Incidents of the Fraudulent Issue of Public Key Certificates

In this report we examine an Apache vulnerability and its handling, analyze the SpyEye crimeware kit that is now often used as an attack platform for monetary gain, and examine incidents of the fraudulent issue of public key certificates.

1.1 Introduction

This report summarizes incidents to which IIJ responded, based on general information obtained by IIJ itself related to the stable operation of the Internet, information from observations of incidents, information acquired through our services, and information obtained from companies and organizations with which IIJ has cooperative relationships. This volume covers the period of time from July 1 through September 30, 2011. In this period a number of vulnerabilities related to servers and browsers were discovered and fixed. There were also many reports of DDoS attacks, etc., on companies and government-related organizations in a number of countries, such as attacks on the Hong Kong Stock Exchange. Targeted attacks were also reported to have been made on military contractors in Japan in late September. As seen above, the Internet continues to experience many security-related incidents.

1.2 Incident Summary

Here, we discuss the IIJ handling and response to incidents that occurred between July 1 and September 30, 2011. Figure 1 shows the distribution of incidents handled during this period*1. From this report onward, in addition to incidents that IIJ directly responded to, we will also cover information obtained indirectly such as incidents occurring in foreign countries that may affect Japan.

Next, we will discuss the major incidents that occurred during this period.

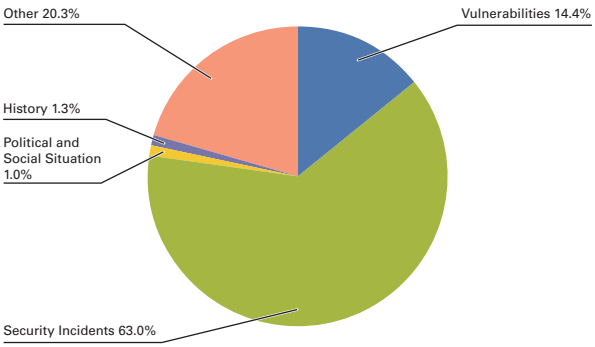


Figure 1: Incident Ratio by Category (July 1 to September 30, 2011)

■ Activities of Anonymous, etc.

Attacks by hacktivists*2 such as Anonymous continued during this period. DDoS attacks were made on the sites of government-related organizations and companies in countries such as the United States, India, Chile, Columbia, and Mexico stemming from a variety of incidents and causes. Between mid-July and August there were a number of attacks on sites related to Anonymous, but it is not known who instigated them. There were also a large number of information leaks from government-related sites and company sites.

*1 Incidents discussed in this report are categorized as vulnerabilities, political and social situation, history, security incident and other.
Vulnerabilities: Responses to vulnerabilities associated with network equipment, server equipment or software commonly used over the Internet or in user environments.
Political and Social Situations: Responses to incidents related to domestic and foreign circumstances and international events such as international conferences attended by VIPs and attacks originating in international disputes.
History: Historically significant dates; warning/alarms, detection of incidents, measures taken in response, etc., related to attacks in connection with a past historical fact.
Security Incidents: Unexpected incidents and related responses such as wide propagation of network worms and other malware; DDoS attacks against certain websites.
Other: Security-related information, and incidents not directly associated with security problems, including highly concentrated traffic associated with a notable event.

*2 Anonymous and related activities are discussed in IIR Vol.12 under "1.4.1 Continuing Attacks on Companies and Government-Related Organizations" (http://www.iij.ad.jp/en/company/development/iir/pdf/iir_vol12_EN.pdf).

For example, in August after incidents at San Francisco public transit authority BART there were demonstrations at stations, as well as DDoS attacks on BART-related websites, and information leaks from said websites. Actions related to the demonstrations held on Wall Street in New York were also ongoing at the time of writing.

■ Attacks Based on Political and Social Situation and Historical Context

On September 18 of last year, multiple large-scale DDoS attacks were made on a number of sites in Japan as part of a series of incidents originating from a boat collision off the Senkaku Islands*3. This year a number of attacks were observed on and around this day. Table 1 summarizes the attacks determined to be related to this incident based on attack warnings, etc. As this demonstrates, there were fewer attack targets and incidents than the attacks the previous year. The largest attack was a 635Mbps UDP flood, and 1.2Mpps/600Mbps SYN flood and HTTP GET flood compound attacks were also observed. The longest attack continued for approximately two hours.

Other characteristics of this year's incidents were the attacks made on general companies such as financial institutions, and the large number of SQL injection and brute-force password attacks used to hack, leak information, and alter content that occurred alongside DDoS attacks.

■ Targeted Attacks

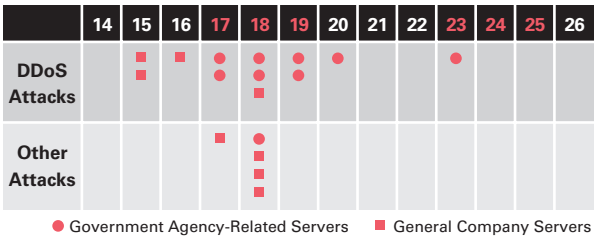
In mid-August a virus infection was discovered on the internal network of a major Japanese corporation, and was announced be a targeted attack via email. It was subsequently reported that similar attacks had been made on a number of other companies in the same industry.

In response to these attacks a number of public agency-led countermeasure activities have been assessed or implemented. Additionally, a concept proposed in the IPA guidelines*4 that were published ahead of the announcement of these attacks called outbound measures*5 is gaining attention as an effective countermeasure.

■ Vulnerabilities and their Handling

During this period a large number of vulnerabilities were discovered and fixed in Web browsers and user applications such as Microsoft's Internet Explorer*6, Adobe Systems' Adobe Reader and Acrobat, Flash Player, and Shockwave Player, and Apple's QuickTime.

Vulnerabilities were also found in the ISC BIND DNS Server and the Apache HTTP Server. There was no fix available for the Apache vulnerability when it was disclosed through a proof-of-concept program called Apache Killer, and we have confirmed that it was exploited before a fix was released. See "1.4.1 Apache Killer and its Handling" for more information about this vulnerability. In addition to these, vulnerabilities were also patched in the CMS platform WordPress and Microsoft's Windows DNS Server*7. A quarterly update for the Oracle database server was released, fixing a number of vulnerabilities. Cisco also released a scheduled update that patched a number of vulnerabilities including router firmware fixes that were delayed in consideration of the effects of the Great East Japan Earthquake.



This summarizes the attacks observed by IIJ during the current period that correspond to attack warnings. Attacks such as SQL injection attacks on Web servers and brute-force password search attacks on FTP servers are classified as "Other Attacks." The marks indicate days in which an attack on specific servers occurred. A single mark is used even when a server was attacked multiple times on a given day.

Table 1: Overview of Serial Attacks (September 2011)

*3 The series of attacks that took place during the same period last year are discussed in IIR Vo.10 under "1.4.1 An Overview of the Large-Scale DDoS Attacks in September 2010" (http://www.iij.ad.jp/en/company/development/iir/pdf/iir_vol10.pdf).

*4 IPA, "Design and Operational Guide to Cope with 'Advanced Persistent Threats'" (http://www.ipa.go.jp/security/vuln/documents/eg_newattack.pdf).

*5 Outbound measures stop the behavior and external control of malware that has infected an organization and prevents information leaks by regulating communications to the Internet from within the organization's network. It is implemented using methods such as black lists for servers identified through malware analysis. For more information see "4. Points to Counter New Threats" from P17 of the Design and Operational Guide to Cope with "Advanced Persistent Threats". Regarding IIJ services, we have provided this feature over the IIJ Secure Web Gateway Service since August 2009.

*6 "Microsoft Security Bulletin MS11-057 - Critical: Cumulative Security Update for Internet Explorer (2559049)" (http://technet.microsoft.com/en-us/security/bulletin/ms11-057).

*7 "Microsoft Security Bulletin MS11-058 - Critical: Vulnerability in DNS Server Could Allow Remote Code Execution (2562485)" (http://technet.microsoft.com/en-us/security/bulletin/ms11-058).

July Incidents

1 **1st:** Google deleted .co.cc subdomain sites from its index due to a large volume of malicious use, so they were no longer displayed in search results.
This policy was announced in the following Google Online Security Blog post in June. "Protecting users from malware hosted on bulk subdomain services" (http://googleonlinesecurity.blogspot.com/2011/06/protecting-users-from-malware-hosted-on.html).

2

3 **1st:** JNSA published the "Information Security Measure Guidebook for Telecommuters" to encourage office energy savings.
JNSA, "Information Security Measure Guidebook for Telecommuters" (http://www.jnsa.org/result/2011/zaitaku_guide.html) (in Japanese).

4 **1st:** JNSA published the "2010 Survey Report of Information Security Incidents - Personal Information Leak Edition".
JNSA, "2010 Survey Report of Information Security Incidents - Personal Information Leak Edition" (http://www.jnsa.org/result/incident/2010.html) (in Japanese).

5

6 **4th:** It was discovered that the vsftpd FTP server application had been altered to include a backdoor in the package. It was possible to detect altered packages by verifying their signature.
"Alert: vsftpd download backdoored" (http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html).

7

8 **5th:** BIND 9.7.3-P3 was released. It fixed several issues, including CVE-2011-2464 that led to DoS attacks on DNS servers when exploited.
Internet Systems Consortium, "ISC BIND 9 Remote packet Denial of Service against Authoritative and Recursive Servers" (https://www.isc.org/software/bind/advisories/CVE-2011-2464).

9

10 **8th:** A cell-phone carrier in Japan announced that a large-scale communications failure occurring in the Kansai area in May was an inside job caused by a former employee of a subcontractor.

11 **10th:** A DDoS attack was launched on the website of the National Police Agency, temporarily making their homepage inaccessible.
The following NPA announcement has more information on this incident. "Response to Cyber Attacks on the National Police Agency in July 2011" (http://www.npa.go.jp/keibi/biki3/230826kouhou.pdf) (in Japanese).

12

13 **13th:** Microsoft published their Security Bulletin Summary for July 2011, and released one critical and three important updates.
"Microsoft Security Bulletin Summary for July 2011" (http://technet.microsoft.com/en-us/security/bulletin/ms11-jul).

14

15 **14th:** At Janog28 discussions were held regarding the impact of the Great East Japan Earthquake on communications such as the Internet.
Janog Meeting, "Was the Internet in Japan robust against the earthquake?" (http://www.janog.gr.jp/en/index.php?JANOG28%20Programs#j8204a59).

16 **14th:** The United States Department of Defense announced its new cyber security strategy. At the same time they disclosed that 24,000 files had been leaked through attacks made from other countries in March on companies involved with national defense.
"Lynn: Cyber Strategy's Thrust is Defensive" (http://www.defense.gov/news/newsarticle.aspx?id=64682).

17

18 **20th:** The creator of the "octopus-squid virus" who was on trial for charges of property destruction was sentenced to prison.
TrendLabs SECURITY BLOG, "'Octopus-Squid Virus' Creator Receives Prison Sentence" (http://blog.trendmicro.co.jp/archives/4377) (in Japanese).

19 **20th:** Oracle released their quarterly scheduled update, fixing a total of 78 vulnerabilities.
"Oracle Critical Patch Update Advisory - July 2011" (http://www.oracle.com/technetwork/topics/security/cpujuly2011-313328.html).

20

21 **21th:** The first arrests were made for suspicion of storage of electromagnetic records of a computer virus based on the revised Penal Code that was enacted on July 14, which includes offenses related to virus creation.

22 **23th:** It was revealed that there was a vulnerability in the MacBook battery management interface that, if exploited by malware, could cause malfunctions or overheating.
ISC Diary, "Apple Battery Firmware Default Password" (http://isc.sans.edu/diary.html?storyid=11248).

23

24 **25th:** iOS 4.2.10/4.3.5 were released. These included fixes for SSL-related issues pointed out in CVE-2011-0228.
Apple, "About the security content of iOS 4.3.5 Software Update for iPhone" (http://support.apple.com/kb/HT4824).

25 **25th:** A large-scale alteration of sites using osCommerce took place.
Armorize Malware Blog, "willysy.com Mass Injection ongoing, over 8 million infected pages, targets osCommerce sites" (http://blog.armorize.com/2011/07/willysycom-mass-injection-ongoing.html).

