IIR

Internet
Infrastructure
Review

Jun.2025
Vol. 66

Periodic Observation Report

SOC Report

Focused Research

Development of Remote Access Services —Evolution of ID Gateway

IIJ
Internet Initiative Japan

Internet Infrastructure Review

June 2025 Vol.66

© Internet Initiative Japan Inc.

Executive Summary

I am writing this as the annual Mobile World Congress (MWC) takes place in Barcelona. As many of you know, MWC is the mobile communications industry's biggest event, hosted by the GSM Association, an industry organization comprising mobile network operators, device and equipment manufacturers, and other companies involved in mobile communications.

The theme of this year's MWC is "Converge. Connect. Create." And looking back, the themes over the past five years were "Velocity," "Connected Impact," "Connectivity Unleashed," "Connected Impact," and "Limitless Intelligent Connectivity." With this being a mobile communications industry event, connectivity remains consistently in focus, but with AI now garnering attention, as at other IT-related events, the key phrase "beyond connectivity" is also being used.

With companies around the globe apparently working hard to monetize 5G, the issue of how to go about expanding businesses other than telecommunications services is being discussed, and alongside this there is also a lot of discussion about what the infrastructure that would make such businesses possible should look like and what sort of technologies need to be developed. Creating new services not confined to the traditional notion of connectivity and new infrastructure to enable such services is an important mission for us as a telecommunications business.

The IIR introduces the wide range of technology that IIJ researches and develops, comprising periodic observation reports that provide an outline of various data IIJ obtains through the daily operation of services, as well as focused research examining specific areas of technology.

The periodic observation report for this issue in Chapter 1 is our annual SOC Report. This report looks at the major security topics over the past year that IIJ's SOC was focused on, with a deep dive into DDoS attacks this year. Many DDoS attacks were observed throughout 2024, and Japan faced a constant onslaught of DDoS attacks from the end of 2024 through early 2025, with incidents of this impacting on everyday life also being reported. The report discusses the methods used and the evolving trends in these DDoS attacks, and then takes a detailed look at mirai-toushi, an analysis support tool for Mirai and variants, which are used in many DDoS attacks.

Our focused research report in Chapter 2 takes you behind the scenes of the ID Gateway Service, which IIJ provided for 26 years. Today, with the evolution of mobile communication networks and cloud services, we have access to the information and computing resources we need anytime, from anywhere. When the ID Gateway Service was launched in 1998, dial-up Internet access was still the mainstream, and both information and computing resources resided within corporate networks. Various improvements were made to the ID Gateway Service over the years in response to technological advances and societal changes, and the article walks through the history of IIJ's in-house software that supported this service.

Through activities such as these, IIJ continues striving to improve and develop its services on a daily basis while maintaining the stability of the Internet. We will continue to provide a variety of services and solutions that our customers can take full advantage of as infrastructure for their corporate activities.

Junichi Shimagami

Mr. Shimagami is a Director and Executive Vice President and the CTO of IIJ. His interest in the Internet led to him joining IIJ in September 1996. After engaging in the design and construction of the A-Bone Asia region network spearheaded by IIJ, as well as IIJ's backbone network, he was put in charge of IIJ network services. Since 2015, he has been responsible for network, cloud, and security technology across the board as CTO. In April 2017, he became chairman of the Telecom Services Association of Japan's MVNO Council, stepping down from that post in May 2023. In June 2021, he also became a vice-chairman of the association.

© Internet Initiative Japan Inc.

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/).

© Internet Initiative Japan Inc.

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.

© Internet Initiative Japan Inc.

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.

© Internet Initiative Japan Inc.

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).

© Internet Initiative Japan Inc.

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.

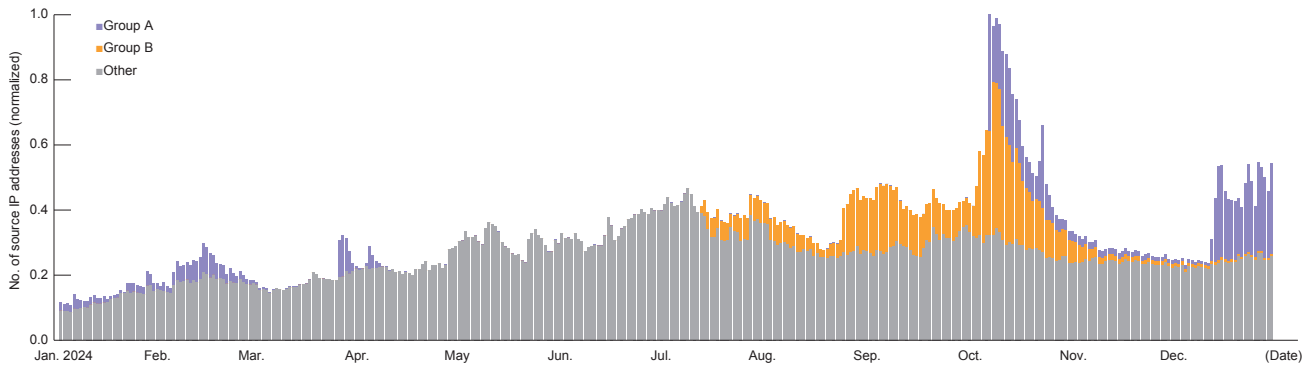


Figure 1: Number of Source IP Addresses (Normalized) in Different Categories Observed by IIJ's Honeypots (Sources with Mirai Characteristics)

*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.

