Computer Science and Application 计算机科学与应用, 2017, 7(3), 183-191
Published Online March 2017 in Hans. http://www.hanspub.org/journal/csa
https://doi.org/10.12677/csa.2017.73023

Hans汉斯

Research on Using Trusted Hardware for Data Location Assurance in Cloud Storage

Guan Wang, Ruixue Xu

Beijing University of Technology, Beijing
Email: timelysnow@emails.bjut.edu.cn

Received: Mar. 3rd, 2017; accepted: Mar. 20th, 2017; published: Mar. 23rd, 2017

Abstract

Recently, the lack of geo-location assurance of data stored in cloud storage has become a main reason which restricts organizations that deal with sensitive data (e.g., financial data, health data) to adopt cloud storage. This paper proposed a mechanism for verifying the geographic location of the stored data. We use Trusted Cryptographic Module (TCM) to identify physical machines and use a trusted third party to verify the actual location. In addition, our approach enables the verification of the trustworthiness of the physical machines which the cloud storage operators provide. The discussion shows that the approach of this paper has an adequate level of security, and by the usage of TCM and additional location audits, can enable a reliable location verification of the stored data.

Keywords

Cloud Storage Security, Trusted Computing, Data Geo-Location, TCM

云存储中基于可信硬件的数据存储位置保障方法研究

王　冠，许瑞雪

北京工业大学，北京
Email: timelysnow@emails.bjut.edu.cn

收稿日期：2017年3月3日；录用日期：2017年3月20日；发布日期：2017年3月23日

摘　要

在云存储中，一些处理敏感数据(如政府文件、资产数据、健康数据等)的用户需要限制数据存储的地理位置，而目前大多数云存储供应商并没有向用户提供验证数据存储位置的方法。针对这个问题，采用可

文章引用: 王冠, 许瑞雪. 云存储中基于可信硬件的数据存储位置保障方法研究[J]. 计算机科学与应用, 2017, 7(3): 183-191. https://doi.org/10.12677/csa.2017.73023

信硬件来标识物理机器并利用可信第三方来验证数据实际存储位置。此外，还可以验证云存储供应商提供的物理机器的可信性。分析结果表明，本文提出的方法具有很高的安全性，通过采用TCM芯片及额外的位置审计，能够可靠地验证数据存储的实际位置。

关键词

云存储安全，可信计算，数据存储位置，可信密码模块

1. 背景

基于云的存储服务越来越流行，用户可以采用云存储服务远程备份自己的数据、从任何连接设备上访问数据并且能够通过访问共享数据与其他用户合作。例如，Amazon S3 和 Google Storage 可以为用户、企业和其他云存储服务供应商提供可扩展的存储服务。Dropbox，Apple iCloud 和 Google Drive 可以在多用户和多设备的环境下为用户提供文件共享和文件同步服务[1]。云存储服务使用户在较低成本下便利地存储和访问数据，并且减轻了用户维护巨大的本地数据存储的成本。

除了云存储服务的安全和隐私问题，一些处理敏感数据(如财务数据、健康数据以及个人身份数据)的用户对数据存储的地理位置有要求，如何对要存储的数据提供有效的地理位置保障和检测机制，也成为云存储供应商和用户面临的一个比较重要的问题。

目前，一些云存储服务提供商(如 Amazon S3)允许用户在服务协定中选择数据存储的地理位置，但是作为用户，并没有途径去检测提供商是否履行了协议，而当你发现他们违反协定时(无意地或偶然地)，损害却已经造成了。近年来，发生了几起出租的商业服务中断运行的情况，导致用户在相当长的一段时间无法访问自己的数据。例如，2012 年 12 月 24 日，部署在 AWS 上的 Netflix 服务，对用户中断服务长达 12 小时[2]。类似的情况还在 Dropbox 用户[2] [3]以及 Xbox 用户中也发生过[3]。为了解决这个问题，云存储服务商可能会把这些数据转移到其他地方，那么如何确定云存储服务商是否遵守了服务协议中的位置条款就成了用户关心的一个重要问题。并且，为了提高数据的可用性，一些云存储提供商可能在多个数据中心存储数据的复制版本，比如 Microsoft Azure，这使情况变得更糟，用户需要能够验证云存储中数据的实际存储位置的要求就更加强烈。

