Engineering Village

经检索“Engineering Village 2”， 以下论文被《Ei Compendex》收录(含“Ei controlled terms”的论文)。(检索时间2011 年 4 月 19 日)。

<RECORD 1>
Accession number:20104313326853
Title:Trusted computing based on virtualization
Authors:Bin, Jiang (1)
Author affiliation:(1) Communication Engineering School, Hangzhou Dianzi University, Hangzhou, China
Corresponding author:Bin, J. (jiangbin1980@163.com)
Source title:International Conference on Internet Technology and Applications, ITAP 2010 - Proceedings
Abbreviated source title:Int. Conf. Internet Technol. Appl., ITAP - Proc.
Monograph title:International Conference on Internet Technology and Applications, ITAP 2010 - Proceedings
Issue date:2010
Publication year:2010
Article number:5566640
Language:Chinese
ISBN-13:9781424451432
Document type:Conference article (CA)
Conference name:International Conference on Internet Technology and Applications, ITAP 2010
Conference date:August 21, 2010 - August 23, 2010
Conference location:Wuhan, China
Conference code:81945
Sponsor:IEEE Wuhan Section; University of Wisconsin at La Crosse; Wuhan University of Technology; Wuhan University
Publisher:IEEE Computer Society, 445 Hoes Lane - P.O.Box 1331, Piscataway, NJ 08855-1331, United States
Main heading:Security of data
Controlled terms:Internet - Thermoelectric power
Uncontrolled terms:Information security - Key component - TPM - Trust chain - Trusted chain - Trusted computing - Trusted computing platform - Virtualizations
Classification code:701.1 Electricity: Basic Concepts and Phenomena - 716 Telecommunication; Radar, Radio and Television - 717 Optical Communication - 718 Telephone Systems and Related Technologies; Line Communications - 723 Computer Software, Data Handling and Applications - 723.2 Data Processing and Image Processing
DOI:10.1109/ITAPP.2010.5566640
Database:Compendex

注：
以上检索结果均得到被检索人的确认。

《Engineering Index》检索结果
杭州电子科技大学图书馆 信息咨询部
检索人(签章):
2011 年 4 月 19 日

2010 International Conference on Internet Technology and Applications

Aug. 21-23, 2010 Wuhan, China

Conference Information

Title Page

IEEE

iTAP

Proceedings

International Conference on Internet Technology and Applications (iTAP 2010)

http://www.itapconf.org/2010 **Aug. 21-23, 2010, Wuhan, China**

IEEE Catalog Number: CFP1014I-CDR
ISBN: 978-1-4244-5143-2
Library of Congress: 2009934056

Papers by Author

Bin, Jiang

- Trusted Computing Based on Virtualization

Bo, Song

- A New Operation Model of E-Business

Brancaleone, C.

- A Mobile PC Workstation for Bedside Nursing Activities

Bu, Wei

- The Correlation Empirical Analysis of China's Manufacturing Development Level and Personnel Training

Cai, Jun

- Secure Physical Layer Network Coding: Challenges and Directions

Click on a title to see the paper.

Trusted Computing Based on Virtualization

Bin Jiang
Communication Engineering School, Hangzhou Dianzi University, Hangzhou, China
jiangbin1980@163.com

Abstract—**Virtualization and Trusted Computing are two important approaches in the information security field. How two combine them to form more powerful information security solustions deserves researching. This paper introduces how to virtulize the key components of trusted computing, including the Trusted Computing Platform, the Trust Platform Module (TPM) and the Trusted Chain, by using virtualization technology.**

Keywords-Virtualization; Trusted Computing; TPM; Trust Chain

基于虚拟机的可信计算

姜斌
杭州电子科技大学通信工程学院，杭州，中国，310018

【摘要】虚拟计算和可信计算是处理信息安全问题的两个重要方法，如何将两者结合起来以构成功能更强大的信息安全解决方案是一个值得研究的问题。研究了如何利用虚拟化技术对可信计算的关键部分——可信计算平台、可信平台模块 TPM、可信链 Trust Chain——进行虚拟化。

