基于模糊测试的虚拟机安全漏洞挖掘工具的设计与实现

牛新立，双锴
（北京邮电大学网络与交换技术国家重点实验室，北京 100876）

摘要：虚拟化技术可以使得一台物理计算机上同时运行多个操作系统，并且在多个操作系统之间进行有效的资源隔离和数据隔离，能够充分利用硬件资源，节省 IT 成本，同时虚拟化技术也是云计算的核心技术之一。随着虚拟化技术的广泛应用，虚拟化技术的安全问题也暴露出来了。由于虚拟化技术的特殊性，虚拟化一旦出现安全问题，后果比传统的安全问题更为严重。保证安全的一个重要措施就是及时安装软件厂商或开源社区提供的漏洞补丁。如何抢在恶意攻击者前面发现虚拟化软件中存在的安全漏洞并且及时发布更新补丁是一个很重要的问题。本文以此为出发点，设计一个基于模糊测试的虚拟化软件漏洞发掘工具，以便于快速的发现虚拟化软件的漏洞并为厂商提供修补意见。

关键词：虚拟化；安全；模糊测试；漏洞挖掘
中图分类号：TP309.2

Design and Implementation of Virtual Machine Vulnerability Digging Tool based on Fuzz Test

NIU Xinli, SHUANG Kai
(State Key Laboratory of Networking and Switching Technology, BUPT, Beijing 100876)

Abstract: Virtualization technology can make multiple operating systems run on the same physical machine and offer effective resource isolation and data isolation, which makes more efficient use of hardware and saves IT cost. Also, virtualization is one of the key technologies of Cloud Computing. With the rapid development and deployment of virtualization technology, its security problems have emerged as an unavoidable issue. Due to the specificity of virtualization, its security problem is more harmful than traditional security threats. One key to asure security is to install updates timely after vulnerability is exposed. Finding vulnerabilities of virtualization software ahead of malicious guys is very important. In this paper, we designed and implemented a virtualization software vulnerability digging tool based on fuzz test, with which we can find vulnerabilities more efficiently and release updates timely.

Key words: Virtualization; Security; Fuzz Test; Vulnerability Digging

0 引言

虚拟化技术可以使得一台物理计算机上同时运行多个操作系统，并且在多个操作系统之间进行有效的资源隔离和数据隔离，使它们互不干扰，解决了 IT 资源使用率低以及使用不灵活的问题。利用虚拟化技术进行服务器融合，可以充分利用 IT 资源，大大降低企业的 IT 成本[1]。

除了用于服务器融合，虚拟化技术还广泛运用于软件调试与测试，恶意软件行为检测等领域[2]。最为重要的是，虚拟化技术是云计算的核心技术，虚拟化技术使得云计算成为现实。目前最成熟的亚马逊弹性计算云(Elastic Compute Cloud,EC2)，使用的就是开源的 Xen 虚拟机[3]。

作者简介：牛新立(1988-)，男，硕士研究生，主要研究方向：网络安全与虚拟化
通信联系人：双锴（1977-），男，副教授，主要研究方向：下一代网络. E-mail: shuangk@bupt.edu.cn

虚拟化技术起源于上世纪 60 年代，但是直到近几年才得到快速发展并且被人们熟知。虚拟化技术仍在不断发展，从最开始的系统虚拟化，发展出了网络虚拟化，存储虚拟化，桌面虚拟化等新技术。历史经验告诉我们，对于一门新技术，一定要重视其安全问题，不能让安全滞后于技术的发展。而虚拟化技术的安全问题，已经逐渐开始显露出来，并且得到了工业界和学术界的普遍重视。

2007 年，Arrigo Triulzi 首次公开演示了一个 VMware Workstation 中可以使虚拟机进入到宿主机的漏洞，这种漏洞称为虚拟机逃逸(Virtual Machine Escape)[4]，是指虚拟机的用户可以进入虚拟机监视器(Virtual Machine Monitor, VMM)或其他虚拟机，或者窃取其他虚拟机的信息，掠夺其他虚拟机的资源。攻击者可以通过这种漏洞获取位于一台物理机上的所有虚拟机信息，危害性特别大。

