IA 313: Operating Systems Security

Lecture 3: File System Security

What does a file system do?

- A **file system** is a method for storing and organizing computer files and the data they contain to make it easy to find and access them.
- File systems exist on hard drives, pen drives, CD's, DVD's and any other form of data storage medium.
- Most data storage devices have array of fixed-size blocks, sometimes called sectors, and file system is in charge of organizing these sectors into files and directories.
- It is also in charge of indexing the media so it knows where and what each file is.

Types of File Systems

- Disk file systems – FAT (File Allocation Table), NTFS, HFS (Hierarchical File System), ext2, ext3, ISO9660 and UDF
- FAT(FAT12, FAT16, FAT32), and especially NTFS are primarily used on Windows operating systems
- FAT is also the standard file system for floppy drives and is still used today
- HFS is used by Mac OS, and ext2, ext3 are used on various linux operating systems
- ISO9660 and UDF are used on optical media

How does the file system handle security?

- The file system is crucial to data integrity.
- Main method of protection is through access control
  - Accessing file system operations (ex. modifying or deleting a file) are controlled through access control lists or capabilities
  - Capabilities are more secure so they tend to be used by operating systems on file systems like NTFS or ext3.
- Secondary method of protection is through the use of backup and recovery systems

Attacks on the file system

- Race Condition Attacks
- Using ADS to hide files
- Directory traversal

Race Condition Attacks

- A race condition attack happens when a computing system that's designed to handle tasks in a specific sequence is forced to perform two or more operations simultaneously.
- This technique takes advantage of a time gap between the moment a service is initiated and the moment a security control takes effect.
- This attack, which depends on multithreaded applications, can be delivered in one of two ways: interference caused by untrusted processes (essentially a piece of code that slips into a sequence between steps of a secure programs), and interference caused by a trusted process, which may have the "same'' privileges.
- Without proper controls, different processes can interfere with each other.
- Other names used to refer to this vulnerability include Time of Check/Time of Use or TOC/TOU attacks.

Using ADS to hide Files

- Alternate Data Streams(ADS) allows multiple data streams to be attached to a single file.
- A file can be hidden behind a file as an attached stream that could be hundreds of megabytes in size, however a directory listing will only display the file's normal size.

Directory Traversal

- An exploit caused by lack of insufficient security validation of user supplied input file names
- For example the attacker would pass this as input. ../../../../../../../../../etc/password to retrieve the password file from the server.

How does the file system ensure data integrity?

- There are various methods of protecting the files on a file system.
  - Access Controls
  - Encryption
  - RAID
  - Recovery when data is corrupted

Access Control

- Access Control plays a huge part in file system security
- The system should only allow access to files that the user is permitted to access
- Almost all major file systems support ACL's or capabilities in order to prevent malicious activity on the file system
- Depending on the users rights they can be allowed to read, write and/or execute and object.
- In some file systems schemes only certain users are allowed to alter the ACL on a file or see if a file even exists.
- Ultimately the less the user has access to the less that can go wrong and the integrity of the disk can be more guaranteed.

General File System Encryption

- Encryption is also a method used by file systems to secure data, NTFS for example offers file encryption using DESX
- Two method of disk encryption
  - Full Disk Encryption
  - File System Encryption
- File system encryption has a few advantages over full disk encryption for example
  1. File based key management
  2. Individual management of encrypted files
  3. Access control can be further strengthened through the use of public key cryptography
  4. Keys are only held in memory while the file is being used

EFS: what is it?

- Encrypted File System (EFS) provides the core file encryption technology used to store encrypted files on the File System.
- Corporate world is very competitive, so any code, system specifications, often needs to be controlled.
- We have to share data among many users or groups, the potential risk for a computer security from a users perspective.
- Password Security – Does nothing to preventing a disk being mounted on a different system and reading the contents.

Encrypting File System(EFS)

- Provides security beyond user authentication and access control lists. For example when the attacker has physical access to the computer.
- EFS uses public key cryptography however it is susceptible to brute-force attacks against the user account passwords.

