FY 2008

Cyber Clean Center (CCC) Activity Report

Anti-Bot Measures Project
Cyber Clean Center
https://www.ccc.go.jp

Table of Contents

1 Purpose of This Project

While usage of the Internet has widely spread to the general public, damage caused by “malware” is increasing. As such malware spreads its infection activities across the Internet, Internet users will be left in a dangerous situation unless countermeasures are taken. Malware is the generic term for various pieces of malicious software, generally classified into viruses, Trojan horses, spyware, BOTs, and so forth. BOTs can be used to form a network called a “botnet”. Once a PC is infected with a BOT, the PC will become a part of the network and can be remotely controlled by a “Herder” and used for various cyber crimes, such as DDoS attacks, spam mails, and phishing, unbeknownst to the owner of the infected PC. The tactics used by BOTs to spread infections are becoming increasingly sophisticated. While old viruses used to engage in merely amusing activities, such as displaying fireworks in PC windows, or worse, deleting files on a hard disk after a PC is infected, BOTs are characterized by the fact that they secretly engage in infection activities that users remain unaware of. As BOTs have been designed with various techniques, such as having many varied subspecies so that they may not be detected and removed by antivirus software, or preventing antivirus software from being updated on a PC after infection, it is becoming more difficult for users themselves to take countermeasures against them. For this reason, it is important that the government takes the initiative to promote BOT-related countermeasures in cooperation with ISPs, security vendors, and other network security organizations, not leaving BOT countermeasures solely in the hands of the users themselves.

The “Cyber Clean Center” (hereafter, “CCC”), was launched in FY 2006 against this backdrop as an approach to minimize BOT infections in Japan in the form of a joint project with MIC and METI. Since then, CCC has been promoting BOT countermeasures with alert activities in cooperation with ISPs.

This document presents the FY 2008 Activity Report covering the activities of the three groups running CCC: the BOT Countermeasure System Operation Group, BOT Program Analysis Group, and BOT Infection Prevention Promotion Group.

1.1 Current Status of Botnets

A botnet typically consists of hundreds, thousands or even millions of PCs infected by BOTs and formed into a large network controlled by a commander, called a “Herder”, often via a C&C (Command & Control) server. BOT-infected PCs are manipulated with commands issued from the Herder and pose a great threat to the safety of Internet users because they are misused for cyber crimes, such as sending large volumes of spam mail for phishing, advertising and the like, as well as DDoS attacks on specific sites. In this way, users of BOT-infected PCs are “victims” and at the same time “victimizers” without being aware that they are being used as a stepping-stone to cyber crimes.

BOTs had already been identified by 2002 (Trend Micro released information on AGOBOT to the public in the second-half of 2002) and infection incidents by BOTs have grown more noticeable since 2004. A survey conducted in June 2005 by Telecom-ISAC Japan and JPCERT/CC estimated approximately 0.4 to 0.5 million infections among approximately 20 million broadband users* in Japan (infection rate of approximately –2.5%). The survey conducted by CCC on the number of domestic BOT infections in June 2008 estimated approximately 0.3 million infections among approximately 30 million broadband users* in Japan (infection rate of approximately 1%).

(*Information source: MIC statistical data “Changes in the number of contracts for

broadband services, etc.")
As for the reasons why the BOT infection rate fell, BOT countermeasures by CCC may have contributed to it, while environmental factors such as the wider adoption of antivirus software, transition to securer OSs, and the introduction of broadband routers could all be possible factors.
The fact that the domestic BOT infection rate is very low compared to those in other countries is reported in "Distribution Map for the Number of PCs Infected by Malware per 1000 PCs" from Microsoft. We can infer that the activities of BOT countermeasures by CCC may have helped the situation.

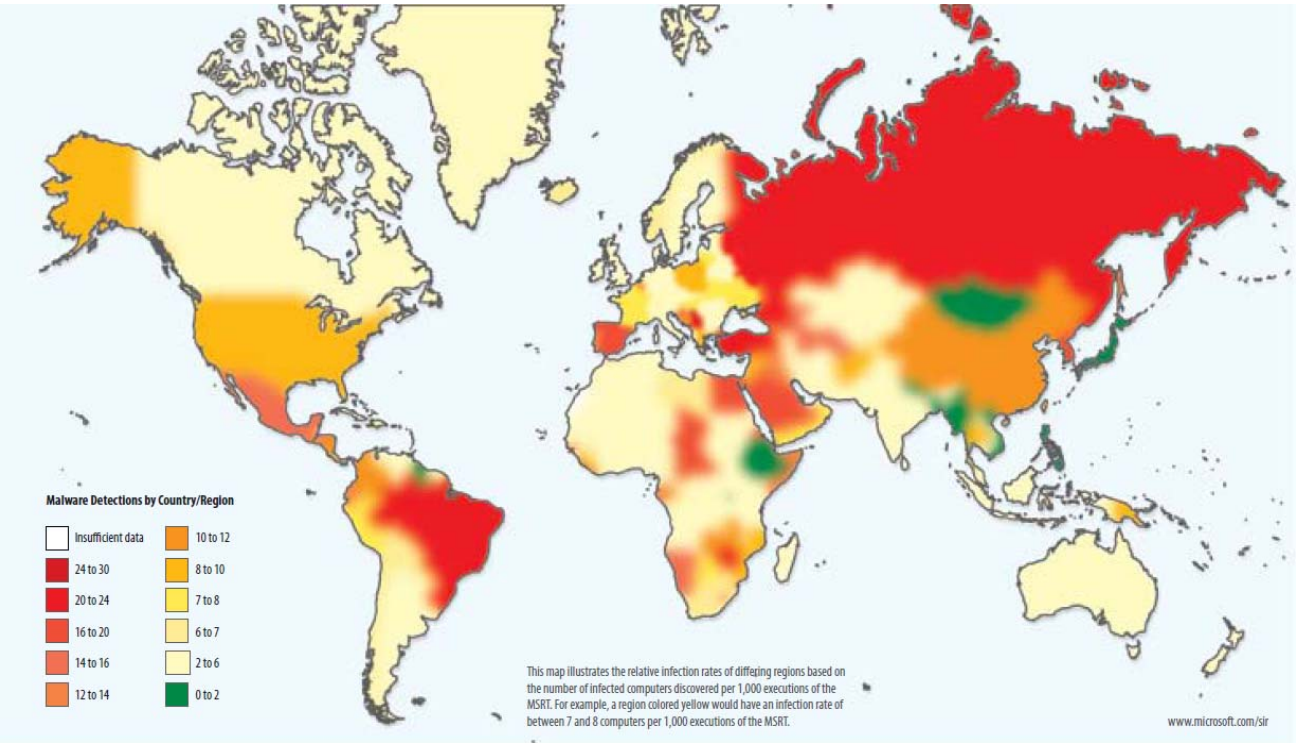

Figure 1.1-1： Distribution Map of the Number of PCs Infected by Malware per 1000 PCs

(Information source: http://www.microsoft.com/japan/security/contents/sir.mspx)

1.2 Overview of the Cyber Clean Center (CCC)

CCC issues alerts for BOT-infected users through the activities illustrated in Figure 1.2-1 in cooperation with Project Participating ISPs and others, such as security vendors, to minimize BOT infections in Japan.

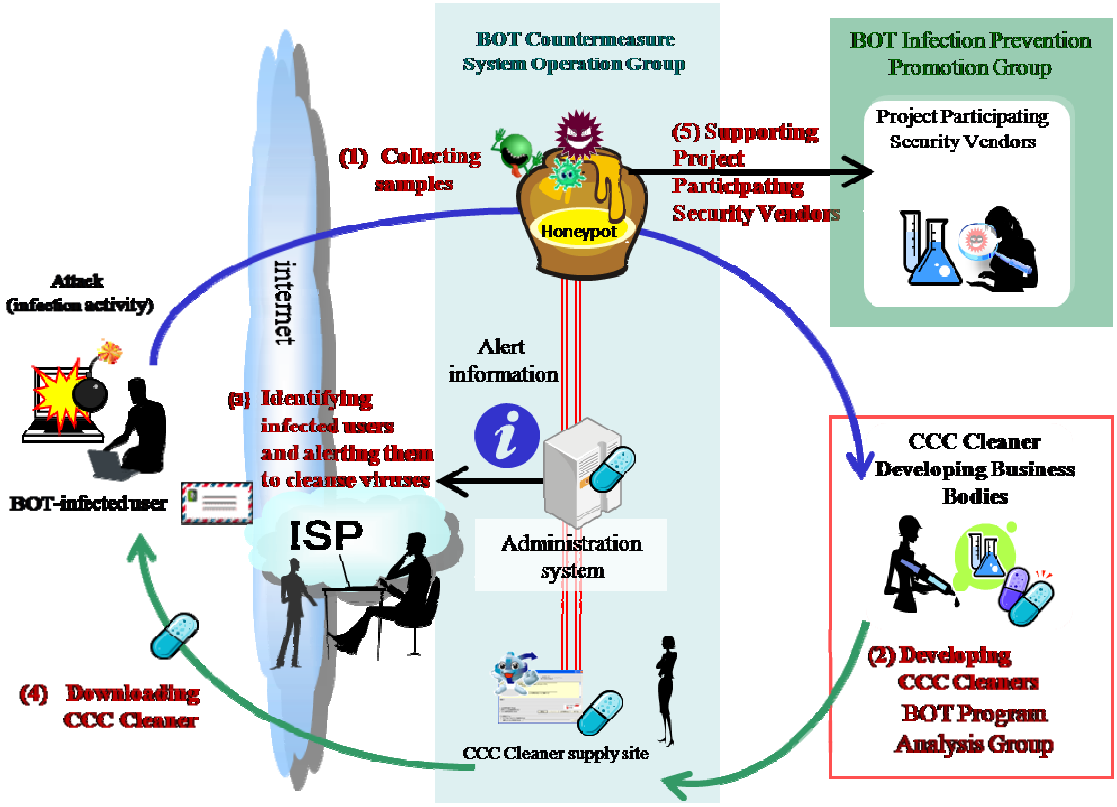

Figure 1.2-1: Overview of CCC Activities

1. Collecting samples	Collect BOT samples by detecting attack events (infection activities) from BOT-infected PCs using “decoy PCs” (honeypots). (BOT Countermeasure System Operation Group)
2. Developing CCC Cleaners	Analyze BOT samples and develop “CCC Cleaner.” (BOT Program Analysis Group)
3. Identifying infected users and instructing them to cleanse viruses	Identify the source from which BOT attacks are executed in cooperation with ISPs and send a BOT cleaning alert mail to the source. (BOT Countermeasure System Operation Group)
4. Downloading a CCC Cleaner	The BOT-infected user who received the cleaning alert mail accesses the CCC Cleaner site specified in the mail and downloads the CCC Cleaner application. (BOT Countermeasure System Operation Group)
5. Supporting Project Participating Security Vendors	Provide BOT samples to Project Participating Security Vendors, the respective vendors reflecting the sample in the pattern files of their antivirus software. (BOT Infection Prevention Promotion Group)

CCC comprises of three groups based on the nature of the work involved and performs its tasks under the Cyber Clean Center Steering Committee (CCC-SC).

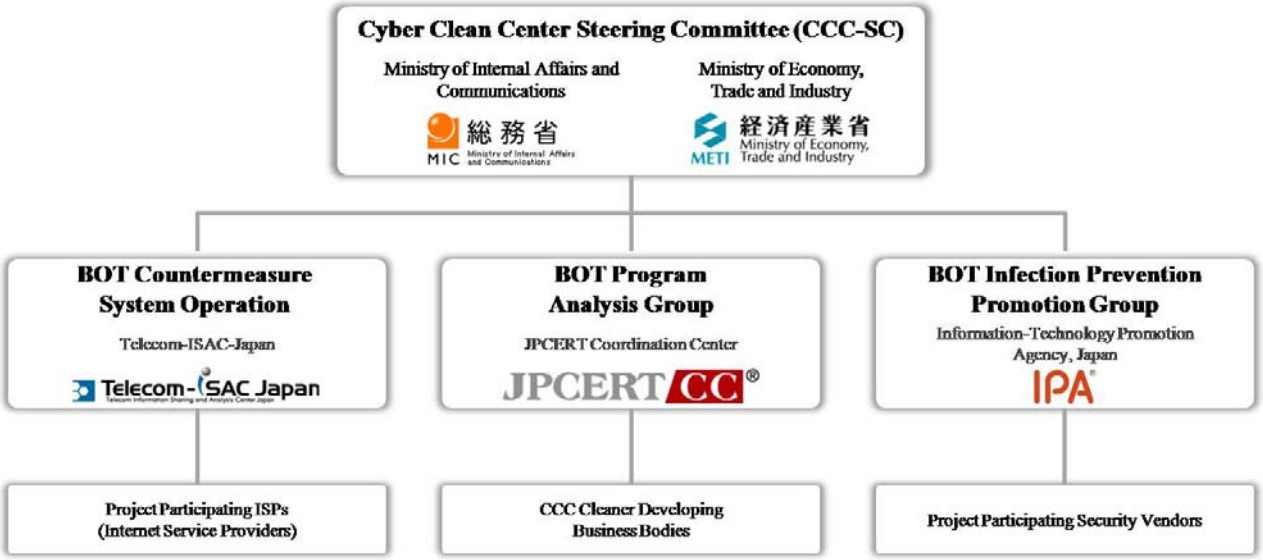

Figure 1.2-2: Organization Chart of CCC Operations

BOT Countermeasure System Operation Group (Telecom-ISAC Japan)

The BOT Countermeasure System Operation Group operates the backbone system of this project, including the sample collection and analysis systems such as honeypots and the alert activity system, collects and analyzes BOTs and alerts BOT-infected users to disinfect BOT through the Project Participating ISPs, and other actions to counteract BOTs. This group passes the collected BOT samples to the BOT Program Analysis Group to reflect them in pattern files for CCC Cleaners. At the same time, it passes them to the major Project Participating Security Vendors in Japan through the BOT Infection Prevention Promotion Group to contribute to reflecting them in the pattern files of antivirus software products.

