重要数据完整性分布式检测系统

邵必林[1]，吴书强[1]，刘　江[2]，胡家发[1]
（1. 西安建筑科技大学，陕西 西安 710055；
2. 中国兵器工业第 203 研究所，陕西 西安 710065）

摘　要：针对现有支持隐私保护的批量检测方案均未考虑重要数据安全性的问题，提出了一种重要数据完整性分布式检测系统，该系统通过 AES-128 加密算法对数据进行加密，保证了数据的安全性。在重要数据完整性检测过程中利用双线性映射性质，采用将多个重要数据证据和标签证据加密后聚合，保护用户隐私。基于多个分布式检测代理节点，采用加权最小连接数调度算法动态调整分布式检测代理节点的检测任务，使得分布式检测代理节点的性能与待检测任务数量高度匹配，避免检测堵塞的情况，减少检测任务整体响应时间，增强检测系统的高可用性。理论分析和实验结果表明，该系统在保证安全的前提下具有明显的性能优势，检测效率是现有方法的 8.24 倍。

关键词：AES-128 加密算法；完整性检测；数据安全；聚合检测；隐私保护
中图分类号：TP309　　**文献标志码**：A　　**文章编号**：1008-1194(2018)02-0093-06

Distributed Detection System of Significant Data Integrity

SHAO Bilin[1], WU Shuqiang[1], LIU Jiang[2], HU Jiafa[1]
(1. Xi'an University of Architecture and Technology, Xi'an 710055, China;
2. 203rd Research Institute of Ordnance Industry, Xi'an 710065, China)

Abstract: In view of the fact that the existing batch detection schemes supporting privacy protection do not consider the security of important data, a distributed detection system of significant data integrity was proposed, which encrypted data through aes-128 encryption algorithm to ensure security of significant data. In the process of significant data integrity detection, this scheme utilized bilinear mapping property, and the design adopted to encrypt the data and the label evidence to protect the user's privacy. This scheme based on multiple distributed detection agent node, used the weighted least connection number scheduling algorithm dynamically adjust the detection task of distributed detection agent nodes , made the performance of the distributed detection agent highly matched with the number of tasks to be detected, so it avoided detection jam, reduces the detection task overall response time, enhanced the detection system of high availability. The theoretical analysis and experimental results showed that the system had obvious performance advantages under the premise of ensuring safety, and the efficiency of detection was 8.24 times as that of the existing methods.

Key words: AES-128 encryption algorithm, provable data possession(PDP), data security, aggregation detection, privacy protect

0　引言

数据拥有者将数据上传到云端后便在物理层面上放弃了数据的控制权，数据就有可能被窃取或被篡改。为了消除用户的顾虑，确保数据安全、可靠地存放在远程的云服务器中，研究者们提出了数据完整性检测的概念[1]。该方法在检测远程数据完整性

＊ 收稿日期：2017-11-12
基金项目：国家自然科学基金项目资助(61672416)
作者简介：邵必林(1965—)，男，云南腾冲人，博士，教授，CCF 会员，研究方向：信息安全管控、云计算以及动态存储安全。E-mail：wu_shu_qiang@163.com。

中得到了广泛应用。

当前，为了满足不同的应用场景，研究者们提出了多种云存储数据检测方案[2-5]。按照检测方式的不同，数据完整性检测方案可分为公开检测方案、批量检测方案和支持隐私保护检测方案。

1)公开检测方案。在现实生活中，用户并不具备强大的计算能力。为了减轻用户的计算负担，Wang Q 等人基于 BLS(Boneh, B. Lynn, and H. Shacham)[6]签名技术和 RSA 结构实现了动态数据的公共检测。Zhu 等人提出了基于片段结构、随机抽样和 index-hash 表技术的检测方案，能够支持外包数据的动态操作和实时的异常检测[7]。ZhuoHao 等人利用 RSA 加密算法，实现了公共检测，并支持数据动态更新[8]。

