# 云计算模式内部威胁综述

王国峰[1),2)] 刘川意[2),3)] 潘鹤中[1),2)] 方滨兴[2),3)]

[1)](北京邮电大学计算机学院 北京 100876)

[2)](哈尔滨工业大学(深圳) 广东 深圳 518055)

[3)](东莞电子科技大学电子信息工程研究院 广东 东莞 523000)

**摘 要** 云计算模式的本质特征是数据所有权和管理权的分离,即用户将数据迁移到云上,失去了对数据的直接控制权,需要借助云计算平台实现对数据的管理.而云计算模式的引入带来了很多新的安全问题和挑战,如恶意的云管理员、可利用的安全漏洞和不当的访问接口等.尤其是当云管理员客观上具备偷窥和泄露用户数据和计算资源的能力时,如何保障用户数据的权益,成为一个极具挑战的问题.因此,内部威胁的研究和应对在云计算模式下变得尤为重要,也是近年来学术界和工业界共同关注的热点.为进一步深入研究云计算模式内部威胁问题及其应对,从具体方式方法上作系统的归纳和梳理,并促进国内在该方向上的研究,该文系统分析和总结了云计算模式下内部威胁的类型、特点,并针对真实的云平台构建和实施切实可行的内部威胁攻击实例,展示了云平台客观存在的可利用内部风险以及典型攻击方法.在此基础上,通过梳理相关研究工作,该文总结了三类应对云计算模式内部威胁的主要技术路线,即人员评价与行为分析、云管理权限划分和执行时验证和用户可控的数据加密.针对每一类技术路线,该文深入分析了其主要原理、关键技术、最新进展和产业实用性.最后给出了云计算模式下内部威胁的重要研究点和未来发展方向.

**关键词** 云计算;内部威胁;行为分析;权限管控;数据加密

**中图法分类号** TP311 **DOI 号** 10.11897/SP.J.1016.2017.00296

## Survey on Insider Threats to Cloud Computing

WANG Guo-Feng[1),2)] LIU Chuan-Yi[2),3)] PAN He-Zhong[1),2)] FANG Bin-Xing[2),3)]

[1)](*School of Computer Science, Beijing University of Posts and Telecommunications, Beijing* 100876)

[2)](*Harbin Institute of Technology (Shenzhen), Shenzhen, Guangdong* 518055)

[3)](*Electronic and Information Engineering Institute, Dongguan University of Electronic Science and Technology, Dongguan, Guangdong* 523000)

**Abstract** The division of data ownership and data management is regarded as the key characteristics of cloud computing. Customers outsource their data to the cloud and need to use cloud computing platform for data management, as a result losing direct control over the data. The introduction of cloud computing model has brought some new security issues and challenges, such as malicious cloud administrator, security vulnerabilities and improper access interface. So insider threats become more crucial, especially with the cloud administrators gaining more control on customers' virtual machines and data in reality. How to defend against malicious insiders, especially who have priorities to access or steal customers' data and computation resources, has become a challenging problem as well as a common focus of attention in both academia and industry in recent years. For further study of the insider threats in cloud computing model, making systematic summary from ways and means to deal with, and promoting domestic research in this direction,

收稿日期:2016-05-19;在线出版日期:2016-09-27.本课题得到国家"八六三"高技术研究发展计划项目基金(2015AA016001)、广东省产学研合作项目(2016B090921001)、山东省自主创新及成果转化专项(2014ZZCX03411)、国家自然科学基金(61370068)资助.**王国峰**,男,1988 年生,博士研究生,主要研究方向为云安全.E-mail:wanggf@yunanbao.com.cn.**刘川意**(通信作者),男,1982 年生,副教授,中国计算机学会(CCF)会员,主要研究方向为云计算与云安全、数据安全与数据保护.E-mail:cy-liu04@mails.tsinghua.edu.cn.**潘鹤中**,男,1991 年生,博士研究生,主要研究方向为云安全.**方滨兴**,男,1960 年生,教授,中国工程院院士,主要研究领域为信息与网络安全、内容安全.

this paper firstly summarizes the major types of internal threats in the cloud environment, and takes an experimental approach to demonstrate typical insider vulnerabilities and possible actionable attacks. This paper summarizes and proposes three approaches to deal with insider threats in the cloud, as: user & entity behavior analysis and evaluation, cloud administration priority division and run-time access control, and customer controlled data encryption. For each approach, this paper thoroughly analyzes its technical principles, key technologies, state of the art, as well as practical possibilities in the real world. At last, this paper points out the future research directions and key technologies against insider threats in the cloud.

**Keywords**　cloud computing; Internal threats; behavior analysis; permissions control; data encryption

## 1　引　言

内部威胁是云计算安全面临最严重挑战之一[1]. 2013 年斯诺登事件即内部人员公开内部数据让媒体引起广泛关注,但这只是内部威胁安全的冰山一角. SailPoint 安全公司曾做过一个安全调查,受访者中 20%的人表示只要价钱合适会出卖自己的工作账号和密码;美国计算机安全协会(CSI)和联邦调查局(FBI)在 2008 年的报告中指出内部安全事件所造成的损失明显高于外部安全事件;2015 年普华永道的调查指出中国大陆与香港地区的企业信息安全事件中 50%以上是由内部人员造成的.

学术界对内部威胁越来越重视,网络安全顶级会议 CCS(ACM Conference on Computer and Communications Security)针对如何防护内部威胁专门成立研讨会. CERT(Computer Emergency Response Teams)研究中心 2000 年即开启内部威胁检测项目,根据 CERT 的定义,内部威胁是指一个或多个现在或以前的公司员工、外包商或合作伙伴,具有对网络、系统或数据的访问权限,故意滥用或误用自己的权限损害公司信息或信息系统保密性、完整性、可用性[2-4].

2016 年 2 月,CSA(Cloud Security Alliance)在报告中列举了云计算模式下的十二大风险①,其中恶意内部人员(Malicious Insiders)威胁赫然在列;对比传统模式下应对内部威胁的典型方法[5-8],云计算模式的引入带来了很多新的安全问题和挑战[9-10],亟需找到有效的解决办法,比如说:

(1) 云恶意管理员.

在云环境下恶意管理员(包括云平台管理员、虚拟镜像管理员、系统管理员、应用程序管理员等)可利用自己的特权窃取用户隐私数据,由于本身属于防范边界内受信任的实体,传统的安全策略难以防范,如云平台管理员转储虚拟机内存并分析其中的用户数据[11-12].

(2) 利用云带来的漏洞.

内部人员了解云的组织结构和应用程序特点,可利用其中的漏洞实施恶意攻击. 如部署在云环境下不同区域的多服务器程序同步消息时,一般情况下同步过程比传统的处于同一区域的本地服务器程序要慢,恶意内部人员可利用这一时间差作恶,窃取数据或获取利益[1].

(3) 复杂的数据资源和访问接口.

在云环境下用户和资源的关系是动态变化的,云服务提供商和用户往往并不在相同的安全域中,用户使用的访问终端也是多种多样的,需要动态访问控制. 基于传统的安全认证并不能防止内部人员作恶,如 2.5 节所述恶意管理员可偷窥到用户正在访问的虚拟机的 URL 并可直接利用此 URL 进入虚拟机.

(4) 使用云作恶.

云环境的引入也为内部人员作恶带来了便利,如恶意内部人员利用云服务的计算处理能力来破解密码文件,或利用云相对便宜、轻松配置的特性发动分布式拒绝服务攻击等.

内部威胁是近年来网络安全领域研究的热点,国际上也发表了很多这方面的学术成果,如研究人员[1,13-14]针对新型的云计算模式内部威胁风险及其

① Alliance C S, "CSA's Cloud Computing Top Threats in 2016" Cloud Security Alliance, Top Threats Working Group, February 2016. [Online]. Available: https://downloads.cloudsecurityalliance.org/assets/research/top-threats/Treacherous-12_Cloud-Computing_Top-Threats.pdf

应对展开了研究工作，并取得了一些进展. 当前国内的内部威胁研究落后于国际，应给予充分重视. 目前CERT 安全事件数据库中关于云服务内部攻击的案例并不多见[1]，很多公司即使发生了内部攻击也不愿主动向外界公开. 对于云计算模式内部人员的作恶手段，大部分研究还停留在设想和理论阶段，缺少实际的内部人员攻击案例. 本文针对真实的云平台，构建和实施切实可行的内部威胁攻击实例，从而让云平台内部安全防护更“有的放矢”.

为进一步深入研究云计算模式内部威胁问题及其应对，从具体方式方法上作系统的归纳和梳理，并促进国内在该方向上的研究，本文结合近几年研究成果从人员评价与行为分析、云管理权限划分和执行时验证、用户可控的数据加密这 3 个方面进行综述，并对云模式下内部威胁问题的研究方向进行了展望.

## 2 典型攻击实例

为揭示云计算模式内部威胁问题，本文真实实现了恶意内部人员窃取用户数据的攻击实例. 实例中内部人员可利用自己的特权成功得到用户的隐私数据，例如云管理员可通过修改 Linux 或 Windows 用户虚拟机登录密码侵入虚拟机，制作虚拟机镜像时安装后门程序，利用内存分析工具将虚拟机内存导出并分析，通过截获虚拟机的访问会话、进入虚拟机等手段窃取用户隐私数据，如图 1 所示.

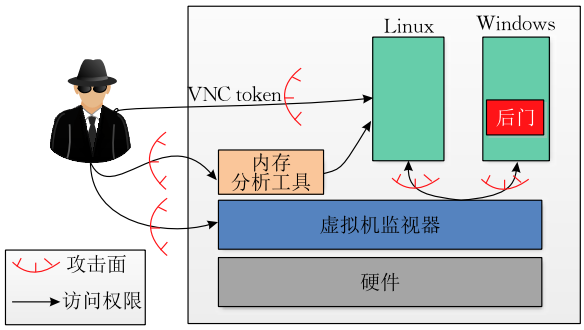



图 1 威胁模型

### 2.1 云管理员删除虚拟机登录密码

用户放在云上的 Linux 虚拟机，其 Root 密码是用户自己设定的，防止别人入侵. 但云管理员可直接对 Linux 虚拟机磁盘文件进行修改绕过检测，登入虚拟机窃取数据. 如图 2 所示，大致过程为：

(1) 在 OpenStack 云平台(也可适用于其它云平台)下，云管理员登录云平台后，获取云主机 ID.

(2) 云管理员登录运行该云主机的物理机，拷贝一个云主机实例(instance)，并将其挂载到另一个操作系统中.

(3) 编辑用户密码文件，将含有登录用户名和密码的一行删除，绕过用户虚拟机的密码登录环节.

(4) 云管理员启动该云主机时，可无需密码直接登录该云主机，对云主机进行操作.

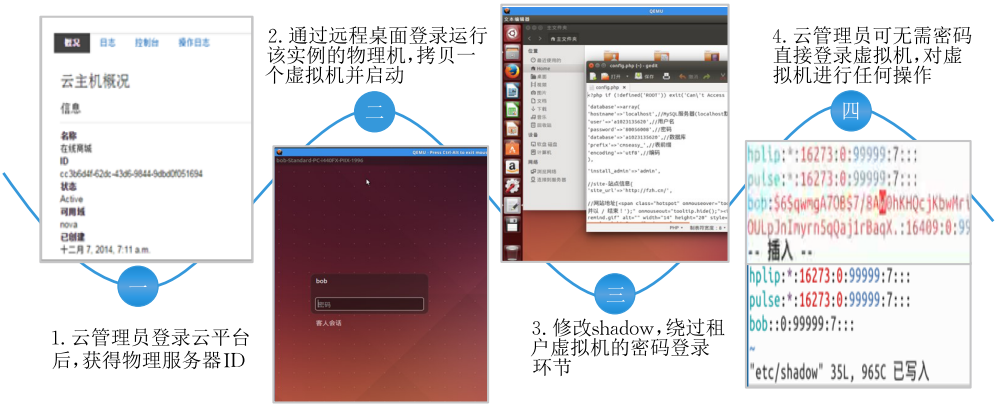



图 2 修改 Linux 虚拟机密码

### 2.2 云管理员篡改虚拟机登录密码

对于主流云平台上的 Windows 虚拟机，同样存在着被入侵的风险. 如图 3 所示，大致过程为：

(1) 在主流云平台下，云管理员将安装光盘挂载到虚拟机，并以安装盘启动虚拟机.

(2) 选择修复计算机，在系统恢复选项中，运行命令提示符. 在 system32 目录下，用 cmd. exe 替换其它可执行程序.

(3) 重新启动虚拟机，执行被替换的程序，此时实际执行 cmd. exe.

(4) 使用命令行方式修改管理员登录密码.

### 2.3 云管理员制作带有后门或木马的虚拟机镜像

云平台一般拥有大量的计算节点，不可能一个个安装，而是利用虚拟机镜像启动虚拟机. 如果镜像

图 3　修改 Windows 虚拟机密码

由云平台内部管理人员制作，则云管理员可在镜像中轻松植入后门或木马程序. 如图 4 所示，在使用该镜像生成用户虚拟机时，系统中将会留下后门，使得云管理员轻松获取虚拟机运行中生成的数据.