目前，也已经有一些研究者提出采用可信硬件来保障虚拟资源(数据或者程序)的存放位置。NIST 做了第一个尝试[4]，他们提出了将服务器的地理位置信息写入可信平台模块 TPM (Trusted Platform Module)的 PCR 中并提出了一个应用原型，当服务器收到存储请求后，它首先检查它的位置请求，然后根据自身的位置决策返回响应，但是此方案仅适用于云存储服务商内部使用，并不对用户提供位置保障和检测机制。Hsin-Jung Yang 等人基于 TPM 提出了通过验证虚拟资源(数据或程序)的位置信息来提供一个可信授权机制[5]。Ali Noman 等人采用 TPM 与数据持有性证明协议 PDP (Provable Data Possession)设计了一个可供用户验证的保障数据存储位置的方法[6]。

本文提出一个检测数据存储地理位置的方法，防止云存储供应商在用户不知情的情况下将数据移动到用户不愿存储的地方。本文将绑定在云存储设施中物理机器上的可信硬件作为唯一身份识别符，每一

个物理机器都含有唯一的可信硬件。可信硬件的地理位置由可信第三方来保证。用户可以利用可信硬件可靠地定位他们的数据存储位置并且验证他们正在使用的云存储设施的可信性，检测是否存在云存储提供商或者其他人的恶意操作。

本文的组织结构如下：第一部分介绍了本文的研究背景；第二部分介绍了可信密码模块相关研究；第三部分介绍了基于可信硬件的数据存储位置验证方法；第四部分分析了方法的安全性；第五部分对本文的研究做出了总结。

2. 可信密码模块 TCM

硬件系统的安全和操作系统的安全是信息系统安全的基础，密码、网络安全等技术是关键技术。只有从信息系统硬件和软件的底层采取安全措施，才能有效地保障信息系统的安全，这促进了可信计算的迅速发展。

为了解决个人计算机结构上的安全问题，并从底层入手提高其可信性，Intel、Microsoft、IBM、HP、Compaq 等著名的信息技术企业在 1999 年 10 月共同发起并成立了可信计算平台联盟(Trusted Computing Platform Alliance, TCPA)。TCPA 定义了具有安全存储和密码功能的可信平台模块(Trusted Platform Module, TPM)，并于 2001 年 1 月发布了基于硬件系统的“可信计算平台规范”(V1.0)。TCPA 的成立标志着可信计算高潮阶段的出现。2003 年 3 月，TCPA 更名为可信计算组织(Trusted Computing Group, TCG) [7]，其目的是在计算和通信系统中广泛使用基于硬件安全模块的可信计算平台，以提高整体的安全性，扩展可信范围。

可信计算的基本思想是在计算机系统中首先建立一个信任根，再建立一条信任链，从信任根到硬件平台到操作系统再到应用，一级测量认证一级，一级信任一级，把信任关系扩大到整个计算机系统，从而确保计算机系统的可信。

可信密码模块(Trusted Cryptography Module, TCM)是我国借鉴国际可信计算框架与技术理念，结合我们信息安全管理国情，在可信计算领域自主研制的安全芯片。可信密码模块的主要核心功能是提供基于硬件的国产密码算法和密钥保护；唯一地标识平台身份；度量平台完整性。

TCM 芯片[8]可以为加密密钥提供安全存储并且为随机数发生器、密钥发生器、哈希计算提供硬件加速计算引擎。TCM 提供的密钥类型包括采用 SM2-3 和 SMS4 算法的存储密钥、采用 SM2-3 和 SMS4 算法的绑定密钥、采用 SM2-1 算法的签名密钥、采用 SM2-1 算法的身份密钥和采用 SM2-3 算法的平台加密密钥。

当一个用户第一次激活一个平台时，TCM 使用 TCM_TakeOwnership 命令进行初始化，此时访问 TCM 的用户密码被设置。这个过程不能远程执行，并且一个新的初始化需要 TCM 的实体存在以及一系列的重置命令。

