Hard Disk Encryption

Software Security

Steffen Helke

Chair of Software Engineering

7th January 2019

Brandenburgische Technische Universität Cottbus - Senftenberg

Objectives of today's lecture

- ➜ Getting to know the challenges of Software Engineering for implementing full disc encryption systems
- ➜ Understanding operating modes focussing on disk encryption
- ➜ Being able to explain how CTS, XEX and XTS work

Encryption of File Systems

Full Disk Encryption (FDE)

Motivation

- Increasing number of mobile computers
- Stealing these devices cannot be completely prevented
- Hardware access allows to bypass the rights management of an operating system

➜ Private and/or internal company data are accessible to unauthorized persons

What can be encrypted?

1. Full hard disk
2. Single partitions or home directories of users
3. Additionally boot sector

➜ Implementations for hardware and software are available

Full Disk Encryption

Challenges of Software Engineering

- User interface should require a minimum of user intervention, i.e. we need a high-level *security transparency* to achieve user acceptance of the full encryption technology
- High-quality *key management* with effective key recovery mechanisms to recover lost keys
- Support of a *group concept* in multi-user environments
- Minimization of *performance* losses inevitably caused by encryption

Full Disk Encryption

Which encryption to use?

- BSI recommends AES-256 in XTS mode for particularly high security requirements
- But weaker encryption systems can also be used

Weaknesses

- Hard disk encryption does not increase security during system operation (e.g. server connected to network)
- Memory can be read out via direct memory access (DMA)
- Virtual memory is often not encrypted

Disk Encryption Operation Modes

How can random access implemented?

Problem

- Random access to encrypted data needs to be guaranteed

➜ Blocks must be encrypted independently as far as possible and still securely encrypted

Approaches

1. CBC (Cipher Block Chaining)
2. LRW (Liskov, Rivest, Wagner)
3. XTS (Extension of LRW)

➜ Newer implementations mainly use XTS!

Which operation modes are suitable for hard disk encryption?

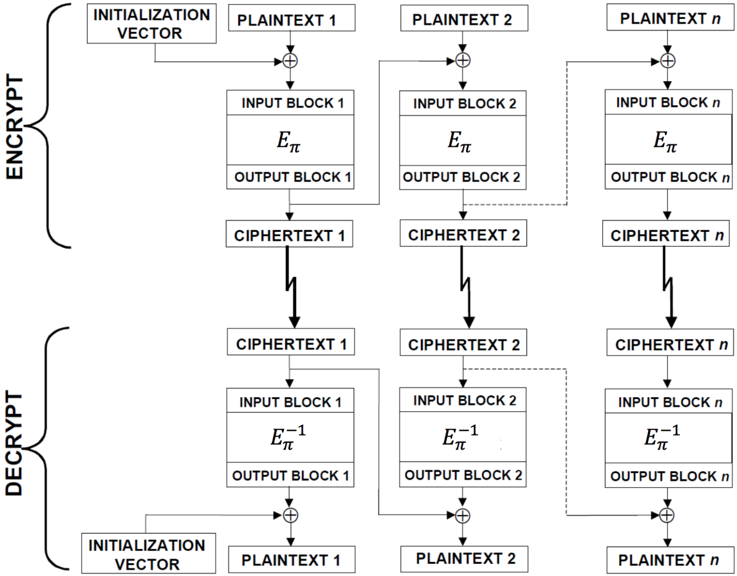
Disk Encryption Operation Modes

CBC

- Ciphertext block i is used as input for encrypting the direct successor block $i+1$ which results in an encryption chain
- Method is unsuitable for encrypting a complete hard disk partition because random access is not possible
- Hence sector by sector encryption is implemented, the initialization vectors are calculated indeterministically (hashing on the key, number of sectors and/or timestamp)

LRW

- In contrast to CBC, isolated block processing
- Random key generation for each block
- Additional 128-bit key required for administration
- Better protection of the management key by **XTS**