由于引入了第三方检测代理，数据的安全性受到了严重挑战，尤其是对于像科研机构的重要数据，它们既需要被共享，又必须保证数据的安全性。

2)支持隐私保护检测方案。Wang 等人提出了一种使用同态令牌预计算来检测修改块的设计[9]，利用擦除编码技术获取来自不同服务器的数据块。Wang BoYang 首次实现了群组共享数据中用户身份隐私保护的公共检测方案[10]，但此方案计算效率较差。研究者利用 Merkle Hash 树以及双线性对技术，实现了支持动态操作的 PDP(Provable Data Possession)[11]。AnirudhaPratap Singh 优化现有的第三方检测协议，使其能够抵抗恶意发起的替换、回放和伪造攻击，并使用改进的变色龙认证树支持细粒度的数据动态更新[12]。

在保护用户隐私的基础上，研究者们实现了数据完整性批量检测功能。

3)批量检测方案。Wang 等人利用 BLS 的同态验证器和 Merkle Hash Tree(MHT)，设计出支持公共验证和批量检测的机制[13]，该机制还支持数据的动态更新。研究者基于 RB232 树，提出了一种改进的公平协议和支持动态数据可持有性证明方案，能够支持动态数据批量检测[14]。Yang 设计了一个高效的、安全的动态检测协议[15]，该协议支持多用户跨多个云服务器对数据进行动态检测和批量检测。He 在 Yang 的基础上提出了将多个数据证据和数据标签聚合后检测，在隐私保护的基础上实现了批量检测[16]。

现有的支持隐私保护的批量检测方案均未考虑重要数据安全性的问题[17-18]。针对上述不足，本文提出了一种重要数据完整性分布式检测系统。

1 系统模型

1.1 系统检测流程

重要数据完整性分布式检测模型由三方构成。如图 1 所示，其中，用户为云资源的使用者。云存储服务提供商掌握着强大的资源，能够针对用户的不同需求实时、动态、高效地提供强大、性价比高的存储资源；由于用户使用资源有限，分布式检测代理可以以自身强大的检测能力对云存储中的重要数据完整性进行检测。

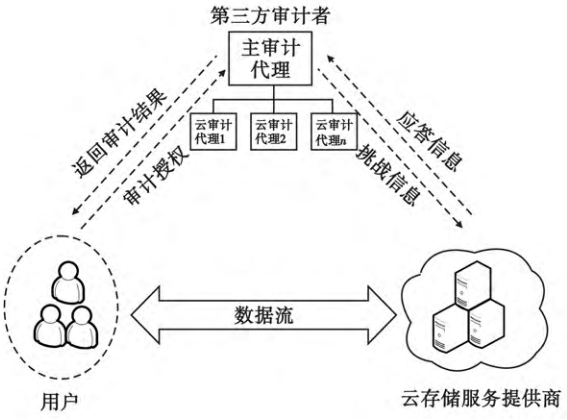

图 1 系统架构

Fig. 1 System architecture

该系统主要由 6 个算法构成，其中 KeyCreat 算法用来生成加密密钥；TagCreate 是认证标签生成算法；ChallRes 是挑战信息生成算法；ProofCreate 是数据持有性的证据生成算法；VerifyProof 是证据检测算法；混合加权轮询调度算法用来动态分配检测任务到分布式检测代理节点。其中 KeyCreat 算法和 TagCreate 算法是在用户端完成，ChallRes 算法和 VerifyProof 算法由检测代理执行，ProofCreate 算法在云服务器中完成。

如图 2 所示，数据完整性云检测流程由 3 个阶段构成，共包含 6 个步骤。

阶段一：初始化阶段

步骤 1 数据拥有者通过执行 KeyCreat 算法，输入安全参数 k，生成密钥对 (sk_t, pk_t) 和加密密钥 sk_h，然后执行 TagCreate 算法，输入密钥 sk_h 和文件 F 生成数据块的签名集合 S 和文件摘要信息 $\boldsymbol{M}_{info}$。然后将 F 和 S 上传给云服务提供商(Cloud Service Provider，CSP)，摘要信息发送给主分布式检测代理

