无证书线性同态聚合签名方案研究

茅 磊

（江苏建筑职业技术学院 信电工程学院，江苏 徐州 221116）

摘 要：在云计算和大数据时代，密码学中的数字签名技术发挥着不可替代的作用。数字签名不仅可以保障数据的真实性和不可否认性，而且可用于实现云存储中各种外包数据完整性的公开审计。从云存储实际需求出发，设计一种适用于云存储中多用户数据公开批验证的无证书线性同态聚合签名方案。在安全性上，该签名方案可抵抗来自聚合签名者内部的“合谋攻击”；在效率上，其可对不同用户各种数据产生的签名进行压缩和聚合。在JPBC库下进行仿真实验，结果显示该方案的验证速度比普通无证书数字签名提高近10倍。此外，该方案只需存储所有数据同态聚合后的一个数字签名，相较于传统的“一块数据，一个签名”的存储方式，从根本上节约了用户在云上的存储空间。

关键词：信息安全；密码学；线性同态签名；聚合签名；无证书密码体制

DOI：10.11907/rjdk.211255 **开放科学（资源服务）标识码（OSID）：**

中图分类号：TP309 **文献标识码：**A **文章编号：**1672-7800（2022）002-0159-06

Research on the Certificateless Linear Homomorphic Aggregate Signature Scheme

MAO Lei

（*School of Information and Telecommunications Engineering, Jiangsu Vocational Institute of Architectural Technology, Xuzhou* 221116, *China*）

Abstract: In the era of cloud computing and big data, digital signature technology plays an irreplaceable role in cryptography. Digital signatures can not only guarantee the authenticity and non-repudiation of data, but are now more used to implement public audits of the integrity of various outsourced data in cloud storage. Starting from the actual needs of cloud storage, this article designs a certificateless linear homomorphic aggregation signature scheme suitable for public batch verification of multi-user data in cloud storage. The security of the signature scheme can resist internal signatories from the aggregation. Collusion attack"; in terms of efficiency, this scheme can compress and aggregate the signatures generated by various data of different users. Through simulation experiments under the JPBC library, it shows that this scheme improves the verification speed nearly 10 times compared with ordinary certificateless digital signatures. Besides, in terms of storage cost, the scheme only needs to store a digital signature after homomorphic aggregation of all data. Compared with the traditional "one piece of data, one signature" storage method, it fundamentally saves users′ storage space on the cloud.

Key Words: information safety; cryptography; linear homomorphic signature; aggregate signature; certificateless cryptosystem

0 引言

随着计算机和互联网技术的不断发展，大数据、云计算、物联网、人工智能等新兴技术步入人们生活。云存储[1-2]是云计算在概念上的延伸与发展。云存储是通过集群应用、网络技术和分布式文件系统，将网络中大量不同类型的存储设备通过软件集合到一起，共同对外提供数据存储和业务访问功能的一个系统。目前，云存储已逐步实现商业化，走进了人们的日常生活中，国内已涌现出一批云存储服务提供商，如百度云、阿里云等。越来越多的企事业单位也习惯将数据搬至云端存储[3]，实现数据外包，不仅可以节约自身存储空间，降低数据维护管理代价，还可以在任何时间、任何地点通过任何可接入互联网的设备高效便捷地访问云端数据。随着云存储的不断商业化，云中数据完整性的保护越来越受到用户关注。基于此，本文采

收稿日期：2021-03-01

基金项目：江苏省高等学校大学生创新创业训练计划项目（202010849026Y）；江苏建筑职业技术学院青年专项（JYQZ20-02）

作者简介：茅磊（1986-），男，硕士，江苏建筑职业技术学院信电工程学院助教，研究方向为密码学与信息安全。

用密码学中的数字签名技术，设计出一种适用于云存储的高效安全文件完整性审计方案。

1 相关研究

数字签名[4-5]是公钥密码学的重要研究方向，其可对被签名文件数据提供真实性、完整性以及不可否认性的技术服务。随着研究的不断深入，许多适用于不同使用环境的特殊数字签名，如同态签名、聚合签名、无证书体制下的数字签名等应运而生。这些数字签名既有普通签名的功能，也有自身独有的特点，若将其有机组合并加以灵活使用，在云存储环境中可起到事半功倍的效果。

