

## 基于网络流量的僵尸网络动态检测模型

成淑萍<sup>1</sup>, 谭 良<sup>2,3</sup>

(1. 四川文理学院计算机学院, 四川 达州 635001; 2. 四川师范大学计算机学院, 成都 610068;  
3. 中国科学院计算技术研究所, 北京 100190)

**摘 要:** 针对利用先验知识不能检测新型或变异僵尸网络(Botnet)的现状, 提出一种基于网络流量的 Botnet 动态检测模型。通过聚类分析通信流量并完成关联分析, 以鉴定 bot 之间的类似通信和恶意行为模式。该模型具有特征库更新和检测模型生成的动态性, 并且可以处理来自不同僵尸网络的数据, 其检测体系结构与协议和 Botnet 的先验知识无关。实验结果验证了该模型的有效性和准确性。

**关键词:** 网络安全; 僵尸网络; 恶意代码; 网络流量; 动态检测

**中文引用格式:** 成淑萍, 谭 良. 基于网络流量的僵尸网络动态检测模型[J]. 计算机工程, 2014, 40(11): 106-112.

**英文引用格式:** Cheng Shuping, Tan Liang. Dynamic Detection Model in Botnet Based on Network Traffic [J]. Computer Engineering, 2014, 40(11): 106-112.

## Dynamic Detection Model in Botnet Based on Network Traffic

CHENG Shuping<sup>1</sup>, TAN Liang<sup>2,3</sup>

(1. College of Computer, Sichuan University of Arts and Science, Dazhou 635001, China;

2. College of Computer, Sichuan Normal University, Chengdu 610068, China;

3. Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China)

**【Abstract】** For the status quo that the Botnet detection of a priori knowledge to get the matching and protocol-related are unable to be suitable for new or mutated Botnet detection, this paper proposes a dynamic Botnet detection model based on network traffic. By using clustering, it analyzes traffic and completes the correlation analysis to identify similar between bot communication and malicious behavior patterns. The test architecture has nothing to do with the agreement and Botnet prior knowledge. The model has three dynamic characteristics, such as the characteristics of library updated, detection model generation, and handling the network traffic from the dynamic Botnet. Finally, the effectiveness and the accuracy are verified by the experimental data.

**【Key words】** network security; Botnet; malicious code; network flow; dynamic detection

**DOI:** 10.3969/j.issn.1000-3428.2014.11.021

### 1 概述

僵尸网络(Botnet)是通过 C&C 通信频道控制的一群相互协作恶意软件实体。一个 Botnet 的基本特性是 bots 以类似或者相关的方式通过一些 C&C 服务器或者节点进行通信, 完成恶意攻击活动。自从僵尸网络的发现、监测和预防成为国内外研究者所共同关注的热点后, 国内外网络安全的研究专家和学者都从不同的角度对僵尸网络的检测与追踪进行了研究。基于流量聚类分析僵尸网络检测机制是通过检测僵尸网络一些异常网络流量影响因子, 如很高的网络延迟、巨大的流量、不寻常端口的流量和

异常的系统行为等都代表网络中有恶意代码的存在<sup>[1]</sup>。

目前的僵尸网络检测方法存在着检测对象单一性、误报率高、漏报率高、效率低等问题, 本文在现有僵尸网络检测的模型结构中加入特征库, 并且使特征库能动态更新, 对不同僵尸网络可以动态生成特征模型, 从而提高检测效率, 降低误报率和漏报率。

