DOI:10.13245/j.hust.2010.04.031
第38卷 第4期 华中科技大学学报(自然科学版) Vol.38 No.4
2010年 4月 J. Huazhong Univ. of Sci. & Tech. (Natural Science Edition) Apr. 2010

基于 Merkle 树的远程证明机制

邱 罡[1] 刘志都[2] 周利华[1]
(1 西安电子科技大学 CNIS 教育部重点实验室, 陕西 西安 710071;
2 南阳师范学院 计算机与信息技术学院, 河南 南阳 473061)

摘要: 针对生产控制环境中设备状态的可信验证问题, 提出了一种远程证明方案. 采用与管理中心协商策略, 由设备平台上的可信平台模块定时地对设备平台运行状态进行完整性度量和评估, 并生成完整性报告. 为保证报告结果的新鲜性和完整性, 在设备平台引入时间戳及 Merkle 树相结合的随机数方法, 并保证报告结果的有效性. 最后, 对该方案的安全性和效率进行了分析.

关 键 词: 完整性评估; 哈希函数; 远程证明; 可信计算; Merkle 树
中图分类号: TP309 文献标识码: A 文章编号: 1671-4512(2010)04-0050-04

Remote attestation scheme based on Merkle tree

Qiu Gang[1] *Liu Zhidu*[2] *Zhou Lihua*[1]
(1 The CNIS Key Laboratory of the Education Ministry, Xidian University, Xi′an 710071, China;
2 College of Computer and Information Technology, Nanyang Normal University, Nanyang 473061, Henan China)

Abstract: A remote attestation scheme was proposed to make device′s report its integrity status actively. According to the policy negotiated between control center and devices, the devices measured and evaluated their runtime integrity status automatically and generated integrity reports periodically with the aid of trusted platform module (TPM). Time stamp and Merkle tree organized nonces were introduced to ensure the freshness, integrity and availability of the reports. The security and efficiency of the scheme were also analyzed.

Key words: integrity evaluations; hash function; remote attestation; trusted computing; Merkle tree

在现场控制环境中, 通常是由管理中心对生产网络中的设备运行状态定时地进行查询, 对设备出现的异常状态进行报警, 并采取相应的措施. 但是这种监控是一种被动式的监控, 仅反映了设备的部分运行状态, 并未反映出设备运行时的全部状态是否可信, 即是否遭受了病毒攻击或被篡改. TCG[1] 定义了一组使硬件和软件能够支持可信计算的工业标准的规范. TCG 的核心是一个称为 TPM[2] (trusted platform module, 可信平台模块)且能防止篡改的硬件安全模块. TPM 作为加密协处理器和保护密钥及秘密信息的保护存储装置, 被用来度量和报告平台的完整性信息, 并且不会受到平台使用者或其上运行的软件的威胁. 本文在 TCG 标准提出的平台证明的基础上, 结合哈希函数提出了一个基于 Merkle 树[3] 的安全的证明方案, 实现了生产设备在运行过程中自主地向管理平台报告设备运行状态, 有效保证了现场设备身份的可信性, 并增加了设备状态证明的灵活性.

1 远程证明

TCG 标准中定义的远程证明(或平台完整性报告)机制, 用于向验证方(verifier)报告存储在 PCR 中的完整性度量值是以可靠方式存储的. 在证明过程中证明方(attester)TPM 对 PCR (plat-

收稿日期: 2009-07-10.
作者简介: 邱 罡(1973-), 男, 博士研究生, E-mail: qiugang7780@163.com.
基金项目: 国家高技术研究发展计划资助项目(2007AA01Z429, 2007AA01Z405); 国家自然科学基金重点研究计划资助项目(60633020); 河南省自然科学基金资助项目(092300410219).

form configuration registers）值和附加的数据（如来自验证方的防止重放攻击的随机数）使用私钥签名，并传给验证方. 证明是一项受保护的原子操作，不能被恶意软件伪造. 平台也可以在证明中发送附加信息，如完整性度量日志，其中包括平台中加载并被 TPM 度量过的组件列表，以及这些组件是如何构成的.