1.1 同态签名

同态签名由 Johnson 等[6]提出。设数字签名的消息空间 $\boldsymbol{M}$ 和签名空间 $\sum$ 上的二元运算符分别为 $\oplus$ 和 $\otimes$，有两个来自 $\boldsymbol{M}$ 和 $\sum$ 上的消息签名对 (m_1,σ_1) 和 (m_2,σ_2)，其中 $\sigma_1 = f(m_1),\sigma_2 = f(m_2)$，若签名算法 f 是代数系统 $(\boldsymbol{M},\oplus)$ 到 $(\sum,\otimes)$ 上的同态映射，则有 $f(m_1\oplus m_2) = f(m_1)\otimes f(m_2) = \sigma_1 \otimes \sigma_2$ 成立。从这一点不难看出，同态签名在一定程度上放宽了数字签名的安全性。一般数字签名方案在安全性上要求达到在适应性选择消息下的存在性不可伪造，而同态签名则要求在同一数据集中，各消息的签名可以由其他已知的消息签名导出，不必再使用复杂的签名算法生成。但在该数据集外，即在不同数据集中的消息签名不能由其他数据集中的消息签名导出，只能由数字签名算法生成。根据目前密码学界的观点，上述同态签名可以分为线性同态签名、多项式同态签名以及全同态签名 3 类[7]，以线性同态签名使用最多最广。线性同态签名是针对一个线性子空间基中的各个向量进行签名，是一种高效快捷的轻量级同态签名方案。例如，Wu 等[8]使用无证书线性同态签名设计了一种抵抗网络编码污染的签名方案，并探讨了该方案在物联网上的应用；Lin 等[9]研究了标准模型下基于格的线性同态签名方案，该方案具有较短的公钥。

同态签名在云存储中也可发挥重要作用。云存储环境中生成的数据文件往往是动态增长的，如果采用普通数字签名保证动态增长数据的完整性并实现数据的可公开验证效果必然不理想。如图 1 所示，假设用户 A 租用云服务器存储自身每天产生的数据，第 1 天用户 A 产生数据块 m_1，第 2 天产生的数据块记为 m_2，以此类推。如果采用一般的数字签名保障数据的可公开验证，该用户需每天对产生的数据进行签名，并将数据和签名均存储在云服务器上。而在验证数据完整性时，又要从云服务器上依次下载这些数据及其对应的签名。若需验证 3 个数据块的完整性，则要分别下载 3 个数据块和 3 个签名，并进行 3 次验证运算。采用同态签名时，每产生一部分新数据就要使用签名算法产生对应的签名并将其存储在云端，而验证时又要下载所有数据块及其对应的签名，还要进行与数据块相同数量级的验证运算，势必会在用户端产生大量验证运算，消耗更多云存储空间，而且下载数据和签名时也会占用更多网络带宽。

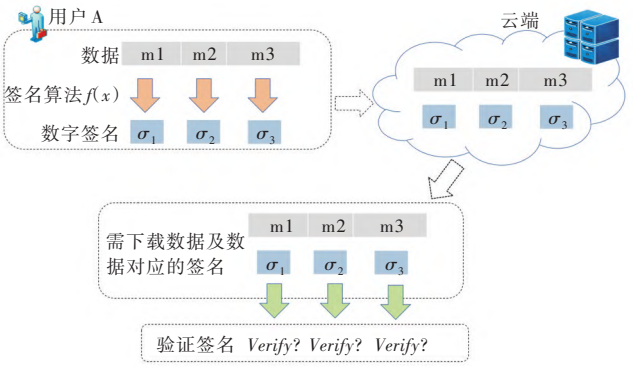

Fig. 1 Using ordinary digital signature to verify data integrity

图 1 使用普通数字签名验证数据完整性

1.2 聚合签名

聚合签名概念由 Boneh 等[10]于 2003 年提出，旨在提高验证大量单个签名的效率。使用普通数字签名方案进行签名合法性验证时，需要对每个签名逐个验证，在云存储中进行数据审计时使用该种方案的繁琐程度和运算存储代价难以承受[11-12]。如果采用聚合签名，则可将来自不同用户的数字签名压缩成一个签名，验证聚合后的签名等同于验证所有聚合前单个用户的签名。使用聚合签名不仅可以保证云存储中数据的完整性[13]，还可实现在云端数据审计工作的公开批量验证。随着对聚合签名研究的深入，Zhang 等[14]发现了来自聚合签名者内部的合谋攻击，并在此基础上提出新的聚合签名安全模型，认为当且仅当每个被聚合的单个签名是合法签名时，最后的聚合签名才是合法的；曹素珍等[15]提出无证书高效聚合签名方案，虽然效率较高，但仍不能抵抗合谋攻击；吴戈[16]设计了能抵抗合谋攻击的基于身份与无证书的聚合签名。

1.3 无证书密码体制

无证书密码体制是 Al-Riyami 等[17]针对身份密码体制的不足所提出的改进方案。基于身份的密码体制往往需要假定密钥生成中心完全可信，这是由于用户密钥完全掌握在密钥生成中心（KGC）手中[18]。但在云存储中通常认为云服务器是不可靠的，云服务商存在有意无意损害用户存储在云端数据的行为（如不可抗的自然灾害、云服务器本身的物理损害、黑客攻击等）。如果使用基于身份的数字签名，当用户数据发生损坏时，云服务商可能会使用身份系统中的用户完整私钥逃避责任[19]，这是由于云服务商掌握用户的签名私钥，当发生数据损坏时，“不诚实”的云存储服务商可能会使用用户签名私钥对损坏后的数据重新签名，而用户却全然不知。无证书密码系统不仅可以方便地搭建在云存储服务系统中，还能克服身份密码系统中密钥管理中心权利过于集中的缺点。