EFS: why do we need it?

- Security—First and Foremost
  - Secures Data from being accessed by any malicious user / hacker.
- Privacy
  - Ensure that private data is not accessed by other users (may not be malicious).
- Reliability – An integral component
  - Only responsible people are provided access to important data
- Resource Sharing
  - Many users can use the same system and still can work independently.

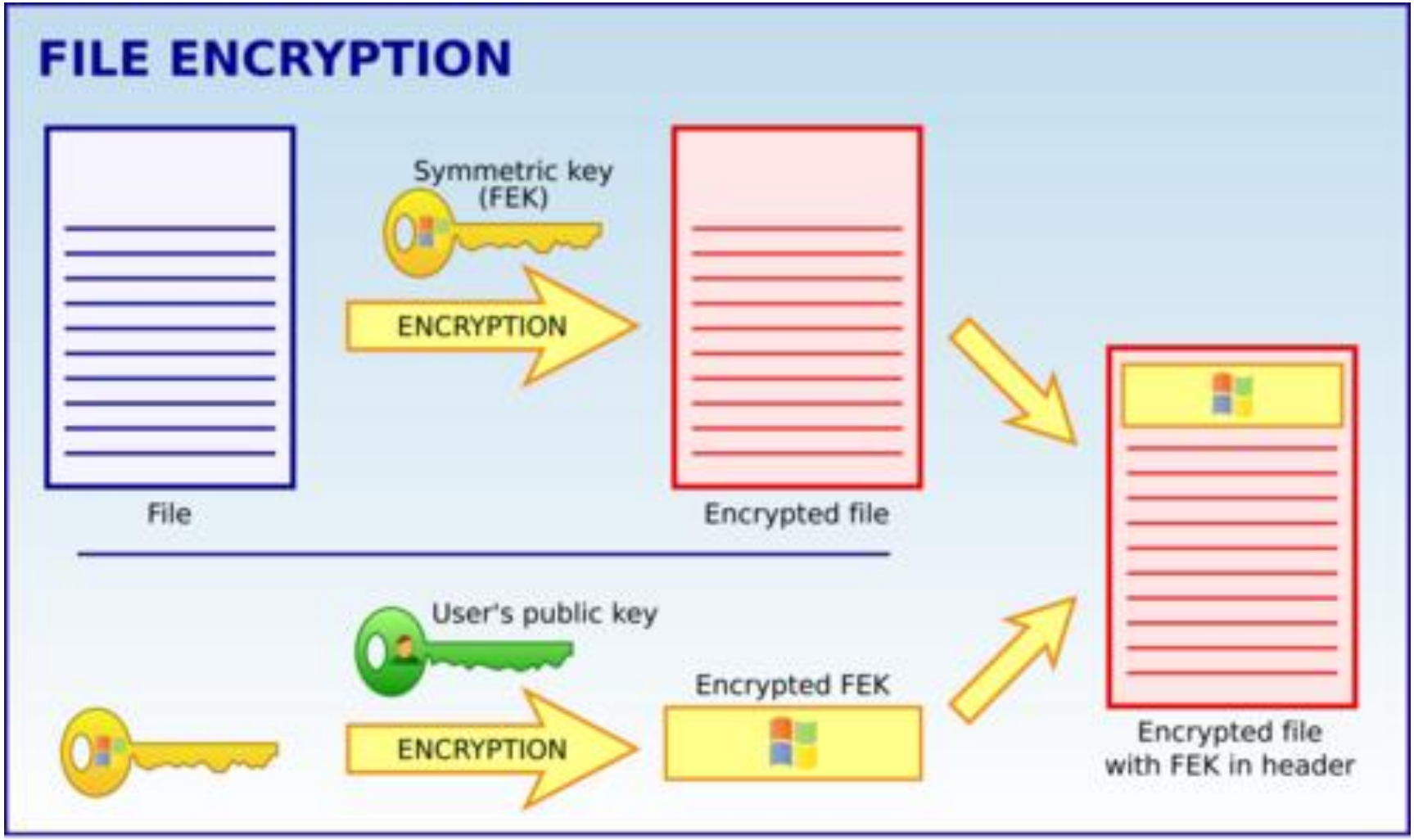
EFS: benefits

- The Disk Encryption reduce risk of data exposure in a specific, if uncommon, scenario.
- To avoid system risks such as:
  - Computer is bodily stolen.
  - Someone inside the company is trying to compromise information.
  - The system is cracked while attached to a network or with some malicious software.
- The primary benefit of the encrypted disk system is defense against device theft, and making your system a more secured one. Though, the risks are partially mitigated.

EFS Encryption

- EFS works by encrypting a file with a bulk symmetric key, aka File Encryption Key or FEK.
- The FEK is encrypted with a public key that is associated with the user that encrypted the file.

