doi :10.3969/j.issn.1671-1122.2014.06.001

云存储服务端数据存储加密机制的设计和实现

吕从东，韩臻，马威

（北京交通大学计算机与信息技术学院，北京 100044）

摘　要：云存储是一种新型的网络存储形式。随着云存储的广泛使用，云存储中的数据安全问题，如数据泄漏、数据篡改，也成了用户广泛关注的问题。云存储可以分为访问层、应用接口层、基础管理层和存储层，云存储安全可以分为访问层安全、应用接口层安全、基础层安全和存储层安全。为保证云存储中服务端数据存储的机密性，文章设计了数据存储加密机制，在基于云桌面的办公系统个人存储的应用环境中，实现了基础管理层和存储层加密机制。基于JAVA、JSP等技术，实现了基础管理层；基于Bash脚本等技术，实现了基础管理层与存储层的接口；基于开源项目TGT实现了存储层数据加密机制，保证服务端存储数据的机密性。

关键词：云存储安全；数据加密；基础管理层；存储层

中图分类号：TP309　　**文献标识码：**A　　**文章编号：**1671-1122（2014）06-0001-05

Design and Implementation of Data Storage Encryption Mechanism in Cloud Storage

LV Cong-dong, HAN Zhen, MA Wei

(*School of Computer and Information Technology, Beijing Jiaotong University, Beijing* 100044,*China*)

Abstract: Cloud storage is a new form of network storage. With the widespread use of cloud storage, data security in cloud storage, such as data leakage, data tampering, has become widespread concern. Cloud storage can be divided into the access layer (AL), the application interface layer (APIL), the infrastructure management layer (IML) and the storage layer (SL). Security of cloud storage can be divided into security of the access layer (SAL), security of the application interface layer (SAPIL), security of the infrastructure management layer (SIML) and security of the storage layer (SoSL). A data storage encryption mechanism is designed to deal with the confidentiality of the data in cloud storage in this paper. The mechanism is implemented in an application environment of the office system based on cloud desktop. The infrastructure management layer is implemented based on JAVA and JSP. The interface between the infrastructure management layer and the storage layer is implemented based on Bash Script. The data encryption on the sever is implemented based on open source projects TGT.

Key words: cloud storage security; data encryption; the infrastructure management layer; the storage layer

0 引言

云存储是云计算的延伸和发展[1,2]，是指通过集群应用、网格技术或分布式系统等将网络中大量各种不同类型的存储设备通过应用软件结合起来协同工作，共同对外提供数据存储和业务访问功能的系统。

如图1所示，云存储从上往下可以分为四层，分别为访问层、应用接口层、基础管理层和存储层[3]。访问层是云存储的最顶层，其主要服务对象为一般用户，主要包括三方面内容：1）个人空间服务和运营商空间租赁等；2）企事业单位或者服务器实现数据备份、数据归档、集中存储和远程共享等；3）视频监控、IPTV（Interactive Personality TV：个性化互动电视）等系统的集中存储、网站大容量在线存储等。应用接口层是云存储的第二层，其主要服务对象为二次应用开发的用户。应

收稿日期：2014-02-28

基金项目：教育部创新团队发展计划 [IRT201206]

作者简介：吕从东（1987-），男，江苏，博士，主要研究方向：信息安全，云安全；韩臻（1962-），男，浙江，教授，博士生导师，主要研究方向：信息安全、可信计算、计算机应用；马威（1985-），男，河南，博士，主要研究方向：信息安全，无干扰模型。

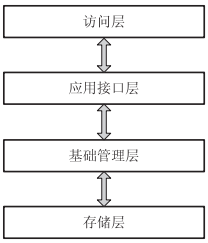

图1 云存储架构

用接口层可以分为两层，上层主要包含网络接入、用户认证和授权管理等，下层主要包含公用应用程序编程接口 API（Application Programming Interface）、应用软件和网络服务等。第三层为基础管理层，主要包含三方面内容：1）集群系统、分布式文件系统；2）内容分发、重复数据删除和数据压缩；3）数据加密、数据备份和数据容灾。最底层为存储层，其主要面向底层硬件存储服务器，可以分为两层。上层主要包含存储虚拟化、存储集中管理、状态监控和维护升级等，称之为网络管理层；下层主要包含 NAS（Network Attached Storage：网络附加存储）、SAN（Storage Area Network：存储区域网络）等存储设备，称之为硬件存储层。

