Challenges for Trusted Computing

Ahmad-Reza Sadeghi

Horst Görtz Institute for It security

Ruhr-University Bochum

sadeghi@crypto.rub.de

CHES, Yokohama 2006

hgi

Horst-Görtz Institut für IT Sicherheit

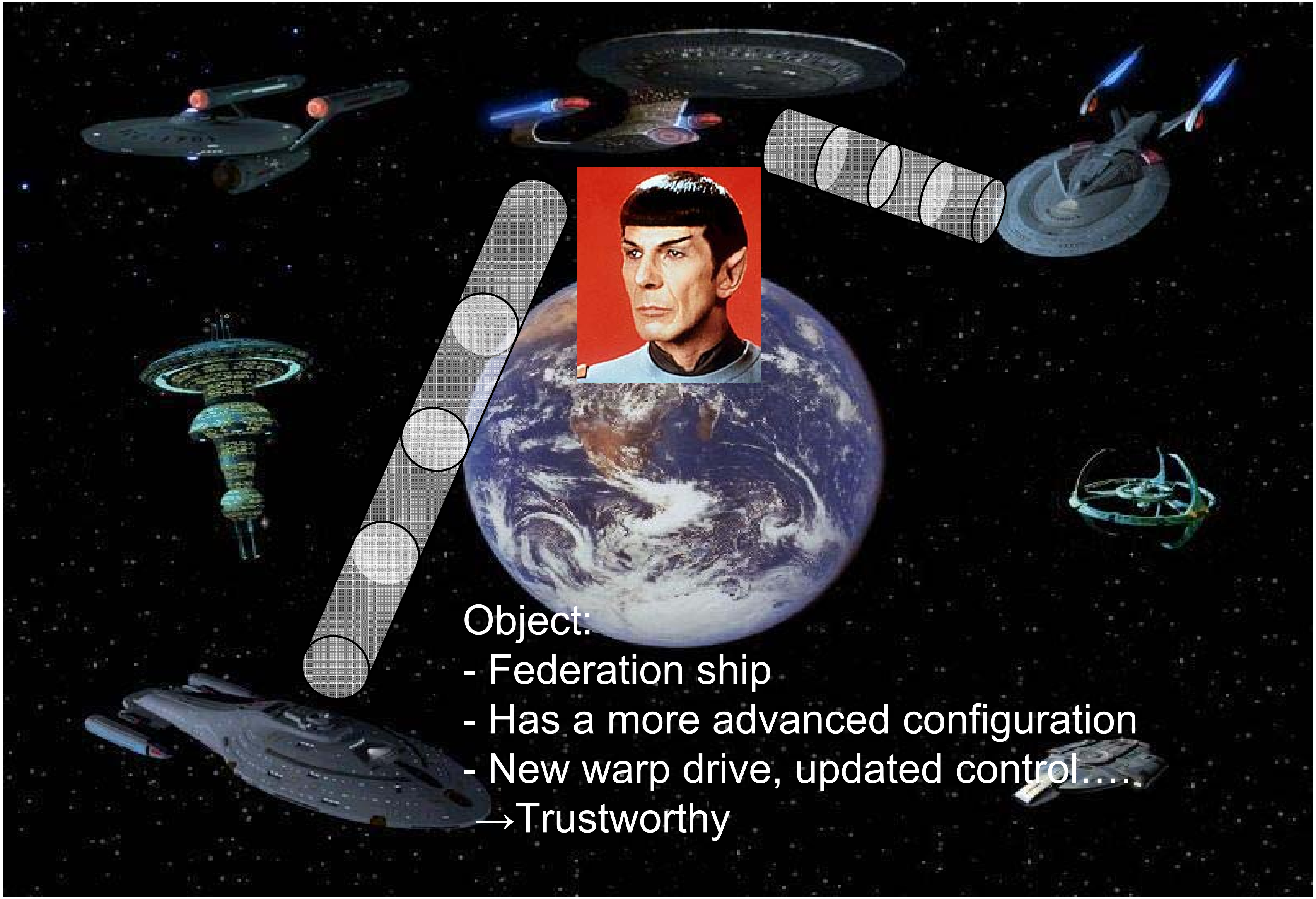
Content

- **Motivation**
  - **Trust Issues and Vocabulary**
  - **Complications in Distributed Application**
- **Towards Trustworthy Computing Platforms**
  - **Objectives and Primary Goals**
  - **Desired Primitives and the Need for Secure Hardware and Software**
- **Trusted Computing Group (TCG) Approach**
- **Security Architectures Based on Virtualization**
- **Selected Research and Development Projects**
- **Reactions to the Trusted Computing Group**
  - **Concerns, open source, law and politics**
- **Some Technical Challenges**
- **Summary and Outlook**

Motivation

- o How do we define „trustworthiness“ in a distributed open IT environment?
- o How can we determine/verify/measure it?
- o How could common computing platforms support such functionality and what are the consequences?

Future….


Object:
- Federation ship
- Has a more advanced configuration
- New warp drive, updated control….
→Trustworthy

A Memo .....

- ”Trustworthy Computing is the highest priority for all the work we are doing. We must lead the industry to a whole new level of trustworthiness in computing”
- “…. Trustworthy Computing is computing that is as available, reliable and secure as electricity, water services and telephony.”
- “Our software should be so fundamentally secure that customers never even worry about it.”
- “No Trustworthy Computing platform exists today. It is only in the context of the basic redesign we have done around”
- ”Keep our customers' trust at every level -- from the way we develop software, to our support efforts, to our operational and business practices. As software has become ever more complex, interdependent and interconnected, our reputation as a company has in turn become more vulnerable.”
- “Key aspects are availability, security, and privacy“
- Trustworthiness is a much broader concept than security, and winning our customers' trust involves more than just fixing bugs

Bill Gates’ email on full-time employees of MS, January 2002

Trust Issues and Vocabulary (1)

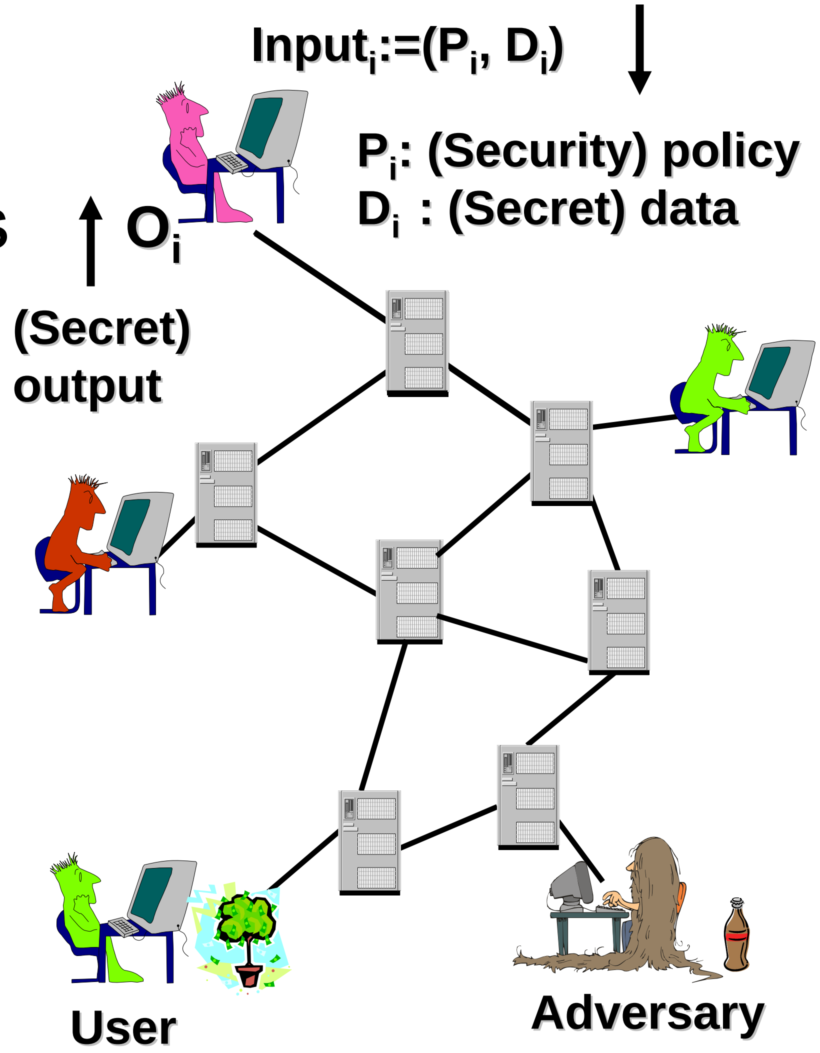
- **Trust:** Complicated notion studied and debated in different areas (social-sciences, philosophy, psychology, computer science,…)
- In **Social Sciences**, trust is
  - a *psychological state* comprising the intention to accept vulnerability based upon positive expectations of the intentions or behavior of another [RoSiBuCa98]
  - a *mechanism* to reduce social complexity (how we think about the world) [Luhm1979]
  - an *action* that involves the voluntary placement of resources (physical, financial, intellectual, or temporal) at the disposal of the trustee with no real commitment from the trustee [Cole1990]
  - temporal and has risk aspects

Trust Issues and Vocabulary (2)

- In **IT security literature**
  - a **Trusted System** or component is one whose failure can break the security policy [Ande2001]
    - Number of trusted components should be minimized
  - **Trustworthiness** is assurance that a system or a component will perform as expected [AvLaLaRa2004]
    - Corresponds to "Trusted" as defined by Trusted Computing Group (TCG)

Complications in Distributed Applications

- Multiple parties involved
- Provide (require) services (resources)
- Have different (possibly conflicting) interests (policies)
- Typically distrust each other (minimal TCB)
  - TCB (Trusted Computing Base)

