Inside Water Barghest's Rapid Exploit-to-Market Strategy for IoT Devices

By By: Feike Hacquebord, Fernando Mercês Nov 18, 2024 Read time: 12 min (3342 words)

Published: 2024-11-18 · Archived: 2026-04-05 14:58:05 UTC

IoT

In this blog entry, we discuss Water Barghest's exploitation of IoT devices, transforming them into profitable assets through advanced automation and monetization techniques.

Summary

- Water Barghest, which comprised over 20,000 IoT devices by October 2024, monetizes IoT devices by exploiting vulnerabilities and quickly enlisting them for sale on a residential proxy marketplace.
- Its botnet uses automated scripts to find and compromise vulnerable IoT devices sourced from public internet scan databases like Shodan.
- Once IoT devices are compromised, the Ngioweb malware is deployed, which runs in memory and connects to command-and-control servers to register the compromised device as a proxy.
- The monetization process, from initial infection to the availability of the device as a proxy on a residential proxy marketplace, can take as little as 10 minutes, indicating a highly efficient and automated operation.

There is a big incentive for both espionage motivated actors and financially motivated actors to set up proxy botnets. These can serve as an anonymization layer, which can provide plausibly geolocated IP addresses to scrape contents of websites, access stolen or compromised online assets, and launch cyber-attacks.

Examples of proxy botnets set up by advanced persistent threat (APT) actors are the VPNFilter botnetopen on a new tab and Cyclops Blinkopen on a new tab, both deployed by Sandworm and disrupted by the Federal Bureau of Investigation (FBI) in 2018 and 2022, respectively. Another example is the SOHO botnetopen on a new tab alleged to be operated by a Chinese company called the Beijing Integrity Technology Group; this botnet was disrupted in September 2024 by the FBI. The cybercriminal group Water Zmeu had a proxy botnet primarily consisting of Ubiquiti EdgeRouter devicesopen on a new tab, which was used by nation state actor Pawn Storm (also known as APT28 and Forest Blizzard) for two years for their espionage campaigns.

In this blog entry, we discuss our findings on another proxy botnet we associate with Water Barghest's intrusion set. This botnet was estimated to have more than 20,000 compromised Internet-of-Things (IoT) devices in October 2024. The starting point of our discovery of Water Barghest's intrusion set was our decade-old research into nation state actor Pawn Storm. Many Ubiquiti EdgeRouter devices had been used by this nation state actor since April 2022 in their espionage campaigns. Ubiquiti routers were the source of spear-phishing e-mails to numerous government organizations all over the world; they were used as SMB reflectors in NTLMv2 hash relay attacks, and they served as proxies to send stolen credentials on phishing websites to upstream servers.

In January 2024, the FBI tried to stop these espionage campaigns by disrupting the third-party criminal router botnetopen on a new tab that Pawn Storm was using. We associate this router botnet that consisted primarily of Ubiquiti EdgeRouter devices with the Water Zmeu intrusion set. During our investigation, we got our hands on a couple of the EdgeRouter devices that had been used by Pawn Storm, and we indeed found traces of espionage campaigns, and the router malware of Water Zmeu. We also found mysterious processes running in memory only, called *um* or *mm*. These processes appeared to be instances of Ngioweb malware running in memory, and this led us to the discovery of the Ngioweb botnet of Water Barghest. Apparently, some cybercriminals and APT actors share compromised infrastructure knowingly or unknowingly.

For more than five years, no significant publications were done on the Ngioweb botnet of Water Barghest while the botnet was up and running. This means that the actor group behind Water Barghest managed to keep a low profile. Like several other cybercriminals, Water Barghest did not make headlines in the news because of their careful operational security and high degree of automation. They had a steady income fueled by their cybercriminal activities, but they did not get the scrutiny they deserved. They quietly erased log files from their servers and made forensic analysis more difficult. They removed human error from their operations by automating almost everything. They also removed financial traceability by using cryptocurrency for anonymous payments.

However, they slipped up and suddenly had the spotlight pointed on them. This was because of a misjudgment, an operational mistake, or by using a vulnerability that made them greedy. One example of this is the well-mediatized usage of the zero-day vulnerability that was used against Cisco IOS XE devicesopen on a new tab in October 2023. Tens of thousands of Cisco routers were affected, and naturally, this sparked the interest of the security industry. We, too, became interested in the ever-intriguing question of whodunnit, and ultimately, we solved it at the technical level: We found that the attackers' infrastructure that was used to compromise thousands of Cisco IOS XE routers belonged to the five-year-old intrusion set of Water Barghest. This makes it very plausible that it was the Water Barghest group who had used the Cisco IOS XE device zero-day in October 2023.

And yet, even without the seemingly reckless usage of the Cisco IOS XE zero-day against tens thousands of routers, we would have discovered Water Barghest's router botnet operations anyway through our decade-long research into Pawn Storm as mentioned above. A series of seemingly unrelated events led us to the discovery of the way Water Barghest had automated every step between finding vulnerable routers and IoT devices on the internet, exploiting these devices, uploading and executing malware on them, and then monetizing the compromised assets for a steady income on an online marketplace of residential proxies.

