doi:10.3969/j.issn.1000-436x.2013.10.006

# 基于多维流量特征的 IRC 僵尸网络频道检测

闫健恩[1]，袁春阳[2]，许海燕[1]，张兆心[1]
（1. 哈尔滨工业大学 计算机科学与技术学院，黑龙江 哈尔滨 150001；2. 国家计算机网络应急技术处理协调中心，北京 100029）

**摘 要：**针对 IRC 僵尸网络频道的检测问题，提出一种基于流量特征的检测方法。分析了僵尸网络频道数据流在不同周期内流量的聚类性、相似性、平均分组长度、流量高峰和协同流量高峰等特征，并以此作为僵尸网络频道检测的依据。检测过程中，采用改进的最大最小距离和 *k*-means 聚类分析算法，改善了数据聚类的效果。最后经过实验测试，验证了方法的有效性。

**关键词：**IRC 协议；僵尸网络；数据流；聚类分析
**中图分类号：**TP393.08 **文献标识码：**A **文章编号：**1000-436X(2013)10-0049-07

# Method of detecting IRC Botnet based on the multi-features of traffic flow

YAN Jian-en[1], YUAN Chun-yang[2], XU Hai-yan[1], ZHANG Zhao-xin[1]
(1.School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China;
2. National Computer Network Emergency Response Technical Team/Coordination Center of China, Beijing 100029, China)

**Abstract:** To resolve the problem of detecting IRC Botnet, a method based on traffic flow characteristics was proposed. The characteristics of Botnet channel traf?c were analyzed in different periods such as data-clustering, data-similarity, the average length of packet, peak of synchronized traf?c, and peak of collaborative synchronized traf?c, and these characteristics were used to detect the botnet. In analyzing, improved max-min distance means and k-means cluster analysis algorithm were also presented to promote the efficiency of data clustering. At last, the availability of the method was verified by experiment.

**Key words:** IRC protocol; Botnet; traffic flow; cluster analysis

## 1 引言

僵尸网络（Botnet）是从传统恶意代码形态的基础上进化的，并通过相互融合发展而成为目前最为复杂的攻击方式之一[1]。僵尸网络作为一种新型的攻击方式，给互联网的安全造成了严重威胁和巨大损失。CNCERT 2012 年 11 月监测报告指出，我国有 165 万余个 IP 地址对应的主机被木马或僵尸程序控制[2]。

从拓扑结构上看，基于 IRC 协议的僵尸网络是一种集中式僵尸网络[3]。虽然分布式僵尸网络比较流行，但 IRC 僵尸网络由于实现简单、操控性灵活，目前在互联网上仍然活跃。如何准确检测僵尸网络及其控制通信信道，是一个很大的挑战。僵尸网络检测技术从数据来源可分为基于网络流量数据和其他系统信息数据(如 DNS 日志数据、入侵检测记录数据

收稿日期：2013-04-28；修回日期：2013-07-01
基金项目：国家高技术研究发展计划(“863”计划)基金资助项目(2007AA010503)；国家自然科学基金资助项目(61100189，61003261)；国家科技支撑计划基金资助项目(2012BAH45B01)；山东省中青年科学家奖励基金资助项目(BS2011DX001)；威海市科技攻关基金资助项目(2010-3-96)；哈尔滨工业大学科研创新基金资助项目(HIT.NSRIF.2011119)
**Foundation Items:** The National High Technology Research and Development Program of China (863 Program) (2007AA010503)；The National Natural Science Foundation of China (61100189, 61003261)；The National Science and Technology Support Program(2012BAH45B01)；Young and Middle-Aged Scientists Research Awards Fund of Shandong Province(BS2011DX001)；Weihai Municipal Science and Technology Research(2010-3-96)；Harbin Institute of Technology Scientific Research Innovation Foundation(HIT.NSRIF.2011119)