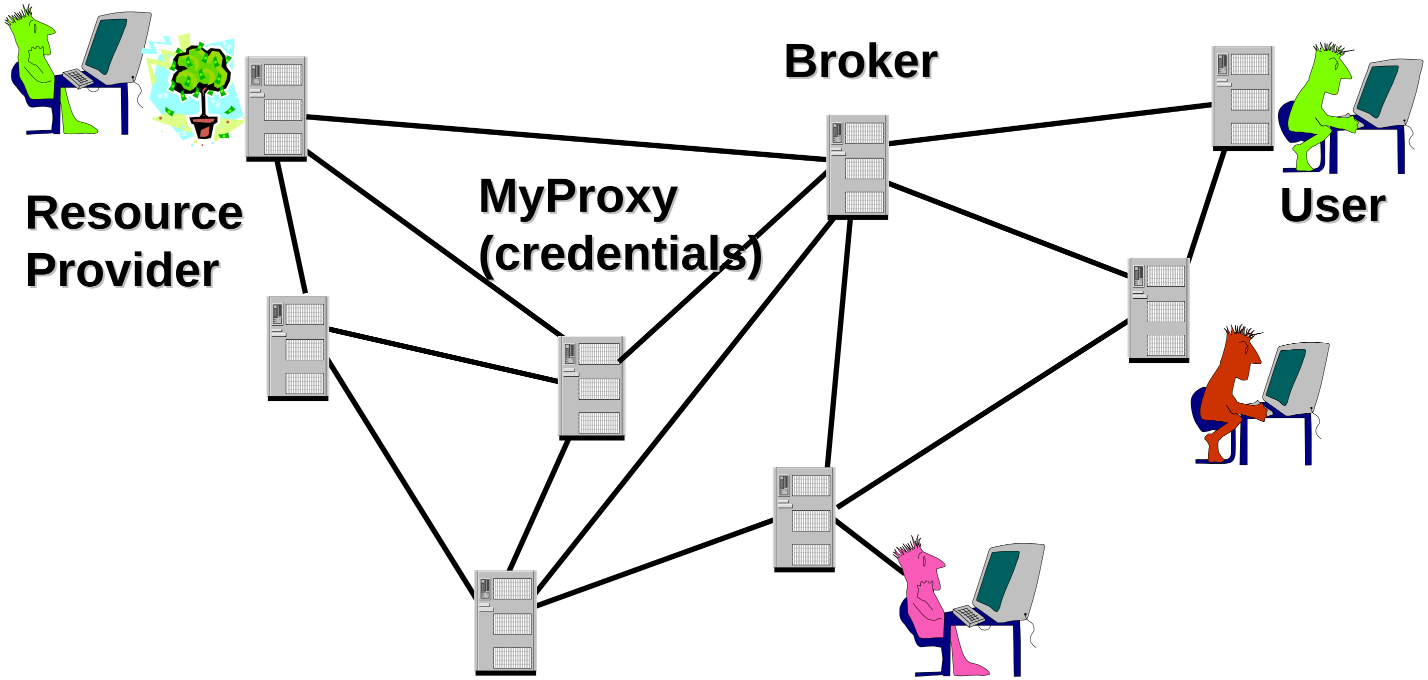


Application Scenarios

- E-Services
  - Government (e.g., e-Voting integrity)
  - Health (confidentiality of sensitive medical records)
  - Commerce ((Non)-enforceability of digital signature)
- Rights and Document Management
  - Enterprise
    - Controlled usage and distribution in Supply Chains
  - Fair use
    - Private copies
    - Copies among different platform types allowed
  - First sale
    - Transfer of digital content
- Outsourcing of services
- Next generation mobile devices

Example: Grid Computing

Model



- **Main parties (simplified): resource providers (RP) and users (U)**
- **In practice more parties: Middleware provider, application provider**
- **Problem: User-provider trust asymmetry** [LoRaSaScSt2006, MaJiMa2006]
    - Grid users forced to place (often, unjustifiable) trust on providers
    - Security measures often assume Grid user as potential adversary
- **Currently used measures**
    - Contracts, standard authentication and authorization mechanisms

Requirements

- **Functional**
  - Sharing resources among different Grid jobs on one platform
  - Interoperability
  - Auditing
  - Delegation and single sign on
  - Accounting and billing
- **Security**
  - Confidentiality and integrity of data
  - Privacy (regarding underlying platform)
  - Authentication
  - Authorization
- **Availability and correctness**
  - Fail-safe short and long term preservation of users data

Towards Trustworthy Platforms

Objectives

- **Multilateral Security** [Rann1994]
  - Considers different and possibly conflicting security requirements of different parties and strives to balance these requirements
  - Refers to (classical) security goals (confidentiality, integrity and availability)
  - Typical conflict occurs between the wish for privacy and the interest in cooperation
- **Problems**
  - Insufficient protection in SW and HW of existing computing platforms
    - Malicious code (viruses, Trojan horses, …)
    - DMA (Direct Memory Access)
    - No secure storage
  - Main reasons
    - High complexity and poor fault isolation of operating systems
    - Lack of functional and protection mechanisms in hardware
    - Security unawareness of users or security measures still not useable enough
- **Main Role of Trusted Computing** [Kuhl2003, KuGe2003]
  - Enable the reasoning about the "trustworthiness" of own and other's IT system (reporting their state)
  - … in contractual sense

Primary Goals

- Improve security of computing platforms
- Reuse existing modules
  - e.g., GUI, common OS
- Applicable for different OS
  - No monopoly, space for innovation (small and mid-sized companies)
- Open architecture
  - Use open standards and open source components
  - Trustworthiness/costs/reliability/compatibility
- Efficient portability
- Allow realization of new applications/business models
  - Providing multilateral security needed for underlying applications
  - Avoiding potential misuse of trusted computing functionalities
  - Based on different sets of assumptions and trust relations

Basic Desired Primitives

- **Integrity verification (Attestation)**
  - Allows a computing platform to export verifiable information about its properties (e.g., identity and initial state)
  - Comes from the requirement of assuring the executing image and environment of an application located on a remote computing platform
- **Sealed/Secure Storage** allows applications
  - to persist data securely between executions using traditional untrusted storage like hard drives
  - To encrypt data and assured to be the only capable of decrypting it
- **Strong process isolation**
  - Assured (memory space) separation between processes
  - Prevents a process from reading or modifying another process's memory
- **Secure I/O**
  - Allows application to assure the end-points of input and output operations
  - A user can be assured to securely interact with the intended application

Need for Secure Hardware and Software

- **Hardware**
  - Even a secure operating system cannot verify its own integrity (another party is needed)
  - Secure storage
  - DMA control
    - Isolation of security-critical programs
  - Hardware-based random numbers
    - Fundamental to cryptography
- **Software (Operating Systems)**
  - Hardening, e.g., SE Linux [LoSm2001]
    - Still too complex and large TCB (Trusted Computing Base)
  - Complete new design
    - e.g., Trusted Mach, EROS (Extremely Reliable Operating System) [TrustedMach1991, Shap1999]
    - Compatibility problem, less market acceptance
  - Secure Virtual Machine Monitors (e.g., [Sailer et al 2005])
    - Allow reuse of legacy software

Trusted Computing Group (TCG) Approach – A Short Introduction

Background

- **TCG (Trusted computing Group)**
  - Consortium 136 enterprises (AMD, HP, IBM, Infineon, Intel, Microsoft, STM, ...)
  - Claimed role: "…to develop, define and promote open, vendor-neutral industry specification for trusted computing. These include hardware building blocks and software interface specifications across multiple platforms and operating environments….. " [TCG]
- **Basic idea**
  - Assurance of a limited set of immutable cryptographic functionalities based on which a larger set of security functions can be provided
  - Minimum tamper-resistant assumptions
- **Uses the concept of roots/chain of trust [ArFaSm1997, Itoi et al 2001]**
  - Entities (functions) trusted to function correctly without external oversight
  - Lower layer verifies the integrity of higher levels before booting them
- **Specified several specifications**
  - Trusted Platform Module (TPM)
    - Set of cryptographic functionalities and features
  - Trusted Software Stack (TSS)
    - TSS is a software specification that provides a standard API for accessing the functions of the TPM (resource management of TPM, ensuring synchronized access)
    - Open source implementation [TrouSerS]
- **Different working groups**
  - e.g., TPM/TSS, Infrastructure, Mobile,...

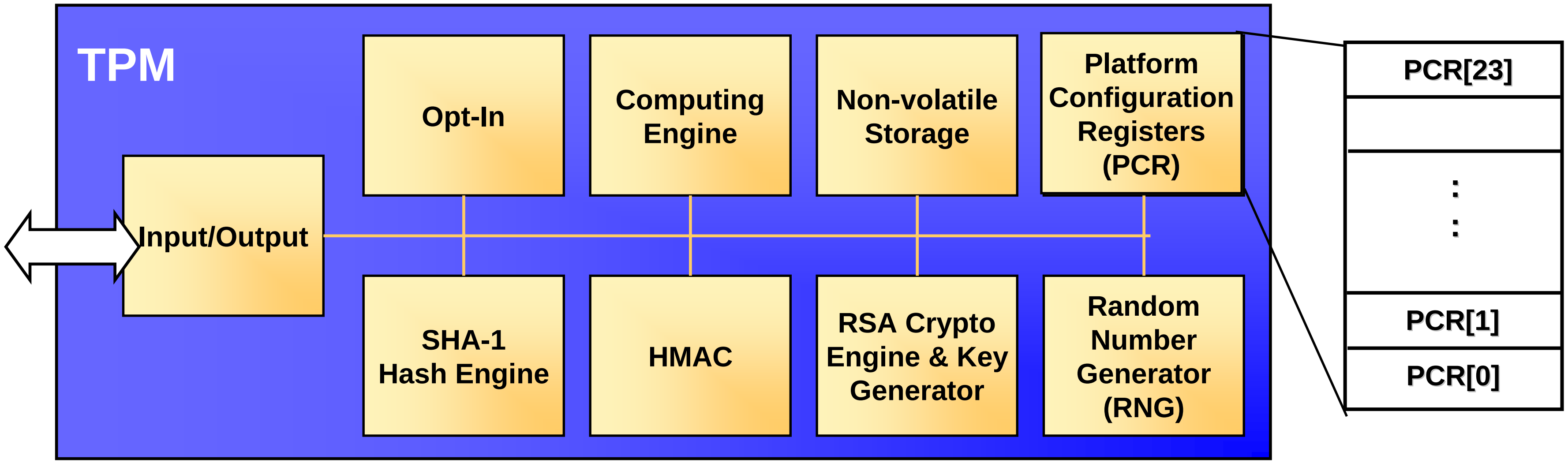
Model

- **Main objectives**
  - Integrity and confidentiality of certain data (e.g., cryptographic keys)
- **Trust model**
  - Roots of Trust for Measurements (RTM): Process that measures platforms integrity
  - Roots of Trust for Storage (RTS): A logical entity capable of maintaining values generated by the RTM
  - Roots of Trust for Reporting (RTR): A mechanism for correctly exporting the values held in RTM to any interested party
  - Minimal essential roots of trust are RTM and TPM
- **Adversary model**
  - Specifications focus on software attacks
- **Remarks**
  - According to TCG an entity can be trusted if it always behaves in the expected manner for the intended purpose

