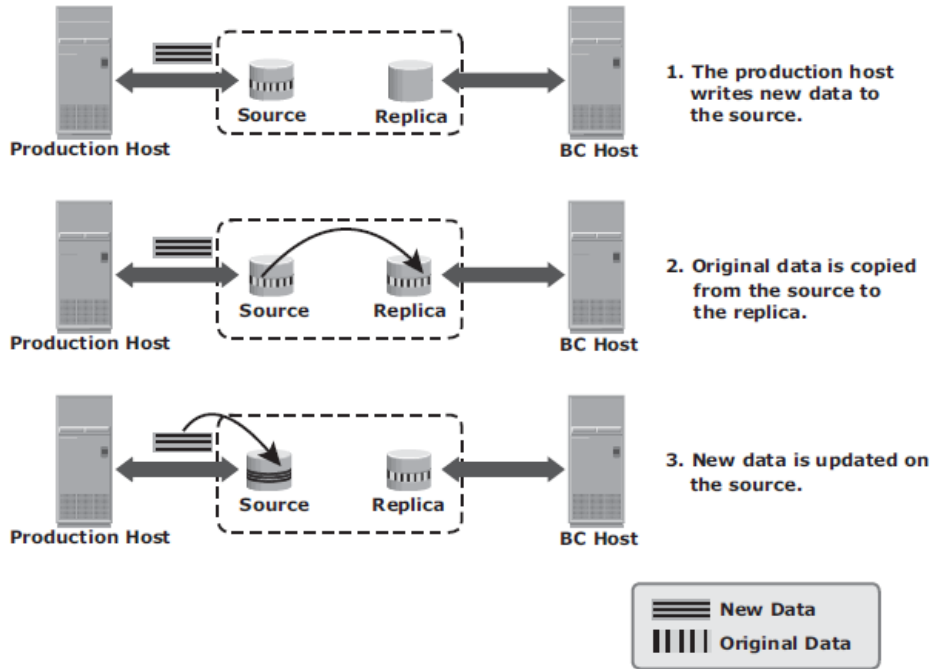


Figure 11-9: Copy on first access (CoFA) — write to source

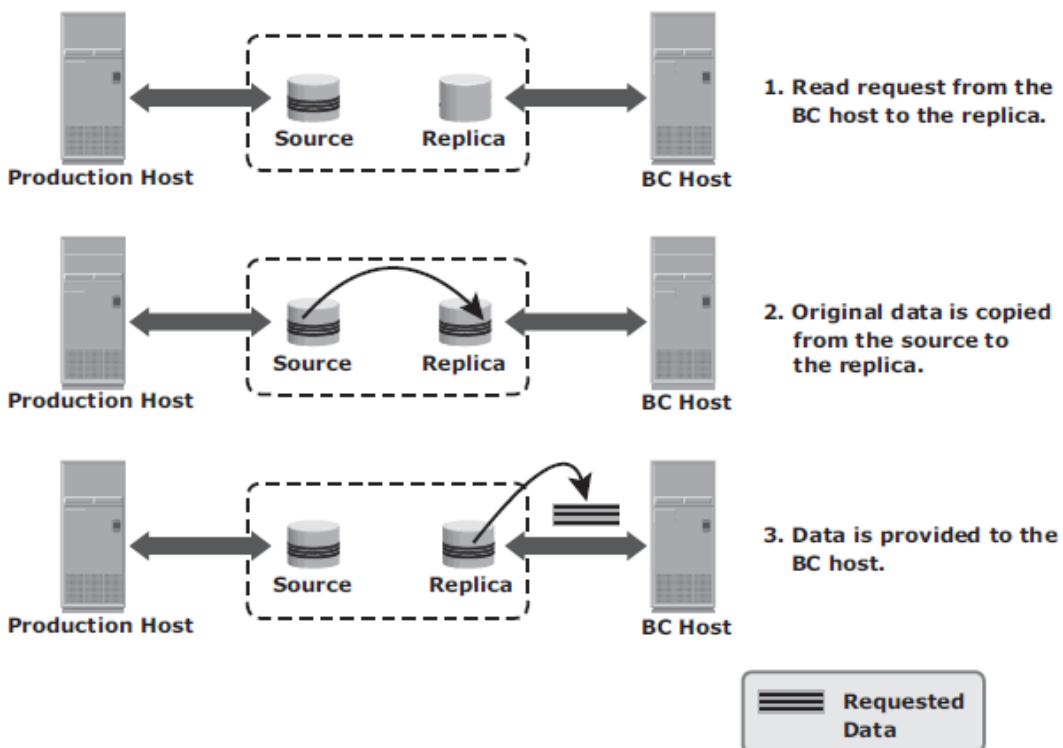
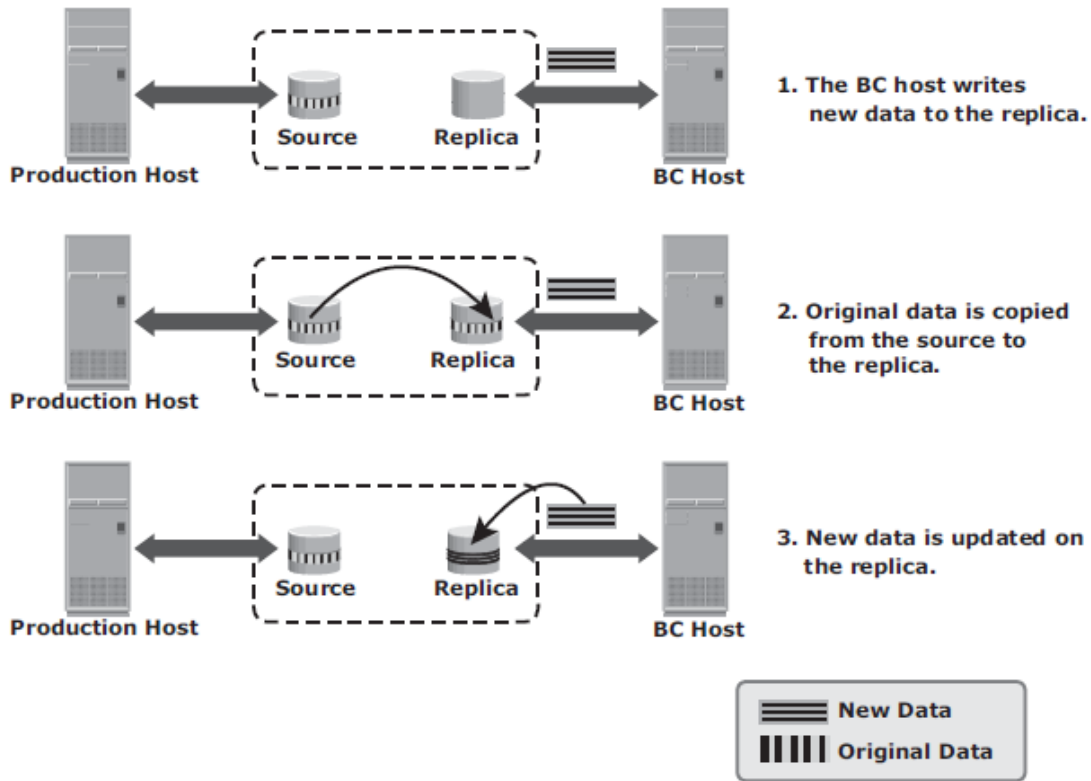


Figure 11-10: Copy on first access (CoFA) — read from target

When a write is issued to the target for the first time after the replication session activation, the original data is copied from the source to the target. After this, the new data is updated on the target (see Figure 11-11).



**Figure 11-11:** Copy on first access (CoFA) — write to target

In all cases, the protection bit for the data block on the source 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 the reads or writes to the same data blocks on the target, do not trigger a copy operation, therefore this method is termed “Copy on First Access.”

If the replication session is terminated, then the target device has only 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 restore because it is not a full replica of the source.

In a 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 to the target 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 contains all the original data from the source at the point-in-time of activation. This makes the target a viable copy for restore or other business continuity operations. The key difference between a pointer-based, Full Copy mode and full-volume mirroring is that the target is immediately accessible upon replication session activation in the Full Copy mode. Both the full-volume mirroring and pointerbased full-volume replication technologies require the target devices to be at least as large as the source devices. In addition, full-volume mirroring and pointerbased full-volume replication in the Full Copy mode can provide incremental resynchronization and restore capabilities.

### ***Pointer-Based Virtual Replication***

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

Pointer-based virtual replication uses the CoFW technology. When a write is issued to the source for the first time after the replication session activation, the original data at that address is copied to a predefined area in the array. This area is generally known as the *save location*. The pointer in the target is updated to point to this data in the save location. After this, the new write is updated on the source. This process is illustrated in Figure 11-12.

When a write is issued to the target for the first time after replication session activation, the data is copied from the source to the save location, and the pointer is updated to the data in the save location. Another copy of the original data is created in the save location before the new write is updated on the save location. Subsequent writes to the same data block on the source or target do not trigger a copy operation. This process is illustrated in Figure 11-13.

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

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. The target contains only pointers to the data, and therefore, 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 the expected data change.

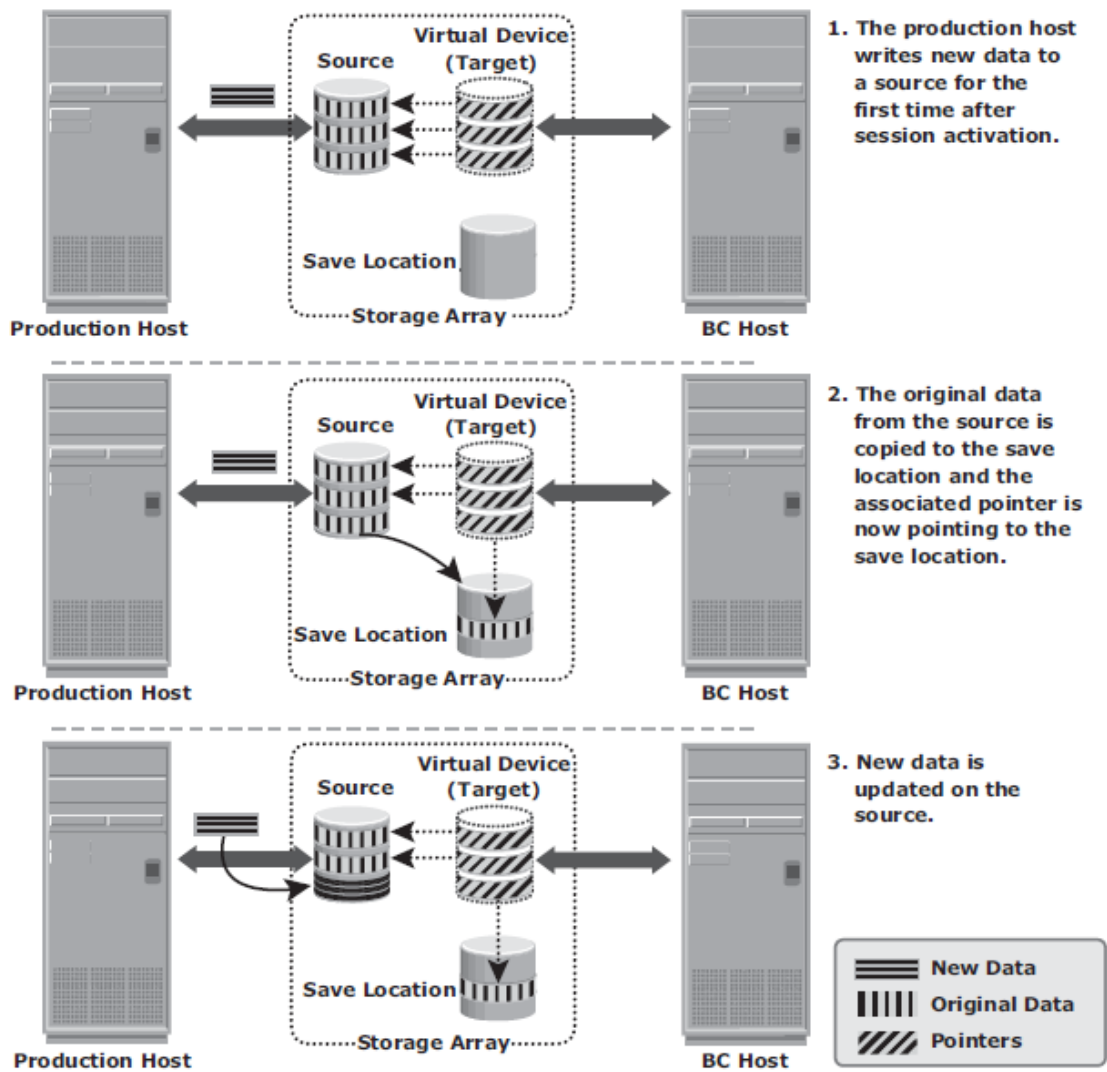
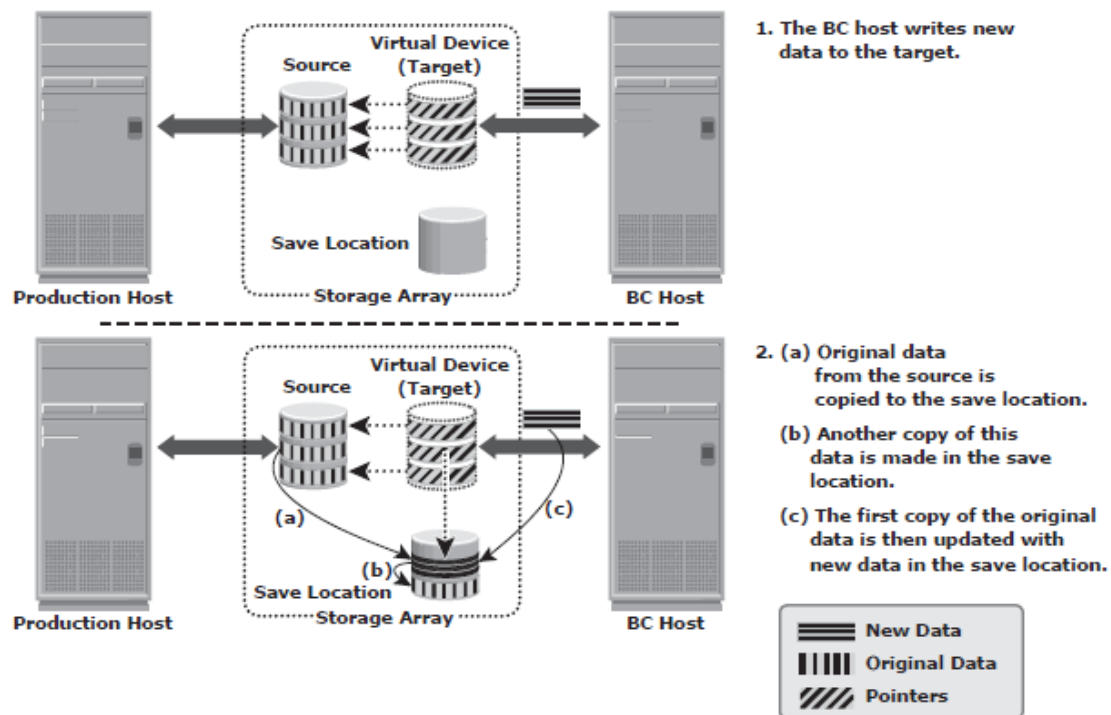


