Virtual platform of trust, a state of the art

Eléonore Hardy[1][0000-0002-6065-5856], Alexis Ulliac[1][0000-0001-7936-316X] and Paul Varela[1][0000-0001-9953-5360]

[1] Thales Secure Communications and Information Systems, 92230 Gennevilliers, France
firstname.lastname@thalesgroup.com

Abstract. The use of virtualized and cloud environments has grown tremendously during the last decade and raised new threats. As privacy and security needs increased, the usage of a Trusted Platform Module (TPM) became trendier. This technology, consisting of a passive crypto coprocessor installed on most modern hardware, helps to provide trust to users. Adapting TPM technology to a virtualized environment brings new security constraints and benefits. This paper presents a state of the art of the virtual TPM technology and how it can improve security and trust on virtualized environments. After an overall presentation of TPM and vTPM principles, this article presents specific architectures, security challenges and solutions. To conclude, standardization initiatives are addressed and a way forward at national level to enhance vTPM security for critical activities is presented.

Keywords: Protection, Virtual platform, TPM, trusted computing, Virtual TPM.

1 Introduction

The Trusted Platform Module (TPM) has been wildly used in hardware architectures for over 15 years. It is a crypto coprocessor embedded in most IT hardware equipment; however most users, even security professionals, use it without noticing [1]. Indeed, TPM is still known as the controversial technology behind digital rights management (including data and license protection preventing illegal copy of software or video games) which has probably slowed down its adoption by general public applications for security usage. In particular, with the growth of virtualized and cloud environment, the TPM technology can bring trust into services, where all third parties are unknown to the end user, and substantially improve applications security.

This article introduces TPM technology and how it can significantly enhance security for virtualized environments. Then, it goes deeper into Virtual TPM (vTPM) design architecture and debates on its security implementation challenges. Before concluding, TPM standardization efforts and virtual TPM technologies are presented, including TCG (Trusted Computing Group) and ISO initiatives. Finally, a way forward is proposed for enhancing vTPM security implementation and usage for national critical activities.

The vocabulary used in this article is referring to the TPM 2.0 specification [8].

2 Trusted Platform Module

2.1 Basic usages and use cases

TPM is a passive crypto coprocessor implemented as hardware or even software [16], depending on requirements. It supports integrity and confidentiality functions contributing to the following applications [1]:

- Secure Boot for auditing the integrity of Operating System (OS) boot process;
- Virtual Private Network (VPN) with device and user authentication;
- Key storage for disk or file encryption;
- Email encryption, signing and authentication;
- Web browsers. e.g. user authentication for logging into online banking;
- Direct Anonymous Attestation (DAA);
- Any application that can take advantage of TPM commands and its TSS (TPM Software Stack) library [10].

Measured Boot. Here is some detail about one of the main functions of the TPM which is to provide a support for a Secure Boot.

It helps to protect the system from rootkits and other malware. Measured Boot checks each start up component, including the firmware, all the way to the OS stores this information in the TPM as defined in **Fig. 1**.

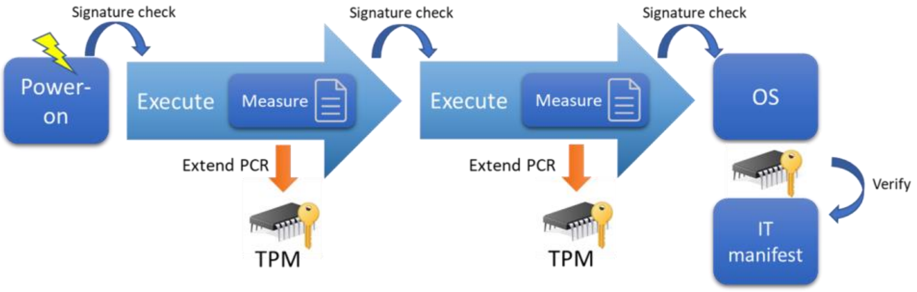

Fig. 1. Measured Boot sequence representation

The Secure Boot is a generic term to designate either a Measured Boot or/and a Trusted Boot [11]. Only the Measured Boot is explained here because it provides measurements that are stored in the PCR (Platform Configuration Registers) of the TPM in order to have evidences to insure that the boot occurred as expected in terms of trust and security. A measurement is defined as below:

PCR: = hash{ [PCR] | "Integrity Metric of the next component" }.

In other words, the new value of the PCR is the digest of the previous PCR value concatenated with the integrity metric of the next component.

