Indirect Detection of Mass Mailing Worm-Infected PC terminals for Learners

Yasuo Musashi,† Ryuichi Matsuba,† and Kenichi Sugitani†

†*Center for Multimedia and Information Technologies, Kumamoto University,*
2-39-1 Kurokami, Kumamoto-City, 860-8555, JAPAN
†*{musashi, matsuba, sugitani}@cc.kumamoto-u.ac.jp*

Abstract

We have developed a new indirect virus detection system that detects IP addresses of the mass mailing worm (MMW)-infected PCterminals for learners by only watching the domain name system (DNS) query traffic between the DNS server and the PC terminals.

1. Introduction

Recent internet worms (IW) are mainly categorized into two types, as follows: one is a mass mailing worm (MMW) like W32/Sobig.F MMW[1], W32/Mydoom.A MMW[2], and W32/Netsky.Q MMW[3] which transfers itself by an attachment file of the E-mail and the other is a system-vulnerability-attack worm (SVAW) like W32/Welchia SVAW[4] and W32/Sasser.D SVAW[5] that transfers itself by attack on vulnerabilities of remote buffer overflow in the operating systems and/or the application software. Especially, since March 29th, 2004, the former MMW like W32/Netsky.Q MMW has attacked a lot of Windows PC terminals worldwidely because of quick development of W32/Netsky MMW variants[3] and the delay of delivering a virus pattern for worm detection in PC terminals. The speed of the MMW development is too much fast to fix it so that we have detected a total of 104,010 worm actions of the W32/Netsky.Q MMW-infected PC terminals in our university from 12:50 to 17:44 at March 29th, 2004. Furthermore, our university has 920 PC terminals for learners that have a large scaled potential for MMW breeding; truly, we have detected 800 IP addresses of W32/Welchia SVAW at August 20th, 2003, which include a wide IP address range of 920 PC terminals for learners *i.e.* the PC terminals are considerably to be a big threat. From these points, it is of considerable importance to detect an IP address of the MMW-infected PC terminal.

One of the attractive solutions to detect of an IP address of the MMW-infected PC terminal is to employ an intrusion detection system (IDS)[6-14]. We know that Snort[14] is an open-sourced package and a rule-based network based IDS, which has a lot of function such as packet capture, IP defragmentation, TCP stream reassembling (stateless/stateful), and content matcher (detection engine). However, it gives a plenty of alert messages when detecting suspicious packets and/or illegal accesses and it needs a highly qualified skill to tune it up. Therefore, we designed and developed a new indirect virus detection system (MXRPDS) that detects IP addresses of the mass mailing worm (MMW)-infected PC terminals for learners by only watching the domain name system (DNS) query traffic between the DNS server and the PC terminals.

We have reported that DNS query access traffic from an SMTP (simple mail transfer protocol) engine of an E-mail server and SMTP traffic in the E-mail server strongly correlate to each other (D_q=$m_{SMTP}N_{SMTP}$; where D_q and N_{SMTP} are the numbers of the DNS query packets and the SMTP accesses, respectively, and $m_{SMTP}\geq 2$) [15-16]. Also, we have found that the DNS query access packets from the SMTP engine includes an MX (mail exchange) record packet[16]. Furthermore, it is found that recent MMWs use their own SMTP engines to propagate themselves.

The present paper is to discuss on (1) illegal DNS query MX record packet accesses from the MMW-infected PC terminals in which PC terminals are infected with a mass mailing worm (MMW) and are hijacked UNIX-like PC terminals with a spam relay (SR-embedded), and (2) how to prevent infection of the MMW. By analyzing syslog messages of the DNS server, we show how to detect IP addresses of the MMW-infected PC terminals and the SR-embedded PC terminals.

