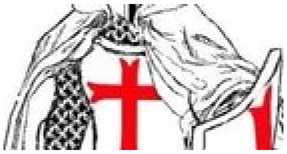
# (memo) RHOMBUS an ELF bot installer/dropper : LinuxMalware

old.reddit.com/r/LinuxMalware/comments/fh3zar/memo_rhombus_an_elf_bot_installerdropper/



(memo) RHOMBUS an ELF bot installer/dropper (self.LinuxMalware)

submitted 2 years ago * by mmd0xFF

A newly seen ELF IoT bot's Dropper/Installer, I firstly handling these hashes for intel 64bit & ARM 32bit, along with other architectures,

It's spotted under file naming of **RHOMBUS.{arch}** :

```
b982276458a85cd3dd7c8aa6cb4bbb2d4885b385053f92395a99abbfb0e43784
83e4fb6e5b042c15c035f399d286690f0382c01b43b84898564315951bb1c375
```

**Several explanation:**

Basically **this is an ELF malware installer (and dropper)**, it drops another ELF as payload & set cron as "autostart" for it. The installed payload is a bot client (embedded in the dropper). It seeks the **/tmp** directory path, extracted embedded binary data and creates file **"/tmp/fileXXXXXX"** (X= combination permutated strings), i.e: **"fileCo70r0"**, then it saves the executable code into dropped ELF bot binary.

The dropped ELF binary is the payload, a DDOS bot client, that is having basic stuff like remote execution, receiving bot commands, encrypted traffic functions, etc), see below for detail.

The interesting part for the dropper is, after dropping the payload, **it installs persistence** startup to **"/etc/cron.hourly/0"** and lastly executed the dropped binary and then cleaned-up itself. The dumped the embedded ELF is with the hash : **269029c1554b13c3eccfaacf0196ff72** (you can check this hash after you drop / extract embedded part).

**How to detect by behavior in a glance:**

The dropped binary is a bot client that will print **"IVEBEENEXECUTED"** on execution, and made below networking:

```
1. listening to (bind to 127.0.0.1) TCP/12645 < likely a command receiver port
2. callback to C2 (bind to LOCALIP:HIGHPORTS) at 209.126.69.167:2020 (IP = AS6428
River City Internet Group, Primary Networks, USA)
```

PS: The source of this infection is also from USA network: 104.244.72.54 on AS53667 at PonyNET, honeypots detected it:

```
104[.]244[.]72[.]54/RHOMBUS.sh4
104[.]244[.]72[.]54/RHOMBUS.x86
104[.]244[.]72[.]54/RHOMBUS.arm5
104[.]244[.]72[.]54/RHOMBUS.x86_64
104[.]244[.]72[.]54/RHOMBUS.mpsl
```

Interesting strings in the embedded (or) the dropped binary, aka **the payload** is:

```
0xZ6c8 48 47 %s %s HTTP/1.1\r\nHost: %s\r\nConnection: close\r\n\r\n
0x_1bd 5 4 \b\n\n\n
0x_1d0 9 8 hlLjztqZ
0x_220 21 20 npxXoudifFeEgGaACScs
0x_235 8 7 +0-#'I
0x_2a8 15 14 Unknown error
0x_2c0 8 7 Success
0x_e1d 8 7 /bin/sh
```

some_ encrypted strings are intact, you can "grep" these:

```
0x_6f8 7 6 {inod\f
0x_6ff 5 4 snnu
0x_704 10 9 0110biho\f
0x_70e 9 8 edg`tmu\f
(etc)
```

**So, What is this threat anyway?**

We were suspecting this ELF dropper is ~~a part of~~ a new ELF/linux DDoSer w/new installer ~~possibilities were varied, maybe CLOUDSNOOPER or,~~ and, well, in the end it is just a dropper to drop "another" DDoSer bots. (one of a kind of: Gafgyt, Kaiten, Mirai and such) just another new coded one.. Made by skids in DDoS ecosystem.

**How the payload works in general:**

The payload has these functions:

1. C2 command receiver
2. Execution of DoS attack variations
3. Sending data to C2 of compromised device
4. Has a remote command execution that can be used to execute downloaded file or crafted pushed command
5. Encryption to process config and receive-transmission comm.

In detail it works as follows:

After static reversing & decrypting payloads distributed via the dropper, and also decrypting/analyzing more recent samples distributed w/o dropper it shows that the connection to C2 will trigger the real activity of this bot (decryption of transmission data, processing receiving commands, and offensive execution for five DOS variation attacks of commands, including one of them are sub packeting forged **"urg", ack", rst", "fin", "psh" attacks**, other is with the **"0" or "1"** sub attack types, and there's also **L7 flood spoofing for HTTP/1.1** too).

The transmission data is using encryption (XOR'ed basis), that's also used for processing embedded & hard coded config in the bot client, and also the pushed one through the listening port.

Activities invoked are not only processing bot command but involving the **command execution triggered by "sh -c"** in the system compromised by this bot.

C2 will be sent (with write method) by encrypted data through connected socket from the bot client, contains data of **bot info and networking (IP)** with this string:

```
"jm:%s:%d OR "jm:_:%d
```

The infrastructure used by the adversaries to spread the payload and as C2 is listed so far as follows, you should block these:

```
209.126.69.167  | I167.datasoft.ws. |6428 | 209.126.64.0/20 | CDM | US
45.135.134.132  |  |51659 | 45.135.134.0/23 | ASBAXET | RU
167.172.128.4   |  |14061 | 167.172.128.0/20 | DIGITALOCEAN-ASN | US
205.185.122.243 | google-public-dns-a.google.com.(fake) |53667 | 205.185.112.0/20 |
PONYNET | US
```

Still, I haven't got enough time to check on this thoroughly, currently busy w/other works too, I am sure I am still missing one or two, so please add or comment, I will update the info regularly.

Let's call this new threat as "RHOMBUS". The OpenIOC is here.

1st found credit: 0xrb (thank you).