一种基于虚拟隔离机制的安全私有云存储系统

鲍爱华[1] 袁晓萍[1] 陈 锋[2] 缪嘉嘉[1]
(解放军理工大学指挥信息系统学院 南京 210007)[1] (第二军医大学 上海 200000)[2]

摘 要 云存储技术是云计算领域的重要研究方向,由于存在隐私泄漏和安全问题,公共云存储服务在持有核心数据的组织(如创新型企业、军队)中往往难以得到广泛应用。提出一种基于虚拟隔离机制的安全私有云存储系统VI-PCS:系统将物理存储介质和公共云存储服务虚拟化为虚拟存储能力,通过集中生命周期管理使其为系统提供服务;系统将文件存储过程划分为元数据管理、文件虚拟存储和文件物理存储3个层面,通过文件重命名、IO透明加解密等方式实现数据安全可靠存储;系统提供基于隔离沙箱技术的安全网盘访问方法,通过虚拟化隔离环境来保证数据安全、可控、可用;系统提出基于有序哈希树的文件双向同步方法来实现数据的高效同步,其离线模式也提高了系统的可用性和适应性。结果表明,系统在可靠性、安全性、可伸缩性和适应性方面具有一定的优势。

关键词 云计算,私有云存储,虚拟隔离,沙盒,双向同步
中图法分类号 TP393 **文献标识码** A

Secure Private Cloud Storage System Based on Virtual Isolation Mechanism

BAO Ai-hua[1] YUAN Xiao-ping[1] CHEN Feng[2] MIAO Jia-jia[1]
(College of Command Information System, PLA University of Science and Technology, Nanjing 210007, China)[1]
(The Second Miltary Medical University, Shanghai 200000, China)[2]

Abstract Cloud storage technology is an important research area of cloud computing, because of the loss of privacy and security concerns, public cloud storage services are often difficult to be widely used in organizations which keep the core data, such as the innovative enterprises or the army. VI-PCS, a secure private cloud storage system based on virtual isolation mechanism, was proposed in which physical storage media and public cloud storage services are virtualized as storage capabilities, which are managed through centralized life-cycle, provide storage services for applications in VI-PCS; file storage procedure is divided into three levels (i. e. Meta-data management, virtual storage and physical storage), in which secure, reliable data storage is achieved by file renaming and transparent encryption and decryption technology; a secure net disk based on isolated sandbox is provided as access method, and data security, controllability and availability are achieved in this isolated environment; a file bidirectional synchronization method based on ordered hash tree is proposed, and its offline mode is also helpful to improve system availability and adaptability. The results show that VI-PCS has certain advantages in reliability, security, scalability and adaptability.

Keywords Cloud computing, Private cloud storage, Virtual isolation, Sandbox, Bidirectional synchronization

1 引言

云存储技术是从云计算概念中发展起来的一项关键技术,它通过集群应用、网格技术或分布式文件系统等功能,将网络中异构存储设备通过软件系统集成,共同对外提供数据存储和业务访问功能[1]。从部署和数据管理方式来看,云存储可以分为公共云、私有云和混合云等类型[2]。由于具有低成本、高可靠、可扩展和便于使用等优势,云存储技术得到了业内的广泛关注,但由于用户需要将其重要数据交由云存储服务提供商(Cloud Service Provider, CSP)托管,因此云存储中数据的安全性和可用性问题也引起了用户的担忧。调查显示,超过70%的用户仍然不愿意将关键数据存储于自身控制域之外[3],这对于保有核心数据的组织(如军队、创新型企业等)而言尤为如此。因此,云存储的安全性和可用性成为其能否被进一步普及和接纳的关键问题。

目前,云存储安全性相关的研究集中在数据完整性、访问控制和数据加密等几个方面。针对云存储数据完整性问题,Ateniese等人提出了可证明数据持有(Provable Data Possession, PDP)模型[4,5],由客户端挑战某些数据块,服务器生成完整性证据,由客户端判断结果;惠普公司Shah等人提出了一种由可信第三方进行检查的审计方案[6],但该方案可能造成信息泄漏而带来新的安全隐患;曹夕等人提出了一种云存储系统中数据完整性验证协议CS-DIV[7],在考虑系统资源和带宽的情况下适应性地随机挑战数据块确认完整性;颜湘涛

到稿日期:2013-03-05 返修日期:2013-06-03 本文受江苏省自然科学基金项目(BK2010131)资助。
鲍爱华(1981—),男,博士,讲师,主要研究方向为云计算、语义Web服务,E-mail:nudtbaoah@gmail.com;**袁晓萍** 硕士,助教,主要研究方向为云计算、虚拟化;**陈 锋** 博士,讲师,主要研究方向为信息安全;**缪嘉嘉** 博士,讲师,主要研究方向为大数据处理。

等人[8]提出了一种基于哈希树的云存储完整性检测算法，利用哈希树结构和大数模运算进行数据完整性检测。在安全方面，Sahai[9]等人提出了一种称为属性基加密(Attribute-based Encryption，ABE)的加密访问控制方法，其能够对共享数据进行细粒度控制；Bethencourt 等人[10]提出了使用与访问控制类应用的密文策略属性基加密 CP-ABE 机制；文献[11]提出了一种用于云存储的 CP-ABE 方案，以防止云存储特权用户的内部攻击。

对于公共云存储而言，尽管目前有许多加密访问控制方法来保障数据安全，但由于数据的存储、加解密和访问控制等关键流程均依赖 CSP 的计算环境完成，因此在 CSP 自身可靠性无法得到保证的情况下，这些安全策略仍然无法从根本上打消用户的疑虑。同时，尽管 CSP 的云存储一般都使用了冗余方法来保障数据的可用性[12]，但仍然存在因 CSP 故障或用户自身网络故障而导致的云存储数据可用性问题。所以，相对于个人用户热衷于使用公共云存储，持有核心数据的组织往往更倾向于构建私有云来存储其重要数据。