© Internet Initiative Japan Inc.

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

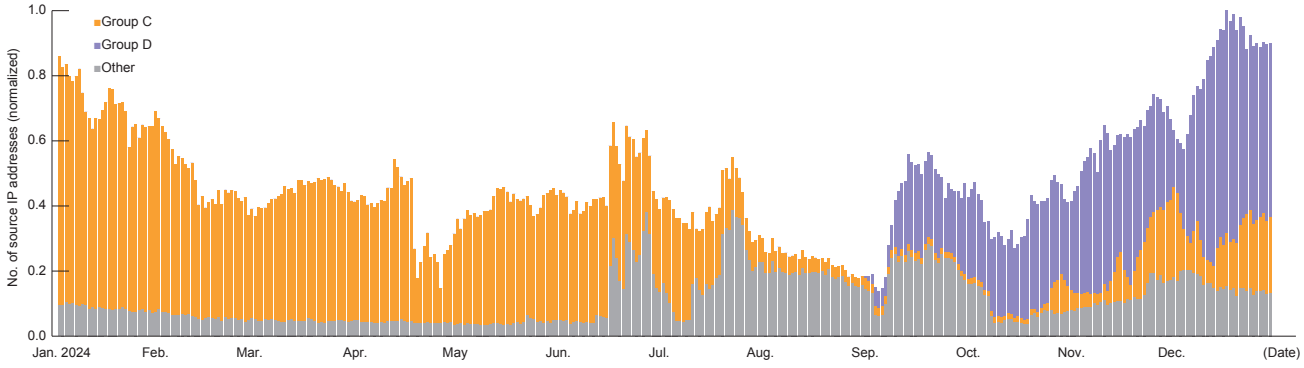


Figure 2: Number of Source IP Addresses (Normalized) in Different Categories Observed by IIJ's Honeypots (Sources without Mirai Characteristics)

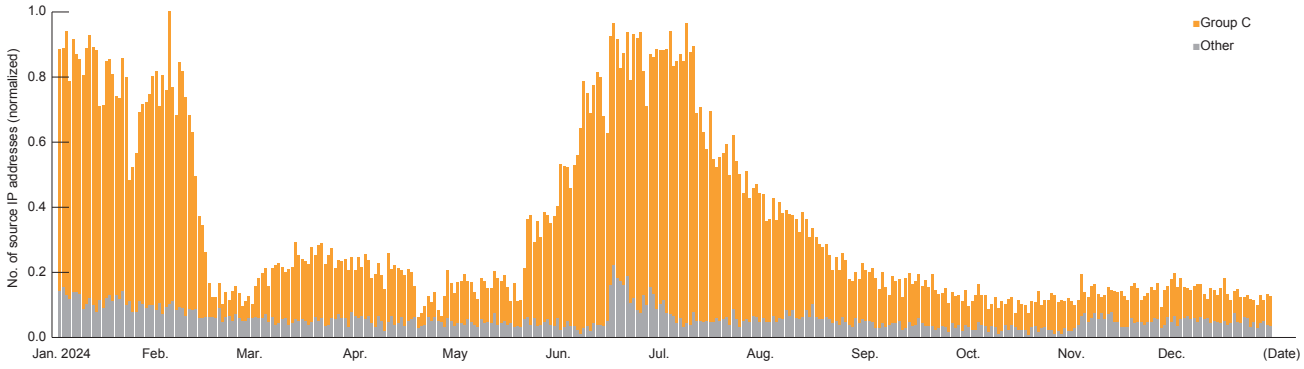


Figure 3: Number of Japanese Source IP Addresses in Different Categories Observed by IIJ's Honeypots (Sources without Mirai Characteristics)

© Internet Initiative Japan Inc.

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).

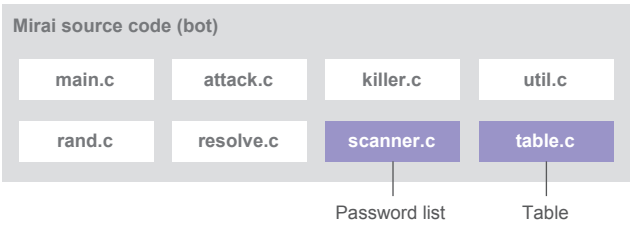


Figure 4: IoT Malware Mirai's Configuration Data

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).

© Internet Initiative Japan Inc.

