“Trusted computing” a Linux

Michal Schmidt
Red Hat
2009-11-07

O čem to bude

- Co má znamenat “trusted”?
  - důvěra, měření, atestace
- TPM čip
  - RNG, SHA1, RSA, klíče, PCR, bind, seal
- Trusted Software Stack (TSS)
  - driver, TDDL, TCS, TSP
- Co s tím v Linuxu
  - kernel, trousers, tpm-tools, openCryptoki, IMA
- Intel TXT
  - MLE, PMR
- “trusted” nebo “treacherous”?

Trust – důvěra

- Důvěřujete svému počítači?
  - Určitě poslouchá právě Vás?
  - malware, viry, rootkity, keyloggery
- S Linuxem jsme sice **relativně** v klidu
  - Nic ale není dokonalé
    - ... a software už vůbec ne

- Jak vlastně definovat důvěryhodnost systému?

Důvěryhodnost podle TCG

- (TCG = Trusted Computing Group)
- “Entita je důvěryhodná, pokud se vždy zachová očekávaným způsobem pro daný účel.”
- Co může být ta entita
  - člověk, klika u dveří, program na počítači, ...
- “očekávaným” neznamená nutně “dobrým”
  - spam zombie
    - negativní důvěra

Rozhodnutí o důvěryhodnosti

- Víme-li, se kterou entitou máme co do činění,
- a známe-li vlastnosti této entity,

- pak se můžeme **my sami rozhodnout**, zda je skutečně důvěryhodná.

- “trusted” nebylo nejlepší pojmenování
  - dává pocit zaručené důvěry
- “trustable” by bylo výstižnější
  - dává pocit, že o důvěře bude rozhodnuto

Zpátky od entit k počítačům

- Jak dlouho vydrží důvěryhodnost?
  - Před týdnem jsme instalovali OS a programy na nové PC.
    - Důvěryhodné. Jiné bychom tam přece nedávali.
  - Lze ověřit, že systém je dnes pořád ještě důvěryhodný?
- Umíme vytvořit dokonale bezpečný software?
  - Ve stovkách milionů řádků kódu se vždy najdou chyby.
    - mnohé z nich s bezpečnostními důsledky
  - I kdybychom to uměli, nenahradíme veškerý používaný software ze dne na den.
  - Podpora v hw může
    - usnadnit detekci škodlivého kódu
    - chránit nejcitlivější data (soukromé klíče) i když dojde k prolomení do softwaru

Trusted Platform Module (TPM)

- čip v počítači
- Navržen tak, aby
  - soukromé klíče nemohly být prozrazeny.
  - umožnil detekci přidání škodlivého kódu.
  - škodlivý kód nemohl soukromé klíče zneužít.
- Poskytuje funkce pro
  - generování a uchovávání klíčových párů
  - měření integrity
  - atestaci (vydávání osvědčení)

Funkce TPM pro práci s klíči

- generování klíčových párů
    - s tím, že nikdy nevydá soukromý klíč ven v otevřeném tvaru
- podepisování
- ověřování
- šifrování
- dešifrování

Měření integrity s TPM

- “trusted boot”
- kontrolní součty během bootu
- jednoznačná identifikace, jaký systém byl nabootován
- Funkce “sealing”
    - TPM vydá uložené tajemství pouze tehdy, když kontrolní součty zvolených komponent mají očekávané hodnoty.

