SOC Report

1.1 Introduction

IIJ launched the wizSafe security brand in 2016 and has continued working to create an environment in which its customers can use the Internet safely. One of its activities in this regard is the regular dissemination of information on security through wizSafe Security Signal[*1]. IIJ's Data Analytics Platform, which aggregates security logs from IIJ services, is used to produce this information, with security issues being analyzed from a variety of angles in combination with threat intelligence that IIJ collects daily.

Section 1.2 of this report looks back at major security topics that arose in 2024 in calendar format, and Section 1.3 presents observational information on DDoS attacks, which have once again come into the spotlight due to their societal impact. This observational information includes the incidence of DDoS attacks detected on IIJ services, as well as IoT malware infection activity, a key element in recent DDoS attacks. Section 1.4 introduces mirai-toushi, an analysis support tool developed to streamline the analysis of IoT malware.

1.2 2024 Security Summary

Tables 1 and 2 show the security incidents that the SOC focused on from among those that rose to prominence in 2024.

*1 wizSafe Security Signal (https://wizsafe.iij.ad.jp/).

Table 1: Security Topic Calendar (January – May)

Month	Summary
January	**Vulnerabilities in Ivanti products** Ivanti disclosed zero-day vulnerabilities (CVE-2023-46805, CVE-2024-21887) in Ivanti Connect Secure (formerly Pulse Connect Secure) and Ivanti Policy Secure gateways. These vulnerabilities could be combined to allow arbitrary command execution without authentication. At the time of disclosure, patched versions had not been released, and the vulnerabilities were already being widely exploited. The patches, when released, also fixed a privilege escalation vulnerability (CVE-2024-21888) and an SSRF vulnerability (CVE-2024-21893). Reports indicate CVE-2024-21893 was being exploited before the patch's release.
January	**Vulnerabilities in Citrix products** Citrix disclosed vulnerabilities (CVE-2023-6548, CVE-2023-6549) in NetScaler ADC and NetScaler Gateway. CVE-2023-6548 could lead to remote code execution and CVE-2023-6549 to denial of service (DoS). Exploits of both vulnerabilities had been observed in the wild at the time of disclosure.
February	**Vulnerability in SonicWall products** SonicWall disclosed an improper authentication vulnerability (CVE-2024-22394) in the SSL-VPN functionality of SonicOS. This vulnerability could allow an attacker to bypass authentication.
February	**Vulnerability in Fortinet products** Fortinet disclosed an out-of-bounds write vulnerability (CVE-2024-21762) in FortiOS and FortiProxy. This vulnerability could allow execution of arbitrary code or commands. Exploits of the vulnerability had already been observed.
February	**International effort against LockBit ransomware** Europol (the European Union Agency for Law Enforcement Cooperation) announced that a joint operation against the cybercriminal group using the LockBit ransomware had led to the arrest of two LockBit actors and the seizure of attack infrastructure. The operation was known as Operation Cronos and involved law enforcement from 10 countries, including Japan. A new LockBit leak site was subsequently established and attacks resumed. In October, as part of Operation Cronos, authorities announced the additional arrest of four individuals, including a LockBit developer, and the seizure of some servers in the attack infrastructure.
February	**Leak of information related to threat actors** Data believed to be from a Chinese security firm's internal materials was found to have been uploaded to GitHub. According to a US security firm, the content indicated that the Chinese company had been providing development support for tools used by threat actors claimed to have ties to China.
March	**Economic organization hit by support scam** An economic organization disclosed that it had suffered financial damage totaling 10 million yen through fraudulent funds transfers via online banking. This resulted from the installation of remote access software on work computers via a support scam tactic that involved displaying fake warning screens in browsers.
March	**Personal information leak at a software developer** A software developer disclosed that customer personal information had been leaked due to misconfigured access restrictions on storage servers used by their service offerings. According to an investigation report released in May, personal data of 158,929 individuals was leaked, with no secondary harm, such as misuse of that personal data, having been observed at the time of reporting.
March	**XZ Utils vulnerability** The Tukaani Project disclosed an incident (CVE-2024-3094) in which malicious code was inserted into XZ Utils, a suite of lossless compression tools. Under specific conditions, there was a risk of external connections being made via SSH ports. XZ Utils is used in several Linux distributions, and a wide range of systems were affected.
April	**Cyberattack on lens manufacturer** A lens manufacturer disclosed a system failure resulting in the shutdown of production facility systems and order processing systems. A cyberattack by a third party was identified as the cause of the failure, and the company reported that its systems had largely been restored as of April 23. The effects of this incident went beyond the company directly targeted by the attack, with several retailers the company does business with halting sales of some lenses, for instance.
April	**Vulnerability in Palo Alto Networks products** Palo Alto Networks disclosed a vulnerability (CVE-2024-3400) involving OS command injection in the GlobalProject feature of PAN-OS, and that attacks exploiting this vulnerability had already been observed.
May	**DDoS attack on railway company** Internet services (including payment systems) provided by a railway company experienced connectivity issues. The system failure was caused by a cyberattack targeting IC transit card-related systems, with reports suggesting it may have been a DDoS attack.
May	**Vulnerability in Check Point Software Technologies products** Check Point Software Technologies disclosed an information leakage vulnerability (CVE-2024-24919) in its security gateway products, and that attacks involving password authentication-based unauthorized login attempts had been observed.
May	**Unauthorized Bitcoin outflow from Japanese cryptocurrency exchange** A Japanese cryptocurrency exchange disclosed an unauthorized outflow of Bitcoin worth around 48.2 billion yen at the then prevailing exchange rate. In September, the Kanto Finance Bureau issued an administrative order (business improvement order) based on Article 63-16 of the Payment Services Act, requiring the exchange to report on its investigation into the cause of the incident and its response to customers. In December, the exchange announced it would transfer customer assets to another domestic cryptocurrency exchange and discontinue its service operations. Subsequently, the FBI, the US Department of Defense Cyber Crime Center (DC3), and Japan's National Police Agency jointly released information identifying the perpetrator as TraderTraitor, an attack group with ties to North Korea. The released document mentioned social engineering tactics, such as attackers posing as recruiters and contacting employees of the developer of the cryptocurrency wallet software used by the exchange.
May	**Ransomware attack on systems developer** A systems developer disclosed that it was the subject of a ransomware attack involving the encryption of files on its servers and PCs. As part of its subsequent investigation, it reported that the intrusion occurred via a VPN and that stolen information, including the personal information of customers, had been temporarily published on the attack group's leak site. As a result of this incident, the company's ISO27001 and ISO27017 certifications were temporarily suspended in September, along with its PrivacyMark in December. The impact of the incident was widespread, with numerous companies and municipalities that had outsourced work to the systems developer reporting personal information leaks.