远程验证方通过验证带有 AIK（attestation identities key，身份证明密钥）签名的报告，并将收到的完整性度量值与本地收集的参照度量日志进行对比来验证证明. 攻破的系统可以修改度量日志，但其不能改变受 TPM 保护的度量结果与修改过的日志匹配，因此通过和签过名的度量结果对照验证度量日志，伪造操作即可被发现. 远程证明的基本原理如下[4]. **a**. Verifier：生成随机数 Nonce；**b**. Verifier → Attester：$Enc_V\{Nonce\}$；**c**. Attester→ Verifier：$Sign_{AIK}\{PCR, Nonce\}$，SML，$Cert_{CA}(AIK)$；**d**. Verifier：首先验证由证书中心（certificate center，CA）为 AIK 签发的证书 $Cert_{CA}(AIK)$，其次验证签名 $Sign_{AIK}\{PCR, Nonce\}$，最后检验 Nonce 的新鲜性，并通过 PCR 值验证 SML 的一致性. 其中，$Enc_V\{\}$ 为使用验证方公钥加密；$Sign_{AIK}\{\}$ 为使用 AIK 私钥的签名. SML（storage measurement log，存储度量日志）其中包括被测量组件的名称以及对应的散列值. 验证方通过 SML 和 PCR 值获得证明方系统运行的软件状态，配置信息等.

上述平台完整性证明协议中，需要验证方（如管理平台）发送随机数 Nonce 给证明方（如设备），以保证度量值的新鲜性，并防范重放攻击. 平台完整性证明只能由验证方发起，证明方被动接受证明，设备无法主动将完整性报告上传，增加了验证方的负担，因此并不适合现场生产环境.

2　基于 Merkle 树的证明方案

本证明方案主要从证明结果的新鲜性，完整性以及设备的主动性出发，对上述方案进行改进.

a. 新鲜性. TPM v1.2 中新增了时间戳功能. TPM 使用 TPM_TickStampBlob 命令在时戳摘要 blob 上加上签名：$TS \leftarrow Sign_{AIK}\{blob \| t \| TSN\}$，其中 AIK 是签名私钥，$t$ 是当前时间，TSN 是由 TPM 产生的时间会话随机数（tick session nonce）. 时戳 TS 中并未包含实际的时钟，而是 TPM 平台从启动时开始的滴答计数，因而其本质上是滴答时戳. TPM 从时间会话开始计数滴答值，时间会话依靠 TSN 值进行区分，即若在时间会话过程中得到不同的 TSN 值，则说明系统发生了重启或遭受到硬件攻击.

设备在发送的证明报告中加入时戳，管理平台比较先后接收到的 TS_1 和 TS_2，若其中 $TSN_1 = TSN_2$，则设备未发生重启；再验证若 $t_1 < t_2$，则证明报告是新鲜的[5~7].

b. 完整性. 管理平台期望接收到的证明结果是未经篡改的，本方案中通过散列链的方式保证了证明结果的完整性. 散列链的起始值是由设备签发的随机数（如图 1 中的叶子节点）. 在初次发送完整性报告时，设备将发送的 PCR 值与该随机数进行合并再散列，并由设备对此结果进行签名后发送. 再次发送完整性报告时，则将上次的散列结果与新的 PCR 值进行合并再散列，签名后发送. 以图 1 中的叶子 B_0 为例，散列链的构成为 H，报告的完整性传递过程如图 2 所示.

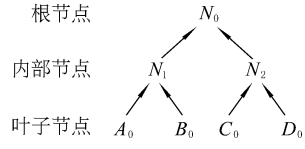

图 1　*Merkle* 树示意图

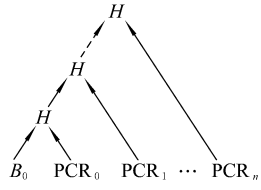

图 2　完整性报告传递链

为提高效率，设备需要向多个设备管理平台进行报告时，由设备生成 *Merkle* 散列树，并对散列树的根节点进行签名，将各个叶子作为设备与各个管理中心之间散列链的初始值. 其主要优点是仅需通过对树根节点的一次签名运算就可以对树上所有的叶节点独立地提供完整性认证[8,9]. 由于各个叶子的完整性是可以验证的，因此保证了后续传输的完整性.

c. 主动性. 在设备平台中增加证明代理（attestation agent，AA）提供证明方案. 当设备首次向管理平台注册时，管理平台保存设备的 TSN 以及证书 $Cert_{CA}(AIK)$，并将证明方案的相关策略发送给 AA. 管理平台发送的策略包括上传完整性报告的时间或时间周期，AA 根据策略可在固定的周期内主动上传完整性报告.

基于 Merkle 散列树的证明过程如下.