2. Observations

2.1 Network systems

We investigated traffic of DNS query accesses between the top domain DNS server (**tDNS**) and DNS clients of A (**cA**), B (**cB**), and C (**cC**), where both **cA** and **cB** are W32/Sobig.F and W32/Mydoom.A MMW-infected PC terminals, respectively, and **cC** is a Compaq Alpha True64 PC terminal in the laboratory of our university. Figure 1 shows a schematic diagram of a network observed in the present study. **tDNS** is one of

the top level secondary domain name (kumamoto-u) server and plays an important role of subdomain delegation and

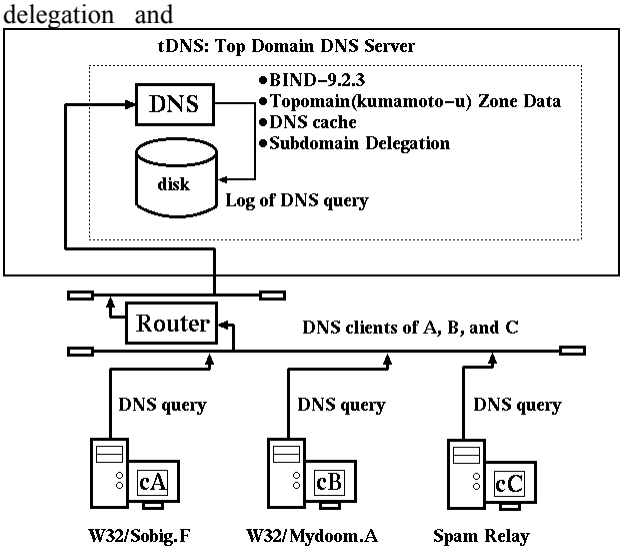

Figure 1. A schematic diagram of a network observed in the present study.

domain name resolution services for many PC terminals. In **tDNS**, the operating system (OS) is employed Linux OS (kernel-2.4.24), and an Intel Xeon 2.40GHz Dual CPU machine. The PC terminals, **cA**, **cB**, and **cC** are DNS clients of tDNS in which the first DNS server is configured to access to **tDNS**.

In **tDNS**, BIND-9.2.3 program package has been employed as DNS server daemon[17]. The DNS query packets and their contents have been recorded by the query logging option (see man named.conf), as follows:

```
logging {
        channel qlog {
            syslog local1;
        };
        category queries {qlog;};
}
```

The log of DNS query access has been recorded in the syslog file[18]. All of the syslog files are daily updated by the crond system. It is known that a DNS server provides mainly a host domain name (A record), an IP address (PTR record), and mail exchange (MX record) to DNS clients.

2.1 A method of analysis

We extract lines described DNS query accesses only including MX records from the syslog file in **tDNS**. After discarding IP addresses of DNS query accesses from the outside of university and the E-mail servers that are authorized in our university, we sorts the lines to get top

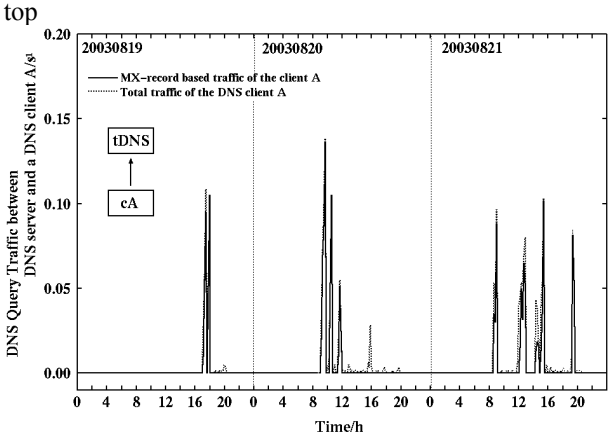

Figure 2. Traffic of the DNS query access between the top domain DNS server and the DNS client A through August 19th to 21st, 2003. The dotted line shows the total the total DNS traffic and the solid line indicates the MX-record based DNS traffic (s^{-1} unit).

