GIAC
CERTIFICATIONS

Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permited without express written permission.

Interested in learning more?

Check out the list of upcoming events offering
"Hacker Tools, Techniques, and Incident Handling (Security 504)"
at http://www.giac.org/registration/gcih

Bot Threats to Corporate Networks

GIAC Certified
Incident Handler

Practical Assignment

Version 4.0



© SANS Institute 2000 - 2005, Author retains full rights.

Suchun Wu

March 18, 2005

© SANS Institute 2000 - 2005 Author retains full rights.

Table of Contents

Executive Summary 2
Statement of Purpose 3
Background Knowledge 4
The Exploit 6
Exploit Name 6
Operating System 7
Protocols/Services/Applications 7
Exploit Variants 7
Description and Exploit Analysis 8
Exploit/Attack Signatures 11
Platforms/Environments 14
Victim's Platform 14
Source Network (Attacker) 15
Target Network 15
Network Diagram 16
Stages of the Attack 16
Stage 1: Attacker is doing homework 17
Stage 2: Attacker installs the bots onto victim's computer 17
Stage 3: Bots drop payloads on vulnerable computers 18
Stage 4: Zombies connect to the IRC server 18
Stage 5: Attacker's commands are sent to zombies 18
Stage 6: Zombies blindly execute the commands 18
The Incident Handling Process 19
Preparation Phase 20
Existing Policy and Incident Handling Procedures 20
Incident Handling Resource and Team 21
Existing Countermeasures 22
Jump Kit Components 24
Identification Phase 24
Containment Measures 30
Eradication Phase 32
Recovery Phase 34
Lessons Learned Phase 35
Timeline of Handling Incident W32/SDBot.Wom 35
After Thoughts 38
References 40

Key fingerprint = AF19 FA27 2F94 998D FDB5 DE3D F8B5 06E4 A169 4E46 Author retains full rights.

© SANS Institute 2000 - 2005 Author retains full rights.

Executive Summary

This paper is to fulfill the practical requirements of GIAC/GCIH version 4.0. It uses the provided assignment template to address an ever rising and substantially dangerous threat to the whole Internet community: ***malicious bots***. This paper presents an introduction to malicious bots, which can be carriers of Trojans, worms, and virus to infect Internet hosts, and be remotely controlled by attackers via private IRC channels. By providing a case study on a particular variant of IRC-Sdbot, I am going to analyze in detail the characteristic, behavior and impact of this threat in a fictional and medium sized financial institution (the GIAC Enterprises) environment. It further demonstrates how the financial institution successfully handled the incident.

© SANS Institute 2000 - 2005, Author retains full rights.

© SANS Institute 2000 - 2005 Author retains full rights.

Statement of Purpose

Nowadays many companies in the world have profoundly realized the need for virus protection in e-mail servers and web traffic. They put many counter measures in place for these two particular applications. In the meantime, computer virus and worm writers (let's call them hackers) have certainly noted this fact and started to produce virus/worms that use other channels to bypass e-mail and web traffic protections. A striking example is that a hacker can install bots[1] on multiple computers to set up *botnets*[2] as a means of delivering virus, worms and even a distributed denial of service (DDoS) attack on selected targets that overwhelm victimized systems' defenses.

According to the security software company Symantec's threat report [1], the first half of 2004 saw a huge increase in zombie (infected bot) hosts on the Internet. The average numbers monitored by Symantec rose between January and June from under 2,000 to more than 30,000 per day - peaking at 75,000 on one day.

In most cases, malicious bots spread using Internet Relay Chat (IRC), Instant Messaging, and peer-to-peer file-sharing networks. Although these channels can be blocked at the gateway level firewalls by security conscious administrators in accordance with the company's policy, it is by no means to prevent employees from using these kinds of applications with working laptops outside company's perimeters. Furthermore, with the proliferation of VPN connections to remote offices and business partners, the trusted employees' and contractors' affected zombie desktops can become a serious threat to the security of the corporate internal networks.

This is one of biggest concerns that the GIAC Enterprises direction comes up. As an incident handler, in order to help the company winning the battle in confronting with the *bot* threats, I decided to examine this concern with a particular bot attack incident that the company's incident handling team dealt with recently.

For preparing this assignment, I begin with by examining what is malicious *bot* threat by presenting its general characteristics and behaviors. I then present a detailed analysis on a particular variant of IRC_SDBot called *W32/SDBot.worm*. Early when the incidence occurred, the employee, the owner of the infected laptop did not, in fact, aware that his laptop was infected by a worm. On his computer the anti-virus software and its signature dat files were updated.

[1] Generally speaking, a bot is a program that operates automatically as an agent for a user or another program.

[2] A **Botnet** [Zombie Network] is a group of computers (Zombies) that have been compromised by malicious bots under a common command and control (C&C) infrastructure.

© SANS Institute 2000 - 2005 Author retains full rights.

However, the bot worm was not detected either by anti-virus software or by company's internal intrusion detection system. It was discovered by company's house-made intrusion detection script over the VPN log database. It is scary signal for the company in terms of security. So, I will, in chapter Incident Handling Process, present how we handle the incidence in a proactive way and what we learned from it.

Background Knowledge

Since IRC_SDBot is a generic term for one kind of malware, before commencing the exploit of a specific incidence of such a threat, I think that it is necessary to introduce some common understanding and background on what are bots and botnets.

IRC Network

IRC (Internet Relay Chat) is a protocol defined 1982 in Oulu, northern Finland. With this protocol it is possible and not difficult to set up servers to form complex networks spreading all over the world to provide chat services for a big number of users. Within the IRC network a user can log in without a registration as it does in ICQ, IM (Instant Messaging), MSN or similar networks. He/she can open new separate channels to talk within. He/she is even able to take control over the channels by restricting other's access, as every user can be an operator of a channel.

Bots & Botnets

Initially a bot is a very useful feature of IRC servers, which allows server operator or client to make scripts for automatic actions in response to activities on the IRC channel. Because the IRC Network is very open and easily usable, it can be often used by computer program too. Programs installed on different computers can log onto an IRC server, open new channels and communicate through these channels. These programs are called *bots* (an abbreviation of *Robots*). There are a numbers of bots that serve for various purposes, e.g. FTP messaging, file sharing synchronization, etc. So these are bots "harmless bots".

Unfortunately, since the protocol is open to every one, malicious persons can use this openness for their malicious purposes, such as, coordinating SPAM delivery, launching virus/vorms and DDoS attacks, gaining financial advantages, etc. These are called "malicious bots". They can spread wildly. In this paper, I use the term *bot* to imply the malicious one.

In the literature, we can find a number of bot famililies [1]:

GaoBot, RBot, SDBot, AgoBot, PhaBot, etc. There are thousands of variants for these Bots (see [18] for a general classification of bot types, and see web pages

© SANS Institute 2000 - 2005 Author retains full rights.

of Anti-Virus software vendors, such as Symantec, McAfee, Sophos, etc. for more detailed description on specific bots.). The bot W32/SDBot.Worm addressed in this paper belongs to SDBot that is said the father of many other bots, like Rbot, RxBot, UrBot … It is written in C programming language under GPL (General Public License). More than 5 its derivatives are listed in Sophos' "Latest 10 virus alerts" at http://www.sophos.com/virusinfo/topten/.

A *Botnet* aggregates computers that have been compromised by malicious bots, allowing them to be remotely controlled by hackers. They are highly evolved versions of DoS tools and remote-control bots that hackers developed in the late '90s. So, it is not a new concept to the security community. The only difference is that the size and functionality of today's botnets is more scalable and more dangerous [2]. Instead of controlling a few hundred machines, today's botnets can control up to 25,000 zombies. Hackers are using them not just to crash target networks, but also to send spam and generate click-throughs to ad-laden porn sites.

Bot Distribution and infection Techniques

Hackers typically send their malicious bots to many computers at one time. The bots then automatically infect the machines that have the backdoors or other vulnerabilities that the bot software was written to exploit via virus, worm, or Trojan horse components (often called payload). There are several ways an attacker can distribute the infected bots:

- Take advantage of system vulnerabilities such as software bugs, including those that enable buffer overflow attacks, hacker-installed backdoors, and various memory-management problems that allow malicious code to infect a system.
- Use a list of common usernames/passwords to gain access to password-protected administrative shares such as D$, E$, IPC$, Print$ and Admin$.
- Use E-mail attachments with mass-mailing worms. In addition, hackers can send bots via Internet relay chat (IRC) file-transfer mechanisms or other means to victims' potentially vulnerable TCP/IP ports.
- Hack Web sites and install bots that can infect surfers' vulnerable browsers. For example, hackers can attack buffer-overflow vulnerabilities in Web servers, changing HTML pages' header and footer information to include scripts. Visiting browsers activate the scripts, which cause the browser to download a bot.
- When installed, the bot will attempt to connect to an IRC server on predefined port, usually port 6667, could be 6660 - 7000.
- Use encryption to evade the IDS detection

Facilitating Facts:

© SANS Institute 2000 - 2005 Author retains full rights.

The following facts facilitate the dramatic growth of botnets over the past year:

- Bots are open sources that allow attackers to easily create new ones with malicious purpose by customizing the source codes.
- The open nature of IRC allows the introduction of malicious codes.
- With high-speed broadband Internet connections, a lot of home-computers can be always turned on. This enables hackers to spread bots widely and quickly.
- The broadband connections make it easier for attackers to both install bots on victim computers and control them at their convenient times.

The Exploit

Now it is time to look at a specific variant of IRC_SDBot we will deal with in this paper. This is a case study on a real incident happened in the GIAC Enterprises. This incident is called "blended" bots incident because it includes both bots *W32/Sdbot.worm* and *Proxy-FBSR* named by McAfee Inc.

It should be noted that in our case, when the incident broke out, the company's Intrusion Detection System (IDS) detected the two above-mentioned bots on the laptop of one of employees on December 6, 2004. These bots were trying to spread over the company's internal networks and to make the possibly infected hosts to join a botnet on the Internet.

Exploit Name

In the following sections we concentrate on both bots ***W32/Sdbot.Worm*** and ***Proxy-FBSR***.[3]

These exploits in question can take advantage of the following vulnerabilities to propagate across networks:

DCOM RPC vulnerability - http://www.microsoft.com/technet/security/bulletin/MS03-026.mspx
WEBDAV vulnerability - http://www.microsoft.com/technet/security/bulletin/MS03-007.mspx
LSASS vulnerability - http://www.microsoft.com/technet/security/bulletin/MS04-011.mspx
Microsoft Messenger Service Buffer Overrun Vulnerability - http://www.microsoft.com/technet/security/bulletin/MS03-043.mspx
Microsoft SQL Server 2000 or MSDE 2000 audit - http://www.microsoft.com/technet/security/bulletin/MS02-061.mspx

[3] These names are later named by McAfee Inc after the outbreak of the incident addressed in this paper.

