云存储中基于虚拟用户的数据完整性验证

徐云云 白光伟 沈 航 黄中平
(南京工业大学计算机科学与技术学院 南京 211800)

摘 要 针对验证数据完整性过程中被撤销用户与云服务器存在共谋的问题,提出基于虚拟用户的数据完整性校验方案。在管理群组用户的过程中,管理员让云服务器作为代理,通过重签名方法将被撤销用户的签名转换为虚拟用户签名,以防止攻击者获取群组用户身份隐私信息。另一方面,管理员在本地存储所有用户的身份隐私信息,用户在访问共享数据之前需要通过管理员的验证,这样既能保证校验者可以正确验证共享数据的完整性,又能保护群组用户的隐私和共享数据的安全。分析证明结果表明,所提方案在用户撤销时不仅能够验证共享数据的完整性,还能降低攻击者精确获取用户身份隐私信息和共享数据内容的概率。

关键词 云存储,隐私保护,代理重签名,用户撤销,虚拟用户
中图法分类号 TP393 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2017.05.017

Virtual-user-based Public Auditing Integrity in Cloud Storage

XU Yun-yun BAI Guang-wei SHEN Hang HUANG Zhong-ping
(College of Computer Science and Technology, Nanjing University of Technology, Nanjing 211800, China)

Abstract A public auditing integrity mechanism based on the virtual user was proposed, addressing collusion issue between the revoked user and cloud. After the user is revoked from the group, the manager lets the proxy resign the blocks with the virtual user's signature, which utilizes proxy re-signature to protect user's privacy. In addition, the manager verifies the user who want to access the shared data via a local list consisting of all users' identity, with objective of auditing data integrity and protecting user's privacy. Theoretical analysis shows that our framework achieves significant performance improvement in security and privacy, and it can decrease the probability of an adversary to get the users' identity privacy and the data in the cloud.

Keywords Cloud storage, Preserve privacy, Proxy re-signature, User revocation, Virtual user

1 引言

云计算是继分布式计算、网格计算、对等计算之后的一种新型计算模式,它以资源租用、应用托管、服务外包为核心,迅速成为计算机技术发展的热点[1]。网络带宽的增加以及移动互联网的普及,为广大云用户提供了廉价的存储空间,使得云存储已经成为云计算中最广泛的应用之一[2]。然而受软硬件或者人为等不确定因素的影响[3-4],存储在云端的数据不可避免地会被服务器错误地修改或者删除,而且用户没有本地存储,所以无法得知数据的完整性和正确性。因此,安全问题成为进一步推广云存储服务的主要障碍之一,涉及的问题主要包括共享数据的完整性、机密性等。

近期,为了验证云存储中共享数据的完整性,许多方案都被提出[4-10]。文献[4]首先提出公开验证的概念以及可证明数据持有(Provable Data Possession),使用户在不需要下载全部数据的情况下验证数据的完整性。文献[6]提出一种校验机制,可以保护数据隐私,使校验者在验证数据完整性的过程中无法获取共享数据的内容。目前大多数的云存储数据完整性的验证方案[5-9]主要用于检验个人数据的完整性,很少考虑群组共享数据的完整性。若群组中有用户离开,则从群组中撤销该用户。对于整个群组来说,该用户所生成的签名也将失效。文献[10]利用代理重签名[11-12]来完成由被撤销用户签名数据块的重签工作。这样不仅可以在用户撤销过程中有效验证共享数据的完整性,同时也可以避免暴露合法用户的私钥。

在前期的研究过程中[5-9,13-15]发现,若被撤销用户与云存储服务器之间存在共谋,将会对用户的身份隐私以及共享数据的安全造成较大威胁。如图1所示,在撤销B用户的情况

到稿日期:2016-04-15 返修日期:2016-07-18 本文受国家自然科学基金项目(61502230,61073197),江苏省自然科学基金项目(BK20150960),江苏省普通高校自然科学研究项目(15KJB520015),中美计算机科学研究中心开放课题(KJR16078),江苏省六大高峰人才基金资助项目(第八批),2015年度普通高校研究生科研创新计划(KYLX15_0804)资助。

徐云云(1990-),女,硕士生,主要研究方向为云计算安全,E-mail:xuyun1234.happy@163.com;白光伟(1961-),男,博士,教授,博士生导师,CCF高级会员,主要研究方向为移动互联网、无线传感器网络、网络体系结构和协议、网络系统性能分析和评价、多媒体网络服务质量等;沈 航(1984-),男,博士生,CCF学生会员,主要研究方向为无线网络编码、移动互联网、无线多媒体通信协议等;黄中平(1993-),男,硕士生,主要研究方向为移动云计算、分布式视频转码等。

