4 Cyber-Security Technologies: Live Network Monitoring and Analysis Technologies

4-1 NIRVANA-Kai: A Real-time Visual SIEM System Against Targeted Attacks

Yu TSUDA, Nobuyuki KANAYA, Takashi TOMINE, Masaki KAMIZONO,
Masato JINGU, Yaichiro TAKAGI, and Koei SUZUKI

Targeted attacks, a specific type of cyber-attacks targeted to a specific organization, are recognized as serious social concerns. Targeted attacks cannot be observed by large-scale darknet monitoring systems like "NICTER". Therefore, against targeted attacks, methods for rapidly detecting attackers' malicious activities are required on the assumption that the attackers have intruded into the internal network of an organization already. In this paper, we present NIRVANA-Kai which is a visual platform to observe live network traffic and to aggregate and analyze various security alerts.

1 Introduction

Various cyber-attacks are being generated via the Internet and becoming a large problem for society. For example, new malware is manufactured daily and is sent out by attackers to infect computers, and these go on to expand the infection by transmitting it to other computers. Moreover, there are many different types of cyber-attacks, including Denial of Service attacks (DoS attacks), which force a denial of service on their targets, attacks designed to steal confidential information, and those intended to destroy hardware.

Up until now at the Network Security Research Institute, research development has been conducted on NICTER, which works to comprehensively monitor and analyze trends in cyber-attacks that have wide-ranging effects on the Internet [1][2]. NICTER monitors unused Internet IP address space (i.e. darknet) on a large scale, and by monitoring and analyzing incoming packets, is able to scan for signs of attempts by self-propagating malware to expand infections and scan for responses to DoS attacks (i.e. backchatter).

On the other hand, "targeted attacks"—a type of cyber-attacks targeting a specific organization—have become a huge threat. Because this type of attack has a distinct target its signs cannot be detected with a darknet monitoring system like NICTER. Thus, the present authors developed NIRVANA-Kai, a system based on the technologies cultivated through NICTER and which is able to monitor, analyze, and visualize organizations' internal live network traffic in real time. NIRVANA-Kai is an integrated analysis platform that aggregates and visualizes [i] traffic flowing through live networks and [ii] alerts given out by security appliances installed inside the organization.

This report is comprised in the following manner: in Section **2** there is an explanation of NIRVANA, one of the core technologies of NIRVANA-Kai. Then, in Section **3**, there is a discussion of items ranging from NIRVANA-Kai's visualization user interface to the components that operate in NIRVANA-Kai's backend. In Section **4**, actual cases of security operations at NICT using NIRVANA-Kai are discussed, and finally in Section **5**, is a summary and a discussion of the future outlook.

2 NIRVANA: A real-time network traffic visualization system

As preceding technology to NIRVANA-Kai, the NIRVANA visualization system acts to support network management and operation [3]. NIRVANA is a system that allows for the real-time visualization of massive network traffic in live networks. With NIRVANA's visualization it is possible to picture network traffic in units of packets and to indicate flow rate. Furthermore, by making use of to-

