Xen system boot attestation with DRTM and TPM2

Xen Developer and Design Summit 2020

Michał Żygowski

3MDEB

Agenda

- Introduction
- What is TrenchBoot?
- Who contributes to TrenchBoot?
- What we have already achieved?
- TrenchBoot project roadmap
- safeboot
    - original boot flow
    - our boot flow
    - tpm2-attest
- Booting Xen with DRTM on AMD platform
- Generating quote for attestation
- Verifying the attestation quote
- DEMO
- Safeboot issues
- What we still need?
- References
- Summary

Introduction

Michał Żygowski
Firmware Engineer

- @*miczyg*
- michal.zygowski@3mdeb.com
- linkedin.com/in/miczyg
- facebook.com/miczyg1395

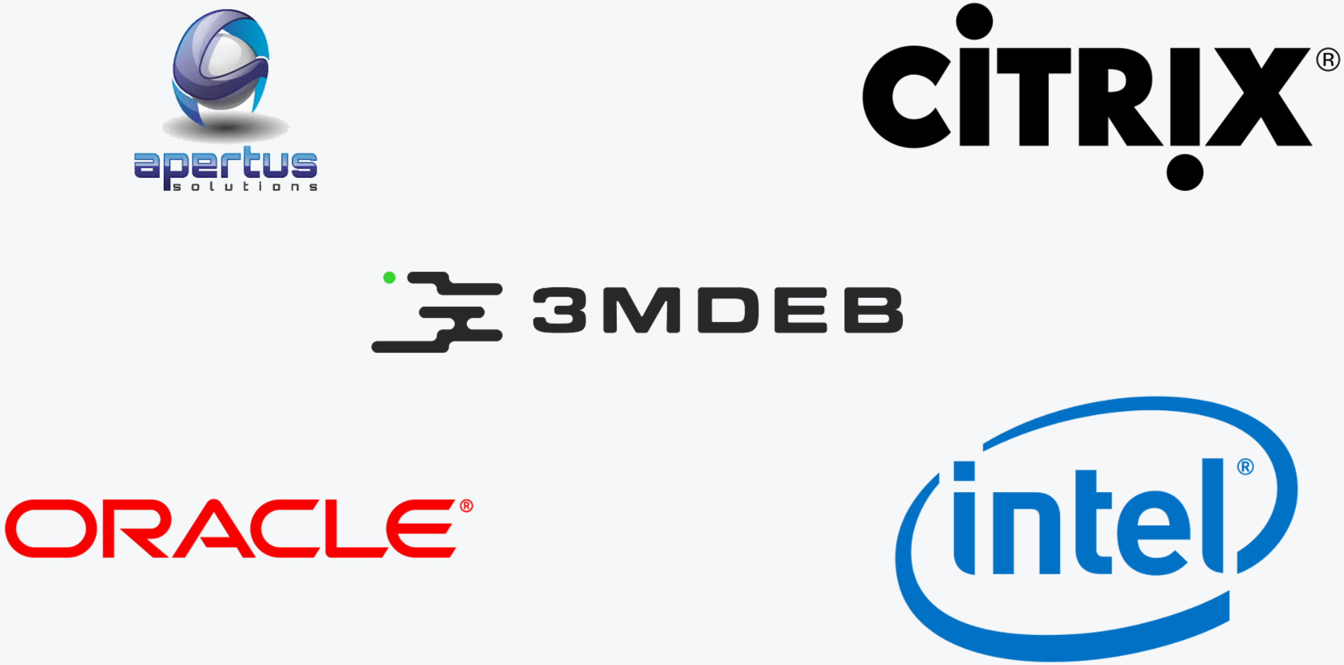
- Braswell SoC, PC Engines and Protectli maintainer in coreboot
- interested in:
  - advanced hardware and firmware features
  - coreboot
  - security solutions

What is TrenchBoot?

TrenchBoot is a cross-community integration project focused on launch integrity

- This means there is no "one thing" that is TrenchBoot
- The name was a play off of dealing with the muddy mess of trying to find a way to unify boot integrity
- The purpose is to develop a common, unified approach to building trust in the platform through launch integrity
- Works with existing open-source ecosystem to integrate the approach into their respective projects

Who contributes to TrenchBoot?



What we have already achieved?