Figure 11-12: Pointer-based virtual replication — write to source

### Network-Based Local Replication

In network-based replication, the replication occurs at the network layer between the hosts and storage arrays. Network-based replication combines the benefits of array-based and host-based replications. By offloading replication from servers and arrays, network-based replication can work across a large number of server platforms and storage arrays, making it ideal for highly heterogeneous environments. *Continuous data protection* (CDP) is a technology used for network-based local and remote replications.



**Figure 11-13:** Pointer-based virtual replication — write to target

### *Continuous Data Protection*

In a data center environment, mission-critical applications often require instant and unlimited data recovery points. Traditional data protection technologies offer limited recovery points. If data loss occurs, the system can be rolled back only to the last available recovery point. Mirroring offers continuous replication; however, if logical corruption occurs to the production data, the error might propagate to the mirror, which makes the replica unusable. In normal operation, CDP provides the ability to restore data to any previous PIT. It enables this capability by tracking all the changes to the production devices and maintaining consistent point-in-time images.

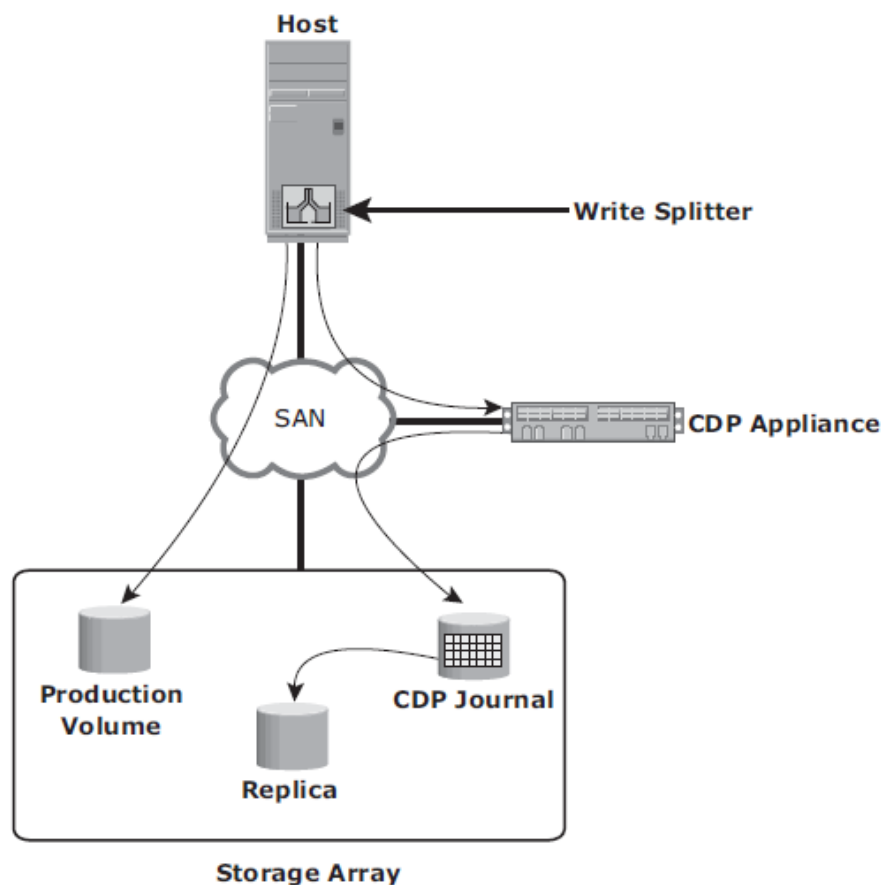
In CDP, data changes are continuously captured and stored in a separate location from the primary storage. Moreover, RPOs are random and do not need to be defined in advance. With CDP, recovery from data corruption poses no problem because it allows going back to a PIT image prior to the data corruption incident. CDP uses a *journal volume* to store all data changes on the primary storage. The journal volume contains all the data that has changed from the time the replication session started. The amount of space that is configured for the journal determines how far back the recovery points can go. CDP is typically implemented using *CDP*

*appliance* and *write splitters*. CDP implementation may also be host-based, in which CDP software is installed on a separate host machine.

CDP appliance is an intelligent hardware platform that runs the CDP software and manages local and remote data replications. Write splitters intercept writes to the production volume from the host and split each write into two copies. Write splitting can be performed at the host, fabric, or storage array.

### ***CDP Local Replication Operation***

Figure 11-14 describes CDP local replication. In this method, before the start of replication, the replica is synchronized with the source and then the replication process starts. After the replication starts, all the writes to the source are split into two copies. One of the copies is sent to the CDP appliance and the other to the production volume. When the CDP appliance receives a copy of a write, it is written to the journal volume along with its timestamp. As a next step, data from the journal volume is sent to the replica at predefined intervals.



**Figure 11-14:** Continuous data protection — local replication

While recovering data to the source, the CDP appliance restores the data from the replica and applies journal entries up to the point in time chosen for recovery.

### **Tracking Changes to Source and Replica**

Updates can occur on the source device after the creation of PIT local replicas. If the primary purpose of local replication is to have a viable PIT copy for data recovery or restore operations, then the replica devices should not be modified. Changes can occur on the replica device if it is used for other business operations. To enable incremental resynchronization or restore operations, changes to both the source and replica devices after the PIT should be tracked.

This is typically done using bitmaps, where each bit represents a block of data. The data block sizes can range from 512 bytes to 64 KB or greater. For example, if the block size is 32 KB, then a 1-GB device would require 32,768 bits (1 GB divided by 32 KB). The size of the bitmap would be 4 KB. If the data in any 32 KB block is changed, the corresponding bit in the bitmap is flagged. If the block size is reduced for tracking purposes, then the bitmap size increases correspondingly.

The bits in the source and target bitmaps are all set to 0 (zero) when the replica is created. Any changes to the source or replica are then flagged by setting the appropriate bits to 1 in the bitmap. When resynchronization or restore is required, a *logical OR* operation between the source bitmap and the target bitmap is performed. The bitmap resulting from this operation references all blocks that have been modified in either the source or replica (see Figure 11-15). This enables an optimized resynchronization or a restore operation because it eliminates the need to copy all the blocks between the source and the replica. The direction of data movement depends on whether a resynchronization or a restore operation is performed.

If resynchronization is required, changes to the replica are overwritten with the corresponding blocks from the source. In this example, that would be blocks labeled 2, 3, and 7 on the replica. If a restore is required, changes to the source are overwritten with the corresponding blocks from the replica. In this example, that would be blocks labelled 0, 3, and 5 on the source. In either case, changes to both the source and the target cannot be simultaneously preserved.

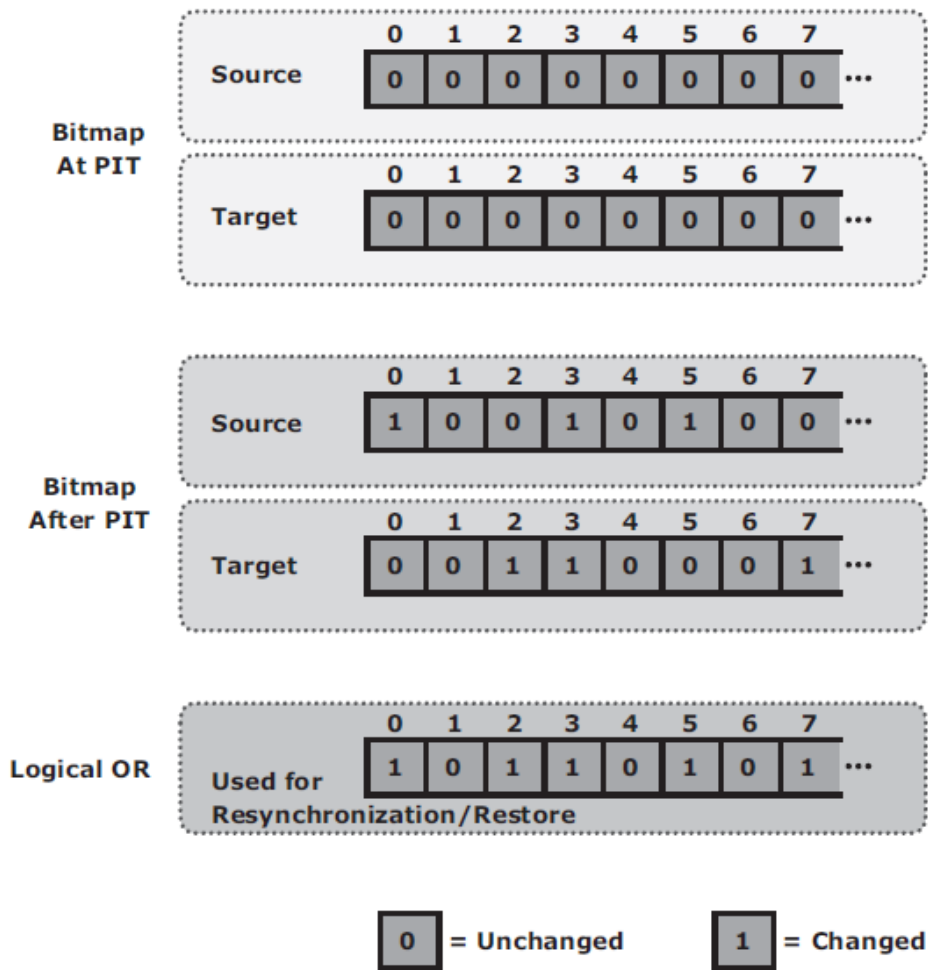


Figure 11-15: Tracking changes

**Restore and Restart Considerations**

Local replicas are used to restore data to production devices. Alternatively, applications can be restarted using the consistent PIT replicas. Replicas are used to restore data to the production devices if logical corruption of data on production devices occurs — that is, the devices are available but the data on them is invalid. Examples of logical corruption include accidental deletion of data (tables or entries in a database), incorrect data entry, and incorrect data updates. Restore operations from a replica are incremental and provide a small RTO. In some instances, the applications can be resumed on the production devices prior to the completion of the data copy. Prior to the restore operation, access to production and replica devices should be stopped. Production devices might also become unavailable due to physical failures, such as the production server or physical drive failure. In this case, applications can be restarted using the data on the latest replica. As a protection against further failures, a Gold Copy (another copy

of replica device) of the replica device should be created to preserve a copy of data in the event of failure or corruption of the replica devices. After the issue has been resolved, the data from the replica devices can be restored back to the production devices.

