3-2 GHOST Sensor: Development of a Proactive Cyber-attack Observation Platform

Masashi ETO

There have been several network monitoring projects launched to address cyber threats on the Internet. Meanwhile there are some difficulties of operations of network monitoring system such as; difficulty of efficient utilization of wide IP address range for monitoring, burden of physical or logical maintenance of remote sensor systems, and obsolescence of the monitored IP addresses due to be recognized by attackers as “monitored”. In order to solve these problems, this paper proposes a multipurpose network monitoring platform that can manage any type of sensors with applying the virtual sensor mechanism and the dynamic address assigning technique. This article introduces the detail of the platform.

1 Introduction

Cyber-attacks are becoming more serious. In response, currently, research and development of various cyber security technologies is advancing all over the world. Among them, several research and development projects have been advancing research and development and operation of wide area network monitoring technologies, for the purpose of understanding the situation of cyber-attacks on the Internet [1]–[5]. As a countermeasure against cyber-attacks, what is common to these projects is that they are concentrating on gathering information on attacks occurring all over the world over a wide scope and at a deep level, and understanding the situation globally.

Meanwhile, methods of cyber-attacks are becoming more and more complicated. Remote exploit attacks on OS and server applications are still active. Recent years have also brought a rapid increase in malware infections via applications such as web sites and emails, for example in drive-by download attacks.

In this way, various attack information and collection methods have been proposed according to the type of threat, in order to respond to cyber-attacks which flexibly change their attack targets. For attack information gathering systems targeting remote exploit attacks, especially widely used systems are high-interaction and low-interaction honeypots [6][7] that observe attacks as a vulnerable real host, and honeypots specialized for HTTP, SSH and DNS services, etc. Also, black hole monitoring [8][9] which observes the state of attack without any response to communications from outside the network is easier to operate than a honeypot, so it is suitable for the network monitoring over a wider scope, and is operated in many research projects.

In addition, web crawlers periodically searching each web server are being researched and developed in various organizations, as one of the countermeasures against drive-by download attacks that attempt to intrude into clients by responses containing exploit code. It is possible to understand the situation of the cyber-attacks on the Internet for the first time by installing these sensors widely in Japan and also overseas (or by sharing information closely with overseas organizations).

However, when operating a cyber-attack observation system constructed using these technologies, there is the problem that it is difficult to “see the desired attack with the optimum honeypot”. For example, in an environment where the honeypot is operated under a fixed IP address even if it has a wide area observation network, it is difficult to observe the attack in detail unless an attack comes to the IP address of that honeypot. Deep information cannot be obtained, unless attacks are observed with an optimal sensor in the form of a web server type honeypot if it is an attack in web communication, or an SSH server type honeypot if it is an attack in SSH communication. However, such flexible monitoring is difficult when operating by preexisting fixed IP addresses.

In addition, there are various issues in operation of sensors with logically and physically wide scope in this way. In particular, from the viewpoint of projects that construct sensor networks in international wide area networks as in the above [1]–[5], the operational issues of a wide area network monitoring system are listed below.

Difficulty in obtaining a wide area darknet As mentioned earlier, black hole observation is suitable for wide area network monitoring due to its ease of operation, but it is a passive attack observation method, so in order to effectively collect attack information, it should be applied to darknets (unused IP address groups) with wider ranges (e.g. /16 subnets, etc.). However, international IPv4 address resources are not abundant in many countries, so it is difficult to observe with a vast darknet like in previous research. Therefore, there is a need for technology used for observation that is effective even if there are few IP addresses (approximately 2 or 3).

Maintenance Cost If one cannot obtain a vast darknet suitable for black hole observation, and only few IP addresses can be used, one could set up high-interaction or low-interaction honeypots to collect deeper attack information. However, these honeypot systems that perform detailed observation by actually receiving attacks by decoys or emulators, on the other hand, require system configuration and machine resources that are more complex than in black hole observation, so their maintenance costs for prevention of secondary infection and system troubleshooting tend to be higher.

IP address blacklisting problem When using the same IP address and operating the observation sensor for a long period of time, an attacker can detect that it is an observation network, that IP address is put on the attacker's blacklist, and attackers avoid it. As a result, with honeypots, it can become more difficult to collect malware specimens, and especially with active sensors such as web crawlers, the access source addresses can be registered in attacker blacklists, and access to attackers' websites can be refused.

In response to these issues, in this research, we developed a GHOST sensor proactive cyber-attack observation platform. This uses physical machines and IP address resources effectively, while having a sensor mechanism that allocates flexibly according to the attacker, with the aim of operating a sensor network stably and continuously. This method has virtual sensor technology and a mechanism for dynamic address allocation to honeypots as its main functions, which makes it possible to integrally operate the various types of sensors described above.

In this paper, first, Section **2** describes previous research on operational technologies of attack observation networks. Next, Section **3** introduces the configuration and functions of the GHOST sensor which is the proposed system of this research. Section **4** investigates effects on collection of attack information when the proposed method is actually used, examines feasibility of the proposed method, and evaluates the proposed method based on the implementation in Section **5**. Finally, Section **6** provides a summary and describes future issues.

2 Related research

As cyber-attack observation techniques, in addition to [3][4][9] which mainly deal with black hole observations, many proposals are being made for the effective use of resources while operating honeypots [10]–[14]. Among them, Collapsar [11] proposed a honeypot operation technique which places so-called virtual sensors at observation sites in remote locations, only forwards packets to the analysis center (or the Internet), and operates high-interaction honeypots comprised of virtual machines in the analysis center.

In addition to the functions of Collapsar, Potemkin [12] is configured so when an attack comes to a specific IP address, it dynamically launches a virtual machine having that IP address and responds. In addition to reducing consumption of machine resources by launching virtual machines only when necessary, it has virtual machines that correspond to all its IP addresses, so one can say that this effectively utilizes the IP addresses of the observation target.

On the other hand, SGNET [13] has a configuration similar to Collapsar and Potemkin. General server responses to queries from attackers are recorded in remote sensors, and they are responded to as much as possible, to reduce the amount of communications between observation sites and the analysis center. It is noteworthy that, in the case of an unknown query, it is possible to respond in real time by transferring the query to a real server on the center side.