### 2 相关研究

命令控制机制是所有类型的僵尸网络核心, 用于实现信息的交互、接收和执行控制者发出的指

**作者简介:** 成淑萍(1988 -), 女, 助教、硕士, 主研方向: 网络安全; 谭 良, 教授、博士。

**收稿日期:** 2013-09-12 **修回日期:** 2013-12-25 **E-mail:** csp16850@163.com

令,再将执行结果信息反馈给控制者。针对这点可以采用流量分析的方法来进行有效检测。文献[2]针对基于IRC协议的僵尸网络的特征,提出一种主动监视网络流量,来寻找可疑的IRC昵称、IRC服务器和不常用的端口的检测方法。Strayer针对基于IRC协议的僵尸网络将流量分析分为2步:(1)将IRC通信流量从正常的网络中分离;(2)将Botnet的C&C流量从正常的IRC流量中分离,但其只对基于IRC协议的僵尸网络起作用。文献[3]针对恶意代码感染主机的5个会话阶段,提出了一种感染对话相关策略,通过发现在感染过程中的通信序列来检测出被僵尸程序成功感染的主机,并在BotHunter的网络监视系统中运用了该策略。该研究成果有助于理解恶意软件感染的生命周期从而找到有效的检测方法。文献[4]在本地网络中没有僵尸网络命令控制的服务器的地址和特征等先验信息的前提下,采用基于异常网络流量的方法来检测和鉴定僵尸网络的命令控制频道,这种方法可以识别网络中被感染的主机和C&C服务器。并将这种方法运用到BotSniffer的原型系统的实现中,用了很多真实世界的网络跟踪,得到的评估结果是BotSniffer拥有高准确率和低错误率的检测出真实的僵尸网络。但BotSniffer只针对基于IRC和HTTP协议僵尸网络的消息进行了检测,对于新型基于P2P技术的僵尸网络就失效了。文献[5]在BotMiner中将数据挖掘技术引入到基于流量分析的僵尸网络的检测方法中,使用流量聚类分析,得到相似的通信流量。文献[6]在研究P2P僵尸网络运行协议和机制的基础上,结合P2P僵尸网络建立连接的主机IP地址不是静态地址就是动态地址,提出了一种基于连接成功率的检测算法,但在大量的流量分析的效率和准确率方面存在缺陷。文献[7]在基于节点连接分布性和突发性的特征,过滤非P2P的节点,进而根据对称度值采用K均值聚类来发现P2P群,最后利用流行行为的相似性检测来确认。

针对目前已有僵尸网络检测工具对协同活动的僵尸网络缺少信息共享和配合,从而产生漏报和误报的现象,文献[8]提出一个层次协同模型,在信息、特征及决策3个级别上进行信息交互与协作。同时针对已有检测体结构的特征提取灵活性不好、缺少协同功能和协同方式单一等缺点,基于该层次协同模型,该文构建了Bot\_CODA——一个

新的僵尸网络协同检测体系结构。但该体系结构只是一个原型系统,并没有在真实的网络中运行,其有效性、效率及准确度也没进行验证。文献[9]提出了一个基于通信特征提取和IP聚集的僵尸网络相似性度量模型,通过在国家网络安全监测平台获取到的僵尸网络IRC服务器与bot的C&C通信数据;然后提取僵尸网络的通信流量特征、IP聚集和估算bot重叠率等度量指标,分别分析每个度量指标的优缺点;最后合并这些指标,提出了僵尸网络相似性度量模型,并用实验验证了其有效性和准确率。但该模型在计算通信特征曲线距离采用的是欧式距离,没有考虑曲线形状、均值、方等相关性;相似性度量模型采用的是没有考虑弃真和取伪错误率及其方差的权值系数。

目前已有异常网络流量分析技术的检测方法大多是要在基于先验知识的基础上或者协议相关,一旦出现新型的僵尸网络就失效了,或者只集中在一个特殊的指挥和控制的协议(IRC,HTTP)和结构(中央集中)的僵尸网络,当僵尸网络的命令控制协议和结构发生改变其检测方法就会失效,或者就依靠其他外部组件(蜜网等)来协同工作。而且各个检测部署点之间没有信息交流,就不能共享信息,每遇到一个新的僵尸网络就必须完成所有的检测步骤,因此减缓了检测的效率。

### 3 僵尸网络检测模型设计

本文在BotSniffer<sup>[4]</sup>的检测机制中加入了特征库,采用网络流量聚类分析,提出了一个新的检测模型——基于网络流量的僵尸网络动态检测模型。该模型的结构体系具有3个动态特性:(1)特征库更新的动态;(2)检测模型生成的动态;(3)处理的网络流量可以来自于动态的僵尸网络,其分别表现在该体系结构的入口网络流量、客户机的监测引擎和服务端处理引擎的特征库的管理模块,这3个动态特性使检测平台能达到检测出不同的僵尸网络,并能在不同的处理阶段加快基于网络流量的僵尸网络检测的处理速度。

图1展示了基于网络流量的僵尸网络动态检测平台的体系结构,该平台由客户端和服务端2个部分组成。其中客户端分为预处理、消息响应检测模块、恶意行为响应检测模块、关联引擎、特征库更新模块和产生报告6个组件构成;服务端由聚类分析、关联分析和特征库管理模块3个组件构成。

