Detection of SSH Dictionary Attack in DNS Reverse Resolution Traffic

Yasuo Musashi,† Masaya Kumagai,†† Shinichiro Kubota,† and Kenichi Sugitani†

We developed and evaluated Euclidian distance based detection method for SSH dictionary attacks in the total PTR resource record (RR) based DNS query request packet traffic from the campus network to the DNS cache server in a university through January 1st to December 31st, 2009. The obtained results are: (1) The network servers, especially, they have a function of SSH services, generated the significant PTR RR based DNS query request packet traffic through 07:30-08:30 in March 14th, 2009. (2) We found eleven SSH dictionary attacks in the score changes for the detection method using the calculated Euclidian distance between the observed query IP address and the last one by employing a distance value of zero and the obtained signature data at March 14th, 2009. Also (3), we found twenty-seven SSH dictionary attacks in the score changes for the detection method employing daily generated signature data. Therefore, it can be concluded that the Euclidian distance based detection method can be useful for detecting the SSH dictionary attacks in the campus network.

1. Introduciton

Unfortunately, the SSH dictionary attack (the brute force attack) has been still used to spread out the bots when hijacking the specific vulnerable network servers on the Internet [1, 2]. This is because the network servers can be easily connected with the SSH clients when the attackers know their user IDs and pass phrases, or when, in other words, the account holders use easy breakable pass phrases. Therefore, it is also important to develop detection technologies as countermeasures against the SSH dictionary attack [1, 2].

Recently, several researchers reported prevention technologies for the SSH dictionary attack by employing the distributed and cooperative active response architectures [3, 4]. Currently, we can find the SSH dictionary attack related alert messages from the IDS/IPS or logging agents (sensors) in the network servers, in which these systems, however, we have to observe directly the inbound SSH communication related packets, and they need a cost of installation, update of their security appliances or network configurations.

Previously, on the other hand, we reported that the DNS traffic and entropy based detection technologies of the inbound and outbound SSH dictionary attacks in the campus network [5-9].

† Center for Multimedia and Information Technologies (CMIT), Kumamoto University
†† Graduate School of Science and Technology, Kumamoto University

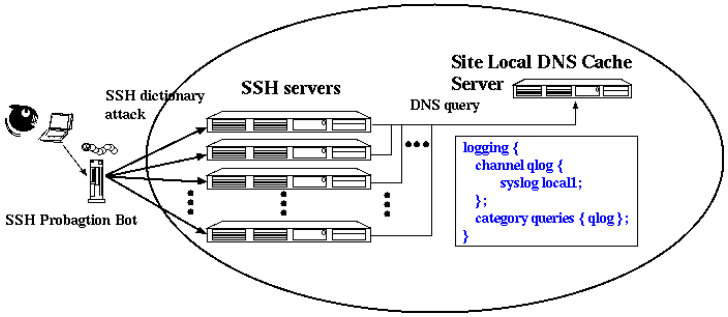

Figure 1 A schematic diagram of a network observed in the present study.

The DNS based detection system has a merit which observes only the DNS query request packet traffic between the DNS server and its clients *i.e.* the DNS resolver has been already installed in almost all the network appliances like PC terminals, routers, switches, servers, network security appliances, etc. It is, however, not only difficult to calculate the thresholds but also in a high-cost for the DNS traffic or DNS traffic entropy based detection technologies [5-9].

In this paper, (1) we carried out statistical analysis on the PTR-resource record (RR) based DNS query packet traffic from the campus network servers that were under inbound SSH dictionary attack through March 14th, 2009, and (2) we assessed the suggested detection technology by calculating the detection rate of the SSH dictionary attack, in the DNS query request packet traffic from the campus network through January 1st to December 31st, 2009.