2 方案设计思路

上述 3 种密码学技术各有所长：无证书密码体制可以保障用户签名私钥的安全，抵抗不诚实的 KGC；同态签名

可以压缩同一数据集内部动态增长的各数据签名;聚合签名可以将不同数据集中的数字签名进一步压缩。因此,本文结合以上3种技术设计出一种适用于云存储中数据审计的无证书线性同态聚合签名。

如图1所示,首先将无证书密码体制搭建在云存储系统上,由云服务商生成无证书密码系统的相关参数,并产生用户签名时的部分私钥。将私钥通过安全信道传送给用户,用户自己生成并掌握签名的秘密值。用户的签名完整私钥由部分私钥和秘密值组成,除用户以外,没有任何人再能掌握完整的签名私钥。因此,将无证书密码体制部署在云存储系统中可避免云服务商逃避责任。

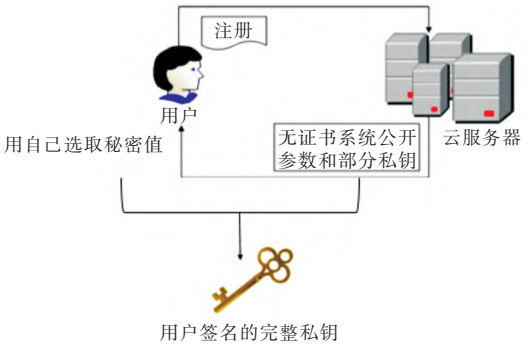

Fig. 2　Deploying certificateless system on cloud server

图2　在云服务器上部署无证书系统

然后,使用同态签名解决同一数据集下动态增长数据的签名问题。如图3所示,用户 A 可将每天产生的数据看作同一数据集(数据集理解为文件夹,同一数据集中的数据即为同一文件夹中的数据,数据集名称可理解为文件夹名称)。用户可使用同态签名中的同态组合算法(Homomorpic-Combine)将3天内生成的数据签名压缩成一个签名 σ_A^*,再将这个签名与3个数据块上传至云服务器中保存。验证完整性时,只需下载对应数据块和一个同态组合签名,并仅进行一次验证。由此可见,使用同态签名可减少同一数据集中数据验证的运算代价、存储代价和网络传输带宽。

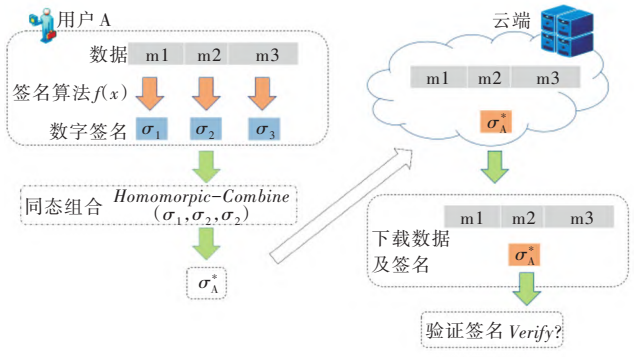

Fig. 3　Using homomorphic signature to verify data integrity

图3　使用同态签名验证数据完整性

最后,基于聚合签名的特点,将来自不同数据集的同态签名进一步聚合压缩。将同态签名与聚合签名结合起来形成同态聚合签名,可解决在云存储中多方数据公开批验证的问题。如图4所示,假设需要验证N个用户数据,只需使用聚合签名算法 *Aggregate* 将这N个用户数据生成的同态组合签名聚合成一个签名然后验证一次即可。因此,使用无证书线性同态聚合签名可以在保障云存储中数据块完整的前提下减少签名验证次数,提高审计效率。

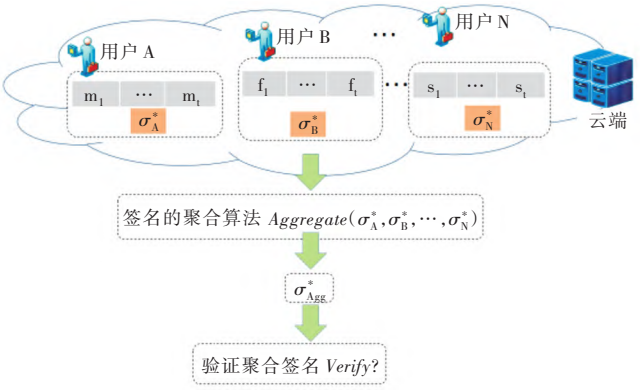

Fig. 4　Multi user data integrity verification using aggregate signature

图4　使用聚合签名验证多用户数据完整性

3　无证书线性同态聚合签名的形式化定义

无证书线性同态聚合签名系统由以下9个概率多项式算法组成,分别为:

(1) *Setup*:密钥生成中心(KGC)输入为系统的安全参数 1^k,产生系统的公开参数 params 和系统主私钥 msk。

(2) *PartialPrivateKeyExtract*:输入用户身份 ID,KGC 生成身份为 ID 用户的部分私钥 D_{ID}。

(3) *ProduceSecretValue*:用户运行生成秘密值 r_{ID}。

(4) *ProducePublicKey*:用户运行生成公钥 PK_{ID}。

(5) *Sign*:用户输入身份 ID、部分私钥 D_{ID}、秘密值 r_{ID},以及待签名的消息向量 $m \in Z_p^n$ 和数据集标签 τ 后,输出消息向量 m 的签名 σ。