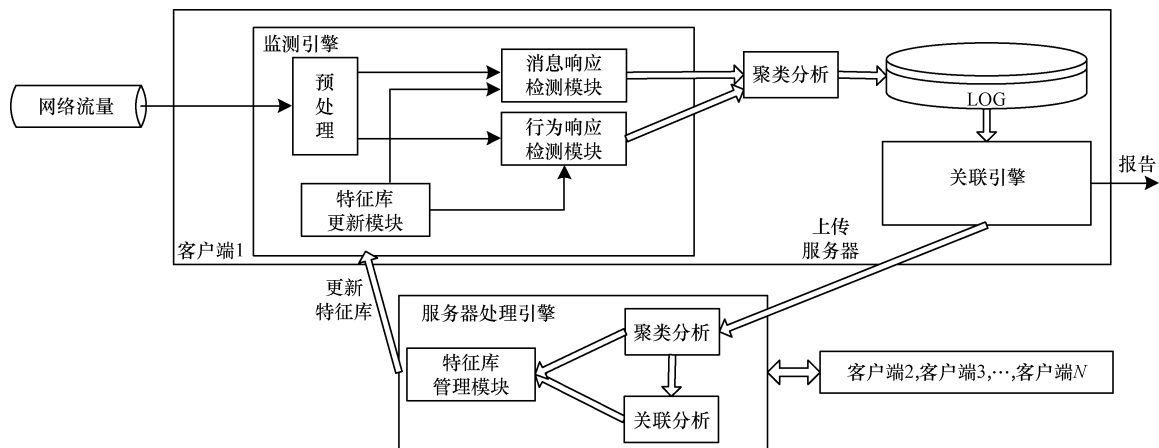


图1 基于网络流量的僵尸网络动态检测平台体系结构

### 3.1 客户端模块

#### 3.1.1 预处理

预处理主要负责过滤不相关的网络流量,以减少网络流量负载和提高检测平台后续的处理效率。在一定时段内捕获到的局域网出入的网络流量的数目是海量的,这为后续的处理会增加很多资源的消耗,增加处理的时间。为此本文在预处理模块的工作分为2步:基础过滤和白名单过滤;基础过滤是过滤那些没有建立连接的流量和目的IP地址不可能是被感染的僵尸网络服务器(如新浪、雅虎、Google等知名网站的服务器),黑名单为该局域网内的被怀疑的主机或者是处于重要位置的主机。

#### 3.1.2 网络流量分析

网络流量分析主要是为通信流量分析和行为响应模块分析提供所需要的数据,并先对数据进行一个提前处理,再次减少数据处理的工作量,在通信流量检测模块中,本文首先采用流量监控是在检查网络流量中检测出一组相类似行为和通信模式的主机。因此,就捕获网络流量和记录每一条数据流的一些特别的信息(源IP地址、目的IP地址、源端口、目的端口、持续时间、通信协议、数据包数量和包的字节数)。

本文使用抓包工具 Omnipcap 来监控流量和记录下所需要的信息,然后把这个信息插入到资源库中去,  $\{f_i\} (i=1,2,\dots,n)$  是网络流量的记录,每一个  $f_i$  代表一条网络流量。流量监控肯定不能一次性全部持续下去,否则即使用再好的配套硬件设施和开发出再好的配套软件都完成不了如此海量的数据,因此,本文把监控时间分成段,可以随意安排一个时间段的时间长度,把每个时间段的监控数据都保存下来。

#### 3.1.3 消息响应检测模块

消息响应检测模块主要工作对象是捕获的网络流和记录信息,这些信息是网络中是“谁跟谁对话”。在前面产生的记录信息库中,在这部分只提取出局

域网外向局域网内发起的基于 TCP 和 UDP 协议的连接或者通信的网络流,每条网络流量记录包括源 IP 地址、目的 IP 地址、源端口、目的端口、持续时间、通信协议、数据包数量和包的字节数。

在网络流量分析中记录了多个时间段监控的流量中,现在就把所有记录的  $n$  条数据流量中有相同的源 IP 地址,目的 IP 地址和相同协议(TCP 或者 UDP)都被标记为一个同类,分别利用式(1)将  $\{f_i\}$  归的同类计算出平均每秒字节数(nbps)、平均每包字节数(nbpp)和平均每秒包数(npss)。

$$\text{平均每秒字节数} = \frac{\text{字节总数}}{\text{持续时间}}$$

$$\text{平均每包字节数} = \frac{\text{字节总数}}{\text{包数}}$$

$$\text{平均每秒包数} = \frac{\text{包总数}}{\text{持续时间}} \quad (1)$$

接下来将 nbps、nbpp 和 npss 作为 3 个新值,加上前一步标记为同类数据流的源 IP 地址和目的 IP 地址插入到另外一个相似的流量资料库。因此,在指定的时间段,可以得到一组 data:  $\{d_i\} (i=1,2,\dots,m)$  这些 data 中每一条都有相同的源 IP 地址、目的 IP 地址、协议。

正如上文前面提到的, Bots 如果属于同一个僵尸网络,它们会有相类似的通信流量和行为响应模式,特别是在它们想从 BotMasters 更新命令、数据或者攻击一个目标,这种相似的模式就更为明显。