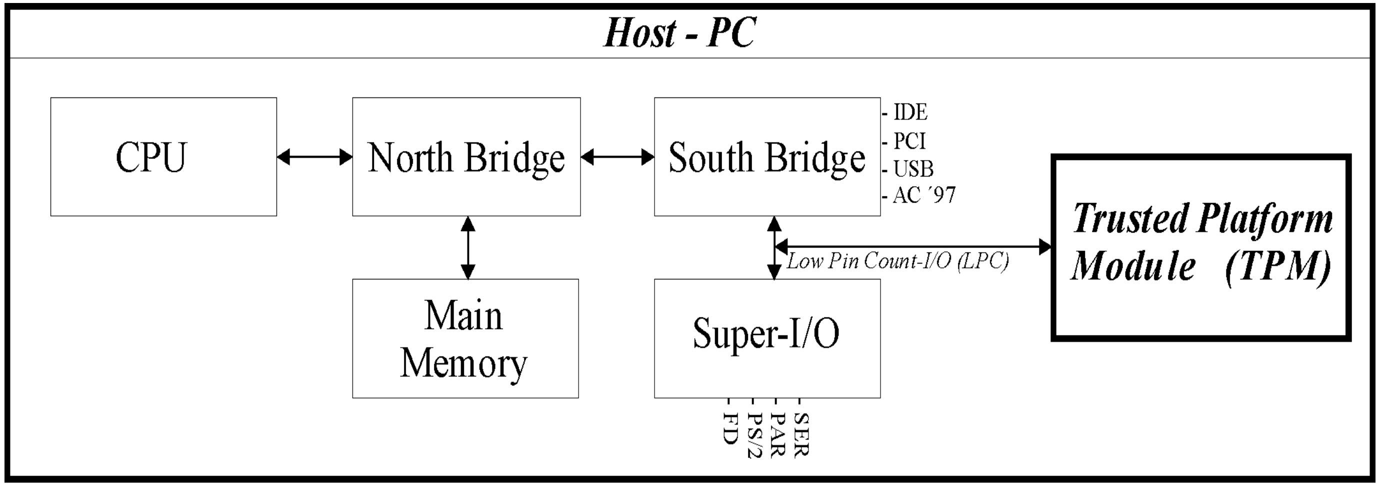
Core TCG Components and Functionalities

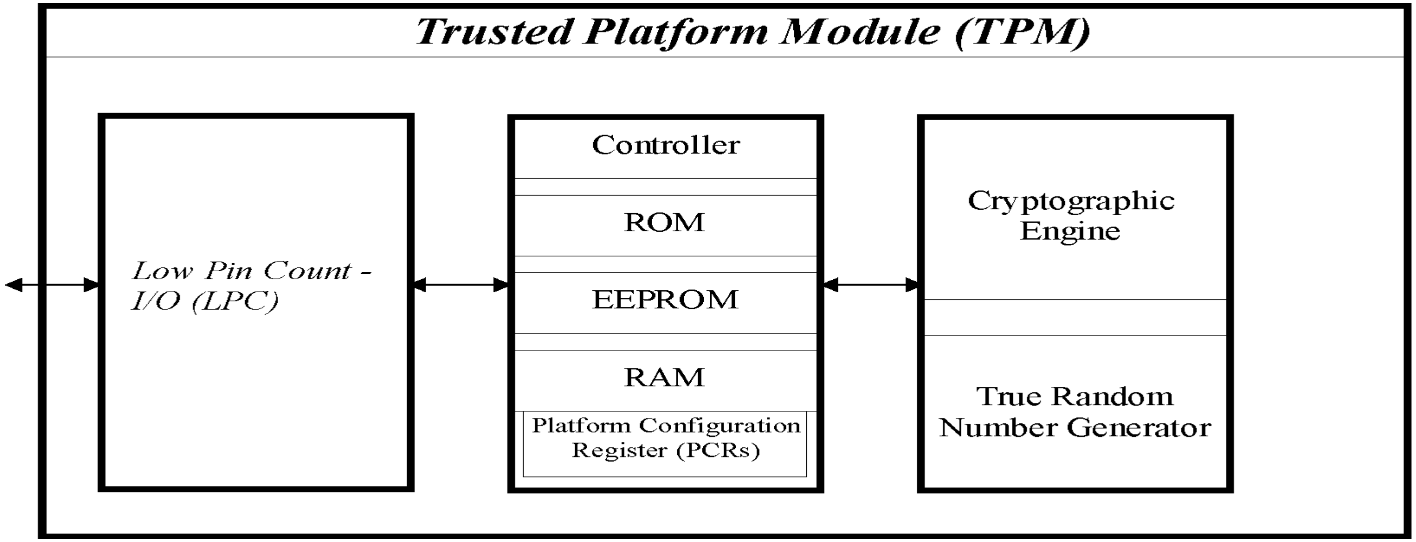
Trusted Platform Module (TPM)

- Current implementation is a dedicated hardware chip on main board
- Two versions 1.1b and 1.2 [TPM2002, TPM2003]
- Passive component
- Manufacturer (Atmel, Infenion, Sinosun, STM,…)



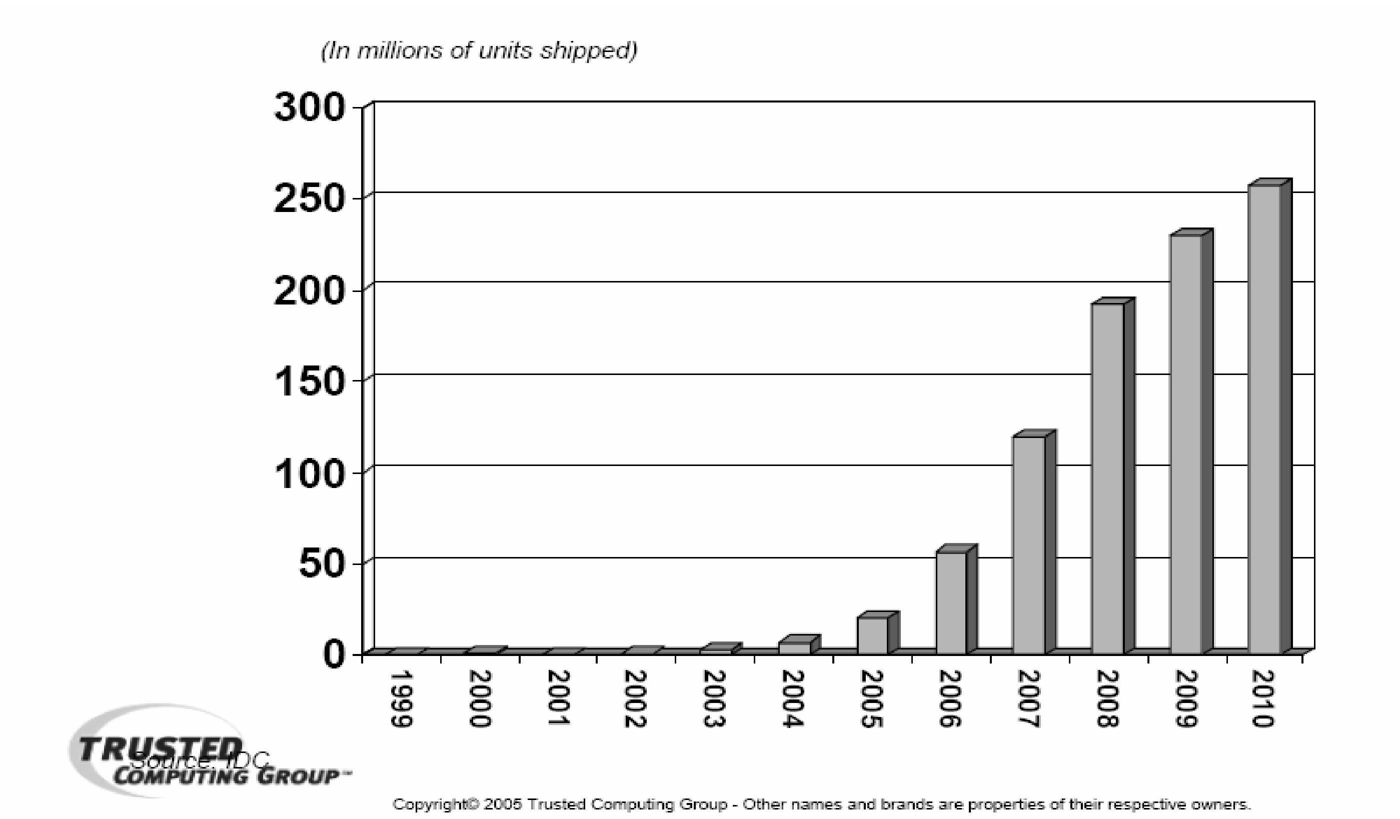
Details


Host - PC
CPU
North Bridge
South Bridge
- IDE
- PCI
- USB
- AC ´97
Trusted Platform Module (TPM)
Low Pin Count-I/O (LPC)
Main Memory
Super-I/O
FD
PS/2
PAR
SER


Trusted Platform Module (TPM)
Low Pin Count - I/O (LPC)
Controller
ROM
EEPROM
RAM
Platform Configuration Register (PCRs)
Cryptographic Engine
True Random Number Generator

TPM Forecast

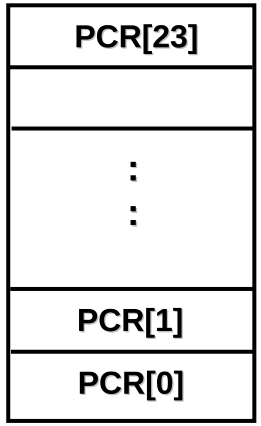
- Many vendors ship platforms equipped with TPM e.g., IBM, HP, Siemens-Fujitsu (see [TPMMatrix2006])
- Microsoft' Vista [Vista2006] uses TPM functionalities for secure setup (requires TPM v1.2 [TPM2003])



TPM Features

- **Hardware-based random number generators**
- **Cryptographic functions**
  - Hash (SHA-1), signature, encryption (RSA), key generation
- **Platform Configuration Registers (PCR)**
  - Storage for (integrity) measurements
  - Metric for measurements is computing hash values
  - PCR values are so-called extensions

    $\text{extend}(PCR_N, \text{Input}) = \text{SHA1}(PCR_N \,||\, \text{Input})$



- **Sealing/Binding**
  - Binding data to TPM state represented by a subset of PCRs
  - S_i current state, S_0 initial state
    - $[\text{Data}]_{S_0}^{PK} \leftarrow \text{Seal}(\text{State},PK,\text{Data})$
    - $\text{Data}=\text{Unseal}([\text{Data}]_{S_0}^{PK}) \Leftrightarrow$

      $[\text{Data}]_{S_0}^{PK} \leftarrow \text{Seal}(\text{State},PK,\text{Data}) \wedge (S_i = S_0))$

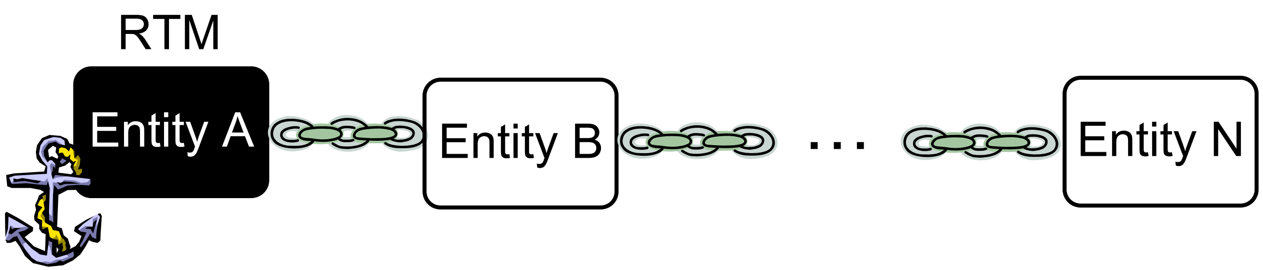
TPM Features: Keys

- **Endorsement Key (EK)**
  - uniquely identifies a TPM (manufacturer may provide certificate for EK)
- **Attestation Identity Key (AIK)**
  - created by TPM, certified by CA, primarily used to sign subset of PCRs
- **Storage keys**
  - used to encrypt data outside TPM (e.g., other keys of TPM)
- **Storage Root Key (RTS)**
  - uniquely created inside TPM, private part in TPM
  - used to encrypt all other keys created by the TPM
- **Migratable and non-migratable keys**
- **Certified-migratable keys**
  - decide to delegate migration upon creation of keys