(6) *Homomorpic-Combine*:用户输入身份 ID、公钥 PK_{ID}、数据集标签 τ 以及二元组 $(\beta_i,\sigma_i)_{i=1}^n$(其中 $\beta \in Z_p$),算法输出同态算法组合后的签名 σ。

(7) *Verify*:输入用户身份 ID、用户公钥 pk_{ID}、数据集标签、消息向量 m、签名 σ 和二元组 $(\beta_i,\sigma_i)_{i=1}^n$,算法输出 1(接受)或 0(拒绝)。

(8) *Aggregate*:输入来自不同用户集 $U = u_1,u_2,\cdots,u_q$,身份为 ID_i,公钥为 PK_{ID_i}(其中 $i = 1,2,\cdots,q$)的合法签名,输出这些用户在消息—签名对 $(m_i,q_i)_{i=1}^q$ 上的聚合签名 σ_{Agg}。

(9) *AggregateVerify*:聚合签名验证者输入在 $\{(m_i,ID_i,PK_{ID_i})_{i=1}^q\}$ 上的聚合签名 σ_{Agg},算法输出 1(接受)或 0(拒绝)。

4　无证书线性同态聚合签名的安全模型

聚合签名的作用是将若干签名者生成的单个签名压缩成一个签名,因此只有当每个参与聚合的无证书线性同态签名合法时,生成的无证书同态聚合签名才合法。无证

书同态聚合签名的特性可分为无证书同态签名方案的安全性和聚合算法的安全性两部分。

4.1 无证书同态签名的安全模型

对于无证书密码体制的安全模型而言，一般有两类攻击者：第一类攻击者可以替换系统的公钥，但不能访问主私钥；第二类攻击者则可以访问主私钥，但不能替换系统的公钥，通常又被称为“诚实而又好奇”的攻击者。同态签名的基本安全性要求为在适应性选择数据集下的存在性不可伪造，因此无证书同态签名方案在适应性选择身份、适应性选择数据集下存在性不可伪造的安全模型可以通过挑战者C和攻击者（$\mathbf{A}_{\mathrm{I}}$或$\mathbf{A}_{\mathrm{II}}$）之间的游戏来刻画。

4.1.1 游戏1（第一类攻击者A_{I}）

（1）系统建立。挑战者C获得系统的安全参数1^k后，生成公开参数*params*、系统主私钥*msk*、保密主私钥*msk*，将公开参数*params*发送给攻击者$\mathbf{A}_{\mathrm{I}}$。

（2）询问。攻击者$\mathbf{A}_{\mathrm{I}}$可向挑战者C进行创建用户询问（CU）、部分私钥提取询问（PPK）、秘密值询问（SVK）、公钥询问（PK）、替换用户公钥询问（PKR）、签名询问（Sign）。挑战者C需要模拟随机预言器，给出上述方案中各算法的正确回答。在签名询问时，挑战者C要模拟超级签名预言器，即攻击者$\mathbf{A}_{\mathrm{I}}$只提供用户身份*ID*和消息向量$\boldsymbol{m}$，而不提供替换公钥后对应的秘密值，挑战者C需要产生当前用户公钥（一般是被$\mathbf{A}_{\mathrm{I}}$替换后的公钥）下该数据集中消息向量$\boldsymbol{m}$的合法签名σ，并发送给攻击者$\mathbf{A}_{\mathrm{I}}$。

（3）伪造输出。攻击者$\mathbf{A}_{\mathrm{I}}$输出一个伪造（ID*，$\mathrm{pk}_{\mathrm{ID}^*}$，$\tau^*$，m*，$\sigma^*$），在以下情况成立时攻击者$\mathbf{A}_{\mathrm{I}}$在游戏中获胜：①$\sigma^*$是在挑战身份ID*和对应公钥$\mathrm{pk}_{\mathrm{ID}^*}$下，由$\tau^*$标记的数据集中消息m*的合法签名；②攻击者$\mathbf{A}_{\mathrm{I}}$没有询问过挑战身份ID*的部分私钥；③$\tau^* \neq \tau_i$，即攻击者$\mathbf{A}_{\mathrm{I}}$没有询问过身份为ID*、公钥$\mathrm{pk}_{\mathrm{ID}^*}$下由$\tau^*$标记的数据集m*上消息的签名。

4.1.2 游戏2（第二类攻击者A_{II}）

（1）系统建立。*C*获得系统的安全参数1^k后，生成公开参数*params*、系统主私钥*msk*、保密主私钥*msk*，将公开参数*params*发送给攻击者$\mathbf{A}_{\mathrm{II}}$。

（2）询问。攻击者$\mathbf{A}_{\mathrm{II}}$可向挑战者C进行创建用户询问（*CU*）、秘密值询问（*SVK*）、公钥询问（*PK*）、替换用户公钥询问（*PKR*）以及签名询问（*Sign*）。挑战者C需要模拟随机预言器，给出上述方案中各算法的正确回答。在签名询问时，挑战者C仍然要模拟超级签名预言器。

