Encrypting removable storage devices

Removable device encryption – R/W compatible with Linux and Windows

for TF-CSIRT, 22nd Meeting – Oporto, PT
September 21, 2007
Background and (my) Motivation

• Some private discussion regarding management of data storage, including backup
• Some accidents
  – Laptop gone missing
  – Memory sticks being forgotten...
• Logistics more interesting than technology
  – experience?
  – discussion?
Environment: University vs. Enterprise ;-)

• There are commercial products available, but
  – Management domain has to include the end users
  – Vendor lock-in (backup!)
  – Platform lock-in (+ version management!)
  – Is the vendor trustworthy and
  – going to be around for a „while“?
  – Trust relationship topology: Tree oder Net?
  – ➔ can be cumbersome..
My Shopping-List…

• As simple as possible for every-day use!
• „Transparent“ to application software
  – has to work with „everything“ 😊
• Compatible with Linux (+Unix) and Windows
  – Windows/XP SP2
  – Fedora Core 6 (+ *BSD, OS X if possible)
• I don‘t trust any Certification/Trust Authority
  – unless I run it myself, or a good friend 😊 does,
• plus …
(continued)

My Shopping-List...

• ...

• I hardly trust any „closed source“ Product,
  – even less if from the United Stans of America

• ➔ Are there Open Source/GPL solutions?

• My 1. assumption was: no, that‘s fairy land,
  – and I was wrong!

• At least 2 Open Source/GPL solutions exist!
  – TrueCrypt, FreeOTFE
Tests and Technology

• Caution: still a „Work in Progress“ for me
• Different approaches:
• Encrypt individual records or files, e.g. GnuPG
  – still: inevitably, cleartext is bound to sit around anyway!
  – still: management of Key/s and Pass-Phrase/s
• Virtual Volume within a container
  – „loopback driver“ concept in Linux
  – quite OK 😊
  – still: management of Key/s and Pass-Phrase/s
Tests and Technology

- Encrypting a whole partition
  - minimises the problem of „littering the environment“ with clear text
  - minimises dependencies and complexity when supporting different filesystem formats, compared to Loop-Back and Virtual Disk
  - still: management of Key/s and Pass-Phrase/s
  - still: authorized access by colleagues?
  - still: loss of Key and/or Pass-Phrase
Tests and Technology

- Encrypting a whole partition, trying to minimise the problems listed, in Linux:
  - Device Mapping Tables
    - Abstraction on Device- / Partition-Plane!
  - LUKS
    - Linux Unified Key Storage
    - almost completely automatic (/etc/fstab) [bugs]
  - LUKS supports multiple Access-Identities
    - still tests to do! (key escrow, emergency access,...)
Tests and Technology

• What‘s the situation for the Windows platform?
  – TrueCrypt is said to support many (all?) of these things
  – FreeOTFE does support all that stuff!

• FreeOTFE Tests:
  – Initially V1.6, V2.0 has become available

• FreeOTFE supports 2 different modes:
  – „Installed“ (Admin) and „Portable“ (User-Mode!)
• My approach:
  – 1. Tests with Linux
    • either using individual commands für Device-Mapping and Friends (according to the „How-To“), or
    • use a version of cryptsetup with LUKS-Support
    • modify /etc/fstab
    • use mkfs.fvat to create a filesystem that is also supported by Windows
    • use mount and supply the Pass-Phrase
Tests and Technology

• ... just to keep in mind:
  – Cryptographic-Mechanisms have to be compatible between the platforms
  – Filesystem formats have to be supported on both platforms
    • ext2 or ext3 is not available for Windows by default (without installing additional software)
    • NTFS is not necessarily available for Linux
  – VFAT, UDF or ISO9660 seem to be the „best“ choices for many applications
Tests and Technology

• My Approach:
  – 2. Tests with Windows/XP
    • install/use FreeOTFE, in Admin- or in User-Mode?
      – For the moment I am (still) using User-Mode
    • Activate file FreeOTFE.exe, ack the warning box,
      – In GUI under „Tools“ select „portable mode“
      – select ➔ Driver-Installation
      – mount the Partition, supply Pass-Phrase and...
    – 😊
Advantages, Problems, Use

• **Installed oder Portable Mode?**
  - during „Installation“ it is possible to deselect unused modules
    • skip unused (weak) algorithms
    • select one from alternative implementations
    • may then lack support for someone else´s Media
    • does leave tracks in the system (being paranoid 😐 ?)
  - in „Portable Mode“
    • flexible, a bit more cumbersome, few traces of use
    • unused mechanisms do consume resources
Advantages, Problems, Use

• **Application Scenarios**
  – Store valuable informationen „proactively“ in encrypted partitions (collaboration within group)
  – Presentations with sensitive content on Sticks,... can be used immediately and remain encrypted
    • avoid giving away an unencrypted copy?
    • Driver-Load in Portable Mode allowed?
  – Move browser environment, eMail folders, certificate- und key-stores to removable medium
    • platform-independent, protection against theft, ...
Discussion

- Is there any feedback?
- That’s the technology, what do the logistics look like?
- Is it worth the effort, to worry about compatibility, when data is (most likely) used on a single platform?
- How stable are the implementations?
- To investigate: additional platforms?
Useful Documentation

• „CryptoPartitionHowTo“ (in German)
  – Wiki @ systemausfall.org
• „Verschlüsselte Festplatten“
  – @ fedorawiki.de
• „Disk cryptography with dm-crypt“
  – @ www.gentoo.org
• „dm-crypt: a device-mapper crypto target“
  – @ www.saout.de
• Google is your friend 😊
Useful Commands

• Prepare media
  - `fdisk` or use `<your preferred tool>`...

• Crypto Stuff Setup
  - `cryptsetup luksFormat /dev/$DEVICE`
  - `cryptsetup luksOpen /dev/$DEVICE $CRYPT`
  - `mkfs -t $XXX /dev/mapper/$CRYPT`
    • `xxx=vfat|udf|ext2|ext3|iso9660|...`
  - `mount /dev/mapper/$CRYPT $MOUNTPNT`

• Key Management
  - `cryptsetup luksAddKey /dev/$DEVICE`
  - `cryptsetup luksDelKey /dev/$DEVICE`
Useful Commands

• Remove device **cleanly**
  - `umount $MOUNTPNT`
  - `cryptsetup luksClose $CRYPT`

• Remove all device mappings
  - `dmsetup remove_all`

• Select algorithms and keys
  - `cryptsetup parameters for lxsFormat, e.g.
    - `--c aes-cbc-essiv:sha256`
    - `--s 256`
Contact-Information

Team-Website: https://cert.ACO.net/
 eMail: cert@aco.net

Wilfried Wöber
Universität Wien, ZID - ACOnet
Universitätsstrasse 7
A-1010 Wien

 eMail: woeber@cc.univie.ac.at

T: +43 1 4277 14033
M: +43 664 8175166
F: +43 1 4277 9 140
K: 0xF0ACB369 (GnuPG: --load idea)