© SANS Institute 2000 - 2005 Author retains full rights.

The filenames detected in this incident include:

C:\WINNT\system32\vdddwq.exe (W32/SDBot.Worm)
C:\WINNT\SYSTEM32\vssddfq.exe (Proxy-FBSR)
C:\WINNT\system32\svchos.exe - (looks like a legitimate one but without a "t" ! !)
C:\WINNT\system32\winupdate.exe (looks a valid for UpdaterUI.exe)
C:\WINNT\system32\system32.exe
C:\WINNT\system32\taskmanager.exe (looks like to TASKMgr.exe)
C:\WINNT\system32\ravmond.exe

We note that the attacker using these Bots has some concerns on how to avoid being detected by possible defense measures implemented on the target network. Some of these bots look similar to other legitimate Windows executable names. As such, a user or system administrator viewing the Task Manager might assume that the names listed are valid.

Operating System

Like many other bots, *W32/Sdbot.Worm* exploits rely on flaws and bugs in Windows systems, Internet Explorer, and other popular applications to slip into your system. These systems are:

Windows 95
Windows 98
Windows Me
Windows NT
Windows 2000
Windows XP
Windows 2003

Protocols/Services/Applications

These are protocols/services/applications involved in this paper:

- Protocol: IRC Protocol, RFC 2810 (see [5])
- Services: corporate Networking and VPN VPN services
- Applications: Email, HTTP

Exploit Variants

The variant list of SDBot is huge. According to McAfee Inc. there are 4000 variants for W32/Sdbot.Worm only by August 2004 (see

© SANS Institute 2000 - 2005 Author retains full rights.

http://vil.nai.com/vil/content/v_100454.htm). In our particular case, the closely relevant variants are shown as follows:

Variants of W32/Sdbot.worm:

W32.HLLW.Donk (Symantec)
W32/Sdbot.worm.gen
W32/Sdbot.worm.gen.b

Variants of *Proxy-FBSR*

Backdoor.Ranky (NAV)
Bck/Ranck (Panda)
Troj/Ranck (Sophos)
TrojanProxy.Win32.Ranky (KAV)
Win32.Ranck (CA)

Description and Exploit Analysis

Detection.1: ***W32/Sdbot.Worm***

This detection is for worms that are based on the IRC-Sdbot Trojan code. The source code for the IRC-Sdbot trojan was published on the Internet some time ago, and a number of worms are based on the same code. The following detections exist for such worms:

W32/Sdbot.worm
W32/Sdbot.worm.gen
W32/Sdbot.worm.gen.b

These worms typically spread via network shares and create a remote access point for attackers to exploit.

According to our statistics over the VPN logs (see Listing 1) and Fport output from the victim's laptop (see Listing 2), we think that It was most likely intended to take advantage of high profile exploits:
http://www.microsoft.com/technet/security/bulletin/MS04-011.mspx

Port	Count
445	3012
139	3399

Listing 1 Ports 445 and 139 used by worm over a period of 3 hours

© SANS Institute 2000 - 2005 Author retains full rights.

```
C:\Documents and Settings\Victim\Desktop\Fport-2.0>fport
FPort v2.0 - TCP/IP Process to Port Mapper
Copyright 2000 by Foundstone, Inc.
http://www.foundstone.com
Pid   Process          Port  Proto Path
416   svchost       -> 135   TCP   C:\WINNT\system32\svchost.exe
8     System        -> 139   TCP
8     System        -> 445   TCP
880   MSTask        -> 1026  TCP   C:\WINNT\system32\MSTask.exe
1768  vdddwq        -> 1315  TCP   C:\WINNT\system32\vdddwq.exe
8     System        -> 1416  TCP
1128  javaw         -> 2001  TCP   C:\PROGRA~1\xpoint\SAS\jre\bin\javaw.exe
8     System        -> 3819  TCP
8     System        -> 4157  TCP
1768  vdddwq        -> 4244  TCP   C:\WINNT\system32\vdddwq.exe
1768  vdddwq        -> 4245  TCP   C:\WINNT\system32\vdddwq.exe
1768  vdddwq        -> 4246  TCP   C:\WINNT\system32\vdddwq.exe
1768  vdddwq        -> 4247  TCP   C:\WINNT\system32\vdddwq.exe
1768  vdddwq        -> 4248  TCP   C:\WINNT\system32\vdddwq.exe
1768  vdddwq        -> 4249  TCP   C:\WINNT\system32\vdddwq.exe
1768  vdddwq        -> 4250  TCP   C:\WINNT\system32\vdddwq.exe
1768  vdddwq        -> 4405  TCP   C:\WINNT\system32\vdddwq.exe
1768  vdddwq        -> 4406  TCP   C:\WINNT\system32\vdddwq.exe
…..
1768  vdddwq        -> 4441  TCP   C:\WINNT\system32\vdddwq.exe
1092  xpclient      -> 7777  TCP   C:\PROGRA~1\xpoint\EEClient\xpclient.exe
1128  javaw         -> 8200  TCP   C:\PROGRA~1\xpoint\SAS\jre\bin\javaw.exe
1128  javaw         -> 8201  TCP   C:\PROGRA~1\xpoint\SAS\jre\bin\javaw.exe
1128  javaw         -> 8500  TCP   C:\PROGRA~1\xpoint\SAS\jre\bin\javaw.exe
1044  XPAGENT    -> 8700  TCP   C:\PROGRA~1\xpoint\agent\XPAGENT.EXE
1028  xpadmin       -> 8886  TCP   C:\PROGRA~1\xpoint\xpadmin\xpadmin.exe
1728  vssddfq       -> 20721 TCP   C:\WINNT\SYSTEM32\vssddfq.exe
8     System        -> 137   UDP
8     System        -> 138   UDP
8     System        -> 445   UDP
1128  javaw         -> 1041  UDP   C:\PROGRA~1\xpoint\SAS\jre\bin\javaw.exe
2660  IEXPLORE   -> 2956  UDP   C:\Program Files\Internet Explorer\IEXPLORE
.EXE
1128  javaw         -> 3001  UDP   C:\PROGRA~1\xpoint\SAS\jre\bin\javaw.exe
```

Listing 2 Fport output on victim's laptop

Within LSASS, there is a component called LSASRV.DLL that will facilitate the exploit to allow for system control via the buffer overrun (see http://www.cultdeadcow.com/cDc_files/cDc-351/page2.html). In order for ***W32/Sdbot.Worm*** to take control of a vulnerable system, the worm must attack an error in LSASS debug log processing. This is accomplished when an infected remote system locates a vulnerable system via TCP port 445.

© SANS Institute 2000 - 2005 Author retains full rights.

Once exploited successfully ***W32/Sdbot.Worm*** will allow for full control over the vulnerable remote system. It then spreads via network shares using NetBEUI functions to get available lists of user names and passwords. It then drops a copy of itself on accessed shared folders.

The worm disables default admin shares (such as C$, D$, and Admin$) on WinNT/2K/XP systems by setting two registry key values:

- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\ lanmanserver\parameters "AutoShareServer" = DWORD:0
- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\ lanmanserver\parameters "AutoShareWks" = DWORD:0

A registry key is set to disable the enumeration of shares during a null sesssion:

- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\ Lsa "restrictanonymous" = DWORD:1

Another registry entry is added to start the worm each time the infected machine is restarted:

HKEY_CURRENT_USER\Software\Microsoft\Windows\ CurrentVersion\Run "Services" = *C:\WINNT\System32* \ vdddwq.exe

Again, according to our log analysis, this worm can perform a denial of service (DoS) attack against random IP addresses. Over three hours, it generated 25,356 events over 265 different TCP/UDP ports.

Detection.2: ***Proxy-FBSR***

This detection is for a malicious bot intended to serve as a proxy on the victim machine. The proxy Trojan acts as a middleman between a requesting system and a destination host. It is designed to listen on a specified TCP port for incoming requests. Those requests are then sent out from the infected system to the desired destination. The response from the destination server can be re-routed back to the originating host by the proxy Trojan.

This proxy bot allows for a Trojan author/distributor to use the infected system as a type of identity shield, allowing them to navigate to different locations on the Internet without divulging who or where they really are.

Such a proxy can be used to surf the web anonymously, hack systems, or relay spam.

There are multiple versions of this Trojan proxy - the details below are specific to such a variant. Exact details such as filename, Registry key name, and file size will vary.

© SANS Institute 2000 - 2005 Author retains full rights.

Upon execution, a port is opened for listening on the victim machine - the exact port is likely to vary in different variants. In our case the bot used port 20271 (See Listing 2: Fport output).

A Registry entry is added to start the proxy again each time the infected machine is restarted:

HKEY_CURRENT_USER\Software\Microsoft\Windows\
CurrentVersion\Run "Services" = *C:\WINNT\System32* \ vssddfq.exe

Trojans do not self-replicate. They are spread manually, often under the premise that the executable is something beneficial. Distribution channels include IRC, peer-to-peer networks, newsgroup postings, email, etc.

It is worthy to note that the chance of success for this Trojan in an environment depicted in Figure 1 is very slim in our particular case, because the attacker may does not know what are the GIAC Enterprises' proxy servers IP addresses and how they function. However, if the attacker is an internal and knowledgeable employee, he/she can modify the proxy Trojan so that it allows the zombie computers infected by ***W32/Sdbot.Worm*** get pass through Corp's proxy and talk or accept the commands from their masters outside the company. This will form a botnet over which the company's security is at great risk.

From above analysis, a combination of the above-described bots could become a powerful attack machine against the company's multilayer defense.

Exploit/Attack Signatures

Like other IRC_bot.Worms traffic, it is not very difficult to detect ***W32/Sdbot.worm*** since it has some strong indications by looking at closely their traffic or examining the log files. For example, some outbound traffic on TCP port 6667-6670 could be an indication of communication between bots and IRC servers. As another example, excessive connections to the internal hosts on TCP port 445 can be an indication of Windows system vulnerability scan or worm propagation (please refer to Listing 3).