（3）伪造输出。攻击者$\mathbf{A}_{\mathrm{II}}$输出一个伪造（ID*，$\mathrm{pk}_{\mathrm{ID}^*}$，$\tau^*$，$\eta^*$，$\sigma^*$，f*），在以下情况成立时$\mathbf{A}_{\mathrm{II}}$赢得游戏2：①$\sigma^*$是在挑战身份ID*和相应的公钥$\mathrm{pk}_{\mathrm{ID}^*}$下，由$\tau^*$标记的数据集中消息m*的合法签名；②$\tau^* \neq \tau_i$，即$\mathbf{A}_{\mathrm{I}}$没有询问过身份为ID*、公钥$\mathrm{pk}_{\mathrm{ID}^*}$下由$\tau^*$标记的数据集m*上消息的签名；③$\mathbf{A}_{\mathrm{II}}$没有询问过挑战身份ID*对应的秘密值；④$\mathbf{A}_{\mathrm{II}}$没有替换过挑战身份对应的公钥。

将攻击者在上述两个游戏中获胜的概率称为攻击者的优势。无证书同态签名方案在超级攻击者适应性选择身份、适应性选择数据集攻击下是存在性不可伪造（*EUF-CLHS-ID-CDA*）的，任何概率多项式时间内的超级攻击者赢得两个游戏的优势是可以忽略的。

4.2 聚合算法的安全模型

张一名[3]提出一种来源于聚合签名者内部的合谋攻击，其认为聚合签名安全模型中攻击者的能力有限，并在原基础上将攻击者的攻击目标修改为攻击者使用一系列单个签名伪造一个合法的聚合签名，在这些单个签名中包含至少一个非法签名。因此，聚合算法的安全模型可描述为：

（1）系统建立。挑战者获得安全参数后生成系统的各项参数*params*，以及验证聚合签名的公私钥对，然后将公钥*PK*和参数发送给攻击者。

（2）询问阶段（*Query*）。询问阶段包括私钥询问（*SKRQ*）和聚合验证询问（*AVRQ*）两种。私钥询问为当攻击者进行私钥询问时，挑战者根据具体用户的ID运行部分私钥提取算法和秘密值设置算法，将用户ID的私钥返回给攻击者；聚合验证询问为当攻击者进行聚合验证询问时，挑战者通过运行聚合签名验证算法，回答攻击者所给的聚合签名是否合法。

（3）伪造输出。攻击者输出一个伪造（待聚合的消息，最后的聚合签名），如果满足以下条件，则认为攻击者在上述游戏中获胜：①攻击者输出的最后聚合签名是合法的；②在攻击者进行聚合的单个签名序列中，至少有一个签名是非法的。

从上述模型可以看出，攻击者在游戏中获胜实际上是使用一系列不完全合法的单个签名产生了一个合法的聚合签名。一个无证书同态聚合签名方案对于上述攻击者是安全的，在任何概率多项式时间内伪造最多n个用户聚合签名的超级攻击者在上述游戏中获胜的优势是可以忽略的。

5 无证书线性同态聚合签名方案

5.1 无证书同态聚合签名算法描述

（1）Setup：输入安全参数1^k和正整数n，KGC生成素数p阶的乘法循环群G_1、G_T，g为G_1的生成元，双线性映射$\hat{e}$：$G_1 \times G_1 \rightarrow G_T$。KGC选择以下密码学Hash函数：$H$：$\{0,1\}^* \rightarrow Z_p^*$，$H_0$：$\{0,1\}^* \rightarrow Z_p^*$，$H_1$：$\{0,1\}^* \rightarrow G_1$和$H_2$：$\{0,1\}^* \rightarrow G_1$。KGC随机选取$s \in Z_p^*$，设置主私钥为msk=s，$P_{pub}=g^s$。聚合签名的验证者随机选取$\alpha \in Z_p^*$，设置验证者验证私钥为$\alpha$，验证公钥为$Y=g^{\alpha}$。最后KGC公开系统参数，params = $\{G_1, G_T, \hat{e}, p, g, P_{pub}, Y, H, H_0, H_1, H_2\}$。

（2）Partial-Private-Key-Extract：由KGC运行，输入为系统公共参数params、主私钥msk=s和用户身份$ID \in \{0,1\}^*$。该用户的部分私钥计算步骤为：KGC随机选取$r \in Z_p^*$，计算$R=g^r$，$D_{ID}=r+sH_0(ID \| R \| P_{pub})$，通过秘密信道将$D_{ID}$和$R_{ID}$传递给身份为ID的用户。