与公有云相比，私有云的构建也存在一定的挑战。首先，目前构建私有云往往需要新建大量的 IT 软硬件基础设施，现有 IT 资产难以得到有效利用；其次，私有云存储缺乏友好的应用模式，使组织成员将数据存入私有云，这也影响了组织对其数据进行集中管控目标的达成；第三，私有云的安全性不仅体现在其自身上，还应当保障访问云存储终端的安全；最后，私有云更加需要考虑脱机情况下的数据可用性问题，而这在构建私有云的进程中往往容易被忽略。

针对这些问题，本文提出了一种基于虚拟隔离机制的安全私有云存储系统 VI-PCS。VI-PCS 系统采用虚拟化方法将磁盘阵列、分布式文件系统、公共云存储服务(如 SkyDrive，Google Drive)等存储服务统一表示，使其成为 VI-PCS 的数据存储服务器，在实现 VI-PCS 的动态伸缩性的同时降低私有云的构建成本；通过自建 CA 中心来管理用户密钥，将数据加密后存储在自建存储或公共云存储服务平台上，保障数据存储的安全性；采用沙盒技术封装 VI-PCS 的网盘，使 VI-PCS 的数据存储形成闭环，促使用户数据集中存储，实现组织关键数据的安全集中管控；采用云+端的工作模式，用文件同步替换传统网盘直接通过网络进行 IO 操作的方式，从而支持用户离线操作云端数据，提高私有云的可用性。

2 VI-PCS 的系统架构

VI-PCS 系统层次结构如图 1 所示，其结构划分为：

(1)物理存储层。物理存储层是 VI-PCS 的基础组件，它使用异构存储介质保存私有云存储数据。VI-PCS 能够兼容各种存储介质，既包括传统磁盘阵列、NAS、分布式文件系统等用户自行部署的存储介质，也包括基于互联网的公共云存储服务，例如微软 SkyDrive 等。灵活的存储介质使 VI-PCS 能够基于各种应用需求进行扩展，并降低其部署成本和复杂度。

(2)存储虚拟层。存储虚拟层是 VI-PCS 中连接上层应用和下层存储媒介的中间件，由 3 部分组成：公有云存储与私有存储访问接口使系统具备访问各种存储媒介数据的能力，文档加密接口提供了对下层存储数据透明加解密访问能力，物理存储资源访问中间件则对下层存储功能进行虚拟化，使上层应用能够以统一的方式访问下层异构存储媒介。

(3)应用管理层。应用管理层是 VI-PCS 系统的核心，它对用户组织结构、文件元数据进行集中统一管理，并使用数字证书进行用户身份认证和文档数据加密，从而保障私有云存储的安全；此外，应用管理层还包括文档共享、文档版本和用户空间快照管理的功能，并使用影子服务器备份的方法来保障系统元数据服务器自身的可靠性。

(4)访问接口层。访问接口是 VI-PCS 面向最终用户的界面，任何通过身份认证的用户都可以通过这些接口安全访问私有云数据。Web 访问接口面向手机等手持终端用户，提供有限的文档访问功能；共享访问接口供共享授权用户使用，仅能使用特定的共享文件；安全网盘用户则面向私有云客户端用户，它通过形成封闭隔离的环境来使用 VI-PCS 数据，能够提供高速、安全、支持离线访问的私有云使用环境，支持 VI-PCS 的所有功能特性。完善的访问接口使 VI-PCS 能够支持私有云的各种应用用例。

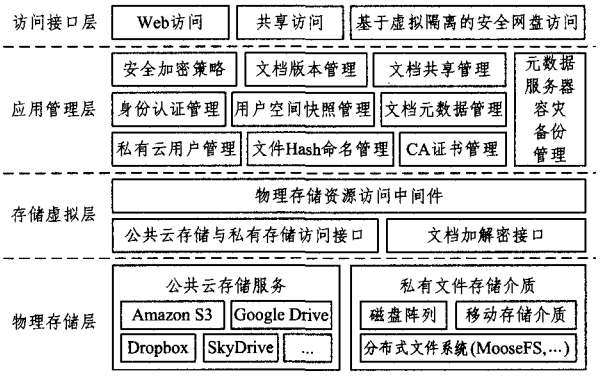

图 1 VI-PCS 私有云存储系统层次结构

VI-PCS 私有云的系统组件及其关联如图 2 所示。总体而言，系统由总控服务器、存储集群和访问终端 3 部分组成。

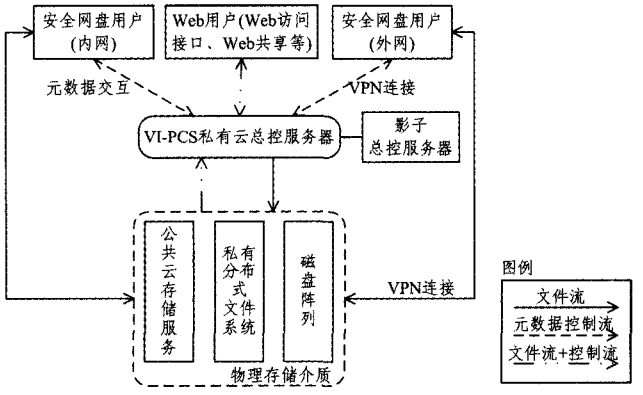

图 2 VI-PCS 的组件及其关联

总控服务器(Master Server，MAS)是 VI-PCS 系统的核心，它对私有云用户信息和文件相关的元数据进行集中统一管理；MAS 由用户自行部署，处于用户的绝对控制中，因此与公有云相比，其安全性可以得到保证；MAS 还包含影子服务器进行热备，一旦出现故障，影子服务器立即替换 MAS 进行服务，从而有效地提高 VI-PCS 的可靠性。

存储集群(Storage Cluster，STC)由异构存储媒介组成，既包括自建的磁盘阵列系统或分布式文件系统，也包括公共云存储服务。数据在 STC 中以密文存储，并采用文件 HASH 值命名，而真实元数据及加密密钥管理均在私有的 MAS 中进行，这有效地保证了 STC 中数据的安全性。同时，由于