【关键词】虚拟化；可信计算；TPM；可信链

1. 引言

近年来，随着计算机和互联网的飞速发展，人们对软件的兼容性、可移植性、安全性等性质的需求日益增强。虚拟化技术[1]能够动态组织多种计算资源，从整体上隔离具体的硬件体系结构和软件系统之间的紧密依赖关系，从而实现透明化的可伸缩计算体系结构。同时，虚拟化技术也能够提高系统的安全性和稳定性：由于虚拟化计算的支持，可以使单独的服务、应用执行在独立的虚拟机中，这种有效隔离减少了因为个别软件的安全问题而导致的系统漏洞；另一方面，基于虚拟机平台，我们可以较为容易地实现多级多域多策略的管理机制，从而大大提高系统运行的安全性和可信性。

在 2003 年，可信计算组织 TCG（Trusted Computing Group）[2]成立。可信计算在产业界和学术界迅速掀起了一股热潮。可信计算通过从芯片、主板、BIOS、操作系统等底层软硬件综合采取措施，采用密码、网络安全，信息安全等关键技术，增强终端平台的安全性，从而增强包括网络在内的整个信息系统的可信和安全。在国内，有多家著名的厂商实现了自己的可信计算产品。可信计算技术正深刻地改变着信息安全的现状。

作为引领新一代计算潮流的两种计算技术，如何把两者结合起来综合利用两者的优势，是一个值得探讨的问题。本文研究了如何基于虚拟机技术，对可信计算的关键组成部分——可信计算平台，可信平台模块 TPM，可信链 Trust Chain——进行虚拟化。

2. 基于虚拟机的可信计算

2.1 节介绍了综合利用虚拟机和可信计算技术，构建一个可信计算安全平台； 2.2 研究了如何对可信链进行虚拟化。

2.1 基于虚拟机的可信计算安全平台

2.1.1 平台架构

如何综合利用结合虚拟机和可信计算技术，以实现可信安全计算平台，已经有相关研究[3-5]。综合考虑这些研究的优点，我们提出了如图 2.1 的可信计算平台安全架构。

首先，该架构中最底层是系统所真实使用的商业硬件，包括具有自主知识产权的可信平台模块 TPM（Trusted Platform Module）[2]。TPM 是一个紧密嵌入在主板上的安全模块，它用来提供平台所需要的所有密码服务，是整个系统可信的起点。

其次，在平台硬件之上是可信虚拟机监视器 Trusted VMM (Virtual Machine Monitor)，它用来监控和管理所有的虚拟机运行。在其内部，插入了两个模块。一个是访问控制模块 ACM（Access Control Module）；一个是安全钩子（Hook）。

978-1-4244-5143-2/10/$26.00 ©2010 IEEE

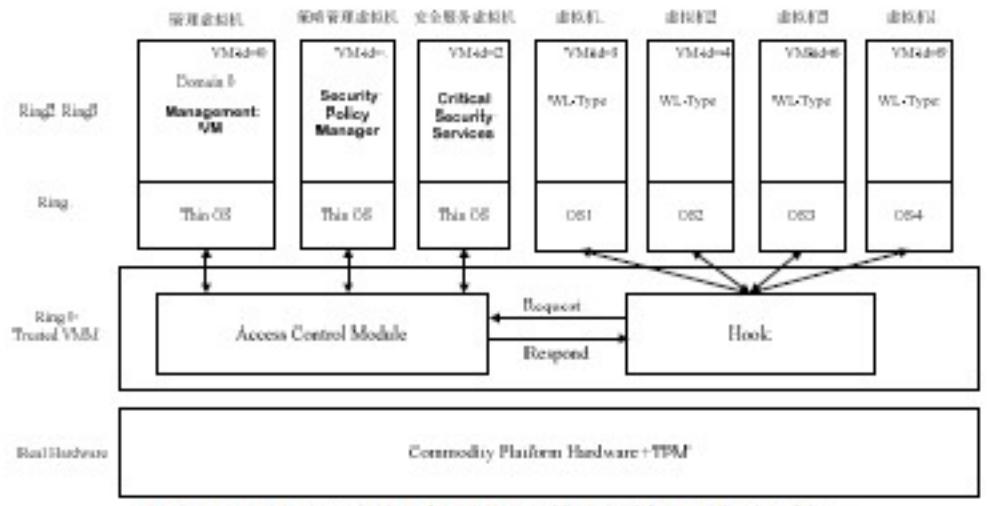

图 2.1 基于虚拟机的可信计算平台架构

任何时候，当某个虚拟机发出访问请求（主客体读写、硬件资源访问，创建回收通道，内存共享等等）以后，该请求将会被 Hook 所钩住；随后，Hook 请求 ACM 检查该访问请求是否能被允许放行。 ACM 通过与 SPMVM 通信，检查访问控制策略，返回响应 True 或者 False 给 Hook。整个 TVMM 运行在 Ring 0 级别。

