Virtual Trusted Platform Technology

Dr. Qingni Shen
Peking University

Intel UPO Supported
Main Points

- An Overview of trust computing technology
- Virtual trusted platform architecture
- TCG VPWG working group
- Key technologies of virtual trusted platform
An Overview of trust computing technology
Concept of trust computing

◆ Generally speaking, the goal of trust computing is to make computing devices (such as computers) more secure through dedicated hardware.

◆ From the perspectives of users, trust computing provides reliable measurements and reports of platform and its state information through some mechanisms, convincing the user that computing platform’s software and hardware environment to run as intended.

◆ From the manufacturer’s point of view, trust computing is an idea that resisting certain software-based attacks through hardware-oriented platform.

◆ Overall, during the formation process of trust computing, the research and development of fault-tolerant computing, secure OS, network security and other fields have expanded the meaning of trust computing, from the concentration on reliability and availability of hardware to the comprehensive trust of hardware platform architecture and software system services.
**Trusted Platform Module (TPM):**

- A small computing system contains a cryptographic computing component and storage component.
- It works before OS and BIOS, and cannot use the computer memory and storage. Therefore, some public security algorithms must be implemented in order to standardize the interfaces for other components and provide internal cryptographic computing during some security operations.

**TPM requires at least four main functions:**

1. Symmetric/asymmetric encryption;
2. Secure storage;
3. Integrity measurement;
4. Signature Verification.

- Asymmetric encryption and signature verification of data is realized through the RSA algorithm.
- The integrity measurement is accomplished by the efficient SHA-1 hash algorithm.
- Symmetric encryption, which can use any algorithm, can be implemented by either the dedicated co-processor or the software.
TPM provides protected function and isolated position

- **Protected function**: A set of commands, with which we can access the shielded position.

- **Isolated position**: Memory, registers and so on. It is safe to operate sensitive data in these fields, but the location of data can only be accessed by protected function.
- **I/O device**: Responsible for the management of information transmitting on the communication bus. Its tasks include the implementation of corresponding encoding/decoding protocol of internal and external bus communication, and sending messages to appropriate target devices.

- **Cipher Coprocessor**: Implementation of encryption, decryption, digital signature and signature verification. TPM uses conventional methods to carry out the encryption operation, including asymmetric key generation (RSA), asymmetric encryption/decryption (RSA), hash (SHA-1), random number generation (RNG). TPM can also use an internal symmetric encryption system, but it should not expose any symmetric algorithm function to users. TCG’s new technical specification requires TPM to provide 2048 bit RSA algorithm.

- **Key generator**: Responsible for the generation of symmetric key and asymmetric key pair. For the key generation process of RSA algorithm, key generator will use RNG to generate random numbers, and completes the large prime number test of these random numbers.

- **HMAC Engine**: Provide data authentication code and message authentication code to ensure the integrity of data and command message. HMAC engine only provides computing function, and does not support command and mechanism of data transmission. HMAC parameters are in accordance with the standard proposed in RF2104, using 20 bytes key and 64 bytes data block.

- **Non-volatile storage**: Used to store permanent identifiers (such as EK) and TPM related states.
Random number generator: Responsible for generating random numbers key generation and signature. It transforms an unpredictable input into a 32 bytes random number through a internal state machine and a single hash function. The input data can be generated by software or hardware, such as noise, clock, temperature etc., which is nor visible to the outside. TPM can also use pseudo random number generation algorithm in RNG.

SHA-1 Engine: Responsible for basic hash functions. Hash interface is exposed to the outside, so that the platform can implement integrity verification at startup. TCG does not provide the minimum throughput requirement of SHA-1.

Power monitor module: Manage TPM’s power according to the power state of TPM, and help to takes appropriate measures while detect the change of power status.

Opt-In: Realize a mechanism of turning on and off internal functions of TPM which is defined in TCG policy. The implementation of turning on, turning off and fully activating of internal function is through changing some permanent variable flags. But this setting must be under TPM owner’s permission or authorization, and it is nor allowed for remote setting in principle.

Execution Engine: Contain CPU and embedded software, run commands that sent to TPM through I/O, and implement the initialization of TPM and monitoring operation.
Trust computing platform

◆ TPM based “Trust Computing” consists of several aspects:
  ➢ Authentication of user identity, reflects the trust in users.
  ➢ Platform integrity, reflects user’s trust in platform runtime environment.
  ➢ Application integrity, reflects the trust in applications.
  ➢ Verifiability between platforms, reflects the mutual trust between terminals in the network environment.