每个 TCM 拥有唯一的背书密钥(Endorsement Key, EK)，TCM 出厂时都会由 TCM 厂商签发背书证书来唯一标识可信平台的身份。EK 的私钥部分 EK_{priv} 受 TCM 保护，永久存储在 TCM 内部。由于安全与隐私问题，EK_{priv} 不能用于签名操作，它只能被用来解密用 EK_{pub} 加密过的敏感数据。背书证书(Endorsement Credentials, EC)包含 EK_{pub}，TCM 厂商使用私钥对背书证书进行签名，以便于验证 TCM 和 EK 的有效性。

除了背书证书，TCM 还含有一致性证书(Conformance Credentials，CC)和平台证书(Platform Credential, PC)来证明 TCM 的组件符合《可信密码模块规范》的要求。一致性证书由评估机构(如平台制造商、供应商或者独立实验室)颁发，确保可信构造基(Trusted Building Blocks, TBB)与 TCM 说明相一致。平台一致性证书由平台制造商、供应商或者其他独立实体颁发，保证平台包含背书证书中的 TCM 芯片。

平台身份密钥(Platform Identity Key, PIK)用于对完整性值和其他密钥的数字签名。平台身份密钥是一个 SM2 密钥对，必须在所有者授权的情况下才能够生成。TCM 采用 TCM_MakeIdentity 命令生成 PIK。平台身份证书(Platform Identity Credential, PIC)用来证明此 PIK 与有效的背书证书、平台证书以及一致性证书绑定。PIK 证书的构建需要认证机构(Certification authority, CA)的参与。

CA 是证书的签发机构，它是公钥基础设施的核心。证书是公钥体制的一种密钥管理媒介。它是一种权威的电子文档，形同网络计算环境中的一种身份证，用于证明某一主体的身份及其公开密钥的合法性。因此，公钥体制环境必须有一个可信的机构 CA 来对任何一个个体的公钥进行公证，证明主体的身份及它与公钥的匹配关系。

TCM 向 CA 发送一个包含 PIK_{pub}、背书证书 EC、一致性证书 CC、平台证书 PC 的请求，并使用 PIK_{priv} 对请求进行签名。如果证书是有效的，CA 产生平台身份证书，使用背书证书中的 EK_{pub} 加密 PIC，将其发送回 TCM。

由于密钥的数量可能非常庞大，而 TCM 内部的存储空间有限，那么一些密钥(如 PIK)，可以选择存储在外部如硬盘上。每一个存储在外部的密钥都需要被存储根密钥(Storage Root Key, SRK)加密。SRK 是一个不可迁移密钥对，它建立了密钥存储的可信根。SRK 以非易失的方式存储在 TCM 内部，并且永远不在 TCM 外部使用。当一个新用户被授权时，SRK 可以被重置。

为了安全存储度量结果，TCM 在内部开辟了专门的完整性值存储空间—平台配置寄存器(Platform Configuration Registers, PCRs)。平台启动时，PCR 值会被重置为默认值。平台启动之后，以扩展的方式更改 PCR 值。软件组件(BIOS，bootloader，操作系统，应用程序)在执行前被度量，通过扩展之前的哈希值得到新的哈希值被写入到特定的 PCR 中：$\text{Extend}(PCR_N, \text{value}) = SHA1(PCR_N \,\|\, \text{value})$。SHA1 代表 TCM 使用的哈希密码函数，$\|$代表连接操作。这条信任链的信任根是可信度量根(Core Root of Trust Measurement, CRTM)，它在 BIOS 中，并且在平台加电时被第一个执行。接着，CRTM 度量它自身以及 BIOS，然后将控制权交给信任链中的下一个软件组件。对每一个被度量的组件，一个事件被创建并存储在存储度量日志(Stored Measurement Log, SML)中。这样，采用[9]中提出的完整性度量框架(Integrity Measurement Architecture, IMA)，一个远程实体就可以通过 PCR 和 SML 来证明平台状态。为开始一个证明，挑战者创建一个随机数并发送给证明系统，证明系统的 TCM 使用 PIK 对随机数和 PCR 签名并将此签名、随机数以及 SML 发送回挑战者。挑战者检验签名与 SML 中对应的值是否一致来判断系统的完整性。

