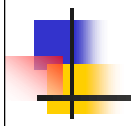


Trusted Computing Technology and Client-Side Access Control Architecture

ISA 767, Secure Electronic Commerce
Iliano Cervesato, icervasa@gmu.edu
George Mason University
Spring 2006

Acknowledgement: Some slides and diagrams are adapted from TCG Architecture Overview, Intel IDF Fall 03, and Boot Camp's TCG 101 Presentation

Outline

- Trusted Computing
  - TCPA/TCG Trusted Platform Module
  - Intel LaGrande Technology
  - Microsoft NGSCB
- Client-side Access Control Architecture and Protocols using TC
  - Motivations
  - Architecture and Protocols
  - Applications

Terminology

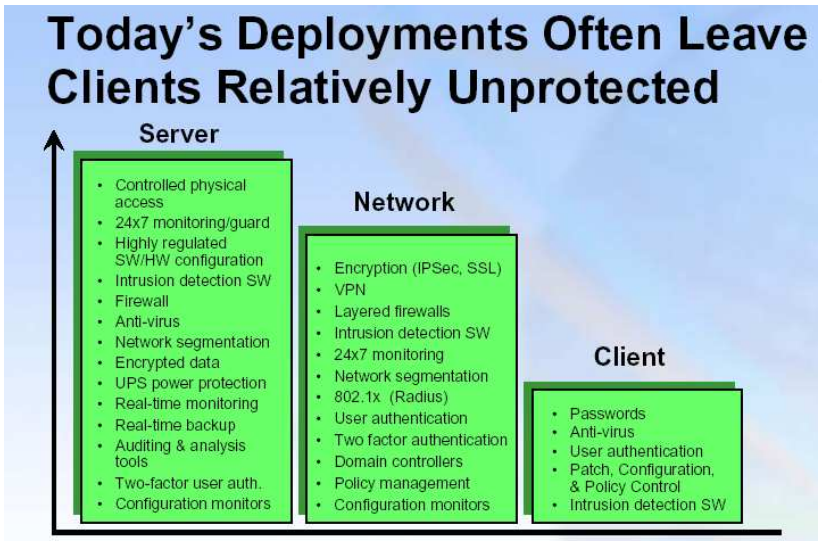
- Trust
  - "An entity can be trusted if it always behaves in the expected manner for the intended purpose."
    - Is the system what it claims to be?
    - Has the system been modified or compromised?
    - Is the system securely storing secrets such that they are protected from adversaries?
- Entity
  - A platform, or an application or service running on a platform.
  - A platform can be a personal computer, PDA, smart phone, etc.
  - A client is a computing platform that can initiate communication with other clients to transfer or share data and resources

2

Trusted Computing

- Traditional Client/Server Architecture
  - Trust is on the server side.
  - Trust is obtained with multi layer protection mechanisms.
    - Access control
    - Firewall
    - Intrusion detection/prevention system
  - There is little trust on client side.
    - Clients are generally lightly protected.
    - Attacks outpacing today's protection models
    - Attack tools readily available
- Information on the client susceptible to software-based attacks.
  - Malicious device drivers and kernels, misconfigured software, virus, Trojan horse, worms, spyware
  - Mismatch between security and high value of data in client platforms

3



4

Trusted Computing

- Evolution of TC
  - **Software alone cannot provide an adequate foundation**
  - Multics system
  - Capability-based computers
  - Trust with security kernel based on military-style security labels
  - Trust in application
    - Totally depends on application
    - With privileged kernel

5

Trusted Computing

- Recent TC activities
  - **Hardware-based root of trust**
  - TCPA/TCG Specifications
  - Hardware: Intel LT, TrustedZone, etc
  - OS/Software: Microsoft NGSCB
  - Provide trusted software execution within a single platform
  - Provide platform-to-platform propagation of trust
  - For open systems
  - Mainly for client side platforms

Trusted Computing Platform Alliance

Trusted Computing Group

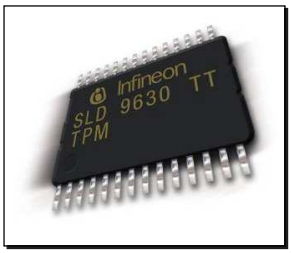
LaGrande Technology

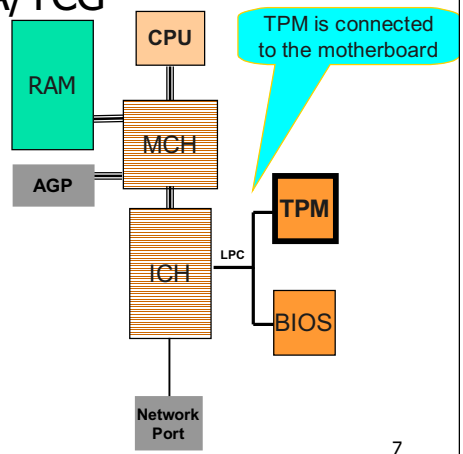
Next Generation Secure Computing Base

6

Trusted Platform Module (TPM)

