基于终端行为特征的 IRC 僵尸网络检测

王 威[1),2)] 方滨兴[1),2)] 崔 翔[2)]

[1)](哈尔滨工业大学计算机网络与信息安全研究中心 哈尔滨 150001)

[2)](中国科学院计算技术研究所信息安全研究中心 北京 100190)

摘 要 目前已有的 IRC 僵尸网络检测算法存在两个问题:需要先验知识以获取匹配模式,无法满足实时处理需求.为解决这两个问题,文中提出了基于昵称和命令序列这两个终端行为特征的 IRC 僵尸网络检测算法.文中提出三种属性分别从内容、组成和结构三方面互补的刻画两个昵称的相似性,给出两个昵称相似性的量化因子,根据这量化因子生成弹性 TRW 算法以进行 IRC 僵尸网络实时检测.文中还在分析僵尸终端登录服务器的行为的基础上,提出了基于命令序列相似性的检测算法.算法评估实验证明两个算法行之有效.最后将这两个算法用于大规模网络环境中实时检测 IRC 僵尸网络,在两周内检测到 162 个僵尸频道.

关键词 僵尸网络;IRC 昵称;命令序列;相似性度量

中图法分类号 TP393 **DOI 号**: 10.3724/SP.J.1016.2009.01980

IRC Botnet Detection Based on Host Behavior

WANG Wei[1),2)] FANG Bin-Xing[1),2)] CUI Xiang[2)]

[1)](*Research Center of Computer Network and Information Security, Harbin Institute of Technology, Harbin* 150001)

[2)](*Research Center of Information Security, Institute of Computing Technology, Chinese Academy of Sciences, Beijing* 100190)

Abstract There are two problems in current algorithms for IRC botnets detection. One is that detection algorithms require some prior knowledge of botnets to generate matching patterns. The other is that algorithms can not perform detection online. To solve these problems, this paper proposes two IRC botnet detection algorithms based on host behavior. Three attributes, LCS_rate, compositive distance and RN_dice coefficient, are discussed to quantify the similarity of nicknames from three aspects: content, composition and structure. To detect IRC botnets online, extended TRW algorithm based on the similarity of nicknames is proposed. This paper also proposes a detection algorithm based on the command sequence of IRC clients. Evaluations of these algorithms indicate that the two algorithms are correct and valid. At last, detection algorithms are used in large-scale network to detect IRC botnets and detect 162 bot channels within two weeks.

Keywords botnet; IRC nickname; command sequence; similarity measurement

1 引 言

近年来,僵尸网络的活跃得到了国内外安全业界的高度重视.简单地说,僵尸网络就是攻击者利用互联网秘密建立的可以集中控制的计算机群[1].僵尸网络不是一个特指的安全事件,它是攻击者手中的一个平台.利用该平台,攻击者可以实现覆盖面更

收稿日期:2009-07-15;最终修改稿收到日期:2009-08-23.本课题得到国家"九七三"重点基础研究发展规划项目基金(2007CB311100)、国家"八六三"高技术研究发展计划项目基金(2009AA01Z437,2007AA010501,2007AA01Z474)资助.**王 威**,女,1981 年生,博士研究生,主要研究方向为网络信息安全、僵尸网络.E-mail: wangwei@software.ict.ac.cn. **方滨兴**,男,1960 年生,教授,博士生导师,中国工程院院士,主要研究领域为计算机体系结构、信息安全和计算机网络等. **崔 翔**,男,1978 年生,博士研究生,助理研究员,主要研究方向为网络安全.

广、力度更高、更难于防范的攻击,最主要的就是分布式拒绝服务攻击 DDoS、发送垃圾邮件 Spam、网络仿冒 Phishing 等等.

僵尸网络具备 4 个属性:(1)恶意性. 僵尸网络控制者利用僵尸网络完成恶意攻击;(2)非配合性. 互联网计算机在其合法用户不合作状态下被植入僵尸程序成为僵尸终端;(3)可控性. 攻击者可以利用命令控制信道远程控制僵尸网络;(4)联合性. 僵尸网络中的僵尸终端联合同步执行攻击者的命令. 僵尸网络包括 4 个必要元素:(1)控制者(botmaster). 这是整个僵尸网络的所有者,也是攻击者;(2)僵尸终端(zombie). 互联网中被攻击者控制的计算机;(3)僵尸程序(bot). 控制者用于控制僵尸终端的恶意程序;(4)命令控制信道(command and control channel). 控制者通过命令控制信道向僵尸终端发布命令,实现远程控制. 显然命令控制信道是整个僵尸网络的核心,目前流行的僵尸网络采用多种协议来实现命令控制,主要有 IRC[2]、HTTP 和 P2P 协议,2008 年底爆发的 conficker① 更是借助于域名池.

目前检测僵尸网络主要有 3 大类方法:(1)基于蜜罐蜜网技术. 蜜罐蜜网技术是检测僵尸网络最基本的方法,德国的蜜网项目②和北京大学的狩猎女神项目[3]是比较突出的代表. (2)基于终端信息的检测技术. 僵尸终端要通过命令与控制信道和控制者通信,因此僵尸终端包含许多有意义的信息,采集这些信息对检测僵尸网络非常重要. 比较有特点的有美国哈佛大学 Malan 等人提出的基于对端的快速检测算法[4]和英国诺丁汉大学 Al-Hammadi 等人提出的基于日志相关性的检测算法[5]. (3)基于网络流的检测技术. 僵尸网络拥有大量的僵尸终端,控制者与僵尸终端之间的通信与正常用户之间的通信具有较大差异,通过监控网络流特征有可能寻找到命令与控制信道. 该类方法有美国波特兰州立大学 Binkley 等人提出的针对 IRC 僵尸网络的异常检测方法[6]、美国 BBN Technologies 的 Strayer 等提出的基于机器学习的 IRC 僵尸网络检测方法[7]、德国 RWTH 亚琛大学 Goebel 等提出的根据 IRC 用户昵称检测僵尸网络的方法 Rishi[8]和美国 AT&T 实验室 Karasaridis 等人提出的大范围检测僵尸网络的方法[9]等等,尤其以乔治亚理工的 Gu 等人完成的 BotHunter[10]、BotSniffer[11]和 BotMiner[12]最具代表性.