All of these honeypot operation technologies are specialized for the purpose of effective use of resources in the operation of high-interaction honeypots and efficient sample acquisition, and one can say they are effective methods in those respects. However, at most approximately several thousand honeypot instances can be launched concurrently using advanced virtualization technology. On the other hand, there are projects that observe tens of thousands of darknet addresses. However, even with advanced operation technology, it is expected that machine resources will be depleted when observing many attacks. Therefore, it is necessary to consider a method to use IP addresses as efficiently as possible without waste. In addition, these operation technologies are mainly technologies that assume operation of passive type honeypots, so it is difficult to apply them as is to other observation methods. It is necessary to consider an operation method while also considering issues in observation by web crawlers as mentioned in the previous chapter.

3 GHOST sensor: A proactive cyber-attack observation platform

In response to the issues in related research listed in the previous chapter, our research dynamically assigns IP addresses to various network monitoring systems such as black hole sensors and web crawlers as well as high-interaction honeypots. We thus developed a GHOST (Global, Heterogeneous, and Optimized Sensing Technology) sensor, a proactive cyber-attack observation platform that effectively utilizes physical and logical resources.

3.1 Outline

Figure 1 shows a diagram of the proposed system.

In the proposed system, as in the previous research by Collapsar etc., the actual sensors that were previously operated at remote observation sites are placed on the analysis center side (Fig. 1: Low-interaction Honeypot, High-interaction Honeypot and Blackhole sensor). On the other hand, only the virtual sensors having the L3 proxy function (Fig. 1: Virtual sensor) are placed on the observation site side. The virtual sensors are focused only on forwarding received attack packets to the analysis center, and the actual sensors in the analysis center respond to specific attacks.

The virtual sensor extends the L3 network (and IP address) provided at the observation site, through the VPN line until the analysis center. Therefore, it seems for the attacker that it is making an attack against the cooperating organization itself, but in reality all attacks are handled at the analysis center.

A major feature of the proposed system is that the connection manager (Fig. 1: Connection manager) dynamically allocates IP addresses according to various operational policies, and enables dynamic attack observation, not only for honeypots, but also web crawlers, black hole

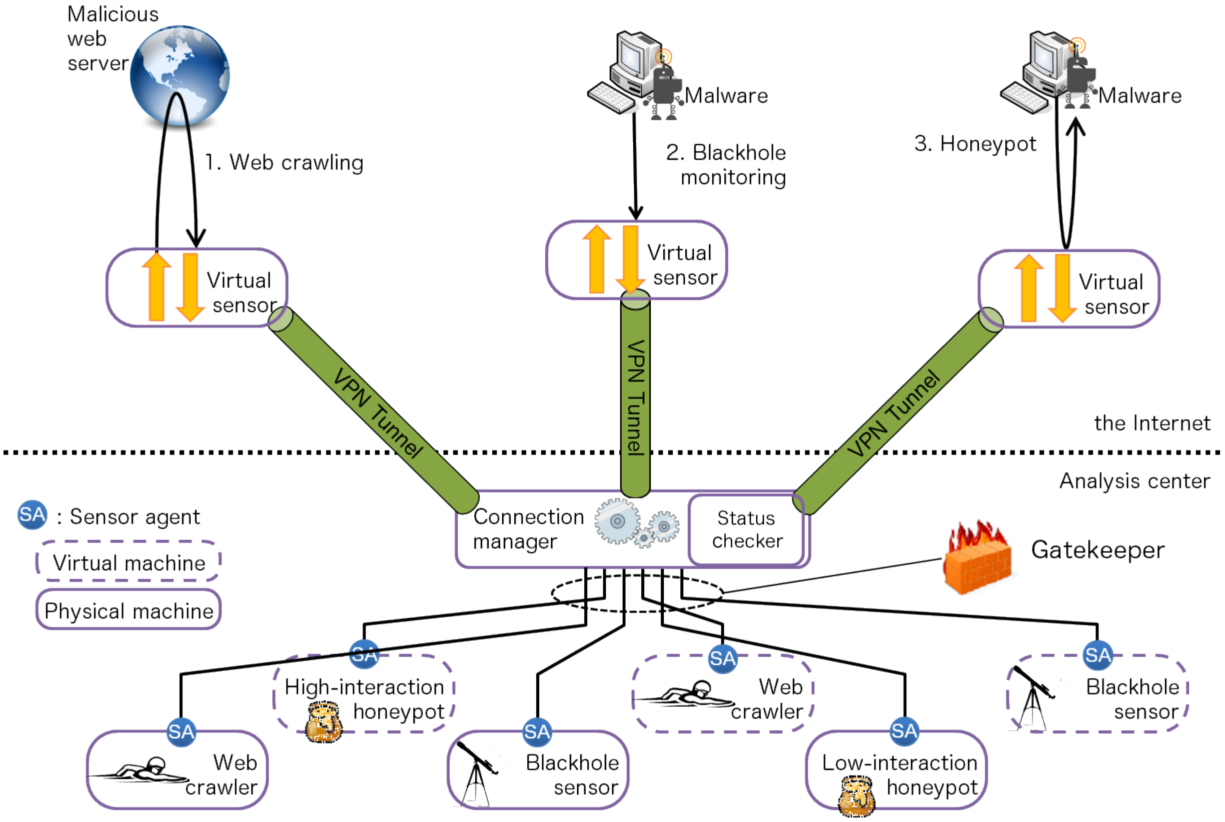

Fig. 1 GHOST sensors diagram

sensors, etc. Furthermore, in order to prevent secondary infection to the outside via a high-interaction honeypot, it is necessary to set up a monitoring system with an IPS and firewall, etc.; until now, it was necessary to install these systems at each analysis site. In contrast, the proposed system can centrally manage this inside the analysis center.

3.2 Main components

The proposed system's main components and their functions are described below.

Virtual Sensor The virtual sensor is an L3 proxy type sensor program installed at an observation site that is geographically separate and is assumed to run on a virtual machine built on the physical machine of the observation site. The virtual sensor connects the VPN line by IPsec with the analysis center, encapsulates all packets destined for itself and transfers them to the analysis center. On the other hand, the response packets from the actual sensors in the analysis center are transmitted to the appropriate destinations.

