Festplattenverschlüsselung mit TPM 2.0

Chemnitzer Linux-Tage 2025

23. März 2025

Susanne Schütze
Linux Consultant
B1 Systems GmbH
schuetze@b1-systems.de

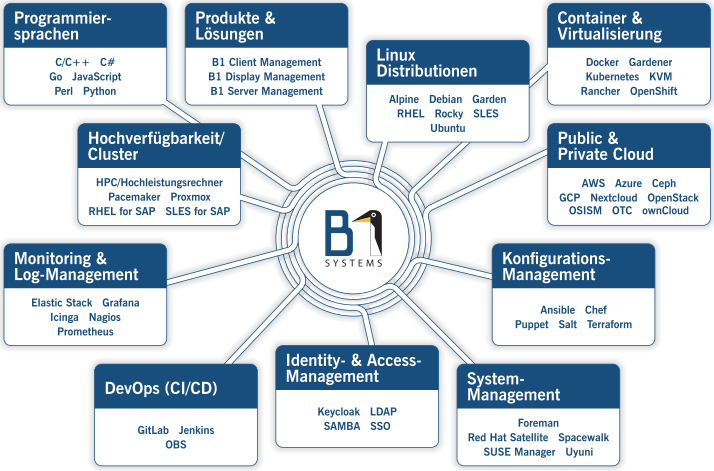
Inhaltsverzeichnis

Vorstellung B1 Systems
Was ist ein TPM?
Vorbereitungen
da-lockout
LUKS-Header
LUKS & TPM
PCR
Konfiguration in System
Unterschiede bei den Distributionen
Debugging
Fazit
Vielen Dank für Ihre Aufmerksamkeit

Vorstellung B1 Systems

- gegründet 2004
- spezialisiert auf Linux/Open Source-Themen
- national & international tätig
- ca. 150 Mitarbeiter:innen
- unabhängig von Soft- & Hardware-Herstellern
- Leistungsangebot:
  - Managed Service & Betrieb
  - Beratung & Consulting
  - Support
  - Training
  - Lösungen & Entwicklung
- Standorte in Rockolding, Köln, Berlin & Dresden

Schwerpunkte



whoami

- Susanne Schütze
- 41 Jahre
- Fachinformatikerin für Systemintegration
- bei B1 Systems GmbH seit Juli 2024
- berufliche Themen: Linux Client Management, Development, Ansible, Salt, etc.

Was ist der mysteriöse TPM?

Fast jeder hat einen TPM-Chip in seinem Gerät, doch was ist dieser mysteriöse TPM-Chip eigentlich?

- Abkürzung für „Trusted Platform Module“
- Aufbau insgesamt kompliziert - hier nicht weiter relevant
- herausragende Fähigkeit: mit Verschlüsselungskeys umgehen zu können und diese sicher zu speichern
- Möglichkeit, Measured Boot über PCRs zu realisieren
- Wichtig: hier ausschließlich TPM 2.0

Was sind PCRs?

- PCR steht für „Platform Configuration Register“
- Ablage für gehashte Werte
- Werte beziehen sich zum Beispiel auf secure-boot-policy, kernel-initrd, kernel-boot, kernel-config, ...
- Integrität anhand der Hash-Werte überprüfen
- Update-Prozesse für TPM PCRs sind noch nicht in Systemupdates integriert

↪ manuelles Eingreifen der User erforderlich

Spielwiese

- Ubuntu 24.04
- Virtuelle Maschine mit aktivierten TPM 2.0

```
--tpm backend.type=emulator,backend.version=2.0,model=tpm-tis
```

- benötigte Pakete:
  - TPM-Pakete:
    - `tpm2-tools`
    - `tpm2-abrmd`
    - `dracut`
  - clevis-TPM-Pakete (Alternative)
    - `clevis-luks`
    - `clevis-tpm2`
- Root-Rechte
- bereits eingerichtete Festplatten-Verschlüsselung (zum Beispiel bei der Installation)

TPM-Erkennung überprüfen

- Einhängepunkte für TPM 2.0
  - `/dev/tpm0`
    TPM allgemein
  - `/dev/tpmrm0`
    Resource Manager, mit dem der Kernel mehrere User-Zugriffe auf den TPM steuert

TPM-Prüfung mit `sytemd-cryptenroll`

```
1 # systemd-cryptenroll --tpm2-device=list
2 PATH        DEVICE      DRIVER
3 /dev/tpmrm0 MSFT0101:00 tpm_tis
```

da-lockout

Anfragen an den TPM können als Dictionary Attack gewertet werden. Zur Verhinderung wird `da-lockout` genutzt, das den Zugriff auf den TPM gesperrt.