- Specified by TCPA/TCG
- A chip on board





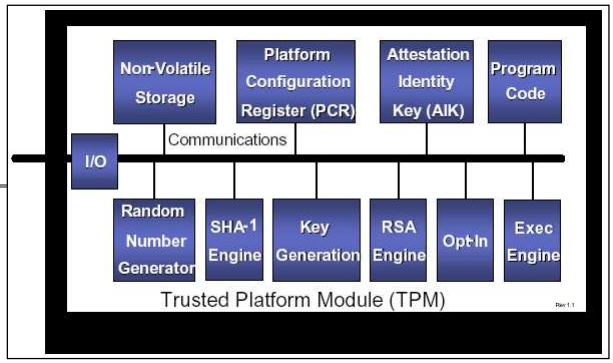
7

TPM

- Basic functions:
  - Integrity measurement, storage, and reporting
    - Ensure that computer reports its configuration parameters in trustworthy manner
  - Cryptographic functions:
    - Random number generation, RSA key generation and public key algorithm, etc.
  - Hardware-based protection of secrets
    - Store root security key inside TPM and never release it
  - Sealed Storage
  - Remote attestation

8

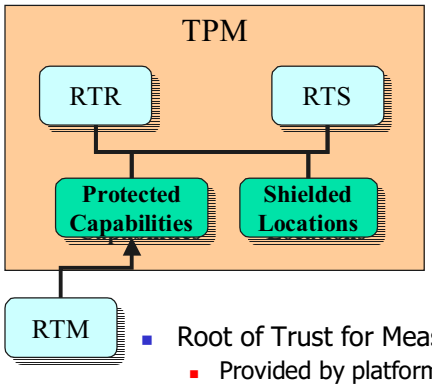
TPM



- Building blocks of a TPM
  - trusted to work properly without additional oversight
  - Trust in these components is derived from good engineering practices, manufacturing process and industry review

9

Functional TPM Diagram

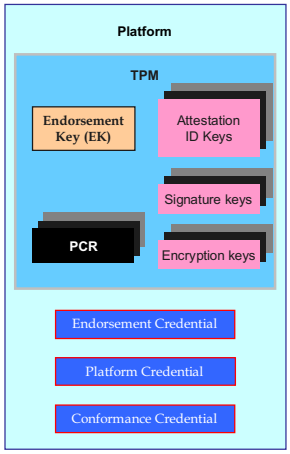


- Root of Trust for Reporting (RTR)
  - Provides cryptographic mechanism to digitally sign TPM state and information held by RTS
- Root of Trust for Storage (RTS)
  - Provides cryptographic mechanism to protect information held outside of the TPM
  - Maintain accurate summary of TPM state
- Root of Trust for Measurement (RTM)
  - Provided by platform to measure platform state
  - Defined by platform specification
- Interaction between RTR and RTS is important TPM capability

10

TPM

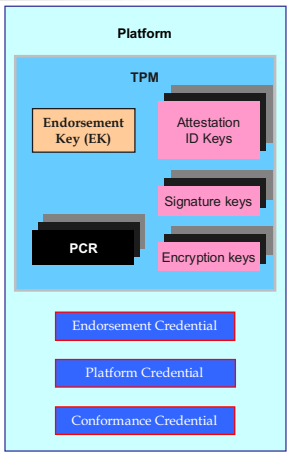
- TPM Credentials:
  - Endorsement credential
    - The EK is a 2048-bit RSA key
    - One per platform
    - Issued by TPM manufacturer
    - Provides attestation that this is a "genuine" TPM
    - Identifies the TPM
    - Provides public key to encrypt the AIKs
  - The EK only participates in two operations
    - Taking TPM ownership
    - Creation of Attestation Identity Keys
  - There are mechanisms to change the EK



11

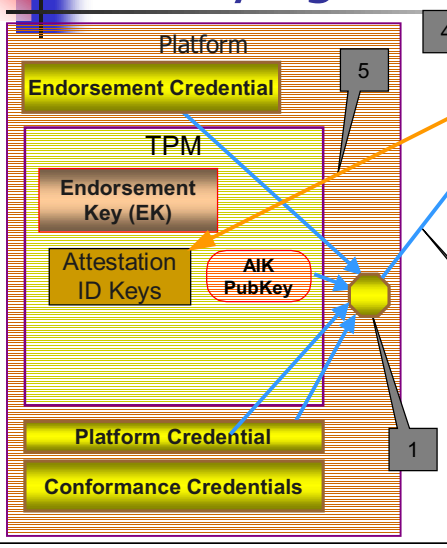
TPM AIK

- Attestation Identity Key (AIK) credentials
  - Many per platform
  - Issued by Privacy CAs (or TPM using EK)
  - Identifies AIKs
  - Provides alias of the platform
  - Provides platform authentication and attestation
- TPM Conformance credential
- Platform credential
- Creation and distribution mechanism is not specified by TCG