Actual Sensors / Sensor Agents The actual sensors are comprised of black hole sensors, high-interaction / low-interaction honeypots and various web crawlers, which were mentioned previously. It can use either a physical machine or a virtual machine as the actual sensor, which differs from existing methods [11][12] which assume a virtual machine. This is because the sensor agents installed on all the actual sensors manage IP addresses on the actual sensors, whereas in previous research, this was managed using a hypervisor of the virtual machine (when the "Agent Method" described in Subsection **3.3** is used). In accordance with messages of the connection manager, the sensor agent changes the IP address of the actual sensor in real time, and also has a function to periodically send statistical data such as the number of acquired specimens and number of packets to the connection manager.

Connection Manager The connection manager is installed at the boundary of the analysis center, and has a function of transferring packets from the virtual sensors to appropriate actual sensors, and returning the response packets to the virtual sensor side. Also, as the most important function, the connection manager transmits address change commands to sensor agents according to the profile of the attacker and the operating status of the actual sensor, and has the role of maintaining the optimum actual sensor configuration at all times. It is implemented according to allocation rules pre-described in the Lua language. A major feature of the proposed system is that the assignment rules can be defined by software in this way.

Gatekeeper A gatekeeper is installed between the actual sensor and the analysis center, and in particular, it monitors and controls traffic to the outside. By consolidating monitoring points in one place this way, it enables reduced burden of operations for preventing secondary infections. Here, for example, only control such as C&C communications by bots infected in actual sensors and connection confirmation communications to a well-known website are permitted, and other communications are cut off.

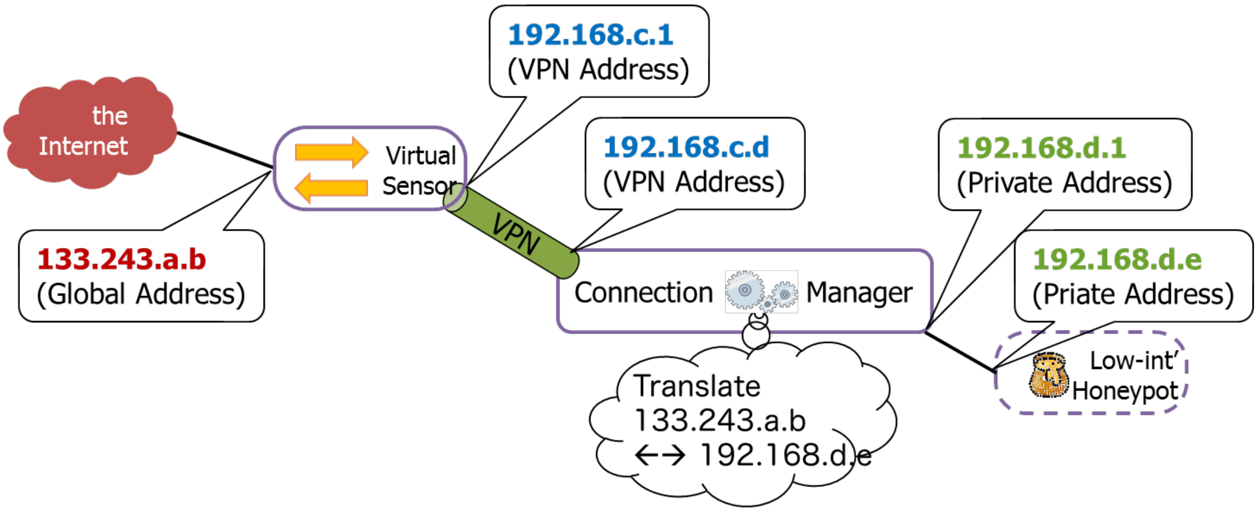

Fig. 2 Dynamic IP address allocation by NAT method

3.3 Investigation of address dynamic allocation function

As mentioned in the previous section, the connection manager has the function of forwarding the packets from virtual sensors to actual sensors, and changing the allocation of actual sensors' IP addresses as necessary. For this case, the following investigation was done as a method of dynamically changing the address of the actual sensor.

Virtual Machine Method Potemkin [12], Dense Ship [14], etc. dynamically allocate IP addresses, utilizing features of virtual machines. In this case, dynamic IP address allocation is achieved in the form of launching a virtual machine corresponding to an arbitrary IP address in real time. On the other hand, in this research, it is necessary to use not only virtual machines but also physical machines as actual sensor platforms, so this method requiring virtual machines is not suitable.

NAT Method A fixed address is permanently allocated to the actual sensor, and the connection manager switches the correspondence between the virtual sensor and the actual sensor by NAT conversion (Fig. 2). It can be achieved by switching the NAT table without dynamically changing settings on the actual sensor side, so there is an advantage that it is possible to allocate faster than in other methods. However, if the actual sensor has an advanced environment such as a high-interaction honeypot, the attacker may notice a difference between the attack target IP address and the actual sensor IP address.

DHCP Method The connection manager becomes a DHCP server, distributing an arbitrary IP address to the actual sensor in an extremely short time, and dynamically changing the allocation as necessary. This is effective in that there is no need to change the actual sensor side. However, it is inappropriate in this research, because it takes at least time in the order of seconds to change the IP address.

Agent Method A sensor agent resident on the OS of the agent type actual sensor dynamically allocates IP addresses, by receiving messages from the connection manager (Fig. 3).

The sensor agent runs on the OS, so high-speed address switching becomes possible. In addition to switching addresses, it is also possible to more flexibly control the actual sensor, for example, by an agent imitating actions of a real user. The sensor agent constantly checks the status of the target actual sensor, and is controlled to not change the address if, for example, an arbitrary TCP session is being established. However, in the event of an attack or the like that hinders communication of all other processes, any control cannot be performed, so it is necessary to consider a backup system such as combining with other methods.

Considering the above points comprehensively, this research adopts a dynamic IP address switching method in both NAT method and agent method, and implements both these forms so switching can be achieved by settings of the GHOST sensor.