TPM funkce pro atestaci

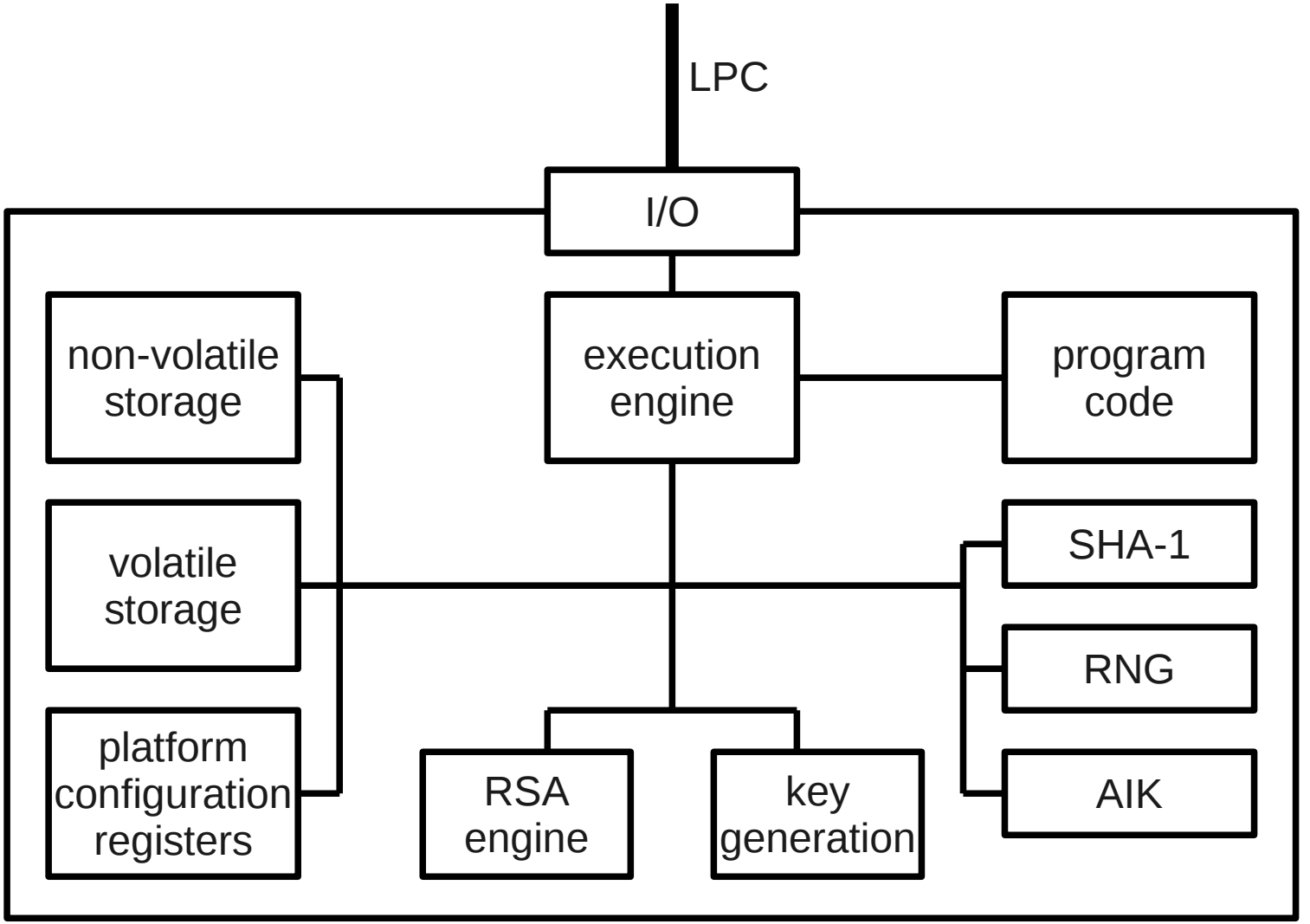
- atestace = podepsané osvědčení o změřené integritě systému.
- Lze tak prokázat vzdálenému systému, že byl spuštěn důvěryhodný software.

Trusted Computing Group

- dříve Trusted Computing Platform Alliance (TCPA)
- AMD, Fujitsu, HP, IBM, Infineon, Intel, Lenovo, Microsoft, Sun, Wave Systems, ... ~120 dalších.
- vývoj TPM a souvisejících standardů
- http://www.trustedcomputinggroup.org/

TPM čip

Blokový diagram TPM čipu



Komponenty TPM čipu

- I/O
  - Low Pin Count (LPC) sběrnice
  - TPM čip je slave – pouze odpovídá na příkazy
- execution engine
  - dekóduje příkazy z příchozího proudu dat
  - podle typu příkazu spustí odpovídající interní program v:
- program code
  - validuje příkaz a jeho parametry
  - ověří autorizaci příkazu
  - provede příkaz
  - sestaví odpověď