Lockout-Mode-Erkennung

```
# tpm2 getcap properties-variables
TPM2_PT_PERMANENT:
  ownerAuthSet:              0
  endorsementAuthSet:        0
  lockoutAuthSet:            0
...
  inLockout:                 1
  tpmGeneratedEPS:           1
...
TPM2_PT_LOCKOUT_COUNTER: 0x0
TPM2_PT_MAX_AUTH_FAIL: 0x3
TPM2_PT_LOCKOUT_INTERVAL: 0x3E8
TPM2_PT_LOCKOUT_RECOVERY: 0x3E8
...
```

da-lockout aufheben

da-lockout aufheben

```
1 # tpm2_dictionarylockout --clear-lockout
```

da-lockout Authentifizierungsversuche erhöhen

```
1 # tpm2_dictionarylockout --setup-parameters --max-tries=5
```

den gesamten TPM beim nächsten Systemstart zurücksetzen

```
1 # echo 5 > /sys/class/tpm/tpm0/ppi/request
```

LUKS-Header

Überblick LUKS-Header mit systemd-cryptenroll

```
# systemd-cryptenroll /dev/vda3
SLOT TYPE
   0 password
```

Überblick LUKS-Header mit cryptsetup (gekürzt)

```
# cryptsetup luksDump /dev/vda3
LUKS header information
...
Keyslots:
  0: luks2
        Key:        512 bits
        Priority:   normal
        Cipher:     aes-xts-plain64
        Cipher key: 512 bits
        PBKDF:      argon2id
        Memory:     82040
        Salt:       9a 9f b7 e2
...
```

Entschlüsselung an den TPM binden

LUKS-Key an TPM binden

```
1 # systemd-cryptenroll --tpm2-device=auto /dev/vda3
2 Please enter current passphrase for disk /dev/vda3: ••••
3 New TPM2 token enrolled as key slot 2.
```

Überprüfung

```
1 # systemd-cryptenroll /dev/vda3
2 SLOT TYPE
3    0 password
4    1 tpm2
```

Clevis als Alternative zu `systemd-cryptenroll`?

Clevis LUKS-Key an TPM binden

```
1 # clevis luks bind -d /dev/vda3 tpm2 '{ "pcr_bank":"sha256"}'
```

Clevis Debugging

systemd-cryptenroll

```
# systemd-cryptenroll /dev/vda3
SLOT TYPE
   0 password
   1 tpm2
   2 other
```

cryptsetup (Ausschnitt)

```
# cryptsetup luksDump /dev/vda3
...
Tokens:
  0: clevis
        Keyslot:    2
...
```

clevis

```
# clevis luks list -d /dev/vda3
2: tpm2 '{"hash":"sha256","key":"ecc"}'
```

TPM und PCR

cryptsetup luksDump 1

```
# cryptsetup luksDump /dev/vda3
...
Keyslots:
  0: luks2
    Key:        512 bits
    Priority:   normal
    Cipher:     aes-xts-plain64
    Cipher key: 512 bits
    PBKDF:      argon2id
...
  1: luks2
    Key:        512 bits
     ...
```

cryptsetup luksDump 2

```
Tokens:
  0: systemd-tpm2
    tpm2-hash-pcrs:   7
    tpm2-pcr-bank:    sha256
    tpm2-pubkey: (null)
    tpm2-pubkey-pcrs:
    tpm2-primary-alg: ecc
...
    tpm2-policy-hash:
...
    tpm2-pin:         false
    tpm2-pcrlock:     false
    tpm2-salt:        false
    tpm2-srk:         true
    Keyslot:    3
```

PCR – Übersicht

PCR 7 wird standardmäßig vor Festplatten-Entschlüsselung geprüft

PCR Ausgabe

```
# systemd-analyze pcrs
NR NAME                SHA256
 0 platform-code       8612f3867faeb0fdadcdad675d95202ba256d1be3aeddc
 1 platform-config     a7eac3ef0871d29a2fb02312786d17f73ef743eb790035
 2 external-code       f1fa93cc54736b2e79a00f4d0ba566abb5d08dff9de70d
 3 external-config     3d458cfe55cc03ea1f443f1562beec8df51c75e14a9fcf
 4 boot-loader-code    30e68a8980f5cbd2cdb82dd5cdd46bae6cc90188dac07d
 5 boot-loader-config  28ff65fd8161c8ab7e909591816be3de4c07ef5af06ed8
 6 host-platform       3d458cfe55cc03ea1f443f1562beec8df51c75e14a9fcf
 7 secure-boot-policy  6dc6aeabe4d8207dad5a99dea56dc1ba5fca17a678c73a
 8 -                   016a167e0c5690c26b4d3c33f8a4fad08655e42b7c11c0
 9 kernel-initrd       516a9ca2c7f7b4786cb800c692caa137471a5809f58eed
10 ima                 9592ad63436356ecafe2c10fadccd4bc51a77b50e69194
11 kernel-boot         0000000000000000000000000000000000000000000000
...
14 shim-policy         306f9d8b94f17d93dc6e7cf8f5c79d652eb4c6c4d13de2
```

