Agenda

Definitions

Using the AIK

TPM Basics

Attestation

Keys

Measurement

Chain of Trust

Copyright © 2006-8 Trusted Computing Group - Other names and brands are properties of their respective owners.
Family in Rome
Around and About

Other names and brands are properties of their respective owners.
Omaha Beach
My Work Over Last Two Months

Rainbow trout (*Oncorhynchus mykiss*)
Definitions

**Attestation** – 1. an act of attesting. 2. an attesting declaration; testimony; evidence

- **Source** – www.dictionary.com

**Trust** – An entity can be trusted if it always behaves in the expected manner for the intended purpose

- **Source** – TCG documents

**Measurement** – The process of obtaining the identity of an entity. Normally this is a cryptographic hash

- **Source** – TCG documents

**Security** – A condition that results from the establishment and maintenance of protective measures that ensure a state of inviolability from hostile acts or influences

- **Source** – www.wikipedia.org
Functional TPM Diagram

TPM

Root of Trust for Reporting RTR
- Provides cryptographic mechanism to digitally sign TPM state and information

Root of Trust for Storage RTS
- Provides cryptographic mechanism to protect information held outside of the TPM

Root of Trust for Measurement RTM
- Provided by platform to measure platform state
- Defined by platform specification

Interaction between RTR and RTS is important TPM capability

Copyright © 2006-8 Trusted Computing Group - Other names and brands are properties of their respective owners.
PCR

Never write directly to PCR, ALWAYS append and hash

- Uses cryptographic properties of hash to mitigate attacks

PCR maintains history of input values in a single hash output register

- Allows for better use of TPM resources
Agenda

Definitions

Using the AIK

TPM Basics

Attestation

Keys

Measurement

Chain of Trust
Agenda

- Definitions
- Using the AIK
- TPM Basics
- Attestation
- Keys
- Measurement
- Chain of Trust
Chain of Trust

Goal is to gain trust in Entity C

Operational standpoint is that A launches B and B launches C

- To trust C one must trust B
- To trust B one must trust A

A to B to C creates a “Chain of Trust”

- Another term in use for this is “Transitive Trust”
- Trust is transitive from A to B to C
- It does not invert, trusting A does NOT imply that I must trust C
- Trusting C REQUIRES me to trust A and B
Chain Measurement

What does one need to “trust” the chain

- The identity of each item in the chain
- From definitions identity = measurement
- Therefore A measures B before passing control to B
- B measures C before passing control to C

Generic flow is

- Receive control
- Measure next entity
- Pass control to entity

That works for the chain but who measured A?

Entity A  Entity B  Entity C
Root of Trust

A Root of Trust is an entity that must be trusted as there is no mechanism available to measure the entity.

When creating a chain of measurements the first entity in the chain MUST be the Root of Trust for Measurement (RTM).

- Becomes the anchor of the chain

A platform may have more than one RTM available.

- The Static RTM (SRTM) gains control on each boot of the platform
- The Dynamic RTM (DRTM) gains control upon invocation of a specific

If more than one RTM is available it means that more than one trust chain is possible.
Chain Size

Once the RTM measures one entity the process is the same for any additional measurements.

To make everything easier to understand the rest of this class will use only the RTM and entity B.

It is possible to make the chain as large as one likes.
Agenda

- Definitions
- TPM Basics
- Attestation
- Keys
- Using the AIK
- Measurement
- Chain of Trust
Normal Measurement

1. RTM measures (SHA) Entity B
2. RTM creates event structure
3. RTM stores event structure in SML
4. RTM extends value into PCR

Details on Event Structure, SML, and TPM in subsequent slides

Only after performing the extend operation into the TPM does the RTM pass control to entity B

Copyright© 2006-8 Trusted Computing Group - Other names and brands are properties of their respective owners.
Event Structure

Each Event Structure contains, at a minimum, two pieces of data:

- Extend value or the actual result of the digest calculation
- Extend data or metadata that describes the entity measured and potentially information regarding the environment that performed the measurement

One structure for each measurement extended into the TPM

---

Copyright © 2006-8 Trusted Computing Group - Other names and brands are properties of their respective owners.
The Stored Measurement Log (SML) contains the event structures for all measurements stored in the TPM.

The SML has no requirements for protection, either confidentiality or integrity.

