基于 Command and Control 通信信道流量属性聚类的僵尸网络检测方法

苏 欣[①] 张大方*[①] 罗章琪[①] 曾 彬[②] 黎文伟[①]
[①](湖南大学信息科学与工程学院 长沙 410082)
[②](中国移动湖南分公司 长沙 410015)

摘 要：僵尸网络(Botnet)是一种从传统恶意代码形态进化而来的新型攻击方式，为攻击者提供了隐匿、灵活且高效的一对多命令与控制信道(Command and Control channel, C&C)机制，可以控制大量僵尸主机实现信息窃取、分布式拒绝服务攻击和垃圾邮件发送等攻击目的。该文提出一种与僵尸网络结构和 C&C 协议无关，不需要分析数据包的特征负载的僵尸网络检测方法。该方法首先使用预过滤规则对捕获的流量进行过滤，去掉与僵尸网络无关的流量；其次对过滤后的流量属性进行统计；接着使用基于 X-means 聚类的两步聚类算法对 C&C 信道的流量属性进行分析与聚类，从而达到对僵尸网络检测的目的。实验证明，该方法高效准确地把僵尸网络流量与其他正常网络流量区分，达到从实际网络中检测僵尸网络的要求，并且具有较低的误判率。

关键词：网络检测；聚类；僵尸网络检测；命令与控制信道；流量属性
中图分类号：TP393 **文献标识码**： A **文章编号**：1009-5896(2012)08-1993-07
DOI: 10.3724/SP.J.1146.2011.01098

Botnet Detecting Method Based on Clustering Flow Attributes of Command and Control Communication Channel

Su Xin[①] Zhang Da-fang[①] Luo Zhang-qi[①] Zeng Bin[②] Li Wen-wei[①]
[①](*Information Science and Engineering College Hunan University*, *Changsha* 410082, *China*)
[②](*China Mobile Group Hunan Company Limited*, *Changsha* 410015, *China*)

Abstract: Botnet is a novel attack strategy evolved from traditional malware forms; It provides the attackers stealthy, flexible and efficient one to many Command and Control (C&C) mechanisms, which can be used to order an army of zombies to achieve the goals including information theft, launching Distributed Denial of Service (DDoS), and sending spam. This paper proposed a botnet detecting method which independent of botnet C&C protocol and structure, and not analysis payload of packets. At first this method use pre-filter rules to filter flow which have irrelevant with botnet; Second, the flow attributes are analyzed; Third, two-steps clustering algorithm which based on X-means clustering is used to analyze and cluster flow attributes of C&C channel, and the botnet detection is implemented. The experiment shows that this method can differentiate traffic of botnet and normal network with high accuracy, low false positive, achieve the goal that detects botnet under real network environment.

Key words: Network detection; Clustering; Botnet detection; Command and Control (C&C) channel; Flow attributes

1 引言

僵尸网络是指攻击者隐藏在僵尸控制者(Botmaster)背后利用各种手段传播僵尸程序来达到控制大量主机的目的，被控制的主机被称为僵尸主机(zombies)，由这些僵尸主机所组成的网络称为攻击者的一个攻击平台。它是在网络蠕虫、特洛伊木马、后门工具等传统恶意代码形态的基础上发展、融合而成的一种新型攻击方式[1]。它通过分布式拒绝服务攻击、垃圾邮件、窃取敏感信息等恶意活动使得全球的网络安全领域面临严峻的考验。

僵尸控制者通过 C&C 信道来操纵这些僵尸主机，这些命令与控制信道和僵尸主机共同组建了僵尸网络。集中化的僵尸网络的 C&C 结构使用网络中继聊天协议(Internet Relay Chat, IRC)[2]，超文本传输协议(HyperText Transfer Protocol, HTTP)[3]协议作为通信协议。这种结构存在单点失效的问题，随着技术的发展，非集中化的 C&C 结构开始出现，这种结构使用 P2P 协议作为通信协议[4]，甚至加强型的 P2P 协议[5]。其中，Nugache[6]和 Storm worm[7]

2011-10-24 收到，2012-04-24 改回
国家自然科学基金项目(61173167, 61173168, 61070194)和国家发改委信息安全专项资助课题
*通信作者：张大方 dfzhang@hnu.edu.cn

