云存储环境下的支持隐私保护方案①

郭 程[1], 张龙军[2], 许钟华[1]

[1](武警工程大学 研究生管理大队, 西安 710086)
[2](武警工程大学 信息工程系, 西安 710086)

摘 要: 作为云存储技术中的突出问题, 安全始终受到用户的关注. 针对云存储安全中的用户身份隐私保护和数据隐私保护设计了一种安全、高效的云存储方案. 在该方案中构建了基于时间序列的多叉树存储结构(MTTS), 并在该结构基础上设计了一种密钥推导算法, 不仅方便了密钥的生成和管理, 并且节省了存储空间. 通过对方案的安全性分析, 结果表明该方案在确保数据存储安全的基础上也能很好地保护用户身份隐私.

关键词: 云存储架构; 隐私保护; MTTS 存储结构; 密钥生成

Privacy-Preserving Scheme in Cloud Storage Environment

GUO Cheng[1], ZHANG Long-Jun[2], XU Zhong-Hua[1]

[1](Graduate Management Brigade, Engineering University of CAPF, Xi'an 71006, China)
[2](Department of Information Engineering, Engineering University of CAPF, Xi'an 71006, China)

Abstract: As the key problem in cloud storage, security is always drawing the users' attention. A safe and efficient cloud storage scheme is designed for users' identity and data privacy protection in cloud storage security. In the scheme, a storage structure based on time series is constructed and designs a key-deriving algorithm based on the structure. It can not only be good for the generation and management of the key but also can save the storage space. By analyzing the security of scheme, it indicates the scheme can insure the security of data and can protect the privacy of users' identity.

Key words: cloud storage architecture; privacy protection; MTTS storage structure; key generation

随着计算机和互联网的飞速发展, 信息成为政府机关、企业和国家的重要资产, 数据作为信息的载体, 是企业、国家政府机关重要的资源和保护对象. 然而随着数据资源呈指数级的飞速增长, 如何存储庞大的数据文件成了各机关单位、公司企业亟待解决的问题[1-2].

云存储是在云计算基础上延伸、发展出来的. 与云计算系统相比, 云存储可以认为是配置了大容量存储空间的一个云计算系统[3]. 在云存储的应用过程中, 安全是使用者考虑的首要问题. 在云存储系统中, 服务提供商在很多情况下是不可信的, 其可能会因为经济利益将用户存储的数据交予第三方, 造成用户数据的泄漏[4,5]. 因此, 如何确保云存储系统中数据存储安全和保护用户的隐私成了云存储安全研究中的关键问题.

关于云存储中隐私保护技术的研究, 王平建在文献[6]中对市场上的云存储服务, 如亚马逊的 S3 服务、微软 Windows Azure 和谷歌的 Google Storage 服务进行了介绍, 指出这些服务在访问控制方式的安全性和灵活性上都还不够完善, 并指出了改进方向. 针对云存储服务中的数据隐私保护问题, 前人设计了一些保护方案, 黄汝维在文献[7]中构建了一种支持隐私保护的云存储方案, 该方案区分了数据拥有者和数据使用者, 确保了数据在云服务器中的存储安全, 然而该方案中数据拥有者轻易能获得数据使用者的身份信息, 极易造成用户身份隐私泄露; 针对用户身份隐私的保护, 毛剑在文献[8]中设计了一种基于可信第三方的云存储架构, 实现了数据存储和用户个人身份信息的隔离, 有效地保护了用户身份隐私, 但该架构中数据未经加

① 基金项目:中央高校基本科研业务费资助项目(3122013SY39)
收稿时间:2014-05-17;收到修改稿时间:2014-06-16

密就存储于服务器，一旦云服务器被攻破或者数据在传输途中被截取，用户的机密数据文件将完全暴露；为确保数据文件的机密性，苏弘逸在文献[9]中利用秘密共享策略设计了一种云计算环境下的分布式加密方案，针对不可信的云计算服务提供商，该方案能有效降低数据泄露的威胁；陈琳娟在文献[10]中采用基于角色的授权访问机制对用户的访问进行控制，能有效防止他人对数据的非法访问，但其认证流程过于繁琐，难以运用于实际；针对文件[10]方案中的不足，魏娟在文献[11]中设计了基于角色的访问控制管理模型CARBAC，该模型较好地实现了用户对其权限范围内数据的合法访问. 本文针对文献[7]和文献[8]方案中存在的不足，设计了一种支持隐私保护的云存储方案，该方案有效确保了数据在云服务器中的存储安全，不仅保证了非法用户对数据文件的不可访问性，还能保证用户身份信息的隐私性；云可信中心采用了基于时间序列的多叉树存储结构(MTTS)，能对数据名称进行有效管理，加快了检索速度. 基于 MTTS 设计了一种密钥生成算法，方便了密钥管理，节省了存储空间，大大地提高了系统运行效率.

