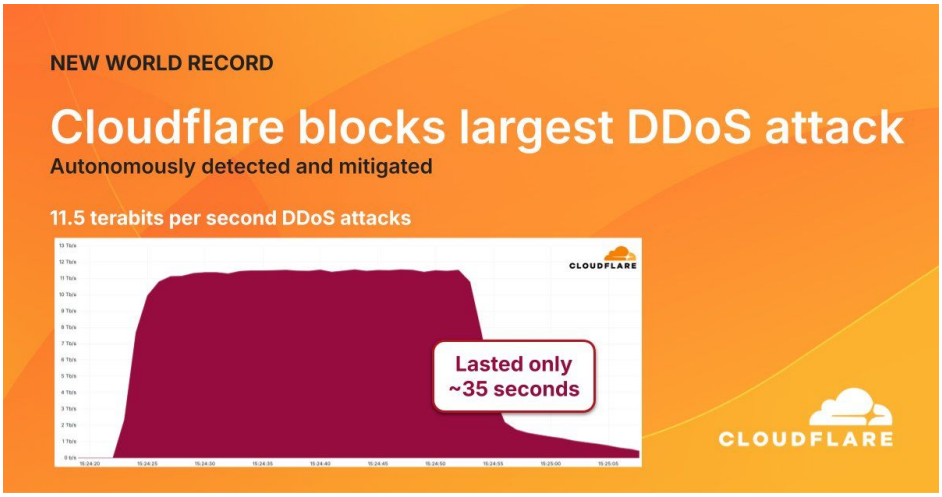
blog.xlab.qianxin.com /super-large-scale-botnet-aisuru/

The strongest in history? Uncovering the inside story of the 11.5T-class hyperscale botnet AISURU

Alex.Turing ⋮ ⋮ 9/14/2025

Overview

Since 2025, the bandwidth peak of global DDoS attacks has continued to set a record, from 3.12 Tbps at the beginning of the year to a staggering 11.5 Tbps in recent days. In a number of high-impact or record-breaking attacks, we have all monitored a botnet called AISURU's frequent activity behind the scenes.



Cloudflare reports 11.5T attack

2025-09-02 07:34:52	coerece.ilovegaysex.su	81.19.140.41	8443	Flood_Attack_0	185.211.78.117
2025-09-02 07:34:52	lane.ilovegaysex.su	88.151.192.129	8443	Flood_Attack_0	185.211.78.117
2025-09-02 07:34:51	lane.ilovegaysex.su	45.138.16.202	9034	Flood_Attack_0	185.211.78.117
2025-09-02 07:34:51	coerece.ilovegaysex.su	5.181.3.41	9034	Flood_Attack_0	185.211.78.117
2025-09-02 07:34:50	approach.ilovegaysex.su	45.80.158.129	9034	Flood_Attack_0	185.211.78.117

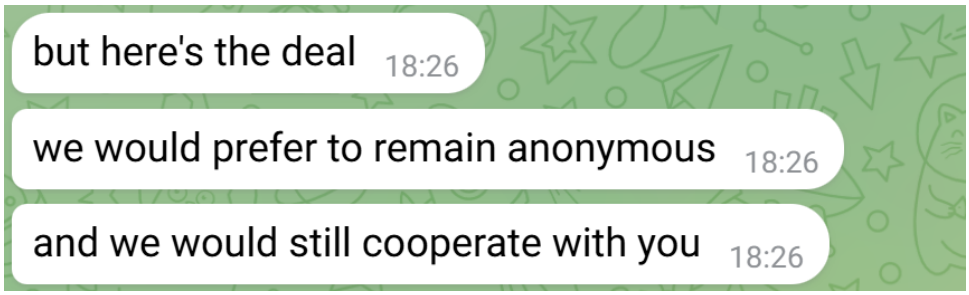
XLAB attack event monitoring data

The AISURU botnet, first revealed by XLab in August 2024, was involved in a DDoS attack on the Black Myth: Goku distribution platform. Since March, the XLab threat monitoring platform has continuously captured new samples of the botnet. According to multi-party information, the gang behind it was suspected of invading a brand router firmware upgrade server in April, and further expanded the scale of the botnet by issuing malicious scripts, and the current number of nodes is said to have reached 300,000.

What is even more alarming is that the "egg" information embedded in some of the AISURU samples has clearly gone beyond the pure intent of attack and instead attempted to convey specific ideological content. Based on this grim situation, we have decided to write this report to open the relevant research results to the safe community, and call on all parties to work together to combat this increasingly rampant cybercrime.

Anonymous sources & XLab Vision

As a result of XLab's long-standing deep-rooted DDoS attacks and continued to publish reliable and in-depth analysis reports, we have gained a strong reputation not only among defenders but also within the attacker community. Recently, for the AISURU/AIRASHI botnet, an informed anonymous source has taken the initiative to provide us with relevant information, hoping to completely disintegrate AISUU like the previous crackdown on the Fodcha botnet. This clue gives us the final opportunity to get closer to the gang behind AISURU and uncover the inside story of the botnet's operations.