2.2 TPM Architecture

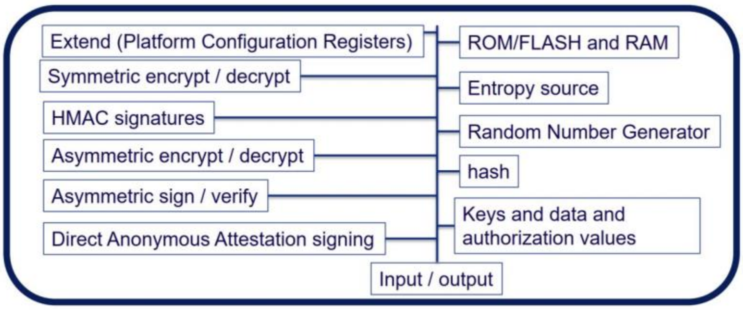
Extend (Platform Configuration Registers)
Symmetric encrypt / decrypt
HMAC signatures
Asymmetric encrypt / decrypt
Asymmetric sign / verify
Direct Anonymous Attestation signing
ROM/FLASH and RAM
Entropy source
Random Number Generator
hash
Keys and data and authorization values
Input / output

Fig. 2. Trusted Platform Module (TPM) 2.0 architecture [28]

TPM main cryptographic functions are the following:

- Random number generator: prevents the platform from relying on software pseudo random numbers generators to generate cryptographic keys (except for the primary keys generated from seeds in 2.0);
- Symmetric and asymmetric cryptographic keys generator;
- Encryption/decryption.

It also provides secure storage capabilities in two memory types, Volatile and Non-Volatile memory (NVRAM) for the following elements:

- Endorsement Key (EK): this is a unique key generated by the trusted TPM manufacturer (eventually signed) in persistent memory;
- Primary Storage Key (known as Storage Root Key in TPM 1.2): this is a root key of a key hierarchy for key derivation process and stored in persistent memory;
- Other entities, such as Indexes, Objects, Platform Configuration Registers (PCR), Keys, Seeds, counters, etc. ([1] Chapter 8).

3 Virtual Trusted Platform Module (vTPM)

The growth of virtualized environments and cloud services usage, including critical systems, made virtual TPM the natural successor to physical TPM (pTPM), as virtual environments require the same level of security as physical ones. Indeed, most of the time, it can be considered that virtual environments are not aware of their underlying virtual or physical platform, including for TPM usage.

3.1 Use cases where TPM support is required in virtualization

Most usages of TPM in virtualized environment can be the same as on a physical host. The common needs for TPM seen in the industry for virtualized environments are the following:

- Offer TPM support to multiple Virtual Systems (VMs or containers);
- Offer TPM support in case of VMM (Virtual Machine Manager) nesting;
- Prevent the virtual machine to have direct access to the memory location where are stored crypto materials at VMM level (keys and certificates) for encryption or signing [7];
- Enable standard services based on TPM such as remote attestation of firmware and guest operating systems [7] on virtualized environments;
- Support virtual disk encryption by the VMM.

3.2 Why is a virtual TPM required?

TPM specification [8] essentially requires the chip to be associated to only one system such as:

- Only one user platform can be enrolled in a TPM;
- It is not possible to have a binding from multiple VMs to a pTPM;
- A VM has a unique and specific lifecycle that is not compatible with hardware TPM chip specification [8] such as suspend and resume operations;
- By definition, a virtual system is transient which needs to be adapted to the persistent key storage principle;
- The possibility to store multiple users on a TPM would require modifying the TCG specification that is tailored to reduce the cost of production of TPM chips by limiting embedded memory requirement and processing power. In addition support to multiple users would be limited to a fixed number anyway.

A specific instantiation of TPM specification has been written by TCG in order to answer to those vTPM constraints [4]. The production of a second version is ongoing to reflect most recent TPM evolutions.

3.3 How to virtualize a TPM?

The TCG, under the VPWG (Virtualized Platform Working Group) described in §7.1, identified several ways to provide TPM interface to Virtual Systems.

Sharing of a physical TPM. This involves reserving PCR registers for each virtual machine: for example, PCR 1 to 4 for the first VM, 5 to 8 for the second, and so on. This implementation is not feasible for several reasons: first, it is not allowed to have a different key hierarchy by VM (which represents a non-compliance with the TPM specification); moreover, it simply does not allow scaling up, because of the physically limited number of PCRs.

Managing virtualization directly in the chip. This results in the introduction of a notion of contexts in the TPM chip, which is a sequence of parameters describing the exhaustive of a single state of the chip, being specific to a machine Virtual. Each time an order is sent to the chip, it is indicated which machine is concerned by indicating the identifier of the context.

In practice, it does not hold either a massive implementation, because the resources of a TPM in storage are necessarily physically limited and because it would imply an expensive solution to implement.

Virtual TPM. This last solution has been retained by the VPWG. It consists of developing software emulated TPMs, which are seen by VMs as virtualized hardware. Since those vTPMs offer all the functionalities of a pTPM, applications that run inside a VM do not need any modification to interact with them.

However, in order to insure trust in those virtual TPMs (vTPMs), the pTPM is used to guarantee the hypervisor boot chain and the integrity of the code and context data of these virtual TPMs, with the aim of degrading as little as possible the expected level of security of this component.

4 vTPM architectures

This section presents the different possible architectures and security challenges involved in implementing a vTPM service into a VMM.

In order to establish a root of trust, bindings need to be insured. First, between the different instances of vTPMs and the pTPM. Then, between the virtual systems and the vTPM. Those different bindings are described below (**Fig. 3**). It should be noted that low levels should not trust higher levels but always check that they have not been compromised.