Lösungen für verwendete PCRs 1/4

Hashsumme im PCR 7 updaten:

systemd-cryptenroll

```
# systemd-cryptenroll --tpm2-device=auto --tpm2-pcrs=7 --wipe-slot=2
  ↪  /dev/vda3
```

clevis

```
# clevis luks regen -d /dev/vda3 -s 4
```

Lösungen für verwendete PCRs 2/4

keine PCRs verwenden:

systemd-cryptenroll

```
# systemd-cryptenroll --tpm2-device=auto --tpm2-pcrs='' --wipe-slot=2
  ↪ /dev/vda3
```

clevis

```
# clevis luks unbind -d /dev/vda3 -s 4
# clevis luks bind -d /dev/vda3 tpm2 '{ "pcr_bank":"sha256"}'
```

Lösungen für verwendete PCRs 3/4

andere PCRs verwenden:

systemd-cryptenroll

```
# systemd-cryptenroll --tpm2-device=auto --tpm2-pcrs=17,18
  ↪  --wipe-slot=2 /dev/vda3
```

clevis

```
# clevis luks unbind -d /dev/vda3 -s 4
# clevis luks bind -d /dev/vda3 tpm2 '{
  ↪  "pcr_bank":"sha256","pcr_ids":"17,18"}'
```

(es können auch eigene Hash-Werte in die PCRs geschrieben und gegen diese geprüft werden)

Lösungen für verwendete PCRs 4/4

keine PCRs verwenden und stattdessen eine TPM-Pin:

```
1 # systemd-cryptenroll --tpm2-device=auto --tpm2-with-pin=yes
  ↪  --wipe-slot=2 /dev/vda3
```

zusätzlichen Recovery-Key verwenden:

```
1 # systemd-cryptenroll --recovery-key /dev/vda3
```

zusätzlich Passwörter zum Entschlüsseln verwenden:

```
1 # systemd-cryptenroll --password /dev/vda3
```

Begrifflichkeiten

Wer glaubt, dass ...

A eine PIN immer für eine Zahlenfolge steht? X Nicht für systemd-cryptenroll

B eine PIN immer eine Form von Passwort ist? X bei Clevis steht PIN für PlugIN

C für `systemd-cryptenroll` eine PIN eine Passphrase ist? ✓

D unter Clevis und `systemd-cryptenroll` mit dem Begriff PIN das gleiche gemeint ist?X Der Begriff wird in den Projekten unterschiedlich definiert

initramfs mit dracut 1/2

Datei /etc/dracut.conf.d/tpm2-tss.conf anlegen

```
1 # Wichtig: Leerzeichen nach und vor den Anführungsstrichen!
2 add_dracutmodules+=" tpm2-tss crypt "
```

- Kernel-Optionen in Grub: Datei /etc/default/grub: GRUB_CMDLINE anpassen

nur TPM:

```
1 GRUB_CMDLINE_LINUX="rd.auto rd.luks=1
  ↪ rd.luks.options=tpm2-device=auto"
```

TPM mit PIN:

```
1 GRUB_CMDLINE_LINUX="rd.auto rd.luks=1
  ↪ rd.luks.options=tpm2-device=auto,tpm2-pin=yes"
```

initramfs mit dracut 2/2

- /etc/crypttab

bis systemd Version 255

```
1 #dm_crypt-0 UUID=ede52a15-f515-4fed-838d-5433999f3f24 none luks
```

ab systemd Version 256

```
1 dm_crypt-0 UUID=ede52a15-f515-4fed-838d-5433999f3f24 none
  ↪  discard,tpm2-device=auto,tpm2-pin=yes
```

initramfs und Grub updaten

```
1 # dracut -f
2 # update-grub
```

Welche Unterschiede bezüglich des TPMs gibt es bei Distributionen?