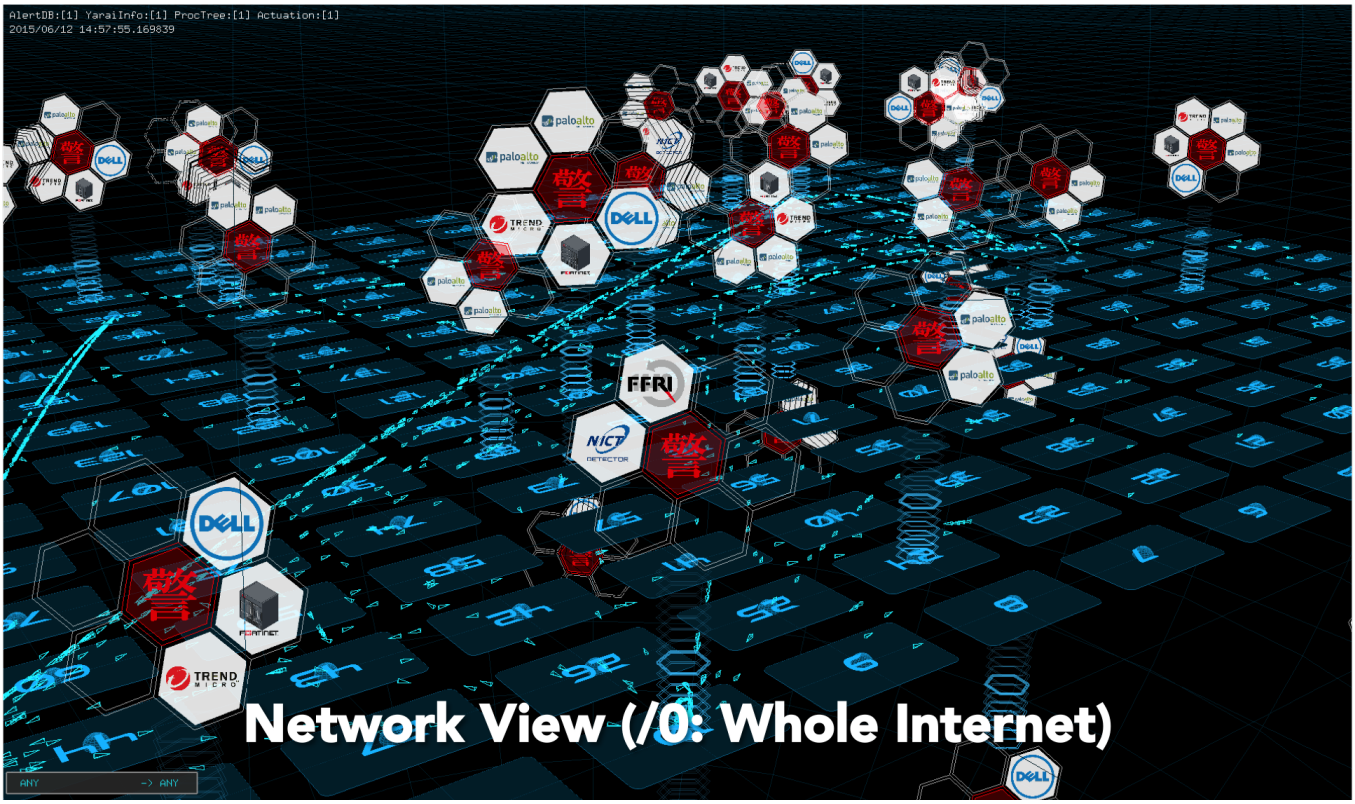

Fig. 1 Visualizing traffic and alerts in NIRVANA-Kai

pologies and network routing information, it is possible to visualize the flow of traffic between different network devices.

Through NIRVANA, functionalities for network management and operation are implemented to reduce workload for network administrators and reduce management costs. However, in the actual management of networks, there needs to be swift and appropriate response to security incidents such as [i] cyber-attacks from outside the organization and [ii] internal malware infections. In particular, as a strategy for combatting targeted attacks, which became a problem for society coming into the 2010s, it became indispensable to not only have security appliances to provide protection in perimeter environments between the organization's interior and exterior—such as firewalls and intrusion detection systems—but also to quickly discover activities of attackers from within the organization. Due to this kind of demand, research development has been advanced on NIRVANA, a platform for allowing the implementation of swift security operations by integrating, analyzing, and visualizing the many types of information that can be drawn from live networks.

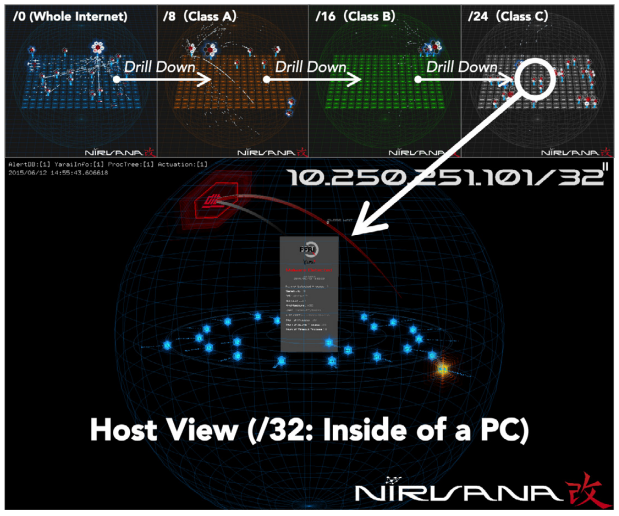

Fig. 2 An image of drill down

3 NIRVANA-Kai: A real-time visual SIEM against targeted attacks

3.1 An overview of NIRVANA-Kai

NIRVANA-Kai is a platform that assists network administrators in swiftly discovering security incidents by integrating the network traffic visualization technologies cultivated in NIRVANA with alert information from security appliances installed inside organizations.