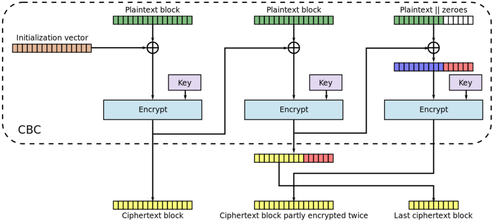
Example: Operation mode with CBC – Cipher Block Chaining

Operation Mode

– XEX-based Tweaked-codebook mode with CTS (XTS) –

CTS as an important building block for XTS

- XTS is short for . . . *XEX-based Tweaked-codebook mode with CTS*
- CTS is short for . . . *CipherText Stealing*
  - ➜ Padding to fill the last block is avoided by this method
  - ➜ Special processing of the last two blocks: The piece required for the last encryption is *stolen* from the second-last block



Example CTS for CBC, Source: https://en.wikipedia.org/wiki/Ciphertext_stealing

Benefit of CTS: Length of ciphertext and plaintext are the same!

Basic Principle of XEX

➜ XEX is short for ... *Xor-Encrypt-Xor*

- Method was developed by Phillip Rogaway in 2004
- Objective: Fast encryption of a sequence of blocks without using initialization vectors and encryption chains
- Ciphertext C is calculated according to the following rule

  $X = E_k(I) \otimes \alpha^j$
  $C = E_k(P \oplus X) \oplus X$, where [1]

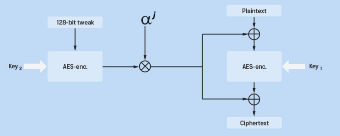
  - P is plaintext
  - I is the address of the sector to be encrypted
  - α is primitive polynomial made of $GF(2^{128})$
  - j is a block index within the given sector

XEX encrypts blocks separately, but in contrast to ECB identical plaintexts are mapped to different ciphertexts, because the *tweak* X is mutable

[1] Operation $\otimes$ describes the multiplication for polynomials modulo $x^{128} + x^7 + x^2 + x + 1$, which can be efficiently calculated for simple α^j, the operation $\oplus$ represents the XOR operation

XTS Encryption for a Sector

How to encrypt using XTS[1]?

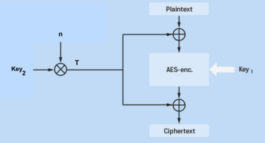


1. Construct a 128-bit *tweak* based on sector properties
   ➜ Result is a constant *master tweak*
2. Calculate AES encryption of the *master tweak* using key_2
3. Decompose the data to be encrypted into 128-bit blocks, with ascending index j, starting at 0
4. Multiply the primitive polynomial α^j with the encrypted *tweak value* in $GF(2^{128})$ which can be efficiently implemented using a left shift by j places ➜ Result is a mutable *subtweak*
5. Add the plaintext of block j to the *subtweak* using XOR, then calculate an AES encryption for the intermediate result using the key_1 and finally add again the subtweak to the result using XOR

[1] Note, XEX mode uses a single key for two different purposes, whereas XTS mode uses two independent keys

Differences between XTS and LRW

- LRW is a generic tweaked cipher design, proposed as the basis for a variety of tweaked modes and based on suitable hash functions



Weak instantiation of the LRW design

- XTS is in principle also an instantiation of the generic LRW design
- Note, also weaker instantiations exist, e.g. the draft SISWG proposal for tweakable narrow-block encryption (LRW-AES)[1]
- There the *tweak* T is just calculated from the polynomial multiplication of key_2 and the logical index n of the data block to be encrypted, the rest of the encryption works quite similar to XTS

➜ Note that this specific LRW-AES instantiation in particular has some security concerns, so XTS mode is now recommended for use