除了虚拟机逃逸，虚拟化环境中还可能存在其它诸多安全问题。例如由虚拟机引起的拒绝服务，一台虚拟机可以利用某些漏洞让宿主机崩溃或者工作异常，导致其它虚拟机不能正常运行[5]。

这种针对虚拟化系统的攻击危害性要远大于传统的对单服务器的攻击。因为一个虚拟化系统中运行着多个虚拟机，这些虚拟机可能属于多个不同的企业或组织。一旦被攻击者控制，后果不堪设想。

对于虚拟化软件的使用者来说，保证安全的一个重要措施就是及时安装软件厂商或开源社区提供的漏洞补丁。如何抢在恶意攻击者前面发现虚拟化软件中存在的安全漏洞并且及时发布更新补丁是一个很重要的问题。本课题以此为出发点，设计一个基于模糊测试(Fuzz test)[6]的虚拟化软件漏洞发掘工具，以便于快速的发现虚拟化软件的漏洞并为厂商提供修补意见。

1 虚拟化安全与模糊测试

1.1 传统互联网安全威胁与虚拟化环境中安全威胁的区别

传统的互联网安全威胁主要来自外部。由于网络服务程序的缺陷，攻击者通过互联网向网络服务程序发送特定的网络数据包，导致网络服务程序异常，进而导致拒绝服务或者攻击者获取服务器访问权限。

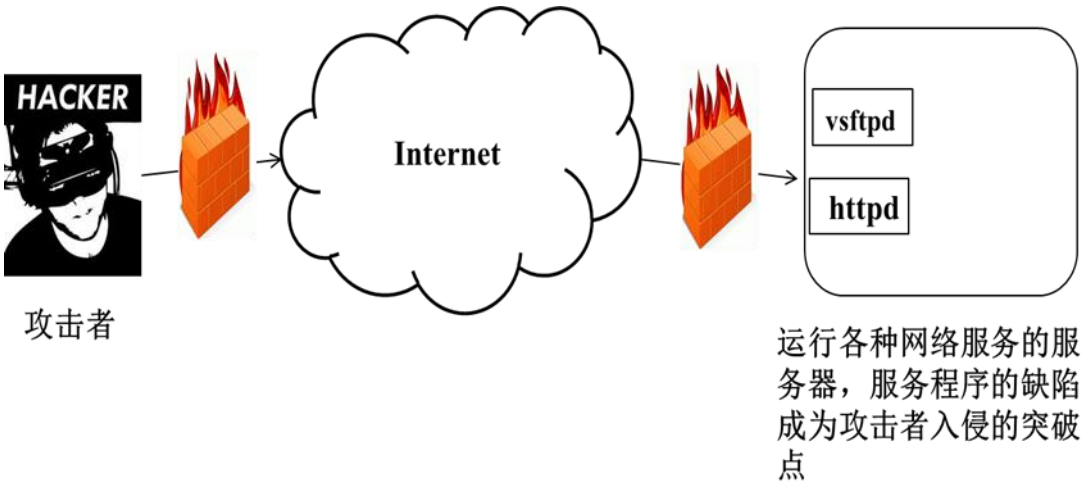

图 1 传统互联网安全威胁模型

Fig. 1 Traditional Internet Security Threat Model

在虚拟化环境中，网络服务程序一般运行于虚拟机中，同一宿主机中可能有多个这样的虚拟机，这些虚拟机用户之间是相互不信任的。这种情况下，服务器除了受到来自外部的攻击，还要防范内部的攻击者。内部攻击者除了可以用传统方式对网络服务程序进行攻击，还可以利用 VMM 或者 VMM 的组件的缺陷，先获得整个宿主机的权限，进而获取这个宿主机上所有虚拟机的权限，例如一个虚拟机用户可以利用 Xen 的 pygrub 组件缺陷获取整个宿主机的权限进而控制所有虚拟机[7]，Xen 的 hypercall 机制也可以被攻击者利用[8]。这种攻击的危害性要远远大于传统的网络攻击。