4 Advance survey

In this research, before implementation of the proposed

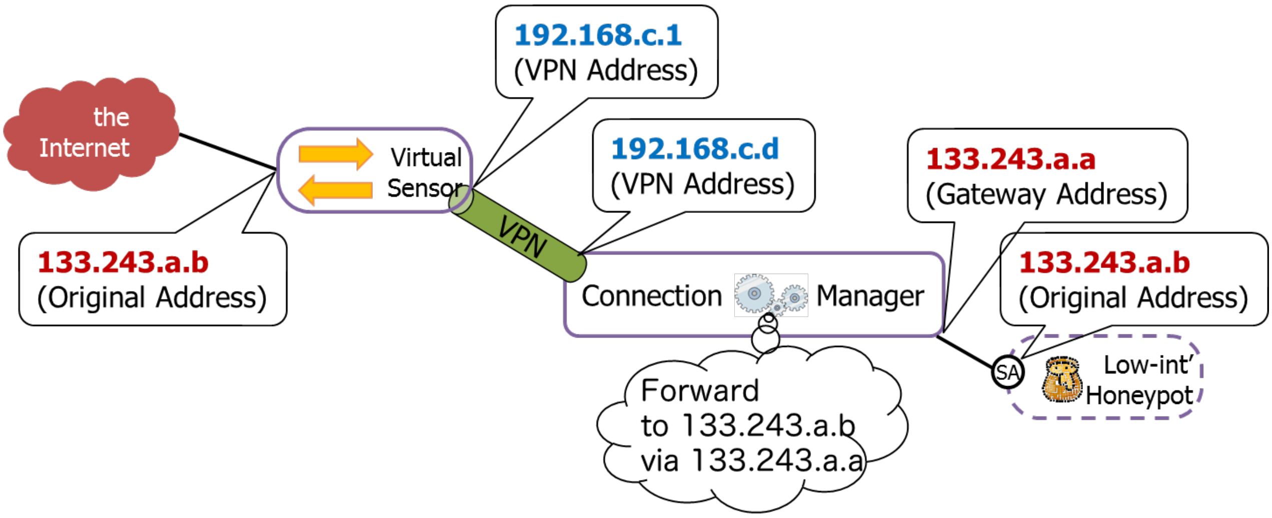

Fig. 3 Dynamic IP address allocation by agent method

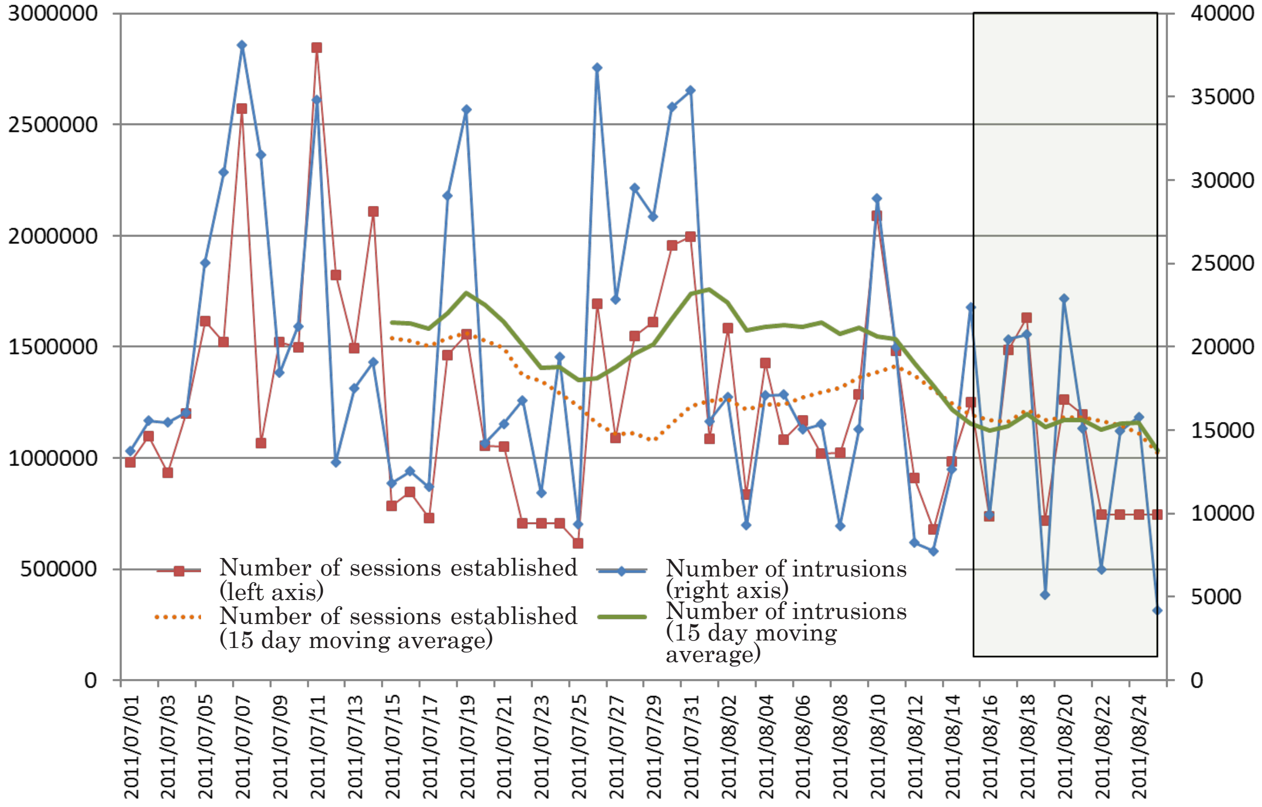

Fig. 4 Status of attack observation in low-interaction honeypot

system, we investigated the effects of the proposed system on attack observation. In the proposed system, it is the actual sensor in the analysis center that makes the actual responses to all attacks. Therefore, all the packets pass between the virtual sensor and the analysis center, and a corresponding round trip delay will occur. Therefore, in this research, we examined how round-trip delay during communication affects collection of attack information.

The authors assume that a virtual sensor is installed overseas using the proposed system, and the observation range is expanded. There may be a large delay of several hundred milliseconds to overseas, so this evaluation investigated effects of delays on specimen collection in an environment introducing a one-way 500 millisecond delay.

4.1 Effect on attack caused by round-trip delay between sensor and gate

In this evaluation, we installed a delay generator on the upstream interface of the low-interaction honeypot which consists of high-interaction and low-interaction honeypots (Nepenthes) configured as a real machine in Windows XP. Then, by intentionally generating delays, we constructed an environment similar to the proposed system. In this state, observation was carried out for 10 days from August 16 to 25, 2011. Compared with the case of no delays, we verified changes in the number of specimens acquired, the number of attacks (exploit establishment), and the number of TCP sessions.