网络附加存储是一种专门的数据存储技术的名称，它可以直接连接在电脑网络上，对异质网络用户提供了集中式数据访问服务。

网络附加存储是文件级存储，用的是以文件为单位的通信协议，如 NFS(Network File System)、SMB/CIFS (Server Message Block/Common Internet File System) 或者 AFP(Apple Talk File Protocol) 等。网络附加存储设备上有简单的操作系统 (包括文本服务和相关的通信协议)。操作系统只提供数据存储、数据访问和相关的管理功能。

在数据量大的环境中，网络附加存储为用户提供存储空间。网络附加存储可以给系统 (如负载均衡系统、容错电子邮件系统和网络服务系统等) 提供存储服务，从而使系统简单，并且花费更低。网络附加存储可以为网络中的其他服务器提供文件服务，提高了文件的使用率。当服务器被关闭后，用户仍然可以和网络附加存储设备通信，从而避免了因为服务器关闭而无法使用数据的情况。

存储区域网络是一种连接外接存储设备和服务器的架构。人们采用包括光纤通道技术、磁盘阵列、磁带柜、光盘等各种技术进行实现。该架构的特点是，连接到服务器的存储设备，将被操作系统视为直接连接的存储设备。

存储区域网络是块级数据存储，用的是以区块为单位的通信协议，一般是通过 SCSI 再转为光纤通道或者 iSCSI。存储区域网络不提供文件虚拟，只提供块级数据存储操作。

存储区域网络可以提高计算机存储资源的可扩展性和可靠性，降低实施成本，减少管理成本。存储区域网络用于要求可用性、可伸缩性和性能的计算环境中。例如，视频编辑工作组对于数据的传输速率要求非常高，存储区域网络可以很好地满足这方面的性能要求。

存储区域网络可以卸掉主网上大量的数据流量，减免数据拥塞。

网络附加存储与存储区域网络比较如下。

1）相同点

网络附加存储和存储区域网络都依靠于网络。在网络中为用户提供存储服务。与传统的存储不同，二者在可用性和扩展性方面有很大的提高。

2）不同点

首先，使用的通信协议是不同的。网络附加存储是文件级数据存储，使用以文件为单位的通信协议；存储区域网络是块级数据存储，用的是以块为单位的通信协议。

其次，技术上实施不同。网络附加存储通常是一个服务器群：邮件服务器、应用服务器等，存储设备附加于系统之上；存储区域网络大多部署于电子商务应用中，大量的数据备份和其他应用数据在网络中传输。

硬件存储设备可以通过 Internet 小型计算机系统接口 iSCSI（Internet Small Computer System Interface）协议实现。iSCSI 协议 [4] 利用 TCP/IP[5,6] 网络传送本机的 SCSI 协议，把 SCSI 数据块映射成以太网数据包。由于 iSCSI 协议是服务器架构，因此可以实现在线扩容以及动态部署。

iSCSI 又称为 IP–SAN，是一种基于因特网及 SCSI–3 协议下的存储技术，由 IETF 提出，并于 2003 年 2 月 11 日成为正式的标准。

iSCSI 协议本身提供了一些基本安全措施。IPSec（Internet Protocol Security：Internet 协议安全）在 IP 层为 iSCSI 通信两端 iSCSI PDU（Protocol Data Unit：协议数据单元）的传输提供数据完整性、认证和机密保护。带内认证

对始发端（Initiator）和目标端（Target）在登录阶段对对方进行身份认证[7,8]。IPSec 和带内认证提供了基础管理层的安全保护。基于已有商业存储 Amazon S3 的增强系统[9] 和网关，可以保护应用接口层的安全。但基于 iSCSI 的云存储，没有提供对存储服务端用户数据机密性的保护机制。

本文研究的云存储服务端数据存储加密机制主要保护用户数据的机密性。本文的主要工作包括：1）设计了一种对用户透明的云存储数据加密机制，给出了由基础管理层和存储层组成的二层架构；2）基于 JAVA、JSP 等技术，实现了基础管理层；3）基于 Bash 脚本等技术，实现了基础管理层与存储层的接口；4）基于开源项目 TGT 实现了存储层数据加解密机制，保证数据的机密性。测试表明，本文实现的加密机制适用于基于云桌面的办公系统个人存储的应用环境，而且用户无需参与加密过程。