下，若云服务器和被撤销用户 B 存在共谋，那么在代理重签名中，转换为新签名的合法用户 A 的私钥将会被暴露，这使得 A 用户的身份隐私信息以及存储在云服务器上的共享数据内容的安全也将受到威胁。

文献[16]提出一种安全的用户撤销机制，使校验者在用户撤销的情况下仍然能正确验证共享数据的完整性，忽略了用户的身份隐私保护。文献[17]设计了一种基于多项式验证标签以及代理标签更新的多用户撤销机制。在多个用户撤销的情况下，验证者仍然能够验证共享数据的完整性，同样，该机制也忽略了用户的身份隐私保护。通过上述分析可知，被撤销用户与云端服务器之间存在共谋时会威胁合法用户的身份隐私以及共享数据的安全。

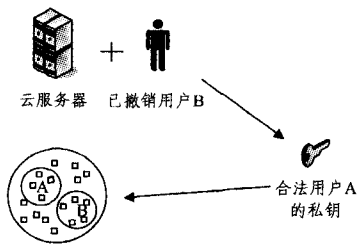

图 1 被撤销用户与云服务器存在共谋

针对上述问题，本文提出基于虚拟用户的共享数据完整性验证方案，目的是解决在被撤销用户与云服务器之间存在共谋时，合法用户私钥暴露所带来的身份隐私泄露以及共享数据安全问题。当有用户撤销时，代理对需要重新签名的数据块进行重新计算，将它转化为虚拟用户生成的签名。这样若云服务器与被撤销用户之间存在共谋，则攻击者即使得到虚拟用户私钥，也无法获取群组中任一合法用户的身份隐私信息，同时也无法得知共享数据的内容，达到保护用户身份隐私以及共享数据安全的目的。与现在的研究工作相比，本文既考虑了云存储中共享数据完整性验证过程中存在的用户撤销以及被撤销用户与云服务器共谋的情况，还引入了虚拟用户这一概念，保护了用户的身份隐私信息。

本文第 2 节介绍相关的数学知识；第 3 节引入系统模型和设计目标；在此基础上，第 4 节介绍基于虚拟用户的公开校验方案以及算法；第 5 节对本方案进行安全性分析与评价；最后总结全文。

2 预备知识

本文为了完成在用户撤销的情况下基于虚拟用户的共享数据完整性校验，需要用到密码学理论的基础知识，现在对相关机制进行相应阐述，具体如下：

(1)双线性对映射。设 p 是一个大素数；$\mathbf{Z}_p$ 表示模 p 的同余类集合，G_1 和 G_2 是 2 个有着相同阶数 p 的乘法循环群，G_1 的生成元为 g。e 是一个双线性对映射，即 $G_1\times G_1\rightarrow G_2$。

1)双线性。对于所有的 $u,v\in G_1$ 以及所有的 $a,b\in\mathbf{Z}_p$，$e(u^a,v^b)=e(u,v)^{ab}$ 均成立。

2)可计算性。对于所有的 $u,v\in G_1$，都存在一个有效的算法计算出映射 $e(u,v)$。

3)非退化性。存在 $a,b\in\mathbf{Z}_p$，使得 $e(g^a,g^b)\neq 1$。

(2)安全性假设。本文涉及到以下安全性假设。

定义 1(计算 Diffie-Hellman 假设) 给定任意随机元素 a，$b\in\mathbf{Z}_p^*$，已知 $g,g^a,g^b\in G_1$，求解 $g^{ab}\in G_1$。

对于一个多项式时间的攻击者 A_{CDH}，其在 G_1 上求解 CDH 问题的优势是可以忽略的，具体如下：

$$\Pr[A_{CDH}(g,g^a,g^b)=(g^{ab}):a,b\xleftarrow{R}\mathbf{Z}_p^*]\leqslant\varepsilon$$

其中，R 表示 a，b 为 $\mathbf{Z}_p^*$ 中的随机数，ε 表示一个可以忽略的优势。

定义 2(离散对数假设，Discrete Logarithm Assumption) 给定任意元素 $a\in\mathbf{Z}_p^*$，已知 g，$g^a\in G_1$，求解 a。

