中国科学: 信息科学 2011 年 第 41 卷 第 10 期: 1227–1237
www.scichina.com info.scichina.com
《中国科学》杂志社
SCIENCE CHINA PRESS

论 文

数据可恢复性的零知识证明

朱岩[①②*], 王怀习[③], 胡泽行[①], Gail-Joon AHN[④], 胡宏新[④*]

① 北京大学计算机科学技术研究所, 北京 100871
② 北京大学互联网安全技术北京市重点实验室, 北京 100871
③ 北京大学数学科学学院, 北京 100871
④ School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, Tempe, AZ 85287, USA
* 通信作者. E-mail: yan.zhu@pku.edu.cn, hxhu@asu.edu

收稿日期: 2010–04–28; 接受日期: 2010–12–14
国家自然科学基金 (批准号: 61003216) 和美国国家自然科学基金 (批准号: NSF-IIS-0900970, NSF-CNS-0831360) 资助项目

摘要 可恢复性证明是一种外包存储服务中的数据完整性检测技术. 本文致力于交互式证明系统标准模型下的可恢复性证明协议构造方法研究, 提出了首个能够防止证明者欺骗和验证数据泄露的交互式可恢复性证明协议, 并通过构造多项式时间的可回卷知识抽取器, 给出了基于计算 Diffie-Hellman 假设下的协议完备性、稳固性, 以及零知识性证明. 特别是, 所提方案的验证过程只要求低的、固定大小的负载, 达到了最小化通信复杂性的目的.

关键词 密码学 存储安全 可恢复性证明 交互协议 零知识

1 引言

可恢复性证明 (proof of retrievability, POR)[1] 是一种存储提供者向数据使用者证明所存储数据仍保持完整性的密码证明技术, 从而保证使用者能够完全地恢复出这些被存储的数据, 并可安全地使用这些数据. 与通常的完整性认证技术不同, 可恢复性证明技术能够在不下载数据的情况下对数据是否已被篡改或删除进行检验, 这一特点对于大数据量的外包数据和文档存储是极其重要的. 例如, 随着云计算的广泛应用, 云存储服务通过提供相对低廉的、可扩展的、地域无关的数据管理平台, 正逐渐变成了一个新的信息技术利润增长点; 但是, 由于这种服务将用户的数据和文档存储于企业以外的不确定存储池中, 如果这样一个重要的服务易于受到各种恶意攻击的危害, 那么它也会给客户带来不可挽回的损失. 然而, POR 技术所提供认证能力为解决这一外包存储问题提供了技术支持, 因此, 云服务提供商使用 POR 技术实现安全数据管理是完全必要的.

Juels 和 Kaliski[1] 首先提出了可恢复性证明的形式化模型, 此后在近几年里一些实用方案 [2–5] 陆续被提出, 在这些方案中, 由 Shacham 和 Waters 所提出的紧凑可恢复性证明 (CPOR) 方案 [6] 是一个具有代表性的工作. 该方案具有如下的一般框架和特征: 1) 文件被分割成块并且每块生成一个签名标签; 2) 验证者能够通过随机采样的方式来检验文件完整性, 这一特点对于大或特大文件是极其有效的; 3) 同态性质被用来聚合全部采样标签生成一个固定长度的响应 (Response), 这有利于最小化网络通信带宽.

引用格式: 朱岩, 王怀习, 胡泽行, 等. 数据可恢复性的零知识证明. 中国科学: 信息科学, 2011, 41: 1227–1237

虽然各种密码学敌手模型已经被用于证明 POR 方案的安全, 但是这些已存在的方案并不能完全满足交互证明系统 (interactive proof systems, IPS) 的标准安全模型 [7], 因此, 这些 POR 方案的安全性, 特别是验证协议的完备性, 仍然无法保证. 这意味着在验证过程中证明者能够通过篡改数据来欺骗验证者. 更重要的是, 对于公开可验证的 POR 协议, 外包存储数据的保密性也无法保证, 也就是恶意敌手 (验证者) 能够通过分析公开挑战 (Public Challenges) 的响应 (Responses), 轻易地获得所有待验证数据. 因此, 有必要构建了一种交互式证明系统标准模型下的 POR 方案, 以防止欺诈和保护数据证明者的隐私.

1.1 相关工作

为了检查数据的可用性和存储外包数据的完整性, Juels 和 Kaliski[1] 首先提出了可恢复性证明 (POR) 概念和方案, 该方案很大程度上依赖于客户在把文件发送到存储服务器前的预处理过程: 一个被称为 "哨兵" 的数据块被随机插入数据中用于发现异常, 并将文件加密用以隐藏这些 "哨兵", 而且纠错码被用于恢复数据所受到的损坏. 不幸的是, 该方案只能处理有限数量的查询, 这个查询数量需要事先确定, 并且这种方法只适用于加密文件.

类似于 POR 概念, 为确保在不可信存储服务上的文件拥有性证明, Ateniese 等人 [2] 提出了一种被称为 PDP 的模型, 并提供了一个基于 RSA 的方案, 该方案能够对静态数据 (即, 数据不可被修改) 实现 $O(1)$ 的通讯复杂度. 该文作者还提出了该方案的一个公开可验证的版本, 它允许任何人 (不只是数据所有者) 可以向拥有数据的服务器发送 "挑战查询" 实现完整性证明. 由于数据所有者和使用者的分离, 这一性质大大扩展了 PDP 方案的应用领域. 但是, 由于签名信息对于数据块索引的严重依赖, 采用类似于重放攻击的方法, 该文所述方案对动态数据都是不安全的. 为了解决这一问题, Erway 等人 [8] 提出了两个基于 Hash 函数树的动态 PDP 方案. 对于 n 块数据构成的文件, 这些方案实现了 $O(\log n)$ 的通信和计算复杂性.

在上述工作基础上, Shacham 和 Waters[6] 提出了一种基于数据分片技术的通用思想, 即所谓的紧凑 POR(CPOR) 模型. 该模型通过使用数学上的同态性质, 对于 t 个挑战块构成的证明, 能够在 $O(t)$ 计算复杂性下聚合该证明生成一个 $O(1)$ 长度认证值. 事实上, 这种模型可被认为是一些现有方案的通常表示, 并很容易转换成基于 MAC, ECC 或 RSA 的方案. 这些方案被建立在 BLS 签名 [9] 和随机预言模型基础上, 在可公开验证模式下也具有较短的查询和响应. 然而, 这种模型并没有建在交互证明系统基础上, 攻击者仍然可以利用公开验证协议来获得外包存储中的数据.

