

一种可信接入云安全协议

常亚楠, 张志勇, 韩林茜

(河南科技大学 信息工程学院, 河南 洛阳 471023)

摘 要: 为提高用户接入云计算的安全性, 利用可信计算的远程证明机制构建云环境下的可信接入认证模型, 提出一种可信接入安全协议. 该协议实现移动用户和云服务器身份与平台的双重认证. 分析表明该协议具有更高的安全性和效率, 并且满足云环境下平台隐私保护需求.

关键词: 云计算; 可信认证; 远程证明; 安全协议

中图分类号: TP391.9

文献标识码: A

文章编号: 1000-7180(2016)02-0034-05

A Trusted Access Cloud Security Protocol

CHANG Ya-nan, ZHANG Zhi-yong, HAN Lin-qian

(Information Engineering College, Henan University of Science and Technology, Luoyang 471023, China)

Abstract: In order to improve the security when users access to cloud computing, constructed of trusted authentication model using remote attestation, and proposed a trusted authentication cloud security protocol. The proposed protocol realizes a two-factor authentication of identity and platform for user and cloud server. It shows that the protocol with higher security and efficiency, as well as satisfies the privacy protection requirement of the platform configuration in cloud environment.

Key words: cloud computing; trusted access; remote attestation; security protocol

1 引言

云计算用户共享云中资源, 导致隐私机密信息遭到非法访问, 所以要增强云计算的身份认证^[1]. 云计算主要采用传统的用户名/口令认证方式^[2]. 文献[3]利用图像与文字结合增加云计算接入安全级别, 但是该方案中云平台受到拒绝服务攻击. 于是, 文献[4]提出一种数据屏蔽访问控制机制, 避免未授权用户的访问使用. 文献[5]虽然实现了用户和云服务器的身份认证, 但没有验证他们的平台是否可信, 这对于云环境下的用户和服务器来说, 即使通过合法的身份认证, 并不能保证其平台是可信的. 文献[6]仅实现了对终端用户的平台验证. 因此本文提出一种可信接入云安全协议, 该协议不仅实现双向的身份认证, 而且对通信平台双方进行可信验

证, 同时也满足云计算平台的隐私安全需求, 安全性高. 在完成相同功能的前提下, 在计算量上减少哈希运算和连接运算次数, 缩短运算时间, 减少通信时延, 性能优势明显.

2 云计算可信接入模型

云环境计算模式与传统模式不同, 在云计算的虚拟状态下, 用户将重要的数据信息及复杂的计算任务交给云服务器, 用户失去对其数据的物理控制, 其敏感数据会带来隐私保护及信任安全问题, 因此必须加强对云服务的安全认证^[7]. 而移动终端频繁的断开、连接和移动性特征需要进一步增强终端安全接入^[8]. 本节提出一种面向云计算环境的可信认证远程证明模型, 如图 1 所示.

该模型实现了移动用户与云服务器双向的身份

收稿日期: 2015-04-04; 修回日期: 2015-05-20

基金项目: 国家自然科学基金(61370220); 河南省高校科技创新团队支持计划项目(15IRTSTHN010); 河南省科技创新人才计划杰出青年基金(134100510006); 河南省科技攻关项目(142102210425); 河南省教育厅科学技术研究重点项目基础研究计划(13A520240, 14A520048); 河南科技大学科研创新能力培育基金(2013ZCX022)