The group investigates the latest trends in malware in cooperation with security vendors, etc. to address new threats from BOTs and take effective action against them.

Project participating ISPs, as of the end of March 2009

Internet Initiative Japan Inc. (IIJ), NEC BIGLOBE, Ltd. (BIGLOBE), NTT Communications Corporation (OCN), KDDI CORPORATION (au one net), NIFTY Corporation (@nifty), hi-ho Inc. (hi-ho), SOFTBANK TELECOM Corp. (ODN), SOFTBANK BB Corp. (Yahoo! BB), IMS Corporation (Internet MAGMA), IC-NET Co., Ltd. (IC-NET), iTEC Hankyu Hanshin Co., Ltd. (GAONET, tigers-net.com, BaycomNet, bai Service), ASAHI Net, Inc. (ASAHI Net), Technology Networks Inc. (@NetHome), INTERLINK Inc. (ZOOT), ipc-Tokai Co., Ltd. (ipc Tokai Internet Service), VECTANT Ltd. (VECTANT), STNet, Incorporated (PIKARA, NETWAVE), NTT-ME Corporation (WAKWAK), NTT DATA SANYO Corporation (SANNET), NTT DOCOMO INC. (mopera/mopera U), NTTPC Communications, Inc. (InfoSphere), NTT Media Supply Inc. (DoCANVAS, Pokke, BB-WEST, Suruga, Wellith, SUISUI, MAST-BB), NDS Corporation (TikiTiki Internet), Energia Communications, Inc. (MEGA EGG, Urban Internet), LCV Corporation (LCV-Net), Kawaguchiko cable television Inc. (LCNet),

KANSAI MULTIMEDIA SERVICE COMPANY (ZAQ), KATCH NETWORK INC. (KATCH Cable Internet Service), Kintetsu Cable Network Co., Ltd. (KCN-Net), Good Communications Co., Ltd. (SYNAPSE), KUMAMOTO CABLE NETWORK CORPORATION (JCN Kumamoto) (KCN Internet Service), Gunma Internet Co., Ltd. (Gunma Internet), KMN Corporation (ROSENET, MediaCat), K-Opticom Corporation (eo), Kintetsu Cable Network Kyoto Corporation（KCN Kyoto Internet), KIP Co., Ltd. (KIP-Internet), Cable TV Yamagata Co., Ltd. (CATVY Internet), Cable One Corporation (Cable Internet), TOKAI CORPORATION (TOKAI Network Club), Sunrise Systems Corporation（RYOMO Internet), JWAY Co., LTD. (Cable Internet), Shonan Cable Network Co., Ltd. (SCN Network Service), Shiratsuyu Company Corporation (DAC System), SENDAI CATV CO., LTD. (CAT-V NET), TAKAOKA CABLE NETWORK CO., LTD. (Takaoka Cable Network Internet Connection Service), CHUBU TELECOMMUNICATIONS CO., INC. (Commuf@), TSUNAGU NETWORK COMMUNICATIONS INC. (e-mansion), TAM InternetService CO., LTD. (TAM Internet Service, Net3 Internet), DEODEO Corporation (DEODEO Enjoy Net), TelecomWAKAYAMA, Inc. (aikis), Densan Co., Ltd. (avis), Tokyo Cable Network, Inc. (TCN Cable NET), TONAMI Transportation Co., Ltd. (CORALNET), Tonami Satellite Communications Television Inc. (TSTnet), DREAM TRAIN INTERNET INC. (DTI, Cilas.net, BroadStar, isao.net), Nagasaki Cable Media Co., Ltd. (NCM Cable Internet Service), Nagano Kyodo Densan Co. Ltd. (JANIS), Global Network Core Co., Ltd. (N-plus), NOETSU CABLENET Inc.(NOETSU Net), ParkNet Corporation (ParkNet), Hanno Cable Television Co., Ltd. (@hanno), Himawari Network Co., Ltd. (Aitainet), FAMILYNET・JAPAN CORPORATION (CYBERHOME), VR Tecno Center, Inc. (VRTC Net), FUJITSU SOFTWARE TECHNOLOGIES LIMITED (Web Shizuoka), FUJITSU NAGANO SYSTEMS ENGINEERING LIMITED (Infovalley), Fusion Network Services Corporation (FUSION GOL), NTT Plala Inc. (Plala), Fureai Channel Inc. (Ai Net), Mie Data Tsusin Corporation (Mie Internet Service), Micsnetwork Corporation (mics Internet), Mirai Communication Network Inc. (Mirai Net), Mediatti Communications, Inc. (Mediatti NET), Yamaguchi Cable Vision Co., Ltd. (C-able Internet)

BOT Program Analysis Group (JPCERT Coordination Center)

The BOT Program Analysis Group analyzes the features and techniques used by the BOT samples collected by the BOT Countermeasure System Operation Group, developing CCC Cleaners in cooperation with the CCC Cleaner software developers. This group also conducts studies on effective analysis systems and develops countermeasure techniques in cooperation with CCC Cleaner developers.

CCC Cleaner Developing Business Bodies
Trend Micro Incorporated

BOT Infection Prevention Promotion Group (Information-Technology Promotion Agency, Japan)

The BOT Infection Prevention Promotion Group promotes the prevention of BOT infections by taking final custody of the BOT samples collected through CCC and by providing samples to the Project Participating Security Vendors in an appropriate manner for incorporation into the pattern files of vendors' antivirus software.

Project Participating Security Vendors

AhnLab Incorporated, Kaspersky Labs Japan Limited, Symantec Corporation, Sourcenext Corporation, Trend Micro Incorporated, Microsoft Corporation, McAfee Incorporated

2 Activity Report – BOT Countermeasure System Operation Group

2.1 Overview

The BOT Countermeasure System Operation Group collects and analyzes BOT samples as well as promoting alert activities with the aim of minimizing BOT infections in Japan.
In the sample collection and analysis phase, the group detects attacks (BOT infection activities) from BOT-infected users and collects BOT samples. It performs “known” and “unknown” isolation analysis on the collected samples in addition to dynamic analysis to observe the actual operation of the samples, along with determining antivirus software detection rates. The collected BOT samples are then passed to the BOT Program Analysis Group, which in turn develops CCC Cleaners.
In the alert promotion phase, the Group identifies the ISP(s) used by BOT-infected users based on attack events from them and passes relevant information to the appropriate ISP(s). In turn, the ISPs identify users based on this information and sends alert emails to them, informing them that their PCs are infected by a BOT. The BOT-infected users may follow the instructions included in the alert email, access the CCC Cleaner web site operated by the BOT Countermeasure System Operation Group, and conduct appropriate BOT countermeasures. From the CCC Cleaner supply site, users can download CCC Cleaners that have been developed by the BOT Program Analysis Group, and at the same time they can obtain various information necessary for BOT countermeasures, such as about Windows Update, installing antivirus software, and usage of broadband routers. The group provides such BOT countermeasure information to BOT-infected users who received alert emails, as well as to general users through the official CCC site.

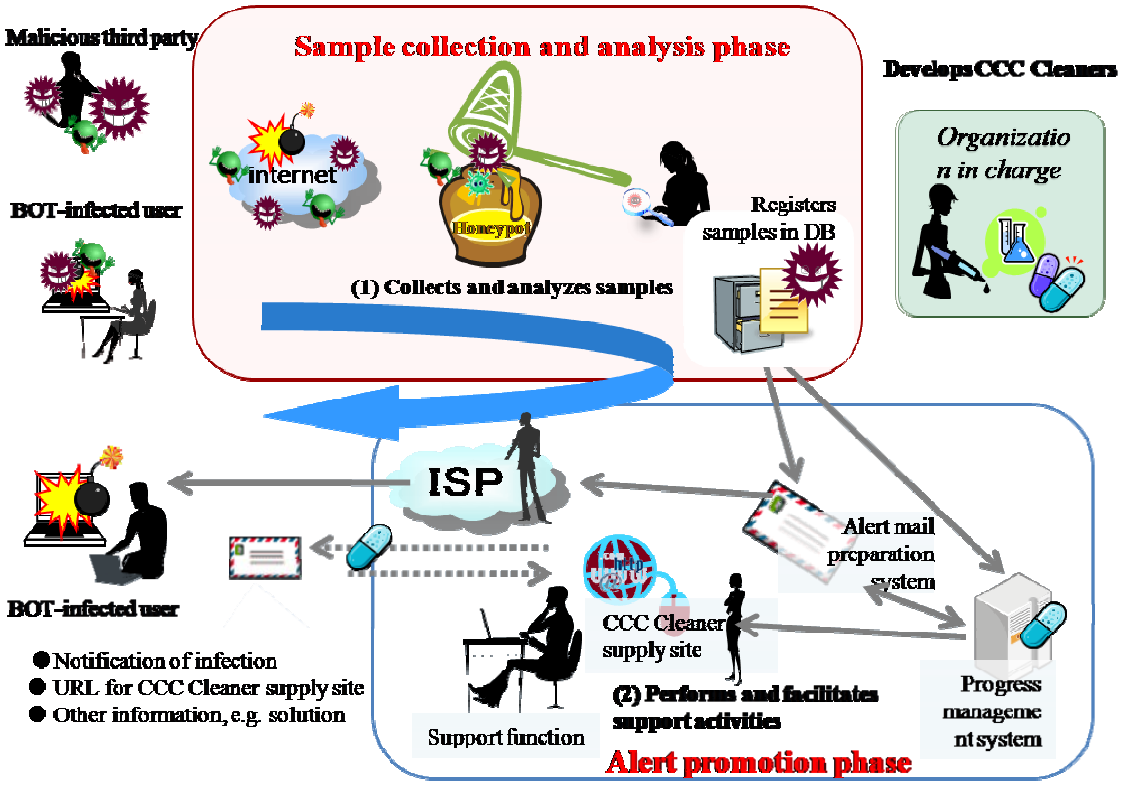

Figure 2.1-1: Overall View of BOT Countermeasure System Operation Group

2.1.1 Sample collection and analysis

It is important to detect attacking events (BOT infection activities) coming from BOT-infected users, and collect and analyze the BOT samples as the first step in the BOT countermeasures in CCC. For this reason, the BOT Countermeasure System Operation Group engages in collecting BOT samples by using “honeypots”—decoy systems on which OS vulnerabilities are deliberately left open.
The collected BOT samples include many duplicates and those that have already been regarded as being able to be addressed by antivirus software. In light of this, the group first extracts unique samples from the duplicates (identification analysis). Next, it confirms whether the identified, unique samples can be addressed with antivirus software—using the latest pattern Trend Micro antivirus software uses at the time of virus scanning, (known and unknown isolation analysis). Finally, it confirms whether the samples regarded as unidentified are actually BOTs by executing them (dynamic analysis). Those that are confirmed to be BOTs are assigned to the BOT Program Analysis Group so CCC Cleaners can be developed.

2.1.2 Alert activities

The BOT Countermeasure System Operation Group alerts BOT-infected users of the possibility that they might be infected with a BOT virus, based on attacking events (attack source IP address and time information) detected by the honeypots at the same time that the group collects and analyzes BOT samples as described in *2.1.1 Sample collection and analysis.* When alerting users to the possibility of BOT infection, the group identifies the ISPs used by BOT-infected users from the attack source’s IP addresses, passes the relevant attack event information to each ISP, and asks the ISP to alert the BOT-infected users. The ISP identifies the BOT-infected users based on the information passed from the BOT Countermeasure System Operation Group and sends alert emails to them. The BOT Countermeasure System Operation Group prepares a page in the countermeasures site for each BOT-infected user, and provides the infected users who received alert emails with information essential to carry out BOT countermeasures, such as the delivery of CCC Cleaners, Windows Update, installation of antivirus software, and usage of broadband routers. Preparing a page in the countermeasures site for each BOT-infected user enables the group to keep track of how much progress the user makes in carrying out the countermeasures and provide the user with fine-tuned support by ISPs.