3. 基于 TCM 的数据存储位置保障方法

本文采用物理机器中的 TCM 芯片作为验证数据存储位置的可信根。绑定有地理位置信息的 TCM 芯片在认证机构(Certification Authority, CA)中注册。用户运行虚拟机中的客户端软件来初始化一个证明协议，通过这个协议清楚地识别 TCM 的身份并且在 CA 的协助下验证位置。此外，此证明协议可以保证虚拟机上运行的软件组件(BIOS、bootloader、hypervisor 等)没有受到任何损害。

3.1. 整体框架

图 1 展示了基于 TCM 的数据存储位置保障整体框架。虚拟机 VM_1 运行在云存储提供商提供的位于某个特定地理位置的一台机器上，此台机器含有 TCM 芯片并安装了云存储服务。某用户的数据通过使用云存储服务存储在VM_1上。虚拟机监视器运行在硬件之上并负责执行所有的虚拟机。LICT 模块提供了位置验证、完整性检查以及 TCM 驱动的功能。LICT 位于虚拟机监视器中，它管理位置验证请求对 TCM 的访问并实现上文中提出的 IMA 的必要部分，来保证虚拟机监视器以及 LICT 模块没有受到损害。运行在虚拟机监视器上的所有虚拟机通过 LICT 模块访问 TCM 芯片。虚拟机中的 LocCheck 模块通过 LICT

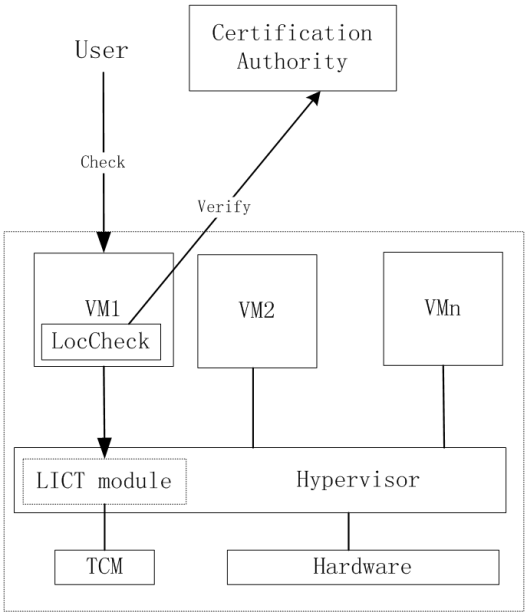

Figure 1. Overall framework
图 1. 整体框架

模块获取 TCM 身份。绑定物理机器位置信息的 TCM 芯片在 CA 中注册。我们假设可信服务提供商或者 CA 执行定期系统检查来验证物理机器的位置。例如，依据标准 ISO/IEC 27001，位置验证是一个需要定期执行的安全审计。简单起见，我们还需假设 CA 维护了一个 IMA 要求的可信软件组件的指纹数据库。可以通过与 CA 通信来验证 TCM 以及机器的地理位置。在本文中，我们还假定用户、云存储提供商以及 CA 之间的通信是安全的。

3.2. 详细设计

本文的原型分为两个部分：初始化阶段和验证阶段。初始化阶段只执行一次，当云存储提供商提供一个新的绑定有 TCM 芯片的机器，并将此机器以及其位置信息在 CA 中注册时，执行初始化协议。平台身份证书在此阶段生成。在初始化阶段，CA 通过证明协议验证机器的可信性。当 CA 验证了机器位置信息的正确性并且系统是可信的，此机器的用户就可以通过执行验证阶段来验证位置了。

初始化阶段：当云存储供应商提供一台新的机器 M 时，需执行下面几步进行初始化，如表 1 所示。

首先，LICT 模块执行 TCM_TakeOwnership 命令来初始化 TCM，设置访问 TCM 的用户密码，并将此密码存储在 LICT 模块中。然后，执行 TCM_MakeIdentity 命令使 TCM 生成 PIK 密钥对(PIK_{priv}, PIK_{pub})。每一个这样的 SM2 密钥对都与一个健柄 h_{PIK} 绑定，因为在一个 TCM 内部可以生成多个不同的 PIK。在生成 PIK 的同时，用来访问 EK_{priv} 的密码 $pass_{PIK}$ 也被设置。