Figure 4 is a graph showing the number of attacks and number of TCP establishment sessions in the low-interaction honeypots from July 1 to August 25, 2011. Note that the low-interaction honeypots observe 245 IP addresses. Basically, regardless of whether they are high-interaction / low-interaction honeypots or black hole sensors, the honeypots passively wait for attacks, so the information that can be observed per day varies widely from day to day. Therefore, we decided to check by obtaining the moving average. It can be seen that the average per day is approximately 1.2 million TCP sessions, and approximately 15,000 attacks during the period when the delay generator was installed (shaded part starting August 16). This is lower than before August 16 when the delay generator was not yet installed, but it is also possible to see that the number of attacks that were on a downward trend, that downward trend continues to decrease.

Figure 5 shows the number of specimens obtained and the number of TCP sessions established, when observations were made under the same conditions as described above, for a high-interaction honeypots observing with three IP addresses. In the high-interaction honeypots, after the

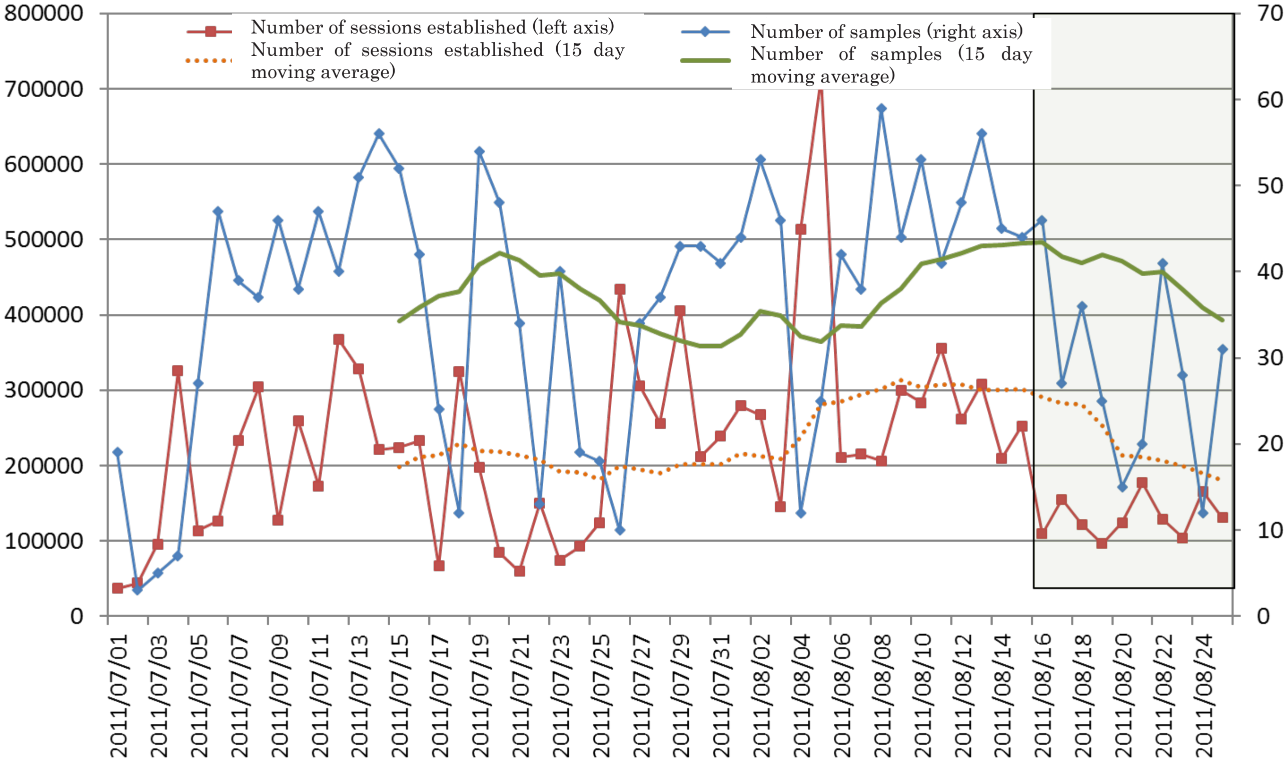

Fig. 5 Status of attack observation in low-interaction honeypot

delay generator was installed, the moving average is 35 to 40 samples obtained per day, and approximately 400,000 to 500,000 sessions established per day; it can be confirmed that these are approximately the same amounts as in the period before the delay generator was installed.

4.2 Discussion on round trip delay

When we introduced a one-way delay of 500 milliseconds on two types of honeypot sensors and looked at the 15 day moving average, we confirmed that the detected attacks were roughly comparable to the period before the delay was introduced. If the delay affects the number of attacks etc., then it should show a sharp drop starting the day in which the delay generator was installed. However, even with actual measurements each day, there were many days on which values recorded were about the same as they were before the delay generator was installed. From this, it was found that the round-trip delay between the virtual sensor and the analysis center does not significantly affect the attack observation rate.

5 Evaluation

Based on the investigation until now, we implemented prototype GHOST sensor, constructed a small scale observation environment, and actually connected it to the Internet to perform a demonstration experiment. In this environment, a black hole sensor and a low-interaction honeypots are installed as a sensor, and an allocation rule was adopted that uses the low-interaction honeypots (more sophisticated than a black hole sensor) for newer attackers (source IP address). New source IP addresses are handled by the low-interaction honeypots, and other (known) attackers are sent to the black hole sensor, which reduces the opportunities in the low-interaction honeypots to collect duplicate malware samples from the same IP address.

Therefore, in this evaluation work, we evaluated how few of the same malware, that is, uniqueness of the observed samples, in the low-interaction honeypot of the experimental environment. In terms of effective use of resources, one of the important objectives of the GHOST sensor, we evaluated the utilization ratios of sensors and IP addresses.

5.1 Experimental environment

Figure 6 shows a diagram of the experiment environment.