#### 3.1.4 行为响应检测模块

行为响应监视器主要分析对象是捕获的网络流和记录信息是“在做什么”,它通过监控局域网向外的网络流量,能够检测出局域网内主机可能正执行的恶意行为,如扫描活动(应用在恶意代码的传播或 DoS 攻击)<sup>[10]</sup>、垃圾邮件<sup>[11]</sup>、二进制下载(应用在恶意代码的更新)和尝试连接(应用在恶意代码的传播或者攻击)<sup>[12]</sup>,这些是最常见的僵尸网络能控制他的受控主机执行的恶意活动。同样本文在前面产生的记录信息中,提取出局域网外向局域网内发出



的基于 TCP 和 UDP 协议的连接或者通信网络流,为下一步的行为聚类分析提供数据。

目前对行为的检测的研究工作已经有很多成功和可用的成果。其中, Botminer<sup>[5]</sup> 检测系统的行为响应数据聚类分析能对目前所存在的大多数恶意行为进行分析,本文研究就直接采用了其结构和方法。

Botminer 结构如图 2 所示,在对于可能完成的恶意行为列表中,首先按照它们集群的活动类型(如扫描、发送垃圾邮件、恶意代码更新等)进行分类。这是第一层的聚类,然后是针对每类的恶意行为的具体特征进行聚类。对于扫描行为包括扫描端口、扫描目标等,扫描端口具体特征是如果 2 个主机扫描同一个端口就把它们归为一类,扫描目标具体特征是扫描的目标主机是同一台主机或者同一个子网;对于发送垃圾邮件的行为的聚类依据是如果 2 个主机的 SMTP 的连接目的地址是高度重叠就将他们聚在一起,对于此 Bot 可以被配置使用不同 SMTP 服务器来逃避检测,进一步可以考虑通过捕获整个 SMTP 服务器的流量来得到垃圾邮件的内容来进行检测,但这样工作量就增大不少,所以本文暂时只考虑前一种方法;对于恶意代码更新行为的聚类依据是(如果) 2 台主机下载相类似二进制数据, BotSniffer 中的一个距离函数可以实现计算出 2 个二进制文件的任何字符或者字符串之间的距离。在目前的实现中,本文根据不同的扫描端口进行扫描行为的聚类,对于垃圾邮件活动依据是一起执行滥发邮件所有主机。

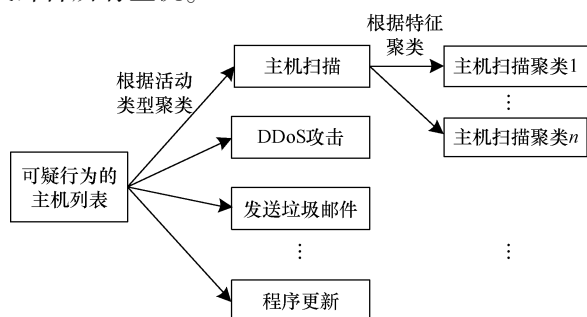


图2 BotMiner 的行为响应数据聚类分析结构

### 3.1.5 关联分析

所谓关联分析模块就是把从通信流量分析和行为响应的检测 2 个模块得到的数据进行分析,来判定一个主机是否是僵尸受控机,或者属于某个僵尸网络。该模块的主要目的是在这 2 个类群之间找出联系,从而为证明该主机属于某个僵尸网络提供更强检测依据。

定义通信流量检测结果的集合为  $C$ ,用  $C_i$  表示一个消息通信记录,行为响应检测结果的集合为  $A$ ,用  $A_i$  表示一个行为响应通信记录。因此,要判断一个主机是否属于某个僵尸网络就得对集合中的一个主机 host 可以计算出一个值,用以判断的权值<sup>[5]</sup>,这

个值要考虑的关系有:行为响应类之间的关系,行为响应类与通信流量类的关系, $\varphi(A_m)$ 是指  $A_m$  的行为响应的类型的危害系数,式(2)表示一个主机同时执行 2 个恶意行为的权值,其中, $|A_m \cap A_n|$ 是 2 种恶意行为在记录库中的交集的目, $|A_m \cup A_n|$ 是 2 种恶意行为在记录库中的并集的目,将 2 种恶意行为的交集的目与并集的目的倍数乘以其危害系数,就得一个主机同时执行 2 个恶意行为的权值;在式(3)中, $|A_i \cup C_j|$ 是指恶意行为  $A_i$  在通信流量记录的  $C_j$  中的并集的目, $|A_i \cap C_j|$ 是指恶意行为  $A_i$  在通信流量记录的  $C_j$  中的交集的目,它们再乘以  $A_i$  的危害系数就能得出行为响应  $A_i$  在通信流量集  $C_j$  中的分布情况。而计算权值的总分数如式(2)~式(4)如示, $S(h)$ 中的  $h$  是出现在行为响应集  $A_m$  中的 IP 地址的主机所计算出的权值; $A_m$  是主机  $h$  在行为响应中的流分类记录; $C_m$  是主机  $h$  在通信流量中的流分类记录;为某种恶意行为的活动级别,重量级活动的值取 2,轻量级活动的值取 1,当然活动级别的取值也将直接影响最后的可疑权值。