One of the striking characteristics of the Water Barghest botnet is its high degree of automation, which will be discussed in the following section.

Water Barghest's automation

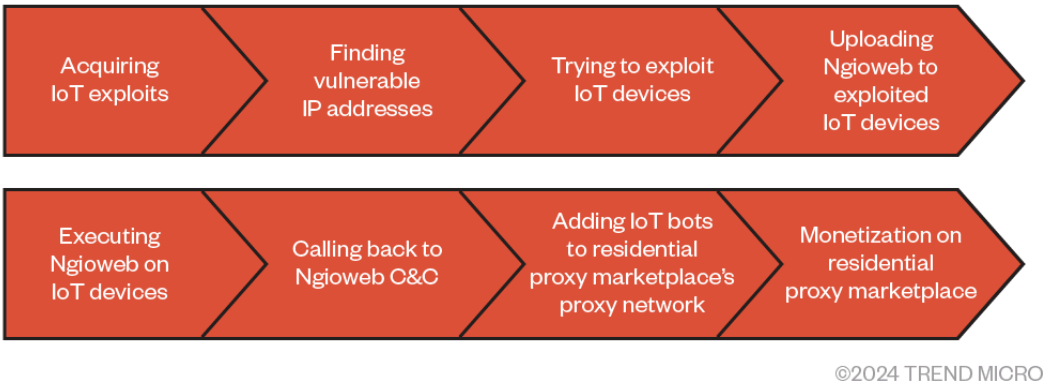

Figure 1. Automation by Water Barghest: from using IoT exploits to monetizing IoT bots on a residential proxy marketplace

As far as we know, apart from acquiring IoT exploits, Water Barghest has automated each step between finding vulnerable IoT devices and putting them for sale on a residential proxy marketplace (Figure 1). However, it all starts with acquiring IoT device vulnerabilities: Oftentimes these will be n-days, but in at least one case Water Barghest utilized a zero-day. With a list of exploits in hand, Water Barghest uses search queries on a publicly available Internet scan database like Shodan to find vulnerable devices and their IP addresses.

After retrieving these IP addresses, Water Barghest uses a set of data-center IP addresses with an oftentimes big longevity to try the exploits against potentially vulnerable IoT devices. When an exploit is successful, the compromised IoT devices download a script that iterates through Ngioweb malware samples that are compiled for different Linux architectures. When one of the samples runs fine, the malware Ngioweb will run in memory on the victim's IoT device. This means that the infection is not persistent; a reboot would remove the infection. When Ngioweb runs, it will register with a command-and-control (C&C) server. Oftentimes, within minutes the bot will receive instructions to connect to one of the residential proxy provider's 150 entry points (Figure 2). A speed test and name server test will follow, and the information will be sent to and be listed on the marketplace. The whole procedure between initial infection and making the bot available as a proxy on the marketplace may take no longer than 10 minutes. This shows again the professionalism and maturity of this threat actor, who has been around for more than five years.

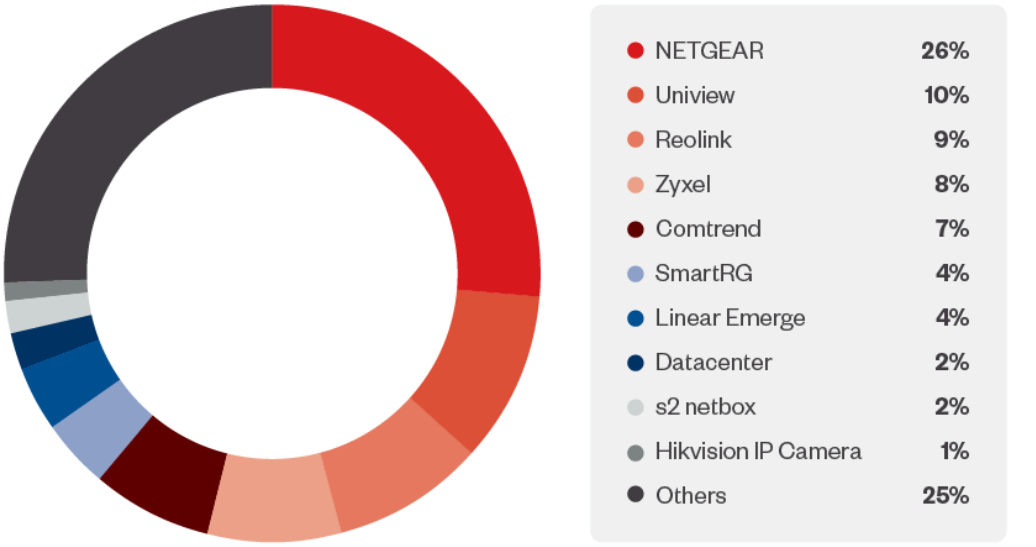

Figure 2. Estimated breakdown of the residential proxy provider's proxies by device type, based on 2,900 IP addresses we verified to be exit nodes of the marketplace at the end of October 2024

At the time of writing, Water Barghest deploys about 17 workers on virtual private servers (VPS) that continuously scan routers and IoT devices for known vulnerabilities. The same workers are also used to upload Ngioweb malware to freshly compromised IoT devices. Water Barghest has probably been using this mode of operation for years, with the worker IP addresses changing slowly over time. This setup allowed for a steady income for Water Barghest for years.