（3）Produce-Secret-Value：输入用户身份*ID*，随机选择

$x_{ID}\in Z_p^*$,输出用户的秘密值为x_{ID}。

(4)Produce-Public-Key:输入身份ID和秘密值x_{ID},产生用户的公钥$PK=g^{x_{ID}}$。

(5)Sign:输入用户身份ID、部分私钥D_{ID}、秘密值x_{ID},对于l维向量子空间$M\subset Z_p^n$,设$m_1,m_2,\cdots,m_l$为M的基(其中$m=(m_1,m_2,\cdots,m_n)$),该向量子空间的数据集标签$\tau\in 0,1^*$。分别计算$T_i=H_1(ID\|R\|PK\|P_{pub},\tau,i)$,$T_i'=H_2(ID\|R\|PK\|P_{pub},\tau,i)$,然后通过式(1)计算该消息向量$m$的签名,最后输出签名$\sigma$。

$$\sigma=\left(\prod_{i=1}^{n}T_i^{m_i}\right)^{D_{ID}}\left(\prod_{i=1}^{n}T_i'^{m_i}\right)^{x_{ID}} \tag{1}$$

(6)Homomorpic-Combine:输入用户身份ID、公钥PK和l个元组$\left(\beta_i,m_i,\sigma_i\right)_{i=1}^{k}$,输出导出的向量签名对:$\psi=\sum_{i=1}^{k}\beta_i m_i$,$\sigma=\prod_{i=1}^{k}\sigma_i^{\beta_i}$。

(7)Verify:输入ID用户对应的公钥PK,数据集标签τ、R,系数β_i,向量$m_i=\left(m_{i,1},m_{i,2},\cdots,m_{i,n}\right)\in Z_p^n$(其中$i=1,2,\cdots,l$),签名$\sigma$,并计算$h=H_0\left(ID\|R\|P_{pub}\right)$,$T_i=H_1\left(ID\|R\|PK\|P_{pub},\tau,i\right)$,$T_i'=H_2\left(ID\|R\|PK\|P_{pub},\tau,i\right)$,$y=\sum_{i=1}^{l}\beta_i m_i$,$W=R\left(P_{pub}\right)^h$。然后验证式(2)是否成立,如果成立则输出1,否则输出0。

$$\hat{e}(\sigma,g)=\hat{e}\left(\prod\nolimits_{i=1}^{n}T_i^{y_i},W\right)\hat{e}\left(\prod_{i=1}^{n}T_i'^{y_i},PK\right) \tag{2}$$

(8)Aggregate:由签名聚合者执行,设聚合签名用户子集为U∈U ,q=|U|,每个用户在公钥PK_j下对各自数据集标签τ_j标记的l维消息向量子空间基$m^j=\left(m_1^j,m_2^j,\cdots,m_l^j\right)\subset Z_p^n$,对应的签名为$\sigma^j=\left(\sigma_1^j,\sigma_2^j,\cdots,\sigma_l^j\right)$,其中$m_k^j=\left(m_{k,1}^j,m_{k,2}^j,\cdots,m_{k,n}^j\right)\in Z_p^n$且k=1,2,⋯,$l$;j=1,⋯,q。聚合签名验证者的公钥为Y=$g^{\alpha}$,采用式(3)计算聚合签名t。

$$\begin{aligned}t=H(&\hat{e}\left(\sigma_1^1,Y\right),\cdots\hat{e}\left(\sigma_l^1,Y\right),\\&\hat{e}\left(\sigma_1^2,Y\right),\cdots\hat{e}\left(\sigma_l^2,Y\right),\\&\cdots,\\&\hat{e}\left(\sigma_1^q,Y\right),\cdots\hat{e}\left(\sigma_l^q,Y\right))\end{aligned} \tag{3}$$

(9)AggregateVerify:由聚合签名的验证者运行。输入q个用户,在其公钥PK_j下对各自数据集标签τ_j标记的l维消息向量子空间基$m^j=\left(m_1^j,m_2^j,\cdots,m_l^j\right)$,其中$m_k^j=\left(m_{k,1}^j,m_{k,2}^j,\cdots,m_{k,n}^j\right)\in Z_p^n$且k=1,2,⋯,$l$;j=1,⋯,q,对应的聚合签名为t。然后计算$h_j=H_0\left(ID\|R_j\|P_{pub}\right)$,$T_{i,j}=H_1\left(ID\|R_j\|PK_j\|P_{pub},\tau,i\right)$,$T_{i,j}'=H_2\left(ID\|R_j\|PK_j\|P_{pub},\tau,i\right)$,$W_j=R_j\left(P_{pub}\right)^h$。聚合签名的验证者如下进行验证:

$$\begin{aligned}t'=H(&\hat{e}\left(\prod\nolimits_{i=1}^{n}T_i^{m_{k,i}^1},W_1{}^{\alpha}\right)\hat{e}\left(\prod_{i=1}^{n}T_i'^{m_{k,i}^1},(PK_j)^{\alpha}\right),\\&\cdots,\\&\hat{e}\left(\prod\nolimits_{i=1}^{n}T_i^{m_{k,i}^q},W_q{}^{\alpha}\right)\hat{e}\left(\prod_{i=1}^{n}T_i'^{m_{k,i}^q},(PK_j)^{\alpha}\right)\end{aligned} \tag{4}$$