$$S(h_A) = \sum_{\substack{m,n \\ m>n}} \varphi(A_m) \varphi(A_n) \frac{|A_m \cap A_n|}{|A_m \cup A_n|} \quad (2)$$

$$S(h_c) = \sum_{i,j} \varphi(A_i) \frac{|A_i \cap C_j|}{|A_i \cup A_j|} \quad (3)$$

$$S(h) = \sum_{\substack{m,n \\ m>n}} \varphi(A_m) \varphi(A_n) \frac{|A_m \cap A_n|}{|A_m \cup A_n|} + \sum_{i,j} \varphi(A_i) \frac{|A_i \cap C_j|}{|A_i \cup A_j|} \quad (4)$$

利用式(4)计算得出权值后,就需要设定一个阈值  $\lambda$  作为判断值,当  $S(h_i) > \lambda, S(h_j) > \lambda, h_i$  和  $h_j$  同属于一个通信流量流类,则认定这 2 台主机属于一个僵尸网络。

### 3.1.6 特征库管理模块

特征库管理模块主要包括两方面的工作:(1)特征库的更新工作,包括 2 类更新:第 1 类是将客户端检测出的结果的通信流量特征存入特征库中,第 2 类就是从服务器端更新其他客户端检测到的而在自己的特征库中没有的僵尸网络的特征数据;(2)在检测过程中为检测结果提供已有特征信息数据进行对比,以便快速得到检测结果。

### 3.2 服务器模块

服务器模块主要就是特征库管理模块,主要负责将客户端上传的特征数据与特征库中现有的特征数据进行比较,如果存在就不加入特征库,否则添加一条新的记录,同时向各个客户端发起要求更新的命令。客户端的特征管理模块在接收到更新命令后就在客户端空闲的时候进行更新,这样不仅能减轻服务器的压力,也能提高客户端检测的利用率。它

的主要作用是使各个客户检测端能够合作共享信息实现协同检测,提高检测效率。

## 4 部署与实施

### 4.1 部署

按照计算机网络的作用范围,检测平台的网络部署有 3 种方法:广域网部署、城域网部署和局域网部署。其中,广域网部署对整个网络的流量进行监控是对研究分析僵尸网络是最有用的,但这样的部署会有高昂的代价,还要各个网络运营商的相互协助。而且对后期的分析处理带来更多的工作,这对平台的可行性不太大。因此,在检测平台的网络部

署中选择了局域网部署连接服务器,进行联动部署和综合检测,从而达到实现广域网部署的检测效果。

针对基于网络流量的僵尸网络动态检测平台的特点,设计了一个基于局域网的检测平台网络部署结构,僵尸网络所产生的通信流量都要经过网络的三层架构设备,最后达到僵尸主机或者攻击目标主机,核心层的数据通信量太大,如果在核心层上部署检测客户端的话,所在的主机或者服务器的工作负荷将会相应的大,工作的效率将受到影响。因此本文采用分散部署、集中控制的思路<sup>[13-14]</sup>,对僵尸网络所产生的异常流量采取综合治理。具体部署结构如图 3 所示。

