学号 2017282110201

姓名 王子豪

海量存储技术论文

多云环境下云存储的安全框架技术

院（系）名 称：计算机学院

专 业 名 称 ：计算机技术

学 生 姓 名 ：王子豪

授 课 教 师 ：何水兵

二〇一七年十二月

摘要

云存储随着云计算的迅速发展也逐渐在我们生活中占据了重要的地位。如何保障和提高云存储的安全性是现在云存储用户重要的需求。本文针对云存储的安全性，研究了云存储的安全策略、架构、面临的问题等。

关键词：云存储；安全

目 录

1 背景

如今，云计算的迅速发展正在改变我们的日常生活。云存储是数据存储和管理云计算系统的重要组成部分，它提供了易于访问数据的 API。云存储服务可以通过集群应用程序、网络技术和分布式文件系统等方式访问数据。本质上，用户可以随时随地地通过网络连接到云并使用任何设备。

在传统的存储服务模式中，企业需要购买服务器，网络通信设备，存储设备和人力资源来建立和维护数据中心，特别是在物联网领域。相比之下，云存储为用户提供了更大的可扩展性和便利的数据存储服务。用户只需根据需要的存储容量付费，而不需要了解如何创建和操作存储服务。目前，许多著名的互联网公司提供了公共的云存储服务，比如 Amazon Cloud Drive, Apple iCloud, Box, Microsoft SkyDrive 和百度网盘等等。

云存储服务市场的发展特别依赖于建立的经济的规模。在“Digital Single Market Strategy”中，欧盟通过促进云服务提供商的数据可移植性和可交换性，在欧洲建立了自由的数据流。研究“SMART 2013/0043-Uptake of Cloud in Europe”表明，云计算的发展可能导致欧洲云存储市场从 2013 年的 95 亿欧元增长到 2020 年的 448 亿欧元(几乎是 2013 年的五倍)。大约有 19%的欧盟企业在 2014 年使用了云计算，主要用于托管电子邮件系统和以电子形式存储文件。此外，对这项研究的进一步估计显示，使用云计算的十家公司中有四分之一报告说，安全漏洞的风险是使用云计算服务的主要限制因素。

但是，用户在使用云存储服务的时候，用户的数据必须在云环境中传输。在云存储的分布式环境中，用户几乎不知道数据的存储位置，这使得用户无法控制数据。因此，用户不得不依靠云服务提供商来保护他们数据的隐私和安全。Twinstrata 在 2012 年的调查显示，只有 20%的人愿意将他们的私人数据存储在云存储中。云存储服务的提供商政面临信任危机，这限制了云应用的发展。

为了解决云存储用户的安全问题，各种云服务提供商已经引入了大量的数据加密服务（DEaaS）。这些服务有两个主要的局限性：（1）它们经常遭受“提供商锁定”的问题，即加密服务受限于特定的云服务提供商或云计算平台；（2）像访问控制策略这样的基本的安全基元经常超出它们的范围，由于用户之间需要共享的密钥，导致这些

服务对用户之间数据共享的支持不够灵活。这些限制迫使数据所有者完全信任其云存储服务提供商，并且在使用云存储过程中共享或者访问云时失去对其数据的控制权。

2 相关工作

在[1]中，作者提出了一种在不可信的移动云环境中使用多云保护用户数据的新方法，这种方法将数据分割成片段，这些片段通过多云被连续加密，压缩和分发，同时在移动设备存储器上保存一部分。在用户设备中保留一个分段将防止任何尝试恢复分发的数据，从而避免可能的未授权用户将所有分段与密钥一起抓取。

在[2]中，作者提出了一个加密存储服务的体系结构。它由四个组件组成：一个服务器，在发送到云之前处理和加密（AES256）数据，一个保存元数据信息的私有云和两个分别存档每个用户文件一般的云。作者假设远程服务器是可信的，没有指定任何有关如何实现这个信任的信息。元数据信息（例如密码，每个文件的密钥，加密的访问路径）安全地存储在私有云中。

在[3]中介绍的 TwinCloud 客户端加密解决方案的中点是云端的安全共享，而不需要明确的密钥管理。为此，他们强调了基于公钥基础设施（PKI）的解决方案问题，即由于从证书颁发机构（CA）获得证书而导致的成本以及维护 PKI 基础设施。