◆ Users can achieve more protection through the protection of TPM hardware against software-level attacks.
TPM can not only establish a trust chain in kinds of terminal systems, but also extend the trust chain into network. We need to solve:

- **Trusted transfer.** TPM increases the transmission security by providing key management and configuration management (storage protection, measurement and report).

- **Identity authentication.** In TCG system, identity authentication generally uses the AIK (Attestation Identity Key) as a alias of platform’s permanent identity EK (Endorsement Key). Also, there is another method which is called direct anonymous attestation.

- **Trusted interconnection.** TCG organization has established a network connection specification based on trust computing technology, including an open terminal integrity architecture and a set of secure interoperability standards.
Integrity Measurement Architecture (IMA)
Integrity Measurement Architecture (IMA) is usually implemented as an OS kernel, which makes the kernel has the ability to measure the loaded softwares, so that builds a complete chain of trust.

★ When system kernel boots up, this module measures every loaded application, runtime library and kernel module.
★ It creates and maintains a measurement list, and send the measurement value to TPM while adding items.
★ When the local platform needs to provide platform status proof to remote platform, IMA can send local stored measurement lists and measurement signed by TPM to remote side, and the latter will validate measurement list by computing the hash value of measurement list and comparing them with measurement values, and then check the status of platform according to specific measurement values.
Trusted Software Stack

Trusted software stack (TSS) provides trusted services, which are offered by TPM, to applications through a hierarchical structure. It is divided into 4 layers, from bottom to top:

⭐ **TPM Device Driver (TDD):** Generally provided by TPM manufacturer, and the code is loaded and executed in kernel mode.

⭐ **TPM Device Driver Library (TDDL):** It provides user mode interface, and ensures that different forms of TSS can communicate with any TPM, and offers interface which has nothing to do with OS to TPM application.

⭐ **TSS Core Service (TCS):** This interface is designed to provide a direct and simple way to control and request TPM’s service.

⭐ **TSS Service Provider (TSP):** Provide a rich and object-oriented interface to applications in order to show all the features of TCG platform. Every process can have a TSP, and in the OS with multiple threads, there may be more TSP reside on the platform.
Virtual trusted platform Architecture
Virtual TPM (vTPM) is the core of virtual trusted platform, which provides trust computing services for VM application through virtualized TPM.
The implementation structure of vTPM:

★ In the virtual trusted platform, driver of physical device exists in privileged VM, which is responsible for the management and scheduling of underlying resources. But in user VM (non-privileged), the use of the hardware is through interior front-end driver. The front-end driver does not interact with hardware directly, and accesses underlying resource by drivers of privileged VM (known as back-end driver) indirectly.

★ TPM’s virtualization uses the same approach. Privileged VM loads TPM back-end driver and creates multiple vTPM instances in this VM using vTPM manager. These instances interact with TPM front-end driver of user VM, and provide trusted services for applications in the VM.
**Requirements:**

Each VM that needs TPM wants to access its own private TPM, even if the system has more VMs than physical TPM (usually each platform has a unique hardware TPM).

**Object:**

Create multiple virtual TPM instances so that every instance can truthfully emulate the function of a hardware TPM.
Four problems to solve in order to keep TPM security property during the design of TPM virtualization

◆ (1) Virtual TPM must provide the same usage model and TPM command set for OS that runs in VM as those that hardware TPM provides for OS which is running on hardware platform.

◆ (2) During the life cycle of VM, there must be a strong association between VM and its virtual TPM, including that VMs and their associated virtual TPMs should be migrated from one physical machine to another machine together.
(3) A strong association between each virtual TPM and its underlying Trusted Computing Base (TCB) should be maintained.