Full-volume replicas (both full-volume mirrors and pointer-based in Full Copy mode) can be restored to the original source devices or to a new set of source devices. Restores to the original source devices can be incremental, but restores to a new set of devices are full-volume copy operations.

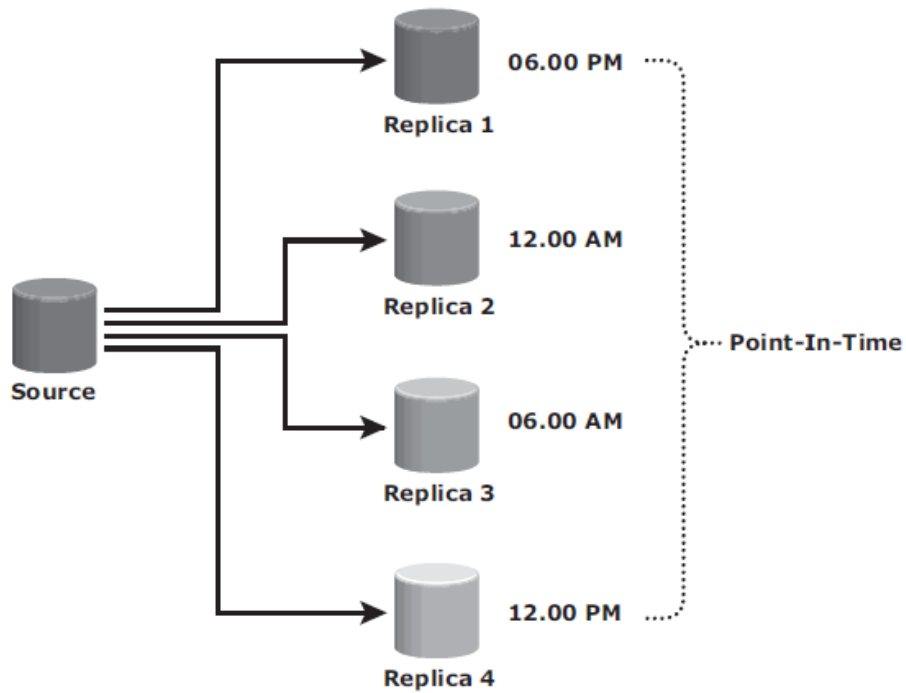
In pointer-based virtual and pointer-based full-volume replication in CoFA mode, access to data on the replica is dependent on the health and accessibility of the source volumes. If the source volume is inaccessible for any reason, these replicas cannot be used for a restore or a restart operation. Table 11-1 presents a comparative analysis of the various storage array-based replication technologies.

**Table 11-1: Comparison of Local Replication Technologies**

<b>FACTOR</b>	<b>FULL-VOLUME MIRRORING</b>	<b>POINTER-BASED, FULL-VOLUME REPLICATION</b>	<b>POINTER-BASED VIRTUAL REPLICATION</b>
Performance impact on source due to replica	No impact	CoFA mode – some impact Full copy mode – no impact	High impact
Size of target	At least the same as the source	At least the same as the source	Small fraction of the source
Availability of source for restoration	Not required	CoFA mode – required Full copy mode – not required	Required
Accessibility to target	Only after synchronization and detachment from the source	Immediately accessible	Immediately accessible

### Creating Multiple Replicas

Most storage array-based replication technologies enable source devices to maintain replication relationships with multiple targets. Changes made to the source and each of the targets can be tracked. This enables incremental resynchronization of the targets. Each PIT copy can be used for different BC activities and as a restore point. Figure 11-16 shows an example in which a copy is created every 6 hours from the same source.



**Figure 11-16:** Multiple replicas created at different PIT

If the source is corrupted, the data can be restored from the latest PIT copy. The maximum RPO in the example shown in Figure 11-16 is 6 hours. More frequent replicas further reduce the RPO. Array-based local replication technologies also enable the creation of multiple *concurrent* PIT replicas. In this case, all replicas contain identical data. One or more of the replicas can be set aside for restore operations. Decision support activities can be performed using the other replicas.

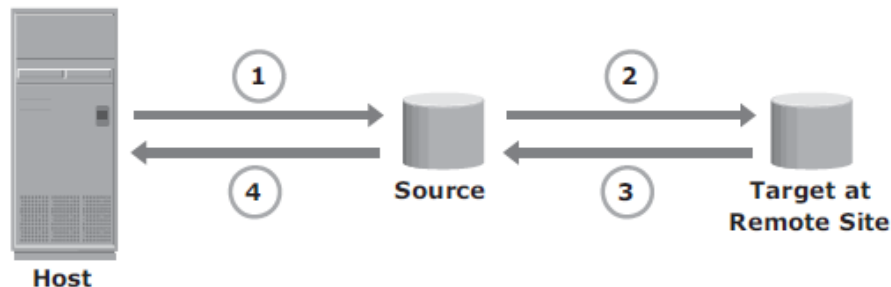
## Remote Replication

*Remote replication* is the process to create replicas of information assets at remote sites (locations). Remote replication helps organizations mitigate the risks associated with regionally driven outages resulting from natural or human-made disasters. During disasters, the workload can be moved to a remote site to ensure continuous business operation. Similar to local replicas, remote replicas can also be used for other business operations.

### Modes of Remote Replication

The two basic modes of remote replication are synchronous and asynchronous.

In *synchronous remote replication*, writes must be committed to the source and remote replica (or target), prior to acknowledging “write complete” to the host (see Figure 12-1). Additional writes on the source cannot occur until each preceding write has been completed and acknowledged. This ensures that data is identical on the source and replica at all times. Further, writes are transmitted to the remote site exactly in the order in which they are received at the source. Therefore, write ordering is maintained. If a source-site failure occurs, synchronous remote replication provides zero or near-zero recovery point objective (RPO).



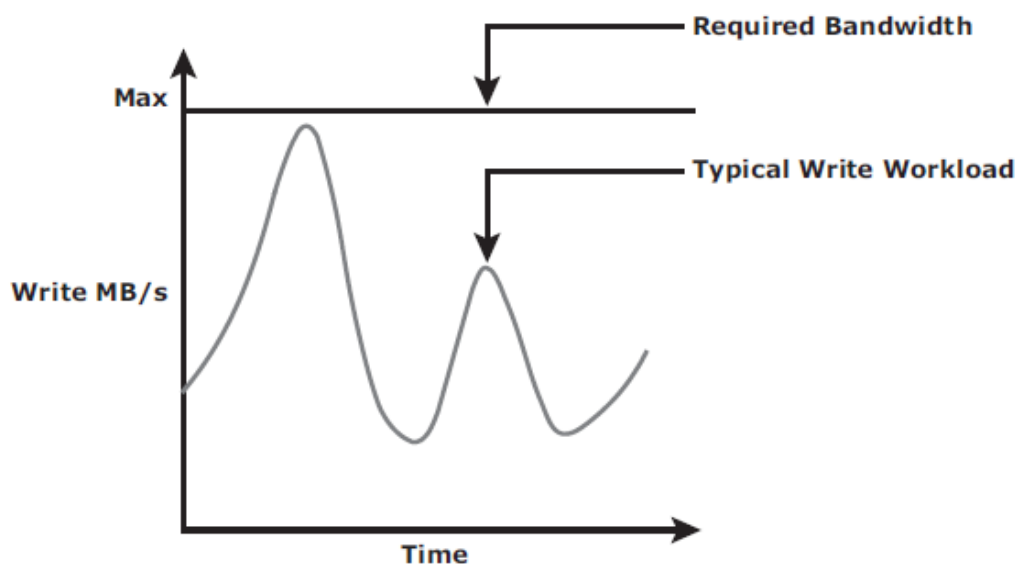
- ① The host writes data to the source.
- ② Data from the source is replicated to the target at a remote site.
- ③ The target acknowledges back to the source.
- ④ The source acknowledges write complete to the host.

**Figure 12-1:** Synchronous replication

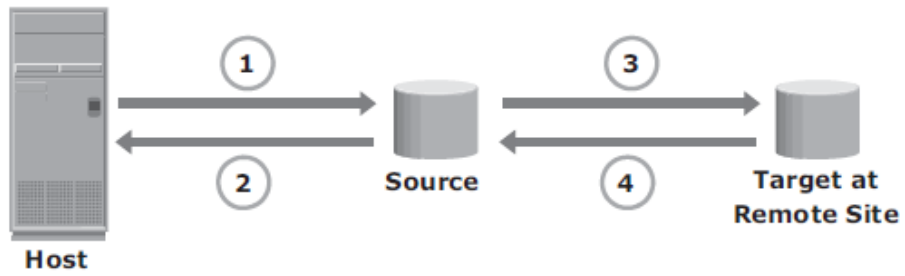
However, application response time is increased with synchronous remote replication because writes must be committed on both the source and target before sending the “write complete” acknowledgment to the host. The degree of impact on response time depends primarily on the distance between sites, bandwidth, and quality of service (QOS) of the network connectivity

infrastructure. Figure 12-2 represents the network bandwidth requirement for synchronous replication. If the bandwidth provided for synchronous remote replication is less than the maximum write workload, there will be times during the day when the response time might be excessively elongated, causing applications to time out. The distances over which synchronous replication can be deployed depend on the application's capability to tolerate extensions in response time. Typically, it is deployed for distances less than 200 KM (125 miles) between the two sites.

In *asynchronous remote replication*, a write is committed to the source and immediately acknowledged to the host. In this mode, data is buffered at the source and transmitted to the remote site later (see Figure 12-3). Asynchronous replication eliminates the impact to the application's response time because the writes are acknowledged immediately to the source host. This enables deployment of asynchronous replication over distances ranging from several hundred to several thousand kilometers between the primary and remote sites. Figure 12-4 shows the network bandwidth requirement for asynchronous replication. In this case, the required bandwidth can be provisioned equal to or greater than the average write workload. Data can be buffered during times when the bandwidth is not enough and moved later to the remote site. Therefore, sufficient buffer capacity should be provisioned.



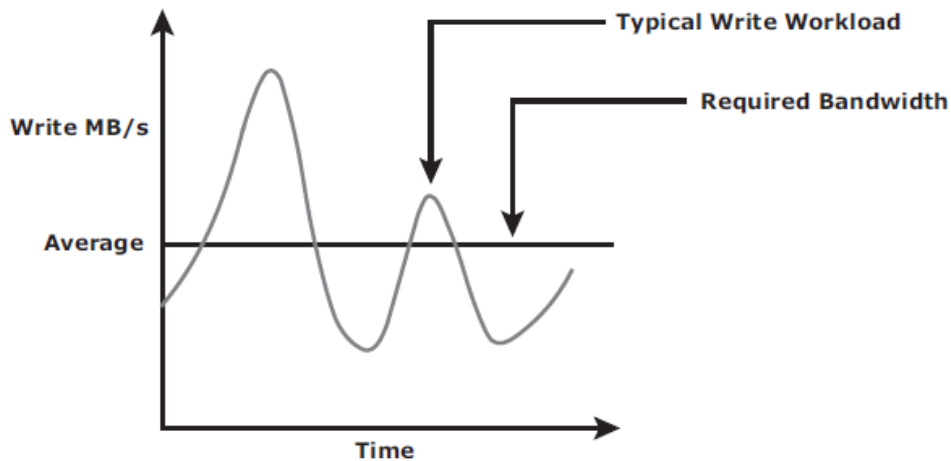
**Figure 12-2:** Bandwidth requirement for synchronous replication



- ① The host writes data to the source.
- ② The write is immediately acknowledged to the host.
- ③ Data is transmitted to the target at a remote site later.
- ④ The target acknowledges back to the source.

**Figure 12-3:** Asynchronous replication

In asynchronous replication, data at the remote site will be behind the source by at least the size of the buffer. Therefore, asynchronous remote replication provides a finite (nonzero) RPO disaster recovery solution. RPO depends on the size of the buffer, the available network bandwidth, and the write workload to the source.



**Figure 12-4:** Bandwidth requirement for asynchronous replication

Asynchronous replication implementation can take advantage of *locality of reference* (repeated writes to the same location). If the same location is written multiple times in the buffer prior to transmission to the remote site, only the final version of the data is transmitted. This feature conserves link bandwidth.

In both synchronous and asynchronous modes of replication, only writes to the source are replicated; reads are still served from the source.

### **Remote Replication Technologies**

Remote replication of data can be handled by the hosts or storage arrays. Other options include specialized network-based appliances to replicate data over the LAN or SAN. An advanced replication option such as three-site replication is used.