等)；从异常模式分析可分为基于内容特征分析(如通信端口、特殊二进制代码等)和基于行为特征分析(如网络行为特征或流量行为特征)。在分析和检测研究中，最早开展跟踪研究工作的团队是德国蜜网项目组，他们对近100个Botnet活动进行了跟踪和分析。此外，FEDYNYSHYN等[4]使用不同的分类算法，分析了正常网络流量和僵尸网络流量在发送和接收数据数量上的差异，结合域名和IP地址，实现了僵尸网络的检测，但其只分析了流量数量单一的特征，具有一定的缺陷。文献[5]分析僵尸网络发起攻击时的异常TCP流量，以此进行僵尸网络检测，不过，此方法对非TCP流量的攻击或者非攻击活动的僵尸网络检测无效。另外，文献[6]中使用恶意URL黑名单，分析了僵尸网络使用域名的特性，从而实现了对僵尸网络活动的追踪，并对可疑URL进行判别。文献[7]利用僵尸网络客户端定期持续向控制者主动进行连接的特性，借助正常目的地址白名单，检测僵尸网络。还有LU等[8]使用隐式马尔可夫链模型对僵尸网络的流量进行分析和检测。文献[9]中对僵尸网络的整体情况和目前的检测技术进行了较全面的介绍。文献[10]介绍了基于智能手机短信技术构建新型移动互联网僵尸网络。

基于流量特征的检测适应于大规模数据量的情况，不进行数据内容分析，相对执行效率较高，能够满足实时分析的需求。不过，以往流量技术检测僵尸网络的方法中，关注的只是数据流的部分或局部时段属性特征，从而不能全面刻画僵尸网络数据流的特点，故本文从多种僵尸网络频道数据流的特征出发，考虑不同周期内数据流的特点，从而实现了对僵尸网络频道的检测。

## 2 僵尸网络频道流量特征

由于正常的网络活动可以看成是没有主观控制意识的随机活动，而IRC僵尸网络的活动具有人为控制的特点，因而从网络数据流量角度看，正常的网络流量随机性较强，即使有突发流量变化，也不会出现规律性的周期变化。而僵尸网络的数据流量由于人为主观控制的原因，具有周期性和规律性特点，下面分析一下僵尸网络流量的特征。

1) 聚类和相似性特征

同一频道中的不同客户端的流量存在聚类性质和相似性。文献[11]中，作者把同一频道内的正常主机和同一频道内的僵尸主机的数据流进行分析得到：同一僵尸频道内的僵尸主机与服务器之间形成的数据流具有很明显的聚类和相似性特征，而同一正常频道内的正常主机与服务器之间的数据流差距较大，不具有聚类和相似性特征。产生这种特征原因是：处于同一个频道的僵尸主机只能被动地接收来自服务器端的命令，执行相应的操作并返回操作结果，因此，同一频道上的所有僵尸主机在与服务器交互组成的数据流之间具有聚类和相似性特征；并且不同僵尸频道上的客户端数据流之间不存在聚类和相似性特征。对于正常的频道内的客户端数据流，由于客户端交互内容的随意性，因此不存在聚类和相似性特征。

2) 命令反应时间特征

僵尸程序内置的命令执行功能可以完成控制者通过服务器推送过来的命令，在较短的固定时间周期内向服务器返回回馈信息，而正常的IRC用户聊天，由于用户随机的思考时间和消息输入时间，则会存在数据之间不固定的时间间隔，它与僵尸程序的反应时间的特征差距很大。例如，正常IRC聊天过程中，用户在登录过程中需要人工输入用户名和密码，在选择频道时也会需要一定时间，而僵尸程序中已经内置登录过程中的一切信息，完成登录频道的时间仅需十几到几十毫秒。

3) 平均分组长度特征

在僵尸网络没有发动恶意行为时，绝大部分时间处于静默期，在这段时间内服务器和客户端之间主要通过PING/PONG分组维持网络连通。因为PING/PING分组长度较小且发送的时间间隔较大，因此，僵尸网络的平均分组长度通常较小，并且同一个服务器的PING/PONG数据分组的长度是固定不变的[12]。

4) 相邻数据分组的时间间隔特征

由于处于静默期的僵尸网络通过PING/PONG分组进行交互，而同一个服务器的相邻PING/PONG分组间的时间间隔相对稳定，文献[6]中对僵尸网络中的相邻PING/PONG数据分组之间的时间间隔进行测试，发现PING/PONG数据分组的时间间隔为90~110 s。因此，如果数据分组的到达时间间隔一直处在90~110 s，而且持续时间很长，则可以认为其处于僵尸网络的静默期。