是两种具有代表性的 P2P 僵尸网络。目前国内外已经逐步深化了对僵尸网络功能结构、工作原理、命令与控制机制、传播模型等方面的研究。但是僵尸网络存在行为隐蔽，产生的网络流量很少，P2P 僵尸网络的流加密等特点，对检测技术提出了巨大的挑战。

在文献[8,9]使用蜜罐技术去采集研究僵尸网络。文献[10]通过使用多方面的方法来收集波特程序和跟踪僵尸网络来实现对僵尸网络活动的深度测量。文献[11]对多个常见的僵尸程序的代码进行了详细的研究。文献[12]阐述了他们对于僵尸网络和扫描、发送垃圾邮件等活动之间的联系。文献[13]提出了使用 DNS-based blackhole list(DNSBL)智能计数器来发现发送垃圾邮件的僵尸程序。该方法对于特定的发送垃圾邮件的僵尸网络非常有效。文献[14]提出了一种基于异常行为的 IRC 僵尸网络自动检测系统，该系统通过识别 IRC 网络的异常行为来实现对 IRC 僵尸网络的检测。文献[15]中阐明了恶意 HTTP 僵尸程序是以规律性的间隔反复连接 HTTP 服务器，不同于正常的用户程序。实验结果表明，检测效果良好，但如果正常用户用程序自动连接HTTP服务器，则可能会产生误报。Botgraph[16]是一种基于图论的系统，研究发现僵尸程序会冒充用户注册和发送电子邮件时会共享 IP 地址。文献[17,18]中，研究者使用机器学习的方法来识别 P2P 僵尸网络和 IRC 僵尸网络流量。在文献[19]中，作者提出了一种根据僵尸网络通信流量负载特征来建立决策树模型，利用该模型对僵尸网络进行分类。实验结果表明该方法对检测 IRC 僵尸网络流量具有较高的准确率和可以接受的误判率。但是该方法不适用于流量内容加密的僵尸网络。以上所述的所有方法中，有些适合检测具有特定类型的通信机制或者发起特定类型的攻击的僵尸网络，有些只针对僵尸网络所产生的流量的某一方面的特征进行数据挖掘或机器学习，有些只适合于负责特征为明文的流量。

本文提出的僵尸网络检测方法的主要思想是：首先对所捕获的流量进行过滤，把不相关的流量过滤掉；然后对僵尸网络的 C&C 信道所产生的流量特征进行统计；接着使用聚类算法对统计得到的数据进行聚类，从而得到僵尸网络不同于正常网络应用或者网络行为所产生的流量的特征，从而实现对僵尸网络的检测。该方法所要达到的目的：(1)独立于僵尸网络中的结点进行通信的协议和结构；(2)独立于 Botmaster 或者 P2P 中的结点进行通信的内容；(3)具有较低的误判率和漏判率。

本文的组织结构如下：第 2 节介绍本文所提出的方法，实现僵尸网络聚类检测方法；第 3 节对僵尸网络的流量属性进行聚类并验证聚类结果；第 4 节是结束语。

2 基于 C&C 信道流量属性的僵尸网络检测方法

2.1 预过滤规则

由于聚类算法无法很好地处理噪声数据，并且受僵尸程序感染的主机所产生的流量并不仅仅只有僵尸网络 C&C 信道所产生的流量，大部分是正常网络的流量，这就会对聚类僵尸网络 C&C 信道流量带来不利的影响。本文预先过滤掉与僵尸网络流量不相关的流量。

本文定义了 3 个预过滤规则：规则 1(R1)：过滤由内部主机之间所产生的流量；规则 2(R2)：过滤没有完全确立连接的流量，这些流量只包含一次握手，比如 SYN-FLOOD 的流量；规则 3(R3)：过滤主机访问众所周知的，合法的服务器的流量，比如访问百度、新浪等，这些流量属于正常网络行为所产生的。在本文的实验中，白名单中国内网站 TOP 100 是从 chinarank.org.cn 获得，国际网站 TOP 100 是从 alexa.com 获得。

过滤流量后，剩下的流量基本上是僵尸网络 C&C 信道或者被疑为僵尸网络 C&C 信道所产生的流量。在流量经过预过滤规则过滤后，本文为了进一步减少处理网络流量的工作量，对具有相同五元组的流进行聚合，即 TCP/UDP 的流具有相同的协议、源 IP、目的 IP、源端口、目的端口，在本文中被认为是同一条流；如果这五元组中任意某个或者多个元素不同，那么即认为是不同的流。