- es gibt unterschiedliche Tools zum Erstellen des initramfs
  - Dracut
  - Initramfstools
- es werden unterschiedliche Versionen von systemd eingesetzt
- es werden verschiedene Bootloader verwendet
- es werden unterschiedliche TPM-Bibliotheken eingesetzt

  `ibmtss2` IBM

  `tpm2-tss` Intel

Debugging mit dem TPM 1/5

Wer glaubt, dass folgende Meldung auf einen Fehler hinweist?

```
localhost kernel: Unknown kernel command line parameters "splash
↪ BOOT_IMAGE=/vmlinuz-6.8.0-49-generic tpm2-pin=yes", will be passed
↪ to user space.
```

X

Debugging mit dem TPM 2/5

Wer glaubt, dass folgende Meldung auf einen Fehler hinweist?

```
1 localhost systemd-udevd[278]:
  ↪ /usr/lib/udev/rules.d/60-tpm-udev.rules:3 Unknown user 'tss',
  ↪ ignoring.
```

X

Debugging mit dem TPM 3/5

Wer glaubt, dass folgende Meldung auf einen Fehler hinweist?

```
systemd[1]: systemd-tpm2-setup-early.service - TPM2 SRK Setup (Early)
↪ was skipped because of an unmet condition check
↪ (ConditionSecurity=measured-uki).
```

X

Debugging mit dem TPM 4/5

Wer glaubt, dass folgende Meldung auf einen Fehler hinweist?

```
gnome-remote-de[1092]: Init TPM credentials failed because Failed to
↪ initialize transmission interface context: tcti:IO failure, using
↪ GKeyFile as fallback
```

X

Debugging mit dem TPM 5/5

Wer glaubt, dass folgende Meldung auf einen Fehler hinweist? ✓

```
1 systemd-cryptsetup[513]: WARNING:esys:src/tss2-esys/api/Esys_StartAuthSession.c:391
  ↪  :Esys_StartAuthSession_Finish() Received TPM Error
2 systemd-cryptsetup[513]: ERROR:esys:src/tss2-esys/api/Esys_StartAuthSession.c:136:E
  ↪  sys_StartAuthSession() Esys Finish ErrorCode (0x0000098e)
3 systemd-cryptsetup[513]: Failed to unseal secret using TPM2: State not recoverable
4 systemd-cryptsetup[513]: Set cipher aes, mode xts-plain64, key size 512 bits for
  ↪  device /dev/disk/by-uuid/6b563880-9b80-4b25-a317-267617bbd26c.
5 systemd-cryptsetup[513]: WARNING:esys:src/tss2-esys/api/Esys_StartAuthSession.c:391
  ↪  :Esys_StartAuthSession_Finish() Received TPM Error
6 systemd-cryptsetup[513]: ERROR:esys:src/tss2-esys/api/Esys_StartAuthSession.c:136:E
  ↪  sys_StartAuthSession() Esys Finish ErrorCode (0x0000098e)
7 systemd-cryptsetup[513]: Failed to unseal secret using TPM2: State not recoverable
8 systemd-cryptsetup[513]: TPM2 operation failed, falling back to traditional
  ↪  unlocking: State not recoverable
```

TPM-Fehlercodes entschlüsseln

Fehlercodes 0x0000098e sind sehr verständlich für Menschen ;)

Übersetzung der TPM-Fehlercodes

```
# tpm2_rc_decode 0x0000098e
tpm:session(1):the authorization HMAC check failed and DA counter
  ↪  incremented
```

Dieser Fehlercode wird auch angezeigt, wenn die PIN falsch eingegeben wurde.

Nachteile des TPM in Bezug auf `systemd-cryptenroll`

- LUKS-Header müssen zum Bearbeiten mit Passphrase entschlüsselt werden; zukünftig geht auch `systemd-cryptenroll --unlock-tpm2-device=auto` ...
- wenn eine Entschlüsselungs-Option nicht funktioniert, wird nicht automatisch auf die nächste gewechselt (bis systemd Version: 255)
- PCR Measurements sind zur Zeit noch nicht Update-sicher
- mehrere Versuche die TPM-PIN einzugeben, nur wenn tpm2-measure-pcr=yes in `crypttab` gesetzt sind
- Debugging:
  - TPM-Fehlercodes müssen erst übersetzt werden
  - fehlerhafte PIN-Eingabe ist nicht erkennbar
- bei Diebstahl des Rechners ist eine Festplatten-Verschlüsselung mit TPM ohne Pin so sicher wie keine Verschlüsselung

Vorteile des TPM

- Festplatte kann mit dem TPM automatisch entschlüsselt werden
- User müssen sich keine langen und umständlichen Passphrases merken
- wenn sich am System etwas verändert hat, kann die Entschlüsselung gesperrt werden
- der TPM kann nun auch unter Linux genutzt werden, weitere Features sind in aktiver Entwicklung
  (weitere Vorträge zum Thema TPM in Planung)

Vielen Dank für Ihre Aufmerksamkeit!

Bei weiteren Fragen wenden Sie sich bitte an info@b1-systems.de oder +49 (0)8457 - 931096