虽然 P2P 和 HTTP 僵尸网络逐渐盛行,但由于 IRC 僵尸网络简单、灵活、易控等优点,IRC 僵尸网络仍然是攻击者手中的重要手段. 目前已有的 IRC 僵尸网络检测算法或者需要先验知识,或者不能达到轻量实时处理,都不能满足大规模网络的需要. 本文提出了一种基于终端行为特征的 IRC 僵尸网络检测方法,该方法基于网络流检测,提取 IRC 终端昵称和活动的命令序列,近而分析一个频道是否为僵尸频道,该方法不需要先验知识,能够检测未知僵尸网络,同时又可以满足大规模网络的在线处理的需求.

2　相关工作

Racine 等人在文献[13]中提出了一种基于 IRC 网络流的检测方法,关注僵尸程序的特殊动作行为,例如:僵尸程序登陆 IRC 频道后处于“发呆”状态,仅通过 PING-PONG 命令与服务器保持连接. 基于统计 IRC 数据包包长和 IRC 消息分布特性,Chen 提出了一种可行策略来检测 IRC 僵尸网络[14]. 与这两个方法相似,本文提出的检测算法也关注行为,同时部署于边界网关和路由器.

Binkley 等人在文献[6,15]中提出基于异常检测的方法去发现僵尸网络. 该方法基于一个观察:如果一个 IRC 频道内的大部分终端执行大量的 TCP SYN 扫描,那么这个频道为僵尸频道. 本文提出的算法与此类似,都是关注 IRC 客户端行为特性. Binkley 关注客户端的恶意行为,本文关注客户端的相似特征. 在僵尸网络建立初期,客户端可能没有恶意行为,所以可看出本文方法可以较早地发现僵尸网络,从而更优.

Goebel 和 Holz 刻画了一种基于用户昵称的 IRC 僵尸网络检测方法[8]. Rishi 使用正则表达式来描述僵尸昵称模式,使用评分函数来分析昵称. 本文提出的方法也关注用户昵称,但两种方法是截然不同的. Rishi 用已存在僵尸昵称模式来对新捕获的昵称进行评分,从而判断新昵称是否为僵尸昵称. 本文的算法关注同一个频道下的昵称相似度,从而不需要先验知识,并且能检测未知的僵尸网络.

本文提出的算法运行于一个高性能网络捕包分析平台[16-17],这是作者所在研究中心的已有科研成

① Leder F, Werner T. Know your enemy: Containing conficker. Form the Honeynet Project. http://honeynet.org. April 2009

② Laboratory for Dependable Distributed Systems of the University of Mannheim. German Honeynet Project. http://pil.informatik.uni-mannheim.de/index.php?pagecontent=site/Research.menu/Honeynet.page

果.该平台采用了零拷贝和多线程并行协议栈等技术,有效地解决网络数据包处理的瓶颈问题,为能实时检测 IRC 僵尸网络奠定了基础.在作者的早期研究中[18]提出了一种基于用户昵称长度相似性的 IRC 僵尸网络检测方法,该算法是本文算法的雏形.该算法利用昵称各个组成部分的长度作为衡量昵称相似性的度量,这虽然有效但抗攻击性不强.本文提出的算法采用更丰富的属性刻画昵称相似性,并提出基于命令序列相似性的辅助检测算法.

3 基于昵称相似性的检测算法

IRC 客户端登陆服务器成功后发送的第一条命令是:NICK nickname,参数 nickname(昵称)是客户端的唯一标识.IRC 协议要求在同一 IRC 网络中,任意两个客户端的昵称是相异的.僵尸终端的昵称由僵尸程序自动生成,所以僵尸程序必须采取一定随机策略避免昵称的重复.现在的僵尸程序常常采用固定串+特殊字符+随机串的方式,例如 Rbot.210944 采用的昵称构成方式为:国家名称缩写|九位数字,如 USA|016887436、DE|028509327 等.还有一些僵尸程序采用相反的策略,构成方式为变长英文前缀加固定数字串,例如 Korgo.F.var 用"_13"作为固定后缀,如 bmdut_13、tlmtj_13 等.

僵尸昵称的生成规则初始硬编码于僵尸程序中,所以同一僵尸频道内的僵尸昵称必然具有相似性,相似性包括内容相似性、组成相似性和结构相似性.干净昵称反应了终端用户的个人习惯与喜好,所以正常频道下的昵称具有高随机性.本节充分考虑僵尸频道下昵称的相似性,来判断一个频道是否为僵尸频道.

3.1 相似性度量属性

为刻画两个昵称的相似性,我们定义 3 个属性:公共子串比率、组成距离和 *RN_Dice* 系数.

3.1.1 公共子串比率

定义 1. 字符串 X 和字符串 Y 的公共子串比率 *LCS_rate* 定义为

$$LCS\_rate(X,Y)=\frac{2\times|LCS\ \text{of}\ X\ \text{and}\ Y|}{|X|+|Y|} \quad (1)$$

考虑很多僵尸程序的昵称组成方式为固定字符串混合变长随机串,固定字符串为僵尸网络的控制者提供了成员检查机制,对非法主机直接踢出频道或拒绝服务攻击.公共子串比率反映了两个昵称是否包含"长"公共子串,显然如果两个昵称具有高公共子串比率,则说明两个昵称具备高内容相似性,那么这两个昵称为僵尸昵称的概率较高.