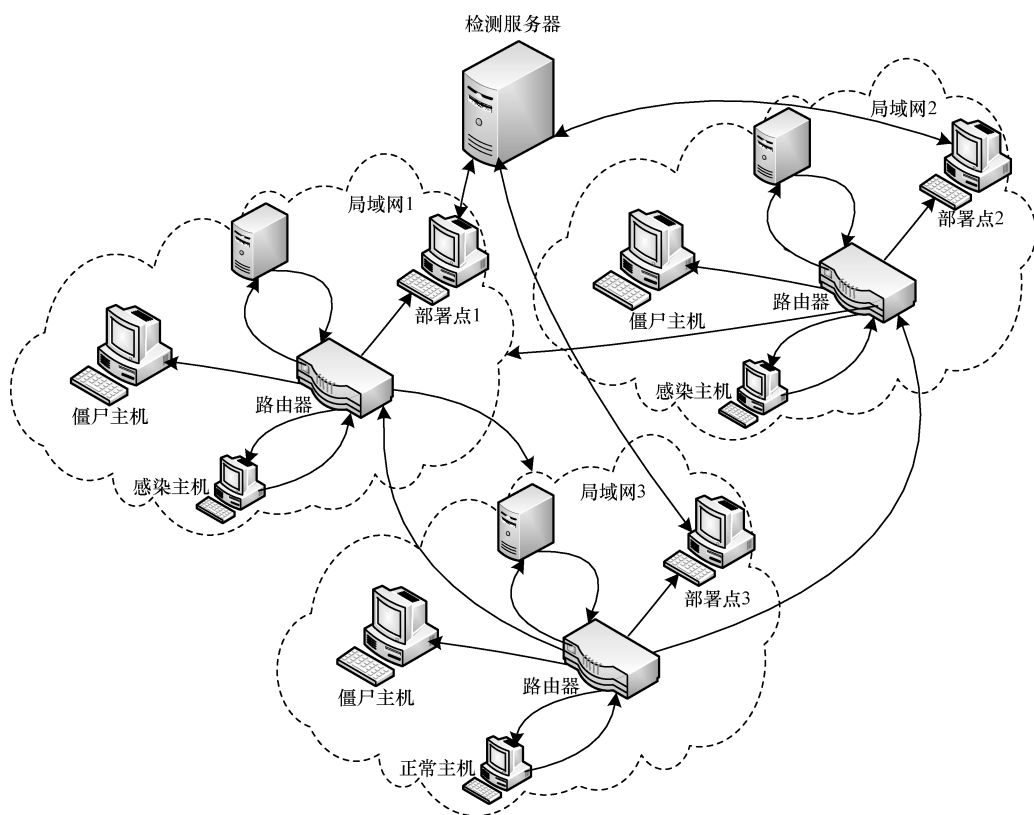


图 3 网络部署结构

### 4.2 实施

在僵尸网络平台的搭建过程中,本文采用风云僵尸网络,第一步在僵尸程序的传播传染过程,没有采用伪装、社会工程学等方式进行传播,而是直接把僵尸网络生成器生成的控制器端和 server 分别发送给主机,登录控制器端设置好监听端口开始监听,当

收到 server.exe 程序的主机执行后,就会被控制器监听到,受控主机通过不同端口连接到控制器,并等候命令。

受控主机的情况如图 4 所示,显示有受控主机的 IP 地址/端口、计算机名、所在地域、操作系统和状态等参数。

IP地址/端口	计算机名	所在地域	操作系统	内存	版本	状态
<input type="checkbox"/> 222.196.190.19:1950	SYSTEM6	四川省成都市 四川师范大学	Win XP SP3 (Build 2600)	512MB	V14.0...	空闲
<input type="checkbox"/> 222.196.190.66:1139	SYSTEM1	四川省成都市 四川师范大学	Win XP SP3 (Build 2600)	512MB	V14.0...	空闲
<input type="checkbox"/> 222.196.190.58:1318	SYSTEM2	四川省成都市 四川师范大学	Win XP SP3 (Build 2600)	256MB	V14.0...	空闲
<input type="checkbox"/> 222.196.190.27:54423	SYSTEM2	四川省成都市 四川师范大学	Win XP SP3 (Build 2600)	256MB	V14.0...	空闲

图 4 僵尸网络受控主机的情况

首先利用构建好的僵尸网络中的受控主机向 IP 地址 222.196.190.25 发起了攻击,然后在 Omnipeek

中设定好捕捉流量的条件,在僵尸网络发起攻击的同时也开始捕捉,得到僵尸主机发起攻击时的流量

数据,用于下文的实验数据分析。

5 实验数据分析

实验数据分析首先把攻击过程中捕获到的数据导入到原型系统中进行网络流量分析,部分数据如表 1 所示,其次是预处理,把不可能是僵尸网络的数据过滤掉,处理的结果如表 2 所示,再次是计算主机参数,如表 3 所示,分别计算出捕获流量的每台计算机的主机参数,然后利用这些主机参数计算出其特征参数,如表 4 所示,分别得到 IP 地址 222. 196.

190. 19 和 222. 196. 190. 58 向攻击目标 222. 196. 190. 25 特征参数,如果系统中已经有该僵尸网络的特征参数值,就可以直接得出检测结果,如图 5 所示。其中,实线表示 IP 地址为 222. 196. 190. 19 的特征参数,虚线是特征库中风云僵尸网络的特征参数,两者之间的距离小于阈值 1,就成功检测出 IP 地址为 222. 196. 190. 19 属于风云僵尸网络。阈值的取定是一个很复杂的问题,在这里所得到的特征参数进行归一化后,如果 2 组特征参数的距离小于 1,两者之间的相似度大。因此,本文将阈值设定为 1。