5) 流量高峰和协同流量高峰特征

僵尸网络不同时期内，频道数据流量的大小是

不同的。处于静默期的僵尸网络，频道上只有少量的 PING/PONG 分组在服务器端和客户端进行交互；而处于攻击期的僵尸网络，会通过服务器在短时间内会发送大量的攻击命令，随后 bot 主机也会向服务器反馈其执行命令的结果。由于这种行为，僵尸网络在客户端会存在流量高峰的特征，并且服务器端和客户端会存在协同流量高峰特征。而正常 IRC 频道中主机之间的交互流量一般比较平缓，很难出现较大的波幅。

利用以上介绍的僵尸网络多种流量数据特征，本文提出了一个检测 IRC 僵尸网络频道的方法，通过流量特征的分析，发现僵尸网络活动。

## 3　基于多维流量特征的僵尸网络流频道检测

数据流可以为 2 个 IP 地址之间的所有数据分组组成的流，也可以为 2 个主机的（IP,PORT）对组成的流。因此，需要在研究中给出明确的数据流定义。

定义　数据流 $S=(sip,sport,dip,dport,data)$，其中 $sip$ 是 IRC 服务器 IP 地址；$sport$ 是 IRC 服务器端口；$dip$ 是客户端 IP 地址；$dport$ 为客户端端口；$data$ 是满足上述 IP 地址和端口条件的 2 个主机之间发送的所有数据分组。

僵尸网络频道数据的流量高峰和协同流量高峰、命令反应时间、平均分组长度和相邻数据分组的时间间隔这些特征，通过分析捕获的数据流即可获得，在此不做过多描述，下面主要讲述分析数据流的聚类和相似性的方法。

### 3.1　数据流聚类算法

在实际应用过程中，使用最大最小距离算法和 $k$-means 算法结合的方式进行数据流聚类分析，但是原始的聚类算法不能满足要求，故对 2 种算法进行改进，从而满足效率和性能的要求。原始的最大最小距离算法[13]和 $k$-means 算法[14,15]在此不做描述，着重描述改进的算法。

1) 改进的最大最小距离算法

最大最小距离算法中的第一个聚类中心 $c_1$ 是随机产生的，这样得到的距离 $|c_2-c_1|$ 也具有随机性。而在算法的最后需要判断是否需要产生新的聚类中心时要对参数 $m$ 进行设定，判断公式为 $\max_S\{Ds_j\}>m(|c_2-c_1|)$。因此，对参数 $m$ 设定一个固定值并得到很好的聚类效果是非常困难的。下面对最大最小聚类算法进行相关的改进以消减原始算法中的随机性。改进的最大最小算法如下。

算法输入：待聚类的数据集 $S=\{s_1,s_2,\cdots,s_n\}$。

算法输出：带有类别标识的数据集合。

算法描述：

步骤 1

$c_1=\forall s_i$，$c_2=s_j$ ，且 $\max|s_j-s_i|$

步骤 2

```
for each s_m∈S 且 s_m≠s_i && s_m≠s_j do
    d_m=min{|s_m-s_i|,|s_m-s_j|};
end for
for  each d_m do
    x=max{d_m};
end for
if  x> threshold then
     c_3=s_m;
     else
     exit;
end if
```

步骤 3

重复步骤 2 的方法直到没有新的聚类中心产生。

算法步骤 2 中的 threshold 是设定的固定值，可以根据实验数据和专业领域知识确定，这样处理的优点是聚类中心的产生不会因第一个聚类中心的产生而具有很大的随机性。

由于 $k$-means 算法需要事先设定聚类中心个数，如果任意设定该数量，那么可能得到的聚类结果不是十分精确，因此如果结合改进的最大最小距离算法，不仅可以按照需要的距离划分聚类中心，同时也解决了 $k$-means 算法的聚类结果受初始凝聚点影响很大的缺点。改进的 $k$-means 算法如下。

2) 改进的 $k$-means 算法

算法输入：待聚类的数据集 $S=\{s_1,s_2,\cdots,s_n\}$。

算法输出：带有类别标识的数据集合 Z。

算法描述：

步骤 1

使用改进的最大最小距离算法得到最初的若干个聚类中心，$C=\{c_1,c_2,\cdots,c_k\}$

步骤 2

```
    for each s_i∈ S  do
        for each c_j∈ C  do
            d_ij=|s_i-c_j|;
        end for
        if  min{d_ij}  then
          Z_j←s_i;//将s_i加入c_j为中心的Z_j分类中
```

```
        end if
    end for
```