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

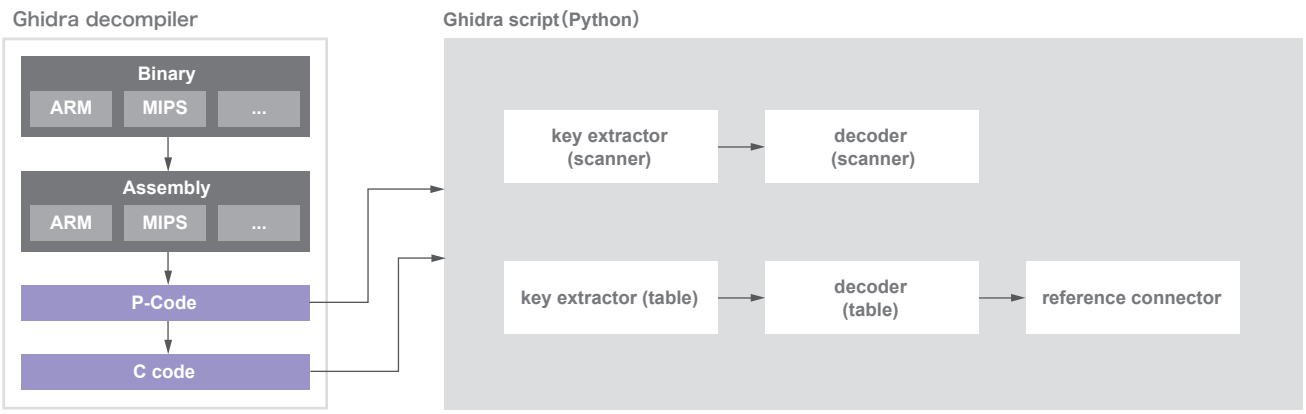


Figure 5: Overview of mirai-toushi

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

© Internet Initiative Japan Inc.

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).

© Internet Initiative Japan Inc.

- 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

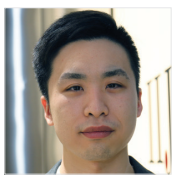
In this article, we reviewed security topics of note in 2024, discussing observational information and our analysis efforts, with a focus on DDoS attacks.

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

© Internet Initiative Japan Inc.

Development of Remote Access Services —Evolution of ID Gateway

2.1 Introduction

At the end of September 2024, the ID Gateway Service IIJ had been providing for around 26 years was finally discontinued. Having begun in December 1998[*1], it was one of the longest-running services in IIJ's history.

The ID Gateway Service was a service that provided remote access, a technology for connecting to computers in remote locations and something that is essential for the now widespread practice of remote work. VPN technology is now considered essential for remote access, but the ID Gateway Service did not use VPN when it was first released in 1998. Even site-to-site VPNs were still only just starting to gain traction at the time, and while there were client-side VPN implementations, they could hardly have been called practical. Here, we look back on developments from this era through to the present, where VPN has become an essential technology in our society, alongside the history of IIJ's ID Gateway Service.

2.2 The Early Days of Remote Access Services

Around the time IIJ launched the ID Gateway Service in 1998, remote access meant installing network devices for dial-up access, such as the Ascend MAX, within an organization, dialing into it, and establishing an IP connection via PPP. The key with these dial-up routers was that they needed to provide internal connectivity, because if you simply wanted to connect to the Internet, you could connect to an ISP. At the time, however, such devices could not implement access controls such as firewalls, and there were apparently cases in which they connected to internal networks with virtually no restrictions.

IIJ, meanwhile, had been recommending to users that they keep their internal networks separate from the Internet, with firewalls at the boundary. Remote access that provided a back door around that separation was a vulnerability in and of itself, and IIJ believed that when providing remote access as a service, it should be incorporated within the same access control policy as firewalls. Moreover, instead of putting dial-up routers on the user's network, the system was designed to use IIJ's dial-up service. And thus ID Gateway 1.0 was developed to use existing Internet connectivity as the means for remote access while requiring firewall-like access control for access to internal networks.

2.3 ID Gateway 1.0

ID Gateway 1.0 was developed based on an application-level gateway firewall. The base OS was BSD/OS 3.1. With an application-level gateway firewall, a proxy intercepts all communications passing through the firewall and only relays subsequent communications if permitted according to access control rules. Since communications are relayed at the application level, protocols such as HTTP and SMTP are interpreted before the communications are forwarded. The access control portion of this application-level gateway firewall was extended to allow or deny communications for individual destinations based on the PPP account username (Figure 1).

We anticipated that it would often be necessary to grant permission for each specific destination rather than using a broad rule to allow access to the entire network simply because a user had been authenticated. This was because it had been pointed out through design reviews and internal beta testing that it would be crucial to protect not only connecting users but also the services within organizations to which communications were being directed. The team therefore aimed to implement fine-grained access control.

Let's examine the details a little more closely. As indicated in the lower left of Figure 2, when a user access request comes in, it is passed through the operating system to the relay program (proxy). At this point, the user's source IP address, destination IP address, and port number are passed to the relay program, which then queries the ID authentication daemon to determine whether the communication is permitted under the access control rules. Since

*1 "Launch of ID Gateway Service," iij.ad.jp, IIJ (October 2, 1998) (https://www.iij.ad.jp/news/pressrelease/1998/pdf/gateway.pdf, in Japanese).

© Internet Initiative Japan Inc.

access control rules are in Link-ID form (PPP account user ID), the ID authentication daemon needs to convert the source IP address to a Link-ID. If a previous query result exists in the cache, the ID authentication daemon immediately completes the conversion from source IP address to Link-ID using the cached information. If no such result exists in the cache, it sends a query to what is called IIJ's ID server. Communication with the ID authentication daemon and the ID server uses a protocol developed by IIJ called the Link-ID protocol.

