文章编号：2096-1472(2020)-02-34-05
DOI:10.19644/j.cnki.issn2096-1472.2020.02.010

基于虚拟化技术设计信息安全攻防实训环境

张月红

(上海电子信息职业技术学院，上海 201411)

摘　要：本文分析了信息安全人才培养及院校的技能实训环境现状，通过对KVM虚拟化和SDN虚拟网络运行原理及功能特性的剖析，将信息安全实训演练与虚拟化技术结合，为院校提供高度仿真的攻防环境；在虚拟化调度与运行方面做出优化，使院校计算资源能够充分利用；最后采用高负载压力测试验证相关优化技术的实际效果。

关键词：KVM；SDN；信息安全；攻防实训

中图分类号：TP311.1　**文献标识码**：A

The Design of Information Security Attack and Defense Training Environments Based on Virtualization

ZHANG Yuehong

(*Shanghai Technical Institute of Electronics&Information*,*Shanghai* 201411,*China*)

Abstract: This paper analyses the current situation of information security personnel training and skills training environment in colleges.Through the analysis of the operation principle and function characteristics of KVM virtualization and SDN virtual network,this paper combines information security training exercise with virtualization technology to provide a highly simulated attack and defense environment for colleges and universities,which optimizes the virtual scheduling and operation,and makes the full use of computing resources in colleges.Then high loading pressure test is used to verify the actual effect of the optimization technology.

Keywords:KVM;SDN;information security;attack and defense training

1 引言(Introduction)

2016年，赛迪智库发布《信息安全产业发展白皮书(2015版)》中指出：“信息安全技术本身具有专业性强、技术更新速度快等特点，现有的信息安全人才培育和引进机制尚不能满足产业和企业发展的需求。”

随着我国信息服务业逐渐兴盛，信息安全保障压力必将更加巨大，对信息安全相关人才需求也将更加迫切，其中懂技术、懂管理和懂业务的综合型人才更是稀缺人才。信息安全人才培养需要系统化和专业化，需要从当前产业和企业用人现状出发，优先进行安全技能型人才的着重培养，以满足对实战型安全人才的需求。

本文通过虚拟化的方式开发信息安全攻防实训系统，旨在普及信息安全技能，通过实训演练的方式提升相关专业学生或从业人员的信息安全动手能力。

2 目标(Objective)

采用KVM虚拟化技术作为底层支撑，通过PHP及Python等动态语言调用虚拟化，为每位学生提供独立的虚拟化实训环境，并提供复杂网络拓扑模拟功能，能实现多级应用场景下的安全攻防实训。

(1)在实际攻防环境中通常存在多级网络场景，但尚未应用于虚拟化实训教学中。通过对SDN技术的研究，实现入侵路径攻击链的场景复原，并批量分配给客户端。

(2)通过对学生实训过程的输入捕获，实现实训结果自动化评估，并能追踪学生在攻防实训环境中的关键操作。

(3)实训环境随用随开，完成立即销毁，减轻对系统的性能压力。如遇到高并发，启动环境可提前生成，学生登录系统后，系统自动推送给各学生端。

3 关键技术(Key technologies)

3.1 KVM 虚拟化技术

KVM[1]是基于虚拟化扩展X86硬件的开源Linux原生全虚拟化解决方案。在KVM中的虚拟机被实现为常规的Linux进程，由标准Linux调度程序进行调度；虚拟机的每个虚拟CPU被实现为一个常规的Linux进程。这使得KMV能够使用Linux内核的已有功能。KVM本身不执行任何硬件模拟，需要客户

空间程序通过/dev/kvm接口设置一个客户机虚拟服务器的地址空间，向它提供模拟的I/O，并将它的视频显示映射回宿主的显示屏。KVM的逻辑架构如图1所示，目前主流的应用程序是QEMU。