**步骤 3**

计算每个 $Z_j$ 中新的均值作为聚类中心，重复步骤 2 和步骤 3，直到初始分类的均值没有变化。

### 3.2 数据流相似性分析

相似性分析是分析单位时间内各条数据流在各个时间段内的分组数和字节数特征。对聚类中的任意 2 条数据流，分别计算在对应时间段内的分组数之差的均值、分组数之差平方的均值、字节数之差的均值以及字节数之差平方的均值，采用欧式距离计算相似性。距离越小就说明 2 条数据流之间的相似性程度越大。相似性分析算法如下。

算法输入：

数据流聚类集合 $Z=\{Z_1,Z_2,\cdots,Z_k\}$，每个类别中的数据流集 $S=\{s_1,s_2,\cdots,s_m\}$。

算法输出：分类中的数据流相似性特征标识。

```
for each  Zi∈Z  do
  if  | Zi | <2 then
      Zi 中的数据流不具有相似性;
    break; //类别中数据流少于 2 条，则不具备流相似性判别条件
  else
    for each  sj∈ Zi do
      计算属性向量 vj(apj, sapj, acj,sacj);
      //数据流Δt 时间内分组数之差的均值、分组数之差平方的均值、字节数之差的均值和字节数之差平方的均值
      end for
      for ∀ (vi,vj) do //类别中任意 2 个数据流的属性向量
          dij=| vi- vj |;//计算属性向量间的欧式距离
      end for
      if  max{dij}≤threshold then
          Zi 中的数据流具有相似性
      else
          Zi 中的数据流没有相似性
      end if
  end if
end for
```

### 3.3 僵尸网络频道检测

通过对僵尸网络频道数据流特征的分析，本文使用流量聚类特征、相似性特征、平均分组长特征、流量高峰特征和协同流量高峰特征 5 个特征作为检测依据。聚类特征和相似性特征可以通过频道数据流分析判别。统计数据流中数据分组的平均长度，然后与给定的阈值进行对比，即可决定数据流是否满足条件，平均分组长度阈值可以通过机器学习等方法确定。流量高峰特征和协同流量高峰特征通过单位时间的数据流对比分析即可判断。检测僵尸频道时，根据 5 个流量特征计算检测特征值，当特征值达到检测阈值时，可以认为检测到了僵尸网络。下面给出检测特征值函数 $f(x)$的定义为

$$f(x)=\sum_{i=1}^{n} w_i x_i \tag{1}$$

其中，$w_i$ 为影响力系数，$0<w_i<1$，且 $\sum_{i=1}^{n} w_i = 1$，$x_i$ 表示参与检测的流量特征，当一个考察流量特征符合检测条件时，$x_i=1$，否则 $x_i=0$，$n$ 为检测过程中流量特征数量。

影响力系数表明每种流量特性在 IRC 僵尸网络频道数据流中表现的强弱和重要性。IRC 僵尸网络活动的群体性和一致性是重要行为表现，尤其是相似性特点，故设置聚类特征影响系数$w_1=0.2$、相似性特征影响系数$w_2=0.3$，而静默期的正常 IRC 频道和僵尸频道都使用 PING/PONG 分组维持通信，数据分组大小类似，只是在时间长短上有差异，因此设置平均分组长度特征影响系数$w_3=0.1$，同样，流量高峰和协同流量高峰也是群体性和一致性是重要行为表现，所以，设置流量高峰特征影响系数$w_4=0.2$、协同流量高峰特征影响系数$w_5=0.2$。通过实验数据统计分析，将僵尸网络数据流检测函数的阈值设定 0.5，当超过此阈值时可以判断检测到僵尸网络流量。阈值的选取与检测数据的数量、僵尸网络的周期性有关联，因此可以通过不断的积累和自学习更新阈值，达到更好的检测效果。

## 4 实验及结果分析

在 Windows XP 环境搭建 IRC 测试环境，包括 1 台部署 IRC 服务器、1 台部署正常 IRC 客户端、6 台部署 bot 客户端、1 台作为僵尸网络控制者的主机。在 IRC 服务器上建立多个频道。其中包括 2 个正常频道 normal1 和 normal2，normal1 内正常 IRC 客户端保持静默状态，normal2 内正常 IRC 客户端进行聊天活动，一台 IRC 客户端主机分别登录这 2 个频道；3 个僵尸网络频道 botnet1、botnet2、botnet3，