STC 能够兼容各类私有或共有存储服务，因此能够根据需要进行快速扩充，达到云存储动态可伸缩的特性。

访问终端(Visit Client Agent，VCA)是用户访问 VI-PCS 数据的统一接口，安全网盘用户能够直接连接 MAS 进行数据操作，而外网安全网盘用户则可以通过 VPN 接入 VI-PCS，Web 用户主要提供给手持终端用户或文件共享用户使用 VI-PCS 中的数据。

从数据流来看，内外网安全网盘与 MAS 进行元数据交互，真实的文件存取由网盘与存储集群直接进行，通过控制流与文件流的分离来降低 MAS 的负载，提高 MAS 的 IO 能力。Web 用户对 VI-PCS 数据的访问需要由 MAS 转发获取。此外，MAS 还直接将其元数据定期备份到 STC 中，通过 STC 的高可靠存储来有效保障 MAS 元数据的可靠性。

3 VI-PCS 系统关键技术

3.1 异构存储能力虚拟化表示及生命周期管理

由于 VI-PCS 私有云能够支持各种异构的自建存储介质或公共云存储服务，只有对其进行虚拟化统一表示，才能被上层应用系统使用。

定义 1(存储能力) 存储能力(Storage Capability，SC)是 VI-PCS 系统对物理存储介质功能的虚拟化表示，形式上 $SC=\langle N,T,URI,C,TBs,FBs,BS\rangle$。其中，$N$ 代表存储介质名称标识；T 代表存储介质类型；URI 代表存储介质资源定位信息，如 IP 地址等地址标识；C 代表访问存储介质的安全认证信息；TBs 表示存储介质总容量；FBs 表示当前剩余容量；BS 表示该存储介质中文件存储最佳的尺寸粒度。

存储能力虚拟化是 VI-PCS 系统的基础，它包含了应用层所关注的功能属性，对于存储介质的访问方法、性能差异等则通过中间件进行封装，使 VI-PCS 的上层应用专注于用户文件数据的安全存储和管理。在 VI-PCS 私有云中，存储介质在虚拟化所得的存储能力的生命周期如图 3 所示。

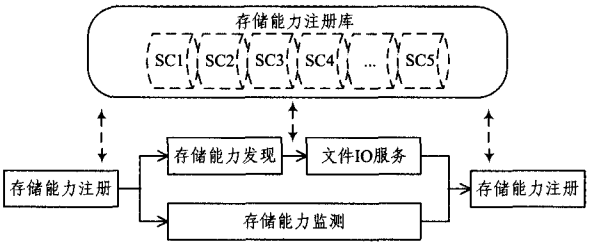

图 3 VI-PCS 存储能力的生命周期

(1)存储能力注册。存储能力注册是物理存储介质进入 VI-PCS 系统并为用户提供服务的基础，存储介质的注册由管理员人工进行，非授权存储介质无法进入 VI-PCS 系统提供服务。在注册成功后，总控服务器将对应虚拟化后的存储能力加入内置的存储能力注册库，供元数据服务器使用。

(2)存储能力监测。存储能力监测持续监测分析存储介质的运行状态，供总控服务器使用。系统需要侦测的指标包括总空间 TBs、剩余空间 FBs、带宽、在线状态等。在线状态可以帮助管理员发现存储介质异常，剩余空间和带宽则有利于系统为用户选择更合适的存储介质。存储能力监测是持续运行的过程，新的存储介质加入后，系统将开启线程进行监测分析，并将结果实时反馈至存储能力注册库中。

(3)存储能力发现。存储能力发现的功能在于为新加入 VI-PCS 系统的文件选择最为合适的存储介质进行存储。在出现新的文件存储请求时，系统主要使用文件尺寸和重要程度这两个因素来选择合适的存储介质，一般而言，重要程度越高的文件越倾向于使用自建存储，而剩余空间较大的存储介质也优先为用户提供服务。

(4)文件 IO。由于 VI-PCS 能够兼容常见的存储能力，因此在文件 IO 阶段，系统对存储能力的访问进行区分处理：对于提供访问 API 的公共云存储服务而言，系统为其构建中间件进行适配，通过 API 封装为系统提供标准访问接口；对于其他自建存储等介质，则在挂载后运行系统提供的 Agent 来提供标准访问接口。所以，在存储介质虚拟化为存储能力后，用户或应用可以以统一的方式使用这些存储能力，而不必关心存储能力的来源、位置或差异特性。

(5)存储能力注销。存储能力注销是存储介质退出 VI-PCS 必须进行的步骤，主要将待注销存储介质上的用户文件数据迁移到其他存储介质上，并更改对应用户文件元数据，确保存储介质的退出不影响用户的文件使用。

在 VI-PCS 系统中，系统整体的存储空间由各类存储能力所组成，因此系统的全局存储空间(Global Total Space，GTS)等同于所有可用存储能力的空间综合，全局可用空间(Global Free Space，GFS)等同于所有可用存储能力的剩余空间综合，即 $GTS=\sum SC.TBs$，$GFS=\sum SC.FBs$。

3.2 VI-PCS 文件安全管理方法

如何对用户文件数据进行有效组织管理，是私有云存储系统需要解决的关键问题。下面，本文首先对其中的关键概念进行定义，然后进一步阐述系统的文件组织结构。

定义 2(文件) 文件(File)是 VI-PCS 系统中存储的数据集合，形式上，文件 $F=\langle FN,D,A,SC\rangle$，其中 FN 表示文件名，D 表示文件数据流，A 表示文件属性集合，SC 表示存储该文件的虚拟存储能力 SC。