本文第一部分给出了数据存储加密机制的基本架构，第二部分是原型系统实现及测试，第三部分是本文的总结和未来进一步的工作。

1 加密机制基本架构

本文的设计主要针对云存储的基础管理层和存储层，如图 2 所示。基础管理层主要包含存储虚拟化、存储集中管理、状态监控和维护升级等，实现对存储层硬件服务器的管理；存储层主要包含硬件服务器、操作系统及软件。可信第三方负责发放和存储密钥。

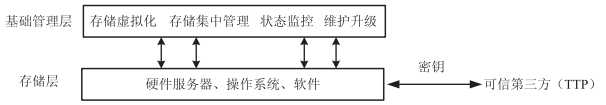

图2 总体架构

1.1 基础管理层

基础管理层面对的对象是存储系统的管理角色，其主要功能模块如图 3 所示，管理角色对整个存储进行管理。

存储虚拟化将硬件存储资源进行抽象化表现，将多台物理存储服务器虚拟成一个存储池，在管理和建立存储时不用考虑存储实际所建的物理机器。存储集中管理对存储进行集中管理，包括存储的添加、删除和修改。状态监控对物理存储进行状态监控，包括 CPU 使用率，设置 CPU 使用率的正常值范围，超出该范围给出警告；内存空闲率，设置内存空闲了的正常值的范围，超出该范围，给出警告；其他相关参数的监控。维护升级对物理存储的软件进行升级，保证物理存储的可用性和安全性，并回收系统的垃圾存储。

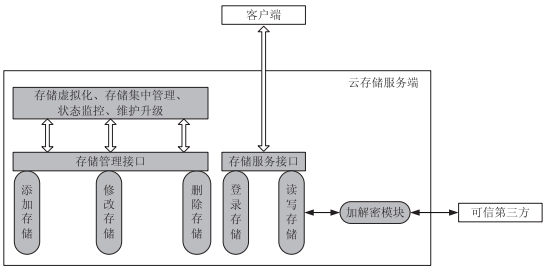

图3 基础管理层功能模块示意图

存储管理接口将底层的添加存储、修改存储和删除存储等功能通过接口提供给存储管理层。

1.2 存储层

存储层主要是硬件服务器、操作系统和软件，它们组成了云存储的服务端。如图 3 所示，它包括存储管理接口、添加存储、修改存储和删除存储，存储管理接口为基础管理层提供添加、修改和删除存储的接口；它还包括存储服务接口、登录存储模块、读写模块和加解密模块，存储服务接口直接为客户端提供登录存储和读写存储的接口；读写存储调用加解密模块，保证客户端存储数据的机密性。用户与服务端、服务端与可信第三方交互流程如图 4 所示。

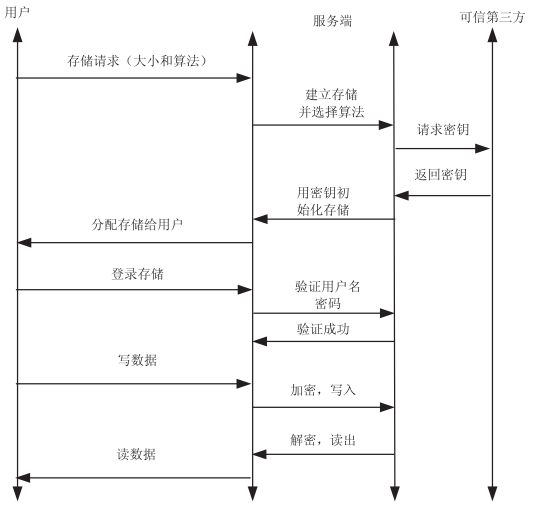

图4 客户端与服务端交互图

1）用户获得存储

（1）首先，用户发起存储请求，可以指定存储大小或系统默认大小；

（2）服务端接收到请求后，建立存储并选择算法；

（3）服务端向可信第三方请求密钥；

（4）可信第三方接收到密钥请求后，返回密钥；

（5）服务端使用密钥初始化存储；

（6）服务端将存储分配给用户；

（7）用户登录存储；

（8）服务端对用户名和密码进行验证，当用户名和密码验证通过后，建立会话。

2）用户读写数据

（1）用户写数据，服务端将数据加密，然后写入存储；

（2）用户读数据，服务端解密数据，然后将数据返回给用户。

2 原型实现

本文基于JAVA、JSP、Bash脚本等，实现了基础管理层基本功能；基于开源项目TGT实现了物理存储服务器加解密功能和可信度量。本文存储的应用环境为云桌面用户存储，用户在存储中存放自己的文件。