步骤 1　Attester：生成 Merkle 散列树，并用 AIK 对根节点 N_0 签名；

步骤 2 Attester→Verifier: $Cert_{CA}$ (AIK), $Enc_{AIK}\{B_0, A_0, N_0\}$, $Sign_{AIK}\{N_0, TSN, t_0\}$, 其中 B_0 和 A_0 是散列树的叶子节点;

步骤 3 Verifier→Attester: 将报告策略发送给设备;

步骤 4 Attester→Verifier: $Sign_{AIK}\{PCR, H(B_0 \| PCR), TS\}$, SML;

步骤 5 Verifier: 首先将接收到的 PCR 值与步骤2 中收到的 B_0 进行散列, 并和接收到的签名中的散列值比较, 验证 PCR 值的完整性; 其次验证时戳 TS 中的 TSN 值是否发生改变, 以判断设备是否发生重启, 并判断其中的滴答值是否大于步骤 2 中的滴答值 t_0, 以验证其新鲜性; 最后通过 PCR 值验证 SML 的一致性.

3 安全性及效率分析

3.1 安全性分析

a. 数据完整性. 在本方案中, 利用发布 Merkle 散列树认证路径来保证初始随机数的完整性, 因为如果发送的随机数发生了改变, 则在验证阶段计算出的根节点值将与公开值不符, 这将导致验证失败. 同时散列函数的单向性也保证了发送数据信息的完整性.

b. 防抵赖性. Merkle 散列树安全性是依赖于散列函数的安全性[10]. 散列函数 $y=F(x)$ 具有如下特点: 对于给定的 x 值计算 y 是很容易的, 但仅知道 y 值想要计算 x 值是不可能的; 并且不存在两个不同的 x 值 x_1 和 x_2, 使 $F(x_1)=F(x_2)$ 成立. 即只有知道正确的 x 值的人才能得到正确的 y 值. 因此, 若双方使用散列函数验证成功, 则可以确定是具有正确的 x 值的一方发出的信息.

c. 抗重放攻击. 通过时戳中的滴答值以及每次报告 PCR 值与上次报告结果的散列值可以有效地防止重放攻击的发生. 由于散列函数的抗碰撞性, 任何对于散列值的插入和改变, 都将使验证结果发生改变, 保证了远程证明的完整性.

3.2 效率分析

在设备平台上需要构造适当规模的 Merkle 散列树, 在生成 Merkle 散列树时应考虑到散列树的存储规模和计算开销. 散列树的叶子节点选用 160 bit 的随机数构成, 散列函数采用输入和输出均为 160 bit 的 SHA1 函数. 如构造叶子节点为 m 个(m 的值为 2 的整数次幂)随机数的散列树, 则计算量如下: 构造散列树过程需要 $m-1$ 次散列运算; 存储散列树需要 $(2m-1)\times 20$ byte 的存储空间; 初次向管理平台传输带有验证信息的节点需要 $(\log_2[m]+1)\times 20$ byte, 在后续认证过程中, 只需传送 20 byte 的散列值即可. 特别情况下, 当现场控制网络只有惟一的控制中心时, 设备只需构造具有惟一节点的散列树, 因此本方案具有很大的灵活性和适用性. 本方案仅需 AIK 对根节点签名, 对于需要提供多个叶节点的证明情况下, 共用一个根节点数字签名, 节省了计算时间[7].

文献[11] 中针对 Merkle 树计算的效率问题, 给出了 Intel Pentium 4 CPU 1700 MHz 机器上 SHA1 函数运行时间的线性方程: $H=\alpha b+\beta$. 其中 b 为数据块的字节数, $\alpha=0.012\ 234\ 8\ \mu s/byte$, $\beta=1\ \mu s$. 由于 Merkle 树的分枝节点为两个孩子节点连接后的摘要值, 每一个孩子节点的长度为 20 byte, 所以生成一个分枝节点的时间 $H_i=\alpha\times 2\times 20+\beta$. 一个叶子节点数为 m 的满 Merkle 树, 具有个 $m-1$ 个分枝节点, 生成根节点的时间为 $(m-1)\times H_i$.

在管理平台, 只需要记忆一个与其通信的设备平台发送的第一个节点随机数, 并将此随机数与设备平台绑定, 用于再次证明时验证, 即针对不同的管理平台, 设备使用不同的叶子节点, 避免了重放攻击; 管理平台初次验证散列树时需要进行 $\log_2[m]$ 次散列计算, 在以后的认证过程中只需计算一次散列函数. 证明方采取主动证明的方式, 并按照约定的时间间隔向验证方发送度量报告, 因此可以减轻验证方的负担.