2.2 Status of Activities and Results

The CCC project publicly discloses the results of alert activities on a monthly basis through the official CCC site (https://www.ccc.go.jp/) (Figure 2.2-1）.

As of March 2009, the project has collected a total of 13,534,588 samples, and identified unique samples of 870,277 types. Among them, 22,871 samples were identified as unable to be detected with commercially available antivirus software at the time of collection. Regarding alert activities, 373,207 emails were sent to 79,050 persons and about 30 percent of infected users downloaded CCC Cleaners to take active measures against the BOTs.

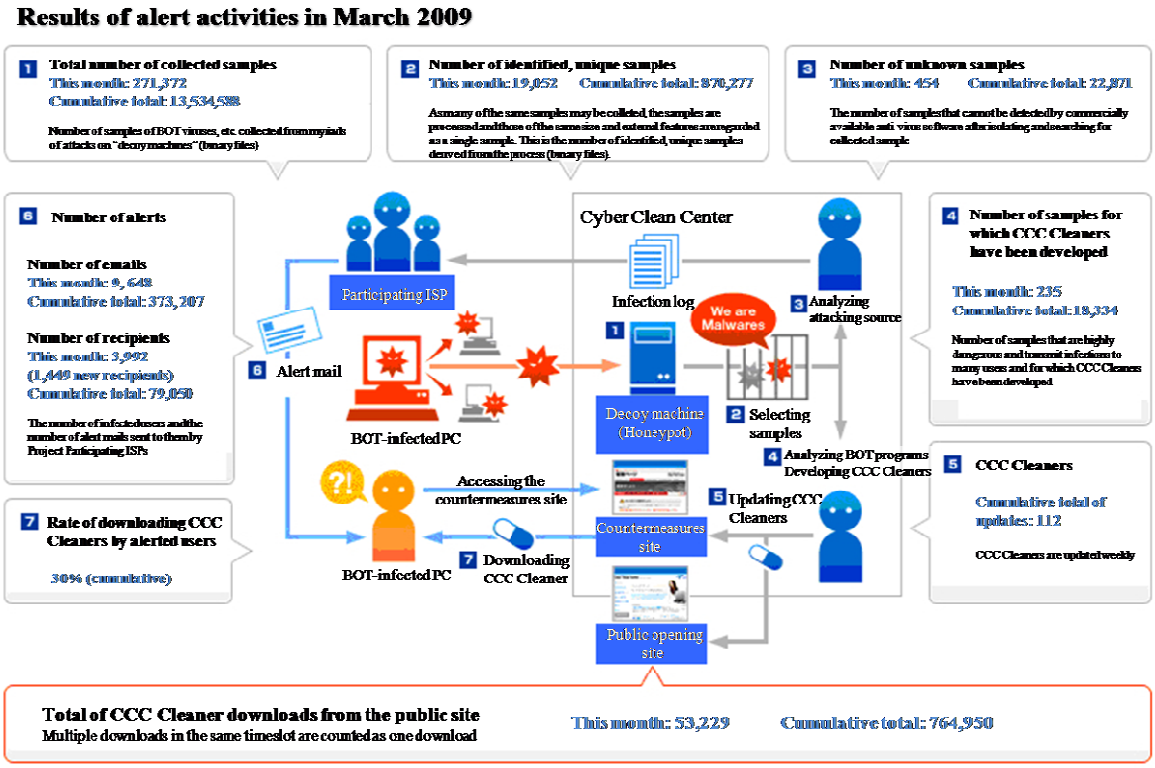

Figure 2.2-1: Activity Result – Actual Result in March 2009 and Cumulative Total of Actual Results from February 2007 to March 2009

2.2.1 Sample collection and analysis

To identify BOT-affected users, it is necessary to capture infection and attacking activity from BOTs and collect and analyze the BOT samples to develop CCC Cleaners.
This section shows the status of sample collection and analysis activities from April 2008 to March 2009.

(1) Changes in number of collected samples

The sample collection and analysis system lures BOTs into a system known as a honeypot and collects them as BOT samples.
The average number of collected samples per month in FY 2008 was approximately 490,000 samples. The duplicate and known samples were counted in the number. Figure 2.2-2 shows changes in the number of collected samples on a monthly basis.

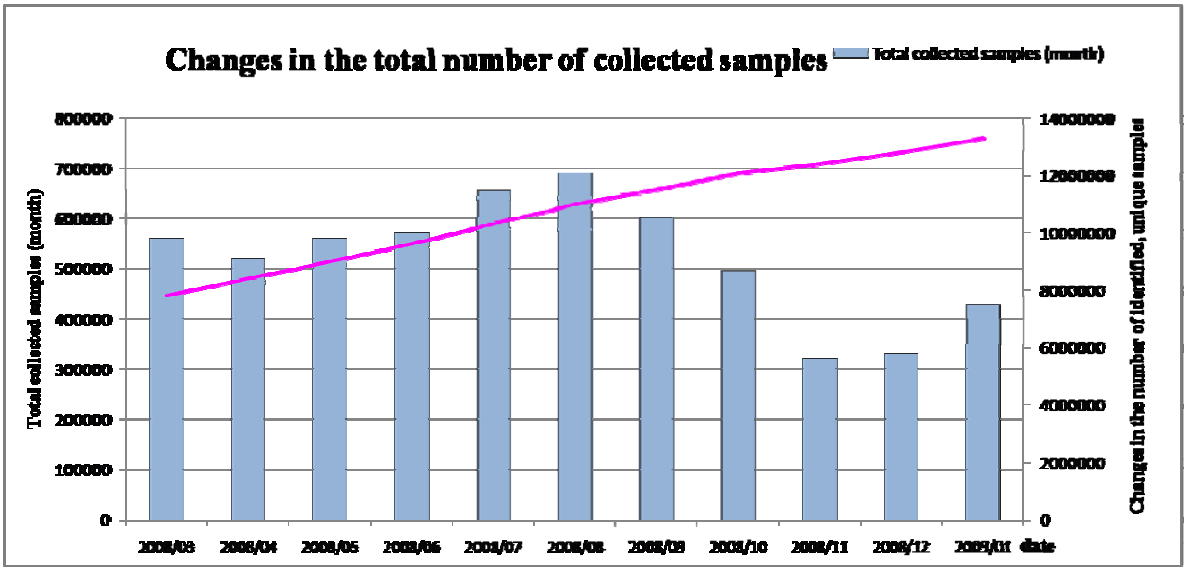

Figure 2.2-2: Changes in Total Number of Collected Samples

(2) Changes in the number of identified, unique samples

The monthly average number of identified, unique samples derived from identifying and analyzing the collected samples is approximately 55,000, which amounts to approximately 1,800 unique samples on a daily basis.

Figure 2.2-3 shows the changes in the number of identified, unique samples by month. The figure shows that the number of identified, unique samples increased in July and August 2008. The reason for this is that the types of BOTs increased in number due to propagation of file infectors, which infect files such as EXE and SCR files (screen savers) running in processes within a system. In the second half of FY 2008, the number of identified, unique samples decreased as these file infectors decreased.

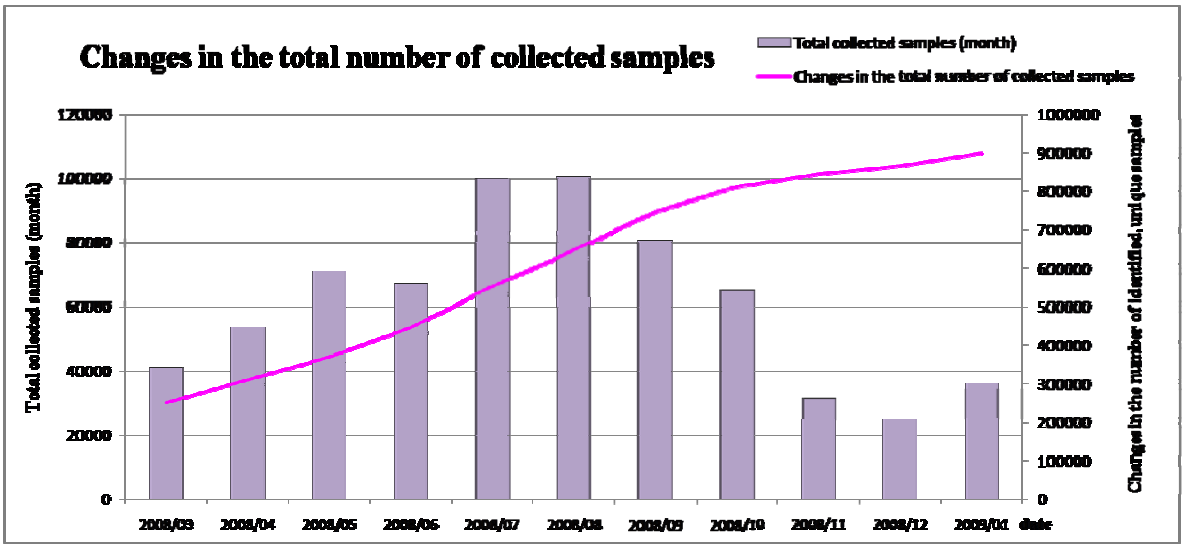

Figure 2.2-3: Changes in Number of Identified, Unique Samples

(3) Changes in identified, unique samples, known and unknown

According to the results of performing the known and unknown isolation analysis on the

identified, unique samples, analysis determined that the known samples totaled approximately 54,000 while the unknown totaled approximately 1,000 among approximately 55,000 identified, unique samples on average per month. This means that approximately 30 unknown samples were collected on a daily basis. Figure 2.2-4 shows the changes in the number of known and unknown identified, unique samples by month. The figure indicates that the number of unknown identified, unique samples decreased, although it fluctuates each month.

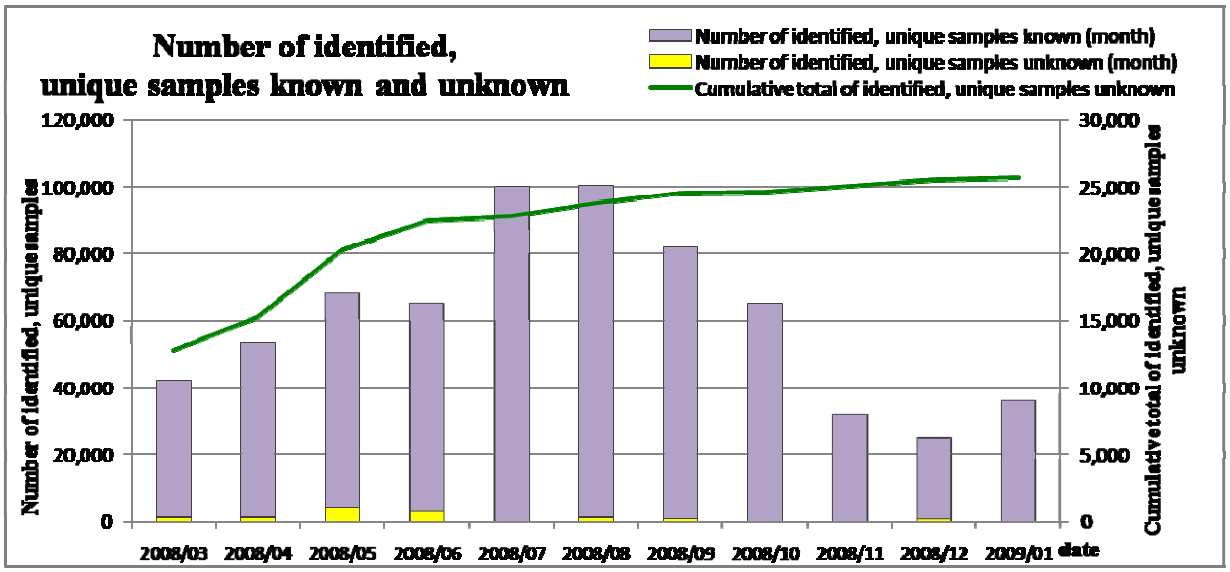

Figure 2.2-4: Changes in Number of Known and Unknown Identified, Unique Samples

Figure 2.2-5 shows the trend for changes in the number of collected samples by month throughout the year. TSPY_KOLABC.CH, which indicated a conspicuous move among the collected samples, ballooned in February 2009 and decreased sharply in March 2009. The cause was likely a high volume of samples received from specific sites overseas between December 2008 and February 2009. The number of collected instances of TSPY_KOLABC.CH decreased as the specific sites stopped delivering the viruses at the beginning of March 2009.