Table 2: Security Topic Calendar (June – December)

Month	Summary
June	**Ransomware attack on entertainment company** An entertainment company engaged in publishing, web services, educational business, etc. revealed that a ransomware attack had caused widespread outages affecting its publishing, web services, and merchandising businesses. It worked to restore the affected business activities, and over August and September reported that shipment volumes had recovered to normal levels and that its services had been fully restored. The attack group repeatedly claimed to have leaked information stolen from the company. An investigation ultimately confirmed personal information on 254,241 individuals and internal and external corporate information had been leaked.
July	**Major outage caused by security product malfunction** Faults in some update files for Windows hosts running a US security company's endpoint security product resulted in updated hosts crashing worldwide. This event affected a wide range of industries, resulting, for instance, in flight cancellations and delays and retail store POS registers being unusable.
July	**International effort targeting DDoS-for-hire service** The UK National Crime Agency (NCA) announced that following an investigation by the Police Service of Northern Ireland (PSNI) and the FBI, it had seized the platform of the digitalstress DDoS-for-hire service and arrested one of the site's suspected controllers. This operation was carried out as part of Operation Power Off, a joint initiative aimed at shutting down DDoS-for-hire services.
August	**Arrest of DDoS-for-hire service user** Multiple media outlets reported that a man who had used a DDoS-for-hire service to attack Japanese websites had been arrested by Japan's National Police Agency. DDoS-for-hire services are the target of a coordinated international response called Operation Power Off. Separate arrests related to the use of DDoS-for-hire services were subsequently reported in November and December.
September	**Vulnerability in SonicWall products** SonicWall updated its security advisory on an improper access control vulnerability (CVE-2024-40766) in SonicOS, which it had disclosed in August, to indicate that not only the management access interface but also SSL-VPN was affected. This vulnerability allows an unauthenticated attacker to mount remote attacks. The company recommended updating affected products, changing local user passwords, enabling multi-factor authentication, logging SSL-VPN login events, and implementing account lockout mechanisms.
September	**Ransomware attack on logistics service provider** A logistics service provider disclosed that several of its servers were the subject of a ransomware attack. Several business-related systems were shut down, causing delays and stoppages to business partners' inbound and outbound goods processing. The company initially mentioned the possibility of personal information being leaked by the attack, but in October, it reported that no such information leaks had been confirmed as the data had not been published on leak sites etc. Several companies in the supply chain that used the affected company's services also reported they had been affected by the attack.
October	**Ransomware attack on childcare facility management company** A childcare facility management company disclosed that its servers were the subject of a ransomware attack. When making the disclosure, the company mentioned the possibility of personal information being leaked, but according to a follow-up report in November, an investigation by an external expert found that while personal information on the servers could possibly have been viewed, there was no evidence it had been exfiltrated. Several municipalities that had outsourced facility operations to the company also reported being affected by the cyberattack.
October	**Ransomware attack on insurance claims assessment firm** An insurance claims assessment firm disclosed that its servers had been infected with ransomware. An investigation revealed that the intrusion may have occurred through RDP connections to the servers following brute force attacks on UTM devices. While nothing was found to indicate that personal data leaks had occurred, the firm said the possibility of data leaks could not be ruled out as logs had been deleted for the timeframe during which unauthorized access to local folders and the incident itself had occurred. Several companies in the supply chain that used the affected firm's services also reported they had been affected by the attack.
October	**Vulnerability in Fortinet products** Fortinet disclosed a missing authentication vulnerability (CVE-2024-47575) in the fgfmd daemon of its FortiManager security management platform. This vulnerability could allow remote execution of arbitrary code or commands. Attacks exploiting this vulnerability had already been observed at the time of disclosure.
November	**Vulnerabilities in Palo Alto Networks products** Palo Alto Networks disclosed an authentication bypass (CVE-2024-0012) and a privilege escalation (CVE-2024-9474) vulnerability in the PAN-OS web management interface. These vulnerabilities were already being exploited at the time of disclosure, with commands having been executed on affected products and a webshell having been deployed.
December	**Multiple DDoS attacks in Japan** A spate of service disruptions was attributed to DDoS attacks involving massive amounts of data being sent, primarily targeting domestic airlines and financial institutions in Japan. These attacks affected web services operated by the targeted companies and airline operations.