将可信计算标准中的远程证明技术引入到现场控制环境中, 能够有效地提高设备状态报告的安全性和可信性, 但是现有远程证明方案中, 只能由管理中心进行轮询, 加重了管理中心的通信和计算的负担. 本文提出的基于 Merkle 散列树的证明方案能够使设备主动、定时地向管理中心报告自己的平台状态, 仅增加了设备少量的存储开销和计算开销, 减轻了管理中心的负担. 在完整性报告上绑定时戳保证了认证的新鲜性, 基于 Merkle 散列树的完整性报告传递保证了认证的完整性.

参 考 文 献

[1] Trusted Computing Group. TCG specification architecture overview[EB/OL]. [2008-12-08]. https://www.trustedcomputinggroup.org/groups/TCG_1_0_Architecture_Overview.pdf.

[2] Trusted Computing Group. TPM main part 1 design principles[EB/OL]. [2008-11-01]. https://www.

trustedcomputinggroup. org/ specs/ TPM/ tpmwgmainrev62- Part1- Design- Principles. pdf.

[3] Merkle R. A certified digital signature[C] //Advances in Cryptology-CRYPTO′89. Berlin: Springer-Verlag, 1989: 218-238.

[4] Sailer R, Zhang X, Jaeger T, et al. Design and implementation of a TCG-based integrity measurement architecture[C] //Proceedings of the 13th USENIX Security Symposium. Sa Diego California: ACM Press, 2004: 223-238.

[5] Schellekens D, Wyseur B, Preneel B. Remote attestation on legacy operating systems with trusted platform modules[J]. Electronic Notes in Theoretical Computer Science (ENTCS), 2008, 197(1): 59-72.

[6] Stumpf F, Fuchs A, Katzenbeisser S, et al. Improving the scalability of platform attestation[C] //Proceedings of the 3rd ACM workshop on Scalable Trusted Computing. Fairfax, VA: ACM, 2008: 1-10.

[7] 徐国愚, 常朝稳, 黄　坚, 等. 基于时间的平台完整性证明[J]. 计算机工程, 2009, 35(6): 153-155.

[8] 谭运猛, 郎为民, 杨宗凯. 基于 Merkle 树的微支付方案[J]. 华中科技大学学报(自然科学版), 2004, 32(6): 27-28.

[9] Chang C, He R, Xie H, et al. A high efficiency protocol for reporting integrity measurements[C] //Proceedings of the Eighth International Conference on Intelligent Systems Design and Applications. Kaohsiung: IEEE Press, 2008: 358-362.

[10] 蔡永泉, 刘　芳. DMSS-动态 Merkle 可信树签名方案[J]. 电子学报, 2009, 37(4A): 97-101.

[11] Williams D, Sirer E G. Optimal parameter selection for efficient memory integrity verification using merkle hash trees[C] //Proceedings of the Third IEEE International Symposium on Network Computing and Applications. Washington: IEEE Press, 2004: 383-388.

(上接第 30 页)

[9] Ge Tong, Chang J S, Shu Wei. PSRR of bridge-tied load PWM class D amps[C] //IEEE International Symposium on Circuits and Systems. Seattle: IEEE, 2008: 284-287.

[10] Chun Kit Lam, Meng Tong Tan. A class D amplifier output stage with low THD and high PSRR [C] //IEEE International Symposium on Circuits and Systems. Taipei: IEEE, 2009: 1 945-1 948.

[11] Oliva A R, Ang S S, Thuy V Vo. A multi-loop voltage-feedback filterless class-D switching audio amplifier using unipolar pulse-width-modulation[J]. IEEE Transactions on Consumer Electronics, 2004, 50(1): 312-319.

[12] Karstin Nielson, Thomas Taul, Michael A E Andersen. A comparison of linear and non-linear control methods for power stage error correction in switching power amplifiers[C] //104th AES Convention. Amsterdam: Audio Engineering Society, 1998: 4 673.

[13] Shu Wei, Chang J S, Tong Ge, et al. Fourier series analysis of the nonlinearities in analog closed-loop PWM class D amplifiers[C] //IEEE International Symposium on Circuits and Systems. Island of Kos: IEEE, 2006, 4: 1 382-1 385.