在[4]中，作者提出了使用冗余余数系统（RRNS）并行利用不同云存储提供商的三种不同方法，以实施长期可用性，混淆和加密。他们进行了几个实验，设计了一个测试平台。在这个测试平台中，一个客户端与三个不同的提供商（Google Drive，Copy，Dropbox）交互。他们的解决方案允许用户检索文件即使在之前使用的一个云存储提供商不可用的时候。

在[5]中，作者提出了一个可靠的存储系统 TrustyDrive，可以文件匿名以及用户匿名。系统体系结构由三层组成：使用存储系统作为服务的终端用户对文件进行拆分和编码；调度员，它为终端用户和云服务提供者提供入口；云服务提供商在成本和性能方面提供不同的存储空间。在客户端（终端用户）中，文档被分割成块，并且每个块元数据相关联（用户元数据）。用户将其元数据分成块，发送给调度员，然后计算丢失的信息，以在云服务提供商处存储元数据块。然而，作者没有提供关于最终用户调度员和云服务提供者调度员之间如何进行安全通信的描述。

在[6]中，作者提出了一个架构，将上传的文件的数据分成三份。他们的体系结构包括一个系统数据库和一个用于用户数据切片和合并以及数据加密和解密的中间件。在这项工作中，数据在每个用户文件的三分之一处被静态固定拆分。他们使用系统数据库来存储中间件操作所需要的信息，而不指定其内容（即信息的类型）。

3 云存储

3.1 云存储的模式

云存储可以分为公共云存储、私有云存储和混合云存储。公共云存储设施通常包括低成本存储节点和管理跨节点内容分布的基于对象的存储体。它能够为每个客户提供数据隔离、访问与安全性的服务。私有云存储设施一般是数据中心的专用基础设施，可以完全满足用户所关注的安全性和性能这两点。在其它方面，私有云存储也具有与公共云存储一样的特点，但私有云存储的可扩展性不如公共云存储，成本也比公共云存储稍高。混合云存储一般以传统存储系统或私有云存储为主，公共云存储作为补充。

从云存储的运作模式来看，用户们上传的资料并不能全部都可以密文保存，对于未加密的这部分文件，管理员可以从服务端的平台中直接查看和删除。一旦发生管理员违反职业操守的情况，就有可能造成严重的后果。此外，如果用户的移动终端或客户端用户名和密码泄露或被非法窃取，服务器上用户的隐私数据安全将难以保证。这意味着，用户上传到云端的资料信息越多，个人隐私越多，信息安全隐患越大。

3.2 云存储安全策略

3.2.1 多副本策略

云存储起始于 Google 发布的 Google File System。Google 的分布式文件系统构建在大

量的廉价的机器之上，系统需要忍受硬件的失效，所以硬件失效在分布式文件系统中是允许的。如果某一个硬件机器失效，那么存储在该台机器上的数据是无法访问的，为了防止数据的丢失和为保证数据安全性，可以采取多副本策略。每个数据块在整个集群之上有多个备份，备份的数量可以由用户自己决定。这些备份根据系统的分布情况分布不同的物理位置，防止一个节点失效导致多个备份无法访问。

3.2.2 密钥策略

通常的云存储是通过服务的方式由第三方提供给用户使用，用户不知道自己的数据存放在何处，这个时候对数据是否被别人使用就多了一重疑问，为了消除用户方的顾虑，我们可以通过加密的方式来实现。用户通过一定的加密手段来对数据加密，加密的密钥由用户自身掌控，第三方无法直接访问到用户的数据，访问时用户通过自身的密钥来访问对应的数据块。

3.2.3 数据的差异性保存

云存储出来之前，用户的数据都是存储在自己的私有服务器中，为了数据的安全性，数据的保密等级是必不可少的。这种策略可以运用到云存储上面，将关键的数据由用户自己保存，剩下的通用型的数据存放在云上，这样在私有存储和云存储上找到一个折中，可以使安全性和实用性都得到一个很好的保证。