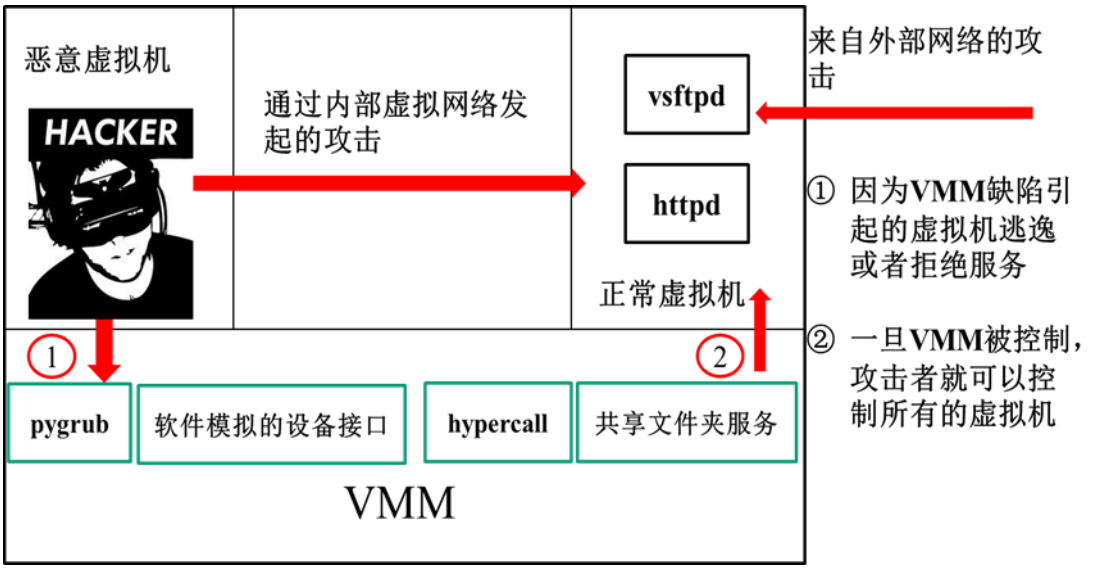

图 2　虚拟化环境下的安全威胁模型

Fig. 2　Security Threat Model Under Virtualization Environment

1.2　模糊测试在虚拟化安全中的应用

模糊测试(Fuzz test)是一种软件安全测试技术。所谓模糊测试，就是用随机产生的输入对程序进行测试，观察程序是否产生异常。一般说来，模糊测试中发现的问题都具备以下特点：

1)　严重的安全漏洞和程序异常

2)　普通人工测试和自动化测试很难复现的 bug

大部分虚拟化漏洞具有以下特点：

1)　输入在虚拟机内，由虚拟机用户控制

2)　处理程序在宿主机或者 VMM 中，而且是以最高权限执行

3)　处理程序不能正确检查或者处理来自虚拟机的某些输入

这与网络服务程序有异曲同工之处。如果具有最高权限的处理程序不能正确处理虚拟机用户的输入，那么就会存在非常严重的安全问题，轻则引起拒绝服务，重则导致整个宿主机被攻陷。

