presented by

Advanced Trusted Platform Module (TPM) Usage

Fall 2018 UEFI Plugfest

October 15 – 19, 2018

Presented by HPBirdChen (AMI, Inc.)

Agenda

- Introduction and TPM Usage Overview
- Industry Updates on TPMs
- UEFI TPM Protocol Interface
- Using TPM to Secure a Platform
- Using Additional TPM Features
- Call to Action

Introduction and TPM Usage Overview

Introduction

The Trusted Platform Module (TPM) is a hardware based security chip that provides

- Root of trust for a system through measurements
- Attestation and authentication of data
- Security through the use of protected and shielded locations

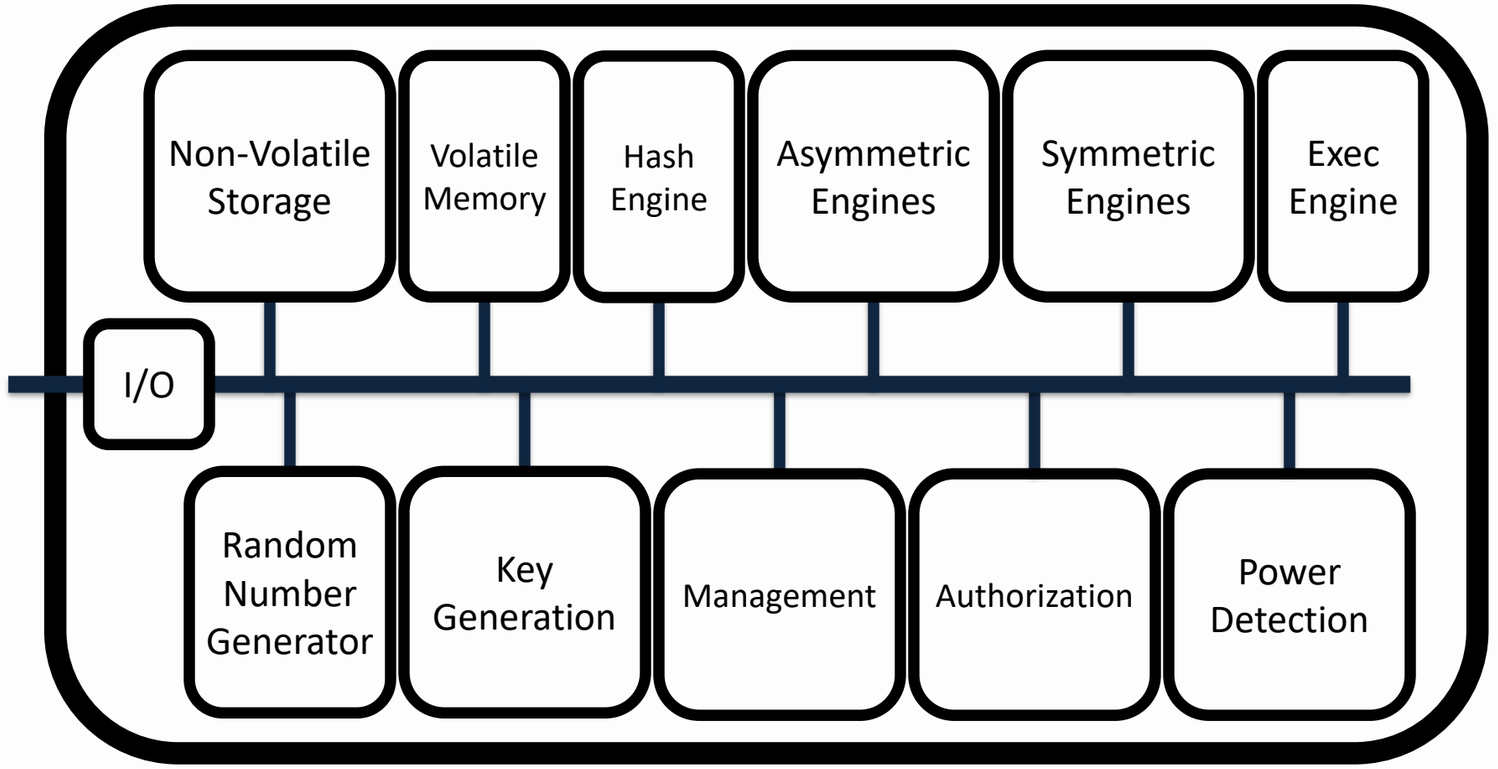
TPM usage is managed by the TPM specification developed by the TPM working group