26

27 **26th:** JPCERT/CC and others jointly conducted a briefing on the Act on the Partial Revision of the Penal Code, etc., in Response to the Sophistication of Information Processing (Cyber Penal Code, Code of Criminal Procedure).
JPCERT/CC, "Announcing a briefing on the Act on the Partial Revision of the Penal Code, etc., in Response to the Sophistication of Information Processing (Cyber Penal Code, Code of Criminal Procedure)" (http://www.jpcert.or.jp/event/keiji.html) (in Japanese).

28

29 **28th:** The personal information of up to 35 million individuals was leaked from South Korean portal sites (Nate and Cyworld). The information leaked included user IDs, names, mobile phone numbers, email addresses, passwords, and resident registration numbers. The passwords and resident registry numbers were encrypted.
TrendLabs MALWARE BLOG, "Large Data Breach in South Korea, Data of 35M Users Stolen" (http://blog.trendmicro.com/large-data-breach-in-south-korea-data-of-35m-users-stolen/).
TrendLabs MALWARE BLOG, "Updates on the SK Comms Data Breach" (http://blog.trendmicro.com/updates-on-the-sk-comms-data-breach/).

30

31

[Legend] V Vulnerabilities | S Security Incidents | P Political and Social Situation | H History | O Other

*Dates are in Japan Standard Time

■ Large-Scale Data Breach

A number of portal sites in South Korea were attacked, resulting in the leak of personal information for up to 35 million users. The leaked data is thought to have been encrypted, but it became a serious issue due to the leak affecting as many as 70% of South Korean citizens and the fact that personal information was involved.

In Japan it was also discovered that the personal information of a total of 25,000 individuals, including contract information for a number of insurance providers, had been obtained without authorization and sold to third parties. There was also an incident of a game server being hacked, leading to the leak of up to 203,000 user IDs, passwords, and email addresses.

■ Offenses Related to Virus Creation

On July 14 of this period the "Act on the Partial Revision of the Penal Code, etc., in Response to the Sophistication of Information Processing"*8 was enacted. On July 21 the first arrests of individuals suspected of storage of electromagnetic records of a computer virus were made. The individual believed to have created the octopus-squid virus who was arrested the year before this act was enacted received a prison sentence for the destruction of property.

Until now it was not possible to charge someone directly for actions such as the creation of malicious software, but with the enactment of this act the creation, sharing, or storage of viruses for malicious purposes can be treated as crimes. Meanwhile, because there are also concerns about the interpretation and application of the law, JPCERT/CC and others jointly held briefings and other events to explain these points.

■ Hacking Incidents and the Alteration of Web Content and Packages

There were also many incidents of hacking and hack-related alterations. There were a large number of website alteration incidents targeting a vulnerability in the osCommerce server application used for the construction of online shops*9. These incidents involved a technique whereby visiting users were redirected from an altered website to a malicious site, where they were infected with malware. It is thought that as many as 7,690,000 sites were altered through these incidents. There was also an incident involving the alteration of MySQL.com, where the same method was used to redirect visitors to a malicious website.

The Kernel.org site that manages the Linux kernel was also hacked, resulting in issues such as the alteration of SSH-related files and the creation of a backdoor into the system. This led to a large-scale verification effort to ensure that the Linux kernel programs distributed had not been altered. In a related incident, it was discovered that the Linux Foundation had also been hacked*10, with measures such as the temporary closure of the site being necessary to identify the extent of the impact and confirm what had been altered. Additionally, files for the vsftpd FTP server used with Linux and other systems were altered, and packages including a backdoor for use by hackers were distributed*11. Incidents involving the placing of maliciously altered software on legitimate distribution sites have also occurred frequently in the past*12, for example Sendmail and OpenSSH in 2002, and ProFTPD in 2010.

File signatures were not altered in the latest incidents, so it was possible to identify software that was not legitimate by checking signatures and hash values.

*8 Details of this act can be found on Japan's Ministry of Justice site. "Act on the Partial Revision of the Penal Code, etc., in Response to the Sophistication of Information Processing" (http://www.moj.go.jp/keiji1/keiji12_00025.html) (in Japanese).

*9 Information on these incidents is being reported on the following Armorize Malware Blog on an ongoing basis. "willysy.com Mass Injection ongoing, over 8 million infected pages, targets osCommerce sites" (http://blog.armorize.com/2011/07/willysycom-mass-injection-ongoing.html).

*10 Details of this incident can be found in the following Sophos blog post. "Security breach: Kernel.org and Linux Foundation remain 'temporarily unavailable'" (http://nakedsecurity.sophos.com/2011/09/12/linux-world-in-security-spinout/).

*11 The creator of vsftpd posted an explanation of this incident. "Alert: vsftpd download backdoored" (http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html).

*12 Previous incidents of alterations and methods for detecting them are explained in Vol.10 of this report under "1.4.3 Alteration of Software Distribution Packages" (http://www.iij.ad.jp/en/company/development/iir/pdf/iir_vol10.pdf).

August Incidents

1 **1st:** The IPA published the Design and Operational Guide to Cope with 'Advanced Persistent Threats.'
"Design and Operational Guide to Cope with 'Advanced Persistent Threats'" (http://www.ipa.go.jp/security/vuln/documents/eg_newattack.pdf).

2 **2nd:** ENISA identified a vulnerability in the HTML5 specifications.
"Agency ENISA flags security fixes for new web standards/HTML5"
(http://www.enisa.europa.eu/media/press-releases/web-security-eu-cyber-security-agency-enisa-flags-security-fixes-for-new-web-standards).

3 **2nd:** A phishing incident involving emails purportedly from MasterCard occurred in Japan.

4 **2nd:** An eavesdropping virus that records conversations on Android was discovered.
Total Defense: GLOBAL SECURITY ADVISOR RESEARCH BLOG, "A Trojan spying on your conversations"
(http://totaldefense.com/securityblog/2011/08/26/A-Trojan-spying-on-your-conversations.aspx).

5

6 **3rd:** A warning was issued regarding a series of incidents of unauthorized access at Internet banks in Japan.
IPA, "Regarding repeated incidents of unauthorized access of Internet banking in Japan" (http://www.ipa.go.jp/security/topics/alert20110803.html) (in Japanese).

7 **3rd:** A number of anti-virus software vendors published reports analyzing the Shady RAT targeted attacks.
McAfee, "Revealed: Operation Shady RAT" (http://www.mcafee.com/us/resources/white-papers/wp-operation-shady-rat.pdf).

8 **4th:** QuickTime 7.7 was released, fixing several vulnerabilities including CVE-2011-0245.
Apple, "About the security content of QuickTime 7.7" (http://support.apple.com/kb/HT4826).

9 **8th:** A large-scale power failure occurred at a data center in Dublin, United Kingdom, affecting the services of a number of cloud providers.

10 **9th:** A security update for Adobe Shockwave Player was released, fixing seven vulnerabilities.
"APSB11-10: Security update available for Adobe Shockwave Player" (http://www.adobe.com/support/security/bulletins/apsb11-19.html).

11 **10th:** Microsoft published their August 2011 security bulletin, and released fixes for two critical, nine important, and two warning updates, such as MS11-057 and MS11-058.
"Microsoft Security Bulletin Summary for August 2011" (http://technet.microsoft.com/en-us/security/bulletin/ms11-aug).

12 **10th:** A DDoS attack was launched on the Hong Kong Stock Exchange, preventing trading in a number of stocks for two consecutive days through to the 11th.
HKEx, "Disruption of HKExnews Website Services" (http://www.hkex.com.hk/eng/newsconsul/hkexnews/2011/1108104news.htm).
Sophos, naked security "Hong Kong stock exchange (HKEx) website hacked, impacts trades"
(http://nakedsecurity.sophos.com/2011/08/10/hong-kong-stock-exchange-hkex-website-hacked-impacts-trades/).

13

14

15 **12th:** It came to light that the source code for the "SpyEye" crimeware kit had leaked.
DAMBALLA, The Day Before Zero "First Zeus, now SpyEye. Look at the source code now!" (http://blog.damballa.com/?p=1357).

16 **12th:** A number of vulnerabilities were discovered and fixed in the Xen virtualization server, including the CVE-2011-3131 vulnerability that causes system service outages.
Red Hat Bugzilla, "CVE-2011-3131 kernel: xen: IOMMU fault livelock" (https://bugzilla.redhat.com/show_bug.cgi?id=730341).

17 **14th:** A number of vulnerabilities were discovered and fixed in Ruby.
Red Hat Bugzilla, "CVE-2011-2686 CVE-2011-2705 CVE-2011-3009 ruby: Properly initialize the random number generator when forking new process" (https://bugzilla.redhat.com/show_bug.cgi?id=722415).

18

19 **15th:** An attack was launched on a major message board in Japan in relation to the anniversary of the end of World War II.

20 **17th:** Researchers discovered a vulnerability in the AES cryptographic algorithm. However, it was also confirmed that its use was not an immediate threat to security.
Microsoft Research, "Biclique cryptanalysis of the full AES" (http://research.microsoft.com/en-us/projects/cryptanalysis/aes.aspx).

21 **19th:** There were reports of the spread of emails with malware attachments disguised as invoices.
Sophos, naked security "Inter-company invoice emails carry malware" (http://nakedsecurity.sophos.com/2011/08/18/inter-company-invoice-emails-malware/).

22 **20th:** The CVE-2011-3192 vulnerability in Apache was disclosed and later fixed. At the time that this vulnerability was made public it had not yet been fixed.
See the Full Disclosure mailing list for more information on Apache Killer. Details of this vulnerability can be found in the following Apache Foundation advisory. "Apache HTTPD Security ADVISORY UPDATE 3 - FINAL" (http://httpd.apache.org/security/CVE-2011-3192.txt).

23

24 **24th:** A phishing site masquerading as that of a major Japanese service provider was confirmed, and a warning was issued. Because this phishing site requested the entry of IDs and passwords, it is thought to have been aimed at identity theft or unauthorized use of services.

25 **25th:** A major message board with many users was attacked using the Apache Killer tool for detecting an unpatched vulnerability in Apache.

26 **28th:** It was revealed that Kernel.org had been hacked about a month earlier. Account information was leaked and files were altered.
Linux Foundation, "The cracking of kernel.org" (https://www.linuxfoundation.org/news-media/blogs/browse/2011/08/cracking-kernelorg).

27 **28th:** The CVE-2011-3205 vulnerability in Squid that could trigger a DoS attack when connected to a gopher server was fixed.
"Squid Proxy Cache Security Update Advisory SQUID-2011:3" (http://www.squid-cache.org/Advisories/SQUID-2011_3.txt).

28 **28th:** It was reported that the Morto Worm that infects via RDP was spreading.
F-Secure Weblog, "Windows Remote Desktop Worm 'Morto' Spreading" (http://www.f-secure.com/weblog/archives/00002227.html).

29 **30th:** It came to light that the Dutch certificate authority DigiNotar had been hacked in July, and a large volume of fraudulent SSL certificates had been issued.
F-Secure Weblog, "DigiNotar Hacked by Black.Spook and Iranian Hackers" (http://www.f-secure.com/weblog/archives/00002228.html).

30 **31st:** A smartphone application that revealed information such as the location and call logs of phones became an issue due to it failing to obtain sufficient confirmation from the actual users.

31 **31st:** The CVE-2011-2901 vulnerability in Xen that could cause system service outages from a guest OS was fixed.

[Legend] V Vulnerabilities | S Security Incidents | P Political and Social Situation | H History | O Other

*Dates are in Japan Standard Time

DDoS Attacks

During this period a large number of DDoS attacks also took place, including an incident in which the National Police Agency website was temporarily made inaccessible, and DDoS attacks targeting a major message board on August 15. IIJ also observed a UDP flood attack in July with a maximum bandwidth of 3Gbps using up to 450,000pps.

Overseas there were DDoS attacks on the Hong Kong Stock Exchange that resulted in some stocks not being tradable for two consecutive days, and this made headlines as a case of cyberterrorism against the crucial infrastructure of the financial sector that led to actual damages.

Warnings Regarding Phishing and Internet Banking in Japan

There were continued incidents of users being directed to phishing sites through email, and attacks via attachments containing malware. Phishing sites that masqueraded as organizations such as Internet banks, credit card companies, and providers were confirmed, as well as related malware that steals IDs and passwords. With regard to Internet banking in Japan in particular, because of a string of incidents involving customer information such as passwords being stolen via suspicious emails and spyware and used to commit fraudulent bank transfers, the IPA issued a warning regarding malicious software and spam.

