OpenStack认证安全问题研究

Research on OpenStack Identification Security Issues

熊　微[1]，房秉毅[1]，张云勇[1]，吴　俊[2]，李素粉[1]（1. 中国联通研究院，北京 100048；2. 北京邮电大学，北京 100876）
Xiong Wei[1]，Fang Bingyi[1]，Zhang Yunyong[1]，Wu Jun[2]，Li Sufen[1]（1. China Unicom Research Institute，Beijing 100048，China；2. Beijing University of Posts and Telecommunications，Beijing 100876，China）

摘　要：
选择Keystone组件作为OpenStack安全问题研究的切入点，分析了该组件相关的安全问题。首先简要介绍了OpenStack的主要功能组件及组件间的交互关系，详细描述了Keystone的对象模型，并深入研究了其认证机制。在充分分析了Keystone运行机制的基础上，提出了Keystone中存在的安全问题，并给出了相应的改进方案。

关键词：
云计算；安全；OpenStack
中图分类号：TP939
文献标识码：A
文章编号：1007-3043（2014）07-0021-05

Abstract：
Selecting Keystone component as the beginning of researching OpenStacek security issues, it analyzes the security issues on this component. It firstly describes the major functional components of OpenStack and the interactive relation between the components, presents object model of Keystone in detail, and researches its identification mechanism. Based on the analysis of Keystone operation mechanism, it presents the security issues on Keystone and gives the relative improvement solution.

Keywords：
Cloud computing; Security; OpenStack

0 前言

OpenStack是一个由Rackspace和NASA发起、全球开发者共同参与的开源项目，旨在打造容易部署、功能丰富且易于扩展的云计算平台，它可以为用户提供类似Amazon Web Service的服务。到目前为止，OpenStack已经发布了8个版本，其系统功能在版本演进的过程中得到了不断的完善，以安全功能方面的认证授权功能为例，在Essex版本之前，OpenStack的各个子项目如Nova、Swift均使用各自独立的认证系统，Swift组件中使用一种名为“swauth”的身份验证和用户授权方法，从Essex版本之后开始全面支持Keystone。OpenStack作为开源云平台，虽然有良好的跟踪记录，但是仍存在漏洞和脆弱性，其安全问题也是作为OpenStack发展过程中需重点考虑的问题之一。OpenStack安全组织及相关研究人员在OpenStack安全问题研究方面进行了积极探索。OpenStack安全组在其发布的安全指南[1]中明确指出了OpenStack的安全范围，该指南指出了OpenStack的用户和实施者在使用过程中存在各种安全痛点，但是并没有直接解决OpenStack中的安全漏洞。Cooper J D[2]以OpenStack作为研究案例，对云平台的安全特征进行了研究，识别与授权和身份管理、数据管理相关的安全漏洞，并针对所发现的问题，提供不同的建议去解决这些问题。Ristov S[3]分析了私有或公有网络的安全漏洞，针对虚拟机实例和OpenStack云节点，提出了3个假设并验证其正确性。相关的OpenStack安全方面的研究工作主要是从云平

基金项目：国家自然科学基金资助项目（71172134）；国家科技重大基金资助项目（2012ZX03002001-002、2013ZX03002004-002、2013ZX03002003-005）
收稿日期：2014-05-29

台的整体安全性角度出发,给出部分组件中存在的漏洞,提出改进的方向或建议,并没有针对某个特定的功能组件展开深入的研究。

1 OpenStack系统框架

OpenStack是一个纯IaaS交付模型,任何组织均可以通过OpenStack基于标准化的硬件设施创建和提供云计算服务。OpenStack的主要7个核心组件关系如图1所示。