TCM 生成一个 PIK 请求(PIK-Request, PR)并使用 PIK_{priv} 对 PR 进行签名，此请求包含 PIK_{pub}，背书证书 EC，平台证书 PC，一致性证书 CC。TCM 使用 PIK_{priv} 对 PR 进行签名并将此签名与 PR 发送到 LICT 模块，由 LICT 模块发送到 CA。由于背书证书包含背书公钥 EK_{pub}，PIK 密钥对唯一地绑定到特定 TCM，因此，LICT 模块存储访问 TCM 的用户密码、h_{PIK}、$pass_{PIK}$、PIK_{pub} 以及所有的证书，以便使用。

收到消息之后，CA 验证证书和签名的有效性。若验证通过，CA 生成一个平台身份证书(Platform IdentityCredential, PIC)并且使用 CA 的私钥对 PIK_{pub} 签名。采用对称加密方案 Sym_Enc 用生成的会话密钥 K 对 PIC 加密。会话密钥 K 本身采用 SM2 算法用 EK_{pub} 加密后与 PIC 一起被发送到 LICT 模块。

Table 1. Initialization phase steps
表 1. 初始化阶段步骤

步骤	执行模块	命令
1a.	LICT→TCM:	TCM_TakeOwnership
1b.	LICT→TCM:	TCM_MakeIdentity
1c.	TCM	Generate (PIK_{priv}, PIK_{pub})
1d.	TCM	PR = PIK_{pub}, EC, PC, CC
1d.	TCM	SM2_Sign (PR|PIK_{priv}) = Sig
2.	LICT→CA:	PR, Sig
3a.	CA	Verify Credentials
3b.	CA	Verify SM2_Verify(Sig|EK_{pub}) = PR
3c.	CA	Generate PIC
3d.	CA	Generate K
3e.	CA	SM2_Enc(K|EK_{pub})
3f.	CA	Sym_Enc(PIC|K)
4.	CA→LICT:	SM2(K|EK_{pub}), Enc(PIC|K)
5a.	LICT→TCM:	TCM_ActivateIdentity
5b.	TCM	Decrypt K
5c.	TCM	Decrypt PIC
6a.	LICT→TCM:	TCM_Quote(h_{PIK}, $pass_{PIK}$, loc_M, S_{PCR})
6b.	TCM	SM2_Sign(loc_M, S_{PCR}|PIK_{priv}) = SigL
6c.	LICT→CA:	loc_M, PCR[S_{PCR}], SML, SigL
7a.	CA	Verify SigL
7b.	CA	Verify platform integrity
7c.	CA	Verify loc_M
7d.	CA	Mark loc_M of M as verified

LICT 模块调用 TCM_ActivateIdentity 命令使用 EK_{priv} 解密会话密钥 K。这个操作保证了在后续步骤中只有 TCM 可以解密 PIC。

此时，LICT 模块开始在 CA 注册物理机器 M 的地理位置 loc_M。如上文 3.1 中所说，我们将位置注册与平台完整性验证、CA 验证机器 M 没有收到任何损害，BIOS、Bootloader、Hypervisor、LICT 模块是可信的结合起来。TCM 激活 TCM_Quote 命令，使用 PIK_{priv} 对 loc_M 以及相应的用于启动过程完整性验证的 PCR 值 S_{PCR} 进行签名。参数 h_{PIK} 用来指定正确的 PIK，$pass_{PIK}$ 用来获取此 PIK 密钥的使用权。发送到 CA 的消息包括位置 loc_M、设置的 PCR 的值 PCR[S_{PCR}]、存储度量日志 SML 以及签名 SigL。

当 CA 收到消息之后，它首先使用 PIK_{pub} 验证签名的有效性。然后比较 PCR 值与 SML 的一致性来验证平台可信性。当位置信息被某外部机构(任何形式的审计)验证之后，CA 认为这台机器(包括它所绑定的 TCM)及其位置是可信的。

当初始化阶段结束之后，绑定到云存储供应商提供的机器上的 TCM 的身份标识密钥 PIK 以及它目

前的位置就成功在 CA 中注册了并可以被下文的验证阶段所使用。

验证阶段：验证阶段的协议步骤如表 2 所示。

当用户 U 想要验证他的数据存储实际位置是否与供应商声称的位置信息一致时，用户向存储数据的虚拟机 VM_1 发出验证请求。位于虚拟机 VM_1 中的 LocCheck 模块选择几块随机的内存区域 $MA_1, MA_2, \cdots, MA_n$，并将指针 MA 指向此内存区域。将这些随机内存区域应用在验证阶段，可防止供应商的重放攻击。