Integrity Measurement

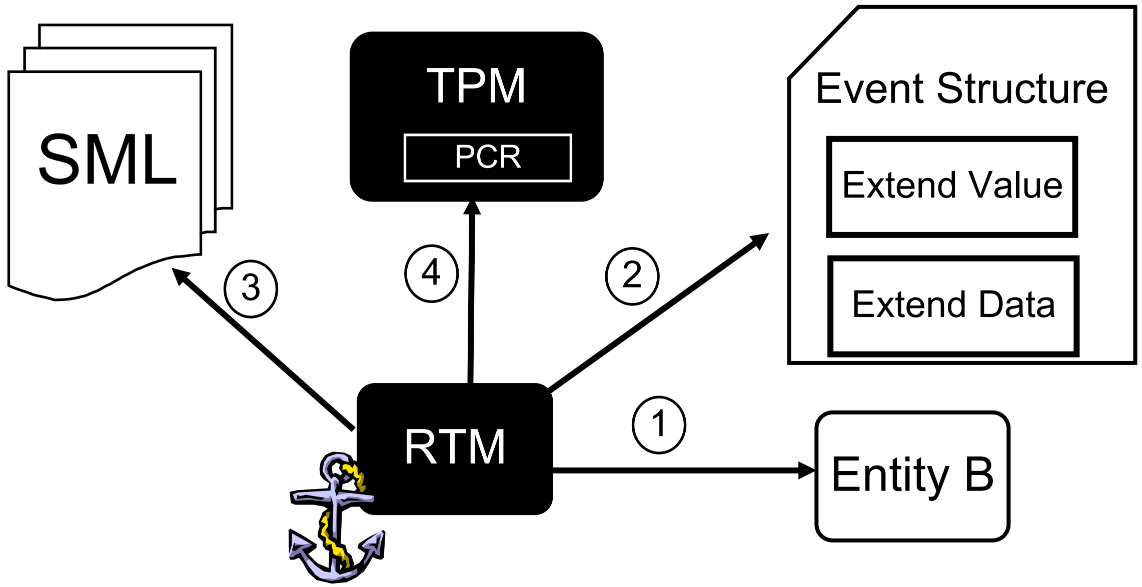
Chain of Trust and Measurements

- **Chain of Trust**
- **Chain measurement**
  - To trust the chain the identity of each member is needed
  - Identity = measurement according to TCG definition
  - Generic flow: Each member measures its successor before passing the control to it
- **Root of Trust**
  - Must be trusted, no mechanism to measure it
  - For creating chain of trust the first entity is RTM



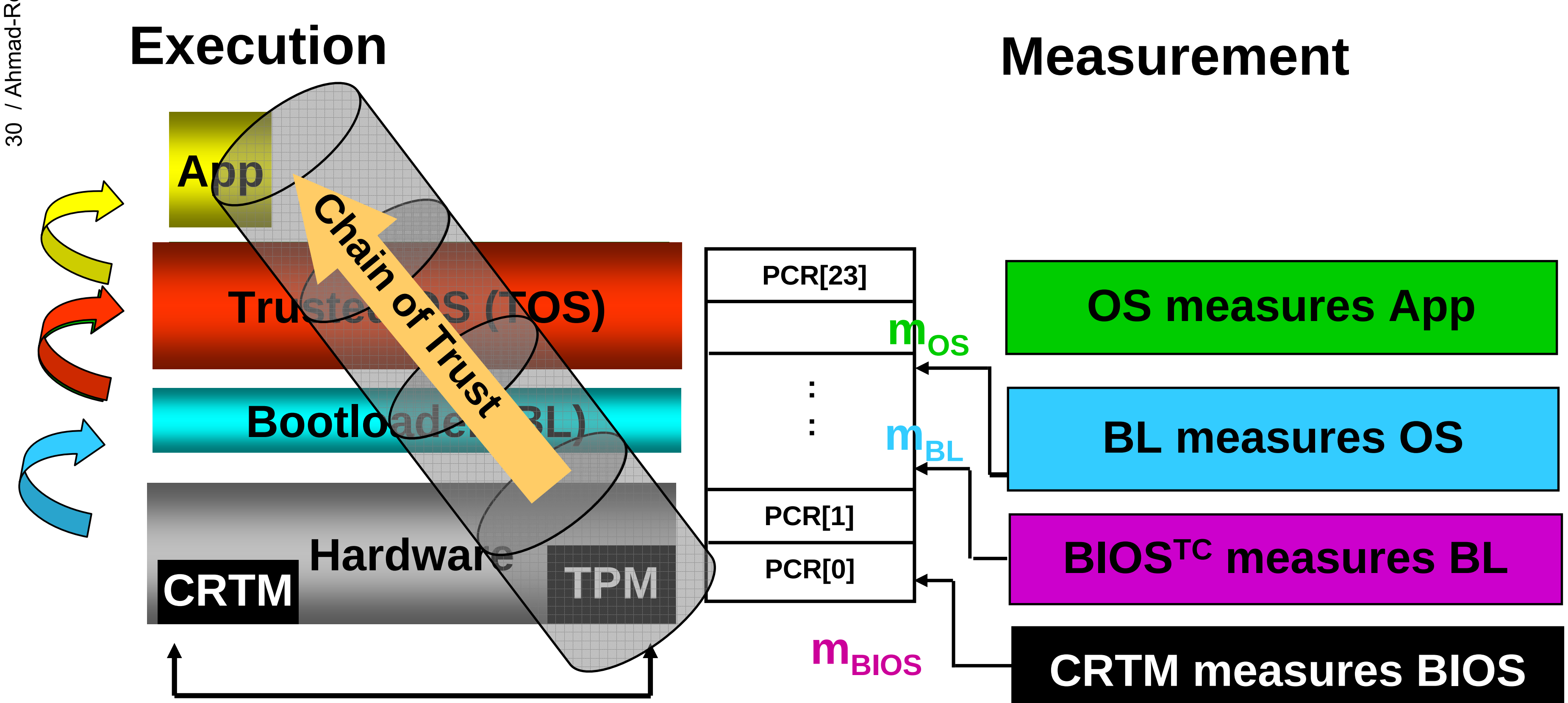
Measurements

- **Measurements**
  1. RTM measures entity B
  2. RTM creates Event Structure in SML (Stored Measurement Log)
     - SML contains the Event Structures for all measurements in the TPM
     - SML can be stored on any storage media, e.g., storage device
  - 3.RTM
  - RTM extends value into PCR
- **Event Structure**
  - Contains extend value (actual result of digest) and extend data
  - One structure for each measurement extended into the TPM



Bootstrap and Integrity Measurement

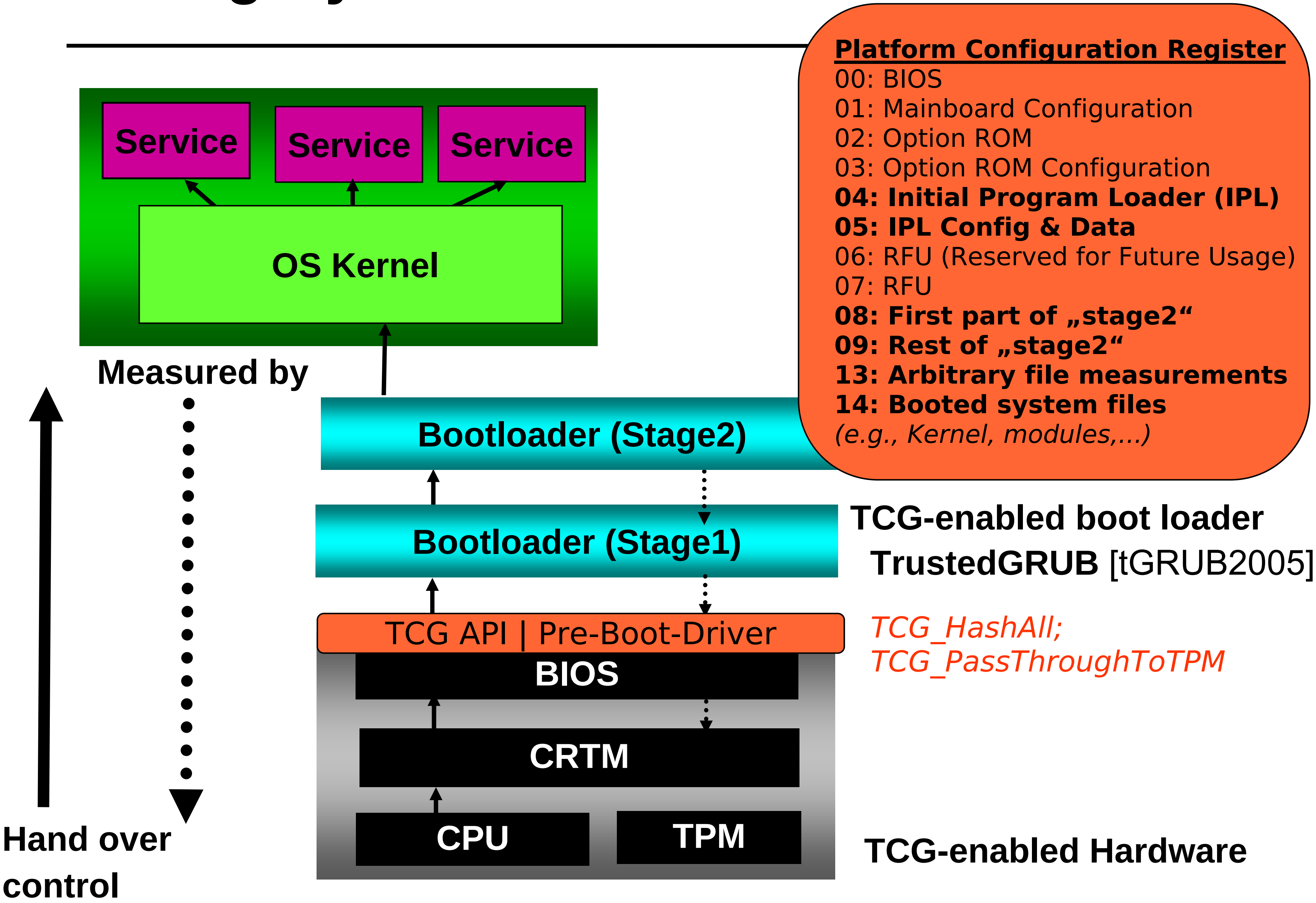
- Instantiation based on TCG approach



Trusted Components:

- Core Root of Trust for Measurement (CRTM)
- Trusted Platform Module (TPM)