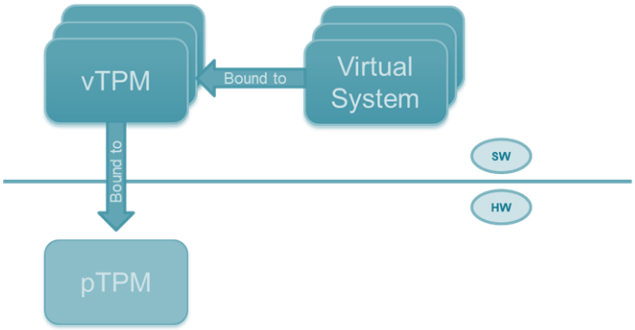

Fig. 3. Link between pTPM, vTPMs and Virtual domains

4.1 Binding between vTPM and pTPM

Binding vTPM(s) with pTPM. There are different methods [29] to protect a vTPM and ensure a root of trust. One of them is to bind vTPM(s) to the pTPM by using it to seal the NVRAM file which is the virtual replacement of the physical NVRAM to hold secrets in a virtual environment. The downside is that it cannot insure a run-time protection of the vTPM as the NVRAM file remains accessible.

Table 1. Binding between vTPM and pTPM pros and cons

Pros	Cons
- The vTPM secrets are sealed by the pTPM when at rest which provides a strong binding and protection for the vTPM to the hardware. - It is possible to guarantee the chain of trust from the pTPM to the Virtual Systems.	- For Virtual Systems migration or backup restoration, it is not possible to deploy the vTPM to another VMM without rebinding with the pTPM. This can add delay to this process.

Note: Intel SGX (Software Guard Extensions) [29] can enhance the protection of vTPM. Nevertheless, SGX and TPM are not providing trust in the same way. TPM chip is tamper-proof whereas SGX is not because it is based on the CPU instructions and not on a dedicated secure chip with a proper root of trust. This does not prevent a compromised hypervisor or a malicious administrator to alter the NVRAM file stored at rest.

vTPM without binding to pTPM. In this implementation the Virtual TPM is fully emulated and totally detached from the pTPM present in the physical host.

Table 2. vTPM without binding to pTPM pros and cons

Pros	Cons
- No pTPM are required on the host as the vTPM is completely emulated. - Migration easy to implement as the vTPM is not linked to hardware (no crypto binding). - All the VMs present on the same host can possess its own vTPM.	- The vTPM secrets are stored in NVRAM file. To avoid security issues, these secrets need to be encrypted but in current solutions the level of protection can vary from a vendor to another. - The absence of pTPM cannot permit to attest that a vTPM has not been tempered or replaced.

Pass-through to pTPM (1 to 1). In a pass-through configuration the TPM offered to the Virtual System is the actual pTPM of the physical host. The vTPM is mapped with the pTPM.

Table 3. Pass-through to pTPM pros and cons

Pros	Cons
- All the functionalities and measurement present in the pTPM can be reflected on the vTPM. - The Virtual System secrets are protected by the pTPM as a hardware-based solution. - Useful when prototyping a system or application in a VM that will run on a non-virtual environment in operational phase.	- As the mapping is one to one, only one instance of Virtual System on the host can use the pTPM. - Migration of Virtual Systems is not supported. Indeed, pTPM registers and NVRAM cannot be extracted from the pTPM and moved to another pTPM.

4.2 Binding between Virtual Systems and vTPM

This section presents the binding strategy of Virtual Systems to vTPMs. To insure isolation of data managed by vTPM, an instance of virtual TPM shall be used by only one Virtual System (VM or container). The **Fig. 4** presents this relation for binding Virtual Systems to vTPM.