1　支持隐私保护的云存储框架

1.1 云存储框架设计

本文设计的支持隐私保护的云存储框架包含四个实体，分别为：用户(U)、数据拥有者(O)、云存储中心(CSC)、云可信中心(CTC). 该框架的基本结构如图 1 所示.

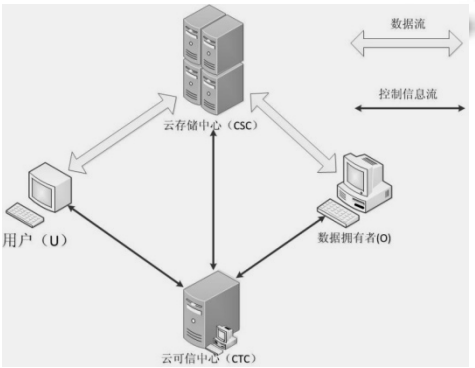

图 1　支持隐私保护的云存储框架

数据拥有者是将存储在本地的数据上传至云存储中心的实体. 数据拥有者需通过云可信中心认证，使用其分发的密钥将数据加密后上传. 另一方面，数据拥有者也能申请访问云服务器中存储的其它数据，在该情况下，数据拥有者即为用户.

用户是申请访问云中存储的数据的实体. 在访问前必须先通过云可信中心的身份认证，使用其分发的密钥对从云存储中心收到的密文解密后进行访问. 另一方面，也可上传本地数据至云存储中心，在该情况下，其身份为数据拥有者.

云存储中心由存储服务器组成，存储了大量的文件数据，其接收数据拥有者上传的文件并为用户提供数据访问服务.

云可信中心由可信服务器组成，具有高度的安全性，黑客极难将其攻破. 主要负责对用户和数据拥有者的身份进行认证，对密钥进行管理等.

从图 1 中可以看出，数据拥有者存储和用户读取数据的数据流只发生在二者和云存储中心之间，可信中心只进行存储授权和反馈存储信息的操作，因此可信中心不会成为数据传输的瓶颈.

1.2 MTTS 存储结构

云可信中心中储存了每一个数据拥有者的身份信息和其上传的文件名称以及文件对应的加密密钥，面对海量的数据，如何使得这些数据有序地存储是一个关键问题. 本文设计了一种基于时间序列的多叉树数据存储结构 MTTS(Multi-Tree based on Time Series)，以每个数据拥有者为根节点，根据数据上传时间的不同建立多叉树型存储结构，较好地解决了云可信中心的数据存储问题. 具体存储结构如图 2 所示.

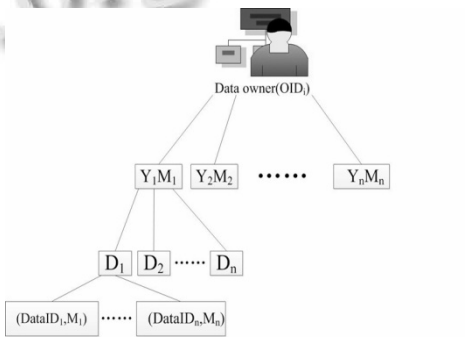

图 2　云可信中心中的 MTTS 存储结构

在该结构中，以每个数据拥有者为根建立多叉树，并将文件名称($DataID_i$)存储在以其拥有者为根节点的多叉树中，分别根据文件上传的年(Y)、月(M)、日(D)建立多级索引.

从图 2 可见，二级节点代表文件上传的年、月，三

级节点代表文件上传的日期, 而叶节点对应文件的名称, 并且在文件名称后加上根据文件存储时间生成的序列码 $M_1,M_2,\ldots\ldots M_n$, 其中序列码生成算法为 $M_f = Hash(\mathrm{FID,Y,M,D})\ (1 \le f \le n)$, 根据 Hash 散列的特性, 给定一个文件名称和上传的时间, 能快速求出其序列码, 而且对于不同的输入值, 所得的序列码也必定不同. 由于该架构根据时间来建立索引, 能确保有序地存储上传的数据名称, 并能快速检索到所需文件. 此外, 由于存储的数据名称后还加上了序列码, 即使同一个数据拥有者上传了两份名称一样的数据, 也能根据序列码的不同将其区分.