此外, 一些 POR 方案和模式也已经在最近被提出, 例如文献 [4,5,10]. Dodis 等人在文献 [4] 中讨论了几个变种的问题 (如有界使用和无界使用的对比, 知识稳健和信息稳健的对比), 并给出了在这些变种方案中最优的 POR 方案. Wang 等人 [5] 通过整合上述 CPOR 方案和 Merkle 哈希树 (MHT), 提出了代价为 $O(\log n)$ 的动态方案. Bowers 等人 [10] 在 Juels-Kaliski 和 Shacham-Waters 工作的基础上, 提出了 POR 方案的理论设计框架, 该框架支持对噪声通道下的完全拜占庭敌手模型和纠错编码方法.

1.2 本文工作

本文着眼于具有防止证明者欺诈和验证信息泄露的有效 POR 协议构造. 首先, 基于交互式证明系统的标准模型, 我们首次提出了交互式可恢复性证明 (interactive proofs of retrievability, IPOR) 的形式化定义. 其次, 我们提出了一个更加实用的零知识可恢复性证明 (zero-knowledge POR, ZK-POR), 它能防止公开验证过程中的数据泄漏. 同时, 我们在 Diffie-Hellman (CDH) 假设下, 证明了该方案的完

备性、稳固性, 以及零知识性, 该证明是通过建立多项式时间的知识提取器 (Extractor) 实现的, 并且允许提取器可回卷地黑盒访问证明者. 最后, 性能分析表明所提出 ZK-POR 协议只需发送固定大小的数据量就能够实现承诺、挑战、响应过程, 并可最大限度地减少网络通信. 因此, 我们的方案可用于大范围分布存储系统中实现大尺寸文件的公共远程验证.

本文其余部分组织如下: 在第 2 节, 我们介绍一些基本的符号、共同的 POR 结构, 以及对现有方案的一些攻击. 在第 3 节, 定义了一种基于交互式证明系统的形式化 IPOR 模型. 在第 4 节, 针对该模型提出了一种实用的 ZK-POR 方案. 第 5 和 6 节证明了所提方案的安全性和对其性能予以了评价. 最后, 第 7 节总结全文.

2 预备知识

假设 $\mathcal{H}=\{H_k\}$ 为带密钥的 Hash 函数族 $H_k:\{0,1\}^*\to\{0,1\}^n$, 密钥由 $k\in\mathcal{K}$ 标记. 称算法 $\mathcal{A}$ 在攻击 $\mathcal{H}$ 的抗碰撞游戏中具有优势 ϵ, 如果

$$\Pr[\mathcal{A}(k)=(m_0,m_1):m_0\neq m_1,H_k(m_0)=H_k(m_1)]\geqslant\epsilon,$$

这里, 不确定性是由所有 $k\in\mathcal{K}$ 以及算法 $\mathcal{A}$ 的随机选择引入. 这种 Hash 函数可以从 BLS 签名 [9] 中获取.

定义 1 (抗碰撞 Hash 函数族) 如果不存在 t- 时间的敌手在攻击 $\mathcal{H}$ 的抗碰撞游戏中具有至少 ϵ 优势, 那么称该 Hash 函数族 $\mathcal{H}$ 为 (t,ϵ)– 抗碰撞的.

本文采用 Boneh 和 Franklin[11] 提出的双线性映射来构造方案. 假设 $\mathbb{G}$ 和 $\mathbb{G}_T$ 为两个乘法群, 其阶均为大素数 p. 函数 e 为具有如下性质的可计算双线性映射 $e:\mathbb{G}\times\mathbb{G}\to\mathbb{G}_T$: 对于任意的 $G,H\in\mathbb{G}$ 和所有的 $a,b\in\mathbb{Z}_p$, 有, 1) 双线性性: $e(G^a,H^b)=e(G,H)^{ab}$; 2) 非退化性: $e(G,H)\neq1$ 除非 G 或 $H=1$; 3) 可计算性: $e(G,H)$ 可有效地计算.

定义 2 (双线性映射群系统) 双线性映射群系统是指一个满足上述性质的四元组 $\mathbb{S}=\langle p,\mathbb{G},\mathbb{G}_T,e\rangle$.

Shacham 和 Waters[6] 提出了一种普遍的紧凑可恢复性证明 (CPOR) 模型, 具体过程如下: 给定文件 F, 客户将文件 F 分成 n 块 $(m_1,\ldots,m_n)$, 并且每一块 m_i 又分为 s 个分区 (sector) $(m_{i,1},\ldots,m_{i,s})\in\mathbb{Z}_p^s$, 其中, p 为充分大的素数. 假设 $e:\mathbb{G}\times\mathbb{G}\to\mathbb{G}_T$ 为双线性映射, g 为群 $\mathbb{G}$ 的生成元, 以及一个 BLS Hash 函数 $H:\{0,1\}^*\to\mathbb{G}$. 私钥为 $sk=x\in_R\mathbb{Z}_p$, 相应的公钥为 $pk=(g,v=g^x)$. 客户选择 s 个随机元素 $u_1,\ldots,u_s\in_R\mathbb{G}$ 作为验证信息 $t=(Fn,u_1,\ldots,u_s)$, 其中, Fn 为文件名. 对于每个 $i\in[1,n]$, 第 i 块的标签 (Tag) 为 $\sigma_i=(H(Fn||i)\cdot\prod_{j=1}^s u_j^{m_{i,j}})^x$. 对于索引集 (Index Set) I, 服务端在收到询问 $Q=\{(i,v_i)\}_{i\in I}$ 之后, 计算并送回响应 $\sigma'\leftarrow\prod_{(i,v_i)\in Q}\sigma_i^{v_i}$ 和 $\mu=(\mu_1,\ldots,\mu_s)$, 其中 $\mu_j\leftarrow\sum_{(i,v_i)\in Q}v_im_{i,j}$. 验证方程为

$$e(\sigma',g)=e\left(\prod_{(i,v_i)\in Q}H(Fn||i)^{v_i}\cdot\prod_{j=1}^s u_j^{\mu_j},v\right).$$

由于该方案存在文件信息的泄露问题, 因此方案是不安全的. 这种攻击如下:

攻击 1 对于具有 $n\times s$ 个分区的文件, 敌手可以通过执行 n 次验证或窃听 n 次验证信息的方式得到文件和标签信息.

证明 假设 s 为每个文件块中分区的个数. 给定 n 次挑战 (Challenge) $(Q^{(1)},\ldots,Q^{(n)})$ 以及它们的反应 $((\sigma'^{(1)},\mu^{(1)}),\ldots,(\sigma'^{(n)},\mu^{(n)}))$, 敌手可以计算方程组 $\mu_i^{(k)}=m_{1,i}\cdot v_1^{(k)}+\cdots+m_{n,i}\cdot v_n^{(k)}$, 其中 $k\in[1,n]$, 从中获取 $\{m_{1,i},\ldots,m_{n,i}\}$, 其中 $\mu^{(k)}=(\mu_1^{(k)},\ldots,\mu_s^{(k)})$ 和 $Q^{(k)}=\{(i,v_i)\}_{i\in I}$. 通过 s 次计算这些方程组 $(i\in[1,s])$, 敌手可以得到整个文件 $F=\{m_{i,j}\}_{j\in[1,s]}^{i\in[1,n]}$. 类似地, 敌手可以通过 $\sigma'^{(1)},\ldots,\sigma'^{(n)}$ 得到所有的标签 $\sigma_1,\ldots,\sigma_n$.