Bot name	Possible indications
W32/Sdbot.worm	- Antivirus software alert on infected files (in our case, at its first occurrence, the Antivirus software did not pick them up) - Port scan activities (e.g. http, and open windows shares) - Outbound traffic on 6660-6670 - Increased network traffic in both

© SANS Institute 2000 - 2005 Author retains full rights.

	incoming and outbound directions - Applications and programs appear running slowly - Systems crash without knowing real reasons - Excessive outbound TCP ports 445 and 139 traffic from a single source IP - Unknown processes running
Proxy-FBSR	- Antivirus software alert on infected files (in our case, at its first occurrence, the Antivirus software did not pick them up) - Actual communication traffic between a server and clients - Unusual and unexpected ports open - Increased outbound network traffic from a single source IP - Applications and programs appear running slowly - Systems crash with unknown reasons - Unknown processes running

Listing 3 Attack indications

There is one interested point in terms of detecting both bots W32\SDBot.Worm and Proxy-FBSR. It is that many RealSecure SiteProtect network sensors (see www.iss.net) have been deployed at the network perimeters of the GIAC Enterprises. These are well-known and commercial IDS products. However, at the earlier stage of the incident, there were no particular relevant events that alerted by these sensors or attracted the attention of the IDS analysts on monitoring duty. The alert was first coming from a house-made Perl script which looks into the VPN log database. In our case, a few lines in Listing 4 triggered the email alert to the company IDS analysts. This script played a remarkable “dog-guard” role in our case.

From the listing 4, the reader can find some alert triggering conditions that can be served as the signatures for detecting IRC_SDBot. Two of them are particularly useful in our case:

1) From one single source IP, the triggered events contained more than 100 distinct destination IP addresses within one hour, or
2) The events used “TCP SYN” as protocol and {(6665 – 6669) or (6777)} as destination ports.

© SANS Institute 2000 - 2005 Author retains full rights.

```
SELECT
      *,
      (
            SELECT TOP 1 t.MsgUser + ';' + t.MsgGroup + ';' + CONVERT( VARCHAR(32),
t.MsgDateTime, 120 )
            FROM Tunnel t
            WHERE t.MsgStatus='assigned' AND t.MsgSIP=s.Source
            AND t.MsgDateTime<s.EndDate
            ORDER BY MsgDateTime DESC
      )
FROM
      (
            SELECT TOP 10
                  MsgSIP AS Source,
                  MsgHostName AS GatewayHost,
                  MIN(MsgDateTime) AS BeginDate, MAX(MsgDateTime) AS EndDate,
                  COUNT(*) AS CountProbes,
                  COUNT(DISTINCT MsgDIP) AS CountTargets,
                  MsgProto AS Protocol,
                  MsgDP AS TargetPort
            FROM
                  syslogd
            WHERE
                  MsgDateTime>='$startTime'
                  AND NOT ( MsgProto='udp' AND MsgDP=5632 )
            GROUP BY
                  MsgHostName, MsgSIP, MsgProto, MsgDP
            HAVING
                  COUNT(DISTINCT MsgDIP)>=100
                  OR ( COUNT(DISTINCT MsgDIP)>=20 AND MsgProto='tcp syn' AND ( MsgDP=1214
OR MsgDP=4661 OR MsgDP=4662 OR ( MsgDP>=6346 AND MsgDP<=6348 ) ) )
                  OR ( MsgProto='tcp syn' AND ( MsgDP=6777 OR ( MsgDP>=6665 AND
MsgDP<=6669 ) ) )
                  OR ( MsgProto='tcp syn' AND MsgDP=25 AND COUNT(DISTINCT
SUBSTRING(MsgDIP, 0, 6) )>=5 )

            ORDER BY
                  COUNT(*) DESC
      ) s
```

Listing 4 IRC_SDBot Detection Script

To give a bit hint on what log could contain in our case, I show a very very small portion of the VPN log entries on our bots detection in Listing 4. As you see, this portion of data was produced within less one minute. The logging date is December, 6, 2004.

Time	Src-IP	Src-Port	Dest-IP	Dest-Port
14:04.0	172.30.207.10	15000	69.158.142.28	139
14:04.0	172.30.207.10	15000	69.158.142.28	139
14:05.0	172.30.207.10	15010	69.158.150.43	5554
14:05.0	172.30.207.10	15011	69.158.150.43	80
14:05.0	172.30.207.10	15026	69.150.240.170	445
14:05.0	172.30.207.10	15027	69.150.240.170	139
14:05.0	172.30.207.10	15028	69.16.32.197	135
14:05.0	172.30.207.10	15029	69.52.236.198	445
14:05.0	172.30.207.10	15030	69.52.236.198	139
14:05.0	172.30.207.10	15031	69.150.240.170	1025
14:05.0	172.30.207.10	15040	69.84.94.239	445

© SANS Institute 2000 - 2005 Author retains full rights.

14:05.0	172.30.207.10	15045	69.84.94.239	1025
14:05.0	172.30.207.10	15002	69.175.132.48	135
14:05.0	172.30.207.10	15003	69.175.132.48	3410
14:05.0	172.30.207.10	15004	69.158.126.86	135
14:05.0	172.30.207.10	15005	69.175.132.48	5554
14:05.0	172.30.207.10	2864	68.64.46.217	139
14:05.0	172.30.207.10	15021	69.158.142.237	135
14:05.0	172.30.207.10	15022	68.159.127.248	139
14:05.0	172.30.207.10	15023	69.158.142.237	1025
14:05.0	172.30.207.10	2876	68.201.175.147	139
14:05.0	172.30.207.10	15000	69.158.196.192	1025
14:05.0	172.30.207.10	15000	69.158.196.192	135
14:05.0	172.30.207.10	15000	68.209.123.40	139
14:05.0	172.30.207.10	15000	69.158.142.28	1025
14:05.0	172.30.207.10	15000	69.158.142.28	6129
14:05.0	172.30.207.10	15000	69.158.142.28	3410
14:05.0	172.30.207.10	2972	68.78.209.209	139
14:05.0	172.30.207.10	2973	69.158.142.76	135
14:05.0	172.30.207.10	2974	69.158.142.76	1025
14:05.0	172.30.207.10	2975	69.158.142.76	445
14:05.0	172.30.207.10	2976	68.70.240.36	139
14:05.0	172.30.207.10	2977	68.86.52.113	139
14:05.0	172.30.207.10	2978	69.251.125.172	135
14:05.0	172.30.207.10	2979	69.251.125.172	1025
14:05.0	172.30.207.10	2980	69.251.125.172	445
14:05.0	172.30.207.10	2981	69.251.125.172	6129
14:05.0	172.30.207.10	15000	69.111.25.62	135
14:05.0	172.30.207.10	15000	69.111.25.62	3410
14:05.0	172.30.207.10	15000	69.111.25.62	5554
14:05.0	172.30.207.10	15000	69.158.142.43	445
14:05.0	172.30.207.10	15000	69.158.142.131	1433
14:05.0	172.30.207.10	15000	69.158.142.131	6129
14:05.0	172.30.207.10	15000	69.158.142.131	3410

Listing 5 Excerpt of VPN Log

Platforms/Environments

As shown in Figure 1, in the GIAC Enterprises, there are thousands of servers and desktops with Windows platform from different manufactures, such HP, IBM, and Dell. The OS are Windows 2000, Windows XP, and Windows 2003. A lot of machines have IE 6.0 and MS SQL installed on them.

Victim's Platform

© SANS Institute 2000 - 2005 Author retains full rights.

Victim's IBM ThinkPad T21 is set up with a direct connection to a cable modem with no firewall protection from the Internet. His computer runs Windows 2000 with Service Pack (SP) 4 and IE6.0 with SP1.

Possible victims could be any computers including desktops and servers running Windows as OS on the interconnected corporate networks.

Source Network (Attacker)

The source network includes both attacker(s) and victim. Both of them obtained an IP address from their own ISPs. The attacker's IP address was obtained via ISP's DHCP server and within a C class network of 24.150.22.0/24.

In order to attack the corporate internal networks, the attacker found a way with malicious bots to infect an employee's laptop connecting on the Internet. Because his job requires, this employee like some of other company's remote employees needs to access to the target corporate network from time to time. The remote access from home to the corporate networks is via a VPN tunnel. The login authentication over the VPN is through RSA's Security ID. Once an employee gains the access through VPN, his/her computer will be assigned an IP address of 172.30.207.0/24 range by the VPN server.

Target Network

The target network includes both victim one and corporate's ones. The latter can become victims if the former could successfully spread worms or Trojan onto the them on the target network. Note that at this time, the victim was on the target network – corporate one via VPN tunnel.

As shown in Figure 1, the GIAC Enterprises uses firewall systems and routers' access control lists to protect itself from the Internet.

In order for an employee outside the company to remotely access to the company's internal networks, he/she has to go through VPN tunnel by using SecureID authentication from RSA Security Inc. (http://www.rsasecurity.com).

The communications between Toronto Headquarter and Ottawa R&D take place within leased T3 line. The traffic is encrypted according to the sensitivity of the transferred data.

In addition, because of the confidentiality of the data the company deals with, a second line of defense is established by sub-zoning the whole internal network into several zones, e.g. between the general internal service operation environment and public service zone (DMZ). Between these sub-zones, firewalls are used to make more granular and appropriate controls over the traffic flow

© SANS Institute 2000 - 2005 Author retains full rights.

(see Figure 1). Furthermore, some IDSs have deployed at the perimeters of the corporate networks (i.e., Tor-Internal-Net and Ott-Internal-Net).

The whole company's network is a B class network 172.16.0.0/16. All company's subnets are divided from this B class network.

Network Diagram

A high level and logical network diagram of the GIAC Enterprises is shown in Figure 1.