2. Observation

2.1 Network Systems and DNS Query Packet Capturing

We investigated on the DNS query request packet traffic between the top domain (tDNS) DNS server and the DNS clients. Figure 1 shows an observed network system in the present study, which consists of the tDNS server and the PC clients as bots like a Kaminsky attack bot or a spam bot in the campus or enterprise network, and the victim hosts like the DNS servers on the campus network. The tDNS server is one of the top level domain name (kumamoto-u) system servers and plays an important role of domain name resolution including DNS cache function, and subdomain name delegation services for many PC clients and the subdomain network servers, respectively, and the operating system is Linux OS (CentOS 5.5) in which the kernel-2.6.18 is currently employed with the Intel Xeon X5660 2.8 GHz 6 Cores Dual node system, the 16GB core memory, and Intel Corporation 82575EB Gigabit Ethernet Controller.

In the tDNS server, the BIND-9.3.6-P1 program package has been employed as a DNS

©2011 Information Processing Society of Japan

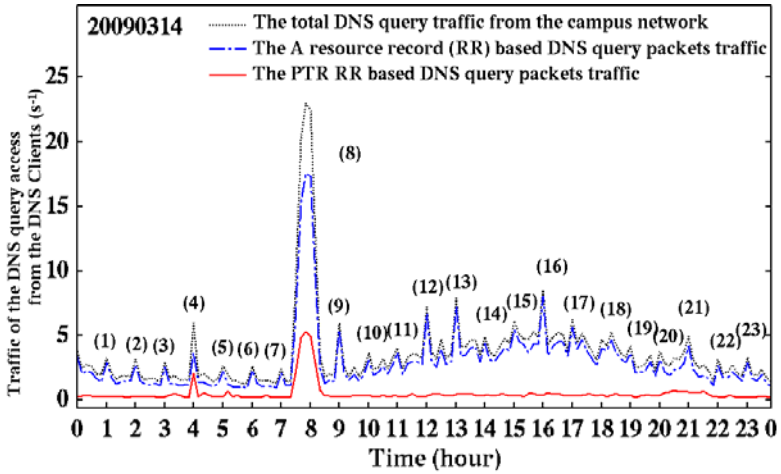

Figure 2 The total, A and PTR resource records (RRs) based DNS query request packet traffics between the top domain DNS (tDNS) server and the DNS clients on the campus network at March 14th, 2009 (s^{-1} unit).

server daemon [10]. The DNS query request packet and their query keywords have been captured and decoded by a query logging option (see Figure 1 and the named.conf manual of the BIND program in more detail). The log message of DNS query request packet access has been recorded in the syslog files. All of the syslog files are daily updated by the cron system. The line of syslog message consists of the contents of the DNS query request packet like a time, a source IP address of the DNS client, a fully qualified domain name (A-resource record (RR)) type, an IP address (PTR RR) type, or a mail exchange (MX RR) type.

2.2 Observed DNS Query Request Packet Traffic

Firstly, we can demonstrate the observed total DNS query packet traffic, and A- and PTR-resource records (RRs) based DNS query request packet traffics from the campus network to the top domain name system (tDNS) server in March 14th, 2009, as shown in Figure 2.

In Figure 2, we can find twenty three peaks and they are categorized into two groups, as: {(1)-(3), (5)-(7), (9)-(23)} and {(4), (8)}. In the former group, the total DNS query packet traffic correlates only with the A RR based DNS query request packet traffic, while in the latter one, the total DNS query packet traffic does with the both A- and PTR-RRs based DNS query request packet traffics, simultaneously. These results indicate that we should concentrate the source IP addresses of the DNS clients at the peak (8).

