基于软件定义网络的恶意网站防护系统

陈晓帆,黎志勇,李宁

中山大学信息科学与技术学院,广州 510006

摘要　基于恶意网站防护及软件定义网络等相关技术,设计了基于软件定义网络的恶意网站防护系统,通过多组实验验证该系统各模块功能,并将系统部署于真实的校园网环境。实验结果表明,基于软件定义网络的恶意网站防护系统能有效防范恶意网站的攻击,为对抗恶意网站攻击提供良好的技术支持。

关键词　软件定义网络;恶意网站;恶意网站防护;恶意域名;未来互联网

中图分类号　TP393.1　　**文献标志码**　A　　**doi**　10.3981/j.issn.1000-7857.2015.05.015

Protection system against malicious web sites based on software defined network

CHEN Xiaofan, LI Zhiyong, LI Ning

School of Information Science and Technology, Sun Yat-sen University, Guangzhou 510006, China

Abstract　Based on the technology of the protection against malicious web sites and the SDN (software defined networks), this paper puts forward a protection system against malicious web sites based on the SDN. Several experiments are conducted to verify the performance of each module of the system. The system is applied to a real campus network environment. And the testing results show that our system can effectively prevent malicious web sites and provide a good support against malicious attacks.

Keywords　software defined network; malicious web site; SDN; malicious URL; future internet

长期以来,恶意网站一直都是木马、病毒传播和网民染毒的主要途径之一。恶意网站利用操作系统或软件的安全漏洞,在网页内嵌入恶意的病毒、蠕虫以及木马等,当用户访问这些网页时,内嵌的恶意程序会在用户不知情的情况下,强行修改用户操作系统和应用软件的配置信息,导致成为僵尸系统,严重影响互联网的可信性,制约互联网应用发展[1,2]。

传统的针对恶意网站的安全工具,如防火墙、入侵检测设备、漏洞扫描、杀毒软件等要求用户在PC客户端上安装杀毒软件或安全浏览器,或者在网络出口端安装流量审计软件。而客户端安装软件并非所有用户都会安装或使用。网络出口流量审计一般是事后审计,可能已经对用户造成了损失。为了避免这种情况,本文提出基于软件定义网络的恶意网站防护系统。该系统主要利用软件定义网络控制层面和数据层面分离的特点,将恶意网站防护系统部署于控制器端,运用黑白名单检测和可疑域名智能检测的方法,最大限度对恶意和可疑网址域名进行过滤,保障用户上网安全。

1　恶意网站防护和软件定义网络技术

1.1　恶意网站防护技术

随着中国互联网应用的不断普及和网民数量的不断增加,网页种植木马技术不断发展,恶意网站给网民带来的安全威胁正在迅速增加。根据恶意网站的攻击类型,可以将恶意网页分为3类:1) 挂马,主要指被黑客或恶意程序修改的正常网页文件,在用户不知情下,访问网页资源的同时也连

收稿日期:2014-07-14;修回日期:2014-08-13

作者简介:陈晓帆,博士研究生,研究方向为网络安全、未来互联网及SDN的分布式入侵检测,电子信箱:chxfan@mail2.sysu.edu.cn;黎志勇(共同第一作者),硕士研究生,研究方向为计算机网络与信息处理,电子信箱:lizhiy8@mail2.sysu.edu.cn

引用格式:陈晓帆, 黎志勇, 李宁. 基于软件定义网络的恶意网站防护系统[J]. 科技导报, 2015, 33(5): 93-99.

接到黑客控制的服务器,并下载执行相应的恶意脚本;2)中继,将http访问请求进一步中继到下一个指定的服务器;3)恶意脚本,利用客户端的系统和应用程序漏洞下载执行木马程序,是实际危害客户端系统的罪魁祸首[3]。

目前对恶意网站的防护主要采用静态和动态两种方式。静态方式已经被大多数防病毒软件采纳,并集成在目前的软件产品中。该方式主要使用传统的特征匹配原理,对恶意网页进行扫描,且需要实时维护升级跟上恶意网页的增加速度,其优势在于判定速度快、消耗资源少。动态方式主要通过蜜罐蜜网技术[4]和沙盘(sandboxie)技术,具有代表性的主要有:微软公司的HoneyMonkey项目[5],Honeynet项目下的Capture-HPC[6]及奇虎公司的360安全浏览器,动态实时访问可疑网站,并对所有浏览造成的影响进行监控。此外,美国华盛顿大学Moshchuk等[7]提出了将网络搜索引擎技术和虚拟机技术结合的对恶意网站进行防护。