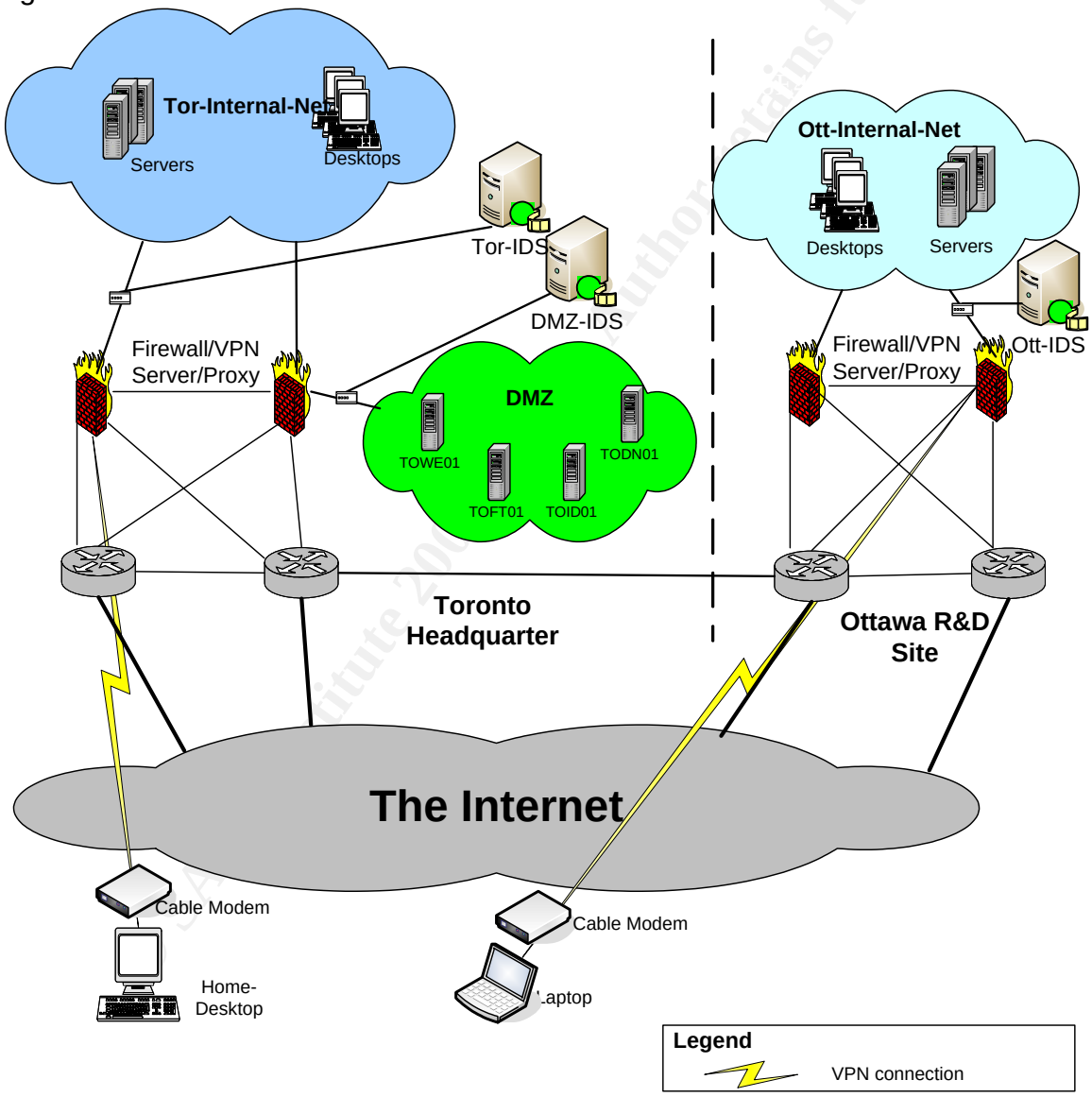


Figure 1: Attacking Network Diagram

Stages of the Attack

© SANS Institute 2000 - 2005 Author retains full rights.

In this section, I will provide a more detailed description of the events that transpired during a specific security incident involving W32/SDbot.Worm. First, I detail the way the SDBot.worm works according to the literature and log information on the incident. In order to help the reader to gain a good understanding, an illustrated graph for attacking stages is also provided.

Stage 1: Attacker is doing homework

In order to fulfill his/her ultimate goal to compromise the security of the GIAC Enterprises' network, an attacker needs to do some homework:

- Finding and customizing the appropriate bots to carry out virus, worms, and/or Trojan Horse as payloads.
- Selecting a victim who can bring the malicious bots onto the corporate networks. Here, we assume that the attacker him/her self is not an employee of the GIAC Enterprises.
- Gaining knowledge about target networks by using various reconnaissance, scanning, and even social engineering tools.

Stage 2: Attacker installs the bots onto victim's computer

After finishing homework, the attacker is facing a challenge to decide how to install the intended bots onto victim's computer (shown in Figure 2 as VictimA). He/she may:

- Send a email with attached bots to the selected targets
- Compromise the victim's computer by taking advantage of system vulnerabilities and install the bots on the computer without victim's knowledge. According to our investigation of the root cause on why the victim's computer is infected (see section Identification), we consider that this is most likely the way the infection happened on VictimA's computer. This is because the victim has never used IRC, and P2P on his laptop. However, with an interview with the victim and an after-incident assessment, we do find that the victim used a very simple password for his laptop's administrator account. Furthermore, a patch scan with MS Baseline Security Analyzer did find that on his laptop, there exists a LSAS vulnerability related to patch MS04-011.
- Hack web site and install the bots that infect victim's vulnerable web browser.
- Trick the victim into executing a malicious program that leads to bots installation.

© SANS Institute 2000 - 2005 Author retains full rights.

Stage 3: Bots drop payloads on vulnerable computers

Once the bots are installed on the victim's machine, they copy themselves to the desired directories and update the related registry keys (see section The Exploit). The bots can carry other malware as payload, such as virus, worm, and Trojan. These malware can further find other possible victims by using methods described in stage 2 in an automatic way.

Stage 4: Zombies connect to the IRC server

The compromised corporate computers become zombies, and can join now with VictimA. They could connect to the IRC server through Command & Control (C&C) channel waiting for further orders from the attacker. With multiple compromised zombies, a botnet under the control of the attacker is formed.

Stage 5: Attacker's commands are sent to zombies

Using the newly established botnet to download more attack tools, the hacker can comfortably login onto a corporate network to issue more commands to zombies even if they are behind a firewall. This is totally possible because most companies' firewalls allow inside-outside connections without much restriction. This way, the bots can effectively render company's firewall transparent to the attacker.

Note that some more sophisticated bots can render any zombie as a "master" or "commander" to do more scalable attacks FREELY for the attacker.

Stage 6: Zombies blindly execute the commands

If the botnet is successfully established, the hacker can take advantage of it, for examples, to distribute large quantities of spam, to launch DDoS attacks by sending large numbers of messages to the target network.

What a zombie can do is blindly executing the commands from its direct "master", just like a "slaver".

© SANS Institute 2000 - 2005 Author retains full rights.

© SANS Institute 2000 - 2005 Author retains full rights.

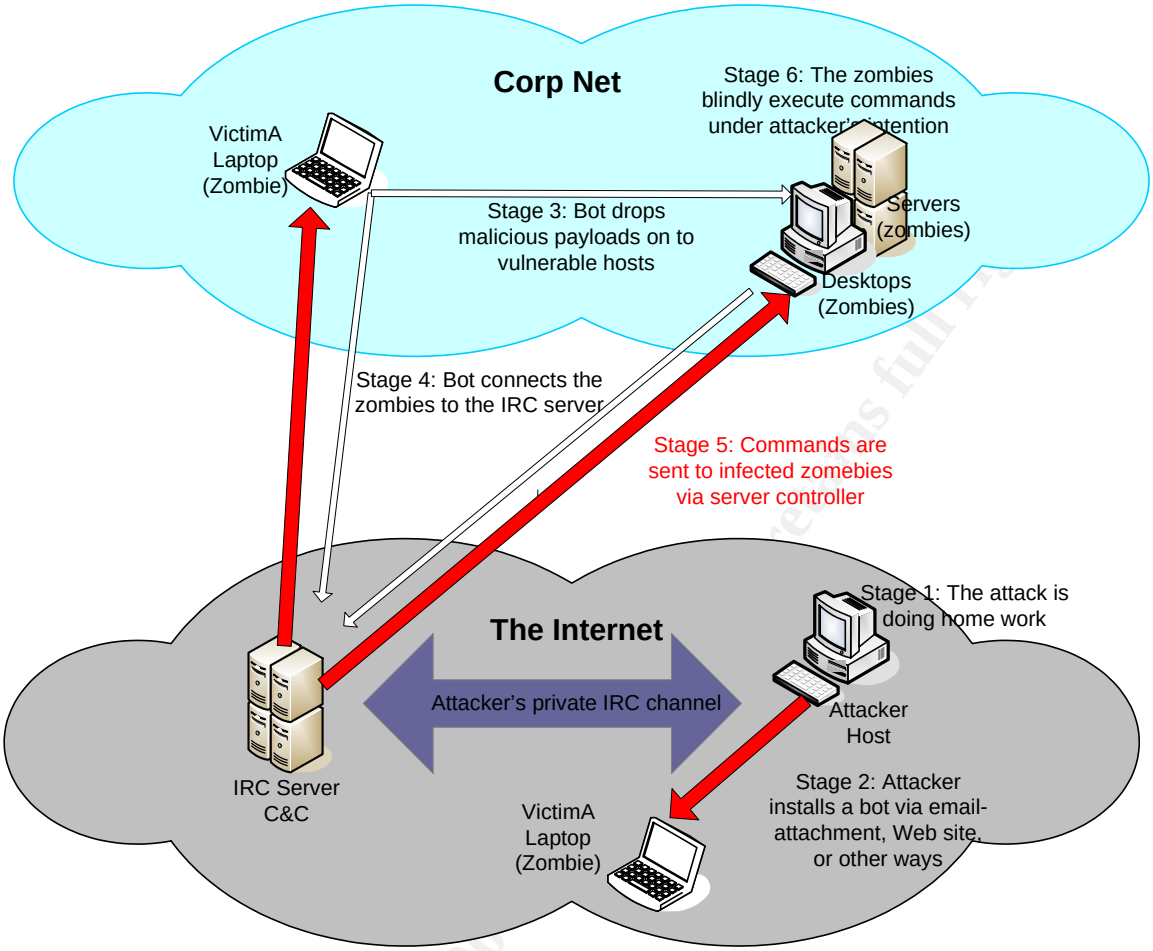


Figure 2: Illustrated Attack Stages

The Incident Handling Process

Having an established incident handling process is extremely important for the survival of an enterprise in today's cyber-computing world [13].

In this section, the application of incident handling methodology with six phases (Preparation, Identification, Containment, Eradication, Recovery, and Lessons Learned) on the process of responding to the security incidents will be demonstrated.

I profit the occasion of dealing with an actual incident case to examine the effectiveness of the existing incident handling process within the GIAC Enterprises. By applying the process to the real case, we can further check out its strength and weakness. I will, thereafter, make some improvement

© SANS Institute 2000 - 2005 Author retains full rights.

recommendations to the company in countering against bot threats to the company's networks in the last subsection of this part.

Preparation Phase

It is commonly acknowledged that this first step is also the most important step. Preparation means being ready to response to future incidents. The main underlying concerns of preparation are:

- Having a set of policies and procedures in place for dealing with incidents
- Obtaining necessary resources and personnel
- Establishing an infrastructure to support incident response activities.

Let's examine now how the GIAC Enterprises addresses these concerns.

Existing Policy and Incident Handling Procedures

The GIAC Enterprises direction clearly understands the first paramount need is to establish a security policy along with a comprehensive process in order to lay the ground rules for incident handling.

Although the full description of the policy is out of scope of this paper, it is worthy to outline some important issues that the policy contains:

- Clarify higher-level organizational objectives and rules for the governance of the incident handling process.
- Determine the resources including personnel involved in the process; hardware, software, and technologies used for incident response.
- Define resource ownership.
- Assign responsibilities to personnel of incident response team.
- Draw the guidelines on how to deal with incidents and how to escalate an incident according to its severity.
- Policy enforcement.