Incidents Caused by Insiders

A communications failure at a mobile phone network in the Kansai area of Japan in May was found to have been caused by a malicious program created by a former contract worker, who was arrested on suspicion of obstruction of business by damaging a computer. In an incident involving the alteration of game data for 1.3 million users of a mobile social game, an individual who did temp work at the game developer until March was arrested on suspicion of violation of the Prohibition of Unauthorized Computer Access Law and obstruction of business by damaging a computer. He is believed to have accessed a computer without authorization by installing a program that could disable server access restrictions. In the United States, content on a number of internal virtual servers belonging to a pharmaceutical company was deleted in February of this year, and the culprit was a former employee who had been fired*13. Because these virtual servers contained many business systems, the deletions reportedly had significant impact.

The sharing of information and countermeasures regarding incidents caused by insiders has been under review for a long time. For example, CERT/CC in the United States began research into this in 2001, and has provided information on their blog on a regular basis since last year*14. A related survey was also conducted in Japan last year*15.

Shady RAT

In September a number of anti-virus vendors reported on attacks known as Shady RAT. The first of these was a report published by McAfee based on actual incidents of targeted attacks and hacking that took place as part of these attacks*16. This report noted that attacks using this system had been carried out on a number of organizations including government agencies over a span of five years. Symantec also published a detailed explanation of these same attacks*17. It has been revealed that some of the attacks disguised control communications as normal HTTP traffic using techniques such as hiding commands in image files via steganography, or hiding commands in encrypted HTML comments.

*13 Details of these incidents can be found in the following United States Department of Justice press release. (http://www.justice.gov/usao/nj/Press/files/Cornish,%20Jason%20Plea%20News%20Release.html).

*14 CERT/CC, "Insider Threat Research" (http://www.cert.org/insider_threat/).

*15 Research Foundation for Safe Society - Research Committee on Countermeasures for Human Threats in Information Security, "Research Report on Countermeasures for Human Threats in Information Security" (http://www.syaanken.or.jp/02_goannai/08_cyber/cyber2203_01/pdf/cyber2203_01.pdf) (in Japanese).

*16 See McAfee's "Revealed: Operation Shady RAT" (http://www.mcafee.com/us/resources/white-papers/wp-operation-shady-rat.pdf).

*17 Symantec Security Response Blog, "The Truth Behind the Shady RAT" (http://www.symantec.com/connect/blogs/truth-behind-shady-rat).

September Incidents

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

V **7th:** Fixes were made to vulnerabilities CVE-2011-3207, which made it possible to bypass certificate revocation list validation in OpenSSL, and CVE-2011-3210, which caused system service outages.
"OpenSSL Security Advisory [6 September 2011]" (http://www.openssl.org/news/secadv_20110906.txt).

S **8th:** A suspect was arrested for an incident in which the data of 1.3 million social game users was altered. He is suspected of installing a backdoor on the server to access it without authorization after finishing temp work at the game developer.

S **10th:** The Twitter account of NBC news was hacked and used to spread fake news stories about Ground Zero. One of the three individuals responsible for managing the Twitter account was subject to a targeted email attack in which a keylogger was installed.
Sophos, naked security "Script Kiddies" (http://nakedsecurity.sophos.com/2011/09/13/christmas-tree-trojan-blamed-for-nbc-news-twitter-hack/).

S **12th:** The Linux Foundation site was temporarily taken offline after it was discovered that it had been hacked in relation to the hacking of Kernel.org.

V **13th:** A security update (APSB11-24) that fixed a number of vulnerabilities in Adobe Reader and Acrobat was released.
"Security updates available for Adobe Reader and Acrobat" (http://www.adobe.com/support/security/bulletins/apsb11-24.html).

V **14th:** Apache 2.2.21 was released with further fixes related to the CVE-2011-3192 vulnerability.
"Apache HTTP Server 2.2.21 Released" (http://www.apache.org/dist/httpd/Announcement2.2.html).

V **14th:** Microsoft published their September 2011 security bulletin, and released five important updates including MS11-071 and MS11-073.
"Microsoft Security Bulletin Summary for September 2011" (http://technet.microsoft.com/en-us/security/bulletin/ms11-sep).

S **18th:** A number of attacks were launched on Japanese government agencies and private-sector businesses on and around this day.

S **19th:** It was discovered that a major Japanese corporation had been infected with malware in a targeted attack via email. It later came to light that similar attacks had been made on other companies.

S **20th:** It was reported that a number of websites had been altered in relation to attacks on Japanese websites that occurred on September 18.

S **20th:** Dutch company DigiNotar filed for bankruptcy in the wake of the security breach that took place there.
ISC Diary, "Diginotar declared bankrupt" (http://isc.sans.edu/diary.html?storyid=11614).

V **21th:** A security update (APSB11-26) that fixed a number of vulnerabilities in Adobe Flash Player was released.
"Security update available for Adobe Flash Player" (http://www.adobe.com/support/security/bulletins/apsb11-26.html).

V **22nd:** A vulnerability in WordPress that made clickjacking attacks possible was made public. This vulnerability was fixed after protective functions were implemented in version 3.1.3 released in May.
Full Disclosure mailing list, "WordPress <=v3.1.2 Clickjacking Vulnerability Advisory" (http://seclists.org/fulldisclosure/2011/Sep/219).
"WordPress 3.1.3 (and WordPress 3.2 Beta 2)" (http://wordpress.org/news/2011/05/wordpress-3-1-3/).

V **24th:** A new technique for exploiting SSL/TLS vulnerabilities called BEAST was made public.
Thai Duong, "BEAST" (http://vnhacker.blogspot.com/2011/09/beast.html).

V **26th:** A vulnerability (CVE-2011-2483) in PostgreSQL that caused passwords to be saved using weak encryption in relation to the blowfish algorithm was fixed.
"PostgreSQL 2011-09-26 Cumulative Bug-Fix Release" (http://www.postgresql.org/about/news.1355).

S **27th:** MySQL.com was altered so that users viewing the website would be redirected to a malware infection site.
"MySQL.com Security Notice" (http://www.mysql.com/news-and-events/generate-article.php?id=1691).

S **27th:** Microsoft and anti-virus software vendors shut down the activity of the "Kelihos" botnet.
"Microsoft Neutralizes Kelihos Botnet, Names Defendant in Case"
(http://blogs.technet.com/b/microsoft_blog/archive/2011/09/27/microsoft-neutralizes-kelihos-botnet-names-defendant-in-case.aspx).

V **27th:** Firefox 7.0 was released, fixing multiple vulnerabilities such as CVE-2011-3002, which made it possible for external parties to cause service outages.
"Security Advisories for Firefox" (http://www.mozilla.org/security/known-vulnerabilities/firefox.html).

V **29th:** A Cisco Security Advisory was released, fixing 10 vulnerabilities.
"Cisco Event Response: Semi-Annual Cisco IOS Software Security Advisory Bundled Publication"
(http://www.cisco.com/web/about/security/intelligence/Cisco_ERP_sep11.html).

[Legend] V Vulnerabilities | S Security Incidents | P Political and Social Situation | H History | O Other

*Dates are in Japan Standard Time

Cryptographic Technology Trends

At the international conference on cryptography (CRYPTO 2011) in August, a new cryptanalytic attack for the AES cryptographic algorithm was made public*18. This technique deciphers encryption with comparatively less computational effort than the theoretical value necessary for brute force attacks on keys. However, at this point it does not critically impact cryptographic strength, and it will not cause issues with the use of AES.

A tool for eavesdropping cookies via vulnerabilities in SSL and TLS1.0 when using the CBC block cipher mode was also made public*19. This issue was dealt with comprehensively in the 2006 revision of the protocol specification (RFC4346), and does not affect TLS1.1 or 1.2. Attacks can be avoided by using the RC4 cryptographic algorithm or by using modes other than CBC (for example CTR).

Regarding the security breach of Dutch certificate authority DigiNotar in July, major browsers implemented measures to remove trust for self-signed certificates held by affected certificate authorities*20. See "1.4.3 Incidents of the Fraudulent Issue of Public Key Certificates" for more information about this incident. Additionally, because the culprit alluded to also hacking certificate authorities other than DigiNotar, named certificate authorities took measures such as suspending their operations to investigate. As this demonstrates, the impact was much larger than originally expected, and certificate reliability was seriously threatened.

Other Trends

In other trends, JNSA published their "2010 Survey Report of Information Security Incidents - Personal Information Leak Edition," which reports data on personal information leaks in 2010 gathered and analyzed via their independent survey model. They also published their "Information Security Measure Guidebook for Telecommuters," which provides guidelines for implementing telecommuting. Interest in telecommuting rose after the Great East Japan Earthquake and the energy savings measures this summer.

*18 This technique is discussed on the following Microsoft Research site. "Biclique cryptanalysis of the full AES" (http://research.microsoft.com/en-us/projects/cryptanalysis/aes.aspx).

*19 See the blog of presenter Thai Duong for more details. "BEAST" (http://vnhacker.blogspot.com/2011/09/beast.html).

*20 Measures such as removing certificates via updates were taken for each browser and application. The responses for major browsers are as follows. "Mozilla Foundation Security Advisory 2011-34" (http://www.mozilla.org/security/announce/2011/mfsa2011-34.html). "Mozilla Foundation Security Advisory 2011-35" (http://www.mozilla.org/security/announce/2011/mfsa2011-35.html). "Security fixes and Opera's phishing and malware prevention features in 11.51" (http://my.opera.com/chooseopera-Japan/blog/2011/09/01/11-51-opera) (in Japanese). "Stable Channel Update" (http://googlechromereleases.blogspot.com/2011/09/stable-channel-update.html). Similar measures were also taken for OSes and applications. "Microsoft Security Advisory (2607712) Fraudulent Digital Certificates Could Allow Spoofing" (http://technet.microsoft.com/en-us/security/advisory/2607712). "About Security Update 2011-005" (http://support.apple.com/kb/HT4920). "JPCERT/CC Alert 14.09.11 Vulnerabilities in Adobe Reader and Acrobat" (http://www.jpcert.or.jp/english/at/2011/at110025.html).

1.3 Incident Survey

1.3.1 DDoS Attacks

Today, DDoS attacks on corporate servers are almost a daily occurrence, and the methods involved vary widely. However, most of these attacks are not the type that utilizes advanced knowledge such as that of vulnerabilities, but rather cause large volumes of unnecessary traffic to overwhelm network bandwidth or server processes for the purpose of hindering services.

■ Direct Observations

Figure 2 shows the circumstances of DDoS attacks handled by the IIJ DDoS Defense Service between July 1 and September 30, 2011. This information shows traffic anomalies judged to be attacks based on IIJ DDoS Defense Service standards. IIJ has also responded to other DDoS attacks, but these incidents are excluded from the figure due to the difficulty in accurately ascertaining the facts of each situation.

There are many methods that can be used to carry out a DDoS attack, and the capacity of the environment attacked (bandwidth and server performance) will largely determine the degree of impact. Figure 2 categorizes DDoS attacks into three types: attacks on bandwidth capacity*21, attacks on servers*22, and compound attacks (when both types of attacks are conducted at the same time).

During the three months under study, IIJ dealt with 561 DDoS attacks. This comes to 6.1 attacks per day, indicating an increase in the average daily number of attacks compared to our prior report. Bandwidth capacity attacks accounted for 0.2% of all incidents, server attacks accounted for 80.7% of all incidents, and compound attacks accounted for the remaining 19.1%.

The largest attack observed during the period under study was classified as a compound attack, and resulted in 1.5Gbps of bandwidth using up to 4,350,000pps packets over the course of two hours and 15 minutes. Of all attacks, 84.8% ended within 30 minutes of commencement, while the remaining 15.2% lasted between 30 minutes and 24 hours. The longest sustained attack was a compound attack that lasted for two hours and 47 minutes. Regarding the relationship between the scale and duration of attacks, we have observed a trend in which smaller attacks are sustained for shorter periods of time, while larger attacks are sustained for longer.

