SANS PENETRATION TESTING

Use offense to inform defense.
Find flaws before the bad guys do.

Copyright SANS Institute
Author Retains Full Rights

This paper is from the SANS Penetration Testing site. Reposting is not permited without express written permission.

Interested in learning more?

Check out the list of upcoming events offering
"Hacker Tools, Techniques, Exploits, and Incident Handling (SEC504)"
at https://pen-testing.sans.org/events/

False Alarm…Or Was It?

Lessons Learned from a Badly Handled Incident

by

Dana K. Graesser

Advanced Incident Handling and Hacker Exploits
SANS GCIH Practical Assignment v2.0

© SANS Institute 2000 - 2002, Author retains full rights.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

Table of Contents

Overview 3
The Exploit 4
  Name 4
  Operating System 4
    Brief Description 4
    Variants 5
    References 6
The Attack 7
  Description and Diagram of the Network 7
  Protocol Description 14
    Windows 14
    Linux 14
  How the Exploit Works 15
    Windows 15
    Linux 16
  Description and Diagram of the Attack 17
  Signature of the Attack 18
  How to Protect Against It 19
    Windows 19
    Linux 20
The Incident Handling Process 22
  Preparation 23
    Policy 24
    Systems 26
    Forensic Tools 26
    Incident Response Plan 28
    Incident Response Report 30
    Training 31
  Assessment 32
    Identification 32
    Containment 34
    Eradication 37
  Recovery 37
  Lessons Learned 38
References 42
Appendix B 45

© SANS Institute 2000 - 2002, Author retains full rights.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

False Alarm…Or Was It?
Lessons Learned from a Badly Handled Incident

Overview

Companies have many reasons for securing their networks. Many have only recently determined that security is an issue that they cannot ignore. With the terrorist attacks of September 11, 2001, the need for security became a top priority for companies that had not previously considered it.

Incidents can start, the incident response process can be initiated, and actions taken while pursuing the wrong path. That is what happened on 17 January 2002 at Company X. The security console showed that there was unusual activity from three machines within minutes of each other. A junior member of the team was assigned to serve as incident manager.

The incident was presented to the security manager on duty as a worm. A set of actions was then taken based on the standard procedures set up for a virus or worm.

Further investigation, however, made it apparent that the original diagnosis of worm was incorrect. However, evidence contained on the affected machines was destroyed because the personnel to whom they belonged were able to make changes and delete files.

This is a cautionary tale about fully researching and understanding the exploit before assuming that is what is in play. Training of personnel is probably the most important component of incident handling. Poorly trained personnel can damage or destroy forensic evidence used to validate an attack or prosecute suspected attackers.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

The Exploit

Name

Initial Diagnosis - Millennium Worm
Final Analysis – Exploit Unknown

Operating System

According to information available on the Internet, the operating systems affected by the Millennium worm are Windows 95, Windows 98 and Windows NT. This impacts all versions of the three operating systems.

No testing has been conducted to see whether this worm would infect Windows 2000, Windows ME or Windows XP. This testing was not possible at Company X because the two standard images were based on Windows 98 for laptops and Windows NT for desktops and servers. Therefore it is not possible to confirm or deny whether these additional platforms are vulnerable.

Millennium worm also has a variant that affects Linux-based machines.

Brief Description

Note that the worm has been referred to as Millenium worm or Millennium worm. Both terms will be used interchangeably throughout. This worm is an older worm developed and originally released in 1999. In character, the worm truly acts as a trojan because it allows remote access into the infected clients. However, based on the naming, it will be referred to as a worm throughout the paper.

However it is not unprecedented for one virus or worm to be re-written to fix some of the problems in the original or to get past anti-virus protection. A recent example of re-release to get past virus protection was the re-release of the Klez worm which is known as klez.g, klez.h or klez.k (Lemos, 2002).

The Millennium backdoor is one of many backdoor programs that attackers can use to access your computer system without your knowledge or consent. With the Millennium backdoor, an attacker can do the following:

- log keystrokes
- capture an image of your screen
- execute programs
- send messages to you that appear on your screen

http://www.iss.net/security_center/static/3111.php

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

The Windows worm is purported to use ports 20000 and 20001. This incarnation of the worm is purported to be written in Visual Basic.
http://www.sans.org/newlook/resources/IDFAQ/oddports.htm
http://www.aroundtownnc.com/security/computer_ports.html
http://www.onctek.com/trojanports.html

There is an analysis of an actual infection in the wild that appears to show that the worm actually uses port 6667. It is the variant that is written in Delphi.

The worm is built on a client and server model. Target machines are infected with the client and are controlled by the server.

The confusion about which ports the worm uses and what it was written in is based on the fact that most websites do not provide the specifics about what was used to write the worm. An analysis from the wild states that is was written in Delphi. Most of the virus reference sites do not list anything, but several mention Visual Basic.

Based on the fact that worms require no specific action on the part of the user to enable infection and propagation (Kerby, 2001), infection by a worm is a highly worrisome incident for the security team to contain.

Variants

The Security Focus library contains a small amount of information about the Linux variant. The summary of the paper is included below.

The Millennium Internet Worm is a collection of scripts and programs whose function is to exploit common remote vulnerabilities in linux systems in order to gain access and propagate itself throughout the net. The Millennium Worm discovered currently uses Linux specific x86 remote exploits for the imap4 v10.X, Qualcomm popper, bind with iquery, and rpc.mountd services. This worm does one thing that we can appreciate, which is to fix the security holes. It has been reported to have infected at least one person in the wild (http://www.securityfocus.com/library/1432).

G-Lock Software provides the following information about the Millennium variant.

Name:	Millenium Worm
Aliases:	N/A
Ports:	1338
Files:	Mworm.tgz -
Created:	N/A
Requires:	N/A
Actions:	Worm
Versions:	N/A
Registers:	N/A
Notes:	Works on Unix (Linux). Password = millennium.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

(http://www.glocksoft.com/trojan_list/Millenium_Worm.htm)

The Linux variant is purported to use port 1338.
http://www.sans.org/newlook/resources/IDFAQ/oddports.htm
http://www.aroundtownnc.com/security/computer_ports.html
http://www.onctek.com/trojanports.html

References

No specific notices for Millennium worm were found at cve.mitre.org or www.cert.org after searching the incident notes and the vulnerability notes database. There were notices about some of the services used.

The Windows worm uses the following services to gain privileges:

Service	Applicable CVE Notices	Applicable CERT Notices
mIRC	CAN-1999-0399	

The Linux variant uses the following services to gain privileges:

Service	Applicable CVE Notices	Applicable CERT Notices
Imapd Overflow	CVE-1999-0005, CVE-1999-0920	CA-98.09.imapd
Bind with iquery	CVE-1999-0009	
Rpc.mountd	CVE-1999-0210	
Qpopper Overflow	CVE-1999-0006	

The Millenium worm can be downloaded for educational purposes from:http://www.thenewbiesarea.com/home_index.html.

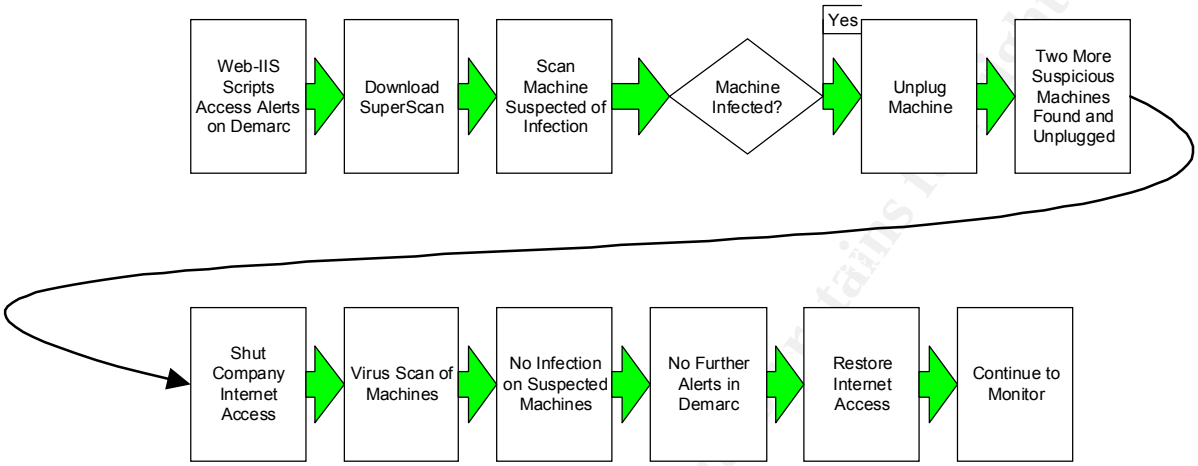
Once on the website, go to the category Hacking on the left hand side and choose Trojans from the list. The site requires registration before downloading, but it is a no cost registration.