3.2.4 公共云存储云服务的数据加密机制

针对公共云存储服务，使用混合加密机制来保证公共云存储数据安全是一个合适的选择。混合加密机制包括密钥封装机制（key encapsulation mechanism,KEM）和数据封装机制（DEM）。其中 DEM 采用对称密码算法加密数据量较大的数据文件，以保证加解密运算具有高速度和低复杂度。而 KEM 则以公钥密码算法封装了用户加密数据文件的对称密钥。

（1）传统非对称加密封装

在单用户环境下，数据拥有者使用云存储服务，可以基于传统非对称密码体制实现 KEM，及自己产生一个加解密的密钥对，用公钥实现 KEM，自己保存私钥。当从 CSP 取回加密文件后，用自己保存的私钥解开 KEM，从而能够解密数据。

（2）代理重加密机制

代理重加密是指允许第三方（代理）改变由用户 A 加密的密文，使得用户 B 可以解密，而代理并不知道原来的明文。在公共云存储应用中，如果数据所有者用户 A 想将其加密的文件分享给一个特定的数据使用者用户 B 时，可以使用代理重加密，委托 CSP 将由用户 A 公钥分装的 KEM 转换为用户 B 公钥分装的 KEM，而 CSP 并不能解开 KEM。重加密的操作由 CSP 完成，以节省用户 A 的运算开销。

除此之外，还利用广播加密机制、基于属性的加密机制等。

4 SSME 架构

SSME（Secure Storage in Multi-Cloud Environment）是一个云存储安全架构，解决了有关数据存储和分布式云环境中数据的机密性和完整性的问题。该架构实现了一种主要在客户端使用加密的新颖解决方案，以及一个用户数据分解，传播和检索的分布式中间件。

图 4.1 显示了 SSME 架构的方案。它由一个 JAVA 客户端-服务器应用程序组成，这个应用程序使用无状态的 RESTfull 方法进行通信，客户端与完成大部分计算的中间件进行协作。该 SSME 中间件主要由两个主要部分组成：一个“可信任的云服务”（TCS），实现所有与 SSME 客户的基本接口功能，和一个“服务器”提供所有的中间件文件操作。

符合 SSME 的应用程序可以在两种不同的模式下工作：上传模式和下载模式。

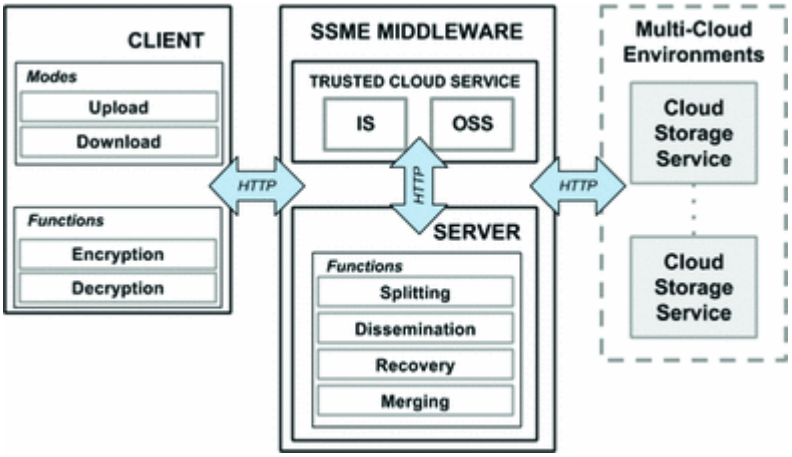

图 4.1

4.1 SSME中间件

组件受信任的云服务（TCS）提供一些显著云计算功能的 SSME 客户端-服务器应用程序。它由两个不同的子服务组成：“身份服务”（IS）和“对象存储服务”（OSS）。顾名思义，IS 通过运行 SSME 客户端-服务器应用程序来识别用户希望采用的自己的可信身份服务。IS 通过利用自己的令牌服务机制，为 SSME 服务请求提供 Cloud 认证和授权。同样，OSS 通过使用 SSME 应用标识用户可能想要采用的自己的可信对象存储服务。OSS 提供 SSME 内部操作期间需要的临时备份。OSS 还可以用来存储加密的 JSON 文件，该文件作为 SSME 应用程序的输出结果。这个输出文件代表位置的标