Figure 1 shows the visualization of network conditions

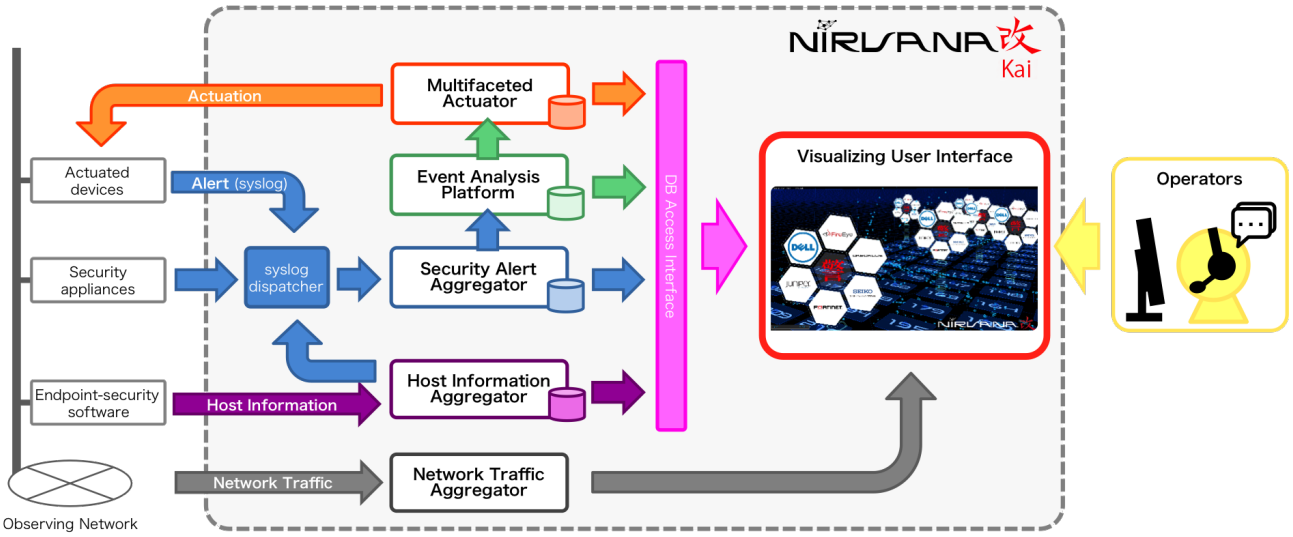

Fig. 3 The system composition of NIRVANA-Kai

using NIRVANA-Kai. This shows the entirety of the IPv4 Internet space (/0 network), and the panel listing values from 0 through to 255 corresponds to a Class A address block. The trigonal pyramid objects that jump between the panels show the sending and receiving of the packets in that space. The flower-shaped objects at the top of the address block panel visualize the alerts aggregated by the different types of security appliance. If any of the individual parts that look like flower petals correspond with a security appliance, this means that an irregularity has been detected within that address block.

In NIRVANA-Kai there is a "drill down function" which allows a drilling down from big Class A address (/8 network), Class B address (/16 network), and Class C address (/24 network) blocks from the entirety of the Internet space, down to small address blocks with an alert source, and finally arriving down at the interior of one host (/32 network). Figure 2 shows an example of drill down.

In the case of NIRVANA-Kai, in addition to the previously introduced visualization user interface operated by a security operator, there are also the components (as listed below) that operate at the back-end and execute actions such as the aggregation and analysis of information needed for visualization. Figure 3 shows the composition of the NIRVANA-Kai system.

- Network Traffic Aggregator
- Security Alert Aggregator
- Host Information Aggregator
- Event Analysis Platform
- Multifaceted Actuator

Each of the components is implemented so it operates independently, and the information it accumulates in its individual databases is visualized by way of a database access interface. From among these components, the Network Traffic Aggregator had already been implemented as a primary function of NIRVANA. In this report, the next section onward will explain the following components, which have been uniquely implemented in NIRVANA-Kai: the Security Alert Aggregator, the Host Information Aggregator, the Event Analysis Platform, and the Multifaceted Actuator.

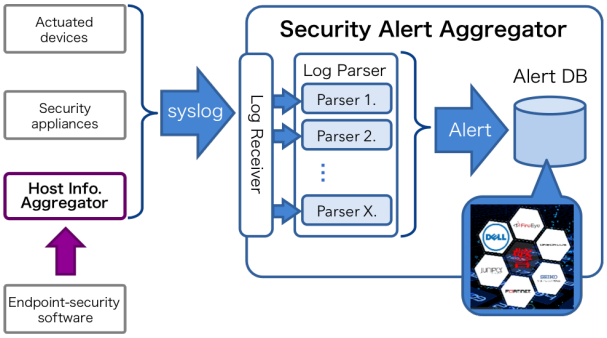

Fig. 4 Overview of the security alert aggregator

3.2 Security alert aggregator

One type of information that NIRVANA-Kai aggregates is "alert information," which includes attack detection information from devices introduced inside an organization and malware detection information from endpoint-security software. NIRVANA-Kai's Security Alert Aggregator specifically aggregates syslog messages, which are standardly used when the log message of a device is transferred.

Figure 4 shows an overview of the Security Alert Aggregator. Because a log message via syslog differs in format for each device, the received syslog message must