图 1 对比了 50 对正常昵称和 50 对来自不同僵尸频道的僵尸昵称的公共子串比率,可以看出,僵尸昵称的 *LCS_rate* 普遍高于 0.4,而正常昵称的 *LCS_rate* 普遍低于 0.2,从而说明公共子串这个属性是行之有效的且该属性强度较高.需要说明的是第 37～40 对这 4 组僵尸昵称,从图中可以看出,他们的 *LCS_rate* 较低,这是因为这些昵称的构成方式是长度为 8 的随机字母串.虽然公共子串比率这个属性无法有效识别该模式,但通过后面的介绍可以看出,我们可以通过另外两个互补属性对这种模式进行有效的判定.

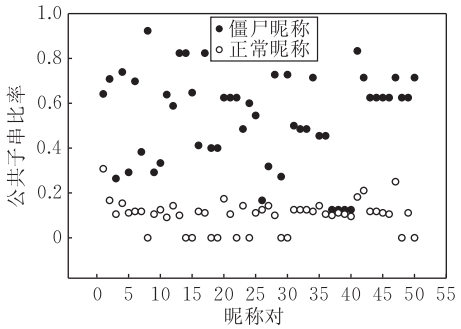

图 1 昵称公共字串比率属性比较

3.1.2 组成距离

昵称公共子串比率反映了昵称的内容相似性,为描述昵称的组成相似性,定义昵称组成距离.一个昵称由若干字母、数字和特殊字符组成,对已有的僵尸昵称进行分析发现,一个频道下的昵称的各个组成部分的内容可能不同,但各个组成部分的长度基本相同.于是将昵称表示为四元向量(昵称长度、字母长度、数字长度、特殊字符长度).在使用欧几里德距离来刻画两个昵称向量相似性时,我们发现两对昵称[D|00]|23403,[PD|00]|26743 和 ross,rosey 具有相同的欧式距离,但显然第一对昵称为僵尸昵称的可能性更高,于是需要突出特殊字符长度和数字串长度对昵称组成距离的贡献.

定义 2. 字符串 X 的总长度记为 $LT(X)$,字母长度记为 $LL(X)$,数字长度记为 $LN(X)$,特殊字符长度记为 $LS(X)$,则定义字符串 X 和字符串 Y 的组成距离为

$$Dis(X,Y)=\sqrt{\frac{(LT(X)-LT(Y))^2+(LL(X)-LL(Y))^2+(LN(X)-LN(Y))^2+(LS(X)-LS(Y))^2}{\max\{\min\{LN(X),LN(Y)\},1\}\times\max\{\min\{LS(X),LS(Y)\},1\}}} \quad (2)$$

为验证组成距离属性的有效性,分别考察僵尸昵称和正常昵称各 50 对(测试数据与 3.1.1 节相同),结果如图 2 所示.可以看出,75%的僵尸昵称对组成距离为 0,僵尸昵称对的组成距离普遍低于 2,而正常昵称的组成距离普遍高于 2.于是可以选择该属性作为僵尸昵称相似性度量标准之一,但强度稍弱于公共子串比率.

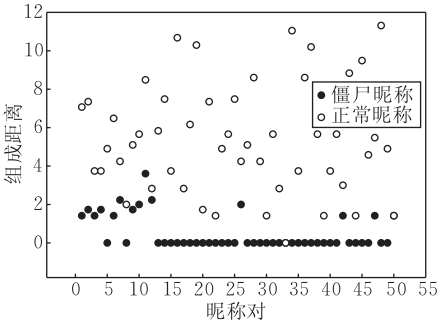

图 2　昵称组成距离属性比较

3.1.3　*RN_Dice* 系数

除了内容相似性和组成相似性两个指标外,僵尸昵称还具备高结构相似性.为忽略内容,首先对昵称字符串进行标准化映射 f:

$$f(x)=\begin{cases}A, & x\in\{a\sim z,\ A\sim Z\}\\ B, & x\in\{0\sim 9\}\\ C, & \text{其它}\end{cases}.$$

称映射后的字符串为原字符串的标准字符串.例如字符串 USA|00|XP|719 的标准字符串为 AAACBBCAACBBB.为忽略组成长度,最后对标准字符串进行缩减:去掉连续的相同字母.例如 AAACBBCAACBBB 的缩减标准字符串为 ACBCACB.可以看出,缩减标准字符串反映了原字符串的结构.

定义 3.　字符串 X 的缩减标准字符串计为 $RN(X)$,则字符串 X 和字符串 Y 的 *RN_Dice* 系数定义为

$$RN\_Dice(X,Y)=\frac{2\times|2grams(RN(X))\bigcap 2grams(RN(Y))|}{|2grams(RN(X))|+|2grams(RN(Y))|}\quad(3)$$

同样的,我们考察各 50 对正常昵称和僵尸昵称的 *RN_Dice* 系数(测试数据与 3.1.1 节相同),如图 3 所示.可以看出,僵尸昵称的 *RN_Dice* 系数大多为 1,这是因为实验中的僵尸昵称普遍具备相同的结构.但一些正常昵称的 *RN_Dice* 系数同样为 1,这是因为我们采集到的正常昵称多数是纯字母或字母数字组合,很容易使两个昵称具备相同的结构(尤其是纯字母串).可以看出,*RN_Dice* 系数仅仅是衡量僵尸昵称的充分条件,属性强度最弱.