每个频道中，有 2 台包含 bot 客户端的主机，其中 botnet1 内 bot 客户端保持僵尸网络的静默状态，botnet2 内 bot 客户端由攻击期转向静默期，botnet3 内 bot 客户端接收控制命令，执行攻击。使用 wincap 技术分别采集测试环境客户端和服务器端 48 h 的数据流量，由于 normal1 频道保持静默状态，故使用了 normal2 中数据流 $s_4$ 作为正常频道数据分析，8 条数据流如表 1 所示。下面对数据流进行分析。

**表 1　数据流与频道关系**

| 编号 | 所属频道 |
|---|---|
| $s_5$,$s_6$ | botnet1 |
| $s_7$,$s_8$ | botnet2 |
| $s_1$,$s_2$ | botnet3 |
| $s_4$ | normal2 |
| $s_3$ | 控制者主机数据流 |

### 4.1　数据流聚类分析

聚类分析结果如表 2 所示。从表中可以看到，8 条数据流被分为 5 个类别，改进的 *k*-means 算法的聚类 1 中包含数据流 $s_5$ 和 $s_6$；聚类 2 包含数据流 $s_1$ 和 $s_2$ ；聚类 4 包含数据流 $s_7$ 和 $s_8$。因为它们分别处于僵尸频道 botnet1、botnet3 和 botnet2，接收控制者发送控制命令并将命令的执行结果返回给控制者，或保持僵尸网络静默状态，因此具有相同的特性，被聚类到一起。聚类 3 只有数据流 $s_3$，因为是控制者数据流，不同于其他频道中的数据；聚类 5 只有数据流 $s_4$，由于处于正常频道，与其他僵尸客户端通信内容不同。而原始 *k*-means 算法将 $s_5$、$s_7$ 和 $s_6$、$s_8$ 聚集到一类，原因是初始中心数量的随机选取产生的影响，并且 $s_7$ 和 $s_8$ 所在的频道是从攻击期进入静默期，攻击持续时间较短，流量也较小，从而影响了聚类的结果。另外，改进的 *k*-means 算法在收敛速度上也比原始方法快，尤其是在大量数据情况下，在本次实验处理时，由于数据量较小，收敛速度差别不是很大。

**表 2　数据流聚类比较分析结果**

| 聚类类别编号 | *k*-mean 算法数据流编号 | 改进的 *k*-mean 算法数据流编号 |
|---|---|---|
| class1 | $s_5$, $s_8$ | $s_5$, $s_6$ |
| class2 | $s_1$,$s_2$ | $s_1$,$s_2$ |
| class3 | $s_3$ | $s_3$ |
| class4 | $s_6$, $s_7$ | $s_7$, $s_8$ |
| class5 | $s_4$ | $s_4$ |

### 4.2　数据流相似性分析

相似性分析的结果如表 3 所示。可以看到聚类 1、2、4 中的所有数据流具有很大的相似性。产生上述结果的原因是这 3 个聚类中的数据流分别处于同一个僵尸频道。每一个僵尸频道中的所有数据流几乎会在同一时间接收到控制者的控制命令并几乎同时向控制者返回命令执行结果信息。因此，僵尸频道中的每个客户端的数据流很相似，具有相似性特征。

### 4.3　数据分组平均长度分析

数据分组平均长度分析结果如表 4 所示。数据表明：由于 $s_5$ 和 $s_6$ 为 botnet1 中的僵尸流，处于静默期，之间只有 PING/PONG 数据分组，平均分组大小最小；$s_7$ 和 $s_8$ 为 botnet2 中的僵尸流，从攻击期转向静默期，平均数据分组大小稍大；$s_4$ 为 normal1 和 normal2 的正常聊天流，平均数据分组大小较大；而 $s_3$ 为控制者流，向僵尸主机发送控制命令，平均数据分组大小较 $s_4$ 大；流 $s_1$ 和 $s_2$ 为 botnet3 中的数据流处于攻击期，不断执行控制命令并返回命令执行结果，平均数据分组大小最大。