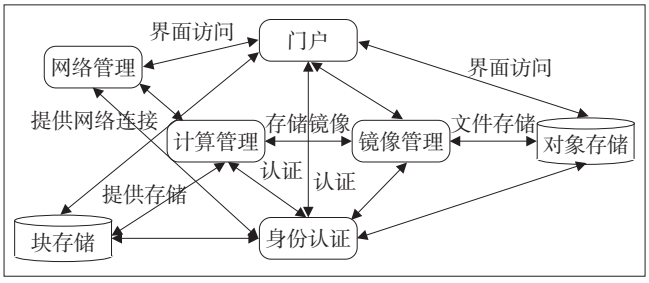

图1 OpenStack核心系统架构

计算管理(Nova):提供虚拟主机,对应亚马逊的EC2,包括虚拟机、弹性云硬盘等。通过虚拟化技术(如KVM、Xen、VMware Esxi等实现计算、网络、存储等资源池的构建及应用),将计算能力通过虚拟机的方式交付用户。

镜像管理(Glance):提供虚拟磁盘镜像的目录分类管理以及镜像库存储管理。

身份认证(Keystone):为OpenStack所有的系统提供统一的授权和身份验证。

网络管理(Quantum):实现了虚拟机的网络资源管理,包括网络连接、子网IP管理、L3的公网映射、后续的负载均衡等。Folsom版中的Quantum对应于Essex版中Nova中的nova-network。

对象存储(Swift):对应亚马逊的S3,通过简单key/value的方式实现对象文件的存储读取。

块存储(Cinder):与亚马逊的EBS弹性云硬盘类似,实现了对块存储的管理,为虚拟机提供云硬盘服务。在Essex版本中是由nova-volume实现块存储的管理,在Folsom版本之后,独立新增的Cinder项目增强了块存储方面的管理能力。

门户(Horizon):基于OpenStack API接口开发的Web呈现。

从图1可以看出OpenStack所有模块系统之间都是通过标准的API接口进行服务调用,因此所有涉及服务API的调用都脱离不了Keystone。Keystone为OpenStack所有的系统提供统一的授权和身份认证服务,负责身份验证、服务规则和服务令牌的功能,它实现了OpenStack的Identity API,其他服务需要通过Keystone来注册其服务的Endpoint(服务访问的URL),任何服务之间相互的调用,都需要经过Keystone进行身份验证,来获得目标服务的Endpoint,从而找到目标服务。Keystone作为OpenStack身份验证的中央安全中心,其安全问题将直接影响着其他模块子系统的安全。下面就对Keystone的运行机制及其安全问题进行分析和介绍。

2 Keystone运行机制分析

Keystone使用基于Token/PKI的身份认证机制提供单点登录认证和访问控制服务。身份认证服务提供2个功能:用户管理、追踪用户及其许可和服务目录,提供可用服务的目录及其API端点。OpenStack所有组件都依赖于Keystone提供3A(Account、Authentication、Authorization)服务。除了3A之外,Keystone还对外提供服务目录(Service Catalog)服务,用户(无论是Dashboard还是API Client)都需要访问Keystone获取服务列表以及每个服务的地址(OpenStack中称为Endpoint)。OpenStack身份管理是基于标准PKI功能的令牌,允许客户端无须身份服务呼叫便可获取离线令牌认证。OpenStack身份管理还针对多租户环境提供了更为系统化的管理能力,支持群组、角色扮演、角色接入控制(RBAC),赋予管理员更大的权力。

2.1 Keystone对象模型

在介绍Keystone对象模型之前,先要介绍一下Keystone中的几个基本概念。

a) 用户(User):数字代表了使用OpenStack云服务的人、系统或服务。身份验证服务会验证传入的请求。用户有可能被分配一个登录名和令牌来访问资源,并标识其为API的一个特定使用者,属于一个指定的域。同时,可以赋予用户角色,每一个用户域或用户项目都可以有一组角色。

b) 证书(Credential):保证数据只有能证明他们是谁的用户知道。例如,用户名和密码、用户名和API密钥或由身份认证服务提供的身份认证令牌。它也是与用户关联的认证凭据。一个用户可能有一个或多个证书,一个证书与某一个项目关联。

