

# TrenchBoot - the only AEM-way to boot Qubes OS

Qubes OS Summit 2022

Michał Żygowski



- Introduction
- Qubes OS Anti Evil Maid
- Qubes OS AEM status
- TrenchBoot project
- TrenchBoot as DRTM provider for Anti Evil Maid
- Project plans
- Demo
- Q&A



Michał Żygowski  
*Firmware Engineer*

-  [@\\_miczyg](https://twitter.com/_miczyg)
-  [michal.zygowski@3mdeb.com](mailto:michal.zygowski@3mdeb.com)
-  [linkedin.com/in/miczyg](https://linkedin.com/in/miczyg)
-  [facebook.com/miczyg1395](https://facebook.com/miczyg1395)
- Braswell SoC, PC Engines and Protectli maintainer in coreboot
- OpenPOWER System Software Technical Workgroup chair
- 5 years in Open Source Firmware
- interested in advanced hardware and firmware security features
- OST2 instructor
- TrenchBoot developer



- coreboot licensed service providers since 2016 and leadership participants
- UEFI Adopters since 2018
- Yocto Participants and Embedded Linux experts since 2019
- Official consultants for Linux Foundation fwupd/LVFS project since 2020
- IBM OpenPOWER Foundation members since 2020

- Qubes OS Anti Evil Maid is a set of software packages and utilities to aid against [Evil Maid attacks](#)
- **Requires TPM and Dynamic Root of Trust for Measurement (DRTM)** technology from silicon vendor to be present and supported by the firmware



## Questions:

- Can we trust hardware features silicon vendors provide?
- If we can trust the hardware and software we use, can we feel safe?
- How to determine if the state of the platform is trusted and hardware/firmware/software has not been tampered?

## Solution:

- Protection by ensuring the state of the platform
- Additional TOTP codes and secret sealing in TPM
- Trusted Execution / Trusted Computing:
  - TPM module by TCG
  - Intel TXT (DRTM)
  - AMD Secure Startup (DRTM)

## Intel TXT

- TPM required (discrete or integrated)
- BIOS ACM and SINIT ACM required
- Implementation: tboot
- BIOS needs to enable IOMMU, load and execute BIOS ACM
- Software needs to execute SINIT ACM (GETSEC[SENTER])
- Many GETSEC sub-instructions called leaf functions

## AMD Secure Startup

- Discrete TPM required (integrated not supported?)
- No blobs required
- Implementation: TrenchBoot
- BIOS only needs to enable SVM
- Software needs to execute a 64KB module (can be self-written) with SKINIT instruction
- Only 3 instructions: SKINIT/STGI/CLGI

```
sudo qubes-dom0-update anti-evil-maid
```

### Additional protection:

- Multi-factor with AEM USB boot device and TOTP
- Using 2 AEM USB sticks in case one could be stolen
- Using non-default SRK password
- Using additional secret key file for LUKS on AEM USB

### Attack still not prevented:

- Attacker can sniff passwords, keystrokes and access AEM USB stick
- Fake motherboard injection with radio link
- Successful measurement bypass by buggy CRTM implementations in BIOS
- Buggy BIOS updates leading to BIOS compromise
- SMM attacks leading to Intel TXT compromise

Current upstream status:

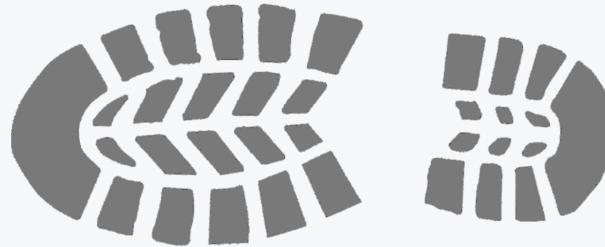
- **only for Intel silicon**
- **not supported on UEFI installations**
- **TPM 1.2 only**

Ongoing work:

- [On QubesOS Summit 2020](#) a PoC has been shown on AMD hardware that integrated TrenchBoot framework into GRUB and used SKINIT to extend the Xen and Dom0 to PCRs 17 and 18.
- Efforts to further extend the AEM support with TrenchBoot are ongoing

TrenchBoot is a framework that allows individuals and projects to build security engines to perform launch integrity actions for their systems.