对于任何一个多项式时间的攻击者 A_{DL}，其求解离散对数问题的优势可以忽略。

$$\Pr[A_{DL}(g,g^a)=(a):a\xleftarrow{R}\mathbf{Z}_p^*]\leqslant\varepsilon$$

其中，R 表示 a,b 为 $\mathbf{Z}_p^*$ 中的随机数，ε 表示一个可以忽略的优势。

(3)同态标签。同态标签指，允许一个公开校验者在不下载所有数据的情况下，验证全部数据的完整性。该工具作为云端数据完整性验证的必要工具，被广泛应用于多个方案。一个同态标签具有以下性质。

令 (pk,sk) 为签名者的密钥对，pk 表示公钥，sk 表示私钥，σ_k 表示数据块(block) m_k 的签名，$k\in[1,n]$。

1)数据块简化验证(Blockless Verifiability)：已知 σ_1 和 σ_2，两个随机数 $\alpha_1,\alpha_2\in\mathbf{Z}_p$ 和一个数据块 $m'=\alpha_1 m_1+\alpha_2 m_2\in\mathbf{Z}_p$，一个校验者可以在无需知道 m_1 和 m_2 的情况下，验证数据块 m' 的正确性。

2)非扩展性(Non-malleability)：已知 m_1,m_2,σ_1 以及 σ_2，两个随机数 $\alpha_1,\alpha_2\in\mathbf{Z}_p$，一个用户若没有私钥 sk，则无法通过线性结合 σ_1 和 σ_2 的方式来产生一个合法的、基于数据块 m' 的标签。

数据块简化验证实质是在拥有部分数据的前提下，一个校验者与云服务器之间通过挑战与回应(challenge-and-response)的协议得到的一个基于所有数据块的线性组合数据块来判断共享数据的完整性。非扩展性实质上是指任何攻击者在没有私钥的情况下，无法通过线性组合已有签名的方式来获得新的有效标签。

(4)代理重签名。代理重签名算法是由 Blaze 等人[11]首先提出的，该签名允许一个半可信代理来转换不同用户之间的签名，例如 A 和 B 两个用户，该代理可以将同一个数据块上 A 的签名转化为 B 的签名。在转换过程中，代理无法获得任何用户的私钥，即代理无法生成任何一个数据块的有效签名。

3 系统模型及设计目标

3.1 系统模型

如图 2 所示，系统由 5 个部分组成：用户、虚拟用户、管理员、云服务器以及公开校验者。本文中的用户指一个群组用户$\{u_1,\cdots,u_d\}$，群组中任何一个用户在通过管理员的验证后都有权限访问或者修改共享数据，并为之生成相应的签名。虚拟用户指存在用户撤销的情况时，将被撤销用户所签名的数据块上的签名转化为虚拟用户所生成的签名(虚拟用户的信息只有在用户撤销的情况下，公开校验者验证共享数据完整性时才有效)。管理员存储所有用户的身份隐私信息和共享数据分块后的数目信息(如数据块标识符、数据块数量)。云服务器利用自身具有较强的存储能力和较高的资源利用率，为用户提供海量数据存储以及计算等业务。公开校验者指代替用户为共享数据提供完整性验证功能的第三方，通过

与云服务器之间的一个挑战与回应协议来完成共享数据的完整性校验。

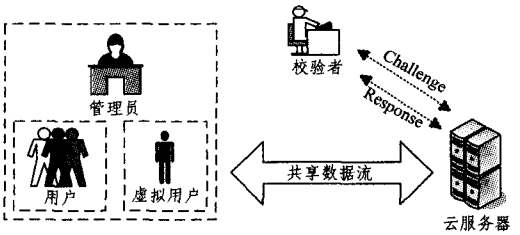

图 2　系统模型

在本文所提方案中，云服务器是半可信的，即它不会恶意攻击共享数据，但考虑到自身利益，可能会对用户隐瞒共享数据的不正确性。另外，假如被撤销用户与云服务器之间存在共谋，那么群组用户的身份隐私信息以及共享数据内容的安全都将会受到威胁。本文引入虚拟用户这一概念，在有用户撤销的情况下，管理员让云服务器作为代理，将被撤销用户所签名的数据块签名转化为虚拟用户生成的签名。这样即使被撤销用户与云服务器之间存在共谋，攻击者获取虚拟用户的私钥，也无法通过管理员的验证，从而有效保护群组用户的身份隐私信息以及共享数据的内容。

