ABSTRACT

The Domain Name System (DNS) is vital to the Internet, providing a mechanism for resolving host names into Internet Protocol (IP) addresses. Insecure underlying protocols and lack of authentication and integrity checking of the information within the DNS threaten the proper functionality of the DNS. The Internet Engineering Task Force (IETF) is working on DNS security extensions to increase security within the DNS, known as DNSSEC. These security issues and solutions are presented in this paper. The threats that surround the DNS are due in part to the lack of authenticity and integrity checking of the data held within the DNS and in part to other protocols that use host names as an access control mechanism. In response to this, the IETF formed a working group to add DNS Security (DNSSEC) extensions to the existing DNS protocol. This paper gives an overview of the DNS, its security weaknesses, and the new security extensions being worked on by the IETF’s DNSSEC Working Group (WG).

Keywords: - BW, PC clients and Hardware security alarms, DDOS, Proxy server, Mail exchange

1 INTRODUCTION

It is of considerable importance to find out new countermeasure technologies for detecting boot worms (BW), since they infect with the PC clients as well as hijacks the compromised PC clients [1-4]. The BW-infected PC clients become usually components of the boot network (bots) that are very useful for transmitting a lot of unsolicited E-mails like spam, phishing, and mass mailing (a SMTP proxy; spam boot), to carry out a distributed denial of service (DDoS) attack (a base for cyber attack; a DDoS boot), to launch new upgraded internet worms that infect with the next victim PC clients (boot propagation), to spy out or disclose private information (information leakage), and so on [1]. From these points, it is quickly required to develop a new detection method for BW activity.

Conventionally, we can detect BW activity candidates by observing the clients based MX (Mail Exchange) or PTR (Pointer: reverse name resolution) resource record DNS query access when supposing that the client based MX or PTR RR based DNS resolution access is suspicious because the usual PC clients send only Address (A) RR based DNS query packets [7-10]. This spam boot detection model is currently very useful to detected the compromised PC clients infected with a classical mass mailing worm (MMW) like W32/Netsky and W32/Mydoom MMWs [12,13] as well as the BW-infected PC clients when transmitting spam mails like W32/Mytob and W32/Zotob BWs [14,15]. However, it is generally difficult to detect recent BW-infected/compromised PC clients. We have recently started to investigate on the entropy based DNS query traffic analysis method in order to confirm whether this method is useful or not.

In this paper, we discuss on (1) the DNS query traffic from the DNS clients to the top domain DNS server (tDNS) through January 1st to July 31st, 2007, (2) the source IP addresses- and query keywords-based entropy analysis on the DNS query traffic, and (3) how to detect the suspicious candidates like BW-infected PC clients in the campus network.

Figure 1: A schematic diagram of the network observed in the present study
2 OBSERVATIONS

2.1 Network System
We investigated traffic of the DNS query packets access between the top domain DNS server (tDNS) and the PC clients. Figure 1 shows an observed network system in the present study, an optional configuration of BIND-9.2.6 server program daemon in tDNS, and the three typical DNS query types. The DNS server, tDNS, is one of the top level DNS (kumamoto-u) servers and plays an important role of domain name resolution and subdomain delegation services for many PC clients and the subdomain network servers in the university, respectively, and the operating system is CentOS 4.3Final and is currently employed kernel-2.6.9 with the Intel Xeon 3.20GHz Quadruple SMP system, the 2GB core memory, and Intel 1000Mbps Ethernet Pro Network Interface Card.

2.2 Capture of DNS Query Packets
In tDNS, BIND-9.2.6 program package has been employed as a DNS server daemon [16]. The DNS query packets and their query keywords (query contents) have been captured and decoded by a query logging option (Figure 1, see % man named.conf in more detail). The log of DNS query access has been recorded in the syslog files which are daily updated/rotated by the crond system. The line of syslog messages mainly consists of a source IP address and query keywords (payloads) in the DNS query packets like a fully qualified domain name (an A resource record (RR) type: standard name resolution), an IP address (a PTR RR type: reverse name resolution), and a mail exchange (an MX RR type).

