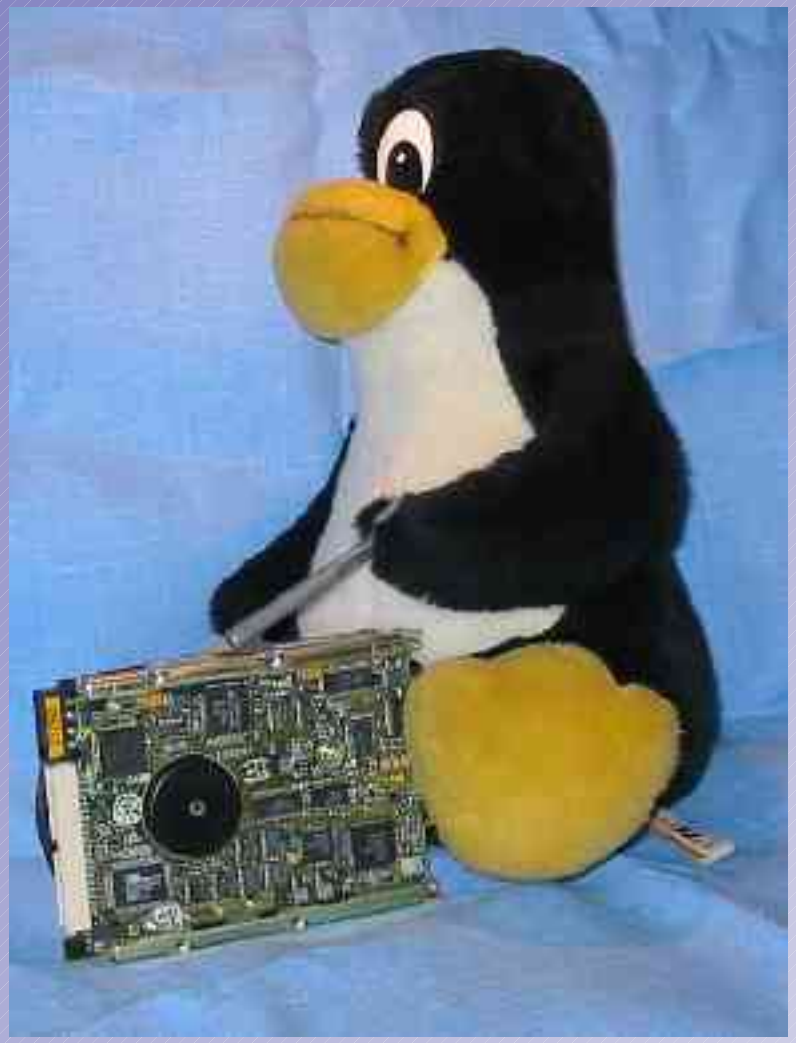
Plattenschlüssel



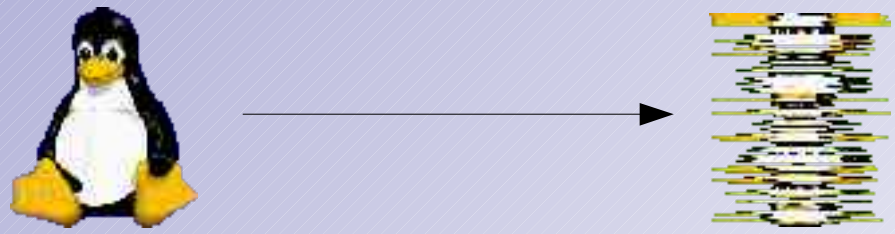
Crypto-Dateisysteme auf Linux

Grundlagen

- Cipher
- Hashes
- Schlüssellängen
- symmetrische vs. asymmetrische Verfahren
- Zufall
- angewandte Paranoia

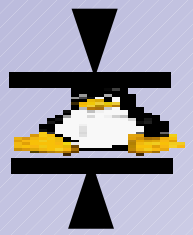
Cipher

- komplexe mathematische Funktionen
- transformieren Input ➔ Output
- parametrisierbar: Schlüssel
- ohne Wissen des Keys (fast) nicht erratbar