虚拟机安全问题的这个特性，非常适合用来模糊测试。在虚拟机中构造随机的输入，然后检查宿主机是否产生异常，例如崩溃或死机，以此来发现潜在的安全问题。

2　漏洞挖掘工具的设计与实现

2.1　总体架构

漏洞挖掘工具主要分为以下几个模块：模糊测试模块、模糊测试监控模块、异常收集模

块、数据存储模块。

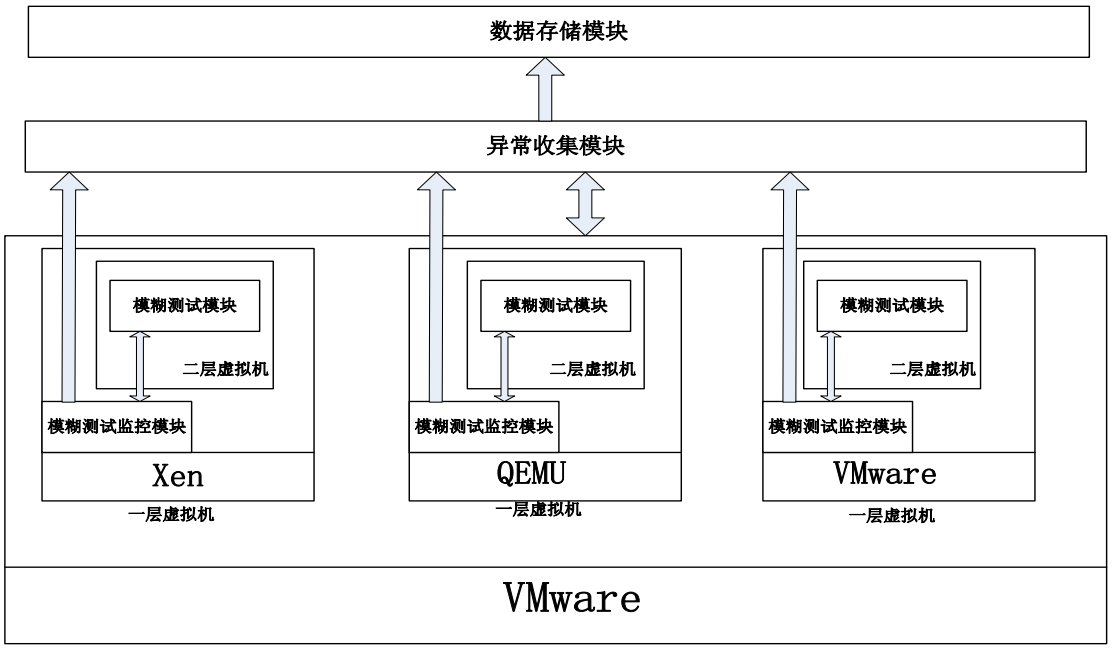

图 3　漏洞挖掘工具总体架构

Fig. 3　Framework of Vulnerability Digging Tool

2.1.1　模糊测试模块

模糊测试模块是本系统的核心模块，其中又包括对虚拟机通用的模糊测试模块，还有针对具体虚拟机的模糊测试模块。

通用的模糊测试模块：

1)　对 VMM 指令解析系统的模糊测试功能：在虚拟机内产生随机的机器指令序列，测试 VMM 的指令解析系统。

2)　对 VMM 的虚拟 I/O 系统的模糊测试功能：在虚拟机内对虚拟机的 I/O 端口进行随机的读写，测试 VMM 的 I/O 虚拟化系统。

针对特定虚拟机的模糊测试模块：

1)　对 Xen 的 Hypercall 的模糊测试功能：在 Xen 半虚拟化的虚拟机内，调用 Hypercall，随机产生参数，测试 Xen 的 Hypercall。

2)　对 Xen 的 PyGrub 组件的模糊测试功能：在 Xen 的宿主机内，构造随机的虚拟硬盘，内核文件以及 grub.conf 文件，对 PyGrub 进行测试。

3)　对 VMware Workstation 的 DHCP 服务的模糊测试功能：在 VMware Workstation 虚拟机内部，随机发送 DHCP 协议的数据包，测试 VMware Workstation 的 DHCP 服务。