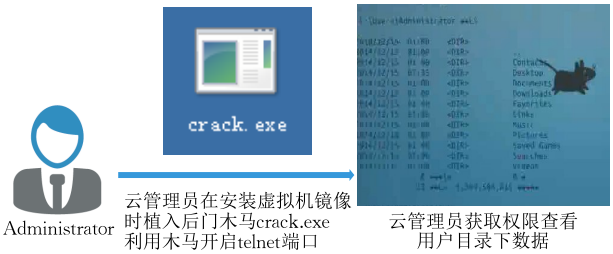



图 4　植入后门程序到虚拟机

## 2.4　云管理员获取虚拟机敏感信息

如图 5 所示，在主流云平台下，虚拟机用户使用虚拟机时输入了敏感信息. 云管理员利用 VMI (Virtual Machine Introspection)[15] 工具，导出正在运行的虚拟机的内存，并使用取证工具，如 Volatility① 分析内存内容，通过关键字符串查找匹配，找到敏感信息的内容.

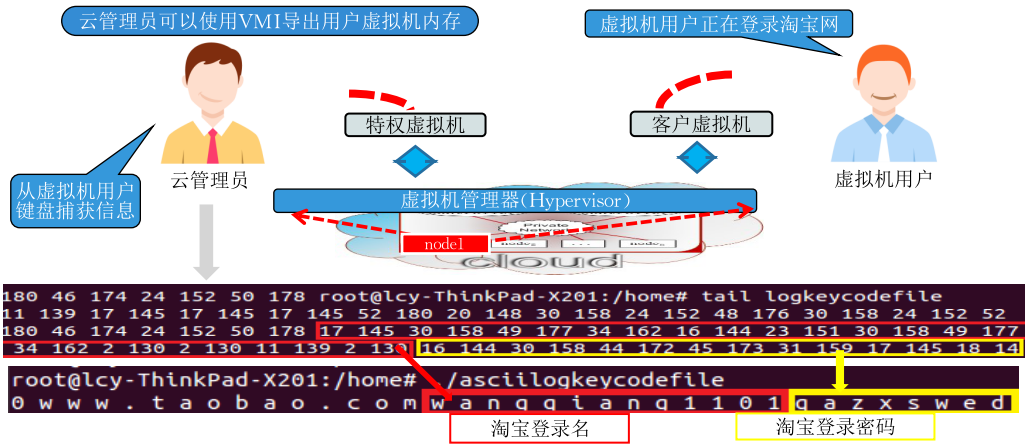



图 5　内存导出与分析

## 2.5　云管理员伪造 URL 连接租户虚拟机

如图 6 所示，在 OpenStack 云平台下，云管理员通过更改计算服务配置，以导出虚拟机用户 VNC 方式访问虚拟机的令牌(token)值，从而可以通过构造含有该 token 的 URL，连接进入到用户虚拟机活动会话的 VNC 桌面.

# 3　人员评价与行为分析

人员评价主要利用心理学和社会学技术，监控并分析内部人员的心理状态变化，一定程度上可以预测他们的恶意入侵倾向，做到防患于未然. 行为分析主要通过深度解析历史记录中的用户行为数据形成用户行为特征，然后比对用户操作的实时数据和基于历史操作记录得到的模型就能够识别出恶意的用户行为并提出预警. 同时人员评价和行为分析可互为补充，协同工作.

## 3.1　人员评价

根据以往的研究报告，内部人员进行恶意操作

① Walters A. Volatility: An Advanced Memory Forensics Framework. http://www.volatilityfoundation.org/. [Online; accessed Jun. 2015]

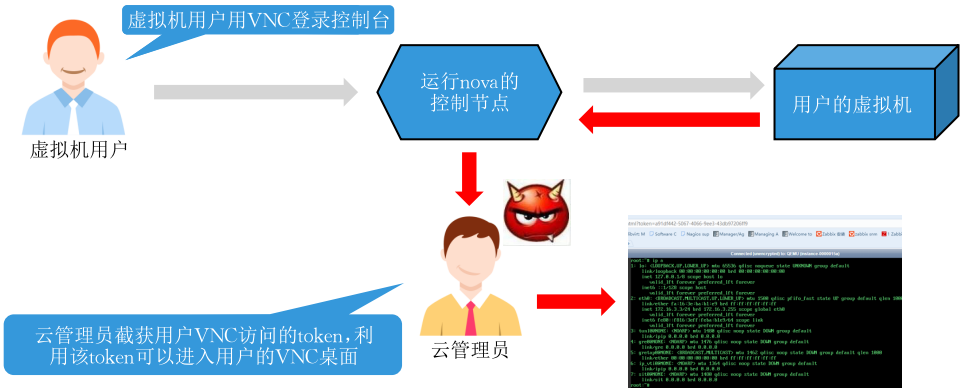



图 6 截获 token 进入虚拟机

之前常常伴有较为明显的心理学波动和社会学消极言论[16]. 监控并分析内部人员的心理状态变化一定程度上可以预测他们的恶意入侵倾向，做到防患于未然.

内部人员心理分析角度可能包括他们是否以自我为中心、是否为人冷漠、是否承受压力、是否傲慢自大、是否敢于铤而走险、是否有很强的抵触情绪等等[17]. 较为公认的心理学分析体系包括“黑暗三元组”(Dark Triad)理论和“五大人格”(Big Five)理论[18]. 这些理论体系可以将心理学指标数字化，以便更加准确地找出有恶意企图的内部人员. 内部人员心理学指标的确定，可以依据多方面数据的综合评估. 分析内部人员的网络言论、工作状态或对内部人员进行心理测验可以直接有效地得到心理学数据. 也可以根据内部人员的工作环境或人员之间的互相评价间接地分析出他们的心理状态. 下面将简要介绍几个较有代表性的相关工作.

Greitzer 等人[19]基于大量的事件实例和行为心理学关联研究构建了较为成熟的内部人员潜在威胁分析模型. 为收集内部人员的心理学数据，该研究项目组首先咨询了组织内部的管理阶层和人力资源团队，收集整理出人为观测的结果. 而后数据驱动的心理学推理系统将观测结果对应到不同的心理学指示器上(包括压力指数、不满指数等等). 该模型的实现主要可分为两大步骤：第 1 步将心理学指示器和心理学风险指数对应为“贝叶斯信念网”的二进制变量节点，并进行有监督的机器学习[20]. 该学习的目的是模拟出各个指示器优先级和权值. 第 2 步利用学习得到的模型分析新的员工心理指标，得出他们造成内部威胁的可能值. 这种方法的优点在于将抽象的心理学指标数字化，有助于进行准确的量化分析. 随后 Greitzer 等人[21]分析内部人员作恶前的行为表现，提出的心理学指标包括愤怒、不接受反馈、情绪管理问题、空闲时间、无视权威、效率、压力、对抗性行为、个人问题、自我、可靠性、旷工等. Nurse 等人[22]构建表征内部威胁的一个基础框架，提出内部威胁问题空间的分类，包括突发事件、人格特征、历史行为、心理状态、对待工作态度、技能、机会和攻击动机. Hashem 等人[23]利用人类生物信号(脑电图和心电图信号)来检测恶意活动. Tausczik 等人[24]分析日常使用的词语中在心理学有意义的类别词，用以检测人员的心理状态，如注意焦点、情感、社会关系、思维样式和个体差异等.

通过案例研究证明这种方法对预防内部威胁的发生具有一定的有效性，但这种方法容易侵犯用户隐私，且应用与实践的成熟系统很少，准确性不好验证.

### 3.2 行为分析

用户在使用云平台服务时，会留下大量的日志信息和行为轨迹，我们将其称之为“用户行为数据”. 这些数据一定程度上反映出了用户的个人特征和使用意图. 作为用户行为分析的基石，用户行为数据的采集和处理可谓至关重要. 因此，如何采集数据和采集什么数据这两大问题是用户行为分析工作开展之前要着重调研和解决的问题. 本文汇总了目前用户行为分析中常见的数据类别，并对相关工作进行了总结和评价. 本文将目前可用于用户行为分析的数据概括为以下 4 个类别：用户的认证信息、网络数据信息、用户的活动日志、用户的生物学行为特征. 其具体阐述如下：

(1)用户的认证信息. 用户在接入云服务时用于确认用户身份的基本信息，包括用户标志、用户密码、用户指纹等. 更进一步地采集用户所在的位置(IP 地址)、用户的权限等级[25]、用户的物理地址[26]也属于直接认证的范畴. 直接采集用户的基本认证信息可以准确快速地识别用户，当用户进行恶意操

作时，这一类数据有助于直接定位恶意用户. 然而仅仅采集用户基本认证信息来定位恶意用户的方式并不完善，例如用户名和密码等信息已被他人窃取[27]，用户的 IP 地址等信息已被伪造. Kent 等人[28]基于图的用户分类和入侵检测方法，把网络身份认证活动的数据通过用户特定的图和图特征表现出来，提供综合分析用户异常行为的方法.

(2) 网络数据信息. 用户在使用云服务的过程中，会在网络上留下大量的数据信息(网站接入信息、云主机记录、网络输出记录、网站访问记录、防火墙信息、DNS 记录等)，深入分析这些信息可以及时发现内部危险行为并进行预警. 采集这些细节的网络数据信息，可以从根本上对内部入侵进行防护，但其缺点在于数据收集和整理的工作量较大. Gavai 等人[29]首先提出标志内部威胁的相关特征，如电子邮件交流模式和内容特征、电子邮件频率、登入登出特征、应用程序使用特征、浏览网页使用特征及文件机器访问模式，结合异常分析算法预测异常行为. Mayhew 等人[25]使用基于行为的访问控制(BBAC)计算访问者的可信性(如行为和使用模式)及验证文件的属性(如数据依赖关系)，以判断活动是否合法. 观测内容包括网络连接、HTTP 请求、电子邮件或聊天信息及文档编辑序列.

(3) 用户的活动日志. 用户在使用系统时系统对用户具体操作进行的详细记录. 用户日志详尽地描述了用户进行的一系列活动，例如 Web 服务器日志[30]、活动目录日志数据[31]、用户使用 Unix 命令日志[32]等. 因此，它被广泛地用于检测组织内部人员的可疑活动. 由于用户日志数据数量庞大，且随着用户的使用不断增长，自动化数据分析成为目前面临的一大挑战.

(4) 用户的生物学行为特征. 用户在使用终端进行操作时，硬件可以采集用户使用模式特征或固有特性. 如用户使用键盘和鼠标[33-36]的最常用的使用习惯信息、用户的指纹或虹膜信息[37-38]等. 分析这类数据可以准确地判断操作者是否为本人，从而及时发现用户账号密码被窃取情况.

分析某一用户的生物学特征可以模拟出他的使用习惯和个人特征. 当其他人员(可能包括恶意人员)得到该用户的账户和密码并接入系统时，系统会发现其使用习惯或个人特征异常之处，从而引发预警. 评价用户生物、行为学特征的认证分析系统可以从漏报率、误报率、学习构建时间和健壮性四个方面着手. 不同角度和侧重的用户生物学特征具有各自的优势和不足. 例如：采集并分析用户的指纹或虹膜信息可以最准确地识别用户是否为本人，但需要昂贵的硬件支持和特定采集环境；采集用户键盘的击键序列和击键习惯信息也是一种常见的用户身份验证方式，这种方式的采集不需要额外的硬件添加方案实施简单，但是容易因为硬件的更换而受到影响，并且在采集生物学特征的同时收集了大量的用户个人信息(个人密码、输入信息等). 本文选取了以下两种生物学特征分析方法进行研究和评估.

利用键盘和鼠标使用习惯的分析可以实时验证用户的行为特征，防止了用户登录后暂离时的安全隐患. Zheng 等人[33]提出了一种基于鼠标轨迹分析的用户身份验证系统. 该系统具有三大优势：较低的漏报率和误报率、快速的验证响应速度、强大的抗模仿能力. 根据具体功能的不同，我们可将该系统划分为四大部分：记录器(Recorder)、预处理器(Pre-processor)、分类器(Classifier)和决策器(Decision). 记录器以四元组(操作类型，时间，$X$ 坐标，$Y$ 坐标)的形式实时记录用户的鼠标轨迹，而后输出到预处理器. 预处理器以鼠标点击为分隔整理成多组序列. 与此同时，预处理器还将原本的四元组数据换算成向量方向(Direction)、曲率角(Angle of Curvature)、曲率距离比例(Curvature Distance)的三元组，从而降低了硬件平台的依赖性，提高了用户之间的独特性. 分类器针对每一用户，将本用户的鼠标轨迹信息作为正样本、将非本用户的鼠标轨迹信息作为负样本输入支持向量机(SVM)中，从而得到本用户的鼠标特征分类. 最后，身份验证系统实时采集用户的鼠标轨迹信息，根据分类器的结果进行决策. 为使得决策结果更加准确，该系统使用了多数投票法(Majority Votes)作为决策器的主要决策算法.