Paměť TPM čipu

- non-volatile storage (trvalá paměť)
  - Endorsement Key (EK)
    - obvykle nastaven už od výrobce, nemění se
    - Root of Trust for Reporting
  - Storage Root Key (SRK)
    - generuje se při nastavení vlastníka čipu (TPM_TakeOwnership)
    - Root of Trust for Storage
    - nadřazený všem dalším storage klíčům
- volatile storage (dočasná paměť)
  - aktuální stav čipu, kryptografické klíče, autentizační relace, transportní relace

Attestation Identity Key (AIK)

- Pro atestaci by šlo použít přímo EK, ale
  - ten je unikátní a neměnný => byl by tak při každém použití identifikován **konkrétní** TPM čip.
  - obvykle stačí prokázat, že podpis pochází od **nějakého** TPM čipu.
- S ohledem na ochranu soukromí se proto zavádí AIK.

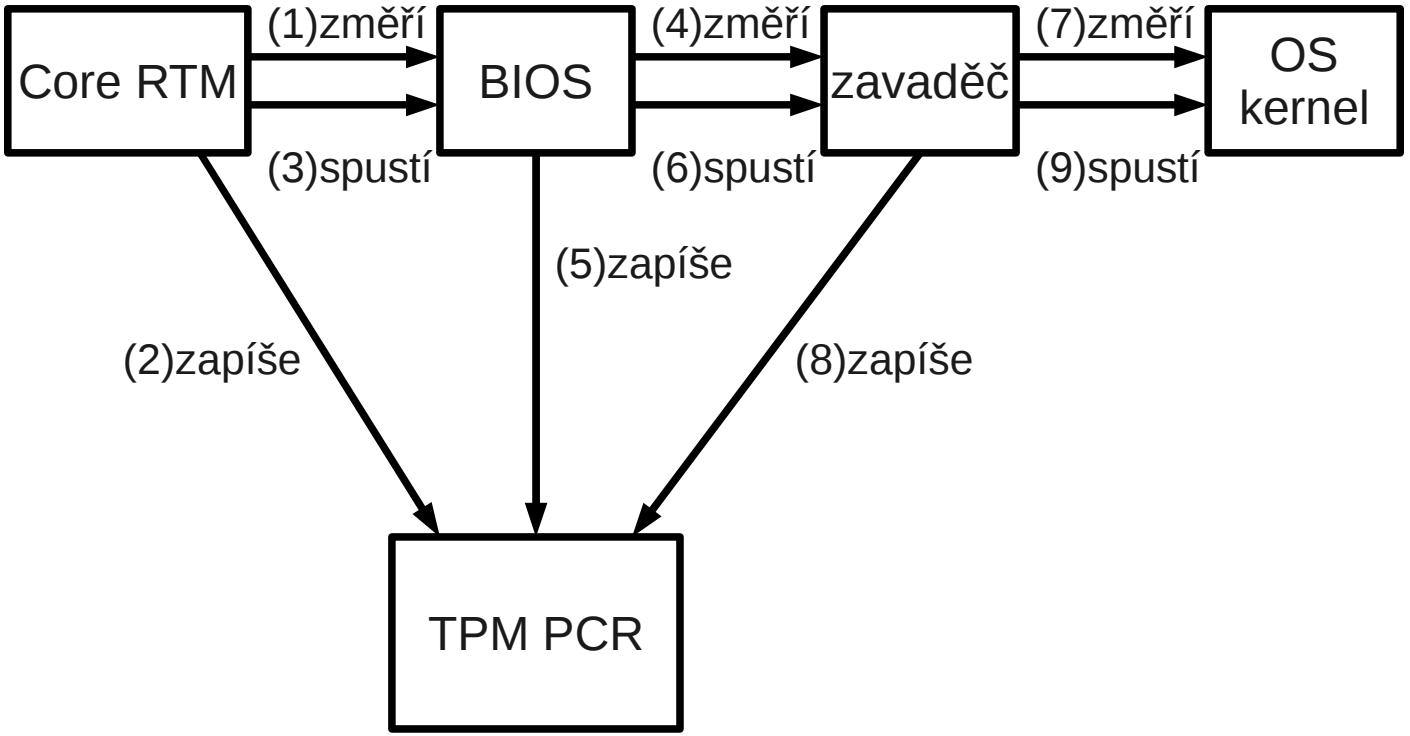
Platform Configuration Registers (PCR)