In the peak (8), 07:30-08:30 March 14th, 2009, the almost observed source IP addresses in the DNS query request packets are assigned to the SSH network servers in the campus network. Fortunately, we found several SSH login-failed messages in the syslog files

```
Mar 14 07:40:56 kun named[32126]: client 133.95.*.122#41612: query: **.15.9.*4 IN PTR
Mar 14 07:40:56 kun named[32126]: client 133.95.*.180#32860: query: **.15.9.*4 IN PTR
Mar 14 07:40:57 kun named[32126]: client 133.95.*.145#49339: query: **.15.9.*4 IN PTR
Mar 14 07:40:57 kun named[32126]: client 133.95.**.29#32947: query: **.15.9.*4 IN PTR
Mar 14 07:40:57 kun named[32126]: client 133.95.**.30#34540: query: **.15.9.*4 IN PTR
Mar 14 07:40:57 kun named[32126]: client 133.95.*.115#33050: query: **.15.9.*4 IN PTR
Mar 14 07:40:57 kun named[32126]: client 133.95.*.137#32827: query: **.15.9.*4 IN PTR
Mar 14 07:40:57 kun named[32126]: client 133.95.*.143#32783: query: **.15.9.*4 IN PTR
Mar 14 07:40:58 kun named[32126]: client 133.95.*.101#32799: query: **.15.9.*4 IN PTR
Mar 14 07:40:58 kun named[32126]: client 133.95.**.27#32876: query: **.15.9.*4 IN PTR
Mar 14 07:40:58 kun named[32126]: client 133.95.*.107#37557: query: **.15.9.*4 IN PTR
Mar 14 07:40:58 kun named[32126]: client 133.95.*.121#47403: query: **.15.9.*4 IN PTR
Mar 14 07:40:58 kun named[32126]: client 133.95.*.249#63358: query: **.15.9.*4 IN PTR
Mar 14 07:40:59 kun named[32126]: client 133.95.*.118#43359: query: **.15.9.*4 IN PTR
Mar 14 07:40:59 kun named[32126]: client 133.95.*.109#32779: query: **.15.9.*4 IN PTR
```

Figure 3 Changes in the IP address as the DNS query keywords in the total PTR resource record (RR) based DNS query request packet traffic from the campus network to top domain DNS (tDNS) server at March 14th, 2009.

(*/var/log/secure*) in the SSH network servers through 07:30-08:30 March 14th, 2009. This feature shows that the PTR RR based DNS query request packet traffic at the peak (8) can be assigned to the inbound SSH dictionary attack based DNS query request packet traffic. This is because the PTR RR based DNS query request packet traffic can be generated by the SSH server daemon program to check out their SSH clients and to log their IP addresses or fully qualified domain names (FQDNs) into the syslog files.

In the peak (8), we also investigated the DNS query keywords in the PTR RR based DNS query request packet traffic and the results are shown in Figure 3. In Figure 3, we can view the scenery that the IP addresses as DNS query keyword are consecutively unchanged. Therefore, it has a possibility that this consecutive unchanged IP addresses can be useful to detect the SSH dictionary related PTR RR based DNS query request packet traffic.

2.3 SSH Dictionary Attack Model

We define here an SSH dictionary attack model (See Figure 1).

― *An SSH dictionary attack model* ― the SSH dictionary attack can be mainly carried out by a small number of IP hosts on the Internet or in the campus network like bot compromised PCs or hijacked network servers. Since these IP hosts send a lot of the SSH session trials to the SSH servers in the campus network, the SSH servers record the source IP address for the SSH clients IP address in the system log files and carry out the reverse name resolution on the SSH clients, simultaneously. Then, the reverse name resolution generates a lot of the PTR RR based DNS query request packets to the DNS cache servers, in other words, the DNS cache server received a lot of the PTR RR based DNS query packets including the IP addresses of the SSH clients (SSH dictionary attackers) as DNS query keywords.

From these results, we need to take into consideration on the consecutive query keyword based model in order to develop an SSH dictionary attack detection system *i.e.* we also

ⓒ2011 Information Processing Society of Japan

```
1 #!/bin/tcsh -f
2 set TH=10
3 # Step 1 Preprocessing
4 cat /var/log/querylog | grep "IN PTR" | arpa | \
5 clgrep -cclients | grep -v -f noise | \
6 # Step 2 Detection
7 qdis 0.0 0.0 | \
8 # Step 3 Calculation of Query Frequency
9 awk '{print $9}' | sort -r | uniq -c | sort -r | \
10 awk '{printf("%s\t%s\n",$2,$1);}' | \
11 qdos $TH >query.freq
12 # Step 4 Scoring
13 cat /var/log/querylog | clgrep -cSSHDA.conf | \
14 grep "IN PTR" | arpa | cngrep -Dquery.freq | \
15 wc -l
16 exit 0
```

```
133.95.**.**
133.95.**.**
133.95.**.**
133.95.**.**
b.*dns***.udp
lb.*dns***.udp
db.*dns***.udp
r.*dns***.udp
dr.*dns***.*dp
1.0.0.127.dnsbugtest.127.0.0.1
1.0.0.127.dnsbugtest.1.0.0.127
```