目前，触屏智能手机得到了广泛地应用于公司内部人员的日常工作，然而相比于个人 PC，终端设备更易丢失或泄漏重要办公信息. 为解决触屏移动终端被入侵的问题，Sae-Bae 等人[34]和 DeLuca 等人[35]等人针对目前广泛使用的智能手机，设计了基于触屏使用轨迹的认证系统. 相比上述两者的工作，Frank 等人[36]实现了触屏轨迹的连续认证(Continuous Authentication)系统，其优点在于认证过程透明于用户且可以分析检验日常的用户操作. 从宏观上来讲，Frank 等人构建的系统可分为两大部分：登记学习阶段(Enrollment Phase)和连续认证阶段(Continuous Authentication Phase). 前者是认证系统的数据训练阶段. 由于该阶段使用有监督学习，系

统需要依赖常规认证手段(如账户密码等).为更好地采集数据,作者定义了触发动作(Trigger-Actions)这一概念,触发动作在所有的用户用例中都很常见而且几乎是所有复杂动作的一部分.该系统选定的触发动作有两种:水平划过屏幕、以页面上下移动为目的的垂直划过屏幕.通过不断监听、分析这些动作,系统提取出用户触屏轨迹特征并加以分类.分类学习完成之后,系统进入连续认证阶段.这一阶段中,系统不间断地采集所有用户触屏轨迹,并通过分类器识别这些轨迹是否来自合法用户.这一部分的性能(包括认证准确性和响应时间)主要受到登记学习阶段过程中分类精度的影响.当用户的移动终端被窃取时,系统可以及时发现手机遗失,并根据GPS定位窃取者.

行为分析需要收集大量用户历史操作数据,存在一定的误报率或漏报率,结合其他技术如人员评价可在一定程度上缓解处理大量数据的压力,降低误报率或漏报率.Greitzer 等人[16]描述了一个用于检测内部风险行为并触发报警的预测模型,行为分析结合心理数据,捕获恶意内部人员从事恶意活动的迹象,用于告警.心理学指标包括职场行为、个人行为、人力资源数据等.技术指标包括系统日志、入侵检测、数据防丢失等.本方法使用人员评价过滤数据,准确性有待验证,需进一步优化改进.

Brdiczka 等人[39]提出了一种基于用户日志的图结构行为分析系统,并结合心理学剖析进行威胁综合评定,心理学剖析的加入大大降低了结构异常分析的误警率,提高了系统的可用性.相对而言,结构异常分析的加入使得该系统比单纯进行心理学、社会学分析的系统更加精确、更加严谨.其系统概况如图7所示.结构异常分析可分为4个步骤:

(1)图结构分析(Graph Structure Analysis).内部员工在进行日常工作时总是大量地重复某些操作.我们认为这些日常操作是无害、安全的“正常行为”.图结构分析的作用是将“正常行为”与“异常行为”进行区分,从而发现信息网络数据的异常特征,根据这种特征将类似数据进行关联,从而减少数据分析的工作量.

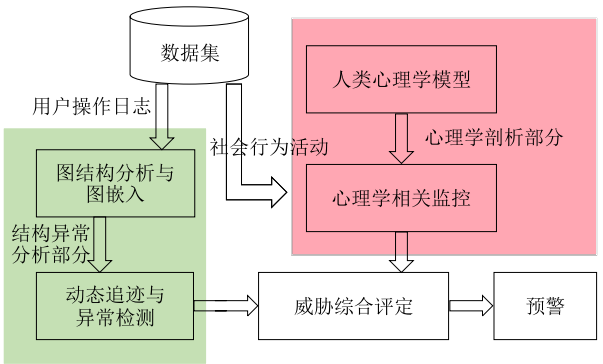



图7 行为分析系统架构

(2)图嵌入(Graph Embedding).将图结构分析中得的关联数据进行聚类,同一分类中的图节点(代表用户)具有更高的关联性,这当中运用到了机器学习的一些分类算法.

(3)动态追迹(Dynamic Tracking).动态追迹用于监视异常行为模式下的个人行为,其中用到了一些固定欧式空间下的动态实体追迹方法(如贝叶斯方法).

(4)异常检测(Anomaly Detection).动态追迹过程中,追迹系统将主体行为处理成多变量行为序列,序列中的变量是一些内部人员的属性.异常检测利用这些序列和已知的异常行为构建一个概率模型,并利用这一模型来识别新的异常行为.

行为分析可以有效地验证用户是否是真正的用户,但其本身为被动防御,如果用户本身想要作恶,且操作极其隐蔽,则可能逃过检测,怎样提高拦截的时效性也是研究的重点.

## 4 云管理权限划分和执行时验证

这种技术路线主要针对云系统中资源或数据制定细粒度权限划分;或是在硬件层或虚拟机管理器层对用户数据的访问进行执行时验证,使管理员即使拥有系统权限也无法得到客户的真实数据.

### 4.1 云管理权限划分

在虚拟化平台中,虚拟机管理器(VMM)是云平台最有权限的软件,可以访问包括用户虚拟机(VM)的所有资源.特权域(Dom0)是用户VM管理域,管理和复用硬件资源,管理客户虚拟机,比客户虚拟机拥有更多特权.传统的VMM赋予Dom0的权限太大,致使恶意的内部管理员可以利用Dom0的权限侵害用户数据隐私,如云管理员可利用虚拟机管理域导出虚拟机内存窃取用户数据.

自服务云计算模型(SSC)[40]利用隔离虚拟机、分割Dom0特权来防止特权域对用户隐私造成威胁,VMM将原有Dom0的权限分离,以防止用户数据受到来自管理域的攻击威胁.SSC中的虚拟机管理器将原有管理域的权限分离,并提供了两类管理域:(1)由云管理员使用的解除特权的系统管理域.

(2) 基于用户的管理域，用户可以使用该管理域管理自身虚拟机. 这种新型权限模型阻止了恶意的内部管理员去监听或者修改用户虚拟机状态，并且使用户更加灵活地控制自己的虚拟机.

如图 8 所示，SSC 特权模型划分为 4 类计算模型：Sdom0 是由云管理员使用的解除特权的系统管理域. Sdom0 中保留启动或停止 Udom0 域的权限，并运行虚拟化设备的驱动程序，管理硬件资源. SSC 的权限模型禁止 Sdom0 检查客户域的状态，没有访问客户虚拟机的权限，确保客户域的安全性和隐私性. Udom0 为基于用户的管理域，用户可以使用该管理域实现对自身虚拟机的管理，委托其特权到服务域. Service VM 为客户虚拟机提供客户定制的服务，如验证 VM 完整性、入侵检测和 VM 数据加密等. Mutually Trusted Service VM 按照云平台和用户之间的协商执行对虚拟机的监控，防止虚拟机用户滥用服务.

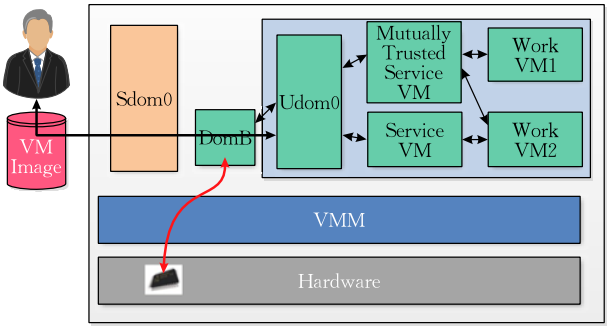



图 8　SSC 系统架构

客户首先与 DomB 建立加密通道传输数据，如 Udom0 镜像. Udom0 由 DomB 验证镜像完整性并启动 Udom0，从而确保 Udom0 的安全可信. DomB 建立 Udom0 虚拟机，将用户密钥置于 Udom0 内存中，之后 Udom0 与用户建立加密可信通道，传输虚拟机镜像或服务镜像. Udom0 调用 DomB 建立镜像对应的虚拟机. 这种新型权限模型的不足是需要在云平台安装安全芯片，且需要改变虚拟机管理程序架构和用户镜像，不利于实施.

### 4.2　云执行时验证

云执行时验证通过仲裁 VMM 和虚拟机之间的控制转移保护用户的数据隐私. 为防止恶意或好奇的管理员偷窥或修改租户的虚拟机状态，缩小可信计算域，CloudVisor[41] 利用嵌套虚拟化技术保护虚拟化资源，使恶意管理员即使完全控制 VMM 和 Dom0 也不能窃取用户虚拟机隐私，大大减少了可信计算域的大小和风险.

CloudVisor 通过在一般云操作系统上加一层安全管理层，来保证云操作系统本身的安全性，提供虚拟机状态的保密性和完整性. CloudVisor 运行在最高特权模式下，而 VMM 运行在非特权模式. CloudVisor 利用最高特权层，使用硬件辅助虚拟化技术实施内存隔离，VMM 和虚拟机之间的控制转移通过 CloudVisor 仲裁，不允许 VMM 到 VM 内存的非法映射. 其中，控制转移过程中虚拟机的执行上下文被 CloudVisor 安全的保存和恢复. 这种方法也需要在云平台安装安全芯片，并且需要修改 VMM 程序，存在兼容性问题.

为保证平台可信启动，在 BIOS 载入启动程序之前，BIOS 首先会利用哈希验证启动程序. 哈希存储在硬件安全芯片中. 启动程序会去验证 CloudVisor 二进制程序. 平台启动后，远程用户可利用云平台安全芯片远程验证运行的 CloudVisor 是否完整可信，没有受到损害. 客户 VM 镜像加密传输存储到云平台，并可利用元数据验证是否被篡改.

在 CloudVisor 系统中，VMM 的扩展页表 (Extended Page Tables，EPT) 只具有读权限，CloudVisor 维持一份相同的 EPT 页表完成客户物理地址到主机物理地址的转换. 分配给 VM 的页面会改变其所有者为 VM，CloudVisor 将此页面从 VMM 的 EPT 页表中解除映射. 当页面回收时，页面重新被 VMM 所有. 页面只能被所有者访问，这样当 VMM 访问客户虚拟机页面时会出现错误陷入 CloudVisor，CloudVisor 加密后映射到 VMM 页表，把页面所有权设为 VMM. 当 VMM 要改变 VM 页表时也会引起修改页表错误，陷入 CloudVisor 处理，CloudVisor 判断页面是否属于 VMM.

如图 9 所示，VM 特权事件执行流程为：

(1) 当 VM 执行特权事件时会先陷入 CloudVisor；

(2) CloudVisor 执行必要的操作，如保护 CPU 上下文和解析 I/O 指令，如果需要再交给 VMM 执行；

(3) VMM 执行后回到 CloudVisor；

(4) 经过 CloudVisor 处理后如恢复 CPU 的上下文，再转到 VM 执行.

用户的虚拟机是有可能被云管理员非法入侵和掌控的，所以针对虚拟机被入侵情况下用户程序的安全保护也是尤为重要的. Overshadow[42] 和 Inktag[43] 研究怎样在用户虚拟机被入侵情况下保证用户虚拟机重要程序的安全性. 假定 VMM 是可

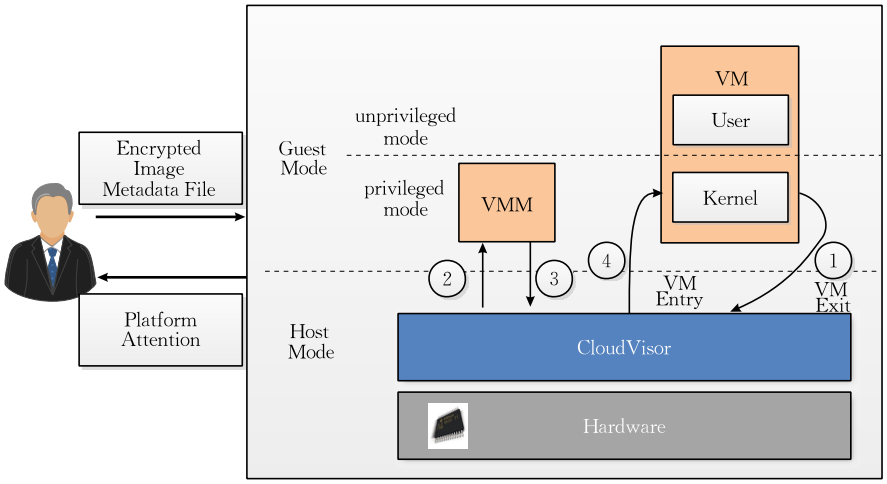



图 9 CloudVisor 系统架构

信的,而操作系统是不可信的,通过 VMM 和保护程序之间的协商,仲裁程序与系统交互以验证不可信系统行为. Overshadow 基于虚拟化技术,采用额外的阴影(shadow)页表映射,包括系统 shadow 页表和保护应用程序 shadow 页表,提供不同的客户机物理内存视图,对虚拟机中要保护的应用程序(Cloaked APP)所有内存页加密的方法来保护用户的隐私性,并根据当前访问视图动态地加密和解密内容,向保护程序提供明文视图,向操作系统提供加密视图,保证用户数据不被侵入虚拟机的恶意用户窃取.