Hash/Message Digest

- transformiert beliebig langen “Text” in Block fester Länge
- (fast) nicht umkehrbar



Asymmetrische Verfahren

- benutzen unterschiedliche Schlüssel zum ver- und ent-schlüsseln
- geheimer Schlüssel nur sehr schwer aus öffentlichem Schlüssel zu berechnen

Zufall

- “guter” Zufall ist nötig:
  - Block n+1 darf nicht aus Block n zu folgern sein
  - Zufallsstrom darf nicht aus bekannten Parametern ableitbar sein
- Linux: /dev/random, /dev/hw_random
- Nicht: /dev/urandom – Sequenz ist vorhersagbar!

Schlüssellänge

- ...oder: mein Pinguin ist grösser als Deiner!
- sollte nicht zu klein sein (sonst knackbar)
- sollte nicht zu gross sein (Rechenzeit)

- 128bit für symmetrisch
- 1024-2048bit für asymmetrisch
- 160bit für Hashes

angewandte Paranoia

- trust no one!
  - keine sensitiven Daten herumliegen lassen
  - keine Software einsetzen, die niemand sehen darf
- be conservative!
  - bewährte Konzepte einsetzen
  - ausgetestete Programme benutzen
- onion configuration
  - mehrere Schichten aufbauen, statt nur einer zu vertrauen

Schlüssel

Key erzeugen

- altes Cryptoloop:
  - dd if=/dev/random count=1 bs=128 | od -x
- dm-crypt:
  - dd... | hashalot -x rmd160
  - cryptsetup
- Vorsicht!
  - Key nicht auf der Platte ablegen!
    - Floppy oder USB-Stick (+Backup)
  - kein Suspend-to-disk!
  - kein Swap od. verschlüsselter Swap

Umgang mit Schlüsseln

- Schüssel auf anderem Medium
  - auf USB-Stick
  - per ssh von anderem System mounten

```
cat ~/.key/fs1.vol1-key |  ssh root@fs1 "cryptsetup \
   -c twofish -s 192 -h ripemd160 create vol1-crypt \
   /dev/md0; /bin/mount /mnt/vol1"
```

- GnuPG gesichert
-
- …

USB-Stick

- Vorteil: kann sicher am Körper getragen werden (nicht in der Laptop-Tasche!!)
- Wenn alleine angewandt: sobald der Stick verloren geht ist der Key kompromittiert

GnuPG

- Man muss sich nur die Passphrase merken
  - Sichere Passphrase bitte!!
- kombinierbar mit anderen Verfahren
  - z.B. Key GPG-verschlüsselt auf USB-Stick

...andere Ideen

- Krypto-Karte
  - GnuPG Key per KryptoCard sichern
  - Key auf KryptoCard sichern
- Key im TPM speichern

Praxis

Verschlüsselungsmethoden

- Userspace
  - CFS (Crypto File System, NFS)
    http://www.crypto.com/software/
  - EncFS http://arg0.net/users/vgough/encfs.html
  - LUFS (Linux Userland File System)
    http://lufs.sourceforge.net
- Loopback Device
  - cryptoloop
  - loop-aes
- Devicemapper (ab Kernel 2.6.4)
  - dm-crypt

Userspace: LUFS

- speichert Dateien und Namen verschlüsselt
- Mountpoint enthält unverschlüsselte Dateien
- Bsp:

```
lufsmount cryptofs:///<source> <dest>

-[/tmp:] lufsmount cryptofs:///tmp/encrypted /tmp/plain/
Enter password:

echo "foo" > /tmp/plain/bar

root@adolar:/tmp# ls -l encrypted/
total 4
-rw-r--r--  1 root root 4 2005-02-25 00:03 raTI

sudo lufsumount /tmp/plain/

-[/tmp:] ls -l /tmp/plain/
total 0
```