Additional references can be found at:
http://www.iss.net/security_center/static/3111.php
http://www.sans.org/newlook/resources/IDFAQ/oddports.htm
http://www.aroundtownnc.com/security/computer_ports.html
http://www.onctek.com/trojanports.html
http://securityresponse.symantec.com/avcenter/venc/dyn/7670.html
http://www.securityfocus.com/library/1432
http://www.glocksoft.com/trojan_list/Millenium_Worm.htm

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

The Attack

Description and Diagram of the Network



The above is a graphical representation of what occurred at Company X on 17 January 2002. It is a picture of what actually happened, not of what should have ideally happened. It is immediately apparent that a major mistake was made during the attack and our response. The incident was classified as a worm without having performed even a basic virus scan on the suspected machine.

The response stance and heightened alarm level of management were the main reasons that Internet access for the entire Company was removed. The management alarm level was based on concerns related to terrorist hacking and the description of the worm as presented to them.

Note – All real IP addresses have been changed to start with 555.444 to protect the real addresses.

The network at Company X is comprised of two pieces. The first piece is what is referred to as the Admin network. The Admin network is the location of all administrative end-user workstations. These workstations are used for office applications, email and Internet access. Servers, such as Admin network domain controllers, are on the Admin network to support the servers available.

The Admin network has several layers of protection from the Internet. The first layer is a border router with well-constructed access control lists (ACLs). The next layer is a firewall with additional restrictions with the security stance that all traffic is denied except that which is specifically permitted.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

The intrusion detection probe is a pattern-matching device. The probe between the firewall and the router does not use any automated response. Company management determined that all changes and connection resets should require human intervention.

Another layer of protection is provided by the fact that only a small, identified community of users can access the Internet while on the network. A proxy moderates the traffic.

The second piece of the network is referred to as the Production network. The Production network contains all servers that house the business-critical applications and terminals used to access those servers.

The Admin and Production networks are connected via firewall. The Production network does not have Internet access. The Production network is a highly sensitive environment with strict performance service levels required.

Every precaution is taken if there is even the suspicion of infection on the Admin network because, despite all attempts control contact, personnel insert machines to try and circumvent security measures in place.

Despite the fact that the two networks are only supposed to be connected via firewall, the information security team has repeatedly found personnel who purchase and install a second network interface card to serve as a jump host from the Admin network to the Production network. These dual-NIC'd machines are regularly scanned for, but a scan had not occurred for approximately three weeks.

© SANS Institute 2000 - 2002 Author retains full rights.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

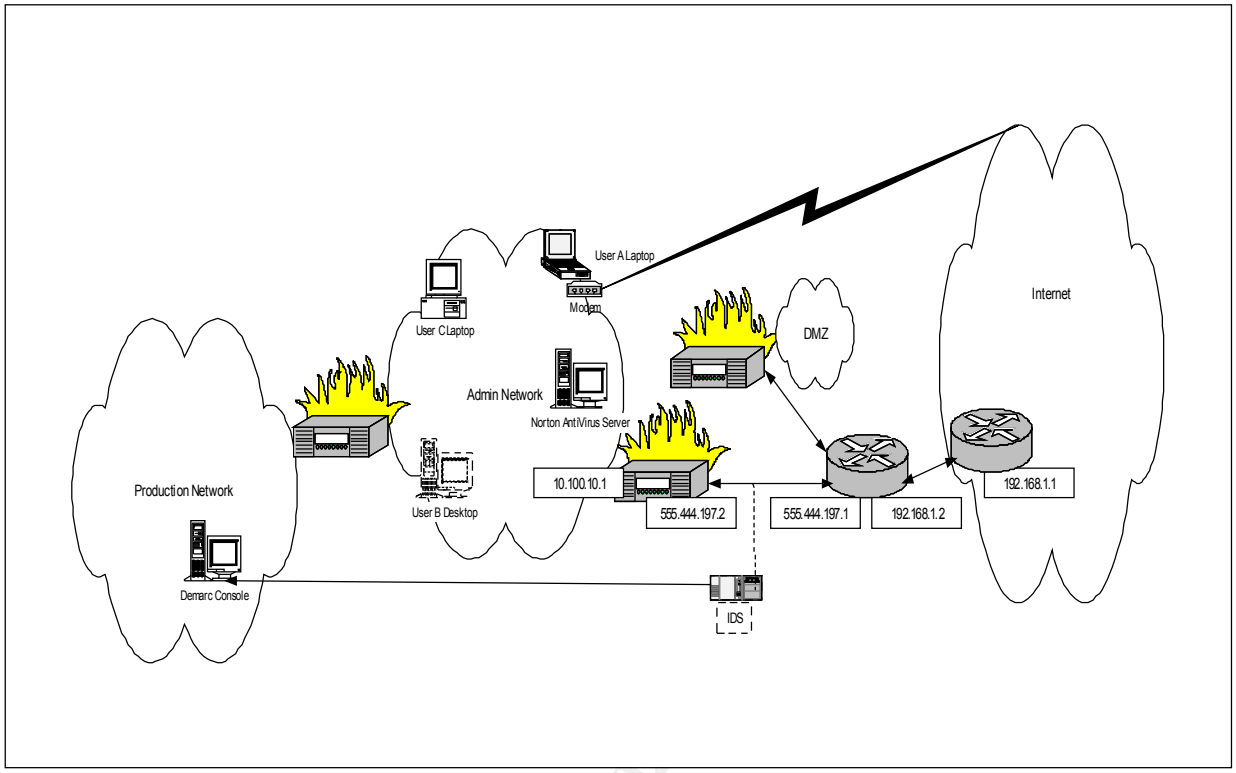


Figure 0-1 Network Drawing

<u>Internet Router</u> – The Internet Router was a Cisco 2620 running IOS 12.2.2. The IOS was chosen by the Network team without input or consideration for security because of functionality concerns. Pertinent parts of the configuration:

```
!
interface FastEthernet0/0
 no ip address
 speed 100
 full-duplex
!
interface FastEthernet0/0.2
 encapsulation isl 2
 ip address 555.444.197.1 255.255.255.224
 ip access-group 105 out
 no ip redirects
 no ip mroute-cache
 ip policy route-map Ecommerce
 service-policy input mark-inbound-http-hacks
!
interface Serial0/1
 description circuit abc123 - Internet 1
 bandwidth 1536
 no ip address
```


© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

```
 encapsulation frame-relay IETF
 down-when-looped
 frame-relay lmi-type ansi
!
interface Serial0/1.2 point-to-point
 description Internet Link 1
 bandwidth 768
 ip address 192.168.1.2 255.255.255.252
 ip access-group RDP in
 ip access-group 105 out
 service-policy input mark-inbound-http-hacks
 no cdp enable
 frame-relay interface-dlci 16
!
class-map match-any http-hacks
  match protocol http url "*.ida*"
  match protocol http url "*cmd.exe*"
  match protocol http url "*root.exe*"
  match protocol http url "*Admin.dll*"
  match protocol http url "*.idq*"
  match protocol http url "*.nws*"
  match protocol http url "*riched20.dll"
  match protocol http url "*readme.exe*"
  match protocol http url "*mmc.exe*"
  match protocol http url "*WTC.exe*"
  match protocol http url "sample.exe"
  match protocol http url "*_vti_bin*"
  match protocol http url "*_mem_bin*"
  match protocol http url " *.scr"
!
access-list 105 deny   ip any any dscp 1 log
access-list 105 permit ip any any
!
ip access-list extended RDP
 deny   udp any any eq 259 log
 deny   ip 1.2.3.0 0.0.0.255 any log
 deny   ip host 1.2.3.4 any log
 deny   ip host 2.3.4.1 any log
 deny   ip host 3.4.1.2 any log
 deny   ip host 4.1.2.3 any log
 permit ip any any
```

Specific hosts are denied at the border router if it is determined that the Company is being scanned or attacked by that address. The hosts were being blocked at the firewall, but policy dictated that habitual attackers be stopped at the furthest point of the network possible.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

Determining who is a habitual attacker or scanner is a subjective call by security team members in association with the security manager. The host IP address is checked against the Top 10 Most Wanted list, which is located at www.dshield.org/top10.html. The security team does not have the authority to make the change to the router configuration. The change must be approved by the Change Control Board and implemented by the Network team.

The hosts are removed after a month if further attempts are not found in the log. The router is running Cisco Express Forwarding (CEF) and has a static route to the Internet Service Provider. According to Cisco, CEF is a scalable, distributed, layer 3 switching solution designed to meet the future performance requirements of the Internet and Enterprise networks. Express Forwarding evolved to best accommodate the changing network dynamics and traffic characteristics resulting from increasing numbers of short duration flows typically associated with Web-based applications and interactive type sessions (Cisco, 2002).

The rest of the routers on both the Admin network and the Production network are running EIGRP, the Cisco proprietary routing protocol.

Internet Firewall – The Internet firewall is a Cisco PIX 515 running IOS 6.0.1. The IOS was chosen by the Network team without input or consideration for security because of functionality concerns.

```
ip address outside 555.444.197.2 255.255.255.224
ip address inside 10.100.10.1 255.255.255.0

global (outside) 1 interface
nat (inside) 0 access-list Internet-2
nat (inside) 1 0.0.0.0 0.0.0.0 0 0

static (inside,outside) 555.444.197.69 10.100.19.49 netmask 255.255.255.255 0 0
static (inside,outside) 555.444.197.70 10.100.19.43 netmask 255.255.255.255 0 0
static (inside,outside) 555.444.197.71 10.100.19.44 netmask 255.255.255.255 0 0
static (inside,outside) 555.444.197.72 10.100.19.45 netmask 255.255.255.255 0 0
static (inside,outside) 555.444.197.77 10.100.15.20 netmask 255.255.255.255 0 0

access-list Internet-2 permit ip 10.0.0.0 255.0.0.0 172.18.0.0 255.255.248.0
access-list Internet-2 permit ip 10.0.0.0 255.0.0.0 555.210.180.160 255.255.255.240
access-list Internet-2 deny ip 10.100.19.0 255.255.255.0 any
access-list Internet-2 deny ip 10.100.0.31 255.255.0.255 any
access-list Internet-2 deny ip any 555.75.26.0 255.255.255.0
access-list Internet-2 deny ip any host 555.10.144.221
access-list Internet-2 deny ip any host 555.28.82.48
access-list Internet-2 deny ip host 10.100.101.30 any
access-list Internet-2 deny ip host 10.100.73.28 any
```


© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

```
access-list Internet-2 deny ip host 10.100.101.134 any
access-list Internet-2 deny ip host 10.100.4.52 host 555.444.81.3
access-list Internet-2 deny ip host 10.100.4.4 host 555.147.81.3
access-list Internet-2 deny ip host 10.100.8.179 any
access-list Internet-2 deny ip host 10.100.4.9 host 555.147.81.3
access-list Internet-2 deny ip host 10.100.73.25 any
access-list Internet-2 deny ip host 10.100.15.20 any
access-list Internet-2 permit ip any any

access-group outside in interface outside
access-list outside deny tcp any any eq irc
access-list outside deny udp any any eq 194
access-list outside deny tcp any eq irc any
access-list outside deny udp any eq 194 any
access-list outside deny tcp any any eq daytime
access-list outside deny udp any any eq 13
access-list outside deny tcp any eq daytime any
access-list outside deny udp any eq 13 any
access-list outside deny tcp any any eq 27374
access-list outside deny udp any any eq 27374
access-list outside deny tcp any eq 27374 any
access-list outside deny udp any eq 27374 any
access-list outside deny tcp any eq 6346 any
access-list outside deny tcp any any eq 6346
```

Norton AntiVirus Server – The Norton AntiVirus Server was built on an x86 server platform with dual processors. The server is a backup domain controller running Windows NT 4.0 Server with Service Pack 6a. It is the central distribution point for updates to the antivirus running on the Admin clients.

Microsoft Windows NT 4.0 Server
Microsoft Windows NT 4.0 Server Service Pack 6a
Veritas, Backup Exec 8.5
NetIQ Monitoring Software 4.0

Demarc Console – The Demarc console was built on an x86 server platform with dual processors. The console OS was Linux Mandrake 8.0 Kernel 2.4.3. The security team, based on the requirement that the device be run on Linux and the comfort level with Mandrake in particular, chose Mandrake. Additionally the console had loaded:

Mysql ver. 3.23.42-1
Msyslogd ver. 1.0.1
Gsheild
Mysql-php

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

OpenSsh
DBI
Mysql-perl
Apache web server 1.3.20

Figure 0-2 Demarc Console

Snort Probe – The probe was built on an x86 workstation platform. The probe OS was Mandrake 8.0 Kernel 2.4.3. The security team, based on the requirement that the device be run on Linux and the comfort level with Mandrake in particular, chose Mandrake. Additionally loaded was:

Gshield iptables firewall
Ssh 2.5.2.p2
Snort 1.8.1
Snort 1.8.2 for Demarc

Figure 0-3 Snort Probe

Standard Laptop Configuration – All laptops were built on a standard image designed by the LAN administrators without input from the Security. The reason that security was not involved is that the laptop standard image was built before mid-August 2001.

A major issue is that all users were set up as local administrators for their laptop. The laptops were designed to be connected only to the Admin network. No laptop was supposed to be connected to the Production network.

Another significant issue for this incident was the fact that the laptops were part of the Admin network. Since the security effort had started at Company X in mid-August 2001, a decision was made that Admin network machines were low priority for hardening and patching. This was a business decision by the management team since the critical applications all ran on the Production network. It was decided that time and resources should first be concentrated on the Production network.

Considering that the users had local administrator access the original standard image, which was not designed with security in mind, many laptops included settings and programs that are not part of security best practices.

Microsoft Internet Explorer 5.0
Norton AntiVirus 7.0.1
Lotus Notes Client 5.0.5
Microsoft Office 97

Figure 0-4 Laptop

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

Protocol Description

Windows

The protocols involved in the Millennium Worm are:

TCP/IP - As with all other communications protocol, TCP/IP is composed of layers:

- IP - is responsible for moving packet of data from node to node. IP forwards each packet based on a four byte destination address (the IP number). The Internet authorities assign ranges of numbers to different organizations. The organizations assign groups of their numbers to departments. IP operates on gateway machines that move data from department to organization to region and then around the world.
- TCP - is responsible for verifying the correct delivery of data from client to server. Data can be lost in the intermediate network. TCP adds support to detect errors or lost data and to trigger retransmission until the data is correctly and completely received.
- Sockets - is a name given to the package of subroutines that provide access to TCP/IP on most systems.

http://www.yale.edu/pclt/COMM/TCPIP.HTM
TCP/IP is the only protocol that is used on both the Admin network and the Production network. It is the method by which users connect to the Internet either at the office or when connected at home to an Internet Service Provider (ISP)

IRC - (Internet Relay Chat) is a multi-user chat system that originated in 1988. IRC uses a client program to connect to a server (usually via port 6667, but in the range of 6660-6670). IRC is officially assigned port 194 according to RFC 1700. IRC was prohibited at Company X, but it is used Business Y in the analysis in Appendices A and B that serves as the basis for analysis in the next section “How the Exploit Works”.

