KISA Cyber Security Issue Report : Q1 2019

KOREA INTERNET & SECURITY AGENCY


KOREA INTERNET &
SECURITY AGENCY

Contents


KOREA INTERNET &
SECURITY AGENCY

Experts column

1. Honeypot Strategy as a Defense against Intrusion to the Internal Network

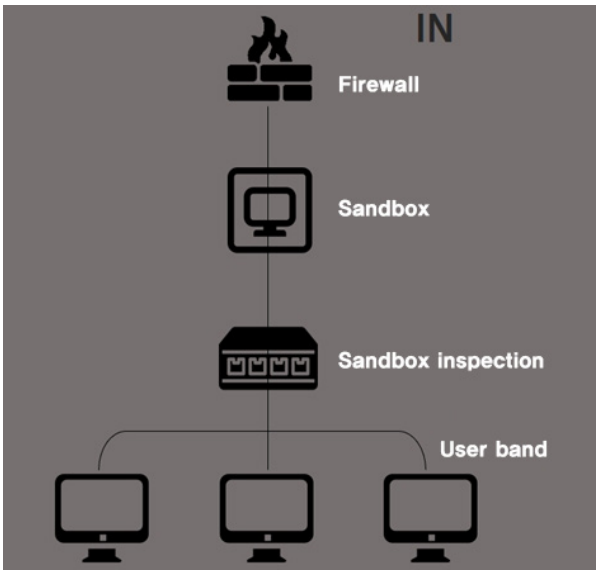
Financial Security Institute / Assistant Manager, Park Mohyun
Financial Security Institute / Associate Senior Assessor, Lee Sangsik
Korea Racing Authority / Manager, Min Moohong

1) Participation in HDCON (Hacking Defense CONtest)

October 1, 2018 was the first day of registering for the 15th HDCON. From the day to December 5 when the final result was announced, we were able to win after we applied all contemplation and ideas we had for the response to intrusion incidents. We now intend to share this contemplation and ideas in the form of an article.

2) Description of problem

The 15th HDCON is the contest of idea planning. The contestants select one of three problems and present their solutions to the problem. After examining the problems, we selected the problem related to intrusion to internal networks as it could show our capability most clearly.



[Figure 1] Description of problem

Company C generates and manages sensitive national technical information and operates its website by constructing the DMZ network separately from user networks. It has outsourced the monitoring and control service to an outside company to protect the service network and operates the security systems such as the firewall and web firewall. It also operates the intrusion incident detection and response system such as the anti-virus, DLP/DRM and NAC for the security of the internal network.

However, an attacker established the initial base in the internal network and configured the beacon-type command control channel by bypassing the preventive systems. The attacker then installed a chain-type backdoor and has breached the information collected from each host by attaching it in a file of 5MB or less and transferring through the webmail server located overseas.

The following prerequisites are given to the problem. Although the company performs the sandbox inspection to block the data transfer of 10MB or more, the security operation focuses on responding to detected events. Contestants cannot purchase the new signature-based security system or outsource the problem-solving to third-party expert. They must suggest security architecture to respond to changing attackers. The defender can access the logs of the internal security system and the network flow data.

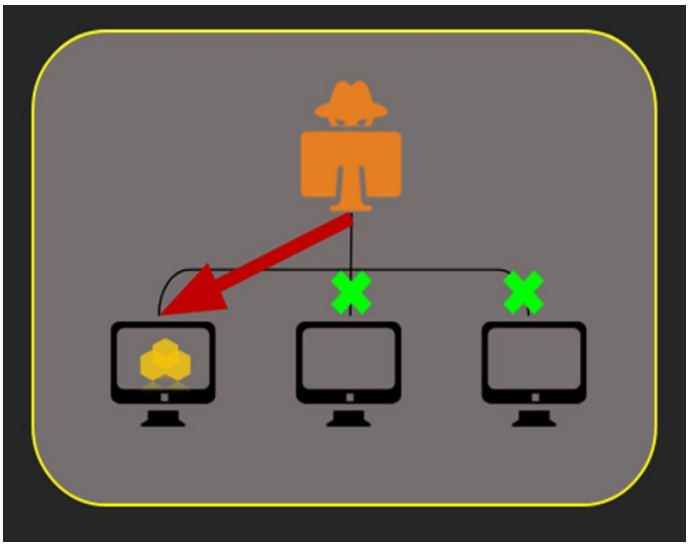
3) Analysis of the problem

We analyzed the problem in depth to find the answer. We paid attention that the problem stated many security systems. It was safe to say that almost all systems that the security manager can procure and operate were stated. However, the situation in the problem was that the attacker had already bypassed or neutralized these systems and compromised the network. Therefore, we had to present the answer in a different direction.

The network flow data caught our eyes initially. The series of all actions such as the beacon-type C2 channel, lateral movement, chain-type backdoor, and data breach is accompanied by network traffic. We thought that the phrase network flow stated in the problem situation was the hint. We decided to include the network flow analysis in the countermeasures as the first step. However, it did not seem sufficient, and we contemplated more.

As we could not come up with adequate response measures from the defender's position, we shifted our thinking. Contemplating which actions the attacker would intend to perform after intruding the internal network and how the attacker would fulfill the intention, we identified honeypot as the countermeasures. We thought that we could detect even the changed attack technique if we use the honeypot to monitor the intention of the attacker and the actions that are essential in the attack process.

4) Typical honeypot

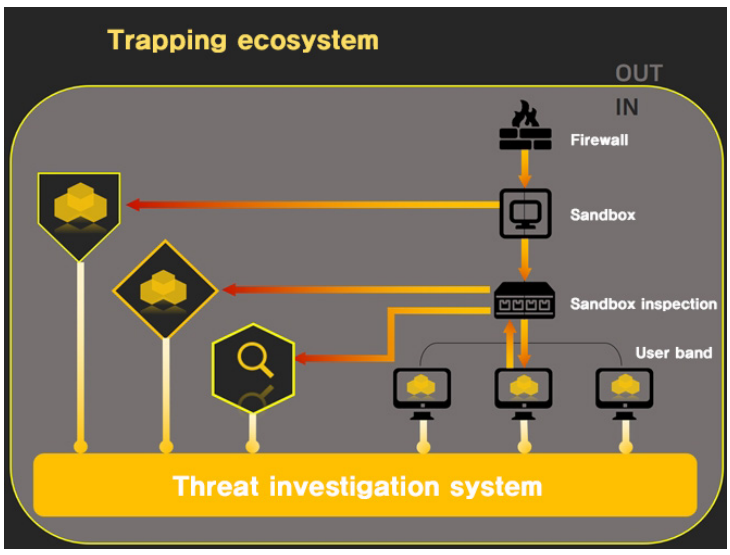


[Figure 2] Typical honeypot

A typical honeypot virtually implements the service that the attacker may access. When the attacker mistakes the honeypot as a normal service and performs various attacks afterward, it analyzes the attacker's actions to find the attack path and hacking technique. It also has the function to protect normal services at the same time by inducing the attacker to the honeypot.

We decided to expand the concept and propose three types of honeypot. We called them the sandbox honeypot, host honeypot, and service honeypot. We then added the network flow analysis contemplated above and the system to visualize the individual honeypot system and traffic analysis log. We called these series of systems the “trapping ecosystem”.

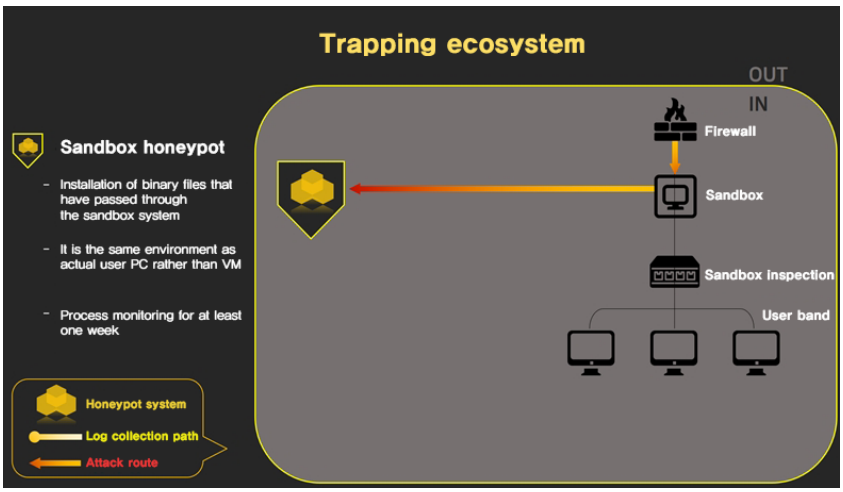
5) Trapping ecosystem



[Figure 3] Intruder trapping system

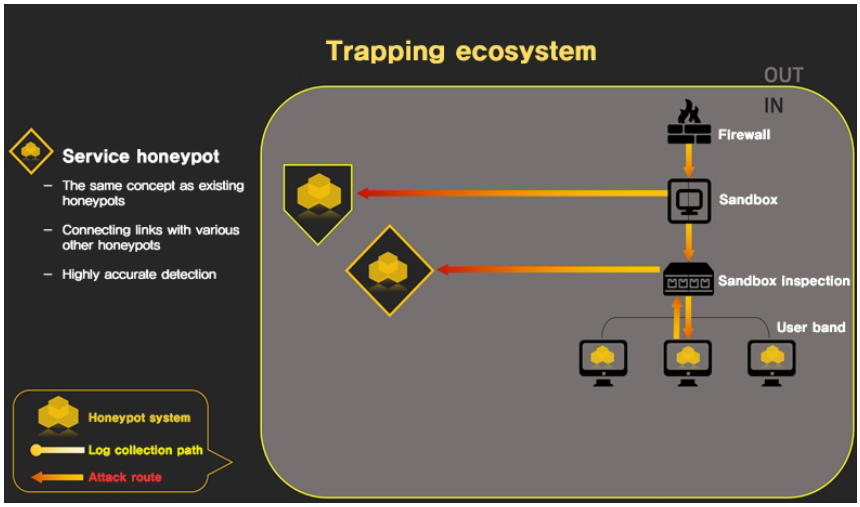
This section describes the trapping ecosystem. The trapping ecosystem consists of five elements that are the sandbox honeypot, host honeypot, service honeypot, threat investigation system, and threat monitoring system. We suggest the layered security strategy that supplements the existing security measures.

The first element is the sandbox honeypot system. The malware types include the malicious code that uses the anti-VM technique that does not operate in a virtual machine and the logic bomb that runs only after a specific time or at the preset date and time. Even the sandbox may not detect the type of malware. Therefore, the sandbox honeypot should be configured to have the same environment as the user terminal and monitor the traffic for a long time to analyze the malware overlooked by the conventional sandbox. The existing sandbox sends the binary file that is suspicious but has no action to the sandbox honeypot for the analysis in the processing unit. If there is a malicious action, the process and binary data are saved in the threat monitoring system.