3.2　设计目标

本方案需要实现以下目标：

(1)正确性。公开校验者可正确验证共享数据的完整性。

(2)有效且安全的用户撤销。当一个用户从群组中被撤销之后，该用户之前计算所生成的签名可以被其他合法用户重新计算；另一方面，只有合法用户才能为共享数据生成签名，已被撤销用户无法再为共享数据生成新的签名。

(3)公开可验证性。在没有与群组用户共享密钥的前提下，公开校验者不用加载全部的共享数据，即可验证它的完整性，且云服务器无法为不正确的数据伪造完整性证明。

(4)身份隐私保护。校验者或攻击者在验证数据完整性的过程中无法获取整个群组用户的身份隐私信息。

4　基于虚拟用户的数据完整性验证

本节介绍基于虚拟用户的数据完整性验证方案。为了解决被撤销用户与云服务器之间存在共谋所带来的合法用户私钥泄露问题，引入了虚拟用户这一概念来解决用户身份隐私泄露对群组用户的威胁，同时保护共享数据的安全。

4.1　基本思路

数据完整性验证协议[18]通常包括初始化和验证两个阶段。

(1)初始化阶段。用户首先对共享数据 F 进行分块，$F=\{m_1,\cdots,m_k\}$，然后将其存储在云服务器上。用户生成自己的密钥对(pk,sk)，利用同态签名机制给每个数据块生成相应的签名 σ_i，组成验证元数据集合 $\Phi=\{\sigma_1,\cdots,\sigma_k\}$，并且将共享数据 F 及 Φ 一起存放在云服务器上。另外，用户需要将公钥以及数据块数目等信息发送给校验者。

(2)验证阶段。校验者首先向云服务器发送挑战协议 $chal$；云服务器接收协议后计算生成相应的证据 P，并发送给校验者；最后校验者只需验证证据 P 的正确性来判断云服务器中的共享数据是否完整。具体来说，校验者从分块索引[1, k]中随机抽取 c 个块索引作为挑战索引 L，并选取随机数 $t_i\in\mathbf{Z}_p$，然后将$\{l,t_l\}$作为挑战请求发送给云服务器。云服务器在接收到 $chal$ 后，将 L 分成 d 个部分，即 $L=\{L_1,\cdots,L_d\}$，并计算相应的证据 $P=\{\mu,\beta\}$，其中 μ 表示数据块值的线性组合，β 表示数据块标签的线性组合。

$$\mu_i=\sum_{l\in L_i}t_l m_l\in\mathbf{Z}_p; \tag{1}$$

$$\beta_i=\prod_{l\in L_i}\sigma_l^{t_l}\in\mathbf{G}_1 \tag{2}$$

然后云服务器将证据 P 发送给校验者，由校验者来验证共享数据的完整性。

为了防止被撤销用户与云服务器之间存在共谋，攻击者获取用户私钥对的共享数据进行恶意破坏，管理员在本地创建一个 user-key-list (UKL)来存储群组用户以及虚拟用户的身份隐私信息(见表 1)，用户需要通过管理员的验证才能够访问或者修改共享数据。当用户需要访问或者修改共享数据时，管理员验证其是否在 UKL 中，若在，则返回相应的密钥；否则，将忽略该用户的请求。当有用户撤销时，管理员需要将被撤销用户的身份信息从 UKL 中删除，同时增加一个虚拟用户的身份信息。

表 1　一个 user-key-list(UKL)

Num	User	Key
1	u_1	sk_1
2	u_2	sk_2
…	…	…
d	u_d	sk_d
$d+1$	u_v	sk_v
…	…	…

在验证共享数据的完整性时，先计算用户的签名 σ_i 以及代理重签密钥 $rk_{i\to j}$，生成代理重签名 σ_i'。本方案中所用到的符号以及相关参数如表 2 所列。

表 2　符号以及相关参数

符号	说明
$\mathbf{G}_i$	2 个乘法循环组：$\mathbf{G}_1,\mathbf{G}_2$ ($g_i,w\in\mathbf{G}_i$)
e	$\mathbf{G}_1\times\mathbf{G}_1\to\mathbf{G}_2$；$e(g_1^a,g_2^b)=e(g_1,g_2)^{a*b}$
a_i,b_i,v_i,r,t	随机参数($a_i,v_i,r\in\mathbf{Z}_p^*$)
m_i	待签名数据块
id	数据块标示
sk_i	用户数字签名私钥 $sk_i=a_i$
pk_i	用户数字签名公钥 $pk_i=g^{*}i$
$rk_{i\to j}$	代理重签密钥 $rk_{i\to j}=sk_j/sk_i$
σ_i	各个数据块的部分签名
σ_i'	各个数据块的代理重签名
L	数据块索引值