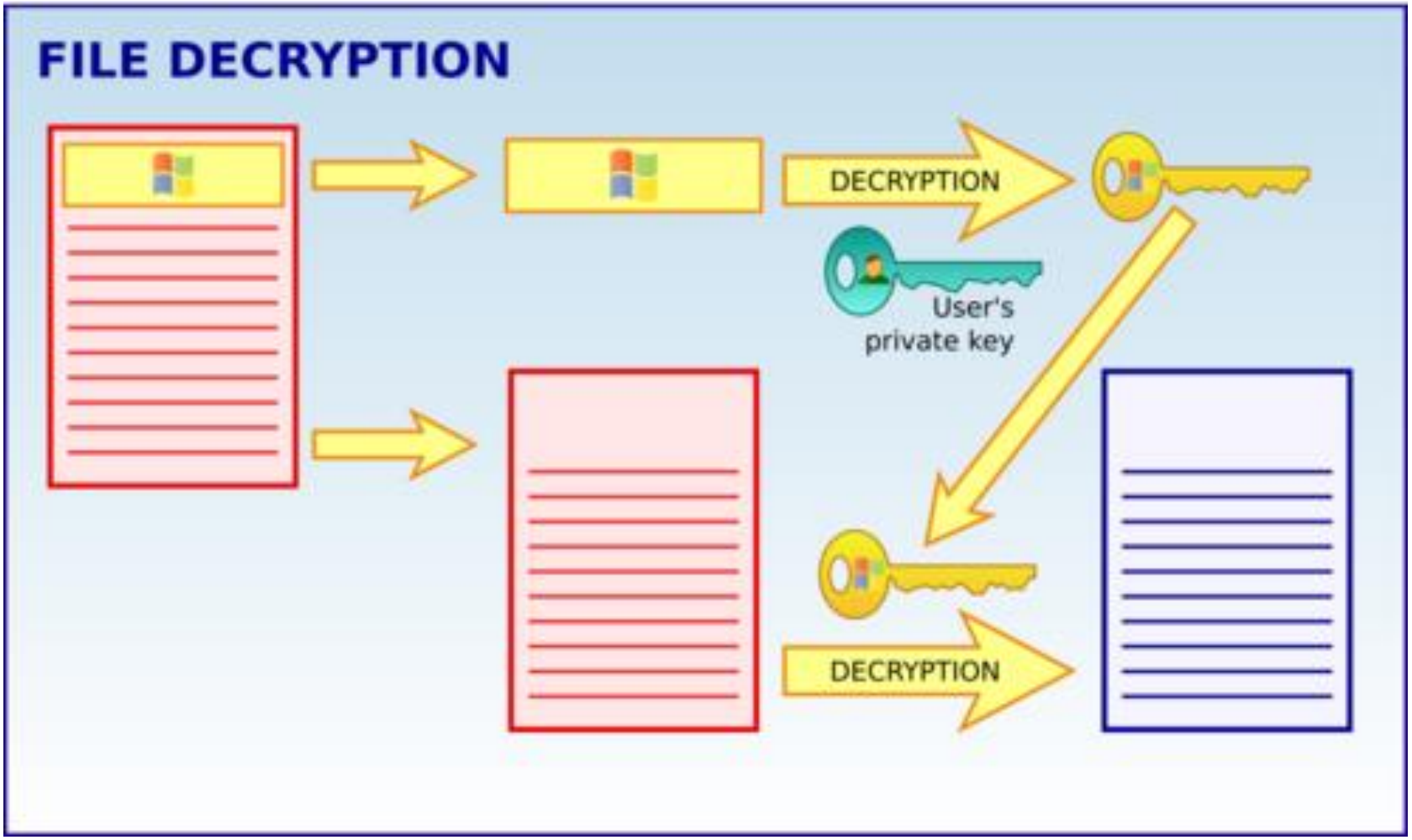
EFS Encryption



EFS Decryption

- The EFS uses the private key that matches the EFS digital certificate (that was used to encrypt the file) to decrypt the symmetric key.
- The resulting symmetric key is then used to decrypt the file.

EFS Decryption

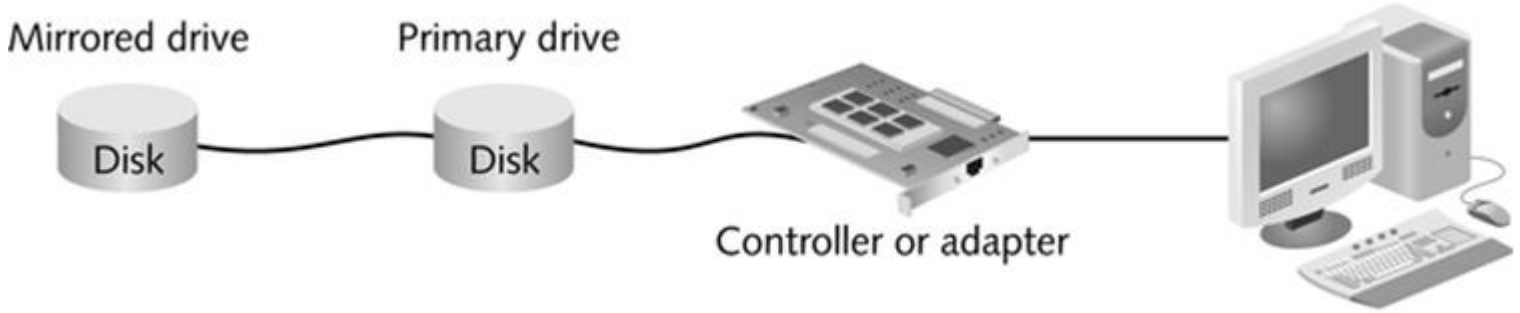


Fault-Tolerance Options

- Disk mirroring
- Disk duplexing
- Redundant array of inexpensive (or independent) disks (RAID)

Disk Mirroring

- A technique in which data is written to two duplicate disks simultaneously.
- This way if one of the disk drives fails, the system can instantly switch to the other disk without any loss of data or service.
- Disk mirroring is used commonly in on-line database systems where it's critical that the data be accessible at all times.