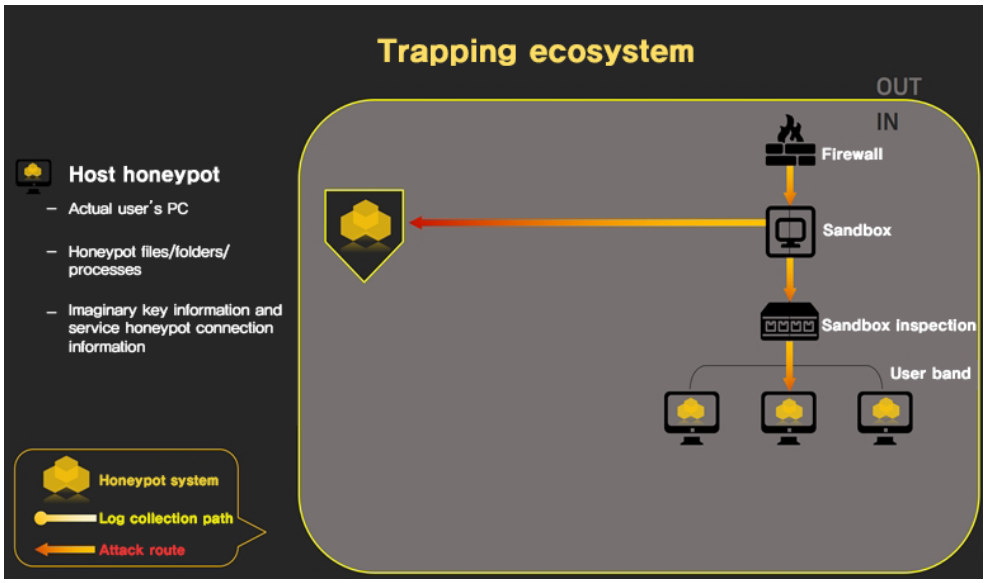
[Figure 4] Sandbox honeypot

The second element is the service honeypot system. This system is the same as the already well-known honeypot concept. It opens the service that may entice the attackers to attack and wait for logins. When an attacker accesses it, the data of the accessing terminal and all actions occurring in the terminal are recorded and saved in the threat monitoring system.



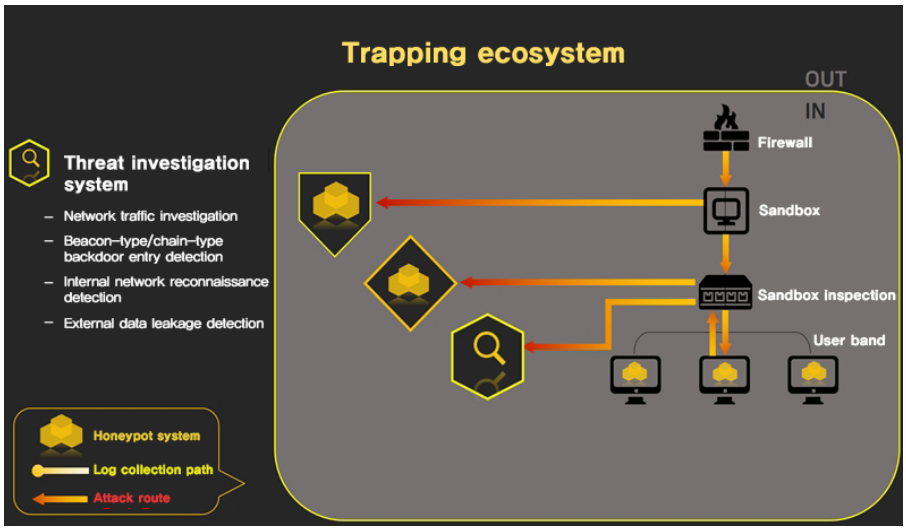
[Figure 5] Service honeypot

The third element is the host honeypot. A host honeypot is not a separate system but files installed in user PCs. The files are those that can entice the intruding attacker to read or breach to outside. All processes that access these files are recorded and saved in the threat monitoring system. The files may be the virtual customer list, false personal information, or confidential documents, and the server and account data that the service honeypot can use for login. For example, the FTP account data open in the service honeypot may be recorded in a file and monitor the access to this file.



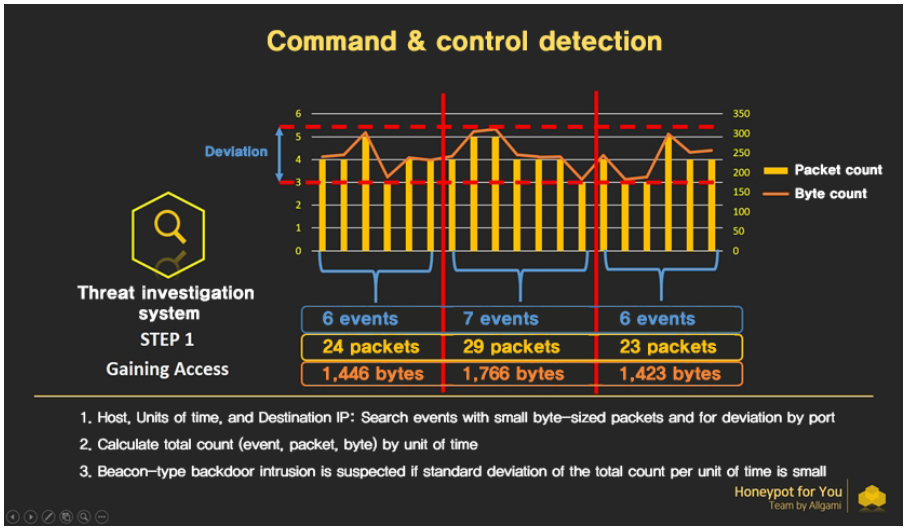
[Figure 6] Host honeypot

The fourth is the threat investigation system. The threat investigation system analyzes the network flow to detect the attacker's actions. The system is designed to detect the actions needed to intrude the internal network and the attack methods given in the problem. It can detect the periodic traffic of beacon–type backdoor implemented at the time of initial infection or detect the chain–time backdoor communication connected between internal hosts afterward. Moreover, it also analyzes whether the port scan is generated for internal reconnaissance and analyzes the traffic related to data breach which is the final step of an attack. Like other systems, all attack data detected by the threat investigation system are saved in the threat monitoring system.



[Figure 7] Intrusion threat investigation system

The threat investigation system detects attack actions through the statistical analysis of the network flow. The internal reconnaissance detection function of the investigation system finds the port scan by identifying the traffic to multiple destination ports from a specific host in a short period.



[Figure 8] Beacon-type backdoor detection

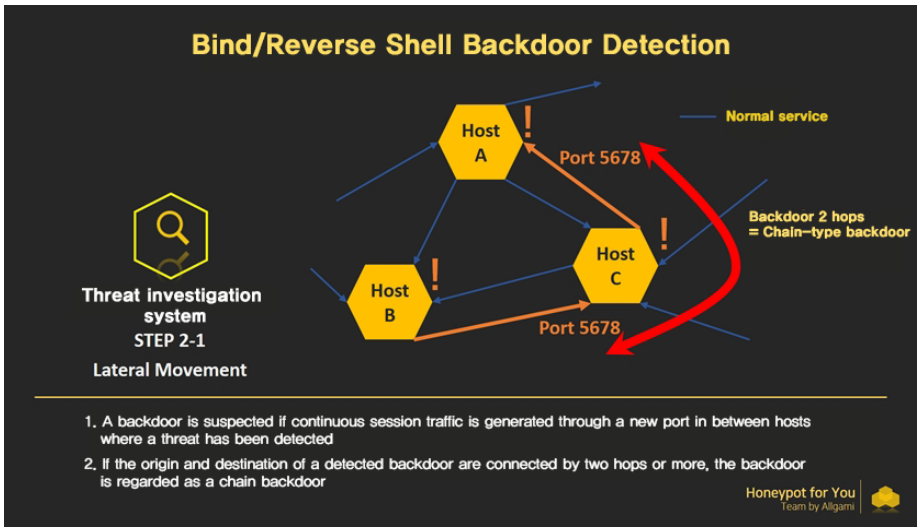
The beacon-type backdoor detection is designed by referring to the characteristics of the beacon traffic. The beacon-type backdoor generates small traffic in the steady interval until the command is sent from the C2 server. We decided to perform the statistical analysis to capture these characteristics.

First, we calculate the traffic statistics of each source-destination IP subject to the packets with small packets and number of bytes. Then the events that have a small standard deviation of packet counts and the small standard deviation of the number of bytes at the same time during an hour. The traffic extracted during the step include the beacon-type backdoor traffic as well as the single event of traffic.

Therefore, we perform the second statistical analysis to extract only the suspicious beacon-type backdoor traffic. We collect the calculated hourly traffic statistics for another analysis to compare the number of traffic events and the total of packet counts and bytes calculated hourly and extract the events that exist in multiple hour bands and have the mall deviation of traffic size in each hour band. We consider the traffics that exist only in a specific hour as the single event traffic and exclude them in the second step.

In other words, the first step is to extract the cases of generating similarly sized traffics between the specific source and destination IP within a unit time (hour), and the second step is to determine if the similarly sized traffics in each unit time. We compared the standard deviation divided by the mean to check the deviation of traffics and suspected those with a specific value or lower as the backdoor traffic. We tried to overcome the scale problem by dividing the standard deviation by mean.

Although the above description may seem long and complicated, we tried to configure the algorithm with the basic statistical values such as the means, count, and standard deviation that the existing commercial or open source big data software can support. In article "3. Detection of and Response to Threats to Internal Networks" of the other column, it is calculated the interval value between the traffics and detected the cases with less dispersion of the traffic intervals. The method can seem more intuitive if it can calculate the traffic interval.