节点。用户通过授权主分布式检测代理节点对远程数据进行完整性验证。

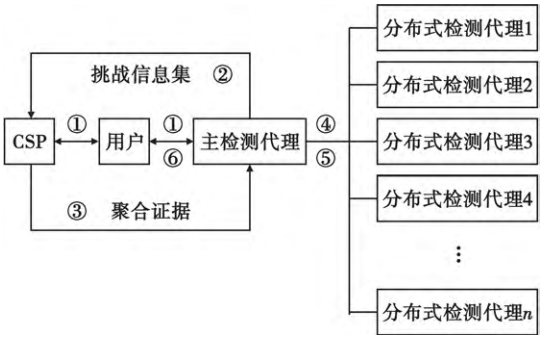

图 2 系统检测流程图

Fig. 2 System audit flow chart

阶段二:证据生成阶段

步骤 2 分布式检测代理节点通过执行 ChallRes 算法,输入 M_{info},输出挑战集 C_i。主检测代理节点将挑战信息集 $C=\{C_1,C_2,\cdots,C_i\}$ 发给 CSP。

步骤 3 CSP 执行 ProofCreate 算法,输入为 F、S 和 C,输出为数据证据 DP 和标签证据 TP。CSP 利用双线性对对 DP 和 TP 加密。然后将加密后的 DP 和 TP 聚合。

阶段三:确认检测阶段

步骤 4 CSP 将聚合后的证据发送给主检测代理。主检测代理将接收到的批量聚合证据通过混合加权轮询调度算法动态分配给分布式检测代理节点。

步骤 5 分布式检测代理节点执行 VerifyProof 算法,验证数据的正确性,将验证的结果传递给主检测代理节点。

步骤 6 最后由主检测代理节点与用户通信,返回检测结果。

1.2 AES-128 加密流程

AES-128 算法输入密钥被扩展成由 44 个 32 位字所组成的数组 $w[i]$,它的密码由 10 轮组成。如图 3 所示,每轮由 4 个不同的字作为该轮的轮密钥。每一轮由 4 个不同的阶段组成,包括字节代替、行移位、列混淆和轮密钥加[17]。

进行加密和解密操作时,算法由轮密钥加开始,接着执行 9 轮迭代运算,每轮都包含所有四个阶段的代替,接着是第 10 轮的 3 个阶段。加密和解密过程的最后一轮只包含 3 个阶段,这是由 AES 的特定结构决定的,也是密码算法可逆性要求的。

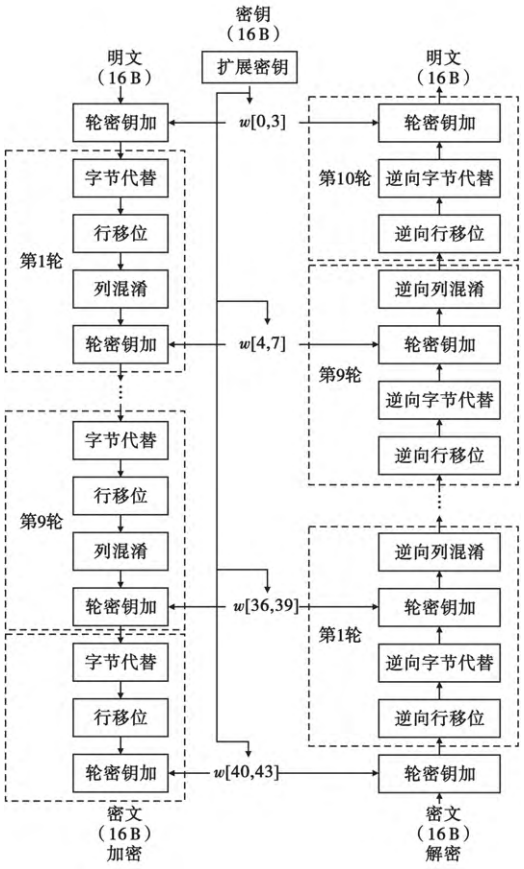

图 3 AES-128 加密解密过程

Fig. 3 AES-128 encryption and decryption process

1.3 聚合检测

为了高效地对重要数据进行完整性检测,检测的方法必须高效。本文采用聚合检测将分布式检测代理节点的若干挑战信息构成集合发送至 CSP,CSP 通过执行 ProofCreate 算法生成证据后,合并证据并将聚合后的结果发送给分布式检测代理节点。聚合检测由初始化阶段和聚合检测阶段组成。