1.3 DDoS Attack Topics

1.3.1 Observational Information on DDoS Attacks

This section presents information on DDoS attacks detected on IIJ services. Table 3 summarizes attacks detected in 2024 on a monthly basis.

The largest attack observed in 2024 involved 174.80Gbps of traffic and employed a DNS-based UDP amplification method. The longest attack, meanwhile, lasted around 3 hours and 24 minutes and was determined to have employed a TCP SYN Flood technique. We note no major changes from the past in the methods used for the other DDoS attacks we observed, regardless of scale, with UDP amplification and TCP/UDP Flood methodologies again being used.

An investigation of source IP addresses involved in the UDP Flood attacks observed in December 2024, conducted using external information sites, revealed that over half of the IP addresses on which information could be obtained belonged to IoT devices such as TP-Link routers and Hikvision IP cameras. These devices have hardcoded default passwords.

It seems that external parties were able to log in because the default password had not been changed.

Even as IoT devices are becoming more common, security measures and such tend to be given short shrift. This is why many devices around the world remain in a vulnerable state and are actively targeted by attacks. When attacks succeed, the devices are at risk of being infected by IoT malware and recruited for DDoS attacks as described in this section. It is important to take steps against this, such as changing default passwords to difficult-to-predict passwords, updating firmware to the latest version, and discontinuing the use of out-of-support devices.

1.3.2 Trends in Attack Traffic Targeting IoT Devices as Observed by Honeypots

IIJ ties attacks targeting IoT devices observed in its honeypots[*2] into its Data Analytics Platform for analysis and utilization of the results. When IoT devices are infected with IoT malware via external attacks, they may be exploited as part of botnets used to conduct DDoS attacks. Here, we look at IoT malware infection activities as observed by IIJ's honeypots.

Many IoT devices allow users to log in via Telnet over the Internet. In some cases, such devices are used with their factory default Telnet username and password settings, and this is one potential entry point for IoT malware infections. IoT malware that forms botnets like

Table 3: DDoS Attack Detections (2024)

	Observational data on the largest-scale attacks				Observational data on the longest-duration attacks	
Month	Detection count (daily avg.)	Packet count (10,000pps)	Bandwidth (Gbps)	Main attack method	Duration	Main attack method
1	7.58	533	2.47	TCP SYN/ACK Reflection Attack	13 minutes	TCP SYN Flood
2	9.79	690	71.11	UDP Amplification using DNS Protocol	25 minutes	TCP ACK Flood
3	7.61	6	0.61	HTTP Flood	1 hour 41 minutes	HTTP Flood
4	7.97	526	54.90	UDP Amplification combining multiple protocols (WSD, SADP, CoAP, etc.)	39 minutes	UDP Amplification combining multiple protocols (WSD, SADP, CoAP, etc.)
5	9.13	70	8.15	HTTP Flood	16 minutes	HTTP Flood
6	13	1228	110.88	UDP Amplification using DNS Protocol	1 hour 10 minutes	TCP ACK Flood
7	10.87	1050	90.64	UDP Flood	3 hours 24 minutes	TCP SYN Flood
8	12.23	426	44.34	UDP Amplification using DNS Protocol	12 minutes	UDP Amplification using DNS Protocol
9	8.4	262	9.74	UDP Amplification using NTP Protocol	2 hours 3 minutes	UDP Amplification using NTP Protocol
10	7.77	477	49.63	UDP Amplification using DNS Protocol	12 minutes	UDP Amplification using DNS Protocol
11	7.23	1680	174.80	UDP Amplification using DNS Protocol	1 hour 14 minutes	UDP Amplification using DNS Protocol
12	7.06	332	12.12	UDP Flood	2 hours 3 minutes	UDP Flood