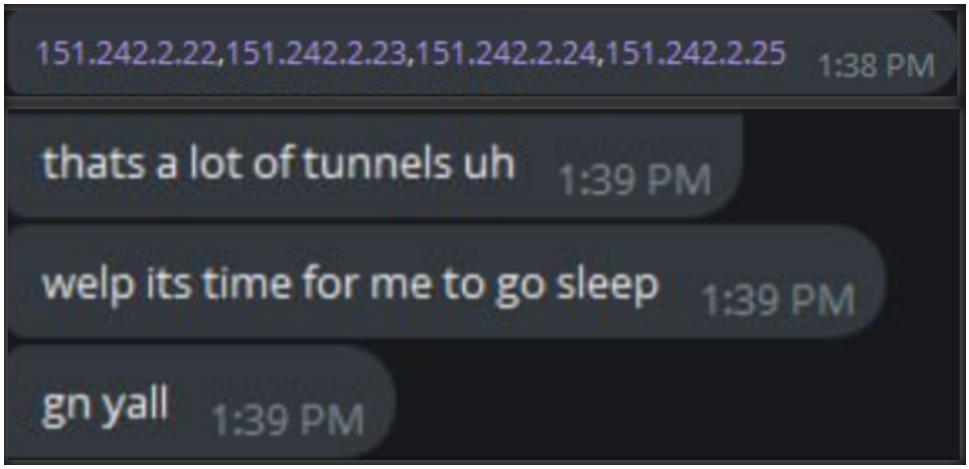
Anonymous sources

According to anonymous sources, the AISURU gang has three key characters, codenamed Snow, Tom, Forky. In 2022, Forky met with Snow and Tom, who was in the middle of the fashion, after several happy collaborations such as the catddos botnet, and decided to set up the current AISURU team.

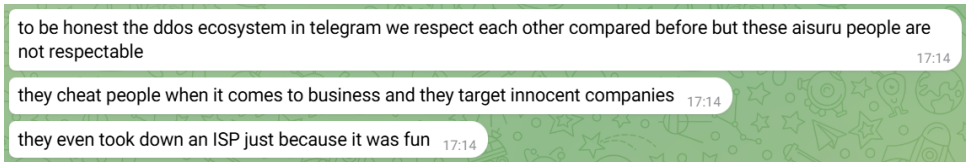
- Snow: responsible for the development of botnets
- Tom: Responsible for vulnerabilities, including 0-day discovery, Nday integration
- Forky: responsible for the sale of botnets

In April 2025, Tom successfully broke into one of Totolink's routing upgrade servers, setting the URL of the firmware upgrade to download and execute malicious scripts. This means that every totolink router that performs an upgrade operation is likely to infect the AISURU botnet.

The intrusion allowed the AISURU botnet to climb rapidly, breaking through 100,000 in a very short period of time. Such a huge scale, let them also a little caught off guard, had to sacrifice sleep time, overtime on several C2 IP configuration strategy, with GRE TUNNEL for shunting.



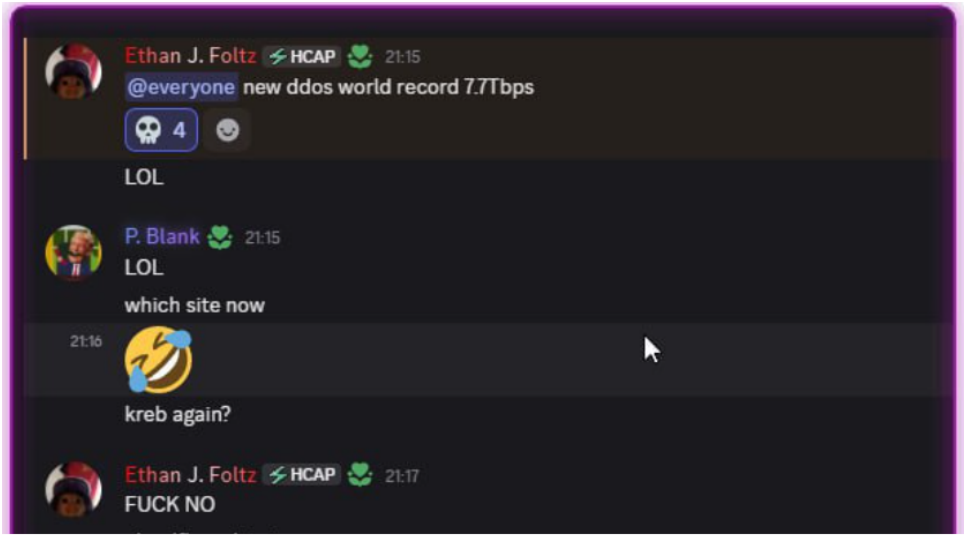
Members of the AISURU gang are openly aggressive and often attack ISPs on the grounds of "fun" and are very destructive. This makes them very bad reputation in the DDoS circle, often jokingly called "mental abnormal" by others, which can be said to be countless enemies.