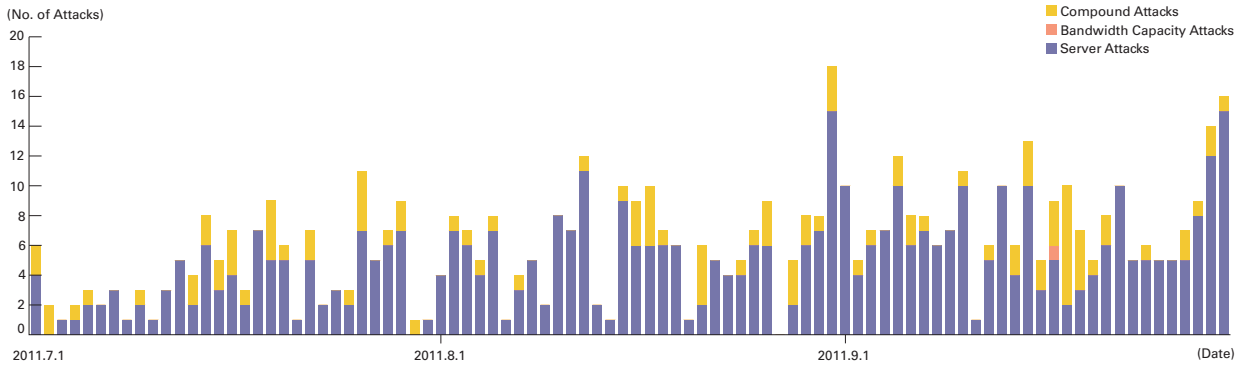


Figure 2: Trends in DDoS Attacks

*21 Attack that overwhelms the network bandwidth capacity of a target by sending massive volumes of larger-than-necessary IP packets and fragments. The use of UDP packets is called a UDP flood, while the use of ICMP packets is called an ICMP flood.

*22 TCP SYN flood, TCP connection flood, and HTTP GET flood attacks. TCP SYN flood attacks send mass volumes of SYN packets that signal the start of TCP connections, forcing the target to prepare for major incoming connections, causing the wastage of processing capacity and memory. TCP connection flood attacks establish mass volumes of actual TCP connections. HTTP GET flood attacks establish TCP connections on a Web server, and then send mass volumes of HTTP GET protocol commands, wasting processing capacity and memory.

In most cases, we observed an extremely large number of IP addresses, whether domestic or foreign. We believe this is accounted for by the use of IP spoofing*23 and botnet*24 usage as the method for conducting DDoS attacks.

Backscatter Observations

Next we present our observations of DDoS attack backscatter using the honeypots*25 set up by the MITF, a malware activity observation project operated by IIJ*26. By monitoring backscatter it is possible to detect DDoS attacks occurring on external networks as a third party without any interposition.

For the backscatter observed between July 1 and September 30, 2011, Figure 3 shows trends in packet numbers by port, and Figure 4 shows the sender's IP addresses classified by country. The port most commonly targeted by the DDoS attacks observed was the 80/TCP port used for Web services, accounting for 71.4% of the total during the target period. Attacks on 3389/TCP used for remote desktop and 21/TCP used by FTP were also observed. Looking at the origin of backscatter thought to indicate IP addresses targeted by DDoS attacks by country in Figure 4, China and the United States accounted for large proportions at 42.4% and 28.6%, respectively, with other countries following in order.

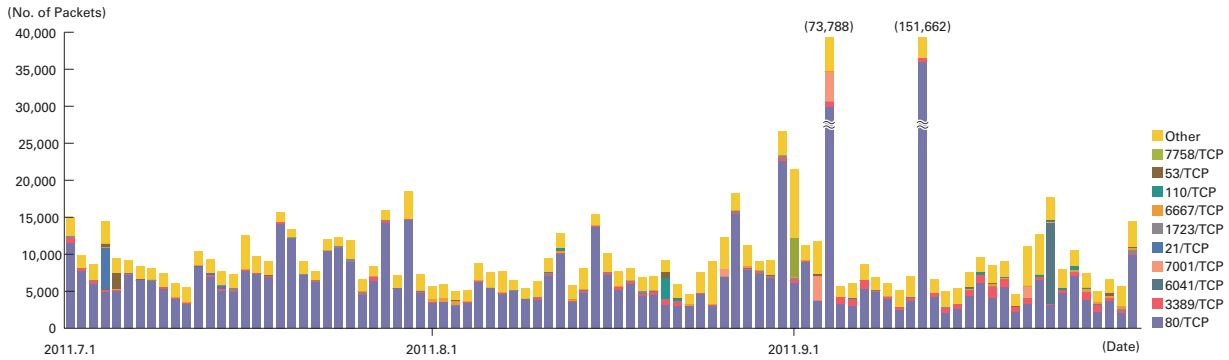


Figure 3: Observations of Backscatter Caused by DDoS Attacks (Observed Packets, Trends by Port)

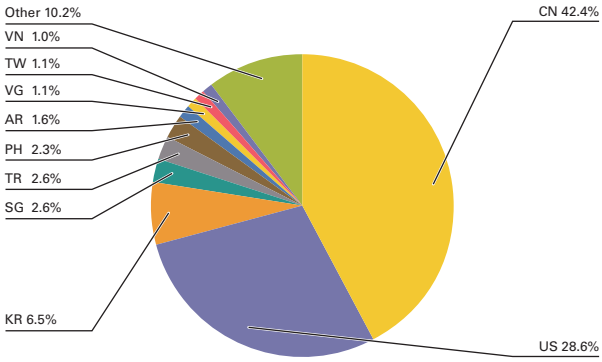


Figure 4: Distribution of DDoS Attack Targets According to Backscatter Observations (by Country, Entire Period under Study)

*23 Misrepresentation of a sender's IP address. Creates and sends an attack packet that has been given an address other than the actual IP address of the attacker in order to make it appear as if the attack is coming from a different location, or from a large number of individuals.

*24 A "bot" is a type of malware that institutes an attack after receiving a command from an external C&C server. A network constructed of a large number of bots acting in concert is called a "botnet."

*25 Honeypots established by the MITF, a malware activity observation project operated by IIJ. See also "1.3.2 Malware Activities."

*26 The mechanism and limitations of this observation method as well as some of the results of IIJ's observations are presented in Vol.8 of this report under "1.4.2 Observations on Backscatter Caused by DDoS Attacks" (http://www.iij.ad.jp/en/company/development/iir/pdf/iir_vol08_EN.pdf).

Regarding particularly large numbers of backscatter packets observed, there were two attacks in September targeting Web servers (80/TCP) in China. We also observed attacks on the FTP servers (21/TCP) of U.S. companies on July 4, and DNS servers (53/TCP) and a POP3 server (110/TCP) in the United States on July 5 and August 21, respectively. In China an attack targeting 7758/TCP took place on September 1, and an attack on 6041/TCP was observed on September 23. The applications corresponding to these two ports are unclear, but it is known that the latter attack targeted a server of a game-related company.

1.3.2 Malware Activities

Here, we will discuss the results of the observations of the MITF*27, a malware activity observation project operated by IIJ. The MITF uses honeypots*28 connected to the Internet in a manner similar to general users in order to observe communications arriving over the Internet. Most appear to be communications by malware selecting a target at random, or scans attempting to locate a target for attack.

■ Status of Random Communications

Figure 5 shows trends in the total volumes of communications coming into the honeypots (incoming packets) between July 1 and September 30, 2011. Figure 6 shows the distribution of sender's IP addresses by country. The MITF has set up numerous honeypots for the purpose of observation. We have taken the average per honeypot, showing the trends for incoming packet types (top ten) over the entire period subject to study. Additionally, in these observations we corrected data to count multiple TCP connections as a single attack when the attack involved multiple connections to a specific port, such as attacks on MSRPC.

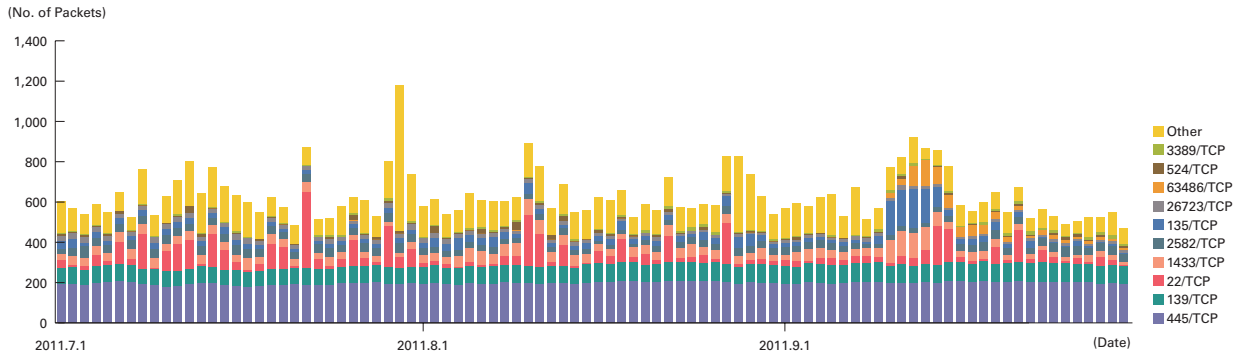


Figure 5: Communications Arriving at Honeypots (by Date, by Target Port, per Honeypot)

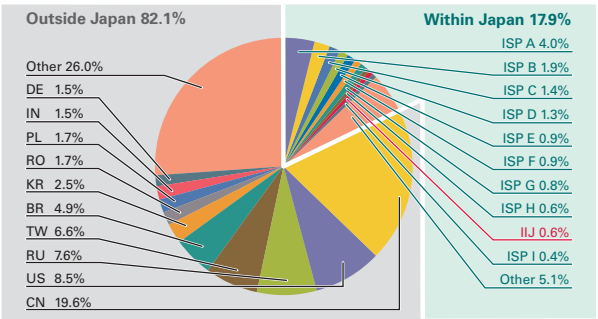


Figure 6: Sender Distribution (by Country, Entire Period under Study)

*27 An abbreviation of Malware Investigation Task Force. The Malware Investigation Task Force (MITF) began activities in May 2007 observing malware network activity through the use of honeypots in an attempt to understand the state of malware activities, to gather technical information for countermeasures, and to link these findings to actual countermeasures.

*28 A system designed to simulate damages from attacks by emulating vulnerabilities, recording the behavior of attackers, and the activities of malware.

Much of the communications arriving at the honeypots demonstrated scanning behavior targeting TCP ports utilized by Microsoft operating systems. We also observed scanning behavior for 1433/TCP used by Microsoft's SQL Server and 22/TCP used for SSH. Additionally, communications of an unknown purpose were observed on ports not used by common applications, such as 2582/TCP, 26723/TCP, and 63486/TCP. Looking at the overall sender distribution by country in Figure 6, we see that attacks sourced to China at 19.6% and Japan at 17.9% were comparatively higher than the rest.

Between September 10 and September 15, communications from a specific ISP in Japan and the United States targeting 135/TCP and 1433/TCP increased. Communications thought to be SSH dictionary attacks also occurred intermittently. For example, concentrated communications was observed coming from IP addresses in China on July 22, the United States on August 10, and the Netherlands on August 27.

■ The Morto Worm

During the current period the Morto worm that spreads by launching dictionary attacks on the RDP Windows remote login function surfaced*[29]. Figure 7 shows the RDP (3389/TCP) communications arriving at honeypots. RDP is probed on a daily basis, but we can see that this behavior increased between August 5 and August 13, before once again rising intermittently between August 17 and August 27. Communications also repeatedly increased then fell between September 9 and September 13, and from September 18 onwards, maintaining a higher level than normal. We believe that this is due to the spread of the Morto worm. In the past communications targeting RDP were mainly from China, so the fact that communications also arrived from other countries during the abovementioned periods suggests that the worm may have spread worldwide.

