Confidential data storage

Problems

The protections provided by a traditional filesystem are limited

Physical Protections

- File system is limited to a physical device

Logical Protections

- Access control to files, controlled by the operating system
- Using ACLs and other confinement mechanisms

Problems

There is a relevant number of situations where standard protections are irrelevant

When there is direct and physical access to devices

- Access to host devices (laptops, smartphones, servers)
- Access to external storage devices
  - Tapes, CDs, DVDs, SSDs, NAS

Access through the system with the correct rights

- Non-ethical access by system administrators
- With impersonation attacks

Problems

There is a prevalence of distributed storage

It imposes trusting multiple administrators, sometimes unknown

Authentication is made remotely

- Sometimes it is not clear what is the security level of said methods
- Storage Provider may have unknown integrations
- Interaction models are complex, through external networks
- Multiple entities involved

Information is transmitted through communication channels

- May violate confidentiality, integrity and create privacy issues

Solution: Encrypt data

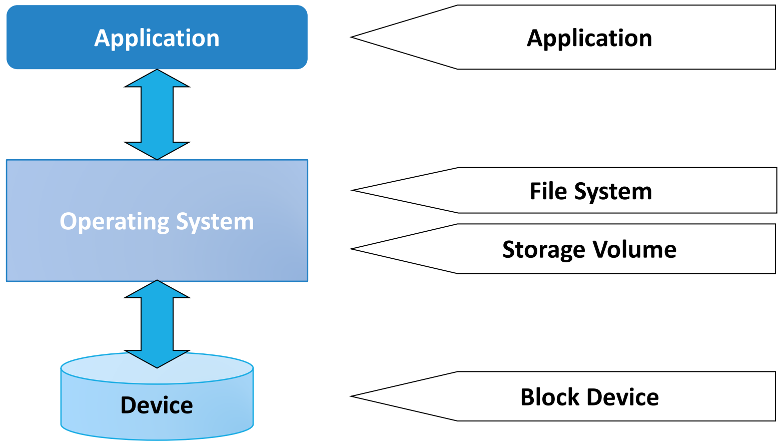
Encryption/Decryption of file contents

- Enable secure transfer over insecure networks
- Enable secure storage in insecure locations
  - Managed by external entities, or in shared storages

Problems of encryption

- Access to information
  - Users may lose the keys
    - Key loss = data loss
    - Key storage may reduce overall security
- File sharing
  - Sharing data implies sharing keys
- May interfere with standard management and recovery tasks
  - Content analysis, deduplication, indexing

Approaches



Encryption in Applications

Information is transformed by each application

- Little or no integration with other applications
- Usually, it is clear what is secure or not
  - Specific files with known file extensions

Present vulnerability windows

- Data must be decrypted to other files before being accessed

Information may be processed by different algorithms/keys

- Adapted to a specific operating system or the security level
- May complicate the data recovery processes

May difficult sharing data inside the encrypted package

- May imply extract data which is stored in a clear format

Examples:

- PGP, AxCrypt, TrueCrypt, Veracrypt, etc.
- Also: RAR, ZIP, 7Zip, LZMA…

Encryption in the File Systems

Information is transformed when is sent from memory to the filesystem

- May be broad, from the entire filesystem into the global memory cache
  - No protection in shared servers as data is available to all applications
  - Security mechanism is harder to implement in distributed environments
    - Coordination of ACLs
- May be specific to the cache of a specific process
  - Protection in the case of shared servers as data access is context-bound
  - Client API decrypts data

Examples

- EncFS, EXT4, NTFS, CFS

Encryption at the volume level

Information is transformed by the volume driver

- Transparent to applications and almost transparent to the OS
  - Requires support through a specific driver
- The entire volume will be made available (partition)

Policies defined through applications or the controller

- Agnostic to the actual filesystem on top
  - Protects everything, including metadata
- But it doesn't differentiate between individual users

Unable to solve problems related with distributed systems, but solves those related with mobile devices

- Distributed systems expose the filesystem after decryption
- Mobile devices: lost of stolen devices will keep data secure

Examples:

- PGPDisk, LUKS, BitLocker, Filevault

Encryption at the Device Level

Block Device applies security policy internally

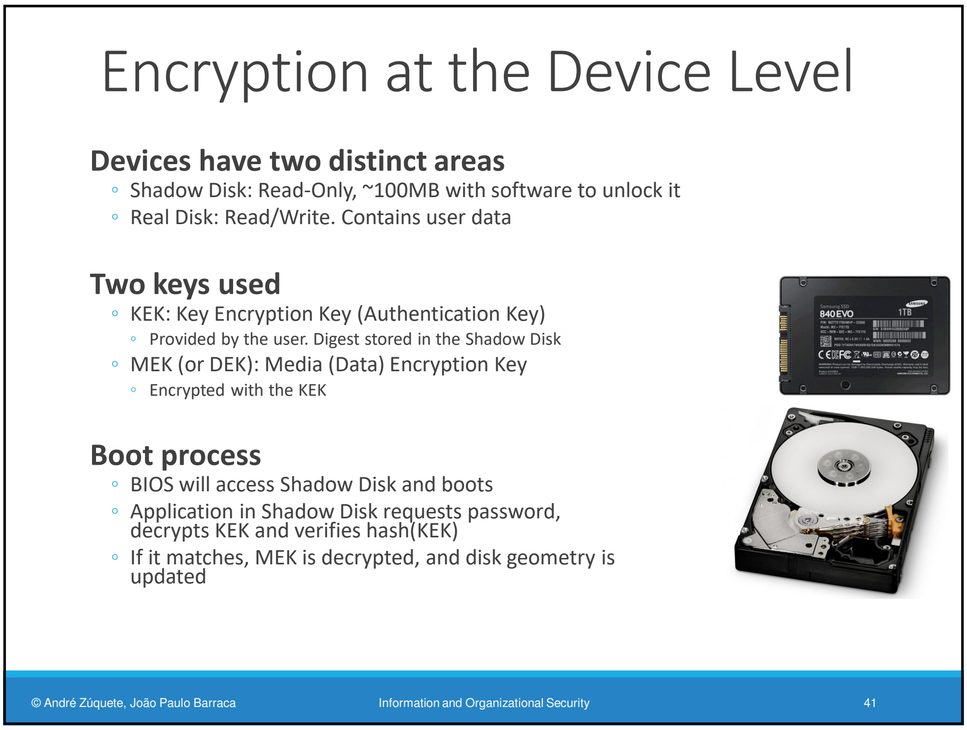
- At boot, the device must be unlocked
  - After the correct credentials are provided
- Encryption is implemented at the hardware/firmware

Advantages

- No performance loss
- Data access is not trivial as keys are internal
- May be coordinated with applications (e.g., USB devices)

Disadvantages

- After the device is unlocked, all data is made available
- Security is limited by the algorithms present
- The possible existence of backdoors is difficult to find and correct



Encryption at the Device Level

Devices have two distinct areas

- Shadow Disk: Read-Only, ~100MB with software to unlock it
- Real Disk: Read/Write. Contains user data

Two keys used

- KEK: Key Encryption Key (Authentication Key)
  - Provided by the user. Digest stored in the Shadow Disk
- MEK (or DEK): Media (Data) Encryption Key
  - Encrypted with the KEK

Boot process

- BIOS will access Shadow Disk and boots
- Application in Shadow Disk requests password, decrypts KEK and verifies hash(KEK)
- If it matches, MEK is decrypted, and disk geometry is updated