2.2 僵尸网络 C&C 通信信道流量特征分析

2.2.1 僵尸网络 C&C 信道流量采集和观察 本文通过蜜罐主机技术对僵尸网络所产生的流量进行采集和观察。首先在一台服务器上安装多个虚拟机(VM)，每个 VM 安装 Windows XP SP2 的操作系统；其次把不同的应用分别安装在不同的 VM 上，对每个 VM 以不同的应用的名称命名；然后分别运行 VM 里面的应用，并使用 Wireshark 来收集应用所产生的流量。图 1 分别是僵尸网络、网页浏览、在线游戏和 P2P 应用产生的流量的数据包的大小分布比例图。

从图 1(a)中可以看到，僵尸网络 C&C 信道所产生的流量的数据包分布主要集中在长度为 0–79 Bytes 区间，因为僵尸程序在感染主机后需要与 C&C 服务器进行互相通信来告知 botmaster 该主机已经成功感染，并且与 botmaster 协商来确定发起

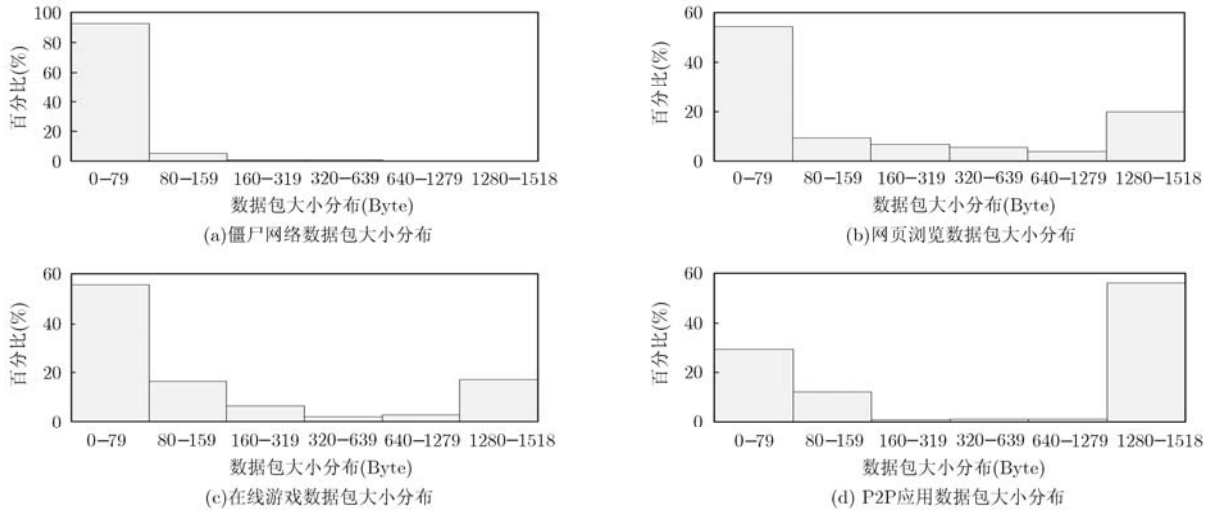

图 1 僵尸网络与一般网络应用数据包长度分布对比图

攻击的信息。图 1(b), 1(c), 1(d)中，网页浏览和在线游戏所产生的流量的数据包长度分布主要集中在 0–79, 80–159 和 1280–1518 Byte 这 3 个区间，因为网页浏览需要一个请求连接建立过程，在线游戏需要一个连接服务器账号密码验证过程；P2P 应用所产生的流量的数据包长度主要分布在 0–79 和 1280–1518 Byte 这 2 个区间，因为 P2P 应用的网络连接建立过程是首先协商阶段，协商完毕就开始大数据的文件传输。此外，我们可以看出僵尸程序与 C&C 服务器之间通信的流量在数据包长度分布与其他的一般网络应用是存在一定的相似性，这也说明僵尸网络 C&C 信道的流量是具有隐密性的，善于伪装在其他的网络应用的流量中。