c) 令牌(Token):用于访问资源的任意字节的文本。每个令牌都有一个范围,该范围描述了这个令牌

所能访问的资源。一个令牌可以随时被取消，其有效性为有限的持续时间。Keystone通过验证一组由用户提供的证书来验证出入的请求，为了响应这些证书，Keystone为用户发布一个身份验证令牌，用户在随后的请求中需要提供该令牌。

d）组（Group）：表示一组拥有某权限的用户，属于一个指定的域。可以赋予组特定的角色，此时组内的用户会被赋予该角色所具有的权限。

e）域（Domain）：表示一组项目和用户的集合。每一个项目或用户只能属于一个域，但用户可以属于多个项目。域有命名空间的概念，即在一个命名空间内的名称是否是全局唯一。

f）租户/项目（Tenant/Project）：在OpenStack中表示一组资源（在Folsom版本中叫Tenant，在Grizzly版中叫Project），是一个用户分组或隔离资源/身份对象的容器，一个项目属于某一个域。

g）角色（Role）：表示一组项目或者域范围内所允许的操作。一个角色包含一组权利和特权。在认证服务中，分发给用户的令牌包含用户拥有的角色列表。用户所能调用的服务能表明用户所拥有角色的集合，以及每个角色能够访问的资源和操作。

上述的主要对象之间的关系模型如图2所示。

2.2 Keystone认证机制

OpenStack中身份认证分为2个部分：首先，用户进行初始化身份认证时，会为用户分配一个Token；然后使用该Token进行单点登录和委派验证。下面介绍OpenStack中Token的工作原理。OpenStack在G版之前使用的是UUID Token，在G版之后引入新的方法验证Token，这是因为PKI（Public Key Infrastructure）能够增强第一步中的安全性，同时也能提高第二步中的安全性和扩展性。

2.2.1 UUID Token

OpenStack在G版之前使用的是UUID Token，图3显示了Keystone是如何生成Token及用户是如何利用Token“签署”每一个后续API请求的过程。

基于提供的用户名/密码对（假设它在这个场景和图3中是正确的），生成1个UUID Token过程如下。

a）在Keystone的后端存储UUID Token。

b）将UUID Token的拷贝发送给用户。

c）用户缓存Token到本地。

d）UUID将会和用户发起的每个API调用一起传递。

e）基于每个用户的请求，API端点会将这个UUID传回给Keystone去验证。

f）Keystone会拿UUID与其授权后台匹配（检查UUID字符串和失效日期）。

g）Keystone会给API端点返回“成功”或“失败”消息。

从图3可知，对每个用户调用API端点需要使用Keystone服务执行在线验证。假设上千个用户执行VM listings、网络创建等，这样就会导致为Keystone服务扩大流量。