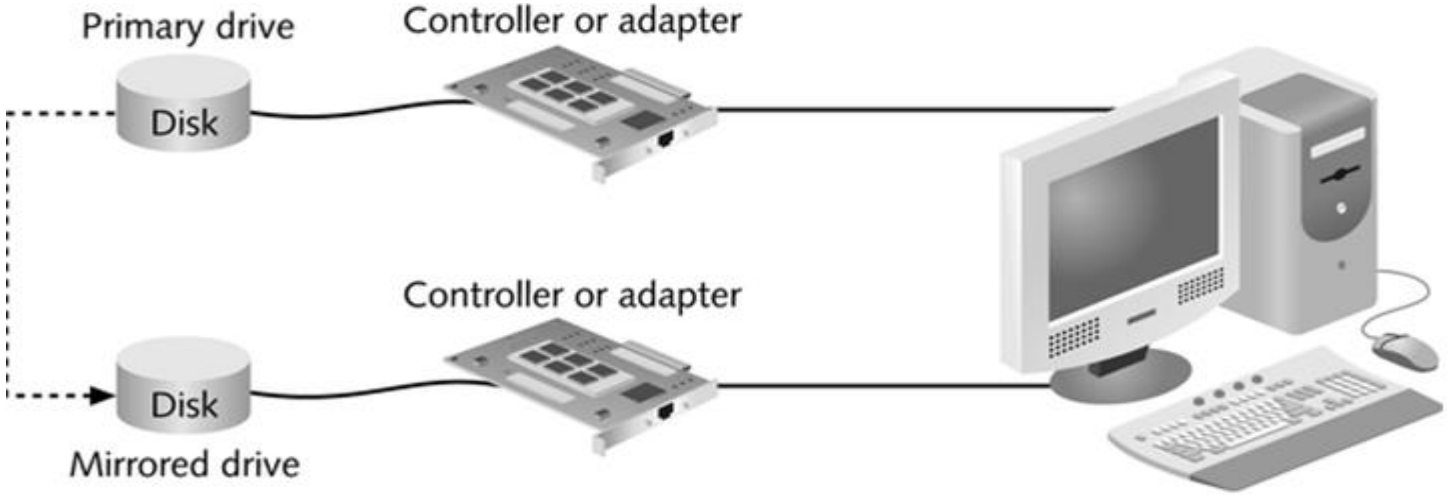


Disk Mirroring …

- Keep two copies of data on two separate disks
- Gives good error recovery
  - if some data is lost, get it from the other source
- Expensive
  - requires twice as many disks
- Write performance can be slow
  - have to write data to two different spots
- Read performance is enhanced
  - can read data from file in parallel

Disk Duplexing

- Disk duplexing is a variation of *disk mirroring* in which each of multiple storage disks has its own controller or adapter.
- Mirroring provides data protection in the case of disk failure, because data is constantly updated to both disks.