The TPM working group is of part of Trust Computing Group (TCG), all published specifications can retrieved from:

- https://trustedcomputinggroup.org/

TPM Hardware Component

- TPM 2.0 Component Architecture



PCRs and Measurements

The TPM has a collection of registers called Platform Configuration Registers (PCRs)

- PCRs are shielded locations used to validate the contents of a log of measurement
- Data inside PCRs will be hashed using industry standard hashing algorithms:
  - PCR.digestnew[x] = HashAlg{PCR.digestold[x] || extend data digest}
- Hashing algorithms are irreversible functions that guarantee each extend operation can not be forged
- As system boots, binaries are hashed to PCRs

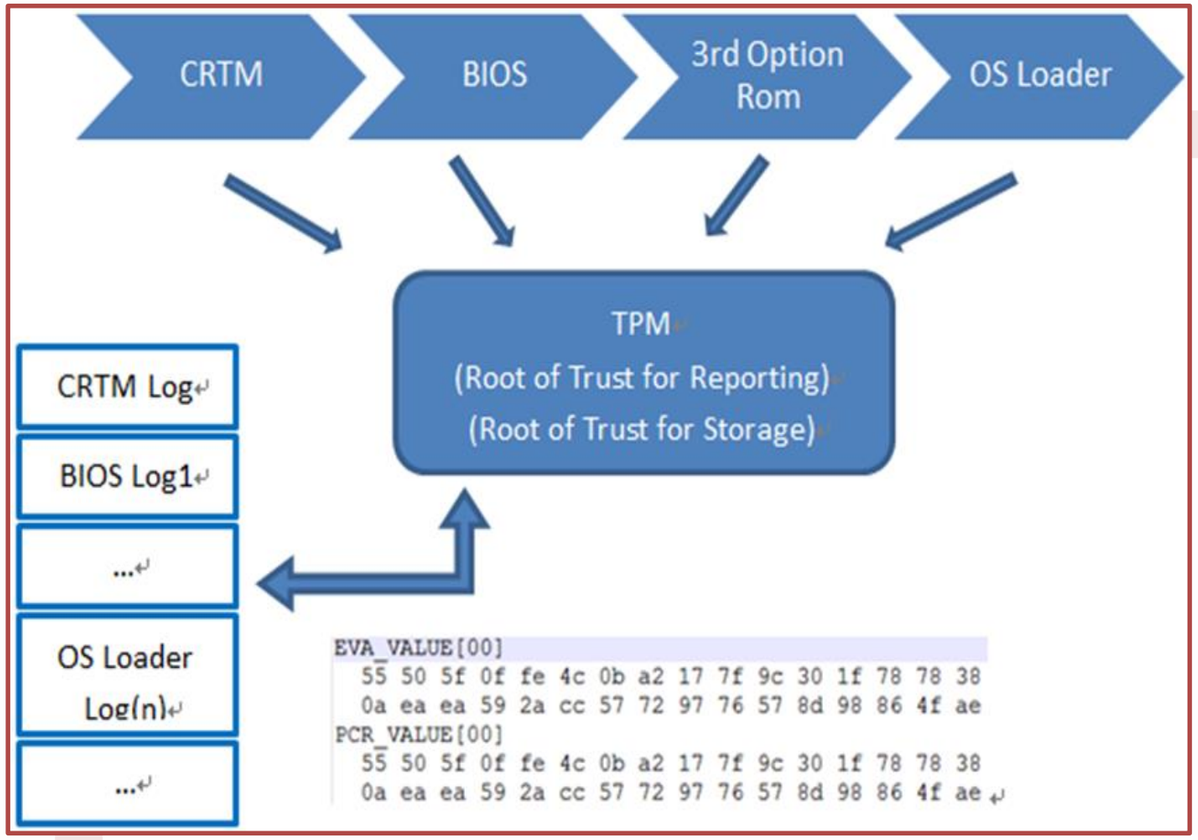
Measurements and Reporting

The process of adding a measurement into a PCR is as follows:

- The Initial starting point of measurement, called “Core Root of Trust for Measurement (CRTM),” is the first thing executed after power-on
- Many subsequent important items are measured into the TPM as well
- All measurements are reported to the OS for verification that system has not been modified
- Presentation by BIOS/Firmware Event Log

```
PCRIndex  : [00000007]
EventType : [80000007]
AlgorithmId:
  0b 00
Digest:
  a6 2b d6 7b 2c c2 95 97 66 51 b3 54 46 8c 00 47
  f8 d1 54 7d 25 05 6d ed 59 52 aa f5 99 17 62 a3
EventData: Size[000f]
  55 45 46 49 20 44 65 62 75 67 20 4d 6f 64 65    | UEFI Debug Mode
```

Chain of Trust



Each item in the boot sequence is required to be measured

Everything in the boot sequence is considered within the “Trust Boundary”

Validation chain of trust measurement.