At the end of April, DDoS insiders decided to give the AISURU gang a little color and began to break all kinds of news on social media. First, in a tweet that Cloudlare said to ease the record-breaking 5.8Tbps, he replied: “This is an attack from the 340k totolink route!”; and a few days later exposed more heavy-weight evidence: a background screenshot of the botnet. From the statistics, the total number of bots online at that time exceeded 300,000, of which 30,000 were from China. While chanting “welcome to totolink botnet”, he **@totolink and Interpol** hope to attract the attention of the public, law enforcement agencies, to achieve the intent of the attack on AISURU.

At present, the vulnerability of Totolink upgrade server has been patched, and the AIRUSU gang also humorously said.`RIP TOTOLINK 2025-2025`But in fact, the scale of the AIRUSU botnet has not been affected, and it remains at about 300,000.

Before the record-breaking 12.1 Tbps in September 2025, Aisuru conducted several attack tests, including an attack on the personal website of prominent journalist Brian Krebs, which set a “world record” for the attack traffic at the time.

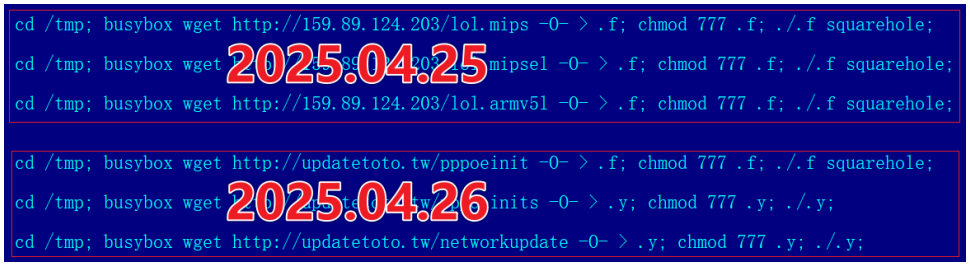


Interestingly, "Ethan J Foltz" is the real name of the author of Rapper Botnet, who was arrested on August 6, 2025; the person behind the ID "Ethan J Foltz" above is actually Snow, who uses this method to nakedly ridicule Rapper, which may be one of the reasons why the AISURU gang is shouting in the DDoS circle.

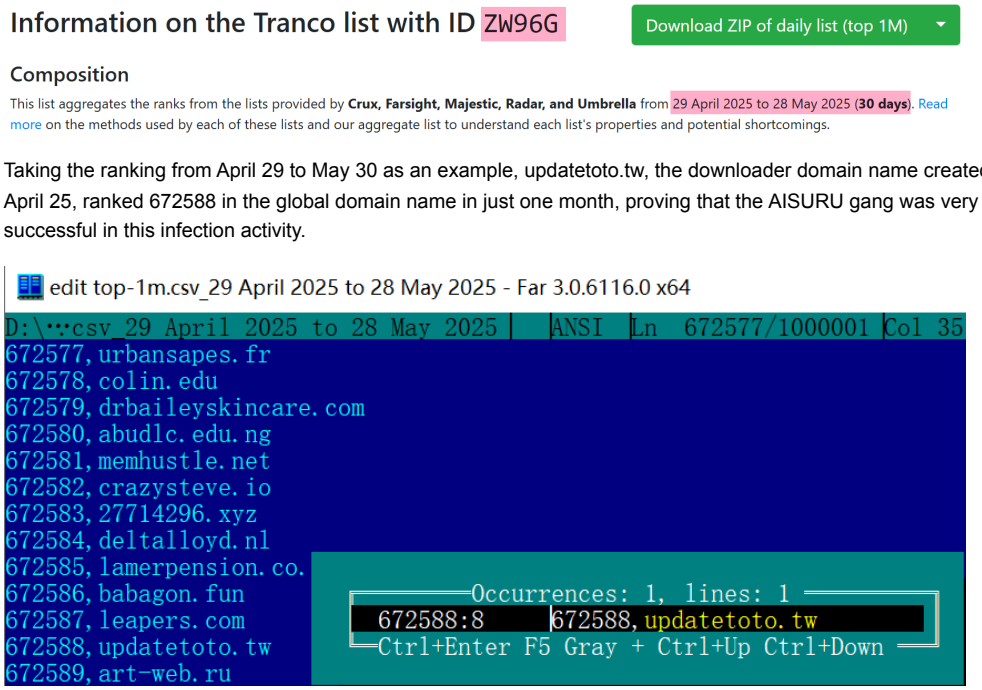
XLab Vision

For stories provided by anonymous sources, readers will certainly have similar ideas:"这的确是很有趣的瓜，可你这瓜保熟不？"We may not be able to verify these people, but relying on the powerful monitoring capabilities of the XLab threat awareness system, we样本，C2，攻击事件All have a good view. Taking the group's key activities as a clue, through data cross-checking, **we believe that the intelligence provided by the anonymous source of attacks has a high degree of credibility**.