12

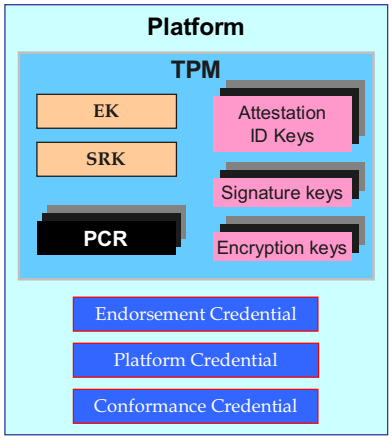
Certifying an AIK



1. Owner bundles into an AIK request:
   - New AIK PubKey
   - Endorsement Cred,
   - Platform Cred,
   - Conformance Creds
2. Owner sends AIK request to P-CA
3. TTP verifies Credentials
4. TTP signs AIK
5. Signed AIK sent to TPM

13

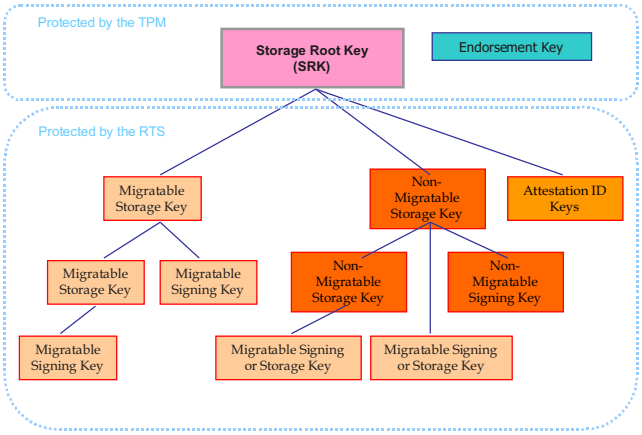
Persistent Keys



- Endorsement Key (EK)
  - Not part of the key hierarchy
- Storage Root Key (SRK)
  - All keys are protected by this key
    - Except EK and AIKs
    - Root of Key Hierarchy
  - Changed on new owner

14

TPM



- Key Hierarchy
  - Non-Migratable Keys: Permanently bound to specific TPM, platform
    - EK, AIK
  - Migratable Keys: Can be exchanged between platforms, follow user
    - Validation key of hardware or software component

15

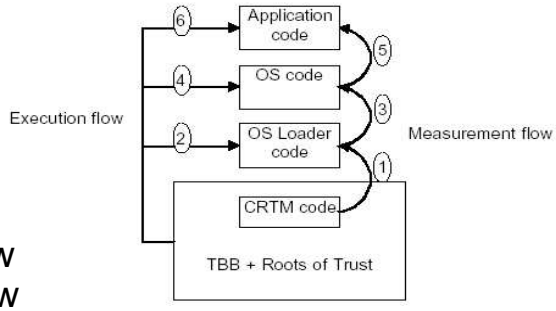
TPM

- Trusted Boot
  - Each boot step is measured and stored
  - Each measurement event consists of:
    - Measured values: integrity, configuration, state, code, etc.
    - Value digests: Hash of measured values
  - Stored Measurement Log (SML): sequences of measured values
  - Value digests are stored in PCRs:
    - *PCR[new]=SHA1{PCR[old] || measured value}*
    - TPM v1.2 requires 24 PCRs
  - Verification requires all SML entries and signed PCRs by an AIK

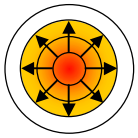
Platform Configuration Registers

16

TPM

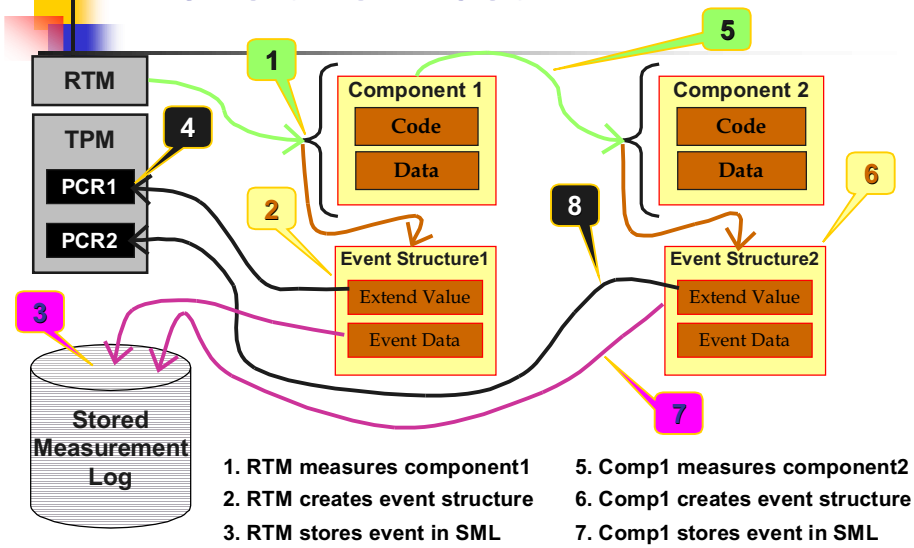


- Measurement flow and execution flow
  - Trust boundary is extended to include measured code.
  - the target code is first measured before execution control is transferred.