最后，在 TVMM 之上分别是：管理虚拟机 MVM（Management VM）；安全策略管理虚拟机 SPMVM（Security Policy Management VM）；安全服务虚拟机 CSSVM（Critical Security Service VM）；客户虚拟机 GVM（Guest VM）。其中：

（1）MVM 运行在 Domain 0，并且是 TVMM 所启动的第一个虚拟机。MVM 用来对其它后面所有的虚拟机——包括 SPMVM，CSSVM——进行管理。

（2）SPMVM 用来对所有的策略进行管理。根据用户需求，在 SPMVM 当中可以实现多种安全策略。例如：常用的商业模型 Chinese Wall，Clark-Wilson，以及经典的 BLP，Biba 模型等等。SPMVM 当中的安全策略，不仅包含每个 VM 内部的访问控制，而且包括 VM 与 VM 之间的访问控制。例如，假设图 2.1 当中的虚拟机 2 和虚拟机 4 都运行的是订单服务（Order），因此，它们有可能需要联合协作（Coalition）。在这种情况下，有可能虚拟机 2 和虚拟机 4 同时访问相同的数据。但是，根据 Chinese-Wall 模型，如果恰好虚拟机 2 和虚拟机 4 位于冲突集当中，则该访问请求被拒绝；否则该请求可以执行。

安全策略的制订，由 MVM，SPMVM，ACM 以及下面的 CSSVM 共同制订，并以策略表 PT（Policy Table）的形式存放在 SPMVM 当中。在实现时的多级安全策略的结构体如下：

```
struct MLS_POLICY {
struct subject;          //operation subject
struct object; //operation object
struct sub_vm            //vm info of the subject
struct obj_vm            //vm info of the object
unsigned int sub_level;          //security label of the subject
unsigned int obj_level;//security label of the object
//observe, modify, invoke, share memory, domain management, etc.
unsigned int operations;
unsigned int respond;  //grant, deny, warn, audit etc.
struct MLS_POLICY *pre_policy;
struct MLS_POLICY *next_policy;
}
```

（3）CSSVM 当中运行平台关键的安全应用软件。这些软件服务提供平台最基本和关键的安全服务，因此，它们单独运行在独立的 CSSVM 当中，以实现与其它 VM 的隔离。

由上述架构说明我们可以看到，图 2.1 构成了一个多级、多域、多策略的可信安全计算平台。

2.1.2 虚拟化 TPM

构建了可信计算安全架构之后还需要解决一个问题：实现对硬件 TPM 的虚拟化，以便每个 VM 都有一个独立的虚拟化 TPM（Virtualized TPM，简称 vTPM）可以使用。参考[6]的方法，图 2.2 给出了虚拟化 TPM 的主要过程和思路。