(1)虚拟化调度接口

在KVM虚拟化管理中，最常使用的远程管理接口为Libvirt，该组件提供了对虚拟化及虚拟机的管理功能，例如存储和网络等。Libvirt相关软件包括稳定的C语言底层API及表层二次接口、守护进程libvirtd,以及命令行工具virsh。Libvirt的主要目标是提供能够独立管理多种不同的虚拟化技术及相关虚拟机，包括KVM/QEMU、Xen、LXC、OpenVZ或VirtualBox。

在Python中，对Libvirt进行相应的接口调用的依赖组件为libvirt-python，该软件包提供基于python2的API，安装完成后可在/usr/lib/python2.7/site-packages/libvirt.py中调用。

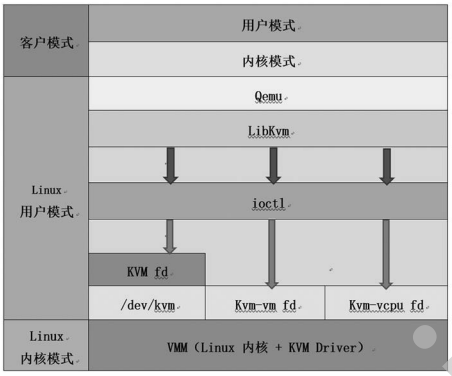

图1 KVM逻辑架构

Fig.1 The logic framework of the KVM

(2)后端镜像

KVM提供后端镜像backing_file技术支持，在KVM中后端镜像技术则更为方便，用户可随时对后端镜像进行维护，并可快速创建派生镜像，派生镜像初始文件大小为192kB，仅记录基本信息，随着派生镜像的启动运行，镜像文件随之增长，但仅记录与后端镜像产生差异的部分，并且后端镜像文件不会被修改，除非在QEMU monitor中使用commit命令或者使用qemu-img commit命令去手动提交这些改动。在KVM及QEMU后端镜像中较其他虚拟化技术更有优势的是Python API接口生态较为完整。

3.2 浏览器远程控制技术

Guacamole是一种基于浏览器的远程管理控制软件，用户通过浏览器访问其控制台生成一条链路，通过调用后端的GUACD数据中转组件，浏览器无须配置任何接入组件即可通过GUACD连接目标虚拟机中的VNC、RDP、SSH等远程管理方式。同时考虑到性能因素，该软件支持分布式与集群部署，并能通过其他第三方软件如Nginx或Keepalive实现高可用。

(1)Guacamole技术架构

Guacamole远程控制软件的Web应用本身不支持任何远程管理协议，也不支持直接的TCP数据传输，其仅对内部前后端数据通信的Guacamole传输协议提供完整的兼容和调用。

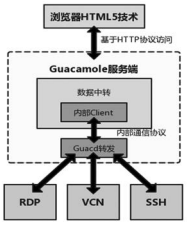

图2 Guacamole技术架构

Fig.2 The technical architecture of the Guacamole

完整的数据流转链路为：学生通过浏览器访问Guacamole前端展示，并通过JavaScript脚本向Server端发送需要访问的服务器标识数据，Server端接收数据并处理为Guacd可处理的格式，而后由Guacd向最终目标服务器发起RDP/SSH/VNC等协议的远程访问，并将画面及控制数据层返回至用户浏览器前端，由JavaScript进行渲染。

(2)鉴权与身份认证

Guacamole支持多种身份认证模式，包括默认的XML文本身份认证、数据库认证、LDAP统一身份认证、HTTP头认证、CAS统一身份认证、OpenID统一身份认证、Radius身份认证等。

3.3 SDN虚拟网络

信息安全的攻防不是单一网络场景，而是由办公网络、DMZ、IDC等多种交换路由网络组成，完整的模拟一条攻击链路需要将各个网络、主机、服务、应用全面结合，而这种融合的基础则是网络层面的全面仿真。采用SDN[2]软件定义网络的方式来实现多层级拓扑结构，包含多层级路由结构与多个VXLAN结合，动态的按需创建内部虚拟化网络，并将实验环境的每一个虚拟机配置于合理的网络结构中，实现一套拓扑多个主机多层网络多套网络设备与接口，而不同的实训学生则可生成互不影响的多套拓扑结构，解除了网络设备对实训环境搭建的限制。