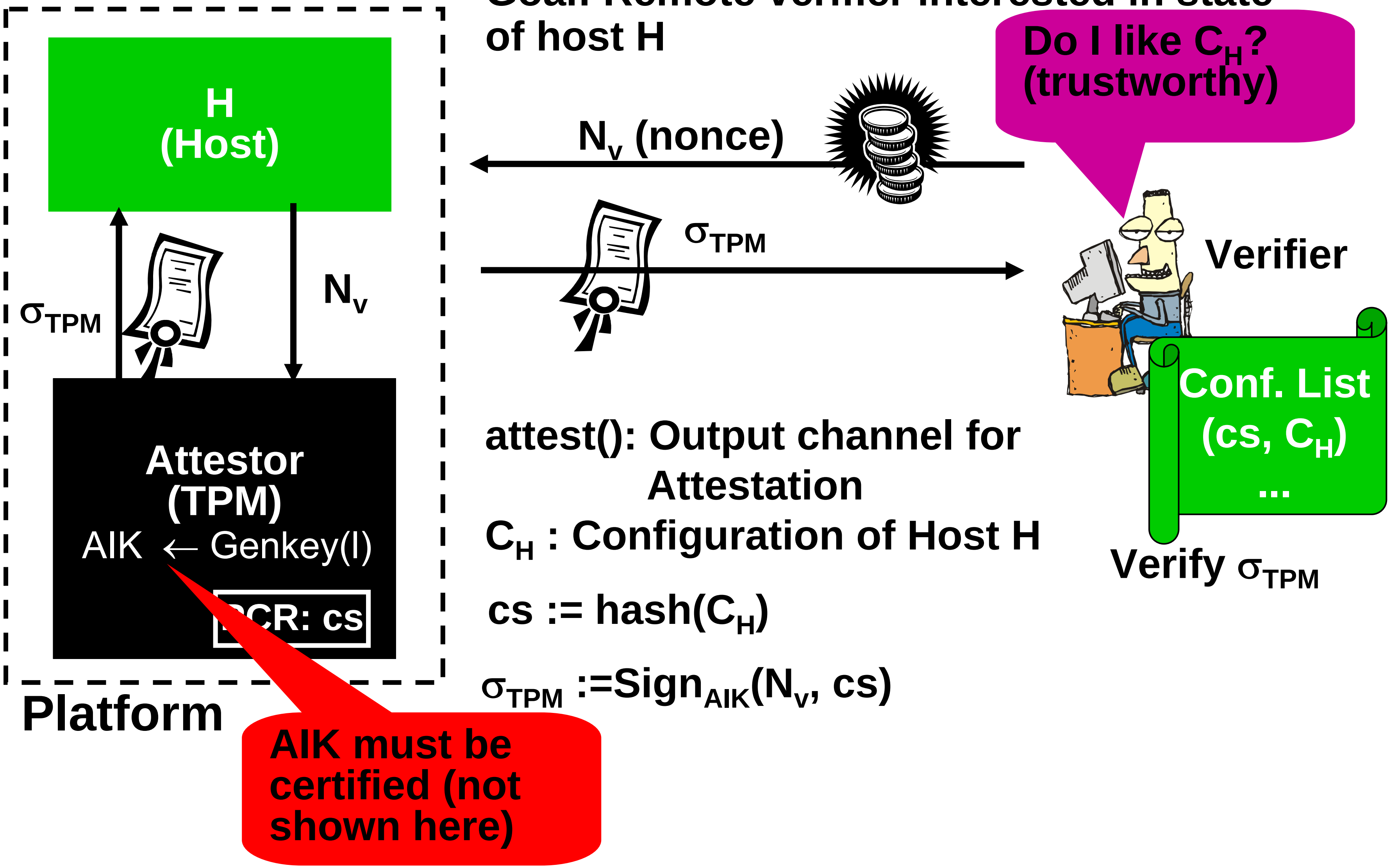
Integrity Measurement: More Details



Attestation

TCG Attestation (simplified)

Goal: Remote verifier interested in state of host H



Attestation Identity Key (AIK): Overview

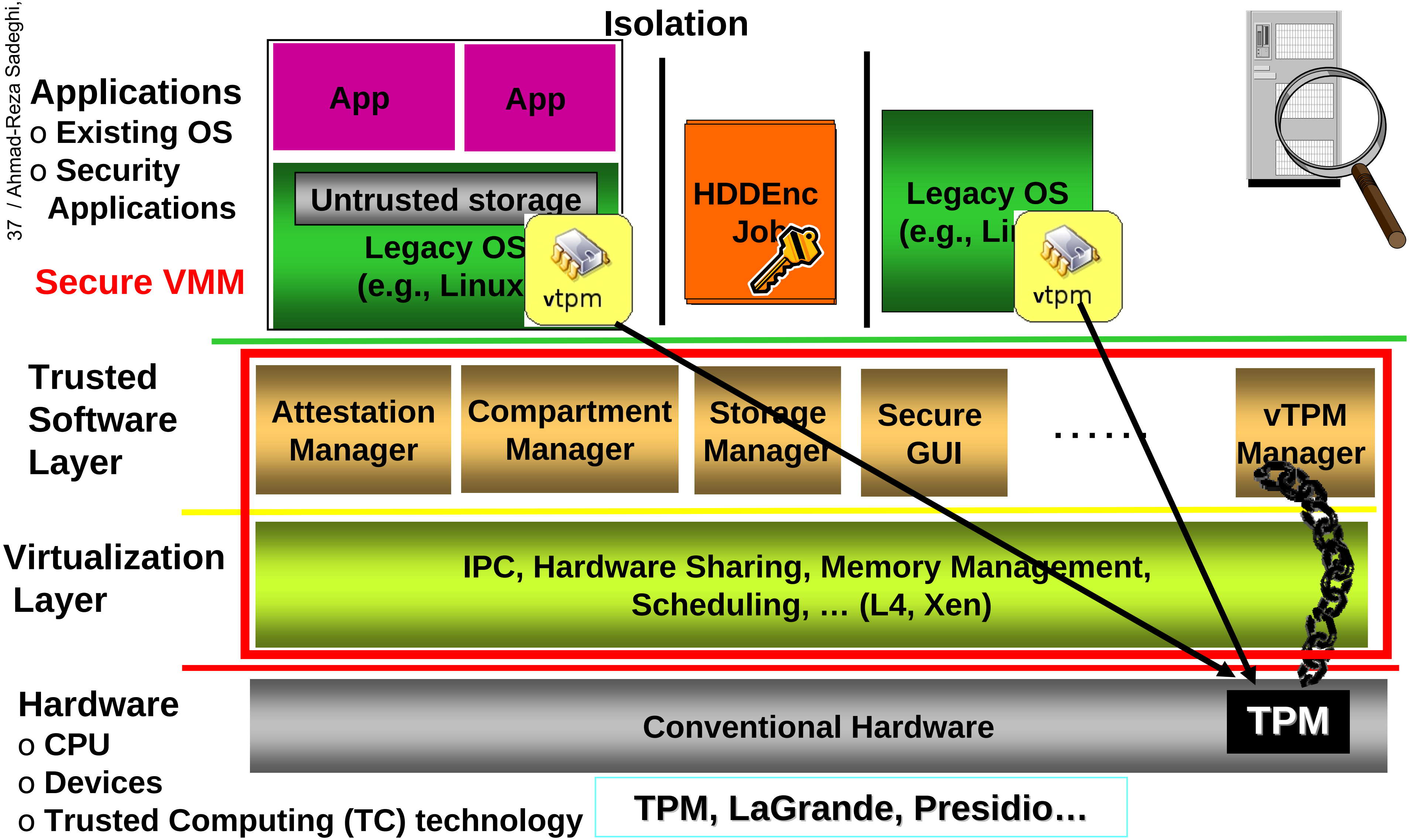
- Provides a signature key that can act a pseudonym
- Theoretically a TPM can have unlimited number of AIK (different key for each transaction)
- Certification Authority
  - Requires certification by a Trusted Third Party (Privacy-CA in TCG Terminology) certifying that an AIK comes from a TPM
  - Unlinkability achieved by DAA (Direct Anonymous Attestation) Protocols [BrCaCh2004]
    - No privacy-CA needed
    - A zero-knowledge proof of knowledge of possession of a valid certificate

Security Architectures Based on Virtualization

Some Terms

- **Compartment**
  - A process logically isolated from other processes
- **Configuration**
  - I/O behavior of a state machine based on an initial state
    - e.g., represented by the hash value of the binary code
- **Trusted Channel**
  - A secure channel verifying expected configuration of an endpoint compartment
    - e.g., verify hash of the compartment against a reference value

Proposed Architecture

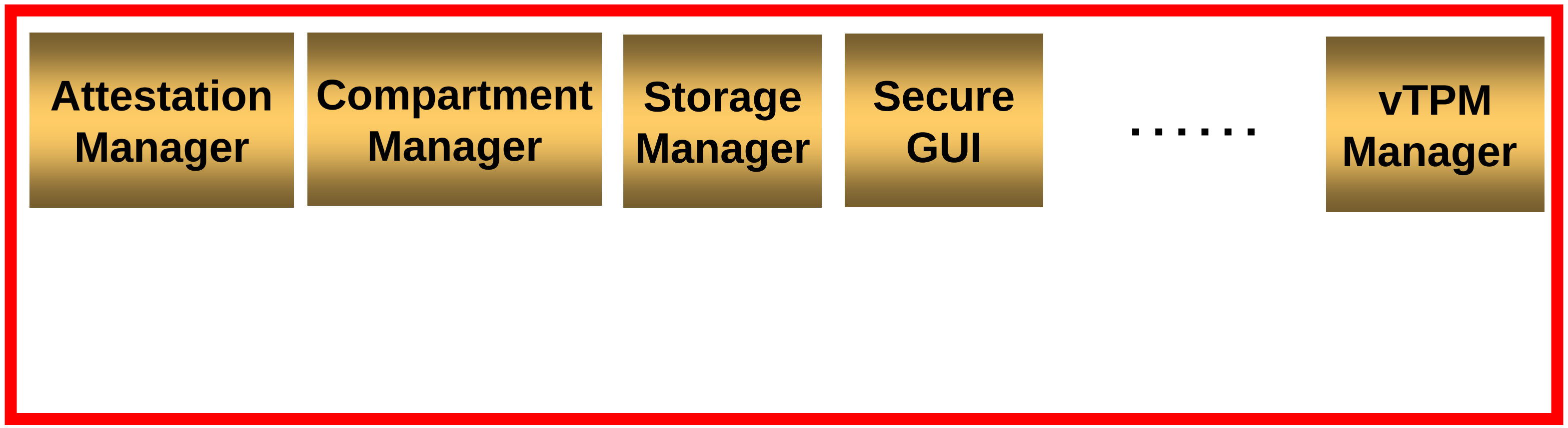


Virtualization Layer

- Provides an abstraction of underlying hardware
  - e.g., CPU, devices, interrupts
- Offers management primitives
- Access control polices for resources
- Examples
  - Based on microkernels (L4 family) [Liedke1996]
  - Based on hypervisors (Xen) [Barham et al 2003]

Trusted Software Layer

- Provides elementary security properties
  - Trusted channels
  - Strong compartment isolation
- Main services
  - Trust Manager
  - Compartment Manager
  - Storage Manager
  - Secure GUI