The SML could be on the hard drive or some other storage device.

- The SML is NOT internal to the TPM.
The TPM is the repository for measurements, but so is the SML

The difference is the extend operation

- PCR = digest(old, new)
- PCR value equals the digest of the old value with the requested extend value
- The TPM hardware enforces the extend and no outside entity can force the PCR to a specific value
  - Other than the result of the digest calculation
  - Can’t say set the PCR to 1, but can say extend PCR with value A which results in PCR value of D
Agenda

- Definitions
- Using the AIK
- TPM Basics
- Attestation
- Keys
- Measurement
- Chain of Trust
Verification

1. Verifier reads SML for event structure
2. Verifier calculates expected value
   Calculation is the extend algorithm
3. Verifier obtains PCR value from TPM
4. Verifier compares expected with actual

If the expected does not agree with the actual the verifier knows that the SML does not agree with the TPM, there is no indication of where any attack did occur.

What evidence does the verifier need to be validate the response from the TPM
Step 3 obtains the current value from the TPM

The verifier needs a mechanism to validate the report from the TPM

The TPM can perform a digital signature of the PCR value

The mathematics of a digital signature prove that the value is unchanged from when signed

But how to prove that the signature comes from a specific TPM and not from a software masquerade
Simple idea is to put a unique signature key in each TPM

Create a list of known TPM signing keys

Just validate that public key for signing the PCR is from the list and voila one knows that the attestation comes from a known TPM

Is there a problem here?
Verifier 1 and verifier 2 both obtain a report from the same digital signature key.

Verifier 1 and verifier 2 now can correlate that the platform connected to the TPM has had contact with both of the verifiers.

Using the same key provides a mathematical proof that of the correlation.

Need a mechanism to break the correlation.
What Is This?
Writing the Spec
Agenda

- Definitions
- TPM Basics
- Using the AIK
- Keys
- Attestation
- Measurement
- Chain of Trust
Attestation Identity Key (AIK)

The attestation identity key (AIK) provides the mechanism to break the correlation property

The TPM can create a virtual unlimited number of AIK

- In a 1.2 TPM the AIK is a 2048-bit RSA signature key

With an unlimited number of AIK there is the ability to use a different AIK for each verifier

But now there needs to be a way to link the AIK to the credentials such that the verifier knows the AIK represents a valid TPM
Creating the AIK Credential

1. TPM Owner creates new AIK
2. Software creates AIK credential request
   Bundle includes EK pub, AIK pub, platform and EK credentials
3. Bundle sent to AIK CA
4. AIK CA verifies credentials
5. AIK CA determines validity of platform
6. AIK CA creates AIK credential signing using CA priv
7. AIK CA encrypts AIK credential using EK Pub
What does the AIK CA need to have sufficient evidence to create the AIK credential?

The bundle contains the EK and platform credential and the new AIK pub key

AIK CA validates signatures on EK credential and platform credential

- Implies that CA has access to the public keys associated with the signature keys
- CA needs to get these public keys and validate authenticity
  - Could be done previously to this request for AIK credential

AIK CA validates that platform not on list of known bad platforms

- More details in following slides

AIK CA validates that platform meets the CA’s requirements for trusted platforms
Delivering AIK Credential

The AIK CA has no proof, to this point, that the AIK credential will only be in use by the TPM that contains the AIK
- The next step provides the proof

Remember that the AIK credential is encrypted by the EK public key
- Technically the EK pub only encrypts a symmetric key which does the encryption of the credential, but the point remains the same

The only entity with access to the EK private key is the TPM

The only way to obtain the credential is to ask the EK to decrypt the credential
- The only way to use the EK is to ask the TPM to perform the operation
- Hence the only way to gain access to the credential is to use the TPM to decrypt the credential

Copyright© 2006-8 Trusted Computing Group - Other names and brands are properties of their respective owners.
Using An AIK

1. Platform request service from verifier
2. Verifier requests PCR value
3. TPM uses AIK to sign PCR value
4. Platform returns signed PCR and AIK credential
5. Verifier performs digital signature validation
6. Verifier validates validity of AIK Credential
7. Verifier knows PCR from valid TPM
8. Verifier can make trust decision based on PCR value
Sealing Data to the TPM