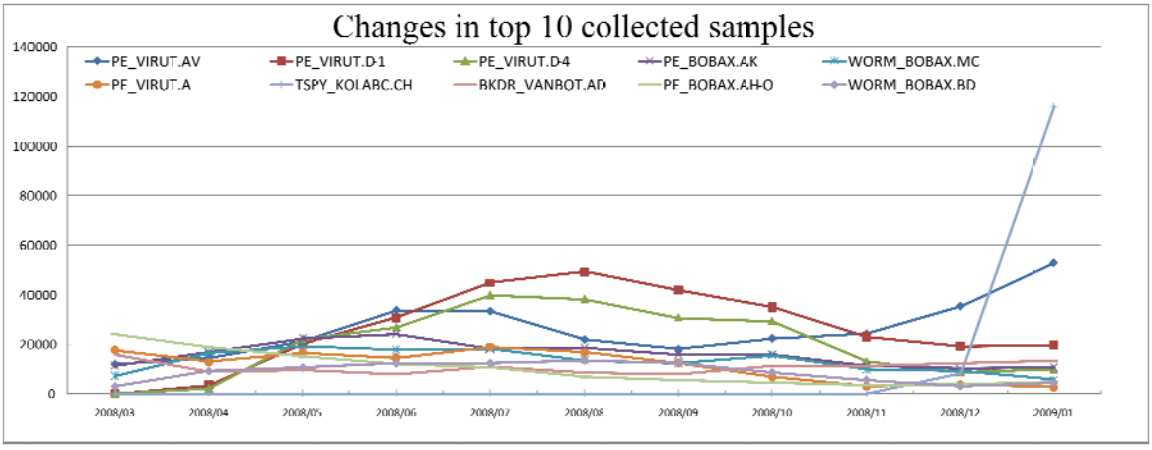

Figure 2.2-5: Changes in the Top 10 Collected Samples

(4) Sample collection in special environments

BOTs spread infection and propagate from attacks that focus on vulnerabilities in operating systems and software accessible via networks. Communication ports used in attacks tend to target specific ports. For this reason, the group investigated whether BOT infections could be suppressed by deliberately, externally interrupting communications involving widely-used, targeted ports in some CCC honeypot environments and whether this could be an effective measure in controlling BOT infections.

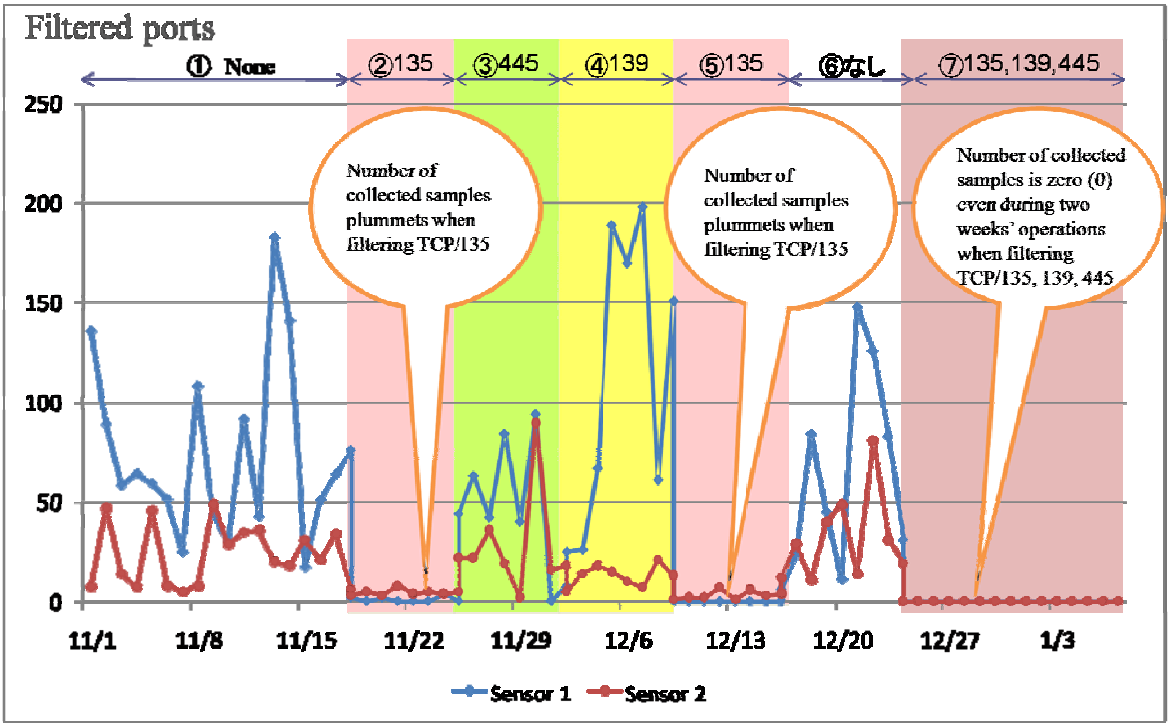

Figure 2.2-6: Changes in the Number of Collected Samples When Filtering Specific Ports

The group used two honeypots (Sensor 1 and Sensor 2) when conducting the investigation and observed changes in the number of collected samples while altering the filtering conditions. The filtering on the TCP/135 port showed the greatest effect, and the trial was then continued by adding TCP/139 and TCP/445. The result obtained showed that the number of collected samples was zero (0) during about two weeks of honeypot

operation period through filtering on these three ports. This means that if we carry out similar measures against malware and the types of BOTs that actively spread infection, this may lead to preventing them from spreading further. However, it is highly likely that the cause of continued infections by such malware is that systems do not have basic countermeasures applied, such as applying Windows Update, installing antivirus software, and installing broadband routers. For users who do not seem to be able to carry out these basic countermeasures, there remain various issues, but we could expect significant impact if measures such as filtering and blocking ports could be implemented on the side of networks providers, such as ISPs, in the future.

2.2.2 Alert activities

(1) Status of user measures performed

To eliminate BOT infections on computers, it is vital to identify such infections and alert the user to this fact. The BOT Countermeasure System Operation Group has been alerting infected users through emails in cooperation with the Project Participating ISPs. In FY 2008, the number of alert emails sent was 373,207 and the number of alerted users was 79,050 (see Figure 2.2-1). The alert mails in FY 2008 achieved the following: The rate of visiting the countermeasures site was approximately 41%, the Windows Update rate was approximately 31%, and the rate of downloading CCC Cleaners was approximately 30%. In addition, the rate of users reporting that all the countermeasures had been completed was approximately 15% (refer to Figure 2.2-7).

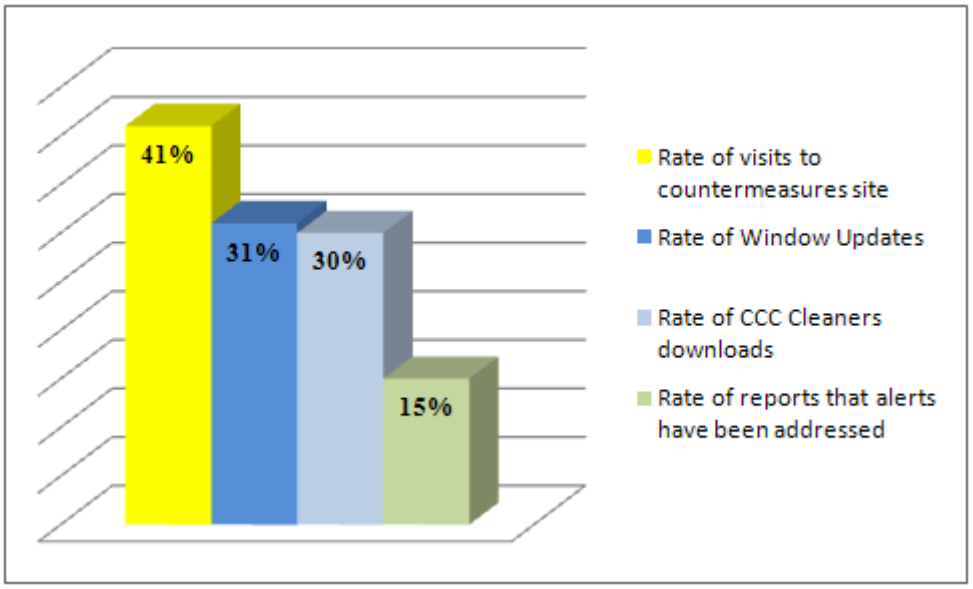

Figure 2.2-7: Status of Users Responding to Alerts

(2) Outcome

Figure 2.2-8 shows the outcomes achieved with each stage alert activities in this project. In the status of alert activities from April 2008 to March 2009, the total number of attacks (number of collected samples) amounted to 1,775,068, and the number of IP addresses for which alert activities were conducted amounted to 206,896. The reason why 1,775,068 attacking sources were reduced to 206,896 IP addresses is that multiple attacks were waged from the same IP addresses. The Project Participating ISPs identified users based on these 206,896 IP addresses, and 24,836 users were identified as being infected as a result. The Project Participating ISPs carried out 97,935 alert activities for these infected users, and

12,665 users accessed the countermeasures site for infected users and 7,664 users downloaded CCC Cleaners as a result.

Total number of attacks	1,775,068
↓	
Alert target IP addresses	206,896
↓	
Number of identified users	24,836
↓	
Number of alert activities (number of emails)	97,935
↓	
Number of users accessing countermeasures site	12,665
↓	
Number of CCC Cleaner downloads	7,664

Figure 2.2-8: Outcomes of alert activities

In addition, Figure 2.2-9 shows changes in the number of new alert target users by month (February 2007 to March 2009). This supports the belief that alert activities undertaken by this project have borne fruit as the number of alert activities by ISPs decreased over time.

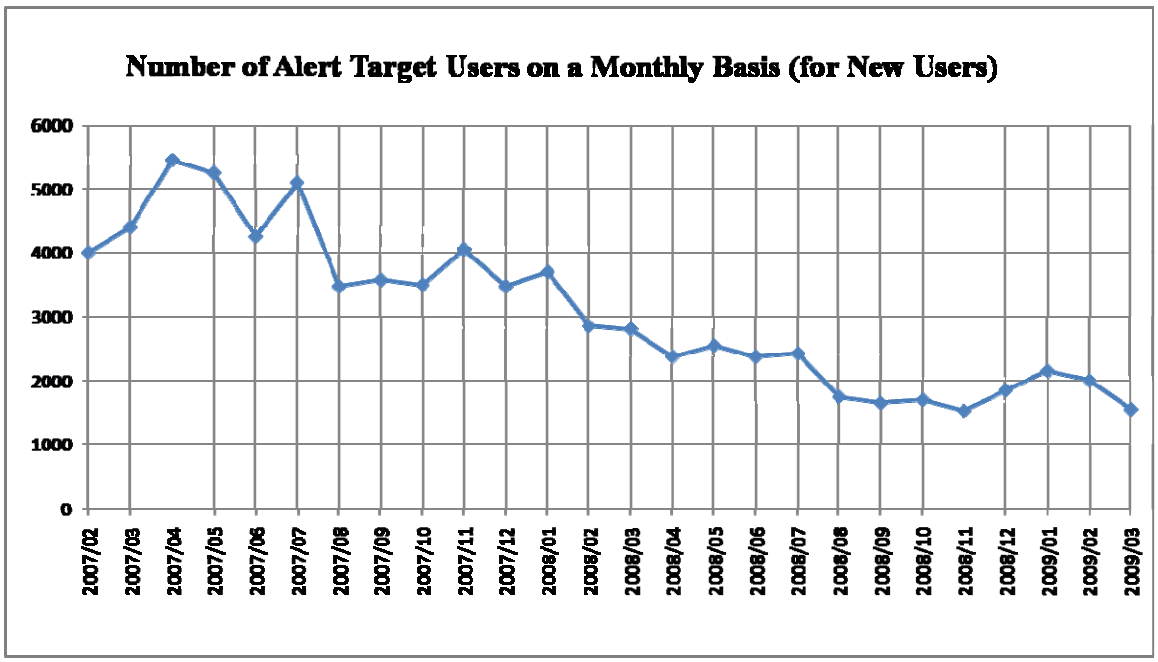

Figure 2.2-9: Changes in New Alert Target Users (02/2007 to 03/2009)

(3) Other approaches

In FY 2008, the group carried out a "BOT Disinfection Activity Campaign" as one of

events related to “Information Security Day”* (February 2009) to disseminate information about the BOT countermeasures project in Japan. The “BOT Disinfection Activity Campaign” became a cross-industry approach that requested not only Project Participating ISPs but also companies in several industries (municipalities, securities, electric utilities, gas utilities, railroads, and so forth) introduced to the project through preparatory meetings for establishment of the CEPTOAR-Council** to set up links to CCC in their respective Web sites.