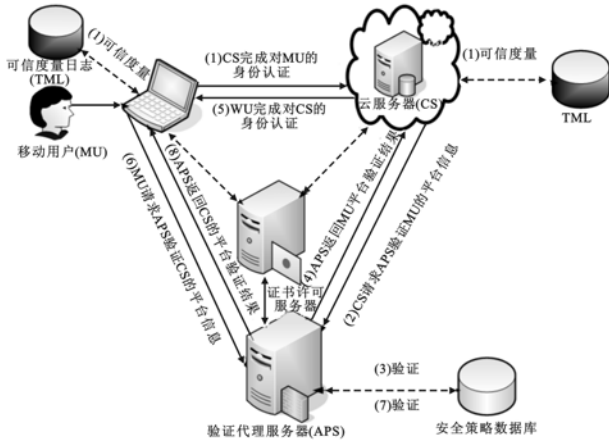


图1 云计算可信认证模型

与平台认证. 包括带有 TPM 芯片的移动用户、云服务器、智能卡及验证代理服务器(APS)、证书与许可服务器、可信度量日志(TML)、完整性度量参考与安全策略数据库. 将移动用户(Mobile User, MU)和云服务器(Cloud Server, CS)的平台完整性度量值及获得的安全属性存储在 TML 中作为平台认证消息. 设备完整性度量值与安全策略值存储在 APS 的完整性度量参考与安全策略数据库中作为 APS 对平台身份验证的参考标准值. MU 和 CS 在本地进行完整性度量,把度量值以 PCRs 的形式存储在 TML 中. (1)首先智能卡在本地系统验证 MU 的口令和密码,验证通过后,把 MU 的身份信息和度量的完整性平台信息发送给 CS. (2)待 CS 完成 MU 的身份认证后把 MU 的平台信息发送给 APS,请求 APS 验证 MU 平台的身份. (3)APS 把 MU 的平台信息与安全策略数据库比较,完成对 MU 平台身份验证. (4)为避免泄露 MU 平台的隐私信息,APS 以平台状态报告的形式发送给 CS,CS 根据 APS 发送的平台状态报告完成对 MU 的平台认证. 步骤(1)~(5)实现了 CS 对 MU 的身份与平台的双重认证. 同理, (6)CS 将其身份信息和度量的完整性平台信息发送给 MU,完成 CS 的身份认证后把 CS 的平台信息发送给 APS,请求 APS 验证 MU 平台的身份. (7)APS 把 CS 的平台信息与安全策略数据库比较,完成对 CS 平台身份验证. (8)APS 把 CS 的平台信息以平台状态报告的形式发送给 MU, MU 根据 APS 发送的平台状态报告完成对 CS 的平台认证. 步骤(6)~(8)实现了 MU 对 CS 的身份与平台的双重认证.

3 云计算可信接入安全协议

$\text{Cert}(\text{MU. AIK})$ 、 $\text{Cert}(\text{CS. AIK})$ 和 $\text{Cert}(\text{APS. AIK})$ 分别表示 MU、CS 和 APS 证明身份密钥 AIK 证书. $\text{SK}^{\text{MU. AIK}}$ 、 $\text{SK}^{\text{CS. AIK}}$ 、 $\text{SK}^{\text{APS. AIK}}$ 分别表示 MU、CS 和 APS 的 AIK 私钥; $\text{PK}^{\text{MU. AIK}}$ 、 $\text{PK}^{\text{CS. AIK}}$ 、 $\text{PK}^{\text{APS. AIK}}$ 分别表示 MU、CS 和 APS 的 AIK 公钥. $K^{\text{MU-APS}}$ 、 $K^{\text{CS-APS}}$ 表示 MU、CS 与 APS 在平台验证前产生的共享密钥. x, y 分别为 MU 和 CS 的私钥. MU 成功登陆 CS 前,需要通信双方进行身份和平台认证. 包括注册、接入认证和口令更新三个阶段:

3.1 注册阶段

MU 登陆 CS 前,需要先在 CS 注册. 具体注册过程如下所示.

3.1 注册阶段

MU 登陆 CS 前,需要先在 CS 注册. 具体注册过程如下所示.

(1) MU 计算 $h(\text{pw} \oplus x)$, 把消息 $\{\text{ID}, h(\text{pw} \oplus x), h(x), \text{Cert}(\text{MU. AIK})\}$ 发送给 CS.

(2) CS 检查证书 $\text{Cert}(\text{MU. AIK})$, 若证书不在有效期内不允许注册. 检查 ID 是否存在, 否则需要 MU 重新注册.

(3) CS 计算 $I = h(\text{ID} \oplus h(\text{PW} \oplus x))$, $J = h(\text{ID} \parallel y)$ 和 $V = gh(I \parallel J) + h(I) + h(J) \bmod p$, CS 把 $\{I, J, V, p, g, h(\cdot)\}$ 存储在智能卡中.