Figure 4 Fixed Signature based Algorithm and Script Code, and Noises in the PTR resource record (RR) based DNS query request packet traffic from the campus network.

suggest hereafter the Euclidian distance based detection model.

2.4 Estimation of Euclidean Distances of IP addresses of IP addresses as DNS Query Keywords

The Euclidean distances, d (IP_i, IP_{i-1}), are calculated, as

$$d(IP_i, IP_{i-1}) = \sqrt{\sum_{j=1}^{4} (x_{i,j} - x_{i-1,j})^2} \quad (1)$$

where both IP_i and IP_{i-1} are the current IP address i and the last IP address i-1 of the DNS query keywords, respectively, and where $x_{i,1}$, $x_{i,2}$, $x_{i,3}$, and $x_{i,4}$ correspond to an IPv4 address like A.B.C.D, respectively. For instance, if an IP address is 192.168.1.1, the vector $(x_{i,1}, x_{i,2}, x_{i,3}, x_{i,4})^T$ can be represented as $(192.0, 168.0, 1.0, 1.0)^T$. The detection is decided by thresholds d_{min}=0.0 and d_{max}=0.0, ass

$$d_{min} \le d(IP_i, IP_{i-1}) \le d_{max} \quad (2)$$

2.5 Fixed Signature based Detection Algorithm for SSH Dictionary Attack

We suggest the following fix signature based detection and scoring algorithm of the SSH dictionary attack and we show a prototype program in Figure 4:

—— **Step 1** *Preprocessing*—— In this step, the first **grep** command extracts the total PTR RR based DNS query request packet messages from the DNS query log file (*/var/log/querylog*), the **arpa** command converts the reverse query format "D.C.B.A.in-addr.arpa" into the usual IPv4 format "A.B.C.D" (A, B, C, and D represent digit numbers of {0-255}), the **clgrep** command extracts only the DNS query traffic from the campus network, and the second **grep** command discards the noises shown in Figure 4.

```
1 #!/bin/tcsh -f
2 set TH=10
3 # Step 1 Preprocessing
4 cat /var/log/querylog | grep "IN PTR" | arpa | \
5 clgrep -cclients.conf | grep -v -f noise | \
6 # Step 2 Detection
7 qdis 0.0 0.0 | \
8 # Step 3 Calculation of Query Frequency
9 awk '{print $9}' | sort -r | uniq -c | sort -r | \
10 awk '{printf("%s\t%s\n",$2,$1);}' | \
11 qdos $TH >query.freq
12 # Step 4 Calculation of source IP Frequency
13 cat /var/log/querylog | grep "IN PTR" | arpa | \
14 clgrep -cclients.conf | \
15 cngrep -Dquery.freq | tr '#' ' ' | \
16 awk '{print $7}' | sort -r | uniq -c | sort -r | \
17 awk '{printf("%s\t%s\n",$2,$1);}' | \
18 $QDOS $TH >sourceIP.freq
19 # Step 5 Updating SSHDA.conf
20 cat SSHDA.conf sourceIP.freq | \
21 awk '{print $1}' | sort -r | uniq -c | sort -r | \
22 awk '{printf("%s\t%s\n",$2,$1);}' | \
23 grep -v -f noise >SSHDA.conf
24 exit 0
```

Figure 5 Updater Script Code for *SSHDA.conf* file.

—— **Step 2** *Detection* —— In the second step, the **qdis** command prints out a syslog message if it is calculated to be zero in the Euclidean distance, d(IP_i, IP_{i-1}), between the two IP addresses IP_i, IP_{i-1}, as DNS query keywords..