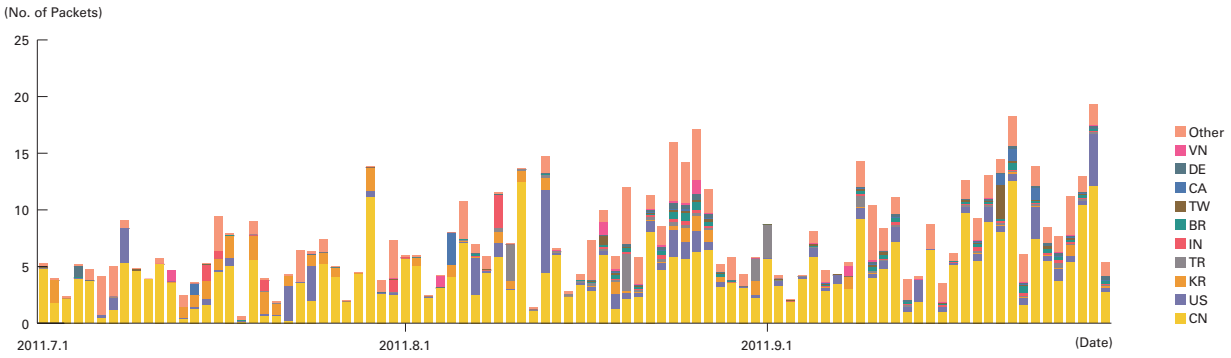


Figure 7: RDP (3389/TCP) Communications Arriving at Honeypots (by Date, by Country, per Honeypot)

*29 The behavior of this malware is examined in the following F-Secure blog post: "Windows Remote Desktop Worm 'Morto' Spreading" (http://www.f-secure.com/weblog/archives/00002227.html).

Malware Network Activity

Figure 8 shows the distribution of the specimen acquisition source for malware during the period under study, while Figure 9 shows trends in the total number of malware specimens acquired. Figure 10 shows trends in the number of unique specimens. In Figure 9 and Figure 10, the trends in the number of acquired specimens show the total number of specimens acquired per day*30, while the number of unique specimens is the number of specimen variants categorized according to their digest of a hash function*31. Specimens are also identified using anti-virus software, and a breakdown of the top 10 variants is displayed color coded by malware name.

On average, 57,352 specimens were acquired per day during the period under study, representing 1,294 different malware variants. Conficker variants were the dominant form of malware, accounting for 73.7% of the total number of specimens acquired, and 70.1% of unique specimens. The distribution of specimens according to source country in Figure 8 had Japan at 2.9%, with other countries accounting for the 97.1% balance. This is because Conficker was mainly active on a large-scale outside Japan. During the current period the number of unique specimens remained constant, while an upward trend was seen in the total number of specimens. This is because the activity of some Conficker variants rose slightly.

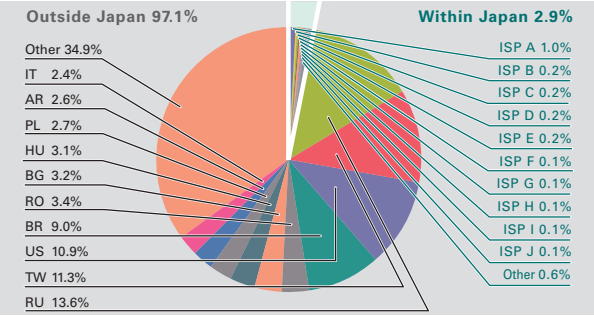


Figure 8: Distribution of Acquired Specimens by Source (by Country, Entire Period under Study)

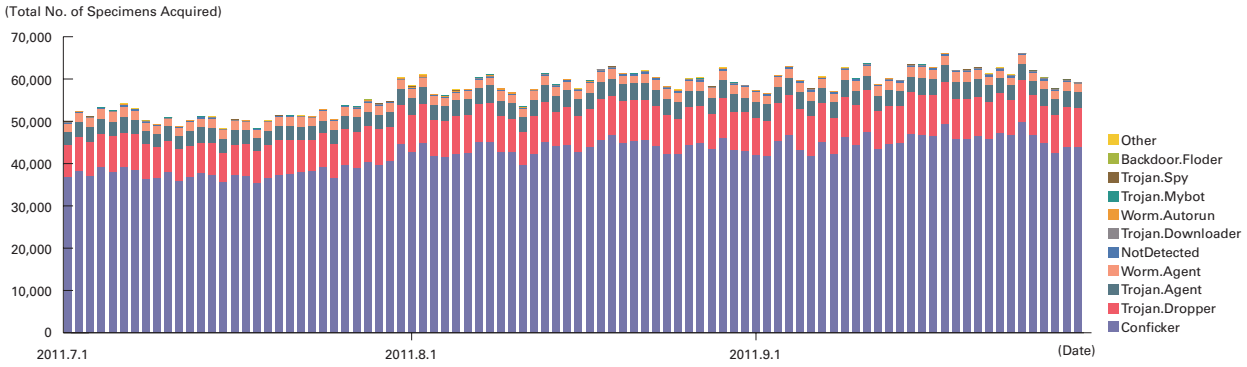


Figure 9: Trends in the Number of Malware Specimens Acquired

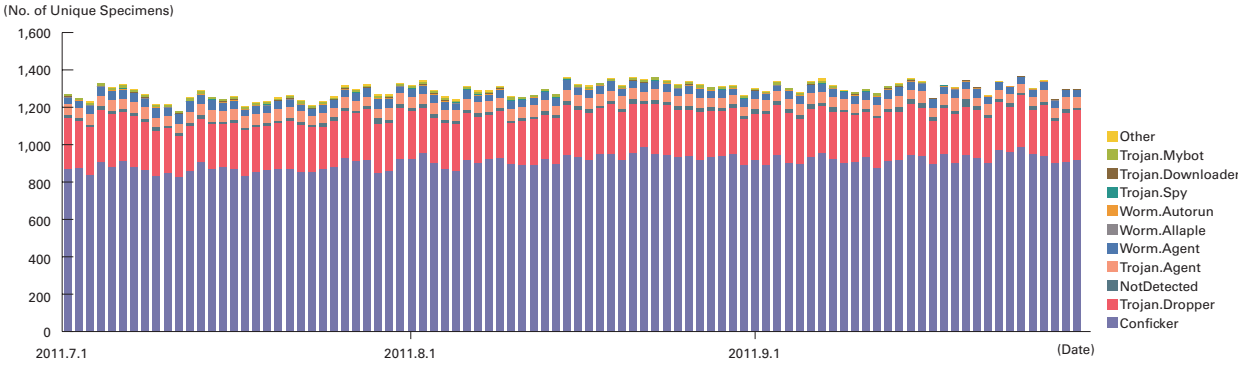


Figure 10: Trends in the Number of Unique Specimens

*30 This indicates the malware acquired by honeypots.

*31 This figure is derived by utilizing a one-way function (hash function) that outputs a fixed-length value for various input. The hash function is designed to produce as many different outputs as possible for different inputs. While we cannot guarantee the uniqueness of specimens by hash value, given that obfuscation and padding may result in specimens of the same malware having different hash values, in this section we take this into consideration when using this methodology as a measurement index.

Next, for the same period we use the method discussed below to exclude specimens determined to be Conficker variants, and show the distribution of the specimen acquisition source for malware in Figure 11, and trends in the total number of malware specimens acquired in Figure 12. Figure 13 shows trends in the number of unique specimens. In Figure 12 and Figure 13, the trends in the number of acquired specimens show the total number of specimens acquired per day, while the number of unique specimens is the number of specimen variants categorized according to their digest of a hash function. In Figure 11, specimens acquired from Thailand accounted for a large proportion at 27.6%. This is due to the fact that specimens active for just a day or two were observed in a large number. In Figure 12 we can see that Mybot infection activity is taking place. Most of this activity came from IP addresses allocated to Taiwan. However, for unknown reasons Mybot activity ceased all at once on September 16 worldwide. Breaking down the unknown specimens (NotDetected) that account for the largest proportion in the figure, 87.1% were executable files, 11.6% were text files such as HTML or XML, and the remaining 1.3% were unknown binary data.

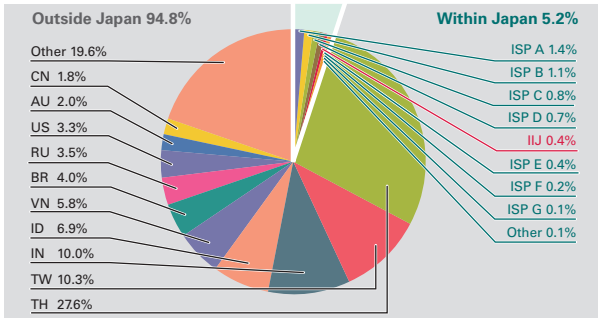


Figure 11: Distribution of Acquired Specimens by Source (by Country, Entire Period under Study, Excluding Conficker)

Under MITF's independent analysis, during the current period under observation 86.9% of malware specimens acquired were worms, 1.4% were bots, and 11.7% were downloaders. In addition, the MITF confirmed through the analyses the presence of 16 botnet C&C servers*32 and 16 malware distribution sites.

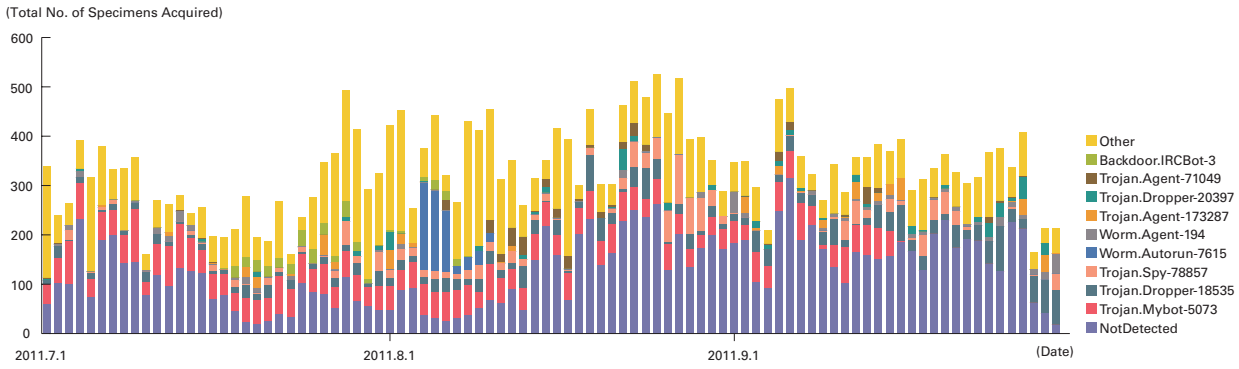


Figure 12: Trends in the Number of Malware Specimens Acquired (Excluding Conficker)

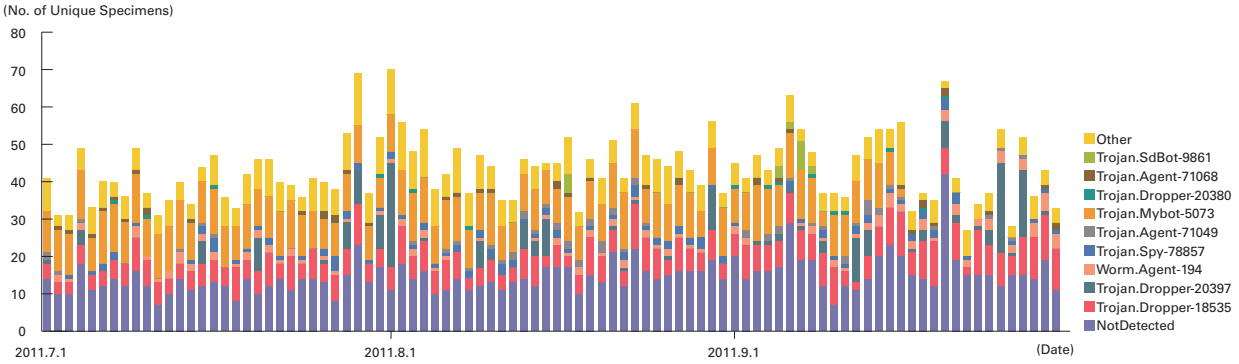


Figure 13: Trends in the Number of Unique Specimens (Excluding Conficker)

*32 An abbreviation of "Command & Control." A server that provides commands to a botnet consisting of a large number of bots.

Identifying Conficker

In this report we previously used the ClamAV anti-virus software for detection. However, we have learned that when using this software some specimens that exhibit behavior similar to Conficker are identified as other malware. For this reason, in this report we have determined whether a specimen is Conficker by the name assigned most often using multiple anti-virus software. Through this process we determined that 99.4% of the total number of malware specimens acquired and 96.6% of the unique specimens were Conficker, and based on these results we created Figures 11 through 13.

Anti-virus software is designed to discover and remove malware promptly, and malware can be removed as long as it is detected regardless of the name assigned. In other words, differences in the identification of specific specimens are due to varying detection methods that take into account the speed and workload of detection processing, and do not indicate differences in the capabilities of anti-virus software. It is important to keep naming consistent when the intent is to indicate the current state of malware active on networks such as in this report, and for this reason we have used results from a single anti-virus software solution in the past. However, from this report we adopted a new method after it became clear that a single solution would not indicate the current status of the Conficker malware that is spreading quickly around the world.