3mdeb is beneficiary of the NLnet Foundation Next Generation Internet grant for Privacy and Trust Enhancing Technologies (NGI0 PET): https://nlnet.nl/project/OpenDRTM/

Thanks to the grant we were able to rapidly improve the support of AMD DRTM.

What we have already achieved?

- How TrenchBoot is Enabling Measured Launch for Open-Source Platform Security - Daniel Smith
    - https://youtu.be/f0LZFSq4Ack
    - "TrenchBoot was born out of limitations of using tboot to launch Xen for OpenXT project"
    - tboot "only supports Intel TXT, no love for AMD's Secure Startup" (2018/2019)
- OSFC 2019 - TrenchBoot - Open DRTM implementation for AMD platforms (3mdeb Piotr Król)
    - https://youtu.be/9NcVjsSu59w
    - First working implementation of TrenchBoot for AMD platform and first such open DRTM implementation in the world (Q3/Q4 2019)

What we have already achieved?

You may track our monthly progress on 3mdeb blog https://blog.3mdeb.com/tags/trenchboot/.

- Tested on variety of processors: family 16h G-series Embedded SoC, family 17h Ryzen and EPYC Embedded
- CI/CD for TrenchBoot related projects:
  - meta-trenchboot
  - GRUB2
  - Linux kernel
  - Landing zone
- Network boot with DRTM using iPXE
- Support legacy and UEFI environments (UEFI multiboot2 not yet tested/verified)
- TPM event log support with DRTM ACPI table
- Can launch Xen in legacy boot mode

TrenchBoot project roadmap

This is the planned roadmap for TrenchBoot AMD part:

- August 2020: Xen hypervisor support for TrenchBoot
    - Improve the security of the measured launch process
- November 2020: Remote attestation Proof of Concept with TrenchBoot and IETF RATS
- (now) - ??: upstream of the work
    - GRUB Intel TXT Secure Launcher RFC
    - x86: Trenchboot secure late launch Linux kernel support

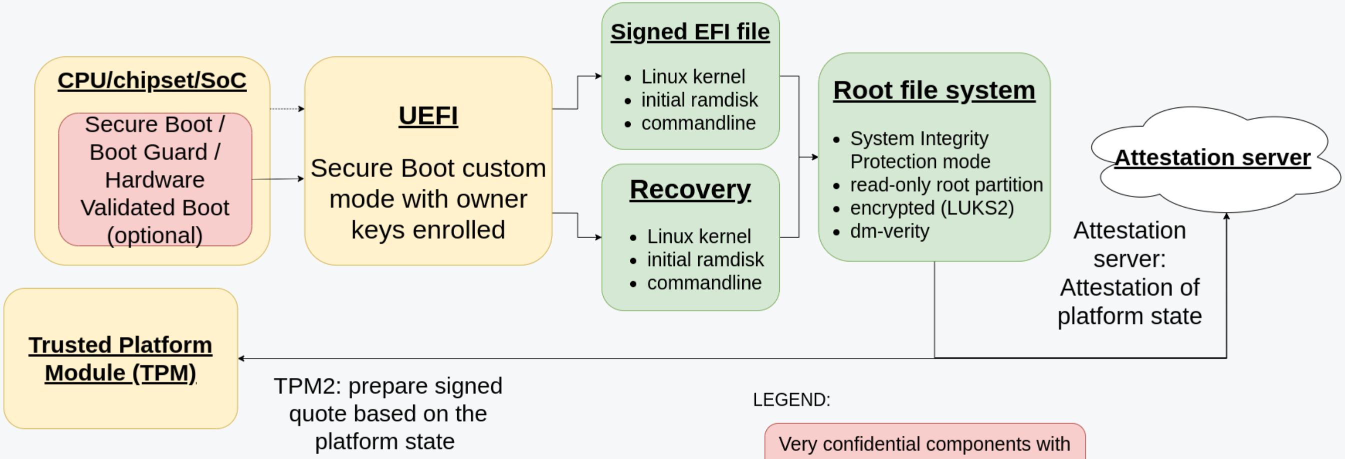
safeboot

Safe Boot (https://safeboot.dev) - booting Linux safely

Safe Boot has five goals to improve the safety of booting Linux on normal laptops:

- Booting only code that is authorized by the system owner (by installing a hardware protected platform key for the kernel and initrd)
- Streamlining the encrypted disk boot process (by storing keys in the TPM, and only unsealing them if the firmware and configuration is unmodified)
- Reducing the attack surface (by enabling Linux kernel features to enable hardware protection features and to de-privilege the root account)
- Protecting the runtime system integrity (by optionally booting from a read-only root with dmverity and signed root hash)
- Proving to remote systems that the local machine is safe (using a remote attestation protocol built with the TPM2)

safeboot original boot flow



CPU/chipset/SoC

Secure Boot / Boot Guard / Hardware Validated Boot (optional)

UEFI

Secure Boot custom mode with owner keys enrolled

Signed EFI file

- Linux kernel
- initial ramdisk
- commandline

Recovery

- Linux kernel
- initial ramdisk
- commandline

Root file system

- System Integrity Protection mode
- read-only root partition
- encrypted (LUKS2)
- dm-verity

Attestation server

Attestation server: Attestation of platform state

Trusted Platform Module (TPM)

TPM2: prepare signed quote based on the platform state

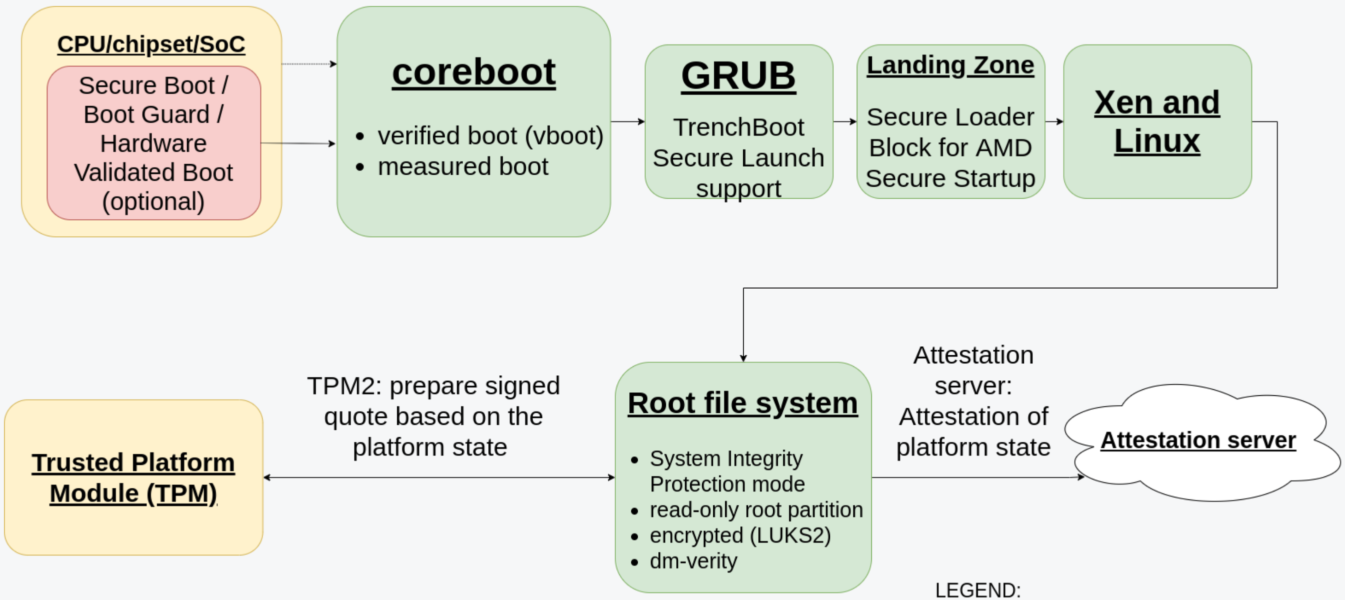
LEGEND:

🟥 Very confidential components with only scraps of knowledge in public how it works

🟨 Closed components, but the specifications are open, so the behavior of the component is known

🟩 Completely open-source component, fully controlled by the owner

safeboot our custom boot flow



CPU/chipset/SoC

Secure Boot / Boot Guard / Hardware Validated Boot (optional)

coreboot

- verified boot (vboot)
- measured boot

GRUB

TrenchBoot Secure Launch support

Landing Zone

Secure Loader Block for AMD Secure Startup

Xen and Linux

Trusted Platform Module (TPM)

TPM2: prepare signed quote based on the platform state

Root file system

- System Integrity Protection mode
- read-only root partition
- encrypted (LUKS2)
- dm-verity