1.2 软件定义网络技术

2008年,美国斯坦福大学Cleanslate研究组[8]提出一种新型的基于软件技术的网络架构——软件定义网络(software defined network,SDN),其最大的特点是控制平面与数据平面的松耦合性、网络状态控制的集中化支持和实现底层网络设施对上层应用的透明化。SDN具有灵活的软件编程能力,促使网络的自动化管理与控制能力得到空前的提升,可以有效解决当前网络系统面临的资源规模受限、组网灵活性不足、难以快速部署业务等问题[9,10]。

如图1所示,该技术是把原有封闭的网络体系解耦为数据平面、控制平面和应用平面,将控制功能从网络设备中分离出来,并为网络应用提供可编程的接口,自底向上分别为基础设施层、控制层和应用层。其中,控制层中控制软件与基础设施中的交换/路由等网络设备经由控制数据面接口(也称南向接口)交互,与应用层各种应用经由开放应用程序编程接口API(也称北向接口)交互,网络基础设施由原交换/路由设计中的转发面而抽象为数据面角色,从而革命性地改变了现有的网络架构[11]。

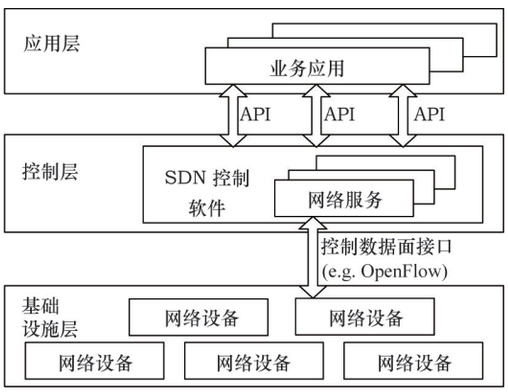

图1 SDN体系结构

Fig. 1 Architecture of SDN

使用OpenFlow技术的网络被称为OpenFlow网络。OpenFlow主要由OpenFlow交换机、FlowVisor和Controller 3个部分组成。其中OpenFlow交换机由流表、安全通道和OpenFlow协议3个部分组成。OpenFlow交换机主要进行数据平面的数据转发,而其中的流表由包头域、计数器和操作3个部分组成;FlowVisor负责对网络进行虚拟化;Controller则负责对网络进行集中控制,实现控制层功能。

目前为止,OpenFlow技术已经部署在美国斯坦福大学、日本JGN2plus等多个科研机构。网络设备生产商Cisco、HP、NEC、Juniper等也陆续推出支持OpenFlow技术的有线和无线交换设备。Google公司的广域流量管理系统[12]和Nicira公司的网络虚拟化平台等早期商用成功案例,也表明OpenFlow技术已经被使用,但相关应用仍相对较少。

2 基于SDN的恶意网站防护系统的架构设计

2.1 系统物理架构

如图2所示,实验网络采用两层网络架构园区网模型,接入+汇聚设备。接入PC客户端的交换机作为接入交换机,连接汇聚交换机,汇聚交换机接入互联网,汇聚交换机与接入交换机都与SDN控制器直接相连。

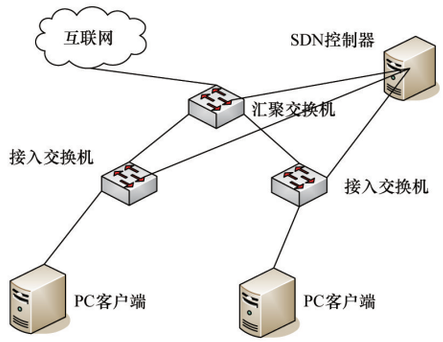

图2 实验网络系统物理架构

Fig. 2 Architecture of lab network system

2.2 系统逻辑结构

基于软件定义网络的恶意网站防护系统逻辑结构如图3所示,主要包括数据包解析模块、白名单检测模块、黑名单检测模块、名单修改模块、可疑域名智能检测模块及转发模块等。

2.3 数据包解析模块

OpenFlow Switch会收到很多来自各层设备的数据包,对于没有流表可循的会询问SDN控制器。因此,如图4所示,在数据包解析模块中,每个Switch连接时,SDN控制器就下发流表,让DNS request包直接送到控制器。接下来,对OpenFlow Switch送上来的Packet_in消息进行解析,对于DNS请求的数据包,解析DNS头部获取相关信息,对于Opcode数值小于10的DNS请求,提取相关域名信息直接进入后续各检测模块,否则经过名单修改模块后,再进入各检测模块。