如图 10 所示,VMM 与 Shim 合作仲裁保护程序和操作系统(OS)之间的交互. Shim 是与 OS 相关的用户级程序,链接在保护程序地址空间. Overshadow 加载程序时在保护程序的地址空间引入 Shim,仲裁程序和操作系统的交互,分为 Uncloaked Shim 和 Cloaked Shim,OS 调用 Uncloaked Shim 超级调用(Hypercall)初始化新的保护程序上下文,并且进入并恢复保存的保护程序执行环境,程序调用

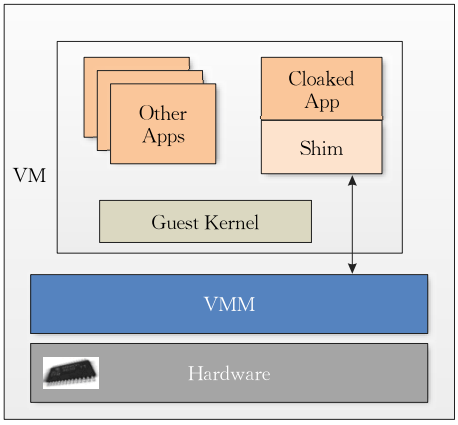



图 10 Overshadow 系统架构

Cloaked Shim,可以保护新的内存区域,解密保护数据,建立新的 shadow 上下文等.

当启动一个保护程序时,Shim 必须首先安全启动,调用 Hypercall 验证自身的完整性并启动保护程序. 然后 Shim 运行它的加载程序,该程序读取应用程序二进制文件,将适当段映射到内存中. 当页面在系统 shadow 页表上无法映射时发生错误陷入 VMM,VMM 加密页面,计算哈希,并将页面重新映射到系统 shadow 页表. 当页面在程序 shadow 页表上无法映射时发生错误陷入 VMM,VMM 验证完整性,解密页面,将明文页面重新映射到程序 shadow 页表. 解密用到的元数据存储在 VMM 或程序内存,OS 访问不到.

VMM 执行 shadow 页表转换时必须识别程序上下文,CTC(Cloaked Thread Context)结构用于保存程序和系统控制转换的隐私数据,CTC 置于保护内存区域,受加密和完整性保护,OS 不能读或修改它. 当执行从 OS 恢复到保护程序时,系统调用 Uncloaked Shim 传入 ASID(Address Space Identifier)和 CTC 指针给 VMM,VMM 验证 ASID 和随机数. 如果 OS 在不合适的地点恢复保护程序,完整性验证不会通过,所以只能在适当的程序代码及适当的地点恢复程序.

通过以上机制确保进入或退出保护程序时的数据隐私,如系统调用、错误、中断和信号传递等. VMM 保证保护程序内存页面完整性和安全性,维持页面和页面元数据的映射关系. 典型错误或中断处理流程:

(1) Cloaked APP 执行错误或中断,陷入 VMM.

(2) VMM 保护程序寄存器内容到 Cloaked Shim CTC,并修改寄存器内容,执行 shadow 页表

切换.

(3) 修改 IP 和 SP 寄存器指向 Uncloaked Shim 地址，设置内核执行完返回的处理函数.

(4) VMM 转换控制权到内核执行.

(5) 内核处理完返回到 Uncloaked Shim 处理函数.

(6) Uncloaked Shim 调用 Hypercall 进入 VMM.

(7) VMM 恢复寄存器内容返回 Cloaked APP，执行 shadow 页表切换.

Overshadow 目前使用单密钥，由 VMM 管理来加密所有页；这种方法需要在云平台安装安全芯片，并且每个安全程序需要两套页表，内存占用率高，一次控制转换需要在保护程序和 VMM 之间执行数据保护及加解密等多步操作，从而带来一定的性能损耗.

Inktag 基于虚拟化技术在不可信的操作系统中保护关键应用，当属于某被保护程序的页表更新时，InkTag 要验证虚拟机操作系统内存映射和程序请求是一致的，确保该更新反映了应用程序的意图. InkTag 提供以下安全保证：(1) 控制流完整性：OS 不能改变程序计数器、寄存器；(2) 地址空间完整性：OS 不能读、改变程序数据；(3) 文件 I/O：保证文件数据的隐私性和完整性.

InkTag 目前基于半虚拟化技术实现，从而不具有普遍性；这种方法也需要在云平台安装安全芯片，并且每个安全程序需要 VMM 和程序配合实现，不具有通用性.

## 5　用户可控的数据加密

第 3 节和第 4 节主要是针对如何管控云端的管理权限. 但是云模式的实质是数据的所有权和控制权分离了，用户失去了对其数据的直接管理. 因此，用户的最后“杀手锏”即是把数据加密以后再存储到云端. 若用户密钥也在云中存储和使用，则云管理员可通过特权域得到密钥进而窃取用户数据[11-12]. 所以数据最好在客户端加密再上传到云服务器，解密密钥在用户手里，而服务商只能看到密文，数据控制权完全在用户手中. 我们把这种思路称为“用户可控的数据加密”. 进一步从是否依赖云服务商的角度可将现有工作分为不依赖云服务提供商的用户数据加密和依赖云服务提供商的用户数据加密两类. 表 1 为典型用户可控的数据加密原型系统对比.

**表 1　典型用户可控的数据加密系统对比**

| 加密系统 | 需要云配合 | 可信第三方 | 加解密位置 | 特殊功能 |
|---|---|---|---|---|
| CryptDB[44] | 不需要 | 不需要 | 数据库代理 | 常用数据库操作，如搜索、比较、排序等 |
| Mylar[45] | 不需要 | 不需要 | 客户端与服务端之间 | 多用户、搜索 |
| ShadowCrypt[46] | 不需要 | 不需要 | 用户与客户端之间 | 易适配、搜索 |
| Over-encryption[47] | 需要 | 不需要 | 客户端与服务端之间 | 多用户、密文访问控制 |
| PasS[48] | 需要 | 需要 | 客户端与服务端之间 | 在云端利用硬件先解密再处理 |

### 5.1　对云服务商透明的用户数据加密

不依赖云服务提供商的用户数据加密不需要云服务提供商配合，如 CryptDB[44] 有效预防第三方数据存储管理人员的内部威胁；Mylar[45] 有效预防服务器端管理人员的内部威胁；ShadowCrypt[46] 可预防客户端恶意代码及服务器端的内部威胁.

#### 5.1.1　CryptDB

CryptDB 对数据在存入数据库前进行加密，并在加密数据上进行数据查询处理，从而有效防范了恶意的数据库管理员. CryptDB 利用数据库代理，对用户的操作进行转换，得到加密结果，然后解密返回给用户.

如图 11 所示，CryptDB 主要由数据库代理(proxy)、CryptDB UDFs(User-Defined Functions)模块组成，proxy 执行加解密操作，CryptDB UDFs 在 DBMS 端执行用户定义的操作. 针对不同的操作有不同的加密方案，如 RND 加密安全性最高，但不支持任何数据操作；HOM 加密方案用于密文加法操作；SEARCH 加密可用于密文搜索；DET 用于相等比较；JOIN 加密用于聚合操作；OPE 加密用于排序比较操作，其安全性依次降低. 对于相同加密层同一列加密密钥是相同的，而不同列加密密钥则不相同. 数据存入数据库前多层加密，为了使用安全性先在操作列 RND 加密层上操作，若不支持则把 RND 层密钥传到服务端执行 UDF 操作降到可支持的加密层继续处理，并维持在支持的加密层，同时 proxy 更新相应列加密层，当服务器返回加密数据时对数据解密并返回给用户.

针对一次请求操作，可能需要以下过程：

(1) 当用户程序发起数据查询请求.

(2) proxy 拦截并重写请求，隐匿并转换表名和

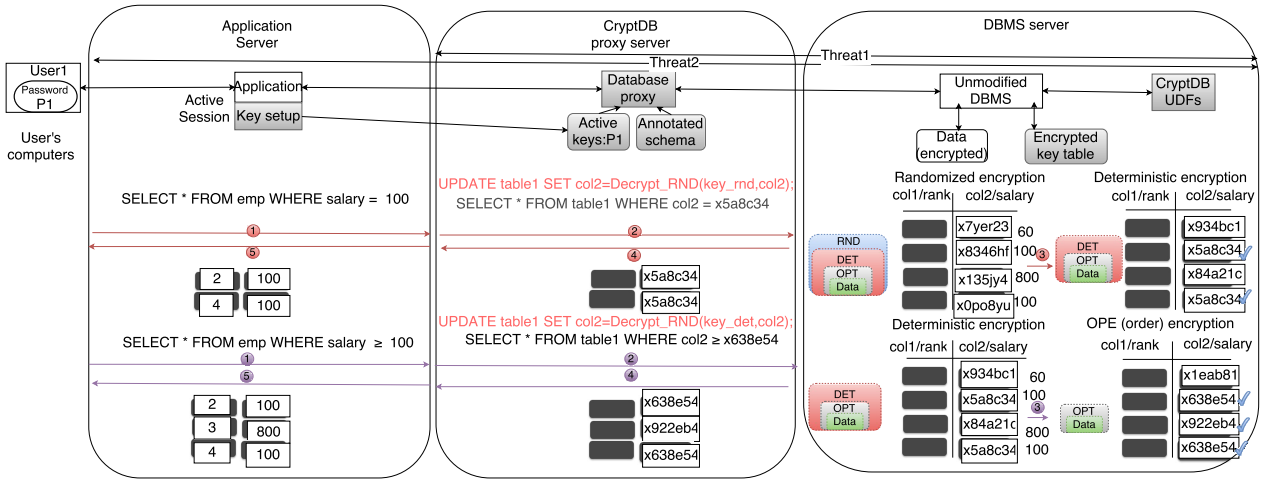



图 11 CryptDB 系统架构

列名(如 emp 对于 table1,salary 对应 col2),并利用主密钥选择合适的加密方案加密请求中含有的常量. 当 proxy 判断 DBMS server 需要调整加密层执行请求操作时,代理会传送相应层的解密密钥给 DBMS server,并发起 UPDATE 请求使 DBMS server 调用 UDF 调整相应操作列加密层到适合的加密方案层(如调整 RND 层到 DET 层).

(3) proxy 再转发加密请求到 DBMS server,DBMS server 利用标准的 SQL 处理语句处理(有时需要调用相应的 UDF 处理聚合或关键字搜索).

(4) 最后 DBMS server 返回加密结果给 proxy,并传送相应的密钥.

(5) 代理解密并返回客户相应数据.

针对密文搜索 SEARCH 的实现,CryptDB 采用 Song 等人[49]提出的密文搜索方案 SSKE(Searchable Symmetric Key Encryption),该方案首先把全文处理成固定长度(如 $n$ 位)的单词,对任意单词 $Wi$ 利用密钥 $k$ 加密生成 $E_k(W_i)$,拆分成 $L_i$(前 $n-m$ 位)和 $R_i$(后 $m$ 位);利用伪随机函数 $f$ 生成 $K_i=f_{K'}(L_i)$;生成($n-m$)位伪随机位 $S_i$,利用伪随机函数 $F$ 生成后 $m$ 位 $F_{K_i}(S_i)$,拼接在一起为 $n$ 位与 $E_k(W_i)$亦或生成密文. 查询时用户传给服务器 $E_k(W_i)$和 $K_i$,服务器利用密文单词与 $E_k(W_i)$亦或看结果是否满足$(s,F_{K_i}(s))$的形式,如满足则返回对应的密文文件. 这种方法优点是搜索运算简单,缺点是云端需要全文线性搜索,计算量与文件大小成正比;不能应用于公私钥加密体系;此外,服务器可利用查询请求获得额外的用户信息.

针对聚合操作 JOIN,若两列使用相同密钥,则很容易支持聚合操作. 若两列使用不同密钥加密,CryptDB 采用 JOIN-ADJ 方案,直观地说,JOIN-ADJ 可以使 DBMS server 在不去解密的情况下调整某列的密钥使相同的明文有相同的密钥. 两列首先在 JOIN 层使用不用密钥加密,当有聚合请求时,proxy 计算两列密钥之间的陷门并发送给 DBMS server,DBMS server 调用 UDF 使用陷门调整某列,使两列有相同的密钥,而后可执行聚合操作.

针对排序比较操作 OPE(Order-Preserving Encryption)的实现,CryptDB 采用第一个符合安全性需求的保序加密技术 OPSE[50](Order-Preserving Symmetric Encryption)方案. OPE 最早由 Agrawal 等人[51]提出,支持对加密数值数据的各种比较操作,但不支持其他数据类型. Wang 等人[52]基于 OPSE 实现检索结果可排序方案. 近年来,为进一步提高安全性,Chenette 等人[53]通过使用"虚拟查询"避免泄露 MOPE[54]秘密偏移. 然而,如果分布极不均匀,虚拟查询的数目可能会很大. Liu 等人[55]指出理想的安全 OPE 方案[56]效率低,在某些方面不适用,其通过信息空间扩展和非线性空间分割隐藏数据分布和频率. Kerschbaum[57]使用随机化密文来隐藏明文的频率,如对精确值的查询转变为对精确值所属范围的查询.