用户 u_i 生成自己的公私密钥对(pk_i,sk_i)，并为数据块 m_k 计算签名 σ_k。

$$\sigma_k=(H(id_k)w^{m_k})^{a_i} \tag{3}$$

当有用户被撤销之后，云服务器、合法用户以及被撤销用户将共同参与生成重签密钥 $rk_{i\to j}$，本方案选用云服务器作为代理，对需要重签的数据块计算新的签名 σ_k'。

$$\sigma_k'=\sigma^{*}i\to j=(H(id_k),w^{m_k})^{a_i*a_j/a_i}=(H(id_k)w^{m_k})^{a_j} \tag{4}$$

云服务器在接收到校验者发送的挑战 $chal$ 后，计算生成相应的完整性证据 P，并发送给校验者。

$$e(\prod_{i=1}^{d}\beta_i,g)=\prod_{i=1}^{d}e(\prod_{l\in L_i}H(id_l)^{t_l}w^{\mu_i},pk_i) \tag{5}$$

校验者通过式(5)来验证共享数据的完整性。若校验者的返回值为 1，则表明数据完整；若返回值为 0，则表明数据不完整。

4.2 代理重签名

因为传统的代理重签名方案未考虑云服务器与被撤销用户之间存在共谋的情况，所以不可避免地会对群组中合法用户的身份隐私信息以及共享数据的内容安全造成威胁。暴露出以上隐私安全问题的主要原因在于被撤销用户的私钥参与了代理重签密钥的生成，并且当用户被撤销后，撤销用户与云服务器之间存在共谋。

为了解决上述隐私安全问题，本文在用户中加入了虚拟用户这一概念，当有用户被撤销之后，该用户之前所生成的签名将失效，这时通过代理重签名将被撤销用户的签名转换成虚拟用户的签名。基于虚拟用户的共享数据的完整性验证方案主要由 6 个部分组成，包括 *Keygen*，*Rekey*，*Sign*，*Resign*，*Genproof* 和 *Verify*，具体实现如下。

(1)$Keygen(1^n)$：管理员为用户 u_i 随机生成 $a_i\in\mathbb{Z}_p^*$ 作为输入，生成公私密钥对(pk_i,sk_i)，其中 $pk_i=g^{a_i}$，$sk_i=a_i$，并将其存储在 UKL 中。

(2)$Rekey(1^n,sk_i,sk_j)$：1)云服务器产生一个随机数 $r\in\mathbb{Z}_p$，并发送给用户 u_i；2)在经过管理员验证后，u_i 将 r/a_i 发送给用户 u_j，$sk_i=a_i$；3)用户 u_j 再将 ra_j/a_i 发送给云服务器，$sk_j=a_j$；4)云服务器得到重签密钥 $rk_{i\to j}=a_j/a_i$。

(3)$Sign(sk,id_k,F)$：以私钥 $sk_i=a_i$、数据文件 $F=\{m_1,\cdots,m_k\}$ 以及数据块标识符 id_k 作为输入，用户 u_i 根据式(3)可计算出数据块 m_k 的签名 σ_k，即 $\Phi=\{\sigma_1,\cdots,\sigma_k\}$。

(4)$Resign(rk_{i\to j},pk_i,\sigma_k)$：以重签密钥 $rk_{i\to j}$、用户 u_i 的公钥 pk_i 以及数据块标识符 id_k 作为输入，根据式(4)对数据块进行重签，即 $\Phi'=\{\sigma_1',\cdots,\sigma_k'\}$。

(5)$Genproof(F,\Phi',chal)$：云服务器在接收校验者发送的挑战请求 $chal=\{l,t_l\}_{l\in L}$ 后，根据共享数据信息 F 以及重签后验证元集合 Φ' 计算出证据 P，并发送给校验者。

(6)$Verify(chal,P,pk_i)$：根据自己发出的完整性挑战 $chal$、接收到的 P 以及用户的公钥$(pk_1,\cdots,pk_d)$，校验者利用式(5)验证共享数据的完整性。