Once the ID authentication daemon obtains the Link-ID, it determines whether the communication is permitted for the resolved user by assessing the access control rules and returns an allow or deny value to the relay program. This ensured that remote access was only allowed when permitted under the access control rules.

2.4 ID Gateway 2.0

With ID Gateway 2.0, released in September 2000, we overhauled the configuration user interface from a console application to a graphical user interface (GUI) called ID Gateway Policy Manager. After extensive discussion about how to make the access control rules more intuitive to configure, we eventually designed a unique spreadsheet-like interface for the access control rule panel in which users

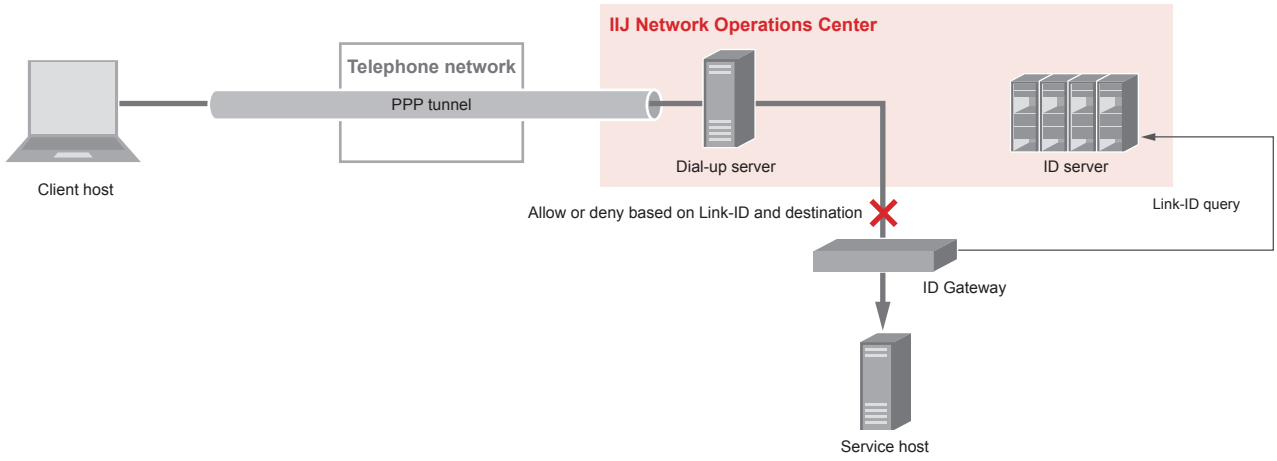


Figure 1: Overall Structure of ID Gateway 1.0Figure 1: Overall Structure of ID Gateway 1.0

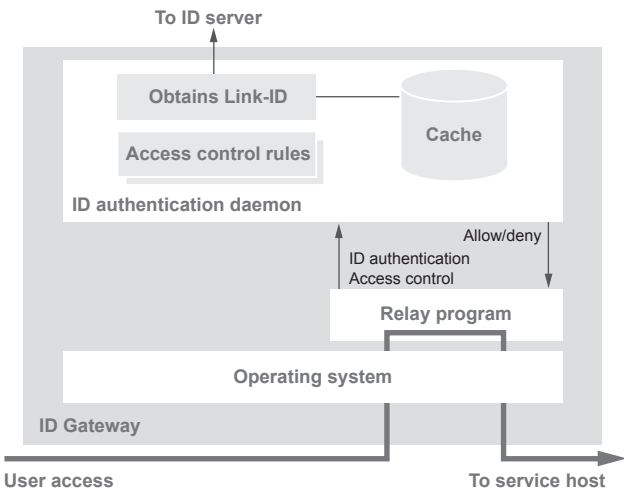


Figure 2: ID Gateway 1.0's Access Control

© Internet Initiative Japan Inc.

could mark cells with ○ or ×, as shown in Figure 3. We also switched to NetBSD 1.4 for the base OS.

2.5 ID Gateway 3.0

ID Gateway 3.0, released in April 2002, provided support for VPN connections based on PPPs such as L2TP/IPsec and PPTP. At the time, however, we had to overcome several challenges to make this happen.

We first needed to implement tunneling protocols L2TP and PPTP. Existing open-source implementations at the time used a process-forking model that used one process per tunnel, and at the anticipated scale of the ID Gateway Service, this would have involved hundreds of processes, which would have been impractical due to memory constraints. We had implemented the daemons on ID Gateway using an event-driven model since version 1.0, and the tunneling protocols also needed to be implemented with an event-driven model. We did this from scratch using C++.