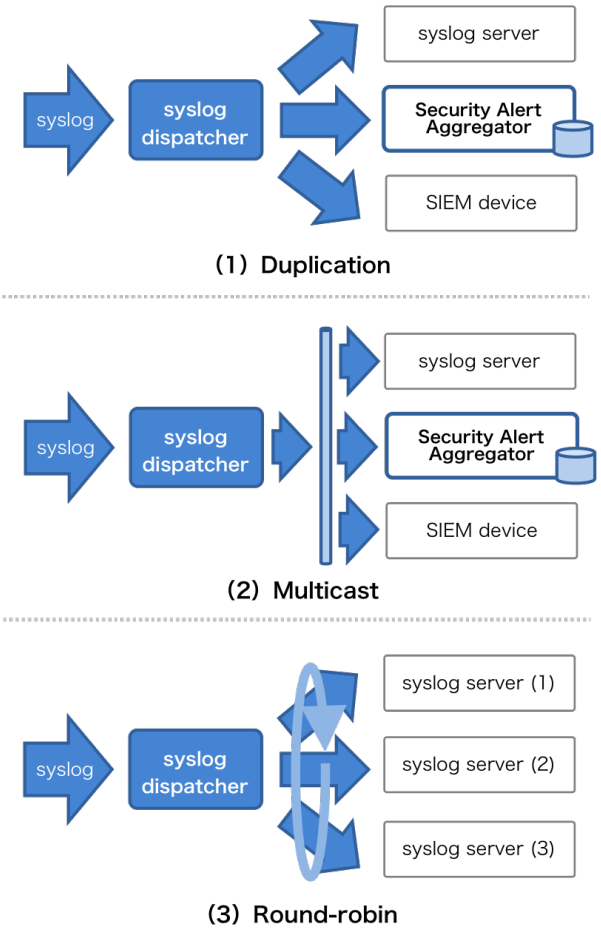

Fig. 5 Functions of the syslog dispatcher

be shaped so it can be handled by NIRVANA-Kai. First, all messages sent by syslog are received by the receiver module. The log receiver module assigns log messages to each device based on the IP address of the sender of the syslog transmission and runs a log parser for each device in order to shape its log message. In the log parser, a regular expression is used and data is extract on sender/recipient IP address, port number, alert contents, severity, and occurrence time, etc. Then, the extracted data goes on to be stored in a database (alert database). The individual pieces of data accumulated in the alert database are displayed on the visualization user interface as individual petals of flower-shaped objects.

On the other hand, when considering organization-internal operations such as network management and security operations, syslog log messages need to also be sent to devices and servers other than NIRVANA-Kai. In this situation, a syslog transmission method suited to the purpose is implemented by the Syslog Dispatcher, as one of the functions with which NIRVANA-Kai is equipped. The Syslog Dispatcher is equipped with the following three types of transfer method: (1) duplication, (2) multicast, and

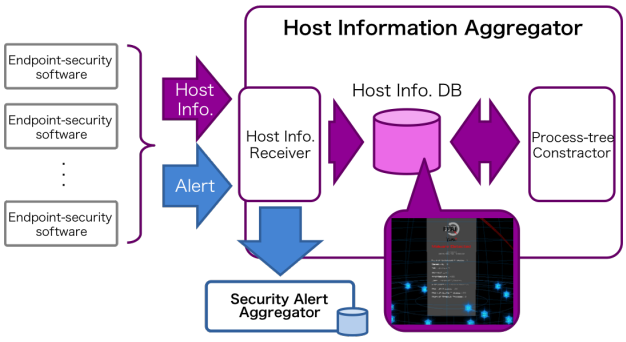

Fig. 6 Overview of the host information aggregator

(3) round-robin (Fig. 5).

In addition to these transmission methods, there is also the functionality for filtering log messages with the desired character strings. When using these functions it becomes possible to realize a system that can transfer all log messages to a syslog server and save them on a disk, filter out only the highly important log messages, and visualize them on NIRVANA-Kai.

3.3 Host information aggregator

In addition to alert information, NIRVANA-Kai also aggregates information from inside hosts. Figure 6 shows an overview of the Host Information Aggregator [4][5]. In order to aggregate the different types of information from the hosts, an agent tool that works in coordination with endpoint-security software needs to be introduced in advance. When malware is detected by the endpoint-security software, this agent tool notifies the Host Information Receiver of processes that have been judged as malware and detection reason. Additionally, at fixed intervals, the agent tool sends basic information (e.g. OS type/version, user information, MAC address, etc.), running processes information and transmission generated by processes to the host information receiver. Following this, malware detection information is sent by syslog to the Security Alert Aggregator, and other information is stored in a database (host information database).

The Process-tree Constructor constructs process trees based on process ID (PID) and parent process ID (PPID) from out of the process information accumulated in the host information database. The basic information and process trees of the host that are accumulated in the host information database are visualized in "Host View." Figure 7 shows this structure. The basic information of the host is listed on the surface of a lithograph-shaped object (monolith) located in the central portion of the host view. Objects that orbit the monolith on a satellite-like path indicate the