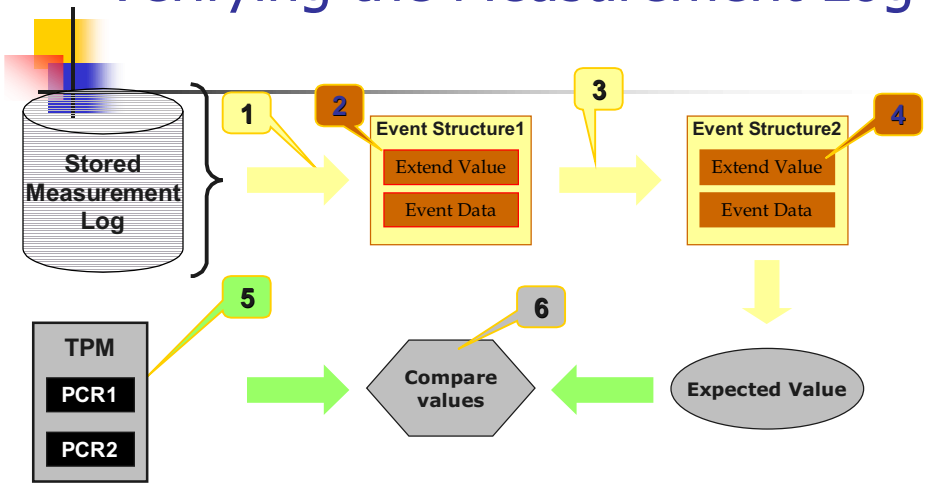
17

Transitive Trust



1. RTM measures component1
2. RTM creates event structure
3. RTM stores event in SML
4. RTM extends PCR with value
5. Comp1 measures component2
6. Comp1 creates event structure
7. Comp1 stores event in SML
8. Comp1 extends PCR with value

Verifying the Measurement Log



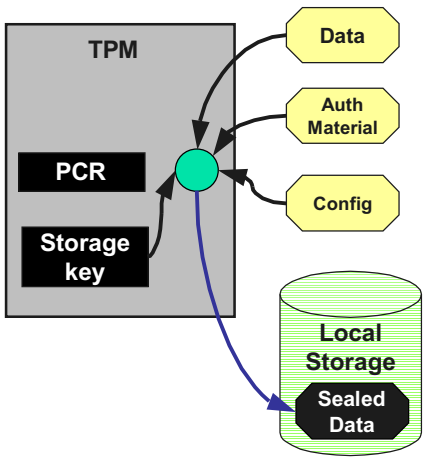
1. Read struct1 from SML
2. Calculate PCR value
3. Read struct2 from SML
4. Calculate PCR value
5. Get PCR value from TPM
6. Compare calculated value with PCR – mismatch indicates problem no information as to what the problem is

TPM

- Sealed Storage:
  - Use one or more PCR values in encryption
  - PCR(s) are part of the sealed message
  - Allows software to explicitly state the environment that can Unseal
  - Sealed Data is inaccessible to any other environment
- Sealed Signing:
  - Signing message with a set of PCR values
  - The platform that signs a message meets specific configuration.
  - Signature is verified by
    - Integrity of the message
    - Trusted PCR values when the signature was generated.

20

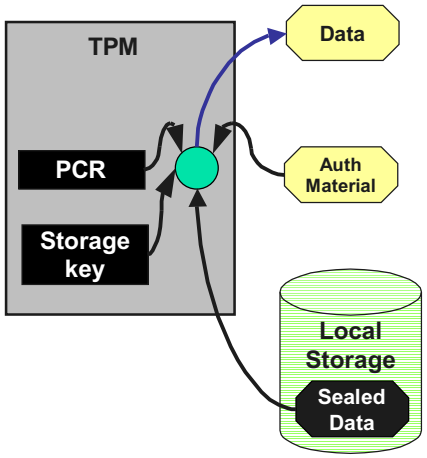
Sealing Data to the TPM



- Send data, authorization value and requested PCR value
  - Not the PCR value at the time of sealing
- TPM encrypts data to create a bound blob
  - Including the request PCR values
- Blob stored outside TPM

21

Unsealing Data



- Load sealed blob into TPM
  - Send in authorization values to use storage key
- TPM decrypts blob
- After decryption TPM validates that current PCR values match requested PCR values in sealed blob
- Data only returned on match