阶段一:初始化阶段

重要数据拥有者通过执行 KeyCreat 算法生成私钥公钥对 (sk, pk),并随机选取 $\text{sk}\in Z_p$ 作为私钥,然后得出对应的公钥 $\text{pk}=g_2^{\text{sk}}$,通过 TagCreat 算法,输入文件 F 和私钥 sk,生成重要数据块 m_i 对应的重要数据标签 $t_i=(h(w_i)\prod_{j=1}^{s}\mu_j^{mij})^{\text{sk}}$ 和重要数据标签集合 $T=\{t_i\}_{i\in[1,n]}$。针对 K 个重要数据持有性检测任务,它们的重要数据文件和重要数据标签可以分别表示为 $F_k=\{m_{k,i}\}_{k\in K,i\in[1,n]}$ 和 $T_k=\{t_{k,i}\}_{k\in K,i\in[1,n]}$。其中 $w_i=\text{fid}\,\|\,i$,i 是块号,fid 是文件标识,s 为每块重要数据元素的数量,$\|$ 是连接操

作符。用户把文件 F 和标签集合 T 发送给远程云端，把摘要信息 F_{info} 传输给主检测代理节点。

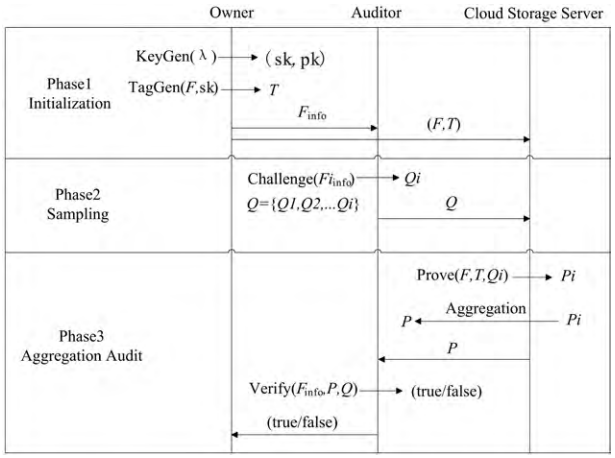

图 4　聚合检测框架

Fig. 4　Aggregated audit framework

阶段二：聚合检测

主检测代理通过 ChallRes 算法为 K 个重要数据完整性检测任务随机选择重要数据块生成挑战子集 $I \in (\text{subset}([1,n]))$，并计算出相应的随机值 v_i，计算 $R_j = g_1^{r_1}$，$R_k = pk_k^{r_2}$，$R_2 = g_1^{r_1 r_2}$，主检测代理将多个挑战消息生成挑战信息集合 Q；主检测代理将挑战信息集合传输给 CSP，发起重要数据持有性挑战。CSP 执行 ProofCreate 算法，输入为文件 F，标签 T 以及挑战信息集合 Q。生成重要数据证据 DP 和标签证据 TP。

$$\text{DP} = e\left(\prod_{j=1}^{s} R_j^{\sum_{i\in I} v_i m_{ij}}, R_1\right), \text{TP} = e\left(\prod_{i\in I} (t_i)^{v_i}, R_2\right)$$

为了节省通信开销将 K 个重要数据证据聚合成 DP_k。K 个标签证据聚合成 TP_k。

$$\text{DP}_k = \prod_{k\in K} e\left(\prod_{j=1}^{s} R_j^{\sum_{i\in I} v_i m_{k,ij}}, R_k\right)$$

$$\text{TP}_k = e\left(\prod_{k\in K}\prod_{i\in I} (t_{k,t})^{v_i}, R_2\right))$$

将 DP_k 和 TP_k 合成证据 P_k。

$$P_k = \frac{\text{TP}_k}{\text{DP}_k}$$

分布式检测代理节点接收来自 CSP 的证据 P_k，并通过执行 $\prod_{k=1}^{s} P_k \stackrel{?}{=} \prod_{k=1}^{k} e\left(\prod_{i\in I} h(w_i)^{v_i r_1 r_2}, \text{pk}_k\right)$ 公式检测结果是否正确。