识从云中检索该特定的文件。每个 SSME 用户都可以拥有自己的可信云服务，例如个人或者公司提供的云服务，用户可以在 SSME 场景中使用这些服务。

SSME 服务器的所有出力都不会在本地存储（硬盘）上留下任何痕迹，因为它只能在易失性存储器中工作。简而言之，服务器执行的功能如下：

（1）从 TCS 上传和下载文件；

（2）碎片的恢复和合并，这些服务是目前使用的多云环境的一部分；

（3）对从客户端接收到的 HTTP 请求的头部中包含的信息进行解密；

（4）创建和加密表示上传模式的输出的 json 加密文件。此文件是从云中恢复文件的唯一方法。

4.2 SSME客户端

客户端工作的时候需要一个名为 json-conf-file 的 JSON 配置文件。为了客户端的服务顺利运行，必须以正确的方式填写 json-conf-file 文件。json-conf-file 文件包含了有关客户端配置的信息，包括：

1.内部设置（包括使用的对称加密密钥）；

2.与服务器通信（包括服务器的公钥）；

3.组成 TCS 的服务；

4.这些云存储服务提供商是用户使用的多云环境的一部分；

在上传或下载模式下，客户端执行的功能如下：

1.加密要发送到多云环境的文件；

2.揭秘想要从多云环境接收的文件；

3.加密从服务器和 TCS 发送的 HTTP 请求头中包含的信息；

4.解密包含在从服务器和 TCS 接收到的 HTTP 请求头中的信息。

在上传模式期间，客户端读取 json-conf-fie 文件内的信息，以指示如何联系服务。客户端根据所报告的信息开始处理。客户端根据 Sect 中描述的规则向服务器发送 HTTP 请求。一旦服务器收到了所有需要的信息，就详细阐述它们并返回给客户端一个 AES256 加密的 JSON 文件，称之为 json-encrypted-file 文件。该 JSON 加密文件包含所有相关信息，以便检索和重建分散在云存储服务中的所有碎片。在没有 json-encrypted-file 文件的情况下，或者如果它被破坏，则无法恢复和重建存储在云中的碎

片。在上传模式中，上传的每个文件将返回一个 JSON 加密文件。json 加密文件的模版结构如图 4.2 所示。

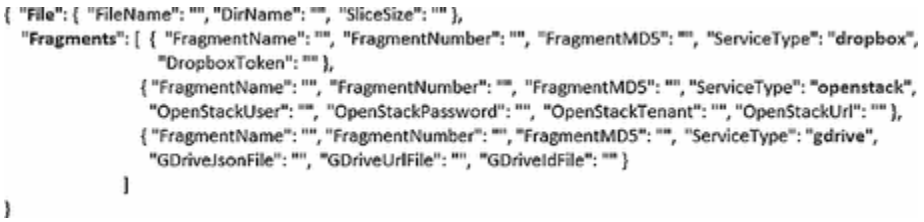

图 4.2

该结构由两个主要部分组成：“文件”和“片段”。“文件”包含原始文件的名称（FileName），碎片存储在 Cloud 存储服务内的容器 / 目录的名称（DirName）以及单个碎片的大小（SliceSize）。“碎片”包含检索所有片段所需的所有信息。这个字段被构造成一个 JSON 向量。在实际的实现中，这个向量的元素可以采取三种不同的类型：dropbox，openstack，gdrive。

在下载模式下，客户端读取 json-encrypted-file 文件中的信息，以遵循 Ariadne 的线程。该 JSON 加密的文件与客户端从云端检索原始文件。客户端通过使用 json-conf-file 文件中指定的对称密钥来解密 json-encrypted-file 文件，并提取包含的信息进行处理。客户端根据规则向服务器发送包含在 json-encrypted-file 文件中的所有信息。服务器需要这些信息来检索和重建用户想要从其自己的多云环境中下载文件的所有片段。这些片段以前在上传模式中分散在云存储服务中。

在下载模式之后，客户端将获得其原始文件（仍然是加密的）。此时，客户端首先删除与刚接收的文件相关的 json 加密文件，然后为了删除多云环境中存储的所有分片，客户端向服务器发送一些删除分片请求。最有客户端将解密原始文件。