3 可恢复性的交互式证明

3.1 定义

基于交互证明系统 (IPS) 的定义, 我们给出如下一种交互式可恢复性证明的形式化定义:

定义 3 (交互式可恢复性证明) 交互式可恢复性证明方案 $\mathcal{S}=(\mathcal{K}eyGen,\mathcal{T}agGen,\mathcal{P}roof)$ 是由两个算法和一个交互式证明系统组成的:

$\mathcal{K}eyGen(1^\kappa)$: 输入为安全参数 κ, 输出私钥 sk 或者公私钥对 (pk,sk);

$\mathcal{T}agGen(sk,F)$: 输入为私钥 sk 和文件 F, 输出三元组 (ζ,ϕ,σ), 其中 ζ 表示用来生成验证标签的秘密, ϕ 表示公开验证参数 u 组成的集合以及索引信息 χ, 即 $\phi=(u,\chi)$, σ 表示验证标签集;

$\mathcal{P}roof(P,V)$: 为证明者 (prover, P) 和验证者 (verifier, V) 之间的可恢复性证明协议. 运行协议后, V 返回 {0|1}, 其中 1 代表服务器上的文件得到正确完整的存储. 这包含两种情形:

•$\langle P(F,\sigma),V(sk,\zeta)\rangle$ 是一个私密证明, 其中 P 的输入为文件 F 和一组标签集 σ, V 的输入为私钥 sk 和私密标签 ζ;

•$\langle P(F,\sigma),V\rangle(pk,\phi)$ 是一个公开证明, 其中 P 的输入为文件 F 和和一组标签集 σ, V 的输入为公钥 pk 和 P 与 V 之间的一组公开参数 ϕ.

其中, $P(x)$ 表示实体 P 具有秘密 x, $\langle P,V\rangle(x)$ 表示成员 P 和 V 在协议中共享公共的数据 x.

这是一个较目前已有的 POR 模型更为广义的模型. 由于验证过程可以视为一个交互式协议, 该定义没有对验证的具体参数提出限制, 诸如: 规模、顺序, 以及协议步数等, 所以该定义可以对协议构造提供更多的便利. 需要说明的是, 本文只关注公开证明协议的构造.

3.2 安全要求

按照 Bellare 和 Goldreich[7] 提出的交互式证明系统的标准定义, 协议 $\mathcal{P}roof(P,V)$ 需满足两个要求:

定义 4 (安全的 IPOR) 给定文件 F, 交互算法 (P,V) 被称为是一个安全的交互式可恢复性证明, 如果 P 为 (无界的) 概率算法, V 为确定性多项式时间算法, 并且存在多项式 $p_1(\cdot)$ 和 $p_2(\cdot)$, 对于所有的 $\kappa\in\mathbb{N}$, 下面两个安全属性成立:

• 完备性: 对于所有的 $\sigma\in\mathcal{T}agGen(sk,F)$, 有

$$\Pr[\langle P(F,\sigma),V\rangle(pk,\phi)=1]\geqslant 1-1/p_1(\kappa); \tag{1}$$

• 稳固性: 对于每个 $\sigma^*\notin\mathcal{T}agGen(sk,F)$, 任意的交互算法 P^*,

$$\Pr[\langle P^*(F,\sigma^*),V\rangle(pk,\phi)=1]\leqslant 1/p_2(\kappa); \tag{2}$$

其中 $p_1(\cdot)$ 和 $p_2(\cdot)$ 为两个多项式, κ 为 $\mathcal{K}eyGen(1^\kappa)$ 中的安全参数.

该定义中, 函数 $1/p_1(\kappa)$ 称为完备性误差, 函数 $1/p_2(\kappa)$ 称为稳固性误差. 一般地, 我们要求 $1/p_1(\kappa)+1/p_2(\kappa)\leqslant 1-1/poly(\kappa)$.

稳固性意味着误导验证者接受错误的陈述是困难的. 稳固性也可以看做是一种比文件标签的不可伪造更严格的安全要求. 因此, 上述定义表明了如果稳固性成立的话, 那么证明者无法伪造文件标签或者修改数据.

在现有的方案中, 验证者易于得到数据块及其标签. 为了保护检验数据的保密性，验证过程中私密信息的泄露问题也必须引起格外的关注. 为了解决这个问题, 我们将交互系统中的零知识性 (zero-knowledge) 引入 IPOR 系统中来, 具体如下:

定义 5(零知识性) 交互式可恢复性证明 (IPOR) 方案被称为计算零知识的, 如果存在概率多项式时间算法 S^*(称为模拟器) 使得对于任意的概率多项式时间速算法 D、任意多项式 $p(\cdot)$, 以及任意充分大的 κ, 满足如下等式

$$\left|\begin{array}{c}\Pr[D(pk,\phi,S^*(pk,\phi))=1]-\\ \Pr[D(pk,\phi,\langle P(F,\sigma),V^*\rangle(pk,\phi))=1]\end{array}\right|\leqslant 1/p(\kappa),$$

其中, $S^*(pk,\phi)$ 表示模拟器 S 在输入为 (pk,ϕ) 时的输出, $\langle P(F,\sigma),V^*\rangle(pk,\phi)$ 表示 V^* 和 $P(F,\sigma)$ 在公共输入为 (pk,ϕ) 时的交互协议输出. 也就是说, 对于任意的 $\sigma\in TagGen(sk,F)$, 两个集合 $S^*(pk,\phi)$ 和 $\langle P(F,\sigma),V^*\rangle$ (pk,ϕ) 是计算不可区分的.

事实上, 零知识性刻画了证明者 P 抵抗证明者在交互过程中获取知识的鲁棒性. 对于 POR 方案而言, 我们利用零知识性保证数据块和签名标签的隐私性. 综上所述, 定义如下零知识可恢复性证明 (ZK-POR):

定义 6 (零知识可恢复性证明) 交互式可恢复性证明 (IPOR) 方案被称为零知识可恢复性证明, 如果完备性、知识稳固性和零知识性都成立.