Userspace: EncFS

- ähnlich wie CryptoFS
- basiert auf FUSE, kein Kernelmodul
- geht als user

```
encfs ~/tmp/encrypted/ ~/tmp/plain/

echo "foo" > plain/bar

-[~/tmp:] encfsctl info encrypted/

Version 5 configuration; created by EncFS 1.2.0 (revision
20040813)
Filesystem cipher: "ssl/blowfish", version 2:1:1
Filename encoding: "nameio/block", version 3:0:1
Key Size: 160 bits
Block Size: 512 bytes
Each file contains 8 byte header with unique IV data.
Filenames encoded using IV chaining mode.
```

Beispiel 1: Crypto-Swap

- Warum?
  - Daten werden ausgelagert
    - Key
    - GnuPG-Daten
    - Liebesbriefe
    - ...
- Womit?
  - loop-aes
  - dmcrypt
- beim Booten neuer zufälliger Key

Crypto-Swap: dm-crypt

- Swap mit zufälligem Schlüssel per Devicemapper

```
dd if=/dev/zero of=/dev/hda1 bs=64k conv=notrunc
cryptsetup -c blowfish -s 128 -d /dev/random create swap0 /dev/hda1
mkswap -L swap0 /dev/mapper/swap0
swapon LABEL=swap0
```

- -c Cipher (siehe /proc/crypto)
- -s Schlüssellänge in Bit
- dd um alte Daten auf /dev/hda1 zu löschen

Crypto-Swap: loop-aes

- Swap mit zufälligem Schlüssel per Loopback Device

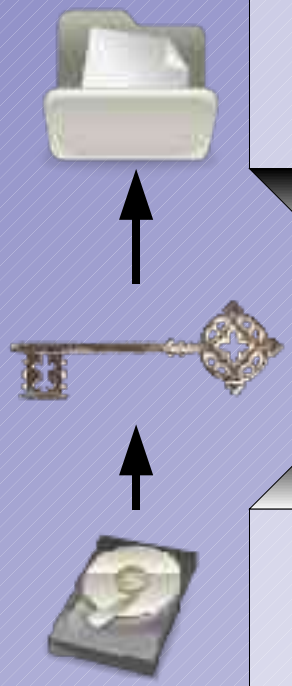
```
cat << EOF >> /etc/fstab
/dev/hda1 none swap sw,loop=/dev/loop6,encryption=AES128   0   0
EOF

dd if=/dev/zero of=/dev/hda1 bs=64k conv=notrunc
mkswap /dev/hda1
swapon -a
Setting up swapspace version 1, size = 1052794 kB


losetup -a
/dev/loop6: [0302]:163882 (/dev/hda1) offset=4096 encryption=AES128 multi-key
```

loop-aes Intro

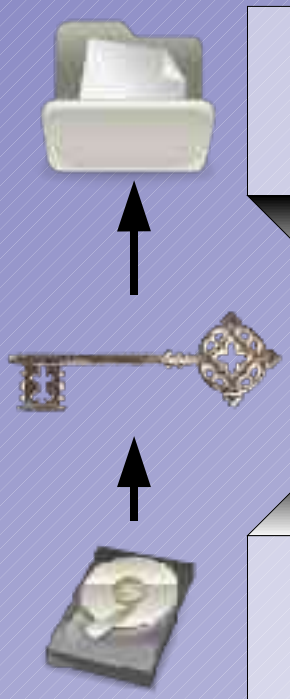
- verschlüsseltes Loopback Device
- v1(ein Key),v2 (64 Keys),v3 (64 Keys + 1Key für MD5 IV Berechnung) Keys



Dateisystem	ext3,xfs,reiserfs
verschlüsselndes Loopback Device	/dev/loop0 (max 8)
Blockdevice oder Datei	/dev/sda1,/dev/md0 /var/fs/home.img

dm-crypt Intro

- Device Mapper zwischen Block Device oder File und “virtuellem” Block Device



Dateisystem	ext3,xfs,reiserfs
verschlüsselnder Device Mapper	/dev/mapper/foo
Blockdevice oder Datei	/dev/sda1, /dev/md0

loop-aes: Partition

- einfaches Beispiel

```
dd if=/dev/hw_random of=/tmp/img count=204800 bs=512
losetup -e AES128 /dev/loop0 /tmp/img

losetup -a
/dev/loop/0: [0302]:203551 (/tmp/img) encryption=AES128

mkfs.ext2 /dev/loop0
mount -t ext2 /dev/loop0  /mnt/
```