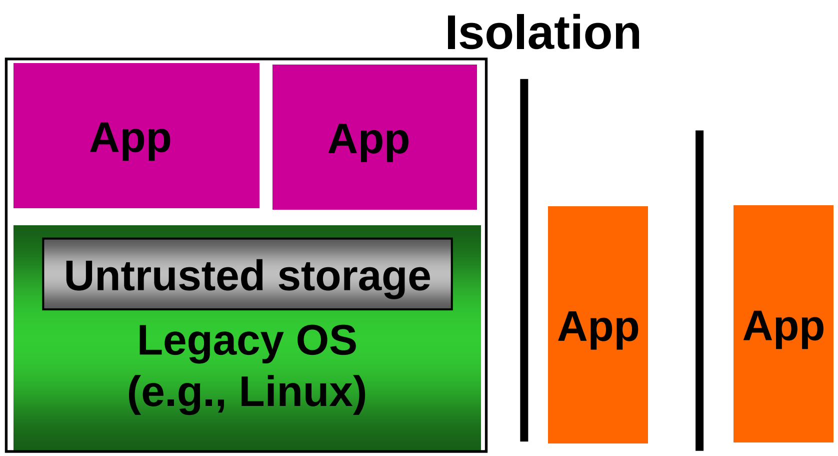


Trusted Software Layer Services

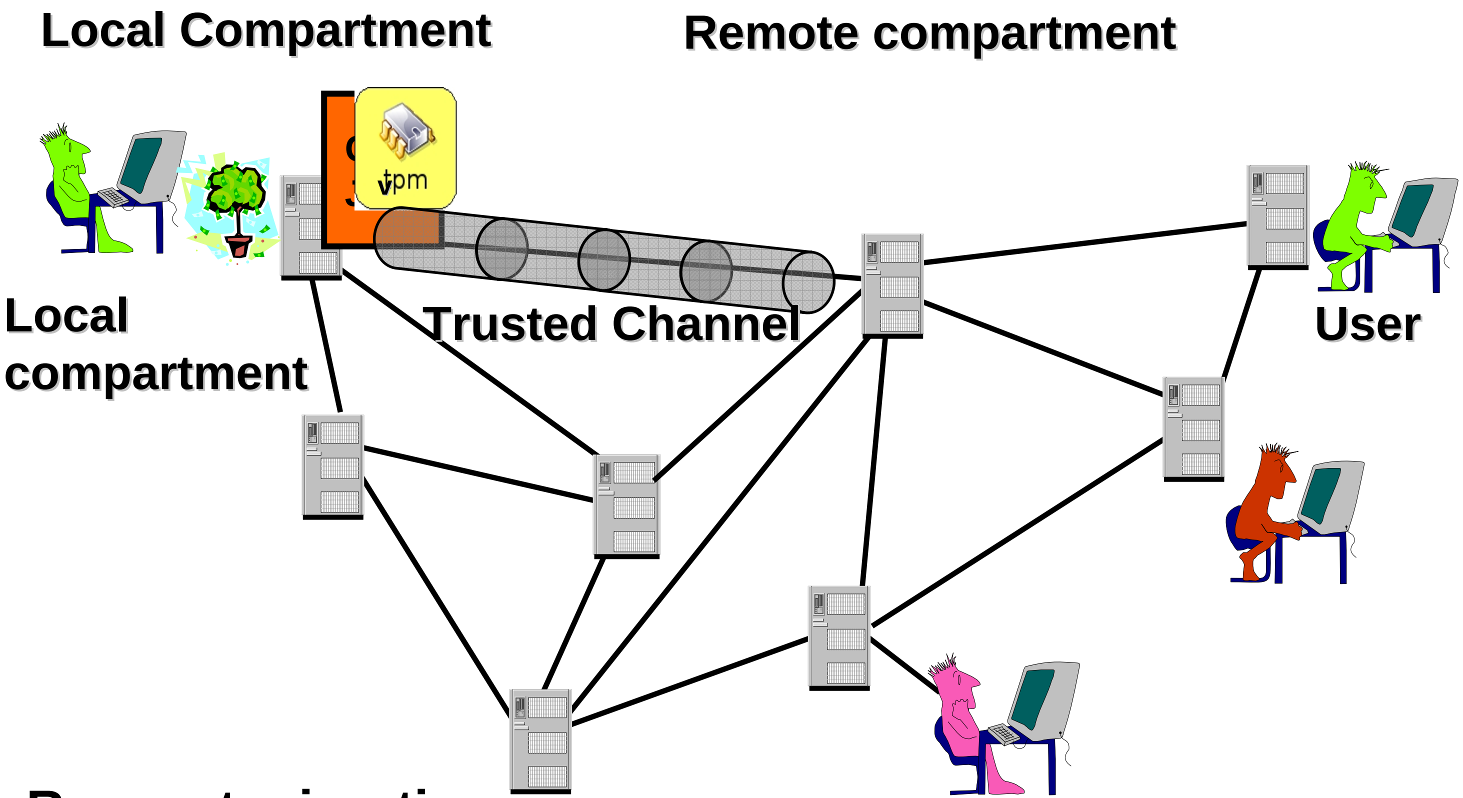
- **Compartment Manager**
  - Manages creation, updates, and deletion of compartments
- **Storage Manager**
  - Provide persistent storage while preserving integrity, confidentiality, freshness, …
  - Has access to configuration of clients it is communicating to over trusted channel
- **Attestation Manager**
  - Determines/Attests the properties of local and remote compartments

Application Layer

- Efficient migration of legacy software possible
- Isolation between applications of legacy services can be achieved by parallel instances of legacy OS



Job Migration in Data Center/Grid



- o **Request migration**
- o **Establish trusted channel to destination node**
- o **Transfer image and vTPM**
  - o vTPM state must not be subject to modification, duplication or comprise
- o **Update state of storage manager**

Selected TC related Research Activities/Projects

Overview

- **Trusted Virtual Domains**
  - Partly supported by METI Japan
  - www.trl.ibm.com/projects/tvd/
- **Open Trusted Computing (OpenTC)**
  - Funded by European Union
  - www.opentc.net
- **European Multilaterally Secure Computing Base**
  - Partly funded by the German Government
  - www.emscb.org
- **Trusted Mobile Computing (TRUCOM)**
  - Partly funded by the German Government
- **Trusted Embedded Computing (TECOM)**
  - European Project
  - In evaluation phase

Open Trusted Computing

- Building on the cost-efficient widely deployed TPM and the new generation of x86 CPUs from Intel and AMD ([LaGrande2003], [Pacifica2005])
- Define and implement an open Trusted Computing framework
  - across different platform and OS types
  - Distribution as Open Source software, supporting Linux in particular
- Consensus driven introduction of a transparent Trusted Computing framework and solutions
- Providing choice between proprietary and non-proprietary solutions for Trusted Computing
- Wide distribution by SUSE
- Collaborative, academic/industrial research project co-funded by the European commission
- 23 Partners
  - Academic: Bochum University (security architecture), Cambridge University (XEN), Dresden University (L4 microkernel)
  - Industrial: AMD, HP, Infineon, IBM, SuSE/Novell

OTC
Open Trusted Computing

OpenTC Use Cases

- **Personal Electronic Transaction**
  - Based on idea of colored computing (red for untrusted and green for trusted)
  - Trusted Virtual Machine
  - Initialization via Trusted GUI
  - Planed Demo November 2006
- **Cooperate computing at home**
  - Home PC
  - Virtual cooperate PC (CPC)
  - Trusted computing to enable corporation to trust CPC
- **Virtual data center**
  - Virtual customer infrastructure
  - Deployed on a smaller number of physical machines

EMSCB-Project

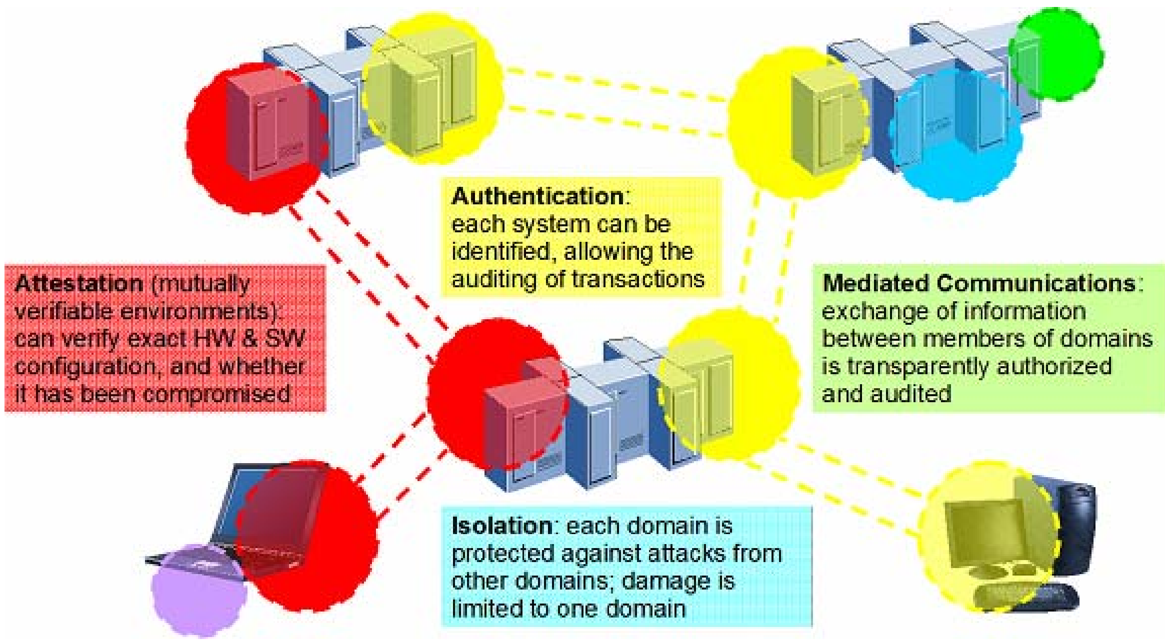
- European Multilaterally-Secure Computing Platform [SaStPo2004]
- Develop an open multilaterally-secure computing platform that is *secure enough* to allow new and innovative business models
- Based on
  - PERSEUS/Nizza ([Pfitzmann et al 2001] / [Haertig et al 2005])
  - L4 (Microkernel)
- 7 Partners from academia and industry
  - Academic: Bochum University (Security Architecture), Dresden University (L4 microkernel), Institute for Internet Security (Gelsenkirchen)
  - Industrial: Bosch/Blaupunkt, escrypt, Infineon, Sirrix, SAP

EMSCB Use Cases

- HDD-Encrypter (Prototype available)
  - Secure Booting
  - Isolated encryption keys
  - See also [Alkassar et al 2006]
- Secure VPN Module (Prototype available)
  - Isolated Certificates
  - Application Attestation
  - See also [Alkassar et al 2006]
- Fair DRM Prototype (End of 2006)
  - Protection of digital content
  - Enforcement of pragmatic security policies
- Enterprise Rights Management (End of 2007)
  - Isolation of Linux compartments
  - Enforcement of different security policies
- Embedded DRM Viewer (End of 2007)
  - Navigation System in cars

Trusted Virtual Domains

- Simplifying management and providing explicit infrastructure-level [Bussani et al 2005]
  - Containment: Isolation of the computing entities used to perform a service regardless of the physical machine or network topology configuration of those entities (domains)
  - and trust guarantees by conveying integrity verification each entity within the domain
- Use case: System management in strategic outsourcing (Data Centers processing data of different customers)
- Project: IBM Tokyo and METI



Reactions to Trusted Computing Group Approach

Concerns

- Since its announcement, TCG has been subject to much criticism
  - Potential basis for DRM
  - Less freedom (including freedom of choice and user control)
  - Privacy violation (disclosing platform identity and configuration)
  - Confusing language: Trust, Control, Opt-in
  - Core specifications unreadable (leads to misunderstanding)
- Much of the criticism is related to Microsoft's NGSCB
  - Several name changes from Palladium to NGSCB, Longhorn to Vista [Microsoft2003a, Microsoft2003b, Microsoft2003c, Vista2006]
  - Bad publicity or legal challenges on rights to the name (see, e.g., [Lemo2003, Bech2003])
- Danger of restricting competition
  - Misuse of sealed storage capabilities to prevent other applications from accessing data, thus locking out alternative applications and inhibiting interoperation [Scho2003], [Ande2002, Ande2003, Cour2002]

Legal Requirements on TC/TCG

- German Government requirements catalogue on TCG
- Electronic Frontier Foundation (EFF) [Scho2003]
- European Commission Article 29 (Data Protection Working Party) [EC2004]
- Main common requests:
  - User’s privacy
    - Assurance: no back doors
    - No collection of user profiles
  - Unrestricted user control (e.g., over keys and IT technology)
  - Transparency of certification
  - Option for transferring secrets between different machines
  - Functional separation of TPM and CPU / chipsets
  - Product discrimination
- New Zealand Government’s initiative [NZG2006]
  - Defines principles and policies for TC/DRM composed system to ensure that the use of TC/DRM technologies does not adversely affect the integrity, availability and confidentiality of government-held information or related government systems