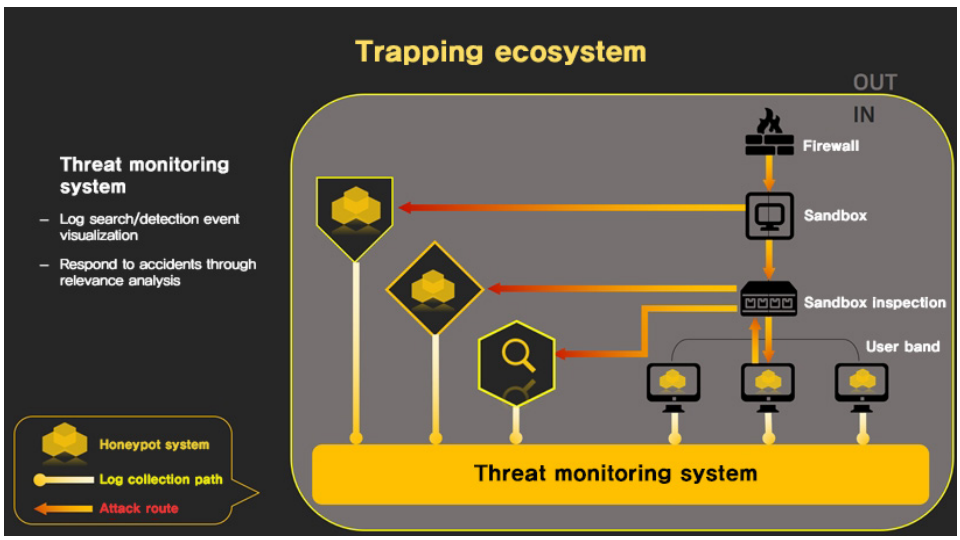
[Figure 9] Chain-type backdoor detection

The chain-type backdoor detection was designed to detect based on the previously detected threats and the traffic generation between the hosts. We can suspect the case of a continuous stream of traffic through the new port between three or more hosts that have already detected the threat as a chain-like backdoor. We collected the security logs recorded by the Windows firewall to determine if the service port was opened and collected the traffic sizes through the network flow data.

If the traffics between user terminals are well controlled, and thus all communications between the terminals can be checked, there is no need to limit the analysis to the hosts with detected threats. The fact that the uncontrolled traffic between multiple hosts is generated is sufficiently suspicious. We can decide whether to analyze the hosts with detected threats or all hosts according to the size and personnel of the whole system.

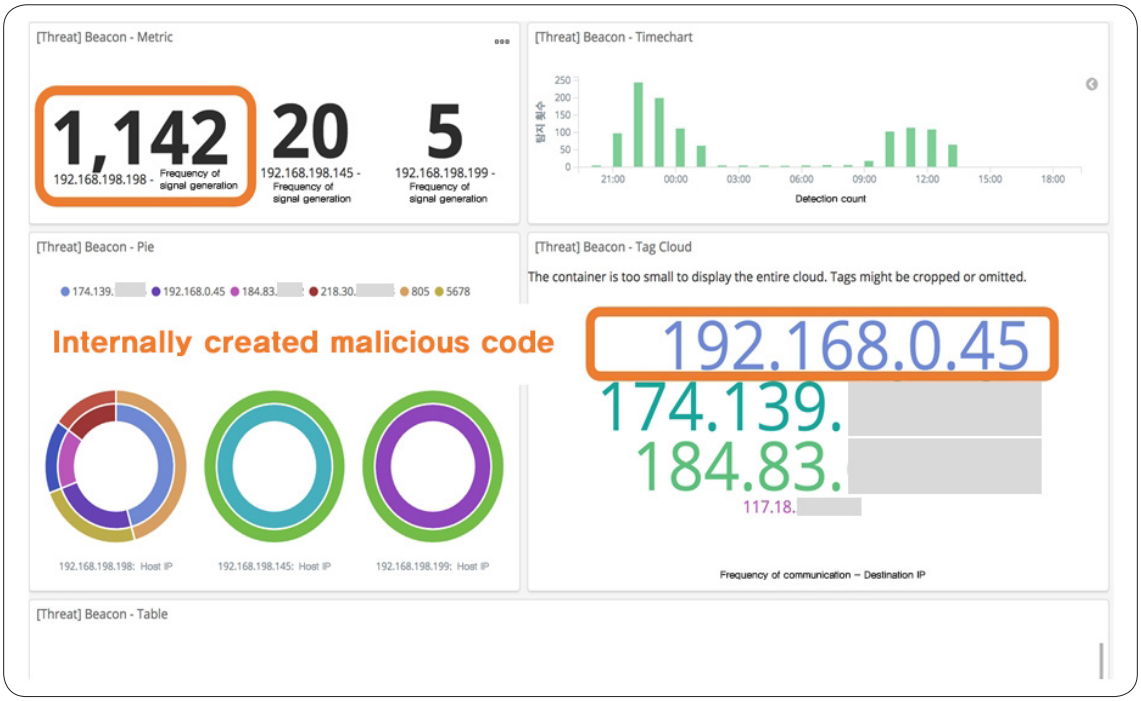
6) Threat monitoring system

We can detect the attack attempts at each stage of security threats using the trapping ecosystem. However, analysts need to check the information at a glance, and the threat monitoring system is devised for that purpose.



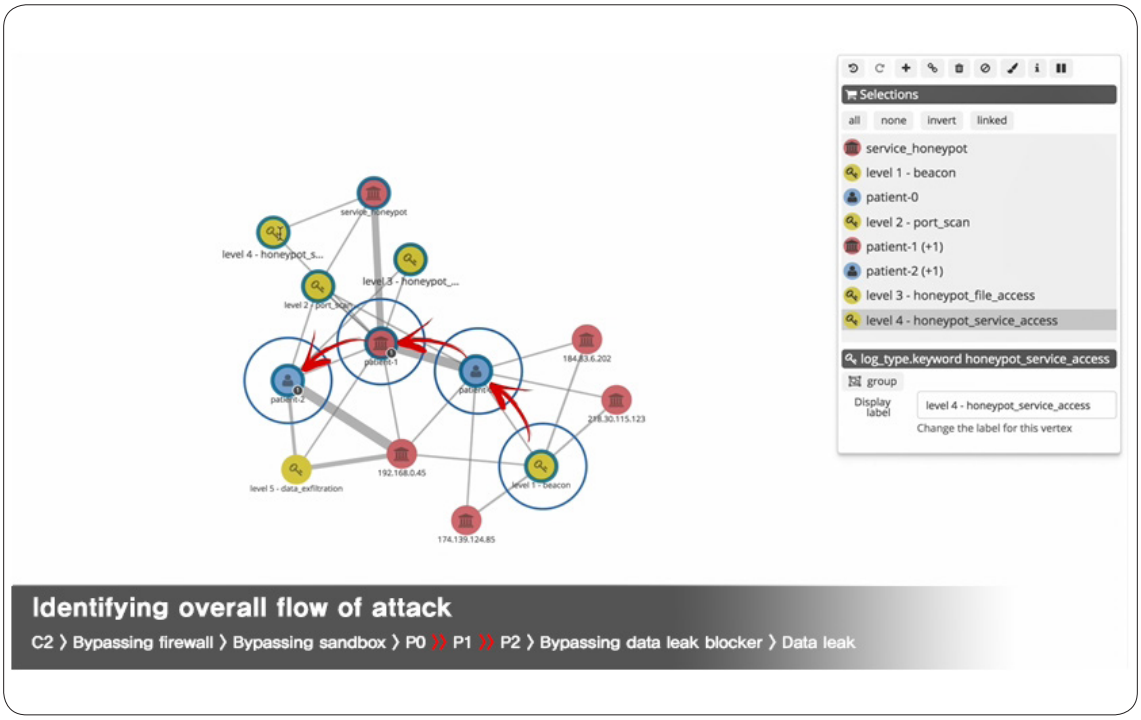
[Figure 10] Threat monitoring system

The threat monitoring system is designed to understand the paths of the attack and compromised hosts intuitively by saving all logs generated in a trapping ecosystem and analyzing the correlation between the logs. Although we implemented it as an open source system because of the constraints of the problem for this HDCON, we can do it more easily and faster with the commercial software.



[Figure 11] Threat monitoring dashboard

One of the key parts of a threat monitoring system is the dashboard. The purpose of the dashboard shown in the figure is to monitor the malware infection. The analysts can check which hosts were infected and which C2 server to which the beacon–type backdoor communication is established through the specifically designed dashboard instead of checking the complicated logs one at a time. The analysis can quickly detect new threats and begin an investigation of existing intrusion using the visual effect of the dashboard.

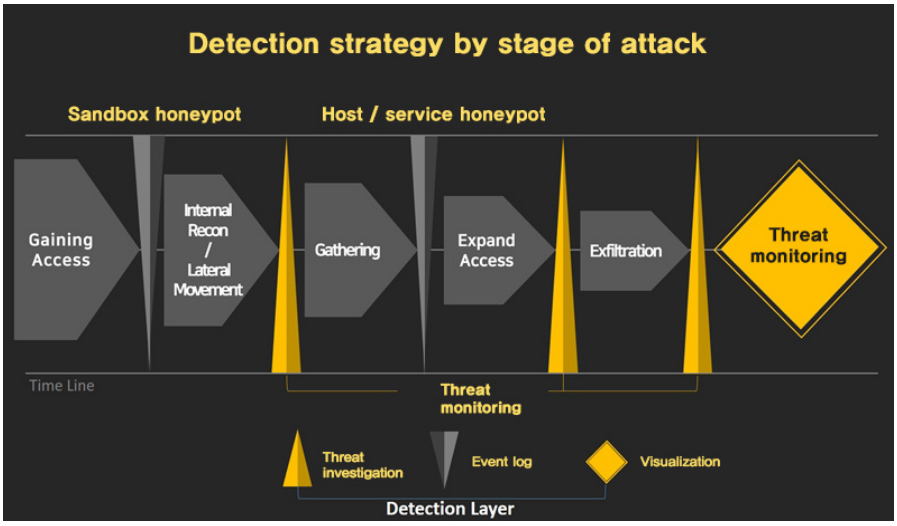


[Figure 12] Threat monitoring graph

The threat monitoring system includes not only the dashboard to check the detection status at each stage of the attack and the graphs to understand the overall flow of attacks. The analysts can check the graph of the nodes that are the hosts related to the compromise types saved in the monitoring system and the edge that is the number of logs to check the relationship between the sizes of the compromised hosts and the hosts.

Using the dashboard and graph that visualize the honeypot events and traffic analysis results saved in the threat monitoring system, the analysts can find and respond to the attack types more quickly and accurately than existing response methods.

7) Layered security



[Figure 13] Detection strategy by stage of attack

A trapping ecosystem takes the layered security strategy that combines multiple systems at each attack stage instead of solving all problems with a solution. The approach enables to supplement in the next stage if it fails to detect an attack in a stage. The system becomes particularly robust by combining the honeypot which has high a accuracy of attack detection but can be avoidable and the network flow analysis which has low accuracy of attack detection but almost impossible to avoid detection. Moreover, the threat monitoring system that can save and visualize the huge volume of log enables the analysts to respond to the threat of the intrusion to the internal network quickly and accurately.

8) In closing

The honeypot is an old concept. It can be a powerful tool for the analysis since the detected events can be considered as the intrusion attempts. Moreover, expanding the concept in existing service honeypot by shifting the paradigm with the active response in the host level and adding the traffic analysis can make the system even more robust. Since each subsystem is implemented as a module, it is possible to borrow only parts of the concept as needed without having to implement the whole system. We conclude this article with the hope that the idea proposed in the HDCON can help the security industry.