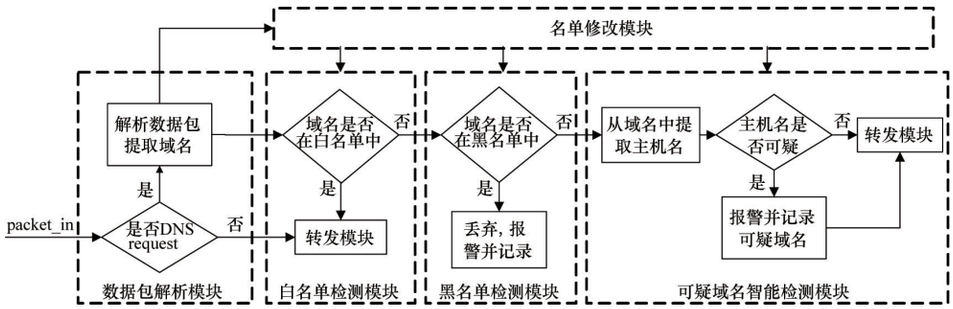

图3　恶意网站防护系统框架

Fig. 3　Framework of the protection system against malicious web sites

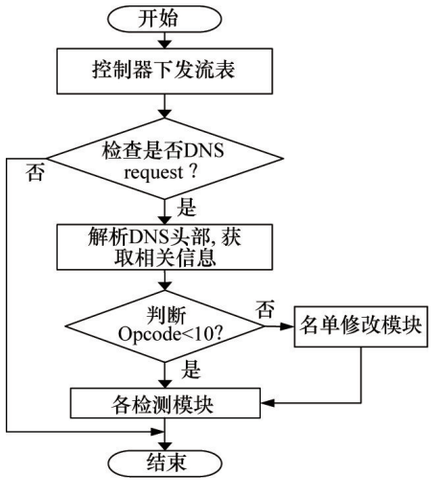

图4　数据包解析模块

Fig. 4　Packet parsing module

2.4　白名单检测模块

白名单是基于可信网站URL优先通过的规则。如图5所示，依次将可信域名列表读入，逐条存放到CBF（counting bloom filter）[13]，形成WL_CBF（white list CBF）。然后，对解析模块得到的域名信息进行多个哈希运算，判断该元素是否在WL_CBF中。从数据包解析模块提取的域名信息，凡符合白名单列表规则的，就直接进入转发模块，否则进入黑名单检测模块。数据来源：GFW白名单（https://github.com/n0wa11/gfw_whitelist/blob/master/whitelist.pac），以此白名单为基础，用网络爬虫获取了每个可信网址首页上的链接，经过消除重复、删除恶意网址等处理后，获得的白名单包含21371个域名。

2.5　黑名单检测模块

黑名单是基于恶意网站URL不能通过的规则。如图6所示，与白名单检测模块类似，将恶意域名列表读入，逐条存放到CBF[13]中，形成BL_CBF（black list CBF）。对送来的域名进行多个哈希运算，判断其是否在BL_CBF中。从白名单检测模块送来的域名信息，若符合黑名单列表的规则，则直接丢弃并报警和记录。数据来源：恶意网站实验室的恶意网站屏蔽列表（http://www.mwsl.org.cn/hosts/hosts），表中收录13923个恶意域名。