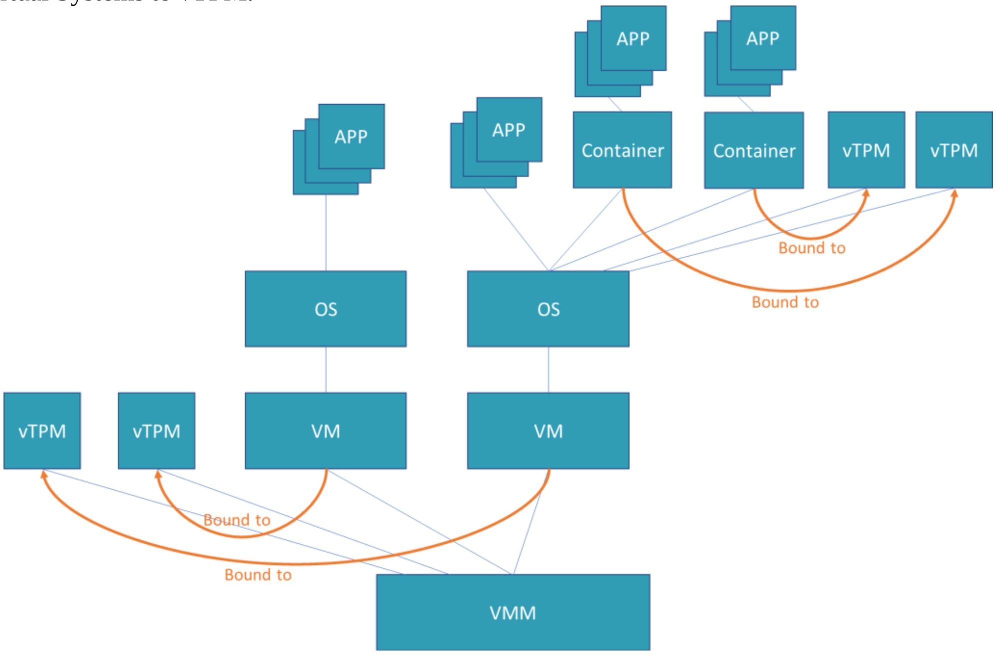

Fig. 4. Relation between vTPM and Virtual Systems

vTPM relation with Virtual Systems. The vTPM serves as a root of trust for a virtualized system which can be hypervisor-based (with VMs) or container-based.

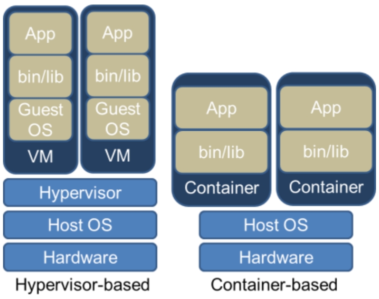

Fig. 5. Hypervisor-based vs container-based virtualization

A container is a lightweight, standalone, isolated and executable package of software where all dependencies required to run an application are included. This means that they are really useful for virtualization.

There are 2 types of Virtual Systems [12]. A Virtual Machine is an independent platform as the guest OS is hosted by a VM that is managed by the hypervisor. A Container is platform-dependent and is seen as applications that depend on the kernel of the host [12]. This requires a specific implementation for containers as having a vTPM included in the operating system kernel [13].

Depending on the technology used (virtualization or containers) the binding to the vTPM is managed differently. A VM has a binding managed by a VMM whereas a container has a binding managed by the OS. To achieve separation, a VMM uses hardware-assisted virtualization and extended/nested memory virtualization. For Containers, the OS uses process separation and name spaces to achieve separation.

As for VM, Trusted containers implementation [26] follows security checks using vTPM to insure a chain of trust using measurements such as:

- Boot-time integrity of Host and container engine;
- Insure that container images are not tampered prior to launch.

Limitations. Containers use pure software-based solutions to provide isolation insures that no process belonging to a different container can access software TPM's state in another container. It is less trusted and secure than the security properties offered by virtualization mechanisms. An attacker could gain access to other containers in case of a lateral move from the container to the host.

5 vTPM implementation challenges

Implementing a vTPM can bring new security challenges compared to pTPM due to its software and virtualized environment. Indeed a vTPM is a full software solution where crypto materials are not protected by an anti-temper hardening in a physical TPM. The following challenges need to be handled in order to avoid compromising secret material.

Secure key storage/management in a vTPM. The part of a vTPM memory (NVRAM) that requires being persistent cannot be stored in a physical sealed hardware TPM. However, it shall be stored securely in a storage unit when at rest. A vTPM developer should take care of carefully manage the memory implementation in order to prevent leakage of sensitive crypto material and keys at rest (when vTPM is shutdown). For instance, if the computer hosting the VMM is put into power-saving mode, the vTPM process in the host RAM shall not be saved to the hard drive.

Reset of volatile memory in case of virtualization platform (vPlatform) reboot. When a hardware platform reboots or is shutdown, the data physically stored in the volatile memory is not erased; it slowly decays over time as the electrons discharge from the memory. In the meantime it is possible to read and get access to temporary remnant data in memory especially if the system is rebooted to another OS.

This issue is addressed by the TCG specification [14] but brings specific challenges to vTPM. Indeed, all information is stored at some point in the memory, even derived keys used by VMs that shall not survive shutdown. Software and hardware implementation shall take care of a careful critical volatile data erase in memory during reboot and shutdown operations [3].

Continuity of service for VM migration and backup. During live migration VMs are moved from one physical platform to another without disconnection. In order to speed up the process, the running system is duplicated and memory pages are gradually copied onto the new system. The switch from the old to the new physical platform is only performed once all pages have been copied. This behavior creates issues for the vTPM implementation and requires deciding about the possibility of cloning vTPM.

Indeed, pTPM security is based on unicity. But to insure security during a live migration two possibilities exist for the vTPM:

- The first one is to clone the vTPM in a short period of time. This solution is close to the usual behavior of VMs but has to be performed carefully to keep vTPM secure.
- The other one is to avoid cloning and create a new vTPM on the new system then migrating information from the old TPM to the new one.

Hypervisors protection. In order to protect the content of a Virtual Machine various elements such as the virtual disk, snapshots and RAM are ciphered to prevent an unauthorized access when the Virtual Machine is in a cold state.