定义 3(用户命名空间) 用户命名空间(User Name Space，UNS)是 VI-PCS 中用户文件及其隶属关联的集合，形式上 $UNS=\langle FileSet,RSet\rangle$，其中 $FileSet$ 表示文件集合，$RSet\subset FileSet\times FileSet$ 是文件间隶属关系集合，$\langle f_1,f_2\rangle\in RSet$ 表示 f_2 是目录 f_1 的直属子文件或子目录。

UNS 表示了用户某一时刻存储在 VI-PCS 中所有文件的目录结构，由于用户实际的用户目录结构都是树形结构，因此我们采用符号 f_r 表示 UNS 的根目录。

定义 4(文件路径) 对于命名空间 uns 而言，$\forall f\in uns.FileSet$，若存在序列 $\langle f_1,f_2,\cdots,f_n\rangle$ 满足条件 $f_1=f_r\wedge f_n=f\wedge(\langle f_{i+1},f_i\rangle\in uns.RSet,1\leqslant i\leqslant n-1)$，则称 $s=\backslash f_2.FN\backslash f_3.FN\backslash\cdots\backslash f_n.FN$ 为文件 f 的文件路径，记为 $P(f)$。特别地，$P(f_r)=\backslash$。若无法为文件 f 找到上述匹配序列，则记 $P(f)=\phi$。

由于 UNS 是树形结构，因此 UNS 符合如下性质：

性质 1 若 uns 为用户某一时刻的命名空间，则：(1) $\exists f=f_r,f\in uns.FileSet$；(2) $\forall f\in uns.FileSet,P(f)\neq\phi$；(3)若 $\langle f_1,f_2\rangle\in uns.RSet\wedge\langle f_1,f_3\rangle\in uns.RSet$，那么 $f_2=f_3$；(4) $\forall f\in uns.FileSet,s_1\in P(f)\wedge s_2\in P(f)\wedge s_1\neq s_2$，那么 $P(f)=\phi$。

图 4 给出了 VI-PCS 系统中的文件组织管理结构，我们从 3 个层面来分析 VI-PCS 中的文件组织方式。

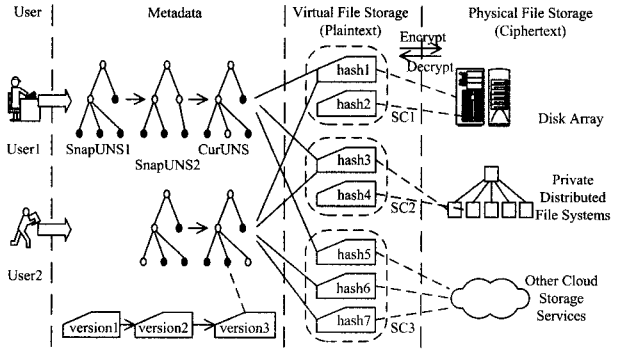

图 4 VI-PCS 的文件组织结构

(1)元数据层面

从元数据(Metadata)层来看，用户的文件以命名空间 UNS 的形态呈现，每个用户可以拥有多个命名空间，表示用户不同时间节点上所有文件及目录结构的快照；这些快照对应的 UNS 具有时间单调递增性，时间最近的 UNS 为当前使用的命名空间。从文件的角度看，每个文件均隶属于一个命名空间，由于文件处于持续的修改进程中，因此每个文件都存在若干历史版本，这些版本同样具有时间单调递增性，时间最近的文件为当前使用的文件。

由于 VI-PCS 系统在元数据层面具有 UNS 快照及文件版本特性，用户可以很方便地进行目录空间的回溯及文件的版本管理。同时，由于为 UNS 创建新的快照仅涉及到元数据拷贝操作，因此并不会显著增大系统的文件存储开销。

(2)虚拟存储层面

在 VI-PCS 系统中，各类物理存储介质都被虚拟化成存储能力注册到系统中提供文件存储服务。因此，从上层应用的角度看，其文件并不是直接存储到物理存储介质上，中间存在着虚拟存储层，其操作对象是虚拟化的存储能力 SC。在文件虚拟存储层面，文件被重命名为其数据 Hash 值，所有的文件均以扁平的方式在 SC 中存储，使命名空间的目录结构及文件名等元数据信息对虚拟存储能力 SC 屏蔽。

这样处理的优点在于：(1)文件以其数据 Hash 值命名，可以避免相同内容文件上传，在提高存储空间利用率的同时避免冗余的文件同步操作；(2)文件重命名后，无论文件存储在自建存储介质或公共云存储服务上，用户文件元数据信息都不会泄漏，这也提高了用户数据的安全性，防止隐私泄漏。

VI-PCS 系统依据如下原则为文件 f 选择合适的虚拟存储能力 SC：(1)若存在某用户的命名空间 uns，$\exists f' \in uns.FileSet, Hash(f'.D)=Hash(f.D)$，则选取 $f'.SC$ 为文件 f 提供存储服务；(2)若上述条件不成立，则选取剩余空间最大的 SC 为文件 f 提供存储服务。

从虚拟存储层面看，文件仍然以明文存储，因为虚拟存储层面仍然处于 VI-PCS 系统的绝对控制之下，数据安全是可以得到保障的。

(3)物理存储层面

在物理存储层面，VI-PCS 将文件存储到真实的物理存储介质中，但为确保数据安全性，VI-PCS 系统对虚拟存储到物理存储之间流动的数据进行透明加解密处理，无论文件最终存储在用户自建存储或公共云存储服务上，文件数据都以密文形式存在，除了用户通过 VI-PCS 系统进行文件存取，其他非授权用户都无法从物理存储层直接获取数据。

综上，在 VI-PCS 系统中，元数据层以命名空间的形式管理用户的所有文件元数据，并能够通过命名空间快照及文件多版本管理实现用户数据回溯；虚拟存储层对文件的组织管理功能在于对下层屏蔽文件元数据信息，使用文件数据的 Hash 值对物理文件重命名，防止用户文件元数据泄露；物理存储层的主要作用在于以密文形式可靠地存储物理文件，避免用户文件数据自身泄露。通过分层协作，VI-PCS 能够安全可靠地为用户提供文件存储服务。