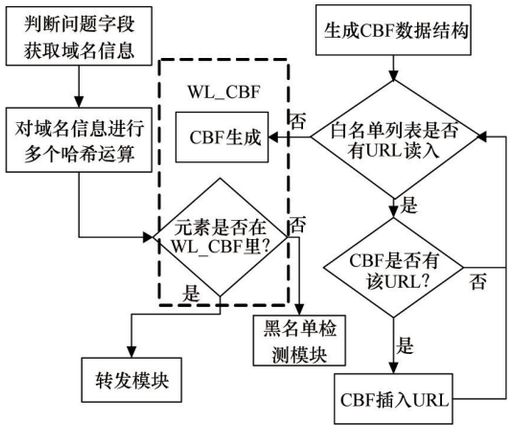

图5　白名单检测模块

Fig. 5　White list detection module

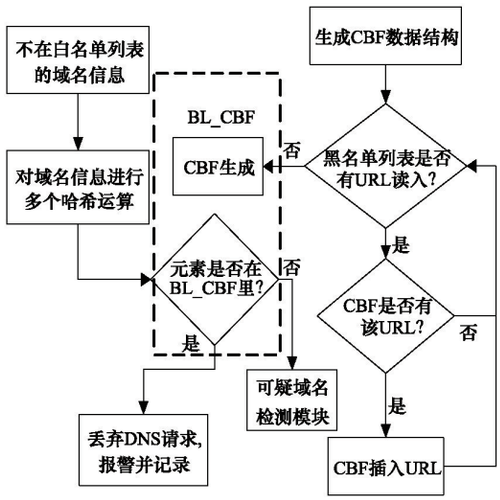

图6　黑名单检测模块

Fig. 6　Black list detection module

2.6 可疑域名智能检测模块

不满足白黑名单检测模块条件的网站URL暂且被认定为潜在可疑域名。如图7所示,该模块对潜在可疑域名进行必要的处理与检测。

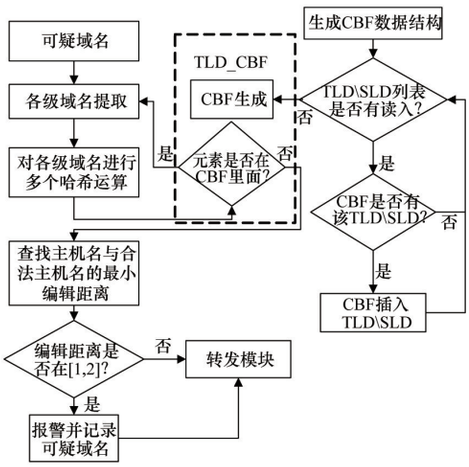

图7 可疑域名智能检测模块

Fig. 7 Intelligent detecting module for suspected domain name

首先,对潜在可疑域名进行各级域名提取,如图8所示,最右边的顶级域名(top level domain, TLD)以及二级域名(second level domain, SLD),以此类推对域名进行提取。只要主机名是可信的,因为低一级的所有域名只能由高一级的域名分配,其对应的下一或多级域名也是可信;如果主机名是可疑的,则其下一或多级域名也是可疑的。为了得到其主机名,需要对各级域名进行多次提取并对各级域名进行多个哈希运算判断其是否在TLD_CBF,直到提取出最后的主机名再与真实的可信主机名进行比较。可疑主机名的特点是与可信主机名编辑距离[14]很小,但不为零。如可信网址www.baidu.com的主机名baidu和可疑域名www.baidv.com的主机名baidv,其编辑距离为1。从白名单中提取所有不重复的主机名,并存放在BK-tree[15]数据结构中。BK-tree在每次访问5%~8%个节点的情况下就能找到编辑距离最近的节点,并得到最小的编辑距离min *Dist*,计算开销非常小。对于编辑距离,采用动态规划算法求解,能有效降低计算开销。当$\alpha \leqslant \min Dist \leqslant \beta$时(经多次实验,选取$\alpha=1$, $\beta=2$时效果最优),认为域名是可疑的,控制器进行报警并记录可疑域名;否则认为域名是可信的,直接进入转发模块。

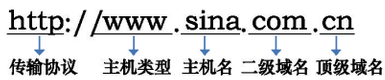

图8 完整域名示例

Fig. 8 Example of complete domain

2.7 名单修改模块

为方便网络用户在终端可对黑白名单及TLD\SLD名单列表进行个性化的删减,该模块对DNS报文首部标志位的Opcode字段的空闲值加以利用。如图9所示,Opcode字段占4位,用于指定查询的类型,值为0表示标准查询,值为1表示逆向查询,值为2表示查询服务器状态,值为3表示保留,值为4表示通知,值为5表示更新报文,值为6~15的作为新增操作。该模块重新定义了Opcode的10~15的数值及对应的操作(表1)。

图9 DNS报文首部标志位

Fig. 9 Flags in the head of DNS packet

表1 Opcode的数值与对应的操作

Table 1 Values of Opcode and the corresponding operations

Opcode	10	11	12	13	14	15
对应操作	将域名添加到BL_CBF中	从BL_CBF中删除域名	将域名添加到WL_CBF中	从WL_CBF中删除域名	将域名添加到TLD_CBF中	从TLD_CBF中删除域名

2.8 转发模块

OpenFlow交换机的处理单元由流表构成,每个流表由许多表项组成,流表项则代表转发规则,进入交换机的数据包通过查询流表取得对应的操作。因此,在恶意网站防护系统中,在SDN控制器上建立一个动态的源MAC地址与交换机端口的映射表,然后根据交换机送上来的数据包,把流表项动态下发安装到各台OpenFlow Switch上,使得OpenFlow交换机能对数据包正确转发。

3 系统功能测试

设计的恶意网站防护系统采用POX[16]和基于NetFPGA的OpenFlow Switch搭建了一个真实的校园网SDN网络平台,控制器与OpenFlow之间的通信协议为OpenFlow V1.0[8](图10),3台装有可编程网卡1 G NetFPGA[17,18]的主机PC-A、PC-B、PC-C实现OpenFlow Switch的功能,PC-D实现OpenFlow Switch Controller的功能,图中PC-0、PC-1是客户端PC机,实时发起DNS访问请求。其中,PC客户端接入层交换机PC-A与PC-C,PC-B作为汇聚交换机接入Internet,PC-A、PC-B、PC-C作为OpenFlow Switch与PC-D相连,PC-D作为Controller,对网络中的OpenFlow Switch集中控制,实现控制层功能。

