doi:10.16104/j.issn.1673-1891.2020.03.009

Docker技术在恶意代码检测环境部署中的应用

雷惊鹏,朱小娟
(安徽国防科技职业学院信息技术学院,安徽 六安 237011)

摘要:通过对恶意代码行为和特征提取技术的分析,提出了基于虚拟环境下实现恶意代码检测的方法,设计了相应的检测系统;利用虚拟化技术,通过Docker容器简化检测环境的配置,增强了代码检测的隔离性、安全性;并建立相应的实验平台开展测试,为检测恶意的网络行为提供了支持。

关键词:虚拟化技术;容器;恶意代码;检测
中图分类号:TP393.081 **文献标志码**:A **文章编号**:1673-1891(2020)03-0047-03

Application of Docker Technology in Malicious Code Detection Environment

LEI Jingpeng, ZHU Xiaojuan
(School of Information Technology, Anhui Vocational College of Defense Technology, Lu'an, Anhui 237011, China)

Abstract: Through analyses of malicious code behavior and feature extraction technology, a method for detecting malicious code based on virtual environment is proposed, and a corresponding detection system is designed. With virtualization technology, the Docker container can simplify the configuration of the detection environment and enhance the isolation and security of code detection. A corresponding test platform is created for tests, which provides support for detecting malicious network behavior.

Keywords: virtualization technology; container; malicious code; detection

0 引言

网络入侵事件检测、响应的数据源之一来自对恶意代码的分析结果。通过不同手段对恶意代码的特征码进行提取和分析,为恶意代码检测提供依据。Twman、Cuckoo[1]等都是较为流行的开源恶意代码分析系统,但是其部署环境较为复杂,部署周期较长。应用Docker容器技术,在嵌套的虚拟平台下构建测试环境,研究恶意代码的通信特征,进一步增强代码检测工作的安全性和隔离性。

1 相关理论

1.1 高级持续性威胁

高级持续性威胁(Advanced Persistent Threat,APT)攻击由于其隐蔽性强、持续时间长等特征[2],成为当前威胁网络资源安全的重要技术手段。APT需要进行全面而深入的信息收集、检索,以便发起对目标系统的攻击,其主要攻击载体是恶意代码,归根结底是一种计算机程序,其运行离不开操作系统的支持[3]。计算机病毒之父Fred Cohen认为:“一个程序是否为恶意代码的问题是不可判定问题”。缺少统一的衡量标准,使得研究者可以从不同角度进行分析、判断、刻画,从而在对恶意代码的理解上会存在差异。通过行为分析、特征分析等手段,提供入侵检测和响应的信息源,而态势感知则需要进一步分析入侵事件的意图所在。因此,对恶意代码的分析难度会明显提升。

1.2 恶意代码分析技术

恶意代码的结构分析和特征分析需要采集代码样本。分析方式主要有静态分析和动态分析2种,2种技术各有技术优劣。

静态分析的主要对象是源代码,或是利用反汇编工具将源码进行反汇编操作,形成汇编代码。由于能查阅到源码,因此对代码执行路径有比较明确

收稿日期:2020-04-13
基金项目:安徽省2018年度高校自然科学研究重大项目(KJ2018ZD066);安徽省2017年度高校优秀拔尖人才培育资助项目(gxyq2017182)。
作者简介:雷惊鹏(1982—),男,安徽潜山人,副教授,硕士,研究方向:信息安全。

的认知,完整性很高。采用反汇编方式难度较大,对检测人员技术要求较高,而且对于经过特殊处理(例如加壳)的恶意代码分析的效果一般。常见的静态分析技术主要有特征码分析、语义分析等。

借助沙箱[4]技术,可以在受控环境下查看恶意代码的运行结果来实现动态分析,该技术受多态、加密等手段限制的影响较小。另一方面,沙箱技术受操作系统、文件调用等因素影响较大,在运行一些特定的程序(例如需要提供命令操作参数或特定条件)时,可能会出现难以被启动的情况,从而导致测试结果的不完整和不确定。

目前采用较多的沙箱技术主要基于Hook(钩子)技术和基于虚拟化技术2种实现方式[5]。前者的安全性和隔离性不强,容易产生逃逸,导致沙箱的有效性易受破坏;其主要思路是通过诸如DLL注入等技术手段,拦截软件对系统资源的访问,并将可疑软件的写操作重定向到沙箱。 后者的技术思路是在虚拟环境中执行可疑的恶意代码,传统的虚拟机技术可以实现,但是虚拟机需要消耗宿主机的资源、需要安装GuestOS、需要配置应用程序环境,部署过程耗费较多时间,但是安全性较好。