**表 3　数据流相似性分析结果**

| 编号 | $s_1$ | $s_2$ | $s_3$ | $s_4$ | $s_5$ | $s_6$ | $s_7$ | $s_8$ |
|---|---|---|---|---|---|---|---|---|
| $s_1$ | — | 5 554.40 | 2 899 120.04 | 17 354 190.63 | 19 095 778.52 | 19 078 586.52 | 19 016 386.65 | 19 017 606.65 |
| $s_2$ | 5 554.40 | — | 2 911 296.54 | 17 915 036.63 | 19 293 992.65 | 19 309 982.65 | 192 324 068.65 | 19 232 928.65 |
| $s_3$ | 2 899 120.04 | 2 911 296.54 | — | 31 671 884.74 | 34 331 848.79 | 34 310 488.79 | 33 582 316.77 | 33 584 500.77 |
| $s_4$ | 17 354 190.63 | 17 915 036.63 | 31 671 884.74 | — | 312 306.91 | 308 995.25 | 419 095.17 | 418 329.45 |
| $s_5$ | 19 095 778.52 | 19 293 992.65 | 34 331 848.79 | 312 306.91 | — | 2 153.04 | 18 689 563.71 | 18 685 582.71 |
| $s_6$ | 19 078 586.52 | 19 309 982.65 | 34 310 488.79 | 308 995.25 | 2 153.04 | — | 18 701 250.06 | 18 710 147.12 |
| $s_7$ | 19 016 386.65 | 192 324 068.65 | 33 582 316.77 | 419 095.17 | 18 689 563.71 | 18 701 250.06 | — | 682.02 |
| $s_8$ | 19 017 606.65 | 19 232 928.65 | 33 584 500.77 | 418 329.45 | 18 685 582.71 | 18 710 147.12 | 682.02 | — |

表 4 平均分组长度分析结果

| 编号 | 平均数据分组长度 |
|---|---|
| $s_1$ | 104.53 |
| $s_2$ | 104.81 |
| $s_3$ | 95.53 |
| $s_4$ | 80.67 |
| $s_5$ | 66.89 |
| $s_6$ | 66.76 |
| $s_7$ | 77.49 |
| $s_8$ | 77.80 |

### 4.4 流量高峰和协同流量高峰分析

由于数据流 $s_1$ 与 $s_2$，$s_5$ 与 $s_6$，$s_7$ 与 $s_8$ 分别是同一僵尸频道中的数据，具有相同的行为活动，因此，分别使用其中的一个来描述各条数据流的变化情况。数据流 $s_1$ 和 $s_3$ 的数据流量如图 1 所示，由于 $s_1$ 是 botnet3 中的数据流，且处于攻击期，因此数据流量比较大；$s_3$ 由于是控制者，它控制整个僵尸网络的频道，因此数据流量最大，另外，bot 端接收攻击命令后，要对其返回结果，故流量 $s_1$ 和 $s_3$ 具有基本同步的流量高峰。数据流 $s_4$ 和 $s_5$ 的数据流量情况如图 2 所示，$s_4$ 为正常 IRC 聊天频道数据流，因此流量变化相对稳定，没有明显的流量高峰，$s_5$ 虽然是僵尸网络频道数据流，不过由于处于静默期，数据流量较少，故高峰情况不明显。图 3 中数据流 $s_7$ 由于僵尸活动是从攻击期转向静默期，在前期攻击期出现流量的高峰，不过大部分时间数据量较少。从以上分析可知，处于攻击期的僵尸网络流量高峰特征明显，如数据流 $s_3$、$s_1(s_2)$和 $s_7(s_8)$，而正常频道 $s_4$ 和静默期的僵尸网络流量 $s_5(s_6)$，这个特征不明显。