Based up the established policy, the existing major procedures involved in the whole incident handling process are as follows:

- Reporting Procedure
- Escalation Procedure
- Security Incident Response Procedure
- Security Incident Track Procedure
- Periodical Team Training and Awareness education
- Incident Monitoring procedure

© SANS Institute 2000 - 2005 Author retains full rights.

Although the GIAC Enterprises has established these incident handling procedures, and the incident response cases have been well documented over past a few years, the company realizes that it is far away from perfect in coping with the complexity of actually happened incidents. For this reason, the company decided to review the old process and procedures, and convert a pre-established Emergency Management Team (EMT) into a new Computer Security Incident Handling Team (CSIHT) as described in the next subsection.

Incident Handling Resource and Team

In accordance with the company's policy, the incident handling process should define the roles and responsibilities of groups and individuals who will be involved in the process. So, the first concrete action or step for us is to define these groups and individuals, and their responsibilities.

The whole enterprise computer security incident handling team (CSIHT) consists of three groups of people who playing important roles in incident handling process. All these groups are under the direction of CISO (Chief Information Security Officer).

EIRG (Enterprise Incident Response Group)

This group facilitates the detection and alerting of incidents within the whole GIAC Enterprises. It is also offered by high-level management to accomplish mission of incident response, including access to all organization resources and systems. It is constituted of a manager and a dozen of technical personnel with different security experience and expertise. The manager is accountable for the entire security incident response process. Each staff will undertake one or more duties listed as follows:

1) 24x7 IDS monitoring and incident alerting for the whole company
2) Constant threat tracking from security sources (mainly from the credible web sites on the Internet).
3) Assessing security vulnerabilities for different corporate networking zones.
4) Creating an enterprise-wide incident handling procedures.
5) Compiling and publishing timely the security vulnerability advisory pertaining to the GIAC environment.
6) Coordinating and guiding incident handling for different environments with local system administrators.
7) Proceeding forensics analysis in case of needs.
8) Perform research on mitigate technologies against the risks caused by various incidents.
9) Creating historical database on past incidents.
10) Assisting local security officers in handling incidents.
11) Ensuring all phases of incident handling are completed.

© SANS Institute 2000 - 2005 Author retains full rights.

As we see from above, one of key activities for EIRG is *coordination*. This is extremely important in view of the fact that today's computer security incidents are very distributed in their nature.

LSO (Local Security Officers)

This group of staff takes care of the systems and their security within their networking zones. LSO are technical personnel who could be security officers or system administrators. These people know their networking environments and security needs within their environments. Their duties in this process include:

1) Communicating security initiatives, best practice, policies and threats to their departments
2) Assisting incident investigation requested from EIRG
3) Participating the incident conference calls
4) Assisting incident victims in data recovery and service operation restoration
5) Analyzing, documenting and reporting to EIRG in detailed the incidents and their activities in the whole incident handling process

SM (Senior Managers)

The policy requires that each department must assign a SM being involved in the incident handling process depend up the severity and scalability of the incidents to handle. SMs' responsibilities include:

1) Responding within 24 hours to requests from the EIRG to assist in the analysis of incidents and development of a suitable response.
2) Calling and attending relevant meetings, as required; leading LSOs to determine the impact of incidents on the systems for which they are responsible.
3) Leading an instant response group from the technical resources through their departments in case of incidents.
4) Ensuring a post-mortem analysis and lessons learned.

Existing Countermeasures

a) Secure network infrastructure

The GIAC Enterprises' network was designed with the principle of defense-in-depth, i.e. the entire network is divided into different zones such as, external zone, DMZ zone, application zone, database zone and desktop zone. Firewalls are deployed between zones. In addition, both ingress and egress traffic are

© SANS Institute 2000 - 2005 Author retains full rights.

controlled by both perimeter routers and firewalls. Furthermore, all the connections from the internal network to the Internet have to go through a proxy server.

b) Intrusion detection

In order to build the second and even third line of defenses, people are increasingly using IDS (Intrusion Detection System) and IPS (Intrusion Protection System) to make the defense more solid and complete than that with only firewall in place.

An IDS helps the system administrator to see how their systems and networks are scanned, probed and possibly exploited, not only from outside the organization but also from inside. There are two kinds of IDS: network-based or host-based. For a comprehensive coverage for such a company like GIAC Enterprises, using both systems could be quite cost. For this reason, the company has decided to deploy network-based IDS as the first step to some critical points of the network. The next step is to deploy host-based IDS onto the critical servers.

Actually, company has deployed 30 network sensors and 20 host-based sensors around the key network entries to the Internet and on the very critical servers.

In addition, EIRG has created some scripts that look into the company's firewall/proxy server log databases. These scripts are extremely useful in detecting some known and rapidly spread worms. In the following section, the reader can see that the actual incident addressed in this paper was first alerted by these scripts, not by other commercialized IDSs.

EIRG monitors intrusion detection traffic through the consoles on a 24x7 basis.

c) Vulnerability assessment (VA)

By using various assessment tools, EIRG conducts vulnerability assessment on a regular basis to identify vulnerabilities seen from the internal networks and external networks. Thanks to these regular assessments, EIRG maintains a database that contains all previously pursued assessment results. This is a very valuable database from which an incident handler can get a lot of helps to his/her incident analysis and handling for a particular environment.

d) Anti-Virus

It is well known that the windows platform is a major target for many viruses. Since most of company's staff uses windows PCs, the company has established a particular policy to cope with this risk. This policy dictates that every

© SANS Institute 2000 - 2005 Author retains full rights.

workstation and windows servers should use anti-virus software signatures updates and virus scan should be automatically executed at least once a day.

In order to enforce this policy, the company has carefully chosen an Anti-virus software of enterprise version that allows the administrators in EIRG to initiate remote scanning and signature status checking to every workstations throughout the whole company. In case of any problem found by the central servers, such as an obsolete dat file on a computer, the servers can automatically push the new dat file onto the computer without knowing by its users.

Jump Kit Components

In case of incident, incident handlers can use a pre-prepared “Jump Bag”. The main items of the kit include the following:

Hardware
Laptop (dual boot Windows XP and Redhat Linux)
Knoppix CD
80 GB USB External Hard Drive
1GB USB Memory Sticker
3Com 8 Port Hub
2 cross-over cables
2 straight-through cables

Software
Ethereal
Windows Resource Kit
LogCollector Utility
Encase Forensics Tools Kit
PepiMK FileAlyzer
Symantec Ghost 6.5 Enterprise Edition

Other Items
An updated document on incidence handling and escalation procedures
Notebook and pens
The GIAC Enterprises phone list
Contact lists for each Lines of Business
Historical notes of Incidence handling cases over last 3 years
A list of managers and Security Officers in each department

Identification Phase

In this section, we will discuss the detection and identification of W32/SDbot.Worm and Proxy-FBSR attack.

This phase involves:

© SANS Institute 2000 - 2005 Author retains full rights.

a) Determine if an incident really occurs

On December 6, 2004, the IDS team of EIRG was as usual executing its IDS monitoring process. Later afternoon around 18:10, one member of IDS monitoring team in EIRG received an email alert (see Listing 6) from the company's firewall/VPN log analysis server.

```
SendTo:  ids_eirg@giac-entprises.com
Subject:  IDS Alert [2004-12-06 18:10]: Virus Infection on VPN: victimA/VPN#111

The following user is showing symptoms of virus infection.

IDS Alert [2004-12-06 18:10]: Virus Infection on VPN: victimA/VPN#111

Date/Time: from 2004-12-06 17:09 to 2004-12-06 18:06 (Eastern Time)
IP address: 172.30.207.10
VPN User ID: VictimA
VPN Group: VPN#111
Type: network worm (TCP 135,139,445,1433 scan) and IRC backdoor (TCP 6667 probes)

This is an automatically generated email.
```

Listing 6 The initial incident alerting email

According to his experience, the alerting source was credible because this alert was coming from a Perl program made by own team. Since this program was well tailed to the company's VPN environments, in the past, it had made several alerts without one false positive. Upon incident reporting procedure, after receiving this alerting message, the IDS analyst informed the incident handler on duty.

b) Perform preliminary assessment

When the incident handler on duty received the call from the IDS analyst, first thing he had to do is to perform a preliminary assessment according to the incident response process. Apart from the alerting information from the IDS analyst, he had to:

- Look into the source logs to have an overview of the incident,
- Look for other evidences from other sources, for examples, from other IDS systems. Remember that there are more than one kind of IDSs deployed in the GIAC Enterprises.
- Estimate the possible impacts to the infected environments so as to help an incident manager to prioritize the incident handling among other possible incidents

© SANS Institute 2000 - 2005 Author retains full rights.

- Make a preliminary action recommendation.

By taking above-mentioned actions, at 18:45pm, the incident handler finished a preliminary assessment as shown in Listing 7:

Incident name	Start time Incident name	Brief description of the Incident	Location/ Env.	Victim info.	Impact	Risk
Unknown for exact name; possible name worm, IRC connection	2004-12-6 14:0.0	Scanning the Corp net on tcp ports 139,445,1433 A faire amount traffic outbound traffic on ports 139,135,445,and 6667	Corp's VPN environment with VPN#111	Name: VictimA VPN IP: 172.30.207.10 He is actually working from home through VPN No other computers on Corp net are found infected	1. Could compromise company's Windows machines, 2. Could degrade the network performance	High: violation of the corporate policy on IRC usage

Listing 7 Preliminary assessment of the incident

In the meantime, he made a list of recommended actions shown in Listing 8.

Recommended actions:

1. Collect evidences from all related sources and logs and start an in-depth investigation and analysis by EIRG members
2. Ask for Remote Access Control Group to disable the VPN access of the affected user and notify the support group.
3. Contact the desktop support group and departmental LSO:
   a. ask the infected user stopping using the computer until EIRG allows
   b. Make an image of infected hard disk and ship it to EIRG
   c. Check that the anti-virus software, engine and DAT are up to date;
   d. Check that the Windows software patches are up to date;
   e. Perform a full antivirus scan and clean any virus found;
   f. Report actions taken and findings back to IH_EIRG@GIAC-Enterprises.com

Listing 8 Preliminary recommended actions on the incident

© SANS Institute 2000 - 2005 Author retains full rights.

c) Logging the incident and get system snapshot

Once the initial assessment was completed and a security incident declared. The first action taken is the establishment of an Incident ID to be used as the incident tracking number required for the incident handling process.