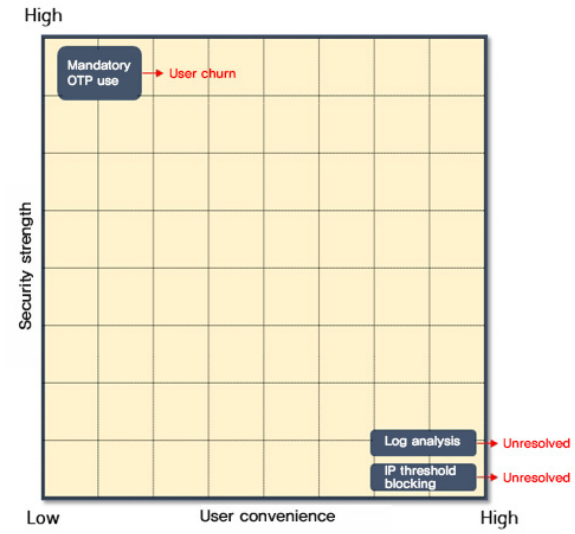
2. Use of Login Status Model to Detect Account Hijacking

Netmarble / Suwon Chon
Netmarble / Geonhui Lee

1) Introduction

Accounts stolen as the result of recent security incidents in vulnerable sites are generating secondary damages. These are cases of account hijacking. The basic problem leading to account hijacking is an authentication process that only checks for matching of the user attempting login and the account user using only their account data (ID and password). The damage by account hijacking can thus be reduced by providing various authentication means in addition to the account data and using the method of multi-factor authentication through active user intervention.

One of the most well-known multi-factor authentication methods is OTP (One-Time Password). OTP generates a single-use random number by a special algorithm instead of a pre-specified password. When a user attempts to log in, the system performs authentication with the account data, then with the additional authentication using OTP input to complete the login process. Although multi-factor authentication is effective in preventing account hijacking by authenticating the matching of the user attempting login and the account user via active intervention of the user, it reduces user convenience as it requires additional user input each time.



[Figure 14] Analysis of account steal problem

In other words, using multi-factor authentication can increase the security level but with the downside of lower user convenience. It is therefore used mostly in the financial sector where additional security can be mandatorily enforced. As low convenience in the general service environment can lead to rejection by users, solutions such as blocking by IP critical value and log analysis are being used in response.

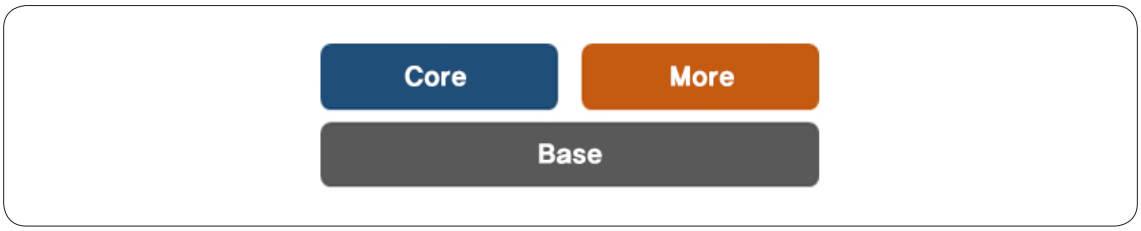
However, these methods cannot solve the intrinsic problem of account hijacking due to low-security strength even though it provides convenience to users. Thus, the dilemma of security level vs. convenience has limitations as shown in [Figure 14]. Many enterprises struggle with business strategies to deal with account hijacking while balancing the security level and user convenience. The following method presents a hybrid solution to account hijacking with the appropriate balance of the security level and user convenience.

2) Countermeasures to Account Hijacking

[Figure 15] shows the architecture with a Base and formed by Core and More to effectively deal with account hijacking.

First, the Base is the method of detecting mechanical login attempts. Mechanical attempts include repetitive login attempts and attempts to log in by altering of IP. Detecting mechanical login is the first requirement in preventing account hijacking. Second, the Core verifies whether the user attempting to log in matches the user of the account. This represents the intrinsic problem leading to account hijacking and is the most important method needed for solving the problem. Finally, More is the method of responding to account hijacking.

As described above, effective response is necessary after detecting an account hijacking with Base and Core. In conclusion, a good solution for prevention of account hijacking must fulfill all of the above criteria.



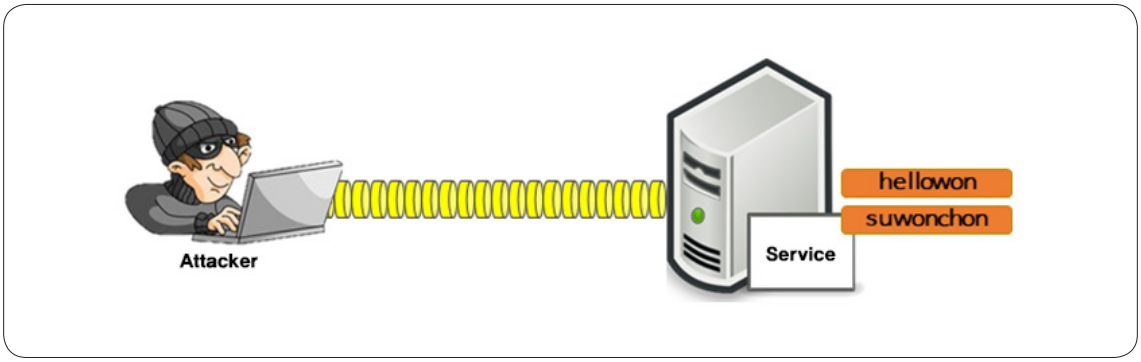
[Figure 15] Summary of suggested method

2-1) Base: Detecting mechanical login attempts

A mechanical login refers to a login attempt using a program or a macro. Such behaviors can be expressed by the model shown in [Figure 16] and [Figure 17].

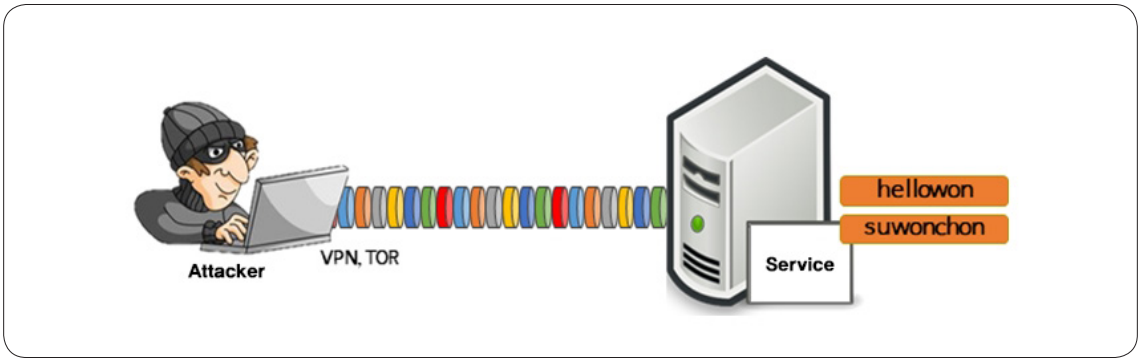
An attacker attempts to log in to multiple accounts from an IP.

※ There can be one or more accounts that attempt to log in.



[Figure 16] Login attempt by single IP

It uses a proxy, VPN (Virtual Private Network), or Tor (The Onion Router) to alter the IP and attempt to log in to multiple accounts.



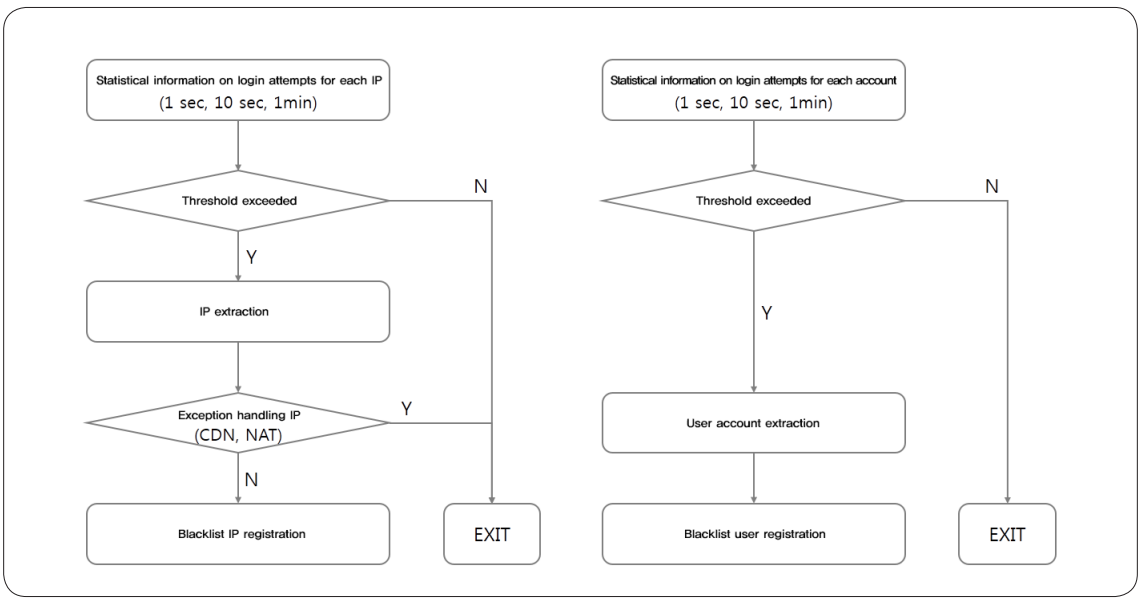
[Figure 17] Login attempt by modulated IP

The reason for defining these models is the need to approach the mechanical login attempts differently according to each case. Login attempts can be categorized by four cases.

- Multiple login attempts to a single account of a single IP
- Multiple login attempts to multiple accounts of a single IP
- Multiple login attempts to a single account of multiple IPs
- Multiple login attempts to multiple accounts of multiple IPs

A typical method is used to respond to the login attempt of an IP. The method accumulates the login attempts to each IP and registers the IP in the blacklist to block if the accumulated value during a specific period is higher than the critical value. However, the IP that attempts to log in may be CDN or NAT and have the same IP. Since blocking such IPs can cause service problems, exception handling is necessary.

Another case is login attempts to multiple IPs. A query of account data in the account DB is necessary for a user to log in. We can generate the statistics for login attempts to each account by accumulating the number of queries. Then, users that show abnormal login attempts during a specific period are registered in the blacklist.



[Figure 18] Mechanical method of login attempt detection

The difference in the above two cases is that the single IP case uses a blacklist of IPs while the latter uses a blacklist of users. Only the security administrator can release the blacklist of IPs, and the blacklist of users can be released through multiple-factor authentication. "2-3) Countermeasures to Account Hijacking" describes the blacklist.