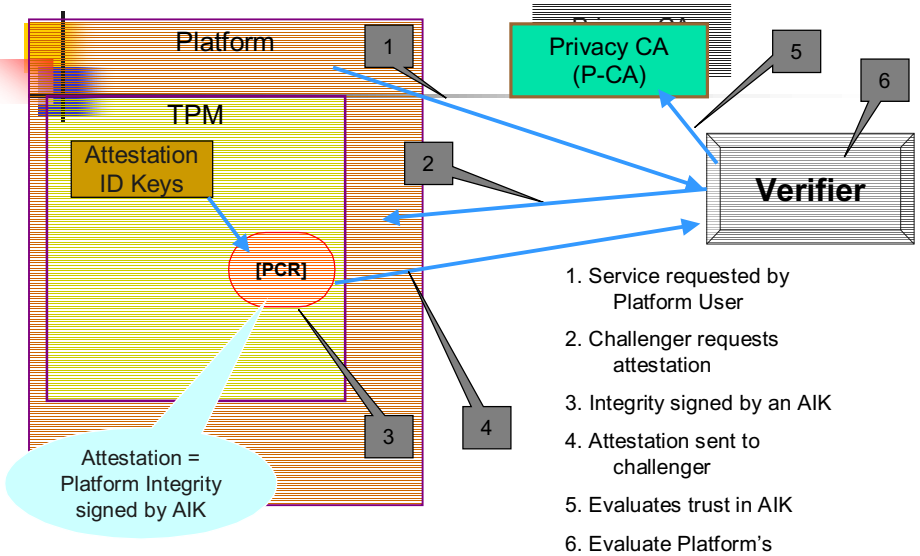
22

TPM

- Integrity reporting: **Attestation**
  - A challenge-response protocol
  - a platform (challenger) sends attestation challenge message to another platform (attestor)
    - Request PCR values
  - One or more PCR values are signed with an AIK protected by the TPM of the attestor and provided to the challenger
    - SML entries are attached.
    - AIK credential is attached.
  - The challenger verifies this attestation
    - Re-generate the hash with values in SML
    - Evaluate credential
    - Compare the signed values with expected values
- Attestation = authentication + integrity

23

Using an AIK



1. Service requested by Platform User
2. Challenger requests attestation
3. Integrity signed by an AIK
4. Attestation sent to challenger
5. Evaluates trust in AIK
6. Evaluate Platform's Integrity

24

Privacy Models

- Don't tell anyone anything
  - Works locally; no distributed trust
- Identity Service Provider (privacy CA)
  - Use a third-party for proof of identity
- Direct Proof
  - Prove identity directly without revealing unique information
- User decides which of these to use and when – can use all or some in combination

25

Identity Service Provider

- A Web-based service that validates identity
  - It gives you a key you can show to third parties to attest to an identity
    - Which identity depends on the service and needs
- Using the ISP model, the MS nexus will:
  - Only release HW key/cert (EK) to certified/trusted parties
    - Privacy CA
  - These parties issue second-level keys
    - Attestation Identity Keys

26

Direct Proof

- Zero Knowledge Proof (ZK)
  - Prove that the system has knowledge of an important something
    - Doesn't reveal the actual piece of knowledge
- Direct Proof (DP)
  - A ZK proof that proves the association between an hardware and AIK
    - Does not reveal the identity of the specific hardware

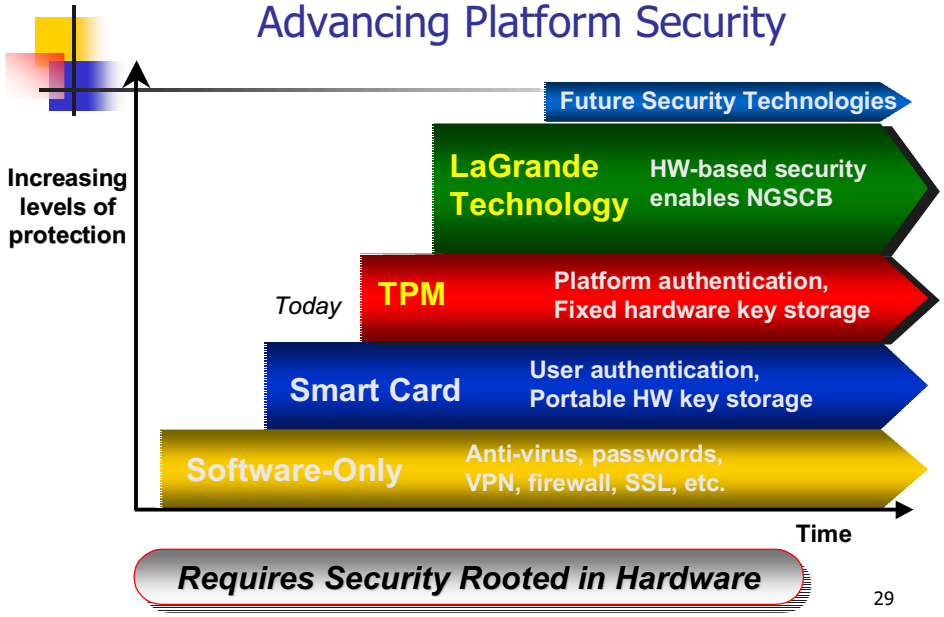
27

Direct Proof Process

- In DP the platform attests to its identity by proving that it has unique "knowledge" which only it can "know"
  - Can be used in a P2P model
    - Two platforms validate each other
    - This would establish a session which uses an identity
      - Which identity depends on the service and needs
- Using the DP model the MS nexus will never release HW key/cert to anyone