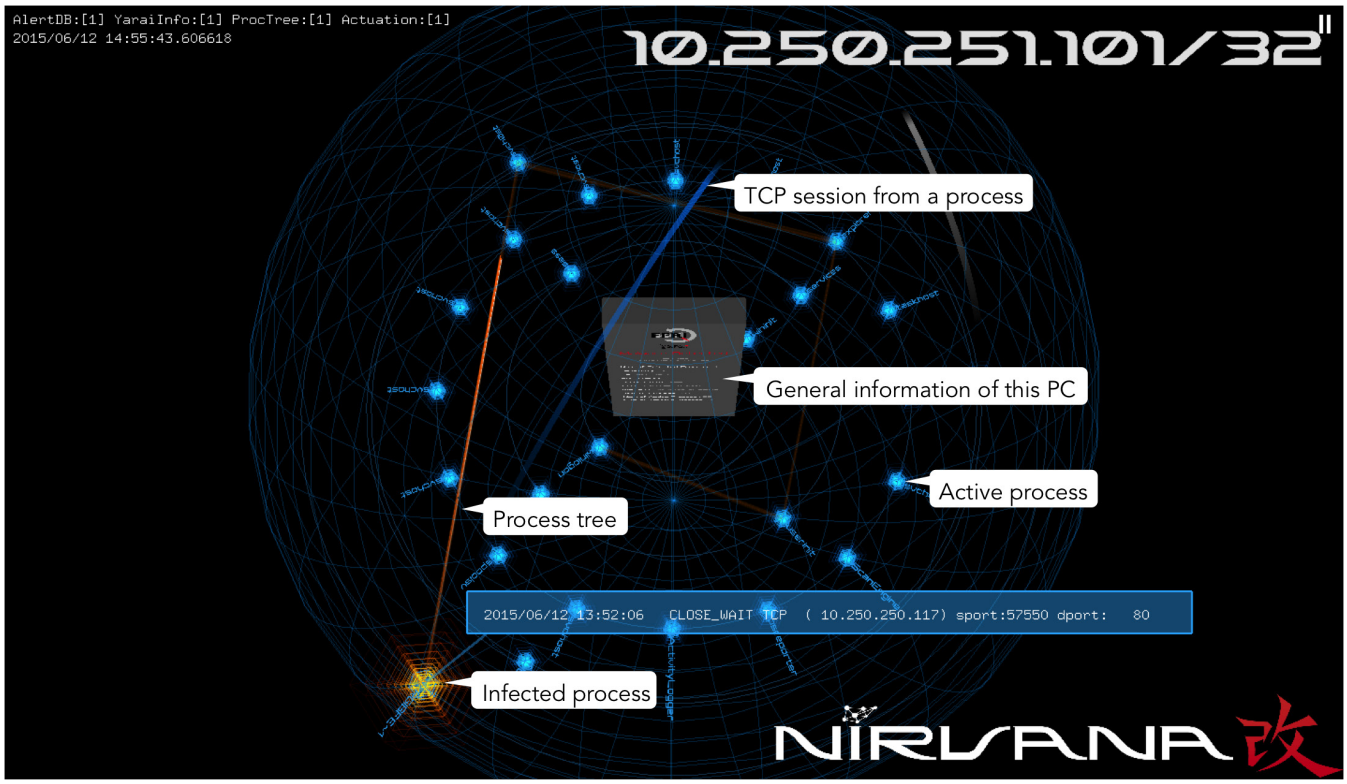

Fig. 7 Host view

currently running processes in the host, and the radius of that trajectory indicates the process's generation. Processes orbiting on the outermost trajectory highlight those that have been detected as malware.

If such process is clicked, the parent/child relationship of processes is indicated as in the figure, and the generation steps of that process can be ascertained. Further, the establishment of the TCP session against an external network from the malware can also be visualized. In this case, a host in the external network is called a C&C server, and the malware and the external host are exchanging attack commands via HTTP transmission.

3.4 Event analysis platform

Up to the previous section, there has been a discussion of the information that can be aggregated with NIRVANA-Kai. However, as the scope of network/security devices and executing PCs in organizations grows larger, it can be predicted that alert information will increase, and it is possible that important alert information will be overlooked. For this reason, NIRVANA-Kai analyzes the aggregated alert information and is equipped with an Event Analysis Platform[6] that works to extract important events that need to be dealt with by a security operator. Figure 8 shows an overview of the Event Analysis Platform. The Event Analysis Platform has a mechanism that allows for

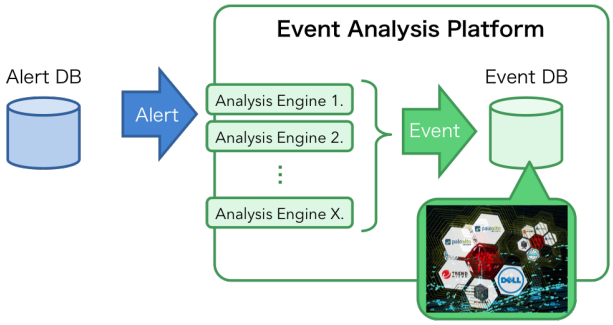

Fig. 8 Overview of the event analysis platform

easy implementation of an "analysis engine" that analyzes the information accumulated in the alert database, and the security operator is able to independently add in an analysis engine. The Event Analysis Platform has the mechanisms below that allows it to write to and operate the analysis engine.

- Plugin mechanism to allow implementation of an analysis engine (analysis plugin)
- Template to allow implementation of an analysis plugin
- Domain-Specific Language (DSL) to allow writing to an analysis engine

When a new analysis engine is written to, DSL is written and multiple analysis plugins are linked. The different types of analysis plugin include those that input data, those

that filter data, and those that output data. In the Event Analysis Platform there is an analysis plugin included, as in Table 1. The input/output of the analysis plugin is prescribed and when a new analysis plugin is needed it can be implemented in accordance with the prepared template.

The event information extracted from the analysis engine is stored in a database (event database). On the visualization user interface, event information is represented in the shape of an effect on the flower-shaped object that is the alert information. Figure 9 shows event information being highlighted. The soundwaves that are emitted from the perimeters of the flower-shaped objects show that an event is being generated from that address block.