- However, since the separate disks rely upon a common controller, access to both copies of data is threatened if the controller fails.
- Disk duplexing overcomes this problem; the use of redundant controllers enables continued data access as long as one of the controllers continues to function.



RAID

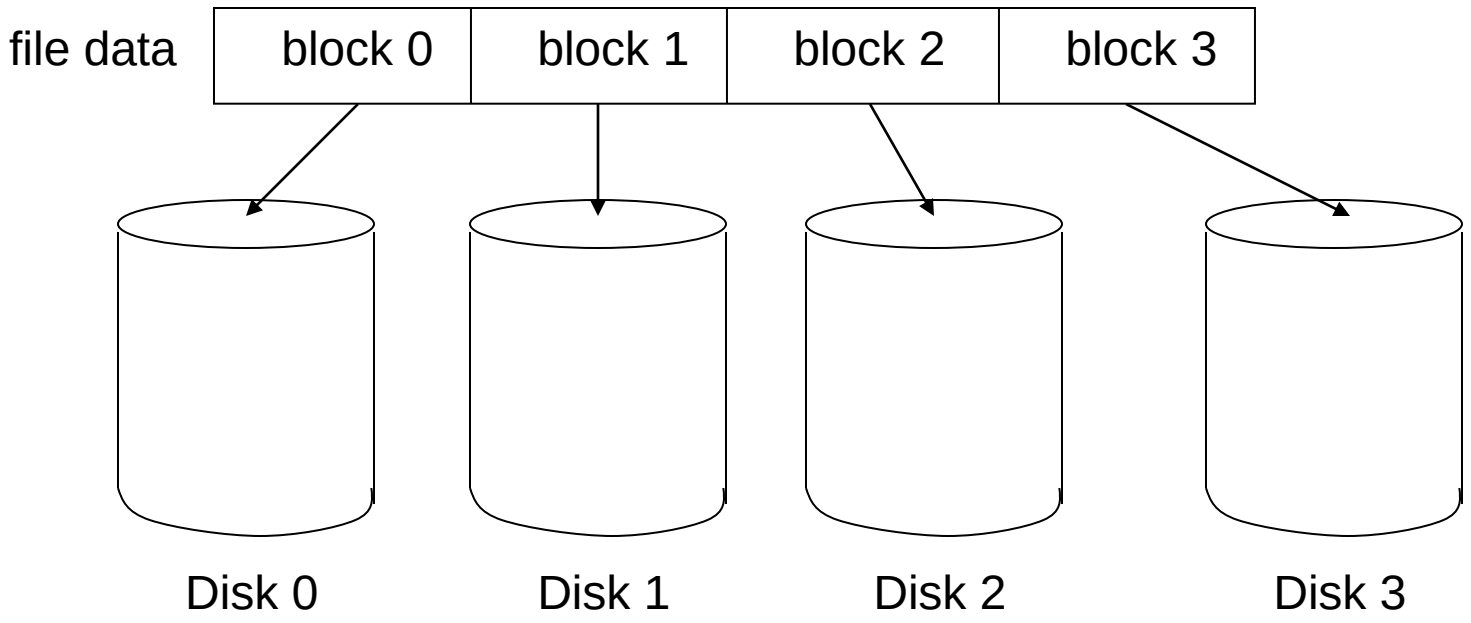
- Set of standards for lengthening disk life and preventing data loss
- Goal: to spread disk activity equally across all volumes
- Basic idea is to connect multiple disks together to provide
  - large storage capacity
  - faster access to reading data
  - redundant data
- Many different levels of RAID systems
  - differing levels of redundancy, error checking, capacity, and cost