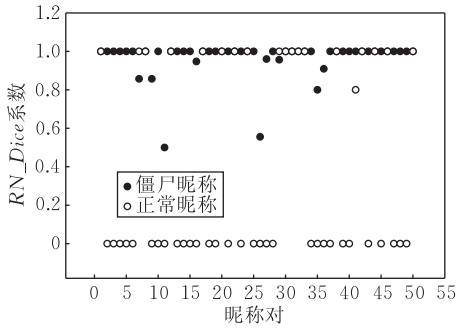

图 3　昵称 *RN_Dice* 系数比较

3.2　IRC 僵尸频道检测算法

上节讨论的 3 个属性从内容、组成和结构三方面互补刻画了昵称的相似性,为了实时检测僵尸频道,我们借鉴 TRW(Threshold Random Walk)算法[19].

3.2.1　TRW 算法

待观察频道为 C,令 Y_i 标记当前加入 C 的第 i 对昵称的相似性:

$$Y_i=\begin{cases}0, & \text{两个昵称具有低相似性}\\ 1, & \text{两个昵称具有高相似性}\end{cases}.$$

当获得观察 $Y_1,Y_2,\cdots$ 时,利用 TRW 算法可以确定频道 C 是否为僵尸频道.TRW 算法是一种假设检验方法,支持两个假设 H_0 和 H_1,H_0 假定频道 C 是一个正常频道,H_1 假定频道 C 是一个僵尸频道.定义

$$Pr[Y_i=0|H_0=0]=\theta_0,$$
$$Pr[Y_i=1|H_0=0]=1-\theta_0,$$
$$Pr[Y_i=0|H_0=1]=\theta_1,$$
$$Pr[Y_i=1|H_0=1]=1-\theta_1,$$

其中 $\theta_0>\theta_1$,表示如果频道是正常频道,那么频道内的昵称具备低相似性的概率较高.给定两个假设之后,输出结果有 4 种可能:接受假设 H_0,但该频道是僵尸频道,这是漏报;接受假设 H_1,但该频道是正常频道,这是误报;接受假设 H_0,该频道是正常频道,正解;接受假设 H_1,该频道是僵尸频道,正解.为保证 TRW 算法的检测性能,用误报率 P_F 和检测率 P_D 来进行限制.用户指定参数 α 和 β,算法的输出保证

$$P_F\leq\alpha,\quad P_D\geq\beta.$$

随着新昵称对加入频道 C,似然比计算如下:

$$\Lambda(Y)=\frac{Pr[Y|H_1]}{Pr[Y|H_0]}=\prod_i\frac{Pr[Y_i|H_1]}{Pr[Y_i|H_0]},$$

$$\Lambda(Y^i)=\begin{cases}\Lambda(Y^{i-1})\times\dfrac{1-\theta_1}{1-\theta_0}, & Y_i=1\\ \Lambda(Y^{i-1})\times\dfrac{\theta_1}{\theta_0}, & Y_i=0\end{cases},$$

$$\Lambda(Y^0)=1,\ i=1,2,\cdots,$$

$$\text{算法输出}=\begin{cases}\text{接受 } H_1, & \Lambda(Y)\geq\eta_1\\ \text{接受 } H_0, & \Lambda(Y)\leq\eta_0\\ \text{继续}, & \text{其它}\end{cases}.$$

文献[20]指出，当阈值满足 $\eta_1=\dfrac{\beta}{\alpha}$，$\eta_0=\dfrac{1-\beta}{1-\alpha}$ 时，即可保证 $P_F\leq\alpha$，$P_D\geq\beta$.

3.2.2 弹性 TRW 算法

使用 TRW 算法的初衷是为了尽可能快速地确定一个频道是否为僵尸频道，根据文献[19]，为确定一个频道是否为僵尸频道，需要分析该频道下 N 对昵称，有

$$E[N|H_1]=\frac{\beta\ln\dfrac{\beta}{\alpha}+(1-\beta)\ln\dfrac{1-\beta}{1-\alpha}}{\theta_1\ln\dfrac{\theta_1}{\theta_0}+(1-\theta_1)\ln\dfrac{1-\theta_1}{1-\theta_0}},$$

$$E[N|H_0]=\frac{\alpha\ln\dfrac{\beta}{\alpha}+(1-\alpha)\ln\dfrac{1-\beta}{1-\alpha}}{\theta_0\ln\dfrac{\theta_1}{\theta_0}+(1-\theta_0)\ln\dfrac{1-\theta_1}{1-\theta_0}}.$$

各个参数对期望的影响如图 4 所示，可以看出，在确定误报率和检测率的情况下，θ_0 越大，θ_1 越小，越能通过较少轮次的计算来确定一个频道是否为僵尸频道. θ_0 和 θ_1 反映了相似昵称对确定僵尸频道的影响程度.

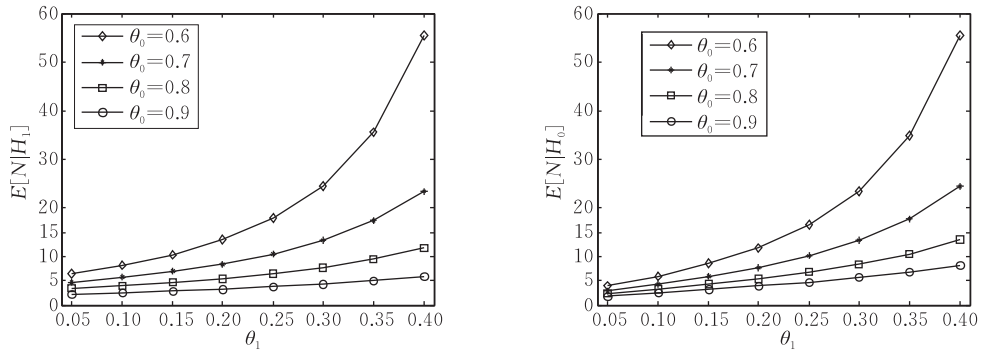

图 4 $\alpha=0.01,\beta=0.99$ 时，$E[N|H_1]$ 及 $E[N|H_0]$ 随 θ_1 和 θ_0 的变化曲线

根据 TRW 算法，当 $Y_i=1$ 时，算法上行 $\Lambda(Y)$ 累乘 $(1-\theta_1)/(1-\theta_0)$；当 $Y_i=0$ 时，算法下行 $\Lambda(Y)$ 累乘 θ_1/θ_0. Y_i 作为传统 TRW 算法的输入，表示一对昵称是否相似. 为了量化从而突出昵称相似性对检测算法的影响，引入弹性因子 $\lambda(0<\lambda\leq 2)$，λ 量性反映昵称相似或相异的程度. 从而生成弹性 TRW 算法，引入弹性因子后，可以预先设置基准 θ_0、θ_1，算法执行时相当于动态调整每一步 walk 的 θ_0 和 θ_1，从而正确且有效地根据昵称相似程度减少或增加判定轮次. 弹性 TRW 的似然比计算如下：