28

Intel Safer Computing
Advancing Platform Security



29

LaGrande Technology

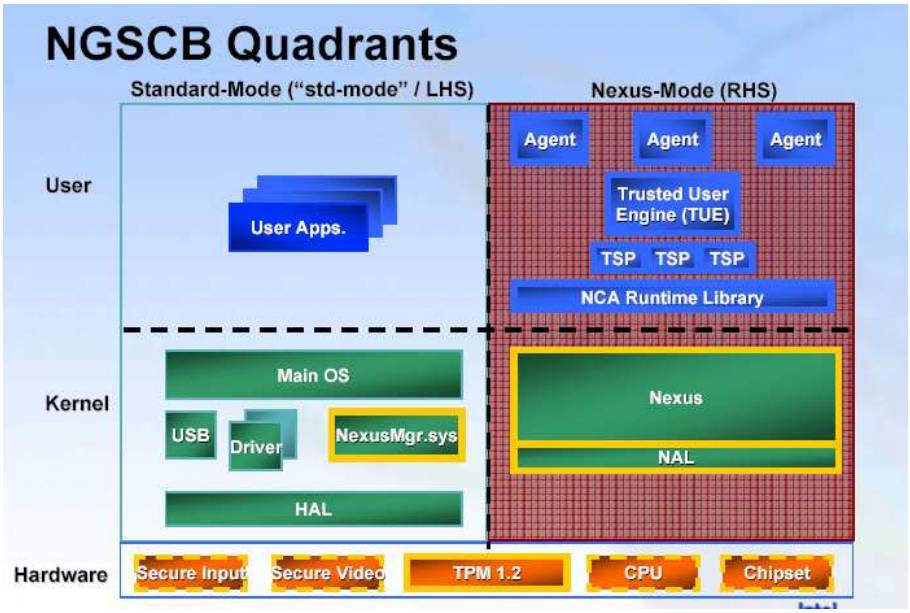
- Extended CPU
  - Enable domain separation
    - Multiple OSs
  - Set policy for protected memory
- Chipset
  - Protected graphics and memory management
- Protected I/O:
  - Trusted channel between keyboard/mouse and trusted software
- TCG TPM v1.2
  - Protect keys
  - Provide platform authentication and attestation

30

LT High-level Functions

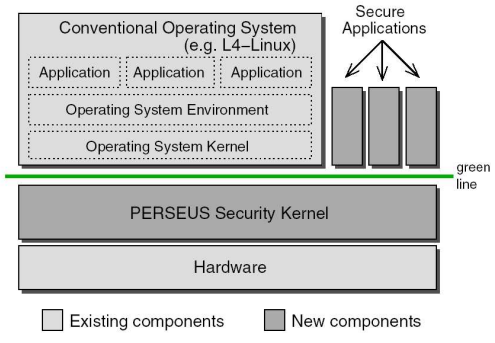
- Protected execution environments
  - Separation of processes, memory pages, and devices
  - Enforced by hardware
- Attestation: Prove platform properties
  - Hardware nature of the platform
  - Current running state and configurations
    - Provided by TPM
- Sealed storage
  - Provided by TPM
- Trusted channels and trusted paths
  - Secure channel between two applications
  - Secure path between application and human
    - between keyboard and keyboard manager
    - between mouse and mouse manager
    - between graphics manager and display adaptor

31



Platform Architectures

- Bright side:
  - Enable client side control after object/content distribution
  - DRM technologies
- Dark side:
  - Platform owner loses partial control of the system
- A platform for multilateral security policies:
  - **Sadeghi and Stüble, Taming Trusted Platforms by Operating System Design, LNCS 2908.**



Related Work

- Secure Boot:
  - Arbaugh et al., Oakland97
  - Boot only signed and verified software
- Secure coprocessors
  - IBM 4758 crypto coprocessor
  - Closed system to run certified and signed software
- Behavior-based attestation
  - Haldar et al. USENIX'04.
  - Trusted language-based VM
- Trusted operating systems
  - SELinux, Trusted Solaris, TrustedBSD
  - Security-enhanced kernel

34

Peer-to-Peer Access Control Architecture Using Trusted Computing Technology

Ravi Sandhu and Xinwen Zhang
George Mason University

SACMAT05, June 1--3, 2005, Stockholm, Sweden

Contributions

- Leverage access control architectures and mechanisms between platforms and users with TC
- Integrate user attributes into TC architecture
- Support a user's ability to roam between platforms by migrating subject identities and attribute certificates.