1. Caller determines data to seal and desired PCR value
2. Caller collects data, PCR, and authorization to unseal
3. Caller sends to TPM collected data
4. TPM encrypts collected data and adds proof to blob
5. TPM outputs sealed blob
6. Note that SEAL does not use current PCR value
7. Note use of storage public key
Unsealing Data from the TPM

1. Caller authorizes UNSEAL of blob
2. TPM decrypts blob if authorization correct
3. TPM validates that proof value correct
4. TPM validates that PCR contain desired value
5. TPM outputs sealed data
6. What value is proof?
QUESTIONS?
Generic Architecture

- **Signature keys and encryption keys held in shielded locations**
  - TPM supports unlimited number of signature and encryption keys

- **Credentials held outside TPM**
  - Endorsement credential normally provided by TPM manufacturer
  - Platform credential normally provided by platform manufacturer
  - Conformance credential provided by lab

---

Endorsement credential

Platform credential

Conformance credential

Shielded Locations

TPM

- EK
- AIK
- PCR

Platform

Copyright© 2006-8 Trusted Computing Group - Other names and brands are properties of their respective owners.
What is TCG Technology

**Trusted Platform Module (TPM)**
- Defines a set of services
- Adds protocols and messages that take advantage of the TPM

**TPM is a platform component**
- Bound and attached to the platform

**The TPM contains**
- Cryptographic engine
- Protected storage

**Functions and storage are isolated**
- Provides a “Trust Boundary”
Agenda

Definitions

Using the AIK

TPM Basics

Attestation

Keys

Measurement

Chain of Trust
Basic TPM Block Diagram

I/O

Storage
- Volatile
- Non-volatile

TCG
- Opt In
- Seal

Keys
- Create
- Use
- AIK

Execution
- Code
- Engine

Crypto
- RNG
- Hash
- PCR

Copyright © 2006-2008 Trusted Computing Group - Other names and brands are properties of their respective owners.
TCG PC Client H/W Design

TPM attached to motherboard

- Attachment is permanent

TPM communicates to chipset through Low Pin Count (LPC) bus

- All PC client implementations use LPC bus
- Other busses possible on other platforms
Other Features

Protected Storage

• TPM Owner can allow use of TPM persistent storage
• TPM protected storage provides authenticated access to storage area

Monotonic Counter

• Persistent long term counter capable of an increment every 5 seconds for 7 years
TSS Block Architecture

- **TSS SPI**
- **TSS CSI**
- **TPMDDL**

**Process 1**
- TSS Service Provider

**Process 2**
- RPC Client

**Remote Process**
- TSS Service Provider
  - RPC Client

**TSS Core Services**

**TPM Device Driver Library**

**TPM Device Driver**

**TPM**

*Copyright© 2006-8 Trusted Computing Group - Other names and brands are properties of their respective owners.*
Persistent Keys

TPM provides persistence for two keys, EK and SRK

- Endorsement Key (EK)
- Storage Root Key (SRK) is the RTS for the TPM
- EK is not part of external storage hierarchy

Signature and encryption keys rooted in SRK

- Signature and encryption keys only loaded when needed
- More later
Endorsement Key (EK) Details

Each TPM has unique EK

- 2048-bit RSA
- Generated when entity willing to create endorsement credential in control
- EK is changeable under specific circumstances

EK only participates in two operations

- Take ownership of the TPM
- Creation of Attestation Identity keys
Storage Root Key (SRK) Details

Each TPM has a single SRK
- 2048-bit RSA
- Generated when TPM ownership occurs
- Changes each time TPM owner changes

Key operations rooted in SRK
- Loading a storage key or signature key always based on SRK
- Only SRK permanently resident in TPM
Loaded Keys

Signature and encryption (storage) keys always loaded

- Multiple RSA key sizes possible
- All loaded keys rooted in SRK
- Possible for TPM owner to allow loaded key to use persistent storage

Loaded keys only useable when loaded by SRK chain

- Key encrypted by SRK (or parent) and hence only useable with correct SRK
Changing the Trust Arrow

**Trust Platform as Built**
- Build
- Evaluate
- Ship
- Use

**Trust Based on Platform properties**
- Measure
- Attest
- Evaluate
- Use

Copyright© 2006-8 Trusted Computing Group - Other names and brands are properties of their respective owners.