(*http://www.nisc.go.jp/isd/index.html)

(**Important infrastructure liaison council)

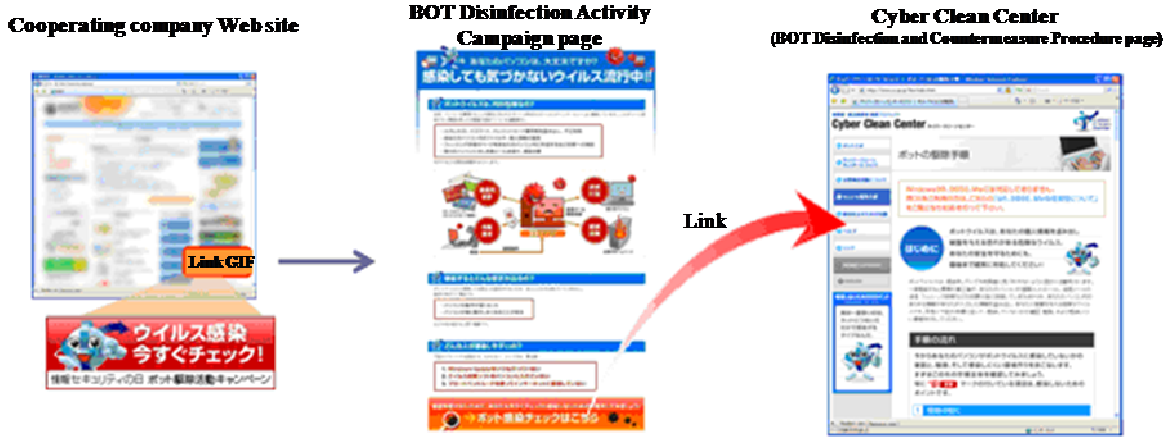

Figure 2.2-10: BOT Disinfection Activity Campaign

2.3 Future Developments

The practical approaches this project has taken are producing results because the number of new users receiving alerts has continued to decrease throughout the course of this project. Awareness of the approaches taken by this project has also been heightened, indicated by the fact that the number of accesses to the official CCC Web site is increasing due to media coverage of the work being undertaken.

In FY 2009 just as in FY 2008, the group aims to further reduce the number of BOT-infected users by continuing its alerting activities. In order to achieve this, it intends to take various approaches to ensure that users will be encouraged to take action in response to alerting activities, in addition to pressing on with creating mechanisms to efficiently detect infected users.

3 Activity Report – BOT Program Analysis Group

3.1 Overview

The BOT Program Analysis Group analyzes the BOT samples collected using “honeypots” and develops CCC Cleaners that reflect the analysis results. In this fiscal year, this group has enhanced its operations to learn the actual conditions of the environment that users are running CCC Cleaner in and promoting counter-measures intended for users. These include functions to check CCC Cleaner operating environments (such as checking if service packs have been applied) and extending information collected in the detection status reporting function.

The group also performs detailed analysis with static analysis techniques on collected samples with an emphasis on those that are unique or interesting, aiming to reflect the analysis results in future threat forecasts and countermeasures while carrying out analysis in order to reflect its results in CCC Cleaners.

In addition to the above efforts, the BOT Program Analysis Group continues to provide BOT samples to the BOT Infection Prevention Promotion Group just as in FY 2007.

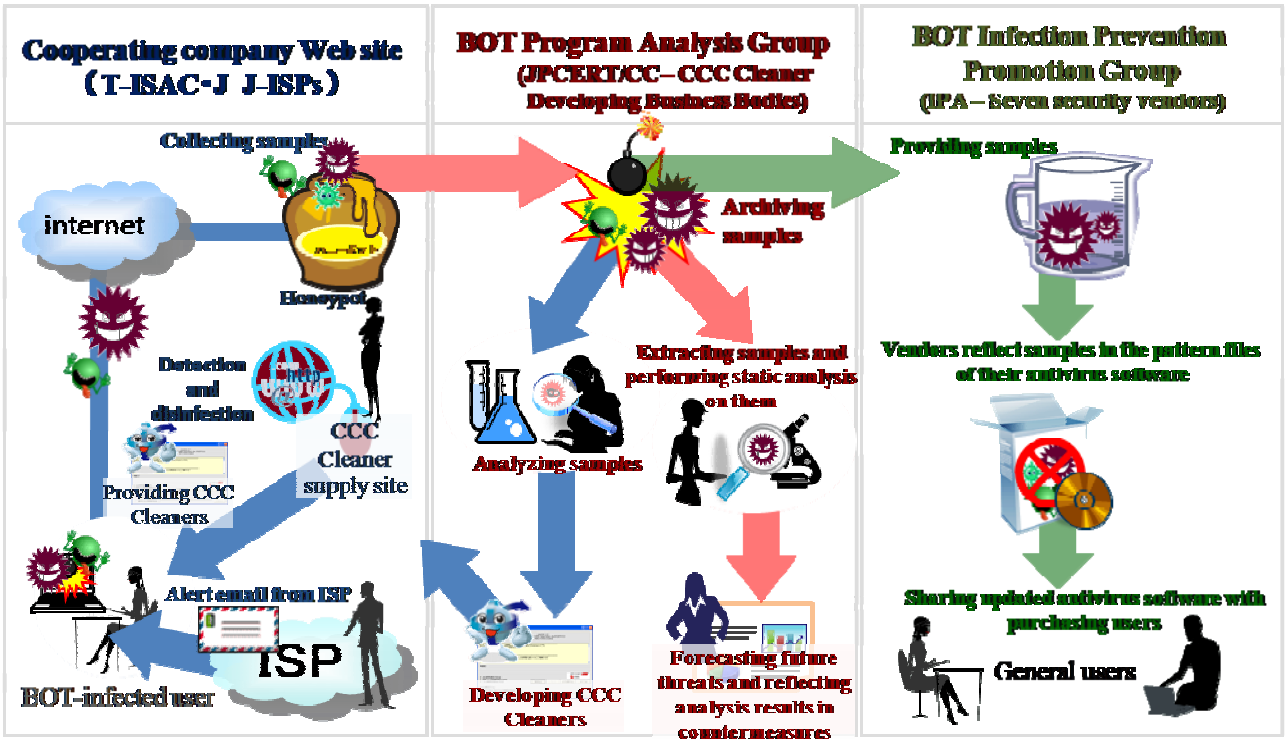

Figure 3.1-1: Role of Each CCC Group

3.1.1 Achievements related to the number of sample analyses and the number of BOT samples reflected in CCC Cleaners

Assuming the cumulative total of identified, unique samples from February 2007 as a base, we derive 99.38% as the CCC Cleaner coverage rate from using the number of samples for which CCC Cleaners were developed and the number of simplified analyses (known samples).

(1) Number of samples for which CCC Cleaners were developed = 18,334
(2) Number of simplified analyses (known samples) = 846,510
(3) Number of identified, unique samples = 870,227
(4) CCC Cleaner coverage rate = ((1) + (2)) / (3) = 99.38 %

This means that 99.38% of the collected samples can be detected as BOT samples with antivirus software and CCC Cleaners, suggesting that the samples collected were being fully utilized.

(1) Number of samples for which CCC Cleaners were developed:
Number of samples regarded as highly dangerous and having many infected users for which CCC Cleaners have been developed
(2) Number of simplified analyses (known samples):
Already known samples that have been identified as ones even existing tools can deal with, from among those collected
(3) Number of identified, unique samples:
As many identical BOT samples are collected, this is the number of unique samples derived from regarding the duplicate ones in terms of their sizes and external characteristics as a single sample.

3.2 Developing CCC Cleaners

With regard to the samples that have not been addressed by commercially available antivirus software, the group analyzes information, such as the file types related to the infection of BOT viruses, and develops CCC Cleaners.

3.2.1 Adding further functions to CCC Cleaners

In FY 2008, the group added additional functions with the aim of improving CCC Cleaners from the viewpoint of users. The following describes the additional functions:

(1) Adding support for processing file infection-type BOTs
In the case when files under the system folder are infected with a BOT and cannot be removed, the group has changed the way of dealing with such cases by now displaying a popup window and aborting the search and disinfection process without trying to disinfect the files. For file infection BOTs that are detected other than in the above case, the group has added changes to give a warning with a popup indicator. Adding this function enables the group to alert users to the fact that they are infected with a file infection BOT and inform them of ways to deal with it.

(2) Improving the detection status reporting function

The group has added additional information that is sent with the detection status reporting function already implemented in existing CCC Cleaners. Improving this function enables the group to collect more information on CCC Cleaner running environments and to use them as statistical information described in the next section, *3.2.2 Analysis of detection status*, and for consideration of more effective countermeasures and other purposes. The following lists the information items sent by the detection status transmission function:

- Information on OS version
- Time and date executed
- Number of detections/number of removed BOTs/number of non-removed BOTs
- Detected malware name
- Error information
- Amount of memory present
- Detection results of file infection BOTs
- Results from checking hosts file alterations
- Results from determining connection status

Note that this function has also been changed to display a popup window after the search and disinfection process has been completed, so that users can select whether they want to send back this information.

(3) Function to check if service packs have been applied
The group has implemented a function to check if the most recent service pack has been applied in the Windows environment of each user. As of March 31, 2009, the function displays a warming popup when it finds that users have not applied SP3 for Windows XP, SP1 for Windows Vista, and SP4 for Windows 2000. Adding this function enables the group to easily notify users with information on how to apply the service packs.

(4) Function to restore unauthorized hosts file alterations
The group has implemented a function that checks the hosts file when running CCC Cleaner to prevent any blocking of Windows Update and updates to antivirus software, and to stop "pharming" attacks. It alerts users by displaying a popup and at the same time renames the existing hosts file and creates the renamed hosts file with default settings if hosts file alterations are suspected.

(5) Function to determine connection type
The group has added a function to check whether the IP address of the PC that is running a CCC Cleaner is a private IP address or a global IP address, and will display a warning popup if it is a global IP address. Adding this function enables the group to check if a broadband router is present on the network CCC Cleaner is running on.

3.2.2 Analysis of detection status

The following section describes results from collecting detection status reporting logs (hereafter, "transmission logs") that have been sent to the group by users at their discretion with the detection status transmission function in CCC Cleaners.

(1) Changes in the number of transmitted logs (target period: April 2008 to March 2009)
The following figure shows the changes in the number of collected transmission logs. The

figure shows that the number of transmission logs rose sharply in February 2009. The likely reason is that this time fell in the period during which events related to "Information Security Day" were being held and access to the CCC site was increasing due to the effect of the events, media coverage, and so forth.

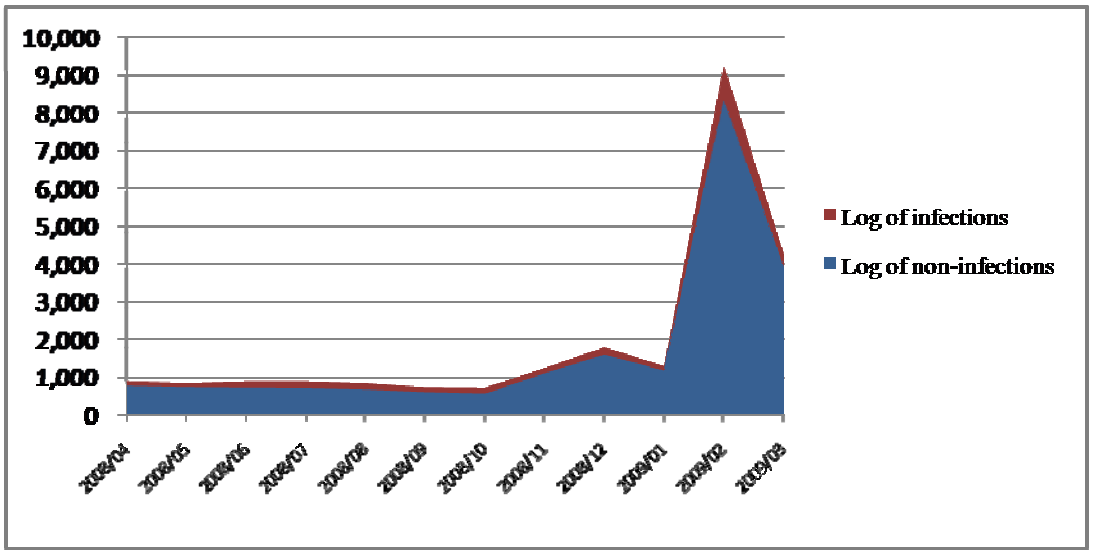

Figure 3.2-1: Changes in the Number of Collected Transmission Logs

(2) Collection rate by the OS (target period: April 2008 to March 2009)
The following figure shows the collection rate by OS. The group could confirm that Windows XP still ran on most of the PCs used by log transmission users.