4 零知识可恢复性证明的构造

本节中, 我们将给出一种实用的零知识可恢复性证明构造. 在该构造中, 验证协议为 3– 步 (3-move) 结构: 承诺、挑战和响应. 该协议类似于 Schnorr Σ 协议 [12], 由于 Schnorr Σ 协议为经典的零知识证明系统, 因此我们的构造也享有一些好的安全性质. 我们给出 IPOR 的构造如下：

$\mathcal{K}eyGen(1^\kappa)$: 假设 $\mathbb{S}=(p,\mathbb{G},\mathbb{G}_T,e)$ 为双线性映射群系统, 具有随机生成元 $g,h\in_R\mathbb{G}$, 其中 $\mathbb{G}$, $\mathbb{G}_T$ 为两个大素数 p 阶群, $|p|=O(\kappa)$. 生成一个抗碰撞的 Hash 函数 $H_k(\cdot)$, 选定两个随机数 $\alpha,\beta\in_R\mathbb{Z}_p$ 并计算 $H_1=h^\alpha$ 和 $H_2=h^\beta\in\mathbb{G}$. 因此, 私钥为 $sk=(\alpha,\beta)$, 公钥为 $pk=(g,h,H_1,H_2)$.

$\mathcal{T}agGen(sk,F)$: 将文件 F 分成 $n\times s$ 个分区 $F=\{m_{i,j}\}\in\mathbb{Z}_p^{n\times s}$. 选择 s 个随机数 $\tau_1,\ldots,\tau_s\in\mathbb{Z}_p$ 作为该文件的秘密, 并对于每个 $i\in[1,s]$ 计算 $u_i=g^{\tau_i}\in\mathbb{G}$ 和 $\xi^{(1)}=H_\xi(\text{“}Fn\text{”})$, 其中, $\xi=\sum_{i=1}^s\tau_i$ 和 Fn 为文件名. 建立索引表 $\chi=\{\chi_i\}_{i=1}^n$, 对于 $i\in[1,n]$ 在 χ 中加入 χ_i, 其中, 索引表 $\chi=\{\chi_i\}_{i\in[1,n]}$ 可以用来支持动态数据操作, 例如, 定义 $\chi_i=(B_i||V_i||R_i)$ 并初始化为 $\chi_i=(B_i=i,V_i=1,R_i\in_R\{0,1\}^*)$, B_i 为文件块的序列号, R_i 为该数据块的版本号以及 R_i 为避免碰撞加入的随机整数. 然后, 标签可计算如下:

$$\sigma_i\leftarrow(\xi_i^{(2)})^\alpha\cdot g^{\sum_{j=1}^s\tau_j\cdot m_{i,j}\cdot\beta}\in\mathbb{G},$$

其中 $\xi_i^{(2)} = H_{\xi^{(1)}}(\chi_i), i \in [1,n]$. 最后, 令 $u = (\xi^{(1)}, u_1, \ldots, u_s)$ 并输出 $\zeta = (\tau_1, \ldots, \tau_s)$, $\phi = (u, \chi)$ 给可信第三方 (TTP, trusted third part), 将 $\sigma = (\sigma_1, \ldots, \sigma_n)$ 发送给存储服务提供者 (storage service provider, SSP).

$\mathcal{P}roof(P,V)$: 这是一个证明者 P (SSP) 和验证者 V (client) 之间的 3– 步协议, 它们的共同输入 (pk, ϕ) 存储在一个 TTP 之上. 交互协议具体如下:

• 承诺 $(P \to V)$: P 选择随机数 $\gamma \in_R \mathbb{Z}_p$; 并对每个 $j \in [1,s]$ 选取 $\lambda_j \in_R \mathbb{Z}_p$; 而后, 将承诺 $C = (H_1', \pi)$ 发送给 V, 其中, $H_1' = H_1^{\gamma}$ 和 $\pi \leftarrow e(\prod_{i=1}^{s} u_j^{\lambda_j}, H_2) \in \mathbb{G}_T$;

• 挑战 $(P \leftarrow V)$: V 随机选取具有 t 个索引的挑战集合 I 以及 t 个随机系数 $v_i \in \mathbb{Z}_p^*$, 其中 $t = |I|$; 假设 $Q = \{(i, v_i)\}_{i \in I}$ 为挑战索引系数组成的集合, V 将 Q 发送给 P;

• 反应 $(P \to V)$: P 计算响应 θ 和 μ 如下:

$$\sigma' \leftarrow \prod_{(i,v_i) \in Q} \sigma_i^{\gamma \cdot v_i}, \quad \mu_j \leftarrow \lambda_j + \gamma \cdot \sum_{(i,v_i) \in Q} v_i \cdot m_{i,j},$$

其中 $\mu = \{\mu_j\}_{j \in [1,s]}$; 最后, P 将 $\theta = (\sigma', \mu)$ 发送给 V.

验证 在 3– 步交互过程之后, 验证者 V 可以通过下面的等式检验反应是否正确

$$\pi \cdot e(\sigma', h) \stackrel{?}{=} e\Big(\prod_{(i,v_i) \in Q} (\xi_i^{(2)})^{v_i}, H_1' \Big) \cdot e\Big(\prod_{j=1}^{s} u_j^{\mu_j}, H_2 \Big). \tag{3}$$

为了防止验证过程中存储数据和标签的泄露, 秘密数据 $\{m_{i,j}\}$ 由随机数 $\lambda_j \in \mathbb{Z}_p$ 来保护, 而且标签 $\{\sigma_i\}$ 被随机数 $\gamma \in \mathbb{Z}_p$ 进行随机化. 进一步, 为了避免敌手获得数值 $\{\lambda_j\}$ 和 γ, 我们利用简单的承诺的方法来保护, 即 H_1^{γ} 和 $e(\prod_{i=1}^{s} u_j^{\lambda_j}, H_2)$.

5 安全性证明

我们的构造是一个满足完备性和稳固性的有效交互证明系统, 具体如下:

1) 完备性: 对于每个可用的标签 $\sigma \in \mathcal{T}agGen(sk, F)$ 和随机挑战 $Q = (i, v_i)_{i \in I}$, 协议的完备性具体阐述如下:

$$\begin{aligned}
\pi \cdot e(\sigma', h) &= e(g,h)^{\beta \sum_{j=1}^{s} \tau_j \cdot \lambda_j} \cdot e\Big(\prod_{(i,v_i) \in Q} (\xi_i^{(2)})^{v_i}, h \Big)^{\alpha \cdot \gamma} \cdot e(g,h)^{\gamma \cdot \beta \sum_{j=1}^{s} (\tau_j \cdot \sum_{(i,v_i) \in Q} v_i \cdot m_{i,j})} \\
&= e(g,h)^{\beta \sum_{j=1}^{s} \tau_j \cdot \lambda_j} \cdot e\Big(\prod_{(i,v_i) \in Q} (\xi_i^{(2)})^{v_i}, h \Big)^{\alpha \cdot \gamma} \cdot e(g,h)^{\beta \sum_{j=1}^{s} (\tau_j \cdot \mu_j - \tau_j \cdot \lambda_j)} \\
&= e\Big(\prod_{(i,v_i) \in Q} (\xi_i^{(2)})^{v_i}, h^{\alpha \cdot \gamma} \Big) \cdot \prod_{j=1}^{s} e(u_j^{\mu_j}, h^{\beta}).
\end{aligned}$$

当所有的 $i \in I$ 满足 $v_i = 0$ 时, 存在平凡解. 此时, $\sigma' = 1$, $\mu_j = \lambda_j$, 和 $\pi_j = u_j^{\mu_j}$, 上面的方程无法判定访问的文件可恢复性. 因此, 协议的完备性成立:

$$\Pr[\langle P(F, \sigma), V \rangle (pk, \phi) = 1] \geqslant 1 - 1/p^t,$$

其中, t 为 Q 中索引系数对的个数. 事实上, 我们要求 $v_i \in_R \mathbb{Z}_p^*$.