1: Upgraded totolink malicious script t.sh for server implantation in April 2025



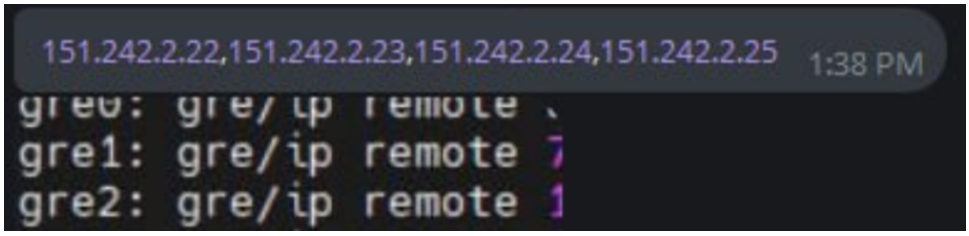
From the 26th, the script starts using a domain name updatetoto.tw, which can be used.Domain Name Ranking System TrancoMeasure its active procedures.



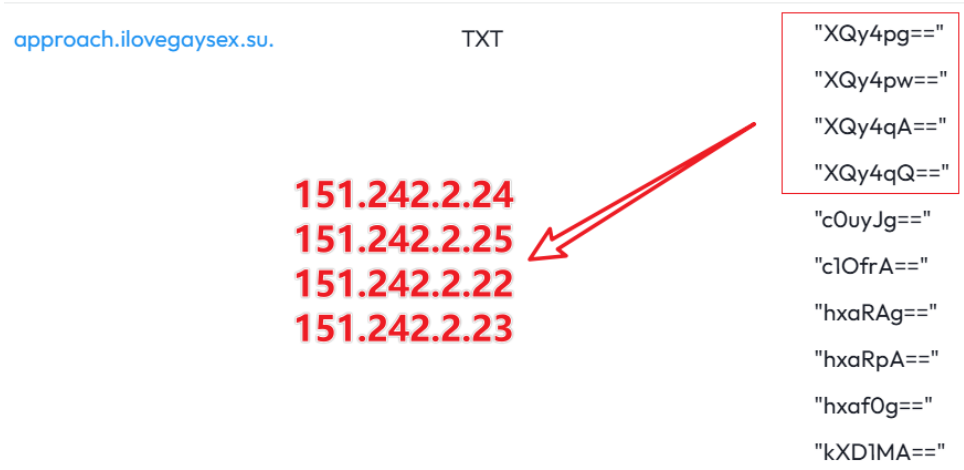
Taking the ranking from April 29 to May 30 as an example, updatetoto.tw, the downloader domain name created on April 25, ranked 672588 in the global domain name in just one month, proving that the AISURU gang was very successful in this infection activity.

2: GRE TUNNEL's C2 IP opens in April 2025

Aisuru gang in 151.242.2.[] 22 - 25] These four IPs configure the GRE Tunnel, which are actually C2 servers.



And we captured C2 in April `approach.ilovegaysex[.]su`The TXT record was declassified to cover these four IPs, indicating that the C2 belonged to the Aisuru gang.



May 2025 attack on KrebsOnSecurity

The C2 server associated with the malicious domain name ilovegaysex was followed and a cyberattack was detected in May of this year against its personal blog, Brian Krebs, a cybersecurity investigative journalist.

2025-05-13 02:15:13	coerece.ilovegaysex.su	151.242.2.23	8443	Flood_Attack_9	krebsonsecurity.com
2025-05-13 02:15:11	approach.ilovegaysex.su	185.173.36.137	8443	Flood_Attack_9	krebsonsecurity.com
2025-05-13 02:15:10	lane.ilovegaysex.su	151.242.2.24	8443	Flood_Attack_9	krebsonsecurity.com
2025-05-13 02:13:48	lane.ilovegaysex.su	151.242.2.24	8443	Flood_Attack_0	krebsonsecurity.com
2025-05-13 02:13:46	coerece.ilovegaysex.su	151.242.2.23	8443	Flood_Attack_0	krebsonsecurity.com
2025-05-13 02:13:46	approach.ilovegaysex.su	185.173.36.137	8443	Flood_Attack_0	krebsonsecurity.com

4: September 2025 attack on 185.211.78.117

The C2 server associated with the malicious domain ilovegaysex was followed, and a cyberattack on 185.211.78.117 was detected in September this year, with a staggering 11.5 Tbps of traffic.