如果被撤销用户与云服务器之间存在共谋，虚拟用户的隐私(这里特别指用户的私钥)将会暴露，这时若攻击者想要访问共享数据，管理员查找 UKL 后未发现该用户的身份信息，则忽略其访问请求，保证了共享数据的安全。

4.3 批验证支持

在很多情况下，校验者需要在短时间内完成多个验证任务。如果校验者逐一验证，将会消耗大量的资源。为了提高本方案在该情况下的扩展性，基于双线性对映射的性质对本方案进行扩展，使其支持批验证，提高校验者的验证效率。假定校验者需验证来自 N 个不同群组的文件 F_n，每个群组的大小为 $d_n(1\leqslant n\leqslant N)$。

(1)根据算法 *Keygen*，群组中每个用户生成相应密钥对$(pk_{n|1},sk_{n|d_n})$；

(2)对每个文件 F_n 进行分块，表示为 $F_n=\{m_{n1},\cdots,m_{nk}\}$；

(3)根据算法 *Sign* 和 *Resign* 为每个数据文件生成验证元数据集合 $\Phi_n=\{\sigma_{n1},\cdots,\sigma_{nk}\}$；

(4)每个 *chal* 可以表示为$\{l,t_{n|l}\}_{l\in L_n}$；

然后利用式(6)完成对 N 个不同群组文件 F_n 的完整性验证：

$$e(\prod_{n=1}^{N}\prod_{i=1}^{d_n}\beta_{n|i}^{b_n},g)=\prod_{n=1}^{N}\prod_{i=1}^{d_n}e(\prod_{l\in L_{n|i}}H(id_{n|l})^{t_{n|l}}w^{\mu_{n|i}},pk_{n|i})^{b_n} \tag{6}$$

若满足式(6)，则表明来自 N 个群组的共享文件 F_n 都是正确且完整的；否则，表明来自 N 个群组的共享文件 F_n 中至少有一个文件是不正确的。

5 安全性分析与评价

根据典型云存储中数据完整性验证的相关参数，本节将从 4 个方面对本方案进行安全性分析，包括正确性、有效且安全的用户撤销、公开验证以及身份隐私保护。

定理 1 公开校验者能够正确验证基于虚拟用户签名的共享数据的完整性。

证明：根据本文代理重签名的生成过程可知，要证明该论点，即证明在已知所有用户(包括群组用户和虚拟用户)公钥 pk_i、共享数据 $F=\{m_1,\cdots,m_k\}$、验证元数据集合 $\Phi=\{\sigma_1,\cdots,\sigma_k\}$以及聚合签名 β 的情况下，当用户 u_j 被撤销之后，校验者验证存储在云服务器上的共享数据是否完整，根据式(5)，即需要验证 $e(\prod_{i=1}^{d}\beta_i,g)$ 与 $\prod_{i=1}^{d}e(\prod_{l\in L_i}H(id_l)^{t_l}w^{u_i},pk_i)$ 是否相等。若结论成立则能证明基于虚拟用户的共享数据完整。证明过程如下：

$$\begin{aligned}
&e(\prod_{i=1}^{d}\beta_i,g)\\
&=\prod_{i=1}^{d}e(\prod_{l\in L_i}\sigma_l^{t_l},g)\\
&=[\prod_{i\neq j}e(\prod_{l\in L_i}\sigma_l^{t_l},g)]\cdot e(\prod_{l\in L_i}\sigma_l^{t_l},g)\\
&=[\prod_{i\neq j}e(\prod_{l\in L_i}(H(id_l)w^{m_l})^{a_it_l},g)]\cdot[e(\prod_{l\in L_j}H(id_l)w^{m_l})^{a_jt_l},g]\\
&=\prod_{i=1}^{d}e(\prod_{l\in L_i}(H(id_l)w^{m_l})^{a_it_l\cdot a_jt_l},g)\\
&=\prod_{i=1}^{d}e(\prod_{l\in L_i}(H(id_l)^{t_l}\cdot w^{m_lt_l}),g^{a_ia_j})\\
&=\prod_{i=1}^{d}e(\prod_{l\in L_i}(H(id_l)^{t_l}\cdot w^{\mu_i}),pk_i)
\end{aligned}$$

由上述证明过程可得：当用户被撤销后，为了使校验者能够正确验证共享数据的完整性，云服务器将被撤销用户生成的数据块上的签名转换为虚拟用户的签名后，第三方校验者仍可以正确地验证共享数据的完整性。