2.2.2 僵尸网络 C&C 信道流量属性统计　在僵尸网络中，Botmaster 通过 C&C 信道来对僵尸程序进行命令和控制的，并且这些命令和控制行为是预先编程设定好的，这是不同于一般的网络应用的行为。本文通过对已经捕获的僵尸网络 C&C 信道的流量的各个属性进行统计，并与一般网络应用所产生的流量属性进行对比，来发现僵尸网络 C&C 信道流量固有的特征。

本文根据 2.2.1 节中方法分别采集了僵尸网络、网页浏览，在线游戏和 P2P 应用(迅雷，PPS, PPLIVE 等)的流量，分别根据这些流量的平均流个数、平均每个流包含数据包个数、平均数据包长度、每小时流个数等属性来进行统计和对比。其中，正常网络应用的流量数据的采集来自于 4 台 windows XP SP2 的主机，采集时间为 24 h 左右。僵尸网络的流量数据的采集来自于 4 台 windows XP SP2 的虚拟机下所运行的僵尸程序，采集时间为 24 h 左右。具体数据见表 1。

表 1 不同网络应用流量属性信息

流量属性信息	僵尸网络	网页浏览	在线游戏	P2P 应用
平均每条流包含数据包个数	10	30	24	17
数据包个数标准方差	12	35	56	86
小包流所占比例(%)	76	6	8	12
小包所占比例(%)	72	6	58	11
数据包平均长度	93	593	317	448
数据包长度标准方差	99	637	729	580
平均每秒数据包个数	2	84	38	352
每小时流个数	2486	12025	5669	38760
每秒字节数	1179	19077	24369	157708

其中，小包指长度在 64–399 Byte 范围内的数据包，小包流是指这个流所包含的数据包的平均长度在 64–399 Byte。因为这个范围的数据包一般是双方主机通信或者协商阶段互发的数据包，很少不存在文件的传输的数据包。从表 1 中看到，僵尸网络产生的流量在平均每条流包含数据包个数、小包流所占比例、小包所占比例、数据包平均长度等属性方面的数据统计符合第 1 节中所提到的僵尸网络的所具有的通信隐秘，产生的网络流量少等特点，并且在这些属性方面与其他正常网络应用所产生的流量有明显的不同，所以这些属性适合作为聚类的指标来实现僵尸网络的检测。

2.3 基于 C&C 通信信道流量聚类的僵尸网络检测方法

2.3.1 流的组属性　属于同样僵尸网络的僵尸程序与 C&C 服务器之通信具有相似的行为。比如属于同一僵尸网络的两个僵尸程序连接两个不同的 C&C 服务器(因为某些僵尸网络可能有多个 C&C 服务器)，尽管从僵尸程序连接 C&C 服务器的信息看来，这

两个连接由于源目的 IP&PORT 不同而被认为是不同的流，但是与 C&C 服务器通信的流的行为存在相似性，所以这两条流理论上是可以聚类的。本文给定一个时间周期 T(trace 的采集时间)，所有 m 条 TCP/UDP 具有相同五元组的流可以聚集为一条流 $f_i=\{p_j\}_{j=1,\cdots,m}$，其中 p_j 表示单个 TCP/UDP 的流。数据集 $D=\{f_i\}_{i=1,\cdots,n}$ 表示在 T 时间内所观察到的 TCP/UDP 的流，这些流包含了主机与网络之间的交互流。本文把数据集 D 的属性定义为一个 d 维的组属性，定义如下：

定义 1 设集合 $S=\{s_1,s_2,\cdots,s_d\}$，其中 $d\in N^+$，若 s_i 和 $s_j(1\le i,j\le d)$ 表示 2 个不同的流属性，d 是一个固定常数，那么集合 S 可以表示为流 $D=\{f_i\}_{i=1,\cdots,n}$ 的组属性，s_i 或 s_j 称为组属性 S 的属性成员，d 称为组属性的维度，本文取值为 9。当使用维度为 d 的组属性进行聚类分析的时候，等价于使用 $s_i(s_i\in S,1\le i\le d)$ 对数据集 D 进行聚类分析。