As well, with the helps of other team members, the incident handler started to create the timeline of the incident and to treat all notes, logs, events, etc., and to update the incident history database.

d) Escalation

At this stage, the incident handler needs to alert all the related parties. Before doing so, he/she must escalate the incident to EIRG manager for his/her support and approval on recommended actions.

Since the incident happened at a remote location and in a particular functional department of the company, according to the incident handling process, the incident handler of EIRG has to collaborate with local security officer (LSO) and SMs in pursuing further incident analysis and handling.

In order to show a clear picture what we did in this handling phase, a time-line table (see Listing 10) is shown in the final phase of the process.

e) Continuous Investigation

After the preliminary assessment was made and the event was escalated to the EIRG manager, the incident handler continued his investigation in depth. He knew that soon or later the management would need more information about the incident.

Note that the actions described below are not necessarily accomplished within Identification phase. They can be undertaken during the course of the incident handling process or even after for forensic analysis. I address these actions in this phase because I believe that as an incident handler, he/she needs often to ask himself/herself the following questions. The earlier are these questions answered, the more helpful for the incident handling phases that follow.

1) What are the worms exactly? Are they known in the literature? Why were they not detected by company's recommended Anti-virus software?

According to the company's VPN alerting scripts, the incident handler had a rough idea about what is the nature of the incident at the beginning: it is a kind worm. But, what is exactly? It is important because if we know exactly what, we can deal with it in an accordingly and efficient way. For this, after he obtained the viral binary files (there are vdddwq.exe and vssddfq.exe in our case) from the

© SANS Institute 2000 - 2005 Author retains full rights.

victim's laptop, the incident handler first went to McAfee's WebImmune page[4] to determine if these are two known worms. The result was negative. He then submitted these files to web site: http://virusscan.jotti.org. The scanning output confirmed that the one binary is a bot worm and the other a bot Trojan (see Figure 3 and Figure 4).

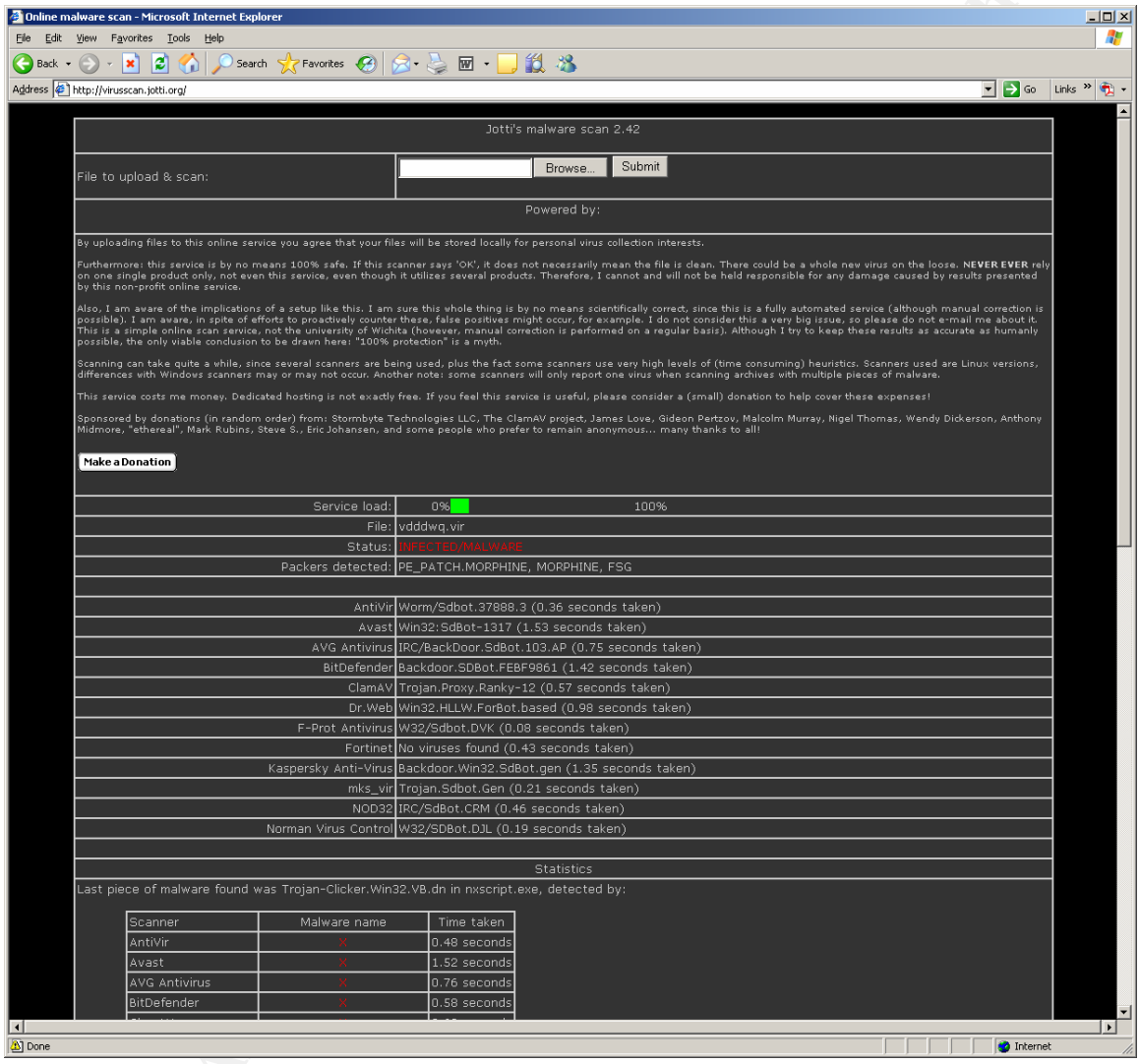


Figure 3 Malware scan for vdddwq.exe binary

[4] McAfee Anti-Virus software is the GIAC Enterprises' prefer one. The policy dictates that this software should be deployed on every Windows machine.

© SANS Institute 2000 - 2005 Author retains full rights.

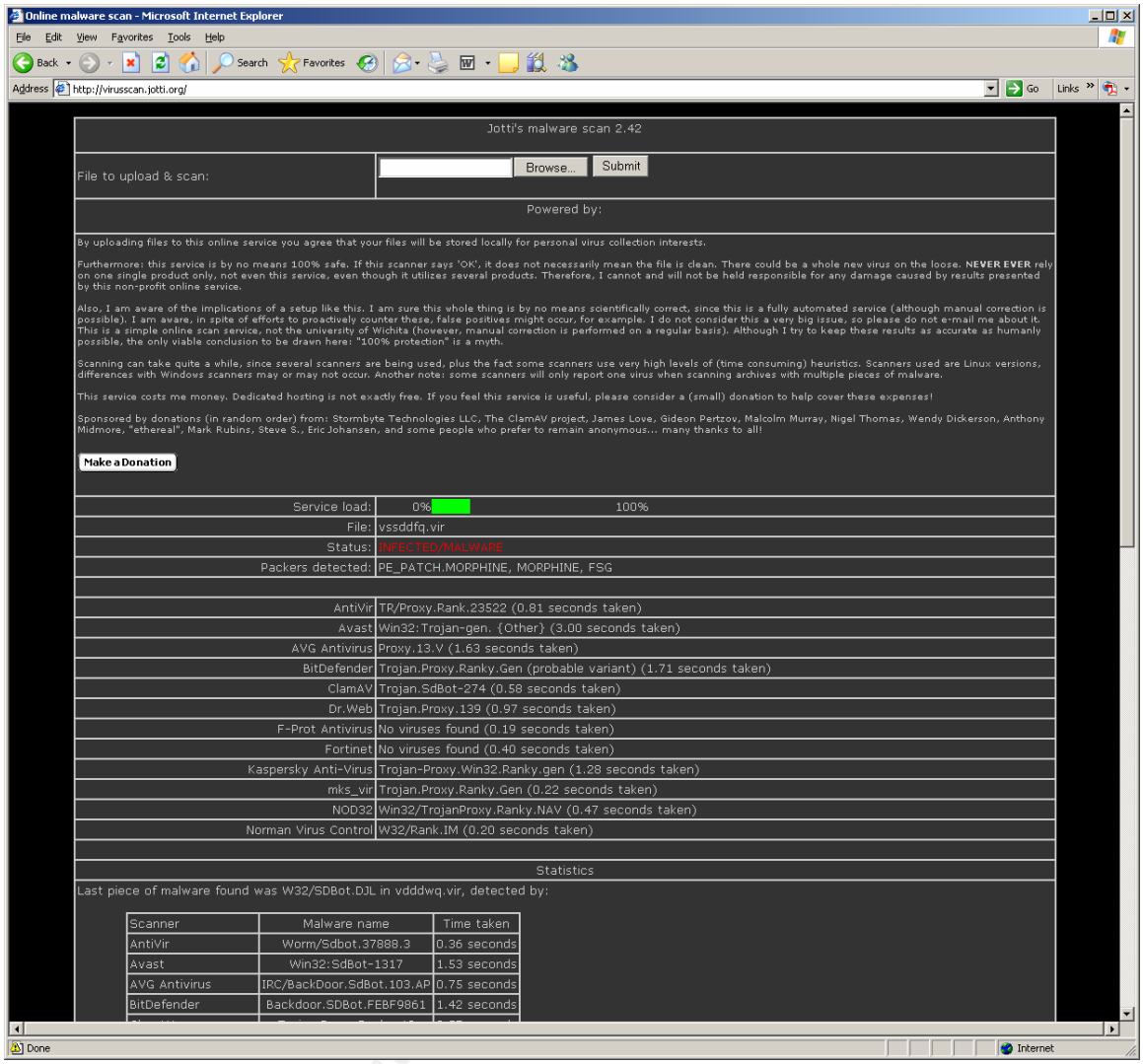


Figure 4 Malware scan for vssddfq.exe binary

Note that in order to avoid these two files to be deleted accidentally by the anti-virus software on the incident handler's computer, he changed the file's postfix "exe" into "vir".

2) Why it was infected?

Before we determine how the infection did not spread over the corporate networks, we needed to determine how the VictimA's laptop was infected in the first place. There is no definitive answer to this question. According to the assessment on VictimA's laptop and an interview with the victim himself, the incident handler knew the fact the victim did not use Chat, IM, P2P, or MSN with his laptop and he also did not receive any email with attachment before the incident outbroke. So, as mentioned earlier in section Stages of Attacks, the most

© SANS Institute 2000 - 2005 Author retains full rights.

possible answer is that VictimA was useing a very simple password which can be found in the word list embedded with the binary file vdddwq.exe [10].

3) What is the possible infection scope? And are they any other corporate servers or desktops are infected?

