DM-VERITY ROOTFS INTEGRITY

Presented by frehberg <Frank Rehberger> (fr@frehberg.com)
FOSDEM 2023

Demonstrator: https://opencritis.org
WHAT IS DM-VERITY?

Family of Kernel Device Mapper (DM) Modules mapping physical block devs onto higher-level virtual block devs, for example

- dm-crypt: encryption/confidentiality (rw)
- dm-integrity: journaling (rw)
- dm-verity: authenticity/integrity (ro)

Since kernel v3.4, Android version 4.4, (late 2013)

https://docs.kernel.org/admin-guide/device-mapper/verity.html
$ veritysetup format rootfs.ext2 hash.img

DM-VERITY HASH TREE FORMATTING

Root Hash Signature

Root Hash
hash H-1..H-2

Hash H-1
hash H-1-1..H-1-2

Hash H-1-1
hash Block-1
Block B-1

Hash H-1-2
hash Block-2
Block B-2

Hash H-2
hash H-2-1..H-2-2

Hash H-2-1
hash Block-3
Block B-3

Hash H-2-2
hash Block-4
Block B-4
**DM-VERITY MAPS VIRTUAL BLOCK DEVICE**

Kernel device mapper module

- Integrity checking root filesystem
- (Authenticity) with signed root hash
DM-VERITY AS COUNTER MEASURE

Major threat: Manipulation of rootfs

- Detect Manipulation at startup
- Detect manipulation during runtime
- Terminate execution if manipulation is detected
- Deal with forward error correction (FEC)
- Minimal runtime overhead, ~Zero latency at startup. (compare to full image hash 15MB/s)
**DM-VERITY USE CASE: RICH LINUX**

- **Bare Metal**
  - Signed Binary
  - Fused SoC

- **Embedded Linux**
  - Signed Kernel
    - Init FS: ca 10MB
  - Signed Boot Loader
    - Device Tree
  - Fused SoC

- **Feature Rich Linux**
  - Signed FIT Image ca. 20MB
  - Kernel
    - Init FS: dmsetup
  - Device Tree
    - chosen.bootargs
  - Hash Tree
  - RootFS > 70MB
  - dm-verity verifies hash per block

- Signed Boot Loader
  - Fused SoC
INIT DM-VERITY WITHOUT INITFS

Kernel patch by Nathan Barrett-Morrison (Timesys)

DM-Verity Without an Initramfs
KERNEL COMMAND LINE

1 console=ttyAMA0,115200 ro rootwait
2 root=/dev/dm-0
3 systemd.volatile=overlay
4 dm-mod.create="verity,,,ro,0 122880 verity 1
5 /dev/sda4 /dev/sda5 1024 4096
6 61440 1 sha256
7 a100bab0d2e49b665bb18a0ee202b0fbc78d084e0d6cfe3b115b6e3908b8ac1
8 5027c03524f90df6e71f623ce059591a0aa3b48a39807826244c5f5e1db3a
9 3 ignore_zero_blocks
10 root_hash_sig_hex
11 3081df06092a864886f70d010702a081d13081ce020101310f300d06096086
12 dm_verity.require_signatures=1 rauc.slot=B
KERNEL OPTIONS

1 CONFIG_BLK_DEV=y
2 CONFIG_BLK_DEV_LOOP=y
3 CONFIG_MD=y
4 CONFIG_BLK_DEV_DM=y
5 CONFIG_BLK_DEV_MD=y
6 CONFIG_DM_INIT=y
7 CONFIG_DM_VERITY=y
8 CONFIG_DM_VERITY_FEC=y
9 CONFIG_DM_VERITY_VERIFY_ROOTHASH_SIG=y
10 CONFIG_SYSTEM_TRUSTED_KEYRING=y
11 CONFIG_SYSTEM_TRUSTED_KEYS="rootfs_cert.pem"
BENEFITS

- Rootfs Integrity during startup
- Rootfs Integrity during run-time
- Termination in case kernel detects manipulation
- Almost zero cost
THANK YOU