In this experiment, in order to confirm the effectiveness of the GHOST sensor, we prepared two environments: one using the GHOST sensor (GS environment) and another

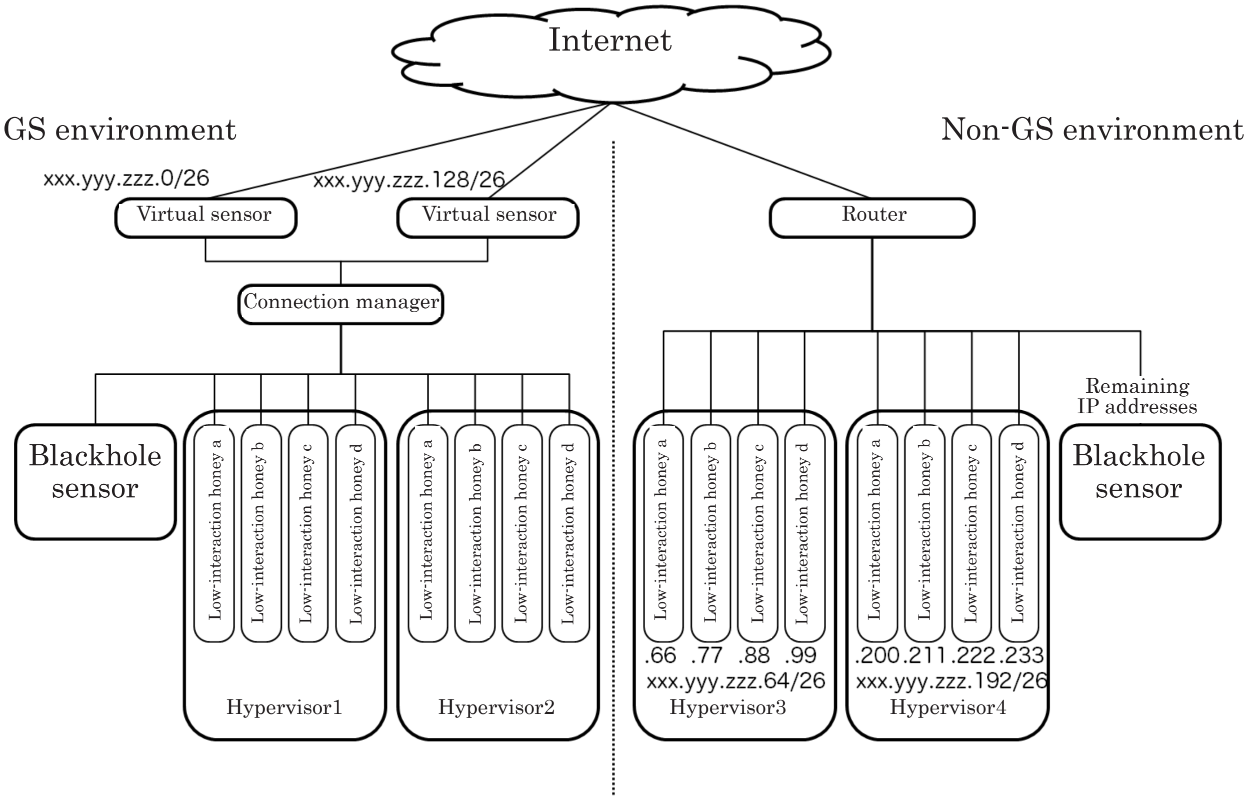

Fig. 6 Experiment environment

not using GS (non-GS environment). In addition, we used a Dionaea low-interaction honeypot (high priority) and darknet sensor (low priority) as the sensor group. Each of the GS and non-GS environments has eight interactive honeypots and one darknet sensor installed. For IP addresses used for observation, in order to compare in more similar environments, we divided single global class C (/24 subnet) addresses into four (/26 subnets of A, B, C, D), then allocated A and C to the GS environment, and B and D to the non-GS environment. In the GS environment, these addresses are dynamically allocated to the sensor by a function of the GHOST sensor. In the non-GS environment, one static IP address is allocated to each of eight low-interaction honeypots, and the others are allocated to the black hole sensor.

In this environment, attack observations were made in 24 hours from 0:00:00 November 19, 2013 to 23:59:59 November 20, 2013. As a parameter of IP address allocation in the GS environment, each sensor is set to automatically release the IP address 300 seconds after the IP address is allocated. In addition, IP addresses (of the known host) registered in the processed database are set to be deleted 5 hours after registration.

5.2 Evaluation: Uniqueness of samples obtained

In this section, we focus on the hash values of malware specimens obtained by the Dionaea low-interaction honeypots, and verify the uniqueness of the specimens obtained. Each of the eight interactive honeypots installed in the GS / non-GS environments independently collects specimens, and a specimen with hash value that was not obtained by any of the other seven units is defined here as a unique specimen. For reference, Tables 1 and 2 show the malware specimens actually obtained in each of the low-interaction honeypots in the GS environment (d1 a - d2 d), and each of the low-interaction honeypots in the non-GS environment (d3 a - d4 d), and each of their hash values.

In each table, the specimens indicated by underlined bold text are unique specimens obtained only by that honeypot. Table 3 shows the number of specimens obtained in both GS and non-GS environments and the number of unique specimens among in them and their proportions.

The number of specimens obtained in the GS and non-GS environments were relatively close, at 33 and 37, whereas the number of unique specimens among them were 17 (51.5%) in GS, nearly double the 9 (24.3%) in non-GS. On the contrary, looking at non-unique specimens, we obtained 76.7% of the specimens were non-unique in the non-GS environment, so we see that more duplicate

Table 1 Samples obtained in GS environment

Honeypot	Hash
d1a	3c3011089708c7a49346f648f1e79384
	9b175f5f727bcf1153e1aaf99798556a
	4f37e1e3ab27feba48038ea03dc55901
	65de48b370a61412435074479c6219fc
d1b	3c3011089708c7a49346f648f1e79384
	9b175f5f727bcf1153e1aaf99798556a
	9521d5fe45b1211e886da8b7ba813ac3
	cc32d0ee45e3f69e4e9b689c8c01c01c
	4d56562a6019c05c592b9681e9ca2737
	ffb4628a96fa19abab9bbded0324fecd
	64b4345a946bc9388412fedd53fb21cf
	7867de13bf22a7f3e3559044053e33e7
d1c	3c3011089708c7a49346f648f1e79384
	8535926634662a4e332121a6d2b01032
d1d	3c3011089708c7a49346f648f1e79384
	eb073edcb3340705a0a45f1d14231d47
	a4619b7dc17f18ef00b714db37a0ef19
	cb4c05cae975d30d7cac15df3cdbfe3e
	64b4345a946bc9388412fedd53fb21cf