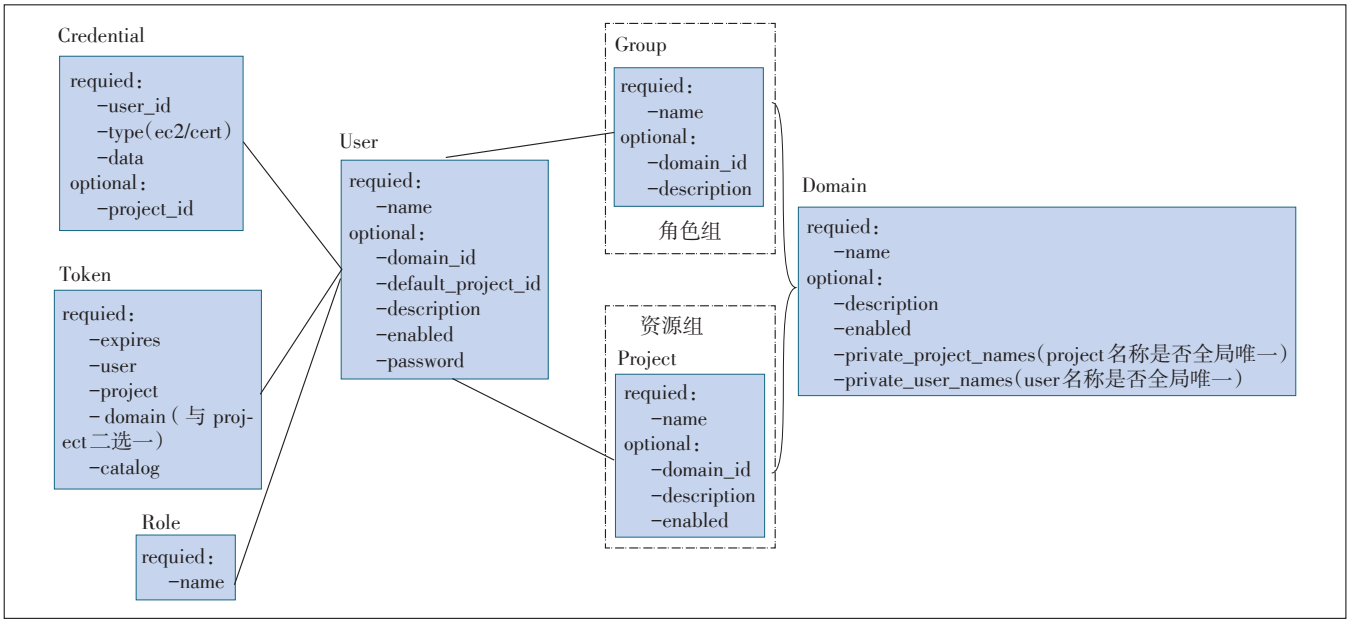

图2　Keystone对象关系模型

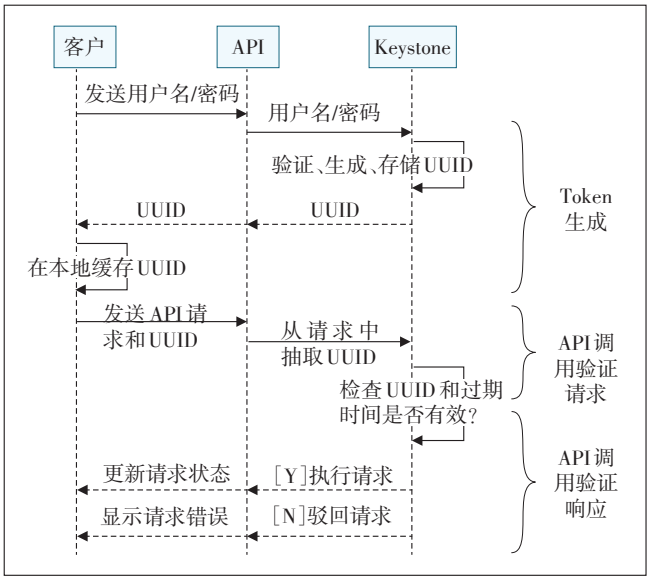

图3 UUID Token过程

2.2.2 PKI Token

OpenStack在Grizzly版中引入的新方法验证Token(见图4)。

概括地说,使用PKI Token,Keystone逐渐成为一个数字签名认证中心(CA)。它使用签名密钥和数字证书来签名用户的Token。每个API端点持有一份Keystone的拷贝,包括签名证书、撤销列表、CA数字签