Ngioweb malware evolution

2018: Ramnit-powered Windows botnet

The Ngioweb malware strain goes back to 2018, when Check Point Research revealed it was being dropped by a Ramnit Trojanopen on a new tab. At the time, Ngioweb targeted computers using the Microsoft Windows operating system. The malware was already designed for turning an infected machine into a malicious proxy server. A few samples even go back to 2017, but the command-and-control (C&C) domain that gives the malware name was registered in 2018: ngioweb[.]su. If you're curious about the .su top-level domain (TLD), it's associated with the Soviet Union, and although the USSR doesn't exist anymore, the TLD is still valid.

2019: WordPress servers botnet

In 2019, Netlab researchers found the Linux variant of Ngiowebopen on a new tab. The malware worked similarly to its previous Windows version, but it had domain generation algorithm (DGA) features added. According to Netlab, the botnet was built mostly of web servers with WordPress installed, which suggests the threat actor could be exploiting a WordPress – or a WordPress plugin – vulnerability.

One of the parameters sent to the first stage C&C server was the 'sv' parameter (likely short for "software version"), which contained the value 5003. Just like its Windows version, Ngioweb used two-stage C&C servers

and implemented its own binary protocol over TCP for communicating with the second-stage C&C server.

2020: IoT devices botnet

In 2020, Water Barghest changed their targets to IoT devices. We found Ngioweb samples compiled for many different architectures. Additionally, Netlab published a blog entryopen on a new tab and Intezer posted on Xopen on a new tab about a live Ngioweb botnet. According to Netlab, the threat actor was exploiting nine different n-day vulnerabilities in IoT devices; this included NAS devices from QNAP and Netgear, but also D-Link devices, among others. The software version defined at the 'sv' field was changed to 0005.The software version defined at the 'sv' field was changed to 0005.

2024: Expanded targets

In 2024, we saw the IoT botnet created by Water Barghest at its full potential. The processes we found running in a bunch of EdgeRouter devices turned out to be a new version of Ngioweb. It works very similarly to its previous versions. When running, the malware performs the following actions:

- Initialize function pointers in runtime, which makes static analysis harder.
- Ignore any ignorable signals received by the kernel.
- Renames itself to "[kworker/0:1]" in a tentative to look like a kernel thread in the process list.
- Closes stdin, stdout, and stderr file descriptors to prevent any error reporting.
- Disables kernel's watchdog, effectively preventing it from rebooting the device.
- Reads the contents of /etc/machine-id (will be sent to the first-stage C&C later).
- Decrypts its AES-256-ECB (no padding) encrypted configuration.
- Generates and tries to resolve the DGA domains of the first-stage C&C.

Its main function is shown in Figure 3.

```
int __fastcall main(int argc, const char **argv, const char **envp)
{
  char *process_name; // rdi
  char buff[13]; // [rsp+2h] [rbp-206h] BYREF
  struct fn_ptr fn; // [rsp+10h] [rbp-1F8h] BYREF

  mw_init_function_pointers(&fn);
  mw_ignore_signals(&fn);
  if ( argc )
  {
    process_name = *argv;
    strcpy(buff, "[kworker/0:1]");
    mw_change_process_name(process_name, 13uLL, buff);
  }
  mw_check_functions_and_close_fds(&fn);        // creates a child process and exits
  strcpy(buff, "/");
  (fn.chdir)(buff);
  mw_disable_watchdog(&fn);
  mw_run(&fn);
  (fn.exit)(0LL);
  return 0;
}
```

Figure 3. Ngioweb's main function

The encrypted configuration is usually at the beginning of the .data section. In the following sample the key is at offset 0x0c from the start of the .data section and the encrypted data blob is 512 byte in size (Figure 4).

Figure 4. AES key and encrypted data

The encrypted configuration includes the ‘sv’ value, DGA seed and count, and C&C URL path, among other settings we didn’t fully analyze. Figure 5 shows a decrypted configuration with highlighted values of ‘sv’, DGA seed and count, C&C port and C&C URL path, respectively.