2) 稳固性: 对于每个标签 $\sigma^* \notin \mathcal{T}agGen(sk, F)$, 为了证明不存在欺骗性的证明者 P^*, 我们采取反证法来证明协议稳固性如下: 假定存在一个多项式时间内的欺骗性的证明者 P^*, 那么我们利用该证明者 P^* 可构造知识提取器 $\mathcal{M}$[7,13] 来求解计算性 Diffie-Hellman 问题 (CDH), 即, $\mathcal{M}$ 的公共输入为 (pk, ϕ), 它可回卷地黑盒访问证明者 P^*, 而后希望求解 $\mathbb{G}$ 中的计算性 Diffie-Hellman 问题: 给定 3 个随机数 $G, G_1 = G^a, G_2 = G^b \in_R \mathbb{G}$, 输出 $G^{ab} \in \mathbb{G}$. 具体来讲, 有如下定理:

引理 1 假设群 $\mathbb{G}$ 中的计算 Diffie-Hellman 问题 (CDH) 是 (t, ϵ) 困难的, 那么所提 IPOR 方案在随机预言机和回卷的知识提取器模型下具有 (t, ϵ') 的知识稳固性, 其中 $\epsilon' \geqslant \epsilon$.

证明 对于某个无效的标签 $\{\sigma^*\} \notin \mathcal{T}agGen(sk, F)$, 我们假设存在交互机 P^* 能够以显著的概率通过验证, 也就是说, 存在多项式 $p(\cdot)$, 对充分大的 κ,

$$\Pr[\langle P^*(F, \{\sigma^*\}), V\rangle(pk, \phi) = 1] \geqslant 1/p(\kappa). \tag{4}$$

利用 P^*, 我们可构造一个概率算法 $\mathcal{M}$ (称作知识提取器) 来求解 p 阶循环群 $\mathbb{G} \in \mathbb{S}$ 中的计算 Diffie-Hellman CDH 问题. 即给定 $G, G_1, G_2 \in_R \mathbb{G}$, 输出 $G^{ab} \in \mathbb{G}$, 其中 $G_1 = G^a$, $G_2 = G^b$. 算法 $\mathcal{M}$ 通过与 P^* 进行下面的交互过程:

初始化: $\mathcal{M}$ 选取随机的 $r \in_R \mathbb{Z}_p$ 并设定 $g = G$, $h = G^r$, $H_1 = G_1^r$, $H_2 = G_2^r$ 作为公钥 $pk = (g, h, H_1, H_2)$, 并将公钥发送给 P^*;

学习: 给定文件 $F = \{m_{i,j}\}_{j\in[1,s]}^{i\in[1,n]}$, 首先对于 $i \in [1, s]$, $\mathcal{M}$ 选择 s 个随机的 $\tau_i \in_R \mathbb{Z}_p$ 和 $u_i = G_2^{\tau_i}$. 其次, $\mathcal{M}$ 将索引 $\{1, \ldots, n\}$ 分为两个集合 $T = \{t_1, \ldots, t_{\frac{n}{2}}\}$ 和 $T' = \{t'_1, \ldots, t'_{\frac{n}{2}}\}$. 假设对于所有的 $i \in [1, n/2]$ 和 $j \in [1, s]$, 我们有 $m_{t_i,j} \neq m_{t'_i,j}$. 然后, $\mathcal{M}$ 利用原始方案生成索引表 χ 和 $\xi^{(1)}$ 并计算每块文件的标签, 具体如下:

- 对每个 $t_i \in T$, $\mathcal{M}$ 选择 $r_i \in_R \mathbb{Z}_p$ 并设定 $\xi_{t_i}^{(2)} = H_{\xi^{(1)}}(\chi_{t_i}) = G^{r_i}$ 和 $\sigma_{t_i} = G_1^{r_i} \cdot G_2^{\sum_{j=1}^{s} \tau_j \cdot m_{t_i,j}}$.
- 对每个 $t'_i \in T'$, $\mathcal{M}$ 利用 r_i 和两个随机数 $r'_i, \zeta_i \in_R \mathbb{Z}_p$ 计算 $\xi_{t'_i}^{(2)} = H_{\xi^{(1)}}(\chi_{t'_i}) = G^{r_i} \cdot G_2^{r'_i}$ 和 $\sigma_{t'_i} = G_1^{\zeta_i} \cdot G_2^{\sum_{j=1}^{s} \tau_j \cdot m_{t'_i,j}}$.

对于所有的 $t'_i \in T'$, $\mathcal{M}$ 检验等式 $e(\sigma_{t'_i}, h) \stackrel{?}{=} e(\xi_{t'_i}^{(2)}, H_1) \cdot e(\prod_{j=1}^{s} u_j^{m_{t'_i,j}}, H_2)$ 是否成立. 如果所有等式成立, $\mathcal{M}$ 输出 $G^{ab} = G_2^a = (G^{\zeta_i} \cdot G_1^{r_i})^{(r'_i)^{-1}}$, 否则 $\mathcal{M}$ 发送 $(F, \sigma^* = \{\sigma_i\}_{i=1}^n)$ 和 $\phi = (\xi^{(1)}, u = \{u_i\}, \chi)$ 给 P^*.

Hash 查询: 在任何时间, P^* 可以询问 Hash 函数 $H_{\xi^{(1)}}(\chi_k)$, $\mathcal{M}$ 必须保证响应 $\xi_{t_i}^{(2)}$ 或者 $\xi_{t'_i}^{(2)}$ 保持一致, 其中, $k = t_i$ 或 t'_i.

输出: $\mathcal{M}$ 选择索引集合 $I \subset [1, \frac{n}{2}]$ 和两个子集 I_1, I_2, 其中 $I = I_1 \bigcup I_2$, $|I_2| > 0$. $\mathcal{M}$ 构造挑战 $\{v_i\}_{i\in I}$, 其中所有的 $v_i \neq 0$. 然后 $\mathcal{M}$ 模拟 V 运行下面的交互协议 $\langle P^*, \mathcal{M}\rangle$:

- 承诺 (Commitment): $\mathcal{M}$ 从 P^* 处收到 (H'_1, π');
- 挑战 (Challenge): $\mathcal{M}$ 向 P^* 发送挑战 $Q_1 = \{(t_i, v_i)\}_{i\in I}$;
- 反应 (Response): $\mathcal{M}$ 从 P^* 处收到 $(\sigma', \{\mu'_j\}_{j=1}^s)$.