(4) MU 把 x 输入智能卡, 智能卡的信息为 $\{I, J, V, p, g, h(\cdot), x\}$.

3.2 接入认证阶段

在 MU 成功登陆 CS 前,需要通信双方进行身份和平台认证. 具体流程如下所示.

(1) MU 插入智能卡, 输入用户账号 ID 和密码 PW, 本地系统计算 $I1 = h(\text{ID} \oplus h(\text{PW} \oplus x))$, 并检查是否 $I1 = I$, 否则, 终止用户的登陆.

(2) MU 生成一个随机数 Nonce; 计算 $U = h(I1 \parallel J)$, $V1 = Vg - h(x) \bmod p$, $R = h(\text{Nonce} \parallel V1)$. MU 对平台进行本地完整性度量, 其度量值放在 PCRs 中, 并获取安全属性特征 Secure Attributes, 将这个�程写入 TML 中; 用 $\text{SK}^{\text{MU. AIK}}$ 对平台信息 PCRs、SecureAttributes 与包含有平台标识值 TML 等内容签名 $\text{Sig}_{\text{MU}} = (\text{PCRs} \parallel \text{SecureAttributes} \parallel \text{TML} \parallel \text{Nonce}) \text{SK}^{\text{MU. AIK}}$, 并用 $K^{\text{MU-APS}}$ 加密 Sig_{MU} 为 $\text{MU}_{\text{TPM}} = (\text{Sig}_{\text{MU}}, \text{PCRs}, \text{SecureAttributes}, \text{TML}) K^{\text{MU-APS}}$. 把消息 M1 发送给 CS.

$M1 = \{\text{MU}_{\text{TPM}}, \text{Cert}(\text{MU. AIK}), U, h(R)\}$ (1)

(3) CS 首先验证 $\text{Cert}(\text{MU. AIK})$ 的有效性, 否则终止会话. CS 生成随机数 y , 计算 $V2 = g^{U+h(y)} \bmod p$, $R * = h(\text{Nonce} \parallel V2)$; 判断 $h(R *) = h(R)$ 是否成立, 否则对 MU 的身份认证失败. 最后, CS 把消息 M2 发送给 APS.

$M2 = \{\text{MU}_{\text{TPM}}, \text{Cert}(\text{CS. AIK})\}$ (2)

(4) APS 首先判断 $\text{Cert}(\text{CS.AIK})$ 的有效性,再用 $\text{PK}^{\text{MU.AIK}}$ 验证 MU 签名的平台身份信息.通过参考安全策略数据库,验证平台当前的配置信息.最后把 MU 的平台状态值 MU_platformStatus 及用其私钥签名的平台状态值 $(\text{MU_platformStatus} || \text{Nonce})$ $\text{SK}^{\text{APS.AIK}}$ 和 $\text{Cert}(\text{APS.AIK})$ 发送给 CS.

$$\text{M3} = \{ (\text{MU_platformStatus} || \text{Nonce}) \text{SK}^{\text{APS.AIK}}, \text{MU_platformStatus}, \text{Cert}(\text{APS.AIK}) \} \text{K}^{\text{CS-APS}} \quad (3)$$

(5) CS 首先验证 $\text{Cert}(\text{APS.AIK})$ 的有效性,用 $\text{PK}^{\text{APS.AIK}}$ 解密经 APS 签名的平台状态值判定 MU 平台身份是否可信.若 MU 平台身份是可信的,CS 对自身平台进行本地完整性度量,其度量散列值存放在中 PCRs,并获取安全属性特征值 SecureAttributes ,将这个过程写入 TML 中;使用 $\text{SK}^{\text{CS.AIK}}$ 对 PCRs, SecureAttributes 与包含有平台标识值的 TML 等内容进行签名 $\text{Sig}_{\text{CS}} = (\text{PCRs} || \text{SecureAttributes} || \text{TML} || \text{Nonce}) \text{SK}^{\text{CS.AIK}}$,用 $\text{K}^{\text{CS-APS}}$ 加密 Sig_{CS} 、PCRs、 SecureAttributes 和 TML 即 $\text{CS}_{\text{TPM}} = (\text{Sig}_{\text{CS}}, \text{PCRs}, \text{SecureAttributes}, \text{TML}) \text{K}^{\text{CS-APS}}$,把消息 M4 发送给用户.