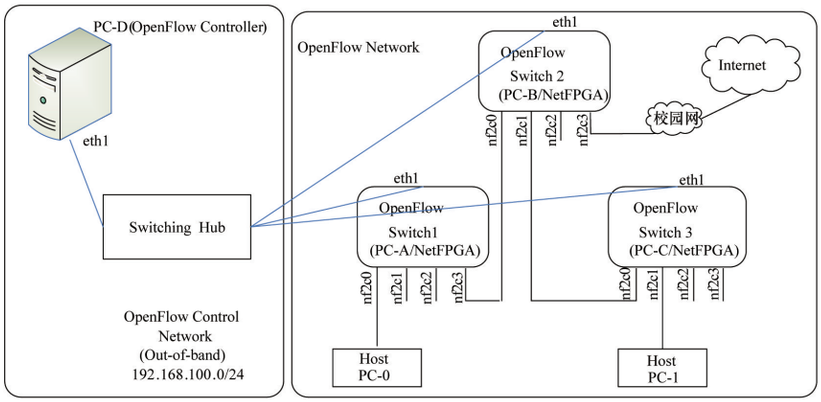

图10　采用POX和基于Netfpga的OpenFlow Switch校园网SDN网络平台

Fig. 10　SDN network platform based on POX and Netfpga-based OpenFlow Switch

3.1　持续的域名访问服务

PC客户端访问网站时使用域名访问,需要进行DNS查询翻译成IP地址。本系统实验需要模拟PC客户端持续发起域名访问服务,因此客户端将使用Scapy[19]编写一个dnsv1.py的持续域名访问服务模块,该模块每隔10 s生成相应列表(黑名单、白名单或者TLD\SLD名单)中的域名访问包,模拟PC客户端对名单内多个网站进行访问的情况。

3.2　系统功能工作流程

根据图3,恶意网站防护方案工作流程为:

1) PC客户端使用Scapy编写的dnsv1.py模拟发起相应名单列表域名的DNS请求。

2) 每个OpenFlow Switch与SDN控制器建立连接时,SDN控制器就下发静态流表给各OpenFlow Switch,使OpenFlow Switch收到的所有DNS request的流封装成Packet_in消息直接送到SDN控制器。

3) SDN控制器根据OpenFlow Switch上发的报文进行包解析,先进入白名单检测模块与本地的白名单进行匹配,匹配成功则进入转发模块,否则进入黑名单检测模块,与本地的黑名单进行匹配,匹配成功,则对黑名单内的域名的包进行丢弃,否则进入可疑域名智能检测系统。进入可疑域名智能检测系统的域名首先去TLD\SLD,然后对hostname(主机名)进行相似度对比,记录并报警其编辑距离在1到2之间的可疑域名,并且进行动态流表分发,使之接入校园网,经过网关接入Internet,实现DNS请求响应。

3.3　系统实验及结果分析

为验证本文恶意网站防护系统的可用性,设定了4组部署于真实校园网环境的实验,分别为白名单网站访问、恶意网站访问、可疑网站访问、黑白和TLD\SLD名单列表的增减。

3.3.1　白名单网站访问

如图11所示,客户端PC-0对白名单内“www.sina.com.cn”发出DNS访问请求,通过Wireshark观察客户端端口数据包情况,抓到“www.sina.com.cn”的DNS请求和响应包,网页正常访问。如图12所示,OpenFlow Switch 1以及OpenFlow Switch 2已安装对应流表项。方框中显示OpenFlow Switch 1基于目标端口号53和输出端口为4接入汇聚层交换机的动态流表项,OpenFlow Switch 2基于目标端口号53和输出端口为4接入校园网的动态流表项。

8558 80.173703	172.18.216.49	222.200.160.1	DNS	Standard query A www.sina.com.cn
8559 80.180928	222.200.160.1	172.18.216.49	DNS	Standard query response CNAME jupiter.sina.com.cn

图11　Wireshark观察客户端端口数据包情况

Fig. 11　View of wireshark in client PC

```
 cookie=0, duration_sec=1s, duration_nsec=750000000s, table_id=0, priority=6553
5, n_packets=1, n_bytes=81, idle_timeout=10,hard_timeout=30,udp,in_port=1,dl_vla
n=0xffff,dl_vlan_pcp=0x00,dl_src=14:cf:92:f9:ff:6b,dl_dst=00:0f:e2:c9:3d:83,nw_s
rc=172.18.216.49,nw_dst=222.200.160.1,nw_tos=0x00,tp_src=48556,tp_dst=53 actions
=output:4
```