接下来，LocCheck 模块向 LICT 模块请求 TCM 的身份标识公钥 PIK_{pub} 并将 PIK_{pub} 以及内存区域的哈希 $\text{hash}(MA_1, MA_2, \cdots, MA_n)$ 发送到 CA 来请求确认位置。其中，PIK_{pub} 用来标识 TCM 身份。

当 CA 接收到 PIK_{pub} 后，从数据库中查找验证机器 M 平台完整性的 PCR 值并将 S_{PCR} 返回 LocCheck 模块。LocCheck 模块将 S_{PCR} 以及内存区域指针 MA 发送到 LICT 模块。

LICT 模块计算 $\text{hash}(\text{MA}) = \text{hash}(MA_1, MA_2, \cdots, MA_n)$ 并请求 TCM 使用 TCM_Quote 命令用 PIK_{priv} 对 hash(MA)和 S_{PCR} 签名。同样，参数 h_{PIK} 用来指定正确的 PIK，$pass_{PIK}$ 用来获取此 PIK 密钥的使用权。TCM 将签名 SigMA、PCR 值 PCR[S_{PCR}]发送到 CA。

CA 验证签名的有效性、平台完整性、验证的 loc_M 与注册的一致性以及 LocCheck 模块计算出的所选内存区块的哈希与 LICT 模块签名的一致性。如果所有的验证均通过，CA 使用自身的私钥对 loc_M 的确认信息进行签名并将其发送到 LocCheck 模块，由 LocCheck 模块将此信息发送给用户。最后，用户使用 CA

Table 2. Validation phase step
表 2. 验证阶段步骤

步骤	执行模块	命令
0.	User→LocCheck:	Memory areas MA
1a.	LocCheck→LICT:	Request PIK
1b.	LICT→LocCheck:	PIK_{pub}
1c.	LocCheck	Generate hash (MA)
1d.	LocCheck→CA:	Hash (MA), PIK_{pub}
2a.	CA	Select S_{PCR}
2b.	CA→LocCheck:	S_{PCR}
2c.	LocCheck→LICT:	MA, S_{PCR}
3a.	LICT	Generate hash (MA)
3a.	LICT→TCM:	TCM_Quote(h_{PIK}, $pass_{PIK}$, hash(MA), S_{PCR})
3b.	TCM	SM2(hash(MA), S_{PCR}|PIK_{priv}) = SigMA
3c.	TCM→LICT	PCR[S_{PCR}]
4a.	LICT→LocCheck:	PCR[S_{PCR}], SML, SigMA
4b.	LocCheck→CA:	PCR[S_{PCR}], SML, SigMA
5a.	CA	Verify SigMA
5b.	CA	Verify platform integrity
5c.	CA	Verify loc_M marked as verified
5d.	CA	Verify hash(MA)
6a.	CA→LocCheck:	Location loc_M confirmed
6b.	LocCheck→User:	Location loc_M confirmed

的公钥验证确认信息的签名并验证位置 loc_M 是否正确。

4. 安全性分析

本节主要分析上文提出的原型的安全性。我们假设云存储提供商试图将用户的数据移动到更便宜的地方来节约成本。

对我们的原型而言，我们假设 TCM 的位置是被检测过的并且它们的实际位置与它们在 CA 中注册的位置一致。我们假设敌手无法破坏 TCM 的抗损坏机制来读取密钥或者损坏任何密码机制，例如，敌手无法在不知道私钥的情况下伪造签名或者破解哈希函数。

对本文而言，敌手可以采用两个方法攻击我们的原型：伪造机器位置或者进行重放攻击。

对于第一种情况，敌手试图伪造机器位置来满足用户要求。由于初始化阶段是在审计期间被验证的，敌手只能在验证阶段进行攻击。敌手需要通过步骤 5a 到 5d (表 2)的验证，直接访问安全存储在 TCM 中的 PIK_{priv} 来产生签名 SigMA。为了获得 PIK_{priv}，敌手必须破坏 TCM 的抗损坏机制，这与上文的假设相冲突。或者，敌手可以将绑定到某机器上的 TCM 芯片移动到其他位置的机器上。然而，这会在下一次审计的最后一步被检测出来。所以，审计操作需要以一个合理的频率定期执行，这样，敌手这种卸载、转移、重装 TCM 芯片的经济成本就会很大。