1.3 基于 MTTS 结构的密钥生成算法

本方案采用对称加密算法对文件进行加密, 为确保灵活、细粒度的数据访问控制机制, 每个文件都要有不同的密钥, 如何生成和管理这些为数众多的密钥成了本设计中的关键问题. 本文基于 MTTS 结构, 设计了一种高效、灵活的密钥生成算法, 该算法首先为每一个数据拥有者生成一个根密钥, 凡是由该数据拥有者上传的数据文件的加密密钥均由数据拥有者的根密钥进一步生成.

具体如下: 1)云可信中心为每一个数据拥有者生成 128 位的根密钥 k_p, $k_p = r \cdot Hash(\mathrm{OID}_i \| \mathrm{s})$, 其中 r 为由 CTC 随机选取的数, $r \in Z^*$; s 为 CTC 保存的私钥. 可以看到对于每一个数据拥有者, 都有特定的每一个数据拥有者的根密钥, CTC 必须妥善保存. 2)对于名称为 $\mathrm{DataID_i}$ 的数据, CTC 根据数据拥有者的根密钥和该数据在存储结构中的索引计算其密钥 k_i, $k_i = Hash(k_p \| \mathrm{Y}_i, \mathrm{M}_i, \mathrm{D}_i \| \mathrm{M}_i)$.

在本密钥生成算法中, 对于某个数据拥有者上传的每一个数据, CTC 不必进行繁琐的密钥生成运算, 只需根据数据拥有者的根密钥和数据存储的索引推导出加密密钥即可, 大大降低了运算开销; 同时, CTC 只需存储数据拥有者的根密钥, 既便于管理又节省了存储空间.

2 云存储环境下的隐私保护方案

在本文提出的云存储架构基础上分别就数据存储和数据读取两个过程设计了数据隐私保护方案.

2.1 数据存储

当数据拥有者需要将本地的数据上传至云服务器中时, 必须先通过云可信中心认证, 然后使用分配的密钥对数据进行加密并上传至云存储中心. 数据存储的流程如图 3 所示.

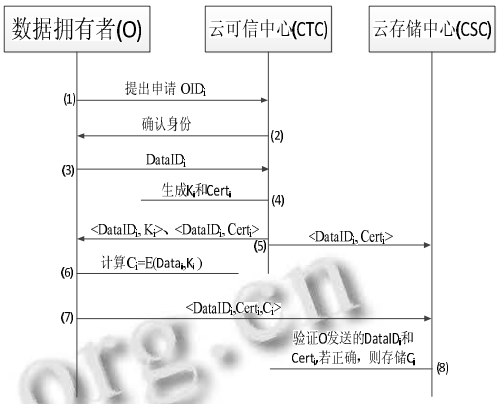

图 3 数据存储流程图

1)数据拥有者(O)向云可信中心(CTC)提出上传数据申请并将自己的身份信息 $\mathrm{OID_i}$ 发送给 CTC.

2)CTC 验证数据拥有者是否合法, 若身份认证通过, 则 CTC 向 O 发送确认信息, 允许 O 上传数据.

3)O 将需要上传的数据的名称 $\mathrm{DataID_i}$ 发送给 CTC.

4)CTC 根据发送的时间将文件名称存储于以 O 为根节点的多叉树中, 查询 O 的根密钥 k_p, 根据密钥生成算法得到数据加密密钥 $\mathrm{K_i}$. 而后, CTC 计算存储认证码, $Cert_i = g \cdot Hash(\mathrm{DataID}_i \| \mathrm{timestamp} \| \mathrm{secret})$ g 由 CTC 随机选取, 使得 $g \in Z^*$, secret 为 CTC 的秘密数据, timestamp 为时间戳. CTC 将数据拥有者身份信息、其根密钥, 文件的名称及存储认证码储存在数据库中.

5)CTC 将 $<DataID_i, K_i>$ 和 $<DataID_i, \mathrm{Cert}_i>$ 发送给数据拥有者, 同时将 $<DataID_i, \mathrm{Cert}_i>$ 发送给云存储中心(CSC).