TC and Open Source

- Customer concerns
  - "Will TC be supported for Open Source based solutions?"
  - OSS systems frequently used in security critical environments
    - Strict requirements (audit, compliance, 'state of the art' mechanisms)
  - Main reason: transparency, vendor-independence
    - Important market segment of institutional and professional users
    - Government, public administration, financial, insurance, aerospace
- Concerns from parts of the OSS spectrum, typical reactions
  - TC may put OSS at a disadvantage
  - TC may lead to customer lock-in
    - No alternative to using a particular piece of software
  - TC could be "philosophically incompatible" with OSS
    - 'Treacherous Computing' (Stallman) has become issue for GPLv3 [GPLv3]
    - Highly controversial debate: Stallman vs. Torvalds
    - As of Sep. 2006: Stallman vs. Linux kernel developer community
    - Might lead to deep split in OSS communities & licensing models

Some Technical Challenges

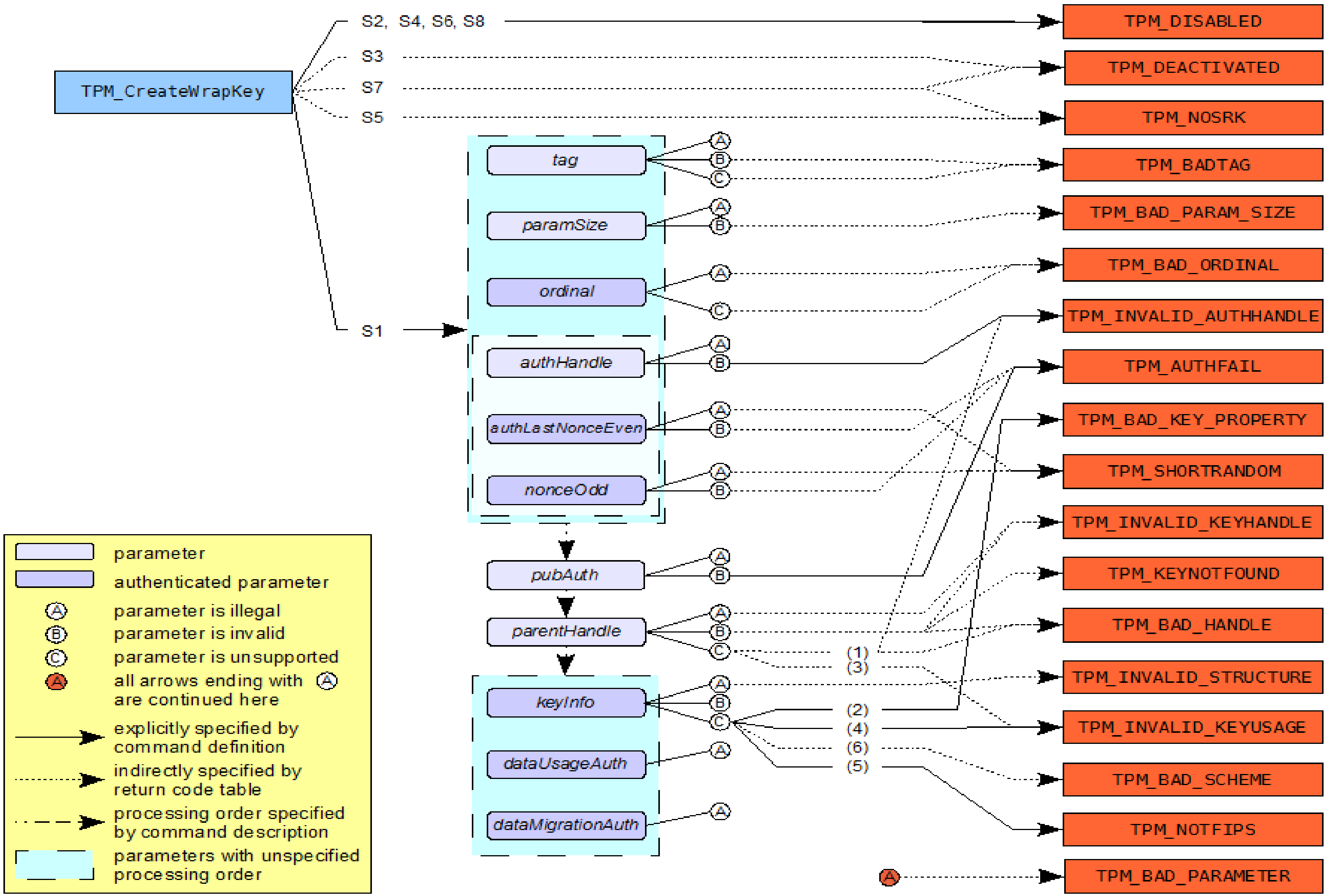
Overview

- In this talk
  - TPM complexity, compliance and security
  - Attesting properties instead of integrity
  - Efficient maintenance
  - Malicious virtualization
  - Widespread commercial applications
- Others
  - Computing platforms with dynamic HW Configuration
  - PKI problems
  - Formal models & methods

TPM Functionality and Complexity

- **Specification very complex & complicated**
  - Many commands (123) with many parameters (3 to19)
  - Which functionalities (and commands) are really needed?
- **TPM Compliance and Security Test**
  - Recent tests show *majority* of TPMs are not compliant with specification [Sadeghi et al 2006]
    - Need new and efficient test strategies and concepts
  - Some TPMs vulnerable to attacks due to weak implementations
    - e.g., dictionary attack, accessing keys without valid SRK authorization) [Sadeghi et al 2006]
  - In particular necessary from users' perspectives
- **TPM Emulation**
  - Based on existing functionalities (e.g., secure storage)
- **Integration of TPM into CPU or chipset**
  - Engineering trade off between security and technical evaluation
  - TPM Construction Kit
  - Towards more security against hardware attacks (see also [KuScPr2005])

TPM Functionality and Complexity: Command Structure and Relation



Conceptual Problems of Attestation / Sealing I

- **Discrimination**
  - Sealing/attestation has the potential to exclude alternative software products systems (e.g., Linux)
  - Sealing allows content providers to enforce usage of a specific platform configuration
  - Application vendors can exclude alternative software
- **Observable**
  - Verifier can obtain information about remote platform configuration

Conceptual Problems of Attestation/Sealing II

- **Inflexible**
  - System update: Sealed data is inaccessible after updating measured system components (e.g., patching TCB)
  - Might affect: cryptographic keys for accessing networks, documents, media files, etc
- **Complexity and management**
  - Vast number of different platform configurations
    - (constantly growing through patches, compiler options and software versions)
  - This makes it hard to keep track
    - “evolution of trustworthiness” of a given configuration

Property-Based Attestation (PBA)

- Verifier usually interested in whether the attested object provides the desired properties instead of specific configuration [SaSt2004]
- Property (informally)
  - describes an aspect of the behaviour of the underlying object with respect to certain requirements (e.g., a security-related)
- Properties on different abstraction levels
  - privacy-preserving, i.e., it has built-in measures conform to the privacy laws
  - provides Multi-Level Security (MLS)
  - security evaluated by a governmental organisation
- The choice of correct or useful property set and its correct definition depends strongly on the underlying use case and its requirements

PBA: Possible Approaches

- **Code control**
  - Property attestor is trusted to enforce that a machine can only behave as expected.
    - In a machine model this means that attestor compares the I/O behavior of M with that defined by the desired property P
    - Example: reference monitor and to attest both OS and the enforced security policy (e.g., [MaSmBaSt2004] for SE Linux [LoSm2001])
- **Code analysis**
  - property attestor directly analyses the code of the machine to derive properties
  - Exp.: proof-carrying code and semantic code analysis ([Necu2002], [HaChFr2003])
- **Delegation**
  - property attestor proves that another party has certified the presence of the desired properties [SaSt2004, Chen et al 2006]
    - Obviously, this third party has to be trusted by both

Sealed Data & Hardware Migration

- o TPM maintenance procedure [TPM2005]
  - o Process is optional
  - o No information on whether mechanism is implemented in any existing TPM
  - o Works only for TPMs of same vendor
  - o Needs interaction with vendor
    - o Vendor out of business?
    - o Price?
- o Efficient recovering of sealed data when HW breaks?

Platform Updates

- o Requirements for a patched TCB
  - o Security: Remote party wants that new platform configuration that adheres to the existing security policy.
  - o Availability: Owner/User wants protected information to be accessible before and after patch.
- o Solution proposals [KuKoSaSt2005]
  - o Software-supported
  - o TPM-supported
  - o Property-based sealing

Migration

- Requirements for TPM migration
  - Completeness: Platform owners should be able to securely transfer complete TPM state
  - Security:
    - Migration only if destination TPM at least as secure as source TPM
    - The state of the source TPM should be cleared afterwards
    - Confidentiality of TPM data
    - Delegate decision to trusted third party
  - Fairness: openly specified process
    - No need for interaction with vendor
- Solution proposal [KuKoSaSt2005]
  - A migration protocol with above properties

Virtualization Attacks

- Virtual-machine based rootkits
  - Compromise computing platforms
    - e.g., Blue Pill [Rutk2006], [Ligu2006], [Ou2006] and SubVirt [King et al]
  - Malicious virtual machine monitors have full access to the internal state of Virtual Machines (VM), thus to all secrets
  - Virtualized operating system cannot always detect the existence of malicious VMM
- Solutions must guarantee anti discrimination
- Solution proposal
  - Trusted Computing can help to prevent virtualization attacks
    - e.g., using property based-attestation [SaSt2004]
    - but, is it essential?
  - Efficient and flexible solutions needed

Secure Multiparty Computation

- Protocols will be more efficient bounds will not change (see, e.g., [BeDoFe2006])
- Note that a TPM has limited functionality and resources

Summary and Conclusion

- **Trusted Computing is an emerging technology**
  - Still needs many improvements
  - It is not restricted to the TPM technology (although competition on market segments already started)
  - Possible deriving/pushing technology for secure operating systems?
  - Europe plays an important role (TPM manufacturing, research in TC)
- **Careful deployment of TC**
  - Protect end-user rights
  - Provide the right environment
    - No discrimination and space for innovation (small and mid-sized enterprises)
  - Understanding TC and having impact
- **Long term solutions require international and joint efforts**
  - Academia, governments and industry
  - Establishing reasonable standards
  - Not to forget our purpose (more security for IT Systems) and not only extending them with functionalities

References

[Alkassar et al 2006] Ammar Alkassar, Michael Scheibel, Ahmad-Reza Sadeghi, Christian Stüble, Marcel Winandy: Security Architecture for Device Encryption and VPN. Accepted for ISSE (Information Security Solution Europe) 2006.

[Ande2001] Ross Anderson: Security Engineering: A Guide to Building Dependable Distributed Systems. Wiley (2001), ISBN 0-471-38922-6, 2001.

[Ande2002] Ross Anderson: Cryptography and Competition Policy Issues with 'Trusted Computing'. Technical report, Cambridge University, 2002.

[Ande2003] Ross Anderson: 'Trusted Computing' Frequently Asked Questions: TC/TCG/LaGrande/NGSCB/Longhorn/Palladium/TCPA. Available at `http://www.cl.cam.ac.uk/~rja14/tcpa-faq.html`, 2006.

[ArFaSm1997] William A. Arbaugh, David J. Farber and Jonathan M. Smith: A secure and reliable bootstrap architecture. In Proceedings of the IEEE Symposium on Research in Security and Privacy, pages 65-71, Oakland, CA, May 1997. IEEE Computer Society, Technical Committee on Security and Privacy, IEEE Computer Society Press.

