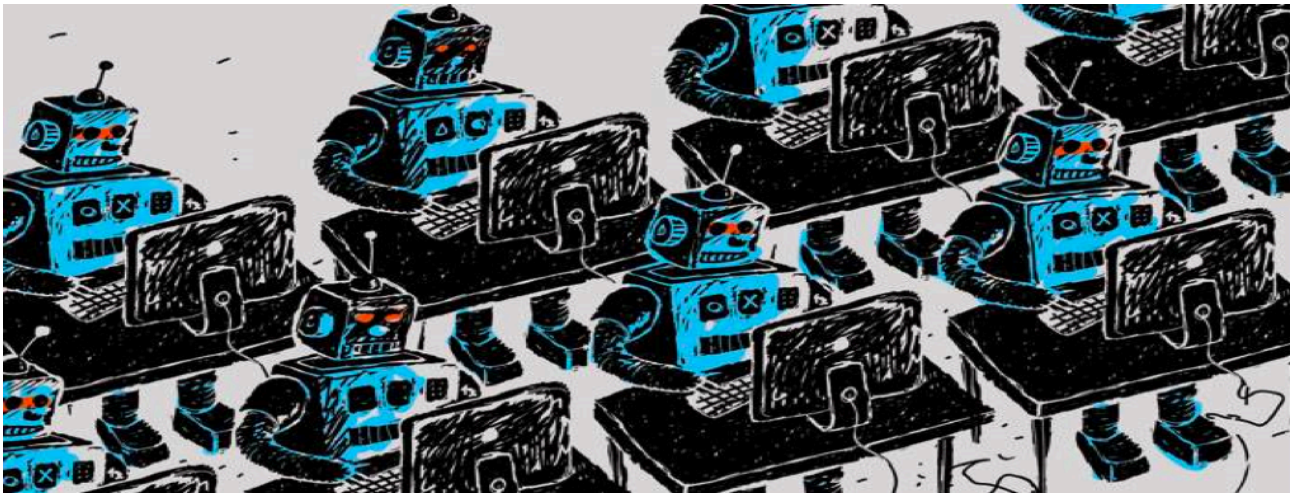
Masuta : Satori Creators' Second Botnet Weaponizes A New Router Exploit. - New Sky Security

Published: 2018-01-23 · Archived: 2026-04-02 10:45:24 UTC



Introduction

Since the inception of the Mirai code leak, many botnets have been seen in the IoT threat landscape. While some of them are clearly Mirai carbon copies, others have added new attack methods, often taking the route of exploits to perform an attack. We analyzed two variants of an IoT botnet named "Masuta" where we observed the involvement of a well-known IoT threat actor and discovered a router exploit being weaponized for the first time in a botnet campaign.

Masuta Code Leak & Attribution

We were able to get hands on the source code of Masuta (Japanese for "master") botnet in an invite only dark forum. After analyzing the configuration file., we saw that Masuta uses 0xdedeffba instead of Mirai's 0xdeadbeef as the seed of the cipher key, hence the strings in the configuration files were effectively xored by ((DE^DE)^FF) ^BA or 0x45.

```
uint32_t table_key = 0xdedeffba;
struct table_value table[TABLE_MAX_KEYS];
```

Now xoring the configuration file with 0x45, we get the domain *nexusiotsolutions(dot)net* which is a known C2 URL of Nexus Zeta involved with recent Satori attacks, where a Huawei router zero day was used.

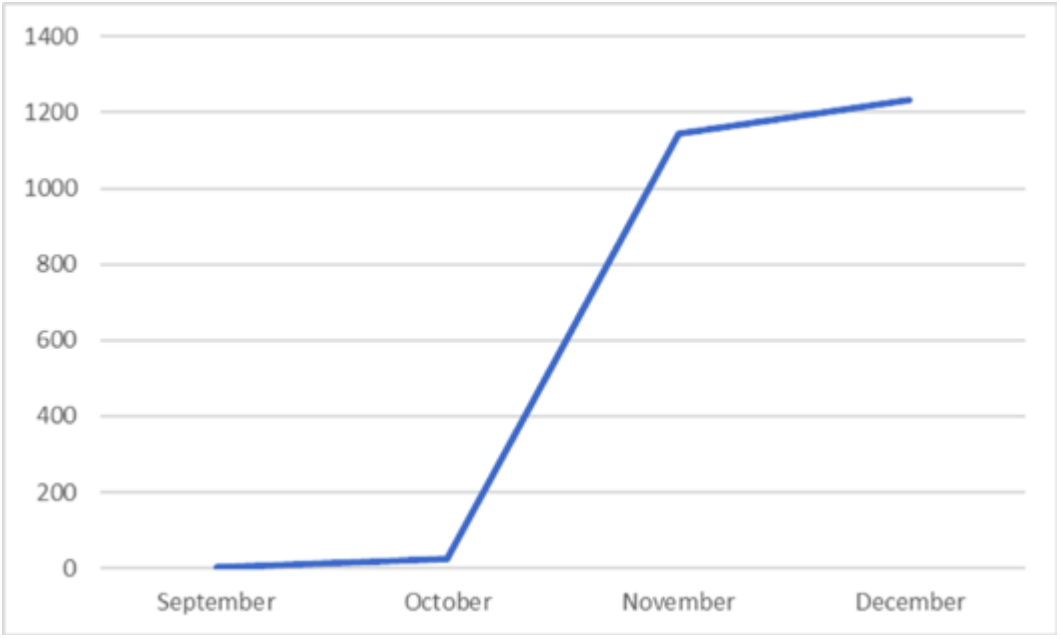
```
add_entry(TABLE_SCAN_CB_DOMAIN, "\x2B\x20\x3D\x30\x36\x2C\x2A\x31\x36\x2A\x29\x30\x31\x2C\x2A\x2B\x36\x6B\x2B\x20\x31\x45", 22);
```