IP addresses of DNS query accesses by using " sort –r" and "uniq –c" commands, as two and one times, respectively, and to show a frequency of the DNS access. If the frequency takes over 50 times, we investigate DNS query contents to get how many MX, A, and PTR records. These procedures are properly worked out to an "mmwip" program compiled with gcc-2.95.3 C compiler.

3. Results and Discussion

3.1 Mass mailing worm

We observed DNS query access traffic from a DNS client A (**cA**) to the top domain name server (**tDNS**) for August 19th-21st, 2003. We show the observed DNS query access traffic in Figure 2. The abscissa is times in units of hour and the ordinate is access count rates from **cA** to **tDNS**. Since **cA** is an Windows/XP system as used only PC terminal *i.e.* **cA** is not a server, **cA** generates only very small DNS query traffic and the traffic includes only A record packet in usual (see before 17:00 at August 19th, 2003 the dotted line in Figure 2). The **cA** DNS query traffic changes in a large scale manner after 17:00 at August 19th, 2003, and the traffic is continued to 20:30 at August 19th, 2003. The large change in traffic was taken place with an infection of mass mailing worm (MMW) in **cA**. How do we recognize the change as the infection of MMW ?

Table 1 gives the total number of lines described MX, A, and PTR records on **cA** for the observed days.

Interestingly, the total traffic consists of MX and A records. No PTR record can be found in the syslog messages for **cA**. Also, the MX record based traffic curve emerges as the solid line in Figure 2. These features

Table 1. The total number of lines for MX, A, and PTR records per a day in the syslog file in tDNS, relating to the DNS client access from cA.

day	MX	A	PTR
Aug. 19th	190	36	0
Aug. 20th	335	89	0
Aug. 21th	422	201	0

provide important information that **cA** has an SMTP engine. We confirm that the DNS query traffic is dominated by MX record and that the query drastically increases when **cA** is turned on in the latter two days.

When we see the syslog file for DNS query packets from **cB**, we encounter head lines in which, "A.ROOT-SERVERS.NET", "A.ROOT-SERVERS.NET", "B.ROOT-SERVERS.NET", "B.ROOT-SERVERS.NET",..., etc are written. These head lines are included in the virus database as the W32/Sobig.F mass mailing worm and its infection is detected in public at the August 19th, 2003[1]. Therefore, we can clearly detect that **cA** is surely infected with the W32/Sobig.F. It is noted that the head lines includes two same lines. This is because **cA** is also infected with the W32/Sobig.C.

We illustrate the DNS query traffic between **tDNS** and the DNS client B **cB** in Figure 3 through January 28th-30th, 2004. The DNS traffic includes only MX and A records. No PTR record is written in the syslog messages for **cB**. This feature is observed in the case of W32/Sobig.F MMW.

When we see the syslog file for DNS query packets from cA, we encounter head lines in which, "mx.xxxxx.co.jp", "mail.xxxxx.co.jp", ``smtp.xxxxx.co.jp", "mx1.xxxxx.co.jp", "mxs.xxx-xx.co.jp", "mail1.xxxxx.co.jp", "relay.xxxxx.co.jp", "ns.xxxxx.co.jp", "gate.xxxxx.co.jp",..., etc are written. These head lines are included in the virus database as the 32/Mydoom.A mass mailing worm and its infection is detected in public at January 28th, 2004[2]. Therefore, we can clearly detect that **cB** is surely infected with the W32/Mydoom.A MMW.

Interestingly, the traffic of MX record packet is totally less than that of total traffic *i.e.* that of A record packet. This result differs from the case of W32/Sobig.F (see Figure 2). It is fact that the total DNS query packets (5630) consist of 807 MX and 4823 A record packets at January 28th, 2004. This feature is interpreted in terms that the W32/Mydoom.A initially searches fully qualified domain name (FQDN) of the next victim PC terminals with the complement of host name keywords as, "mx.", "mail.", "smtp.", "mx1.", "mxs.", "mail1.", "relay.", "ns.", and "gate."[2]. Therefore, this scan generates a lot of A record packets more than MX record ones.