*2 For details of honeypot support for IoT devices, see "Focused Research (1)" in IIR Vol. 36 (https://www.iij.ad.jp/en/dev/iir/036.html).

Mirai[*3] is known to scan Telnet services across random global IP addresses to expand its infection base.

Honeypots make it possible to observe scanning and attacks targeting a wide range of global IP addresses, and when combined with data on increases in the number of source devices and the like, they can be expected to capture phenomena such as the spread of specific IoT malware. Some IoT malware like Mirai, however, has had its source code made public, and so a range of variants has been created through modifications to the original source code. So it must be noted that the characteristics of communications and attack payloads may differ from what was observed with the original source code.

Below are the results of an analysis of attack communications targeting Telnet (23/TCP, 2323/TCP) observed by our honeypots in 2024 showing changes in the number of source IPs and attack payloads, partitioned into sources with and without Mirai characteristics[*4].

Figure 1 plots the number of source IP addresses with Mirai characteristics (normalized to a maximum value of 1 during the observation period) classified based on attack payload. This investigation revealed a notable increase in October of sources that included the echo command being used to execute the hex-escaped string "gayfgt" (\x67\x61\x79\x66\x67\x74) (which we classify into Group A) and sources that included the command "/bin/busybox NUG" in their payloads (which we classify into Group B). There was also another increase in Group A sources in December.

An investigation of the malware infecting systems via attack communications classified into Group A revealed that the majority were characteristic of the Mirai variant RapperBot. Specifically, we found the malware to be congruent with previous RapperBot samples in terms of using the same encryption method for configuration data (single-byte XOR), using the same pattern for encryption keys (the last byte of the encrypted data), and having the same YouTube URLs in the configuration. Note that while the "gayfgt" output string characteristic of Group A is known to be a feature of the IoT malware called BASHLITE (aka Gafgyt), we determined that the specimens we examined were not BASHLITE.

An investigation of the malware infecting systems via attack communications classified into Group B revealed that the majority exhibited some characteristics of the Mirai variant FICORA observed in May 2024 (specifically, the 8-byte XOR encryption of configuration data and identical encryption keys). As such, it is likely that Group B comprises FICORA or a similar Mirai variant.

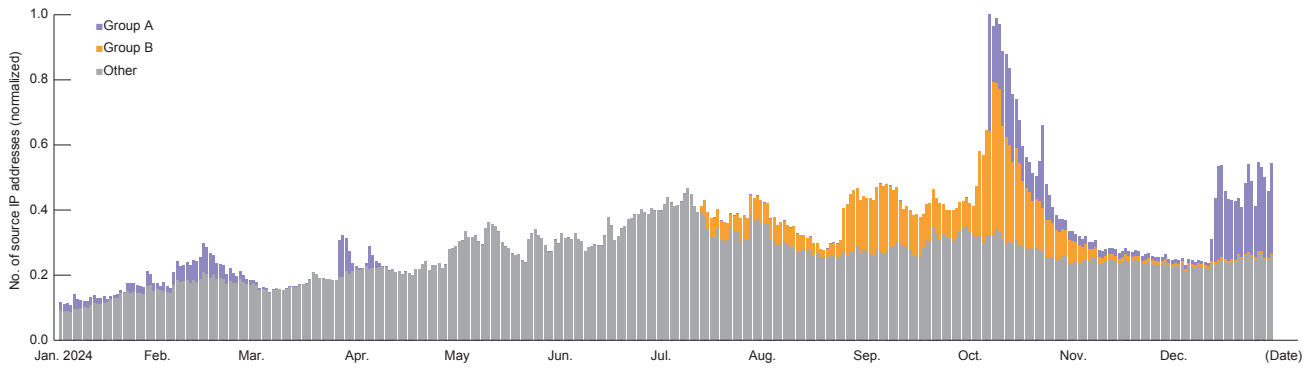


Figure 1: Number of Source IP Addresses (Normalized) in Different Categories Observed by IIJ's Honeypots (Sources with Mirai Characteristics)

*3 A detailed explanation of Mirai can be found in "1. Infrastructure Security: Mirai Botnet Detection and Countermeasures" in IIR Vol. 33 (https://www.iij.ad.jp/en/dev/iir/033.html).