The WHOIS information for the URL also states contact as nexuszeta1337@gmail(.)com, indicating that Nexus Zeta is not a one hit wonder creator of Satori, but also has been involved in the creation of the Masuta botnet.

The standard Masuta variant used several known/weak/default credentials to get access to the IoT device it attacked.

```
// Set up passwords
add_auth_entry("\x37\x2A\x2A\x31", "", 5);      //root
add_auth_entry("\x24\x21\x28\x2C\x2B", "\x24\x21\x28\x2C\x2B", 10);  //admin admin
add_auth_entry("\x24\x21\x28\x2C\x2B", "\x74\x77\x76\x71", 9); // admin 1234
```

The Masuta attacks (defined by the recon indicator /bin/busybox MASUTA) have been on the rise since September as honeypots observed 2400 IPs involved in the botnet in last three months. The rising trend is shown in the graph below.



One of the prominent command and control servers involved in Masuta attacks is n(.)cf0(.)pw or 93.174.93.63.

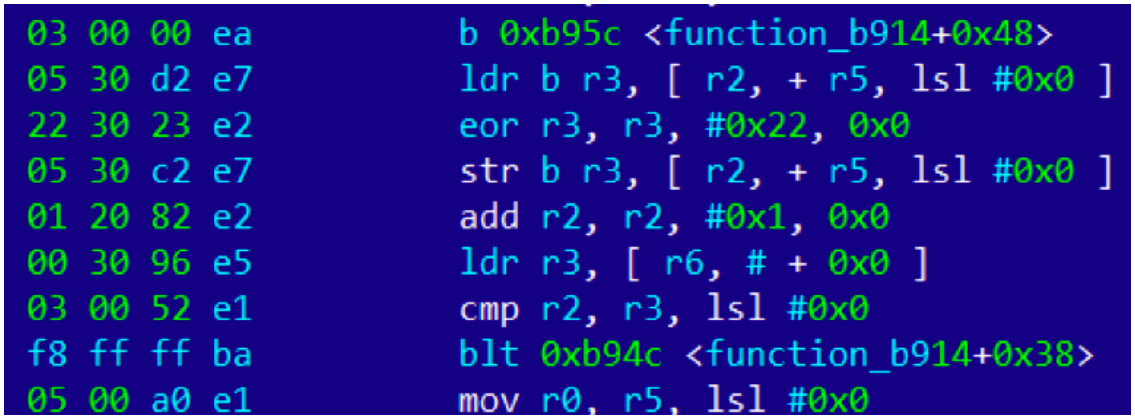
PureMasuta Variant & Exploit Usage

This IP address *93.17.93.63* gave us a way inside another evolved variant of the Masuta botnet. Although we did not obtain the source code of this variant in Blackhat forums, on analyzing the compiled ARM binary it was clear that this was not just a usual Masuta sample.

The Masuta variant (dubbed as PureMasuta) contains the most typical of Mirai style code, with a weak credential list (PMMV = “root”, TKXZT = “vizxv”, CFOKL = “admin”).

```
function_b974((int32_t)"PMMV", (int32_t)&g13, 10);
function_b974((int32_t)"PMMV", (int32_t)"TKXZT", 9);
function_b974((int32_t)"PMMV", (int32_t)"CFOKL", 8);
```

The credentials are hidden by a single byte XOR by 0x22 as shown in the figure below, another inspiration from the Mirai leak.



However, what makes PureMasuta stand out of common Mirai/Masuta is the usage of EDB 38722 D-Link exploit.