# TrenchBoot



- The framework builds upon Boot Integrity Technologies (BITs) that establish one or more Roots of Trust (RoT) from which a degree of confidence that integrity actions were not subverted.
- <https://trenchboot.org/>

- Boot Integrity Technologies (BITs):
  - Intel TXT
  - AMD Secure Startup
- Currently targets Linux and GRUB
  - Patches for Intel TXT on [grub-devel](#)
  - Patches for Intel TXT on [lkml](#)
- 3mdeb implemented AMD Secure Startup Support thanks to [NINet NGI](#)  
[ZERO PET](#)
  - Linux Secure Launch
  - Xen Secure Launch
  - GRUB support for SKINIT
  - Secure Kernel Loader extension with TPM event log



- TrenchBoot may fill the gap of missing hardware support
- Hardware agnostic support for DRTM: both Intel and AMD
- Support for TPM2 regardless of boot mode: UEFI or legacy
- Decreased TCB due to removal of persistent tboot kernel

## Phase 1 (currently ongoing)

- Replace existing tboot implementation with TrenchBoot equivalent
- Support for Intel TXT and TPM 1.2
- Remove tboot kernel
- Reference hardware for testing:
  - Dell OptiPlex 9010 SFF (Intel Ivybridge, TPM 1.2 legacy boot)
  - Lenovo Thinkpad x230 (Intel Ivybridge, TPM 1.2 legacy boot)



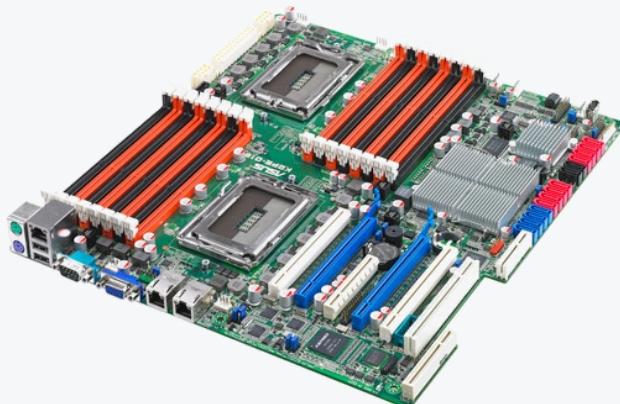
## Phase 2

- Extend AEM scripts with TPM 2.0 support
- Reference hardware for testing:
  - Protectli VP4670 (Intel Gen Comet Lake with TPM1.2 and TPM 2.0, legacy boot)



## Phase 3

- Integrate AMD Secure Startup support in AEM
- Reference hardware for testing:
  - ASUS KGPE-D16 (AMD Opteron 15h family with TPM 1.2 and TPM 2.0, legacy boot)
  - Supermicro MS11SDV (AMD EPYC 3000 with TPM1.2 and TPM2.0, legacy boot)



## Phase 4

- Support DRTM in Xen in UEFI mode
- Remove dependency on UEFI Boot Services in Xen for a cleaner separation between firmware and Qubes OS
- Make GRUB pass all information required by Xen via multiboot tags
- Reference hardware for testing:
  - Supermicro MS11SDV (AMD EPYC 3000 with TPM1.2 and TPM2.0, legacy and UEFI boot)



# DEMO time!

- Anti Evil Maid is an awesome feature of Qubes OS
- Not easy to maintain and improve (mainly due to complexity of DRTM technologies and/or firmware stacks - UEFI vs legacy)
  - Probably the main reason why it hasn't moved forward much for the past few years
- AEM requires DRTM technology to be present and supported by firmware, which limits the hardware choice drastically (at least Intel-based)
- Open-source firmware still pursues correct support for Intel TXT on newer devices
- TrenchBoot can bring solution to almost all missing pieces in Qubes OS Anti Evil Maid

# Q&A