6)O 收到 CTC 发来的信息后, 用 $\mathrm{K_i}$ 对数据 $\mathrm{Data_i}$ 加密, 得密文 $C_i = E(\mathrm{Data}_i, \mathrm{K}_i)$.

7)O 将 $<DataID_i, \mathrm{Cert}_i, C_i>$ 发送给 CSC, 由于 O 知道云存储中心的网络地址和端口, 因此该操作可以实现.

8)CSC 收到 O 发来的数据后, 通过比较 O 发来的文件名称和存储认证码与 CTC 发来的是否一致来判断 O 是否通过 CTC 认证. 若文件名称和存储认证码一致, 则 CSC 接收数据 $<DataID_i, C_i>$ 并储存. 至此, 该云存储架构中数据存储的流程完成.

在数据的存储过程中，用户只将身份信息发送给了 CTC，而 CSC 是通过存储认证码来判断数据拥有者身份的合法性，其并不知道数据拥有者的身份信息，因此能确保数据拥有者身份隐私的保密性. 其次，数据是经过加密后传到 CSC 中的，即使 CSC 被攻破或者在数据传输过程中被窃取，得到的也是加密后的数据，非法操作者无法获得明文数据的.

2.2 数据读取

当用户想要访问云端存储的数据时，需要先通过 CTC 认证. 通过认证后，用户才能对数据进行正常访问. 用户的数据读取流程如图 4 所示.

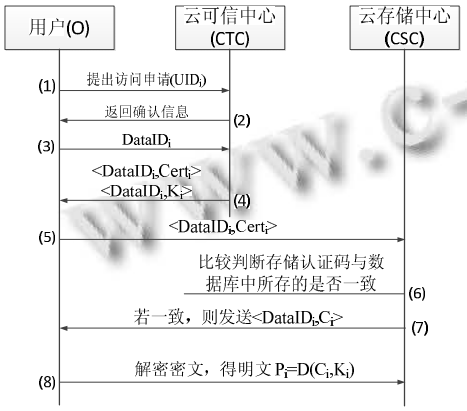

图 4　数据读取流程图

1) 用户向 CTC 提出数据访问申请，将自己的身份信息 UID_i 发送给 CTC.

2) CTC 验证用户的身份信息，若合法，则向其发送确认信息.

3) 用户收到确认信息后，将需要访问的文件的名称 $DataID_i$ 发给 CTC.

4) CTC 收到后通过查询数据库，将该文件对应的存储认证码和密钥发送给用户，即 $< DataID_i, \mathrm{Cert}_i >$，$< DataID_i, \mathrm{K}_i >$.

5) 用户收到 CTC 发来的信息后，保存好 $< DataID_i, \mathrm{K}_i >$，并将 $< DataID_i, \mathrm{Cert}_i >$ 发送给 CSC，由于用户知道 CSC 的网络地址和端口，因此该操作是可以实现.

6) 由于 CSC 的数据库中存有文件的名称和对应的存储认证码，CSC 通过比较用户发来的 $\mathrm{Cert_i}$ 与数据库中存有的是否一致来判断用户是否通过 CTC 认证.

7) 若二者一致，则 CSC 可以确认用户身份合法. CSC 将文件名称 $DataID_i$ 和密文 C_i 发送给用户.

8) 用户使用从 CTC 得到的密钥 K_i 对密文进行解密，得明文 $P_i = D(\mathrm{C}_i, \mathrm{K}_i)$. 至此，数据读取操作结束.

从数据读取的流程可以看出，CSC 是通过比较用户的存储认证码来判断用户的合法性，其并不知道用户的身份信息，因此能很好地防止用户身份隐私泄露；另外，数据都是以密文形式进行传输，即使有非法操作者截取了数据，也只能得到密文信息，无法获得数据的原文.

3 方案安全性分析

3.1 密钥的不可伪造性

在密钥生成算法中，数据拥有者根密钥的生成算法为 $k_p = r \cdot Hash(\mathrm{OID}_i \| \mathrm{s})$，其中 r 为由可信中心随机选取的数，在产生每一个数据拥有者的根密钥时其选取的随机数 r 均不同，因此即使可信中心保存的私钥 s 被非法用户窃取，也无法生成正确的根密钥. 即使非法用户获得了数据拥有者的根密钥，但由于其不知道数据文件名称在云可信中心中的存储位置，仍旧不能得到密文的密钥. 因此，文件的密钥只能由云可信中心生成，非法用户是无法伪造的.