(4) Virtual TPM can clearly be distinguished from hardware TPM, while these two types of TPM should have some different security properties.
# Open source resources

<table>
<thead>
<tr>
<th>Resource</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>TPM Emulator</td>
<td><a href="http://tpm-emulator.berlios.de">http://tpm-emulator.berlios.de</a></td>
</tr>
<tr>
<td>Trusted Grub</td>
<td><a href="https://projects.sirrix.com/trac/trustedgrub">https://projects.sirrix.com/trac/trustedgrub</a></td>
</tr>
<tr>
<td>TPM Driver</td>
<td>Linux kernel 2.6.12</td>
</tr>
<tr>
<td>IMA</td>
<td><a href="http://sourceforge.net/projects/linux-ima">http://sourceforge.net/projects/linux-ima</a></td>
</tr>
<tr>
<td>Trousers</td>
<td><a href="http://trousers.sourceforge.net/">http://trousers.sourceforge.net/</a></td>
</tr>
<tr>
<td>XEN</td>
<td><a href="http://www.xen.org">http://www.xen.org</a></td>
</tr>
<tr>
<td>vTPM</td>
<td><a href="http://www.xen.org">http://www.xen.org</a></td>
</tr>
</tbody>
</table>
Virtual Trusted Platform Architecture based on open source software.
1. The underlying platform has a TPM chip support, and has been activated.

2. Deploy trusted grub.
   - Trusted Grub is a patched trusted boot program according to TCG specification. It adds support of IMA for Grub, so that the trust chain can extend from bootloader to OS. It will measure the hash value of the image file before it is loaded and stores the value into TPM.

3. Deploy virtual platform that supports management of TPM and vTPM.
   - Linux kernel version 2.6.12 and above, which is integrated with the TPM driver.
   - Open source Xen 3.0 and above provides the support for vTPM. It realizes front-end and back-end drivers of vTPM and vTPM manager. TPM driver(This driver is called back-end TPM driver in Xen) will also be compiled while deploying Dom0. TPM front-end driver should be added to DomU’s kernel, and vTPM support is also needed while deploying DomU. We can get hardware TPM services by corresponding back-end TPM driver in Dom0.
4. Deploy kernel patches of IMA that supports application layer integrity measurement, that is, install open source IMA patches provided by IBM for Dom0’s and DomU’s Linux kernel, in order to support the integrity measurement, storage and reporting of applications.

5. Finally, deploy the trusted software stack, that is, use open source Trousers software packages provided by IBM on the platform, which provides open source TPM drivers and toolkits, and supports daily management, setting and application of TPM.
TCG VPWG working group
TCG organization established the TCG Virtualized Platform WG(VPWG) formally in August, 2007. It focuses on regulating the new trusted virtualization computing platform that built on independent physical platform, and plans to produce a series of specifications, including:

★ Specification for trusted attributes of virtual trusted computing platform
★ Interface specification for expressing trusted attributes of virtual trusted computing platform
★ Trusted attributes specification for virtual trusted computing platform’s host machine
★ TPM function and interface specification for virtual trusted computing platform
★ Attributes specification for virtual TPMs
Key technologies of virtual trusted platform
Main Points

◆ Extension of trust chain
◆ Remote(Recursive) attestation
◆ Migration authorization and vTPM’s migration
1. Extension of trust chain

In the VPWG specification, any component that contains local TPM information is not allowed to migrate during the migration process, but vTPM, which is generated in virtual platform, is permitted to migrate.
In order to keep original security attributes of hardware TPM, we need to carefully manage signature key and certificates, and extend trust chain from physical TPM to every virtual TPM instance.

Selecting a solution of generating EK certificate is a key point to realize the trust extension of virtual TPM instance. There are several alternative solutions below:
Solution 1

- Send EK certificate of a virtual TPM instance to AIK certificate of hardware TPM through a link to create a certificate chain. It means that private CA issues certificate according to its EK. The advantages of doing this is to maintain the conventional way to get the AIK certificate by submitting a EK certificate to CA for evaluation.

Figure 6: Certification of Endorsement Key using an AIK
Solution 2

- Issue AIKs certificate for virtual TPM instance according to AIK released by hardware TPM, rather than using virtual EK certificate. As a result, the certificate chain binds virtual TPM instance’s AIK to hardware TPM’s AIK, and thus binds to the host system.

- One of the advantage is that once the hardware TPM’s AIK is released, AIKs of guest VMs’ virtual TPM may be soon proven. A link with the hardware TPM platform is established. The drawback is that we need to change the conventional way of getting AIK certificate from CA.
Solution 3

◆ Issue EK certificate for virtual TPM instance through a local CA.