- nejméně 24 těchto registrů, každý má 20 bajtů
- Ukládají výsledky měření integrity.
- Nikdy do nich nelze zapsat hodnotu přímo, lze je pouze “rozširovat” (extend):
    - hodnota_PCR = SHA1(stará_hodnota_PCR, nová_hodnota)
- Díky tomu nelze výsledek měření dodatečně zfalšovat.
- Navíc tak každý PCR může pojmout neomezený počet měření.
- Záznam, co všechno bylo změřeno, se v TPM neukládá.
    - nutno ho udržovat externě

Trusted boot

- Nastartování do důvěryhodného stavu
- Měří se všechny kusy softwaru účastnící se bootování.
- Výsledky se ukládají do PCR.
- Postupný řetěz důvěry
  - Každý kousek změří následující kousek, dříve než mu předá řízení.

Trusted boot (zjednodušené schéma)



Co s výsledkem v PCR

- Máme nabootováno, hodnotami PCR je jednoznačně identifikován řetěz, který k tomu vedl.

- Správnými hodnotami PCR může být např. podmíněno vydání šifrovacího klíče pro odemknutí disku.
- Nebo tyto hodnoty bude chtít znát firemní server, než umožní přístup...

Vzdálená atestace

- Chceme serveru prokázat, že jsme provedli trusted boot do známé konfigurace.
- Server nám pošle *nonce.*
- Zavoláme *TPM_Quote*(*PCR_composite*, *nonce*)
  - TPM vygeneruje zprávu o aktuálním stavu PCR, přilepí *nonce* a celé to podepíše.
- Výsledek z *TPM_Quote* pošleme serveru.

Další klíče v TPM

- Tvoří stromovou hierarchii, kořenem je SRK.
- “Storage” klíče slouží k uchovávání dalších klíčů.
  - Podřízeny klíč je vždy zašifrován veřejnou částí storage klíče.
  - K načtení klíče do TPM tedy musí být předtím načten jeho nadřazený storage klíč a musí být splněna autorizace k použití tohoto nadřazeného klíče.
- Migrovatelné vs. nemigrovatelné
  - možnost přestěhování do jiného TPM
- Typy klíčů dle účelu:
  - storage, binding, identity, signature