[AvLaLaRa2004] A. Avizienis, J.-C. Laprie, B. Randell and C. Landwehr: Taxonomy of Dependable and Secure Computing. IEEE Transactions on Dependable and Secure Computing, Volume 1, Issue 1, pp 11-33 IEEE, 2004.

[Barham et al 2003] Paul Barham, Boris Dragovic, Keir Fraser, Steven Hand, Tim Harris, Alex Ho, Rolf Neugebauer, Ian Pratt and Adrew Warfield: Xen and the art of virtualization. In Proceedings of the 19th ACM Symposium on Operating Systems Principles (SOSP'03), Bolton Landing, NY, USA, October 2003.

[Bech2003] S. Bechtold: The Present and Future of Digital Rights Management — Musings on Emerging Legal Problems. Digital Rights Management, LNCS 2770:597-654, 2003.

[BeDoFe2006] Zinaida Benenson, Milan Fort, Felix C. Freiling, Dogan Kesdogan, Lucia Draque Penso: TrustedPals: Secure Multiparty Computation Implemented with Smart Cards. 11th European Symposium on Research in Computer Security (ESORICS 2006), September 2006, Hamburg, Germany.

[BrCaCh2004] Ernie Brickell, Jan Camenisch, and Liqun Chen. Direct anonymous attestation. In Proceedings of the 11th ACM Conference on Computer and Communications Security, Washington, DC, USA, October 2004. ACM Press.

[Bussani et al 2005] A. Bussani, J.L. Griffin, B.Jansen, K. Julisch, G. Karjoth, H. Maruyama, M. Nakamura, R. Perez, M. Schunter, A. Tanner, L. Van Doorn, E.A. Van Herreweghen, M. Waidner, S. Yoshihama: Trusted Virtual Domains: Secure Foundations for Business and IT Services. Whitepaper, RC23792, November 9, 2005.

[Chen et al 2006] Liqun Chen, Rainer Landfermann, Hans Loehr, Markus Rohe, Ahmad-Reza Sadeghi and Christian Stüble: A Protocol for Property-Based Attestation. Accepted for The First ACM Workshop on Scalable Trusted Computing (STC'06).

[Cole1990] James Coleman: Foundations of Social Theory, Harvard Edition World, 1990

[Cour2002] D. Coursey: Why we can't trust Microsoft's 'trustworthy' OS. ZDNet, available at `http://www.zdnet.com.au/newstech/os/story/0,2000048630,20266389,00.htm`, 2002.

[EC2004] European Commission Article 29 (Data Protection Working Party), `http://ec.europa.eu/justice_home/fsj/privacy/workinggroup/`.

[GPLv3] GNU General Public License, Version 3. Available at `http://gplv3.fsf.org/`.

[HaChFr2003] V. Haldar, D. Chandra and M. Franz: Semantic remote attestation: A virtual machine directed approach to trusted computing. In USENIX Virtual Machine Research and Technology Symposium, May 2004. Also TechnicalReport No. 03-20, School of Information and Computer Science, University of California, Irvine, October 2003.

[Haertig et al 2005] Härtig, Hohmuth, Feske, Helmuth, Lackorzynski, Mehnert and Peter: The Nizza Secure-System Architecture. 12/2005 - CollaborateCom 2005.

[Itoi et al 2001] Naomaru Itoi, William A. Arbaugh, Samuela J. Pollack and Daniel M. Reeves: Information Security and Privacy. 6th Australasian Conference, ACISP 2001, Sydney, Australia, July 11-13, 2001, Proceedings.

[King et al] Samuel T. King, Peter M. Chen, Yi-Min Wang, Chad Verbowski, Helen J. Wang, and Jacob R. Lorch: SubVirt: Implementing malware with virtual machines. Conditionally accepted pending shepherd approval to the 2006 IEEE Symposium on Security and Privacy, May 2006.

[KuGe2003] Dirk Kuhlmann, Robert A. Gehring: Trusted Platforms, DRM, and Beyond. In: Eberhard Becker, Willms Buhse, Dirk Günnewig / Niels Rump (Hrsg.): Digital Rights Management: Technological, Economic, Legal and Political Aspects, Springer, Berlin, Heidelberg, New York 2003, S. 178-205.

[Kuhl2003] Dirk Kuhlmann: On TCPA. FC 2003, LNCS 2742, pp. 255-269, 2003.

[KuKoSaSt2005] Ulrich Kühn, Klaus Kursawe, Stefan Lucks, Ahmad-Reza Sadeghi, and Christian Stüble: Secure data management in trusted computing. In Cryptographic Hardware and Embedded Systems — CHES 2005, volume 3659 of Lecture Notes in Computer Science, pages 324–338. Springer-Verlag, Berlin Germany, 2005.

[KuScPr2005] K. Kursawe, D. Schellekens, and B. Preneel: Analyzing trusted platform communication. In ECRYPT Workshop, CRASH - Cryptographic Advances in Secure Hardware, 8 pages, 2005.

[LaGrande2003] Intel Corporation: LaGrande technology architectural overview. Technical Report 252491-001, Intel Corporation, September 2003.

[Lemo2003] R. Lemos: What's in a name? Not Palladium. C-Net News, available at http://news.com.com/2100-1001-982127.html, 2003.

[Liedke1996] J. Liedke: Towards real micro-kernels. Communications of the ACM, 39(9), 1996.

[Ligu2006] Anthony Liguori: Debunking blue pill myth. Available at http://www.virtualization.info/2006/08/debunking-bluepill-myth.html, August 2006.

[LoRaSaScSt2006] Hans Löhr, Hari Govind V. Ramasamy, Stefan Schulz, Matthias Schunter, Christian Stüble: Enhancing Grid Security Using Trusted Virtualization. Accepted to be presented at The Second Workshop on Advances in Trusted Computing (WATC '06 Fall).

[LoSm2001] Peter Loscocco and Stephen Smalley: Integrating flexible support for security policies into the Linux operating system. Technical report, U.S. National Security Agency (NSA), February 2001.

[Luhm1979] Niklas Luhmann: Trust as a Reduction of Complexity. In: Trust and Power: Two Works of Niklas Luhmann, New York: John Wiley and Sons, 1979, pp. 24-31.

[MaJiMa2006] W. Mao, H. Jin, and A. Martin: Innovations for grid security from trusted computing. Made available online at http://www.hpl.hp.com/personal/Wenbo_Mao/research/tcgridsec.pdf, 2006.

[MaSmBaSt2004] J. Marchesini, S. Smith, O. Wild, A. Barsamian, and J. Stabiner: Open source applications of TCPA hardware. In 20th Annual Computer Security Applications Conference. ACM, Dec. 2004.

[Microsoft2003a] Microsoft: NGSCB Technical FAQ. Available at http://www.microsoft.com/technet/security/news/ngscb.mspx.

[Microsoft2003b] Microsoft: Next Generation Secure Computing Base. Available at http://www.microsoft.com/resources/ngscb/default.mspx.

[Microsoft2003c] Microsoft: Next-Generation Secure Computing Base Product Information. Available at http://www.microsoft.com/resources/ngscb/productinfo.mspx.

[Necu2002] G. Necula: Proof-carrying code. In 24th Symposium on Principles of Pro-gramming Languages (POPL), pages 106-119, Paris, France, Jan. 1997, ACM Press.

[NZG2006] The New Zealand Government State Service Commission, http://www.e.govt.nz/policy/tc-and-drm.

[Ou2006] George Ou: Detecting the blue pill hypervisor rootkit is possible but not trivial. Available at http://blogs.zdnet.com/Ou/?p=297, August 2006.

[Pacifica2005] Advanced Micro Devices, Inc: AMD64 Virtualization Codenamed 'Pacifica' Technology. 33047-rev. 3.01 edition, May 2005.

[Pfitzmann et al 2001] B. Pfitzmann, J. Riordan, C. Stüble, M. Waidner, A. Weber: The PERSEUS System Architecture. IBM Technical Report RZ 3335 (#93381), IBM Research Division, Zurich Laboratory, 2001.

[Rann1994] Kai Rannenberg: Recent Development in Information Technology Security Evaluation – The Need for Evaluation Criteria for multilateral Security. In Richard Sizer et al.: Security and Control of Information Technology in Society — Proceedings of the IFIP TC9/WG 9.6 Working Conference August 12-17, 1993, pp. 113-128, onboard M/S Ilich and ashore at St. Petersburg, Russia; IFIP Transactions A-43; North-Holland, Amsterdam et al. 1994; ISBN 0-444-81831-6.

[RoSiBuCa98] D. M. Rousseau, S. B. Sitkin, R. S. Burt and C. Camerer: Not So Different After All: A Cross-Discipline View of Trust. The Academy of Management Review. Vol. 23, Num. 3, pp. 393-404, 1998.

[Rutk2006] Joanna Rutkowska: Subverting vista kernel for fun and profit. Available at http://blackhat.com/presentations/bh-usa-06/BHUS-06-Rutkowska.pdf, July 2006.

[Sadeghi et al 2006] Ahmad-Reza Sadeghi, Marcel Selhorst, Christian Stüble, Christian Wachsmann and Marcel Winandy: TCG Inside? - A Note on TPM Specification Compliance. Accepted for The First ACM Workshop on Scalable Trusted Computing (STC'06).

[Sailer et al 2005] R. Sailer, E. Valdez, T. Jaeger, R. Perez, L. van Doorn, J. L. Griffin, S. Berger: sHype: Secure Hypervisor Approach to Trusted Virtualized Systems. IBM research report Research Report RC23511, 2005.

[SaSt2004] Ahmad-Reza Sadeghi, Christian Stüble: Property-based Attestation for Computing Platforms: Caring about policies, not mechanisms. Panel on Themes and Highlights of the New Security Paradigms Workshop 2004, presented at 20th Annual Computer Security Applications Conference (ACSAC) December, 2004.

[SaStPo2004] Ahmad-Reza Sadeghi, Christian Stüble, Norbert Pohlmann: European Multilateral Secure Computing Base - Open Trusted Computing for You and Me. Datenschutz und Datensicherheit (DUD) 9/2004, Vieweg Verlag, pp. 548-554, 2004.

[Scho2003] S. Schoen: Trusted Computing: Promise and Risk. Technical report, Electronic Frontiers Foundation, available at http://www.eff.org/Infra/trustedcomputing/20031001tc.php.

[Shap1999] Jonathan S. Shapiro: EROS: A Capability System. PhD thesis, University of Pennsylvania, April 1999.

[TCG] Trusted Computing Group Website available at www.tcg.org.

[tGRUB2005] Trusted GRUB Project Homepage available at http://www.prosec.rub.de/trusted_grub.html.

[TPM2002] Trusted Computing Platform Alliance (TCPA): TCG Main Specification Version 1.1b. February 2002.

[TPM2003] Trusted Computing Group: TCG TPM Specification Version 1.2 Revision 85. February 2005.

[TPMMatrix2006] Available online at http://www.tonymcfadden.net/tpmvendors.htm

[TrouSerS] TrouSerS: An open-source TCG Software Stack implementation. Available at http://trousers.sf.net.

[TrustedMach1991] Kernel Primer: Trusted Information Systems Inc., Draft 01, November 27, 1991.

[Vista2006] Microsoft Windows Vista Product Homepage available at http://www.microsoft.com/windowsvista/.