$$\text{M4} = (\text{CS}_{\text{TPM}}, \text{Cert}(\text{CS.AIK}), h(\text{V2})) \quad (4)$$

(6) MU 验证 $\text{Cert}(\text{CS.AIK})$ 的有效性.计算 $\text{V1} = \text{Vg}^{-h(x)} \text{mode } p$,判断 $h(\text{V1}) = h(\text{V2})$ 是否成立,否则对 CS 身份认证失败.把消息 M5 发送给 APS.

$$\text{M5} = \{ \text{CS}_{\text{TPM}}, \text{Cert}(\text{MU.AIK}) \} \quad (5)$$

(7) APS 判定 $\text{Cert}(\text{MU.AIK})$ 的有效性,再用 $\text{PK}^{\text{CS.AIK}}$ 验证 CS 签名的平台身份信息.通过参考安全策略数据库,验证平台当前平台配置信息.最后把 CS 的平台状态值 CS_platformStatus ,用其私钥签名的平台状态值 $(\text{CS_platformStatus} || \text{Nonce})$ $\text{SK}^{\text{APS.AIK}}$ 和 $\text{Cert}(\text{APS.AIK})$ 发送给 MU.

$$\text{M6} = \{ (\text{CS_platformStatus} || \text{Nonce}) \text{SK}^{\text{APS.AIK}}, \text{CS_platformStatus}, \text{Cert}(\text{APS.AIK}) \} \text{K}^{\text{MU-APS}} \quad (6)$$

(8) 用户验证 $\text{Cert}(\text{APS.AIK})$ 的有效性,通过 APS 的平台状态报告判定 MU 平台身份是否可信,完成对 CS 平台身份的认证, MU 成功登陆到 CS.如图 2 所示可信接入认证图.

3.3 口令更新阶段

用户插入智能卡在本地系统完成口令更新.具体如下步骤:

(1) MU 输入 ID、PW,计算 $I * = h(\text{ID} \oplus h(\text{PW} \oplus x))$.

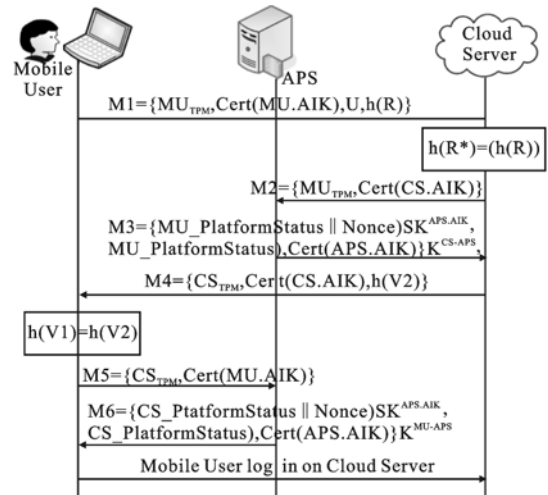


图2 云计算可信认证时序图

(2) 本地系统验证是否 $I * = I$, 否则拒绝请求.

(3) MU 输入新密码 PW 并生成新的秘钥 x' .

(4) 计算 $I' = h(\text{ID} \oplus h(\text{PW} \oplus x'))$.

(5) I 和 x 代替智能卡中原来的 I 和 x , 完成口令更新.

4 协议分析

4.1 安全性分析

可抵御具有特权内部攻击者攻击:在协议的注册阶段,用户把 ID 和 $h(\text{PW} \oplus x)$ 通过安全信道传送给云服务,用户的密码并非以明文的方式出现,因此,即使是云服务的特权用户也不知道用户密码,能够有效抵御云服务器中特权用户的攻击.

可抵御密钥猜测攻击:该密钥由云服务和客户端共同协商,并且客户端的 x 和服务端端的 y 都是受单向哈希函数 $h(\cdot)$ 的保护,对于非法用户是不可能推测出 x 和 y .