2.3.2 两步聚类方法 由于本文分析的流量数据比较大，并且所选择的流的组属性维度 d 也比较大，所以导致数据集 D 的基数比较大；其次，被僵尸程序感染的主机在网络所有的主机中所占的比例还是相对少的，需要把小部分和僵尸网络有关的流从大部分正常的流中分离出来。以上两点使得对数据集 D 的聚类分析变得比较困难。为了解决这个困难，本文使用两步聚类的方法。该方法步骤如下：第 1 步，减少组属性 S 的维度 d 的大小，进行粗粒度的聚类，聚类完成后得到 r_1 个簇 $\{C_i'\}_{i=1,\cdots,r_1}$。按照这个方法把数据集 D 划分成多个 C_i'；第 2 步，在第 1 步得到的结果的基础上对每个不同的数据集 C_i' 使用相同的聚类算法，得到了相对较小但更精准的聚类 $\{C_i''\}_{i=1,\cdots,r_2}$。

在这两步聚类的过程中，本文使用的是 unmerged X 均值聚类算法[20]，该算法基于 K 均值算法[21]改进，但是 X-means 算法不需要用户去选择 K 个初始化的簇中心，X 均值通过内部运行多次 K 均值算法，并使用贝叶斯信息标准[20]来验证聚类计算得出 K 的最优值。unmerged X 均值是一种快速的，适用于大规模数据集的聚类算法。具体实现如下：

(1)第 1 步聚类中，减少组属性 S 的维度 d，从原有的 9 个属性减少到 4 个属性：每个数据包包含的字节数、每小时流个数、每秒数据包个数和每秒字节数(在文献[22]中使用这 4 个属性进行聚类)，然后使用 unmerged X 均值算法进行粗粒度聚类，聚类后得到簇 $\{C_i'\}_{i=1,\cdots,r_1}$；

(2)第 2 步聚类中，对组属性 S 中的所有属性使用 unmerged X 均值算法进行聚类，进一步从(1)的结果中提取更加准确的聚类结果。

聚类完成后，本文提出使用僵尸网络实例比例(Botnet Instances Percentage, BIP)和正常网络实例比例(Normal Instances Percentage, NIP)来验证聚类的结果。计算公式如式(1)：

$$\left.\begin{aligned}\text{BIP}&=\frac{\text{簇内的僵尸网络实例个数}}{\text{簇内所有网络实例个数}}\\ \text{NIP}&=\frac{\text{簇内的正常网络实例个数}}{\text{簇内所有网络实例个数}}\end{aligned}\right\} \tag{1}$$

通过这两个指标来验证对所捕获的数据集在聚类后的结果，BIP 表示簇内的主要成分是僵尸网络实例，NIP 表示簇内的主要成分是正常网络实例。这两个指标的值范围在 0%到 100%之间，而且在理想情况下，在聚类后，划分的簇的这两个指标应该是反比关系，随着一个指标趋近 1，而另一个指标趋近 0。

3 实验

3.1 实验准备和数据采集

本文实验中所用的正常网络 trace 来自湖南大学网络中心出口链路这些 trace 包括了 HTTP, SMTP, FTP, QQ，迅雷，QQ 游戏等正常网络应用所产生的流量，僵尸网络 trace 是由 4 台装有僵尸程序的虚拟机所产生的僵尸网络 C&C 信道所产生的流量。根据 2.1 节中所介绍的过滤规则来对包含了僵尸网络流量的 trace 进行过滤，去掉与僵尸网络无关的流量，剩下的包含了僵尸网络 C&C 通信信道所产生的流量。表 2 列出了这些流量的基本信息。

表 2 中看到，僵尸网络的 trace 包含了基于 IRC 的僵尸网络(sdbot, Black energy bot)，基于 HTTP 的僵尸网络(Backdoor cybot)和基于 P2P 的僵尸网络(Peacomm)。其中，基于 IRC 的僵尸网络的 trace 分别包含了 9 min 和 18 h 的 IRC C&C 通信的流量，botmaster 通过 C&C 信道发送命令，僵尸程序通过扫描来进入到这个信道中。加入信道后，基于 IRC 的僵尸程序通过二进制的信息来通知 botmaster 该主机已经成功被感染。而基于 HTTP 的僵尸网络的

表 2 各种僵尸网络 C&C 信道所产生的流量的基本信息

僵尸网络 trace	大小	持续时间	数据包个数	TCP/UDP 流个数
sdbot	64 kB	9 min	474	19
Black energy bot	43 mB	18 h	282001	70101
Backdoor cybot	15 mB	4 h	96301	24952
Peacomm	322 mB	56 h	2215241	189931

trace 包含了 4 h 的 C&C 通信流量，这些流量相当隐秘，僵尸程序在 0 到 10 min 之间任意选择一个时间来与 botmaster 通信。Peacomm 是一个主要以 UDP 为主的 P2P 僵尸程序，它是基于 Kademlia 协议[23]并通过覆盖网来定位相关数据。