Next, for the IPsec part of L2TP/IPsec, we were able to use the IKE and IPsec implementations from WIDE's KAME project*2, which had been incorporated into NetBSD. There was one problem, however. To use IPsec through NAT, you need to implement IPsec NAT-T, but the version incorporated into NetBSD 1.4 did not have this. Unless IPsec NAT-T was implemented, the UDP checksums would not match, and even if we were to bypass that, the problem was that only one host behind NAT would be able to connect. On the ID Gateway Service, we called this the "first come, only served" problem. For ID Gateway 3.0, we decided to accept this "first come, only served" problem as a given restriction and only solve the UDP checksum mismatch issue. Here, we decided to skip UDP checksum verification since ESP's HMAC is already used to perform a check on incoming transmissions. For outgoing transmissions, we set the optional UDP checksum field to 0, meaning it would not be used, so that checksum verification would also be skipped on the peer host.

Issues can potentially arise with PPTP when traversing NAT. PPTP uses TCP for control and GRE for transmitting data. There are no NAT traversal issues with the control portion since it uses TCP, but issues similar to those with L2TP/IPsec can arise with the GRE-based data transmissions. But implementations that used the Call-ID

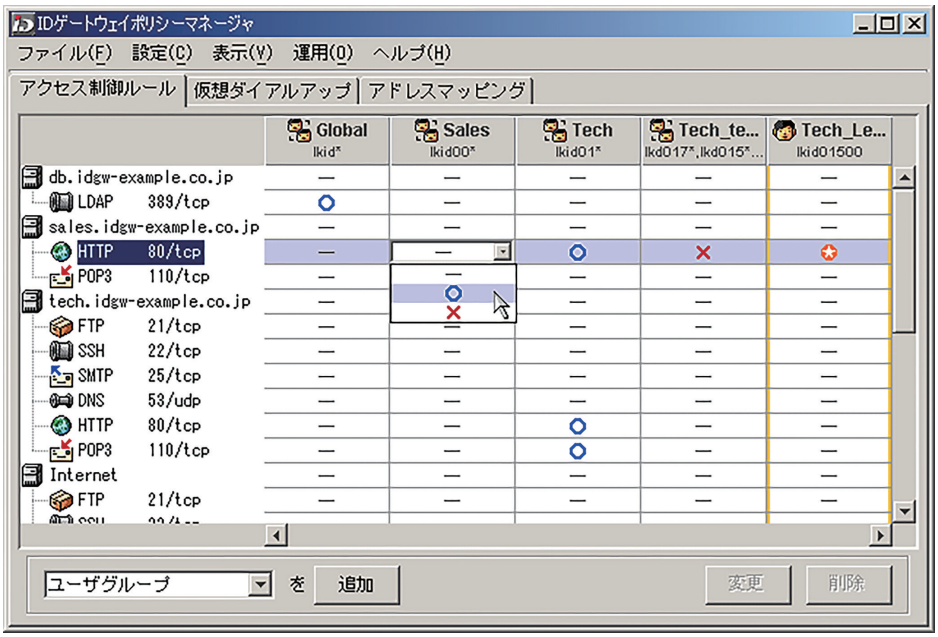


Figure 3: ID Gateway 2.0's Policy Manager

*2 KAME Project (https://www.kame.net/index.html).

© Internet Initiative Japan Inc.

in the GRE header for NAT masquerading had started to become widespread, particularly in consumer routers. We thus expected the "first come, only served" problem to be less of an issue with PPTP than with L2TP/IPsec. Additionally, PPTP effectively requires MPPE for packet encryption, and MPPE uses the proprietary encryption algorithm RC4. We therefore obtained a license to use RC4 from RSA Data Security.

For the PPP implementation, we decided to use FreeBSD's ppp, which was based on iij-ppp[*3] and had been extended with Multi-PPP to enable multiple PPP connections over multiple lines, allowing a single process to act as the endpoint for multiple PPP connections, plus it had originally been developed by IIJ.

We also had to figure out how to handle VPN authentication. For a typical VPN service, you would first connect to the user directory service for authentication. But with ID Gateway up to version 2.0, users had been using dial-up connections, so existing users were already on IIJ's dial-up service, and we had issued PPP accounts to connecting users with access control rules being configured for those accounts. We realized that using the same accounts for VPN authentication would make it possible to use various existing components as well, so that is how we implemented it. Authentication requests from the PPP daemon on the ID Gateway were proxied by the ID authentication daemon and authenticated by the RADIUS server on IIJ's ID server. This allowed users to configure a single PPP account in the same way both for dial-up and for VPN entries via L2TP/IPsec and PPTP. As Figure 4 shows, we were also able to implement the access control mechanism in almost the same manner as before. Because we designed the system so that VPN would be handled in the same way as conventional dial-up, within the ID Gateway Service we called this virtual dial-up (VDIP) and referred to conventional dial-up as real dial-up.