3.3 基于沙箱的安全网盘隔离环境构建技术

通过分层文件安全管理方法，无论用户文件数据最终存储在何种存储介质上，VI-PCS 系统均能够保证用户文件元数据及数据自身的安全，但是如何保证 VI-PCS 私有云终端环境的安全，也是非常关键的问题。如果终端环境安全性无法得到保证，VI-PCS 私有云的安全性仍然存在不足。

本文借鉴沙箱(Sandbox)[13]的思想来增强 VI-PCS 网盘终端环境的安全性。沙箱是信息安全领域的一种虚拟化技术，它通过重定向、访问控制等隔离手段来限制风险进程对操作系统的访问，从而阻止其对系统产生破坏，进而保障系统安全[13]。因此，一般沙箱中运行的都是风险进程，沙箱能够保证风险进程“无痕”地在系统中运行。对于 VI-PCS 网盘终端环境的保护而言，其核心目的在于非法进程无法访问私有云中的数据，而可信进程也无法通过未授权的通道泄漏私有云中的用户数据，因此其保护方法恰恰与沙箱相反。在 VI-PCS 私有云中，沙箱的作用在于限制可信进程及私有云数据的使用、隔离风险环境与风险进程，从而保障私有云终端环境的安全。其原理如图 5 所示。

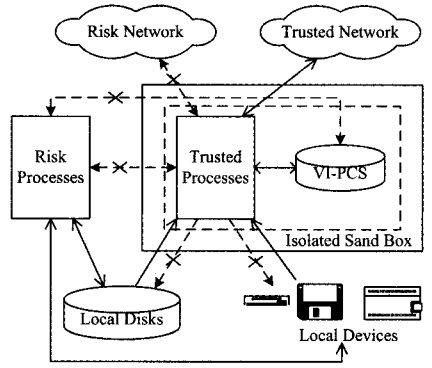

图 5 基于沙箱的 VI-PCS 网盘可信隔离环境

在 VI-PCS 私有云终端环境中，隔离沙箱(Isolated Sandbox，ISB)是一个虚拟化隔离环境，ISB 内包括 VI-PCS 网盘及可信进程(白名单)两个元素，一旦 VI-PCS 私有云用户成功登录，系统将自动为其创建对应的 ISB，并在其中加载与该用户对应的网盘，用户可以运行经过授权的可信进程(Trusted Processes)来操作其存储在私有云中的数据。一旦用户退出系统或出现异常，ISB 将自动关闭并进行环境清理，防止用户数据泄漏。

ISB 对终端环境的隔离主要包括 5 个方面：(1)只有 ISB 中的可信进程能够访问其中的私有云数据，ISB 外的风险进程(Risk Processes)无法穿透 ISB 访问私有云数据；(2)可信进程运行时的内存数据被 ISB 加密保护，无法与风险进程进行交互；(3)可信进程只能以只读方式使用本地磁盘数据，所有文件写操作都通过重定向方式进行隔离；(4)可信进程只能以只读方式使用本地外设(如 USB)，写操作被 ISB 禁止；(5)

可信进程只能访问管理员指定的可信网络，ISB隔离其对风险网络的访问。

通过上述ISB的虚拟化隔离机制，用户登录VI-PCS系统后就如同进入“银行金库”来使用其自身数据，外界无法非法入侵，而用户自身的数据操作也受到一定的限制，无法未经授权泄漏数据。但是，ISB隔离沙箱所提供的是一个终端环境安全增强机制，由于可信进程在ISB中具有较大的权限，如果系统管理员将恶意进程列为可信进程，ISB对私有云数据的保护仍然可能存在风险。“三分技术，七分管理”，ISB对VI-PCS网盘运行环境的增强保护同样需要系统管理员制定合理的可信进程清单，从而在方便私有云用户工作的同时，提高终端环境的安全性。

3.4 基于哈希树的文件双向同步方法

在VI-PCS系统中，安全网盘采用文件同步而非直接网络IO的方式与私有云进行数据交互，同步方式的优点主要有两点：(1)同步方式使VI-PCS安全网盘能够支持用户离线使用，这种“云+端”的使用模式使用户在出差、乘坐飞机等脱机环境下仍然能够安全地操作网盘数据，因而具有更好的适应性；(2)同步方式使用户的文件操作速度不受网络带宽影响，具有更好的用户体验。采用同步方式进行数据交互的关键问题就是，如何快速侦测文件变化，进而在终端和云端之间同步这些文件变化。

VI-PCS终端安全网盘中文件变化侦测的问题，本质上是快速发现命名空间UNS之间差异的过程。一般而言，可以采用遍历UNS的方式进行比较，例如rsync[14]就通过全盘扫描的方式发现变化，但对文件数目较多、变化较频繁的用例而言，全盘扫描效率较低，而且有较大的系统开销。针对这个问题，本文采用一种有序哈希树(Ordered Hash Tree，OHT)对UNS的文件特征进行建模，并通过分析OHT的差异来侦测文件变化。

(1)基于OHT的文件双向同步方法

有序哈希树是与用户命名空间UNS中某个文件(目录)相对应的树，表示该文件(目录)的特征，其节点可以用五元组$\langle f, h, Parent, FirstChild, NextSibling\rangle$表示，其中$f$表示该节点对应的文件信息，$h$表示该节点存储的Hash值，*Parent*、*FirstChild*、*NextSibling*分别代表该节点在OHT中的父节点、头子孙节点和后继兄弟节点。