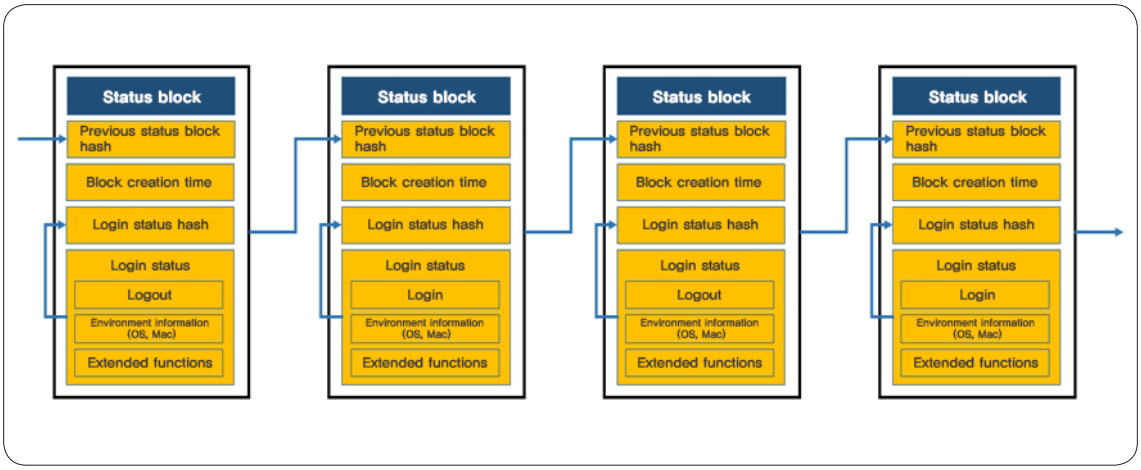
2-2) Verifying the user attempting to log in and the user of the account

It is necessary to verify matching of the user attempting to log in and the user of the account in order to solve the problem of account hijacking. Multiple-factor authentication is effective and powerful since it verifies matching through an additional authentication method such as OTP requiring active intervention of the user. However, requiring additional input for each login is inconvenient to users, and thus improvement is necessary.

We suggest defining the login status model and generating, sharing, and verifying the hash to minimize the inconvenience of multi-factor authentication.

2-2-1) Login status model

A user (hereafter 'client') logs in to use a service (hereafter 'server'). The login status of a client repeats through logout, login and continuously changes. [Figure 19] shows such status changes in the login status model.



[Figure 19] Login status model

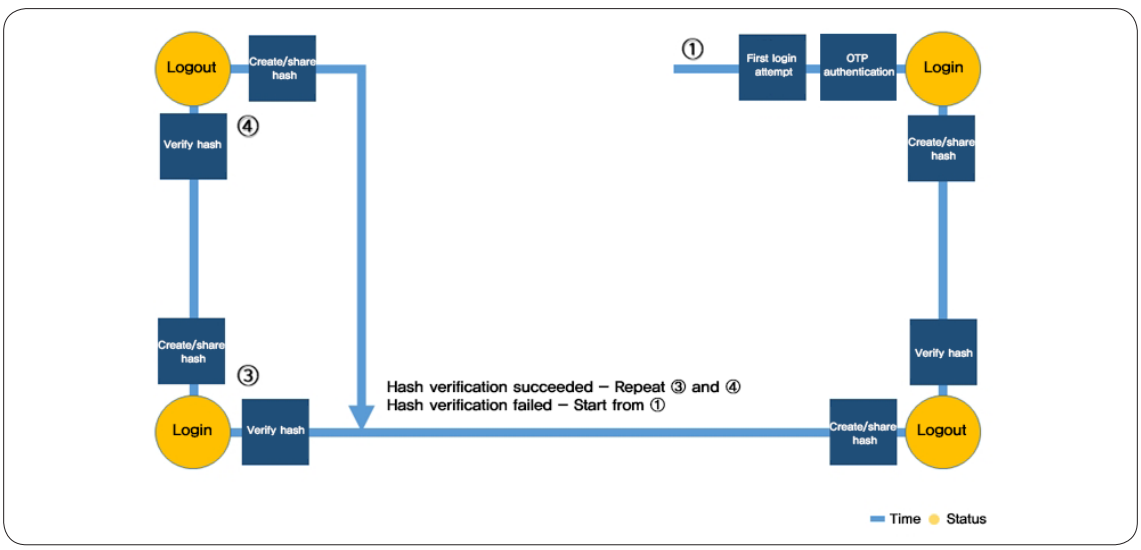
The initial status block is generated after the successful login by the client. The client authenticates using the account data to log in to the service and additionally authenticates using multi-factor authentication to successfully complete login. The server configures the environment data and login status data of the client into a block and generates the login status hash. It then generates a new status block hash with the time of the previous login status hash generation and the previous status block hash and shares it with the client. There is no previous status block for the initial status block hash.

If the client status changes to logout, the server creates a new block hash with the previous status block hash, login status hash, and generation time and shares it with the client. This process repeats generation and sharing and configures the blocks in a chain form.

The environmental data collected from the client is used to prevent theft of the shared hash. Refer to "4) Performance Evaluation and Benefits" for additional information.

2-2-2) Operation of the login status model

[Figure 20] shows the process of application of the login status model and the operation of the client and server generation, sharing, and verification of the status block hash.



[Figure 20] Operation of the login status model

(1) Initial login attempt and authentication with the account data

(2) Request for OTP authentication, followed by completion of multi-factor authentication

(3) Login

(4) Generation of the status block hash by the server and sharing it with the client

(5) Logout

(6) Verification of the hash shared in the previous status

※ It can repeat the verification of whether the user requesting the logout is the same user that logged in in order to respond to attacks like session hijacking.

※ User login can be verified if it is used with main functions such as password change.

(7) Generation and sharing of the hash again after successful logout

(8) Repeat process

※ OTP is not repeated, and the status block hash generated and shared during the login status change plays this role. → If the login status changes, the server verifies the previous status and authenticates whether the user is in the continuous line.

(9) Steps ③ and ④ are repeated if status block hash verification is successful.

※ The process returns to Step ① to perform the OTP if the verification fails.

2-2-3) Failed status hash block verification

Failure of verification can be summarized by the following three cases.

(1) New installation of an application

(2) Intentional deletion of application data

(3) Attempt to log in with a stolen account

The key to the suggested method generates and shares the login status block hash and configures it in a chain according to the login status change. The server then verifies the previous hash at the time of a status change. It improves user convenience compared to using OTP each time as it uses minimal OTP only when hash verification fails. It can also assure a high security level similar to use of OTP. The biggest advantage is in quickly recognizing account hijacking and responding to it. Refer to "4) Performance Evaluation and Benefits" for additional information.

2-3) Countermeasures to Account Hijacking

The blacklist of IPs and blacklist of users are utilized as countermeasures to account hijacking. The purpose of the blacklist of IPs is to block access to the authentication server. The blacklist of users is to block login by listed accounts selectively.

Criteria for registering in the blacklist

(1) A user is registered in the blacklist if OTP failure occurs 5 times consecutively.

(2) An IP is registered in the blacklist if it attempts a mechanical login.

(3) A user attempting a mechanical login to multiple IPs is registered in the blacklist of users.

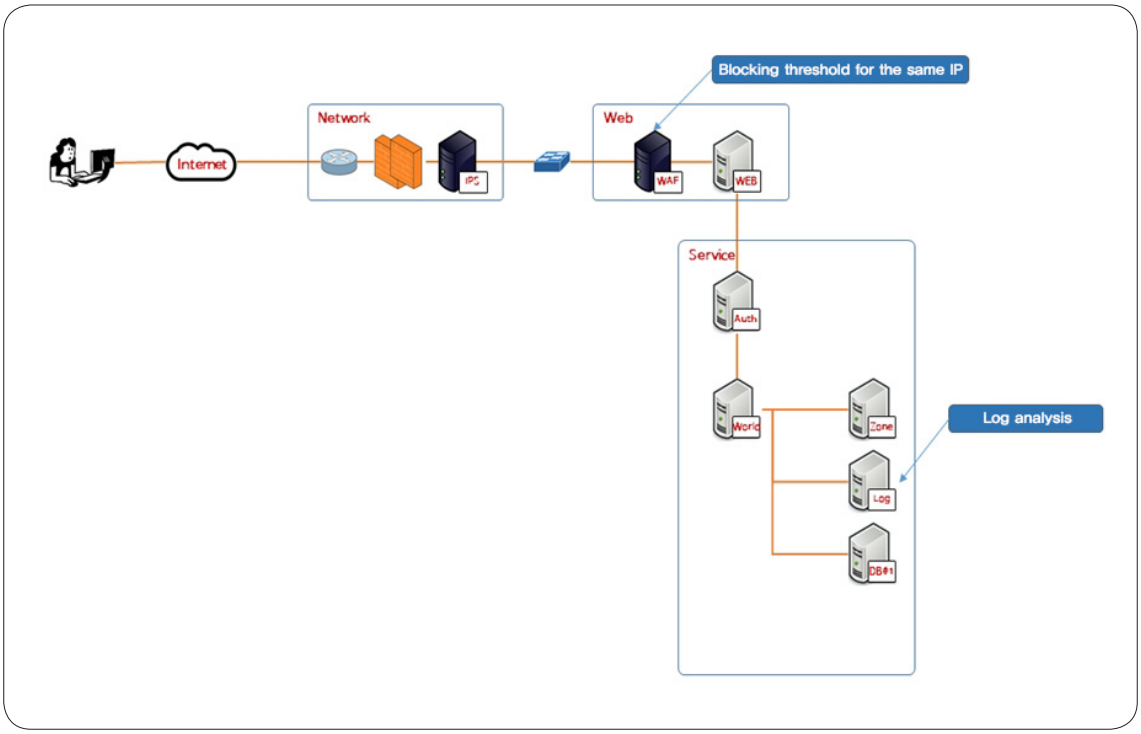
Criteria for removing a user of an IP from the blacklist

(1) Removing an IP from the blacklist requires approval from the security manager.

(2) A user can remove their self from the list without the security manager's approval after OTP verification. Thus, a normal user registered in the blacklist can autonomously remove their self.