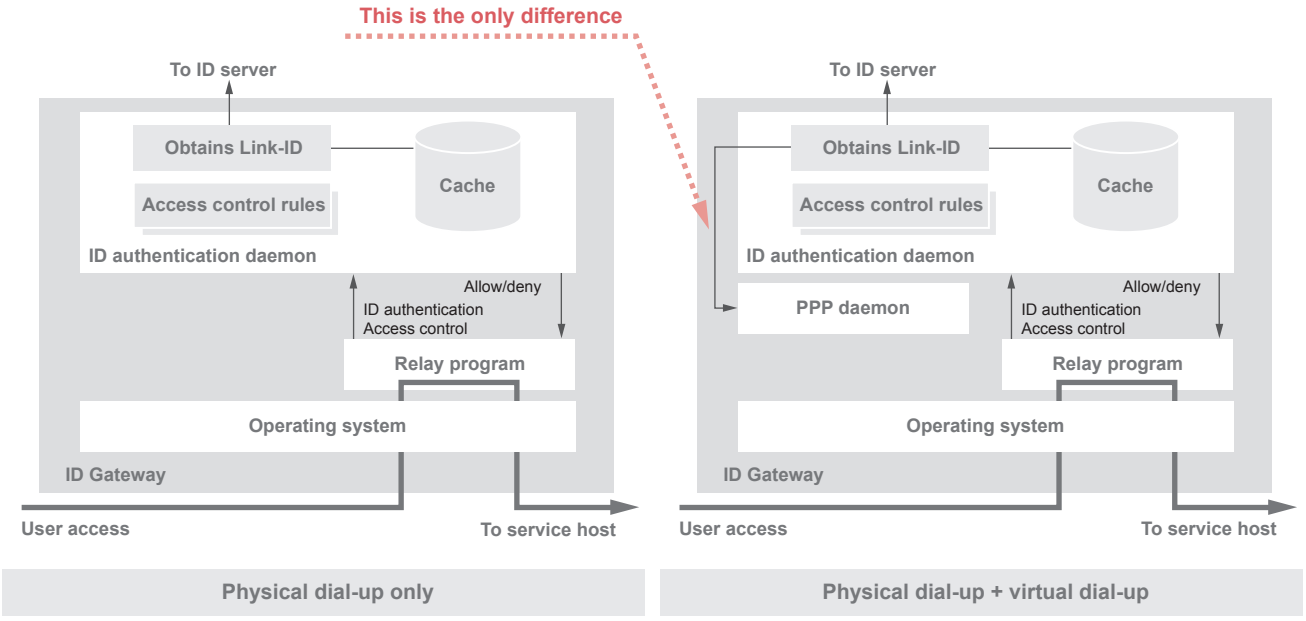


Figure 4: Dial-up Method and Access Control

*3 An open-source PPP implementation developed by Toshiharu Ohno and others at IIJ. It was widely used from the late 1990s to the early 2000s, particularly among BSD users.

© Internet Initiative Japan Inc.

2.6 ID Gateway 4

With ID Gateway 4.00, released in July 2005, we added an authentication server integration feature to allow the use of a RADIUS or LDAP server from the user's network—which could be an Active Directory environment, for example—as the virtual dial-up authentication server. We also overhauled the reporting system and added a rule viewer function.

In ID Gateway 4.02, released in April 2006, we implemented a new VPN protocol called SSL Dial-up (SSLDIP). As noted earlier, L2TP/IPsec and PPTP can experience problems when used across NAT. And these protocols are sometimes completely unusable when only a limited range of ports (e.g., HTTP, HTTPS, DNS) can be used, such as from within restricted organizational networks or networks available at overseas hotels. Meanwhile, SSL-VPN products were beginning to appear on the market, and since SSL-VPN products using SSL for transport are entirely unaffected by these problems, we knew we had to implement equivalent functionality in ID Gateway.

We designed SSLDIP to use SSL (now TLS) as the transport layer and create an L2 tunnel between the client and ID Gateway, with PPTP running on top of that. The aim was to use PPP as the common base for the tunnels that were ultimately created, allowing us to otherwise use the same mechanisms as before, including for authentication and access control. For the L2 tunnel, we decided to use the open-source OpenVPN, and we developed a Windows client to make configuring the system and managing connections easy.

For ID Gateway 4.02, we also rewrote the implementations of the L2TP/IPsec and PPTP server functions from scratch and created a new daemon. In the original implementation, the VPN tunnel processing code was written in C++ rather than C, which caused problems when porting to embedded environments like the SEIL series[*4] (IIJ's series of proprietary high-performance routers for enterprises). The PPP portion had also previously been a separate program, and we simplified this by rewriting it with the bare minimum functionality and incorporated it into the same program as the VPN, improving performance and maintainability. This program was npppd[*5], which would later be incorporated into OpenBSD. Further, the RADIUS portion repurposed code originally used in the ID server, and this served as the prototype for what is now the OpenBSD RADIUS library[*6].

2.7 ID Gateway 5

In ID Gateway 5.00, released in March 2008, we added a client authentication feature. This provided additional authentication on top of the initial VPN connection authentication, making it equivalent to what is now called multi-factor authentication. This feature addressed user requests for the ability to restrict the range of devices able to connect and for contingencies to mitigate the impact of password leaks. We extended the ID authentication daemon and relay program to implement the terminal authentication feature. After a VPN connection was established and up until the point that terminal authentication was completed, the ID Gateway terminal authentication feature restricted access to DNS and authentication pages only, and once terminal authentication was successful, it would switch to the full set of access control rules. This method of

*4 SEIL (https://www.seil.jp/, in Japanese).

*5 src/usr.sbin/npppd/, github.com, GitHub (https://github.com/openbsd/src/tree/8b2d863473/usr.sbin/npppd/).