2025-09-02 07:34:52	coerece.ilovegaysex.su	81.19.140.41	8443	Flood_Attack_0	185.211.78.117
2025-09-02 07:34:52	lane.ilovegaysex.su	88.151.192.129	8443	Flood_Attack_0	185.211.78.117
2025-09-02 07:34:51	lane.ilovegaysex.su	45.138.16.202	9034	Flood_Attack_0	185.211.78.117
2025-09-02 07:34:51	coerece.ilovegaysex.su	5.181.3.41	9034	Flood_Attack_0	185.211.78.117
2025-09-02 07:34:50	approach.ilovegaysex.su	45.80.158.129	9034	Flood_Attack_0	185.211.78.117

Sample propagation

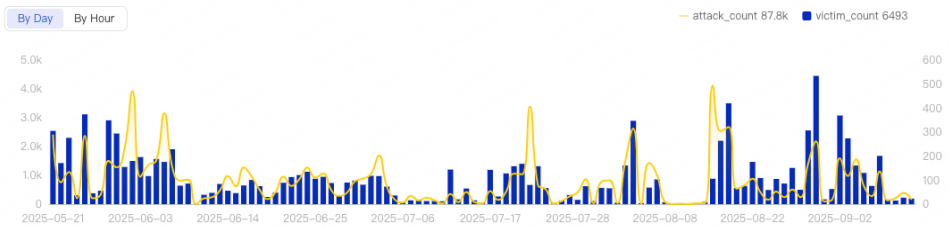
Relying on the capabilities of the XLab threat perception system, we observed that the Aisuru sample was recently propagated primarily through the NDAY vulnerability, with the ability to exploit the 0DAY vulnerability. The U.S.-based Cambium Networks' cnPilot router 0DAY, which began to be used in June last year, is still being used. Some of the vulnerabilities used in Aisuru propagation samples are as follows:

Vulnerability	Affected Vendor	Affected Devices
AMTK-CAMERA-CMD-RCE	A-MTK	Camera
CVE-2013-1599	D-Link	DCS-3411 Firmware
CVE-2013-337	Linksys	Linksys X3000
CVE-2013-5948	T - Mobile	Tm-Ac1900
CVE-2017-5259	Cambiumnetworks	Cnpilot R190V Firmware
CVE-2022-44149	Nexxt	Router
CVE-2023-28771	Zyxel, Zyxel, Zyxel, Zyxel	Zyxel ATP, Zyxel USG FLEX, Zyxel VPN, Zyxel ZyWALL/USG
CVE-2023-50381	Realtek	rtl819x Jungle SDK v3.4.11
LILIN-DVR-RCE	LILIN	DVR
CVE-2022-35733	UNIMO	DVR UDR-JA1004/JA1008/JA101
CVE-2024-3721	TBK	DVR
CNPILOT - 0 Day -RCE	Cambium Networks	cnPilot
SanhUI-GATEWAY-DEBUG-PHP-RCE	Sanhui	Gateway Management Software
TVT-OEM-API-RCE	Shenzhen TVT	DVR

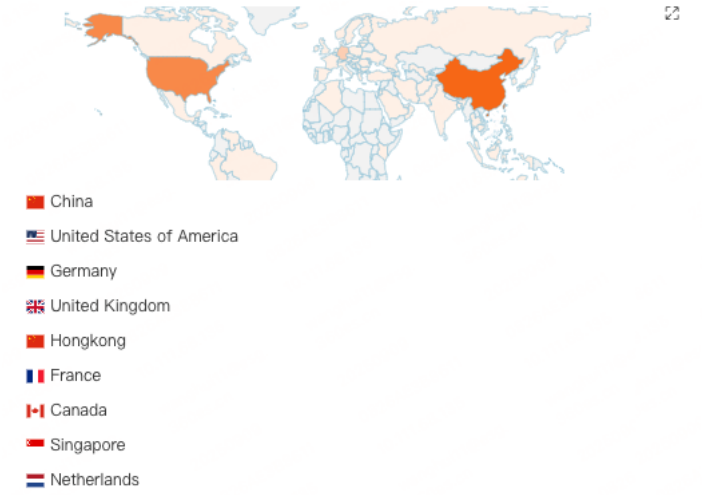
Attack statistics

The Aisuru botnet targets all over the world, distributed in various industries, and the main targets are in China, the United States, Germany, the United Kingdom, Hong Kong and other regions. There is no obvious strong targeting. There are about a few hundred targets per day.

Trends in DDoS attacks:

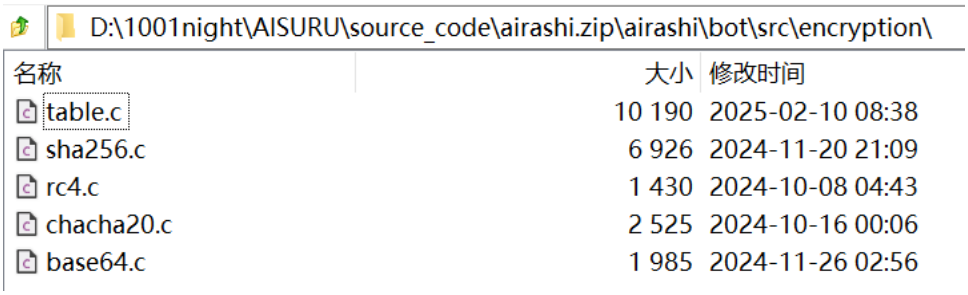


Distribution of victims:



Technical analysis

From March 14, 2025, the AIRURU gang began to deliver new botnet samples, compared with the current source code, we found that the update mainly focused on encryption, as of now found that the update can be divided into two large versions.