1.3.3 SQL Injection Attacks

Among the different types of Web server attacks, IIJ conducts ongoing surveys related to SQL injection attacks*33. SQL injection attacks have flared up in frequency numerous times in the past. SQL injections are known to occur in one of three attack patterns: those that attempt to steal data, those that attempt to overload database servers, and those that attempt to rewrite Web content.

Figure 14 shows the distribution of attacks according to source, and Figure 15 shows trends in the numbers of SQL injection attacks against Web servers detected between July 1 and September 30, 2011. These are a summary of attacks detected by signatures on the IIJ Managed IPS Service.

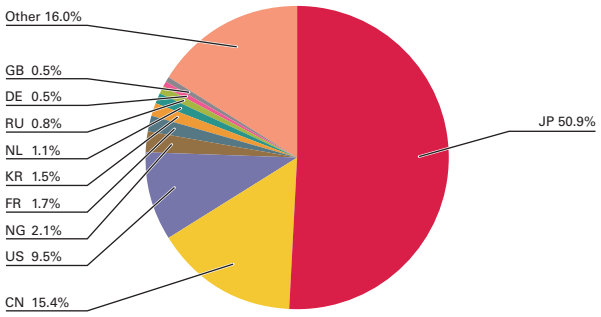


Figure 14: Distribution of SQL Injection Attacks by Source

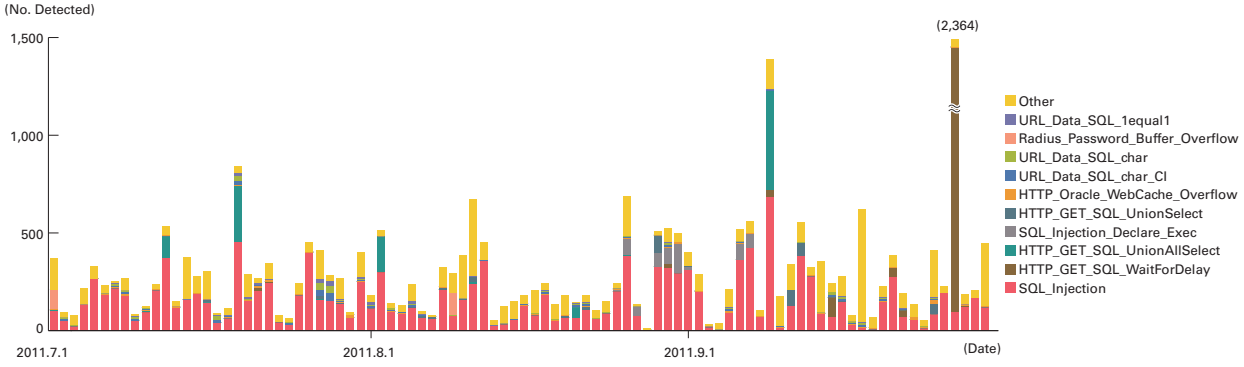


Figure 15: Trends in SQL Injection Attacks (by Day, by Attack Type)

*33 Attacks accessing a Web server to send SQL commands, thereby manipulating an underlying database. Attackers access or alter the database content without proper authorization, and steal sensitive information or rewrite Web content.

Japan was the source for 50.9% of attacks observed, while China and the United States accounted for 15.4% and 9.5%, respectively, with other countries following in order. There was little change from the previous period in the number of SQL injection attacks against Web servers that occurred.

As for communications trends by country, there were attacks from a source in Nigeria on a specific target, and Nigeria also had the 4th highest ratio by country. Because these attacks were brief and took place on a specific day, we believe they were attempts to find vulnerabilities on a specific Web server. The increased number of attacks on September 27 was also from a specific attack source in China against a specific target, and it is likely they had the same purpose. SQL injection attacks also took place in relation to the DDoS attacks that occurred on and around September 18 and drew a lot of attention during this period. Although there was no increase in the actual number of attacks, 75% of the SQL injection attacks on this day were from China.

As previously shown, attacks of various types were properly detected and dealt with in the course of service. However, attack attempts continue, requiring ongoing attention.

1.4 Focused Research

Incidents occurring over the Internet change in type and scope from one minute to the next. Accordingly, IIJ works toward implementing countermeasures by continuing to perform independent surveys and analyses of prevalent incidents. Here, we discuss an Apache vulnerability and its handling, analyze the SpyEye crimeware kit that has come into common use as an attack platform, and examine incidents of the fraudulent issue of public key certificates.

1.4.1 Apache Killer and its Handling

■ About Apache Killer

Apache Killer is a tool for validating an attack concept on the popular Apache HTTPD*34 (henceforth "Apache") Web server software. This tool discloses a technique for launching a DoS attack using a vulnerability*35 that was unknown at the time of its release, but with a simple modification it was possible to exploit it in actual attacks. The tool was posted to a mailing list dealing with vulnerability information on August 20, 2011.

It is possible to launch DoS attacks on Web servers using a variety of methods other than exploiting vulnerabilities, but this incident was serious because, unlike known DoS attacks, it affected all versions of Apache supported at the time of its release, and with the attack tool available to anybody attacks were easy to conduct. The vulnerability was also not dependent on specific modules or settings, so it affected standard configurations. Because the validation tool was released before the vulnerability was reported to the developer, a fixed version was not yet available.

*34 "The Apache HTTP Server Project" (http://httpd.apache.org/).
*35 "Apache HTTPD Security ADVISORY UPDATE 3 - FINAL" (http://httpd.apache.org/security/CVE-2011-3192.txt).

About the Vulnerability and Workarounds

The vulnerability pointed out by Apache Killer originates from processing of the Range: header in an HTTP request. This header is normally used to obtain partial fragments of content. An issue with the fact that it was possible to specify multiple fragments in large numbers and amplify the request for content*36 had already been identified in 2007. With the response to such requests not specified in RFC 2616*37 where the HTTP 1.1 protocol is defined, this affected not only Apache, but also other servers such as Microsoft's IIS. Regarding this protocol, revisions*38 to RFC 2616 have now been proposed.

Meanwhile, although Apache Killer also specifies similar values for the same header, it does not affect other products because the vulnerability originates from the implementation of Apache.

Workarounds for this vulnerability include deleting the Range: header in settings, or denying requests that specify that data be broken into a large number of fragments. The settings that can be changed vary between versions of Apache, but all can limit processing when more than a certain number of fragments are specified. Apache Killer specifies around 1,300 fragments, which has the largest impact on standard configurations. Because software such as browsers and downloaders usually specify a single fragment, this workaround has almost no side effects. Some software, such as older versions of Adobe Systems' Adobe Reader, are known to specify that data be split into multiple fragments, but as long as a realistic number of fragments is set, it is possible to protect against exploit without affecting communications. The higher the number of fragments permitted, the larger the impact of an attack will be, but setting the limit at about 100 fragments neutralizes the risk of the vulnerability, and we believe this value strikes a balance between reducing side effects and providing protection.

Differences from Known DoS Attacks

There are two main types of attacks designed to cause service outages on Web servers. One involves sending a large volume of traffic or requests, and the other involves exploiting vulnerabilities in software or protocols. Apache Killer falls under the latter type, but it affects Web servers differently to previous attack methods of this kind, such as Slowloris*39.

Slowloris attacks only affect HTTPD. Because it has little effect on other processes, administrators can log in to the server, confirm the problem, and resolve it. Meanwhile, Apache Killer consumes memory using a vulnerability that bloats Apache processes, leading to memory running out for not just the corresponding process, but for the entire server. This causes other processes to terminate abnormally or respond extremely sluggishly due to lack of memory. When this happens, administrators cannot even log in to the server, delaying their response.

Timeline of Events

Next, in Table 2 we present the timeline of Apache Killer activity from release to neutralization. First, looking at the number of days that elapsed between release and neutralization, we can see that the tool was dealt with in a short period of time because it was released without contacting the vendor in advance. Workarounds were identified six days after release of the attack tool, and 11 days after the tool was made public a fixed version of Apache (version 2.2.20) was released. However, although this version patched the vulnerability, there were problems with functions that had previously worked correctly. A complete fix (version 2.2.21) was released 25 days after the release of the tool.

Selecting Measures

In response to a vulnerability, developers may sometimes recommend a mandatory upgrade to a fixed version. However, as seen from the incident examined here, this is not always the best option. Because the workarounds proposed for Apache Killer had side effects for a number of clients, they were not a universal solution. However, workarounds were provided much faster than the fixed version, and presented little potential risk, so we can surmise that applying workarounds was the most stable response to take until the release of 2.2.21.

*36 "Vulnerability Summary for CVE-2007-0086" (http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2007-0086).
*37 "Hypertext Transfer Protocol -- HTTP/1.1" (http://www.ietf.org/rfc/rfc2616.txt).
*38 "Add limitations to Range to reduce its use as a denial-of-service tool" (http://trac.tools.ietf.org/wg/httpbis/trac/ticket/311).
*39 "Slowloris HTTP DoS" (http://ha.ckers.org/slowloris/).

On the other hand, it was not possible to confirm what effect fixes would have on operation until the fixed version was actually released, so risk could not be assessed. It is necessary to verify factors such as side effects in advance whether applying a workaround or updating to a fixed version of software, but the fewer changes to software that are involved and the quicker the solution can be applied, the sooner the risk of a vulnerability being exploited can be eliminated.

When operating systems that are open to the Internet, it is important to respond to vulnerabilities before an attack takes place. Software updates are not always the best approach, particularly when a response is needed in a short period of time. As demonstrated in this report, it may be possible to protect a system from vulnerabilities by looking into workarounds via changes to settings, etc., and applying them once their side effects have been assessed. If dealing with a vulnerability causes problems with the primary functions of a system it defeats the purpose, so to continue the stable operation of a system it is necessary to select a low-risk response that has a high probability of working.

Date	Event	Notes	Days Elapsed
08/20	**The Apache Killer attack tool that exploited a new vulnerability was posted to the Full Disclosure mailing list. It was easy to execute, and with no known countermeasure available it could be used for zero-day attacks.**	**Vulnerability made public**	**0**
08/21	Discussions continued on the Full Disclosure mailing list regarding the vulnerability used by the attack tool and the configurations affected. A successful attack on an Apache 2.2 system was reported.		1
08/23	The information available on the Full Disclosure mailing list was posted to the Apache developer mailing list, triggering discussions and work on a fix for the vulnerability.		3
08/24	The individual thought to have disclosed the vulnerability registered a bug in the Apache Bugzilla (bug management system).		4
08/24	An Apache HTTPD developer posted to the Full Disclosure mailing list that a CVE ID (CVE-2011-3192) had been assigned.		4
08/25	**An Apache security advisory was published. The workaround via settings was made public. The advisory stated that all versions released up to that point (1.3, 2.0, and 2.2) were affected, and that a patch or fixed version would be released within 48 hours.**	**Workaround made public (1st)**	**5**
08/25	A large message board with many users was targeted in an attack exploiting this vulnerability.		5
08/26	**Apache security advisory update 2 was published. There were flaws in the workaround via settings, and this portion was amended. The release of the fixed version was postponed, and now stated to be within 24 hours.**	**Workaround made public (2nd)**	**6**
08/27	The deadline announced in Apache security advisory update 2. Code revisions and discussions continued on the developer mailing list and repository. At this stage there were no new announcements.		7
08/30	Fix code was backported to the Apache 2.2.x repository.		10
08/30	A security update was released for the Debian Apache2 package.		10
08/31	Apache 2.2.20 packaging was completed, and developer testing began.		11
08/31	A Debian user reported a bug to the Debian mailing list. Due to lack of information at this stage the source of the problem was not identified.		11
08/31	**Developer testing of Apache 2.2.20 was completed. The release notes and package were made public.**	**Fixed version made public (1st)**	**11**
08/31	Apache developers acknowledged the bug reported on the Debian mailing list, but the details were still unclear at this point.		11
08/31	JPCERT/CC published a security advisory for Japan. It was picked up by various local media.		11
09/01	Red Hat released a fixed package based on the Apache 2.2.20 revisions.		12
09/01	CentOS released a fixed package based on the Red Hat revisions.		12
09/01	The Apache 2.2.20 regression (the new bug found in 2.2.20) was registered in Bugzilla.		12
09/01	Apache developers began discussing measures for resolving the 2.2.20 bug report and regression on the developer mailing list. As a result, all agreed that a 2.2.21 release fixing these issues was necessary, but due to the possibility of other bug reports, the decision was made to delay the release until the middle of the following week. At this stage multiple issues had been discovered in 2.2.20, but no updates were made to the advisory or other information.		12
09/01	A draft of Apache security advisory update 3 was published. This noted that version 1.3 systems were not vulnerable.		12
09/05	The backport of fixed code to the 2.2.x repository began for the release of Apache 2.2.21. The regression and behavior in violation of the RFC in 2.2.20 was fixed.		16
09/10	Apache 2.2.21 packaging was completed, and developer testing began.		21
09/13	Developer testing of Apache 2.2.21 was completed. Syncing of the package to mirror servers began.		24
09/14	**The Apache 2.2.21 release notes and package were made public. Security advisory update 3 was published, officially stating that version 1.3 systems were not vulnerable.**	**Fixed version made public (2nd)**	**25**
09/15	JPCERT/CC updated their security advisory.		26

*Dates are in Japan Standard Time, events on the same day are listed in order of occurrence. Notable events are in bold text.