*6 src/lib/libradius/, github.com, GitHub (https://github.com/openbsd/src/tree/8b2d863473/lib/libradius/).

© Internet Initiative Japan Inc.

access restriction is equivalent to what is now called a captive portal. The authentication mechanism worked by redirecting web access from the client to an authentication page. An applet on the authentication page would send the client device's MAC address to the ID Gateway, which would then check it against the list of registered MAC addresses stored in the ID Gateway's database to verify that the connection was coming from a legitimate device owned by an authorized VPN user.

In addition, to enable hot standby, we added VRRP functionality by porting it from SEIL. We also added support for EAP authentication in the authentication server integration feature.

With ID Gateway 5.02, released in December 2009, we brought the operating system source code and the SEIL/X series source code together to unify the codebase. This enabled IPsec NAT-T and thus resolved the long-standing issue of not being able to accommodate multiple L2TP/IPsec users behind NAT. We switched to NetBSD 3.1 for the base OS.

2.8 ID Gateway 6

ID Gateway 6.00, released in November 2011, added support for SSTP as a new tunneling protocol. SSTP is a protocol developed by Microsoft that operates over TLS and tunnels PPP frames in a manner similar to L2TP/IPsec and the like. As it is TLS-based, there are no issues connecting from behind NAT, and since the client is included as standard in Windows, we did not need to distribute our own client. And because it is PPP-based, we were able to reuse the existing authentication and access control components as is. Although SSTP is an open protocol, we needed to obtain a license from Microsoft for commercial use.

2.9 The End of ID Gateway Development and Challenges Faced

The initial technical challenges in terms of connecting from behind NAT that we faced when we launched the virtual dial-up service were resolved with the addition of support for IPsec NAT-T in version 5.02 and SSTP in version 6.00. Load per user was continuing to increase year by year, however, meaning we could no longer achieve the desired performance on a single gateway.

The first conceivable factor here is that ID Gateway was, from the outset, application gateway software, not a router. All communications were relayed at the application layer, and it basically did not perform IP forwarding. This inevitably results in higher loads compared with IP-level forwarding. What we should have had was a mechanism for offloading to IP-level forwarding on a case-by-case basis.

Another factor was that the relay program providing the application gateway functionality was designed to run as a single process, so it was unable to make use of multiple CPUs even when they were present, and it had issues with multi-core support.

The final factor to consider is the base OS kernel. The base OS of the final version of ID Gateway is NetBSD 3.1, but work on multi-core support for its network stack had not yet started, and thus similar to the relay program, it

© Internet Initiative Japan Inc.

was unable to use multiple CPUs even when present, and it had issues with multi-core support.

These seemingly separate issues are in fact nothing more than the manifestation of software obsolescence. Modern software continues to evolve across the globe via the Internet, and systems are bound to become outdated if neglected. Looking back, we now realize that the issues with the ID Gateway software stem from us not having taken steps to address this inevitable software obsolescence.

We determined that extending ID Gateway to resolve these issues would be difficult for various reasons, and so we decided to discontinue development of the ID Gateway software and pass the torch to its successor service, IIJ GIO Remote Access Service, along with Tornado, a new gateway OS developed at IIJ.

2.10 IIJ GIO Remote Access Service and Tornado

In February 2013, we launched a new remote access service called IIJ GIO Remote Access Service (GAM). GAM is a cloud-based service with the VPN gateway running in IIJ's cloud. For the VPN gateway, we use Tornado, a system newly developed at IIJ to replace ID Gateway.

Tornado is IIJ's in-house gateway software integrating network-related software functions within IIJ, developed based on OpenBSD. We needed the base OS for the successor to ID Gateway to provide enterprise firewall-level packet filtering capabilities and the ability to implement kernel extensions for transparent proxies, and OpenBSD met all these requirements at the time development began in 2011. Moreover, from the outset it also provided features we had wanted for ID Gateway, such as socket splicing, which moves the work of relaying packets from the application level back into the kernel, and VRF for policy routing and the like. It also incorporated the npppd daemon from ID Gateway.

In Tornado, we replaced the relay program with a new multi-core compatible daemon. And in the access control rules, we made it possible to use a single configuration item to switch between application-level relaying and IP-level forwarding through packet filtering.

A major difference between ID Gateway and Tornado is that while ID Gateway was software exclusively for the ID Gateway service, Tornado is general-purpose software. It is designed so that service-specific functions that cannot be standardized are developed as optional packages.