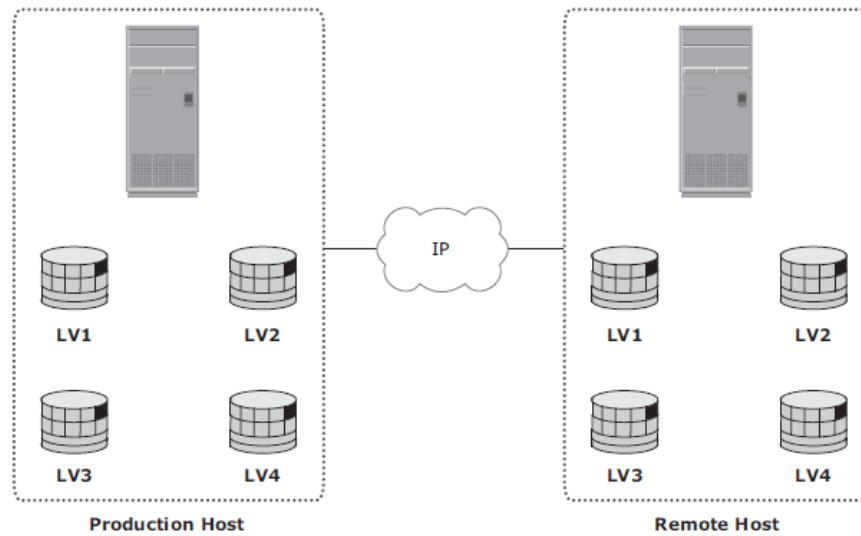
### **Host-Based Remote Replication**

Host-based remote replication uses the host resources to perform and manage the replication operation. There are two basic approaches to host-based remote replication: Logical volume manager (LVM) based replication and database replication via log shipping.

#### ***LVM-Based Remote Replication***

*LVM-based remote 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 replica is performed. One method to perform initial synchronization is to backup the source data and restore the data to the remote replica. Alternatively, it can be performed by replicating over the IP network. Until the completion of the initial synchronization, production work on the source volumes is typically halted. After the initial synchronization, production work can be started on the source volumes and replication of data can be performed over an existing standard IP network (see Figure 12-5).



**Figure 12-5:** LVM-based remote replication

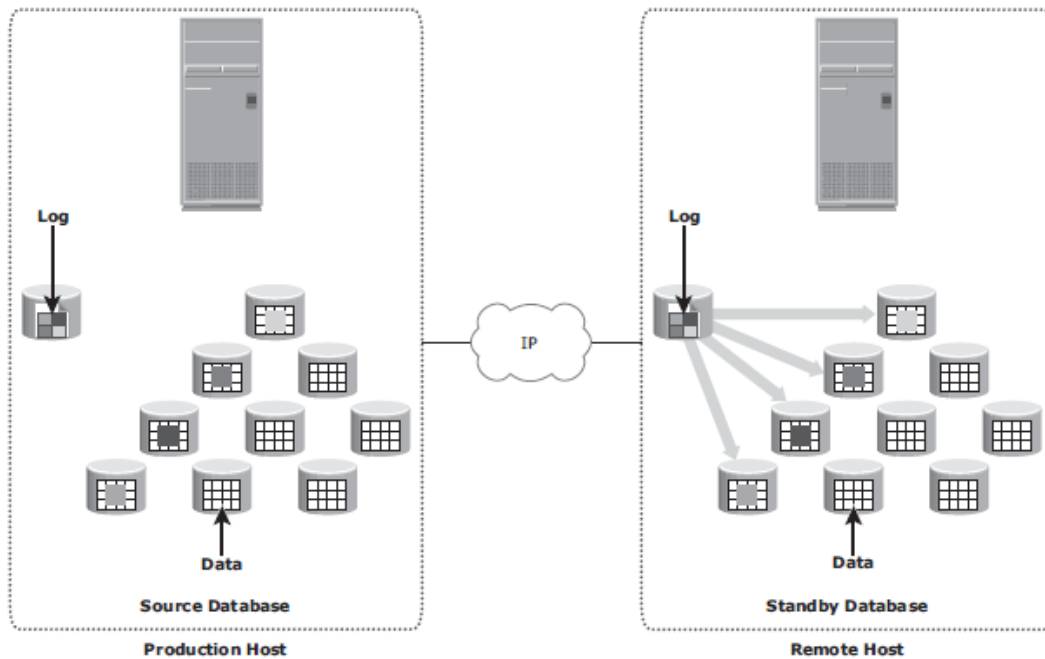
LVM-based remote replication supports both synchronous and asynchronous modes of replication. If a failure occurs at the source site, applications can be restarted on the remote host, using the data on the remote replicas.

LVM-based remote replication is independent of the storage arrays and therefore supports replication between heterogeneous storage arrays. Most operating systems are shipped with LVMs, so additional licenses and specialized hardware are not typically required.

The replication process adds overhead on the host CPUs. CPU resources on the source host are shared between replication tasks and applications. This might cause performance degradation to the applications running on the host. Because the remote host is also involved in the replication process, it must be continuously up and available.

### ***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 (see Figure 12-6). The remote host receives the logs and applies them to the remote database.



**Figure 12-6:** Host-based log shipping

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.

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 file 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, rate of updates to the source database, and the frequency of log switching should be considered when determining the optimal size of the log file.

Similar to LVM-based remote replication, the existing standard IP network can be used for replicating log files. Host-based log shipping requires low network bandwidth because it transmits only the log files at regular intervals.

**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 used for applications running on the host. A source and its replica device reside on different storage arrays. Data can be transmitted from the source storage array to the target storage array over a shared or a dedicated network.

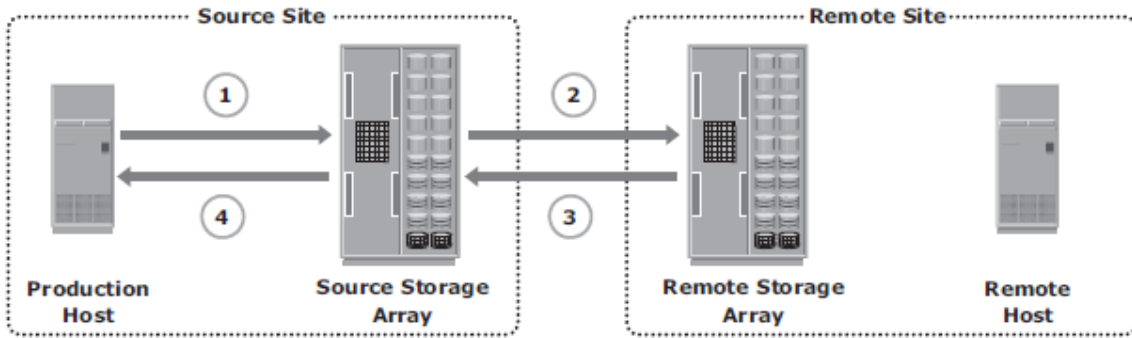
Replication between arrays may be performed in synchronous, asynchronous, or disk-buffered modes.

***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 production host. Additional writes on that source cannot occur until each preceding write has been completed and acknowledged. Figure 12-7 shows the array-based synchronous remote replication process.

In the case of synchronous remote 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 destage 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 keeps track of the writes that are not transmitted to the remote storage array. When the network links are restored, the accumulated data is 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.

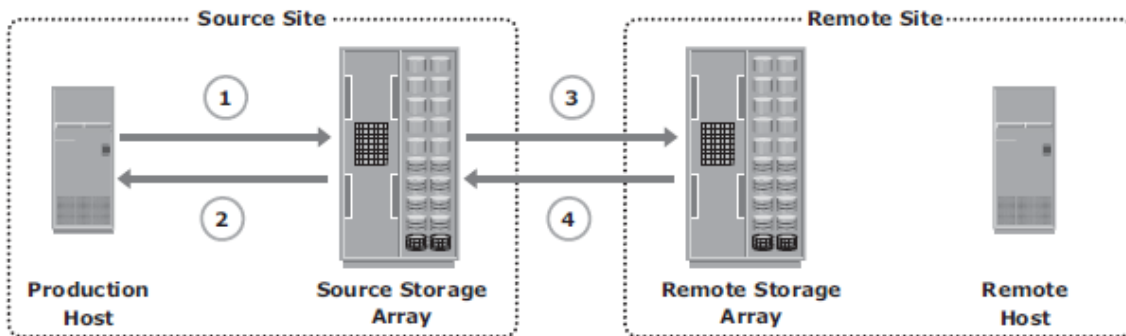


- ① Write from the production host is received by the source storage array.
- ② Write is then transmitted to the remote storage array.
- ③ Acknowledgment is sent to the source storage array by the remote storage array.
- ④ Source storage array signals write-completion to the production host.

**Figure 12-7:** Array-based synchronous remote replication

**Asynchronous Replication Mode**

In array-based *asynchronous remote replication mode*, as shown in Figure 12-8, 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.



- ① The production host writes to the source storage array.
- ② The source array immediately acknowledges the production host.
- ③ These writes are then transmitted to the target array.
- ④ After the writes are received by the target array, it sends an acknowledgment to the source array.

**Figure 12-8:** Array-based asynchronous remote replication

Similar to synchronous replication, asynchronous replication writes are placed in cache on the two arrays and are later destaged to the appropriate disks. Some implementations of

asynchronous remote replication maintain write ordering. A timestamp 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 in most DBMSs. In asynchronous remote replication, 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.

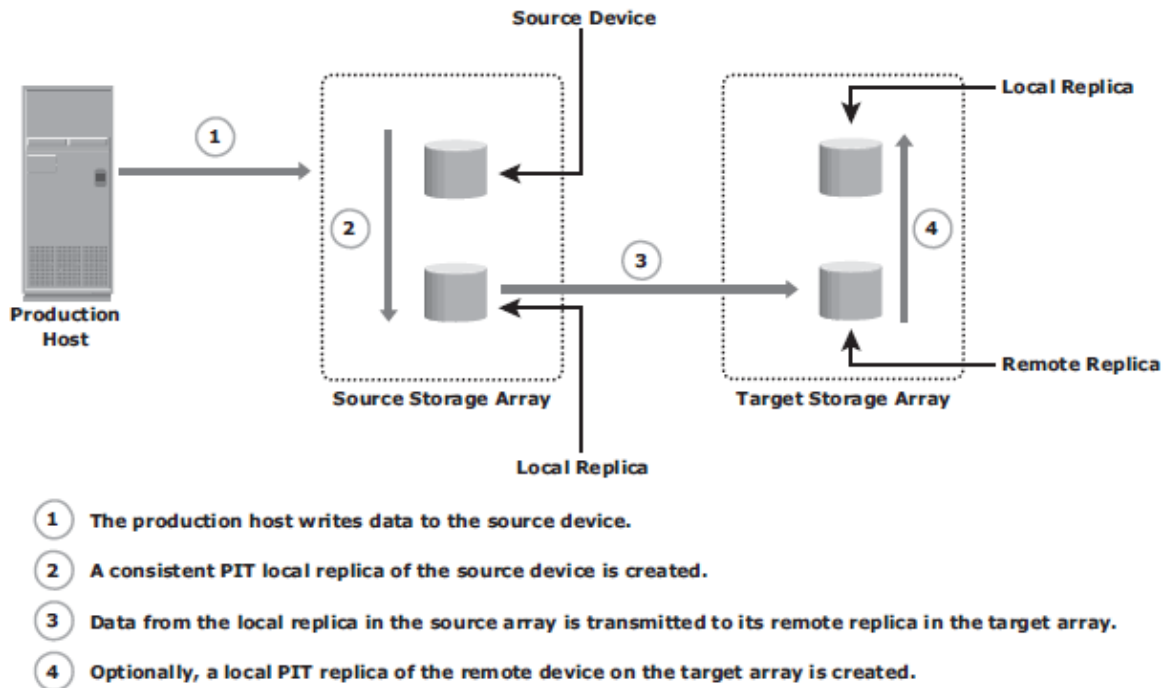
Asynchronous remote replication provides network bandwidth cost-savings because the required bandwidth is lower than the peak write workload. During times when the write workload exceeds the average bandwidth, sufficient buffer space must be configured on the source storage array to hold these writes.

#### ***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.

Figure 12-9 shows the sequence of operations in a disk-buffered remote replication. 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 runs 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 transmits 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 the available link bandwidth and the data change rate on the source device. Because disk-buffered technology uses local replication, changes made to the source and its replica are possible to track. Therefore, all the resynchronization operations between the source and target can be done incrementally. When compared to synchronous and asynchronous replications, disk-buffered remote replication requires less bandwidth. In disk-buffered remote replication, the RPO at the remote site is in the order of hours. For example, a local replica of the source device is created at 10:00 a.m., and this data transmits to the remote replica, which takes 1 hour to complete. Changes made to the source device after 10:00 a.m. are tracked. Another local replica of the source device is

created at 11:00 a.m. by applying track changes between the source and local replica (10:00 a.m. copy). During the next cycle of transmission (11:00 a.m. data), the source data has moved to 12:00 p.m. The local replica in the remote array has the 10:00 a.m. data until the 11:00 a.m. 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 2 hours because the remote site has 10:00 a.m. data.