RAID …

- It’s a technology that enables greater levels of performance, reliability and/or large volumes when dealing with data.
- How?? By concurrent use of two or more ‘**hard disk drives’.**
- How Exactly?? Mirroring, Stripping (of data) and Error correction techniques combined with multiple disk arrays give you the reliability and performance.

Striping

- Take file data and map it to different disks
- Allows for reading data in parallel



Parity

- Way to do error checking and correction
- Add up all the bits that are 1
  - if even number, set parity bit to 0
  - if odd number, set parity bit to 1
- To actually implement this, do an exclusive OR of all the bits being considered
- Consider the following 2 bytes

byte	parity
10110011	1
01101010	0

- If a single bit is bad, it is possible to correct it

RAID flavors

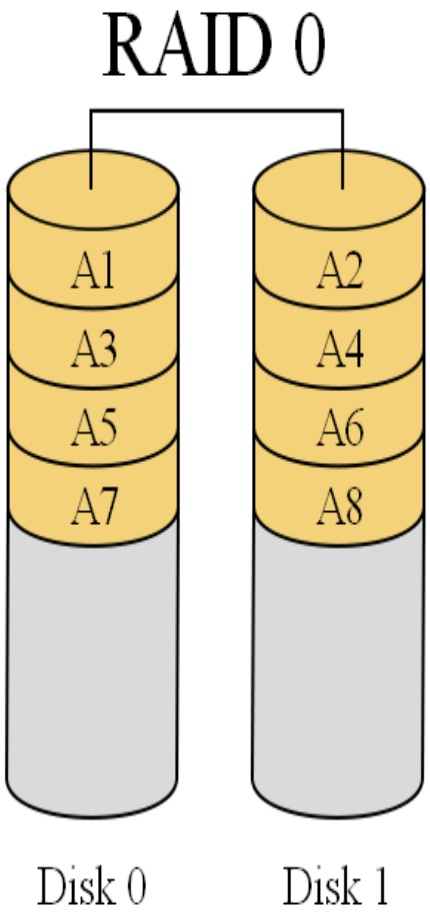
- Commonly used ones:

1. RAID 0
2. RAID 1
3. RAID 5
4. RAID 10

- Other types used…but rarely: RAID 2,3,4,6,50……

RAID 0

a. It splits data among two or more disks.
b. Provides good performance.
c. Lack of data redundancy means there is no fail over support with this configuration.
d. In the diagram to the right, the odd blocks are written to disk 0 and the even blocks to disk 1 such that A1, A2, A3, A4, … would be the order of blocks read if read sequentially from the beginning.
e. Used in read only NFS systems and gaming systems.