模糊测试模块开机启动，对虚拟化组件进行模糊测试，并受模糊测试监控模块监控。

2.1.2　模糊测试监控模块

对于每一个模糊测试模块，都必须有一个模糊测试监控模块与之对应，它需要具备以下功能：

1)　随宿主机开机自动启动：宿主机开机后监控模块自动启动。

2)　检测宿主机的异常情况：检测宿主机异常情况，例如内存耗尽，CPU 使用率过高等。

3) 启动模糊测试模块所在的虚拟机：启动装有模糊测试模块的虚拟机后里面的模糊测试模块会自动启动。

4) 与模糊测试模块通信：通报主机情况，接收模糊测试模块发送的导致异常的输入。

5) 将异常信息发送到异常收集模块：将导致异常的输入以及异常对应的虚拟机组件，版本等信息发送至异常收集模块。

2.1.3 异常收集模块

异常收集模块负责收集各种异常信息。

1) 收集各个模糊测试监控模块发送的异常信息：监听各个模糊测试发送的异常信息

2) 将异常信息交付数据存储模块：将异常信息交给数据存储模块进行结构化存储

2.1.4 数据存储模块

1) 对异常信息进行存储：将异常信息存至数据库或者文件

2) 提供异常信息的查询和处理接口：提供异常信息的查询，例如某个组件的异常情况

2.2 漏洞挖掘工作流程

漏洞挖掘的工作过程：

1)异常收集模块启动(手动)

2)异常收集模块通过服务器控制面板远程启动服务器，并与服务器建立心跳连接

3)服务器开机后，其中的 VMware 一层虚拟机自动启动

4)一层虚拟机自动启动后，其中的模糊测试监控模块自动启动

5)模糊测试启动二层虚拟机，二层虚拟机中的模糊测试模块开机自动启动，并与外层的模糊测试监控模块建立连接，模糊测试模块开始工作

6)如果二层虚拟机崩溃，模糊测试监控模块重启该虚拟机，模糊测试将导致崩溃的输入发送给模糊测试监控模块，模糊测试监控模块将此异常发送至异常收集模块

7)如果一层虚拟机崩溃，该虚拟机会自动重启(VMware 的自动重启功能)，此时模糊测试监控模块会收集信息发送至异常收集模块

8)如果物理服务器崩溃，异常收集模块会监测到，然后利用远程控制面板自动重启服务器

3 利用漏洞挖掘工具挖掘漏洞

3.1 虚拟机异常情况

利用该漏洞挖掘对 Xen,QEMU 和 VMware Workstation 进行测试，出现异常的情况如下：

表 1 虚拟机异常及崩溃次数

Tab. 1 Virtual Machine Exceptions

虚拟机	Xen	QEMU	VMware Workstation
虚拟机崩溃	12	21	2
宿主机崩溃	2	1	0
宿主机异常	4	12	0
合计	18	34	2

从测试结果来看，商业软件 VMware Workstation 相对于开源软件 Xen 和 QEMU 来说安全性更高，没有出现宿主机崩溃或者宿主机异常的情况，只出现过两次虚拟机崩溃的现象。

虚拟机崩溃的危险性很低。

而开源的模拟器 QEMU 出现了最多次的虚拟机崩溃和宿主机异常，却只有一次宿主机崩溃。这是因为 QEMU 只是一个简单的模拟器程序，运行在宿主机的用户控件，对宿主机影响不大。

被广泛使用的 Xen 出现了两次宿主机崩溃，这是因为 Xen 的宿主机实际上也是个虚拟机，只是比较特殊，它具有超级权限，但是运行于 Xen 的 VMM 之上。

3.2 漏洞挖掘实例

3.2.1 漏洞简介

Pygrub 是一个用 Python 模拟的 grub，用于读取 Xen 虚拟机的虚拟磁盘的引导信息并载入虚拟机的内核镜像，主要用于读取半虚拟化的 Xen 虚拟机[9]。

PyGrub 具有如下特点：

1) PyGrub 的输入可以由虚拟机用户控制，例如虚拟机内核文件，grub.conf 配置文件
2) 运行于 Xen 的 Dom0 虚拟机内，Dom0 是用于管理所有虚拟机的
3) 具有最高权限