**Figure 12-9:** Disk-buffered remote replication

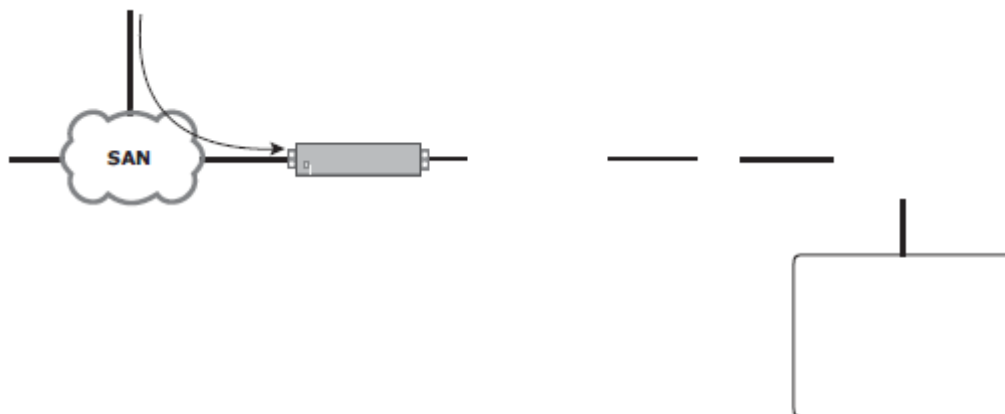
### Network-Based Remote Replication

In network-based remote replication, the replication occurs at the network layer between the host and storage array. Continuous data protection technology provides solutions for network-based remote replication.

#### ***CDP Remote Replication***

In normal operation, CDP remote replication provides any-point-in-time recovery capability, which enables the target LUNs to be rolled back to any previous point in time. Similar to CDP local replication, CDP remote replication typically uses a *journal volume*, *CDP appliance*, or CDP software installed on a separate host (*host-based CDP*), and a *write splitter* to perform replication between sites. The CDP appliance is maintained at both source and remote sites.

Figure 12-10 describes CDP remote replication. In this method, the replica is synchronized with the source, and then the replication process starts. After the replication starts, all the writes from the host to the source are split into two copies. One of the copies is sent to the local CDP appliance at the source site, and the other copy is sent to the production volume. After receiving the write, the appliance at the source site sends it to the appliance at the remote site. Then, the write is applied to the journal volume at the remote site. For an asynchronous operation, writes at the source CDP appliance are accumulated, and redundant blocks are eliminated. Then, the writes are sequenced and stored with their corresponding timestamp. The data is then compressed, and a checksum is generated. It is then scheduled for delivery across the IP or FC network to the remote CDP appliance. After the data is received, the remote appliance verifies the checksum to ensure the integrity of the data. The data is then uncompressed and written to the remote journal volume. As a next step, data from the journal volume is sent to the replica at predefined intervals.



**Figure 12-10: CDP remote replication**

---

## Securing the Storage Infrastructure

Valuable information, including intellectual property, personal identities, and financial transactions, is routinely processed and stored in storage arrays, which are accessed through the network. As a result, storage is now more exposed to various security threats that can potentially damage business-critical data and disrupt critical services. Securing storage infrastructure has become an integral component of the storage management process in traditional and virtualized data centers. It is an intensive and necessary task, essential to managing and protecting vital information. Storage security in a public cloud environment is more complex because organizations have less control over the shared IT infrastructure and security measures' enforcement. Further, multitenancy in a cloud environment enables resource sharing, including storage among multiple consumers. Such sharing might pose a threat of commingling data across tenants.

### 14.1 Information Security Framework

The basic information security framework is built to achieve four security goals: confidentiality, integrity, and availability (CIA), along with accountability. This framework incorporates all security standards, procedures, and controls, required to mitigate threats in the storage infrastructure environment.

- **Confidentiality:** Provides the required secrecy of information and ensures that only authorized users have access to data. This requires authentication of users who need to access information. Data in transit (data transmitted over cables) and data at rest (data residing on a primary storage, backup media, or in the archives) can be encrypted to maintain its confidentiality. In addition to restricting unauthorized users from accessing information, confidentiality also requires implementing 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:** Ensures that the information is unaltered. Ensuring integrity requires detection of and protection against unauthorized alteration or deletion of information. Ensuring integrity stipulates measures such as error detection and correction for both data and systems.
- **Availability:** This ensures that authorized users have reliable and timely access to systems, data, and applications residing on these systems. Availability requires protection against unauthorized deletion of data and denial of service. Availability also implies that sufficient resources are available to provide a service.

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

## 14.2 Risk Triad

Risk triad defines risk in terms of threats, assets, and vulnerabilities. Risk arises when a threat agent (an attacker) uses an existing vulnerability to compromise the security services of an asset, for example, if a sensitive document is transmitted without any protection over an insecure channel, an attacker might get unauthorized access to the document and may violate its confidentiality and integrity. This may, in turn, result in business loss for the organization. In this scenario potential business loss is the risk, which arises because an attacker uses the vulnerability of the unprotected communication to access the document and tamper with it.

To manage risks, organizations primarily focus on vulnerabilities because they cannot eliminate threat agents that appear in various forms and sources to its assets. Organizations can enforce countermeasures to reduce the possibility of occurrence of attacks and the severity of their impact.

Risk assessment is the first step to determine 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. Based on the value of assets, risk assessment helps to prioritize investment in and provisioning of security measures. To determine the probability of an adverse event occurring, threats to an IT system must be analyzed 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 vulnerabilities are considered from the perspective of risk identification and control analysis.

### 14.2.1 Assets

Information is one of the most important *assets* for any organization. Other assets include hardware, software, and other infrastructure components required to access the 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, network infrastructure, and organizational policies.

Security methods have two objectives. The 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.

The second objective is to make it difficult for potential attackers to access and compromise the system. The security methods should provide adequate protection against unauthorized access, viruses, worms, trojans, and other malicious software programs. Security measures should also include options to 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 data compromise. For the security system to function smoothly, all users are informed about the policies governing the use of the network. The effectiveness of a storage security methodology can be measured by two key criteria. One, the cost of implementing the system should be a fraction of the value of the protected data. Two, it should cost heavily to a potential attacker, in terms of money, effort, and time.

### 14.2.2 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, availability, and accountability.

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

*Denial of service* (DoS) attacks prevent legitimate users from accessing resources and services. These attacks generally do not involve access to or modification of information. 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 information. It attempts to provide false information by either impersonating someone or denying that an event or a transaction has

taken place. For example, a repudiation attack may involve performing an action and eliminating any evidence that could prove the identity of the user (attacker) who performed that action. Repudiation attacks include circumventing the logging of security events or tampering with the security log to conceal the identity of the attacker.

### 14.2.3 Vulnerability

The paths that provide access to information are often vulnerable to potential attacks. Each of the paths may contain various access points, which provide different levels of access to the storage resources. It is 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 known as *defense in depth*.

Defense in depth recommends using multiple security measures to reduce the risk of security threats if one component of the protection is compromised. It is also known as a “layered approach to security.” Because there are multiple measures for security at different levels, defense in depth gives additional time to detect and respond to an attack. This can reduce the scope or impact of a security breach.

*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. An attacker can use all the external interfaces supported by that component, such as the hardware and the management interfaces, 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 may 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 environment, organizations can deploy specific control measures. Any control measures should involve all the three aspects of infrastructure: people, process, and technology, and the relationships among them. To secure people, the first step is to establish and assure their identity. Based on their identity, selective controls can be implemented for their access to data and resources. The effectiveness of any security measure is primarily governed by processes and policies. The processes should be based on a thorough understanding of risks in the environment and should recognize the relative sensitivity of different types of data and the needs of various stakeholders to access the data. Without an effective process, the deployment of technology is neither cost-effective nor aligned to organizations' priorities.

Finally, the technologies or controls that are deployed should ensure compliance with the processes, policies, and people for its effectiveness. These security technologies are directed at reducing vulnerability by minimizing attack surfaces and maximizing the work factors. 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 various operations. Physical controls include setting up physical barriers, such as security guards, fences, or locks.

Based on the roles they play, controls are categorized as preventive, detective, and corrective. 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, whereas *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.

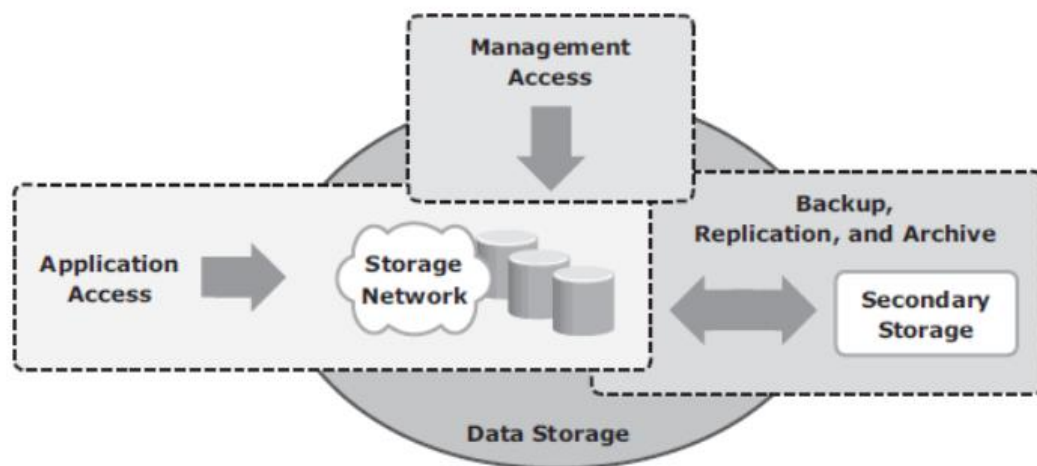
### 14.3 Storage Security Domains

Storage devices connected to a network raise the risk level and are more 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, ensure that these controls are not compromised. If each component within the storage network is considered a potential access point, the attack surface of all these access points must be analyzed to identify the associated vulnerabilities.

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 *backup, replication, and archive*. Figure 14-1 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 backup, replication, and archive access. Along with the access points in this domain, the backup media also needs to be secured.



**Figure 14-1:** Storage security domains

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.

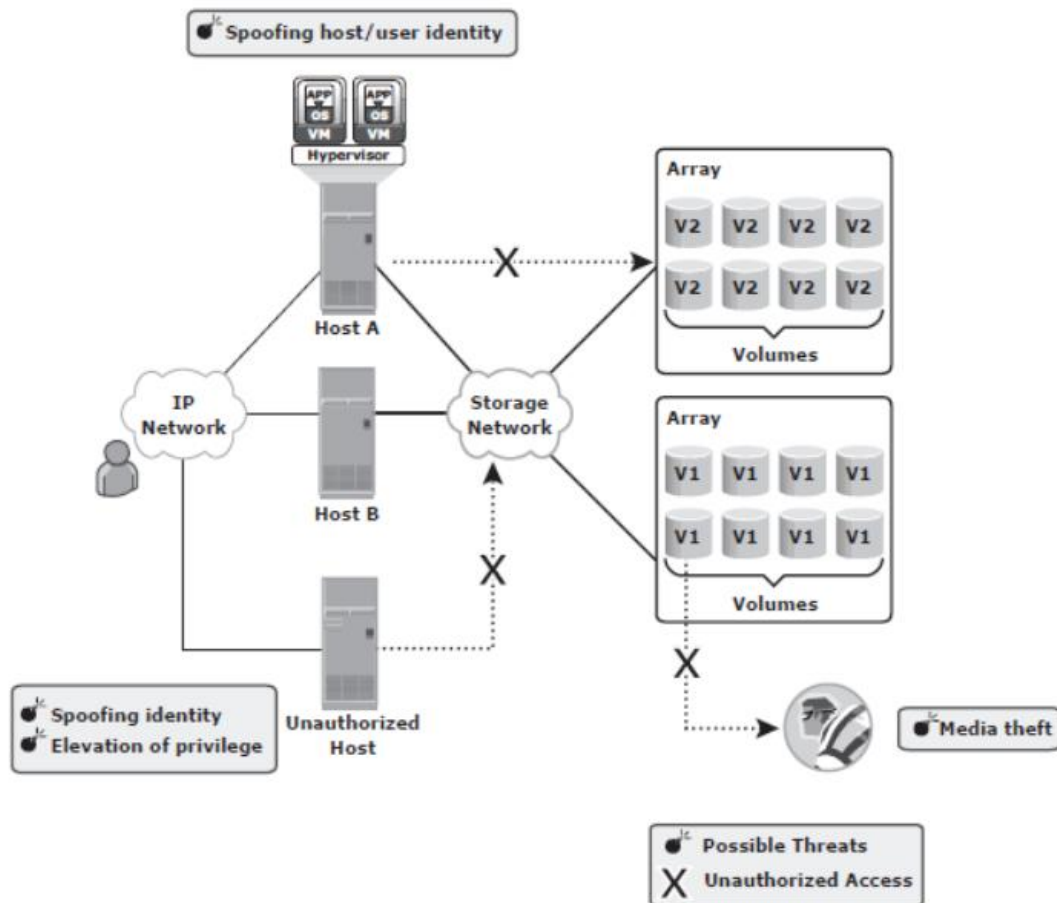
The next step is to select and implement various controls as countermeasures to the threats.