(a) Switch 1

```
 cookie=0, duration_sec=4s, duration_nsec=419000000s, table_id=0, priority=655
35, n_packets=1, n_bytes=72, idle_timeout=10,hard_timeout=30,udp,in_port=1,dl_v
lan=0xffff,dl_vlan_pcp=0x00,dl_src=14:cf:92:f9:ff:6b,dl_dst=00:0f:e2:c9:3d:83,n
w_src=172.18.216.49,nw_dst=222.200.160.1,nw_tos=0x00,tp_src=43409,tp_dst=53,act
ions=output:4
```

(b) Switch 2

图12　动态流表项

Fig. 12　Flow entries

3.3.2 恶意网站访问

客户端PC-0加载持续域名访问服务模块,模拟持续地发起恶意网站域名访问服务,PC客户端对恶意网站进行DNS访问请求,通过Wireshark观察客户端端口数据包情况如图13所示,抓到“download.filesfrog.com”,“www.umof.info”,“varzo.webs.com”,“indah1.webs.com”等网站DNS访问请求,没有收到恶意网站的DNS响应包。

358 231.095398	172.18.216.49	222.200.160.1	DNS	Standard query A download.filesfrog.com
782 241.135246	172.18.216.49	222.200.160.1	DNS	Standard query A www.umof.info
253 251.172629	172.18.216.49	222.200.160.1	DNS	Standard query A varzo.webs.com
532 261.212176	172.18.216.49	222.200.160.1	DNS	Standard query A indah1.webs.com

图13 Wireshark观察客户端端口数据包情况

Fig. 13 View of wireshark in client PC

查看SDN控制器发现恶意网站访问警报信息,如图14所示,表明这类网站为恶意网站,拒绝该网站的DNS解析请求服务,因此客户端端口没有收到恶意网站的DNS响应包。

```
1 find URL in BlackList: download.filesfrog.com
from mac: 14:cf:92:f9:ff:6b, ip: 172.18.216.49
time: Fri Jun  6 18:34:35 2014

2 find URL in BlackList: www.umof.info
from mac: 14:cf:92:f9:ff:6b, ip: 172.18.216.49
time: Fri Jun  6 18:34:45 2014

3 find URL in BlackList: varzo.webs.com
from mac: 14:cf:92:f9:ff:6b, ip: 172.18.216.49
time: Fri Jun  6 18:34:56 2014

4 find URL in BlackList: indah1.webs.com
from mac: 14:cf:92:f9:ff:6b, ip: 172.18.216.49
time: Fri Jun  6 18:35:06 2014
```

图14 SDN控制器恶意网站访问警报信息

Fig. 14 Alarm information of malicious website in SDN controller

3.3.3 可疑网站访问

客户端PC-0加载持续域名访问服务模块,模拟持续地发起可疑域名访问服务,PC客户端对可疑网站进行DNS访问请求,通过Wireshark观察客户端端口数据包情况,如图15所示。

50.581088	172.18.216.49	222.200.160.1	DNS	Standard query A www.baidv.com
60.620904	172.18.216.49	222.200.160.1	DNS	Standard query A www.batde.com
70.662311	172.18.216.49	222.200.160.1	DNS	Standard query A www.baate.com
80.702441	172.18.216.49	222.200.160.1	DNS	Standard query A www.lcace.com
90.738602	172.18.216.49	222.200.160.1	DNS	Standard query A www.lcbc.com
100.773715	172.18.216.49	222.200.160.1	DNS	Standard query A www.lcucee.com
110.811306	172.18.216.49	222.200.160.1	DNS	Standard query A www.167.com