In order to find the answer to the former question, the incident handler dived into the VPN log database to find out what are distinct destination IP addresses and what are belong to the company's internal addresses. At this point, the incident handler found that the vulnerability assessment database (mentioned in the preparation phase of the process) is really helpful because this database contains also a very comprehensive list of the companies network addresses. So the answer to the question comes from a comparison between the address list pulled from the VPN log database and the one from the vulnerability assessment database (see Listing 9 for a statistic result).

Time period	Distinct Dest. IPs	# of Dest IPs also belonging to Corp's network IPs	Total number of events
3 hours	6,536	321	25,536

Listing 9 Destination IPs counts

In answering the latter question, it is sufficient to take a close look into the VPN log database and identify if there are new network scanning activities originated from the corporate network IP addresses. These IP addresses are matched in the destination address list from the VPN log database. The result was non.

Containment Measures

Containment measures are adapted depend up the magnitude of an incident pertaining to the actual incident and the organizational structure of the company. For example, at this stage the EIRG manager, the actual incident manager in charge needs quickly to decide who should be involved and what actions should be taken in the containment process. In our case, the incident manager and his staff had done the following.

a) Assign supportive personnel and responsibility

Since the infected computer was remotely located and only one, it is not a good idea to dispatch an incident handler in EIRG on site at end of working day. The manager asked one of his senior staff to be an incident coordinator whose role is to contact VictimA's LSO and to work directly with desktop support to help out VictimA to contain the infection on his laptop. In the meantime, the incident coordinator informed the incident information to

© SANS Institute 2000 - 2005 Author retains full rights.

network support persons in the concerned VPN networking environment. The network support people were immediately working towards attentive network traffic monitoring.

b) Limit the scope and magnitude

Since the incident was determined as a bot worm attack, the previous experience on worm incidents like Slammer and CodeRed told the manager that he must react very fast. So in order to limit worm spreading, he personally informed the Remote Access Control group to disable immediately the VictimA's VPN access to the corporate network.

Once the VictimA's laptop was cut from the network, the threat of further infection to the corporate networks was theoretically over.

To be more professionalism in following the incident handling process, the manager assigned some concrete tasks to one of his staff. The tasks were mainly:

- to get contact with the victim immediately;
- to interview the victim in order to obtain more information on why his laptop was infected;
- to inform him not using the computer until a desktop person coming.

To be sure there was no further worm spreading over the corporate networks, the manager held a conference call (see Listing 10) with attendance of all related parties. He updated them the incident status and asked everyone to keep an eye on worm spreading in their networking environments for at least one hour. Any suspicious worm activities should be immediately reported to the manager without any delay.

c) Protect critical resources

VictimA is a branch employee who has little technical knowledge of the computer. Because the nature of his job responsibility, his computer contains a lot of customer information. In order to ensure the data on his computer was not compromised or lost, a desktop support person was dispatched to VictimA's home next workday. The support person went there mainly for:

1. Making a full backup VictimA's laptop with the Jump Kit described in preparation phase
2. Using the Symantec Ghost to make disk image for further forensics analysis by EIRG
3. Running the system information tool (in Accessories). Under System Environment: click on each of the following items and extract the output (using File/Export) into a separate or text file:

© SANS Institute 2000 - 2005 Author retains full rights.

- Running Tasks
- Loaded Modules
- Services
- Startup Programs
- Program Groups

Note: I do not present all the outputs in this paper since they are too voluminous.

4. Run FPORT tool by Foundstone Inc. (see Listing 2 for the outputs).

Note again that all these actions are at request of the incident handler.

4) Determine operation status

Another task of the on-site desktop support person was to confirm the infection on VictimA's laptop, assess the operation status, and report back to the incident handler. Based upon his report, the incident handler nand his team will decide whether or not the VictimA's VPN access to corporate network allowed.

Before looking into the report, the incident handler reiterated the point that VictimA was not allowed to connect his laptop directly to the corporate network from anywhere, even he came to office next days. He is only allowed to do so until a special permission notice issued from EIRG after a vigorous reassessment (see the following subsections).

Eradication Phase

Now the infected laptop is off line and no longer threatens the corporate networks, the incident is contained. It is, therefore, the cleanup process begins.

The following two main actions were taken:

a) Get rid of the incident by applying patches/fix

The clean up of W32/SDBot.Worm was relatively easy. As mentioned before, at the beginning of the incident, the incident handler and his colleagues could not get rid of the found bots on VictimA's laptop by using McAfee Anti-Virus software. They have contacted the vendor. In view of the fact that the GIAC Enterprises is a big customer, McAfee immediately organized necessary resources to work out a new dat file to clean up the bot worms.

After obtaining the related dat file from McAfee, we were able to delete the bot programs as shown in Figure 5.

© SANS Institute 2000 - 2005 Author retains full rights.

After cleaning up the two particular bot worms, we had launched a thorough scan on the whole infected system with McAfee AntiVirus Enterprise 7.1 with new dat file.

To verify if all the affected files are indeed cleaned up, we went to the directories/folders indicated in section Description and Exploit Analysis to see whether the "unexpected" files was really gone. They were actually all gone. In the same way, we opened "regedit.exe" program to check whether the affected registry keys were still there. The checking result was that they were all gone.

At this point, some extra cares must be taken. It is not prudent to say that by running a thorough virus scanning, all infected files and registry keys are cleaned up. In reality, we could only checked for those we know. So, it is very subjective and depends up the knowledge on the incident itself and handler's experience on handling it. It could be some other hidden worm files or those spawned by the worms and Trojans. So, in many cases, according to the degree of infection and handlers' knowledge, we recommend the infected system owner just rebuild the system after ensuring no lost of production and customer data.

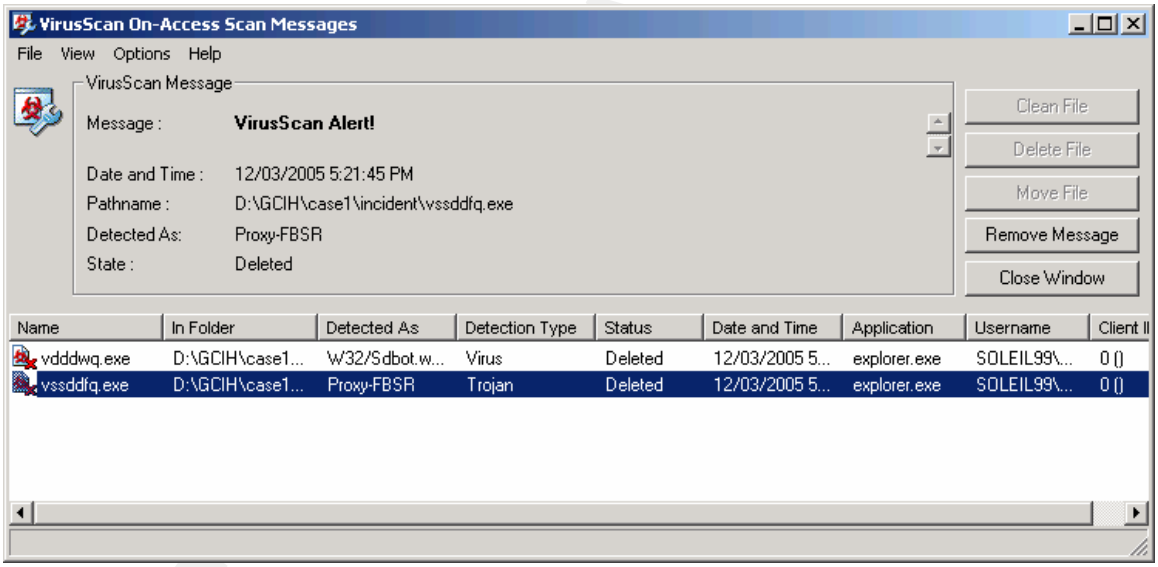


Figure 5 Bots eradication by McAfee Anti-Virus Enterprise 7.1

b) Correcting system mis-configuration and reseting password with a strong one

As mentioned before, there was a very high possibility that the VictimA's laptop was infected by the bot worms because he used a very weak password for his administrator's account. For this reason, the incident handler prepared a document about why and how one needs to choose a strong password for his machine and how to set up local policy for password enforcement on his computer.

© SANS Institute 2000 - 2005 Author retains full rights.

Another proactive action taken by the handler is to send him a document how to do patch update via the enterprise patch manage servers both on the corporate internal network and at home in a VPN tunnel.

Recovery Phase

Before the computer involved in the W32/SDBot.Worm incident can be brought back online, it must be returned to a good working condition. For an incident handler, he/she should ensure in this process:

a) Recover damaged or lost data if any

In our particular case, we did not find any lose of data.

b) Pre-production security assessment

The VictimA's laptop was scanned again by the incident handler to ensure its compliance with the GIAC Enterprises' security policies. Both Microsoft **Baseline Security Analyzer** (MBSA) V.2.1.2 and on-line scanner http://v4.windowsupdate.microsoft.com/en/default.asp were used along the victim himself. This way, the victim learned to how to keep his computer up to date.

All the infected directories, folders, and registry keys were carefully checked again by both VictimA's LSO and the incident handler himself.

c) Restore system to normal operation

The laptop was brought to the most current patch level (see Figure 6).

© SANS Institute 2000 - 2005 Author retains full rights.