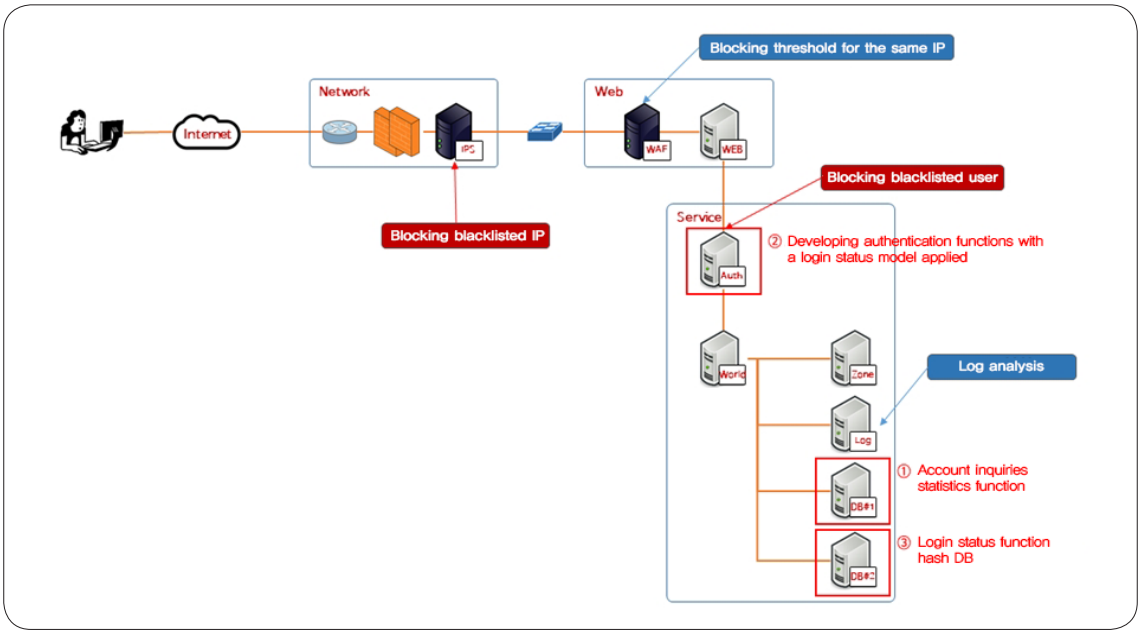
3) System Configuration Using the Suggested Method

[Figure 21] shows the typical service environment. It configures the critical value blocking and log analysis of the matching IP as a solution to prevent damage from account hijacking.



[Figure 21] Typical system environment

As shown in [Figure 21], the suggested method is software configuration and application of mechanical login detection, authentication using the login status model, and countermeasures to account hijacking to the typical system configuration. The following functions are used.



[Figure 22] System configuration that applied by suggested method

First is the statistics of account queries in the Base. This calculates the statistical data of the account and registers in the blacklist of users the account that attempts to log in by more than the critical value. It is effective against mechanical login attempts for not only single IPs but also IP modification.

The second is authentication applying the login status model and storing the status block hash in the DB of the Core. This is configured separately from the account DB. An attacker logging in with a stolen account is subject to secondary authentication since the attacker does not have the status block hash. The attacker cannot log in if the secondary authentication fails, and is registered in the blacklist of users to be blocked from logging in when the failure is repeated. Moreover, the system configured with the suggested method improves user convenience since the user does not have to input OTP at each login. The suggested method detects account hijacking and registers IPs and users in the blacklist according to detection results.

4) Performance Evaluation and Benefits

Since it is difficult to compare the suggested method quantitatively, we compiled the best and worst cases for comparison.

4-1) Performance Evaluation

The best case is using OTP once to update the status block then later logging in without the OTP.

For the worst case, we assumed the following case.

(1) A user account was stolen from a vulnerable site, and the site unfortunately stored the ID and password in plain text.

(2) To make matters worse, we assumed that the attacker also stored the status block hash of the stolen user account. The server and client share the status block hash, and it is assumed that it was stolen by malware from the client.

(3) We also assumed that the attacker found the internal data of the server. The status block collects the MAC address and OS data to prevent theft. The collected data can be configured differently. However, it is assumed that the attacker had identified all internal information to neutralize the security measures.

(4) In the end, the attacker can log in without OTP. In other words, the attacker can log in with the victim's account by bypassing all security measures that we suggest in the worst case.

Despite that, we believe that the suggested method is the required choice.

4-2) Benefits

Users are usually not aware of their accounts being stolen, and change the password only after the damage has occurred. If the suggested method is used, the user whose account is hijacked fails the hash verification and is required to input OTP during login. The user checks the warning message that their account may have been stolen. This is the fastest way to recognize an account hijacking.

However, typical users may not be sensitive to the risk of their account being stolen. Even if the user ignores the warning, in such a case the hash is changed when the user authenticates OTP, and the hash stolen by the attacker is immediately neutralized. The user does not have to change their password. This is because the simple login has the effect of changing the password.

The suggested method can recognize the account hijacking and respond to it faster than any current methods. Since strong security means thorough management and control, it inevitably imposes the obligation of multi-factor authentication on users and thus lowers their convenience. However, the suggested method has the advantage of both security and convenience by having the system, not the user, perform the multi-factor authentication.

5) Conclusion

The current method completes the login when the user enters the account data (ID and password). However, the suggested method applies the login status model and has the client and server share the status block hash generated according to the login status change. The login process then verifies the status block hash.

The process detects and blocks the illegal account hijacking. Moreover, compared to the mandatory use of OTP, this improves user convenience and can detect attacks such as session hijacking in addition to account hijacking. Finally, it can contribute to minimizing damages via the fastest recognition and response to an account hijacking.


KOREA INTERNET &
SECURITY AGENCY

3. Detection of and Response to Threats to Internal Networks

Global Information Security, eBay inc.
APAC Manager, Jay Seo

1) Introduction

Intrusion into internal networks is the most frequent APT attack, and recently is the attack method that has caused the most difficulties for security officials. The personal information breach case at Interpark in 2016 and Good Choice in 2017 and the infection of INTERNETNAYANA by ransomware in 2017 occurring within the past three years all started with intrusions into the internal network. Why do these incidents happen continuously?

Enterprise security systems must be understood in order to understand intrusions into internal networks. The security models currently adopted by most enterprises use the method indicated in the following document.

> "Thinking about firewalls." Proceedings of Second International Conference on Systems and Network Security and Management (SANS-II). Ranum, Marcus J. , Vol. 8. 1993.

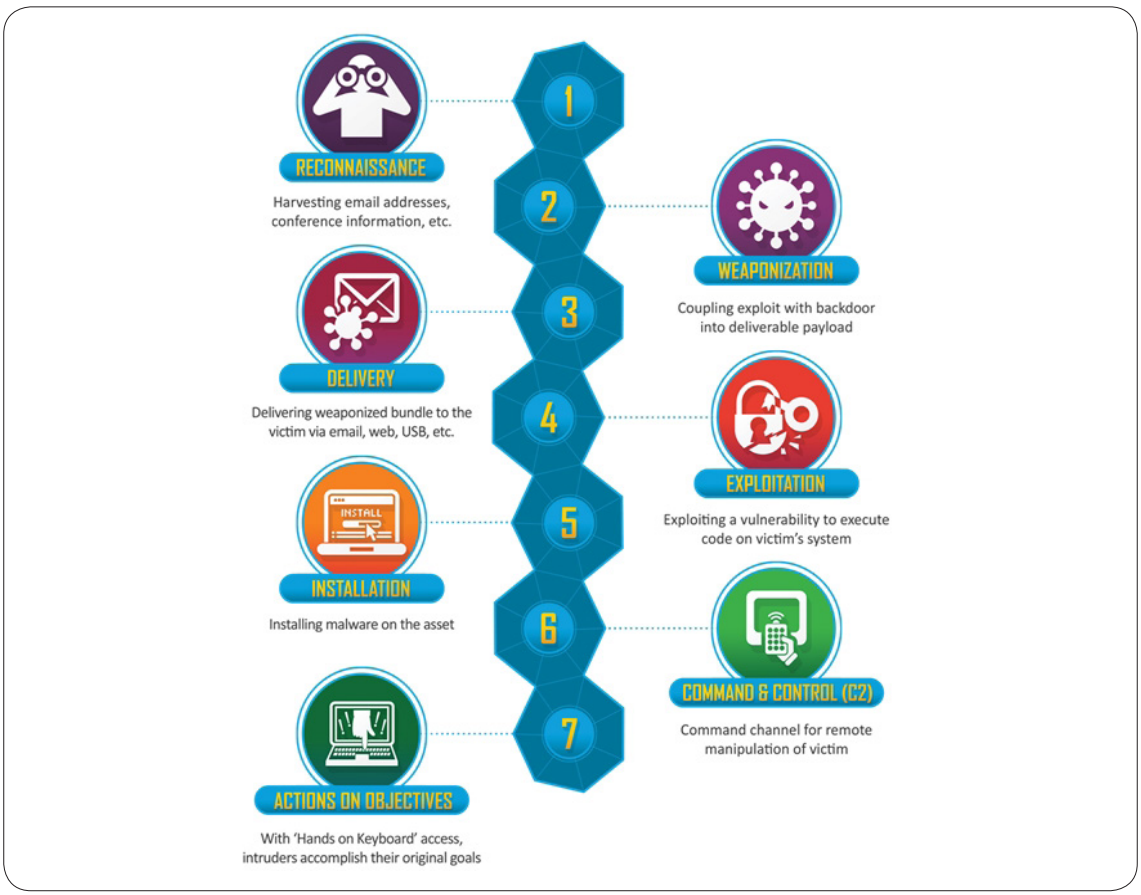
Marcus Ranum, a.k.a the father of the firewall, wrote this document in 1993. The document suggests a method to protect server resources that are connected to the Internet directly in the early years when there were no protective measures. The method involves using a network access control system called a firewall to enhance security by logically separating the Internet and the enterprise network and by applying a security policy to allow or block access.

A firewall blocks attackers from direct access the server from the Internet and contributed greatly to security as a result. However, at the time when Marcus suggested this model, most of the attackers would directly access the target to perform the attack. At the time, there was no automatic attack targeted network using malware.

A firewall functions as a border security system between the Internet and an enterprise network. That is, it only decides whether to allow or block access and cannot apply more detailed policy. The attackers recognized the limitations and began using another method of attack. This is the method of disabling the firewall by bypassing the border security of the firewall.

The intrusion into the internal network (often called an APT attack) begins when an attacker takes control of the internal resources then uses them as an attack base. The attacker uses the resources of the internal network until the exfiltration of the resources. This is the most serious threat to existing security models based on the reliability of internal resources. The Cyber Kill Chain [1] model, announced by the US-based military firm Lockheed Martin shows the typical attack process of such attackers.

1) https://lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html



[Figure 23] Cyber Kill Chain

In this case, how should we effectively respond to such an intrusion into the internal network? Although the boundary–based security models are effective in detecting and blocking attacks initiated from outside, there is still a limitation in blocking attacks that begin inside the firewall.