*4 This refers to instances in which the sequence number in the TCP header and the IP address value are identical, and the source port is greater than 1024. Note, though, that there are Mirai variants that do not exhibit these characteristics.

Moving on, Figure 2 plots the number of source IP addresses without Mirai characteristics (normalized to a maximum value of 1 during the observation period) classified based on attack payload. Our investigation revealed that from January, many sources were sending out attack traffic that included the command "/bin/busybox hostname PBOC" (which we classify into Group C), but this gradually decreased toward September, after which we observed an increase in attack sources sending payloads that included execution of "/bin/busybox hostname whomp" (which we classify into Group D).

An investigation of the malware infecting systems via attack communications classified into Group C revealed that the majority were characteristic of the Mirai variant InfectedSlurs. Specifically, we found that they matched previous InfectedSlurs samples in terms of the encryption method for the configuration data (a combination of RC4 and XOR), the encryption keys, and the strings contained within the malware.

An investigation of the malware infecting systems via attack communications that we classified into Group D revealed that the majority exhibited some characteristics of the Mirai variant CatDDoS observed in April 2024 (ChaCha20 encryption used for configuration data with the same encryption keys and nonce values). This suggests that Group D is likely CatDDoS or a similar Mirai variant.

Within Group C, we also observed attack communications originating from sources within Japan. Figure 3 plots the number of Japanese source IP addresses without Mirai characteristics (normalized to a maximum value of 1 during the observation period), partitioned into Group C and "other".

The number of source IP addresses classified into Group C remained elevated from the end of 2024 until early February, and then dipped before temporarily increasing again from June through mid-July. As mentioned, the majority of malware infecting systems through Group C attacks was InfectedSlurs, suggesting a possible increase

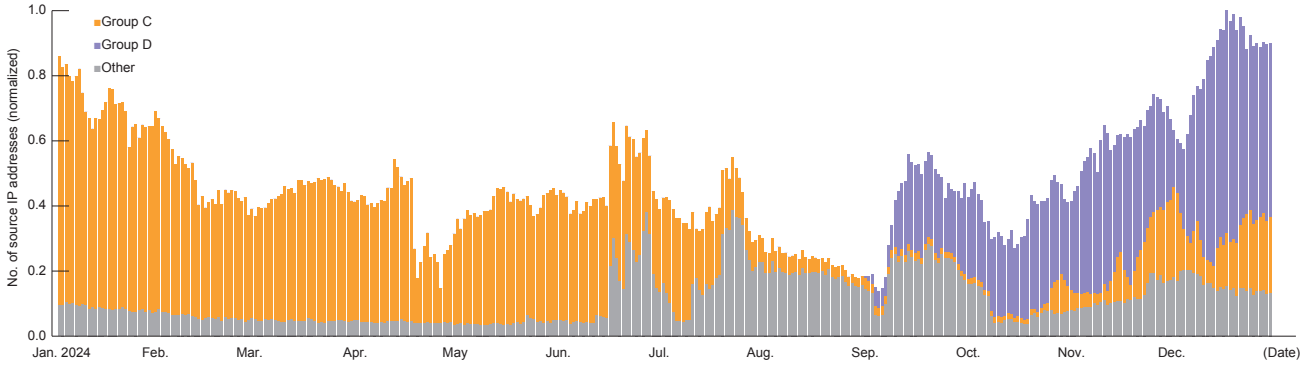


Figure 2: Number of Source IP Addresses (Normalized) in Different Categories Observed by IIJ's Honeypots (Sources without Mirai Characteristics)