$$\Lambda(Y^i)=\begin{cases}\Lambda(Y^{i-1})\times\dfrac{1-\theta_1/\lambda_i}{1-\theta_0}, & Y_i=1\\ \Lambda(Y^{i-1})\times\dfrac{\theta_1}{\theta_0/\lambda_i}, & Y_i=0\end{cases},$$

其中 $0<\lambda_i\leq 2$，$\Lambda(Y^0)=1$，$i=1,2,\cdots$.

为量性刻画昵称相似或相异程度，首先我们要确定 3 个属性的阈值，由 3.1 节可以确定，公共子串比率 *LCS_rate* 阈值为 0.4，组成距离 *Dis* 阈值为 2，*RN_Dice* 系数阈值为 0.8. 并根据 3 个属性的强度分别分配属性所占比率为 0.6，0.3 和 0.1，于是弹性因子 λ 计算如下：

$$\lambda^1=\begin{cases}1+(LCS\_rate-0.4)\times\dfrac{1}{0.6}, & LCS\_rate\geq 0.4\\ 1-(0.4-LCS\_rate)\times\dfrac{1}{0.4}, & LCS\_rate<0.4\end{cases},$$

$$\lambda^2=\begin{cases}1+(2-Dis)\times\dfrac{1}{2}, & Dis\leq 2\\ 1-(Dis-2)\times\dfrac{1}{6}, & 2<Dis\leq 8\\ 0, & Dis>8\end{cases},$$

$$\lambda^3=\begin{cases}1+(RN\_Dice-0.8)\times\dfrac{1}{0.2}, & RN\_Dice\geq 0.8\\ 1-(0.8-RN\_Dice)\times\dfrac{1}{0.8}, & RN\_Dice<0.8\end{cases},$$

$$\lambda=\lambda^1\times 0.6+\lambda^2\times 0.3+\lambda^3\times 0.1$$

若 $\lambda=0$ 则 $\lambda=0.02$.

通过 λ 的计算方法可以看出，λ 反映了昵称对的各个属性对昵称相似性的贡献. 图 5 为 50 对正常昵称和 50 对僵尸昵称的 λ 值比较（实验数据与 3.1.1 节相同），简单地说，$\lambda>1$ 则可以说昵称对的属性值对确定昵称对相似做出了正贡献，相应地，$\lambda<1$ 则可以说昵称对的属性值对确定昵称对相异做出了正贡献. λ 定量的给出昵称相似度偏离阈值的

程度,即昵称相似或相异度.需要说明的是图中第 26 号僵尸昵称对,这两个昵称碰巧分属于不同僵尸频道,可以说,这两个昵称是相异的.

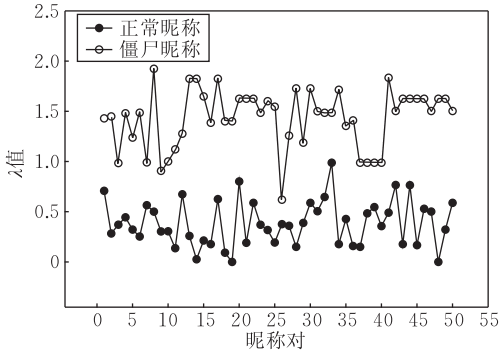

图 5　昵称弹性因子 λ 比较

3.2.3　检测算法描述

本节给出基于昵称相似性的 IRC 僵尸网络检测算法,算法根据公共子串比率、组成距离和 *RN_Dice* 系数这 3 个属性计算两个昵称的相似性并得到因子 λ,在计算弹性因子时,3 个属性所占比例分别是 60%,30%和 10%.然后采用弹性 TRW 算法判断一个频道是否为僵尸频道.具体描述如下:

DETECTION(*flow*)

flow=(*Source_IP*, *Source_Port*, *Dest_IP*, *Dest_Port*, *channel*, *nickname*)

channel is extracted from the command 'JOIN'

nickname is extracted from the command 'NICK'

the function *compute_lcs* computes the *LCS_rate* of the two parameters use formular (1)

the function *compute_dis* computes the compositive distance of the two parameters use formular (2)

the function *compute_dice* computes the *RN_dice* coefficient of the two parameters use formular (3)

```
For each flow
  If channel is new and not in Blacklist and Whitelist
    Then generate a new Channel
         Channel.name=channel
         Channel.ratio=1
         Channel.first=nickname
  If channel is aready exist
    Then If Channel.first==′\0′
           Channel.first=nickname
         Else LCS_rate=compute_lcs(Channel.first, nickname)
              Dis=compute_dis(Channel.first, nickname)
              RN_Dice=compute_dice(Channel.first, nickname)
              Channel.first=′\0′
              If LCS_rate>0.4
                 Y=1
              Else If Dis<2 && RN_Dice≥0.8
                 Y=1
              Else Y=0
              λ=LCS_rate×0.6+Dis×0.3+RN_Dice×0.1
              If λ==0 Then λ=0.02
              If Y==1
                 Channel.ratio=Channel.ratio×(1−θ1/λ)/(1−θ0)
              Else Channel.ratio=Channel.ratio×θ1/(θ0/λ)
              If Channel.ratio≥η1
                 flag Channel as a botnet channel and move Channel to Blacklist
              If Channel.ratio≤η0
                 flag Channel as a clean channel and move Channel to Whitelist
```