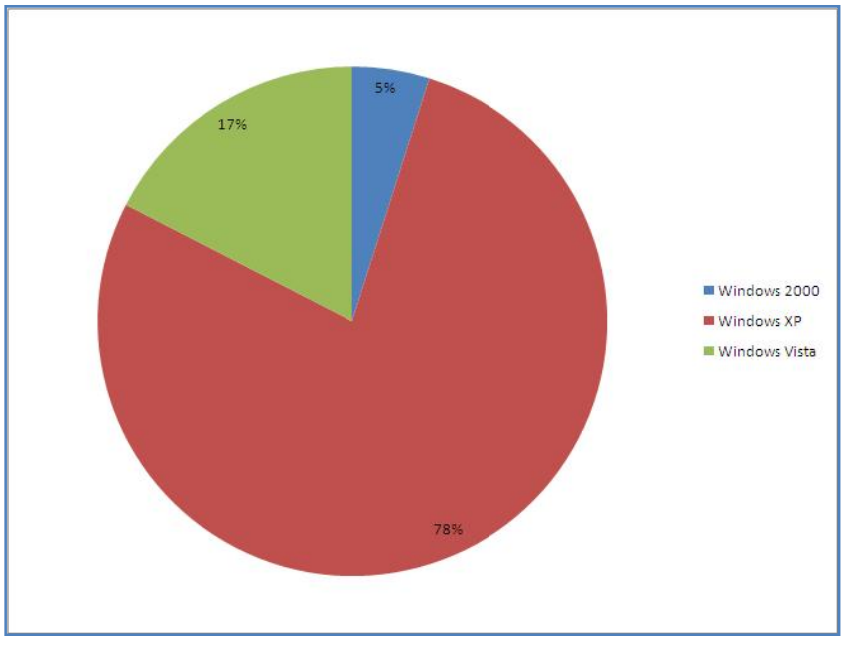

Figure 3.2-2: Collection Rate by OS

(3) Collection trend by OS (target period: April 2008 to March 2009)
The following figure shows the changes in the number of collected transmission logs by OS.

The figure shows that the number of collected transmission logs from Windows XP SP3 and Windows Vista SP1 increased noticeably from November 2008. Note that the sharp rises in the graph from January to February 2009 were affected by the absolute number of log reports due to particular factors, such as events on “Information Security Day.”

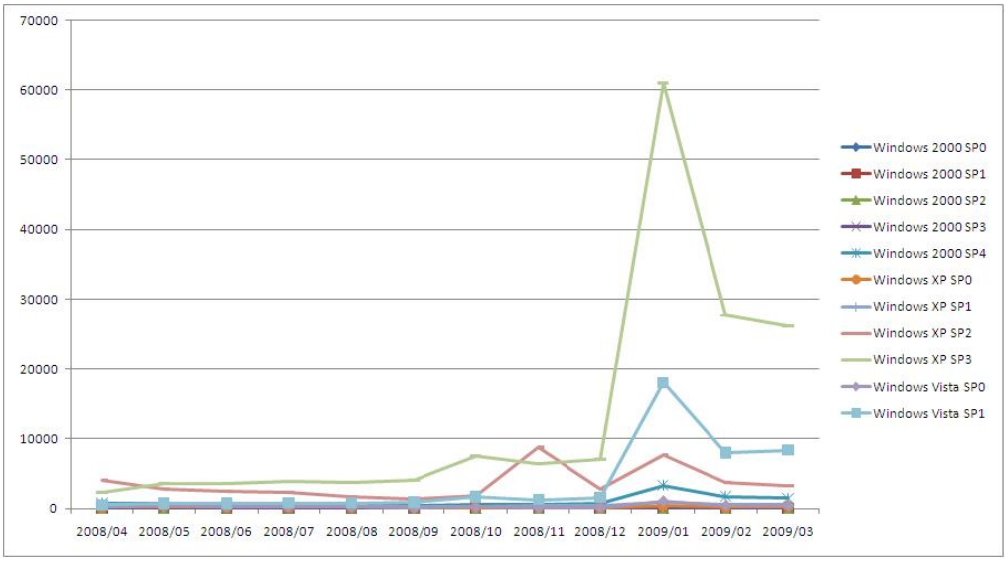

Figure 3.2-3: Changes in the Number of Collected Transmission Logs by OS

(4) Infection rate by the OS service pack (target period: April 2008 to March 2009)
The following figure shows the infection rate by the OS service pack. The total number of transmission logs differs between service packs in each OS, but in the case Windows XP with its many users, the group found a tendency that shows the higher the version of a service packs was, the lower the rate of infection.

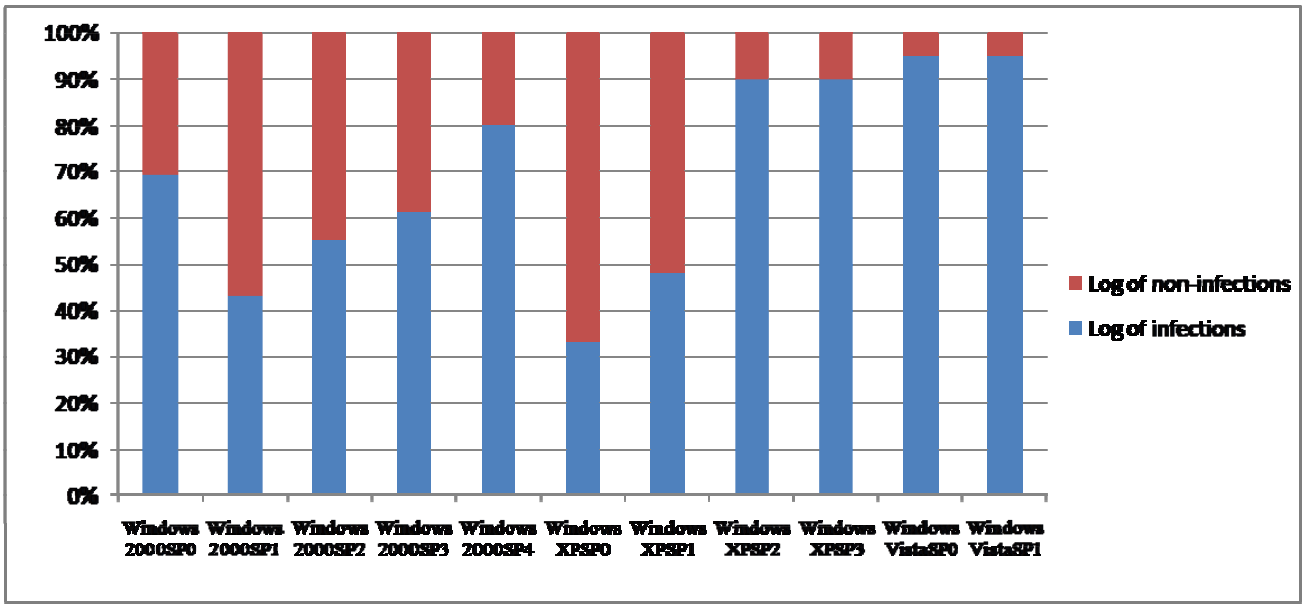

Figure 3.2-4: Infection Rate by OS Service Pack

(5) Status of broadband router introduction (target period: December 2008 to March

2009)
The following figure shows the results from determining the connection types of users identified as infected with BOTs, who occupied 9% of total users. The ratio of global IP addresses of PCs was 19% among the users of the general site and 44% among those who received alert emails. We infer from this that there are still many environments where broadband routers have not been installed.

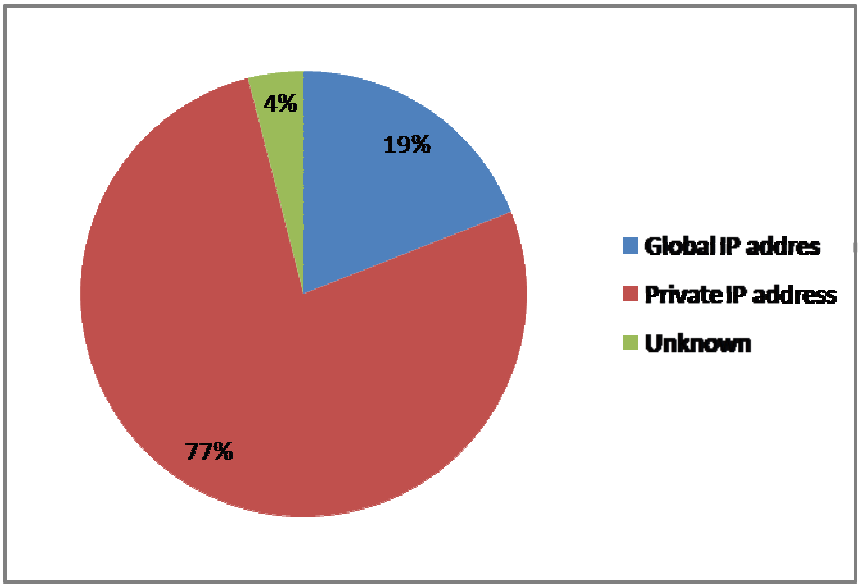

Figure 3.2-5: IP Address Rate among General Site Users

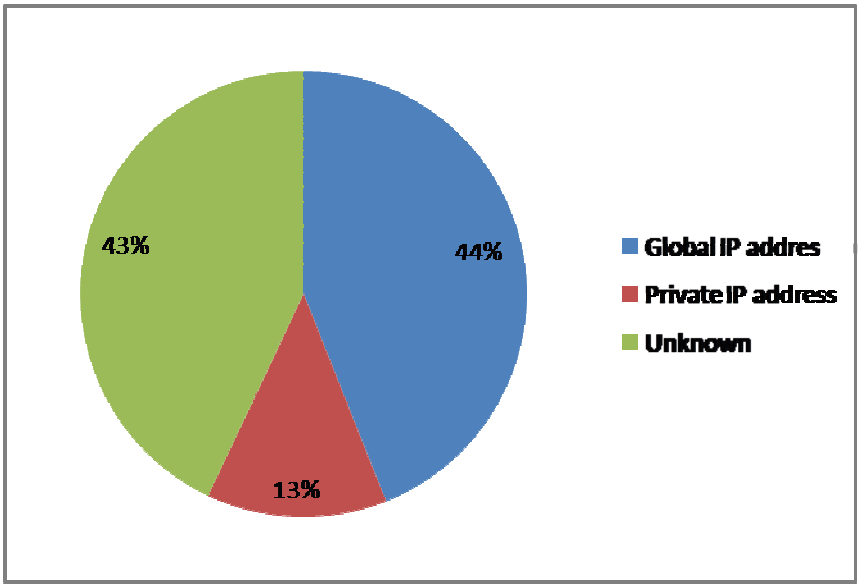

Figure 3.2-6: IP Address Rate for Users Who Received Alert Emails

(6) Status of hosts file alteration (target period: December 2008 to March 2009)
The following figure shows the results for detecting altered hosts files for users identified as infected with a BOT, occupying 9% of total users. Alterations in hosts files were detected in 5% of the general site users and 11% of the users who received alert emails, and we infer from this that there are many users affected by connecting to servers as a result of altered

hosts files.