—— **Step 3** Calculation of Query Frequency ——In the third step, the first **awk** command extracts the source IP address of the SSH dictionary attacker from the query keyword, the two **sort**, **uniq**, and the last **awk** commands calculate query keyword based frequency for the PTR RR based DNS query request packet traffic, the **qdos** command can extract the IP address when the frequency takes more than 10 (set TH=10, as a threshold), and the results of this step are written in the file *query.freq*.

—— **Step 4** *Scoring* ——In the final step, the **clgrep** command extracts the source IP addresses of the frequently attacked SSH servers in the campus network by employing the signature data (*SSHDA.conf*), the **cngrep** command extracts only IP address based DNS queries from the PTR RR based DNS query request packet traffic by referring to the file *query.freq*, and the **wc** command calculates the score for the detection.

ⓒ2011 Information Processing Society of Japan

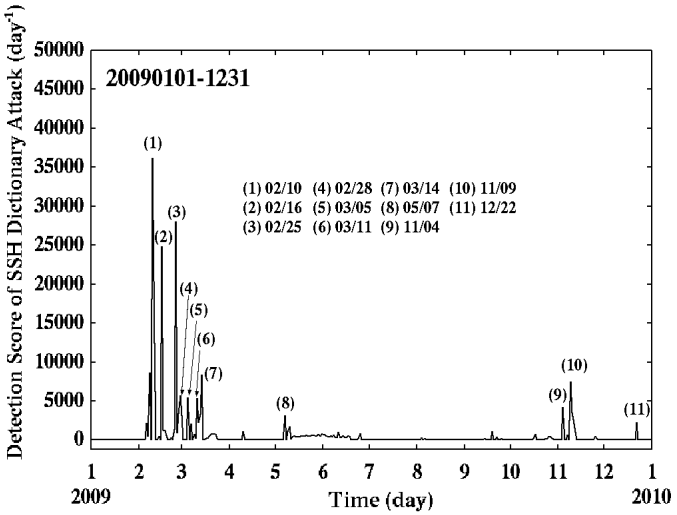

Figure 6 Changes in the signature data file based detection score of SSH dictionary attack in the PTR resource record (RR) based DNS query request packet traffic from the campus network to the top domain DNS (tDNS) server through January 1st to December 31st, 2009 (day^{-1} unit).

2.6 Creating Signature Data File

In order to obtain the IP addresses for creating the file *SSHDA.conf*, we investigated the source IP addresses in the PTR resource recode (RR) based DNS query request packet traffic including the IP addresses that were SSH dictionary attackers at March 14th, 2009, employing the script code (See Figure 5), as follows:

── **Step 1** *Preprocessing*── In this step, all the commands are the same as those in Figure 4.
── **Step 2** *Detection*──In the second step, the **qdis** command is the same as that in Figure 4.
── **Step 3** *Calculation of Query Frequency*──In the third step, all the commands are the same as those in Figure 4.
── **Step 4** *Calculation of Source IP Frequency*──In the step, the **cngrep** command extracts only the PTR RR DNS query request packet traffic including the query IP addresses in the file *query.freq*, the two **sort** and **uniq** commands convert the extracted source IP addresses into the unique source IP addresses, and the last **awk** command writes these IP addresses and their frequencies into the file *sourceIP.freq*.
── **Step 5** *Updating of SSHDA.conf*──In the step, the first **awk** command extracts only IP address included in the files *SSHDA.conf* and *sourceIP.freq*, the two **sort** and **uniq** commands