A compromised hypervisor could request the encryption key of a virtual machine disk in order to cipher/decipher its content. Various solutions of virtualization are now including a control of the hypervisor security state before giving access to the encryption keys as it is explained in [15].

Protect VM against administrators of VM. With the constant growth of virtualization, lots of systems, even critical one, are hosted on cloud services or given to suppliers infrastructures.
As detailed in [15], those choices of infrastructures create new threats which need to be studied. Instead of only focusing on threats between VMs, or from VM to host; VM owner should worry about threats coming from the hosting environment to the VM. In this way, the integrity and confidentiality of VMs need to be protected against:

- Storage administrators: they can have access to VMs disks, so they may extract data or modify it;
- Backup systems administrators: as storage administrators, they can access to disks;
- Network administrators: they can access infrastructure traffic and then obtain sensitive information on VM;
- Hypervisors administrators: they can access to OS, RAM and disk data;
- People with physical access to hypervisors: they can modify hardware.

6 vTPM solutions

This section presents a summary of solutions for vTPM provided by virtualization solution editors and cloud service providers.

6.1 Shielded VM in Microsoft Hyper-V

Hyper-V. Microsoft is an active member of the TCG where it acquired knowledge that has been implemented in their solution, Hyper-V, a virtualization hypervisor. It integrates the following elements described in [21]:

Table 4. Hyper-V components

Name	Description
Host Guardian Service (HGS).	Measure the health of a Hyper-V host and release keys to healthy Hyper-V hosts when powering-on or live migrating shielded VMs.
Guarded Host	A Hyper-V host on which shielded VMs can run.
Shielded-VM	A virtual machine that can only run on guarded hosts and is protected from inspection, tampering and theft from malicious fabric admins and host malware.
Guarded Fabric	Fabric of Hyper-V hosts and their Host Guardian Service that has the ability to manage and run shielded VMs.

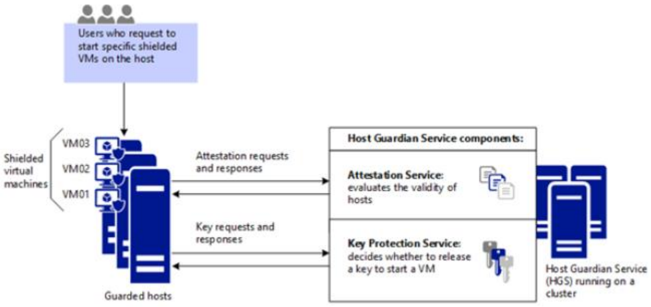
Users who request to start specific shielded VMs on the host
Shielded virtual machines
VM03
VM02
VM01
Guarded hosts
Attestation requests and responses
Key requests and responses
Host Guardian Service components:
Attestation Service: evaluates the validity of hosts
Key Protection Service: decides whether to release a key to start a VM
Host Guardian Service (HGS) running on a cluster

Fig. 6. Host Guardian Service

A Host Guardian Service (HGS) checks whether a hypervisor is safe enough and authorized to have access to a VM. Guarded host does not have the keys needed to power on a shielded VM unless it can provide the current health certificate and the encrypted secret (a Key Protector or KP) to the Key Protection Service (KPS) of the HGS. The secret is encrypted using other keys that only KPS knows.

This implies that the guarded host has already been declared as an authorized host in the shielded data file (PDK file) that contains the NVRAM variables and VM secrets, such as the trusted disk signatures and RDP certificates.

The proposed implementation allows protecting the VM against unauthorized access from a malicious administrator, compromised hypervisor or tampering with a combination of features such as Secure Boot, Bitlocker encryption, vTPM.

Migration. VM migration is a challenge to address when it comes to vTPM. Not only for cloud computing but also for every usage of VM where it is needed to change the host. It is needed to maintain the binding between the VM and the new system host which needs to be authorized to run the VM. For Microsoft technology, an example is described in [19] and [20] regarding importing the destination system’s guardian information on the source host. Nevertheless, in absence of a Protection Profile for vTPM, at the moment this article is written, every solution provider will propose a different solution. Hyper-V allows encrypting virtual machines in saved state and live migration traffic for hot migration [21].

Limitations. With the Guarded Fabric and shielded VM Microsoft implements a solution to protect the hosts and the virtual machines against malicious administrators of compromised hypervisors in order to mitigate the actual threats on vTPM. This adds a level of complexity for the trusted hosts and fabrics management in case of disaster recovery plan and migration or any environment that needs to update the access rights and ownership in the shielded data file. Also, the entire security model relies on the HGS security. An improper implementation or management of this element can compromise the system.

6.2 VMware vSphere

vSphere is VMware's cloud computing virtualization platform to create and run hypervisors and virtual machines.
The hypervisor is hosting the vTPM and is ensuring the storage of encryption keys in the NVRAM file which is encrypted by the vSphere VM encryption key. It is also encrypting the virtual disks. It is possible to backup or migrate VM which possesses a vTPM if the backup includes the NVRAM file.