Binding, Sealing

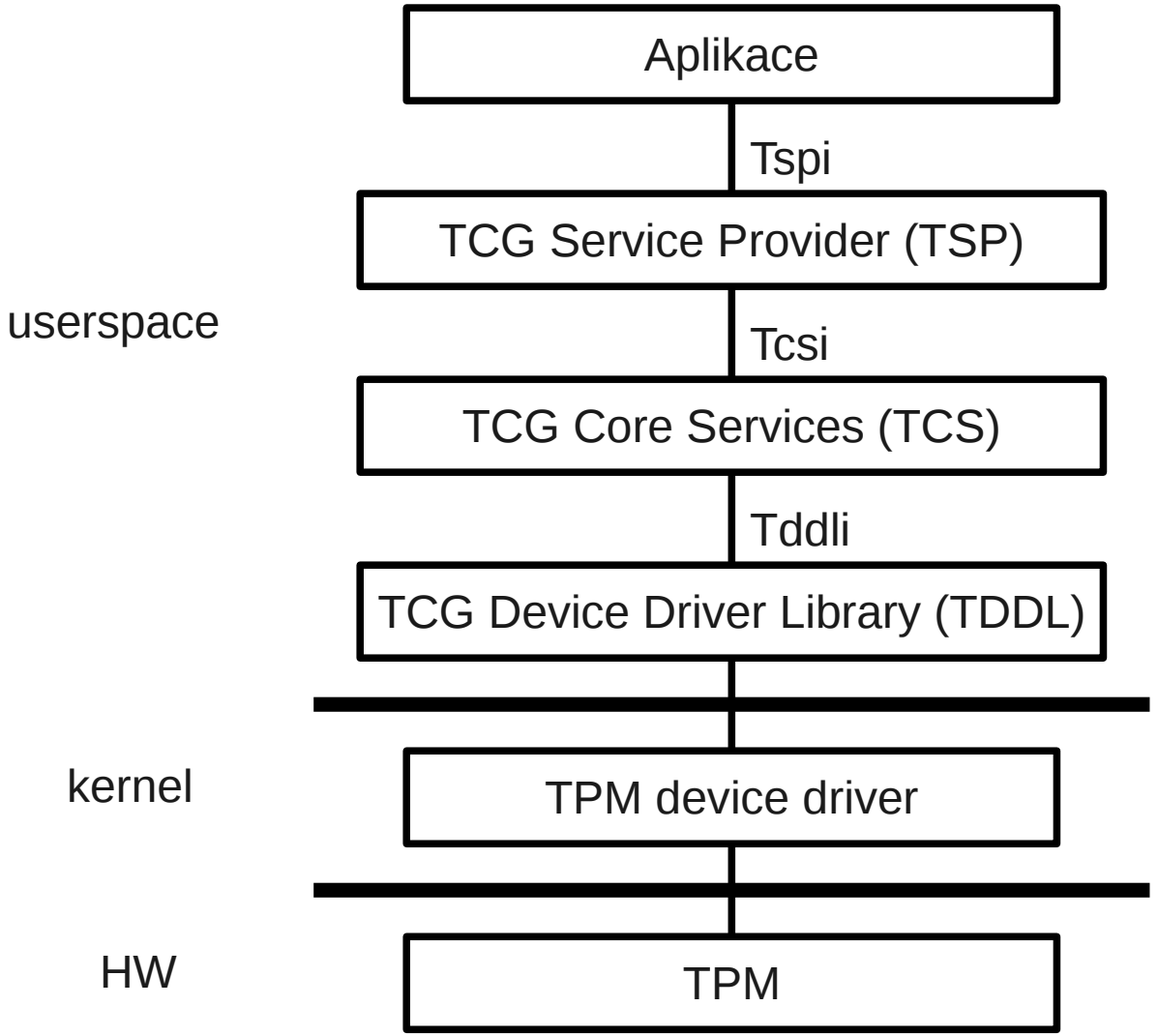
- binding
  - zamknutí symetrického klíče TPM klíčem
  - možnost vyžadování hesla pro odemknutí
- sealing
  - zamknutí symetrického klíče TPM klíčem ke stavu systému (PCR)
  - pro odemknutí musí souhlasit stavy PCR
  - možnost vyžadování hesla pro odemknutí

TCG Software Stack (TSS)

TCG Software Stack (TSS)

- TCG specifikuje nejen chování TPM, ale také softwarové vrstvy nad ním.
- TSS zjednodušuje práci s TPM a poskytuje další služby
  - schopnost ukládání klíčů na disk
  - vzdálený přístup k TPM přes síť
  - konverze dat mezi přenositelnými formáty
- TSS synchronizuje aplikace používající TPM.
- TSS se skládá z vrstev s přesně definovanými rozhraními.

Vrstvy TSS


Aplikace
Tspi
TCG Service Provider (TSP)
userspace
Tcsi
TCG Core Services (TCS)
Tddli
TCG Device Driver Library (TDDL)
kernel
TPM device driver
HW
TPM

TPM driver

- v kernelu OS
- pro starší TPM v1.1 byly různé pro různé výrobce TPM.
- TPM v1.2 standardizovalo i rozhraní TPM z pohledu OS.
  - TPM Interface Specification (TIS)
  - Stačí tedy univerzální TIS driver.