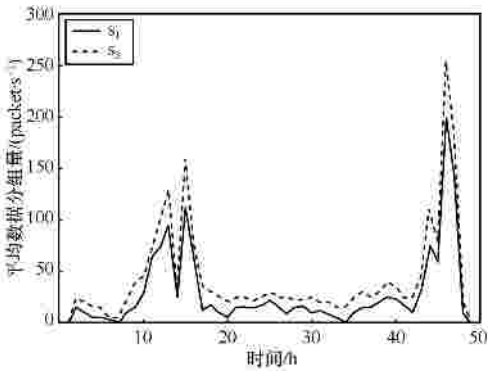


图 1 $s_1$ 和 $s_3$ 流量高峰示意

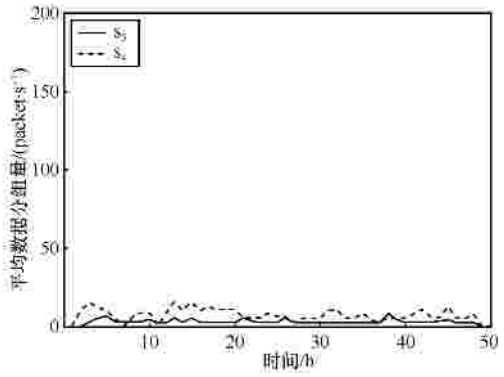


图 2 $s_4$ 和 $s_5$ 流量高峰示意

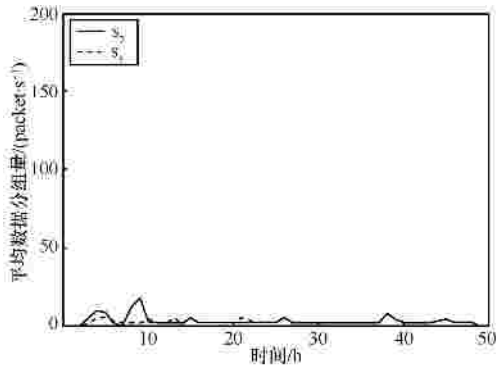


图 3 $s_5$ 和 $s_7$ 流量高峰示意

接下来，分析总数据流和各条数据流之间的协同流量高峰。由于数据流 $s_1(s_2)$是处于攻击期的 botnet3 的数据流，并且数据流 $s_3$ 为控制者数据流，它们会产生大量数据分组并明显影响总的数据分组个数，因此，攻击期间客户端和服务器之间存在协同流量高峰特征，结果如图 4 和图 5 所示。数据流 $s_7(s_8)$的前段时间处于攻击期，具有流量高峰特征，不过由于其发送数据分组较少，故与服务器间没有明显的协同流量高峰特征，结果如图 6 所示。而数据流 $s_4$ 为正常聊天数据流，$s_5(s_6)$是处于静默期的僵尸网络数据流，因此与服务器不存在协同流量高峰特征，结果如图 7 所示。