表 1 Ominipeek 捕获的部分数据

包	源地址	源端口	目的地址	目标端口	大小/Byte	IP 长度	相对时间/s	协议
1	222. 196. 185. 149	ms-dot...	IP Broadcast	corel...	358	340	0. 487 713	UDP
2	222. 196. 185. 149	ctx-br...	IP Broadcast	corel...	358	346	0. 493 167	UDP
3	00:0F:E2:80:02:4C	-	Ethernet Broadcast	-	64	-	1. 005 197	ARP Request
4	00:0F:E2:80:02:54	-	Mcast 802. 1d Br...	-	12 4	-	1. 005 531	802. 1
5	222. 196. 190. 27	62888	IP Broadcast	corel...	358	340	1. 260 736	UDP
6	222. 196. 190. 27	62889	IP Broadcast	corel...	365	347	1. 261 132	UDP
7	222. 196. 190. 53	63912	IP Broadcast	corel...	358	340	1. 576 796	UDP
8	222. 196. 190. 53	63913	IP Broadcast	corel...	363	345	1. 576 796	UDP
9	222. 196. 190. 48	icq-ip...	IP Broadcast	corel...	358	340	1. 674 110	UDP
10	222. 196. 190. 48	disrpn	IP Broadcast	corel...	368	350	1. 675 919	UDP
11	EC:A8:6B:D8:53:D1	-	Ethernet Broadcast	-	64	-	1. 677 062	ARP Request
12	222. 196. 190. 46	62293	IP Broadcast	corel...	358	340	1. 819 649	UDP

表 2 通过预处理的基础数据

编号	源 IP 地址	端口	目地 IP 地址	目的端口	大小/Byte	IP 长度	相对时间/s	协议
570497	222. 196. 190. 58	nbx-ser	222. 196. 190. 25	http	1 518	1500	0. 000 038	UDP
570499	222. 196. 190. 19	netrix-sftm	222. 196. 190. 25	http	1 518	1500	0. 000 552	UDP
570515	222. 196. 190. 19	-	222. 196. 190. 25	-	1 086	1068	0. 001 848	IP Fragment

表 3 主机参数

编号	源 IP 地址	目地 IP 地址	字节总数	包总长度/Byte	包个数	相对时间/s
1	222. 196. 190. 19	222. 196. 190. 25	182 886	180 492	134	3. 012
2	222. 196. 190. 58	222. 196. 190. 25	134 508	132 744	99	2. 770

表 4 不同主机的特征参数

编号	源 IP 地址	目地 IP 地址	nbps	npps	nbpp
1	222. 196. 190. 19	222. 196. 190. 25	115. 383	0. 295	391. 773
2	222. 196. 190. 58	222. 196. 190. 25	1 047. 911	1. 357	753. 685

如果特征库中没有相类似的特征,就按特征参数对所有的计算机进行归类,图 6 中就是由源 IP 地址为 222. 196. 190. 19 和源 IP 地址为 222. 196. 190. 58 的特征参数对比,图中的两线的距离小于阈值 1 就归为一类;面与源 IP 地址 222. 196. 190. 22 的距离大于 1 就不是一类。再把该数据与正在执行的攻击行为检测到的数据进行关联计算,得出源 IP 地址为 222. 196. 190. 19 关联计算值超过阈值。得出源 IP 地址为 222. 196. 190. 19 为僵尸网络,并且可判定与

222. 196. 190. 19 为一类的计算机都为该类僵尸网络。但此时不能判定出为何种僵尸网络。

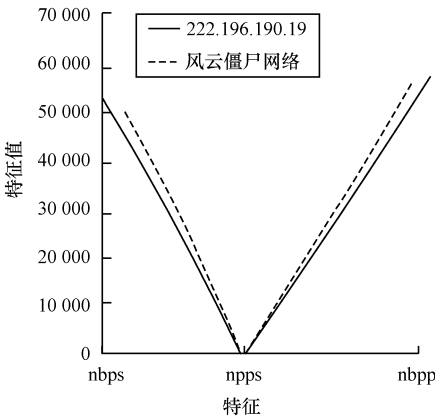


图 5 特征参数对比 1



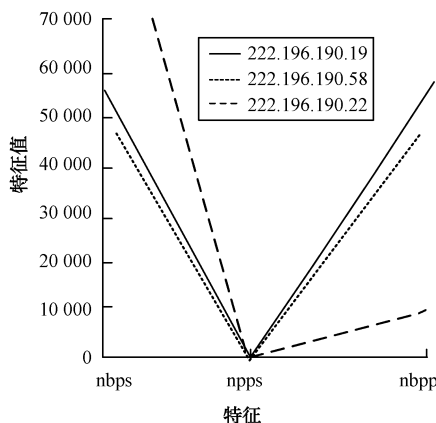


图 6 特征参数对比 2

对改进以后的僵尸网络动态检测平台与改进之前的检测系统对风云僵尸网络进行 4 次检测。在检测效率方面,每次花费的时间如表 5 所示,从表中可以看出两者第一次检测的时间是相同的,但第 2 次以后的检测每次所花时间,改进之后比改进之前的系统少用了 90%,提高了检测效率;在检测准确率方面,其中检测误检率为 3.12%,漏检率为 6.25%,在准确率上还有待进一步的提高。