Table 2: List of Events Relating to Apache Killer

1.4.2 SpyEye

SpyEye is a type of malware known as a crimeware kit. Crimeware kit is the generic name given to frameworks for creating malware that steals personal information such as accounts and passwords (particularly those related to finance) from a computer. ZeuS*40, which made headlines when its source code was recently leaked, is another example of a crimeware kit. Information*41 also suggests that the number of infected users in Japan rose between April and June, 2011. IIJ has independently obtained, studied, and analyzed specimens of SpyEye versions 1.3.10 and 1.3.45. In this section we give an overview of the functions and behavior of SpyEye, and evaluate methods for detecting it.

■ SpyEye Overview

The SpyEye system consists of two main parts. The first is the builder for creating bots that steal accounts and other information once a user's computer has been infected, and the second is the server program for managing infected computers acting as bots as well as the information appropriated from them. Attackers first purchase the framework including these two functions from the creator via underground marketplaces, and then construct the system. Figure 16 shows the process of obtaining information through SpyEye.

Once the system is constructed, the attacker creates a bot using the builder, and then finds a way to install it on the target user's computer. SpyEye bots created through the builder do not have the capability to infect other computers. This means that the attacker must either obtain a separate tool for exploiting vulnerabilities such as Exploitkit*42 and use this in combination with SpyEye, or use other techniques such as social engineering.

When a bot is installed on a user's computer, code injected into other processes by the bot monitors HTTP/HTTPS communications and sends the attacker information such as user accounts that it obtains based on the settings configured when it was built. The attacker can check the information sent on the server's Web UI, send commands to the bot, or update it. As shown in Figure 16, the SpyEye Web UI is divided into a screen for sending commands to bots and a screen for examining the information obtained, so attack specialization is more advanced than ZeuS.

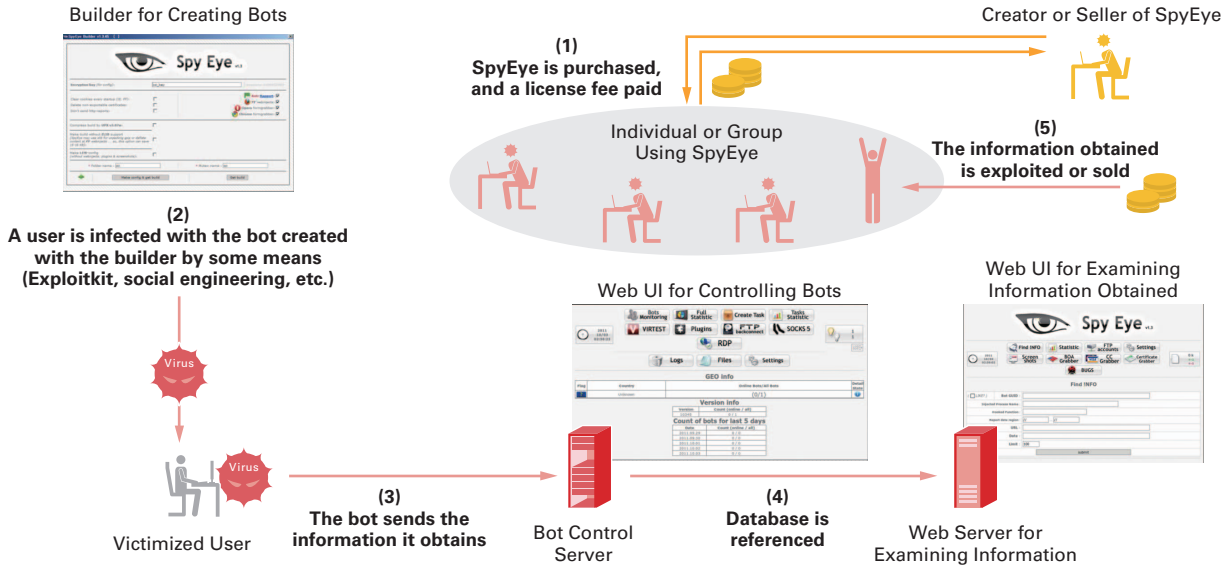


Figure 16: The Process of Obtaining Information with SpyEye

*40 ZeuS is a crimeware kit that surfaced before SpyEye in around 2007. It made headlines when its source code was leaked in May 2011. McAfee Blog, "Zeus Crimeware Toolkit" (http://blogs.mcafee.com/mcafee-labs/zeus-crimeware-toolkit).

*41 The increase in the number of infected users was reported in IBM's Tokyo SOC Report blog and on the IPA website. "An increase in the number of SpyEye viruses detected" (https://www-304.ibm.com/connections/blogs/tokyo-soc/entry/spyeye_20110425?lang=ja) (in Japanese). "An increase in the number of SpyEye viruses detected (continued)" (https://www-304.ibm.com/connections/blogs/tokyo-soc/entry/spyeye_20110817?lang=ja) (in Japanese). "Computer Virus/Unauthorized Computer Access Incident Report - August 2011 -" (http://www.ipa.go.jp/security/english/virus/press/201108/documents/summary1108.pdf).

*42 Expoitkit is also detailed in IIJ Technical WEEK 2010 "Security Trends for 2010 (1) Web Infection Malware Trends" (http://www.iij.ad.jp/development/report/2010/__icsFiles/afieldfile/2011/01/31/techweek_1119_1-3_hiroshi-suzuki.pdf) (in Japanese).

As this demonstrates, even attackers with low technical ability can purchase SpyEye and easily construct a system for obtaining information. Some vendors that offer a pay-per-install (PPI) service to install bots on behalf of others have recently surfaced*43, and we believe using these services lowers the barrier for attackers that steal information.

Next, we will discuss the characteristics of SpyEye bot programs. In addition to a function for simply obtaining information that a user sends over the Internet, SpyEye bots also have a function for capturing screenshots when the mouse is clicked as well as a Web injection function. The function for capturing screenshots is used to steal input made via the software keyboards used on authentication screens for online banking, etc. Web injections are used to steal additional information that the attacker seeks. For example, as shown in Figure 17, it is possible to obtain other information by adding extra input fields on a site that normally only requires input of an ID and password to log in. These alteration settings are described in a file (webinjects.txt) included in bots when they are built. As demonstrated by this example, it is also possible to insert Japanese text.

Attackers can also purchase additional modules known as plug-ins to add a variety of functionality to bots. For example, it is possible to embed plug-ins for functions such as back-connect via FTP, SOCKS or RDP, functions for acquiring credit card details or certificates, functions for launching DDoS attacks, and functions that infect computers via USB devices.

■ Bot Behavior and Characteristics

Like ZeuS, SpyEye bots are designed to operate on Windows XP as well as OSes with User Account Control (UAC)*44 like Windows 7 or Vista without alerting the user. They also make dynamic changes to settings and additional functions possible by storing Web injection settings, bot target URLs, plug-in DLLs, and the configuration file under config.bin*45 in the executable.

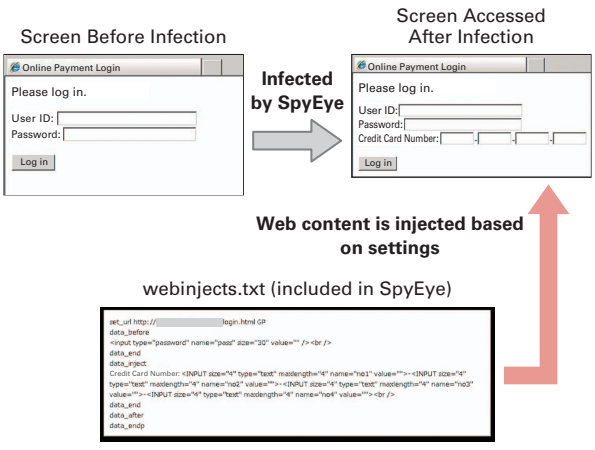


Figure 17: Example of Web Injection Using SpyEye

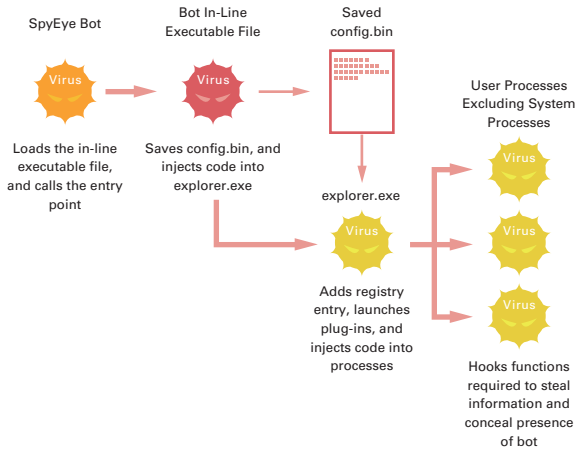


Figure 18: Overview of SpyEye Bot Behavior

*43 Detailed market research and observations of pay-per-install services can be found in the following document. "Measuring Pay-per-Install: The Commoditization of Malware Distribution" (http://usenix.org/events/sec11/tech/full_papers/Caballero.pdf).

*44 UAC is a technological feature of Windows that notifies the user when a program makes changes that require administrator level access privileges, enabling them to retain control of the computer. Microsoft, "What is User Account Control?" (http://windows.microsoft.com/en-US/windows7/What-is-User-Account-Control).

*45 config.bin is a binary file consisting of a password-protected zip encoded with XOR using an immediate value. SpyEye saves the zip password using an environment variable to make it possible to reference the binary file from a variety of processes. In 1.3.10 the file name was fixed as config.bin, but in 1.3.45 the file name is a numeric value generated from a text string such as the folder specified during building or the OS version obtained.

An overview of bot behavior is given above (Figure 18). The bot first loads the in-line executable file, then calls the entry point*46. Config.bin is then saved to the file, and code is injected into explorer.exe. The injected code adds the bot to the registry so it will be executed whenever the system is booted. It also launches plug-ins included in config.bin, and injects code into processes other than selected system processes to hook the functions necessary to thieve information and conceal the presence of the bot.

The creator of SpyEye has designed it to be easy to add or change code. Specifically, it obtains and uses config.bin and code for injections from custom resources. Additionally, as previously mentioned, the dynamic loading and internal execution of the in-line executable file is believed to be designed to separate code so behavior upon installation when initially executed and behavior for the intended purpose can be changed separately and flexibly.

A number of code obfuscation methods are used with SpyEye, but the most distinctive is the calling of a desired library function by specifying a numeric value generated from its name in advance*47. This technique is typically used in shell code. To pinpoint the function being called, analysts must either execute the code in a debugging environment, or analyze the algorithm for generating the numeric value.

■ Evaluation of Detection Methods

The two main methods for detecting computers infected with SpyEye involve either detection via communications or detection on the host. Here we evaluate detection via communications.

SpyEye bots communicate with two main types of server. The first is a program called a collector, and the second is the server's Web UI*48. Communications with the collector takes place when a bot is launched or when stolen information is sent, etc. For example, when a user sends data to a Web server, this data is sent to the collector along with the bot ID and process name, as well as the hooked function. Data sent to the collector is compressed and specially encoded using XOR*49, making it difficult to detect in real time using IDS/IPS, etc.