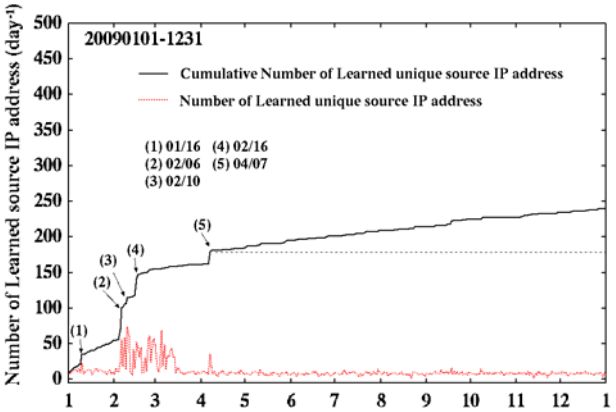

Figure 7 Changes in the cumulative and daily updated numbers of learned unique source IP addresses for the SSH servers in the campus network (day^{-1} unit).

convert the extracted IP addresses into the unique IP addresses, and the last **awk** command writes these IP addresses into the file *SSHDA.conf*.

We carried out the script code and obtained 88 unique IP addresses of the SSH servers in the campus network at March 14th, 2009. Hereafter, we use these 88 unique IP addresses for detection of the SSH dictionary attack.

3. Results and Discussion

3.1 Evaluation of Signature based Detection Technology

We demonstrate the calculated signature based detection score (rate) for the SSH dictionary attack by observing PTR resource record (RR) based DNS query request packet traffic generated by the SSH servers in the campus network through January 1st to December 31st, 2009, as shown in Figure 6.

In this detection technology, we employed misuse intrusion detection (MID) model [11] with the signature file *SSHDA.conf* consisting of the IP addresses which are allocated to the SSH servers in the campus network, in which the SSH servers can have a high probability of the SSH dictionary attack from the Internet *i.e.* we can expect a high detection rate as well as in a lower false positive manner.

As shown in Figure 6, we can find eleven significant peaks ($\leq$ 2,000 day^{-1}). These peaks are assigned to: (1) February 10th, (2) 16th, (3) 25th, and (4) 28th, (5) March 5th, (6) 11th, and (7) 14th, (8) May 7th, (9) November 4th and (10) 9th, and December 22nd, 2009. Interestingly, we can find several peaks through February, March, May, November, and December, 2009.

©2011 Information Processing Society of Japan

Expectedly, it is clear that the signature based detection technology gives us a good precise detection rate in a low false positive manner. However, the signature based detection usually needs to upgrade its signature data, frequently. Also, we have to take the false negative into consideration in the SSH dictionary attack detection technology.

3.2 Upgrading of Signature Data File

We illustrate the totally registered and daily updated numbers of the learned unique source IP addresses for the PTR resource record (RR) based DNS query request packet traffic generated by the SSH servers in the campus network through January 1st to December 31st, 2009, in Figure 7 (updating algorithm signature data file *SSHDA.conf* and its script code shown in Figure 5)

In Figure 7, the totally registered number of unique source IP addresses increases in a large scale at the points (1) January 16th, (2) February 6th, (3) 10th, (4) 16th, and (5) April 7th. This feature shows that the SSH dictionary attackers changes their targets at those points (1)-(5). Also, we can observe the gradual increases after the change point (5) until December 31st, 2009.

From these results, we can conclude that the attacking targets should be always updated i.e. it is required to freshen up the signature data file. Also, the updated the signature data file can give us a high detection score for the SSH dictionary attack, however, the false positive probably still increases when employing the freshened up the signature data file. Furthermore, we also pay much attention on the false negative in the detection score, since the fixed signature data file can also bring about missing the attack. This is also because it has still a possibility for the targeted SSH dictionary attack toward the specific SSH servers in the campus network. Therefore, we combine partially the script codes shown in Figures 4 and 5, and we can show the newly developed dynamic signature based algorithm and the script code in Figure 8.

—**Step 1** *Preprocessing*— In this step, all the commands are the same as those in Figure 5.
—**Step 2** *Detection*—In the second step, the **qdis** command is the same as that in Figure 5.
—**Step 3** *Calculation of Query Frequency*—In the third step, all the commands are the same as those in Figure 5.
—**Step 4** *Calculation of Source IP Frequency*—In the third step, all the commands are the same as those in Figure 5.
—**Step 5** *Scoring*—In the last step, almost the commands are the same as those in Figure 5. Exceptionally, the **clgrep** command extracts only the SSH dictionary based DNS query request packet traffic by the dynamically generated signature data file *sourceIP.freq* and **wc** command calculates the detection score of the SSH dictionary attack.