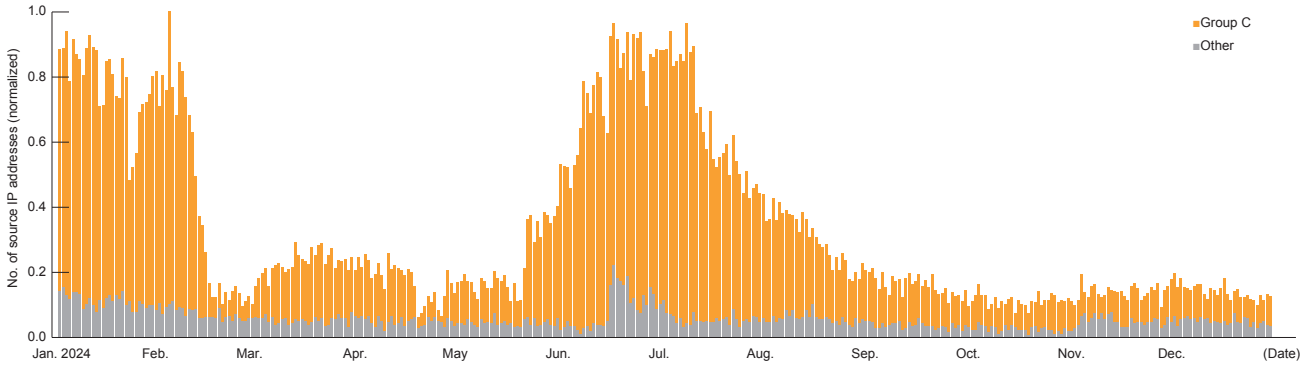


Figure 3: Number of Japanese Source IP Addresses in Different Categories Observed by IIJ's Honeypots (Sources without Mirai Characteristics)

in devices compromised by InfectedSlurs in Japan during this period. As reported by IIJ-SECT[*5] at the end of 2023, InfectedSlurs is known to exploit zero-day vulnerabilities in multiple IoT devices and also to target products primarily used within Japan, so caution is needed even with domestic-market products.

1.4 IoT Malware Analysis Support Tool mirai-toushi

IoT botnets remain widely used today—especially Mirai, the source code for which was leaked in 2016, and variants—and we observe numerous malware samples daily. Attackers do not use the original source code as-is to mount attacks. They edit the configuration and cross-compile the source code to create the malware used in their attacks. So when analyzing Mirai malware, for which the source code has been leaked, extracting the configuration data that differs from that in the original source code is key to understanding the characteristics of that particular malware.

Manually extracting configuration data for each malware sample would be very time-consuming, so using tools to automate the process is essential. While analysis tools that extract Mirai configurations do exist, they have issues in terms of only supporting a limited number of architectures or extracting only partial configurations. It was for this reason that IIJ's SOC developed its own Mirai configuration extraction tool called mirai-toushi. Below, we explain Mirai's configuration data and then describe the implementation of mirai-toushi.

1.4.1 Mirai's Configuration Data

With many types of malware, the configuration is often encrypted, and Mirai is no different here. The original version of Mirai has two types of encrypted configuration data: a password list and a table (Figure 4).

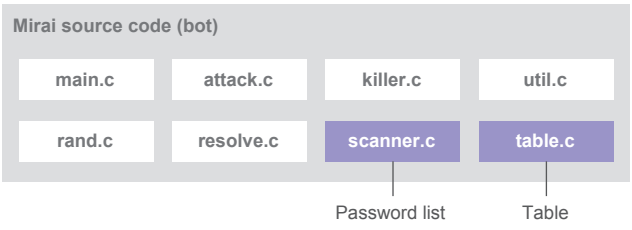


Figure 4: IoT Malware Mirai's Configuration Data

Password List

Mirai stores the password list used for Telnet scanning encrypted using XOR. The source code for password list registrations is in scanner.c. Inside scanner_init(), add_auth_entry() is called to add entries to the password list. The arguments passed to add_auth_entry() are username, password, and weight (Table 4). The username and password are encrypted with a 4-byte XOR key, but since each byte is XORed four times with a 1-byte segment of the XOR key, the process is effectively equivalent to XORing with a single-byte key. For example, if the XOR key is 0xDEADBEEF, the result is equivalent to XORing with 0x22 (byte ⊕ 0xDE ⊕ 0xAD ⊕ 0xBE⊕ 0xEF = byte ⊕ 0x22). The weight argument is the weighted probability used for random selection from the password list, and this value is not encrypted.

Table

Mirai stores various configuration data (in a table) encrypted using XOR. The configuration included in the original Mirai table comprises the C2 server domain and port number, Scan Receiver domain and port number, strings output to standard output, signatures for killing rival malware processes, commands to execute after a successful Telnet scan login, and parameters used for DoS attacks. The source code for processing table entries is in table.c. Inside table_init(), add_entry() is called to add entries to the table. The arguments passed to add_entry() are ID, data, and data length (Table 5). The ID is the value used when calling information from the table. The data are encrypted with a 4-byte XOR key but, for the same reason as with the password list, it is effectively XORed with a single byte. But since the XOR key is defined in a different location, the XOR key for the table may differ from the one used for the password list.

Table 4: Arguments to add_auth_entry()

Variable name	type	Encryptino	Description
enc_user	char*	1-byte XOR	Username
enc_pass	char*	1-byte XOR	Password
weight	uint16_t	-	Weight

*5 Mirai ashu InfectedSlurs no katsudo jokyo [Activity of Mirai variant InfectedSlurs] (https://sect.iij.ad.jp/blog/2023/12/mirai-infectedslurs/, in Japanese).