Linux

IMAP – IMAP is one of several protocols designed for accessing email from an assigned server. It is based on a client server model. The server holds the messages. The user has the ability to look at the heading and the sender to determine whether download to the local machine is acceptable. Problems with IMAP include, but are not limited to: arbitrary command execution via IMAP buffer overflow in authenticate command and buffer overflow in the pop-2d POP daemon in the IMAP package allows remote attackers to gain privileges via the FOLD command.

BIND with iquery – Bind is a popular implementation of DNS. DNS is the way that IP addresses are changed into names for ease of navigation on the Internet. The normal type of DNS query involves taking a name and changing it into an IP address. An inverse query (iquery) is taking an IP address and turning it into a name. DNS has many security vulnerabilities.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

FTP – FTP was developed as a protocol to exchange files between two devices that are IP based. It is most often used to download files via either a graphical interface or via command line.

How the Exploit Works

Windows

This is an older worm so a current signature file on anti-virus should prevent a machine from becoming infected. However there should always be concern that the original code had been found, re-worked, and re-issued.

The Windows based Millenium is based on having a centrally located server with client software loaded on the target machine(s). Web based commentary on the worm states that it was written in Visual Basic. The analysis contained within Appendices A and B about a possible infection at Business Y states that the worm was written in Delphi.

The specifics of how the exploit worked at Company X cannot be detailed because Millennium worm did not truly infect the network so nothing was captured in that manner. Any other evidence was destroyed when the users involved cleaned out their caches.

However, Appendices A and B contain a discussion about a possible Millenium worm infection at another Company (Business Y). According to the email, machines are infected via IRC. This implies that there is user intervention required to infect the machine. The user probably downloaded some sort of content that they did not virus scan. The case involving Business Y does not state whether the infected PC had anti-virus software running on it. It only states that a bootable disk was created and taken to the machine later to scan it.

After downloading the content, the user must have engaged in some action (such as double-clicking) that caused the executable to be run and install the worm. However, at that point the trojan was installed and access to the machine could be gained.

Since Millenium appears to require user intervention to install, it seems likely that email could be used as an infection vector. Users could receive the file and click on it. If they are not running some sort of virus scan they could become infected.

Using IRC as an infection method and a channel for remote control has become more prevalent over time. Some basic rules of safety for IRC include virus scanning all content downloaded before opening and being wary of the intentions of others if they ask you to type commands or run scripts.

An important note is that malicious code may include a hidden extension that would hide malware as a picture or text file. Most of these hidden extensions can be unhidden

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

using the procedures described below (Lo, 2002). An important caveat is that there are three extensions for executables that cannot be unhidden. These extensions are .shs, .pif, and .lnk. An example would be a picture of Anna Kournikova titled “Anna.jpg” that is really “Anna.jpg.pif”.

In Windows 95/98:
- Open Explorer
- Under View menu, select Options
- Check "show all files"
- UNcheck "hide MSDOS file extensions that are registered"
- Click OK to finish

In Windows 2000 (and probably NT):
- select Start | Settings | Control Panels | Folder Options
- select the View tab
- check "show hidden files and folders"
- UNcheck "hide file extensions for known file types"
- Click OK to finish

http://www.irchelp.org/irchelp/security/trojanext.html

With additional testing, I was able to view hidden extensions using the procedure for Windows 2000.

Additional reading can be found at http://www.irchelp.org/irchelp/security/trojan.html.

Once the trojan is installed, there are a number of items that can be controlled remotely based on the code. This list of tasks is a representative sample taken from the strings reference included in Appendix B. Tasks that can be done remotely include sending keystrokes to a remote location, logging off the current user, disabling the keyboard and restarting the system.

Linux

An analysis by G-Lock Software about the functionality of the Linux variant is as follows:

1. After finding a weakness, it ftp's its source code and compiles it on the host;
2. FTP's the patches to fix how it got in.
3. tars itself, and begins probing the hosts in /etc/hosts and then all class A, B, and C addresses it can find;
4. When a weakness is found on another host, it gets in and starts the process over again.
5. OBTW - it forks off 20 processes, so it's working overtime.

Here's an excerpt from the README-ADMIN file:
Dear Admin, if you read this file you have been 0wned by the Millennium Internet Worm. This is a program that exploits some remote bugs to gain access, installs itself and goes on copying itself to other systems. This is a modular worm, which means that other exploits used to spawn itself can be added easily,

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

like a frontend script to a sniffer. For now, this exploits * imap4 v10.X * qualcomm popper * bind with iquery * mountd. This worm is linux specific. This could be changed by porting the exploits and shell code to other systems. This means, do not expect that non-linux boxes will be completely unaffected by variants. We will now try to patch the stuff you should have replaced a long time ago. -Anonymous =oP~

5. If none of the above services installed, the worm downloads them, compiles them, and installs them. Very nice of it.

(http://www.glocksoft.com/trojan_list/Millenium_Worm.htm)

Description and Diagram of the Attack

On 17 January 2002 at approximately 1337 Mountain Standard Time (MST) a case was opened in the Brand Y ticketing system reporting that Demarc showed multiple "WEB-IIS scripts access" on LocA-LAPTOP-127. Based on the name of the device, it was determined that the machine was a laptop. All laptops are built on generic images using Windows 98 as the operating system. Tom was assigned as Incident Manager. Additional personnel from other teams were assigned as Technical Specialists.

The security investigation revealed that LocA-LAPTOP-127 was using UDP port 1338. This is generally an indication of infection by the Millennium Worm. Similar alerts were seen for LocA-DESKTOP-740 and LocA-DESKTOP-026. This led the IS security team to believe there was a possible worm infection.

The above machines were unplugged from the Admin network to prevent any possible further infections. The machines were scanned for any viruses or worms. The results were negative.

The Internet gateway was also shut down temporarily. This was done because investigation showed that the Millennium Worm could:

- log keystrokes
- capture an image of your screen
- execute programs
- send messages to you that appear on your screen

http://www.iss.net/security_center/static/3111.php

No degradation or infection of the Production network was detected. After a period of 1.5 hours, it was determined that no infection was spreading and the Internet gateway was reopened.

A follow-up ticket was opened. Upon further investigation it was shown that classical Millennium worm could not have infected the network. The worm that travels on port 1338 is the variant that runs on Linux. The Windows based worm used ports 20000 and 20001. It was suspected that the suspicious traffic was actually a music-sharing client

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

sending statistical information to a central server on the Internet. This deduction could not be validated because the original machines left the custody of the security team before a thorough forensic analysis was performed.

Signature of the Attack

No screen captures were taken during the incident. It is a known failure in the incident response system.

The initial signature of the attack was determined by watching the Demarc security console. Multiple "Web-IIS scripts access" alerts were seen (at approximately 1300 there were 3234 attempts and at approximately 1400 there were 3703 attempts). The alerts showed multiple attempts to reach 207.68.78.1 from internal hosts. This alert is one that is included in the default install of Snort rules.

At that point one of the team members downloaded SuperScan 3.0 (http://www.webattack.com/get/superscan.shtml). A scan was conducted to determine what ports were open on the machine that was suspected of infection. The machine showed that port 1338. Based on initial investigation, Millennium worm was known to use port 1338.

A tool had to be downloaded because forensics was not a priority of management based on the stance of Contain and Eradicate. However a port scanning tool is valuable for more than forensics. A scanning tool was needed on a machine that is part of the IT Center because those machines are documented and manned 24 hours a day for 365 days a year.