对于第二种情况，敌手需要两台机器来执行一个重放攻击。机器 M_A 对于用户的要求而言，位于一个正确的位置，没有被修改过并且正确运行。这台机器同时运行着恶意供应商的虚拟机 VM_x。敌手的另外一台机器 M_B 位于一个不符合用户要求的位置。在 M_B 上，虚拟机监视器以及 $LICT_B$ 被修改，被修改过的模块将消息传送到机器 M_A 上的虚拟机 VM_x 中的修改过的 $LocCheck_x$ 模块。基于这种方式，敌手可以将虚拟机 VM_1 从 M_A 移动到 M_B 上并且可以转发所有的协议消息，验证依然会通过。为了解决这个问题，我们通过扩展对虚拟机 VM_1 的验证将基于软件的证明应用到我们的协议原型中。绑定在机器上的 TCM 对特定虚拟机的内存区块 $MA_1, MA_2, \cdots, MA_n$ 的哈希值产生签名 SigMA。为了通过机器 M_A 的验证，敌手的虚拟机 VM_x 必须与运行在机器 M_B 上的虚拟机 VM_1 完全相同。这样，敌手的重放攻击就不存在任何意义了，因为他必须在机器 M_A 上保存一个 VM_1 的副本，也就是浪费了机器 M_B 上一倍的存储空间。

远程证明中存在的一个普遍的问题是，在验证和使用的间隙，敌手可能会改变机器的状态。所以，我们建议有规律地进行重复验证。

5. 总结

随着云存储的应用越来越广泛，云存储中的数据安全问题也越来越重要。用户可能需要自定义安全策略来保障自身数据安全，其中一个重要的方面便是数据存储和处理的地理位置，例如，有些用户要求数据存储在国内而不能存储到国外的服务器上。本文提出的方法基于可信密码模块及可信第三方使用户能够验证数据存储的实际位置以及云存储供应商提供的机器的可信性。我们提出的证明协议能够验证云存储供应商提供的机器的平台配置的完整性，并且能够唯一地标识绑定在机器上的 TCM 的身份。TCM 的实际位置以及用于检测云存储供应商提供的机器平台完整性的数据存储在可信第三方上。数据的实际存储位置以及平台完整性可以通过与第三方通信来证明。本文的安全性分析说明了我们的原型具有很高的安全性。与前人的研究相比，我们的方法通过采用 TCM 芯片及额外的位置审计，能够可靠地验证数据存储的实际位置。

参考文献 (References)

[1] Tate, S.R. and Vishwanathan, R. (2013) Multi-User Dynamic Proofs of Data Possession Using Trusted Hardware.

ACM, San Antonio, 353-364.

[2] Shetty, S. and Rogers, T. (2014) Classification Based IP Geolocation Approach to Locate Data in the Cloud Data Centers. ACM, USA.

[3] http://www.zdnet.com

[4] Christoph, K. (2013) Using Trusted Platform Modules for Location Assurance in Cloud Networking. Network and System Security, Springer, Berlin Heidelberg, 109-121.

[5] Yang, H.-J. (2013) Victor Costan. Authenticated Storage Using Small Tusted Hardware. ACM, 35-46.

[6] Noman, A. and Adams, C. (2014) Hardware-Based DLAS: Achieving Geo-Location Guarantees for Cloud Data Using TPM and Provable Data Possession. ACM, Crown, 280-285.

[7] Trusted Computing Group (2004) TCG Specification Architecture Overview Revision 1.2. http://www.trustedcomputinggroup.org

[8] 冯登国, 等. 可信计算-理论与实践[M]. 北京: 清华大学出版社, 2013: 18-46.

[9] Zhang, S.R. (2004) Design and Implementation of a TCG-Based Integrity Measurement Architecture. *Proceedings of the* 13*th Usenix Security Symposium*, USA, 1-20.