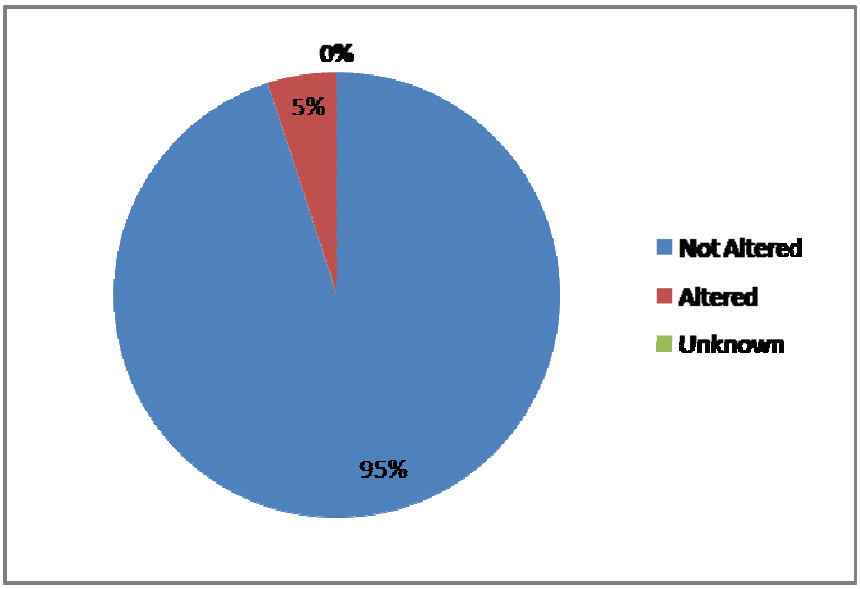

Figure 3.2-7: Detection Rate of Hosts File Alteration in General Site Users

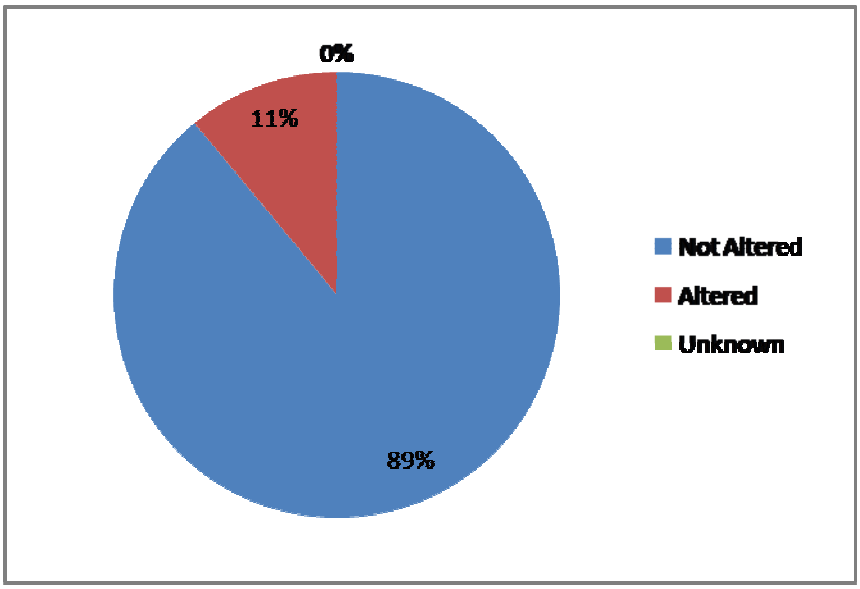

Figure 3.2-8: Detection Rate of Hosts File Alteration in Users Who Received Alert Emails

(7) Infective sample trend (target period: April 2008 to March 2009)
The following figure shows the status of samples detected in user environments by their names. The infection trend in each environment was that PE_VIRUT-family infective samples (file infectors) and samples via networks were collected in “honeypots” (Figure 3.2-9: Rate of Infective Samples in Transmission Logs). PE_VIRUT-family infective samples (file infectors) as well as samples such as WORM_AUTORUN that seemed to spread infection via means other than network attacks were seen in transmission logs (user environments; see Figure 3.2-10: Rate of Samples in CCC Honeypots (Identified, Unique Samples)).

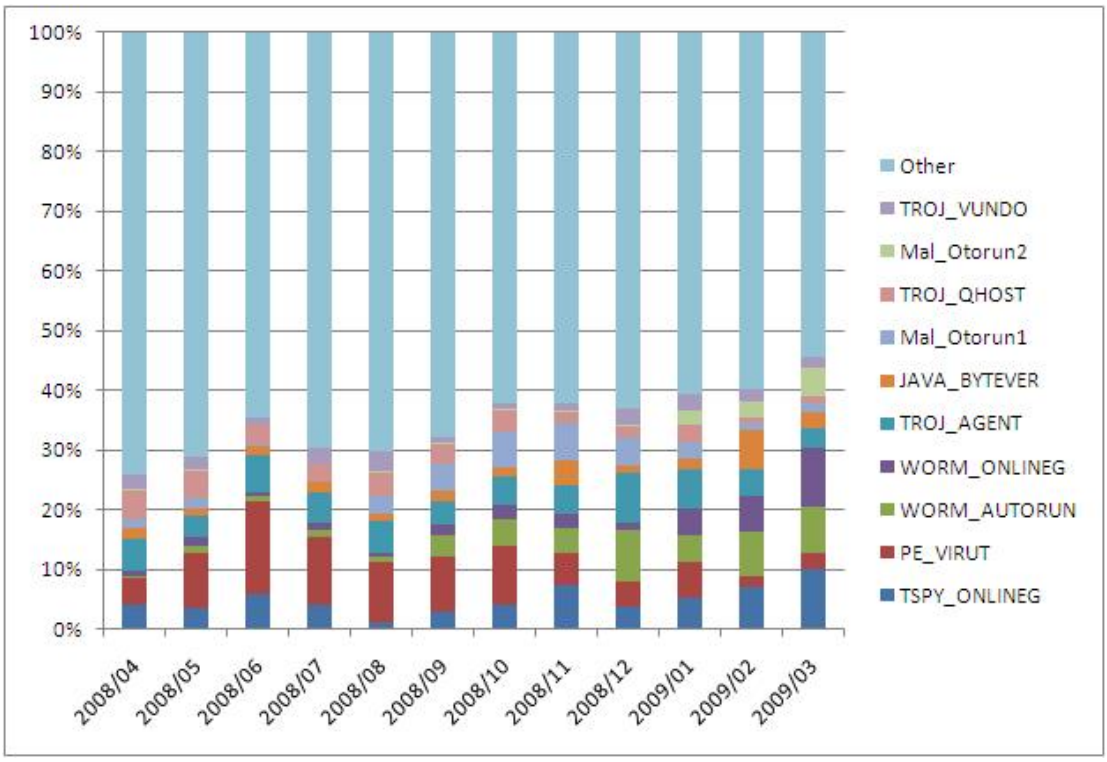

Figure 3.2-9: Rate of Infective Samples in Transmission Logs

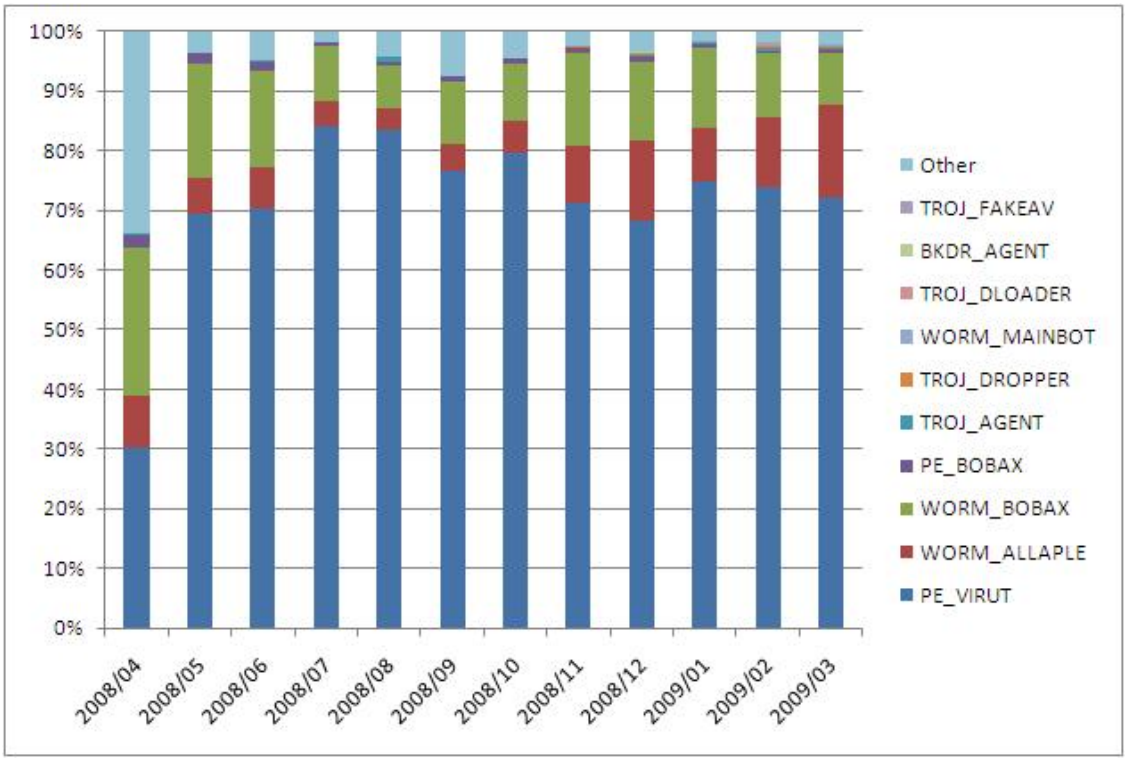

Figure 3.2-10: Rate of Samples in CCC Honeypots (Identified, Unique Samples)

(8) Summary

The group could confirm from collection of transmission logs that many user environments where CCC Cleaners were used did not have broadband routers installed,

nor operating system patches appropriately applied. Consequently, the tools were being used in environments that remained very vulnerable to infection. In addition, the group found that some infection cases seem to have been spread through USB and the Internet. These results indicate that recent malware tends to have more than one infection path. We have concluded from such situations that it is necessary to continue promoting the following campaigns:

- Windows Update promotion
- Broadband router introduction
- Antivirus software installation

3.3 BOT Analysis

The BOT Program Analysis Group aims to identify any threats from prevalent BOTs by periodically sampling and analyzing BOTs that are currently highly active and forecast future threats from them. The group also explores prevention measures against BOTs by accumulating knowledge and experience garnered from such activities.

The following provides details of analysis activities:

3.3.1 Analysis on changes in samples through version upgrading

The group has focused on version upgrades of BOT samples as an approach to forecasting future threats from BOTs. In this investigation, the group examined what changes could be found in BOT samples as the versions of the BOT programs advanced, as well as the relationship between time and version upgrades, in order to study future threats.

Note that the group used the names of mutex algorithms as used by Windows for exclusive control between processes as the method to obtain version information.

(1) Family A
The following four versions have been confirmed in Family A:

- v2.3
- v2.4 tested
- v2.5
- v2.9

In addition, the following features were detected as common to all the versions:

- Feature of code
  - Code written in C language
- Features of functions
  - Connects to IRC to receive commands
    - Runs commands on an infected machine
    - Scans ports
    - Downloads and runs files
    - Sends information about the infected machine

Table 3.3-1 summarizes changes noted across the entire family by comparing results from analyzing each version in Family A.

Table 3.3-1: Changes in Family A

New function	None
Anti-analysis function	Implemented different function, depending on version • Packer change • Code obfuscation • Parameter encoding
Existing functions	Added IRC command Removed the downloading function of sub-files Changed parameters
Appearance	Reconstructed a part of the code

In this family, the functions such as packers or code obfuscation that block analysis noticeably changed. However, although their code was changed, it is not that they have become more sophisticated. The reason for this could be that it is expensive to buy packers with sophisticated anti-analysis functions and implement a complex code obfuscating process, while the major purpose of malware authors is to avoid detection by antivirus software. One of the possible reasons why packers available for purchase were not used is that malware authors only sought to reduce the size of files generated (a basic function of packers), and they do not require the more sophisticated obfuscation features provided by some advanced packers.

(2) Family B
The group confirmed the following version numbers in Family B:

- v011ALPHAA
- v0111ALPHAA
- v0122ALPHAA
- v0122ALPHAA27
- v0.2_Beta_711d43
- v0.66_Beta_erf

The following features were detected as common to all the versions in Family B:

- Features of code
  - Packing with UPX
  - Code whose main section was written in C language
- Features of functions
  - Creates a multi-function backdoor, which receives communications from the outside
    - TCP/UDP proxy, etc.
  - Sends information about clients to the outside

Table 3.3-2 summarizes changes noticed in the entire family by comparing results from analyzing each version in Family B.

Table 3.3-2: Changes in Family B

New function	None
Anti-analysis function	No changes
Existing functions	Added a function to confirm whether connected to the Internet Added a function to check data and duplicate startup Added and removed specific proxy functions Removed a function to check other backdoors Added a function to check a black list Added parameters
Appearance	Reconstructed a part of the code

In this family, the sophistication of backdoors is evident as the proxy that is the main function of the family. Upgrading the version of the program increased the backdoors in number and function. At the same time, upgraded versions began to send host information and information obtained through backdoors to multiple destinations, having used to send to only one. Additionally, it added functions to limit the number of threads to backdoors and to set a sleep timer as the versions become higher. We could infer that malware authors optimized their code or reviewed the functions in the later versions because the backdoor functions and information transmission functions that had been added in the earlier versions were substantially trimmed. These results gave analysts the impression that malware authors performed careful maintenance on their programs with each version.

(3) Family C
The group could confirm that the following versions were present in Family C:

- v2.0
- v3.0
- v3.5
- v6.0

The following features were detected as common to all the versions in Family C:

- Feature of code
  - Code written in a C language
- Features of functions
  - Infection activities through TCP 135 Exploit
  - Downloads and runs files

Table 3.3-3 summarizes changes noticed in the entire family by comparing results from

analyzing each version in Family C.