36

Motivations

- Trust on client platform is needed in modern systems and emerging applications
  - Distributed dissemination control (DCON)
    - *Health records of a patient may be transmitted from a primary physician to a consultant who can access them for some limited period of time and cannot transmit them to anyone else*
  - P2P VOIP application
    - Realtime protection of audio data in a platform
      - conversation is not eavesdropped or illegally recorded.
    - Forward control of audio object (e.g., voice mail)
      - Control the platform and user to forward
  - M-commerce
    - electronic currency between peer platforms
    - payment systems for p2p e-commerce (e.g., micropayment, mobile-payment)

37

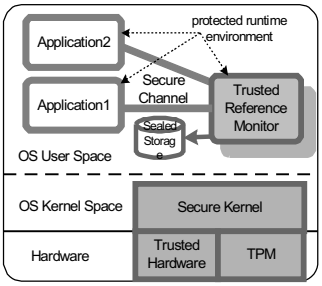
Motivations

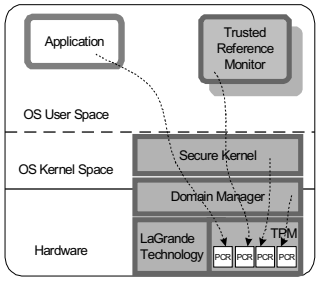
- Need new security model and architecture:
  - Change of trust relation between client and server
    - No centralized and strongly protected server
    - Data located in general client platforms
  - Location of policy enforcement changed:
    - Client-side policy enforcement needs trust
  - Trust of platform and application
    - Dynamic environment
    - Software-based attacks
  - Trusted user authentication and authorization in client platform
  - Trusted path from user to applications and vice versa.
    - Spoofing and ``man-in-the-middle" eavesdropping or modification attacks
    - Trusted input from user to application
    - Trusted output from application to monitor

38

Architecture

- Platform with trusted reference monitor (TRM)
- Assumptions:
  - Tamper resistent hardware
  - A homogeneous environment
    - Each platform is equipped uniformly with necessary TC hardware.





39

Available Credentials

- TPM AIK pair $(PK_{TPM.AIK}, SK_{TPM.AIK})$
  - private key is protected by a TPM with Storage Root Key
  - Public key is certified by a privacy CA.
- TRM key pair (PK_{TRM}, SK_{TRM})
  - The private key is protected by the TPM.
  - The public key is certified by AIK.
- Application key pair (PK_{APP}, SK_{APP})
  - Similar to TRM key pair
- TPM storage key(s)
  - Either the SRK of a TPM, or a key protected by the SRK
  - Protect TRM's credential
  - Protect secrets and policies

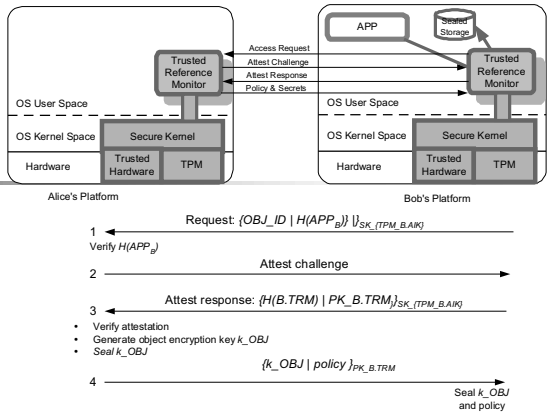
40

Functions of TRM

- *TRM.Seal(H(TRM),x):*
  - seals data *x* by TRM with integrity measurement of *H(TRM).*
  - *x* can only be unsealed under this TRM when the corresponding PCR value is *H(TRM).*
  - In practical a set of PCRs may be included.
- *TRM.GenerateKey(k)*
  - generates a secret key k
- *TRM.Attest(H(TRM), PK_{TRM})*
  - *Return* $\{H(TRM) \,||\, PK_{TRM}\}_{SK\_TPM.AIK}$
  - Attestation response signed by AIK of TPM

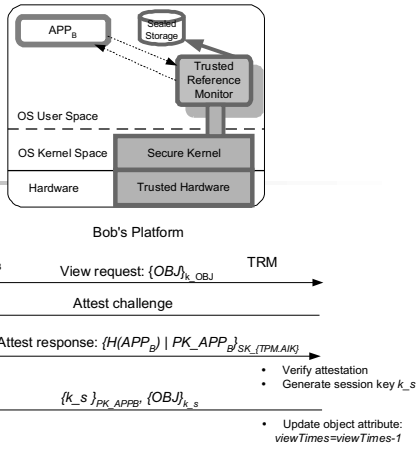
41

Architecture



- Policy and Secret Distribution:
  - Each object has a policy.
  - Object is encrypted with secret key before distribution.
  - Policy specifies what platform and application can access this object
    - migratable or non-migratable policy