1. Updates to version 1 include: using echh-P256 for key exchange, after generating a shared chacha20 key to encrypt network communication messages; DNS-TXT records no longer using base64+cha20 decryption, using base64+xor decryption; new attack directives, message formats

2. Version 2 updates include: streamlining the network protocol, deleting the echh-P256 key exchange process,; magic xxhash algorithm used to verify message integrity; magic RC4 algorithm for decrypting sample strings and communication keys;

The first version lasted only about half a month, and the second version sample was used in the following. The following is a sample of version 2 as the main analysis object, focusing on Aisuru's countermeasures, encryption algorithms, and network protocols.

Environmental testing

When the sample starts, it detects whether the following string is contained in the current process command line:

```
tcpdump
wireshark
tshark
dumpcap
```

Check if the kernel's hardware identity information contains the following string:

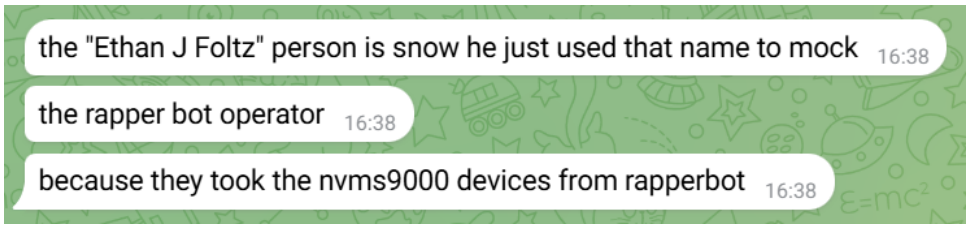
```
VMware
VirtualBox
KVM
Microsoft
QEMU
```

If the above is checked, the program exits to some extent interfere with the dynamic analysis of the sample.

The Killer Confronts

The Linux kernel has an OOM Killer (Out-Of-Memory Killer) that, when the system is running low, selects some process forced to end to release memory. The sample passed in`/proc/self/oom_score_adj`Written`-1000`Disable this function to get more execution time.

The so-called peers are wrong, each Botnet operator wants to monopolize the equipment, the competition for equipment is very fierce, a device today belongs to A, tomorrow by B invasion of the situation is not uncommon. For example, Aisuru and Rapperbot in the nvms9000 equipment competition is very white-hot, when Aisuru as a victory side to take over the equipment, they can not help but jump out to ridicule Rapperbot, close the face.



Most botnet samples compile samples using static links for multi-platform compatibility, causing them to not use any shared libraries; in addition, they delete their files after running. But this also makes many botnets use the above as a feature.`kill`to defeat their competitors.

In order to fight the above killer, the sample will start.`/lib/`Search in`.so`The shared library file at the end is mapped to the current process; no files are deleted and the filename is replaced`libcow.so`The process name is also the key object that is checked, and the sample replaces the process name with one of the following common process names:

```
telnetd
udhcpc
inetd
ntpclient
watchdog
klogd
upnpd
dhclient
```

The RC4 encryption algorithm

Compared with the previous AIRASHI version, the standard RC4 algorithm is no longer used when the new sample decrypts the string, and the standard HMAC-SHA256 algorithm is no longer used when verifying the message.

The new sample uses the magic-modified RC4 algorithm, and the key is`PJbiNbbeasddDfsc`The key has not changed in several versions, perhaps in tribute to the Fodcha botnet. The algorithm retains the RC4's 256-byte S-box, adding new perturbations when initializing and generating key streams, and the equivalent Golang implementations are as follows:

```
func AIRASHI_RC4(data []byte) []byte {
        key := make([]uint32, 4)
        keyBytes := []byte("PJbiNbbeasddDfsc")
        for i := 0; i < 4; i++ {
                key[i] = binary.BigEndian.Uint32(keyBytes[i*4 : (i+1)*4])
        }

        S := make([]byte, 256)
        i := 13
        for j := 0; j < 256; j++ {
                S[j] = byte(i & 0xff)
                i -= 89
        }

        j := 0
        for i := 0; i < 256; i++ {
                j = (j + int(S[i]) + int(key[i%4]>>(i%32))) % 256
                S[i], S[j] = S[j], S[i]
```

```
        }

        seed := uint32(0xE0A4CBD6)
        for i := 0; i < 5; i++ {
                for k := 0; k < 256; k++ {
                        seed = 0x41C64E6D*seed + 12345
                        t := (seed * uint32(S[k])) >> 24
                        t1 := (seed ^ key[(i+k)%4] ^ uint32(S[k])) & 0xff
                        S[k] = byte(t1)
                        j = (int(t1) + j + int(t)) & 0xff
                        S[k] = S[j]
                        S[j] = byte(t1)
                }
        }

        i, j, k := 0, 0, 0
        m := uint32(1)
        result := make([]byte, 0, len(data))
        for _, byteVal := range data {
                i = (i + 1) % 256
                j = (j + int(S[i])) % 256
                k = (k + int(S[(i+j)%256])) % 256
                S[i], S[j] = S[j], S[i]
                m = rol32(m, 1)
                if (m & 1) != 0 {
                        m ^= 0xD800A4
                }
                t := (S[(k+j)%256] + S[(j+i)%256]) & 0xff
                t1 := ((byte(m) ^ S[t]) >> 4) ^ rol8(byte(m)^S[t], 3)&0xff
                result = append(result, byteVal^t1)
        }
        return result
}
```