Honeypot	Hash
d2a	3c3011089708c7a49346f648f1e79384
	ebfaf4383932b3ef39f1b29e1e574459
	9a1f8268805f01a7c3e0bfce07111cf4
d2b	**92675d3f5d76e4170230d1c0294f7be9**
	4d56562a6019c05c592b9681e9ca2737
	e5db14583694d3ff53d3b0b9c95d82b0
	3c3011089708c7a49346f648f1e79384
d2c	3c3011089708c7a49346f648f1e79384
	b202f4b1bdbb2615bb579d64fecd76a6
	7a676b8a1ad9d1efdde6ad9b0a663960
	7867de13bf22a7f3e3559044053e33e7
d2d	3c3011089708c7a49346f648f1e79384
	76e669836f48491f118c8e41c678e230
	b7d4ed11a02cd3f4867299640e1e52a8

Table 2 Samples obtained in Non-GS environment

Honeypot	Hash
d3a	3c3011089708c7a49346f648f1e79384
	e616b165d15a59d672918bf920d4faab
	c0fa3206395854b1eb55c47edd7011b5
	c443480243fbbd8cb11ade4ecdff1d45
	ffc8c1873be79006b4b221fe27e655e9
	7867de13bf22a7f3e3559044053e33e7
d3b	3c3011089708c7a49346f648f1e79384
	e616b165d15a59d672918bf920d4faab
	3c4351bc00f07b94d0fd189d2419d742
	c0fa3206395854b1eb55c47edd7011b5
	7867de13bf22a7f3e3559044053e33e7
d3c	3c3011089708c7a49346f648f1e79384
	e616b165d15a59d672918bf920d4faab
	c5862fe0aeb55594e1f74aa9cfbaa2a8
	c0fa3206395854b1eb55c47edd7011b5
	7867de13bf22a7f3e3559044053e33e7
d3d	3c3011089708c7a49346f648f1e79384
	e616b165d15a59d672918bf920d4faab
	a0194a481b12c590acd6bd8228b4c6d3
	c0fa3206395854b1eb55c47edd7011b5
	7867de13bf22a7f3e3559044053e33e7

Honeypot	Hash
d4a	3c3011089708c7a49346f648f1e79384
	e616b165d15a59d672918bf920d4faab
	207704c559f7b91f24b1b77f0f702da1
	7867de13bf22a7f3e3559044053e33e7
d4b	3c3011089708c7a49346f648f1e79384
	e616b165d15a59d672918bf920d4faab
	42801cfe875896daa5a6990b57567bad
	7867de13bf22a7f3e3559044053e33e7
d4c	3c3011089708c7a49346f648f1e79384
	e616b165d15a59d672918bf920d4faab
	114567ed87eb9723d7be3e9a66fd70d9
	7867de13bf22a7f3e3559044053e33e7
d4d	3c3011089708c7a49346f648f1e79384
	e616b165d15a59d672918bf920d4faab
	41c64356a9618a31785e505e5048047c
	7867de13bf22a7f3e3559044053e33e7

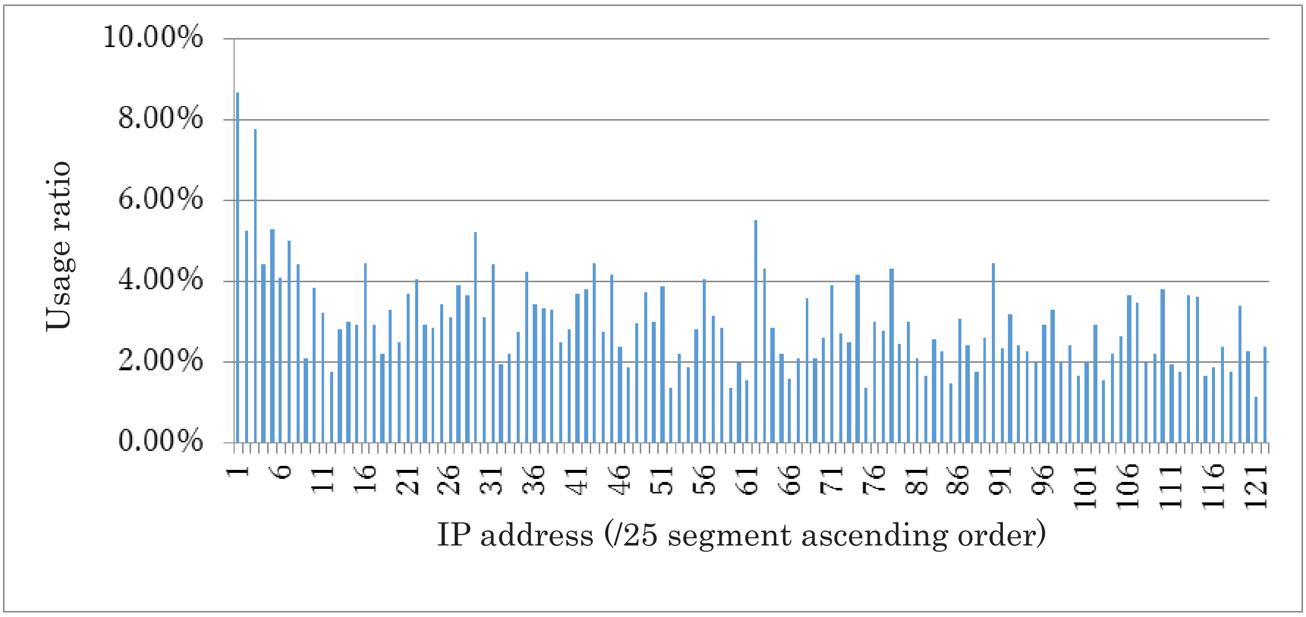

Fig. 7 IP address usage ratios

Table 3 Ratios of unique samples obtained in both environments

	GS Environment	Non-GS Environment
Total no. of samples	33	37
No. of unique samples	22	13
Unique sample ratio	66.7%	35.1%
Non-unique sample ratio	33.3%	64.9%

Table 4 Honeypot machine usage efficiency

Honeypot	Time (second)	Utilization Ratio (/ 172800 second)
d1a	64062.673	37%
d1b	64062.200	37%
d1c	63882.519	37%
d1d	63882.670	37%
d2a	63882.079	37%
d2b	63882.976	37%
d2c	64062.588	37%
d2d	63882.710	37%

specimens are obtained from the same attack source (compared to in the GS environment).