有序哈希树的构建算法如表1所列，其中，我们记与文件f对应的有序哈希树为OHT(f)，记OHT(f)的根节点为RN。

从OHT构建算法可知，对于UNS中的文件而言，OHT中对应节点的哈希值即为文件数据自身的哈希值(步骤1)；对于目录而言，则首先分别创建其直属子目录或子文件的对应节点(步骤2)，然后将其直属子目录或子文件对应节点的哈希值进行拼装，并将拼装结果的哈希值作为其对应节点的哈希值(步骤3)。由于有序哈希树中子节点按照文件名进行了排序(步骤2.2—2.4)，因此只要某个文件内容或目录发生变化，其对应的OHT节点的哈希值必然发生变化；对于用户的某个命名空间UNS而言，如果其根文件f_r对应的OHT根节点RN的哈希值发生变化，也就意味着UNS中发生了目录或文件内容变化。这也是VI-PCS中基于OHT快速侦测文件变化的原理。

表1 有序哈希树构建算法BUILD_OHT

算法输入：用户命名空间uns，uns中的文件(目录)f

算法输出：有序哈希树OHT(f)

BUILD_OHT(uns,f)

1. 为文件f创建有序哈希树的节点rn=⟨f,NULL,NULL,NULL,NULL⟩，若f是uns中的文件，则令rn.h=Hash(f.D)，进入4。
2. 若f是uns中的目录，对于任意的$f'\in$uns.FileSet，若⟨f,f'⟩$\in$uns.RSet，创建与f'对应的有序哈希树OHT(f')=BUILD_OHT(uns,f')，进行如下处理：
   - 2.1 令节点node=OHT(f').RN，若rn.FirstChild=NULL，则rn.FirstChild=node，进入2；若rn.FirstChild≠NULL，则记chNode=rn.FirstChild；
   - 2.2 若f'.FN<chNode.f.FN，则令rn.FirstChild=node，node.NextSibling=chNode，进入2；
   - 2.3 若chNode.NextSibling=NULL，chNode.NextSibling=node，进入2；否则令nextNode=chNode.NextSibling；
   - 2.4 若f'.FN<nextNode.f.FN，则令chNode.NextSibling=node，node.NextSibling=nextNode；否则令chNode=nextNode，进入2.3。
3. 对于节点rn的所有子节点chNode，依次拼接chNode.h，形成字符串s，rn.h=Hash(s)；
4. 令OHT(f).RN=rn，退出。

VI-PCS系统中安全网盘与私有云端进行文件双向同步的流程如图6所示。总体而言，双向同步分为从终端到云端的上传同步以及从云端到终端的下载同步，在上传同步中，系统定期更新本地安全网盘对应的OHT，并与之前版本的OHT进行比较，进而得到命名空间文件变化，并与云端进行同步，使本地的文件变化持久化保存到VI-PCS云端；与此同时，安全网盘也会定期获取云端对应用户当前UNS的OHT，并与本地当前OHT进行比较，得到下行的UNS文件变化，并以此修改本地安全网盘的文件，进而使本地安全网盘的数据与云端保持一致。

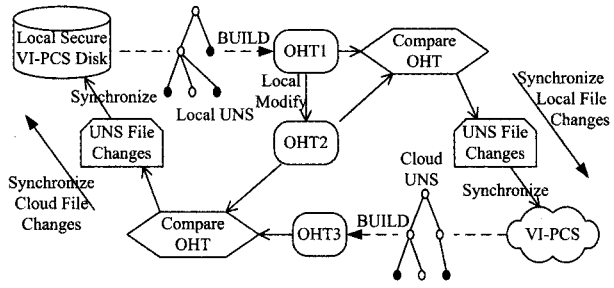

图6 基于OHT的文件双向同步流程

在上述文件双向同步流程中，文件变化是最终进行同步的依据。文件变化的类型较多，例如*CreateFile*、*CreateDirectory*、*Delete*、*Rename*、*Modify*等等，但是为了简化系统，VI-PCS安全网盘将文件变化分为3类，即*Create*、*Delete*和*Update*，其他更为复杂的变化通过这3类变化的组合完成。例如，将文件A重命名为B，则转化为*Delete*(A)、*Create*(B)的组合。由于在VI-PCS的虚拟存储层中文件都以其Hash值进行命名，因此只要文件内容没有发生变化，上述变化组合仅仅会产生元数据修改，不会产生实际的文件数据传输。

在文件上行同步的过程中，安全网盘需要持续检测网盘文件变化，并同步更新OHT。为了提高OHT的更新效率，降低系统开销，VI-PCS安全网盘利用操作系统的Notify机制来得到网盘内的文件变化通知，进而对OHT进行局部更新，避免全局扫描重建OHT。对于*Create*、*Delete*和*Update*这3种文件变化，OHT的更新方式分别为：1)如果发生*Create*操作，假设创建的文件为f，其父目录为f_p，则从OHT中找到与f_p对应的节点N，并将与f对应的节点按照哈希值

和文件名排序插入 N 的子节点中；然后，依次更新从 N 到根节点 RN 路径上所有节点的哈希值。2)如果发生 *Delete* 操作，假设删除的文件为 f，其父目录为 f_p，则从 OHT 中找到与 f_p 对应的节点 N，并从其子节点中删除与 f 对应的节点，然后依次更新从 N 到根节点 RN 路径上所有节点的哈希值。3)如果发生 *Update* 操作，假设更新的文件为 f，OHT 中与 f 对应的节点为 N，则依次更新从 N 到根节点 RN 路径上所有节点的哈希值。

(2)文件同步冲突消解

VI-PCS 系统在多个终端同时与 VI-PCS 进行数据同步时，可能会出现文件变化冲突，这时云端总控服务器会进行冲突检测和冲突消解。VI-PCS 进行冲突消解的原则是尽可能保护私有云用户对其文件的修改、尽可能多地保留用户数据。系统总控服务器中可能出现的冲突包括并发写冲突(写操作包括 *Create*、*Update*)和文件删除冲突两种类型。

(a)并发写冲突。并发写冲突是指在 VI-PCS 云端同步队列中出现来自不同终端针对某用户同一文件的写操作(包括 *Create*、*Update*)。并发写冲突的消解策略是：根据文件变化提交时间，选择最早的写操作进行同步，并向其余终端发送并发写冲突信号，然后撤销其余的文件变化；终端在收到并发写冲突信号后，自动将文件重命名(增加时间及机器标识)，然后再次提交写操作；通过循环进行上述冲突消解，最终将文件变化同步至云端。