1.3 Docker容器技术

Docker作为一种轻量级虚拟化技术,相对于传统的虚拟机技术而言,在资源消耗、启动速度等方面更具优势。Docker可以与宿主系统(本文采用Windows+CentOS的嵌套环境)共享资源,通过使用Namespaces、CGroups、UnionFS等相关技术提供了命名空间、物理资源、文件系统在内的隔离。Docker为应用环境部署提供了非常便利的条件,从本质上来说,Docker就是一套Mini版的Linux环境。在镜像市场提供了种类丰富的运行环境,管理员可以通过镜像拉取、上传等操作实现镜像管理。

2 恶意代码分析

2.1 网络行为分析

本文所涉及的平台主要是针对恶意代码的网络行为特征进行分析。表1列举了部分协议及特征提取程序所依赖的参考字段。其中,HTTP报文中的URL信息,DNS报文中的域名信息,以及IP数据包中的地址和端口信息,通常是重点关注和分析的特征字段。HTTP、URL、HTML是万维网运行的3个基本要素,DNS则为网站访问提供了友好基础。

表1 提取恶意代码特征需关注的参考字段

名称	特征提取字段
IP数据包	①源IP;②目标IP;③源端口;④目标端口
DNS报文	①域名;②请求类型;③请求结果
HTTP报文	①URL;②目标IP;③目标端口;④响应码;⑤实体类型;⑥负载内容

HTTP、URL、HTML是万维网运行的3个基本要素,DNS则为网站访问提供了友好基础。恶意代码分析需要对网络通信流量进行采集、判断。通过采集、解析DNS数据,提取通信双方的域名、地址信息,为黑白名单管理提供数据源。通过采集、判断HTTP的负载内容、状态码等信息,分析恶意代码行为特征,进而对特征进行签名。

网络行为分析的主要步骤如图1所示,具体内容如下:

1)监控网络流量,设计流量数据有效期。不在窗口有效期内的数据进入下一步处理,为提取代码特征提供数据。

2)分析数据属性。提取网络通信双方的源目地址、源目端口、HTTP特征码、状态码等信息,分析通信行为特征,为代码检测提供数据。

3)借助Cuckoo系统,比较当前网络数据与正常通信数据的特征,开展检测。对正常通信流量不做处理,对异常通信流量进行预警。

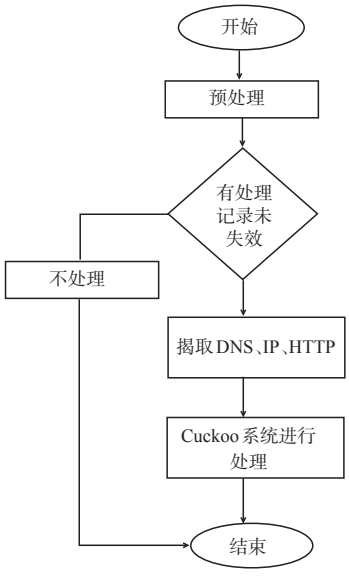

图1 网络行为分析流程

2.2 特征分析

通信双方的IP地址和DNS名称信息是恶意代码的主要提取特征。一些僵尸主机的操控者实施Fast-Flux技术提高僵尸网络的健壮性[6],其本质也属于DNS技术的新应用。通过解析DNS数据包,提取其中的域名信息和IP地址信息、客户端请求信息、响应信息等,构建通信名单。应当注意的是,由于网络通信流量中,恶意流量和正常流量往往是混

淆在一起的,给黑白名单的建立带来了一定程度的不确定性。某些安全研究人员可能更倾向误报率较高、漏报率较低的测试方式。

对疑似恶意代码的特征提取依赖于名单设计。对特定域名或IP地址发起请求的频率、行为类型危险等级等应当作为黑名单设计的参考依据。实际的网络通信会出现各种复杂的情况,例如多个IP地址绑定同一域名或相反、采用了负载均衡机制的域名解析,都有可能造成前述特征在短时间内的变化,因此对网络通信行为的危险程度进行评估和设置黑白名单的时效性非常重要。