表 5 改进前后检测系统的检测时间对比 s

检测系统	第 1 次	第 2 次	第 3 次	第 4 次
改进前的检测系统	60	60	60	60
改进后的检测系统	60	20	20	20

经过实验的测试,得到了以下结果:

(1)检测平台正常运行,能够访问数据库,按照检测流程工作,正常显示界面,但在处理十万条以上的数据记录时会在某些处理的地方出现一些异常现象,这将是下一步的改进工作之一。

(2)能对真实的网络攻击的通信流量进行提取特征,并进行归类。

(3)在特征库没有特征数据的情况下,对于风云僵尸能够提取出检测模型,并加入到客户端的特征库中。

(4)第 2 次检测同一个僵尸网络时,能提高客户端和服务端特征管理模块的工作效率和准确率。

## 6 结束语

本文对僵尸网络工作过程中产生的网络流量进行分析,在研究现有检测机制的存在问题的基础上,提出了一种基于网络流量的僵尸网络动态检测平台,实现了三动态性来完善现在检测机制,并为其设计了检测平台的网络部署结构;同时搭建了僵尸网络检测环境,模拟僵尸网络活动过程,对其进行网络流量监测,提取相关信息,用以检测平台的测试,并

对测试结果数据进行了分析。

下一阶段还需要注重以下方面的工作:(1)完善行为响应模块中行为类型的检测功能;(2)完善原型系统功能以达到检测平台的真正目的;(3)在真实的网络环境中进行测试;(4)改进平台,提高检测结果的准确性。

## 参考文献

- [1] Bacher P, Holz T, Kotter M, et al. Know Your Enemy: Tracking Botnets[J/OL]. [2013-09-15]. <http://www.honeynet.org/papers/bots>.
- [2] Goebel J. Rishi. Identify Bot Contaminated Hosts by IRC Nickname Evaluation[C]//Proceedings of HotBots'07. Berkeley, USA: USENIX, 2007: 47-56.
- [3] Gu Guofei, Porras P, Yegneswaran V, et al. Bothunter: Detecting Malware Infection Through IDS-driven Dialog Correlation[C]//Proceedings of Security'07. [S. l.]: USENIX, 2007: 167-182.
- [4] Gu Guofei, Zhang Junjie, Lee W. BotSniffer: Detecting Botnet Command and Control Channels in Network Traffic[C]//Proceedings of NDSS'08. San Diego, USA: [s. n.], 2008: 234-251.
- [5] Gu Guofei, Perdisci R, Zhang Junjie, et al. BotMiner: Clustering Analysis of Network Traffic for Protocol- and Structure Independent Botnet Detection[C]//Proceedings of USENIX Security'08. [S. l.]: USENIX, 2008: 139-154.
- [6] 刘建波. 基于流量分析的 P2P 僵尸网络检测[J]. 计算机与数字工程, 2011, 39(3): 90-91.
- [7] 刘 丹, 李毅超, 胡 跃. 多阶段过滤的 P2P 僵尸网络检测方法[J]. 计算机应用, 2010, 30(12): 3355-3356.
- [8] 王海龙, 胡 宁. Bot\_CODA: 僵尸网络协同检测体系结构[J]. 通信学报, 2009, 30(10A): 15-22.
- [9] 李润恒, 王明华. 基于通信特征提取和 IP 聚集的僵尸网络相似性度量模型[J]. 计算机学报, 2010, 33(1): 45-54.
- [10] Collins M, Shimeall T, Faber S, et al. Using Uncleanliness to Predict Future Botnet Addresses[C]//Proceedings of ACM/USENIX Internet Measurement Conference. [S. l.]: ACM Press, 2007: 91-102.
- [11] Ramachandran A, Feamster N. Understanding the Network-level Behavior of Spammers[C]//Proceedings of ACM SIGCOMM'06. [S. l.]: ACM Press, 2006: 291-302.
- [12] Zhuge J, Holz T, Han Xinhui, et al. Characterizing the IRC-based Botnet Phenomenon[R]. Peking University & University of Mannheim, Technical Report: TR-2007-010, 2007.
- [13] 冯宗彬, 时 剑, 黄国庆, 等. 一种新的 P2P 僵尸网络综合防御系统框架[J]. 军事通信技术, 2010, 31(1): 66-71.
- [14] 谢文彪. 基于 Agell 协作机制的分布式入侵检测组织结构研究[D]. 长沙: 中南大学, 2006.

编辑 金胡考