Attestation server: Attestation of platform state

Attestation server

Hardware:
- PC Engines apu2
- AMD GX-412TC G-Seriec Embedded SoC
- Infineon SLB9665 TPM 2.0
- 4GB ECC RAM
- open-source firmware coreboot

LEGEND:

🟥 Very confidential components with only scraps of knowledge in public how it works

🟨 Closed components, but the specifications are open, so the behavior of the component is known

🟩 Completely open-source component, fully controlled by the owner

tpm2-attest

- A script that helps leverage certain TPM 2.0 features without deep tpm2-tools knowledge
  - Attestation quote generation
  - Attestation quote verification
  - Attestation quote verification against event log
  - Endorsement Key verification
  - Quote-based sealing/unsealing
- More on https://safeboot.dev/tpm2-attest/
- Used on both attestation server and attested platform
- Two commands are sufficient to attest the platform:

```
# assume we got a nonce from the server
(client) tpm2-attest quote $nonce $pcrs > quote.tgz
(server) tpm2-attest verify quote.tgz $nonce
```

Booting Xen with DRTM on AMD platform

- Modified GRUB2 with TrenchBoot Secure Launch
  - Two additional commands (***slaunch*** and ***slaunch_module***)
  - Hook into ***linux*** or ***multiboot2***
  - Setup the environment, DRTM module and TPM
  - Execute DRTM instruction SKINIT
- Landing zone
  - https://github.com/TrenchBoot/landing-zone
  - Secure Loader Block described in AMD Architecture Programming Manual
  - 64k block of code executed after issuing SKINIT and measured to PCR 17
  - Measures the main kernel to be executed (Linux kernel in case of linux command or Xen hypervisor and multiboot2 modules in case of multiboot2)
- Xen hypervisor
  - Already measured, loads Dom0 kernel

Generating quote for attestation

- Attestation quote generation wrapped in a single ***tpm2-attest*** script which:
  - Reads Endorsement Key (EK)
  - Creates an ephemeral Attestation Key (AK)
  - Gets a quote with the given Attestation Key
  - Attaches the TPM event log for additional verification

Verifying the attestation quote

- Attestation quote verification wrapped in a single ***tpm2-attest*** script which:
    - Unpacks the quote
    - Verifies the signature of the quote with AK public key
    - Verifies the event log and calculates the PCRs that should match those in quote
    - Optionally may verify the PCRs in quote and event log against know good PCRs
    - Verifies that the EK key in quote comes from a valid TPM based on the trusted root CA

DEMO time...

safeboot issues

- Possibly wrong environment variable for TPM access
  https://github.com/osresearch/safeboot/issues/47
- Cannot unseal LUKS key
  https://github.com/osresearch/safeboot/issues/48
- safeboot 0.6 release package does not contain tpm2-attest
  https://github.com/osresearch/safeboot/issues/49
- tpm2-attest script "tpm2 command not found"
  https://github.com/osresearch/safeboot/issues/50
- tpm2-eventlog-csv not working
  https://github.com/osresearch/safeboot/issues/51
- Lack of reference good-pcrs.txt file and format documentation
  https://github.com/osresearch/safeboot/issues/52

What we still need?

- Linux kernel and initrd measured before Xen is launched (it should rather be done before execution)
- add the protection against DMA for the kernel and modules in RAM using IOMMU
- easy way to access TPM event log from DRTM
- DRTM late relaunch (rather long term)
- emulated DRTM for virtual machines (also long term probably)

References

- safeboot: https://github.com/3mdeb/safeboot/tree/drtm_attestation
- Linux: https://github.com/9elements/linux/tree/google_firmware_generic
- GRUB2: https://github.com/3mdeb/grub/tree/trenchboot_support
- landing-zone: https://github.com/3mdeb/landing-zone/tree/mb2_eventlog

Summary

- It is hard to achieve reasonable security.
- We are getting closer and closer to the state when security will be easily available.
- This is kind of breakthrough in platform integrity and security area, since nobody has shown TPM-based attestation in open. safeboot is the first one to show attestation based on BIOS S-CRTM and here we have first DRTM-based attestation.
- Need for more open projects like TrenchBoot, QubesOS etc. which focuses on security, privacy and integrity.

Thanks

Special thanks to:

- Daniel Smith
- Andrew Cooper
- Krystian Hebel
- Piotr Król
- Rich Persaud
- Daniel Kiper

Q&A