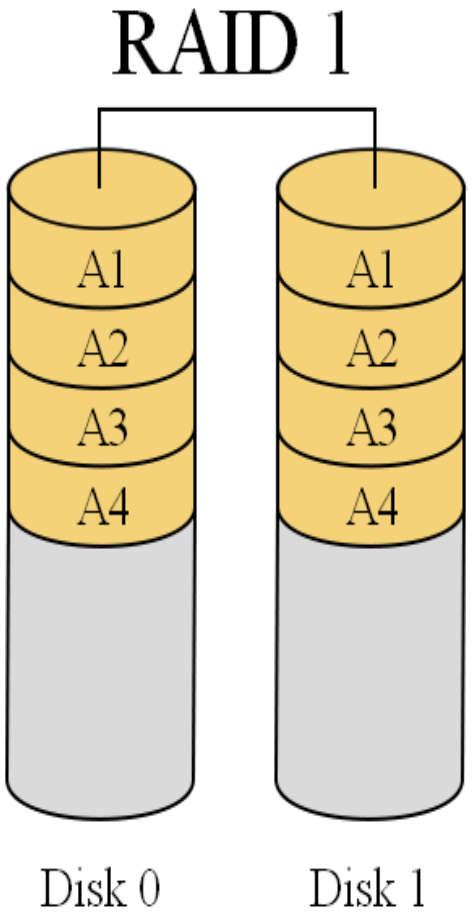


RAID 0 …

- No replication
- ***Advantages:***
  - Simple to implement
  - No overhead
- ***Disadvantage:***
  - If array has *n* disks failure rate is *n* times the failure rate of a single disk

RAID 1

- RAID1 is ‘data mirroring’.
- Two copies of the data are held on two physical disks, and the data is always identical.
- Twice as many disks are required to store the same data when compared to RAID 0.
- Array continues to operate so long as at least one drive is functioning.



RAID 1 …

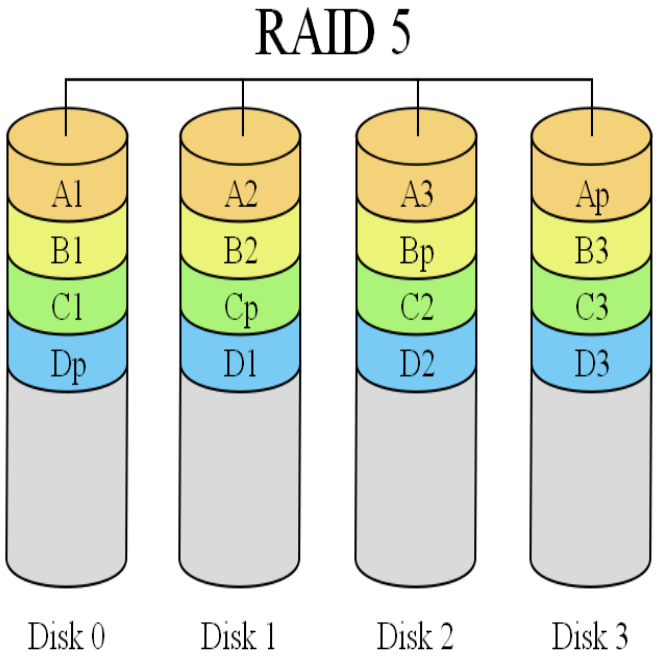
- A complete file is stored on a single disk
- A second disk contains an exact copy of the file
- Provides complete redundancy of data
- Read performance can be improved
  - file data can be read in parallel
- Write performance suffers
  - must write the data out twice
- Most expensive RAID implementation
  - requires twice as much storage space

RAID 1 …

- Mirroring
  - Two copies of each disk block
- Advantages:
  - Simple to implement
  - Fault-tolerant
- Disadvantage:
  - Requires twice the disk capacity of normal file systems

RAID 5

- RAID 5 is an ideal combination of good performance, good fault tolerance and high capacity and storage efficiency.
- An arrangement of parity and CRC to help rebuilding drive data in case of disk failures.
- “Distributed Parity” is the key word here.