密文加法操作 HOM 借鉴了全同态加密计算,即直接对密文进行计算获得密文结果,密文结果解密后即是对应的明文运算结果. Gentry[58]首次设计出了一种基于理想格的全同态加密方案,能同时支持加法和乘法同态. 但该方案伴随着密文同态运算次数增加噪声也会相应增加,且噪声积累到一定量后就不能作密文计算了,故只能支持有限的密文加法和乘法运算次数. Gentry 的解决方案是引入 Bootstrapping 技术,当密文噪声达到一定阈值时,利用同态运算将密文和其对应的密钥用新的密钥加

密并作同态解密操作，由于是对新的密钥产生的密文做操作，从而自动对明文加密，完成对密文的重加密，消除了一定的噪声，从而支持任意次数的密文计算. 该方案理论上可行，但计算复杂①②，导致现有方案在云计算环境中不具有实用性. CryptDB 的解决方案是基于 Paillier 加密系统[59]，可有效地支持特定的运算，如两个数值密文的乘得到相应明文和.

如图 11 所示，CryptDB 可有效防范 Threat1，但针对 Threat2，当用户在线且密钥被服务器程序窃取时不能防止服务器程序作恶，且客户端程序如果被篡改也有可能泄露数据隐私. 同时一次查询有可能需要多次加解密，使执行时间较长.

### 5.1.2　Mylar

针对预防服务器端应用程序，Mylar 扩展 Meteor JavaScript 框架，通过对应用程序传送到服务器的数据加密来保证数据的隐私性和安全性，从而可有效防范恶意的服务器管理员.

如图 12 所示，Mylar 主要由以下部分组成：(1) 浏览器插件，负责验证客户端代码的数字签名（与验证 HTTPS 网页的数字签名类似），保证没被篡改；(2) 客户端库，对发往服务器的数据进行加密，对来自服务器的数据进行解密；(3) 服务器端库，在加密数据中进行搜索；(4) 身份鉴别服务，验证公钥是否属于特定用户.

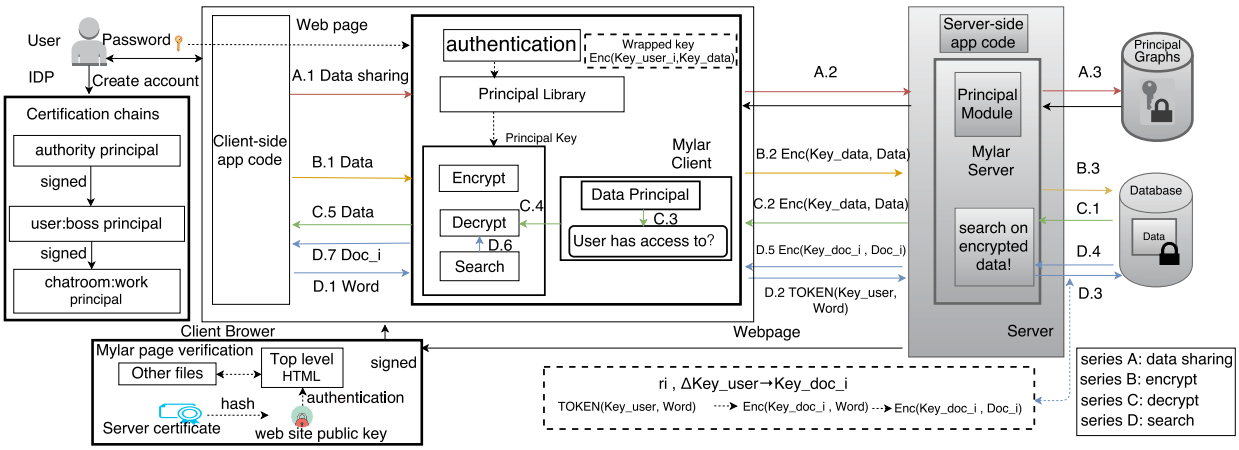



图 12　Mylar 系统架构

如图 12 系列 B 所示，用户上传数据时，利用数据 Principal 的公钥对数据加密并附加加密的元数据，再上传到服务器. 如系列 A 所示，当要共享数据时，获取共享数据对应的 Principal 密钥，加密数据再发送给服务器. 数据共享，也就是密钥分发，要访问共享数据，就要把加密数据的私钥用用户的公钥加密存储在服务器上，这样用户可用自己私钥解密得到加密数据的私钥，进而解密得到数据. 数据访问授权是传递性的，也就是如果 A 授权 B 访问，B 授权 C 访问，则 C 有权访问 A. 因此这种授权关系形成了一张图，称为 Access Graph. 如系列 C 所示，用户登录时经过验证根据 Principal 库获取自己的公私钥，当收到数据时，首先利用密文数据的注解获得对应的 Principal，然后利用自己对应的 Principal 根据 Access Graph 看是否能访问密文数据对应的 Principal，如果能则利用数据对应的 Principal 的私钥解密数据获得明文. 如系列 D 所示，密文搜索仅仅需要提供一个搜索 token，服务器利用私钥差调整 token 到另一个可搜索值，保证服务器不能获取要搜索的词，不能获取文档的内容.

搜索具体算法如下：利用椭圆曲线上双线性映射函数 $e: G_1 \times G_2 \rightarrow G_T$，其中 $G_1$，$G_2$，$G_T$ 是椭圆曲线上阶为 $p$ 的特殊构造群，$g$ 为 $G_2$ 的生成元. $H$ 和 $H_2$ 是椭圆曲线上特定的哈希函数. $e$ 满足方程 $e(H(w)^a, g^b) = e(H(w), g)^{ab}$.

过程 1. 生成密钥：$KEYGEN()$.

$key \leftarrow$ random value from $Z_p$（$p$ 阶循环群）

return $key$

过程 2. 加密：$ENC(key, word)$.

$r \leftarrow$ random value from $G_T$

$c \leftarrow (r, H_2(r, e(H(word), g)^{key}))$

return $c$

① Cooney M. IBM touts encryption innovation: new technology performs calculations on encrypted data without decrypting it. Computer World: 2009 June 25. Available at: http://www.computerworld.com/article/2526031/security0/ibm-touts-encryption-innovation.html

② Greenberg A. DARPA will spend 20 million to search for crypto's holy grail. Forbes. Apr 2011. Available at: http://www.forbes.com/sites/andygreenberg/2011/04/06/darpa-will-spend-20-million-to-search-for-cryptos-holy-grail/#55593fca424c

过程 3. 生成 token：$TOKEN(key,word)$.

$tk \leftarrow H(word)^{key}$ in $G_1$

return $tk$

过程 4. 密钥差计算：$DELTA(key1,key2)$.

$\Delta_{key1\rightarrow key2} \leftarrow g^{key2/key1}$ in $G_2$

return $\Delta_{key1\rightarrow key2}$

过程 5. 服务端调整 token：$ADJUST(tk,\Delta k1\rightarrow k2)$.

$atk \leftarrow e(tk,\Delta_{k1\rightarrow k2})$ in $G_T$

return $atk$

过程 6. 服务端匹配程序：$MATCH(atk, c=\langle r,h\rangle)$.

$h' \leftarrow H_2(r,atk)$

return $h' ?= h$

利用以上算法针对具体程序实现具体的密文搜索功能. 如在聊天室程序中，Bob 被授权访问聊天室 party 时，Mylar 计算出 Bob 的私钥和聊天室 party 的私钥之差，并把这个差存储在服务器上. 当 Bob 被授权访问聊天室 work 时，Mylar 把其私钥和聊天室 work 的私钥之差存储服务器上. 当 Bob 需要在所有聊天室中搜索时，只需将待查词利用自己的私钥形成搜索 token 并发给服务器，服务器会利用之前存储的“私钥差”把待查词用 party 和 work 私钥加密的结果分别计算出来，再利用 party 和 work 聊天室中已经存在的索引，就可以在较短时间内找到包含待查词的文档.

Mylar 程序需要开发者告诉 Mylar 什么数据需要加密，影响了向后兼容性，且针对开发者实施加密，并不是针对用户，需要应用程序在 Meteor JavaScript 框架开发. 客户端代码验证需要服务端配合，需要修改服务端代码. 服务器端无法对加密数据执行计数、排序等操作.

Mylar 构建的面向多用户使用索引的密文搜索需要服务端调用 Mylar 库实现，利用特定的数据加密方案，使用陷门函数获取特定的密文. 密文搜索一直是数据加密研究的热点，表 2 是密文搜索相关文献的功能对比. 针对多用户密文搜索，Boneh 等人[60]利用非对称加密实现了 PEKS(Public Key Encryption with Keyword Search)算法. 数据拥有者在向云服务提供商上传数据前，用数据使用者的公钥加密数据，并将该文件的所有关键字分别用特殊的可搜索算法(PEKS)进行加密. 上传文件时，拥有者一并上传密文和加密后的关键字. 数据使用者进行搜索时，事先将要搜索的关键字换算成陷门(trapdoor)Tw 发送给云服务提供商. 云服务提供商利用比对函数处理 Tw、使用者的公钥和所有加密的关键字，从而断定要搜索的关键字是否在某一文档中，并向用户返回结果. Liu 等人[61]基于 PEKS 提出 SPKS(Secure and Privacy Preserving Keyword Searching)方案，使云服务提供商可以在不知道明文具体内容的前提下参与部分的解密工作，从而降低了数据用户解密密文的运算压力. Goh[62]提出了使用安全索引的方法实现密文搜索. 用户将文档加密并利用 Bloom Filter为每个文档生成 Z-IDX 索引，然后将安全加密索引和加密文件上传到云端. 搜索时，用户将搜索关键字对应的查询单射函数发给云端，云端对索引进行搜索并返回结果. 然而该方案采用的 Bloom Filter 存在错误率，可能导致搜索结果不准确. Curtmola 等人[63]首次使用倒排索引构建密文搜索机制，使搜索效率大大提高，并提升密文搜索的安全性.

**表 2 密文搜索相关文献的功能对比**

| 文献简称 | 模糊匹配 | 多关键字 | 智能排序 | 索引更新 | 服务端配合 | 用户(发/收) | 加密方案 |
|---|---|---|---|---|---|---|---|
| SWP[49] | 不支持 | 不支持 | 不支持 | 全文检索 | 需要 | 单/单 | 对称 |
| Goh[62] | 不支持 | 不支持 | 不支持 | 支持 | 需要 | 单/单 | 对称 |
| CGK[63] | 不支持 | 不支持 | 不支持 | 不支持 | 需要 | 单/单 多/多 | 对称 |
| XWS[66] | 不支持 | 支持 | 支持 | 支持 | 需要 | 单/单 | 对称 |
| YLW[67] | 不支持 | 支持 | 不支持 | 不支持 | 需要 | 单/多 | 对称 |
| BCO[60] | 不支持 | 不支持 | 不支持 | 非索引 | 需要 | 多/单 | 非对称 |
| SLN[68] | 支持 | 不支持 | 不支持 | 非索引 | 需要 | 多/单 | 非对称 |
| WWP[69] | 不支持 | 支持 | 不支持 | 不支持 | 需要 | 多/多 | 非对称 |
| ShadowCrypt[46] | 不支持 | 不支持 | 不支持 | 非索引 | 不需要 | 单/单 | 对称 |
| M-Aegis[70] | 不支持 | 不支持 | 不支持 | 非索引 | 不需要 | 单/单 | 对称 |

近年来，密文搜索研究进一步发展，实现可多关键字搜索、智能排序、索引更新、模糊匹配等高级功能. Sun 等人[64]、Cao 等人[65]支持多关键字和结果排序. Xia 等人[66]在云加密数据上实现多关键字排名的搜索方案，同时支持动态更新操作，如文件删除和插入. 以上密文搜索方案需要将加密索引上传到服务器且需要服务器配合实现特定搜索功能. 针对不改变服务器程序逻辑且不改变用户使用习惯而实现密文搜索，ShadowCrypt[46]利用浏览器插件对关键字建立哈希存在密文数据之后，M-Aegis[70]基于

手机平台提出介于用户和应用程序之间的 7.5 层，可透明地加解密用户数据并设计密文搜索方案 EDESE(Efficiently-Searchable Symmetric Encryption Scheme)，使搜索依然可以使用原来的字符串匹配逻辑.

5.1.3　ShadowCrypt

针对预防客户端应用程序及服务器端应用程序偷窥用户隐私数据，ShadowCrypt 作为浏览器插件运行，为网络应用程序加密文本数据.

如图 13 所示，ShadowCrypt 把用户输入输出数据从原程序 DOM 隔离开来，在浏览器插件中利用 shadow DOM 和 shadow JS 执行加解密等功能. 如图 13 右部分所示，当载入网页时 ShadowCrypt 对数据进行扫描，根据 DOM 找到其中的数据输入部分，建立对应的 shadow DOM 树，利用安全独立的阴影输入替换页面中的输入元素，用户输入的内容先要经过 shadow JS 加密，然后再给应用程序. 这样即使应用程序或服务器被入侵也不会泄露用户的隐私数据. 应用程序返回给用户的数据会回调 shadow JS 程序进行解密密文数据，再返回给用户. 加密后的数据赋予加密特征，如含有密钥指纹及用到的参数，来帮助应用程序识别密文数据并解密返回给用户. 如图 13 左下部分所示，当进行数据搜索时，ShadowCrypt 对文本内的关键字建立哈希存在密文数据之后，这样对数据进行搜索时把数据转换成对应的哈希值进行搜索，安全性较低，容易遭受统计攻击.