Table 1 Analysis plugin included in event analysis platform

Type	Analysis Plugin	Overview
input	database	Input from MySQL, PostgreSQL, SQLite
	csv	Input from CSV file
	yaml	Input from YAML file
	json	Input from JSON file
filter	match	Conditions Met
	sort	Collation
	group_by	Grouping
	unique	Deduplication Counting
	truncate	Subthreshold Truncation
output	database	Input from MySQL, PostgreSQL, SQLite
	csv	Input from CSV file
	yaml	Input from YAML file
	json	Input from JSON file
	stdout	Standard Output

3.5 Multifaceted actuator

NIRVANA-Kai is able to take various actuations (i.e. sending defense commands into several devices) against events obtained from analysis results. Figure 10 shows an overview of the Multifaceted Actuator which fulfills the role of defense in NIRVANA-Kai. Taking the event information obtained from the analysis engine described above, it applies the preset "actuation policy" to extract candidates for actuations. The actuation policy can be written in the desired programming language, and the extracted candidates for measures and the corresponding defense commands are set and stored in a database (actuation database). Network administrators select items to which actuations should be applied via a defense determinant user interface (defense determinant UI). Having done so, an "actuation engine" notifies an "actuation module" of its preset defense measure, and a defense command is actually sent to the device subject to defense measures. An actuation module

