Reference Monitor and TCG

Dawn Song
dawnsong@cs.berkeley.edu

Review

- **System-call interposition**
- **SFI**
- **VMM**

- **Instances of the same concept: Reference Monitor**

Refernece Monitor

- **A reference monitor is a tamperproof, always-invoked, and small-enough-to-be-fully-tested-and-analyzed module that controls all software access to data objects or devices. The reference monitor verifies the nature of the request against a table of allowable access types for each process on the system.**
  - **System call interposition**
  - **SFI**
  - **VMM**

Reference Monitor

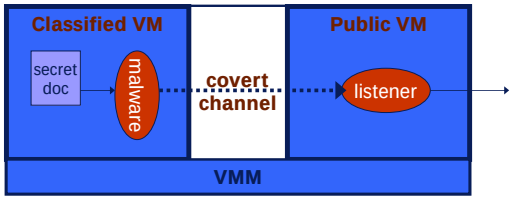
- **Key properties:**
  - Mediates requests **from applications**
    - » **Implements protection policy**
    - » **Enforces isolation and confinement**
  - **Must** always **be invoked:**
    - » **Every application request must be mediated**
  - Tamperproof**:**
    - » **Reference monitor cannot be killed**
    - » **… or if killed, then monitored process is killed too**
  - Small **enough to be analyzed and validated**

Reference Monitor

- **Ensures safety property**
  - **Whether a predicate will hold for a given state**
- **Not liveness property**
  - **Whether a predicate will hold some time in the future**
- **Not information flow property (non-interference)**

Example: covert channels

- **Covert channel: unintended communication channel between isolated components**
  - **Can be used to leak classified data from secure component to public component**



An example covert channel

- Both VMs use the same underlying hardware
- To send a bit b ∈ {0,1} malware does:
  - b= 1: at 1:30.00am do CPU intensive calculation
  - b= 0: at 1:30.00am do nothing
- At 1:30.00am listener does a CPU intensive calculation and measures completion time
  - Now b = 1 ⇔ completion-time > threshold
- Many covert channel exist in running system:
  - File lock status, cache contents, interrupts, …
  - Very difficult to eliminate

TCB

- The trusted computing base (TCB) of a computer system is the set of all hardware, firmware, and/or software components that are critical to its security, in the sense that bugs occurring inside the TCB might jeopardize the security properties of the entire system
- Example: on-line banking application
- Security design principle: minimize TCB
- Security enforcement:
  - Ensure TCB is trust-worthy
- Note the difference btw *trusted* and *trustworthy*

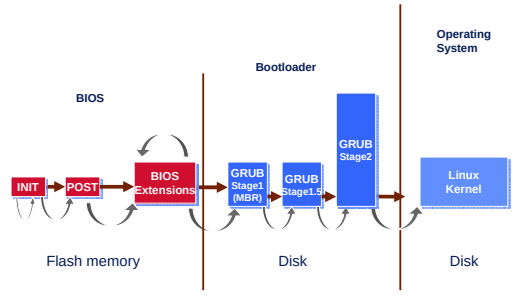
Trusted Path

- Mechanism that provides confidence that the user is communicating with what the user intended to communicate with, ensuring that attackers can't intercept or modify whatever information is being communicated.
- Example: fake log-in program
  - Solution: ctrl+alt+delete guarantees correct log-in program is executed

Trusted Computing Overview

- Goals
  - Make computers a trustworthy execution platform
  - Prove to external entity what software is executing
- Challenges
  - Software vulnerabilities in OS and apps
  - Malware compromises systems
  - Hardware attacks by local user/attacker

Bootstrapping a Typical PC



Trustworthy Booting

- Different flavors of booting
  - "Untrusted booting": no verification, no security guarantees
    - This is how current PCs boot
  - Secure boot: every layer verifies correctness of next layer before passing control to it
    - E.g., BIOS verifies signature of boot loader before passing control to it
  - Trusted/authenticated boot: establishes proof on what software has loaded
- Secure boot and trusted boot assume core root of trust: correctness of BIOS bootloader

Secure Boot Integrity Guarantees

- **Integrity of a layer can only be guaranteed if**
  1. **Base layer is immutable**
  2. **Integrity of the lower layer is verified**
  3. **Transition to higher layer only occurs after valid verification**
- **Secure boot ensures that operating system that is bootstrapped is based on untampered foundation (*integrity guarantee*)**
- **Not a problem in early days when firmware was stored on write-protected EPROMs, nowadays writeable FLASH memory is used**

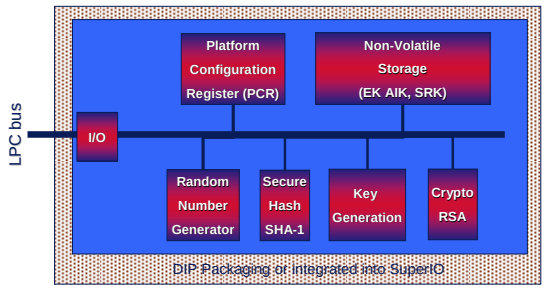
Trusted Computing Group (TCG)