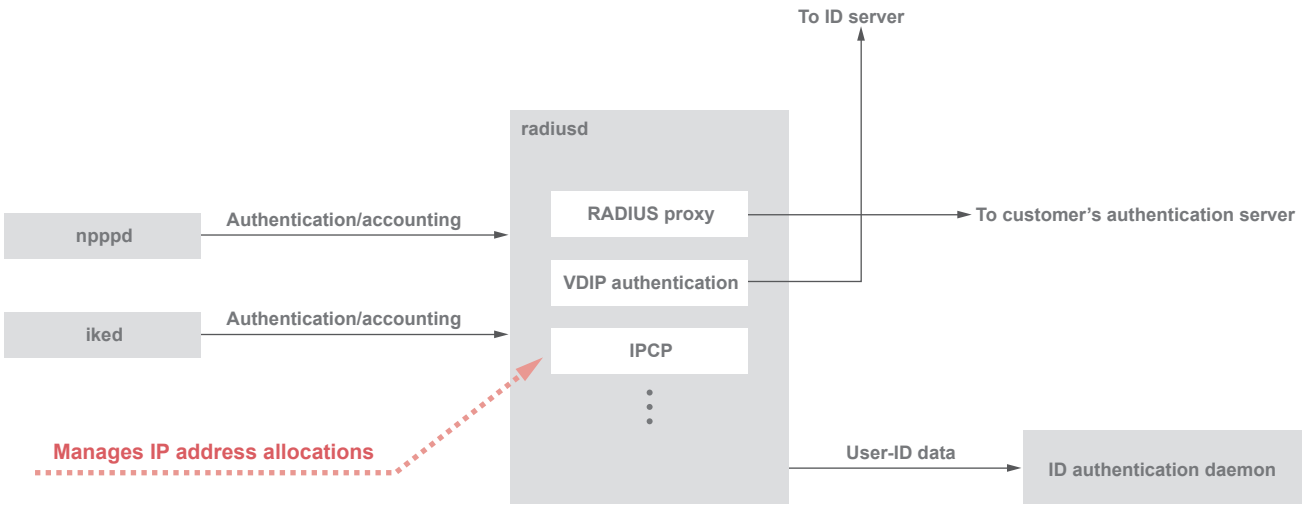


Figure 5: Overview of Connections Using IKEv2

© Internet Initiative Japan Inc.

Service-specific features are subject to a relatively high amount of churn, with new features constantly being desired while old ones become obsolete. We designed the system so that such features could be added and removed easily.

In December 2024, GAM added support for connections using the new VPN protocol IKEv2. As IKEv2 is not a PPP-based protocol, the authentication and access control mechanisms used by other protocols could not be used with the IKEv2 daemon available in OpenBSD at that time. Creating a system entirely separate from the PPP-based one to solve this problem would have meant two different methods had to be managed and maintained. With Tornado, we decided to standardize by adding RADIUS authentication and accounting capabilities to the IKEv2 daemon. We went with a unified authentication system in RADIUS and, as shown in Figure 5, implemented the system such that a local RADIUS daemon centrally manages the allocation of IP addresses to VPNs, and the access control mechanism thus works in the same way for both PPP-based VPNs and IKEv2.

Internally, we use Tornado version 4.5, which is based on an OpenBSD version released about a year ago, so we are using a relatively recent version. Having learnt from the ID Gateway experience, we now update the base OS version regularly as part of continuous integration, so we no longer end up being stuck with stale versions of the base OS.

2.11 Conclusion

Looking back, ID Gateway was never just a remote access service. It provided security features whereby users were authenticated via PPP accounts and only able to engage in the communications permitted under the access control rules for their authenticated IDs. With ID Gateway 5, we also added device-level authentication. These features constitute what is now called a Zero Trust Architecture (ZTA).

The history of ID Gateway and the subsequent transition to IIJ GIO Remote Access Service is also a story of in-house software development at IIJ. Tornado is IIJ's current infrastructure software and successor to ID Gateway. Looking forward, we will continue to work with new technologies and strive to provide even better services that address the changing needs of users as well as changes in the broader landscape.

Masahiko Yasuoka
System Development Section 1, Applied Technology Development Department, System Development Division, Network Services Business Unit, IIJ
Mr. Yasuoka joined IIJ in 1998. After developing ID Gateway and other systems, he proposed and developed Tornado, a gateway OS integrating the functionality of IIJ's internal software. He continues to develop and maintain this system today.

© Internet Initiative Japan Inc.

June 2025 Vol.66

Internet Initiative Japan

About Internet Initiative Japan Inc. (IIJ)

IIJ was established in 1992, mainly by a group of engineers who had been involved in research and development activities related to the Internet, under the concept of promoting the widespread use of the Internet in Japan.
IIJ currently operates one of the largest Internet backbones in Japan, manages Internet infrastructures, and provides comprehensive high-quality system environments (including Internet access, systems integration, and outsourcing services, etc.) to high-end business users including the government and other public offices and financial institutions.
In addition, IIJ actively shares knowledge accumulated through service development and Internet backbone operation, and is making efforts to expand the Internet used as a social infrastructure.

The copyright of this document remains in Internet Initiative Japan Inc. ("IIJ") and the document is protected under the Copyright Law of Japan and treaty provisions. You are prohibited to reproduce, modify, or make the public transmission of or otherwise whole or a part of this document without IIJ's prior written permission. Although the content of this document is paid careful attention to, IIJ does not warrant the accuracy and usefulness of the information in this document.

©Internet Initiative Japan Inc. All rights reserved.
IIJ-MKTG020-0064

Internet Initiative Japan Inc.

Address: Iidabashi Grand Bloom, 2-10-2 Fujimi, Chiyoda-ku,
Tokyo 102-0071, Japan
Email: info@iij.ad.jp URL: https://www.iij.ad.jp/en/