3.3　算法评估

我们随机采集了国内一知名论坛上合法用户昵称 330 个,正常 IRC 频道昵称 10 组,共 400 个正常昵称,分为 15 组;布置于中国科学院计算技术研究所内的蜜罐捕获的僵尸频道 10 组,结合文献[8]中所列 26 组 Rishi 系统监控到的僵尸昵称,去冗余并选择有代表性的僵尸昵称 13 组,具体见表 1.分别对这 15 个正常频道和 13 个僵尸频道利用弹性 TRW 算法进行检测,检测正确率为 100%.检测轮次如图 6 所示.

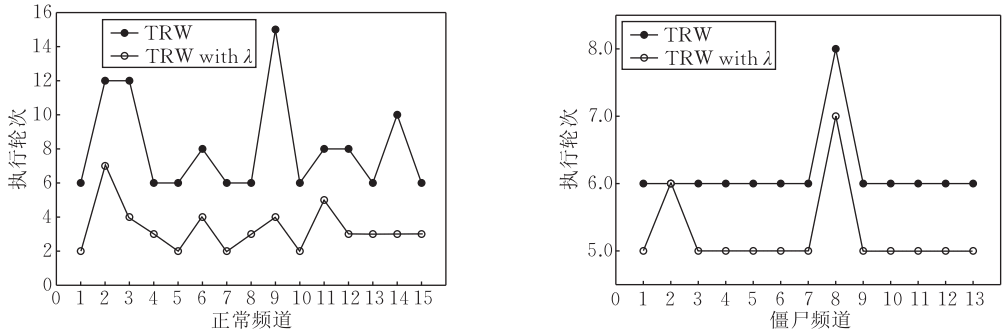

图 6　检测算法执行轮次比较($\theta_1=0.3$, $\theta_0=0.7$, $\alpha=0.01$, $\beta=0.99$)

表 1 测试用僵尸昵称列表

序号	示例	结构
1	[RzC]-[Amilcar]-[Silver]+, [RzC]-[Amilcar]-[click]+, [RzC]-[BaRaCa]-[DVDRip]+	CACACAC
2	ebztjwih, hoqrococ, nbithcrd	A
3	[FUCKOFF]-016649, [FUCKOFF]-192306	CACB
4	USA|00|XP|SP1|719, RUS|00|XP|SP1|317	ACBCACABCB
5	ago-3468, ago-76545	ACB
6	[sx]|840, [sx]|97	CACB
7	cmox31, cmox32	AB
8	'CHN'XPV701, 'CHN'2SIG2K	不同
9	[00|CHN|501054513] [00|CHN|585680095]	CBCACBC
10	CHN[2K|SP4]00[L]990854 CHN[XP|SP1]00[D]924053	不同
11	CHN|419235,CHN|330608	ACB
12	vjlr_13, bmdut_13	ACB
13	[D00|CHN|43744] [P00|CHN|25836]	CABCACBC

4 命令序列相似性检测算法

第 3 节介绍的基于昵称相似性的方法能很好地解决僵尸网络检测问题,本节给出一个 IRC 僵尸网络检测辅助算法.

IRC 僵尸程序运行后登录到 IRC 服务器等待接收命令.通常的登录过程为:发送昵称,加入频道,修改自身模式和频道模式,接收命令,发送 PING/PONG 命令维持连接.相同类别的僵尸程序登录到同一个僵尸频道中,发送的命令序列和相关参数除昵称和主机名外基本一致,正常的 IRC 用户在登录服务器后或修改自身模式,或发送聊天信息,或修改模式参数或者聊天信息内容不一致,表现出一定的随机性.根据对僵尸主机和正常主机登录服务器后发送命令序列的分析,僵尸频道的命令序列相似性较高,而正常频道的命令序列随机性较高.因此同一频道内终端命令序列相似性可以作为检测僵尸网络的方法.

在实际 IRC 数据包统计过程中发现,常用 IRC 命令有 13 个,按照使用频度排列为 PRIVMSG, PONG, PING, JOIN, QUIT, MODE, NOTICE, LIST,END,PART,WHO 和 LINK.其中前 3 位的使用频度分别为 25.31%,12.37%和 10.29%.在生成终端命令序列向量时,我们只关注这 13 个命令.检测步骤简述如下:

(1)根据 IRC 协议生成单词表,单词表含有 13 项(IRC 常用命令);

(2)根据该单词表,构造 AC 自动机;

(3)利用此 AC 自动机对命令文本进行分词,生成命令向量.该向量有 13 个属性,每个属性值为此命令在命令文本中出现位置序列之和;

(4)两个向量的余弦相似度定义为两个命令序列的相似度;

(5)频道内命令序列的相似度的均值定义为频道相似性,相似性高于指定阈值被识别为僵尸频道.

实验表明,僵尸频道的终端命令序列余弦相似度明显高于正常聊天频道,其中 50%的僵尸频道的相似性为 1,即命令序列基本相同,与预先分析的同一版本僵尸程序命令一致的结论相同.正常频道中的终端命令序列相似度分布在 0~0.85,在区间[0.3, 0.7]的频道比例约 85%.

5 大规模网络环境下实时检测

第 3 节和第 4 节分别讨论了两种基于终端行为特征的 IRC 僵尸网络检测算法,通过算法评估实验可以看出,这两种算法是行之有效的.本节把这两种算法以插件的形式加载到一个高性能网络捕包平台[16-17]上,从而运行于实际网络环境中以检测 IRC 僵尸网络.简单地说,该高性能网络捕包平台采用零拷贝接口实时捕获网络流量,多线程并行 TCP/IP 协议栈进行数据流还原,将还原后的数据交付给对应插件处理.