- Merke! kein journaling FS auf eine Datei in einem journaling FS

loop-aes: Multikey

- 65 Schlüssel generieren, mit gnupg verschlüsseln und in Datei speichern
- Partition mit zufälligen Daten überschreiben
- Schlüsseldatei für Nutzer Pubkey verschlüsseln
- /etc/fstab Eintrag

-> siehe loop-AES README

RAID Array mit dm-crypt

- Raid wie immer anlegen unter z. B. /dev/md0
- mit Device Mapper verschlüseln
- LVM geht auch so

```
#!/bin/sh
cat ~/.key/fs1.vol1-key |  ssh root@fs1 "cryptsetup -c twofish  \
       -s 192 -h ripemd160 create vol1-crypt /dev/md0; \
       /bin/mount /mnt/vol1"

ssh fs1 “cat /etc/fstab”
   /dev/mapper/vol1-crypt /mnt/vol1 reiserfs noauto,acl 0 0
```

dm-re-encrypt (1)

- Backup der Daten sichern
- Lese-Mapping erstellen
  - Filesystem prüfen!
- Schreib-Mapping erstellen
- kopieren und beten

- Besser: via Backup

dm-re-encrypt (2)

```
root # cryptsetup -c twofish create alt /dev/hda1
Enter passphrase: ...
root # fsck.ext3 -f /dev/mapper/alt
```

```
[...]

/dev/mapper/alt: clean, 11/125184 files, 8045/249976 blocks

root # cryptsetup -y -c aes -h sha265 create neu /dev/sda1
```

```
Enter passphrase: ...

root # dd if=/dev/mapper/alt of=/dev/mapper/neu bs=4k
```