1.4.2 Implementation of mirai-toushi

We implemented mirai-toushi as a Python Ghidra Script using Ghidra's[*6] decompiler and P-Code intermediate representation (Figure 5). Ghidra is an open-source reverse engineering tool released by the US National Security Agency (NSA) and allows analysis to be automated through Ghidra Scripts written in Java or Python. In Ghidra, the target binary is converted to assembly, and P-Code is then generated from that assembly. Based on that P-Code, the binary is decompiled and C code generated. Since P-Code and the decompiled C code are architecture-independent representations, using them allows for the development of tools with cross-architecture support. And thus mirai-toushi supports eight different architectures (ARM, MC68000, MIPS, PowerPC, SPARC, SuperH4, x86, and x86_64). Below, we explain the implementations for extracting the password list and table, respectively.

■ Extracting the Password List

To extract the password list in scanner.c, we use a key extractor (scanner) to identify the XOR key and a decoder (scanner) to decrypt the password list.

First, the key extractor (scanner) identifies the XOR key used to encrypt the password list. In the source code, a 4-byte XOR key is divided into four parts and XORed four times, but compiler optimization means this is consolidated into a single XOR operation. We therefore obtain the results of decompiling each function and identify the instructions that recursively perform the single-byte XOR on each byte of the data. This single byte is taken to be the XOR key used to encrypt the password list.

Next, the decoder (scanner) decrypts the password list. Since add_auth_entry() is used to add entries to the password list, we identify where add_auth_entry() is called in the decompilation output. In this step, the decompiler sometimes does not correctly interpret the number and types of function arguments, resulting in erroneous decompilation output. To address this, we use Ghidra Script's updateFunction() to properly define the function arguments before decompiling. The first argument (username) and the second argument (password) are decrypted with the identified XOR key, while the third argument (weight) is not encrypted, so it is converted directly to a numerical

Table 5: Arguments to add_entry()

Variable name	Type	Encryption	Description
id	uint8_t	-	ID
buf	char*	1-byte XOR	Data
buf_len	int	-	Data length

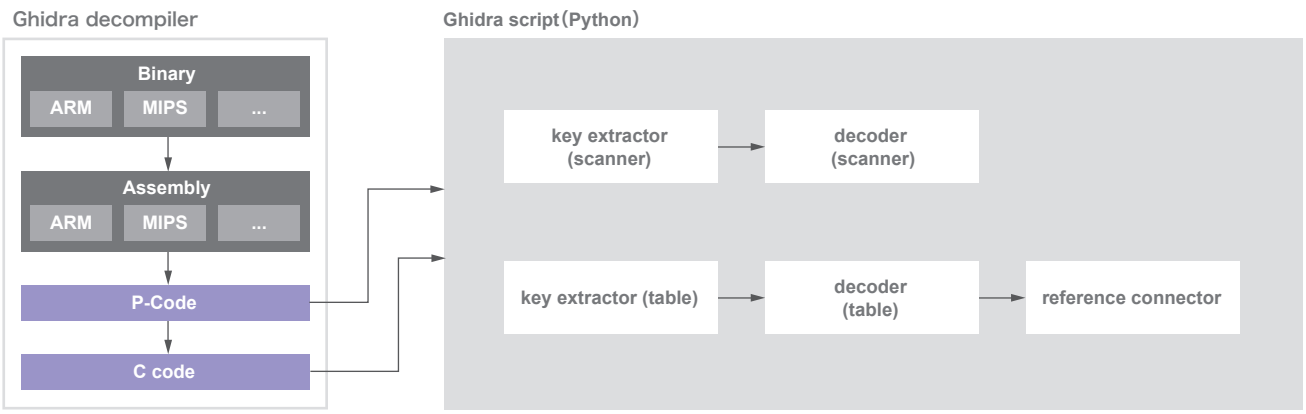


Figure 5: Overview of mirai-toushi

*6 Ghidra (https://ghidra-sre.org/).

value. The extracted password list is output in JSON format (Figure 6).

Extracting the Table

To extract the table in table.c, we use a key extractor (table) to identify the XOR key, a decoder (table) to decrypt the table, and a reference connector to identify where the table is referenced.

First, the key extractor (table) identifies the XOR key used to encrypt the table. Here, we obtain the P-Code for each function and obtain the INT_XOR instructions, which correspond to XOR operations. Since the target function performs XOR four times, we identify the function containing this procedure and obtain the four bytes used for XOR operations. These four bytes are taken to be the XOR key used for encrypting the table.