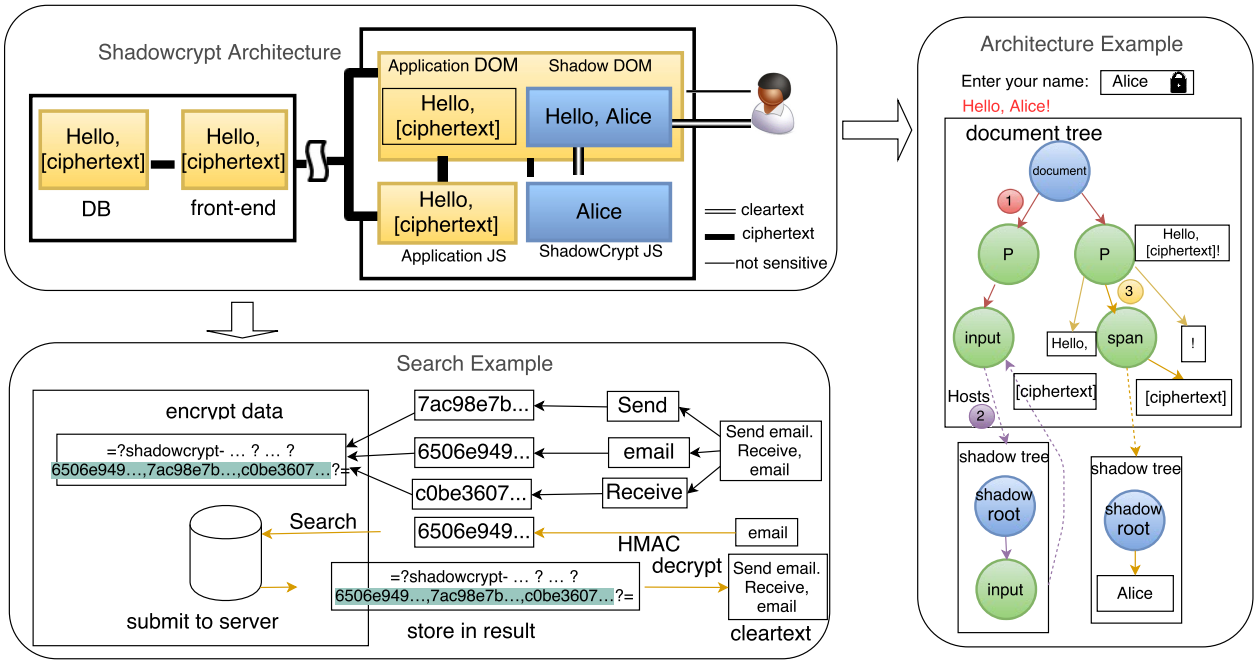



图 13　ShadowCrypt 系统架构

ShadowCrypt 对程序没有特殊要求，针对用户服务，支持目前的 WEB 程序. ShadowCrypt 在用户与客户端程序之间加解密，从而最小化可信任基. 但 ShadowCrypt 只能在 PC 平台 WEB 浏览器如 Chrome 或 Firefox 上运行，不能加密手机应用程序的数据，且不是通过输入元素传输的隐私数据不能被加密.

## 5.2　需要云服务商配合的用户数据加密

依赖云服务提供商的用户数据加密需要云服务提供商配合. 如客户端和云端双层加密需要云服务商执行多用户密文访问控制加密，硬件层配合客户端加密需要云平台硬件层提供加解密密码协处理器. 以上方案虽然需要云服务商配合，但是数据到云服务商之前已经过加密，故可有效预防云服务商内部人员偷窥或窃取用户隐私.

5.2.1　客户端和云端双层加密

Over-encryption[47] 为了实现客户端加密同时实现在云端对用户数据作访问控制，将数据保护加密与动态授权加密分离，而不必请求所有者每次在授权策略改变时下载并重新加密资源. 第一层 Bel 层加密由数据所有者在发送数据到服务器之前执行，在初始化时使每个用户拥有自己的密钥，并分配一些可用的令牌，用户利用公共令牌结合自己的密钥得到其他资源密钥解密自己可访问的资源，从而服务商窃取不到用户的隐私数据. 第二层 Sel 层加密在服务端执行，对数据所有者已经加密的数据选

择性加密,在第一层加密的方案下调整自己的密钥转换方案实行动态资源授权与撤销,用户拥有双层密钥才能访问数据.其中 Sel 层分为Full SEL和 Delta SEL 层,Full SEL 方法提供更安全的保护,但性能较高;Delta SEL 方法提供了优越的性能,但安全性较低.

如图 14 所示,首先根据资源访问策略建立访问控制树,如资源 $r_1$,$r_2$ 可被用户 C 访问,资源 $r_3$,$r_4$ 被用户 C、D 访问,资源 $r_5$,$r_6$,$r_7$ 被用户 A、B、C 访问,资源 $r_8$ 可被用户 A、B、C、E 访问.得到资源加密密钥之间的关系后,对不同资源利用相应的密钥进行加密,如 $r_1$,$r_2$ 用密钥 $v_3.k$ 加密,可被 C 访问;$r_3$,$r_4$ 用密钥 $v_7.k$ 加密,可被 C、D 利用密钥和 token 转化解密密钥访问;$r_5$,$r_6$,$r_7$ 被密钥 $v_6.k$ 加密,可被 A、B、C 访问;$r_8$ 利用密钥 $v_8.k$ 加密,可被 A、B、C、E 访问.然后在服务端初始化相应的 Full SEL 或 Delta SEL,Full SEL 加密方案和 Bel 相同,若是 Delta SEL 则不用再次加密,初始化为 NULL.当分配给 D 用户 $r_5$ 资源时,Bel 层添加 D 用户拥有的密钥到加密 $r_5$ 的密钥之间的转化 token,这样 D 就可以访问 $r_5$ 了.但是加密 $r_5$ 的密钥也用来加密 $r_6$ 和 $r_7$,故在 Sel 层需要做访问控制限制 D 用户访问 $r_6$,$r_7$.Delta SEL 在 SEL 层对 $r_6$,$r_7$ 重新加密,并添加 A、B、C 用户的密钥到加密密钥之间的转化 token;而 D 用户没有可转化的 token,则访问不到 $r_6$,$r_7$.而 Full SEL 在 SEL 层首先添加 A、B、C 密钥到 $r_6$,$r_7$ 密钥的转化 token,再添加 A、B、C、D 密钥到 $r_5$ 密钥的转化 token,也就实现了上述的访问控制.当撤销 C 对 $r_2$ 的访问权时,由于 $r_2$ 现在没有人可访问,则 Full SEL 和 Delta SEL 都是产生新的 Sel 层密钥来加密 $r_2$,这样所有用户都访问不了 $r_2$ 了.

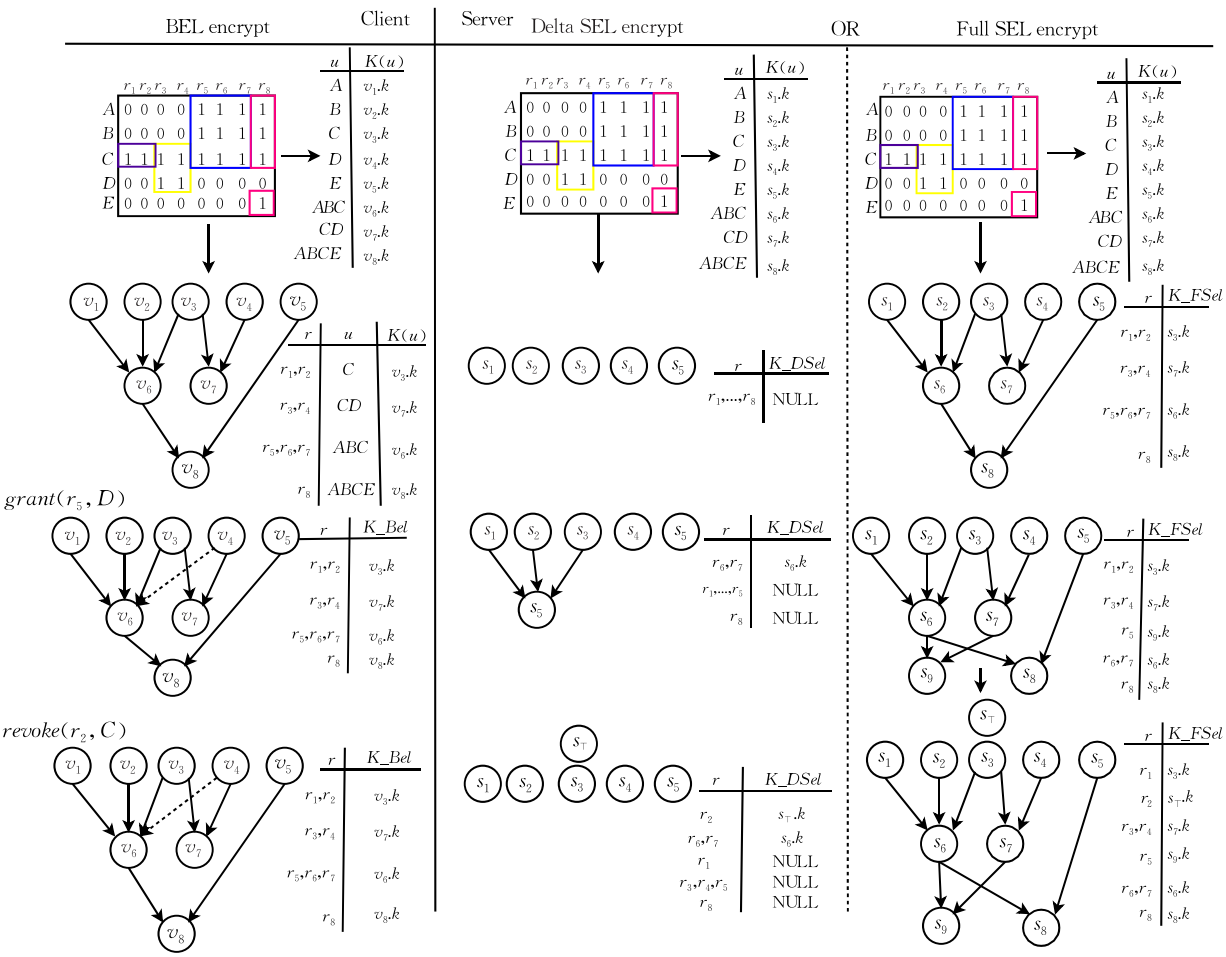



图 14 Over-encryption 密文访问控制

Over-encryption 使用两层加密和密钥转换的思想解决多用户密文访问控制问题,但其首先要在客户端计算资源和根据用户的对应关系初始化加密,实施起来具有一定的限制.针对多用户密文访问控制和密钥分发管理问题,研究人员已经做了大量的研究工作,其中基于属性的加密(Attribute-Based Encryption,ABE)是一个热点研究方向.基于属性加密由 Sahai 等人[71]提出,将数据加密与用户属性相关联,每个

用户拥有不同的属性，数据拥有者根据用户属性制定数据加密策略加密数据生成密文，相应地，只有用户密钥包含的属性满足密文要求的属性时，用户才能解密密文数据. 不同用户的密钥往往是由可信的密钥发布者根据用户属性生成，与不同的随机多项式相关，从而可以有效防范合谋攻击，即不同用户联合各自拥有的不同属性以访问原来都不具有访问权限的资源. 利用秘密分享方案[72]，当用户的身份属性与访问控制的属性之间的交集大于一定阈值时用户即可解密密文，由于阈值在初始化系统时需要确定，具有一定的局限性. Goyal 等人[73]提出 KP-ABE (Key Policy Attribute Based Encryption)，数据由用户属性加密，而用户的密钥与任意单调的树形访问结构相关. 但这种方案加密者只能对密文赋予一组相关的属性，不能控制可以访问自己加密数据的用户. Bethencourtet 等人[74]提出 CP-ABE(Ciphertext Policy Attribute Based Encryption)，使加密者可以控制拥有什么属性的用户可以访问密文数据，用户的密钥与一组属性相关，而密文与一个基于一组属性的访问控制树相关，但不能实现完全授权.

针对细粒度的访问控制和完全授权问题，Wang 等人[75]结合 HIBE(Hierarchical Identity Based Encryption)[76]系统和 CP-ABE 系统提出支持分层的密文访问控制的 HABE(Hierarchical Attribute Based Encryption)系统，实现完全授权，但不能表现分层的属性集合. HASBE[77](Hierarchical Attribute Set Based Encryption)方案扩展了基于属性集的加密方案，支持复合属性. 针对用户撤销问题，Wang 等人[75]赋予云服务商更多的计算任务从而实现可扩展的用户撤销机制. HASBE[77]为访问到期时间采用多重赋值，比现有方案更有效地处理用户撤销. Sieve[78]结合密钥同态首先在云端使用不同密钥重新加密用户数据，然后新数据密钥及数据指针等以用户属性加密替换原来的元数据块，撤销解密密钥且不会泄露新密钥. 针对多授权中心及隐私保护问题，Ruj 等人[79]在云存储信息之前验证用户的真伪，不需要知道用户的身份，并采取分散的认证和访问控制. Lewko 等人[80]设计支持多授权方的 ABE 系统，用户通过自己身份 ID 标志私钥的唯一性，预防多授权方联合恢复出用户的私钥窃取用户数据. Rouselakis 等人[81]和 Han 等人[82]设计的方案中不存在唯一授权中心，而是有不同的授权中心分发密钥给用户，每个授权中心负责一组特定属性的授权密钥分发. Zhang 等人[83]使用两个授权部门，任何一方都不能独立地解密密文. Jung 等人[84]提出半匿名权限控制方案 AnonyControl，分散了中央权威以限制身份泄漏，不仅实现文件访问控制，还实现了文件访问控制的特权控制，可以细粒度地管理数据的所有操作权限. Veluru 等人[85]提出任何属性权威(AA)可以独立加入和离开系统，而无需与现有的 AA 合作. 用户从多授权方获得用户密钥，保护用户的隐私.