(b)文件删除冲突。文件删除冲突是指 VI-PCS 云端同步队列中出现文件删除操作的同时，存在来自其他终端的文件写操作。文件删除冲突的消解策略是：处理文件删除操作，将文件加入回收站，并向其余发起针对该文件写操作的终端发送文件删除冲突信号；终端在收到文件删除冲突信号后，自动将文件重命名(增加时间及机器标识)，再次提交写操作；通过循环进行冲突消解，最终将所有变化同步至云端。

4 VI-PCS 系统特性分析

本文从可靠性、可伸缩性和安全性等 3 个方面对 VI-PCS 系统的特性进行分析。

(1)系统可靠性分析

从系统服务来看，VI-PCS 系统尽可能将文件流与控制流(元数据)进行分离，使总控服务器将主要资源用于为私有云用户提供元数据服务，文件传输则由客户端和存储服务器直接进行，从而降低了总控服务器的负载；同时，VI-PCS 系统为总控服务器建立了影子服务器进行同步元数据备份，一旦总控服务器出现故障，影子节点将自动替换，对外提供私有云元数据服务，从而保障总控服务的可靠性。

从数据存储来看，VI-PCS 系统的物理存储建立在磁盘阵列、分布式文件系统(如 MooseFS 等)或现有公共云存储服务之上，这些物理存储介质自身已经对数据可靠性进行了处理。例如，磁盘阵列可以通过 RAID 对数据进行冗余存储，MooseFS 及常见公共云存储服务采用多份数据冗余存储，这些物理存储介质自身的可靠性足以保障 VI-PCS 私有云数据存储的可靠性。除此以外，VI-PCS 系统会实时侦测虚拟存储能力 SC 的状态，在物理存储介质出现故障时，能够实时屏蔽相关文件访问，避免终端文件访问错误，在系统管理员修复故障后实时恢复数据访问。

(2)系统可伸缩性分析

系统动态可伸缩是云存储的优势所在，用户可以根据实际需求动态调整存储容量，从而节约成本。VI-PCS 私有云的可伸缩性体现在其物理存储介质虚拟化上。在 VI-PCS 系统中，物理存储介质被虚拟化为存储能力 SC，并通过在系统中注册的方式为上层应用提供存储服务。上层应用在需要存储文件时，即时从存储能力注册库中挑选最合适的 SC 使用。因此，如果系统容量不能满足用户需求，系统管理员可以向 VI-PCS 系统中加入新的物理存储介质，实现系统扩容。如果系统容量存在较大闲置，系统管理员也可以通过 SC 生命周期管理中的注销环境移除不需要的物理存储介质，从而为系统减少存储容量，避免资源闲置。因此，VI-PCS 系统的容量可以较为方便地实现伸缩。

(3)系统安全性分析

与公共云存储服务相比，私有云由于由用户自主管理数据，因此数据的安全性也是私有云的优势所在。VI-PCS 系统的安全性主要体现在它的分层数据管理方式上。在虚拟存储层面，系统将文件重命名为其 Hash 值，从而对存储层面屏蔽用户文件元数据；在物理存储层面，系统对数据 IO 进行了透明加解密处理，使数据在物理存储介质中以密文形式存在，避免了用户文件数据的泄漏；同时，由于用户元数据自身由私有云总控服务器进行管理，因此也能够保证用户元数据的安全。综合上述 3 个方面，无论系统管理员选择自建存储介质存储数据，还是使用公共云存储服务作为最终文件存储介质，用户使用 VI-PCS 私有云存储文件都可以保证数据的安全性，可以有效消除用户使用云存储的顾虑。

结束语 本文提出了一种基于虚拟隔离机制的安全私有云存储系统 VI-PCS，对系统的架构进行了设计，阐明了其中的关键技术，并从可靠性、可伸缩性和安全性 3 个方面对 VI-PCS 私有云进行了分析。

VI-PCS 私有云存储系统的主要特点在于：(1)在数据存储层面，系统将各种存储介质(如磁盘阵列、分布式文件系统、公共云存储服务等)虚拟化成虚拟存储能力，并通过有效的存储能力生命周期管理，使其以统一的方式为系统提供文件存储服务；存储介质虚拟化还能够使用户可以灵活地根据成本选择合适的物理存储介质，从而降低私有云构建的成本，同时提高存储容量的可伸缩性。(2)在文件管理层面，系统将整个私有云文件存储过程划分为元数据管理、文件虚拟存储和文件物理存储 3 个层面，分别通过文件重命名、IO 透明加解密等方式，在实现数据安全可靠存储的同时，提高存储空间的利用率。(3)系统采用一种基于隔离沙箱技术的安全网盘终端访问方法，对使用 VI-PCS 私有云的终端环境进行虚拟化隔离，在保证数据安全操控的同时，促进文件数据的集中管理，使数据可控可用。(4)系统提出基于有序哈希树的文件双向同步方法，使终端安全网盘能够与 VI-PCS 云端数据进行高效的数据同步；采用同步的方式与云端交换数据，使私有云终端能够离线使用，“云+端”的模式也提高了系统的可用性和适应性。

本文的重点在于 VI-PCS 系统的架构设计及关键技术实现，后续的工作中，我们将进一步研究用户并发登录副本一致性、私有云用户团队协作等问题，进一步提高系统的效率，并使 VI-PCS 能够更好地与用户既有业务系统更好的集成。

(下转第 216 页)