3.3 Evaluation of Dynamic Signature based Detection Technology

We display the dynamically calculated signature based detection score (rate) for the SSH dictionary attack by observing PTR resource record (RR) based DNS query request packet traffic which was transmitted by the SSH servers in the campus network through January 1st to December 31st, 2009, as shown in Figure 9.

In Figure 9, we can observe twenty seven significant peaks (≤10,000 day-1). These peaks are assigned to: (1) January 16th, (2) 23rd, (3) 25th, (4) February 7th, (5) 10th, (6) 16th, (7)

```
#!/bin/tcsh -f
set TH=10
# Step 1 Preprocessing
cat /var/log/querylog | grep "IN PTR" | arpa | \
clgrep -cclients.conf | grep -v -f noise | \
# Step 2 Detection
qdis 0.0 0.0 | \
# Step 3 Calculation of Query Frequency
awk '{print $9}' | sort -r | uniq -c | sort -r | \
awk '{printf("%s\t%s\n",$2,$1);}' | \
qdos $TH >query.freq
# Step 4 Calculation of source IP Frequency
cat /var/log/querylog | grep "IN PTR" | arpa | \
clgrep -cclients.conf | \
cngrep -Dquery.freq | tr '#' ' ' | \
awk '{print $7}' | sort -r | uniq -c | sort -r | \
awk '{printf("%s\t%s\n",$2,$1);}' | \
$QDOS $TH >sourceIP.freq
# Step 5 Scoring
cat /var/log/querylog | grep "IN PTR" | arpa | \
clgrep -csourceIP.freq | grep -v -f noise | \
wc -l
exit 0
```

Figure 8 Dynamic Signature based Algorithm and Script Code..

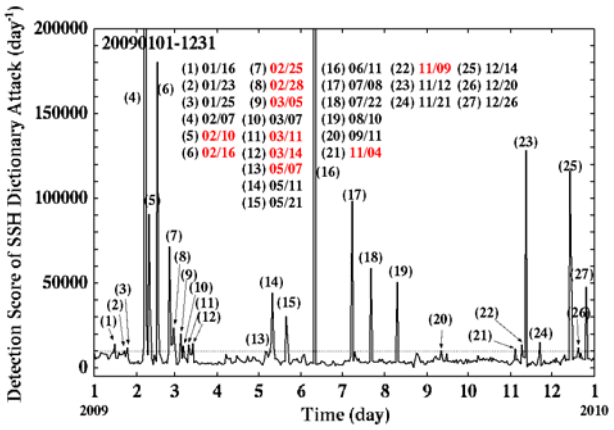

Figure 9 Changes in the signature data file based detection score of SSH dictionary attack in the PTR resource record (RR) based DNS query request packet traffic from the campus network to the top domain DNS (tDNS) server through January 1st to December 31st, 2009 (day^{-1} unit).

©2011 Information Processing Society of Japan

25th, (8) 28th, (9) March 5th, (10) 7th, (11) 11th, (12) 14th, (13) May 7th, (14) 11th, (15) 21st, (16) June 11th, (17) July 8th, (18) 22nd, (19) August 10th, (20) September 11th, (21) November 4th, (22) 9th, (23) 12th, (24) 21st, (25) December 14th, (26) 20th, and (27) 26th, 2009.

Interestingly, we can find no peaks in Figure 9, corresponding to the peaks (1)-(4), (10), (14)-(20), (23)-(27). This feature indicates that we can observe the new peaks by employing the dynamic signature based detection technology *i.e.* we can suppress the false negative.

4. Conclusions

We developed usual signature based detection techonology and evaluated by use of signature file *SSHDA.conf* in which the IP addresses corresponding to the SSH servers that there are possibilities to have an SSH dictionary attack. We can observe eleven significant peaks in a lower false positive manner. However, the rigid signature based detection technology should always require updating or refreshing their pattern data because of false postive.