On the other hand, because communications with Web UI involve frequently sending the bot ID, computer name, and information about the plug-ins supported by that bot using GET parameters in 1.3.10, and base64-encoded POST parameters in 1.3.45, detection via this data should be possible.

Because SpyEye bots operate in user mode, methods of detection include use of a rootkit detection tool such as GMER*50 on the host to check the files that are installed and the registry values that have been added*51.

■ Summary

SpyEye is still under active development. SpyEye versions 1.3.10 and 1.3.45 that were analyzed in this report were released fairly close together, but functions such as code obfuscation and injection behavior have already advanced, so we must continue to monitor its development carefully. SpyEye infections lead to significant issues such as information leaks and financial damages, so preventing and detecting infections is crucial. To prevent infection via drive-by-download attacks using tools such as Exploitkit, it is important to keep software up to date by patching the OS, browser, and browser plug-ins as soon as possible and to install and update anti-virus software. Infection through social engineering via email or other methods is also a possibility, so care must be taken not to blindly open links or attachments.

*46 Upon initial execution computer information is collected, then code is injected into explorer.exe, and finally the bot is copied to the folder specified at build time and installed.

*47 In 1.3.45 numeric values are also assigned to important text strings such as the process names targeted for injection, the names of the executable after installation and config.bin, and plug-in export function names.

*48 A separate plug-in called customconnector is required to communicate with the server's Web UI.

*49 The compression algorithm has not been analyzed in detail, but it is presumed to be a custom one. Additionally, XOR does not use an immediate value as a key.

*50 "GMER-Rootkit Detector and Remover" (http://www.gmer.net/).

*51 However, the folders and file names created and registry values differ for each computer.

1.4.3 Incidents of the Fraudulent Issue of Public Key Certificates

■ Incident Overview and Background

Since the start of this year there have been a string of incidents in which certificate authorities have been hacked and certificates fraudulently issued. In the Comodo incident in March nine certificates were fraudulently issued, and in the DigiNotar incident in August over 500 were fraudulently issued. Both of these were the work of ComodoHacker, who when claiming responsibility for the latter incident announced that he was also able to issue certificates at four other certificate authorities, including StartCom and GlobalSign. In response to this GlobalSign temporarily suspended the issue of certificates to verify the situation. These incidents have brought the reliability of the certificate authority and public key infrastructure (PKI) system into question. In this section we examine the impact of the fraudulent issue of public key certificates and look at countermeasures.

■ The Public Key Certificate System

Public key certificates are data used to certify entities such as individuals or servers over the Internet by a third party. The accuracy of the binding between the public key included in a certificate and an entity is assured through a cryptographic digital signature using public key cryptography such as RSA or ECDSA. Certificates are issued by trusted certificate authorities (CA). Because certificates can be issued hierarchically, the certificates for end entities are sometimes issued through multiple intermediate CAs. Issue of a certificate from issuer to end entity indicates the direction of trust, and accuracy is assured through the system of going back to the self-signed (root) certificate to find the trust anchor via certificate validation in the reverse direction*52.

The public key certificates normally seen by Internet users are server certificates accessed via SSL/TLS protocol, and most are based on X.509 specifications*53. A Fully Qualified Domain Name (FQDN) that specifies the location of the server is described in the certificate, making it possible to confirm whether this matches the URL a user is accessing via a browser. Safe communications are achieved by trusting server certificates issued by a trusted CA. However, trusted anchor settings are normally carried out by the OS or applications such as browsers under the assumption that users trust the reliability of the preinstalled set of CA certificates. Ideally, users would construct their own trust anchor from the preinstalled set of certificates, but currently there are very few users that manipulate this data.

In the recent incidents server authentication could not be carried out correctly because users blindly accepted the trust anchor specified by the vendor. The problem here is that users are made to believe they are conducting legitimate, safe communications when they communicate with a third party (the attacker) instead of the intended server. Because accuracy is not ensured during Web server authentication, even if subsequent confidentiality and data-integrity is achieved, communications are not safe because they involve establishing a secure channel with an attack server.

■ The Impact of the Fraudulent Issue of Certificates

Next, we consider the impact of issuing fraudulent certificates to holders of legitimate FQDN like this. For example, let us imagine a case where an attacker sends a request to a hacked CA to issue a certificate for example.co.jp, obtaining a fraudulently issued certificate that includes a certain FQDN under example.co.jp. When this happens, because the attacker uses their own key to issue a fraudulent certificate, they can conduct legitimate server authentication for this FQDN via SSL/TLS. However, if they cannot place an attack server with the corresponding FQDN under the example.co.jp domain, the previously mentioned browser FQDN check prevents certificates from being accepted (because the URL of the server being accessed and the FQDN in the certificate are different). This limits the potential for a successful attack.

*52 Details of the PKI system can be found in the following paper. R. Perlman, "An Overview of PKI trust models", IEEE Networks vol.13, 38-43 (1999). (http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=806987).

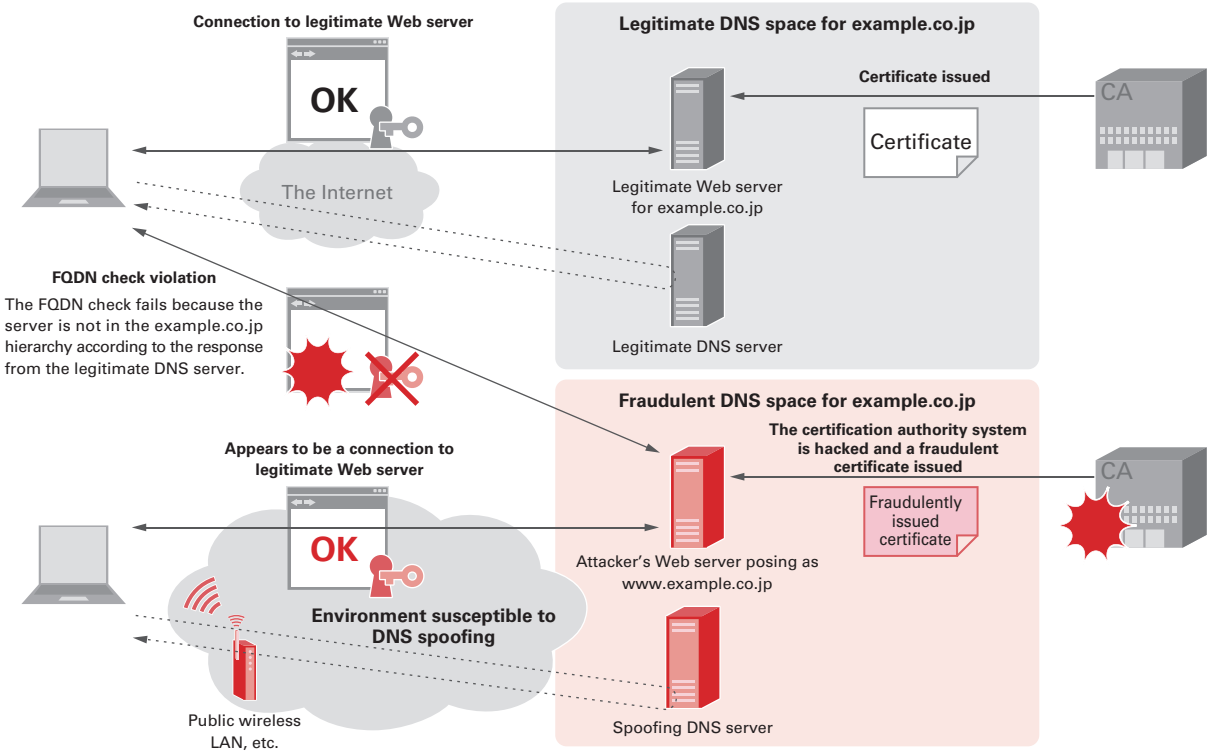
*53 ITU-T Recommendation X.509 (08/05) ISO/IEC 9594-8:2005, Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks. 2005.

Figure 19 shows an example of a successful attack using DNS spoofing to rewrite DNS information when using a public wireless LAN. The method in this example could be used to redirect users to an attack server by making the FQDN of the example.co.jp domain appear as the IP address of a server the attacker controls. This would also pass a browser's FQDN identification check when the corresponding site is accessed via browser using the SSL/TLS protocol. In addition to DNS spoofing, a similar attack could also be launched if it is possible to deploy a Man-In-The-Middle (MITM) attack server via route hijacking.

In the DigiNotar incident, communications via the Online Certificate Status Protocol (OCSP) protocol for verifying certificates online were observed targeting fraudulently issued certificates for google.com from approximately 300,000 addresses*54. This is evidence that fraudulently issued certificates were used via browsers. Because some browsers do not implement OCSP or are configured to disable OCSP communications, we believe that fraudulently issued certificates were actually received on an even higher number of PCs.

■ Countermeasures

Public key certificates have an expiration date set to avoid impact from the compromise of cryptographic algorithms if the same public key continues to be used and to support the PKI business model, and a system to revoke certificates is in place. With incidents such as these in which certificates have been fraudulently issued, it is theoretically feasible to respond by having certificate authorities revoke fraudulently issued certificates to block fraudulent use. However, families of products without an adequate function for revoking certificates that were initially issued as legitimate will need to take significant measures. With embedded products that are not updated frequently in particular, care should be taken if the confirmation of certificate validity is simplified or omitted.



Even if a certificate authority system is hacked and a fraudulent certificate issued for www.example.co.jp, an attack is not possible unless a server is actually placed under the target domain or the DNS is also manipulated by some means. This figure is an example of circumventing the FQDN check using DNS spoofing.

Figure 19: Fraudulent Issue of a Certificate by Hacking a Certificate Authority System

*54 FOX-IT Interim Report, v1.0, "DigiNotar Certificate Authority breach, September 5, 2011" (http://www.rijksoverheid.nl/documenten-en-publicaties/rapporten/2011/09/05/diginotar-public-report-version-1.html).

There are also moves to implement more fundamental countermeasures, such as making improvements in conjunction with the current PKI, or replacing the current CA system altogether. These include proposals such as DANE*55, which has mechanisms for sending certificates and confirming certificate and server accuracy using DNSSEC, and Convergence*56, which ensures accuracy by using multiple notary agencies to improve reliability. It is expected to take a significant amount of time for these technologies to become widespread, but with the reliability of the PKI system itself in question due to the recent incidents, it is possible that public opinion will shift towards making a swifter transition.

1.5 Conclusion

This report has provided a summary of security incidents to which IIJ has responded. In this report we discussed the behavior and response to the Apache Killer tool for detecting an undisclosed vulnerability that was discovered in August, analyzed the SpyEye crimeware kit that has caused damages worldwide, and examined incidents of the fraudulent issue of public key certificates and their impact. By identifying and publicizing incidents and associated responses in reports such as this, IIJ will continue to inform the public about the dangers of Internet usage, providing the necessary countermeasures to allow the safe and secure use of the Internet.

Authors:
Mamoru Saito
Manager of the Office of Emergency Response and Clearinghouse for Security Information, IIJ Service Division. After working in security services development for enterprise customers, Mr. Saito became the representative of the IIJ Group emergency response team, IIJ-SECT in 2001, participating in FIRST, an international group of CSIRTs. Mr. Saito serves as a steering committee member of several industry groups, including Telecom-ISAC Japan, Nippon CSIRT Association, Information Security Operation providers Group Japan, and others.

Hirohide Tsuchiya (1.2 Incident Summary)
Hirohide Tsuchiya, **Hiroshi Suzuki**, **Tadaaki Nagao** (1.3 Incident Survey)
Tadashi Kobayashi (1.4.1 Apache Killer and its Handling)
Takahiro Haruyama (1.4.2 SpyEye)
Yuji Suga (1.4.3 Incidents of the Unauthorized Issue of Public Key Certificates)
Office of Emergency Response and Clearinghouse for Security Information, IIJ Service Division

Contributors:
Masahiko Kato, **Masafumi Negishi**, **Yasunari Momoi**, **Hiroaki Yoshikawa**, **Seigo Saito**, Office of Emergency Response and Clearinghouse for Security Information, IIJ Service Division

*55 IETF DNS-based Authentication of Named Entities (DANE) Working Group (https://datatracker.ietf.org/wg/dane/).
*56 Convergence (http://convergence.io/details.html).