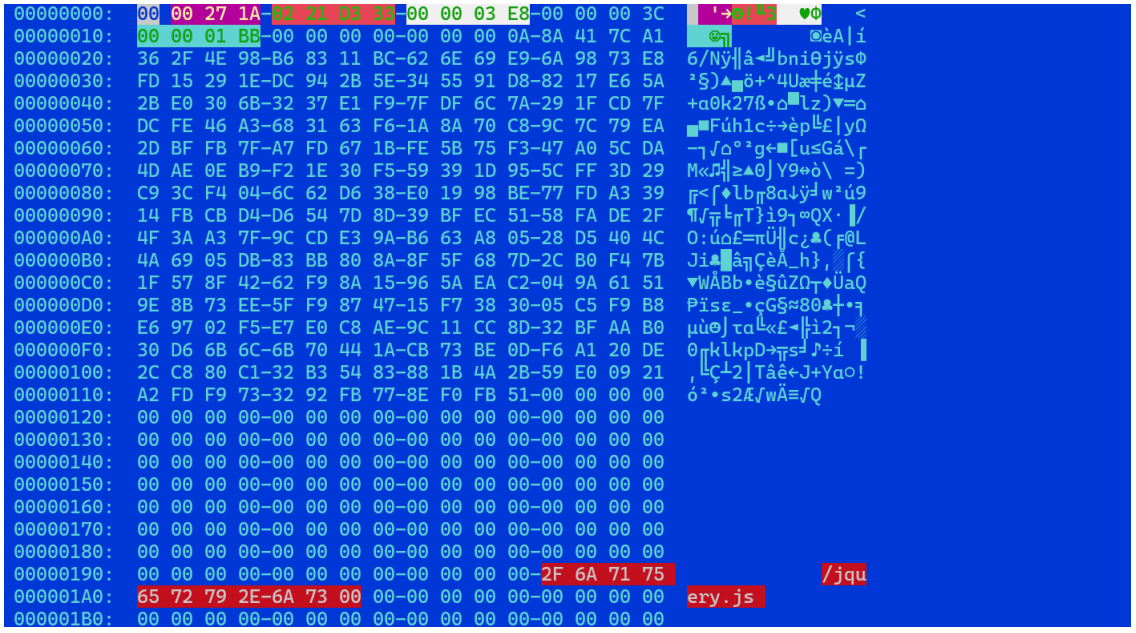

Figure 5. Decrypted configuration

We’ve also created a Python script to decrypt Ngioweb samples configuration, which is available in our GitHub repositoryopen on a new tab. The following is an example output:

```
PS D:\ > python ngioweb_config_extractor.py c267e0
[DEBUG] AES key found at offset 0xc from .data section
AES key (hex): db1f96b20679f9fb9cbd96b242ab8530102c0105b64c83c3ae544f87594a6fa9
DGA seed (hex): 0221d333
DGA count: 1000
URL path: /jquery.js
sv: 271a
```

For every generated domain the malware tries to resolve with A/AAAA requests, it also sends a TXT request expecting a base64-encoded binary blob (Figures 6 and 7).