(1)基础二层网络

默认配置的KVM虚拟化及Libvirt管理工具提供了初始化的二层网络，由基础的静态路由和简单虚拟网络、DHCP等一系列基础服务组成。默认的二层网络基于Libvirt的XML网络配置文件创建，编辑维护管理均存在一定不便，需要配置网络变更时按顺序执行注销网络、修改配置、生成网络、调用网络等多个步骤。例如下面的代码为最小化配置的NAT二层网络，如需修改子网、网关等配置信息均需要对实体文件进行修改，并完成上述步骤。

```
<network>
  <name>default</name>
  <uuid>06fc05f7-7404-49f5-92ec-51d2519fb41d</uuid>
  <bridge name="virbr0" />
  <mac address='52:54:00:27:B7:90'/>
  <forward/>
  <ip address="192.168.122.1"
```

```
netmask="255.255.255.0">
        <dhcp>
            <range start="192.168.122.2"
end="192.168.122.254" />
        </dhcp>
      </ip>
    </network>
```

(2)多级三层网络

在业务系统部署场景中常见多级网络与多个拓扑共同构成一套业务系统的情况，同时网络设备与安全设备的接入链路对虚拟化实训的场景复原能力提出了巨大的挑战，网络旁路、流量镜像、单臂路由、多级三层路由、NAT转换、ARP代理、DHCP服务等在虚拟网络中的实现也需要一套能够管理三层网络的虚拟网络软件。Neutron为整个SDN环境提供网络支持，包括使用虚拟路由实现租户网络的互通和隔离、二层交换、三层路由、负载均衡、防火墙等。最常见的多级网络部署模式为虚拟专用网络VPC，这种拓扑结构中每个用户都可以有多个子网，Neutron负责提供路由服务协调所有子网，用户间可以通过配置路由器的路由表来控制子网间的连通。如图3所示。

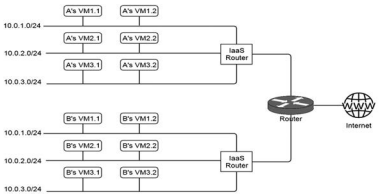

图3 多级网络VPC模式

Fig.3 The multi-level VPC schema

4 设计与实现(Design and implementation)

4.1 总体架构设计

本系统由基础虚拟化环境、攻防实训演练子系统(学生)、实训及教学管理子系统(教师)三部分共同组成[3]。

基础虚拟化环境建设主要工作内容有：物理服务器及基础操作系统部署、KVM虚拟化及SDN Neutron配置、Web和数据库服务部署、浏览器远控服务部署等。基础环境为本文所涉及的系统提供软硬件运行环境。

攻防实训演练子系统主要包括基于课程体系学习、实训演练操作、学习成果信息查看管理等，用户可根据自己的兴趣爱好选择不同的课程体系进入实训环节，并能随时查看成绩排名及实训进度等信息，一键返回对应实训课程继续学习。

实训及教学管理子系统主要包含基础虚拟化管理、教学及教务管理等相关功能，在实训教学管理过程中，除了对实训教学内容的增删改，还能够实时监控实验过程画面并提供远程协助，以及对实训操作结果的自动化评分。

4.2 攻防实训演练子系统设计

攻防实训演练子系统，包含实训课程体系、实训演练操作、课程进度成绩管理三个功能模块，使用PHP开发并使用MySQL数据库，并使用Apache作为前端Web服务器,架构如图4所示。

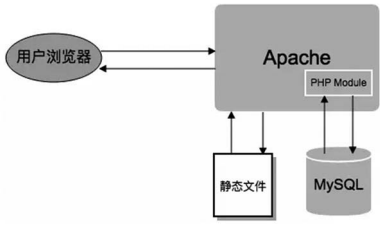

图4 攻防实训演练子系统架构图

Fig.4 The frame diagram of the attack and defense training subsystem

实训演练操作模块是本子系统的核心，学生的全部实训演练及学习行为均发生在该模块中，从实训操作出发，将实训手册、实训远程控制窗口、选择实验环境、查看课前知识介绍、回答课后习题等功能动态结合，在一个单一实训任务中体现教学做评考的完整学习链。