Nessus and nMap were located on a security team members assigned machine for periodic audit scans. However the team member was unavailable at the time of the suspected incident.

This initial categorization was patently false because the worm uses port 1338 on the Linux variant. The machines that exhibited the behavior were running Windows 98 and Windows NT. Therefore the port that should have been open to indicate Millennium was 20000/20001 or 6667.

If the port open had been 6667, the signature would have included continual connection attempts to various IP addresses using that port while no applications were running in the task list. An additional sign would include having a file named kernel32.vxc located in the c:\windows\system directory as a hidden system file without any version information included.

A problem identified is that no screen captures were taken of Demarc, of the SuperScan results, original Snort output or anything else that led to the assumption that the Company was infected with Millenium.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

This problem was further compounded by the fact that the Security team was in the process of moving from the test installation of the intrusion detection system to the production installation. The incident occurred on 17 January. The system became live on 1 February. As part of the cutover the data of incidents from the test period was dumped without being archived in any fashion.

How to Protect Against It

The first global protection measure to protect against infection on either Windows or Linux machines is to ensure that systems are built with all appropriate security patches and that new patches are applied in an expeditious manner.

The second global measure is to ensure that users are fully trained about their responsibilities to assist the security team so that they are not the cause of an infection. At Company X, business card sized reminders were produced that users could wear on the same lanyard as their corporate badge. The specifics of the card are outlined under Preparation.

A further item could have been to the Top Eight to warn users about opening or downloading unknown content via email, IRC or the web. Since the security initiative is still early in its lifecycle, plans for recurring security training need to be developed. Security education needs to be an ongoing process.

Windows

If infected the ISS database recommends:

> To remove the Millennium backdoor from your computer:
>
> 1.Using Regedit, find the HKLM\Software\Microsoft\Windows\CurrentVersion\Run registry key.
> 2.Find the registry entry named Millennium that has a data value of C:\Windows\System\Reg66.exe.
> 3.Delete that registry entry.
> 4.Delete Reg66.exe from the Windows system directory.
> 5.Open the win.ini file in your Windows directory.
> 6.Find and delete the line "run=c:\windows\system\reg66.exe" from win.ini.

http://www.iss.net/security_center/static/3111.php

As always, the most important step to protect against malicious software includes having updated anti-virus software loaded on all computer systems that connect to the network. This is especially important with systems running older operating systems such as Windows 98.

Symantec gives the following information about the Millennium worm and how to prevent infection.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

This threat is detected by the latest Virus Definitions.

All computer users should employ safe computing practices, including:
Keeping your Virus Definitions updated.
Installing AntiVirus program updates, when available.
Deleting suspicious looking emails.
http://securityresponse.symantec.com/avcenter/venc/dyn/7670.html

Linux

G-Lock Software recommends the following:

Prevention

UPGRADE. The Millennium Worm spreads by remotely exploiting vulnerabilities in earlier versions of certain system software. If you upgrade your software to a newer release that is not vulnerable to these particular holes, then you will be effectively immune to this worm. Please note, however, that it is a trivial matter for attackers to create variations of this worm, using other vulnerabilities including ones affecting other platforms than linux. It is always best practice to keep your system and network software current, and watch public security forums for new information that could affect your operating environment.

Repair

To repair an existing infection from the Millennium Worm, you would need to take the following steps:

delete the suid root shell [/bin/rm -rf /tmp/.mwsh]
stop any running worm processes [/usr/bin/killall -9 mworm]
remove the protection flags on the mworm files [/usr/bin/chattr -R -ia /tmp/....]
remove the worm files [/bin/rm -rf /tmp/....]
replace /bin/ps with the original /bin/.ps [/bin/cp /bin/.ps /bin/ps]
remove the mw user from the passwd file [/usr/sbin/userdel -r mw]
remove worm references from startup scripts /etc/rc.d/rc.local and /etc/profile
kill the "bd" backdoor process. If you have removed the worm then this is as easy as rebooting, since it will not be restarted on the next system boot.

If you have been infected by the worm then you have a fairly large problem. Killing the worms processess, deleteing the files, and removing the mw user from the password file only cleans the known part of this attack. The unfortunate issue

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

is that your system has been compromised at the root level, and your IP address has been sent to an attacker. They could have logged in and done any number of things. A good starting point for your path to recovery is CERT's famous "Steps for Recovering from a UNIX Root Compromise (http://www.cert.org/tech_tips/root_compromise.html)".

(http://www.glocksoft.com/trojan_list/Millenium_Worm.htm)

© SANS Institute 2000 - 2002, Author retains full rights.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

The Incident Handling Process

The incident handling process as described by SANS in Computer Security Incident Handling Step by Step includes Preparation, Identification, Containment, Eradication, Recovery and Lessons Learned. However, this model serves as a framework and a starting place. Each company must decide if all the steps meet the standards set forth in their information security policy and the needs of the company. Company X had determined that Notification, Assessment, Recovery and Closure met the requirements of management and service levels designated for the security team.

Preparation was done and is detailed below. Assessment includes parts of the SANS model of Identification, Containment and Eradication. Recovery is analogous to Recovery and Lessons Learned to Closure.

Incident handling cannot begin without a complete understanding of the organization and of all the personnel who might need to be involved in the incident process. Company management should be detailed because it shows the major company stakeholders.

In order to prevent the creation of bad feelings toward the information security team, the team felt it was important to inform the proper personnel within the organization if a security incident took place. Security is a group effort. The security team is ineffective unless all employees are mindful of security while doing their daily tasks and are willing to work with the team in the case of a problem.

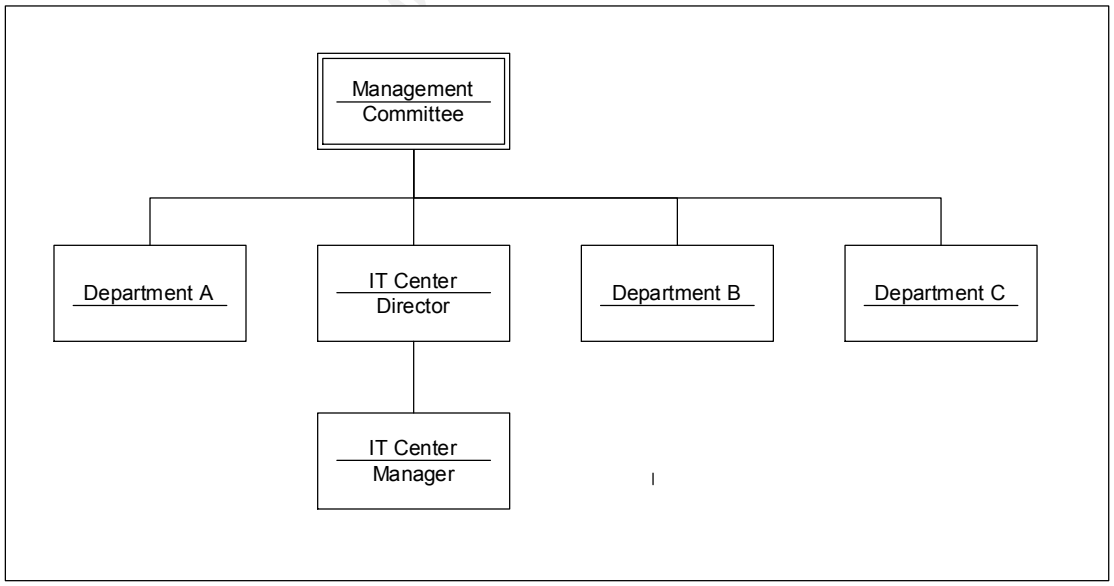


Figure 0-1 Management

The Incident Response Team is an interdisciplinary group made up of personnel from various departments. The core incident response team members are from the security

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

team. A member from the security team serves as the incident manager and directs the efforts of the entire group. The incident manager, in association with the security manager on duty, decides who needs to be called in to assist. Depending on the nature of the incident experts may need to be gathered from different functions.