2　系统性能分析

2.1　安全性分析

1）在挑战信息中添加随机值，保证云存储服务器不能伪造证据，克服了重放攻击。

CSP 希望使用已知或者过期的信息伪造重要数据持有性证据 P，但是在本文方案中挑战信息中包含伴随随机值 $\{i, v_i\}_{i\in I}$，每次产生的随机值是不同的，即使文件被挑战同样的重要数据块，随机值也不一样。因此，CSP 不能依赖历史信息伪造重要数据持有性证据 P，抵挡了恶意云存储服务器的伪造攻击。

2）在重要数据证据检测阶段，分布式检测代理节点不能获取重要数据拥有者的隐私信息。

为了保证用户隐私安全，本系统在证据生成阶段，基于双线性对技术对重要数据证据和标签证据进行加密，然后将它们聚合，最后把聚合后的结果传输给分布式检测代理。分布式检测代理节点使用聚合后的证据检验重要数据的正确性，无需解密。分布式检测代理不能通过聚合后的证据解析出原始重要数据，所以不能获取重要数据。因此，该系统保护了用户隐私。

3）本检测系统可以抵抗替换攻击。

如果 CSP 是不可信的，可能会用正确的重要数据块和标签对 (m_k, t_k) 替换被挑战的重要数据块和标签对 (m_l, t_l)。重要数据证据应为：

$$\text{DP}' = e\left(\prod_{j=1}^{s} R_j^{v_l m_{kj} \sum_{i\in I, i\neq l} v_i m_{ij}}, R_1\right)$$

，标签证据可以表示为：$\text{TP}' = e\left(t_k^{v_l} \prod_{i\in I, i\neq l} (t_i)^{v_i}, R_3\right)$。

根据检测公式，检测结果为：$\text{DP}' \cdot e\left(\prod_{i\in I} h(w_i)^{v_i r_1 r_2}, \text{pk}\right) = e\left[\left[\frac{h(w_l)}{h(w_k)}\right]^{v_l \cdot sk}, R_3\right]\text{TP}'$。

由于散列函数 h 具有防碰撞性，即 $h(w_l) \neq h(w_k)$，所以检测结果不能成立。由此可知，如果 CSP 用正确的 (m_k, t_k) 替换被挑战的 (m_l, t_l)，检测的结果是无法通过的。因此，本系统能够抵抗替换攻击。

4）通过执行 AES 加密算法对本地文件 F 进行加密，保证了重要数据的安全性。

AES-128 加密算法的密码由 10 轮组成。攻击者平均需要进行 2^{127} 次尝试，才能获取正确的密钥。相当 1 000 台中国超级计算机并行运行 10 亿年。

2.2　分布式检测性能分析

分布式检测代理结构由主检测代理和 n 个分布式检测代理节点构成，该结构具有可靠性高、计算能力强、负载均衡和避免单点失效的优势。

1）避免检测堵塞。传统的重要数据完整性检测系统[19]采用最短队列任务优先分配算法。该分配算法是将待处理任务总是分配给待处理任务数最少的执行节点，但在真实的重要数据完整性检测环

境中,每个分布式检测代理的性能和每个检测任务的检测时间常存在差异,这就容易导致检测任务负载不均衡。重要数据完整性分布式检测系统采用加权轮询算法对检测任务进行分配,当分布式检测代理出现负载不均衡时,采用加权最小连接数调度算法,对已分配到分布式检测代理的待检测任务执行任务动态取回重分配,避免了检测堵塞情况的发生,平衡了每个分布式检测代理的检测任务负载量。

2) 避免单点故障。传统的 PDP 方案采用单个代理节点对检测任务进行检测。单检测代理容易造成单点故障,降低了系统的可靠性,制约了性能的扩展。重要数据完整性分布式检测系统可以避免由于单节点失效而引起整个系统崩溃的危险,提高了检测系统的稳定性。

3) 计算开销。重要数据完整性分布式检测系统的主要计算开销集中在证据生成阶段和证据检测阶段。群中一次指数运算用 E 表示,M 是群中一次乘法运算,P 是一次双线性对 e 运算。由于 e 的计算开销远高于其他运算,其计算次数决定着重要数据完整性检测系统的效率。在 Yang[15] 提出的重要数据完整性检测系统中,重要数据块内元素的数量 s 决定了 P 的次数。在计算重要数据证据时,需要聚合 s 个值,其计算开销是 $(Ks+K+1)\times e$,但是本系统采用聚合检测方案[16] 的计算开销为 $(2K+1)\times e$,计算开销明显减小。