(1)课前准备

信息安全领域的技术技能没有孤立的攻防，学生应当了解应用所需的基础知识理论与实操技能。在实训前向学生展示需理解的知识点和关联的实训场景信息。包括前置知识、实训背景、环境介绍、工具及命令讲解等内容。

(2)动手实操

动手实操环节，学生通过浏览UI左侧的指导手册，并基于指导手册中的操作步骤，对UI右侧的远程控制窗口进行实训演练操作。由于信息安全相关实训具备较大的不确定性，可能会对实训环境产生不可逆的破坏，实操模块还需对实训环境对应的一个或一组虚拟机进行监控，当前端触发初始化请求时将对原环境全面销毁，并使用后端镜像技术为学生重新生成完全一致的实训环境，避免重复搭建而浪费课堂时间。

在操作窗口中主要调用到Guacamole的HTML5窗体，通过WebSocket的方式将后端数据传输至前端，再由前端Canvas绘制将后端的数据展现出来。对后端的WebSocket的调用通常嵌套在页面中，示例代码如下，在前端页面中则以ws://IP:Port/?token=FsTY6VOQHjKC3LZU的通信协议连接。

```
<guacd
        data:ng-init="display={ width:100%, fitTo:
'scale' }; viewOnly=false; trueColor=true; states=[];
path='websockify'"
        data:style="{ margin: '5px 0 0 0'}"
        data:view-only="false"
        data:true-color="trueColor"
        data:display="display"
        data:host="服务端IP"
        data:port="服务端socket端口"
        data:path="?token=FsTY6VOQHjKC3LZU"
        data:is-connected="connected"
        data:states="states"
        >
</guacd>
```

(3)课后练习

课后练习是完成学习后对实训演练的考核，检验学生是否理解并掌握课程内容。课后练习方式为触发弹窗式，在正确回答问题后方可结束实训，避免学生因误操作或盲目操作而影响实训效果。

4.3 实训及教学管理子系统设计

实训及教学管理子系统主要承载了管理员及教师的相关管理操作，同时提供底层虚拟化调度及输出、虚拟网络及实验监控等功能。该子系统主要实现虚拟化及网络的管理功能、实训课程管理、实训监控，以及自动化评分等教学相关功能。由于虚拟化调度、监控、输出、网络管理等均采用PythonAPI，调用Django框架作为显示前端。同时考虑到子系统的连续运行等特性，采取UWSGI作为运行Django框架的中间件将数据输出至Nginx，并由Nginx向浏览器输出显示元素。

在功能设计中主要包括五个模块，包括实训虚拟化及网络管理模块、实训课程管理模块、实训监控模块、自动化实训评分模块和用户管理模块。功能如图5所示。

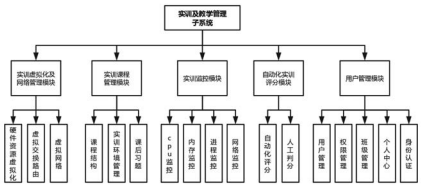

图5 管理子系统功能模块

Fig.5 The functional modules of managerial subsystem

(1)实训虚拟化及网络管理模块

实训虚拟化和网络管理是整体实训系统的核心，是提供教学与实训环境的基础模块。考虑到信息安全相关的攻防实训对运行环境及网络拓扑均有较高要求，此处的设计采用与OpenStack一致虚拟化技术和网络技术，即KVM和Neutron。

实训系统中对虚拟资源的管理主要是虚拟机资源的管理，考虑到虚拟机所处的不同状态，可以将对虚拟机的资源管理分为三种：创建虚拟机的资源管理、启动虚拟机的资源管理及迁移虚拟机的资源管理。创建虚拟机时，实际上只是完成虚拟机文件的创建，虚拟机所需要的CPU、内存等资源暂时还未分配；当创建共享类型的虚拟机时，实训系统在分配资源时采用的是轮询方式，轮询从虚拟机服务器组中选择一台服务器来创建虚拟机。启动虚拟机时的资源管理策略目的是将虚拟机启动到合适的服务器上，以提高实训系统资源池中资源的利用率。