为了便于 X 均值算法对所捕获的流量数据集进行聚类分析，根据第 2 节中所提出的组属性的概念，本节把所捕获的流量按照组属性的格式划分成两类实例，分别是僵尸网络实例(107)和正常网络实例(194)，其中正常网络实例就包括网页浏览、P2P 应用、在线游戏等。

3.2 实验结果

unmerged X 均值算法把聚类的数据集划分成两个不同的簇，而本文实验的目的是通过聚类把僵尸网络实例和正常网络实例分开，所以该方法适用。根据 2.3.2 节中所提出的两步聚类分析方法，经过第 1 步粗粒度的聚类得到各个簇的信息如表 3 所示。

表 3　第 1 步聚类得到的簇信息

指标	聚类 0		聚类 1	
	僵尸网络实例	正常网络实例	僵尸网络实例	正常网络实例
个数	28	142	79	52
BIP/NIP(%)	16.5/83.5		60.3/39.7	

根据表 3，聚类 0 和聚类 1 的 BIP/NIP 分别是 16.5%/83.5%和 60.3%/39.7%，根据前面所提到的利用 BIP 和 NIP 两个指标来验证聚类的效果，这表明粗粒度聚类所选择的 4 个属性不能够很好地把僵尸网络实例和正常网络实例区分，说明这 4 个属性并不能完全区分僵尸网络与正常网络不同。

根据 2.3.2 节的方法，在此基础上进行第 2 步聚类分析，其结果如表 4 所示。

经过第 2 步聚类后，聚类 0 和聚类 2 的 BIP/NIP 分别是 96.6%/3.4%和 95.2%/4.8%，即聚类的准确率为 95.5%，误判率为 4.5%，即为僵尸网络簇；聚类 1 和聚类 3 的 BIP/NIP 分别是 0/100%和 0/100%，即为正常网络簇。综合表 4 的结果，本文一共有 301 个实例，其中 107 个僵尸网络实例被正确的聚类，194 个正常网络实例中有 5 个被误判为僵尸网络实例，所以本文算法的准确率为 98.34%，误判率为 1.66%。这说明根据本文所选用的流量属性，经过两步聚类的方法，可以很好地把僵尸网络流量和正常网络流量区分开来，并且具有较低的误判率。

表 4　第 2 步聚类得到的簇信息

指标	聚类 0		聚类 1		聚类 2		聚类 3	
	僵尸网络实例	正常网络实例	僵尸网络实例	正常网络实例	僵尸网络实例	正常网络实例	僵尸网络实例	正常网络实例
个数	28	1	0	141	79	4	0	48
BIP/NIP(%)	96.6/3.4		0/100		95.2/4.8		0/100	

3.3 与其他检测方法对比

由于网络环境和实验中选择的僵尸程序的不同，很难去公平地比较不同僵尸网络检测方法的性能和识别精度。所以本文选择一些与本文思路类似的僵尸网络检测方法[22,24−26]进行对比。对比中所用到的僵尸网络特征以及描述如表 5 所示。

表 5　基于自动检测的僵尸网络特征描述

特征	描述
基本原理	检测方法的类型：基于主机或者基于网络
IRC	是否依赖特定的 IRC 端口或者对 IRC 通信模型建模
流量属性	通过流量属性关联 C&C 通信信道
时间	采集流量数据是否需要一定时间
语义	是否依赖僵尸的特殊的昵称、命令或者协议语义

在表 6 中列出了对比的结果，发现本文所提出的方法不依赖于特定的僵尸网络结构，也不需要关联其他的 IDS，不需要对 C&C 信道流量的语义进行检测。

文献[17]首先在一台 PC 机中安装多个虚拟机，分别运行 P2P 僵尸程序和其他正常网络应用，并通过 8 M/512 K 的 ADSL 来访问外网。接着对这台 PC 机所产生的网络流量进行 5 min 的监测和捕获，生成了 45 个数据样本，这些数据样本包括 P2P 僵尸网络和其他正常网络应用的流量数据。最后使用了 3 种著名的分类算法对 P2P 僵尸网络和正常网络应用进行区分。其分类结果和本文的算法结果对比如表 7 所示。