WEB服务是因特网上使用非常普遍的业务,基于HTTP/HTTPS的业务量占据了网络通信的重要位置。企业防火墙对正常通信的WEB数据都会采取放行的策略,如果恶意代码流量与WEB访问流量混杂在一起,对恶意行为的监测无疑更加困难。虽然应用层的HTTPS协议提供了数据加密、完整性保护、防重放等功能,但目前仍有很多WEB站点使用不安全的HTTP协议为客户提供服务。同一族恶意代码的HTTP的GET请求内容有很大相似[7],基于这一研究,某些GET请求在返回"Bad Request"状态码的情况下,仍然会尝试持续提起请求。

3 测试环境部署

3.1 Cuckoo开源系统

Cuckoo Sandbox是领先的开源自动化恶意软件分析系统。作为一款专注于恶意代码分析功能的软件,在沙箱内可以自动执行Windows,MacOS,Linux和Android下的任何恶意文件分析任务。其中的一个亮点是可以对网络流量进行转储和分析,测试人员可以根据需要进行自定义。

沙箱实现了运行程序分离。通过这种分离机制在沙箱中执行未经测试的代码,或其他不可信程序,实现了一种隔离的运行环境,提高安全性。

3.2 虚拟化测试环境

Cuckoo Sandbox的部署较为复杂。通过Docker技术,将复杂系统运行在容器内,由于容器技术的隔离性、独立性、封装性,对检测环境的部署更加方便,且更加安全。

表2显示了运行容器的宿主机和虚拟机环境配置。主要思路是在宿主机(物理主机)中运行虚拟机,在虚拟机中启用Docker容器,在容器内自动化配置Cuckoo系统。这种嵌套的虚拟化实现,对保障检测环境的安全性有很大提高,不过是以牺牲物理主机性能为代价。

表2　测试环境配置

	宿主机(物理主机)	客户机(虚拟主机)
操作系统	Microsoft Windows 7 旗舰版	CentOS Linux release 7.3.1611
CPU	Intel64 Family 6 Model 60 Stepping 3 GenuineIntel	虚拟
物理内存	4 096 MB	虚拟(1 024 MB)
硬盘	500 GB	虚拟(20 GB)
网络适配器	2x2 11b/g/n Wireless LAN M.2 Adapter	虚拟(桥接)

从镜像市场获取的Cuckoo系统如图2所示。由于Docker提供了程序运行环境,因此可以直接在容器内启动该镜像,即可获得基础测试环境(图3)。

```
[root@ahgf-kvm ~]# docker images|grep cuckoo
blacktop/cuckoo                     latest              a3d58db8d4fb        13 months ago       5
16MB
```