4. 3 客户端与服务器之间的通信

客户端与服务器之间的所有 HTTP 通信痘使用所选 IS 提供的令牌服务进行认证。在测试平台中，使用 OpenStack Identity Service V2.0（Keystone）提供的令牌服务。客户端发送的每个 HTTP 请求都会嵌入一些字段，这些字段包含在服务器端执行给定任务所需的信息。所有这些字段都使用 AES256 算法进行对称加密。解密这些字段的密钥也嵌入到每个请求的“key”字段中。客户端通过使用 RSA 非对称算法（密钥为 2048 字节），通过使用服务器的公钥来加密该字段。服务器通过自己的私钥解密“密钥”字段，并使用它解密存储在 HTTP 请求头中的所有其他字段的内容。服务器使用从客户

端接收到的对称密钥加密所有分片，然后将其发送到云存储服务。值得注意的是，服务器公钥是已知的，可以从晚上免费下载。

5 总结

面对纷乱复杂的云存储,安全问题是用户最大的质疑和担心。数据如果放到云端的话，用户就丧失了对数据的绝对控制权，只有 20%的用户愿意把个人数据放在云端，剩下的 80%的用户其实担心的有两个问题，第一个问题，文件会不会丢失？第二个问题，文件会不会泄漏？

5.1 云存储面临的三种安全问题

第一种是文件的可用性，防止这些文件在任何情况下丢失。第二种是文件的机密性，防止文件在任何非授权情况下泄漏。第三种是文件传播安全，因为一旦传播开来会不会有木马、病毒导致用户受到损失。

5.1.1 文件的可用性解决措施

云存储对峙安全问题的解决方案有以下集中，从云盘文件的可用性出发：数据中心内的多份拷贝，同时有专门的备份数据中心。另外防止事故发生需要做的有：1、周期性的数据完整性校验去恢复数据，以及数据中心间的高速连接，存储单元的 RAID 方案，文件分级策略，是根据文件的重要性不同。2、对全局做用户的纵向切分，把用户分成很多的集群，当某个集群出现了不可预知的天灾人祸的时候，也只会影响到这部分用户，不会影响到全局用户。

5.1.2 文件的机密性问题解决措施

文件的机密性是从防止泄漏隐私方面出发：首先在云端把文件分片散列混淆，分布在不同的机房、不同的用户之间，即使用户拿到，也只是片段而已。第二是数据元加密。第三，数据本身是加密的，即使拿到也是密文。用户一人一密，用户端文件操作流水，这样用户就可以看到其他人对自己的文件在什么时间什么地点做了哪些操作。第四，匿名举报机制，如果任何某人的操作存在问题的话，可以立刻举报。

5.1.3 云盘文件的传播安全问题解决措施

云盘文件的传播安全中，可以接入云查杀，保障分享的安全性

5.2 云存储安全面临的挑战

云存储安全技术的研究及安全标准的制定迫在眉睫。如果不能保证安全的云存储，就无法顺畅运营。但是，目前大部分研究只针对云存储的安全性作出提醒与建议，而没有提出具体的防护措施。因此，需要进一步研究云存储安全技术，制定云存储业务安全标准，让云存储业务安全、健康地发展。

参考文献

[1]Alqahtani, H.S., Sant, P.: A multi-cloud approach for secure data storage on smart device. In: 2016 Sixth International Conference on Digital Information and Communication Technology and its Applications (DICTAP), pp. 63－69, July 2016

[2]Balasaraswathi, V.R., Manikandan, S.: Enhanced security for multi-cloud storage using cryptographic data splitting with dynamic approach. In: 2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies, pp. 1190－1194, May 2014

[3]Bicakci, K., Yavuz, D.D., Gurkan, S.: Twincloud: a client-side encryption solution for secure sharing on clouds without explicit key management. CoRR abs/1606.04705 (2016)

[4]Celesti, A., Fazio, M., Villari, M., Puliafito, A.: Adding long-term availability, obfuscation, and encryption to multi-cloud storage systems. J. Netw. Comput. Appl. 59, 208－218 (2016)

[5]Pottier, R., Menaud, J.M.: Trustydrive, a multi-cloud storage service that protects your privacy. In: 2016 IEEE 9th International Conference on Cloud Computing (CLOUD), pp. 937－940, June 2016

[6]Vaidya, M.B., Nehe, S.: Data security using data slicing over storage clouds. In: 2015 International Conference on Information Processing (ICIP), pp. 322－325, December 2015