从表 7 可以看出，本文方法的准确率要稍微高于文献[17]所使用的方法中准确率最高的一个。本文方法虽有 1.66%的误判率，但是文献[17]的方法只针对 P2P 僵尸网络，并且实验数据采集时间只有 5 min，可用于实验的数据只有 45 个，远远少于本文的 301 个，缺乏普遍性。

表 6 本文方法与其他方法对比

名称	基本原理	IRC	时间	语义	流量属性
Strayer	基于网络	是	是	否	数据包平均字节数、每秒字节数、每秒数据包个数等
Rishi	基于网络	是	否	是	没有
BotHunter	基于网络	是	是	是	没有
BotMiner	基于网络	否	是	否	数据包平均字节数、每小时流个数、每秒数据包个数和每秒字节数
本文方法	基于网络	否	是	否	流大小、数据包平均长度、数据包个数、每秒字节数等

表 7 本文方法与文献[17]方法结果对比

方法名称	准确率(%)	误判率(%)	漏判率(%)
J48	97.78	0	2.22
NaiveBayes	88.89	0	11.11
BayesNet	86.67	6.67	6.67
本文方法	98.34	1.66	0

4 结束语

本文所介绍的基于 C&C 通信信道流量的僵尸网络检测方法，首先对采集到的流量过滤，去掉与僵尸网络无关的流量；接着对僵尸网络 C&C 通信信道流量和其他正常网络应用流量属性统计和分析；然后使用 unmerged X-means 算法对统计的属性进行聚类，从而实现从网络流量中检测僵尸网络。该方法可以无需检测数据包特征负载，无关于僵尸网络结构，可以检测基于不同的网络协议、网络结构的僵尸网络。实验也验证了该检测方法的可行性、准确性和高效性。

首先，本文的方法需要一定时间来收集数据，不适合于在线检测；其次，随着僵尸网络不断升级，有些僵尸程序可能只和 C&C 服务器进行一次通信，甚至不进行通信，在这种情况下本文的方法就很难去检测僵尸网络，所以如何实现实时地、主动地去检测僵尸网络将是下一步工作的重点。

参 考 文 献

[1] 诸葛建伟, 韩心慧, 周勇林, 等. 僵尸网络研究与进展[J]. 软件学报, 2008, 19(3): 702–715.

[2] Shin Seungwon and Gu Guofei. Conficker and beyond: a large-scale empirical study[C]. Proceedings of 2010 Annual Computer Security Applications Conference (ACSAC'10), Austin, Texas, USA, 2010: 151–160.

[3] Chia Yuan-cho, Juan Caballero, Grier C, *et al.*. Insights from the inside: a view of botnet management from infiltration[C]. Proceedings of the USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET), San Jose, CA, USA, 2010: 1–8.

[4] Yen T F and Reiter M K. Are your hosts trading or plotting? telling P2P file-sharing and bots apart[C]. IEEE 30th International Conference on Distributed Computing Systems (ICDCS), Genoa, Italy, 2010: 241–252.

[5] Wang Ping, Sparks S, and Zou C. An advanced hybrid peer-to-peer botnet[J]. *IEEE Transactions on Dependable and Secure Computing*, 2010, 7(2): 113–127.

[6] Lemos R. Bot software looks to improve peer-age[OL]. http:// www.securityfocus.com/ news/11390, 2006.

[7] Holz T, Steiner M, Dahl F, *et al.*. Measurements and mitigation of peer-to-peer-based botnets: a case study on storm worm[C]. Proceedings of the First USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET'08), San Francisco, CA, USA, 2008: 88–96.

[8] Wang Ying, Jin Zhi-gang, and Zhang Wei. Analysis of botnet attack and defense technology[C]. Computer Science and Service System (CSSS), Pairs, France, 2011: 3021–3023.

[9] Baecher P, Koetter M, Holz T, *et al.*. The nepenthes platform: an efficient approach to collect malware[C]. Proceedings of International Symposium on Recent Advances in Intrusion Detection (RAID'06), Hamburg, September 2006, Vol. 4219: 165–184.

[10] Rajab M, Zarfoss J, Monrose F, *et al.*. A multi-faceted approach to understanding the botnet phenomenon[C]. Proceedings of ACM SIGCOMM/USENIX Internet Measurement Conference (IMC'06), Brazil, October 2006: 41–52.