3.2 用户身份的不可伪造性

若存在非法用户，其未通过云可信中心认证，想要通过伪造存储认证码获得云存储中心的访问权限.

假设非法用户 U' 已经知道文件的名称 $\mathrm{DataID_i}$ 和存储认证码的生成算法，存储认证码的生成算法为，$Cert_i = g \cdot Hash(\mathrm{DataID}_i \| \mathrm{timestamp} \| \mathrm{secret})$ 由于云可信中心安全性极高，U' 几乎不可能获得云可信中心的秘密数据 secret，即使 secret 数据被其窃取，由于 g 为随机数，U' 无法获得正确的值，因此也就无法伪造出正确存储认证码，无法通过云存储中心的验证.

3.3 用户身份隐私的保密性

在数据的存储过程中，用户只将身份信息发送给了 CTC，而 CSC 是通过存储认证码来判断数据拥有者身份的合法性，其并不知道数据拥有者的身份信息，因此能确保数据拥有者身份隐私的保密性.

3.4 文件数据存储的安全性

在很多情况下，云存储中心安全防范级别较低，并且其未必可信，因此云存储服务器易于被非法用户攻破. 在本文设计的方案中，假设云存储中心被攻破，非法用户获得了其中存储的文件. 由于文件是由 128 位的密钥 ki 加密，以密文形式存储，根据上文指出的加密密钥的不可伪造性，非法用户是无法获得密钥对

密文解密的, 因此也就保证了数据存储的安全性.

3.5 与其它相关方案的比较

图 5 直观地展示了与其它类似方案相比, 本文所设计的方案在密钥生成的高效性、用户身份隐私的保密性和文件数据存储的安全性这三个方面的安全性. 图 5 中打勾表示方案能满足这一条件, 打叉表示方案不能满足这一条件.

方案	密钥生成的高效性	用户身份隐私的保密性	数据存储的安全性
文献[5]	×	×	✓
文献[7]	✓	×	✓
文献[8]	✓	✓	×
本文方案	✓	✓	✓

图 5 本文方案与同类方案安全性比较

4 结语

本文设计的支持隐私保护的云存储方案具有存储效率高、安全性强的特征, 其不仅能方便地存储、读取数据, 并且确保了云服务器中数据存储的安全性和用户身份的匿名性. 但本文设计的方案无法抵抗非法用户对传输线路的攻击, 一旦非法用户通过窃听或者重放攻击获得云可信中心传给数据拥有者的文件密钥, 云服务器中存储的密文数据便毫无安全可言. 在下一步的工作中, 应着重解决信息在传输过程中的安全保护问题.

参考文献

1 Ma WJ, Wu HJ, Liu P. Architecture and reliability of cloud storage system MassCloud. Journal of Hohai University: Natural Sciences, 2011, 39(3): 348–354.

2 Brian H, Brunschwiler T, Dill H, et al. Cloud computing. Communications of the ACM, 2008, 51(7): 9–11.

3 成春香,张伟,徐涛.一种基于云存储的数据安全与隐私保护系统.北京信息科技大学学报,2013,28(1).

4 Cachin C, Keidar I, Shraer A. Trusting the cloud. ACM SIGACT News, 2009, 40(2): 81–86.

5 林秦颖,桂小林,史德琴,王小平.面向云存储的安全存储策略研究.计算机研究与发展,2011,48.

6 王平建,荆继武,王琼霄,王展.云存储中的访问控制技术研究.第 26 次全国计算机安全学术交流会论文集.2011,9.

7 黄汝维,桂小林,余思,张进.支持隐私保护的云存储框架设计.西安交通大学学报,2011,45(10).

8 毛剑,李坤,徐先栋.云计算环境下隐私保护方案.清华大学学报(自然科学版),2011,51(10).

9 苏弘逸.云计算数据隐私保护方法的研究[学位论文].南京:南京邮电大学,2012.

10 陈琳娟.云计算数据隐私保护研究[学位论文].南京:南京邮电大学,2013.

11 魏娟.云计算中基于角色的访问控制管理模型研究[学位论文].长沙:湖南大学,2012.