### 14.3.1 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. An important step to secure the application access domain is to identify the threats in the environment and appropriate controls that should be applied. Implementing physical security is also an important consideration to prevent media theft.

Figure 14-2 shows application access in a storage networking environment. Host A can access all V1 volumes; host B can access all V2 volumes. These volumes are classified according to the access level, such as confidential, restricted, and public. Some of the possible threats in this scenario could be host A spoofing the identity or elevating to the privileges of host B to gain access to host B’s resources.

Another threat could be that an unauthorized host gains access to the network; the attacker on this host may try to spoof the identity of another host and tamper with the 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; therefore, they need to be addressed.



**Figure 14-2:** Security threats in an application access domain

**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 these threats affect data integrity and confidentiality.

Access control mechanisms used in the application access domain are user and host authentication (technical control) and authorization (administrative 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 regulate user access to specific files. The Enterprise Content Management application enforces access to data by using Information Rights Management (IRM) that specifies 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 important to ensure that administrative controls, such as defined security 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 logs should also be protected from unauthorized access because they may fail to achieve their goals if the logged content is exposed to unauthorized 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 loss of confidentiality.

The security controls for protecting the network fall into two general categories: *network infrastructure integrity* and *storage network encryption*. Controls for ensuring the infrastructure integrity include a fabric switch function that ensures 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 every user. Instead, *role-based access control (RBAC)* is deployed to assign necessary privileges to users, enabling them to perform their roles. A role may represent a job function, for example, an administrator. Privileges are associated with the roles and users acquire these privileges based upon 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 can 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, and 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; 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

life cycle 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.

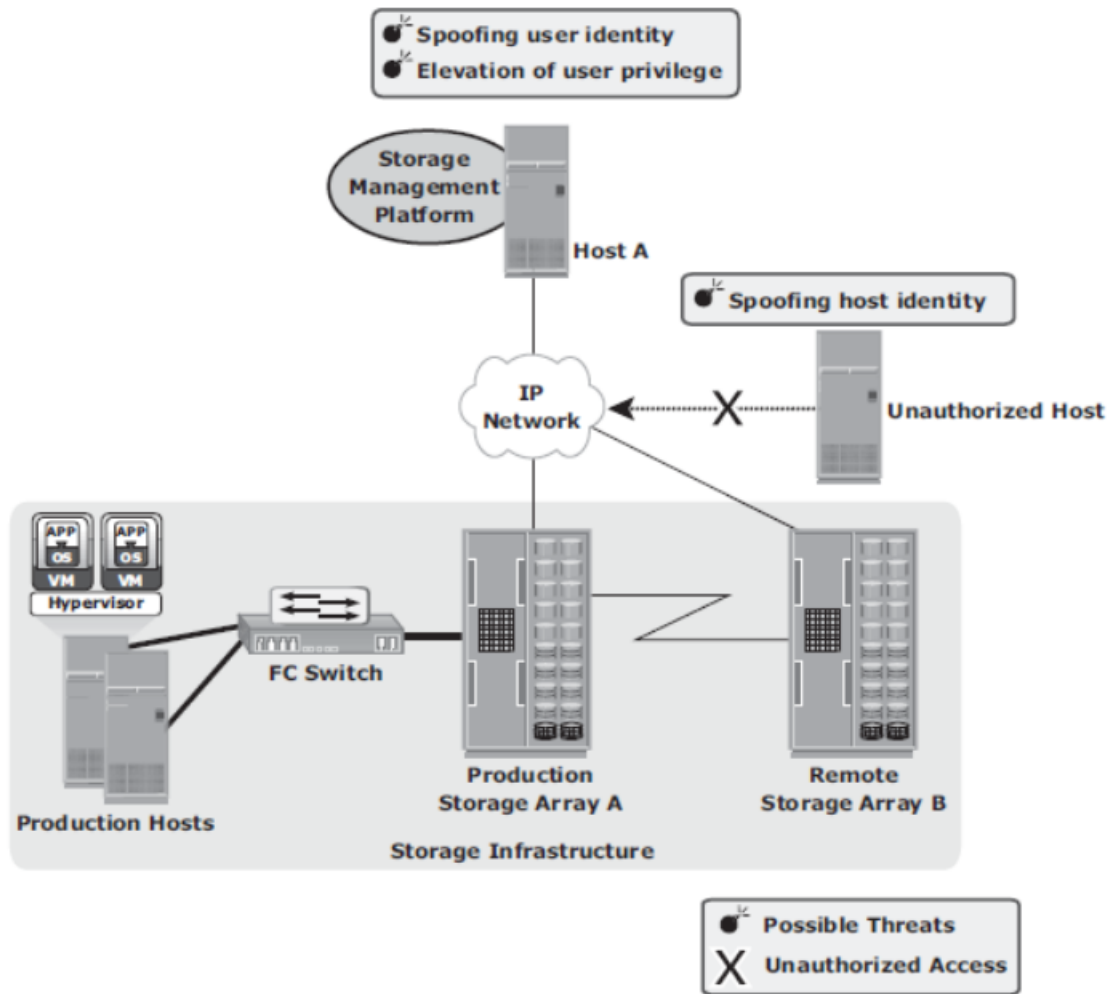
On NAS devices, adding antivirus checks and file extension controls can further enhance data integrity. In the case of CAS, use of MD5 or SHA-256 cryptographic algorithms guarantees data integrity by detecting any change in content bit patterns. In addition, the data erasure service ensures that the data has been completely overwritten by bit sequence before the disk is discarded.

An organization's data classification policy determines whether the disk should actually be scrubbed prior to discarding it and the level of erasure needed based on regulatory requirements.

### **14.3.2 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. Implementing appropriate controls for securing storage management applications is important because the damage that can be caused by using these applications can be far more extensive.

Figure 14-3 depicts a storage networking environment in which production hosts are connected to a SAN fabric and are accessing production storage array A, which is connected to remote storage array B for replication purposes. Further, this configuration has a storage management platform on Host A. A possible threat in this environment is an unauthorized host spoofing the user or host identity to manage the storage arrays or network. For example, an unauthorized host may gain management access to remote array B.



**Figure 14-3:** Security threats in a management access domain

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. Event logs should be placed outside the shared storage systems where they can be reviewed if the storage is compromised. 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 that 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 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 of users and processes. Access control should be enforced for each storage component.

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 administrative 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 and 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*

Mechanisms 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 the traffic to certain devices. 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. 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 the commands that can be issued from each host. A separate private management network is highly recommended for management traffic. If possible, management traffic should not be mixed with either production data traffic or other LAN traffic used in the enterprise. 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.

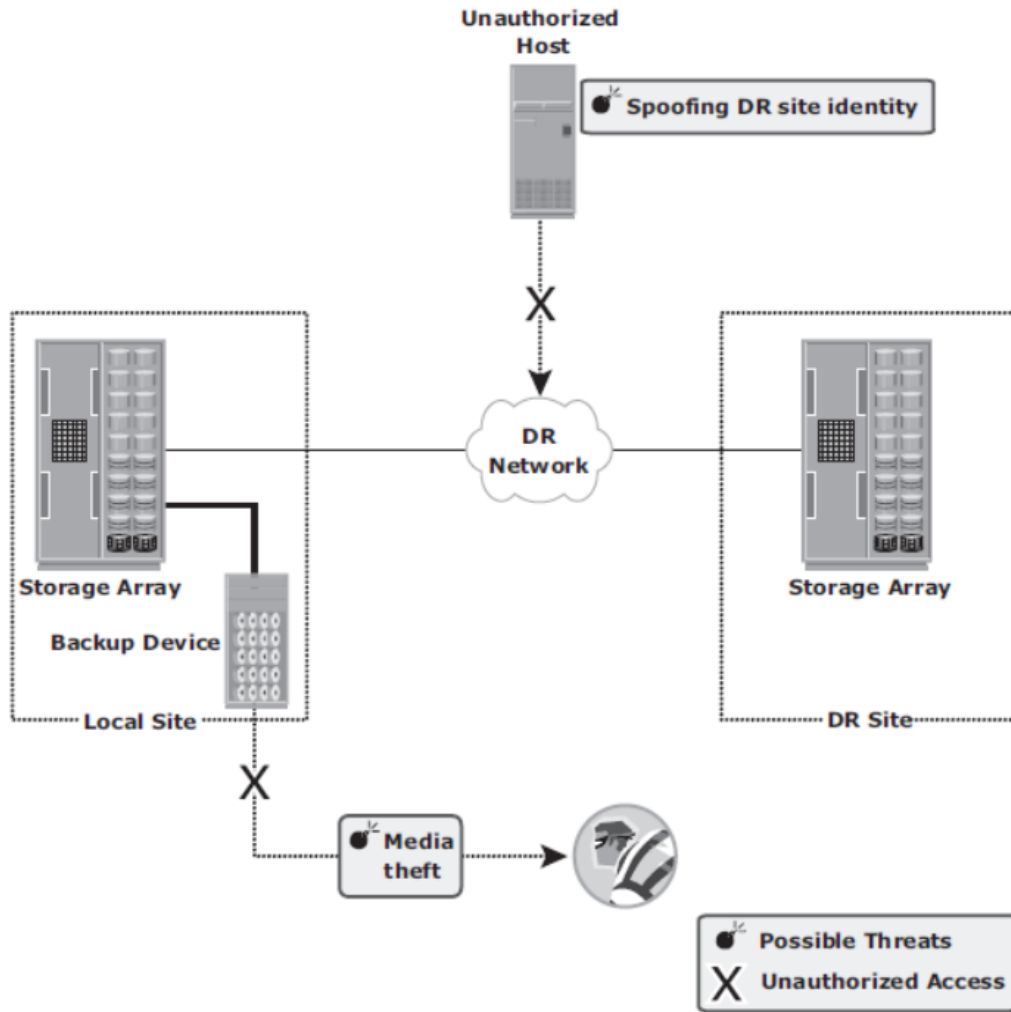
To summarize, security enforcement must focus on the management communication between devices, confidentiality and integrity of management data, and availability of management networks and devices.

### 14.3.3 Securing Backup, Replication, and Archive

Backup, replication, and archive is the third domain that needs to be secured against an attack. As explained in Chapter 10, a backup involves copying the data from a storage array to backup media, such as tapes or disks. Securing backup is complex and is based on the backup software that accesses 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 disaster recovery (DR) site maintains the same level of security for the backed up data. Protecting the backup, replication, and archive 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. Figure 14-4 illustrates a generic remote backup design whereby data on a storage array is replicated over a DR network to a secondary storage at the DR site. 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 identity of the backup server and request the host to send its data. The unauthorized host claiming to be the backup server may lead to a remote backup being performed to an unauthorized and unknown site. In addition, attackers can use the DR network connection to tamper with data, snoop the network, 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.



**Figure 14-4:** Security threats in a backup, replication, and archive environment

### 14.4 Security Implementations in Storage

#### Networking

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

#### 14.4.1 FC SAN

Traditional FC SANs enjoy an inherent 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 converged networks and 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 FC SANs. Many FC SAN security mechanisms have evolved from their counterpart in IP networking, thereby bringing in matured security solutions.

*Fibre Channel Security Protocol (FC-SP)* standards (T11 standards), published in 2006, align security mechanisms and algorithms between IP and FC interconnects. These standards describe protocols to implement security measures in a 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.

### ***FC 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 14-5 illustrates various levels (zones) of a storage networking environment that must be secured and the security measures that can be deployed. FC 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. 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 username/password and an additional security component such as a smart card for authentication.