$\mathcal{M}$ 利用方程 (3) 检验这些反应是否为有效的. 如果所有的反应为有效的, 那么 $\mathcal{M}$ 对证明者 P^* 进行如下可回卷的访问:

- 承诺 (Commitment): $\mathcal{M}$ 从 P^* 处收到 (H''_1, π'');
- 挑战 (Challenge): $\mathcal{M}$ 向 P^* 发送挑战 $Q_2 = \{(t_i, v_i)\}_{i\in I_1} \bigcup \{(t'_i, v_i)\}_{i\in I_2}$;
- 反应 (Response): $\mathcal{M}$ 从 P^* 处收到 $(\sigma'', \{\mu''_j\}_{j=1}^s)$ 或者一个特定的终止符号.

如果反应不是终止符的话, 那么 $\mathcal{M}$ 通过方程 (3) 检验这些反应是否为有效的, 以及 $H'_1 \stackrel{?}{=} H''_1$ 和

$\pi' \stackrel{?}{=} \pi''$. 如果上述检验全部正确, 那么对所有的 $j \in [1,s]$ $\mathcal{M}$ 计算

$$\gamma = \frac{\mu_j'' - \mu_j'}{\sum_{i\in I_2} v_i \cdot (m_{t_i',j} - m_{t_i,j})},$$

并验证方程 $H_1' \stackrel{?}{=} H_1^{\gamma}$ 以保证访问为有效的可回卷的访问. 最后, $\mathcal{M}$ 输出

$$G^{ab} = G_2^a = \left(\sigma'' \cdot \sigma'^{-\phi} \cdot G_1^{\gamma\cdot(\phi-1)\sum_{i\in I} r_i v_i}\right)^{\frac{1}{\gamma\cdot\sum_{i\in I_2} r_i'\cdot v_i}}, \tag{5}$$

其中,

$$\phi = \frac{\sum_{i\in I_1}\sum_{j=1}^{s}\tau_j m_{t_i,j} v_i + \sum_{i\in I_2}\sum_{j=1}^{s}\tau_j m_{t_i',j} v_i}{\sum_{i\in I}\sum_{j=1}^{s}\tau_j m_{t_i,j} v_i},$$

并且 $\phi \neq 1$, 算法终止.

下面我们分析上述算法 $\mathcal{M}$ 求解计算性 Diffie-Hellman 问题的有效性. 显然, 我们在上面的构造中可以设定 $\alpha = a$ 和 $\beta = b$. 由于对于 $\forall t_i \in T$, 标签 σ_{t_i} 为有效的, 第一次交互中的反应满足方程

$$\pi' \cdot e(\sigma', h) = e\Big(\prod_{i\in I}(\xi_{t_i}^{(2)})^{v_i}, H_1'\Big) \cdot e\Big(\prod_{j=1}^{s} u_j^{\mu_j'}, H_2\Big) = e(G^{\sum_{i\in I} r_i\cdot v_i}, H_1') \cdot e\Big(\prod_{j=1}^{s} u_j^{\mu_j'}, H_2\Big).$$

尽管如此, $\forall t_i' \in T'$, $\{\sigma_{t_i'}\}$ 的值仍然无法得到. 在第二次交互中, 我们要求 $\mathcal{M}$ 可以使得证明者 P^* 回卷, 即两次协议执行的参数完全相同 [7,13]. 在上面的构造中, 该性质保证了对于所有的 $i \in [1,s]$, 有 $H_1' = H_1''$, $\pi' = \pi''$ 以及

$$\mu_j'' - \mu_j' = \gamma \cdot \sum_{i\in I} v_i \cdot (m_{t_i',j} - m_{t_i,j}) = \gamma \cdot \sum_{i\in I_2} v_i \cdot (m_{t_i',j} - m_{t_i,j}).$$

对于所有的 γ, 从方程检验等式 $H_1' = H_1^{\gamma}$, 我们可以确定在两次执行中对于 $i \in [1,s], \lambda_i' = \lambda_i''$ 的一致性. 因此, 我们有 $e(\prod_{j=1}^{s} u_j^{\mu_j'}, H_2) \cdot \pi'^{-1} = e(G_2, H_2)^{\sum_{i\in I}\sum_{j=1}^{s}\tau_j m_{t_i,j} v_i}$ 和

$$e\Big(\prod_{j=1}^{s} u_j^{\mu_j''}, H_2\Big) \cdot \pi''^{-1} = e(G_2, H_2)^{\sum_{i\in I_1}\sum_{j=1}^{s}\tau_j m_{t_i,j} v_i} \cdot e(G_2, H_2)^{\sum_{i\in I_2}\sum_{j=1}^{s}\tau_j m_{t_i',j} v_i}.$$

这意味着 $e(\prod_{j=1}^{s} u_j^{\mu_j''}, H_2) \cdot \pi''^{-1} = (e(\prod_{j=1}^{s} u_j^{\mu_j'}, H_2) \cdot \pi'^{-1})^{\phi}$. 利用响应, 我们得到

$$\begin{aligned}
e(\sigma'', h) &= e\Big(\prod_{i\in I_1}(\xi_{t_i}^{(2)})^{v_i} \cdot \prod_{i\in I_2}(\xi_{t_i'}^{(2)})^{v_i}, H_1''\Big) \cdot e\Big(\prod_{j=1}^{s} u_j^{\mu_j''}, H_2\Big) \cdot (\pi'')^{-1} \\
&= e\Big(\prod_{i\in I_1}(G^{r_i})^{v_i} \cdot \prod_{i\in I_2}(G^{r_i} \cdot G_2^{r_i'})^{v_i}, H_1''\Big) \cdot e\Big(\prod_{j=1}^{s} u_j^{\mu_j''}, H_2\Big) \cdot \pi''^{-1} \\
&= e\Big(\prod_{i\in I} G^{r_i\cdot v_i}, H_1'\Big) \cdot e\Big(\prod_{i\in I_2} G_2^{r_i'\cdot v_i}, H_1'\Big) \cdot \Big(e\Big(\prod_{j=1}^{s} u_j^{\mu_j'}, H_2\Big) \cdot \pi'^{-1}\Big)^{\phi} \\
&= e\Big(\prod_{i\in I} G^{r_i\cdot v_i}, H_1'\Big) \cdot e\Big(\prod_{i\in I_2} G_2^{r_i'\cdot v_i}, H_1'\Big) \cdot \Big(e\Big(\sigma', h\Big)^{\phi} \cdot e(\prod_{i\in I} G^{r_i\cdot v_i}, H_1'\Big)^{-\phi}\Big) \\
&= e(\sigma'^{\phi}, h) \cdot e(G_2^{\sum_{i\in I_2} r_i' v_i} \cdot G^{(1-\phi)\sum_{i\in I} r_i v_i}, H_1').
\end{aligned}$$