TCG Device Driver Library (TDDL)

- nejnižší userspace úroveň
- komunikuje s driverem
- poskytuje velmi základní rozhraní
  - open, close, transmit, receive
- neřeší souběžný přístup z více procesů
  - předpokládá synchronizaci na vyšší vrstvě

TCG Core Services (TCS)

- Běží jako daemon.
- Tento daemon je jediný proces v systému, který přistupuje k TPM (přes TDDL).
- Multiplexuje požadavky na TPM služby mnoha procesů, spravuje prostředky TPM, přepíná kontexty, swapuje klíče...
- Spravuje systémové úložiště klíčů.
- Jeho rozhraní Tcsi může být přístupné přes síť.
  - API ve stylu C funkcí. TCS je překládá na zprávy pro TPM.
  - SOAP

TCG Service Provider (TSP)

- Sdílená knihovna se kterou se linkují aplikace
- Přistupuje k TCS a navíc má některé vlastní služby.
- Poskytuje úložiště klíčů individuální pro uživatele.
- C rozhraní (Tspi)
- Základem API je TSP kontext.

TPM v Linuxu

TPM device driver v Linuxu

- Je už dlouho obsažen ve standardním jádře.

```
$ grep TCG /boot/config-2.6.31.5-122.fc12.x86_64
CONFIG_TCG_TPM=y
CONFIG_TCG_TIS=y
CONFIG_TCG_NSC=m
CONFIG_TCG_ATMEL=m
CONFIG_TCG_INFINEON=m

$ dmesg|grep -i tpm
tpm_tis 00:09: 1.2 TPM (device-id 0xB, rev-id 16)

$ ls -l /dev/tpm*
crw-rw----. 1 root root 10, 224  6. lis 17.17 /dev/tpm0

$ cat /sys/class/misc/tpm0/device/caps
Manufacturer: 0x49465800
TCG version: 1.2
Firmware version: 3.16
```


TSS v Linuxu

- Projekt `trousers`
- `http://trousers.sourceforge.net/`
- open-source TCG stack
- běžně obsažen v distribucích
- daemon jménem `tcsd`
- konfigurace `/etc/tcsd.conf`
- TSP: `libtspi.so.1`

tpm-tools

- Nástroje pro základní správu TPM

```
tpm_sealdata tpm_unsealdata tpm_changeownerauth
tpm_clear tpm_createek tpm_getpubek tpm_resetdalock
tpm_restrictpubek tpm_restrictsrk tpm_revokeek
tpm_selftest tpm_setactive tpm_setclearable
tpm_setenable tpm_setoperatorauth tpm_setownable
tpm_setpresence tpm_takeownership tpm_version
```


openCryptoki

- standard PKCS#11 pro kryptografické tokeny
- openCryptoki poskytuje PKCS#11 rozhraní pro (nejen) TPM
- pro kompatibilitu s PKCS#11 aplikacemi
  - i Firefox
- na PKCS#11 nejdou namapovat úplně všechny schopnosti TPM
  - např. omezení na typy použitelných klíčů

Integrity Measurement Architecture (IMA)

- Patch od IBM. V Linuxu od verze 2.6.30.
- Provádí měření spouštěných souborů
  - binárky, knihovny, kernel moduly
- Programy mohou kernel požádat o změření dalších souborů.
- Uchovává v paměti seznam provedených měření.
- Při použití TPM zaznamenává měření také do PCR.
- Parametr `ima_tcb` na příkazové řádce kernelu
- `mount -t securityfs security /sys/kernel/security`