Table 3.3-3: Changes in Family C

New functions	None
Anti-analysis function	Versions of packers were upgraded
Existing functions	Added a function to transmit information about the infected host Added a message transmission function using Messenger Added support to deal with TCP connection restrictions in Windows XP SP2 Added parameters
Appearance	No changes

In this family, it was found that sub-functions such as an attack function using Messenger and an infected host information transmission function had been added. Additionally, changes in the code concerning other common network attacks were not detected. It was found that authors of Family C malware left evidence of maintenance as also seen in Family A, indicated by the fact that they added parameters or support to deal with restrictions on the number of TCP half connections (restrictions on the number of attacking threads), as implemented in Windows XP SP2 or later, as changes to the main function.

The analysis results of all the families can be summarized as follows: the changes by version-upgrading the programs were mainly intended to maintain them and avoid antivirus software; new functions that may lead to threats in the future were not found. The analysis group found that version upgrading did not add new and major changes, similarly to general software, and there may be a limit to the approach of keeping track of version upgrades of BOT programs in forecasting future threats.

Based on the analysis results above, the group concludes that it should adopt different approaches to analyzing samples.

3.3.2 Analysis on changes in samples delivered from the same site

The group performed analysis of future threats by observing changes in samples delivered from the same sites and determining information such as changing trends in the samples.

The group investigated as follows:
(1) Extracted information on specific sites, which continued to change and deliver samples, from the data collected by the BOT Countermeasure System Operation Group
(2) Performed simplified analysis and static analysis on the delivered samples
(3) Analyzed changes in the samples on a per-site basis

The group investigated the following three sites that continue to deliver unknown

samples:

Table 3.3-4: Overview of Target Sites

Items \ Sites	Site 1 (UK)	Site 2 (USA)	Site 3 (Japan)
Target period	10/03/2008 to 12/7/2008	12/31/2008 to 1/14/2009	12/06/2008 to 12/10/2008
Site survival period	Still alive (as of February 20)	15 days	5 days
Number of hash-unique delivered samples	35 samples	19 samples	110 samples

The following table shows the summary of analysis results of the target sites:

Table 3.3-5: Summary of Analysis Results of Target Sites

Items \ Sites	Site 1 (UK)	Site 2 (USA)	Site 3 (Japan)
Delivered samples	IRC BOT Port scanner	IRC BOT Port scanner	Remote shell
Average delivery period	3 days/sample	3 days/sample	1 day/sample
Average number of delivered samples	1 to 2 samples/day	1 to 2 samples/day	20 to 60 samples/day
Changes in samples	• No function changes • Changes found in sections before the entry point of the executable main section	• No function changes • Changes found in sections before the entry point of the executable main section	• No function changes • Changes found in sections before the entry point of the executable main section

The group found that all the sites delivered more than one type of hash-unique samples every day, and the delivered samples had almost the same content on each site although their hash values were different. This means that the sites in the UK and the USA continued to deliver IRC BOTs and port scanners, while the one in Japan continued to deliver remote shells, frequently changing hash values. Note that as the IRC BOTs and port scanners in the UK and USA were the same, we could infer that the sites were somehow related to each other.

Many samples had the same content in spite of different hash values, and one of the purposes of delivering them was to avoid detection by antivirus software. The group

believes that malware authors generate a succession of samples with different hash values from one malware body every day using a tool so that their work may avoid detection by antivirus software.

If malware authors continue to create and deliver samples with different hash values, samples with the same hash values will not be delivered even when the pattern files of antivirus software are updated to reflect the samples after obtaining them. As the same samples with different hash values that cannot be detected will be delivered, a situation where antivirus software cannot deal with the BOT could develop as a result. Recently, the analysis group has dealt with BOTs using new approaches, such as generic patterns or by reputation, but finds itself in a situation where perfect countermeasures are difficult to develop because antivirus software must always consider other issues, such as false positives.

In this situation, one possibility is employing a site closure coordination process to deal with malware delivery sites exploiting the time delay until deployed malware samples are reflected in pattern files of antivirus software. The group plans to investigate issues such as whether site closure coordination techniques can be implemented and how useful they would be as a countermeasure against the continued delivery of unknown samples that antivirus software cannot intercept.

3.3.3 Results from detailed analysis of distinctive samples

The analysis group have not only tried to identify trends in samples collected by the BOT Countermeasure Promotion Project, but also analyzed whether new techniques were being applied or whether they would be used in the future.

They investigated the investigated following:
(1) Carried out dynamic analysis on collected samples and extracted samples that took distinctive actions
(2) Carried out static analysis on the extracted samples, analyzing the details of their functions

As a result of these dynamic analyses, 21 samples were extracted from those collected. The following distinctive functions were found from the results of detailed analyses on each extracted sample:

Table 3.3-6: List of Distinctive Functions Derived from Detailed Analysis Results

Functions		**Description**
BOT functions	IRC	Function to receive instructions from a Herder through IRC
	HTTP	Function to send an HTTP request to a specific server and determine actions to be taken based on the response received
	Unique protocol	Function to exchange commands on well-known TCP/UDP ports using a unique protocol implemented by the malware author

Functions		Description
Self-concealment functions	Information manipulation with API hook	Function to conceal information by rewriting APIs involved in operations on processes, files, and registries
	DLL injection with callback	Function to monitor processes to be started using device drivers and perform DLL injection to started processes
	Manipulation of internal OS information	Function to conceal itself by directly changing internal OS information, such as PEB (Process Environment Block)
Anti-analysis functions	Code injection	Technique for malware to write itself, files or code into another process to let that process run them
	Code obfuscation	Technique that makes assembly code difficult to read by inserting meaningless codes or dividing single functions into smaller units
	Parameter encoding	Technique that holds parameters to be used, such as URLs, in encoded status beforehand and decodes them just before using them
Others	Obtaining information using UPnP	Function to communicate with a router using UPnP and obtain its global IP address
	Obtaining information using external sites	Function to obtain information such as global IP addresses, communication speeds, and items registered in black lists by using external sites
	Changing settings with system alterations	Function for releasing restrictions on connecting to TCP half connections by altering tcpip.sys
	P2P	Function to build a peer-to-peer network to communicate with many hosts

It was found that the samples examined were not BOTs that wage attacks on operating system vulnerabilities or that had multiple functions to spread infections, but that there were many pieces of malware that specialize in a specific purpose or function such as sending SPAM mails or downloading and running files. In addition, such types of malware attempted to conceal evidence of their infection by using “rootkit” functions or “code injection.” The analysis group inferred that the reason they were created this way was so that they could provide their functionality uninterrupted over a long period, most likely for commercial gain.

When each of the distinctive features was inspected, no functions that were observed for the first time in FY 2008 were found. However, with regards to unique protocols, the group found several types they have not yet finished analyzing. Continued analysis is planned due to the anticipation that such unique implementations will continue to advance and become more sophisticated in the future.

As stated, no new functions have been found of late, but for UPnP and processes to alter

tcpip.sys and to obtain IP addresses using external sites, the group can confirm that similar processes are used in the W32.Downadup family malware that exploits MS08-067. According to the malware information provided by Symantec Corporation, W32.Downadup family malware is classified as a worm that aims to spread the infection. However, the analysis group has not found any relation between such worms and other similar samples. We have inferred from the above investigation results that malware developers are sharing information by some means, such as referring to other malware analysis reports, in order to implement the required functions in their own malware.

Using dynamic analysis, we cannot investigate the details of payloads and processes of BOT commands; rather, we can only investigate the behavior of malware when it is executed. Such dynamic analysis allows malware to run for a fixed period of time and records what it processes; however, some malware may not complete its entire execution within the period of time specified in the dynamic analysis by frequently performing a Sleep command.

Using static analysis, the detailed activities of malware, payloads and the processing of BOT commands can be investigated, as well as all the servers to which the malware may connect. However, such static analysis is fairly time-consuming and laborious so may not always be able to fully analyze malware when its code is highly complex.

Based on these facts, the analysis group believes it must continue to undertake the following actions:

- Develop tools that enables it to share information such as analysis information on analyzed malware
- Develop tools that enable it to detect libraries, such as “zlib”, frequently used by malware
- Share techniques and information required to analyze rootkit functions
- Train more malware analysis engineers

3.4 Future Developments

In its activities in FY 2008, the group continued to enhance the functions of CCC Cleaners, analyzed logs involved in such enhancements, and analyzed malware samples. Over the next year, the group will continue its activities as in FY 2008, aiming for continual improvement.

(1) Developing CCC Cleaners and analyzing logs
The group continues to offer a steady supply of CCC Cleaners and at the same time analyze users’ detection status report logs.

(2) BOT analysis
Based on the results derived from activities in FY 2008, the group intends to analyze BOTs by utilizing new approaches, forecast future threats and explore countermeasures against them.

(3) Assisting diffusion and awareness activities
Just as in FY 2008, the group will assist in dissemination and awareness activities concerning BOT infection countermeasures.

4 Activity Report – BOT Infection Prevention Promotion Group

4.1 Overview

The BOT Infection Prevention Promotion Group commits itself to this project in cooperation with security vendors (“Project Participating Security Vendors”) to enhance BOT virus infection countermeasures and prevent the recurrence of damage caused by the same BOT viruses, for general users. Specifically, the group provides samples of the BOT viruses collected in this project to the Project Participating Security Vendors, enabling the vendors to reflect those samples in the pattern files of the antivirus software that they sell. In this way, if users keep the pattern files for their antivirus software updated, the antivirus software can detect and disinfect the BOT viruses collected by the project. Consequently, security measures can continue to be improved.

4.2 Project Participating Security Vendors

The respective Project Participating Security Vendors are legal entities that undertake strict administrative standards on the samples, set up departments to analyze them in Japan, and have a substantial past record of supplying their antivirus software and providing related services within Japan. Alongside the vendors participating in the project, the group campaigns to promote use of infection prevention resources in PCs etc. by users.
List of Project Participating Security Vendors (alphabetical order)

- AhnLab Incorporated
- Kaspersky Labs Japan Limited
- McAfee Incorporated
- Microsoft Corporation
- Symantec Corporation
- Sourcenext Corporation
- Trend Micro Incorporated

4.3 Activity Achievements

Table 4.3-1 shows the average figures in FY 2008 between March 2008 and the end of March 2009 (reported between May 2008 and April 2009), indicating whether the Project Participating Security Vendors had reflected, reflected at this time, or had not yet been reflected the samples collected by this project into the pattern files of their antivirus software.

Table 4.3-1: Samples Reflected in Pattern Files

	Average in FY 2008
Already reflected	98.6 %
Reflected this time	1.1 %
Not yet reflected	0.3 %

A total percentage of 99.7%, derived by adding the figure for “Already reflected” to that for “Reflected this time” indicates that 99.7% of samples collected in this project can be detected with antivirus software. From these figures, the group infers that, as one of the achievements of this project, the collected samples have been fully utilized, thereby significantly contributing to preventing ordinary computer users from becoming infected with BOTs.

4.4 Future Activities

The group intends to continue to take part in this project in cooperation with the Project Participating Security Vendors to strictly manage the collected samples and promote the further reflection of the samples in the pattern files of the antivirus software sold by the vendors.

5 Summary

The Botnet Countermeasure Project, which has operated since December 2006, is the first such project in Japan where MIC, METI and their related organizations and enterprises cooperate with the aim of eliminating domestic BOT infections. From a global standpoint, this could be said to be a rare co-operative project of this type. The approaches adopted by this project contribute to alerting many BOT-infected users and disinfecting BOT-infected machines. Because awareness of this project has been heightened by significant media coverage of the approaches and accomplishments achieved, we can conclude that this project has accomplished concrete results and is gaining wide acceptance.

As the number of BOT infections is still large, however, it is necessary to develop various means of providing alerts for even greater numbers of BOT infections and escalating activity for the disinfection of BOTs. As threats from BOTs are continually increasing, we also need further technical innovation to confront them. In addition, we must consider developing activities with an eye on cooperation with concerned organizations overseas, as the BOT threat exists not only within Japan, but across the globe. The activities of this project will continue in the future, aiming to make a significant contribution to realizing a safer and more secure Internet society.

6 Conclusion

~To Minimize Damage Caused by BOTs~

Cyber Clean Center (CCC) recommends taking BOT countermeasures to minimize the damage that can be caused by BOTs.
Although there are no measures that can guarantee to completely prevent damage caused by infection by BOTs, you can minimize the risk by adopting the following countermeasures:

Infection countermeasures

1. Make sure to keep your computer software up to date
2. Ensure that you install antivirus software
3. Use a personal firewall
4. Use a broadband router for connection to the Internet
5. Do not preview mails in HTML format
6. Pay careful attention to emails with attached files (attachments)
7. Use authentication with IDs and robust passwords

CCC presents detailed instructions to the public at the following Web site:
https://www.ccc.go.jp/knowledge/index.html

Please take appropriate virus countermeasures to protect the security of your computer.