The security manager on duty can request that incident team be gathered from outside the information technology function. For example, if the incident takes place at a remote location the location manager is made part of the incident team. This helps facilitate communications and makes sure that there is buy-in from different groups within the company.

The duty manager can also request that completely external entities, such as law enforcement or product consultants, be added to the team. The request to bring in external entities must be approved by the IT Center Manager or Director.

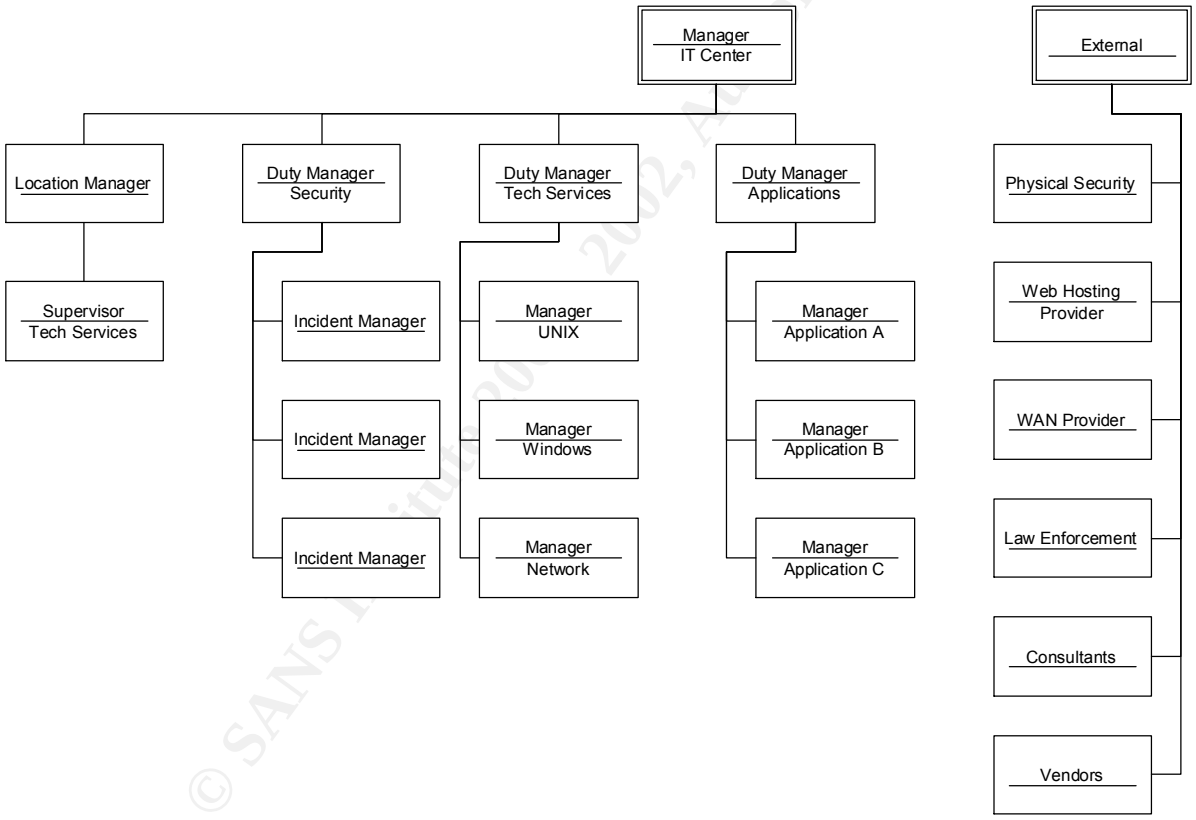


Figure 0-2 Incident Response Team Setup

Preparation

The Company underwent an audit in June because of a change in management. The audit revealed that there were major deficiencies in both physical and logical security. At that point it was determined that the gaps included:

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

- lack of a written security policy
- lack of a written incident response plan
- lack of intrusion detection methods and procedures
- no security tie-ins to network management
- completely decentralized logging and reporting
- an undereducated user population
- change management and configuration management occurred without security

A team was put into place at mid-August 2001 to develop a written security policy and incident response plan. These two documents were to serve as the cornerstone of the security effort. While this effort was in place, an engineering effort started to design and put an intrusion detection system into the network.

Shortly into this effort, the terrorist attacks of September 11, 2001 took place. This added an additional sense of urgency to the security development effort.

An important piece of preparation is having up to date documentation of the network. Documentation should include drawings of the network, a chart or table that indicates major traffic flows and a system that contains pertinent information for all systems on the network.

Pertinent information includes items such as IP address, physical location, point of contact for the system, operating system, applications, and switch port.

Policy

Preparations started by developing and updating an Information Security Policy. The policy requires that all workstations, laptops and servers should be loaded with anti-virus software.

Below is an excerpt from the Information Security Policy. Especially pertinent parts of the excerpt are highlighted.

> Computer viruses are unauthorized programs that can propagate themselves into computer programs or data and cause destruction or damage to workstation programs, data or networks. **All workstations must have virus detection software installed and running.** Virus detection software must be used to check hard drives at startup or shutdown and at regular intervals of operation, at least once per day. Workstations with enabled floppy or CD drives must have virus detection running constantly. The use of software from public or private sources including computer bulletin boards, and shared public domain software must be preceded by a satisfactory virus scan.

This policy included language about not loading software that was not approved by the Company.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

> Only approved hardware may be connected to Company-controlled networks. All exceptions to this standard must be approved by the Information Security Team. No one may knowingly install on any system - test or production - any software (commercial, shareware or public domain) which is not licensed for use on the specific systems or networks. No software will be placed on the system when appropriate licensing agreements do not exist, except during evaluation periods for which the user has documented permission to install and test software under evaluation. **All software installed on Company-controlled hardware must be approved by the Company.** All workstations and servers should be erased and all software reinstalled whenever control of the system passes from one group or individual to another, for example, when a workstation is reassigned to another user.

The policy also delineated the powers explicitly granted to the security team and the limits of that power.

> The Company **reserves the right to search all hardware, software, systems and related equipment and/or accessories for the purpose of investigating all possible policy violations or security incidents.** Further, the company **reserves the right to seize any assets used or under suspicion of being used in violation of any established policy.**
>
> Security scans and audits may only be performed by the Information Security Team. All other employees, vendors, partners are not authorized to do so. **The Information Security will investigate all possible violations of this policy**. All scans performed by the Information Security Team must be documented prior to actually being performed. The documentation process will allow the IRT to compare current or recent scans with possible security breaches within the network. As the IRT responds to possible scans or attacks, having the database with a history of authorized scans will allow the investigation to continue to build the needed information base.

Two areas that impact the incident response process are configuration management and change management. All devices have undergone a baselining process so that the hardware each system is built on and what software is loaded based on the function of the system is known.

Any requests for changes to the configuration the request must go through change control. In support of that regular change meetings were held weekly. If the circumstances warranted it, a special meeting could be called to address emergency change requests. Lists of changes that were pre-approved developed by each group and were presented at a change meeting. An example of a pre-approved change: The security team may ask any Admin network user who appears to be infected with a virus to remove their machine from the network at a physical level (unplug the cable). The same statement could not be applied to users on the Production network.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

Systems

Several systems were put into place to detect any anomalies on the network and protect the infrastructure. The systems include PIX firewall, Snort intrusion detection system, Demarc security console and Norton AntiVirus.

Forensic Tools

Additional preparation steps included developing a forensics disk for Windows NT and an incident response report. The forensics disk included written instructions about what items to check if there was a suspicious machine.

The forensic disk was developed to gather data that remains in system memory that is normally lost once the system is shutdown or rebooted. The data is then sent across an encrypted channel to a specified service for post-incident forensic analysis. The tool resides wholly on the disk and is kept in a secured area. Therefore the disk and the tools on it are considered secure and trusted.