To confirm this, we investigated the changes in the cumulative and daily updated numbers of learned unique source IP addresses for the SSH servers in the campus network. We observed that the daily updated number increased only. As a result, we can conclude that the signature data file needs to be updated, very frequently.

Thus, we developed dynamic signature based detection technology by modifying the rigid signature based detection technology, and evaluated by dynamically obtained signature data file in which the IP addresses corresponding to the SSH servers that there are high possibilities to have an SSH dictionary attack. Finally, we obtained significant twenty seven peaks in detection score for the dynamic signature based detection technology. This means that we can suppress the false negative in the SSH dictionary attack detection. In addition, these peaks can be in good agreement with those in the sample variance changes in the PTR RR based DNS query request packet traffic.

Consequently, although we found that we could detect the inbound SSH dictionary attack by only observing the Euclidian distance between the current IP address and the previous IP address as DNS query keywords in the PTR RR based DNS query request traffic from the campus network, we need to continue further development of detection technology to watch the SSH dictionary attack to the campus network.

Acknowledgment All the studies were carried out in CMIT of Kumamoto University. We gratefully thank all the CMIT staffs and all the members of Kumamoto University.

References

1) Seifert, C.: Analyzing Malicious SSH Login Attempts, Symantec Security Community, Secruity Articles, 2010, http://www.symantec.com/connect/articles/analyzing-malicious-ssh-login-attempts.

2) Ramsbrock, D., Berthier, R., and Cukier, M.: Profiling Attacker Behavior Following SSH Compromises, Proceeding of the thirty-seventh Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN07), Washington D.C., USA, IEEE Computer Society, pp.119-124 (2007).

3) Oosumi, Y. and Yamai, N.: Technique of the countermeasure for brute force attack which can cooperate between the hosts, IPSJ SIG Technical Reports, Distributed System and Management 47th (DSM47), Vol. 2007, No. 93, pp.49-54 (2007).

4) Thames, J. L., Abler ,R., and Keeling, D.: A distributed active response architecture for preventing SSH dictionary attacks, Proceedings of the Southeastcon, 2008, IEEE, Huntsville, AL, USA, pp.84-89 (2008).

5) Ludeña Romaña, D. A., Sugitani, K., and Musashi, Y.: DNS Based Security Incidents Detection in Campus Network, International Journal of Intelligent Engineering and Systems, Vol. 1, No. 1, pp.17-21 (2009).

6) Ludeña Romaña, D. A., Kubota, S., Sugitani, K., and Musashi, Y.: DNS-based Spam Bots Detection in a University, International Journal of Intelligent Engineering and Systems, Vol. 2, No. 3, pp.11-18 (2009).

7) Ludeña Romaña, D. A., Musashi, Y., Takemori, K., Kubota, S., Sugitani, K., Usagawa, T. and Sueyoshi, T.: DNS Based Detection of SSH Dictionary Attack in Campus Network, Proceeding of the 5th International Conference on Information (INFORMATION 2009), Kyoto, Japan, pp.134-137 (2009).

8) Ludeña Romaña, D. A., Musashi, Y., Takemori, K., Kubota, S., Sugitani, K., Usagawa, T. and Sueyoshi, T.: DNS Based Detection of SSH Dictionary Attack in Campus Network, Information, Vol. 13, No. 3(A), pp.701-707 (2010).

9) Takemori, K., Ludeña Romaña, D. A., Kubota, S., Sugitani, K., and Musashi, Y.: Detection of NS Resource Record DNS Resolution Traffic, Host Search, and SSH Dictionary Attack Activities, International Journal of Intelligent Engineering and Systems, Vol. 2, No. 4, pp.35-42 (2009).

10) BIND-9.3.6-P1: http://www.isc.org/products/BIND/

11) Denning, D. E.: An Intrusion-detection model, IEEE Trans. Soft. Eng., Vol. SE-13, No.2, pp.222-232 (1987).

ⓒ2011 Information Processing Society of Japan