- Verify firmware report event logging and PCR.

TPM Support on x86

TPMs have been commonly used on x86 systems since the first TPM

TPM usage has been well defined on x86

- Communication with the TPM has been on a fixed MMIO address
- In addition to HW TPM's from several vendors, both AMD and Intel have even developed their own forms of firmware based TPMs

OS vendors like Microsoft and the Linux community have had drivers to support TPMs on x86 for years

- Ecosystem has evolved over time and features like Microsoft BitLocker are common today

TPM Support on AARCH64

AARCH64 vendors are now entering the server space and are looking to use common technologies like TPMs

Using the fixed MMIO space from x86 does not work here, so new methods of communication are needed

- TCG group also has method of challenge response buffer for TPM transactions
- Using secure communication through SMC on ARM allows interfacing with a TPM through TrustZone

OS usage of TPM can remain the same as long as firmware provides TPM interfaces via ASL

- Even though TrustZone implementation can vary between ARM Si vendors, OS can remain generic because firmware describes the HW specific SMC layer via ASL

Industry Updates on TPMs

Loss of Legacy Interfaces

On x86, TPM communication was done through the LPC interface

- Future x86 systems may not include LPC interfaces
- New architectures like AARCH64 do not include LPC interface

New communication methods must be found!

SPI bus is common to all architectures and within the past 3 years, TPM vendors have provided SPI based TPMs

- SPI is faster than LPC so transactions are faster
- SPI bus design requires less signals than LPC so it is easier
- Chipset can even abstract this communication layer and all access through MMIO is valid

Interrupt Driven TPM Support

Current TPM usage has been sequential

- All TPM transactions are blocking where code execution is halted until completion

For efficient use of a TPM and to improve code performance, OS communication is now looking at using interrupt driven methods

- Similar to DMA, a request is submitted to the TPM and the TPM signals an interrupt when complete

For a TPM to be interrupt capable, HW and FW changes are required

Look for interrupt driven TPM support in 2020 (RS5 spec date?)

UEFI TPM Protocol Interface

TCG UEFI Protocol Definition

The UEFI TCG Protocol is used for communication with a TPM and the interface is defined as follows:

typedef struct tdEFI_TCG2_PROTOCOL {	
EFI_TCG2_GET_CAPABILITY	GetCapability;
EFI_TCG2_GET_EVENT_LOG	GetEventLog;
EFI_TCG2_HASH_LOG_EXTEND_EVENT	HashLogExtendEvent;
EFI_TCG2_SUBMIT_COMMAND	SubmitCommand;
EFI_TCG2_GET_ACTIVE_PCR_BANKS	GetActivePcrBanks;
EFI_TCG2_SET_ACTIVE_PCR_BANKS	SetActivePcrBanks;
EFI_TCG2_GET_RESULT_OF_SET_ACTIVE_PCR_BANKS	GetResultOfSetActivePcrBanks;
} EFI_TCG2_PROTOCOL;	

Using the UEFI TCG Protocol

Common uses of the UEFI TCG Protocol include:

- HashLogExtendEvent: Is called to hash data/executable before using it to a PCR
- SubmitCommand: Is called to send a specific command to the TPM
  - Can include clearing, writing to TPM NVRAM, or any command supported by the TPM
  - Refer to the TCG specification for full list
- Additional functions can be called to switch PCRs, get TPM capabilities, retrieving the event log, but are not within the scope of this presentation

Programming Example

Below is a pseudocode example of hashing a UEFI executable before running it:

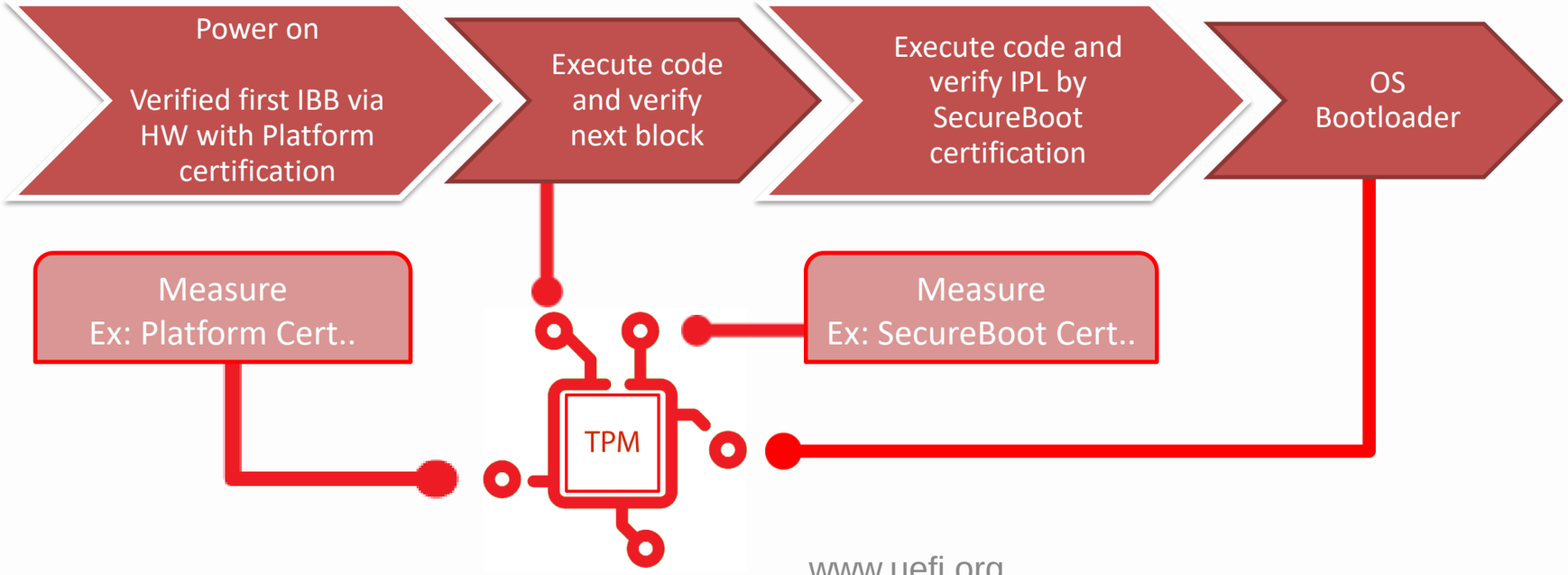
- Status = gBS->LocateProcol(TcgProtocol, TcgProtocolGuid)
- Status = TcgProtocol->HashLogExtendEvent(UEFIExecutable)
- Status = pBS->LoadImage(UEFIExectuable)
- Status = pBS->StartImage(UEFIExecutable)

Using TPM to Further Secure a Platform

TPM Usage in Additional Security Architectures

Verified boot and UEFI SecureBoot are additional security architectures that use the TPM to:

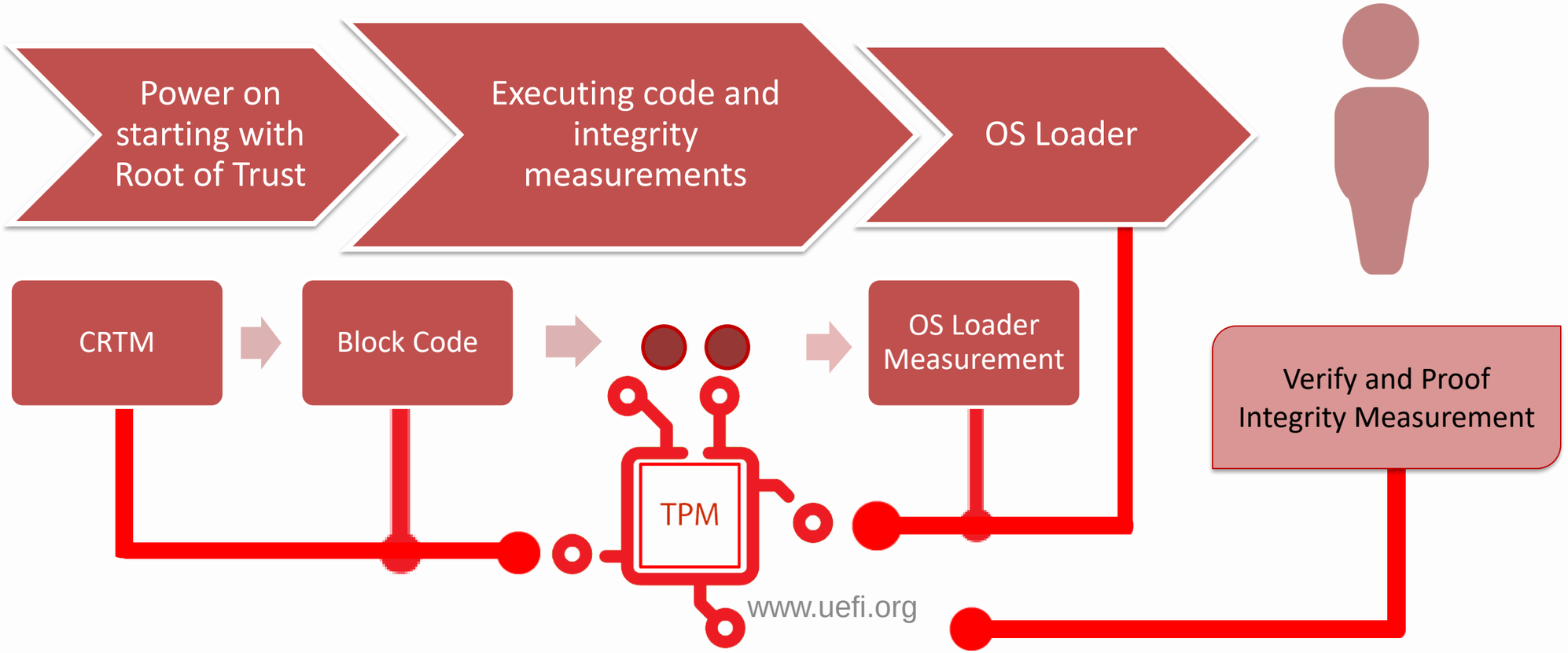
- Preform additional measurements of items like keys
- Do additional measurements before power-on making a true hardware root of trust



Measurements Role in Security Chain

Measured Boot:

- After all measurements are placed into PCRs according to TPM specifications, the OS and other third party applications can uses these measurements to attest system integrity



Proof and Attestation

Example attestation guideline

- NIST SP800-155
- https://csrc.nist.gov/csrc/media/publications/sp/800-155/draft/documents/draft-sp800-155_dec2011.pdf
- Allows IT admins to verify system integrity remotely
- NIST SP800-193
- https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-193.pdf
- Integrity “Root of Trust/Chain of Trust/Reporting” on firmware security chain and resiliency.

Attestation

- TPM has capability of chaining certificates so that a common certificate authority can authenticate multiple measurements from platforms with approved TPMs

Using Additional TPM Features

TPM Secure Storage

TPM storage can be used for more than just measurements

TPM NVRAM Storage provides a method to store data where authentication is required to store/retrieve

- Multiple levels allows for more than one user of NVRAM Store

- Protected from erasure when firmware is updated

- Data is stored inside chip where it cannot be externally extracted

Note TPM NVRAM space is limited!

Additional TPM features

Various flexible set of Algorithm agility and Policies

TPM support various algorithm, such as

- Symmetric keys (DEC, AES, etc.)
- Asymmetric keys (RSA, ECC, etc.)
- Hash algorithm (SHA-1, SHA-256, SM3 .. etc.)

TPM support flexible set of Policies

- Authorization :
  - Password, HMAC, PCR Binding, Signed.
- Restrict : Time and Count.
- Can create a policy by user defined way:
- data context
- Specific command executing policy
- AND (PCR, Data) , OR (Signd)

Putting it Together

TPM secure storage is safer than SPI NVRAM

- Physical access to TPM storage is not available
- Secure PWs, sensitive data, or other items can be moved to TPM based storage, but space is limited

TPM has many cryptographic algorithms built in

- Algorithms can be used to verify OS loader or OEM binaries on top of UEFI SecureBoot before execution
- OS loader can even be encrypted and must be decrypted by the TPM before launching

Call to Action

Call to Action

TPMs usage and design have been very static in the PC industry for many years

Recently TPMs have been updated and moved to new architectures

As TPMs are evolving, try finding new uses for TPMs other than just measurements by the FW

Thanks for attending the Fall 2018
UEFI Seminar and Plugfest

For more information on the Unified EFI Forum and UEFI Specifications, visit http://www.uefi.org

presented by