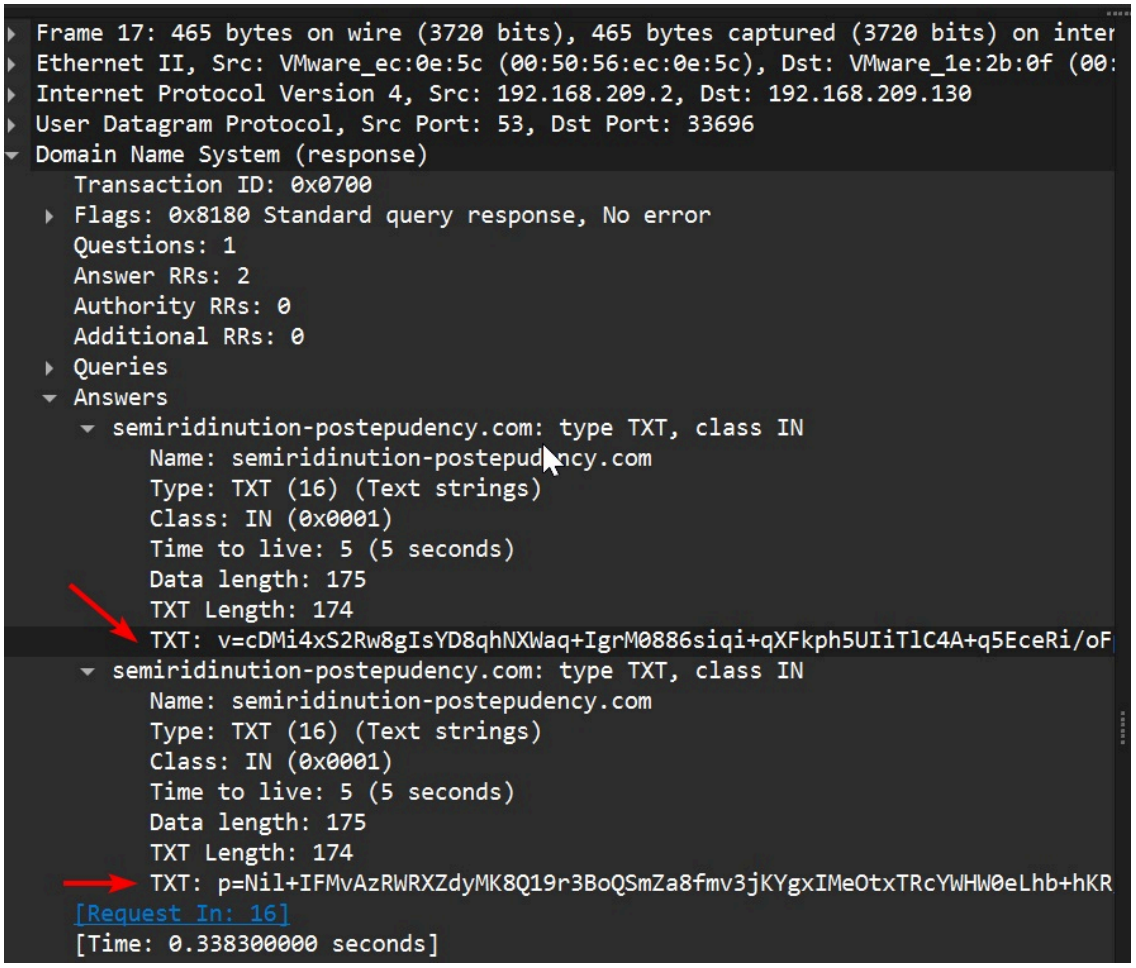

Figure 6. First-stage C&C response for a DNS TXT request

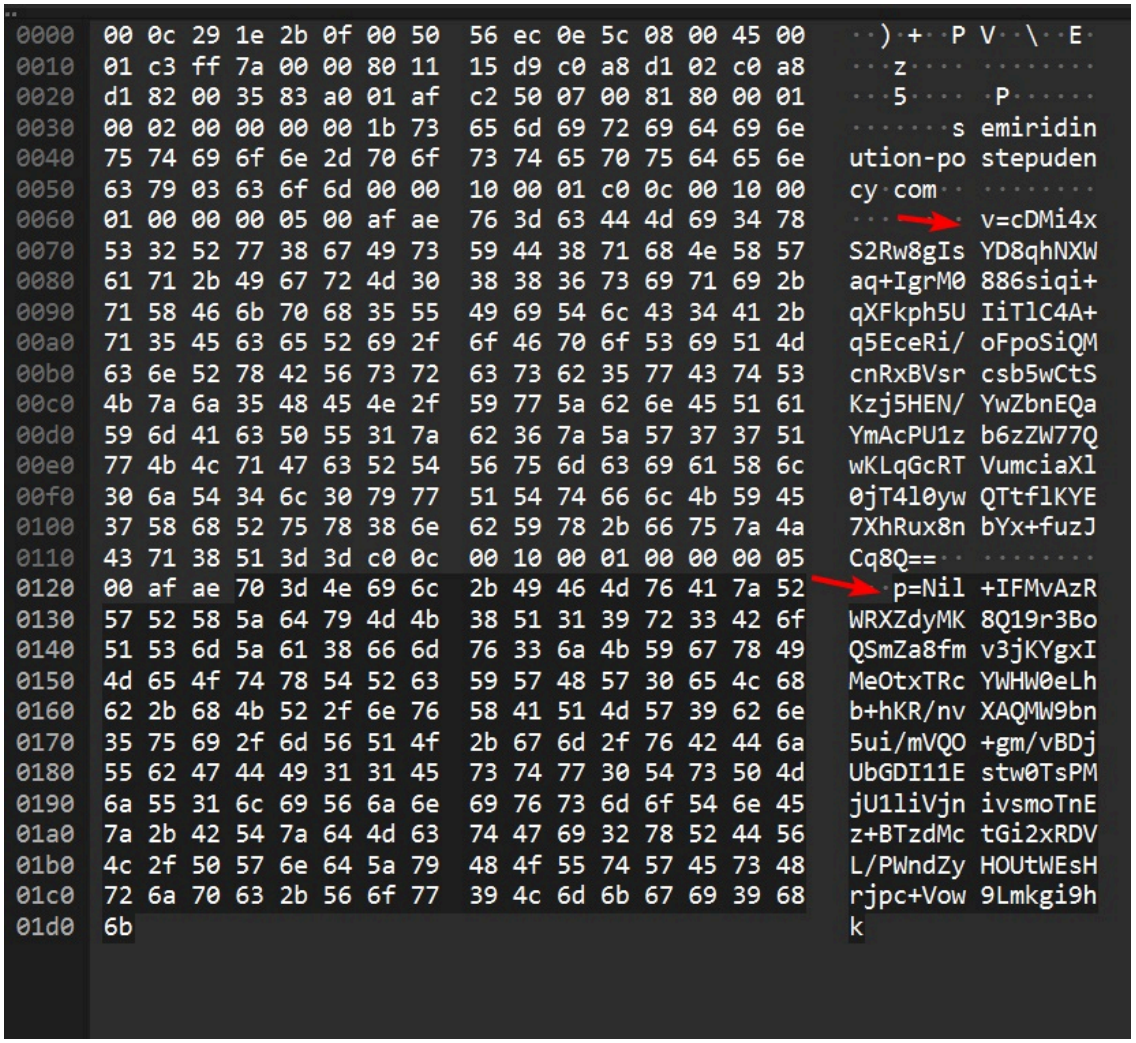

Figure 7. First-stage C&C response for a DNS TXT request

Unfortunately, we didn't finish the full analysis of how this binary blob is used. Nonetheless, the next stage is to send a GET request to the C&C server. Like its previous versions, this request is an unencrypted request at port 443/tcp and contains some base64-encoded data that identifies the victim:

> GET /jquery.js?
> h=aWQ9MDEyMzQ1Njc4OWFiY2RlZiZ2PWFybXY3bCZzdj0yNzFhJnlic25xbndmYXR5anV0c2w=
> HTTP/1.1\r\n
> Host: ultradomafy.net\r\n
> User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:59.0) Gecko/20100101 Firefox/59.0\r\n
> Accept: text/html\r\n
> Connection: close\r\n
> \r\n

The sample base64-decoded data is as follows:

> id=0123456789abcdef&v=armv7l&sv=271a&ybsnqnwfatyjutsl

- id - first 16 characters from /etc/machine-id
- v - architecture of the infected device

- sv - software version (assumed)
- <random string with 16 lowercase letters>

In this version, the ‘sv’ parameter changed to 271a.

The first-stage C&C server response has the following format:

> HTTP/1.1 200 OK\r\n
> Server: openresty/1.19.9.1\r\n
> Date: Mon, 11 Mar 2024 23:23:47 GMT\r\n
> Content-Type: text/plain; charset=utf-8\r\n
> Content-Length: 8\r\n
> Connection: close\r\n
> \r\n
> WAIT 60\n

The above response contains the WAIT command, which instructs the malware to wait a few seconds before querying the C&C server again. Its parameter is the number of seconds to wait for (60 in the example).

Supported commands are:

- WAIT
- CONNECT
- DISCONNECT
- CERT

This is paired with previous versions.

Different child processes check if the following iptables rule is present:

iptables -I INPUT -p tcp --tcp-flags RST RST -j DROP --sport 5000:55000

If the rule is not already active in netfilter, the malware adds it. We believe this is to prevent connection resets, ensuring its communication channels remain open.

After waiting 60 seconds, the malware might get a different answer, as shown in Figure 8:

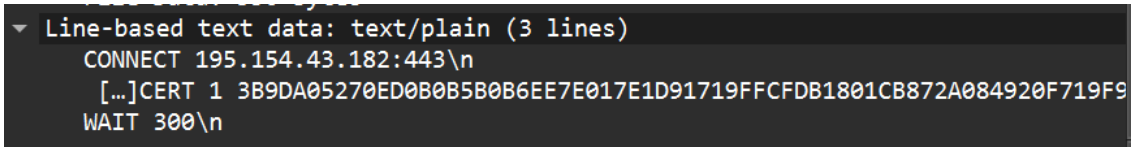

Figure 8. Malware receives three commands: CONNECT, CERT, and WAIT

This instructs the malware to connect to a second-stage C&C, 195.154.43.182 in this case. We associate this second-stage IP address with one of the about 150 entry nodes of the residential proxy service.

Before publishing the new victim’s IP address for sale as a reverse proxy on the residential proxy marketplace’s website, the malware downloads a big file containing random bytes from the second-stage C&C (Figure 9).

```
Hypertext Transfer Protocol
  GET /test.zip HTTP/1.1\r\n
  Host: 195.154.43.182\r\n
  User-Agent: Mozilla/5.0 (Windows NT 6.1; rv:43.0) Gecko/20100101 Firefox/43.0\r\n
  Accept: */*\r\n
  Cookie: \r\n
  \r\n
  [Full request URI: http://195.154.43.182/test.zip]
```

Figure 9. File downloaded from second-stage C&C

This is to estimate the victim's bandwidth, which we believe will be used to calculate the final price on the residential proxy marketplace.

In this version, Water Barghest expanded Ngioweb's list of targeted IoT devices, which now includes IoT devices from more brands, such as:

- Cisco
- DrayTek
- Fritz!Box
- Linksys
- Netgear
- Synology
- Tenda
- Western Digital
- Zyxel

Water Barghest has been targeting devices from the brands above with a range of n-day vulnerabilities and lots of old ones.

Residential proxy marketplace

In our assessment, a significant part of the exit nodes that a particular residential proxy marketplace offers for rent belong to devices that are infected with Ngioweb malware. In a couple of cases, we were able to verify that a fresh Ngioweb infection resulted in the corresponding IP address being offered for rent on the marketplace's website within a few minutes after the initial infection (Figure 10). The residential proxy provider allows for cryptocurrency payments only.