### ***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.

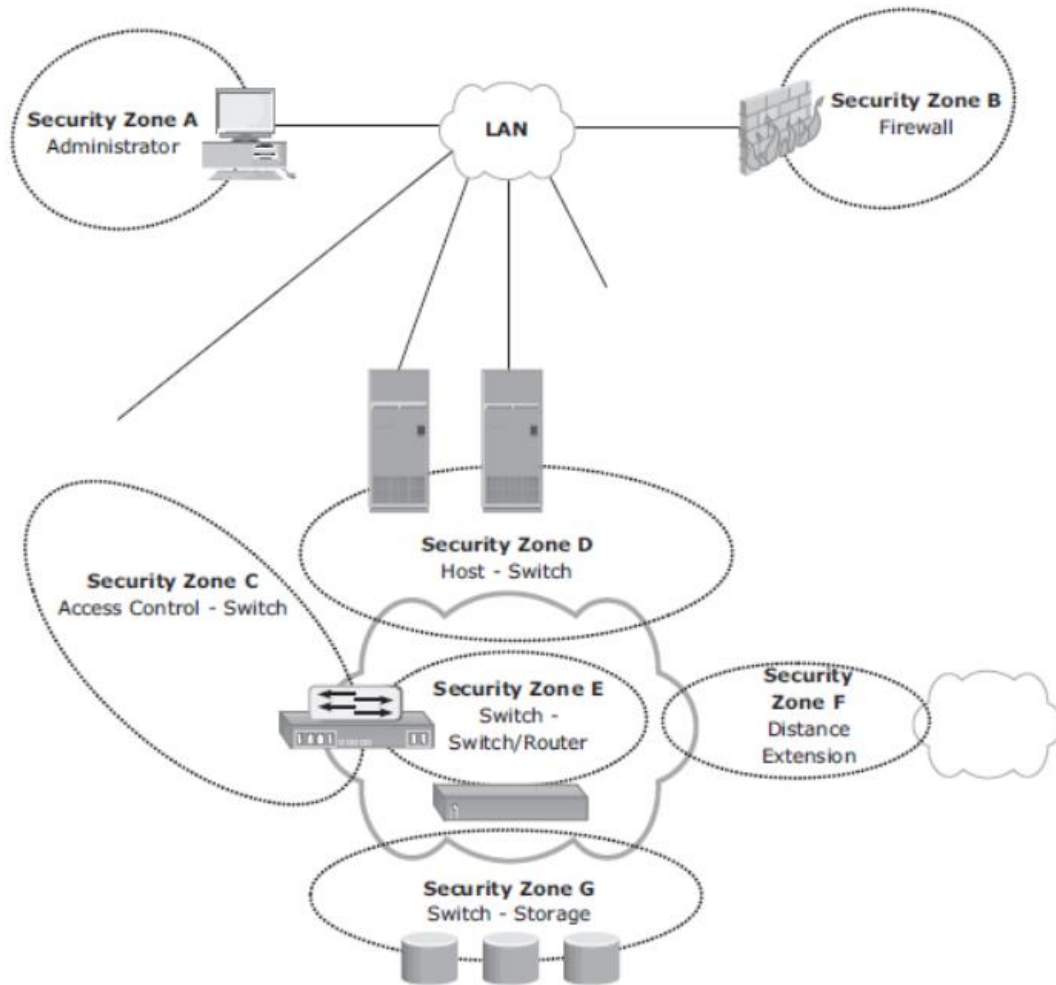


Figure 14-5: FC SAN security architecture

SECURITY ZONES	PROTECTION STRATEGIES
Zone D (Host to switch)	Restrict Fabric access to legitimate hosts by (a) implementing ACLs: Known HBAs can connect on specific switch ports only; and (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 fabric by (a) using E_Port authentication; (b) encrypting the traffic in transit; and (c) implementing FC switch controls and port controls.
Zone F (Distance Extension)	Implement encryption for in-flight data (a) FC-SP for long-distance FC extension; and (b) IPsec for SAN extension via FCIP.
Zone G (Switch to Storage)	Protect the storage arrays on your SAN via (a) WWPN-based LUN masking; and (b) S_ID locking: masking based on source FC address.

### ***LUN Masking and Zoning***

LUN masking and zoning are the basic SAN security mechanisms used to protect against unauthorized access to storage. The standard implementations of LUN masking on storage arrays mask the LUNs presented to a frontend storage port based on the WWPNs of the source HBAs. A stronger variant of LUN masking may sometimes be offered whereby masking can be done on the basis of source FC addresses. It offers a mechanism to lock down the FC address of a given node port to its WWN. *WWPN zoning* is the preferred choice in security-conscious environments.

### ***Securing Switch Ports***

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 from accessing it. Similarly, the switch connection control policy specifies which switches are allowed to be part of the fabric, preventing unauthorized switches 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 any attempt to connect any switch in the fabric by using an ISL causes the fabric to segment.

Role-based access control provides additional security to a SAN by preventing unauthorized 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 can modify the zones on the fabric, whereas a basic user may view only fabric related information, such as port types and logged-in nodes.

#### ***Logical Partitioning of a Fabric: Virtual SAN***

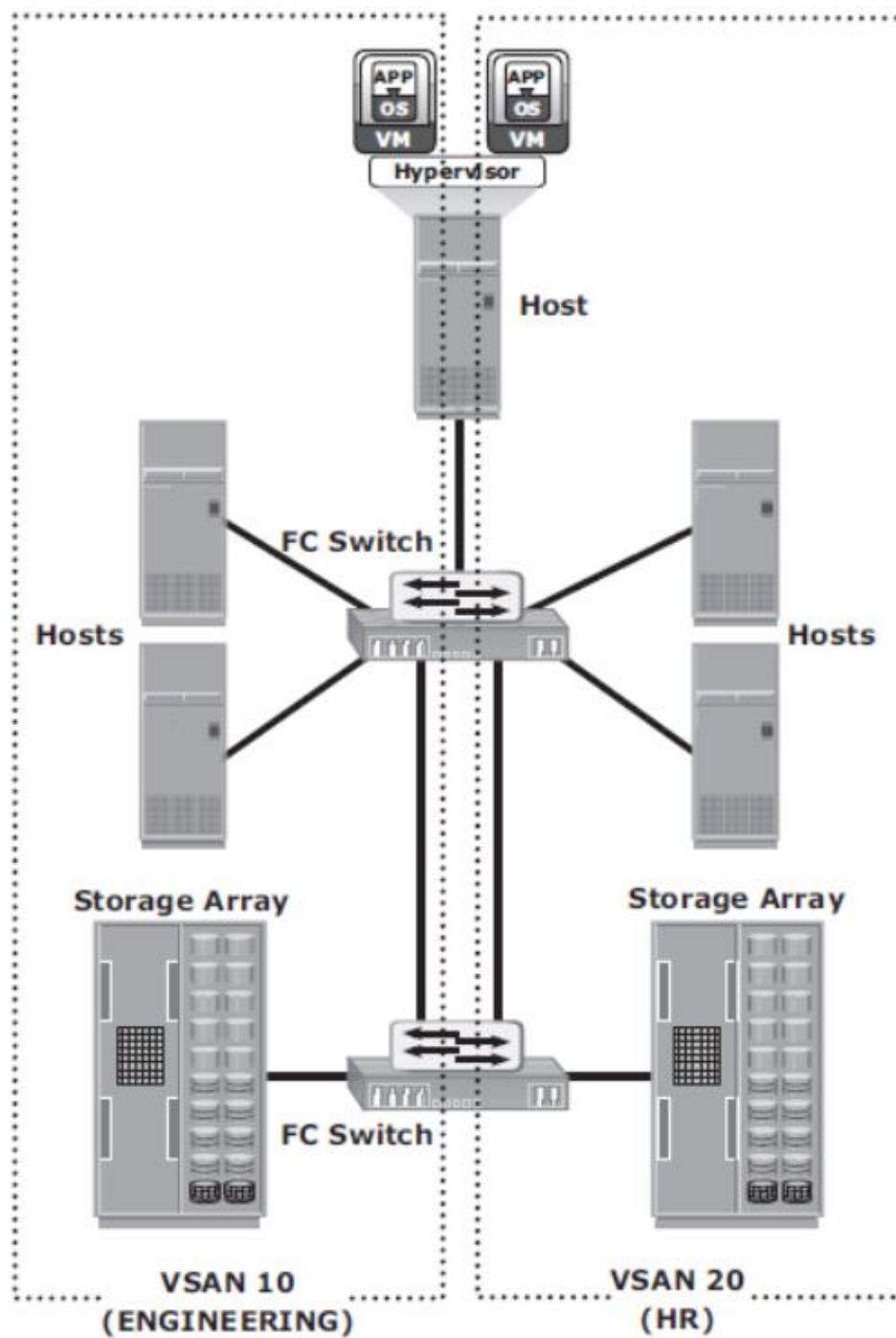
VSANs enable the creation of multiple logical SANs over a common physical SAN. They provide the capability to build larger consolidated fabrics and still maintain the required security and isolation between them. Figure 14-6 depicts logical partitioning in a VSAN.

The SAN administrator can create distinct VSANs by populating each of them with switch ports. In the example, the switch ports are distributed over two VSANs: 10 and 20 — for the Engineering and HR divisions, respectively. Although they share physical switching gear with other divisions, they can be managed individually as standalone fabrics. 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.

VSANs minimize the impact of fabricwide 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 authorization control within a single fabric.

#### **14.4.2 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.



**Figure 14-6:** Securing SAN with VSAN

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 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, that determines 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 need to be explicitly granted in the SACL. The owner, SACL, and DACL are all statically held as attributes 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 security identifiers (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 login ID as “User1,” it is simply a textual representation of the true SID, which is used by the underlying operating system. Internal processes in Windows refer to an account’s SID rather than the account’s username or group name while granting access to an object. 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***

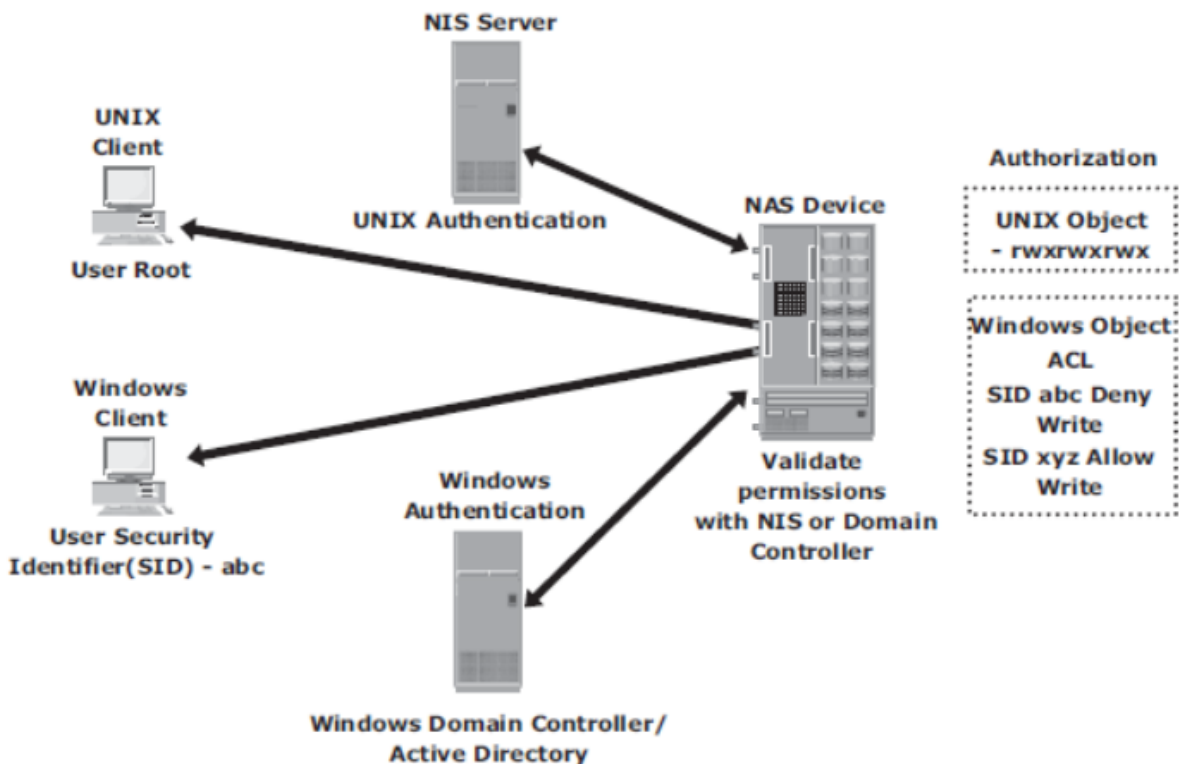
For the UNIX operating system, a *user* is an abstraction that denotes a logical entity for assignment of ownership and operation privileges for the system. A user can be either a person or a system operation. A UNIX system is only aware of the privileges of the user to perform specific operations on the system and identifies each user by a user ID (UID) and a username, regardless of whether it is a person, a system operation, or a device. In UNIX, users can be organized into one or more groups. The concept of group serves the purpose to assign 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. Because 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 read or execute only. When displayed, a character denoting the mode of the file may precede this nine-bit 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.”