3 实验结果及分析

CPU:i5-4200 M 2.50 GHz,内存:4 GB;操作系统:Ubuntu 12.04.3;编程语言:C 语言;编码的实现是基于 PBC 库。文件大小是 4K,重要数据块总数 $n=100\ 000$,其内部元素的数量 s 为 50。

1) 批量检测效率对比。图 5 显示了批量审计不同数量检测任务所花检测时间。分布式检测代理节点数 $n=10$。如图 5 所示,检测任务数量与检测时间呈正比,随着检测任务数量的增加,检测时间也同样增加。但本方法与文献[16]的方法比较,所用时间更少。如图所示,任务数为 500 时,He 所花费时间约为346 s,本方法花费时间约为 42 s,本方法检测效率是 He 的方法的 8.24 倍。由于本系统引用云计算思想,基于多个分布式检测代理并发执行检测任务,大大减少了执行时间。

2) 算法性能对比。图 6 反映的是不同调度算法的效率对比。检测任务调度算法直接影响着检测任务执行的整体响应时间。该实验选取 10 个分布式检测代理节点,分别对 100、200、300、400、500 个检测任务进行动态调度并检测,三种算法的性能对比如图 6。图 6 所示,本系统选用加权最小连接调度算法对重要数据完整性检测任务进行动态调度。该算法相较传统的最短队列优先任务分配算法[19] 和加权轮询算法具有较高的性价比,能很好地实现检测任务的动态分配,有效减少了检测任务整体响应时间。本方案采用的加权最小连接数调度算法充分考虑了现实场景中的问题,在对检测任务进行动态调度时,优先将检测任务分配给负载低的检测节点,从而提高了检测效率。

3) 分布式检测代理数对检测性能的影响。

图 7 展示了分布式检测代理节点个数对重要数据完整性分布式检测系统效率的影响。该实验以 1 000个检测任务为样本,分别选取了 5、10、15、20、25、30 个分布式检测代理节点进行实验。

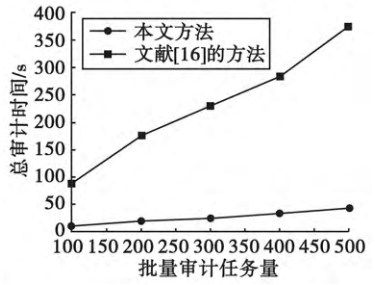

图 5 批量检测时间对比

Fig. 5 Batch audit time comparison

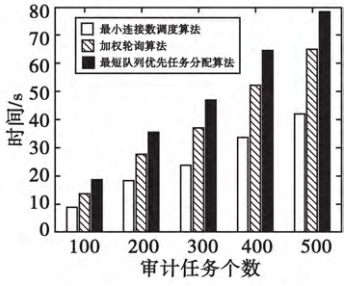

图 6 三种算法效率比较

Fig. 6 Efficiency comparison of the three algorithms

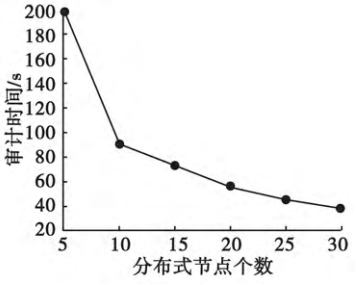

图 7 分布式检测代理节点数对检测性能的影响

Fig. 7 The impact of the number of cloud audit proxy nodes on audit performance

如图 7 所示，当分布式检测代理节点数量增加时，1 000 个检测任务的所用时间呈曲线逐渐减少。说明随着分布式检测代理节点数量的增加系统的性能不断提高，从而使检测重要数据完整性所用时间不断减少。从图中可以看出，检测时间与分布式检测代理节点数并不是呈简单的反比例线性关系，说明分布式检测代理节点的增加，系统性能增强，系统的额外的开销也增大。