The disk is made up to be run in three phases. The first phase is Phase1.bat which calls ir.bat and pipes the output to cryptcat, which is the tool used to encrypt and send the data across the network. It then prompts the user to run Phase2.bat.

IR.bat is written to gather the following information:

- Display system date
- Display system time
- Display system uptime
- Display system specific information
- Display who is logged on locally and remotely
- Display detailed process information
- Display remote connection information
- Display a process to port map
- Display all file and security attributes for the c:\winnt\repair\sam._
- Scan c:\ and d:\ for all hidden files
- Scan c:\ and d:\ for all hidden data streams
- Display information for last logon
- Display information for last null logon
- Dump entire event log

The specifics of ir.bat are:

```
echo off
echo *************NT Forensic Script Output**************
echo *************System Date**************
date /t
```


© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

```
echo off
echo *************System Time**************
time /t
echo off
echo *************System Information**************
psuptime
psinfo
echo off
echo *************Users Logged On**************
psloggedon
echo off
echo *************Process Information**************
pslist
echo off
echo *************Open Ports**************
netstat -an
echo off
echo *************Detailed Port Information**************
fport
echo off
echo *************SAM File Statistics**************
filestat c:\winnt\repair\sam._
echo off
echo *************Detailed Info for Last Logon**************
ntlast -s -f -i -r -v -mil
echo off
echo *************Detailed Info for Last Null Logon**************
ntlast -null -mil
echo off
echo *************Event Log Information**************
psloglist
echo *************System Date**************
date /t
echo off
echo *************System Time**************
time /t
echo off
```

Phase2.bat starts the second phase. It calls the file afind.bat which does the following:

- Search c:\ and d:\ for all files that have been accessed in the past 24 hours
- Prompt user to copy information displayed and then run Phase3.bat

Afind.bat is:

```
echo *************File Access c:\ (1 Day History)**************
afind -d 1 c:\
echo off
```


© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

```
echo *************File Access d:\ (1 Day History)**************
afind -d 1 d:\

echo PLEASE COPY THE INFORMATION ABOVE AND RUN PHASE3.BAT
echo off
echo MAKE SURE TO REMOVE THE FORENSIC TOOL CD PRIOR TO THE SYSTEM
POWERING DOWN!!!
```

- The final phase is contained within Phase3.bat. It uses a trusted shutdown command to bring the system down.

This was included to ensure that the technician does not inadvertently issue a compromised command that has an alias to destroy or modify forensic data that resides on the hard drive of the compromised system.

No forensic disk was created for Windows 98. Windows NT was the standard for all desktop workstations, but Windows 98 was the standard for all laptops. There were approximately 150 laptops throughout the organization.

No other forensics tools were developed or put into place because the response stance of the Company was Protect and Eradicate.

Incident Response Plan

The Company had to determine very early on in the security policy development cycle what response stance would be adopted at the corporate level. The two major stance options the Company considered were Protect and Eradicate or Gather Evidence and Prosecute. Several weeks of discussions between management and technical staff were required to determine what option would be chosen. A careful analysis of the risks and benefits of each stance was conducted.

The major factor on the decision to go with Protect and Eradicate was the ability to quickly restore services. It was determined that there was a limited supply of information technology personnel. If their efforts were divided into gathering evidence as well as restoring services, critical business functions might experience a longer downtime than if all resources were concentrated on restoring full virus-free functionality. This was the main reason that management used to determine the response stance.

Additional reasons for choosing the stance include the possibility of adverse publicity. Many companies choose the same stance because of fear of adverse publicity. The Company also wanted to avoid the possibility of getting involved in prolonged legal battles. Furthermore, this was a new initiative. It was determined that the stance that could be implemented most quickly was the optimum. Over time the stance could be re-evaluated and policies and procedures could be changed.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

A blank chart , like the one below, was filled in by technical personnel and a separate blank chart was filled in by management. This allowed the technical personnel to focus solely on technical risks and benefits of each stance. Management was also allowed to focus solely on business reasons why a stance should be chosen.

	Protect and Eradicate	Gather Evidence and Prosecute
Benefits		
Risks		

An incident response plan is composed of several parts. The plan should include a communication plan, communication details, a severity table, an escalation plan and all other associated details. Information about these parts is detailed below.

The incident response plan should be a living document with regularly scheduled review periods. The Company required that the plan be reviewed once a month to ensure that all information was correctly documented. The review period should be at least once a year so that the information does not become completely outdated.

Communication Plan

A communication plan is an important part of incident response even as it is important with project management. Timely, accurate and concise reports should be made to the appropriate levels of personnel at the appropriate times. The communication plan should be mindful of the categorization of the incident.

Communication Details

The communication details should be included as an appendix to the plan. The details should include an updated phone, email, cell, pager list as well as major schedule details such as shift schedule, holiday schedule and vacation schedule.

Severity Table

For example, the Company had chosen a five level severity as part of the ticketing system.

Severity	Explanation
1 - Critical	This level indicates immediate impact against production.
2 – High	This level indicates immediate impact against supporting infrastructure or systems. However, production may continue.
3 - Major	This level indicates a serious problem that, if left unresolved, may impact infrastructure.
4 - Normal	This level indicates a problem that affects an individual or small group of people's ability to do their jobs, but does not indicate a threat against any production systems or general infrastructure.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

5 - Informational	This level indicates an occurrence that may have a security implication in connection with other occurrences but by itself is not a cause for concern.

Figure 0-3 Severity Levels

Escalation Plan

The communication plan should include the escalation plan so that those involved in the incident are aware of the service levels required by management.

Escalation Level	Critical	High	Major	Normal	Informational
Duty Manager	5 minutes				1 hour
IT Center Manager	10 minutes				4 hours
IT Center Director	15 minutes				8 hours
Management Committee	30 minutes		Never	Never	Never

Figure 0-4 Escalation Plan

Information that should be included in communications should include: date and time of the fault or outage, expected duration, the node(s) affected (if relevant), the circuit(s) affected (if relevant), and the details of the fault/outage.

Incident Response Report

The incident response report includes the major areas that every incident handler would be required to provide information.
Here is the report outline:

Section	Title
1	Incident Overview
1.1	Executive Summary
2	Phase I – Notification
2.1	Notification Details
3	Phase II – Assessment
3.1	Communication and Assessment
3.2	Incident Assessment
3.2.1	Determine Response Stance
3.2.2	Implement Fail-Over Solution
3.2.3	Determine Cause of the Incident
3.2.4	Examine Policies and Procedures
3.2.5	Examine Protection Mechanisms
3.2.6	Examine Detection Mechanisms

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

4	Phase III – Recovery
4.1	Recovery Details
4.1.1	How Were Systems Repaired
4.1.2	Were Systems Re-Imaged
4.1.3	When Was Testing Performed
4.1.4	How Was Testing Performed
4.1.5	Who Performed the Tests
5	Phase IV – Closure
5.1	Closure Details
5.1.1	Elapsed Time
5.1.2	Communications

The report outline provides a useful guide for the incident handler about issues that they need to address and information that they need to gather. This is especially important for less experienced analysts.

Training

Two types of training took place at Company X. One level of training was focused on end-users and the other was focused on members of the security team. Training for end-users consisted of a 30 minute PowerPoint presentation delivered by one of the members of the security team. The main focus of the training was the items that the security team identified that end-users could do to assist the security team in making the network as secure as possible. The Top Eight is as follows:

1. Change passwords often
2. Secure workstations when not in use
3. Use current virus protection
4. Use only Company X authorized software
5. Store critical data on a network drive
6. Protect confidential information
7. Challenge those not following security procedures
8. Report security concerns and incidents promptly

The Top Eight laminated into a credit card sized reminder. The reminder card was hole punched so that users could attach it to the lanyard that they wore that held their corporate badge. The card also included the phone numbers of the security hotline and the helpdesk for easy reference. Several users commented that they removed unauthorized software or changed their passwords after going through training.