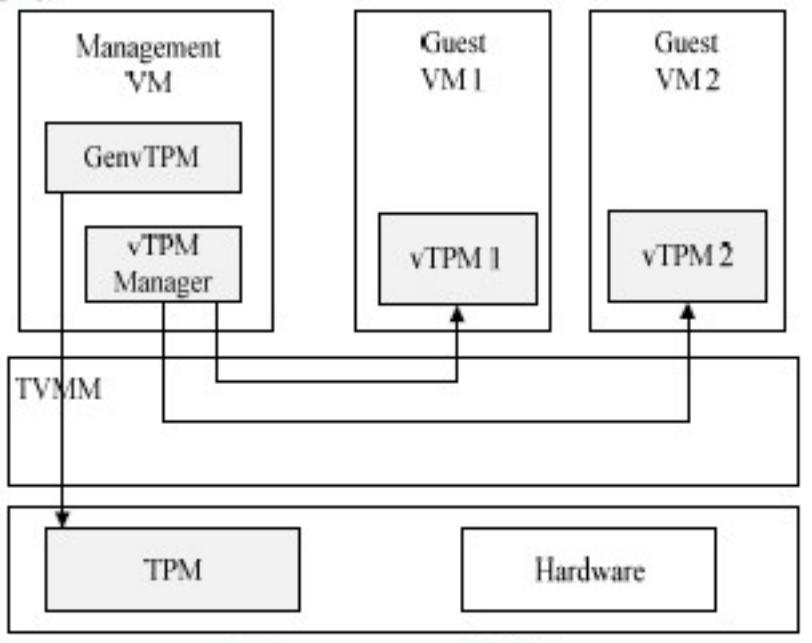

图 2.2 TPM 虚拟化

虚拟化 TPM 采用利用一个总的虚拟化 TPM（图 2.2 中的 GenvTPM）来管理一系列的 vTPM 实例的方法来实现。

首先，GenvTPM 由硬件 TPM 直接提供相应的功能，它以进程的方式运行在 MVM 当中。对于 GenvTPM 来说，它可以像不存在 TVMM 一样而直接使用 TPM 规范[2]当中规定的命令来直接访问 TPM。同时，由于 GenvTPM 的重要性，它要受到严格的访问控制保护。通常，GenvTPM 只能由 MVM 的 Thin OS 所修改，其中 Thin OS 的可信性是由 TVMM 保证的，而 TVMM 的可信性是由

TPM+底层硬件保证的。这种链式的可信性（实际上就是TCG所定义的可信链）在后面2.2有介绍。

任何时候，当一个VM需要一个实例化的vTPM的时候，由GenvTPM创建一个vTPM实例提供给这个VM。当一个VM退出或者被删除的时候，由GenvTPM负责回收与这个VM对应的vTPM实例。所有的vTPM同样也以进程的方式运行在VM当中。

其次，TPM虚拟化另外一项很重要的工作是其内部的密钥体系。对于每一个vTPM（包括GenvTPM），我们都对它们构建其独立的密钥体系，从而方便各个vTPM的密钥使用。根据TPM规范，每一个TPM都要有一个存储根密钥SRK（Storage root key, SRK）用来构成加密的密钥链。另一方面，我们也需要有一个EK（Endorsement key，EK）用来表明TPM的身份。同时，我们还需要身份证明密钥AIK以隐藏TPM的隐私信息，并进行平台身份和配置的证明工作。由于页面限制，EK，AIK和SRK以及其它一些密钥的生成过程这里不再赘述，在TPM规范当中有相关介绍。需要注意的是，上述这些密钥都被硬件TPM当中的一个对称密钥所加密，并存储到硬件TPM的PCR（Platform Configuration Register）当中。这样，任何对vTPM的攻击我们都能够发现。

最后，虚拟TPM内部的PCR与硬件TPM内部PCR的使用不同。由于每个VM都运行在特定的环境之中——例如：每个VM都运行在TVMM之上，并且最终都要使用底层的硬件资源等等——使得在每个VM内部的vTPM当中，除了要记录“该VM的平台配置信息以及软件栈的完整性”之外，还需要记录上述这些环境的完整性。这样，我们规定在每个vTPM的内部，其前8个PCR以只读的方式存储这些“环境的完整性”（由物理TPM所度量得到）；后8个PCR以读写的方式存储该VM平台的的完整性。

2.2 虚拟机上的可信链

TCG定义可信链为：CRTM→BIOS→OSLoader→OS→Applications。其中，CRTM（Core Root of Trust Measurement）通常是BIOS当中的一小块代码。作为可信链的起点，它是绝对可信的。

当平台启动的时候，CRTM首先度量BIOS的完整性。如果BIOS度量是完整的（也就是没有被恶意篡改，其完整性没有被破坏），则信任的边界从CRTM扩大到了CRTM＋BIOS。CRTM＋BIOS随后度量OSLoader的完整性。如果OSLoader也是可信的，则继续度量OS的完整性。这个过程继续下去，直到最后OS无限地度量应用程序的完整性。由于这个可信的度量过程是链式的，因此称之为可信链。

如果以OS为基准，可信链实际上可以分为三个部分：（a）OS启动之前对平台硬件的度量；（b）OS启动；（c）OS启动以后对应用程序进行无限的度量。可信链作为可信计算的关键技术之一，很多研究人员在进行相关研究。其中（b）相对比较成熟，已有成熟的产品[7]。下面重点分析（a）和（c）。