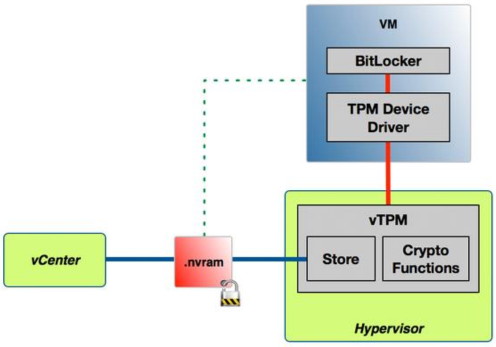

Fig. 7. vSphere architecture

Limitations. vSphere implements the vTPM at the hypervisor level. There is no binding of the vTPM to the pTPM which means there is no root of trust for the VM down to the hardware. In case of a compromised hypervisor there can be an unauthorized access to the NVRAM file with the encryption keys and to the encrypted disks of the VM because everything is stored in the hypervisor. As NVRAM file (VM Home file) and virtual disks are encrypted, this is covering only the protection of the data at rest.

6.3 Google shielded VM in Google Cloud Platform (GCP)

Google Cloud Platform is a cloud service provider proposing an implementation of vTPM that can be used by virtual machines in PaaS (Platform as a Service) mode.

Among the main cloud services providers (Microsoft Azure [24] and Amazon Web Services), as of today, Google is the only one to publicly propose vTPM and related services to their customers.

Virtual Machines protection. The vTPM provided by GCP is compliant with the TPM 2.0 specification and validated by TCG as a recognized TPM implementation from TCG’s approved list of vendors and FIPS 140-2 L1 (physical security not insured as it is a fully virtual cryptographic solution).

As part of its service, GCP provides built-in disk images supporting vTPM for Linux as well as Microsoft Windows. The technology is limited to PaaS service on VM. By design those disk images support the measured boot. Failure to pass the boot startup integrity process is reported to the Google Stackdriver logging tool (logs management tool). It also provides support to all standard features of TPM 2.0 such as digital signature, secrets storage, random number generator, etc.

Hypervisor protection. To establish a hardware root of trust for its customers, Google developed the Titan chip which is a TPM like proprietary design not based on TCG specification. It provides the main following functions for cloud infrastructure security [25]:

- Trust machine unique identity by cryptographic attestation (each hardware can be uniquely identified by the infrastructure management);
- Tamper resistant, events logging and monitoring;
- First instruction integrity;
- Trusted implementation and design of the chip, hardware and software;
- Verification and authentication of every piece of hardware in datacenters;
- Designed and produced within google facilities.

The main difference with TPM is that Titan chip is an active technology where TPM is used to support crypto functions. Titan is mainly able to check the integrity of the system firmware flash at an early hardware startup and provide an active response such as an alert, a stop to the system or detect and drop/rewrite illegal SPI commands (Serial Peripheral Interface data bus used to connect Titan to hardware).

The main limitation of this technology is that events from Titan technology are not shown to Google infrastructure management, internal security events are not displayed to clients that can affect the security of VMs.

Limitations. In general, when a third party is involved in a service provision, the trust in the cloud solution is limited by the trust in the service provider:

- Protection of data can be subject to local regulations;
- Not possible to attest the root of trust and integrity of the VMM in Titan chip used in Google hardware infrastructure. Basically, the service provider can hide potential compromising to the customer;
- Not possible to know the level of protection of NVRAM file used by the vTPM;
- At this point, not possible to prevent tempering from service provider administrators or compromised infrastructure;
- Still linked to a Google account that can be attacked if security policy not properly followed by the customer administrator deploying VMs.

6.4 Xen

The Open Source Xen solution proposes an implementation of vTPM up to TPM 2.0 specification [30]. Each VM from a Xen DomU level gets its own vTPM domain which has its NVRAM sealed by the pTPM. If the process of each Xen domain is trusted then the root of trust can be extended from up to the VM. As each vTPM is running its own domain, VMs do not share their vTPM, thus the risk of compromising of secrets by sealed vTPM between VMs is limited.

Limitations. Xen offers a strong binding of its vTPM to the pTPM. As it strong binding to the pTPM strengthens the chain of trust, it can make backup and migration more complex and affect availability of the system.

7 TPM Standardization

7.1 Presentation of TCG and VPWG

In order to meet the need of standardization on TPM domain, the Trusted Computing Group (TCG) has been an active actor on the subject [5]. The TCG addresses several TPM domains of application within different working groups that are Cloud, Cyber Resilient Technologies, Device Identifier Composition Engine (DICE) Architectures, Embedded Systems, Industrial, Infrastructure, Internet of Things (IoT), Mobile, Network Equipment, PC Client, Regional Forums, Server, Storage, TPM Software Stack (TSS), Trusted Network Communications, Trusted Platform Module (TPM), Virtualized Platform (VPWG).

The VPWG within the TCG is the most relevant in the scope of this paper.