Next, the decoder (table) decrypts the table and calculates IDs for use in the reference connector. The table is populated using add_entry(), but due to compiler optimization, add_entry() is inlined. We therefore obtain the table from the contents of util_memcpy() called within add_entry(). We use updateFunction() in the Ghidra script here to properly define the function arguments before decompiling, and we then decrypt the table data in the second argument using the identified XOR key. Two-byte data entries could be port numbers, so these are decrypted as numerical values. Additionally, we calculate the ID for each data entry. In the table, each piece of data is stored as an array, with the ID serving as the index. So the ID can be calculated by subtracting the starting address of the table from the address of each data entry and then dividing that value by the data size. Data size varies by architecture. On MC68000, it is six bytes; on other 32-bit architectures, it is eight bytes; and on 64-bit architectures, it is 16 bytes.

Finally, the reference connector identifies the functions and addresses where the table is called. When the table is referenced, table_retrieve_val() is called (e.g., table_retrieve_val(TABLE_CNC_DOMAIN, NULL)), so we obtain decompilation output for each function and identify where this function is called. Since the ID value is passed as the first argument, matching this up with the information calculated via the decoder (table) lets us identify which functions and addresses call the table. The extracted table is output in JSON format, with the reference connector results given under “refs” (Figure 7).

We have published mirai-toushi on IIJ’s GitHub repository[*7], where you can find more detailed examples of the output produced when the tool is applied to samples, along with the JSON Schema. The tool can be run without additional settings or libraries in environments where Ghidra is installed, and the extracted configurations have a variety of possible applications.


```
{
  "scanner_init_func": {
    "auth_tables": [
      {
        "user": "root",
        "pass": "admin",
        "weight": 8
      },
      {
        "user": "admin",
        "pass": "admin",
        "weight": 7
      },
```


Figure 6: Example of Extracted Password List Output


```
{
  "table_init_func": {
    "tables": [
      {
        "id": 3,
        "type": "str",
        "str_data": "example.com",
        "table_addr": "080565d8",
        "refs": [
          {
            "func": "resolve_cnc_addr",
            "addr": "0804e552"
          }
        ]
      },
      {
        "id": 4,
        "type": "int",
        "int_data": 23,
        "table_addr": "080565e0",
        "refs": [
          {
            "func": "resolve_cnc_addr",
            "addr": "0804e5a9"
          }
        ]
      },
```


Figure 7: Example of Extracted Table Output

*7 mirai-toushi (https://github.com/iij/mirai-toushi).

- Inferring which device a sample targets based on the password list
- C2 servers and Scan Receiver domains
- Use of domains, IP addresses, and port numbers of C2 servers and Scan Receivers as IoCs
- Detection of new Mirai variants

This tool is only effective against samples using single-byte XOR encryption, but samples using encryption methods other than single-byte XOR have emerged in recent years, so we are looking at how we might deal with these.

1.5 Conclusion

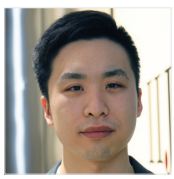
In this article, we reviewed security topics of note in 2024, discussing observational information and our analysis efforts, with a focus on DDoS attacks.

Throughout the year, more than a few reports of serious vulnerabilities that were already being exploited came out. Among ransomware attacks, we observed multiple incidents of the impact spreading through supply chains, disrupting business not only at the companies directly affected but also at their business partners. And from late 2024 to early 2025, we saw a slew of reports about DDoS attacks on companies people commonly encounter in their daily lives, attracting a lot of attention across society at large. As discussed in Section 1.3, the DDoS attack methods IIJ's SOC observed have not changed much from those used in the past, and IoT devices such as routers and IP cameras still feature prominently among the attack traffic sources. Japan is no exception to infections by IoT malware participating in DDoS attacks, and we observed an increase in attack traffic from sources that appeared to be infected with InfectedSlurs. Analyzing the attacks observed requires an analysis of the IoT malware involved, and our efforts include using mirai-toushi, our tool for streamlining the analysis of IoT malware across different platforms, which we covered in Section 1.4.

At IIJ's SOC, we will continue to analyze DDoS attacks and other threats using our Data Analytics Platform and publish our findings in wizSafe Security Signal[*1] and the IIR. We hope that you will continue to turn to these resources and that they will prove useful in your security responses and operations.

Eisei Hombu
Data Analytics Section, Security Operation Department, Advanced Security Division, IIJ

Shota Shinada
Data Analytics Section, Security Operation Department, Advanced Security Division, IIJ



Shun Morishita
Data Analytics Section, Security Operation Department, Advanced Security Division, IIJ

Shota Saito
Data Analytics Section, Security Operation Department, Advanced Security Division, IIJ