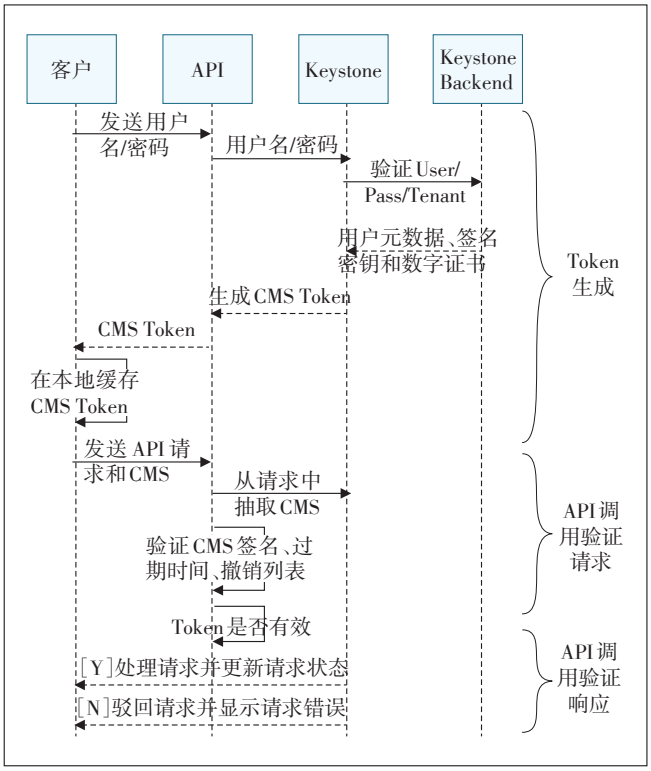

图4 PKI Token过程

名。API端点使用这些字节来验证用户的请求。没有必要为每个验证直接请求Keystone。签名Keystone放在用户令牌和Keystone的撤销列表。API端点使用以上数据离线地执行这个过程。

3 Keystone中的安全问题及改进方案

通过本文第2章中对Keystone安全机制的介绍和研究,本章主要是识别目前OpenStack认证机制中存在的问题,针对其进行分析,并给出了改进的方向。

3.1 Keystone中的安全问题

a) OpenStack使用Keystone作为身份验证的中央安全中心。身份验证机制是基于用户名和密码,以明文的形式发送,当身份验证成功后,Keystone服务器生成一个有效Token,终端用户通过它来获得服务。此外,所有的通信通过HTTP协议执行,因此可能会存在对用户名和密码、通信协议方面的攻击。

b) 即使在OpenStack的Gizzly版中,Keystone也并不提供在重复登录失败后限制登录的方法。重复失败登录有可能是在进行暴力攻击brute-force attacks,可以通过使用一个外部的身份验证系统。

c) OpenStack使用密码用于身份验证且以明文的形式发送。使用HTTP通信协议来进行授权,相比之下,没有HTTPS安全。生成的身份验证的Token不是一个证书,仅有32 bit长。此外,Token的生命周期很短,默认是24 h,这意味着用户在Token过期后需要重新进行身份认证。在Keystone中实施PKI,可以生成证书,其有效期会更长。在身份验证成功后且在Token的有效期内,每个服务都必须借助Keystone来确认Token的可靠性,这可能会引发扩展性方面的问题。32位Token生成的随机性仍然是个问题。作为一个单点故障,Keystone仍然是认证和授权的瓶颈。

3.2 Keystone安全改进方案

3.2.1 基于PKI扩展OpenStack弱认证

PKI可以在Keystone中单独实施,然而,撤销和可扩展性是很重要的方面。这意味着一个单独的证书颁发机构,例如需要“中央安全服务器”来处理认证请求,并为Keystone提供有效的证书。此外,在OpenStack中使用Keystone进行认证分两步:首先,需要用户名和密码、成功的认证、Keystone生成一个有效的Token;然后使用这个有效的Token来提供单点登录(SSO)和委托认证。实施PKI能够提高第一步中的安全性,可以通过提供签名认证和授权Token来增强第二步中的安全