基于这个特点，对 PyGrub 进行模糊测试，发现当在内核文件 vmlinuz 后加填充 0 时并且填充的特别多时，Dom0 的内存会耗尽。

对 Xen 的源代码进行分析，发现 PyGrub 在读取虚拟机内核文件时，会先将整个内核读入内存中，然后才进行解析。如果虚拟机内核文件特别大，那么 Dom0 的内存会耗尽，这样所有的虚拟机都会受影响。

此漏洞已经提交至美国国家安全漏洞库(National Vulnerability Database, NVD)，编号为 CVE-2012-2625，漏洞描述信息为：

The PyGrub boot loader in Xen unstable before changeset 25589:60f09d1ab1fe, 4.2.x, and 4.1.x allows local para-virtualized guest users to cause a denial of service (memory consumption) via a large (1) bzip2 or (2) lzma compressed kernel image[10].

3.2.2 重现漏洞

在一个 VMware Workstation 虚拟机中做实验。

操作系统：Fedora 16 64 位

Xen 版本：Xen-4.1.2

步骤：

1) 建一个半虚拟化的虚拟机，操作系统是 Linux 即可，此处选用 Fedora 16 64 位操作系统。
2) 在虚拟机内重新编译内核，使内核的压缩格式为 bzip2 格式。
3) 创建一个全部为 0 的大文件， dd if=/dev/zero of=/tmp/large_file bs=1G count=10
4) 将此大文件追加到内核文件后，cat large_file >> vmlinuz
5) 重启虚拟机

结果：

当 Dom0 的内存为 4G(Xen 安装在 VMware 虚拟机中，内存可以随意调整)，交换空间为 4G 时，虚拟机不能重启。

当 Dom0 的内存为 12G，交换空间为 4G 时，虚拟机可以重启，但是启动过程耗时接近

5 分钟，而且这 5 分钟内 Dom0 的内存使用率几乎为 100%，CPU 使用率也为 100%，根本不能再为其它虚拟机提供服务。

4 结论

本文介绍了虚拟化技术面临的安全问题，然后提出了一种基于模糊测试的虚拟化软件漏洞挖掘工具的设计和实现方法，并且尝试用此工具对 Xen，QEMU 和 VMware Workstation 等软件进行了漏洞挖掘。在实验过程中，发现了 Xen 的 PyGrub 组件的一个拒绝服务安全漏洞，并提交至美国国家安全漏洞库，证明此工具确实能对虚拟机的安全漏洞进行挖掘，从而提早发现漏洞并发布更新补丁。

[参考文献] (References)

[1] P M Chen, B D Noble. When virtual is better than real[J]. Hot Topics in Operating Systems (HOTOS '01), 2001, 8:133-138

[2] 项国富，金海，邹德清，陈学广. 基于虚拟化的安全监控[J]. 软件学报，2012，23（8）：2173-2187.

[3] 维基百科. 亚马逊弹性云计算[OL]. [2012-11-30]. http://en.wikipedia.org/wiki/Amazon_Elastic_Compute_Cloud

[4] A Triulzi VMware escape and firmware viruses[OL]. [2007-10-30]. http://www.alchemistowl.org/arrigo/Papers/VMware-escape-and-firmware-viruses.pdf

[5] J Reuben. A Survey on Virtual Machine Security[OL]. [2007-10-11]. http://www.tml.tkk.fi/Publications/C/25/papers/Reuben_final.pdf

[6] 维基百科. [Fuzz testing]. [2012-10-21]. http://en.wikipedia.org/wiki/Fuzz_testing

[7] CVE 安全漏洞库. CVE-2007-4993[OL]. [2007-10-3].http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2007-4993

[8] C Le. Protecting Xen hypercalls [OL]. [2009-11-23]. http://www.cs.ubc.ca/grads/resources/thesis/Nov09/Le_Cuong.pdf

[9] Xen Wiki. What is PyGrub?[OL]. [2007]. http://wiki.xen.org/wiki/PyGrub/zh

[10] CVE 安全漏洞库. CVE-2012-2625[OL]. [2012-5-15]. http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2012-2625