Architecture



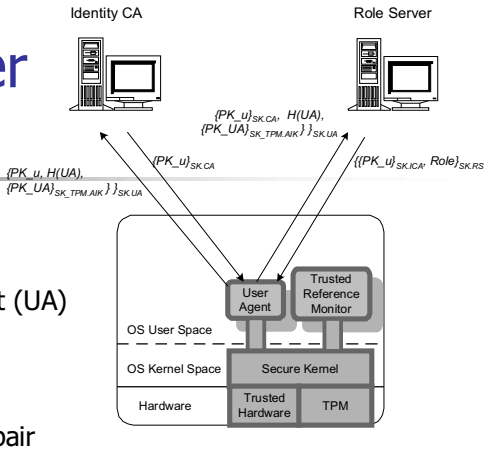
- Policy Enforcement in a client platform
  - Only valid TRM can unseal the policy info and secret.
  - This valid TRM (specified by integrity measurement) can enforce the policy.

Revocation

- Revocation because of
  - Trust revocation of a requesting application
  - Trust revocation of a TRM
  - Trust revocation of a platform
- Two approaches:
  - Push: Object owner sends updated policy to client side
  - Pull: client side check policy update from object owner
  - Both may have delayed revocation
  - Instant revocation needs centralized policy server

44

Support User Attributes

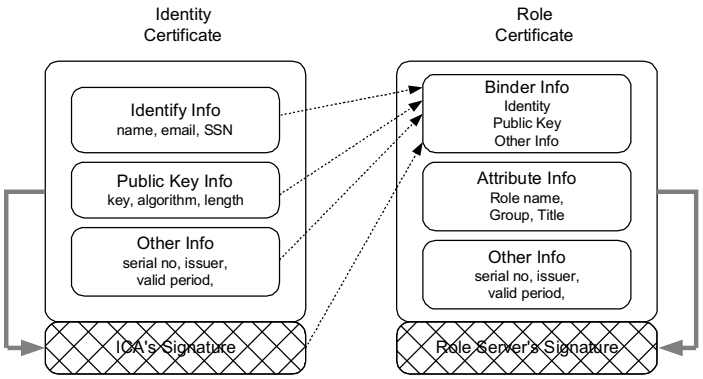


- Each platform has a user agent (UA)
  - Controlled by platform administrator
  - A key pair (PK_{UA}, SK_{UA})
- Each user has an identity key pair (PK_u, SK_u)
  - Migratable key
- Identity and role certificates:

45

Support User Attribute

- Binding of identity and role certificates
  - tightly-coupled binding: by signature
  - loosely-coupled binding: by other components



46

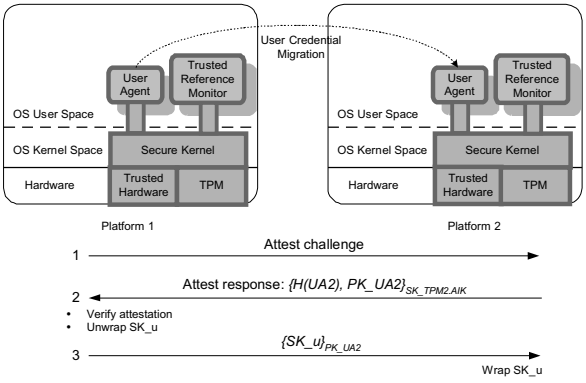
Support User Attribute

- Role-based policy enforcement:
  - TRM sends attestation challenge message to the UA.
  - UA responds with attestation information.
  - If the TRM trusts the running UA, it sends requesting message for role information of the user.
  - The UA sends back the role certificate of the user.
    - UA may submit the proof-of-possession for the corresponding private key of the identity public key
  - Mutual attestation may be needed
    - UA needs to ensure that TRM does not release role information.
    - Role certificate is private information of a user.

47

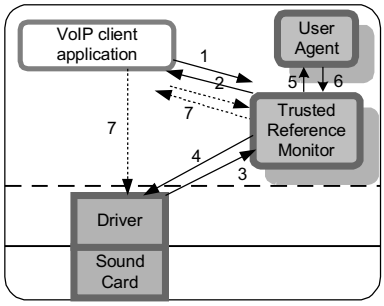
Support User Attribute

- Migration of User Credentials
  - Identity credential and role credential are migratable.
    - Not bounded to specific platform
    - Can be moved or copied between platforms
  - Destination platforms determined by identity owner (user)



Applications

- Secure VOIP:
  - Realtime Protection of Conversation
    - Secure channel between VOIP software and device driver
    - Attestation between TRM and VOIP software
    - Attestation between TRM and UA
    - Attestation between TRM and device driver
  - Secure Storage and Forward of Voice Mail
    - A policy specifying authorized platform and user attribute
    - Similar to DCON



Related Work

- Attestation-based policy enforcement
  - Sailer et al. CCS04
  - Controlled access from client to server by attesting client platform
- P2P content distribution
  - Schecheter et al. 03
  - Admission control by verifying platform and P2P software

50