2.1 基础管理层实现

基于Web图形界面，实现基础管理层的可视化操作，包括存储虚拟化管理界面，集中存储管理界面，状态监控界面和升级维护界面。管理员可以方便快捷管理存储，不用了解和熟悉底层的管理命令。

管理员通过存储创建界面选择创建存储的大小，可以指定物理服务器，也可以不指定物理服务器，如图5所示。

图5 创建存储界面

管理员可以通过图6所示界面查看物理存储存储的信息。

图6 物理存储存储信息

管理员可以通过物理存储状态监控界面查看物理存储当前的状态，包括内存、CPU使用率等，如图7所示。

图7 物理存储状态监控

2.2 存储层加密机制实现

服务端操作系统为rhel-server-6.4-x86_64，iSCSI软件为iscsitarget-1.4.20.2。如图8所示，主要修改的文件为file-io.c和bloc-io.c，并新建立了AES.h、AES.c、DES.h、DES.c、3DES.h、3DES.c、rc4.h、rc4.c等文件。

我们修改了file-io.c和bloc-io.c的读写部分。file-io.c中为static int fileio_make_request(struct iet_volume *lu, struct tio *tio, int rw)函数，其中rw为读取或者写入符号。在rw为READ时，读出数据后，使用解密函数对数据进行解密；否则，在写入数据前，使用加密函数对数据进行加密。bloc-io.c中static int blockio_make_request(struct iet_volume *volume, struct tio *tio, int rw)，rw为读取或者写入符号。在rw为READ时，读出数据后，使用解密函数对数据进行解密；否则，在写入数据前，使用加密函数对数据进行加密。

AES.h、AES.c、DES.h、DES.c、3DES.h、3DES.c、rc4.h、rc4.c等文件分别实现了AES算法、DES算法、3DES算法和rc4算法。

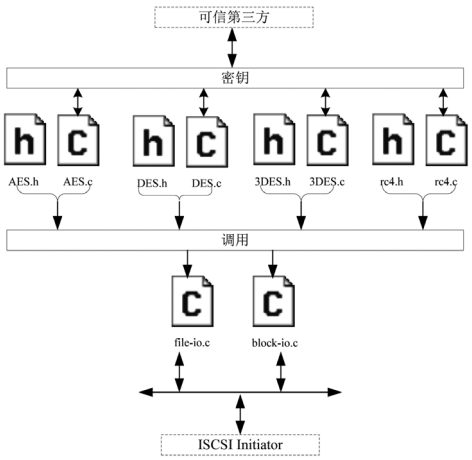

图8 主要添加和修改的代码结构

2.3 数据测试

硬件

处理器：Intel(R) Core(TM) i7-3770 CPU @ 3.40GHz 3.40GHz

安装内存（RAM）：4.00GB

软件

操作系统：Windows 7 专业版

系统类型：32位操作系统

虚拟化软件：Oracle VirtualBox 4.3.2 r90405

客户端操作系统：CentOS-5.5-x86_64 内核版本（2.6.18-371.6.1.e15）；内存：1G；CPU个数：2

服务端操作系统：rhel-server-6.4-x86_64 内核版本（2.6.32-358.e16.i686）；内存：1G；CPU个数：2

客户端软件：iscsi-initiator-utils-6.2.0.837-10.e16.

x86_64.rpm

服务端软件：iscsitarget-1.4.20.2 .tar.gz

Windows 7 作为宿主机系统，使用 VirtualBox 虚拟化软件，建立两个 Linux 虚拟机，一个为 CentOS 系统，安装 iscsi-initiator 软件，作为客户端；另一个为 RedHat 系统，安装修改后 iscsitarget 软件，作为服务端。

由于本文的应用环境为云桌面的用户存储，所以数据测试采用模拟用户写数据的过程，其目的在于测试数据加密后是否会对用户读写文件造成太大延迟，降低系统使用性。测试文件大小分为 1K、10K、100K、256K、512K、1M（如图 9）、10M、100M、256M（如图 10），使用 Bash 脚本，每个文件读写 1000 次计时，然后计算平均时间，作为该大小文件的读写时间。