5.1 检测环境

IRC 僵尸网络检测系统对骨干网网络流量进行捕获分析,系统测试拓扑如图 7 所示.通过分光器将两个边界路由器的流量进行复制,复制的流量经过聚合分流传给检测系统分析机分析,分析日志保存致数据库.通过旁路方式分析流量从而不影响正常的网络通信.

共有 3 台检测机,分别处理不同端口的数据.每台处理机处理流量约为 1.8Gbps,系统运行丢包率为 0.这 3 台检测机的配置为 4CPU Intel(R) Xeon(R) 2.4GHz,4GB 内存,双 1000MB 以太网卡,单 100MB 控制网卡.

5.2 检测结果

检测系统稳定运行两周,每台检测机的 CPU 占用率一直持续在 45%左右,内存使用从 1.1GB 增长到 1.5GB,使用率不足 50%.两周内共检测到

IRC 聊天频道 405 个,聊天主机 5000 多台,其中僵尸频道 162 个,受害主机 4400 台.从数据比例看出目前使用 IRC 协议聊天的客户端中 80%为僵尸终端,具体情况如图 8 所示.

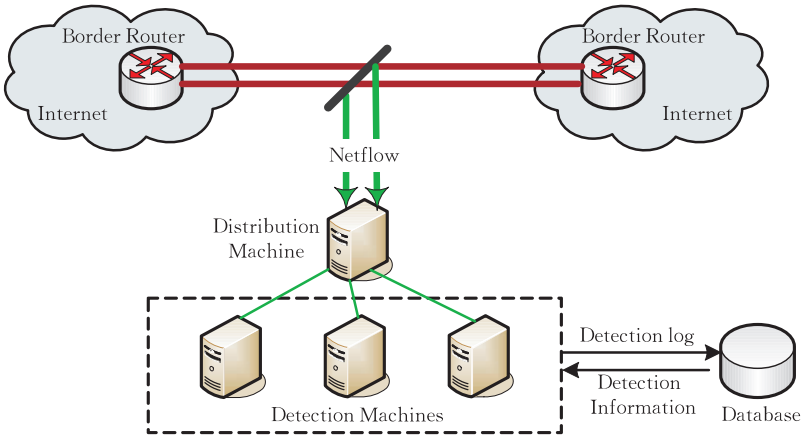

图 7　网络环境下检测拓扑图

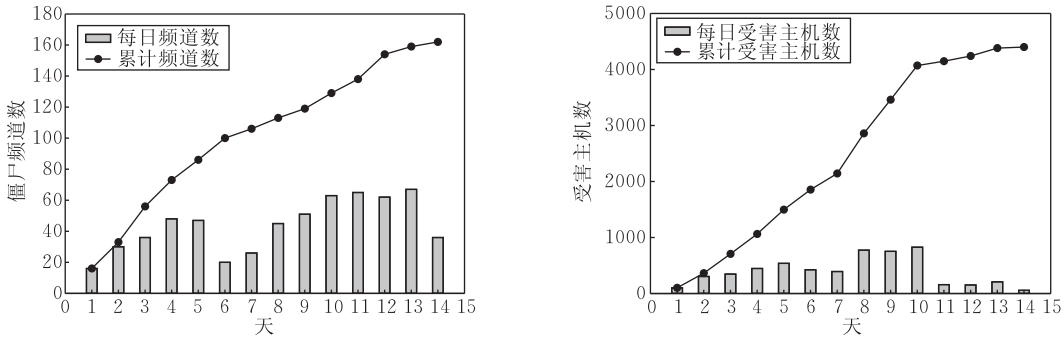

图 8　每日检测结果

6　结束语

虽然 P2P 和 HTTP 僵尸网络逐渐盛行,但 IRC 僵尸网络以其独有的特点,仍然是攻击者手中的一个重要武器.本文从分析 IRC 终端行为特征入手,针对目前已有的检测算法或者需要先验知识,或者无法满足实时检测需求的现状,提出了基于昵称相似性和命令序列相似性的 IRC 僵尸网络检测算法.两种检测算法基于一个事实:僵尸昵称的产生规则和终端登录 IRC 服务器后的动作命令均硬编码于僵尸程序中,于是同属于一个僵尸网络的僵尸终端行为必然具有一定的相似性.本文详细讨论了基于昵称相似性的 IRC 僵尸网络检测算法,提出 3 个属性互补的从内容、组成和结构三方面量化昵称的相似性,并提出弹性 TRW 算法以实时快速检测 IRC 僵尸网络.另外,本文简单地介绍了基于命令序列相似性的 IRC 僵尸网络检测算法,实验表明,该算法也是积极有效的.最后将两种算法以插件形式加载到高性能网络捕包平台中,实现大规模网络环境下 IRC 僵尸网络的实时检测.