### 5.2.2　硬件层配合客户端加密

如图 15 所示，PasS(Privacy as a Service)[48]基于硬件的方法，用户用自己的密钥对数据加密后传到云端，并对云端传回的密文数据进行解密获得明文，从而有效预防云端的内部威胁.

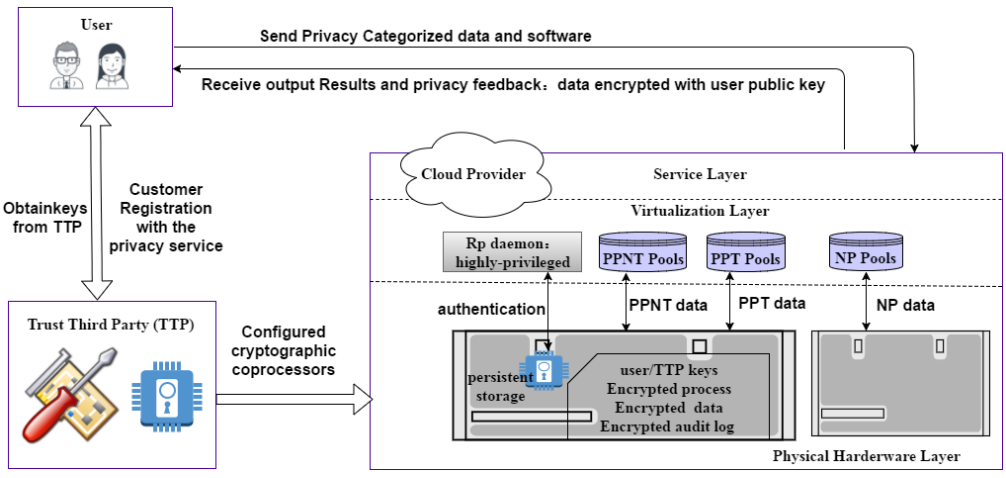



图 15　PasS 系统模型

PasS 云端加解密数据利用密码协处理器，即一个用于加解密的计算机芯片，并具有防窜改的特性. 由可信第三方对注册的用户分发密钥，并把密码协处理器配置在云平台物理硬件上. 密码协处理器里面有专门的存储单元存储用户密钥、可信第三方密钥、用户隐私程序数据等，用户密钥用于加解密用户

数据,第三方密钥用于执行可信第三方传来的加密的命令.云平台上有个特权程序和密码协处理器交互,确保用户的隐私程序和数据在密码协处理器可信的存储单元上执行,数据加密存储在可信的资源池中,确保不被云平台人员窃取.特权程序还对用户程序的执行进行审计并加密反馈给用户,使用户了解自己程序在云平台上的执行是可信的.这种方案需要购买硬件,且需要在云平台上作配置,并需要客户端程序配合,不利于实施.

## 6 总结与展望

本文总结分析了云计算模式下特有的内部威胁,并通过真实攻击实例说明云平台确实可能遭受的内部威胁.通过梳理国内外相关研究文献,本文针对云计算模式内部威胁问题及其应对做了较为全面的介绍和讨论,从人员评价与行为分析、云管理权限划分和执行时验证、用户可控的数据加密这 3 个方面进行综述,并认为进一步的研究重点应解决以下问题:

第一,怎样把人员评价技术更好地应用到实际中来,并提高人员行为分析的时效性和有效性.

怎样实现在不侵犯用户隐私的前提下,使人员评价技术和现有成熟技术相结合,才能更有效地防范内部威胁问题.用户行为分析可以有效地验证用户是否是真正的用户,但需要收集大量用户历史操作数据,往往存在一定的误报率或漏报率.其本身为被动防御,怎样提高拦截的时效性也是研究的重点.如果用户本身想要作恶,且操作极其隐蔽,则可能逃过检测,需要结合其他技术如数据加密来保护用户隐私.

第二,提高云管理权限划分和执行时验证的适用性和可操作性.

云管理权限划分和执行时验证机制往往需要修改虚拟机管理器,需要可信机制验证系统行为,往往机制比较复杂,与现有的公有云平台不兼容.未来的研究重点是如何在不改变虚拟机管理器及虚拟机的前提下,采用带外方式监控虚拟机及程序状态,并可对恶意攻击主动预警.

第三,提高基于用户可控的数据加密的计算性能和计算功能.

加密通常作为一种手段保护用户数据的隐私,但需要妥善管理密钥不被窃取,用户可控的数据加密可有效保护用户数据和密钥,但势必会影响云端数据计算功能,如数据访问控制问题和数据计算问题.未来用户数据加密的研究重点是怎么有效地对密文数据作访问控制和数据计算,同时不会引入过大的计算开销和复杂的机制,提高可用性.

另外,预防云计算模式内部威胁还应包括其他技术,如管理员权限细粒度管控、云平台安全机制防护、云平台操作审计与回溯等措施,做到多措施预防,及早发现,出现后及时应对,将风险降到最低.通过大量分析实践,我们认为以下几点是重要的云安全防护关键技术,包括:

(1)云系统资源访问入口严格管控

所有访问云平台内的资源或数据要经过统一的入口中转,如访问具体云平台、虚拟机或网络设备等,在入口经过认证后方能单点登录到相关的资源或数据,从而避免访问认证上的混乱,做到统一管控,防止内部人员绕开认证系统实施恶意行为.

(2)云系统资源操作留痕

登录云系统资源访问入口后所有对资源或数据的操作都要详细记录,并能做到操作记录审计回放.对用户的操作数据进行处理,通过心理学和行为分析检测到恶意行为,并能对可疑行为进行追溯.

(3)云系统重要数据加密

通过加密保障用户的数据不被恶意内部人员窃取,针对访问控制问题,建立可信的第三方服务,在利用原有系统访问控制的基础上,用户可对授权的不同密钥密文数据向可信第三方认证申请获得明文.同时研究密文搜索技术,从而节约带宽资源,提高效率.

**致 谢** 在此,我们向对本文的工作给予支持和建议的同行表示衷心的感谢!

## 参 考 文 献

[1] Claycomb W R, Nicoll A. Insider threats to cloud computing: Directions for new research challenges//Proceedings of the 2012 IEEE 36th Annual Computer Software and Applications Conference (COMPSAC). Lzmir, Turkey, 2012: 387-394

[2] Cappelli D M, Moore A P, Trzeciak R F. The CERT Guide to Insider Threats: How to Prevent, Detect, and Respond to Information Technology Crimes (theft, Sabotage, Fraud). New Jersey, USA: Addison-Wesley Professional, 2012

[3] Cappelli D M, Moore A P, Trzeciak R F, et al. Common sense guide to prevention and detection of insider threats 3rd edition-version 3.1. Pittsburgh, Pennsylvania: Camegie Mellon Software Engineering Institute, 2009

[4] Silowash G J, Cappelli D M, Moore A P, et al. Common Sense Guide to Mitigating Insider Threats 4th Edition. Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, Technical Note CMU/SEI-2012-TR-012, 2012

[5] Montelibano J, Moore A. Insider threat security reference architecture//Proceedings of the Hawaii International Conference on System Sciences. Hawaii, USA, 2012: 2412-2421

[6] Legg P, Moffat N, Nurse J R, et al. Towards a conceptual model and reasoning structure for insider threat detection. Journal of Wireless and Mobile Networks Ubiquitous Computing and Dependable Applications, 2013, 4(4): 20-37

[7] Bishop M, et al. Insider threat detection by process analysis //Proceedings of the 2014 Workshop on Research for Insider Threat (WRIT). San Jose, USA, 2014: 251-264

[8] Legg P A, Buckley O, Goldsmith M, et al. Caught in the act of an insider attack: Detection and assessment of insider threat//Proceedings of the IEEE International Symposium on Technologies for Homeland Security (HST). Waltham, USA, 2015: 1-6

[9] Flynn L, Porter G, DiFatta C. Cloud service provider methods for managing insider threats: Analysis phase II, Expanded analysis and recommendations. Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania: Technical Note CMU/SEI-2013-TN-030, 2014

[10] Duncan A J, Creese S, Goldsmith M. Insider attacks in cloud computing//Proceedings of the IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom). Liverpool, UK, 2012: 857-862

[11] Bouche J, Kappes M. Attacking the cloud from an Insider perspective//Proceedings of the International Conference on Cloud Technologies and Applications(CloudTech). Marrakech, Morocco, 2015: 175-180

[12] Rocha F, Correia M. Lucy in the sky without diamonds: Stealing confidential data in the cloud//Proceedings of the 2011 IEEE/IFIP 41st International Conference on Dependable Systems and Networks Workshops (DSN-W). Hong Kong, China, 2011: 129-134

[13] Sanzgiri A, Dasgupta D. Classification of insider threat detection techniques//Proceedings of the 11th Annual Cyber and Information Security Research Conference. Oak Ridge, USA, 2016: 25

[14] Mahajan A, Sharma S. The malicious insiders threat in the cloud. International Journal of Engineering Research and General Science, 2015, 3(2): 245-256

[15] Garfinkel T, Rosenblum M. A virtual machine introspection based architecture for intrusion detection//Proceedings of the Network & Distributed Systems Security Symposium. San Diego, USA, 2003: 191-206

[16] Greitzer F L, Hohimer R E. Modeling human behavior to anticipate insider attacks. Journal of Strategic Security, 2011, 4(2): 25

[17] Turner J T, Gelles M G. Threat Assessment: A Risk Management Approach. London, UK: Routledge, 2012

[18] Vernon P A, Villani V C, Vickers L C, et al. A behavioral genetic investigation of the Dark Triad and the Big 5. Personality and Individual Differences, 2008, 44(2): 445-452

[19] Greitzer F L, Noonan C F, Kangas L J, Dalton A C. Identifying at-risk employees: A behavioral model for predicting potential insider threats. Richland, USA: Pacific Northwest National Laboratory, 2010

[20] Moises G. Bayesian network classifiers. Wiley Encyclopedia of Operations Research & Management Science, 2011, 29(2-3): 598-605

[21] Greitzer F L, Kangas L J, Noonan C F, et al. Identifying at-risk employees: Modeling psychosocial precursors of potential insider threats//Proceedings of the 2012 45th Hawaii International Conference on System Sciences (HICSS). Maui, USA, 2012: 2392-2401

[22] Nurse J R C, Buckley O, Legg P A, et al. Understanding insider threat: A framework for characterising attacks// Proceedings of the Workshop on Research for Insider Threat. San Jose, USA, 2014: 214-228

[23] Hashem Y, Takabi H, GhasemiGol M, et al. Inside the mind of the insider: Towards insider threat detection using psychophysiological signals. Journal of Internet Services and Information Security (JISIS), 2016, 6(1): 20-36

[24] Tausczik Y R, Pennebaker J W. The psychological meaning of words: LIWC and computerized text analysis methods. Journal of Language and Social Psychology, 2010, 29(1): 24-54

[25] Mayhew M, Atighetchi M, Adler A, et al. Use of machine learning in big data analytics for insider threat detection// Proceedings of the 2015 IEEE Military Communications Conference (MILCOM). Tampa, USA, 2015: 915-922

[26] Choi S, Zage D. Addressing insider threat using "where you are" as fourth factor authentication//Proceedings of the 46th Annual IEEE International Carnahan Conference on Security Technology. Boston, Massachusetts, USA, 2012: 147-153

[27] Sun H M, Chen Y H, Lin Y H. oPass: A user authentication protocol resistant to password stealing and password reuse attacks. IEEE Transactions on Information Forensics and Security, 2012, 7(2): 651-663

[28] Kent A D, Liebrock L M, Neil J C. Authentication graphs: Analyzing user behavior within an enterprise network. Computers & Security, 2015, 48: 150-166

[29] Gavai G, Sricharan K, Gunning D, et al. Detecting insider threat from enterprise social and online activity data// Proceedings of the 7th ACM Conference on Computer and Communications Security International Workshop on Managing Insider Security Threats. Denver, USA, 2015: 13-20

[30] Myers J, Grimaila M R, Mills R F. Towards insider threat detection using web server logs//Proceedings of the 5th Annual Workshop on Cyber Security and Information Intelligence Research: Cyber Security and Information Intelligence Challenges and Strategies. Knoxville, USA, 2009: 54

[31] Hsieh C H, Lai C M, Mao C H, et al. AD2: Anomaly detection on active directory log data for insider threat monitoring//Proceedings of the 2015 International Carnahan Conference on Security Technology (ICCST). Taiwan, China, 2015: 287-292