可抵御并行会话攻击:在协议的整个认证过程中,双方的挑战值及传送的消息从未以明文的方式出现,攻击者不可能得到任何有用信息冒充合法用户或者服务器.

可抵御重放攻击:在用户和服务器的整个认证过程是挑战—应答机制,并且随机数 Nonce 有效抵抗重放攻击.

可抵御中间人攻击:在消息中 M1 中,即使攻击者知道传输的 U 、 V 也不能推算出 $V2$,因为 $V2$ 是由 y 生成,除了服务器之外,没有知道 y 值,所以中间人攻击不成立.

可抵御拒绝服务攻击:在登陆云服务器之间,本地系统就先对用户的合法性进行验证,能够有效地

抵抗拒绝服务攻击。

可抵御钓鱼攻击:因为在用户认证服务器时需要服务器提供 $h(\cdot)$,而只有真正的服务器能够提供正确值。

可抵御共谋攻击:终端用户使用平台私钥 SKMU. AIK 签名的内容包括平台身份唯一标识的 TML,因此即使终端用户拥有多台终端(可信终端和不可信终端)也无法通过传递的可信平台度量报告获取验证代理者 APS 的平台验证,从而有效阻止了共谋攻击。

身份与平台的双重认证:该方案通过判断 $h(R^*) = h(R)$ 实现了用户和云服务之间双向的身份认证;用可信第三方实现对用户和云服务器平台的完整性验证,确保用户和云服务所在平台的可信性。

平台隐私保护:对双方通信平台的认证是由第三方代理完成,并且以平台的状态报告形式发送给通信方,避免平台配置信息被通信另一方直接获知,有效实现平台隐私信息的保护。如表1所示的相关文献方案在抵抗的攻击类型和功能机制方面的安全性分析表。

表1 安全性与功能机制分析

抵御攻击类型	文献[9]	文献[5]	文献[6]	本文方案
秘钥猜测攻击	✓	✓	✓	✓
并行会话攻击	×	×	×	✓
重放攻击	✓	✓	✓	✓
中间人攻击	✓	✓	×	✓
拒绝服务攻击	✓	×	✓	✓
钓鱼攻击	✓	✓	×	✓
共谋攻击	×	×	✓	✓
双向身份与平台认证	×	×	×	✓
平台隐私保护	×	×	✓	✓

4.2 性能分析

该协议已应用到云媒体社交网络系统中协议的效率主要用协议执行过程中系统完成的各种计算来衡量。其中“H”表示散列运算,“C”表示连接运算,“XR”表示异或运算,用“EX”表示模指数运算,“EK”表示对称加密运算,“DK”表示对称解密运算,用CPU的计算量来衡量协议的效率。本文方案与已有方案的计算性能对比分析,其结果如表2所示。

表2 计算性能分析表

相关文献	主机 CPU 计算量
文献[9]	11H + 6C + 2XR
文献[5]	6H + 3C + 2XR + 3EK + 2DK
文献[6]	4C + 6EK + 6DK
本文方案	6H + 2C + 2EX + 1XR + 2EK + 2DK

该协议执行了6次哈希运算、2次连接操作、2次模指数运算、1次异或操作、2次解密运算,计算量相对较小。与文献[9]相比减少了5次哈希运算,而哈希运算的复杂度要远远高于其他运算,综合考虑该协议的效率高于文献[5]。文献[5]仅实现身份认证,若增加平台的可信验证功能,那么至少增加2次签名运算和2次解密运算,总的计算量仍然比本方案大。文献[6]仅实现对终端用户平台的完整性验证,计算量比较小,若同时实现用户和服务器身份认证,计算量也比本方案大。该方案实现了终端用户和云媒体服务器双向的身份与平台认证,虽然增加了一定的消息负载,但是所增加的消息负载提供了平台的可信验证,且部分参数通过预计算得到,平台CPU与TPM同时完成部分运算,一定程度地缩短运算时间,减少通信时延。综合来看,本方案具有更高的效率并且满足云计算环境下移动终端可信接入需求。