式中,k=1,2,⋯,l。如果$t=t'$,则该算法输出1,否则输出0。

5.2　无证书线性同态聚合签名的安全性

无证书线性同态聚合签名的安全性证明可分为两部分。

(1)本文省略无证书线性同态签名方案在适应性选择身份、适应性选择数据集中含有超级攻击者的证明方法,具体证明过程可参考文献[20]。

(2)聚合算法的安全性:设哈希函数H是抗碰撞的,当且仅当在聚合签名中每个用户的签名是合法的,则由聚合算法产生的聚合签名是合法的。设单个签名元组$\sigma^z=\left(\sigma_1^z,\sigma_2^z,\cdots,\sigma_l^z\right)$为$l$维消息向量子空间基$m^z=\left(m_1^z,m_2^z,\cdots,m_l^z\right)$的合法签名,则有:

$$\hat{e}\left(\sigma_k^z,g\right)=\hat{e}\left(\prod_{i=1}^{n}T_i^{m_{k,i}^z},W_z\right)\hat{e}\left(\prod_{i=1}^{n}T_i'^{m_{k,i}^z},PK_z\right) \tag{5}$$

故有,$\hat{e}\left(\sigma_k^z,Y\right)=\hat{e}\left(\prod\nolimits_{i=1}^{n}T_i^{m_{k,i}^z},W_z{}^{\alpha}\right)\hat{e}\left(\prod_{i=1}^{n}T_i'^{m_{k,i}^z},(PK_z)^{\alpha}\right)$,即$t=t'$。

另一方面,$t=t'$成立,可得:

$$\hat{e}\left(\sigma_k^z,Y\right)=\hat{e}\left(\prod_{i=1}^{n}T_i^{m_{k,i}^z},W_z{}^{\alpha}\right)\hat{e}\left(\prod_{i=1}^{n}T_i'^{m_{k,i}^z},(PK_z)^{\alpha}\right) \tag{6}$$

即$\hat{e}\left(\sigma_k^z,g\right)=\hat{e}\left(\prod\nolimits_{i=1}^{n}T_i^{m_{k,i}^z},W_z\right)\hat{e}\left(\prod_{i=1}^{n}T_i'^{m_{k,i}^z},PK_z\right)$成立。

因此,单个签名元组$\sigma^z=\left(\sigma_1^z,\sigma_2^z,\cdots,\sigma_l^z\right)$为$l$维消息向量子空间基$m^z=\left(m_1^z,m_2^z,\cdots,m_l^z\right)$的合法签名。

6　无证书线性同态聚合签名方案性能实验

在仿真实验过程中使用文献[20]的无证书线性同态签名方案(CLLHS)作为比较对象,对本文提出的无证书同态聚合签名方案的性能进行验证。对5组实验中数据的完整性进行比较,结果如表1、图5所示。

Table 1　Comparison of signature scheme simulation experimental data

表1　签名方案仿真实验数据比较　　单位:s

方案	10Block	20Block	30Block	40Block	50 Block
本文方案	0.56	0.71	0.83	0.94	1.01
CLLHS	2.67	4.89	6.64	9.11	12.2

可以看出,无证书线性同态聚合签名方案在验证大量文件的完整性时所花费的时间远小于CLLHS。当验证数据块达到50个时,其验证速度较CLLHS提高了近10倍。原因在于对N个用户或审计者进行数据完整性公开验证时,如果使用CLLHS,则需要获得所有用户数据块的签名,并对签名逐一验证。随着待验证文件数据块个数的不断增加,用户在验证时的计算代价也会越来越大。而使用无证书线性同态聚合签名方案可极大减少验证时的计算代

价。当用户或审计者想验证N个用户文件时，只需将每个用户数据对应的同态组合签名进行聚合，然后验证聚合签名的合法性即可。通过验证一个同态聚合签名是否有效，便可判定N个用户文件数据块是否保存完好。由此可见，使用无证书线性同态聚合签名方案对大量数据的完整性进行验证时，计算代价不随验证文件数据块数量的增加而显著增大，签名验证效率大大提高。

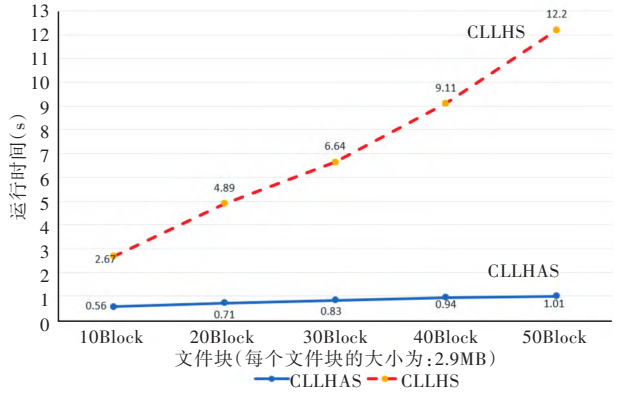

Fig. 5 Comparison between CLLHS and scheme of this paper

图5 CLLHS方案与本文方案性能比较

7 结语

本文提出一种安全高效的无证书线性同态聚合签名方案，并给出了方案的安全证明。结果表明，该方案可以抵抗来自聚合签名者内部的“合谋攻击”。然而，云存储中还存在许多安全技术瓶颈需要解决，如多用户共享数据的权限问题、用户数据审计权限分配问题、各种侧信道攻击、密钥泄露等。这些问题对云存储系统的安全性是一个新挑战，对于密码学也是一个新课题，需要深入细致研究，才能充分保障用户云端数据的安全性。

参考文献：

[1] LI H R, FANG Z C. An architecture and coordination algorithm for improving the comprehensive performance of cloud storage[J]. Microelectronics and Computer, 2020(5): 23-27,32.
李海荣，方中纯. 一种改善云存储综合性能的体系结构及协调算法[J]. 微电子学与计算机, 2020(5):23-27,32.