As defined in [4], this working group is in charge of developing a Virtualized Trusted Platform Architecture Specification that defines a general architecture, terminology and envisioned set of deployment models for what capabilities virtualized trusted computing platforms are expected to offer.

Moreover, it defines how to bind the vTPM to the pTPM and how to bind a Virtual System to a vTPM. This document does not focus on how a particular design or implementation of a virtualized trusted platform should operate on specific hardware (e.g. what functions are done in hardware, hypervisor or VM protection model).

7.2 Relation with ISO/IEC JTC1/SC27

TCG is not the only entity involved in TPM standardization. Another actor is the ISO (International Organization for Standardization) which is a worldwide federation of national standards bodies (ISO member bodies). ISO/IEC JTC 1 is a joint technical committee and its purpose is to develop, maintain and promote standards in the fields of information technology (IT) and Information and Communications Technology (ICT). Under this subcommittee the SC27 is focusing on IT Security techniques.

ISO/IEC JTC1 SC27 is organized in 5 working groups:

- WG 1 Information security management systems
- WG 2 Cryptography and security mechanisms
- WG 3 Security evaluation, testing and specification
- WG 4 Security controls and services
- WG 5 Identity management and privacy technologies

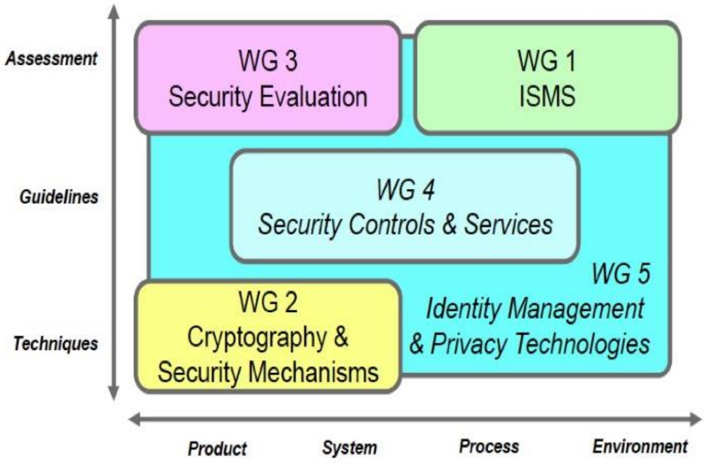

Fig. 8. Working groups of the committee ISO/IEC JTC1/SC 27 [18]

The WG4 covers the services and applications needed to implement the controls and control objectives as defined in ISO/IEC 27001:2005~~,~~ covering aspects related to security breaches, intrusion detection, incident management or business continuity of information systems. The WG4 is in charge of the ISO/IEC 27070, Information technology – Security techniques – Security requirements for virtualized roots of trust. As defined in [6], this standard proposes technical requirements for the establishment and operation of the virtualized root of trust. There is a relation between the ISO and the TCG to allow the VPWG to review the ISO document.

7.3 National design and security certification

The security concept and added value of vTPM technology is undisputable. However, throughout this article, it is shown that limitations exist, especially regarding implementation and design strategies. TCG work groups, as other expert groups, are driven by institutional and private interests contributing to the work of specifying security solutions.

Indeed, virtualization and cloud technology are a reality among solutions used by businesses over the world, even on critical sectors. As an outcome of this paper, it is advised for national authorities to elaborate recommendations and a protection profile for security design of vTPM solutions, integration guidelines and a way to certify it for critical business or public sector usages.

8 Conclusion

In this paper TPM and vTPM technologies, related challenges and design problems are presented. Currently, different vTPM providers implement their own architecture and solutions to answer the security needs of their customers.

Actual implementations have different level of maturity regarding vTPM integration. The biggest challenge is to guarantee the chain of trust for the Virtual System without compromising the performances and availability. Indeed, the growth of Virtual Systems relaying on a strong binding down to the pTPM, to insure the root of trust, creates a strong need to find a solution that balances performance (I/O) and security level, due to hardware limitations.

As seen in limitation of current solutions and implementation challenges, even if a software vTPM cannot bring the same level of security as physically protected crypto material in a pTPM chip, the support of this technology by operating systems and the increasing system administrators' awareness will definitely contribute to tighten the security of virtualized environments

The standardization effort done by TCG and ISO needs to be enforced by the community of experts, including testing organizations, in order to support the development of standards and architectures for vTPM, as done by VPWG, and to allow a common understanding of the technology.