系统采用后端镜像技术对性能进行优化，通过部署好的KVM虚拟机作为实训环境的原始模板，并对原始模板中的操作系统及实训环境进行最大化的性能调优，同时对原始模板的虚拟化配置文件进行优化，降低CPU及内存开销。基于此模板生成的新镜像则无须配置即可达到最小的性能消耗。

在Python中调度后端镜像功能通常使用qemu-img -b 后端镜像文件.img -f qcow2 新镜像文件.img的命令，实现代码如下：

```
def generate_cloudhost_image(self,template,vmname):
        cmd=[]
        cmd.append('qemu-img')
        cmd.append('create')
        cmd.append('-b')
          cmd.append(self.template_path+template+'.img')
        cmd.append('-f')
        cmd.append('qcow2')
          cmd.append(self.cloudhost_path+vmname+'.img')
        print ' '.join(cmd)
        subprocess.call(cmd)
```

将创建的镜像文件运行至KVM中即可面向学生提供实训环境，当面临多个学生同时创建实训环境时也无须复制原始镜像文件，秒级创建实训镜像文件。

(2)实训课程管理模块

实训课程管理模块包含了完整的门、章、节等结构，通过层层递进的方式设计不同的课程体系，并能在小节中管理多种与实训相关的信息，例如课前准备、知识点讲解、实验介绍、实训手册、趣味问答、课后习题等，并通过HTML5的方式展示富文本信息，通过引入外部编辑器UEditor，解决对PPT、WORD、音频视频等多媒体教学的需求，实训课程管理模块结构如图6所示。

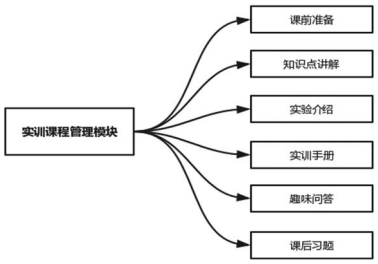

图6 实训课程管理功能模块

Fig.6 The functional modules of training courses management

根据功能需求，本模块主要解决包括教学内容与实训的关联、实训与考核方式的管理问题，通过对后端数据库各个相关数据表的增删改查等操作，向演练子系统动态输出课程内容与实训环境等资源。

(3)实训监控模块

在实训教学过程中主要存在几种常见的管理问题，例如学生需要协助指导、巡查学生操作是否顺利、帮助学生还原初始环境、实训系统性能负载等，因此在实训监控模块应能提供物理机的硬件资源消耗监控、学生实验桌面监控、环境恢复初始化、强制关闭实训环境、操作学生实训虚拟机等功能。

通过监控模块[4]对虚拟实训环境中的操作过程进行实时监控并记录。监控记录用于对实训环境的实操过程把控，对虚

拟实训环境中的操作进行记录可以用于进行实训技能评估。实现系统用户的操作行为记录和查询；通过SNMP、SYSLOG对目标设备的实时状态信息进行采集，通过流量镜像和收取对目标网络的流量信息进行采集；针对采集到的信息实现标准化处理引擎，通过处理规则将信息进行标准化处理并输出，规则实现默认库和自定义添加。

为实现对数据库中的日志数据进行实时解析，采用Storm实现实时数据分析。同时考虑到监控模块需要对物理机操作系统进行性能监控，需调用Python脚本对数据进行获取，Python主要通过psutil组件，可实现对CPU、内存、硬盘、网络、进程等信息的获取，调用代码如下：

```
def get_cpu(interval=1):
    return (str(psutil.cpu_percent(interval)))
def get_memory():
    phymem=psutil.phymem_usage()
      buffers=getattr(psutil, 'phymem_buffers',
lambda:0)()
      cached=getattr(psutil, 'cached_phymem',
lambda:0)()
      used=phymem.total-(phymem.free+buffers+
cached)
    line="%5s%%" % (
        phymem.percent
    )
    return phymem.percent
```