4 结论

本文提出了重要数据完整性分布式检测系统，该系统引用云计算思想，基于多个分布式检测代理并发处理检测任务，采用加权最小连接调度算法对检测任务进行动态调度，避免了检测堵塞的情况，减少了检测任务整体响应时间。在重要数据完整性检测过程中通过执行 AES 加密算法对重要文件加密，保证了重要文件的安全性。利用双线性对性质将云服务器中证据加密，将加密后的重要数据证据和标签证据聚合，保护用户隐私不被泄露，实现了高效、安全的批量检测。理论分析和实验结果表明该系统在保证重要数据安全性的前提下，数据完整性检测效率是传统方法的 8.24 倍。

参 考 文 献：

[1]秦志光，吴世坤，熊虎．云存储服务中重要数据完整性检测方案综述[J]．信息网络安全，2014(7)：1671-1122.

[2]DESWARTE Y，QUISQ J，SAIDANE J. A Remoteintegrity checking[J]. Proceedings of IICIS′03，2004，13(9)：1-11.

[3] Wang C，Wang Q，Ren K，et al. Privacy-preserving public auditing for data storage security in cloud computing[J]. IEEE INFOCOM，2010，62(2)：525-533.

[4]WAN Cong，SSM Chow，WANG Q，et al. Privacy-preserving public auditing for secure cloud storage computers[J]. IEEE Transactions on Computers，2013，62(2)：362-375.

[5]Liu C，Chen J，Yang L T，et al. Authorized public auditing of dynamic big data storage on cloud with efficient verifiable fine-grained updates[J]. IEEE Transactions on Parallel & Distributed Systems，2014，25(9)：2234-2244.

[6]WANG Q，Wang C，Li J，et al. Enabling public verifiability and data dynamics for storage security in cloud computing[J]. ComputerSecurity-ESORICS，2009，22(5)：355-370.

[7] YAN Zhu，WANG Huaixi，HU Zexing，et al. Dynamic audit services for integrity detection of outsourced storages in clouds[J]. Proceedings of the 2011 ACM Symposium on Applied Computing，2011，79(6)：1550-1557.

[8] ZHUO Hao，ZHONG Sheng，YU Nenghai. A privacy-preserving remote data integrity checking protocol with data dynamics and public verifiability[J]. IEEE Transactions on Knowledge and Data Engineering，2011，23(9)：1432-1437.

[9] WANG Cong，REN Kui，LOU WenJing，et al. Toward publicly auditable secure cloud data storage services[J]. IEEENetwork，2010，24(4)：19-24.

[10] WANG Boyang. Oruta：privacy-preserving public auditing for shared data in the cloud[J]. IEEE Transactions on Cloud Computing，2012，2(1)：295-302.

[11]赵洋．一种代理远程重要数据完整性检测协议[J]．电子科技大学学报，2016(1)：81-85.

[12]Singh A P，Pasupuleti S K. Optimized public auditing and data dynamics for data storage security in cloud computing[J]. Procedia Computer Science，2016，93：751-759.

[13]Wang Q，Wang C，Li J，et al. Enabling public verifiability and data dynamics for storage security in cloud computing[C]//European Conference on Research in Computer Security，Springer-Verlag，2009：355-370.

[14]REN Zhengwei，WANG Lina，et al. Improved fair and dynamic provable data possession supporting public detection[J]. Wuhan University Journal of Natural Sciences，2013，18(4)：348-354.

[15]YANG K，JIA X. An efficient and secure dynamic auditing protocol for data storage in cloud computing[J]. IEEE Transactions on Parallel & Distributed Systems，2013，24(9)：1717-1726.

[16]何凯．云存储中重要数据完整性的聚合盲检测系统[J]．通信学报，2015，36(10)：119-132.

[17]刘诚毅，桂延宁，杨燕，等．两种算法相结合的基带数据流加密编解码方法[J]．探测与控制学报，2011，33(5)：56-58.

[18]梁小果，相明．基于分布式检测融合技术的水声信号检测系统[J]．探测与控制学报，2001，23(2)：25-30.

[19] 王惠峰，李战怀，张晓，等．支持多代理的云存储数据完整性检测系统[J]．西北工业大学学报，2016，34(2)：343-348.