**NAS File Sharing: 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 filesharing 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. Figure 14-7 depicts the authentication process in a NAS environment



**Figure 14-7:** 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, consider these complexities of multiprotocol support when designing a NAS solution. At the same time, validate the domain controller and NIS server connectivity and bandwidth.

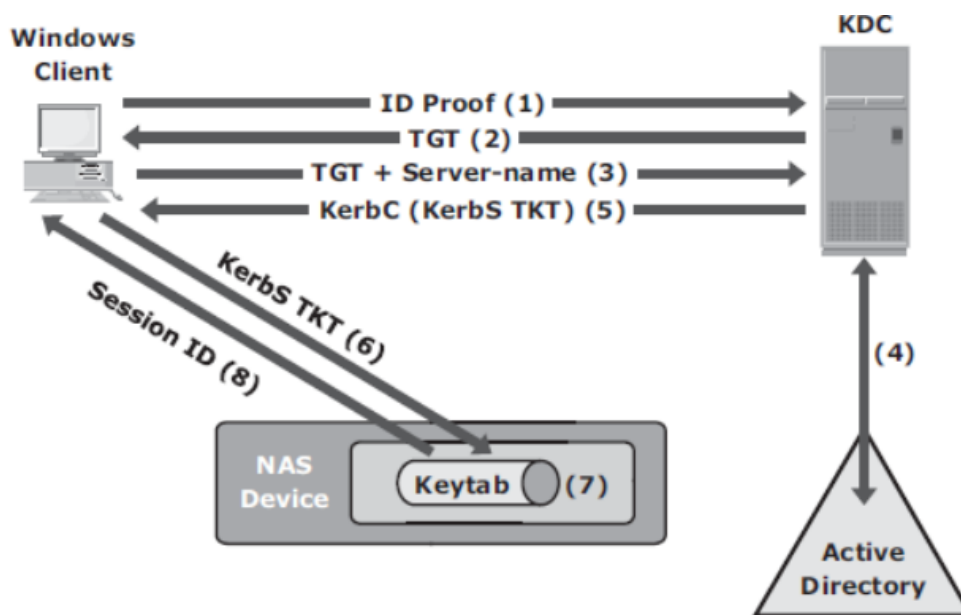
If multiprotocol access is required, specific vendor access policy implementations need to be considered.

***Kerberos***

Kerberos is a network authentication protocol, which 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 identities, they can choose to encrypt all their communications to ensure privacy and data integrity.

In Kerberos, 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 Kerberos, users and servers for which a secret key is stored in the KDC database are known as *principals*. 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 authentication process shown in Figure 14-8 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.
2. The KDC responds with an encrypted Ticket Granting Ticket (TGT) and an encrypted session key. TGT has a limited validity period. TGT can be decrypted only by the KDC, and the client can decrypt only the session key.
3. When the client requests a service from a server, it sends a request, consisting of the previously generated TGT, encrypted with the session key 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 hosting the service.
6. The client then sends the service ticket to the server that houses the required 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 tracks the client activity, such as file locking, as long as the session is active.

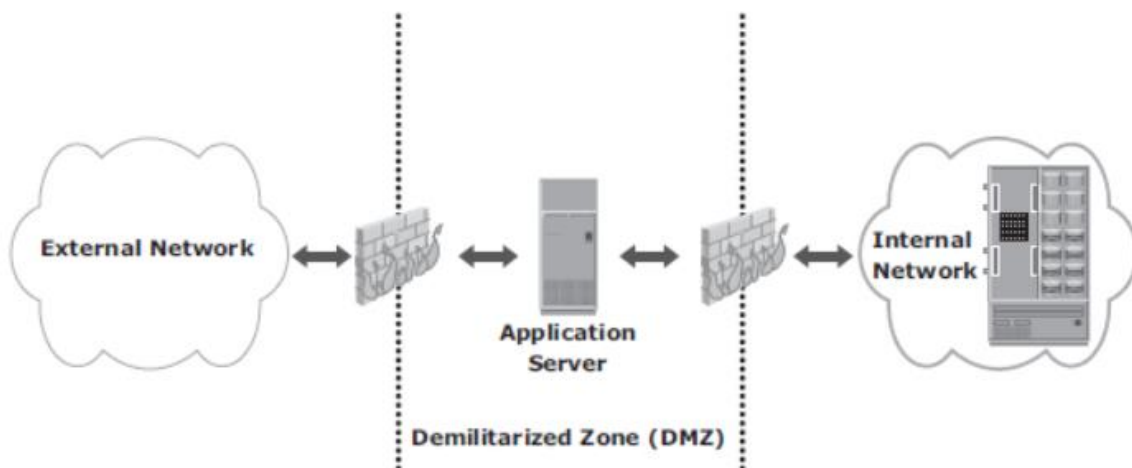


**Figure 14-8:** Kerberos authorization

### *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 can examine network packets and compare 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 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 14-9 depicts a typical firewall implementation. A demilitarized zone (DMZ) is commonly used in networking environments. A DMZ provides a means to secure 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.

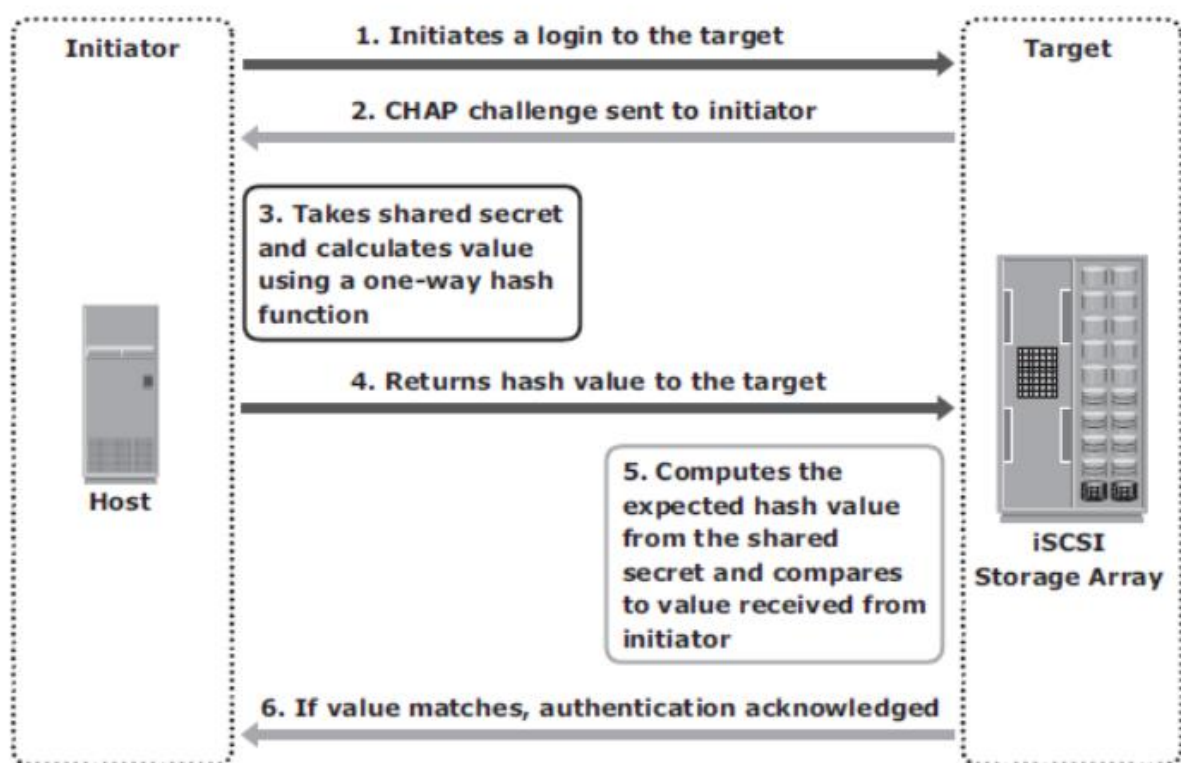


**Figure 14-9:** Securing a NAS environment with a network-layer firewall

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 accessing data stored on a NAS device, which may be located on the internal private network. A secure design would serve only data to internal and external applications through the DMZ.

#### 14.4.3 IP SAN

This section describes some of the basic security mechanisms used in 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 communication channel; 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 14-10 depicts the CHAP authentication process.

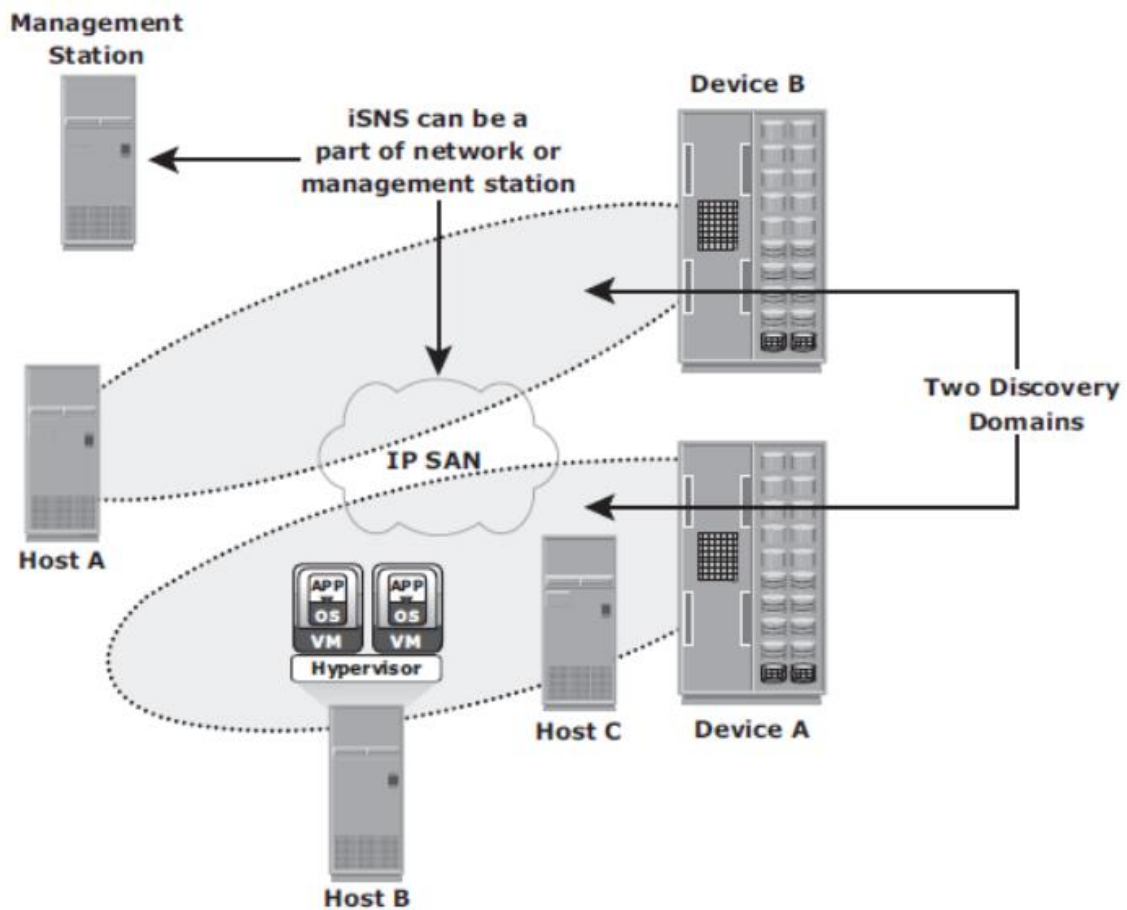


**Figure 14-10:** Securing IPSAN with CHAP authentication

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, composed of 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. For devices to communicate with one another, they must be configured in the same discovery domain. State change notifications (SCNs) inform the iSNS server when devices are added to or removed from a discovery domain. Figure 14-11 depicts the discovery domains in iSNS.



**Figure 14-11:** Securing IPSAN with iSNS discovery domains

---

## Question Bank

- 1) Explain local Replication technology using Host based methods.
- 2) List various uses of local replication.
- 3) Explain storage array based local replication with a neat diagram.
- 4) Recall about FCSAN based security implementation.
- 5) Tell about Host-based Remote Replication and Network-based Remote Replication in detail.
- 6) List and explain about Risk Triad in securing the Storage Infrastructure.
- 7) Explain the IPSAN security implementation in storage networking.
- 8) Explain Network-based Local Replication.
- 9) Show the two modes of Remote Replication and write about them.
- 10) Explain the NAS security implementation in Storage Networking.