PV080 – Disk Encryption

Additional material for Full Disk Encryption use

Milan Brož xbroz@fi.muni.cz
Faculty of Informatics, Masaryk University

CROCS
Centre for Research on Cryptography and Security

Data storage encryption

- external and internal disk drives, laptops, smartphones, appliances, cloud storage

- provide
  - **confidentiality**
  - **for data-at-rest** (offline data protection)

- not for protection for data-in-transit
- no standalone data integrity protection
  can be sub-component in system integrity protection

Full Disk Encryption (FDE)

also known as ~ volume encryption, sector-based encryption, transparent on-the-fly encryption

- symmetric encryption of disk sectors (blocks)
  - transparent for filesystems / applications
  - key management (how to unlock device)

- FDE currently (~2021) massively prevails in comparison with filesystem-based encryption

Full Disk Encryption (FDE)

- data confidentiality
  only authorized user can access plaintext data
  often law requirement or company policy
- easy data disposal (destruction)
  encryption key destruction is enough
- per-application storage (isolation)
  plaintext data not available from other applications

- Different threat models examples
  - *stolen laptop or smartphone*
  - *disk in repair, second-hand hw (warrant claim, hw decomission)*
  - *mandatory data destruction (office printers, IoT, …)*
  - *cloud applications, data leaks, isolation*

FDE algoritms (and limits)

- symmetric encryption of sectors
  - *sectors encrypted independently (transparency)*
  - *almost all systems default to AES in XTS mode*
    exceptions: low-end systems
  - *XTS as trade-off: granularity of change propagation vs performance*

- key management
  - on-disk encrypted metadata
    Key Encryption Key (KEK) > Media Encryption Key (MEK)
  - attack cost, key derivation (PBKDF2 / Argon2)
  - TPM or HSM (only to store volume key)

How is FDE implemented

- in hardware – Self-Encrypted Disks (SED) or chipset-based encryption (USB disk enclosure)
  - *proprietary firmware, OPAL standard*
  - *a lot of problems in the recent past*
  - *use only if you trust hardware vendor*
  - *cheap external USB enclosures are very unreliable*

- software-based disk encryption
  with some hw acceleration (AES-NI, etc)
- *part of OS / distribution, software updates*
- *most of solutions today (demo)*
  BitLocker (Windows), dm-crypt (Linux, Android) + LUKS, VeraCrypt (multiplatform), FileVault (MacOS), ...

Example: Microsoft Windows 10 BitLocker

- (system) drive / disk encryption
- BitLocker to Go – for removable drives
- SED wrapper: BitLocker eDrive – better avoid it

- metadata embedded in NTFS / exFAT
  but functionally it is separate layer (see demo access in Linux)

- Group policy (registry)
- GUI + manage-bde comandline tool
- Key management
  password, TPM, PIN, external key, recovery password

Example: LUKS in Fedora Linux 33

- dm-crypt – kernel encryption driver
  - also used in Android FDE
  - uses kernel Crypto API drivers

- LUKS (Linux Unified Key Setup)
  - key management, multiple keyslots

- Integration in almost all distributions
  - *system encryption (in installer)*
  - *GRUB bootloader*
  - *UDisks GUI + cryptsetup commandline tool*
  - *also suports unlocking of foreign formats (BitLocker, VeraCrypt)*

Conclusion

- use FDE for all mobile and removable devices
  - *performance is no longer problem*
  - *your data is more valuable than cost of hardware*
  - *device lost or theft happens very often, be prepared*
- common password rules (use strong password)
- do not randomly modify encryption defaults
- data backups must include **encryption metadata**
- data backups should be encrypted too
- storage encryption is not a replacement for in-transit encryption (possible replay attacks)

More details in PV204 Security technologies course.