It is clear that the DNS query traffic of the DNS clients like Windows PC terminals includes MX and A record packets without PTR record packet when infected with the MMW like W32/Sobig.F MMW and W32/Mydoom.A MMW.

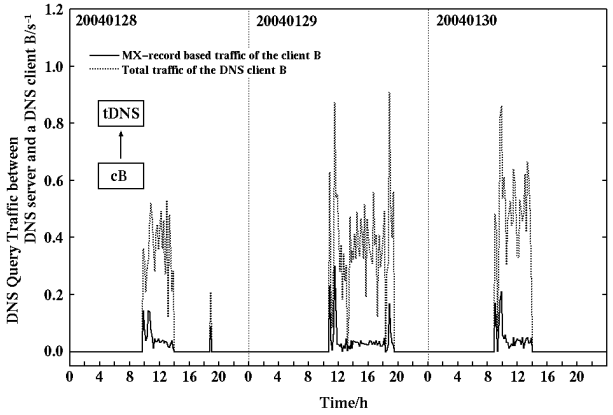

Figure 3. Traffic of the DNS query access between the top domain DNS server and the DNS client B through January 28th to 30th, 2004. The dotted line shows the total the total DNS traffic and the solid line indicates the MX-record based DNS traffic (s^{-1} unit).

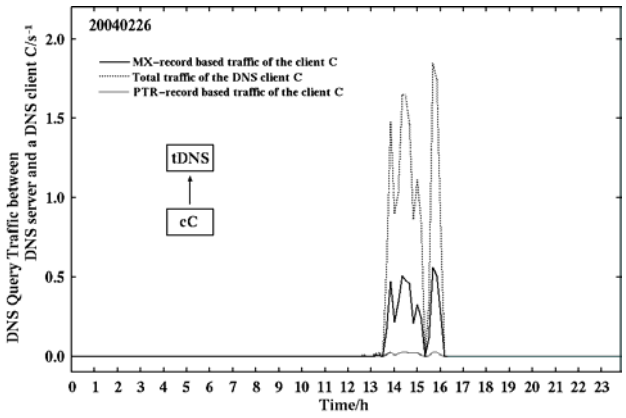

Figure 4. Traffic of the DNS query access between the top domain DNS server and the DNS client C at February 26th, 2004. The dotted line shows the total the total DNS traffic and the solid line indicates the MX-record based DNS traffic (s^{-1} unit).

3.2 Spam relay

We illustrate the DNS traffic between **tDNS** and DNS client C **cC** in Figure 4. The DNS query traffic mainly includes MX and A record packets and slightly includes PTR record packets. This traffic including PTR records clearly differs from the traffic from MMW-infected PC terminals that no PTR record is included. This difference is interpreted in terms that **cC** is a True64 UNIX Compaq Alpha PC terminal.

Since **cC** is left to be a default setting, **cC** becomes easily to be a spam relay by hijacking. In **cC**, a default SMTP server program (MTA; sendmail old version) is running. The SMTP server program like sendmail[19] or postfix[20] initially generates two DNS packets that consist of PTR and A records to check whether or not the SMTP client is authorized[15]. Furthermore, the SMTP server program request at least a couple of packets that consist of MX and A records to get FQDN of domain name E-mail address and to get an IP address of the FQDN that manages an E-mailing destination address[15]. It is fact that the total DNS query packets (9816) consist of 2922 MX, 6755 A, and 139 PTR record packets at February 26th, 2004. Therefore, we can detect a spam relay embedded UNIX-like PC terminal by presence of PTR records in its DNS query packets.

3.3 Development of MXRPDS

We have designed and developed an automatic MX record packet detection system (MXRPDS) consisting of DNS query MX and PTR record packets capture, PTR-record packet preprocessor "apra", a detection engine "mscan", and alert mailer "smail". The "mxrpds.pl" script hooks up the "mscan" script in order to scan the syslog file of **tDNS** in a time per 10 seconds.