图15 Wireshark观察端口数据包情况

Fig. 15 View of wireshark in client PC

结合表2和图16信息可以验证,SDN控制器对于编辑距离在[1,2]的可疑网站域名,进行警报并记录后进入转发模块,否则直接进入转发模块。

表2 可疑网站列表

Table 2 List of suspicious websites

DNS请求域名	主机名	参考主机名	编辑距离
www.baidv.com	baidv	baidu	1
www.batde.com	batde	baidu	1
www.baate.com	baate	baidu	3
www.lcace.com	lcace	lacare	2
www.lcbc.com	lcbc	icbc	1
www.lcucee.com	lcucee	icbc	4
www.167.com	167	163	1

```
suspect url: baidv, similar to: baidu, minEditDist: 1
suspect url: baidv, similar to: baidu, minEditDist: 1
suspect url: batde, similar to: baidu, minEditDist: 2
suspect url: batde, similar to: baidu, minEditDist: 2
suspect url: baate, similar to: baxue, minEditDist: 2
suspect url: baate, similar to: baxue, minEditDist: 2
suspect url: lcace, similar to: lacare, minEditDist: 2
suspect url: lcace, similar to: lacare, minEditDist: 2
suspect url: lcbc, similar to: icbc, minEditDist: 1
suspect url: lcbc, similar to: icbc, minEditDist: 1
suspect url: 167, similar to: 67, minEditDist: 1
suspect url: 167, similar to: 67, minEditDist: 1
```

图16 SDN控制器恶意网站访问警报信息

Fig. 16 Alarm information of malicious website in SDN controller

3.3.4 黑白名单列表和TLD\SLD名单列表的增减

为了满足终端用户可以对黑白名单域名及TLD\SLD列表增减的需求,系统增加了名单修改模块。如表1所示。由于名单修改模块对各名单修改原理一致,因此,仅以黑名单列表增加为例。

客户端PC-0加载持续域名访问服务模块,发起对黑名单列表增加域名的访问服务,如图17所示,其域名列表增加名单为“BLadd.txt”,Opcode数值为10。如图18所示,SDN控制器完成黑名单添加并显示增加列表信息。

dnsv1.py

```
#!/usr/bin/python

from scapy.all import *
from time import sleep

#fd = open("MWSL_ALL.txt",'r')
fd = open("BLadd.txt",'r')
i = 1
dns_server = "222.200.160.1"

while True:
        q = fd.readline().strip()  # use strip to take off '\n'
        if not q: # to the end of file, url is null
            break
        print i ,q
        i += 1
        send(IP(dst=dns_server)/UDP(sport=RandShort())/DNS
(qr=0,opcode=10,rd=1,qd=DNSQR(qname=q,qtype=1,qclass=1)))
        sleep(10)
```

图17 持续黑名单域名访问服务模块

Fig. 17 Accessing service module of persisting black list domain name

```
add URL in BlackList: www.baidv.com
from mac: 14:cf:92:f9:ff:6b, ip: 172.18.216.49
time: Sun Jun  8 13:33:47 2014

add URL in BlackList: www.batde.com
from mac: 14:cf:92:f9:ff:6b, ip: 172.18.216.49
time: Sun Jun  8 13:33:57 2014

add URL in BlackList: www.baate.com
from mac: 14:cf:92:f9:ff:6b, ip: 172.18.216.49
time: Sun Jun  8 13:34:07 2014

add URL in BlackList: www.lcace.com
from mac: 14:cf:92:f9:ff:6b, ip: 172.18.216.49
time: Sun Jun  8 13:34:17 2014

add URL in BlackList: www.lcbc.com
from mac: 14:cf:92:f9:ff:6b, ip: 172.18.216.49
time: Sun Jun  8 13:34:27 2014
```

图18　黑名单列表域名增加检测结果

Fig. 18　Result of adding black list domain

3.4　实验总结

经过多组实验验证,基于软件定义网络的恶意网站防护系统能够有效防范恶意网站的攻击,系统各模块能有机协调工作。与传统的恶意网站防护系统相比,基于软件定义网络的恶意网站防护系统优点为:

1）客户端免于安装安全工具。该系统部署于SDN控制器端,客户端对于恶意网站的访问请求将会在控制器端予以防护,因此该系统所在的SDN网络是安全可靠的,客户端不用安装安全工具。

2）控制中心统一管理。该系统部署于SDN控制器端,对系统黑白名单以及TLD\SLD的升级更新将会在控制器端统一进行,控制器端还可以全网监视各设备的安全情况,对问题设备进行报警。

3）网络可编程及高数据传输速率。该系统的硬件基础设备是基于NetFPGA的OpenFlow交换机,网卡是有4组速率高达1 Gbps的标准以太网接口的可编程网卡。该系统不仅可根据网络要求进行编程,且具备较高的数据传输性能。

4）网络具有良好的扩展性。基于软件定义网络的恶意网站防护系统,可以增加OpenFlow交换机的设备扩展网络的规模。该系统在SDN控制器端独立运行,网络管理员可根据网络业务的需求在SDN控制器端扩展网络应用。

4　结论

基于OpenFlow技术的SDN网络革命性地颠覆了现有的网络架构,是目前研究下一代网络技术的重要方向。SDN实现了网络设备的控制功能与数据转发分离的管控策略,目的是对现有复杂的网络控制面进行抽象简化,使控制面能独立创新发展,使得网络面向应用可编程。基于上述思想,本文设计了基于软件定义网络的恶意网站防护系统,经过实际测试,能有效地防范恶意网站的攻击,解决了当前存在的实际问题,增加了网络的可管可控性。

参考文献(References)

[1] 何公道, 王江民. 我国恶意网站现状及防治对策研究[J]. 中国人民公安大学学报: 自然科学版, 2008(3): 1-4.
He Gongdao, Wang Jiangmin. Study on current situation and countermeasures of malicious web sites in China[J]. Journal of Chinese People's Public Security University: Natural Science Edition, 2008(3): 1-4.