IMA – ukázka

```
# cat /sys/kernel/security/ima/ascii_runtime_measurements
10 9b6b263a50a3c99f480212b9aef0da226e2c8900 ima c3d0ea6ff72be1bef863c2ce7a223555d423689d boot_aggregate
10 76f9b1801c7f37445b2e5b0c0aa92828fe79bb0c ima ca58fcb992f2c0a81b681df6657744c50eb001b0 /init
10 208dcb1aeaad67d4aa1a956b80ac98d4852757ca ima 5d34016daf204a970498de0e7a2242a9b5c15d49 /init
10 614f7a304623a8b56059d383ccd1683769cb9b89 ima 992d433d8f5f6a0f6e41c4e7b7b25eed6985933d ld-2.11.so
10 6ea2de457d1dc66657499a603d9d2318a61e3730 ima 8ddcb0ce63254236211f35d38817a9d85b1ff253 libc-2.11.so
10 09ea82cd2fb5e43595b5bd895e18f9b27f0caaec ima 786a7c75f644e467455914c07dff00de8f48c829 dracut-lib.sh
10 fc451c671c2536c02f6efa69676d972c5471cb1c ima 68191f7821ada3c8fa8c353de7a0a1ada5fc74a2 /bin/mknod
10 f18fa35447dce17b2bf9268f6dbbfe0e8bbfd20b ima 3a8f1ed0244d1ec50f65a27d57eec0cf6615f038 libselinux.so.1
10 ecceb85700770731ea919e35f26c2a881e1f4617 ima dc6446b9e5dfe22b2e737f7c5959016973e248ac libdl-2.11.so
10 467bba34673b20ab36fd29dd95bb4e869c124030 ima 8aea2c016e8b86fbadde869c2cab953b0170b6ad /bin/mount
10 f10425a692c0d5430b4a64b2a3136d4c1caf50ba ima 9eb6f4cfa110e84ae7f501d16ffff9ed3441d808 libblkid.so.1.1.0
10 e6cac1d6105a64e012cdaea7059911ca73229754 ima d97601efc8c35ff75b9dec700aef23da03fa3a7e libuuid.so.1.3.0
10 d2ddb7f8635c048d724453adb9f2cdb3db1a6287 ima cad6a4595760ace0976bc93c1024b15d6b764783 libsepol.so.1
10 22d673e0cc96dd3a16a1f78d7734a0c54f09ed0b ima 8ec6baab66cb24d44574c310a05458646bfb754c /sbin/modprobe
```


IMA

- mount option `iversion`
  - aby se znovu měřily změněné soubory
- /sys/kernel/security/ima/policy
  - lze definovat, které soubory se (ne)mají měřit

```
dont_measure obj_type=var_log_t
dont_measure obj_type=auditd_log_t
```

- aktivní IMA = výrazné zpomalení: načítají se celé soubory při namapování kousku
  - boot F12: z 23,4 s na 55,7 s
- ToMToU konflikt

Trusted boot Linuxu

- Je potřeba mít zavaděč, který provede měření a zapíše do PCR.
- TrustedGRUB
- GRUB-IMA

Intel TXT

Trusted Execution Technology (TXT)

- dříve LaGrande Technology
- Rozšíření CPU a čipsetu o ještě výraznější podporu trusted computing
- TPM je nutnou součástí.
- Silné oddělení prostředí vykonávání (execution environments)
  - k tomu využívá podporu virtualizace: VT-x, VT-d
  - ochrana před nedůvěryhodným kódem v ring 0
  - ochrana před DMA (Protected Memory Range)
  - mechanismy pro zabezpečený vstup z klávesnice/myši a zabezpečený výstup do graf. adaptéru

TXT

- ochrana před jednoduchými hw útoky
  - odpojení napájení, reset
  - šifrovaný přenos na pomalých sběrnicích (LPC)
- nemá ambice vzdorovat náročnějším hw útokům

TXT – launched environment

- dynamický kořen důvěry pro měření (DRTM)
- late launch – trusted prostředí je možno spustit kdykoliv, nezáleží na důvěryhodnosti předchozího stavu
- instrukce SENTER
  - utiší hardware, který by mohl ovlivnit měření
  - zkontroluje Authenticated Code (AC), pustí ho
  - změří Measured Lauch Environment (MLE) a spustí ho
    - což může být OS, hypervisor
- tboot – opensource; ale AC je blob

Trusted vs. treacherous

Kritika “trusted computing”

- ze strany EFF, FSF a dalších
- TC pomůže výrobcům omezovat uživatele.
- TC je nástroj pro posílení DRM (Digital Rights/Restrictions Management).
- TC potlačí hospodářskou soutěž.
- TC omezí svobodu modifikovat software.
- Hrozí ztráta dat při poškození TPM čipu s nemigrovanými klíči.
- Richard Stallman zavedl označení “treacherous computing” (treacherous = zrádný, zákeřný)

Zdroje

- http://www.trustedcomputinggroup.org/
- http://trousers.sourceforge.net/
- http://linux-ima.sourceforge.net/
- Challener, Yoder, Catherman, Safford, Van Doorn – A Practical Guide to Trusted Computing, IBM Press, 2007.
- Grawrock – Dynamics of a Trusted Platform, Intel Press, 2009