[2] ZHENG Z X. Discussion on the cloud storage technology based on “big data” background[J]. Computer Products and Circulation, 2020(1): 154.
郑朝霞. 试论基于“大数据”背景下的云存储技术[J]. 计算机产品与流通,2020(1):154.

[3] ZHANG Y M. Research on cloud data secure storage technology under the background of big data[J]. Information Recording Materials, 2019,20(9):195-196.
张一名. 大数据背景下云数据安全存储技术研究[J]. 信息记录材料,2019,20(9):195-196.

[4] BONEH D, LYNN B, SHACHAM H. Short signatures from the Weil pairing[C]//International Conference on the Theory and Application of Cryptology and Information Security, 2001: 514-532.

[5] ULLAH S, DIN N. Blind signcryption scheme based on hyper elliptic curves cryptosystem[J]. Peer-to-Peer Networking and Applications, 2021,14:917-932.

[6] JOHNSON R, MOLNAR D, SONG D, et al. Homomorphic signature schemes[C]//Cryptographers' Track at the RSA Conference, 2002: 244-262.

[7] GIULIA T, DENISE D, JOHANNES B. Homomorphic signature schemes-a survey[R]. Cryptology ePrint Archive Report, 2016:1-73.

[8] WU B, WANG C, YAO H. A certificateless linearly homomorphic signature scheme for network coding and its application in the IoT[J]. Peer-to-Peer Networking and Applications, 2021,14:852-872.

[9] LIN C J, XUE R, YANG S J, et al. Linearly homomorphic signatures from lattices[J]. The Computer Journal, 2020 ,63(12):1871-1885.

[10] BONEH D, GENTRY C, LYNN B, et al. Aggregate and verifiably encrypted signatures frombilinear maps[C]//Proceedings of the 22nd International Conference on Theory and Applications of Cryptographic Techniques, 2003:416-432.

[11] LIU C L, YOU L. Certificateless aggregate signature scheme[J]. Journal of Hangzhou University of Electronic Science and Technology (Natural Science Edition), 2019,39(6): 12-17.
刘纯璐，游林. 无证书的聚合签名方案[J]. 杭州电子科技大学学报(自然科学版),2019,39(6):12-17.

[12] GUO R, CHEN Y S, ZHENG D. Efficient certificateless aggregate signature scheme based on blockchain in wireless medical sensor networks[J]. Information Network Security, 2020,20(10): 6-18.
郭瑞，陈宇霜，郑东. 无线医疗传感网络中基于区块链的高效无证书聚合签名方案[J]. 信息网络安全,2020,20(10):6-18.

[13] DENG L, YANG Y, GAO R. Certificateless designated verifier anonymous aggregate signature scheme for healthcare wireless sensor networks[J]. IEEE Internet of Things Journal, 2021,8(11):8897-8909.

[14] ZHANG F, SHEN L, WU G. Notes on the security of certificateless aggregate signature schemes[J]. Information Sciences, 2014, 287: 32-37.

[15] CAO S Z, LANG X L, LIU X Z, et al. Provably secure and efficient certificateless aggregate signature scheme[J]. Information Network Security, 2019(1): 42-50.
曹素珍，郎晓丽，刘祥震，等. 可证安全的高效无证书聚合签名方案[J]. 信息网络安全,2019(1):42-50.

[16] WU G. Design and analysis of aggregate signature scheme resistant to collusion attack[D]. Nanjing: Nanjing Normal University, 2015.
吴戈. 抗合谋攻击的聚合签名方案设计与分析[D]. 南京:南京师范大学,2015.

[17] AL-RIYAMI S S, PATERSON K G. Certificateless public key cryptography[C]//International Conference on the Theory and Application of Cryptology and Information Security, 2003: 452-473.

[18] WEN Y, YANG Y, WANG S, et al. A new certificateless aggregate signature scheme for wireless sensor networks[J]. Information Sciences, 2021,514:288-301.

[19] CHEN Y C, TSO R, SUSILO W, et al. Certificateless signatures: structural extensions of security models and new provably secure schemes[R]. IACR Cryptology ePrint Archive, 2013:193-202.

[20] MAO L. Certificateless linear homomorphic signature and its application[D]. Nanjing: Nanjing Normal University, 2017.
茅磊. 无证书线性同态签名及其应用[D]. 南京:南京师范大学,2017.

(责任编辑:尹晨茹)