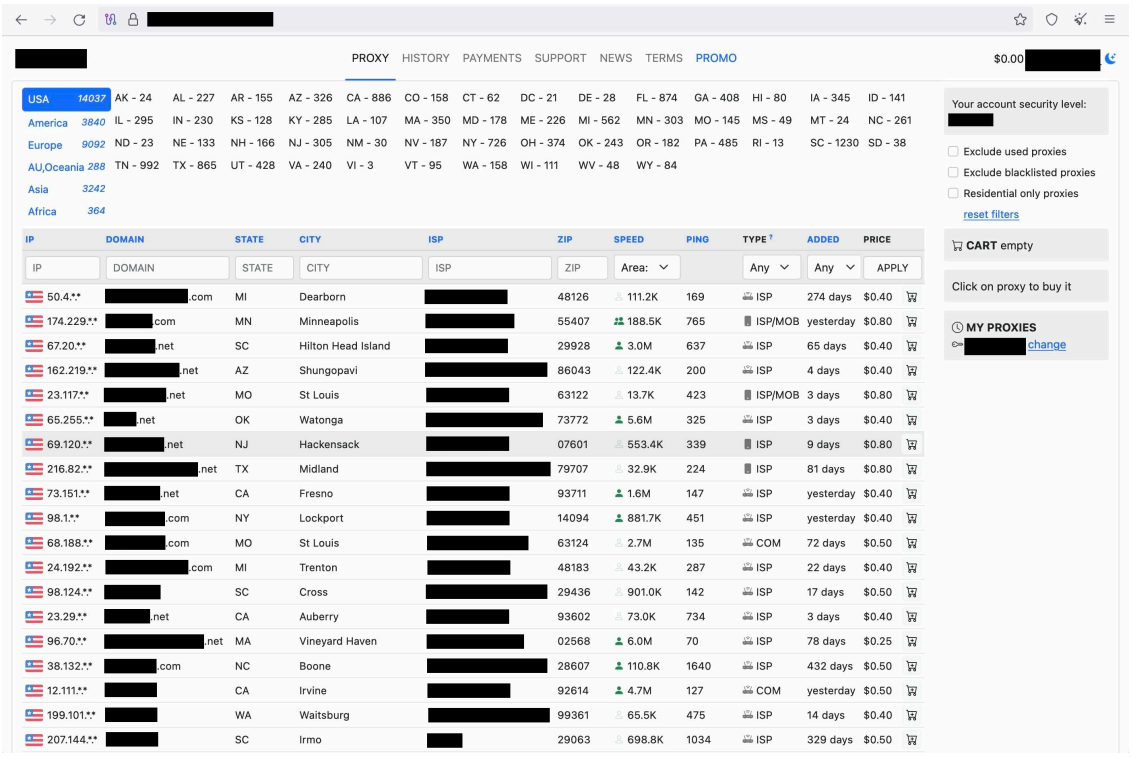

Figure 10. Residential proxy marketplace's website

As far as we can tell, the proxies on the residential proxy marketplace (Figure 11) are back connect proxies. Ngioweb bots are instructed to connect to one of about 150 datacenter IP addresses we associate with the marketplace that are also used as second-stage C&C of Ngioweb-infected devices. Paying users of the residential proxy service can then connect to a temporary high TCP port on one of the 150 datacenter IP addresses, and then route traffic through the Ngioweb bots.

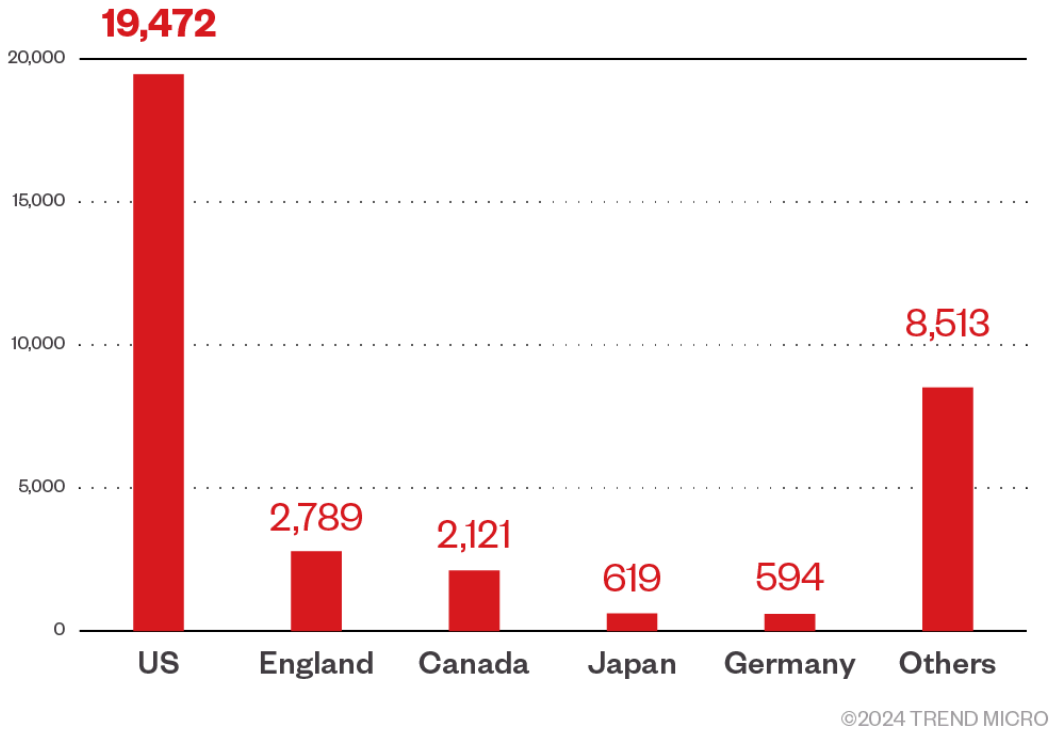

Figure 11. Breakdown of proxies by country according to the residential proxy marketplace‘s website. Data checked in October 2024.