图2　Cuckoo系统镜像

```
[root@ahgf-kvm ~]# docker run --name "ahgf-test" -it  blacktop/cuckoo /bin/sh
/cuckoo # ifconfig -a
eth0      Link encap:Ethernet  HWaddr 02:42:AC:11:00:02
          inet addr:172.17.0.2  Bcast:172.17.255.255  Mask:255.255.0.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:7 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:586 (586.0 B)  TX bytes:0 (0.0 B)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

图3　Cuckoo基础测试环境

由于命令行操作的交互性不太直观,研究人员可以进一步配置并启用MongoDB服务,使得Cuckoo以WEB界面形式与安全人员交互(图4)。

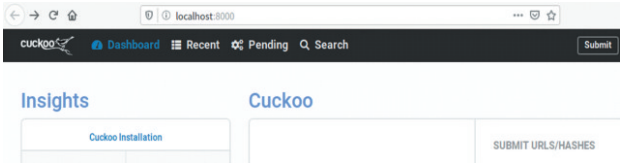

图4　Cuckoo的Web展示界面

4 结语

以APT技术为代表的网络威胁,为了逃避各类安全检测措施,使用了多样化的隐藏技术,对恶意代码检测工作提出了新的、更高的要求。本文对恶意代码的分析技术、特征提取技术做了一些介绍,提出了通过Docker容器部署检测环境的思路,相对于传统的测试环境部署而言,在Docker环境下拉取镜像、启动容器,只需要几分钟时间,大大提高了工作效率。要想实现更加全面的检测,需要安全人员进一步对沙箱和检测代码进行研究。

(下转第69页)

社会上的动力。国家体育总局社体中心要在全国加大对各级社会体育指导员的培训力度,另外对现有的社会体育指导员进行有效的监管和再培训,发挥指导员队伍的作用和价值。

4.3 社区街道等基层部门要不断提高服务意识,加强内涵建设

城市社区街道等基层组织和老年人体协等社会团体要切实提高服务意识,组织更多形式多样的老年体育健身活动和比赛,提高老年人社会参与的意识和习惯,增加老年人感情交流,提升老年人生活质量。充分保障适合老年人进行体育锻炼的体育场地设施建设和健身的专项经费投入,发挥社区各阶层能量,多渠道筹集资金,利用政策优势,吸引社会企业或个人捐赠创办老年体育健身的公益性服务组织。加强老年体育健身内涵建设,大力整合开发社区体育服务人力资源,加强体育专业人员的培训和引进,充分发挥民间体育精英和社区体育积极分子的模范带头作用。

4.4 老年人群体自身要不断提高健身意识,努力构建科学健康的体育生活方式

老年人身体功能退行性衰退,各种慢性病高发,并经常受到寂寞孤独、失落悲观等负面情绪的影响。在这种形势下,养成健康的体育生活方式,不但能增强体质、调节心情,而且能促进社会适应能力,提升生命质量,因此体育健身成为促进老年人身心健康的最佳选择[12]。老年人要提升生命质量和幸福指数,就一定要根据身体状况,努力提高自身健身意识,利用各种渠道,充分利用现有资源,获得科学健身的信息,学习锻炼技巧,逐步养成科学合理健康的体育生活方式。发展迅速的各种居家养老社会服务机构应构建平等互利的自组织模式和自下而上的动员模式为老年人提供丰富的体育资源[13],多元治理行为主体应互为补充,协同合作,推动老年人健身科学健康发展,让改革的成果更多地惠及每一位老年人。

参考文献:

[1] 郅玉玲.和谐社会语境下的老龄问题研究[M].杭州:浙江大学出版社,2011:14-15.

[2] 岳颂东.对我国建立老年护理制度的初步构想[J].决策咨询通讯,2008(3):90-91.

[3] 杜鹏.人口老龄化与老龄问题[M].北京:中国人口出版社,2006:27.

[4] 窦晓璐,约翰·派努斯.城市与积极老龄化:老年友好型城市建设的国际经验[J].国际城市规划,2015(3):117-123..

[5] 耿兴敏,高锋.鲁南地区老年体育与康乐家园建设[J].中国老年学杂志,2018,38(13):3266.

[6] 李宏伟,牛坤,邹家艳,等.基于老年人体育锻炼行为下的健商与幸福指数[J].中国老年学杂志,2019,39(1):100.

[7] 朱佳滨,李松梅,王学如.城市社区社会体育指导员发展构想[J].成人教育,2019(4):85-86.

[8] 周玉强,闫民.山东省公益性社会体育指导员队伍发展困境及优化路径[J].山东体育学院学报,2017,33(6):28.

[9] 周玉强,闫民.山东省社会体育指导员队伍建设存在问题分析[J].中国成人教育,2016(11):120.

[10] 颜小燕,王奇.基于人口老龄化视域:对城市社区老年体育健身服务的实证性研究——以安徽省部分城市社区为例[J].西安体育学院学报,2013,30(3):306.

[11] 邓陈亮,李樑,陆水萍.城市老年人体育锻炼的行为特征[J].中国老年学杂志,2020,40(1):193.

[12] 陈昱全,左群.社会生态学理论视角下我国老年人体育锻炼行为研究展望[J].中国健康教育,2019,35(5):438.

[13] 戴志鹏,马卫平.人口老龄化背景下我国老年人体育的发展动向研究——基于全面推进居家养老服务的思考[J].南京体育学院学报(社会科学版),2017,31(1):22.

(上接第49页)

参考文献:

[1] MILLER C, GLENDOWNE D, COOK H,et al.Insights gained from constructing a large scale dynamic analysis platform[J]. Digital Investigation,2017,22(6):48-56.

[2] 董刚,余伟,玄光哲.高级持续性威胁中攻击特征的分析与检测[J].吉林大学学报(理学版),2019,57(2):339-344.

[3] 刘毅,陈泽茂,沈昌祥.恶意代码的机理与模型研究[J].计算机工程与设计,2008(22):5709-5712.

[4] 钟志明,汪杰.基于电力监控系统虚拟沙箱的异常攻击监测技术[J].网络安全技术与应用,2019(11):126-128.

[5] 赵广强,凌捷.基于HOOK技术的进程管理系统研究[J].计算机工程与设计,2014,35(7):2325-2329.

[6] 左晓军,董立勉,曲武.基于域名系统流量的Fast-Flux僵尸网络检测方法[J].计算机工程,2017,43(9):185-193.

[7] 赵毅.恶意代码分析系统的研究与实现[D].南京:东南大学,2015.