An example of the ciphertext in the following diagram:

```
00000000  09 a5 44 1f 2d d7 55 12 42 b0 0a 0f e8 00 10 a8  |.¥D.-×U.B°..è..¨|
00000010  6f 6f 83 90 c9 83 df 8b af 49 5d 9d ac 7c 7c 7f  |oo..É.ß.¯I].¬||.|
00000020  86 ad 14 99 84 d2 c2 28 64 7e 99 06 77 03 b0 69  |.....ÒÂ(d~..w.°i|
00000030  58 42 5a 5a a7 5a 69 e0 62 e2 9f de 5e 35 38 b0  |XBZZ§Ziàbâ.Þ^58°|
00000040  69 1c 7a 48 d0 d3 2e 75 f9 a7 bf bf 53 0a e4 81  |i.zHÐÓ.uù§¿¿S.ä.|
00000050  e3 b7 aa af 97 63 b1 ff e0 a9 2f 38 55 7e 78 91  |ã·ª¯.c±ÿà©/8U~x.|
```

Using Airashi_RC4 to decrypt, the plaintext we get is a provocative message. In this regard, we just want to respond to a sentence: "Your excellency is itchy?"

> tHiS mOnTh At qiAnXin shitlab a NeW aisurU vErSiOn hIt oUr bOtMoN sYsTeM dOiNg tHe CHAaCha sLiDe
> Translated into Chinese: This month at Chisin's shitlab, a new Aisuru version appears in our BotMon system, is dancing ChaCha.

Of course, AISURU hides much more information in the sample than this one. Interested readers can perform their own decryption analysis of the sample (MD5: 053a0abe0600d16a91b822eb538987bca3f3ab55). Once successfully declassified, you will understand why we are determined to resolutely combat this cyber-attack group.

C2 access

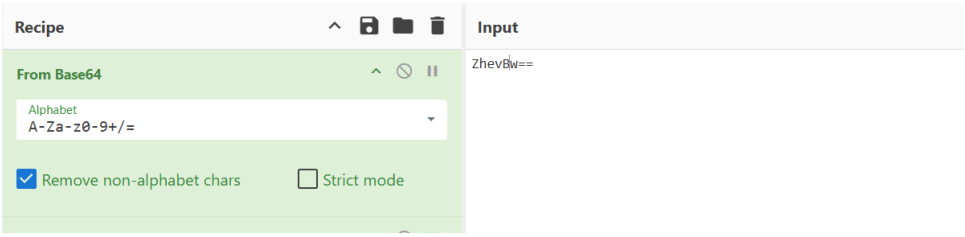
The sample continues to be maintained by the previous C2 decryption method. | Split the C2 string decrypted from the string table, get multiple subdomains and main domain names, and then pass , Split multiple subdomains, as follows:

```
decrypted str: sub1,sub2,sub3|domain.tld

c2_1: sub1.domain.tld
```

```
c2_2: sub2.domain.tld
c2_3: sub3.domain.tld
```

When resolving domain names, encrypted TXT records are still used, and base64+ChaCha20 is used in samples from previous blogs to decrypt them, and the new version simply deprecates ChaCha20, switching to different or acquiring IP. For those interested in C2 decryption, see CyberChef in the Appendix chapter, which only needs to copy C2's TXT records to INPUT.



Speed test

In the latest versions, the developer added the network upload speed test function, which is used.`speedtest`Public services

1. GET /speedtest-servers-static.php Get the test server
2. GET /speedtest/latency.txt Gets the server with the lowest latency
3. POST random data for the lowest latency server for 10s (100ms for partial samples)

However, this feature does not affect the operation of the program and the C2 connection, but only reports to C2 after the results are obtained. We believe that the purpose of this new function of speed measurement is to serve subsequent proxy instructions, and it is clear that C2 will issue proxy instructions to some nodes of the network, so that it will become a part of the residential agent.

Network protocol

Compared with the previous version, the overall process has not changed much compared with the previous version, and the ChaCha20 password and confirmation mechanism are still retained for the shared ChaCha20, but only in terms of message format and instructions and encryption algorithms.