◆ Since no certificate chain is established with hardware TPM certificate, so that the vTPM EK certificate is not bound to the hardware platform. It is common to see that local EK CA is available if hardware TPMs do not provide platform certificate. Compared with solution 2, its benefit is not changing the conventional way of obtaining a AIKs certificate.
Solution 4

- Manufacturer replaces hardware TPM with a secure coprocessor and then link EK certificate with the coprocessor, and makes remote side establish secure property for vTPM running environment. All the previous listed methods to generate the certificate chain are available since manufacturer provides EK certificate.

Figure 2: vTPM running inside a Secure Co-processor.
We must weigh the advantages and disadvantages when we choose a method to authenticate EKs. For example:

- If a strong connection is required between virtual TPM and hardware TPM, we should choose solution 1, 2 or 4.

- It is necessary to invalidate linked certificate and key after migration, in order to rebuild a trust chain with new hardware. According to solution 2, we can provide a better support for dynamic environment, because there is only a virtual environment AIK that needs to be created and proved. And the first solution requires to join EK certificate into a revocation list and create a new EK.
2. Remote(Recursive) Attestation
Research on Remote Attestation

◆ **Remote attestation based on binary code**

◆ **Remote attestation based on software source code**

◆ **Remote attestation based on attribute:** Remote attester cares about whether the target platform owns specific secure properties or not. We can define property of a platform as a need to a specific security requirement, such as the platform should follow privacy protection protocol, support mandatory access control, or have strong isolation control over process.

◆ **Remote attestation based on model:** Policy model of system can have a predetermined behavior for a specific demand, so that remote side only needs to verify the behavior of system security model rather than individual software or hardware.

◆ **Remote attestation based on program behavior:** It first determines the dependency relationship of application at system-level and language-level. The former represents the OS layer, which are necessary software components(data, runtime library etc.) to support a program to run correctly. The latter indicates the instruction layer, which are necessary codes that support a piece of code to run correctly. By analyzing and locating these two types of dependency relationship, remote side can simply rely on monitoring programs to infer the whole program’s correct conversion from input to output by detects the correct conversion of each dependency from input to output, that whether the program is running correctly or not.
Remote attestation based on binary code

Security Attribute Certificate

Hash value of Binary Code
Software Security Attribute

Judge security attribute

Binary Code

Attester

Report binary value
Remote attestation based on software source code
3 Migration Authority and vTPM migration

Secure migration mechanism based on trusted third-party--MA (Migration Authority)
The process of migration should support the safe migration of vTPM instances between physical platform and protect data’s confidentiality and integrity in vTPM instances, and rebuild the trust chain in new platform.

Instance of secure migration protocol based on vTPM
1. TCG_1_4_Architecture_Overview.
   http://www.trustedcomputinggroup.org/resources/tcg_architecture_overview_version_14
2. TPM MainP1DPrev103.
   http://www.trustedcomputinggroup.org/developers/trusted_platform_module
3. TPM MainP2Structrev103
   http://www.trustedcomputinggroup.org/developers/trusted_platform_module
4. TPM MainP3Commandsrev103
   http://www.trustedcomputinggroup.org/developers/trusted_platform_module
5. TCG PC Specific Implementation Specification.
   http://www.trustedcomputinggroup.org/developers/pc_client
6. TCG_Generic_Server_Specification_v1_0_rev0_8
   http://www.trustedcomputinggroup.org/developers/server
   http://developer.berlios.de/projects/tpm-emulator
Bibliography


16. 沈昌祥. 可信计算平台与安全操作系统. 网络安全技术与应用. 2005.4

17. 周明天,谭良. 可信计算及其进展. 电子科技大学学报. 35(4):686-697, 2006.8
Bibliography

18. 沈晴霓, 卿斯汉. 基于可信计算平台的安全操作系统及相关标准研究报告. 中国科学院软件研究所，2006.12
19. 秦宇. 可信虚拟平台安全机制研究. 博士论文. 中国科学院软件研究所, 2008.10
20. 李昊, 冯登国等. 可信密码模块符合性测试方法与实施. 武汉大学学报(理学版), 55(1):31-34. 2009.2
22. 邹德清, 羌卫中, 金海. 可信计算技术原理与应用. 北京：科学出版社，2011
THE END, THANK YOU!