5.3 Evaluation of IP address usage efficiency

In this experiment, one class C (/24 subnet) of IP addresses (256 IP addresses) was divided into four, and two blocks (128 IP addresses) were each allocated to the GS and non-GS environments, then observed. Figure 7 shows the result of investigation from the connection manager log on how long the 128 IP addresses of the GS environment were allocated to the low-interaction honeypots.

In Figure 7, the X axis shows the ratios of the number of seconds each IP address was allocated to the low-interaction honeypot, divided by the total observation period (24 hours: 172,800 seconds). There was some variation, but we found that IP addresses were generally used between 3% and 5% of the time. This indicates that, as a result of the attacker scanning across the entire target network, the attack destination IP addresses were allocated evenly to the honeypots. The usage ratios of IP addresses 1 to 5 are higher than others because the opportunities for these IP addresses to be attacked are probabilistically higher.

From the above, we confirmed that this method can operate any honeypot with any IP address, in contrast to existing methods which can only use IP addresses statically allocated to honeypots.

5.4 Evaluation of machine operating efficiency

Similar to the evaluation of the IP address usage rate in Subsection **5.3,** this section evaluates the operation ratio of the honeypot machines. The operation ratio of the honeypot machines in the GHOST sensor can be calculated by measuring the time when the IP addresses were allocated to the honeypots. Table 4 shows the ratio of the time the IP address was assigned to each honeypot (d1 a to d2 d), divided by the total observation period (24 hours: 172,800 seconds) in the GS environment.

As a result of the verification, we found that the utilization ratio of each honeypot machine is constant at 37%. In a more detailed investigation, we see that one honeypot has an IP address allocated in a frequency of once (responds to attack) approximately each 5 to 10 minutes (average 7 minutes and 45 seconds). This confirmed that if this method is used, it can operate at a constant rate, in contrast to previous methods where the honeypot machine does not operate unless an attack arrives at the corresponding IP address, wasting computation resources.

However, from the viewpoint of efficient use of machines, it is desirable that these utilization ratios are close to 100%. Therefore, we found it necessary to adjust the number of IP addresses and honeypot machines to appropriate numbers.

6 Conclusion

Several network monitoring projects are being implemented internationally in order to follow complex network systems and their threats, but there are various problems in their operation. In order to solve problems in the operation of network monitoring systems, this research proposed a proactive cyber-attack observation platform GHOST sensor. In the proposed system, we designed virtual sensor technology, and designed it to enable flexible attack observation by dynamically allocating addresses to various actual sensors. As an evaluation before implementation, we investigated the effects of delay on specimen collection, and confirmed that the round trip delay between sensors and gates does not significantly affect observations of attacks. Furthermore, we constructed an evaluation environment consisting of Class C IP addresses and eight low-interaction honeypots etc., and confirmed that new specimens are gathered according to allocation rules, and showed the effectiveness of this method.

Currently, the GHOST sensor is incorporated in NICTER's observation network and full-scale operation is being carried out. From now on, we plan to apply effective sensor allocation rules more flexibly so we can "see the attacks you want to see with the optimal honeypots".

References

1 WOMBAT : Worldwide Observatory of Malicious Behaviors and Attack Threats. http://www.wombat-project.eu/.

2 PREDICT : the Protected Repository for the Defense of Infrastructure Against Cyber Threats. http://www.predict.org/.

3 SANS Internet Storm Center. http://isc.sans.org/.

4 F. Pouget, M. Dacier, and V.H. Pham. Leurre.com: On the Advantages of Deploying a Large Scale Distributed Honeypot Platform. E-Crime and Computer Conference (ECCE'05), 2005.

5 K. Nakao, D. Inoue, M. Eto, and K. Yoshioka, "Practical Correlation Analysis between Scan and Malware Profiles against Zero-Day Attacks Based on Darknet Monitoring," IEICE TRANSACTIONS on Information and Systems, vol.92, no.5, pp.787–798, 2009.

6 Nepenthes Development Team. http://nepenthes.carnivore.it/contact.

7 H. Project, "Dionaea honeypot." http://dionaea.carnivore.it/.

8 D. Moore. Network Telescopes: Tracking Denial-of-Service Attacks and Internet Worms around the Globe. In 17th Large Installation Systems Administration Conference (LISA'03), 2003.

9 M. Bailey, E. Cooke, F. Jahanian, J. Nazario, and D. Watson, "The Internet Motion Sensor: A distributed blackhole monitoring system," Proceedings of the 12th ISOC Symposium on Network and Distributed Systems Security (NDSS), pp.167–179, Citeseer, 2005.

10 L. Spitzner. Know your enemy: Genii honeynets, 2003.

11 X. Jiang and D. Xu, "Collapsar: A vm-based architecture for network attack detention center," Proceedings of the 13th conference on USENIX Security Symposium-vol.13, pp.22, USENIX Association, 2004.

12 M. Vrable, J. Ma, J. Chen, D. Moore, E. Vandekieft, A.C. Snoeren, G.M. Voelker, and S. Savage. Scalability, fidelity, and containment in the potemkin virtual honeyfarm. In ACM SIGOPS Operating Systems Review, vol.39(5), pp.148–162. ACM, 2005.

13 C. Leita and M. Dacier, "Sgnet: a worldwide deployable framework to support the analysis of malware threat models," Seventh European Dependable Computing Conference, pp.99–109, IEEE, 2008.

14 Y. Kawakoya M. Iwamura and M. Itoh, Dense Ship: Virtual Machine Monitor Specialized for Server-Type Honeypot, IEICE Technical Report, vol.111, no.82, pp.63–68, 2011.

Masashi ETO, Ph.D.

Research Manager, Cybersecurity Human Resource Development Research Center, Social Innovation Unit
Network Security, Malware Analysis, Network Operation