[11] Barford P and Yegneswaran V. An inside look at botnets[C]. Special Workshop on Malware Detection, Advances in Information Security, Berlin, Germany, 2006: 171–191.

[12] Collins M, Shimeall T, Faber S, *et al.*. Using uncleanliness to predict future Botnet addresses[C]. Proceedings of ACM/USENIX Internet Measurement Conference (IMC'07), San Diego, CA, USA, 2007: 93–104.

[13] Ramachandran A, Feamster N, and Dagon D. Revealing botnet membership using DNSBL counter-intelligence[C]. Proceedings of USENIX SRUTI'06, San Jose, CA, USA, 2006: 49–54.

[14] Wang Zi-long, Wang Jin-song, Huang Wen-yi, *et al.*. The detection of IRC botnet based on abnormal behavior[C]. Multimedia and Information Technology (MMIT), KaiFeng, China 2010, Vol. 2: 146–149.

[15] Lee J S, Jeong H C, Park J H, *et al.*. The activity analysis of

malicious http-based Botnets using degree of periodic reparability[C]. Proceedings of 2008 International Conference on Security Technology (SecTech2008), Washington, DC, IEEE Computer Society, 2008: 83–86.

[16] Wang Zhen-qi and Fu Li. The research of detecting IRC botnet based on k-means algorithms[C]. Communication Systems, Networks and Applications (ICCSNA), Hong Kong, China, 2010: 208–210.

[17] Liao Wen-hwa and Chang Chia-ching. Peer to peer botnet detection using data mining scheme[C]. Conference of Internet Technology and Applications, Wuhan, China, 2010: 1–4.

[18] Langin C, Zhou H, Rahimi S, *et al*.. A self-organizing map and its modeling for discovering malignant network traffic[C]. IEEE Computational Intelligence in Cyber Security (CICS '09), Nashville TN, USA, 2009: 112–119.

[19] Lu W, Rammidi G, and Ghorbani A. Clustering botnet communication traffic based on n-gram feature selection [J]. *Computer Communications*, 2010, 34(3): 502–514.

[20] Pelleg D and Moore A W. X-means: extending k-means with efficient estimation of the number of clusters[C]. Proceedings of the Seventeenth International Conference on Machine Learning (ICML'00), San Francisco CA, USA, 2000: 727–734.

[21] MacQueen J. Some methods for classification and analysis of multivariate observations[C]. Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, San Francisco CA, USA, 1967: 281–297.

[22] Gu G, Perdisci R, Zhang J, *et al*.. BotMiner: clustering analysis of network traffic for protocol- and structure-independent botnet detection[C]. Proceedings of the 17th USENIX Security Symposium (Security'08), San Jose CA, USA, 2008: 378–393.

[23] Maymounkov P and Mazieres D. Kademlia: a peer-to-peer information system based on the XOR metric[C]. Proceedings of the 1st International Workshop on Peer-to-Peer Systems (IPTPS'02), Cambridge MA, USA,2002, Vol. 2429: 53–65.

[24] Strayer T, Walsh R, Livadas C, *et al*.. Detecting botnets with tight command and control[C]. Proceedings of the 31st IEEE Conference on Local Computer Networks (LCN'06), Washington, DC, 2006: 195–202.

[25] Goebel J and Holz T. Rishi: identify bot contanminated hosts by IRC nickname evaluation [C]. Proceedings of USENIX HotBots'07, Berkeley, CA, USA, 2007: 163–174.

[26] Gu G F, Porras P, Yegneswaran V, *et al*.. BotHunter: detecting malware infection through IDS-Driven dialog correlation[C]. Proceedings of the 16th USENIX Security Symposium, Boston, MA, 2007: 167–182.

苏　欣：　男，1983 年生，博士生，研究方向为僵尸网络检测、数据包深度检测、正则表达式匹配.

张大方：　男，1959 年生，教授，博士生导师，研究方向为可信系统与网络、软件容错、软件测试.

罗章琪：　男，1988 年生，硕士生，研究方向为僵尸网络检测.

曾　彬：　男，1979 年生，博士生，研究方向为网络流量检测.

黎文伟：　男，1975 年生，副教授，研究方向为网络测试、可信系统与网络.