5 结束语

本文提出一种双向可信接入安全协议,由可信第三方完成对平台的完整性验证,实现终端用户和云服务器平台与身份双向认证,满足云平台的隐私保护需求,安全性高并且在计算性能上具明显优势。但是,本方案采用的是静态可信验证机制,忽略了文件系统安全问题,下一步工作主要放在协议的形式化证明上,进一步分析协议的安全性。

参考文献:

- [1] Lin G, Wang D, Bie Y, et al. MTBAC: A mutual trust based access control model in Cloud computing [J]. Communications, 2014, 11(4): 154-162.
- [2] Ghazizadeh E, Zamani M, Ab Manan J, et al. Trusted computing strengthens cloud authentication[J]. The Scientific World Journal, 2014(260187): 1-17.
- [3] Popescu D E, Lonea A M. An hybrid text-image based authentication for cloud services[J]. International Journal of Computers Communications, 2013, 8(2): 263-274.

- [4] Wang Z W, Sha KW, Lv W. Slight homomorphic signature for access controlling in cloud computing[J]. Wireless Personal Communications, 2013, 73(1): 51-61.
- [5] Lee S, Kim T R, Lee H J. Mutual authentication scheme for cloud computing[C]// 2013 International Conference on Future Information and Communication Engineering. China, Shenyang, 2013:149-157.
- [6] Hang T, Zhang Z Y, Chen Q L, et al. A method for trusted usage control over digital contents based on cloud computing[J]. International Journal of Digital Content Technology and its Application, 2013, 7(4):795-802.
- [7] Xiao Y L, Wang Y M, Pang L J. Verification of trusted network access protocols in thre strand space mode[J]. IEICE transactions on Fundamentals of Elecronics, Communications and Computer Sciences, 2012, E95-A(3): 665-668.
- [8] Song W T, Hu B. Approach to detection type-flaw attacks based on extended strand spaces[J]. The Computer Journal, 2014(29):84-106.
- [9] Choudhury A J, Kumar P, Sain M, et al. A strong user authentication framework for cloud computing[C]// 2011 IEEE Asia-Pacific Services, Computing Conference. Jeju Island, Korea, 2011:110-115.

作者简介:

常亚楠 女, (1989-) 硕士研究生. 研究方向为云计算安全与可信计算. E-mail: givechangyanan11@163.com.

张志勇 男, (1975-) , 博士后, 硕士生导师. 研究方向为多媒体云计算、可信计算与可信网络连接、数字版权管理等.

韩林茜 女, (1989-) , 硕士研究生. 研究方向为多媒体社交网络与云计算安全.

(上接第 33 页)

器的输出误差 205.9 mV, 线性化输出的误差约 300 μ V, 远远小于前者.

4 结束语

本文分析了传统 4T-APS 像素电路中源极跟随器输出的非线性, 提出了一种新的 CMOS 图像传感器中 4T-APS 线性化技术, 结构简单, 易于实现, 相对于文献[4]的方法, 本方法减少了感光阵列向外的金属布线, 避免了填充率下降, 保证了像元量子效率, 从仿真结果看, 线性化技术处理后输出误差从 205.9 mV 降低到 300 μ V, 从而保证了 CMOS 图像传感器成像质量, 因此可应用于高精度 CMOS 图像传感器中.

参考文献:

- [1] 袁杰, 陈浩洋. 提高 CMOS 图像传感器动态范围和线性度的装置和方法: 中国, CN10800837A[P]. 2010.
- [2] 陈雷, 韩泽耀, 曹庆红. 基于 4T 像素结构的 CMOS 图像传感器设计[J]. 电子科技, 2007, 25(10): 57-63.
- [3] 拉扎维, 模拟 CMOS 集成电路设计[M]. 陈贵灿, 程军, 张瑞智, 译. 西安: 西安交通大学出版社, 2003.
- [4] Morteza Afghahi, Tempe, Ariz imaging system with gain and error correction circuitry: USA Patent, US006111242A[P]. 2000.

作者简介:

闻超 男, (1990-) , 硕士研究生. 研究方向为模拟集成电路设计. E-mail: 1002361761@163.com.