性和可扩展性。

图5示出的是基于PKI的初始化认证。

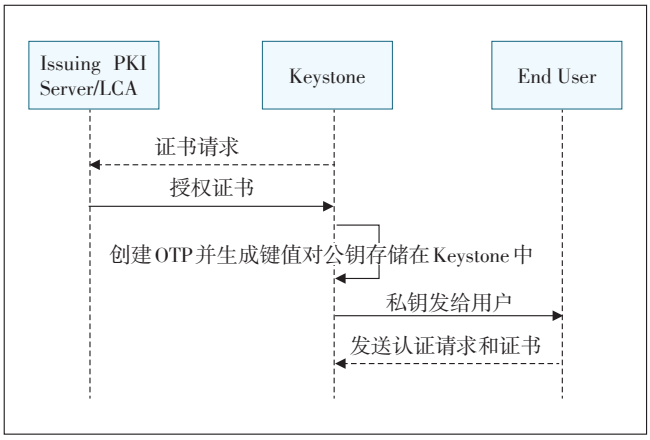

图5　基于PKI的初始化认证

3.2.2 代理授权

Keystone在线验证Token和租户需要用户的积极参与，但是通过实施PKI，可以提高包含认证和授权信息的Token的安全性和可扩展性。通过密码签名和Keystone的私钥对Token的信息内容加密，确保那些有Keystone公钥的服务才能解密信息。因此，也就减少了Token验证对网络的需求及去Keystone中获取角色信息的需求。整个方法包括将Token认证中间件修改为基于Token的CMS（Cryptographic Message Syntax）。这种代理授权的方法仅能用于OpenStack部署中。由于消息签名需要用户的私钥，而用户私钥并非由Keystone模块自己维护，当前的浏览器并不支持代表一个网站去签名消息，门户（Dashboard）需要的代理授权其实代表的是已登录系统的用户。这意味着用户必须首先使用他的证书登录到门户，然后使用Keystone Token进行委托授权。

图6示出的是认证和委托协议。

4 结束语

本文对Keystone安全机制的分析和研究是基于比较新的2个版本，即F版和G版。本文深入研究了OpenStack中的Keystone对象，对Keystone的安全研究着重于基于Token/PKI授权。通过研究其对象模型和认证机制，分析了Keystone中存在的安全问题，并给出相应的改进方案。

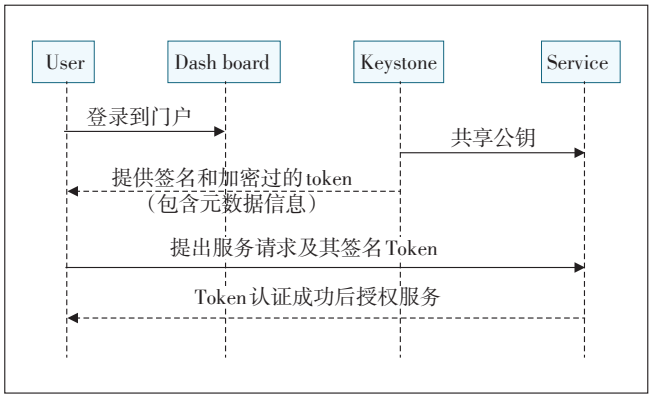

图6　认证和委托协议

参考文献：

[1] The OpenStack Security Group. OpenStack Security Guide[EB/OL]. [2013-11-20]. http://docs.openstack.org/security-guide/content/.

[2] Cooper J D. Analysis of security in cloud platforms using OpenStack as case study[EB/OL]. [2013-11-20]. http://brage.bibsys.no/xmlui/bitstream/handle/11250/137588/Cooper,%20John%20David%20Oppgave.pdf?sequence=1.

[3] Ristov S, Gusev M, Donevski A. OpenStack Cloud Security Vulnerabilities from Inside and Outside[C]//CLOUD COMPUTING 2013, The Fourth International Conference on Cloud Computing, GRIDs, and Virtualization, 2013: 101-107.

作者简介：

熊微，硕士，主要从事云计算相关技术的研究；房秉毅，高级工程师，博士，主要从事云计算、核心网新技术的研究；张云勇，中国联通研究院平台与云计算研究中心主任，博士后，主要研究方向为下一代开放网络、固定移动融合核心网、移动互联网及业务、公共运算；吴俊，副教授，博士，IEEE/AIS/PMI高级会员，主要研究方向为移动互联网应用、云计算商业价值评价等；李素粉，博士，主要从事云计算相关技术的研究。