- **TCG (formerly known as TCPA) goal is to add secure platform primitives to each client (now the focus is also on servers, cell phones, PDAs, etc.)**
- **Industry consortium by AMD, IBM, Intel, HP, Microsoft, …**
- **These secure platform primitives include**
  - **Platform integrity measurements**
  - **Measurement attestation**
  - **Protected storage**
  - **Sealed storage**
- **These can be used to provide trusted boot (as opposed to secure boot)**
- **Provides attestation, which enables an external verifier to check integrity of software running on host**
  - **Goal: ensure absence of malware; detect spyware, viruses, etc.**

TCG Overview (1)

- **Main goals: enable trusted boot and remote attestation**
- **TPM chip provides/contains**
  - **Tamper-resistant enclosure for trusted information**
  - **Secure storage for private key K^{-1}_{TPM}**
  - **Manufacturer certificate, for example $\{K_{TPM}\}_{K^{-1}_{IBM}}$**
  - **Immutable storage for software integrity measurements**
  - **Digital signature capability**

TCG Trusted Platform Module (TPM)



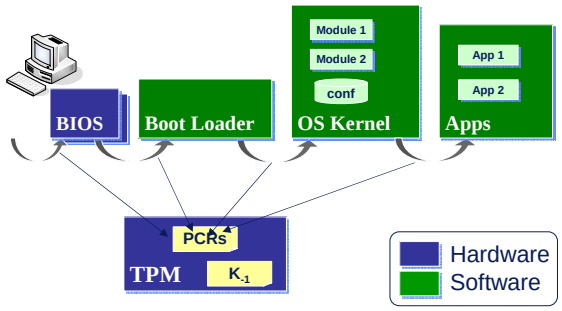
TCG Overview (2)

- **Setting**
  - **External entity A wants to verify correctness of software executing on platform B**
  - **Assume that A trusts manufacturer's public key K_{IBM}**
  - **B is equipped with TPM chip and performed trusted boot process**
- **TCG trusted boot process on B (simplified!)**
  - **BIOS loads OS loads App**
  - **Assume BIOS is Core Root of Trust**
  - **BIOS loads OS, computes H(OS), sends H(OS) to TPM to extend register PCR0, executes OS**
  - **OS loads App, computes H(App), sends H(App) to TPM to extend register PCR1, executes App**

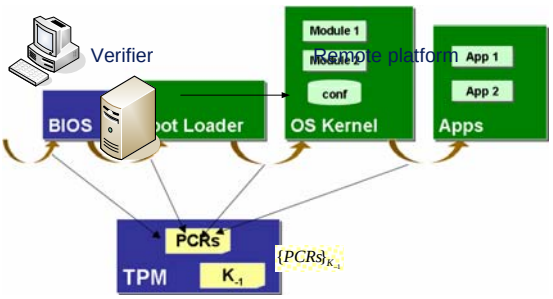
TCG Overview (3)

- **A wants to attest to B's software**
  - **A → B: attestation request, nonce**
  - **B: attestation request & nonce sent to TPM, TPM computes signature of PCRs and nonce**
  - **B → A: $\{K_{TPM}\}_{K^{-1}_{IBM}}$, $\{PCR0, PCR1, nonce\}_{K^{-1}_{TPM}}$**
  - **A verifies certificate, signature and correctness of PCR0 and PCR1**
- **If all checks successful, A trusts that B is executing correct OS and App**

Basic TCG-Style Attestation



Basic TCG-Style Attestation



Trusted Computing Key Components (I)

- **Endorsement Key**
  - Private/public key pair generated on-chip at manufacture time
  - Private key never leaves chip
- **Secure I/O (Trusted path)**
  - a protected path between the computer user and the software with which they believe they are interacting
  - TPM can check software drivers used for I/O have not been tampered with

Trusted Computing Key Components (II)

- **Protected storage**
  - Provide secure storage, not accessible by OS
- **Sealed storage**
  - protects private information by binding it to platform configuration information including the software and hardware being used
  - Data can be read only by the specified combination of software and hardware
- **Remote attestation**
  - Remotely attesting what software is running on the computer

Applications of Trusted Computing

- **Preventing cheating in on-line gaming**
  - Players modify game in order to cheat
  - Remote attestation can verify all players connected to game server are running an unmodified copy
- **Verification of remote computation for grid-computing**
- **Digital Rights Management**
  - Downloading a music file
  - Remote attestation
    - » Refuse to play except on specific music player
      - Windows Media Player
    - » Sealer storage prevent opening file from another player

Problems with Integrity Measurements

- **How do you handle all the different firmware versions, patches, kernel builds? What does a PCR mean in this context?**
- **Integrity measurements are done at load-time not at run-time**
  - Time-of-check-time-of-use (TOCTOU) problem

Policy Issues

- Can TPMs be used for malicious purposes?
  - Could software vendor control all applications that are executed?
  - Could content provider have total control over how we use data? Fair use?
- TPMs can enhance security of computer systems
  - Should government require use of TPMs?

TCG Controversy

- TCG is considered very controversial because it potentially allows content providers to control clients (DRM enforcement)
- This takes away the freedom of the user to use the system as it sees fit (it can be used to lock-out GPL software)
- A privacy concern is that TCG can be used to track users

Conclusion

- Reference monitor
- Trusted computing
  - TCB
  - Trusted path
  - Secure boot/Trusted boot
  - Remote attestation
  - Trusted computing key components

Questionaire

- Pls provide as much feedback as you can