2.3 Conventional Traffic Analysis
Firstly, we can show the DNS query traffic from the DNS clients toward the top domain DNS (tDNS) server through January 1st to July 31st, 2007, in Figure 2. In Figure 2, we can observe several significant peaks of (1) January 15th, (2) February 1st and (3) 7th, (4) March 12th and (5) 22nd, and (6) April, 25th, 2007. We investigated on the security incidents and several peaks can be assigned, as follows: The peaks (1) and (5) are caused by a crash of the NIS server [18], the peaks (2)-(4) are probably including BW activities, the peak (6) is based on the DNS mis configuration, and the peak (7) is occurred with the use of insecure access point for wireless LAN. After May, however, no interesting peak can be found. From this reason, we employed hereafter the entropy based analysis on the DNS query traffic.

2.4 Estimation of Entropy
We employed Shannon’s function in order to calculate entropy (randomness) $H(X)$, as, $\Sigma = - i X H(X) P(i) log 2 P(i)$ (1) where $X$ is the data set of the frequency freq($i$) of IP addresses or that of the DNS query keywords in the DNS query packet traffic from the outside of the campus network, and the probability $P(i)$ is

![Figure 2: Total traffic of the DNS query packets to the top domain DNS server (tDNS) and the traffic from the inside- and the outside DNS clients of a university campus network through January 1st to July 31st, 2007 (day-1 unit) defined, as $P(i) = \frac{freq(i)}{\sum_j freq(j)}$ where $i$ and $j (i, j \in X)$ represent the source IP address or the DNS query keywords in the DNS query packet, and the frequency freq($i$) are estimated with the following script program:

```bash
#!/bin/tcsh
```
We investigated the DNS query traffic from the DNS clients from the inside and outside of the campus network in a university through January 1st to July 31st, 2007 employing entropy based statistical analysis method. The IETF ratified the DNS as an Internet standard to solve the issues of scalability surrounding the hosts.txt file. Since then, the widespread use of the DNS and its ability to resolve host names into IP addresses for both users and applications alike in a timely and fairly reliable manner, makes it a critical component of the Internet. The distributed management of the DNS and support for redundancy of DNS zones across multiple servers promotes its robust characteristics. However, the original DNS protocol specifications did not include security. Without security, the DNS is vulnerable to attacks stemming from cache poisoning techniques, client flooding, dynamic update vulnerabilities, information leakage, and compromise of a DNS server's authoritative files.

In order to add security to the DNS to address these threats, the IETF added security extensions to the DNS, collectively known as DNSSEC. DNSSEC provides authentication and integrity to the DNS. With the exception of information leakage, these extensions address the majority of problems that make such attacks possible. Cache poisoning and client flooding attacks are mitigated with the addition of data origin authentication for RRsSets as signatures are computed on

```
cat querylog | grep -v "client 133\.|95\." | tr ' # ' \
| awk '{print $7}' | sort -r | uniq -c |
sort -r >freq-sIPaddr
cat querylog | grep -v "client 133\.|95\." |
awk '{print $9}' | sort -r | uniq -c |
sort -r >freq-query keywords
```

**Chart 1**

Where “querylog” is a syslog file including syslog messages of the BIND-9.2.6 DNS server daemon program[6]. The syslog message (one line) consists of keywords as “Month”, “Day”, “hours:minutes:seconds”, “server name”, “named [process identifier]”, “client”, “source IP address# source port address:”, “query:”, and “DNS query keywords”. This script program consists of three program groups: (1) The first program group is a first line only including “#!/bin/tcsh -f” means that this script is a TENEX C Shell (tcsh) coded script programs. (2) The second program group estimates frequencies of the unique source IP addresses and the unique source IP addresses, consisting of of unix commands from “cat” to “sort -r” because the backslash “\” connects the line terminated by “\” with the next line in the tcsh program. In this program group, the “cat” shows all the syslog message-lines from the syslog file “querylog”, the “grep -v” (or “grep ”) command extracts only the message-lines excluding (or including) the source IP address of “133.95.x.y”, the “tr” replaces a character ‘#’ with a white space ’ ’, the unix command “awk '{print $7}' ” extracts only a seventh keyword as “source IP addresses” in the message-line, the “sort -r | uniq -c” sorts the dataset of “source IP addresses” into the dataset of “unique source IP addresses” and estimate the frequencies of the unique source IP addresses and the final results are written into the file “freq-sIPaddr”. (3) The last program group extracts the DNS query keywords from the syslog message-lines, sorts the dataset of “DNS query keywords” into the dataset of “unique DNS query keywords” and estimates the frequencies of the unique DNS query keywords.