The new message consists of three parts: the message header, the random byte, and the message body, as shown in the figure, the on-line package after decryption:

The message header is fixed to 8 bytes and consists of 4 fields:

msgType (1 byte) + randSize(1byte) + bodySize(2byte) + bodyHash (4 byte)

New fields in the line package:

```
struct login{
        uint32 stun_ip;
        uint32 botid_len;
        char botid[botid_len];
        uint32 version;
        uint32 nodename_len;
        char nodename[nodename_len];
```

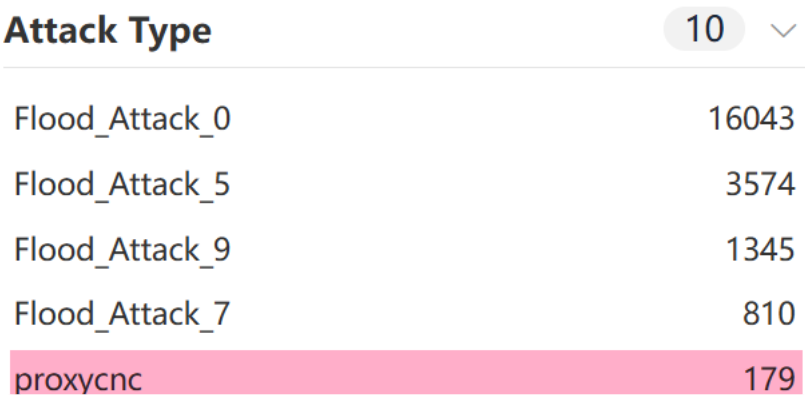
```
        uint32 cwd_len;
        char cwd[cwd_len];
        uint32 kernel_ver_len;
        char kernel_ver[kernel_ver_len];
        uint16 reserve1;
        uint8 reserve2;
        bool support_udp;
}
```

The new version supports the instructions and corresponding functions described as follows:

msgType	DESC
0	get shared net key
1	The key info
2	Confirm key
3	The login info
4	The heartbeat
5	exit
6	Attack
7	execute cmd
8	New Cnc
9	reverse shell
10	Proxy
101	report telnet scan
201	Report Killer
202	Report Netspeed

It can be seen that the new sample not only supports DDoS attacks, but also supports proxy. As law enforcement agencies around the world continue to crack down on cybercrime, there is a growing demand for anonymized services by cybercriminal groups. Where there is demand, there is profit. Botnet-controlled nodes are naturally suited to building residential agents, and from our current cumulative case, this seems to be a trend in the DDoS loop in recent years, expanding the business from a single attack to network agents.

We are`XLab指令跟踪系统`The Aisuru network protocol is implemented, and as expected, it receives not only regular DDoS attack instructions, but also requests Proxy-related instructions.



It is clear that Aisuru is no longer satisfied with DDoS attacks as a single business model and has begun to dabble in the field of proxy services in an attempt to make full use of its vast node resources in order to gain more economic benefits.

IoC

C2

```
coerece[.ilovegaysex[.su
approach[.ilovegaysex[.su
ministry[.ilovegaysex[.su
lane[.ilovegaysex[.su
a.6mv1eyr328y6due83u3js6whtzuxfyhw[.ru
```

Report/Download Server

```
u[.ilovegaysex[.su
updatetoto[.tw
```

Proxy Relay C2

```
194.46.59[.169  United Kingdom|England|Exeter   AS206509|KCOM GROUP LIMITED
104.171.170[.241        United States|Virginia|Ashburn  AS7922|Comcast Cable
Communications, LLC
104.171.170[.253        United States|Virginia|Ashburn  AS7922|Comcast Cable
Communications, LLC
107.173.196[.189        United States|New York|Buffalo  AS36352|ColoCrossing
64.188.68[.193  United States|District of Columbia|Washington   AS46339|CSDVRS, LLC
78.108.178[.100 Czech Republic|Praha, Hlavni mesto|Prague       AS62160|Yes Networks
Unlimited Ltd
```

Sample

```
09894c3414b42addbf12527b0842ee7011e70cfd
51d9a914b8d35bb26d37ff406a712f41d2075bc6
616a3bef8b0be85a3c2bc01bbb5fb4a5f98bf707
ccf40dfe7ae44d5e6922a22beed710f9a1812725
26e9e38ec51d5a31a892e57908cb9727ab60cf88
08e9620a1b36678fe8406d1a231a436a752f5a5e
053a0abe0600d16a91b822eb538987bca3f3ab55
```

Appendix

CyberChef

```
https://gchq.github.io/CyberChef/#recipe=Fork('%5C%5Cn','%5C%5Cn',false)From_Base64('A-
Za-z0-
9%2B/%3D',true,false)XOR(%7B'option':'Hex','string':'ca%20fe%20ba%20be'%7D,'Standard',false)To_Hex('Spac
```