As shown in [Figure 23], the internal resources taken over by an outside entity communicate with the outside in Step 6, Command & Control. In other words, it is where an attacker does not access from outside but rather connects the internal resources with the outside C&C. For this reason it is necessary to monitor the resources that are leaked from inside and focus on monitoring the outbound traffic that detects the attacks.

2) Status of Response to Threats to the Internal Network

Let us consider what constitutes a threat to the internal network. As discussed in the introduction, the firewall securely protects the enterprise network environment. It is therefore impossible for an attacker to access the internal network directly. So why is there a threat to internal networks? And why do attackers target internal networks for attack? The reasons can be identified as follows.

The protection level of web servers and DB servers are high as they are exposed on the Internet. There are standard management processes and dedicated administrators for web servers and DB servers. However, the security levels of PC users are not as high as those of the servers. In other words, the major weakness is that the application of security can vary widely according to the PC user's skill level.

Let us then review how existing information security has been performed.

■ Priority on protecting service networks

Many enterprises perform security monitoring and control of a DMZ (service network). They install an intrusion blocking system and operate an intrusion detection system or intrusion prevention system. Some install a web separate firewall to respond specifically to attacks on web servers. However, the purpose of these systems is to protect web servers, and thus they are not very effective against intrusions into internal networks. The ultimate target of an attacker is a PC, not a web server. An attacker cannot take control of a PC with an attack against a web server.

However, protection of service networks still holds the highest priority in information security for enterprises.

■ Operation of the event-based detection system

Numerous security systems are installed on the service network, since protecting it is the highest priority. In other words, most security systems operate in the DMZ section. The security systems operating in the DMZ section detect events based on specific incidents. This method provides intuitive information to users. It has the advantage that even administrators who are not security experts can understand the events. However, this technique also has the following disadvantages :

- Most detections are based on patterns (signatures). Thus, the ratio of Type 1 errors (false positive) and Type 2 errors (false negative) is relatively high.
- It collects network traffic and analyzes it to detect malicious intent. However, it cannot detect malicious intent if there is no pattern at the time of detection, i.e. if the detection point has passed. This is because it cannot retrieve a packet that has already passed to analyze it again.
- The detected event information only provides the detection result. This information may not be sufficient to explain the threat or overall vulnerability since it is merely a summary of the situation at the time.

■ Operation of anti-virus or DLP (Data Loss Protection) / DRM (Digital Right Management)

While a anti-virus detects only known malware, an attacker uses unknown one. That does not mean that anti-viruses should not be used. It is very important to defend against known malware as well.

DLP and DRM protect data and files. In other words, they do not detect the attacking behavior of attackers. The main function of NAC (Network Access Control) is to authenticate various devices that access the network and block network access by unauthorized systems. Its main purpose is to prevent an access from unauthorized devices within internal network. However, what role can NAC take if an attacker has already taken control of an internal computer using malware? It can apply security review of the accessing devices, but the detection rate would not be very high.

Such security systems might be not effective in blocking an attack against an internal network.

3) Response Measures for Threat to Internal Network

As discussed above, security of an internal network is not completed simply by implementing security systems. Moreover, many enterprises do not operate the security of their internal networks at such a high level. Considering the conflicting relationship between security and convenience and thus difficulty to coexist, it is safe to say that the current security of internal networks completes security at the cost of inconvenience to internal users.

There are prerequisites for enhancing the security of an internal network. What many information security managers overlook is the fact that information protection functions in the IT environment. In other words, IT comes first, and information security next.

In small enterprises, both IT and security are managed in one department without clear separation. However, for large-scale environments they are managed by separate departments, and the tasks of each are clearly defined. Unfortunately, security cannot be the main objective. The reason is that security comes after IT. There is no grounds for security without IT. Thus, the two departments must collaborate. The system is needed that prioritizes IT at normal times and prioritizes security in the case of an incident.

What should be done to enhance the security of an internal network? Until now, security measures have defined threats, analyzed threats, and established countermeasures to offset threats. Since the existing network firewall cannot defend against SQL injection threats, the web firewall was introduced for protection. NAC was introduced due to the threat of unauthorized terminals accessing the internal network to leak data. These cases show that there are threats to all defenses. Unfortunately, however, we cannot identify all possible threats. We would be able to establish security measures perfectly if we could list all possible threats.

The threat to an internal network is a case of an attack having already found its way inside the network by neutralizing the firewall, the boundary security system that we trust the most. The attacker has already acquired the same privileges as an internal user. By this point, the security manager must now judge whether the attack is conducted by an employee on the inside or by an attacker.

Anti-virus, DRM/DLP, firewall, etc. are not much help in this case. The author thus presents an approach that is different from the method of linking the threat and response on a one-to-one basis. This method is detection based on the network access behaviors of attackers. Behavior-based detection is already popularized technology. It can be used in both the host base and network base.

Behavior-based detection requires first collecting a large volume of data and analyzing it to establish a pattern model for the attack technique. Although the technology was not sufficient in the past, this approach has now become feasible thanks to significant improvements in the ability to analyze large volumes of data such as through big data and the calculation capabilities of computers.

The biggest prerequisite in a threat to an internal network is that the internal resources occupied by an attacker continuously communicate with the attacker outside the network. This is because internal resources must link with the attacker to relay commands or to play the role of an access point since the attacker uses an internal resource as an intermediary instead of performing the attack directly.

Such an access channel is called a C2 (Command & Control) channel, and the point of access used by the attacker is called the C2 server (same as the C&D server). In this case, finding the internal resources that communicate with a C2 server is a crucial part of dealing with the threat to the internal network. That is why it is important to monitor outbound traffic to the Internet.

Another point is that the monitoring of outbound traffic should not be event based. As mentioned above, the event-based detection system detects malware only when it meets the detection criteria. The problem is that such detection criteria can be false or a missed detection in many cases. Thus, the traffic must be monitored based on the report that investigates all traffic rather than with event-based monitoring. Although the size of the internal network determines the volume of traffic, the fact that even open sources like Elastic process hundreds of gigabytes of data in a day shows that the collection and analysis should be feasible. The question is what to analyze.

Let us examine the system and operations to satisfy the premise.

■ System needed to enhance the security of the internal network

The logs must be collected first to detect attack behaviors. This means that the system can only detect what it can see. It is unrealistic to establish a security policy that does not assure visibility.

All traffic to the Internet must be collected without loss. Once the collection target is determined, next is the collection range. Up to what range of traffic going to the Internet should we collect? Should we collect all packets? Should we collect up to the TCP/IP header provided by the L3 firewall? The most guaranteed method is, of course, to collect and save all packets. All of the information is contained in the packets. A security manager can reconfigure the attack if all packets are available. They can also understand which information was leaked. However, saving all packets can be costly. Collection can also be done with tcpdump. However, a better system is necessary to extract the desired data of the desired time from the total data to analyze it.

The NetFlow and L3 firewalls can handle only TCP layer data, and thus have limitations in analyzing attacker behaviors in detail. Moreover, NetFlow uses sampling packet instead of whole traffic. This type of information goes against the premise that all data must be collected, and thus is not recommended. We recommend the following open-source programs.

○ Zeek network monitor [2)]

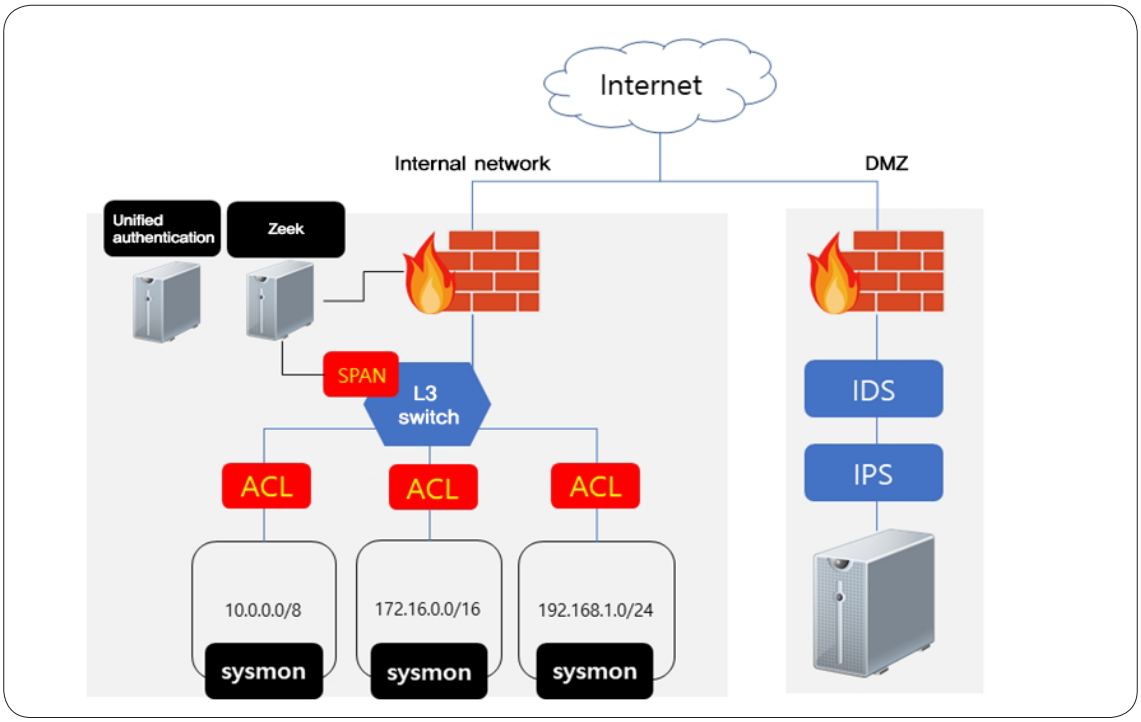
It was called 'IDS' when it used the name 'Bro', but now uses another name, 'Zeek'. This is a reliable open source that can analyze the application layer. Zeek has a very open license even for an open source and does not require any additional cost. However, it needs the Linux server to run the program and an integrated log analysis system to collect the log generated by the program in order to work smoothly as a countermeasure.

2) https://www.zeek.org/

○ Sysmon [3)]

Sysmon is a Windows program distributed by Microsoft. It can be very useful since most of the clients in a business internal network are Windows users. It can provide detailed security information in addition to the events generated by the operating system. The typical information is the process creation and network connection. One can use Sysmon to examine what processes are connected to the network. The logs installed and generated by PCs can be collected and analyzed by the log analysis system using the log collection agent.

These two programs are the minimum configuration to secure visibility for an internal network. Once the collection software is available, the system must be configured.