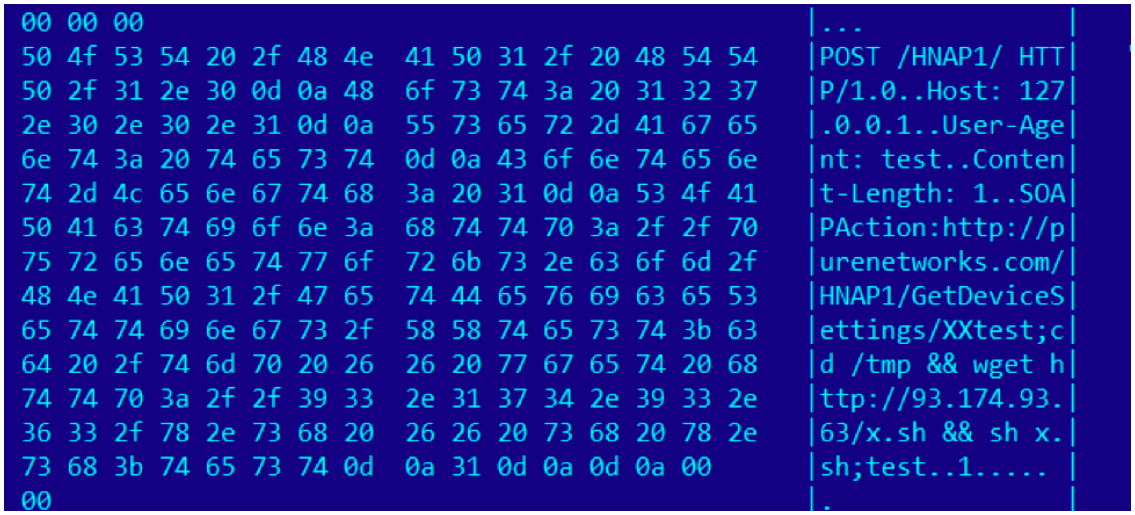
Explaining the EDB 38722 D-Link HNAP Bug

The weaponized bug introduced in PureMasuta botnet is in the HNAP (Home Network Administration Protocol) which itself is based on the SOAP protocol. It is possible to craft a SOAP query which can bypass authentication by using *hxxp://purenetworks.com/HNAP1/GetDeviceSettings.* Also, it is feasible to run system commands (leading to arbitrary code execution) because of improper string handling. When both issues are combined, one can form a SOAP request which first bypasses authentication, and then causes arbitrary code execution. For example, the string below will cause a reboot.

SOAPAction*: "hxxp://purenetworks.com/HNAP1/GetDeviceSettings/`reboot`"*

Hence in simple words, whatever code is written after GetDeviceSettings will be executed.

Instead of the reboot, the PureMasuta botnet downloads a shell script from a command and control server (via wget) and runs it. Following image shows the script in action in the botnet binary.



We noticed that the command and control server (93.174.93.63) is same as used in the original Masuta variants, hence indicating that PureMasuta is an evolved creation of the same Masuta threat actors.

The proof of concept of the exploit is available for public in places like exploit-db and pastebin . Hence, we can assume it will not be very difficult for an attacker to implement the exploit.

Obsession with Brian Krebs

Many IoT botnets mention Brian Krebs, a known journalist who was instrumental in the investigation behind Mirai as an Easter egg. Masuta is no exception as we saw the following message in the source code:

```
this.conn.Write([]byte("\033[37;1m[\033[91m!\033[37m] MASUTA is powered and hosted on Brian Krebs' 4Head!\
for {
    var botCatagory string
    var botCount int
    this.conn.Write([]byte("\033[31;1m" + username + "\033[33m@\033[31mmasuta\033[33m $ \033[0m"))
```

This same message was tweeted out by an unverified twitter account of Nexus Zeta, connecting the dots with his association with the Masuta botnet.

Conclusion

Nexus Zeta is no stranger when it comes to implementing SOAP related exploits. The threat actor has already been observed in implementing two other known SOAP related exploits, CVE-2014–8361 and CVE-2017–17215 in his Satori botnet project. A third SOAP exploit, TR-069 bug has also been observed previously in IoT botnets. This makes EDB 38722 the fourth SOAP related exploit which is discovered in the wild by IoT botnets.

Protocol exploits are more desirable for threat actors as they usually have a wider scope. A protocol can be implemented by various vendors/models and a bug in the protocol itself can get carried on to a wider range of devices. NewSky Security IoT Halo detects all the four SOAP exploits mentioned in this blog.

> *Ankit Anubhav, Principal Researcher, NewSky Security (NewSky Security)*