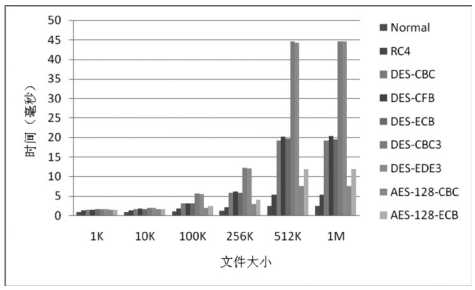

图9 较小文件测试数据图

在文件较小时，加密对于文件写入没有较大影响，而用户的等待时间也不会超过 50ms。

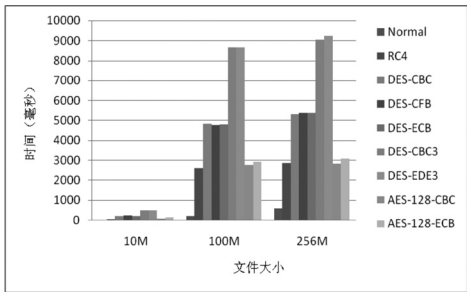

图10 较大数据测试图

在文件相对较大时，如图 10 所示，文件为 256M 时，无加密写入时间不超过 1s，而加密所需最大时间为 9s 多。采用 AES 算法 128 位加密或者 RC4 加密，时间不超过 3s。在用户可接受范围之内。

本文的应用环境是基于云桌面的办公系统的个人存储，办公系统中，文件的读写主要是小数据的频繁读写。图 9 为 1K、10K、100K、256K、512K、1M、10M、100M、256M 等小文件的测试数据，采用 1000 次的读写取平均数据。当文件小于 10M 时，时间差不超过 0.5s；当文件为 256M 时，最大时间差不超过 9s，最小时间差为 2.2s；由于办公系统单个文件为 10M 级已经比较少，所以本文的加密机制适合基于云桌面的办公系统的个人存储使用。

3 结束语

本文主要对云存储的存储层安全进行研究，给出了一种存储层数据加密机制，使用加 / 解密模块加 / 解密服务端（Target）数据，保护用户数据的机密性。针对特定的应用环境，基于 JAVA、JSP 等技术实现了基础管理层，基于 TGT 的开源代码实现数据存储加密机制，并对其性能进行测试。测试表明，该加密机制适合基于云桌面的办公系统中个人存储的应用。

未来的工作主要有以下几个方面：1）对其他层安全的研究，提出总体安全框架或者策略；2）对本文的存储层安全机制性能进行优化，对部分度量服务端（Target）进行探讨；3）如果仅加密或者度量部分服务端（Target），其安全性如何也需要进一步研究。（责编 吴晶）

参考文献

[1] Gkantsidis C, Vytiniotis D, Hodson O, et al. Rhea: automatic filtering for unstructured cloud storage[C].Presented as part of the 10th USENIX Symposium on Networked Systems Design and Implementation. USENIX, 2013: 343-355.

[2] Hao Z, Zhong S, Yu N. A privacy-preserving remote data integrity checking protocol with data dynamics and public verifiability[J]. Knowledge and Data Engineering, IEEE transactions on, 2011, 23(9): 1432-1437.

[3] Grossman R L, Gu Y, Sabala M, et al. Compute and storage clouds using wide area high performance networks[J]. Future Generation Computer Systems, 2009, 25(2): 179-183.

[4] Meth K Z, Satran J. Design of the iSCSI Protocol[C]. Mass Storage Systems and Technologies, 2003.(MSST 2003). Proceedings. 20th IEEE/11th NASA Goddard Conference on. IEEE, 2003: 116-122.

[5] Meth K Z, Satran J. Design of the iSCSI Protocol[C]. Mass Storage Systems and Technologies, 2003.(MSST 2003). Proceedings. 20th IEEE/11th NASA Goddard Conference on. IEEE, 2003: 116-122.

[6] Clark D. The design philosophy of the DARPA Internet protocols[J]. ACM SIGCOMM Computer Communication Review, 1988, 18(4): 106-114.

[7] RFC793, Transmission Control Protocol (TCP), DARPA Internet Program, Protocol Specification, [EB/OL].http://ietf.org/rfc.html. Download on Dec. 18, 2013.

[8] 戴志敏，王倩莉，胡越明，等 . iSCSI 协议研究与实现 [J]. 计算机应用与软件，2005, 22(8): 83-85.

[9] 易非 . iSCSI 协议研究与实现 [D]. 长沙：湖南大学，2004.