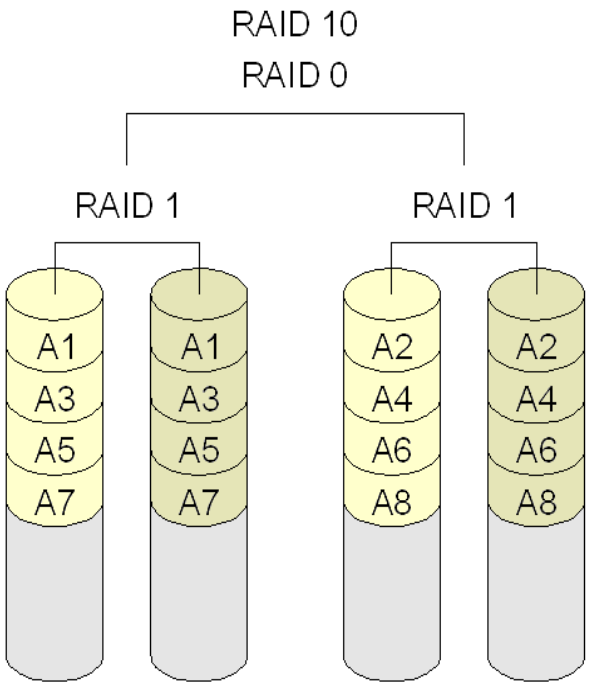


RAID 5 …

- Level-5 stripes file data and check data over all the disks
  - no longer a single check disk
  - no more write bottleneck
- Drastically improves the performance of multiple writes
  - they can now be done in parallel
- Slightly improves reads
  - one more disk to use for reading

RAID 10

a. Combines RAID 1 and RAID 0.
b. Which means having the pleasure of both - good performance and good failover handling.
c. Also called ‘Nested RAID’.



RAID 10 …

- Combine Level-0 and Level-1
- Stripe a files data across multiple disks
  - gives great read/write performance
- Mirror each strip onto a second disk
  - gives the best redundancy
- The most high performance system
- The most expensive system

Software RAID versus Hardware RAID

- Software RAID
  - Implements fault tolerance through computer's operating system
- Hardware RAID
  - Implemented through RAID hardware (eg, adapter)
  - Independent of operating system
  - More expensive than software RAID

Advantages of Hardware RAID

- Faster read and write response
- Ability to place boot and system files on different RAID levels
- Ability to “hot swap” a failed disk with one that works or is new
- More setup options to retrieve damaged data and to combine different RAID levels within one array of disks

Comparing RAID Levels

	RAID 0	RAID 1	RAID 5	RAID 10
Read	High	2X	High	High
Write	High	1X	Medium	High
Fault tolerance	No	Yes	Yes	Yes
Disk utilization	High	Low	High	Low
Key problems	Data lost when any disk fails	Use double the disk space	Lower throughput with disk failure	Very expensive, not scalable
Key advantages	High I/O performance	Very high I/O performance	A good overall balance	High reliability with good performance

Which RAID Level to Use?

- Data and Index Files
  - RAID 5 is best suited for read intensive apps or if the RAID controller cache is effective enough.
  - RAID 10 is best suited for write intensive apps.
- Log File
  - RAID 1 is appropriate
    - Fault tolerance with high write throughput. Writes are synchronous and sequential. No benefits in striping.
- Temporary Files
  - RAID 0 is appropriate.
    - No fault tolerance. High throughput.

What RAID Provides?

- Fault tolerance
  - It does not prevent disk drive failures
  - It enables real-time data recovery
- High I/O performance
- Mass data capacity
- Configuration flexibility
- Lower protected storage costs
- Easy maintenance