Finally, the results of the last program group are written into the file “freq-keywords”. In the last program group, although almost the commands, arguments, and their options take the same as the second program group, the unix command “tr” and its arguments are removed and a new argument “ '{print $9}' ” replaces the arguments of the unix command “awk” in the second program group. Entropy based packet traffic analysis was suggested by Wagner and Platter , recently [17].

**3 RESULTS AND DISCUSSION**

**3.1 Entropy Analysis on DNS Query Traffic**

We illustrate the calculated entropy for the frequencies of the unique source IP addresses and the DNS query keywords in the DNS traffic from the inside and the outside of the campus network to the top domain DNS server through January 1st to July 31st, 2007, we can observe several significant peaks of (i) January 14th, (ii) March 4th, (iii) 11th, (iv) 18th, and (v) 31st, (vi) May 20th, (vii) June 25th, 2007. We also investigated on the peaks and the peaks have been fortunately assigned to the security incidents, as follows: The crash of NIS server for (i) and (iii) [18], the DNS misconfiguration in the campus subdomain DNS server for (ii), (iv), and (v) and the hijacked online fraud web server in the local subdomain for (vi) and (vii), respectively. Interestingly, we can also notice that no peak can be found like ones.

**4 CONCLUSIONS**

We investigated on the DNS query traffic from the DNS clients from the inside and outside of the campus network in a university through January 1st to July 31st, 2007 employing entropy based statistical analysis method. The IETF ratified the DNS as an Internet standard to solve the issues of scalability surrounding the hosts.txt file. Since then, the widespread use of the DNS and its ability to resolve host names into IP addresses for both users and applications alike in a timely and fairly reliable manner, makes it a critical component of the Internet. The distributed management of the DNS and support for redundancy of DNS zones across multiple servers promotes its robust characteristics. However, the original DNS protocol specifications did not include security. Without security, the DNS is vulnerable to attacks stemming from cache poisoning techniques, client flooding, dynamic update vulnerabilities, information leakage, and compromise of a DNS server’s authoritative files.

In order to add security to the DNS to address these threats, the IETF added security extensions to the DNS, collectively known as DNSSEC. DNSSEC provides authentication and integrity to the DNS. With the exception of information leakage, these extensions address the majority of problems that make such attacks possible. Cache poisoning and client flooding attacks are mitigated with the addition of data origin authentication for RRsSets as signatures are computed on
the RRSets to provide proof of authenticity. Dynamic update vulnerabilities are mitigated with the addition of transaction and request authentication, providing the necessary assurance to DNS servers that the update is authentic. Even the threat from compromise of the DNS server’s authoritative files is almost eliminated as the SIG RR are created using a zone’s private key that is kept off-line as to assure key’s integrity which in turn protects the zone file from tampering. Keeping a copy of the zone’s master file off-line when the SIGs are generated takes that assurance one step further. DNSSEC cannot provide protection against threats from information leakage. This is more of an issue of controlling access, which is beyond the scope of coverage for DNSSEC. Adequate protection against information leakage is already provided through such things as split DNS configuration.

The following interesting results are obtained, as: (1) There are two types of changes in the source IP addresses and query keyword-based entropies. One is the simultaneous change in the both entropies, and the other is symmetrical change in both entropies. (2) Conventionally, the latter type change can be observed. Recently, however, the former type change can be observed in 2007. Therefore, it can be concluded that the recent bots worm (BW) infected PC clients as spam bots does not directly send a lot of spam E-mails to the victim PC, however, they transmit via the local vulnerable E-mail servers. We continue to develop detection technology based on the results of the present paper and to evaluate of the detection rate.

REFERENCES


AUTHOR

Umesh Sehgal received the PhD, M.Phil and MCA degrees in computer science and application from Maharishi Dayan and University and Vinankaya Mission University in 2012, 2004 and 2008, respectively. During 2006-2008 worked as a junior lecturer in BITS, Pilani Dubai and also worked as an Assistant Professor in LPU, Phagwara. Presently worked as an Associate Professor in Computer science department in ARNI University.

Naresh Kumar received the B.Tech Degree in CSE from PTU Campus and presently he is doing M.Tech CSE from ARNI University.