对于（a），文献[8]提出硬件完整性度量的思想来判断硬件的完整性。其基本思想是基于校验和的（Checksum-based）。首先，在一个我们认为系统可信的状态下，预先收集平台所有硬件的完整性信息，包括：硬件名称、接口、参数，SHA-1(如果有的话，例如对整个BIOS代码的SHA-1)以及其他特定信息。然后，把这些信息以标准平台设备表（Standard Platform Devices Table, SPDT）的形式存储到TPM当中，并利用TPM的保护性存储加以保护。接下来，当系统加电以后，以TPM为起点，以一种类似于可信链的链式方式：TPM → Motherboard → CPU → Memory → …… 对平台当中所有的硬件完整性信息进行收集。最后，把收集到的硬件完整性信息与SPDT当中的预期值进行比对。如果两者符合，则说明平台硬件完整性得到了保证；否则平台可能存在漏洞，有可能导致基于硬件的攻击发生。

对于（c），OS对应用程序进行度量通常有五种常用方法，分别是：基于校验和的，基于规范的，基于行为的，基于语义的和基于计算的方法。其中，

（1）基于校验和的方法是一种常用方法，只需要对软件预先存储的“指纹”进行比对即可，不再赘述。

（2）基于规范的方法[9]是指：设计一个接近于C语言的规范语言，用这种规范语言来表达一些相关的攻击行为。这样，就能把可能的攻击行为以及对应的防御和响应动作用这种规范语言表达出来。例如下式就表达了一种隐藏恶意进程的攻击行为：

[for_circular_list I as ListHead.next starting init_task.task.netxt], true=>container(I, Task, tasks.next) in AllTasks; [], true=>runquenue,curr in RunningTasks;

基于规范的方法是通过规范语言描述攻击行为的本质，而不是针对每一种攻击构建一组规范。因此，这种方法能够通过一组规范描述一系列同本质的变种攻击，并且不需要先验的攻击知识。这样，基于规范的方法不仅能够减少工作量，而且能够发现未知攻击。这是其优点所在。

（3）基于行为的方法[10]是指：把整个系统的动态

运行行为可以用一棵进程行为树表达出来，然后通过对进程行为树的研究来判定系统的完整性。

由于每个进程的功能都是由一系列的系统调用所完成的。因此，如果把系统调用看作是进程的一次动作，则一个进程的行为就是一系列的动作（系统调用，包括主体和客体）所组成的。进一步地，如果把单个的进程行为用一个节点来表示，则整个系统的动态运行行为可以用一棵进程行为树来表示。通过研究这棵行为树，第三方能够对本平台的完整性进行判定。

（4）基于语义和基于计算[11]的方法通常通过设计相应的语言（例如：SPA，Spi 等），然后利用该语言对整个系统的动态行为进行建模。再利用相应的数学工具（例如 CCS 和 Pi）对整个系统的行为进行精确的推演和计算，从而得到其可信性。这两种方法的理论要求较高，并且已经有相关的工具。

对于可信计算来说，目前仍然是实践超前于理论。特别对于可信链本身而言，并没有相应的模型论述如何实现它。因此，上面分析了在单机上实现可信链系统所要注意的一些问题和一些常见的方法。下面，看虚拟机环境下如何实现可信链。

实际上，对于每一个虚拟机 VM 而言，它就相当于我们通常所使用的单机系统。因此，在每一个虚拟机 VM 的内部，其可信链的实现可以参照上述的方法。但是，在虚拟机环境下，还有一些额外因素需要考虑。

以图 2.3 中运行 OS1 的 VM（假设其为 VM1）为例，其度量流程为：（1）由 TPM 对底层实际使用的硬件进行完整性度量；由 TPM（包括硬件）对 TVMM 进行完整性度量；由 TPM（包括 TVMM）对 MVM 进行度量；由 TPM（包括 TVMM，MVM）对 SPMVM 度量；由 TPM（包括 TVMM，MVM，SPMVM）对 CSSVM 进行度量；最后是对 VM1 的度量。