[1] http://www.siswg.net/docs/LRW-AES-10-19-2004.pdf

Software for Full Disk Encryption

Name	CBC w/ predictable IVs	CBC w/ secret IVs	CBC w/ random per-sector keys	LRW	XTS
Aloaha Crypt Disk	No	No	No	Yes	Yes
ArchiCrypt Live	No	No	No	Legacy support[125]	Yes
BestCrypt	No	Yes	No	Yes[126]	Yes[127]
BitArmor DataControl	No	Yes	Plumb-IV	No	No
BitLocker	No[128]	Yes[128]	No	No	Yes, Windows 10 10547+
Bloombase Keyparc	?	Yes	?	?	?
CGD	No	Yes[129]	No	No	No
CenterTools DriveLock	?	?	?	?	?
Check Point Full Disk Encryption	No	No	Yes	Yes	Yes
CipherShed	Legacy support[130]	No	No	Legacy support[131]	Yes[132]
CrossCrypt	Yes	No	No	No	No
CryFS	No	No	Yes	No	No
CryptArchiver	?	?	?	?	?
Cryhod	No	Yes	No	No	No
Cryptoloop	Yes	No	No	No	No
DiskCryptor	No	No	No	No	Yes
Dm-crypt	Yes	Yes	No	Yes, using *-lrw-benbi[133]	Yes, using *-xts-plain
DriveCrypt	?	?	?	?	?
DriveSentry GoAnywhere 2	?	?	?	?	?
E4M	?	?	?	No	No
e-Capsule Private Safe	?	?	?	?	?
eCryptfs	No	Yes	?	No	No
EgoSecure HDD Encryption	No	Yes	No	No	No
FileVault	Yes[84]	No	No	No	No
FileVault 2	No	No	No	No	Yes[134]
FREE CompuSec	Yes	No	No	No	No
FreeOTFE	Yes	Yes	No	Yes	Yes
GBDE	No	No	Yes[89]	No	No
GELI	No	Yes[135]	No	No	Yes
Loop-AES	single-key, multi-key-v2 modes[91]	multi-key-v3 mode[91]	No	No	No
McAfee Drive Encryption (SafeBoot)	No	Yes	No	No	No
n-Crypt Pro	?	?	No	No	No
PGPDisk	?	?	?	?	?
Private Disk	No	Yes	No	No	No
ProxyCrypt	No	No	No	No	Yes
R-Crypto	?	?	?	?	?
SafeGuard Easy	?	?	?	?	?
SafeGuard Enterprise	?	?	?	?	?
SafeGuard PrivateDisk	?	?	?	?	?
SafeHouse Professional	Yes	No	No	No	No
Scramdisk	No	Yes	No	No	No
Scramdisk 4 Linux	No	Yes[136]	No	Yes[137]	Yes[138]
SecuBox	Yes	No	No	No	No
SecureDoc	?	?	?	?	?
Sentry 2020	?	?	?	?	?
Softraid / RAID C	?	?	?	?	Yes[139]
Svnd / Vnconfig	?	?	?	?	?
Symantec Endpoint Encryption	No	No	Yes	No	No
TrueCrypt	Legacy support[130]	No	No	Legacy support[131]	Yes[140]
USBCrypt	No	Yes	No	No	Yes
VeraCrypt	No	No	No	No	Yes
CyberSafe Top Secret	No	No	No	No	Yes

Source: `https://en.wikipedia.org/wiki/Comparison_of_disk_encryption_software`

References

1 Moses Liskov, Ronald L. Rivest, David Wagner: Tweakable block ciphers, CRYPTO 2002, LNCS 2442, Springer, 2002.

2 Phillip Rogaway: Efficient Instantiations of Tweakable Blockciphers and Refinements to Modes OCB and PMAC, Asiacrypt 2004. LNCS 3329. Springer, 2004.

3 Kazuhiko Minematsu: Improved Security Analysis of XEX and LRW Modes, SAC 2006, LNCS 4356, Springer, 2007.

4 Moses Liskov, Ronald L. Rivest, David Wagner: Tweakable Block Ciphers, Journal of Cryptology, Vol 24, Springer, 2010.

5 Draft Proposal for Tweakable Narrow-block Encryption (2004), `http://www.siswg.net/docs/LRW-AES-10-19-2004.pdf`

6 `http://en.wikipedia.org/wiki/Comparison_of_disk_encryption_software`, Last access 20.12.2017