[2] 周佩颖. 恶意的URL捕获分析系统[D]. 成都: 电子科技大学, 2010.
Zhou Peiying. Malicious URL capture and analysis system[D]. Chengdu: University of Electronic Science and Technology of China, 2010.

[3] Sheng S, Wardman B, Warner G, et al. An empirical analysis of phishing blacklists[C]. Sixth Conference on Email and Anti- Spam (CEAS), California, USA, July 16-17, 2009.

[4] 程杰仁, 殷建平, 刘运, 等. 蜜罐及蜜网技术研究进展[J]. 计算机研究与发展, 2009(S1): 375-378.
Cheng Jieren, Yin Jianping, Liu Yun, et al. Research progress of honeypot and honeynet technology[J]. Research and Development of Computer, 2009(S1): 375-378.

[5] Wang Y M, Beck D, Jiang X, et al. Automated web patrol with strider honeymonkeys[C]//Proceedings of the 2006 Network and Distributed System Security Symposium. San Diego, California, USA: The Internet Society, 2006, 35-49.

[6] Seifert C, Steenson R. Capture-honeypot client (capture-hpc)[EB/OL]. [2014-07-14]. http://projects.honeynet.org/capture-hpc.

[7] Moshchuk A, Bragin T, Gribble S D, et al. A crawler-based study of spyware in the web[C]. The 2006 Network and Distributed System Security Symposium, San Diego, California, USA, February 1-2, 2006.

[8] McKeown N, Anderson T, Balakrishnan H, et al. OpenFlow: Enabling innovation in campus networks[J]. ACM SIGCOMM Computer Communication Review, 2008, 38(2): 69-74.

[9] Open Network Foundation. Software defined networking: The new norm for networks[EB/OL]. [2014-07-14]. https://www.opennetworking.org/images/stories/downloads/sdn-resources/white-papers/wp-sdn-newnorm.pdf.

[10] 范伟. 软件定义网络及应用[J]. 通信技术. 2013, 46(3): 67-70.
Fan Wei. Software defined network and application[J]. Communications Technology, 2013, 46(3): 67-70.

[11] 左青云, 陈鸣, 赵光松, 等. 基于OpenFlow的SDN技术研究[J]. 软件学报, 2013, 24(5): 1078-1097.
Zuo Qingyun, Chen Ming, Zhao Guangsong, et al. Research on OpenFlow-based SDN technologies[J]. Journal of Software, 2013, 24(5): 1078-1097.

[12] Jain S, Kumar A, Mandal S, et al. B4: Experience with a globally-deployed software defined wan[J]. ACM SIGCOMM Computer Communication Review, 2013, 43(4): 3-14.

[13] Fan L, Cao P, Almeida J, et al. Summary cache: A scalable wide-area web cache sharing protocol[J]. IEEE/ACM Transactions on Networking, 2000, 8(3): 281-293.

[14] 郑礼雄, 李青山, 李素科, 等. 基于域名信息的钓鱼URL探测[J]. 计算机工程, 2012, 38(10): 108-110.
Zheng Lixiong, Li Qingshan, Li Suke, et al. Detection of URL domain name information[J]. Computer Engineering, 2012, 38(10): 108-110.

[15] Burkhard W A, Keller R M. Some approaches to best-match file searching [J]. Communications of the ACM, 1973, 16(4): 230-236.

[16] Nicira. NOX repository website[EB/OL]. [2014-07-14]. http://noxrepo.org.

[17] NetFPGA Team. NetFPGA website[EB/OL]. [2014-07-14]. http://netfpga.org.

[18] Naous J, Erickson D, Covington G A, et al. Implementing an OpenFlow switch on the NetFPGA platform[C]. ANCS 2008 Symposium on Architecture for Networking and Communications Systems, San Jose, California, USA, November 06-07, 2008.

[19] Biondi P. Scapy[EB/OL]. [2014-07-14]. http://www. secdev. org/projects/scapy.

（责任编辑　王媛媛）