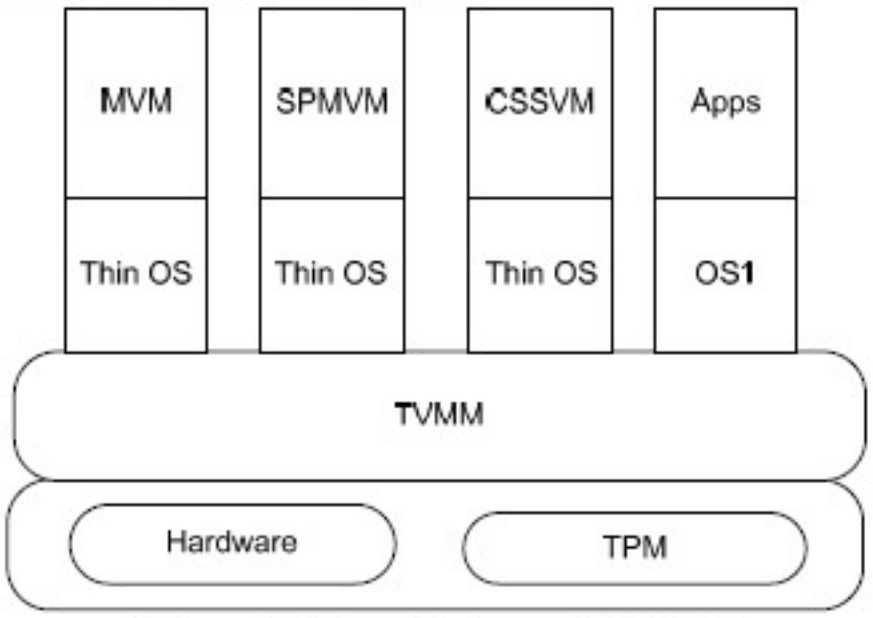

图 2.3 虚拟机环境下可信链的实现

那么，采用类似于可信链的方式，我们把这个过程表达为：

TPM>Hardware>TVMM>MVM>SPMVM>CSSVM>VM1

接下来，对 VM1 的度量则和上述单机内的可信链实现是相同的。唯一需要注意的是：在 VM1 内部度量的起点是 VM1 对应的 vTPM，而不是 CRTM。

3. 结论

虚拟机技术和可信计算技术是当前计算技术的两大热门趋势。如何把两者结合起来，综合利用两者的优势，是一个值得研究的问题。

在本文中，通过结合虚拟机和可信计算，研究了如何设计一个基于虚拟机的可信安全计算平台架构，并进一步考虑了虚拟化 TPM 的问题。同时，我们分析并总结了实现 TCG 定义的可信链的技术，并在此基础上提出了虚拟机环境下可信链的实现方法。由于页面限制，对于实现部分我们没有详谈，在其它文献当中有相关论述。

参考文献

[1] B. Paul, D. Boris, F. Keir, et al., Xen and the art of virtualization, ACM SIGOPS Operating Systems Review, vol. 37, issue 5, Dec., 2003, pp:164-177.

[2] TCG Specification Architecture Overview, Specification Revision 1.2, 28, Apr. 2004, http://www.trustedcomputinggroup.org/downloads/TCG_1_0_Architecture_Overview.pdf.

[3] S. Reiner, V. Enriquillo, J. Trent et.al, sHype: Secure Hypervisor Approach to Trusted Virtualized Systems, IBM Report, rc23511, 2005, http://www.paramecium.org/~leendert/publications/rc23511.pdf.

[4] Q. Huang, C.X. Shen, Y.X. Fang, Security architecture of trusted virtual machine for trusted computing, Wuhan University Journal of Natural Sciences, vol. 12, 2006, no. 1, 2007, pp:13-16.

[5] G. S. Dong, X. Li, X. L. Lu, A MLS Cooperation System Based on Virtual Machine, Monitor, International Conference on Wireless Communication, Networking and Mobile Computing, 2007, Sept., 2007, pp: 6345 – 6348.

[6] B. Stefan, C. Ramon, A. G. Kenneth, et al., vTPM: Virtualizing the Trusted Platform Module, USENIX'06, http://www.usenix.org/events/sec06/tech/full_papers/berger/berger.pdf.

[7] GRUB TCG Patch to support Trusted Boot, http://trousers.sourceforge.net/grub.html.

[8] F. Zhang, G. Q. Wu, J. Tao, M. T. Yuan, Trust of Hardware, Symposia of the IEEE 2008 International Conference on Embedded Software and System, July. 2008.

[9] L. P. J. Nick, F. Timothy, W. AAron, et al., An Architecture for Specification-Based Detection of Semantic Integrity Violations in Kernel Dynamic Data, USENIX'06, http://www.usenix.org/event/sec06/tech/full_papers/petroni/petroni_html/

[10] H. G. zhang, F, Wang, A behavior-based remote trust attestation model, Wuhan University Journal of Natural Sciences, vol. 11, no. 6, 2006, pp: 1819-1822.

[11] F. Riccardo, g. Roberto, Classification of Security Properties, Foundations of Security Analysis and Design, Springer LNCS, vol. 2171, 2001, pp:331-396.