据此, 我们得到方程 $e(\sigma''\cdot\sigma'^{-\phi},h)=e(G_2^{\sum_{i\in I_2}r_i'\cdot v_i}\cdot G^{(1-\phi)\sum_{i\in I}r_iv_i},H_1')$, $H_1'=h^{a\gamma}$, 以及 $G_1=G^a$, 因此方程 (5) 成立. 进一步地, 我们有

$$\Pr[\mathcal{M}(\mathrm{CDH}(G,G^a,G^b))=G^{ab}]\geqslant\Pr[\langle P^*(F,\{\sigma^*\}),\mathcal{M}\rangle(pk,\phi)=1]\geqslant 1/p(\kappa).$$

因此, 算法 $\mathcal{M}$ 能够以至少 ϵ 的优势求解给定的 ϵ-CDH 挑战. 定理得证.

引理 2 所提 IPOR 方案中的验证协议 $\mathcal{P}roof(P,V)$ 为计算零知识系统.

证明 对于协议 $\mathcal{P}roof(P,V)$, 我们构造 P 和 V 交互的模拟器 $S^*(pk,\phi)$. 给定公钥 $pk=(g,h,H_1,H_2)$, 对于文件 F, 公钥验证信息 $\phi=(\xi^{(1)},u_1,\ldots,u_s,\chi)$ 和索引集 I ($t=|I|$), 模拟器 $S^*(pk,\phi)$ 执行如下操作:

1. 选取随机数 $\sigma'\in_R\mathbb{G}$ 并计算 $e(\sigma',h)$;
2. 选取 t 个随机系数 $\{v_i\}_{i\in I}\in_R\mathbb{Z}_p^t$ 和 $\gamma\in_R\mathbb{Z}_p$, 计算 $H_1'\leftarrow H_1^{\gamma}$ 和 $A_1\leftarrow e(\prod_{i\in I}H_{\xi^{(1)}}(\chi_i)^{v_i},H_1')$;
3. 选取 s 个随机数 $\{\mu_i\}\in_R\mathbb{Z}_p^s$ 并计算 $A_2\leftarrow e(\prod_{j=1}^s u_j^{\mu_j},H_2)$;
4. 计算 $\pi\leftarrow A_1\cdot A_2\cdot e(\sigma',h)^{-1}$;
5. 输出 $S^*(pk,\phi)=(C,Q,\theta)=((H_1',\pi),\{(i,v_i)\}_{i=1}^t,(\sigma',\mu))$.

显然, 对于方程 (3) 来说, 模拟器 $S^*(pk,\phi)$ 的输出为有效的验证. 令 $\langle P(F,\sigma),V^*\rangle(pk,\phi)=((\overline{H_1'},\overline{\pi}),\{(i,\overline{v_i})_{i=1}^t\},(\overline{\sigma'},\overline{\mu}))$ 表示交互机 V^* 在公共输入为 (pk,ϕ) 时与交互机 P 交互之后的输出. 事实上, 两个集合中的任意一对变量都是同分布的, 由于 $\gamma,\{v_i\}\in_R\mathbb{Z}_p$, $\overline{H_1'},\{(i,\overline{v_i})\}$ 和 $H_1',\{(i,v_i)\}$ 是同分布的; 同样地, 由于 $\sigma'\in_R\mathbb{G}$, $\lambda_j\in_R\mathbb{Z}_p$, 以及对于 $i\in[1,s]$, $u_j\leftarrow\lambda_j+\gamma\sum_{i\in I}v_i\cdot m_{i,j}$, 我们有 $(\overline{\sigma'},\overline{\mu})$ 和 (σ',μ) 为同分布的. 由于 $\overline{\pi}$ 的分布和所有随机选取的 λ_i 同分布, π 的分布由上述变量的随机赋值决定, 所以变量 $\overline{\pi}$ 和 π 为计算不可区分的.

因此, 集合 $S^*(pk,\phi)$ 和 $\langle P(F,\sigma),V^*\rangle(pk,\phi)$ 为计算不可区分的. 据此, 对于任意的概率多项式时间算法 D, 任意的多项式 $p(\cdot)$ 和充分大的 κ, 我们有如下等式成立:

$$\left|\begin{array}{c}\Pr[D\,(pk,\phi,S^*(pk,\phi))=1]-\\ \Pr[D\,(pk,\phi,\langle P(F,\sigma),V^*\rangle(pk,\phi))]=1\end{array}\right|\leqslant 1/p(\kappa).$$

该模拟器的存在意味着 V^* 无法从 P 中得到任何信息, 由于相同的输出可以在不访问 P 的条件下得到. 也就是说, 交互协议 $\mathcal{P}roof(P,V)$ 为零知识协议.

根据引理 1 和 2, 我们有下面的定理:

定理 1 在 CDH 假设下, 所提 IPOR 方案为随机预言机和回卷的提取器模型下的零知识可恢复性证明 (ZK-POR) 系统.

6 性能分析

首先, 分析 IPOR 方案的计算复杂性. 为了清晰起见, 我们将计算代价在表 1 中逐项列出. 其中, 利用 $[E]$ 表示群 $\mathbb{G}$(或群 $\mathbb{G}_T$) 中一次指数计算的时间, 也就是说, 计算 g^x 所需要的时间, 其中 x 为 $\mathbb{Z}_p$ 中的整数, $g\in\mathbb{G}$ 或者 $g\in\mathbb{G}_T$. 忽略 $\mathbb{Z}_p$ 中的代数运算和群中每次乘法计算的时间, 因为这些运算的效率足够高 [14]. 计算代价最高的运算为双线性映射 $e(\cdot,\cdot)$, 记其每次运算的时间为 $[B]$.