定理 2 虚拟用户不会威胁群组用户以及云服务隐私。

证明：本文在群组用户端加入虚拟用户这一概念，在被撤销用户与云服务器之间存在共谋的情况下，群组中的合法用户能够很好地对云服务器端保密自己的身份隐私信息。当用户因为自己不合法的行为而被撤销之后，该用户所生成的数据块上的签名将不再有效。为了保证校验者能够正确地校验数据的完整性，云服务器端将利用代理重签名将被撤销用户所生成的数据块的签名转换为虚拟用户所生成的签名，这样若被撤销用户与云服务器端存在共谋，则会对群组用户的身份隐私信息以及数据块内容的安全造成威胁。攻击者即使获得虚拟用户的密钥，也无法通过管理员的身份验证，无法访问或者修改共享数据，从而保护了群组用户的身份隐私安全。

定理 3 如果离散对数假设成立，则云服务器无法为不

正确的共享数据伪造证明 P。

证明：根据文献[5]中所定义的安全模型可以证明，假设云服务器能够为不完整的数据伪造一个证据 P'，并能通过校验者的验证，则说明在 G_1 中的 DL 问题是可以解决的。

假设云存储中的共享数据为 F，云服务器在接收校验者发送的挑战请求 $\{l,t_l\}$ 后，为共享数据 F 生成相应的证明 $P=\{\mu,\beta\}$，并能通过校验者的验证。若云服务器为不完整的共享数据 F' 伪造证明 $P'=\{\mu',\beta\}$，其中 $\mu'=\sum_{l\in L_i} t_l m_l'$。设定 $\Delta\mu=\mu-\mu'$ 且 $\Delta\mu\neq 0$。

本文利用归谬法，假设 P' 能够通过校验者的验证，根据式(5)可得：

$$e(\prod_{i=1}^{d}\beta_i,g)=\prod_{i=1}^{d}e(\prod_{l\in L_i}H(id_l)^{t_l}w^{u_i'},pk_i)。$$

根据正确的共享数据的完整性证明 $P=\{\mu,\beta\}$，可得：

$$e(\prod_{i=1}^{d}\beta_i,g)=\prod_{i=1}^{d}e(\prod_{l\in L_i}H(id_l)^{t_l}w^{u_i},pk_i)。$$

根据双线性对的性质，由上述等式可知 $w^{u_i}=w^{u_i'}$，$w^{\Delta\mu}=1$。

假设 $g,h\in G_1$，且存在 $x\in Z_p$ 使得 $h=g^x$。由于 $w\in G_1$，因此 $w=g^{r_1}h^{r_2}$，其中 r_1 和 r_2 为 Z_p 中的随机数，$w^{\Delta\mu}=(g^{r_1}h^{r_2})^{\Delta\mu}=g^{r_1\Delta\mu}\cdot h^{r_2\Delta\mu}=1$。那么

$$h=g^{-\frac{r_1\Delta\mu}{r_2\Delta\mu}},x=-\frac{r_1\Delta\mu}{r_2\Delta\mu}$$

分母 $r_2\Delta\mu$ 不为 0。由于 r_2 为 Z_p 中的随机数，则 r_2 不为 0 的概率为 $1/p$，因此解决 DL 假设问题的概率为 $1-1/p$。考虑到 p 为一个大素数，r_2 不为 0 的概率可以忽略，那么解决 DL 假设问题的概率为 $1-1/p$，这与 DL 假设相矛盾。

定理 4　攻击者即使获得密钥，也无法获取合法用户的身份隐私信息。

证明：假设群组用户的用户数为 d，用户私钥为 $\{sk_1,sk_2,\cdots,sk_d\}$，当有用户被撤销时，被撤销用户被虚拟用户代替，此时管理员更新 UKL，即当前整个群组的私钥为 $\{sk_1,sk_2,\cdots,sk_{j-1},sk_v,sk_{j+1},\cdots,sk_d\}$。即使虚拟用户私钥 sk_v 暴露，攻击者精确获取群组用户的身份隐私信息的概率为 $1/(d-1+n)$，远小于转换为群组中原有用户的概率 $1/(d-1)$。