参　考　文　献

[1] Du Yue-Jin, Cui Xiang. Botnets and its enlightment. China Data Communication, 2005, 7(5): 9-13(in Chinese)
(杜跃进,崔翔. 僵尸网络及其启发. 中国数据通信, 2005, 7(5): 9-13)

[2] Oikarinen J, Reed D. Internet relay chat protocol. Request for Comments (RFC) 1459, IETF, May, 1993

[3] Zhuge Jian-Wei, Han Xin-Hui, Zhou Yong-Lin et al. HoneyBow: An automated malware collection tool based on the high-interaction honeypot principle. Journal of Communications, 2007, 28(12): 8-13(in Chinese)
(诸葛建伟,韩心慧,周勇林等. HoneyBow:一个基于高交互式蜜罐技术的恶意代码自动捕获器. 通信学报, 2007, 28(12): 8-13)

[4] Malan D J. Rapid detection of botnets through collaborative networks of peers[Ph. D. dissertation]. Harvard University, Cambridge, Massachusetts, 2007

[5] Al-Hammadi Y, Aickelin U. Detecting botnets through log correlation//Proceedings of the IEEE/IST Workshop on Monitoring, Attack Detection and Mitigation. Tuebingen, Germany, 2006: 97-100

[6] Binkley J R, Singh S. An algorithm for anomaly-based bot-

net detection//Proceedings of the 2nd Workshop on Steps to Reducing Unwanted Traffic on the Internet. San Jose, CA, 2006: 43-48

[7] Strayer W T, Walsh R et al. Detecting botnets with tight command and control//Proceedings of the 31st IEEE Conference on Local Computer Networks. Tampa, FL, 2006: 195-202

[8] Goebel J et al. Rishi: Identify bot contaminated hosts by IRC nickname evaluation//Proceedings of the HotBots'07, First Workshop on Hot Topics in Understanding Botnets. Cambridge, MA, 2007

[9] Karasaridis A, Rexroad B et al. Wide-scale botnet detection and characterization//Proceedings of the HotBots'07, First Workshop on Hot Topics in Understanding Botnets. Cambridge, MA, 2007

[10] Gu G, Porras P, Yegneswaran V et al. BotHunter: Detecting malware infection through ids-driven dialog correlation//Proceedings of the 16th USENIX Security Symposium (Security'07). Boston, Massachusetts, 2007: 167-182

[11] Gu G, Zhang J, Lee W. BotSniffer: Detecting botnet command and control channels in network traffic//Proceedings of the 15th Annual Network and Distributed System Security Symposium (NDSS'08). San Diego, CA, 2008: 269-286

[12] Gu G, Perdisct R, Zhang J, Lee W. BotMiner: Clustering analysis of network traffic for protocol- and structure-independent botnet detection//Proceedings of the 17th USENIX Security Symposium (Security'08). San Jose, CA, 2008: 139-154

[13] Racine S. Analysis of internet relay chat usage by DDoS zombies[Ph. D. dissertation]. Swiss Federal Institute of Technology, Swiss, Zurich, 2004

[14] Chen Y. Irc-based botnet detection on high speed routers//Proceedings of the ARO-DARPA-DHS Special Workshop on Botnets. Arlington, VA, 2006

[15] Binkley J R. Anomaly-based botnet server detection//Proceedings of the FloCon 2006 Analysis Workshop. Vancouver, WA, 2006

[16] Zhang Zhao-Xin, Fang Bin-Xing, Hu Ming-Zeng. An IDS-supported architecture of information capture for high-speed network. Journal of Beijing University of Posts and Telecommunications, 2006, 29(2): 118-122(in Chinese)
(张兆心, 方滨兴, 胡铭曾. 支持 IDS 的高速网络信息获取体系结构. 北京邮电大学学报, 2006, 29(2): 118-122)

[17] Luo Hao, Yun Xiao-Chun, Fang Bin-Xing. The research of multi-thread parallel TCP/IP protocol assembly technology. High Technology Letters, 2003, 13(11): 15-19(in Chinese)
(罗浩, 云晓春, 方滨兴. 多线程 TCP/IP 还原技术的研究. 高技术通讯, 2003, 13(11): 15-19)

[18] Wang W, Fang B X et al. A novel approach to detect IRC-based botnets//Proceedings of the International Conference on Networks Security, Wireless Communications and Trusted Computing. Hubei, China, 2009: 408-411

[19] Jung J, Paxson et al. Fast portscan detection using sequential hypothesis Testing//Proceedings of the IEEE Symposium on Security and Privacy. Oakland, CA, 2004: 211-225

[20] Wald A. Sequential tests of statistical hypotheses. The Annals of Mathematical Statistics, 1945, 16(2): 117-186

WANG Wei, born in 1981, Ph. D. candidate. Her research interests include network and information security, botnet.

FANG Bin-Xing, born in 1960, professor, Ph. D. supervisor, member of Chinese Academy of Engineering. His current research interests include computer architecture, computer network and information security.

CUI Xiang, born in 1978, Ph. D. candidate, assistant researcher. His research interest is network security.

Background

Botnet is not any kind of malware but a common distributed platform, and the main task of the platform is launching attacks such as distributed denial-of-service (DDoS), sending spam, seeding malwares, identity theft, phishing, and etc. Nowadays, more and more defenders research botnets and the research can be divided into four aspects. The first one is detection. Detection aims to find bot samples and make sure if client or channel belongs to a botnet. The second one is track. Track is composed of measurement and monitoring activities. Measurement aims to find the membership of a botnet, combine detected sub-botnets to one big botnet or divide detected botnet into several small botnets. Monitoring records commands of a botnet. The third one is mitigation. Mitigation aims to reduce the destruction of botnets from C&C level and host level. The last one is hijack. Hijack aims to get the partly or completely control of a detected botnets.

This work researches botnets detection. Although P2P-based botnets and HTTP-based botnets become more and more popular, IRC-based botnets are always the valuable tool for botmasters. There are two problems in existed algorithms for IRC-based botnets detection. One is that detection algorithms require some prior knowledge of botnet to generate matching patterns. The other is that algorithms can not perform detection online. To solve these problems, the authors propose two IRC botnet detection algorithms based on host behavior. One is based on the similarity of nicknames and the other is based on the similarity of command sequences. The authors will study protocol free detection algorithm in the future. This work is partly supported by the National Basic Research Program (973 Program) of China under grant No. 2007CB311100 and the National High Technology Research and Development Program (863 Program) of China under grant Nos. 2009AA01Z437, 2007AA010501 and 2007AA01Z474.