[Figure 24] Diagram of threat detection system

3) https://docs.microsoft.com/en-us/sysinternals/downloads/sysmon)

The configuration is not so complicated. The key objective is to minimize changes of the network in operation. Install sysmon in each PC in operation. You can download the installation document from the Microsoft[4] website. It is recommended to install Zeek in a Linux server or FreeBSD. One way for Zeek to monitor traffic is the reproduction of packets from the backbone switch in a position that enables monitoring of all traffic to the Internet. Another way is to mirror the traffic using a tab switch. The mirroring of traffic requires a careful selection of the section for mirroring. [Figure 24] shows the reproduction of all packets by setting up a span port in the L3 switch. The traffic destined for the firewall, which is the higher level layer, and the traffic that goes over the network sections are visible. In this way it can detect lateral movements that repeat an attack on the internal resources beyond the network sections. However, it is difficult to detect lateral movements generated inside the same network.

There are many network anomalies that can be construed as attacker behavior. Here, we review the detection of the external network connection beacon channel and the upload of a large volume of data as leading examples.

4) https://docs.microsoft.com/ko-kr/sysinternals/downloads/sysmon

■ Beacon channel detection

O Detection principle

Statistical analysis uses the variance value. The variance is a measure of how far each value in the data set is from the mean. In other words, a large variance means the deviation between the data is irregular. Conversely, a small variance means that the deviation between the data is not large. We need the timestamp, origin IP, and destination IP of each log for the analysis.

○ Application of the detection principle

1. Sort the dataset based on the destination IP and segment them by matching destination IP.
2. Sort the dataset of the same destination in descending order of timestamp.
3. Subtract the timestamp of $(N+1)^{th}$ log from the timestamp of the N^{th} log and allocate it in TDn.
4. Repeat Step 3 on all logs. This results in a set of TD1 and TD2–TDn.
5. Calculate the variance of the obtained set.

○ Analysis of the variance value

A smaller variance means that there is less difference between the data, and it means that there is no time difference between the logs. It can be inferred that the login occurs regularly by a similar time difference. Unlike typical user logins, the logins by C2 rarely occur by random time differences since a programming logic generates them. The detection algorithm is simple, but is surprisingly effective. The critical domain of the variance value of suspicious communication differs for each network environment.

○ Considerations for application

It would be nice if the simple variance value could detect malicious behaviors, but there are many cases to consider when applying it in an actual environment. The technique may detect a normal case as the beacon channel in actual application. A typical example is the anti-virus update server. The internal hosts access the update server regularly for anti-virus updates. In these cases, the destination IP must continuously process the white list to eliminate the possibility of false positives in advance. The Windows updates and external backup sites would also show the anomalies if white list processing is continuously performed after checking the destination.

■ Detection of anomalies using the Z-score value

○ Utilization of Z-score

In statistics, the Z-score is a technique used for calculating the standard score. However, here it is used to find outliers that are higher than other users. In other words, it is used to check how many more files a specific user sends to the outside than other users. Let us assume that the system finds a particular user sending 5 GB to the outside. The simple value of 5 GB is an especially large volume of data transfer to the outside. However, the session should be considered normal if most users of the network transfer 4-5 GB of data. This is what the Z-score shows.

○ The Z-score formula

$$Z = \frac{X - \mu}{\sigma}$$

The mean value is subtracted from each data value then divided by the standard deviation. Each term has the following meaning.

- Numerator: How far is the data value from the mean?
- Denominator: How many times the standard deviation is the distance of the further numerator?

○ Application of the Z-score detection principle

X in the original formula can be calculated by substituting the volume of data transfer in each session. However, it is more effective to detect the accumulated data transfer for anomalies. Thus, the Z-score is calculated with the total data transfer to the outside in a session based on the origin and destination. The accumulated value can be calculated in units of time. Both 12-hour and 24-hour bases are needed. The single session value should also be calculated. This is because it is possible to extract the large volume data transfer from a single session and understand the large volume data transfer from the accumulated value as well.

○ Z-Score analysis

According to the standard normal distribution, 68% of all data exist between the Z-score value of −1 and +1, and 95% exist between −2 and +2. 99% of all data exist between −3 and +3.

If ±2 is set as the critical domain, the sessions outside of the 95% normal can be detected. The Z-score value is set to the critical value according to the given network environment to detect the origin or destination of excessive data transfer.

■ Detection using accumulated data transfer

○ Detection principle

- This improves the problem of not being able to detect if the data transfer of a single session does not reach the detection byte volume.
- The data transfers from the origin and to the destination within the specific period are all added and compared with the critical domain bytes.

○ Application of the detection principle

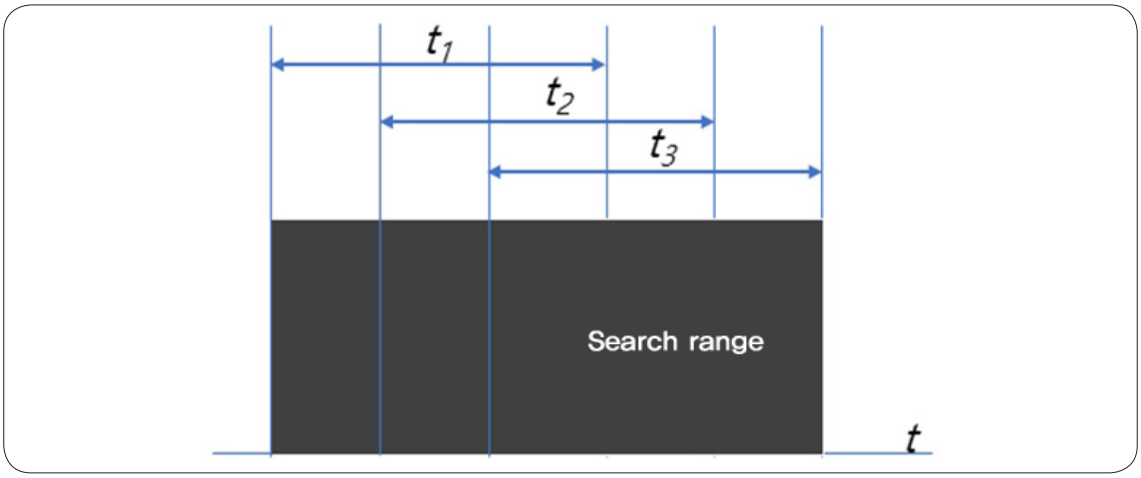
– Real-time detection: The outbound data transfer for the specified period is calculated every 10 minutes (may be user specified).

※ The data transfer occurs in continuous time. The search range should thus be set in an overlapping range according to time. As shown in [Figure 25] below, t2 and t3 should be searched after the search result of t1. The following is an example of a search for cumulative transfer volume.

– Daily: The outbound transfer by origin and destination for the past 24 hours is calculated at 00:05.

– Every 3 days: The outbound transfer by origin and destination for the past 3 days is calculated.

※ This is registered in the cron table or uses the scheduling function of the integrated log analysis system.



[Figure 25] Example of search time

■ Detection of external transfer in download/upload ratio

○ Detection principle

The Internet service organized in a client/server structure operates by means of the client making a request for the data to the server. This means that the clients receive a large volume of data. If a client sends out a large volume of data, this means that the data is sent outside. The download data and upload data ratios are compared to extract the clients that show high uploading activity and detect anomalies.

○ Application of the detection principle

1. Calculate the total uploads and downloads for each origin IP within the search range.
2. Divide the total uploads by the total downloads. The following results can be obtained from the calculation of total uploads ÷ total downloads.

2.1 The value is less than 1 if the total number of downloads is larger.

2.2 The value is greater than 1 if the total number of uploads is larger.

2.3 The value is 1 if the total number of downloads is equal to the total number of uploads, but this case is not realistic.

3. Extract the clients corresponding to Step 2.2 and analyze the destination address.

Zeek saves the logs by protocol. This helps with the analysis since the applications can be categorized according to protocol. Let us also review DNS and SMTP.

■ DNS (Domain Name System)

DNS is the domain–IP conversion service by which all systems transmit upon connecting to the Internet. It shows to which site the user connects and which DNS service the user uses. It can be an indicator of malware infection.

First, all PCs must use the in–house DNS server. The clients that use a destination IP other than the internal DNS server in traffic that uses 53/UDP as the destination port are extracted. The host that generated such a query is judged to be an anomaly. There are cases where the malware makes a direct query to the external DNS server to disable the in–house security related to DNS.

The security should also build a list of domain queries and search the country code. Most users connect to the domestic domain unless it is an overseas business. The country codes extracted by MaxMind's GeoIP can be used with the domain and IP. Records of overseas connection outside of business hours may be considered as anomalies. However, it will not be abnormal at all if the user regularly interacts with overseas personnel.

※ Available at https://www.maxmind.com/en/geoip–demo

■ SMTP(Simple Mail Transfer Protocol)

This email transfer protocol is the next most widely used protocol after HTTP and DNS. Apply the following method to detect file transfer anomalies through webmail.

The email server communicates in 25/TCP mode. Monitor outbound 25/TCP traffic. The normal emails from clients are 25/TCP traffic destined for an internal email server. This means that the destination IP is the internal email server and the destination port is 25/TCP traffic. It can be judged as an anomaly if an internal host has an Internet IP as the destination and the destination port as 25/TCP.

4) Conclusion

This document shows how to detect suspicious activities within internal network. There are diverse analysis techniques to judge intrusions into internal networks. The applicable techniques differ widely according to internal services and operating architecture of the enterprise.

The behavior-based analysis of outbound traffic is more effective than event-based response to attacks coming in from the Internet. This is because the attacker needs an outbound connection since they must take over key internal resources even after succeeding in penetrating the internal network.

However, detecting an attack on an internal network is not an easy issue. Attackers continue to evolve and hide their tracks within the normal pattern of internal users. The security personnel must thus continuously collect and analyze the data for detection. Monitoring of an internal network must take a different approach from existing security operations. It must implement different privileges for each user and authenticate internal network resources in advance while building a directory.

There is no silver bullet for the security of internal networks. The security personnel are in a battle with attackers in an area where there is no correct answer. This is why clear information is greatly helpful. Using the C-TAS (Cyber Threat Analysis & Sharing) service provided by KISA in security defense can help security personnel in this regard. Knowing the IOC (Indicator of Compromise) gives the defender a dominant position over the attacker. Building a trusted channel and sharing information between enterprise security teams will serve as a solid defense against attackers. They all contemplate the same problem but do not require the same analysis and response. Ten teams analyzing, responding, and sharing ten attacks can be much more effective than ten teams analyzing one attack.

KISA Cyber Security Issue Report : Q1 2019

Printed in April 2019
Published in April 2019

Publisher | KOREA INTERNET & SECURITY AGENCY

IT Venture Tower, 135 Jungdae-ro, Songpa-gu,
Seoul, Korea, 05717

TEL : 82-2-405-5597

The contents of this report may be different from the official opinion of KISA. Reproduction or copying of this report without permission of KISA is prohibited and may be against the copyright law in case of violation.

KISA Cyber Security Issue Report : Q1 2019