实训监控模块的功能设计如图7所示。

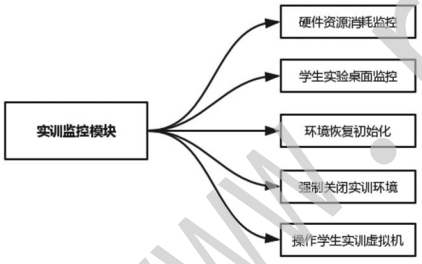

图7 实训监控模块功能

Fig.7 The functional modules of training monitor

(4)性能测试

考虑到本系统提供大量虚拟化操作系统的访问，需要较强的性能负载，而传统的访问式的性能测试工具如LoadRunner无法提供相应功能，因此采用自行编写的脚本批量创建实训环境[5]，并随时观测其性能状况。在物理服务器中分别使用sysstat、iotop、htop、nmon等工具对CPU及硬盘负载进行监控并记录。

测试使用脚本创建100个CentOS实训环境，在进入操作系统后随机时间打开10个门户网站，测试环境的CengOS计算资源指标均为2核CPU、4G内存。

在测试开始时可见第二硬盘的负载瞬时变大，并产生了少量的突发硬盘负载，通过对脚本执行日志的分析得知约每25个虚拟机的并发就会触发一次小规模的I/O资源紧张。如图8所示在全部测试过程中第一硬盘的性能几乎没有任何消耗，同时CPU资源虽然存在持续上升趋势，如图9所示，但服务器总计48核CPU，CPU LOAD为2.5仅占全部CPU性能的约5%。本次测试耗时3分46秒，100个2核CPU、4G内存的CentOS已全部加载操作系统并完成10个网页的浏览器访问。

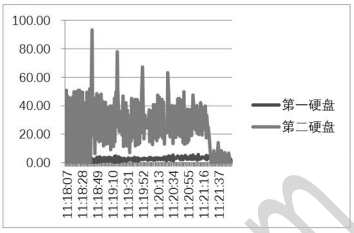

图8 两块硬盘性能消耗分析

Fig.8 The performance consumption analysis of two hard disks

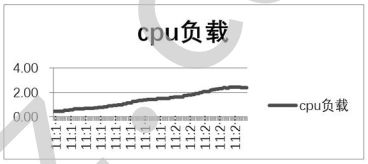

图9 物理服务器的CPU消耗

Fig.9 The CPU consumption of the physical server

通过批量创建不同操作系统的实训环境分析可知，本系统已实现计算性能的全面优化，降低了CPU与硬盘的性能消耗，能够通过一台物理服务器为大量学生提供实训环境，并能在短时间内完成大量实训环境的创建任务，而无须对既有教室或实训室进行改造，提高了资源使用率，提升了教学环境使用效率。

5 结论(Conclusion)

本文通过调度底层KVM虚拟化及SDN等相关技术将信息安全攻防技术与实训教学过程进行了一定程度的结合，并通过对教学和教务管理的深入分析，最终设计并实现了基于底层虚拟化的信息安全教学实训系统。通过测试，该系统能够满足实训演练教学做一体化的要求，可实现预期设计目标。

参考文献(References)

[1] 马震太,张晓梅.BOSS在KVM平台中的性能研究与优化[J].计算机工程,2017,43(7):70-74.

[2] 许磊,顾进广,何亨.基于SDN架构的网络能耗与性能动态调节机制[J].计算机工程,2018,44(4):108-114.

[3] 马丽.网络安全攻防训练平台设计与实现[J].无线互联科技,2017(11):75-76

[4] 任永杰,程舟.KVM实战:原理、进阶与性能调优[M].北京:机械工业出版社,2019.

[5] 杨志国.银行业数据中心性能测试的策略与实践[M].北京:人民邮电出版社,2019.

作者简介：

张月红(1975-)，女，硕士，副教授.研究领域：渗透测试，漏洞挖掘，WEB安全.