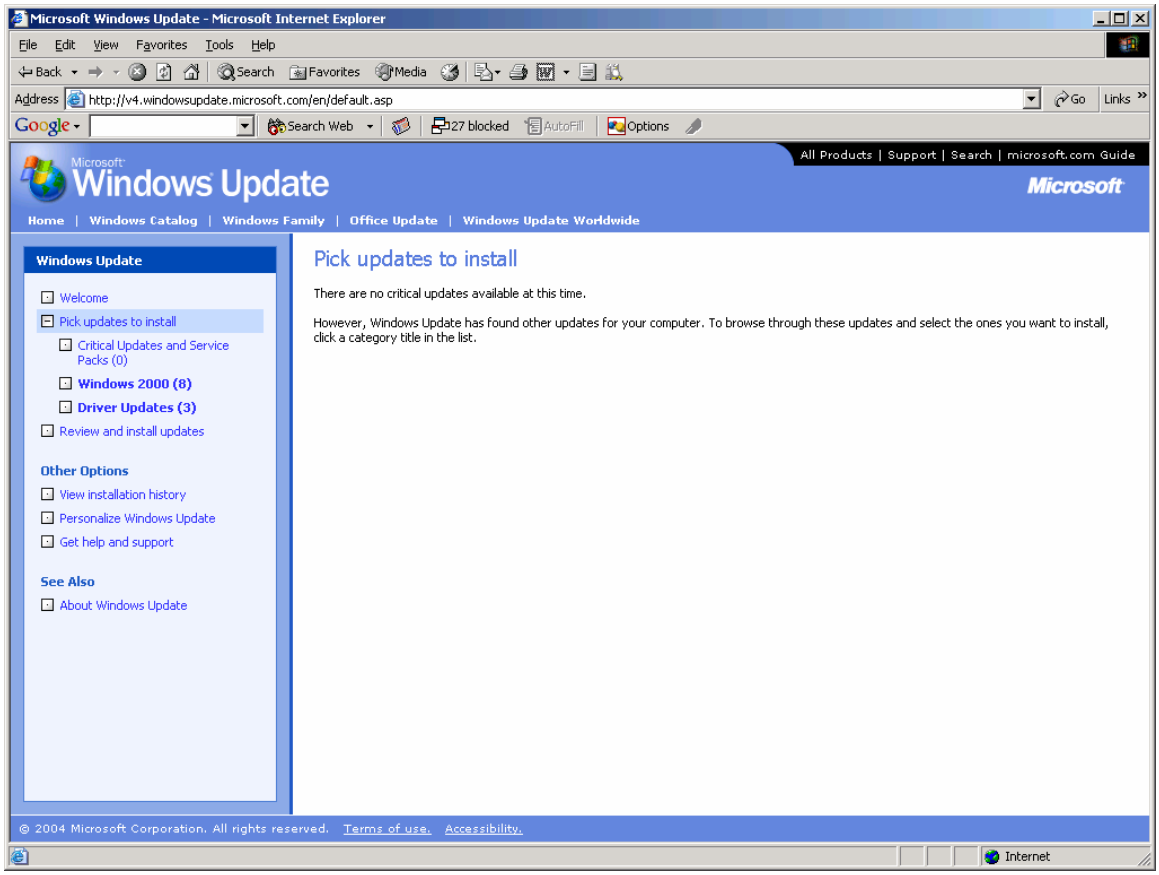


Figure 6 Microsoft on-line patch scan result

All the above three actions were executed by the incident handler, the desktop technician, and the owner of the affected laptop.

Lessons Learned Phase

Restoring a system to normal operation does not mark the end of a security incident handling process. It is always good idea to draw the lessons from the past experience.

Timeline of Handling Incident W32/SDBot.Wom

As the process approaches to its end, I believe that it is a good idea to draw timelines for a quick recapitulation of actions taken during the incident handling process.

Time: Dec. 6, 2007	Person & role	Actions	Handling phase
6:10pm	IDS analyst in EIRG	VPN log IDS alerting script sent out a email alert to IDS-	IDS monitoring

© SANS Institute 2000 - 2005 Author retains full rights.

		EIRG@giac_enterprises.com	
6:15pm	IDS analyst	- Believed that this event should be raised as an incident because the alert showed some network scans going on the corporate networks and unusual IRC traffic that is prohibited by the corporate policy; - Informed the incident handler on duty.	Identification phase
6:18pm – 6:25pm	Incident handler of EIRG	- Made a preliminary assessment; - Escalated the event to manager; - Collected evidences and was continuing the further investigation.	Identification Phase
6:25pm – 6:30pm	Incident manager of EIRG	- reviewed the preliminary assessment and collected data; - registered the event as an incident to handle; - started an incident conferment call.	Identification Phase
6:30pm – 7:00pm	Incident manager, local security officers from affected departments In the conference call	- review the data provided by the incident handlers; - define the incident scope and magnitude; - set up a suitable handling plan; - determine the resource and actions.	Identification Phase
7:05pm – 7:10pm	Incident Manager, LSOs, network administrators, Desktop support person	- related department managers, network administrators, desktop support were informed the decisions and action plan	Containment phase
7:10pm – 7:45pm	Incident handler, manager, LSOs, network	- Incident manager informed Remote Access Control group to disable the identified VictimA's VPN	Containment phase

© SANS Institute 2000 - 2005 Author retains full rights.

	administrators, Desktop support person	access; - Incident handler continues looking into other data resources; - Network administrators started to closely monitor network traffic and report to the incident manager every 15 minutes; - The victim's VPN connection was forcedly cut off; - The victim was informed and asked to shut down his laptop without changing any thing.	
Dec. 6, 2007 (next day) 9:00am – 9:15am	Incident Handler, Desktop support technician	- A desktop support technician is sent on site. He conducted a system assessment and collected system information under the instruction of the incident handler	Containment phase
9:15am - 9:30am	Incident handler	- Analyzed the system information collected by desktop support person - Found out the root-cause of laptop's infection - Established a clean up document and sent back to the support technician	Containment phase
9:30am – 10:00am	Desktop support technician	- Received the instruction of how to clean up the SDBot.worm from the incident handler and helped the victim to pursue this step. - Before this step, a full system back up was made to protect the customer information on the victim's laptop	Eradication
10:00am – 10:30am	Desktop support technician	- Helped the victim to restore his laptop and make a thorough virus and patch check on it.	Recovery phase

© SANS Institute 2000 - 2005 Author retains full rights.

		- Reported the recovery status along with the system information of after recovery to incident handler and manager	
10:30am – 10:40am	Incident manager, Incident handler	- Both incident handler and manager reviewed the recovery and final assessment report. - A permission of using the infected laptop on corporate network is issued.	Recovery phase
11:00am	Incident manager	- Conference call for lessons learned and final remarks	Lesson Learned Phase

Listing 10 Timeline of handling incident W32/SDBot.Worm

After Thoughts

This W32/SDBot.Worm incident was detected fairly quickly and the worm did not spread wildly on the corporate networks. This is largely because the company has effectively implemented a "defense-in-depth" strategy. First of all, the firewall/Proxy rules were set up appropriately so that they leave no chance for the infected machine to go outside of the corporate networks to join the malicious botnets. Secondly, using different IDS systems can help security professionals to discover the attacks more accurately and timely. Thirdly, an enterprise-wise Anti-Virus system and patch management system have greatly enhanced the security for the windows servers and desktop on the network to withstand the worm attacks. Finally, the well-established incident handling policy and process have immensely helped the involved handlers to handle the incident in a controllable and efficient way.

Having said all these nice things, as an incident handler, I just have some concerns on company's security in thinking a possible attacking scenario as follows:

Let's assume the victim in our case is also a "smart" attacker who knows well the corporate networks and defense measures in place. He has also strong programming skills. He knows then to pick up right open-source bots and customize them to carry Back-Door Trojans, worms, and Denial-of-Service (DoS) programs. Under this assumption, does this "internal attacker" have any chance to:

© SANS Institute 2000 - 2005 Author retains full rights.

- Find a way to get through the corporate proxy and set up a IRC channel with outside command master (see Figure 2)?
- Find enough compromised computers (zombies) in order to establish an internal botnet?
- Take the master control and make malicious orders to zombies?

By thinking of this worse case scenario, we may ask ourselves:

> Are we capable of coping with SDBot attacks from this kind of attackers, which can possibly happen in the future?

This paper is not to answer this question. It is, however, as some after-incident-thoughts, to proactively recommend some defense lines in battling this worse case bot-attack in the future.

- Raising the awareness towards network security and best practice for managing and using computers that connect to the corporate networks directly or via VPN.

- Deploying other type IDS systems than signature based IDS, for example, anomaly-based IDS systems. For some critical network points, Intrusion Prevention Systems (IPSs) can consider to be deployed [16].

- Developing an enterprise-wise desktop management system so that there are no unmanaged and un-patched desktops being active on the corporate net [17].

© SANS Institute 2000 - 2005 Author retains full rights.

© SANS Institute 2000 - 2005 Author retains full rights.

References

[1] Symantec Internet Security Threat Report, Volume VI September 2004

[2] Malicious Bots Threaten Network Security, By David Geer http://csdl.computer.org/comp/mags/co/2005/01/r1018.pdf

[3] IRC-Sdbot, McAfee Inc. http://vil.nai.com/vil/content/v_99410.htm

[4] Bots & Botnets: An Overview, GSEC Practical Assignment by Ramneek Puri, August 2003 http://www.giac.org/certified_professionals/practicals/gsec/3372.php

[5] RFC 2810, Contributors: Matthew Green, Michael Neumayer, Volker Paulsen, Kurt Roeckx, Vesa Ruokonen, Magnus Tjernstrom, Stefan Zehl *Internet Relay Chat: Architecture Request for Comments: 2810* April 2000 URL: http://www.irchelp.org/irchelp/rfc/rfc2810.txt

[6] Microsoft Security Web Site: http://www.microsoft.com/technet/security/default.mspx

[7] CERT® Advisory CA-2003-08 Increased Activity Targeting Windows Shares URL: http://www.cert.org/advisories/CA-2003-08.html

[8] Quarterly Report, National Infrastructure Security Coordination Centre (NISCC), URL: http://www.niscc.gov.uk/niscc/docs/re-20041231-00959.pdf?lang=en

[9] SANS Institute. "Hacker Techniques, Exploits and Incident Handling." SANS Online Training Material URL: http://www.sans.org/onlinetraining/track4.php

[10] **W32/Sdbot.worm** analysis web page at McAfee Inc. URL: http://vil.nai.com/vil/content/v_100454.htm

[11] **WORM_AGOBOT.GEN - Description and solution** Trend Micro, Inc. URL: http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_AGOBOT.GEN&VSect=Sn

[12] W32.HLLW.Gaobot.gen Technical Description, Symantec Inc. URL: http://securityresponse.symantec.com/avcenter/venc/data/w32.hllw.gaobot.gen.html#technicaldetails

[13] Incident Response: Strategic Guide to Handling System and Network Security, By E. Eurgene Schultz and Russell Shumway, 2002, New Riders Publishing

© SANS Institute 2000 - 2005 Author retains full rights.

[14] Backdoor.IRC.SdBot at http://www.viruslist.com/eng/viruslist.html?id=51544

[15] Detecting Bots in Internet Relay Chat Systems, by Jonas Bolliger and Thomas Kaufmann, May 2004 to July 2004 ***Institut für Technische Informatik und Kommunikationsnetze***

[16] Intrusion Prevention Systems: the Next Step in the Evolution of IDS, By Neil Desai, Feb. 27, 2003, URL: http://www.securityfocus.com/infocus/1670

[17] Planning a Managed Environment, Microsoft Inc., URL: http://www.microsoft.com/resources/documentation/WindowsServ/2003/all/deployguide/en-us/Default.asp?url=/resources/documentation/WindowsServ/2003/all/deployguide/en-us/dmebf_use_overview.asp

[18] Knowing your Enemy: Tracking Botnets – Using Honeynets to Learn more Bots, The Honeynet Project & Research Alliance, March 13, 2005, URL: http://www.honeynet.org/papers/bots

© SANS Institute 2000 - 2005, Author retains full rights.

© SANS Institute 2000 - 2005 Author retains full rights.