扑是依据链路发生故障概率大小而进行链路保护的。故免疫原理算法能够对存储资源的利用进行优化。

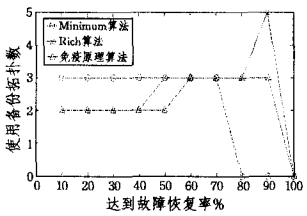

图6 存储资源利用合理性

结束语 本文提出了一种基于免疫原理的多拓扑子层生成算法，它能够有效地保障网络在多故障情况下的业务运行。由于子层中的拓扑是依据链路故障发生概率生成的，因此该算法较其它多拓扑生成算法抵抗多故障的能力更强，且生成过程使用了智能算法，使得生成子拓扑能最大限度满足目标。在后续研究中，将综合考虑网络的链路保护与负载大小，通过调整链路权值实现生成拓扑的负载均衡，进一步提高该方法的网络可生存性。

参 考 文 献

[1] Alaellinoglu C, Jacobson V, Yu H. Towards millisecond IGP convergence [EB/OL]. http://www. nanog. org/meetings/nanog20/abstracts. php? pt = MTA3MiZuYW5vZzlw&nm = nanog20, 2000

[2] 徐明伟，杨芫，李琦. 域内自愈路由研究综述[J]. 电子学报，2009，37(12)：2753-2761

[3] Zhang Xin, Perrig A. Correlation-resilient path selection in multi-path routing[C]//Proceedings of IEEE Globecom. 2010

[4] Bryant S, Shand M, Previdi S. IP fast reroute using notvia addresses[EB/OL]. http://tools. ietf. org/html/draft-ietf-rt-gwg-ipfrr-notvia-addresses-03, 2008

[5] Xu Ming-wei, Yang Yuan, Li Qi. Selecting shorter alternate paths for tunnel-based IP fast reroute[J]. Computer networks, 2012, 56(2): 845-857

[6] Lee S, Yu Yin-zhe, Nelakuditi S, et al. Proactive vs reactive approaches to failure resilient routing[C]//Proceedings of INFOCOM 2004. Hong Kong, IEEE Press, 2004: 176-186

[7] Kvalbein A, Hansen A, Cicic T, et al. Fast recovery from link failures using resilient routing layer[C]//Proceedings of the 10th IEEE symposium on computers and communications. Barcelona, Spain: ISCC Press, 2005: 554-560

[8] Scheffel M C, Gruber C G, Schwabe T, et al. Optimal multi-topology routing for IP resilience[J]. International journal of electronics and communications, 2006(60): 35-39

[9] Kvalbein A, Hansen A, Cicic T, et al. Multiple routing configurations for fast IP network recovery[J]. IEEE/ACM transactions on networking, 2009, 17(2): 473-486

[10] Luan Lin-lin, Wang Zhi-jie, Liu San-ming. Quantum Immune Algorithm for 0/1 Knapsacks Problem[J]. Intelligent Information Managent Systems and Technologies, 2012, 8(1): 117-122

[11] 焦李成，杜海峰，刘芳，等. 免疫优化计算、学习与识别[M]. 北京：科学出版社，2006：92-94

[12] Betker A, Gerlach C, Hulsermann R, et al. Reference transport network scenarios[R]. MultiTeraNet Project, 2004

[13] 黄赫，王晟. 多拓扑路由实现 IP 网络区分服务的优化算法[J]. 计算机应用研究，2010，27(12)：4735-4737

[14] 包学才，戴伏生，韩卫占. 基于拓扑的不相交路径抗毁性评估方法[J]. 系统工程与电子技术，2012，34(1)：168-174

[15] Sterbenz J, Hutchison D, Cetinkaya E, et al. Resilience and survivability in communication networks: strategies, principles, and survey of disciplines[J]. Computer networks, 2010, 54(3): 1245-1265

(上接第 207 页)

参 考 文 献

[1] Zhan Ying, Sun Yong. Cloud storage management technology [C]//2009 Second International Conference on Information and Computing Seience. 2009

[2] Storage networking Industry Association. Cloud storage for cloud computing[EB/OL]. http://www. snia. org

[3] Larry D. Cloud computing hasn't gone fortune 500 yet, but it's coming[EB/OL]. http://blogs. zdnet. corn/BTL/? p=8199

[4] Atenises G, Burns R, Curtmola R, et al. Provable data possession at untrusted stores[C]//CCS'07: Proceedings f the 14th ACM Conference on Computer and Communications Security. New York: ACM Press, 2007: 598-609

[5] Atenises G, Kamara S, Katz J. Proofs of storage from homomorphic identification protocols[C]//ASIACRYPT'09: Proceedings of the 15th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology. Berlin: Springer-Verlag, 2009: 319-333

[6] Shah M A, Baker M, Mogul J C, et al. Auditing to keep online storage services honest[C]//HOTOS'07: Proceedings of the 11th USENIX Workshop on Hot Topics in Operating Systems. Berkeley, CA: USENIX Association, 2007: 1-6

[7] 曹夕，许力，陈兰香. 云存储系统中数据完整性验证协议[J]. 计算机应用，2012，32(1)：8-12

[8] 颜湘涛，李益发. 基于哈希树的云存储完整性检测算法[J]. 计算机科学，2012，39(12)：94-97

[9] Sahai A, Wates B. Fuzzy identify-based encryption [C]// Advances in Cryptology-EUROCRYPT. Berlin: Springer-Verlag, 2005: 457-473

[10] Bethencourt J, Sahai A, Waters B. Ciphertext-policy attribute-based encryptio[C]//Proc of IEEE Sympoisum on Security and privacy. Washington DC: IEEE Computer Society, 2007: 321-334

[11] 刘帆，杨明. 一种用于云存储的密文策略属性基加密方案[J]. 计算机应用研究，2012，29(4)：1452-1456

[12] 刘鹏等. 云计算[M]. 北京：电子工业出版社，2010

[13] Goldberg I, Wagner D, Thomas R, et al. A Secure Environment for Untrusted Helper Applications (Confining the Wily Hacker) [C]//Proceedings of the Sixth USENIX UNIX Security Symposium. 1996

[14] Pierce B C, Vouillon J. What is in Unison[R]. MS-CIS-03-06. Philadelphia, Pennsylvania: Department of Computer and Information Science, University of Pennsylvania, 2004