References

1. Arthur, A., Challener, D., Goldman, K..: A Practical Guide to TPM2.0 Using the New Trusted Platform Module in the New Age of Security. Apress open (2015).
2. Cucurull, F. Guasch, S.: Virtual TPM for secure cloud: fallacy of reality? In: RECSI pp. 197-202. Alicante (2014)
3. TCG. Virtualized Trusted Platform White Paper. Version 1.0, Revision 1.01. Trusted Computing Group (2007)
4. Sangster P., Wilson L., Liberty D.: Virtualized Trusted Platform Architecture Specification. version 1.0 Revision 0.26. Trusted Computing Group (2011).
5. TCG Work Groups, https://trustedcomputinggroup.org/work-groups/, last accessed 2019/06/26.
6. ISO: Information technology – Security techniques – Security requirements for virtualized roots of trust. Text for ISO/IEC 3rd WD 27070. International Organization for Standardization (2009).
7. VMWare vSphere Securing Virtual Machines with Virtual Trusted Platform Module, https://docs.vmware.com/en/VMware-vSphere/6.7/com.vmware.vsphere.security.doc/GUID-A43B6914-E5F9-4CB1-9277-448AC9C467FB.html, last accessed 2019/06/15.
8. TCG. Trusted Platform Module Library Part1: Architecture. Family "2.0" Level 0 Revision 01.48. Trusted Computing Group (2016)
9. QEMU Features/TPM, https://wiki.qemu.org/Features/TPM, last accessed 2019/06/26.
10. Arthur, W., Baggaley, B., Challener, D., Cox, M, Fuchs, A., Goldman, K., Repp, J., Tricca, P., Willson, L.: TCG TSS 2.0 Overview and Common Structures Specification. version 0.90 Revision 02. Trusted Computing Group (2018)
11. TCG: Using the TPM to Solve Today's Most Urgent Cybersecurity Problems. Trusted Computing Group (2014).
12. Le Vinh, T., Bouzefrane, S., Banerjee, S.: Convergence in trusted computing and virtualized systems: A new dimension towards trusted intelligent system. In: The 5th IFIP Performance Evaluation and Modeling in Wired and Wireless Networks (PEMWN 2016), Paris, France (2016).

13. Hosseinzadeh, S., Laurén, S., Leppänen, V.: Security in Container-based Virtualization through vTPM. In: IEEE/ACM 9th International Conference on Utility and Cloud Computing, Shanghai, China (2016).
14. TCG: TCG Platform Reset Attack Mitigation Specification. version 1.0. Trusted Computing Group (2008).
15. Galet, JB. : Machines virtuelles protégées. SSTIC, pp. 217-241, Rennes, France (2018).
16. TCG. Trusted Platform Module (TPM) 2.0: A BRIEF INTRODUCTION. Trusted Computing Group, pp. 3 (2015).
17. Popek, GJ.; Goldberg, RP.: Formal requirements for virtualizable third generation architectures". In : Communications of the ACM, Association for Computing Machinery, vol. 17 num. 7, pp. 412-421 (1974).
18. ISO: ISO/IEC JTC 1/SC27 “Working documents”. International Organization for Standardization. Geneve (2009).
19. Migrating local VM owner certificates for VMs with vTPM, https://docs.microsoft.com/en-us/virtualization/community/team-blog/2017/20171214-migrating-local-vm-owner-certificates-for-vms-with-vtpm, last accessed 2019/09/04.
20. Allowing an additional host to run a VM with virtual TPM, https://docs.microsoft.com/en-us/virtualization/community/team-blog/2016/20161025-allowing-an-additional-host-to-run-a-vm-with-virtual-tpm, last accessed 2019/09/04.
21. Guarded fabric and shielded VMs overview, https://docs.microsoft.com/en-us/windows-server/virtualization/hyper-v/learn-more/generation-2-virtual-machine-security-settings-for-hyper-v, last accessed 2019/09/04.
22. Generation 2 virtual machine security settings for Hyper-V, https://docs.microsoft.com/en-us/windows-server/security/guarded-fabric-shielded-vm/guarded-fabric-and-shielded-vms, last accessed 2019/09/04.
23. TPM and Windows Features, https://docs.microsoft.com/en-us/windows/security/information-protection/tpm/tpm-recommendations, last accessed 2019/09/04.
24. Support for generation 2 VMs (preview) on Azure, https://docs.microsoft.com/en-us/azure/virtual-machines/windows/generation-2, last accessed 2019/09/04.
25. Johnson, S., Rizzo, D., Ranganathan, P., McCune, J., Ho, R. : Titan: enabling a transparent silicon root of trust for Cloud, In: HC30, Hot Chips: A Symposium on High Performance Chips. Cupertino, California (2018).
26. Yeluri, R., Gupta, A.: Trusted Docker Containers and Trusted VMs in OpenStack. In: OpenStack Summit. Vancouver (2015).
27. VMware vTPM functionalities, https://docs.vmware.com/fr/VMware-vSphere/6.7/com.vmware.vsphere.security.doc/GUID-A43B6914-E5F9-4CB1-9277-448AC9C467FB.html, last accessed 2019/09/04.
28. Proudler, GJ,: Introduction to Trusted Computing Concepts and the Trusted Platform Module (TPM) 2.0. Trusted Computing Group (2016).
29. Juan W., Chengyang F., Jie W., Yueqiang C., Yinqian Z., Wenhui Z., Peng L.: SvTPM: A Secure and Efficient vTPM in the Cloud. (2019).
30. XenProject Virtual Trusted Platform Module (vTPM), https://wiki.xenproject.org/wiki/Virtual_Trusted_Platform_Module_(vTPM) , last accessed 2019/09/04.