In the DNS query packet capture, DNS query MX and PTR record request packets and their contents are recorded and decoded with the query-logging system of BIND-9.2.3[17].

The preprocessor "arpa" changes a description format of an IP address in the content of PTR record packet, like sorting "D.C.B.A.in-addr.arpa" to "A.B.C.D", where A, B, C, and D indicate 8 bits unsigned integer (0-255) values. This is because the described IP address in the content of PTR record is complicated for the detection engine. This "arpa" is a packet filter to be sensitive only for a string that includes a key word as "in-addr.arpa" and it is compiled with the gcc-2.95.3 C compiler. The preprocessor is called in the following detection engine and prints out the filtered contents of the PTR record packets into a "newdb" file and the old "newdb" file is renamed as an "olddb" file.

The detection engine "mscan" is a C-shell script program consisting of four components, an MMW IP filter "mmwip", a difference checker for MX record, a difference checker for PTR record, and an E-mailer without a local MTA "smail". The "mmwip" program compiled by the gcc-2.95.3 C compiler is a filter to discard the registered and/or authorized E-mail servers. The difference checker for MX (PTR) record packet is a "diff" command with an option "-c" to check difference between "olddb" ("polddb") and "newdb" ("pnewdb") files. Before the difference checker for PTR record packet, the preprocessor "arpa" is called. After the checker, if the "newdb" file differs from the "olddb" one, the following two results are obtained: (1) If the "pnewdb" file is equal to the "polddb" one, the difference means detecting a MMW-infected PC terminal. (2) If the "pnewdb" file differs the "polddb" one, the difference shows detecting a spam relay. These results are e-mailed to a network manager by the "smail" program.

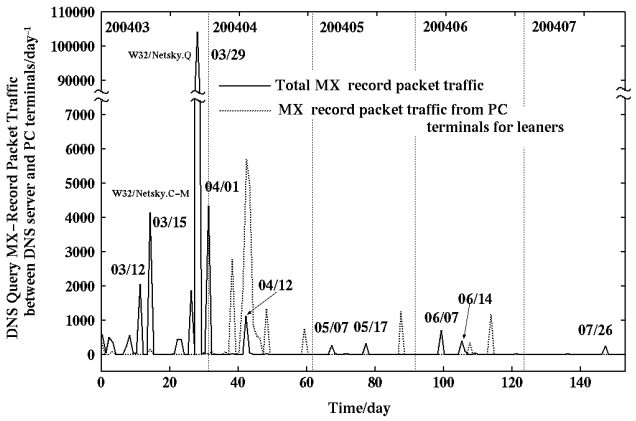

Figure 5. Traffic of the DNS query access between the top domain DNS server and the PC terminals through March 1st to July 31st, 2004. Both solid and dotted lines show the total illegal DNS query MX record traffic (day^{-1} unit) and MX record traffic related to PC terminals for learners (day^{-1} unit $\times 10^3$).

3.4 Evaluation

We implemented MXRPDS into the top domain DNS **tDNS** server and evaluated detection rate (March 1st, 2004). The machine in the evaluation has the following configuration: Intel Xeon 2.40GHz Dual CPU, 1GB main memory, Intel 100Mbps Ethernet NIC, and 80 GB ATA133 hard disk drive (The Linux kernel is currently to be a version of 2.4.26).

In Figure 5, the detection rate of the illegal DNS query MX record packet traffic increases when the W32/Netsky.C-Q MMWs are released suddenly. However, the detection rate drastically decreases after April 2nd, 2004. Note that the detection rate of the IP addresses of PC terminals for learners shown in Figure 5

seems to be very small (low), however, it also indicates that infection of internet worm like a mass mailing worm in the PC terminals for learners should be taken into consideration when designing a next future network system.