Fig. 9 Event information highlighting

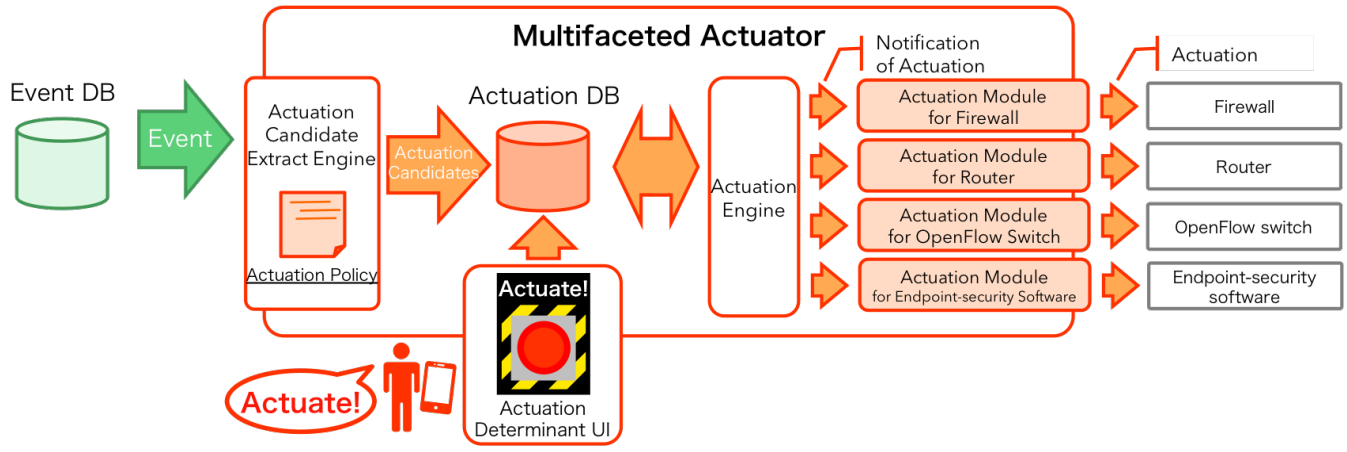

Fig. 10 Overview of the multifaceted actuator

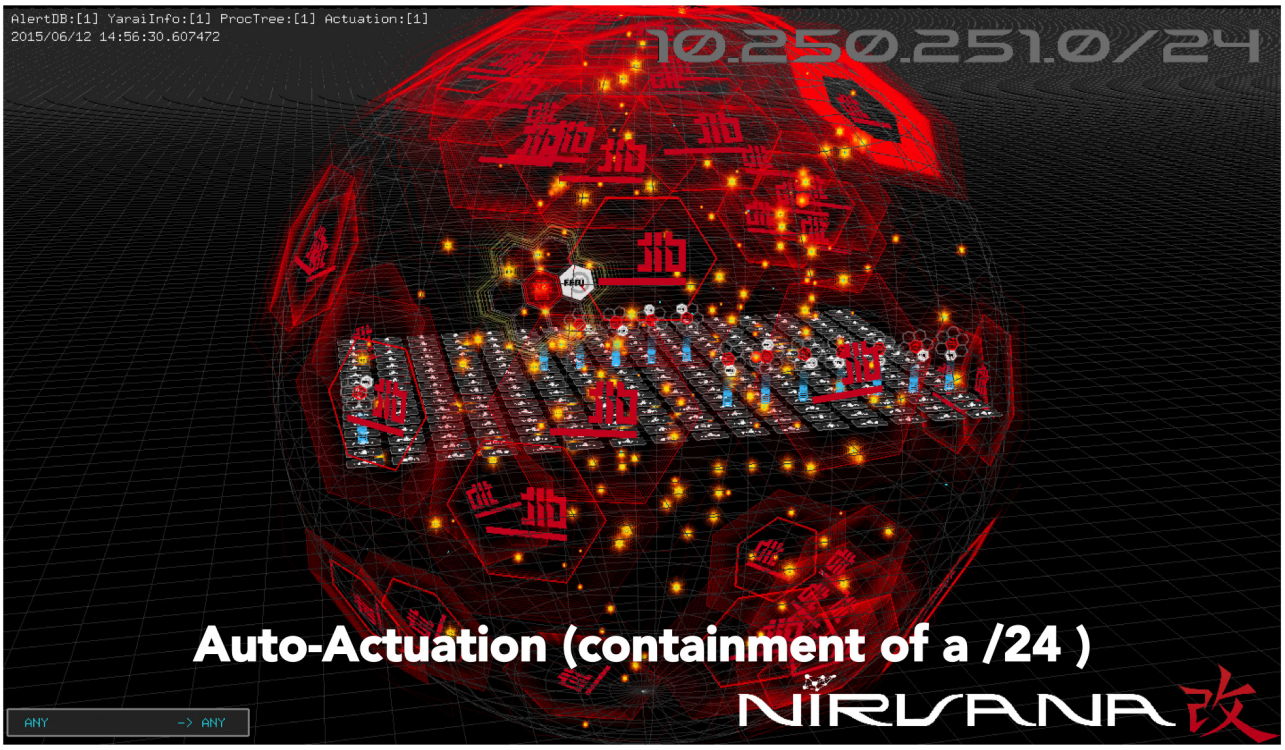

Fig. 11 Isolation of a class C address

can be added when appropriate and combined with protocols and interfaces of devices subject to defense measures, such as NETCONF and REST API, and defense commands can be written in the desired programming language.

Further, when defensive measures are being implemented, this is visualized on NIRVANA-Kai. As an example of actuation visualization, Fig. 11 shows the use of a firewall and the isolation of a Class C address. In this case, when a large-scale infection of malware has been recognized inside a given network, a defense policy is preset to send a defense command to the firewall, and the network administrator can make an actuation determinant judgement concerning the actuation candidate they are presented with. In addition to this example, NIRVANA-Kai can intercept transmissions sent toward a particular host and simultaneously execute malware scans of multiple hosts on a desired network using security software and visualize these actions.

4 Security operations using NIRVANA-Kai

In the previous chapter there was an overview of the use of NIRVANA-Kai in information aggregation and analysis and a discussion of execution sequence flow for defense measures and the different components that fulfil those roles. The different components of NIRVANA-Kai that have been introduced thus far are used in the daily security operations inside NICT, and feedback is obtained on points for improvement related to actual operation.

At NICT since around September, 2015, there has been a step-by-step increase in devices that aggregate alert information and as of today, March 31, 2106, monitoring is carried out on 14 devices/pieces of software. Figure 12 shows the number of alerts during that period by device type, and the number of alerts, which varies widely depending on the device/software, and swift identification much be made of the items from within those types that need to be dealt with. Figure 13 is an example of a visualization user interface that was developed based on feedback arising from actual operation. Set in the background is a picture of the different types of device and the geographical location, and—so the severity of alerts can be determined at a glance—they are visualized in numerical values and colors in the center of flower-shaped objects.

In order to realize even more practical security operations, it is absolutely necessary that they are included in actual operations. Moving forward, too, incremental improvements in actual operational functionality can be built up in NIRVANA-Kai through increasing operational experience with the system. Establishing practical security operations centered on visualization technologies will be an important issue going into the future.