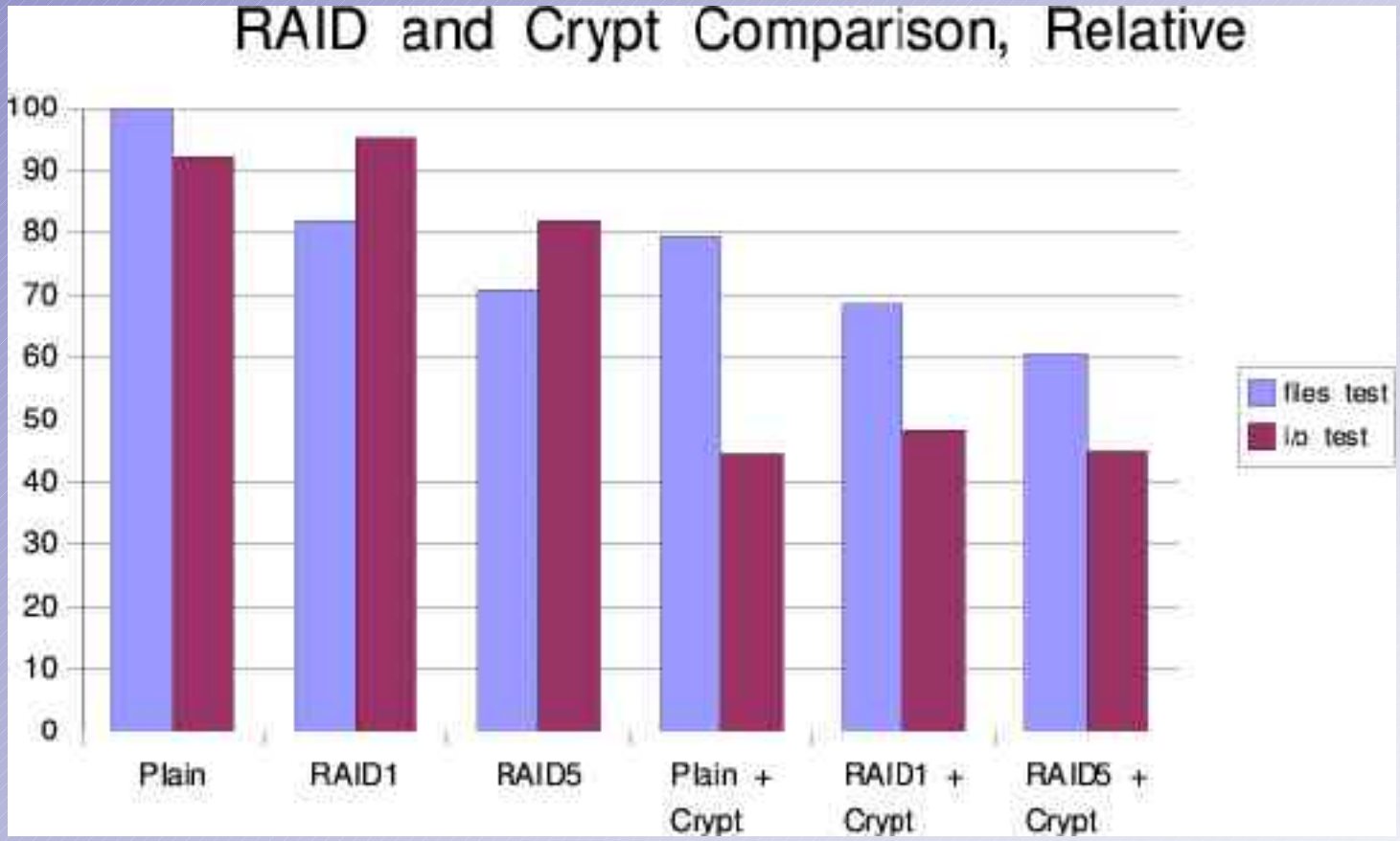
dm-crypt luks

- on-disk Format für Verschlüsselung
  - Parameter speichern
- “Schlüssel-Container” für mehrere User-Passwörter
- basiert auf TKS1, einem Modell für sichere Schlüsselverwaltungkey management on solid state disks.
- http://luks.endorphin.org/about
- http://clemens.endorphin.org/LUKS-on-disk-format.pdf