With data provided by Team Cymru’s Real-time Threat Intelligence Platform, Pure Signal Reconopen on a new tab, we were able to explicitly enumerate a significant part of the marketplace’s residential proxy network over time and verify that Ngioweb bots were added to the marketplace’s offerings within 10 minutes after initial infection.

Outlook and conclusions

For years, mid-sized proxy botnets have existed without them being disrupted and published on. Examples are the botnets we associate with the Water Barghest and Water Zmeu intrusion sets. The actor groups behind these intrusion sets have made refinements in their setup over the years and automated their operations to a high degree. Eventually, some of these botnets were brought to the attention of the security industry. In the case of Water Barghest, this was because of the use of Water Barghest’s infrastructure to deploy a zero-day against Cisco IOS XE devices that infected tens of thousands of routers in October 2023. In the case of Water Zmeu, APT actor Pawn Storm’s use of this criminal botnet for espionage purposes motivated the FBI to disrupt the Water Zmeu-associated router botnet. Upon completing our write-up on Water Barghest's activities, we became aware of a LevelBlue blog entryopen on a new tab that partially overlaps with our findings.

APT actors have also deployed their dedicated IoT botnets sometimes for years, before they were disrupted by the FBI and its partners. APT actors and financially motivated actors will continue to have an interest in building their own IoT botnets for anonymization purposes and espionage. They also will continue to use third-party botnets or commercially available residential proxy services.

We expect that both the commercial market for residential proxy services and the underground market of proxies will grow in the coming years, because the demand from APT actors and cybercriminals actor groups is high. Protecting against these anonymization layers is a challenge for many enterprises and government organizations around the world. Court-approved disruptions of proxy botnets will help put a dent into malign operations, but it is better to do something against the source of the problem: securing IoT devices is of paramount importance, and whenever possible, these devices should not be exposed to incoming connections from the open internet.

Whenever an IoT device accepts incoming connections on the open internet, commercial scanning services will quickly find them online, and malicious actors can find them too via bought or stolen access to these internet scanning services. Using internet scan data, the automated scripts of bad actors can quickly try known vulnerabilities, and possibly even zero-days, against the exposed IoT devices. In the case of Water Barghest, we have seen that the time between exploiting an IoT device and putting them for sale on a residential proxy marketplace can be as little as 10 minutes. Therefore, it is important not to expose IoT devices to incoming internet connections whenever it is not business-essential, and put mitigations in place to avoid their infrastructure being part of the problem itself.

Trend Micro Vision One Threat Intelligence

To stay ahead of evolving threats, Trend Micro customers can access a range of Intelligence Reports and Threat Insights within Trend Micro Vision One. Threat Insights helps customers stay ahead of cyber threats before they happen and better prepared for emerging threats. It offers comprehensive information on threat actors, their malicious activities, and the techniques they use. By leveraging this intelligence, customers can take proactive steps to protect their environments, mitigate risks, and respond effectively to threats.

Trend Micro Vision One Intelligence Reports App [IOC Sweeping]

Ngioweb IoCs used in Water Barghest Campaigns

Trend Micro Vision One Threat Insights App

Threat Actors: Water Barghestopen on a new tab

Emerging Threats: Water Barghest's Rapid Exploit-to-Market Strategy for IoT Devicesopen on a new tab

Hunting Queries

Trend Micro Vision One Search App

Trend Micro Vision Once Customers can use the Search App to match or hunt the malicious indicators mentioned in this blog post with data in their environment.

Detection of Ngioweb Malware

malName:*NGIOWEB* AND eventName:MALWARE_DETECTION

More hunting queries are available for Vision One customers with Threat Insights Entitlement enabledopen on a new tab.

Indicators of Compromise (IOCs)

The full list of IOCs can be found hereopen on a new tab. For DGA-generated domains, please refer to this GitHub repositoryopen on a new tab.

YARA rules

As Ngioweb samples are highly obfuscated, an easy approach is to look for known AES keys in .data section. However, it is possible to find samples without section headers. In this case, searching for the AES key in the whole binary (or in a loadable segment) does the job. There are also samples with an AES KEY c91795b59248562e44d6c07526c7ab89dfe45344293703a94a3ae5ff02eab5a4 that we believe could be part of some test, so we didn't include them in our IOC list. The YARA rules can be found hereopen on a new tab.

Tags