结束语　针对用户撤销过程中共享数据完整性验证问题，提出基于虚拟用户的数据完整性验证方案。当用户从群组中被撤销之后，被撤销用户所生成的签名也将失效，代理需要将被撤销用户所生成的数据块上的签名转换为虚拟用户生成的签名，这样攻击者就无法获取任一群组用户的身份隐私信息。同时，为了防止攻击者获得共享数据的内容，管理员利用本地存储的 UKL 来判断当前用户是否有权限访问共享数据。即使攻击者获得虚拟用户的私钥，也无法通过管理员的验证，从而保护共享数据内容的安全。综上，考虑到被撤销用户与云服务器之间存在共谋的情况，本文为了保护群组用户的身份隐私信息以及共享数据内容的安全，使攻击者无法精确获得用户的身份信息以及共享数据的内容，提出新的方案来验证共享数据的完整性。

参 考 文 献

[1] LING C, SU W B, MENG K, et al. Cloud computing security: Architecture, Mechanism and Modeling[J]. Journal of computer, 2013, 36(9): 1765-1784. (in Chinese)
林闯，苏文博，孟坤，等. 云计算安全：架构，机制与模型评价[J]. 计算机学报，2013，36(9)：1765-1784.

[2] LI H, SUN W H, LI F H, et al. Secure and Privacy-Preserving Data Storage Service in Public Cloud[J]. Journal of Computer Research and Development, 2014, 51(7): 1397-1409. (in Chinese)
李晖，孙文海，李凤华，等. 公共云存储服务数据安全及隐私保护技术综述[J]. 计算机研究与发展，2014，51(7)：1397-1409.

[3] ARMBRUST M, FOX A, GRIFFITH R A, et al. A View of Cloud Computing[J]. Communications of the ACM, 2010, 53(4): 50-58.

[4] ATENIESE G, BURNS R, CURTMOLA R, et al. Provable Data Possession at Untrusted Stores [C]// The Proceedings of ACM CCS 2007. 2007: 598-610.

[5] SHACHAM H, WATERS B. Compact Proofs of Retrievability [C]// Proceedings of ASIAC RYPT 2008, LNCS. Springer, Heidelberg, 2008, 5350: 90-107.

[6] WANG C, WANG Q, REN K, et al. Privacy-preserving Public Auditing for Data Storage Security in Cloud Computing [C]// Proceedings of IEEE INFOCOM. 2010: 525-533.

[7] ZHU Y, WANG H, HU Z, et al. Dynamic Audit Services for Integrity Verification of Outsourced Storage in Clouds [C]// the Proceedings of ACM SAC 2011. 2011: 1550-1557.

[8] WANG C, WANG Q, REN K, et al. Towards Secure and Dependable Storage Services in Cloud Computing [J]. IEEE Transactions on Services Computing, 2011, 5(2): 220-232.

[9] CAO N, YU S, YANG Z, et al. LT Codes-based Secure and Reliable Cloud Storage Service [C]// The Proceedings of IEEE INFOCOM 2012. 2012: 693-701.

[10] WANG B, LI B, LI H. Panda: Public Auditing for Shared Data with Efficient User Revocation in the Cloud [J]. IEEE Transactions on Services Computing, 2015, 8(1): 92-106.

[11] BLAZE M, BLEUMER G, STRAUSS M. Divertible protocols and atomic proxy cryptography [C]// Advance in Cryptology-EUROCRYPT 98. 1998: 127-144.

[12] ATENIESE G, HOHENBERGER S. Proxy Re-signatures: New Definitions, Algorithms and Applications [C]// Proceeding of ACM CCS 2005. 2005: 310-319.

[13] WANG C, CHOW S S M, WANG Q, et al. Privacy-preserving public auditing for secure cloud storage [J]. IEEE Transactions on Computers, 2013, 62(2): 362-375.

[14] ZHANG J, TANG W, MAO J. Efficient public verification proof of retrievability scheme in cloud [J]. Cluster Computing, 2014, 17(17): 1401-1411.

[15] WANG B, LI B, LI H. Oruta: Privacy-preserving public auditing for shared data in the cloud [C]// Proc. of IEEE 5th International Conference on Cloud Computing (CLOUD). 2012: 295-302.

[16] YU Y, NI J, AU M, et al. On the security of a public auditing mechanism for shared cloud data service[J]. IEEE Transaction Services Computing, 1939, 8(6): 1.

[17] YUAN J, YU S. Public integrity auditing for dynamic data sharing with multiuser modification [J]. IEEE Trans. Inf. Forensics and Secur, 2015, 10(8): 1717-1726.

[18] BONEH D, GENTRY C, LYNN B, et al. Aggregate and Verifiably Encrypted Signatures from Bilinear Maps [M]// The Proceedings of EUROCRYPT 2003. Springer-Verlag, 2003: 416-432.