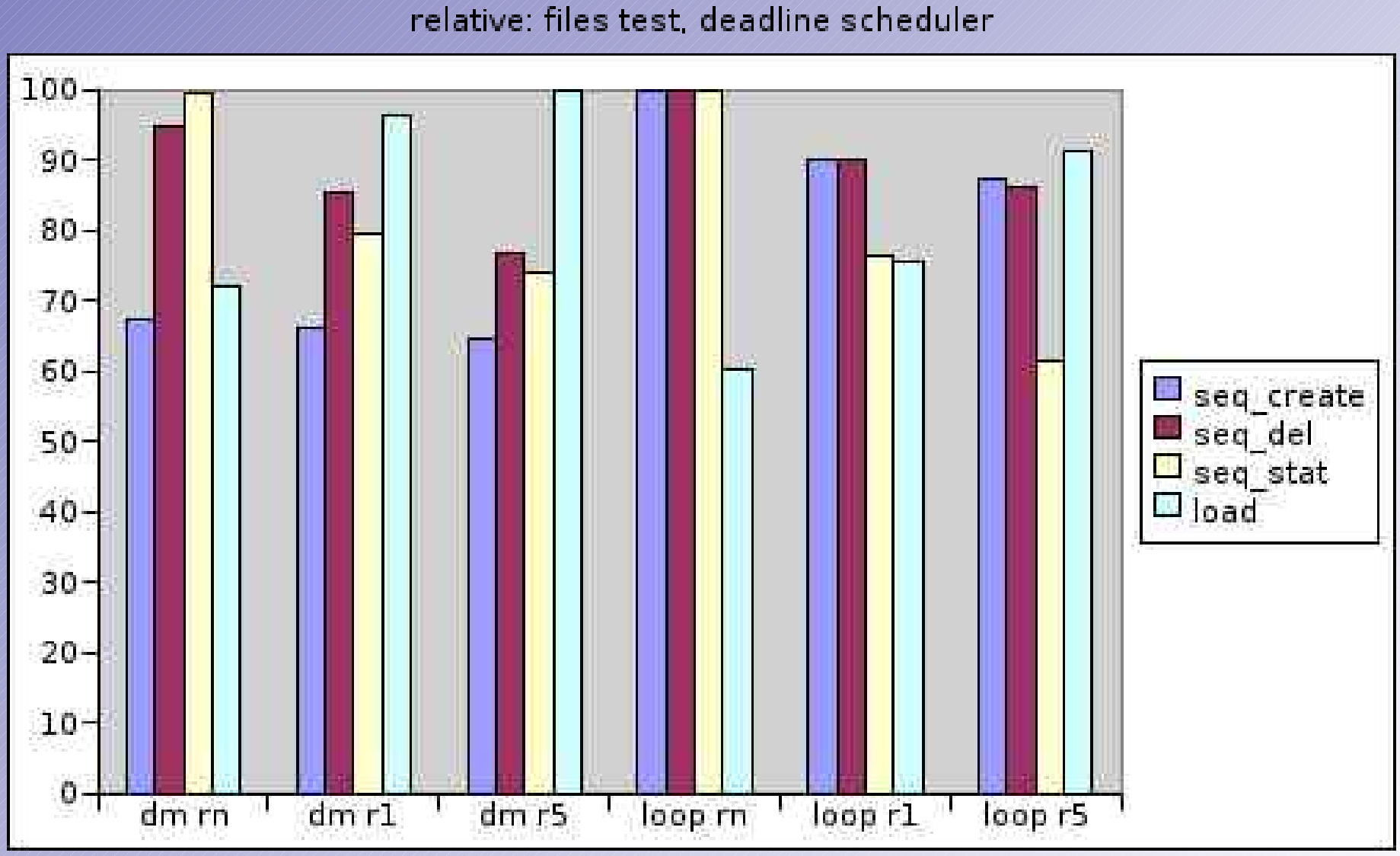
Performance

Performance

- Resultate von
  http://deb.riseup.net/storage/encryption/benchmarks/dmcrypt-v-loopaes/



Performance 2



Angriffe auf Kryptodateisysteme

Angriffe auf KryptoDateisysteme

- Content-Leak
  - Informationen über den Inhalt herausfinden
- Watermarking
  - herausfinen, dass eine Datei vorhanden ist
- Data modification leak
  - herausfinden, ab welchem Byte geändert wurde

Angriffe auf KryptoDateisysteme

- Malleable Plain Text
  - Daten gezielt manipulieren
- Moveability
  - Blöcke fast beliebig vertauschen

- häufig genutztes CBC ist anfällig
- neuere Modi nicht: CMC, EME, LRW

Vorsichtsmaßnahmen

Vorsichtsmaßnahmen

- speichern von /dev/random verhindern
  - bes. wichtig bei Crypto-Swap!
- sichere Passphrasen
- USB-Sticks mit Key niemals liegen lassen
- trotzdem auf Laptop aufpassen
- beim Tippen nicht murmeln
- verschlüsselter Swap, damit Keys nicht auf Platte gelangen können

Links

- http://clemens.endorphin.org/LinuxHDEncSettings
- https://wiki.boum.org/TechStdOut/LinuxCryptoFS
- http://deb.riseup.net/storage/encryption/benchmarks/dmcrypt-v-loopaes/
- http://www.saout.de/tikiwiki/tiki-index.php
- http://loop-aes.sf.net
- http://www.saout.de/tikiwiki/tiki-index.php?page=RootCryptoraid
- http://www.saout.de/tikiwiki/tiki-index.php?page=looptutorial
- http://docs.indymedia.org/view/Local/UkCrypto#Filesystem
- LRW http://clemens.endorphin.org/patches/
- Gentoo: http://forums.gentoo.org/viewtopic-t-242488-start-0.html

Fragen?

?