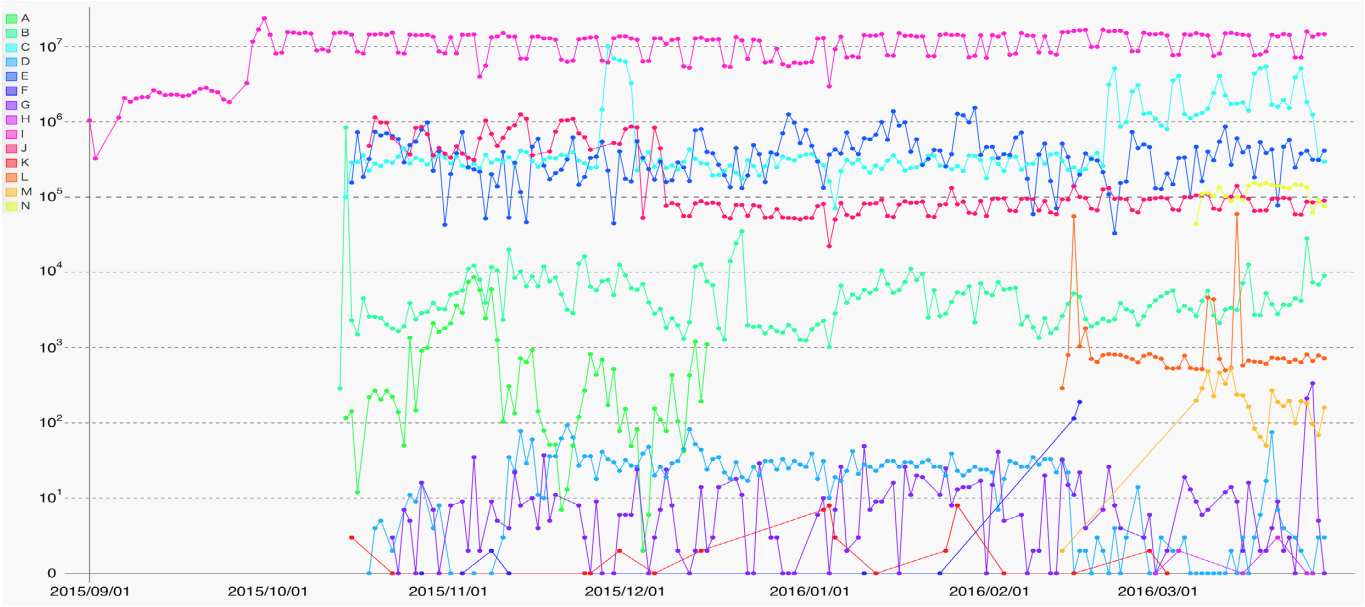

Fig. 12 The number of alerts by device type from September 01, 2015 to March 31, 2016

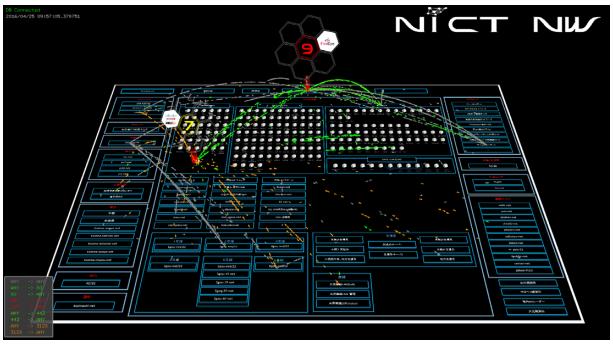

Fig. 13 NIRVANA-Kai being implemented inside the NICT network

5 Conclusion

Discussed in this report were [i] as technology for resisting targeted attacks, the system structure of the NIRVANA-Kai integrated analysis platform for defense measures against cyber-attacks and [ii] examples of its application at NICT. By using NIRVANA-Kai, transmissions and alerts that are generated within an organization can be aggregated and analyzed, and information needed in security operations can be visualized.

In efficient and effective security operations against targeted attacks, changes to the situation inside the organization need to be swiftly recognized. In the future, there are plans to build up functional improvements through security operations at NICT and to continue involvement in research and development that will establish even more practical security operations through the use of visualization technologies.

References

1 D. Inoue, M. Eto, K. Yoshioka, S. Baba, K. Suzuki, J. Nakazato, K. Ohtaka, and K. Nakao, "nicter: An Incident Analysis System Toward Binding Network Monitoring with Malware Analysis," WOMBAT Workshop on Information Security Threats Data Collection and Sharing (WISTDCS '08), pp.58–66, 2008.

2 M. Eto and Y. Takagi, "An Incident Analysis Center "nicter" and its Social Commitment," Journal of NICT, vol.57, no.3/4, pp.17–26, 2011 (in Japanese).

3 K. Suzuki, M. Eto and D. Inoue, "Development and Evaluation of NIRVANA: Real Network Traffic Visualization System," vol.57, no.3/4, pp.63–80, 2011 (in Japanese).

4 J. Nakazato, Y. Tsuda, M. Eto, D. Inoue and K. Nakao, "A Suspicious Processes Detection Scheme using Process Frequency," IEICE Technical Report, vol.115, no.334, pp.61–66, 2015 (in Japanese).

5 J. Nakazato, Y. Tsuda, M. Eto, D. Inoue, and K. Nakao, "A Suspicious Processes Detection Scheme using Process Frequency and Network State," IEICE Technical Report, vol.115, no.488, pp.77–82, 2016 (in Japanese).

6 Y. Tsuda, T. Tomine, M. Kamizono, M. Eto, and D. Inoue, "A Pluggable and Programmable Platform for Analyzing Security Logs," IEICE Technical Report, vol.114, no.489, pp.31–36, 2015 (in Japanese).

Yu TSUDA, Ph.D.

Researcher, Cybersecurity Laboratory,
Cybersecurity Research Institute
Cybersecurity, Countermeasure against APT

Nobuyuki KANAYA

Research Expert, Cybersecurity Laboratory,
Cybersecurity Research Institute
Cybersecurity. Web Security

Takashi TOMINE

Technical Researcher, Cybersecurity Laboratory, Cybersecurity Research Institute
Cybersecurity, Network Management

Masaki KAMIZONO

Guest Researcher, Cybersecurity Laboratory, Cybersecurity Research Institute
Cybersecurity

Masato JINGU

Research Expert, Cybersecurity Laboratory, Cybersecurity Research Institute
Cybersecurity, Incident Response

Yaichiro TAKAGI

Technical Researcher, Cybersecurity Laboratory, Cybersecurity Research Institute
Cybersecurity

Koei SUZUKI

Senior Technical Researcher, Cybersecurity Laboratory, Cybersecurity Research Institute
Cybersecurity