[32] Kim H S, Cha S D. Empirical evaluation of SVM-based masquerade detection using UNIX commands. Computers & Security, 2005, 24(2): 160-168

[33] Zheng N, Paloski A, Wang H. An efficient user verification system via mouse movements//Proceedings of the 18th ACM Conference on Computer and Communications Security. Chicago, USA, 2011: 139-150

[34] Sae-Bae N, Ahmed K, Isbister K, et al. Biometric-rich gestures: A novel approach to authentication on multi-touch devices//Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. Austin, USA, 2012: 145-160

[35] De Luca A, Hang A, Brudy F, et al. Touch me once and i know it's you!: Implicit authentication based on touch screen patterns//Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. Austin, USA, 2012: 987-996

[36] Frank M, Biedert R, Ma E, et al. Touchalytics: On the applicability of touchscreen input as a behavioral biometric for continuous authentication. IEEE Transactions on Information Forensics & Security, 2013, 8(1): 136-148

[37] Eberz S, Rasmussen K B, Lenders V, et al. Preventing lunchtime attacks: Fighting insider threats with eye movement biometrics//Proceedings of the Network and Distributed System Security Symposium (NDSS). San Diego, USA, 2015: 1-13

[38] Eberz S, Rasmussen K B, Lenders V, et al. Looks like eve: Exposing insider threats using eye movement biometrics. ACM Transactions on Privacy and Security, 2016, 19(1): 1

[39] Brdiczka O, Liu J, Price B, et al. Proactive insider threat detection through graph learning and psychological context//Proceedings of the IEEE CS Security and Privacy Workshops. San Francisco, USA, 2012: 142-149

[40] Butt S, Lagar-Cavilla H A, Srivastava A, et al. Self-service cloud computing//Proceedings of the 2012 ACM Conference on Computer and Communications Security. Raleigh, USA, 2012: 253-264

[41] Zhang F, Chen J, Chen H, et al. CloudVisor: Retrofitting protection of virtual machines in multi-tenant cloud with nested virtualization//Proceedings of the 23rd ACM Symposium on Operating Systems Principles (SOSP). Cascais, Portugal, 2011: 203-216

[42] Chen X, Garfinkel T, Lewis E C, et al. Overshadow: A virtualization-based approach to retrofitting protection in commodity operating systems//Proceedings of the 13th International Conference on Architectural Support for Programming Languages and Operating Systems. New York, USA, 2008: 2-13

[43] Hofmann O S, Kim S, Dunn A M, et al. Inktag: Secure applications on an untrusted operating system//Proceedings of the 18th International Conference on Architectural Support for Programming Languages and Operating Systems. New York, USA, 2013: 265-278

[44] Popa R A, Redfield C, Zeldovich N, et al. CryptDB: Protecting confidentiality with encrypted query processing//Proceedings of the 23rd ACM Symposium on Operating Systems Principles. Cascais, Portugal, 2011: 85-100

[45] Popa R A, Stark E, Valdez S, et al. Building web applications on top of encrypted data using Mylar//Proceedings of the 11th USENIX Symposium on Networked Systems Design and Implementation. Seattle Washington, USA, 2014: 157-172

[46] He W, Akhawe D, Jain S, et al. Shadowcrypt: Encrypted web applications for everyone//Proceedings of the 2014 ACM Special Interest Group on Security, Audit and Control. Scottsdale Arizona, USA, 2014: 1028-1039

[47] Di Vimercati S D C, Foresti S, Jajodia S, et al. Over-encryption: Management of access control evolution on outsourced data//Proceedings of the 33rd International Conference on Very Large Data Bases. Vienna, Austria, 2007: 123-134

[48] Itani W, Kayssi A, Chehab A. Privacy as a service: Privacy-aware data storage and processing in cloud computing architectures//Proceedings of the 8th IEEE International Conference on Dependable, Autonomic and Secure Computing. Chengdu, China, 2009: 711-716

[49] Song D X, Wagner D, Perrig A. Practical techniques for searches on encrypted data//Proceedings of the IEEE Symposium on Security & Privacy. Berkeley, USA, 2000: 44-55

[50] Boldyreva A, Chenette N, Lee Y, et al. Order-preserving symmetric encryption//Proceedings of the 2009 International Conference on the Theory and Applications of Cryptographic Techniques. Cologne, Germany, 2009: 224-241

[51] Agrawal R, Kiernan J, Srikant R, et al. Order preserving encryption for numeric data//Proceedings of the 2004 ACM Special Interest Group on Management of Data. Paris, France, 2004: 563-574

[52] Wang C, Cao N, Li J, et al. Secure ranked keyword search over encrypted cloud data//Proceedings of the 2010 IEEE 30th International Conference on Distributed Computing Systems (ICDCS). Genoa, Italy, 2010: 253-262

[53] Mavroforakis C, Chenette N, O'Neill A, et al. Modular order-preserving encryption, revisited//Proceedings of the 2015 ACM Special Interest Group on Management of Data. Melbourne, Australia, 2015: 763-777

[54] Boldyreva A, Chenette N, O'Neill A. Order-preserving encryption revisited: Improved security analysis and alternative solutions//Proceedings of the 2011 Advances in Cryptology Conference. Santa Barbara, USA, 2011: 578-595

[55] Liu Z, Chen X, Yang J, et al. New order preserving encryption model for outsourced databases in cloud environments. Journal of Network and Computer Applications, 2016, 59(C): 198-207

[56] Popa R A, Li F H, Zeldovich N. An ideal-security protocol for order-preserving encoding//Proceedings of the 2013 IEEE Symposium on Security and Privacy. San Francisco, USA, 2013: 463-477

[57] Kerschbaum F. Frequency-hiding order-preserving encryption //Proceedings of the 22nd ACM Conference on Computer and Communications Security. Denver Colorado, USA, 2015: 656-667

[58] Gentry C. Fully homomorphic encryption using ideal lattices. Annual ACM Symposium on Theory of Computing, 2009, 9(4): 169-178

[59] Paillier P. Public-key cryptosystems based on composite degree residuosity classes. Lecture Notes in Computer Science, 1999, 5(1592): 223-238

[60] Boneh D, Di Crescenzo G, Ostrovsky R, et al. Public key encryption with keyword search//Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques. Berlin Heidelberg, Germany, 2004: 506-522

[61] Liu Q, Wang G, Wu J. Secure and privacy preserving keyword searching for cloud storage services. Journal of Network and Computer Applications, 2012, 35(3): 927-933

[62] Goh E J. Secure Indexes. International Association for Cryptologic Research Cryptology ePrint Archive, 2003: 216

[63] Curtmola R, Garay J, Kamara S, et al. Searchable symmetric encryption: Improved definitions and efficient constructions. Journal of Computer Security, 2011, 19(5): 79-88

[64] Sun W, Wang B, Cao N, et al. Privacy-preserving multi-keyword text search in the cloud supporting similarity-based ranking//Proceedings of the 2013 ACM Symposium on Information, Computer and Communications Security. Hangzhou, China, 2013: 71-82

[65] Cao N, Wang C, Li M, et al. Privacy-preserving multi-keyword ranked search over encrypted cloud data. IEEE Transactions on Parallel and Distributed Systems, 2014, 25(1): 222-233

[66] Xia Z, Wang X, Sun X, et al. A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data. IEEE Transactions on Parallel and Distributed Systems, 2016, 27(2): 340-352

[67] Yang Y, Lu H, Weng J. Multi-user private keyword search for cloud computing//Proceedings of the 2011 IEEE 3rd International Conference on Cloud Computing Technology and Science. Washington, USA, 2011: 264-271

[68] Sedghi S, Van Liesdonk P, Nikova S, et al. Searching keywords with wildcards on encrypted data//Proceedings of the International Conference on Security and Cryptography for Networks. Amalfi, Italy, 2010: 138-153

[69] Wang P, Wang H, Pieprzyk J. Threshold privacy preserving keyword searches//Proceedings of the International Conference on Current Trends in Theory and Practice of Computer Science. Nový Smokovec, Slovakia, 2008: 646-658

[70] Lau B, Chung S, Song C, et al. Mimesis aegis: A mimicry privacy shield—A system's approach to data privacy on public cloud//Proceedings of the 23rd USENIX Security Symposium. San Diego, USA, 2014: 33-48

[71] Sahai A, Waters B. Advances in Cryptology—Eurocrypt: Fuzzy Identity-Based Encryption. Berlin Heidelberg: Springer, 2005

[72] Shamir A. How to share a secret. Communications of the ACM, 1979, 22(11): 612-613

[73] Goyal V, Pandey O, Sahai A, et al. Attribute-based encryption for fine-grained access control of encrypted data//Proceedings of the 13th ACM Conference on Computer and Communications Security. Virginia Alexandria, USA, 2006: 89-98

[74] Bethencourt J, Sahai A, Waters B. Ciphertext-policy attribute-based encryption//Proceedings of the 2007 IEEE Symposium on Security and Privacy. Oakland, USA, 2007: 321-334

[75] Wang G, Liu Q, Wu J. Hierarchical attribute-based encryption for fine-grained access control in cloud storage services//Proceedings of the 17th ACM Conference on Computer and Communications Security. Chicago Illinois, USA, 2010: 735-737

[76] Gentry C, Silverberg A. Advances in Cryptology—ASIACRYPT: Hierarchical ID-Based Cryptography. Berlin Heidelberg: Springer, 2002

[77] Wan Z, Liu J, Deng R H. HASBE: A hierarchical attribute-based solution for flexible and scalable access control in cloud computing. IEEE Transactions on Information Forensics & Security, 2012, 7(2): 743-754

[78] Wang F, Mickens J, Zeldovich N, et al. Sieve: Cryptographically enforced access control for user data in untrusted clouds//Proceedings of the 12th USENIX Symposium on Networked Systems Design and Implementation. Savannah Georgia, USA, 2016: 611-626

[79] Ruj S, Stojmenovic M, Nayak A. Privacy preserving access control with authentication for securing data in clouds//Proceedings of the 2012 IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing. Uttawa, Canada, 2012, 25: 556-563

[80] Lewko A, Waters B. Decentralizing attribute-based encryption //Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques. Tallinn, Estonia, 2011: 568-588

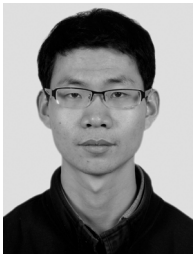
[81] Rouselakis Y, Waters B. Efficient statically-secure large-universe multi-authority attribute-based encryption//Proceedings of the International Conference on Financial Cryptography and Data Security. San Juan, Puerto Rico, 2015: 315-332

[82] Han J, Susilo W, Mu Y, et al. Improving privacy and security in decentralized ciphertext-policy attribute-based encryption. IEEE Transactions on Information Forensics and Security, 2015, 10(3): 665-678

[83] Zhang X, Jin C, Wen Z, et al. Attribute-based encryption without key escrow//Proceedings of the International Conference on Cloud Computing and Security. Nanjing, China, 2015: 74-87

[84] Jung T, Li X Y, Wan Z, et al. Control cloud data access privilege and anonymity with fully anonymous attribute-based encryption. IEEE Transactions on Information Forensics and Security, 2015, 10(1): 190-199

[85] Rahulamathavan Y, Veluru S, Han J, et al. User collusion avoidance scheme for privacy-preserving decentralized key-policy attribute-based encryption. IEEE Transactions on Computers, 2016, 65(9): 2939-2946



**WANG Guo-Feng**, born in 1988, Ph. D. candidate. His current research interest is cloud security.

**LIU Chuan-Yi**, born in 1982, associate professor. His research interests include cloud computing and cloud security, data security and data protection.

**PAN He-Zhong**, born in 1991, Ph. D. candidate. His current research interest is cloud security.

**FANG Bin-Xing**, born in 1960, professor, member of Chinese Academy of Engineering. His current research interests include information and network security, content security.

**Background**

This paper belongs to Cloud Computing and Cloud Security area. Trustworthiness is a critical factor for the large-scale use of cloud services, and this paper focuses on the insider threat of cloud execution environment. Customers' data stored in the cloud, how to protect the security and confidentiality of customer's data is a matter of major concern in the cloud computing model. Security policies installed by the cloud provider against malicious external attacks, but against malicious insiders who has special authority to steal or modify user's data resources, many cloud companies lack effective measures.

This paper describes the peculiar internal threats to the cloud environment, and explains that cloud computing model is indeed vulnerable to internal threats through real case. Combined with domestic and foreign research literature, against the internal threat of cloud computing model, this paper summarizes three typical preventive measures, including personnel evaluation and prediction based on the behavior of sociology and psychology, behavior-based early warning mechanism with data collection and analysis of behavior, based on user-controllable data encryption security mechanisms. Finally, we point future research directions.

This paper is supported by the National High Technology Research and Development Program (863 Program) of China under Grant No. 2015AA016001, the Production-Study-Research Cooperation Project in Guangdong Province under Grant No. 2016B090921001, the Innovation Projects in Shandong Province under Grant No. 2014ZZCX03411, the National Natural Science Foundation of China under Grant No. 61370068. These projects aim to promote the healthy and development of cloud computing and reach the international advanced level of the cloud security field.