其次, 分析方案的存储和通信复杂性. 双线性映射的通常形式为 $e:E(\mathbb{F}_{p^m})\times E(\mathbb{F}_{p^{km}})\rightarrow\mathbb{F}_{p^{km}}^*$(见文献 [15,16]), 其中 p 为素数, m 为正整数, 且 k 为嵌入次数 (或者称为安全乘子). 此时, 我们利用非

表 1　IPOR 方案的存储/通信以及计算复杂度

Table 1　The storage/communication and computation overheads in our IPOR scheme

Algorithm		Computation overheads	Communication overheads
$\mathcal{K}eyGen$		$2[E]$	$2l_0$
$\mathcal{T}agGen$		$(2n+s)[E]$	nsl_0+nl_1
$\mathcal{P}roof$	Commitment	$[B]+(s+1)[E]$	l_2+l_T
	Challenge		$2tl_0$
	Response	$t[E]$	sl_0+l_1
	Verification	$3[B]+(t+s)[E]$	

对称双线性对 $e:\mathbb{G}_1\times\mathbb{G}_2\to\mathbb{G}_T$ 代替原方案中的对称双线性对. 不失一般性, 假设安全参数 κ 为 80 bits, 我们需要在试验中的椭圆曲线基域 $\mathbb{F}_p$ 参数满足 $|p|=160$ bits 和 $m=1$. 这意味着 $\mathbb{Z}_p$ 中整数的长度 $l_0=2\kappa$. 类似地, $\mathbb{G}_1$ 中元素长度 $l_1=4\kappa$, $\mathbb{G}_2$ 中元素长度 $l_2=24\kappa$, $\mathbb{G}_T$ 中元素长度 $l_T=24\kappa$ 并且嵌入指数 $k=6$. 基于上述定义, 所提方案的存储或通信代价如表 1 所示. 例如, 对于某个大小 1 MB 的文件, 选定 $s=200$, 标签的额外存储为 $250*40=10$ KB (其中, $n=250$), 相应承诺和响应阶段的负载分别为 $240+240=480$ Bytes 和 $200*20+40\approx4$ KB. 显然, 方案在验证协议中的承诺和响应阶段具有固定大小的通信负载. 进一步讲, 给定一个具有 $sz=n\cdot s$ 个分区的文件, 假设每个分区被控制的概率为 ρ, 上述方案的检测成功概率为 $P\geqslant1-(1-\rho)^{sz\cdot\omega}$, 其中, ω 为验证协议中的取样概率.

7　结论

本文提出了一种基于交互式证明系统的 POR 定义及其安全要求. 基于交互式零知识证明, 我们提出了交互式 POR 方案 (IPOR), 该方案满足稳固性和零知识性的安全要求, 同时, 性能分析表明该方案只需要固定大小的负载, 使得计算和通信复杂性实现最小化.

致谢　感谢美国亚利桑那州立大学 (Arizona State University) 的黄迪江副教授和 Stephen S. Yau 教授一起讨论研究方向和证明方法, 感谢北京大学信息学院实习学生刘凯南利用 C++ 语言实现并验证方案性能.

参考文献

1　Juels A, Kaliski-Jr B S. Pors: Proofs of retrievability for large files. In: Proceedings of the 2007 ACM Conference on Computer and Communications Security, CCS 2007. Alexandria: ACM, 2007. 584–597

2　Ateniese G, Burns R C, Curtmola R, et al. Provable data possession at untrusted stores. In: Proceedings of the 2007 ACM Conference on Computer and Communications Security, CCS 2007. Alexandria: ACM, 2007. 598–609

3　Bowers K D, Juels A, Oprea A. Proofs of retrievability: Theory and implementation. In: Proceedings of the 2009 ACM Workshop on Cloud Computing Security, CCSW 2009. Chicago: ACM, 2009. 43–54

4　Dodis Y, Vadhan S P, Wichs D. Proofs of retrievability via hardness amplification. In: Reingold O, ed. Theory of Cryptography, 6th Theory of Cryptography Conference, TCC 2009. Lecture Notes in Computer Science, vol. 5444. San Francisco: Springer-Verlag, 2009. 109–127

5　Wang Q, Wang C, Li J, et al. Enabling public verifiability and data dynamics for storage security in cloud computing. In: Proceedings of the 14th European Symposium on Research in Computer Security, ESORICS 2009. Saint-Malo:

Springer-Verlag, 2009. 355–370
6 Shacham H, Waters B. Compact proofs of retrievability. In: Advances in Cryptology - ASIACRYPT 2008, 14th International Conference on the Theory and Application of Cryptology and Information Security. Melbourne: Springer-Verlag, 2008. 90–107
7 Goldreich O. Foundations of Cryptography: Basic Tools. Cambridge: Cambridge University Press, 2001
8 Erway C, Küpçü A, Papamanthou C, et al. Dynamic provable data possession. In: Proceedings of the 2009 ACM Conference on Computer and Communications Security, CCS 2009. Chicago: ACM, 2009. 213–222
9 Boneh D, Boyen X, Shacham H. Short group signatures. In: Proceedings of CRYPTO 2004, LNCS series. Santa Barbara: Springer-Verlag, 2004. 41–55
10 Bowers K D, Juels A, Oprea A. Hail: A high-availability and integrity layer for cloud storage. In: ACM Conference on Computer and Communications Security, CCS 2009. Chicago: ACM, 2009. 187–198
11 Boneh D, Franklin M. Identity-based encryption from the weil pairing. In: Advances in Cryptology (CRYPTO'2001), vol. 2139 of LNCS. Santa Barbara: Springer-Verlag, 2001. 213–229
12 Schnorr C P. Efficient signature generation by smart cards. J Cryptol, 1991, 4: 161–174
13 Cramer R, Damgård I D, MacKenzie P D. Efficient zero-knowledge proofs of knowledge without intractability assumptions. In: Public Key Cryptography. Melbourne: Springer-Verlag, 2000. 354–373
14 Barreto P S L M, Galbraith S D, O'Eigeartaigh C, et al. Efficient pairing computation on supersingular abelian varieties. Des Codes Cryptogr, 2007, 42: 239–271
15 Beuchat J L, Brisebarre N, Detrey J, et al. Arithmetic operators for pairing-based cryptography. In: Cryptographic Hardware and Embedded Systems - CHES 2007, 9th International Workshop. Vienna: Springer-Verlag, 2007. 239–255
16 Hu H G, Hu L, Feng D G. On a class of pseudorandom sequences from elliptic curves over finite fields. IEEE Trans Inf Theory, 2007, 53: 2598–2605

Zero-knowledge proofs of retrievability

ZHU Yan[1,2*], WANG HuaiXi[3], HU ZeXing[1], Gail-Joon AHN[4] & HU HongXin[4*]

1 *Institute of Computer Science and Technology, Peking University, Beijing* 100871*, China;*
2 *Beijing Key Laboratory of Internet Security Technology, Peking University, Beijing* 100871*, China;*
3 *School of Mathematical Sciences, Peking University, Beijing* 100871*, China;*
4 *School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, Tempe, AZ* 85287*, USA*
*E-mail: yan.zhu@pku.edu.cn; hxhu@asu.edu

Abstract Proof of retrievability (POR) is a technique for ensuring the integrity of data in outsourced storage services. In this paper, we address the construction of POR protocol on the standard model of interactive proof systems. We propose the first interactive POR scheme to prevent the fraudulence of prover and the leakage of verified data. We also give full proofs of soundness and zero-knowledge properties by constructing a polynomial-time rewindable knowledge extractor under the computational Diffie-Hellman assumption. In particular, the verification process of this scheme requires a low, constant amount of overhead, which minimizes communication complexity.

Keywords cryptography, storage security, proofs of retrievability, interactive protocol, zero-knowledge