4. Concluding remarks

We statistically investigated system log (syslog) files in the top domain DNS server (**tDNS**) when several PC terminals were infected by mass mailing worm (MMW). By monitoring the DNS query accesses on **tDNS**, we have found information about detection of an IP address of a MMW-infected PC terminal: (1) Usually, the DNS query traffic of the DNS clients like Windows PC terminals includes an A (Address) record only, but it contains MX (Mail Exchange) and A records without PTR (Pointer/Reverse) record when the DNS clients are infected with the MMW like W32/Sobig.F(with W32/Sobig.C) and W32/Mydoom.A. (2) When the DNS clients like UNIX/UNIX-like PC terminals is unauthorized as a network sever in our university, the DNS query traffic is usually small but its traffic increases to a greater extent than that in the usual when the DNS client becomes a spam relay and it includes MX, A, and PTR records. From these points, we have developed and implemented an indirect virus detection system (MXRPDS) into our top domain DNS server. Successfully, we have been preventing the current MMW-infection.

We continue further investigation to get more detailed information on the system-vulnerability-attack worm (SVAW) like W32/Welchia SVAW and/or W32/Sasser.D SVAW.

Acknowledgement

All the calculations and investigations were carried out in Center for Multimedia and Information Technologies, Kumamoto University. We specially thank to technical officers, K. Tsuji, M. Shimamoto and T. Kida, and K. Makino who is a system engineer of MQS (Kumamoto) for daily support and constructive cooperation.

References

[1] http://www.trendmicro.com/vinfo/virusency-clo/default5.asp?VName=WORM_SOBIG.F

[2] http://www.trendmicro.com/vinfo/virusency-clo/default5.asp?VName=WORM_MYDOOM.A

[3] http://www.trendmicro.com/vinfo/virusency-clo/default5.asp?VName=WORM_NETSKY.Q

[4] http://www.trendmicro.com/vinfo/virusency-clo/default5.asp?VName=WORM_NACHI.A

[5] http://www.trendmicro.com/vinfo/virusency-clo/default5.asp?VName=WORM_SASSER.D

[6] S. Northcutt and J. Novak, *Network Intrusion Detection,* 2nd ed; New Riders Publishing: Indianapolis (2001).

[7] D. E. Denning, "An Intrusion-detection model", *IEEE Trans. Soft. Eng*., Vol. SE-13, No.2, 1987, pp.222-232.

[8] Cisco Systems: *The Science of Intrusion Detection System Attack Identification*, http://www.cisco.com/-warp/public/cc/pd/qsw/sqidsz/prodlit/idssa_wp.htm, 2002.

[9] B. Mukherjee, L. Todd, and K. N. Heberlein, "Network Intrusion Detection", *IEEE Network*, Vol.8, No3, 1994, pp.26-41.

[10] C. Warrender, S. Forrest, and B. Pearlmutter, "Detecting Intrusions Using System Calls: Alternative Data Models", *Proc. IEEE Symposium on Security and Privacy*, No.1, 1999, pp.133-145.

[11] S. A. Hofmeyr, A. Somayaji, and S. Forrest, "Intrusion Detection Using Sequences of System Calls", *Computer Security*, 1998, pp.151-180.

[12] D. Anderson, T. F. Lunt, H. Javitz, A. Tamaru, and A. Valdes, "Detecting unusual program behavior using statistical component of the Next-generation Intrusion Detection Expert System (NIDES)", *Computer Science Laboratory*, 1995, SRI-CSL-95-06.

[13] Symantec: ManHunt, http://enterprisesecurity.symantec.com/products/products.cfm?ProductID=156&EID=0

[14] http://www.snort.org/

[15] Y. Musashi, R. Matsuba, and K. Sugitani, "Traffic Analysis on a Domain Name System Server. SMTP Access Generates Many Name-Resolving Packets to a Greater Extent than Does POP3 Access", *Journal for Academic Computing and Networking*, No.6, 2002, pp.21-28.

[16] R. Matsuba, Y. Musashi, and K. Sugitani, "Statistical Analysis in Syslog Files in DNS and Spam SMTP Relay Servers", *IPSJ Symposium Series*, Vol. 2004, No. 3, 2004, pp.31-36.