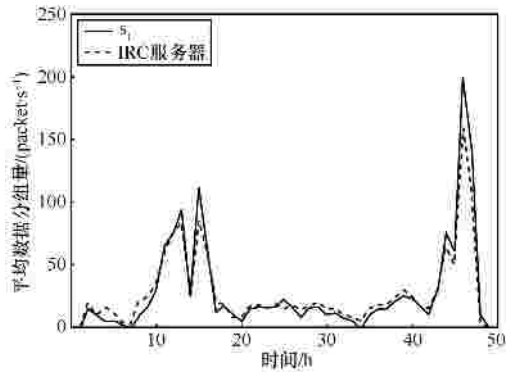


图 4 $s_1$ 和 IRC 服务器协同流量高峰示意

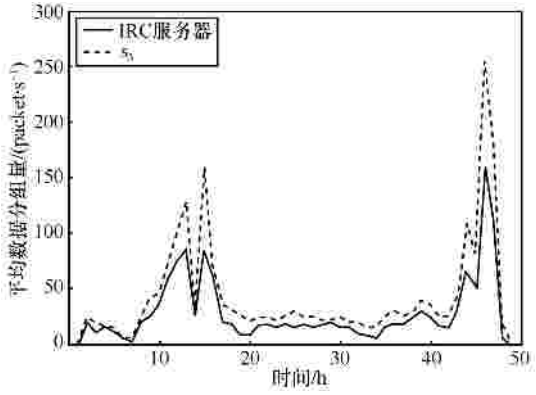


图 5　$s_3$ 和 IRC 服务器流量协同高峰示意

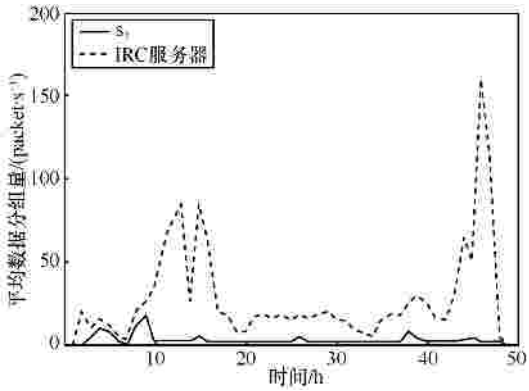


图 6　$s_7$ 和 IRC 服务器流量协同高峰示意

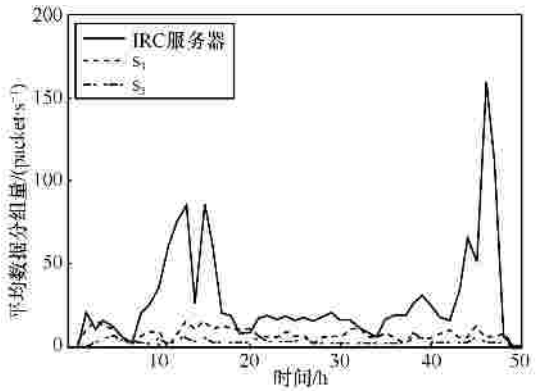


图 7　$s_4$ 与 $s_5$ 和 IRC 服务器流量协同高峰示意

综合上述实验数据结果，使用定义的僵尸网络频道检测特征值函数分别计算每个频道数据流的特征值，结果如表 5 所示。可以看到，僵尸频道中的数据流特征值都在给定阈值之上，故使用流量多特征的检测方法，可以检测出僵尸网络的数据流，从而判断僵尸网络的存在。不过，由于 IRC 僵尸频道中的 bot 是集体活动，流量特征比较明显，便于从流量的角度发现其恶意活动，但控制者由于单独存在或者利用中间跳板，其多种流量特征反映不尽相同，故检测时可能需要更多的信息。

表 5　数据流检测特征值

| 编号 | 特征值 |
|---|---|
| $s_1$ | 0.9 |
| $s_2$ | 0.9 |
| $s_3$ | 0.5 |
| $s_4$ | 0 |
| $s_5$ | 0.6 |
| $s_6$ | 0.6 |
| $s_7$ | 0.7 |
| $s_8$ | 0.7 |

## 5　结束语

本文对基于流量特征的 IRC 僵尸网络频道检测方法进行了探索和研究，分析了僵尸网络在不同活动阶段所反映的流量特点，提出了基于多维僵尸网络流量特征的 IRC 僵尸网络频道检测方法，通过实验表明，方法在检测的有效性方面是可行的。不过方法也存在不足，例如，如果僵尸网络处于非攻击活动阶段时，很难从流量特征进行检测等，因此还需要其他的检测方法辅助进行，从而更加准确地发现僵尸网络活动。

## 参考文献：

[1] 诸葛建伟，韩心慧，周勇林等. 僵尸网络研究[J]. 软件学报，2008(3): 702-715.
ZHUGE J W, HAN X H, ZHOU Y L, *et al*. Research and development of Botnets[J]. Journal of Software, 2008(3):702-715.

[2] CNCERT/CC 互联网安全威胁报告[EB/OL]. http://www.cert.org.cn/publish/main/upload/File/20121227monthly11.pdf, 2012.
CNCERT/CC Internet security threat report[EB/OL]. http://www.cert.org.cn/publish/main/upload/File/20121227monthly11.pdf, 2012.

[3] 江健，诸葛建伟，段海新等. 僵尸网络机理与防御技术[J]. 软件学报，2012, 23(1): 82-96.
JIANG J, ZHUGE J W, DUAN X H, *et al*. Research on Botnet mechanisms and defenses[J]. Journal of Software, 2012, 23(1): 82-96.

[4] FEDYNYSHYN G, CHUAH M, TAN G. Detection and classification of different Botnet C&C channels[A]. Proceedings of the 8th International Conference[C]. Banff, Canada, 2011.228-242.

[5] BINKLEY J R, SINGH S. An algorithm for anomaly-based Botnet detection[A]. Proceedings of the 2nd Workshop on Steps to Reducing Unwanted Traffic on the Internet[C]. Berkeley, CA, USA, 2006.43-48.

[6] TSAI M H, CHANG K C, LIN C C, *et al*. C&C tracer: Botnet command and control behavior tracing[A]. Proceedings of the 2011 IEEE International Conference on Systems, Man and Cybernetics (SMC)[C]. Anchorage, AK, 2011. 1859-1864.

[7] GIROIRE F, CHANDRASHEKAR J, TAFT N, *et al*. Exploiting temporal persistence to detect covert botnet channels[A]. Proceedings of the Recent Advances in Intrusion Detection[C]. Springer Berlin Heidelberg, Brittany, France, 2009. 326-345.

[8] LU C, BROOKS R. Botnet traffic detection using hidden models[A]. Proceedings of the Seventh Annual Workshop Cyber Security and Information Intelligence Research[C]. Oak Ridge, TN, USA, 2011. 31.

（下转第 64 页）