An important point to remember about training is that users should go through periodic refresher courses to ensure that they continue to remember and focus on security. Most personnel concentrate on their specific job function and forget that they play a part in security.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

Training for incident handlers included training on all the detection tools. Familiarity with Snort and Demarc via classroom training and on-the-job experience was a major component of the program. Each system had a champion that served as the main point of contact on that system. That person developed a one to two-hour training course about the basics of the system. Many also had required reading, which was usually the documentation provided with the product, before staff could attend the class. All team members were required to attend the training.

Incident response tabletops were held. A tabletop is a short meeting where the entire team gathered together and talked through incident scenarios. Tabletops proved to be a very valuable training method. One team member was given a scenario and asked what steps he or she would take. Then the entire group would discuss the proposed steps and determine if they were appropriate especially as to whether they followed the written incident response plan. The junior members of the team regularly stated that tabletops were the most valuable part of the training because it helped them understand the process and the mindset of an incident handler.

As the first part of the Incident Response (IR) process, the Information Security was notified of a potential security incident. The following details outline the first phase in the Incident Response Plan – Notification, which captures all the first contact information, required before activating the Incident Response Team (IRT).

The notification details are captured in a chart that is included in the incident response report. The incident was originally classified a Level 3. Tom opened the case, which led him to be listed as the first contact. The incident was first seen via the Demarc security console and was classified as an incident at 1337 on 17 January.

Assessment

Company X does not break apart Identification, Containment and Eradication. An attempt was made to take the data collected and re-identify it into the appropriate categories.

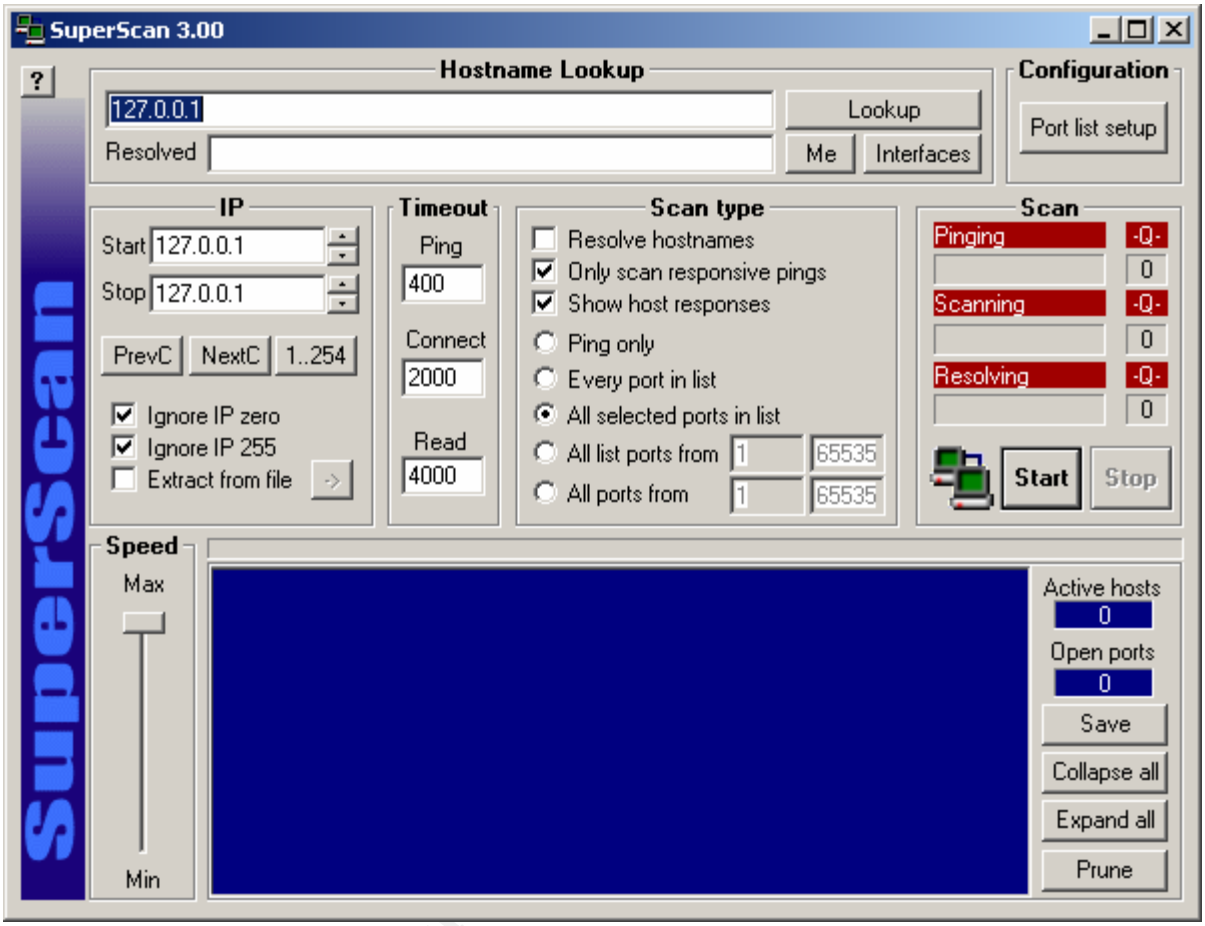
Identification

As has been previously noted, no screen shots were taken to document the process of what was done. No thought or consideration had been given to the value of having such shots available. This is a major shortcoming because it does not allow the report to give a full picture of what occurred. An additional shortcoming is that reports from previous incidents are not as effective as a training tool for new security personnel.

A member of the security team was monitoring the Demarc console and noticed alerts titled “Web-IIS scripts access”. This is one of the generic rules from the default Snort install. This was a new sensor that had not yet been tuned.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

A second member of the team was tasked with downloading a port scanner tool. The tool chosen was SuperScan (http://www.webattack.com/get/superscan.shtml). Below is a screen capture from the opening screen of SuperScan 3.0.



The reason for classifying the traffic as an incident is listed in the chart below labeled Original Notification Communication. This information is captured as part of an effort to build a local knowledge database. The information in the database is intended for use as a supplement to knowledge bases available on the Internet such as the CVEs at cve.mitre.org.

Knowing the location of the incident is important because the manager might need to contact technical specialists at a remote location or to dispatch a member of the security team to the location. In this instance the two locations listed are LocA (the headquarters building where the security team is located) and LocB (the secondary headquarters located approximately three blocks away). The LocB building does not have dedicated technical staff. All needs are handled by dispatching personnel from LocA.

Normally the Security DM would have dispatched a team member to LocB. However, since the device that appeared to be infected at LocB was a laptop, it was decided that

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

it would be more prudent to have the machine owner bring the device to the headquarters building.

No virus scans were conducted on the machines at this point. That should have occurred before the incident was classified an infection by Millenium worm.

The Incident Manager assigned was Tom, a junior member of the team. Based on the fact that only one machine indicated infection, it was determined that this incident would be good for on-the-job training purposes. Additional personnel were identified to assist the IM with the process.

Brand Y Ticketing Case ID	50819
Who Initiated First Contact	Tom
How was the IRT Contacted	Demarc
When was the IRT Contacted	Approximately 1337 on 17 January 2002
What were the symptoms	Multiple Web-IIS alerts were seen on Demarc. A port scan of the source machines showed the UDP 1338 was open.
Where did the incident occur	LocA and LocB
Who was the IM Assigned	Tom
Technical Specialist(s) Assigned	Mark, Nathan, Mike
IM's initial Severity Level	3 – Major
IM's initial Categorization	Virus-Trojan-Worm

Figure 0-5 Original Notification Communication

Containment

The containment strategy used was the most draconian option available. The suspected worm was contained by physically removing systems from the network. No commands were issued remotely.

The IS Security Duty Manager was alerted about the case and briefed regarding the incident at about 1340 on 17 January 2002. The user removed the infected machine at the remote site physically from the network.

At that point two other machines started to exhibit the same behavior of attempting to contact the same IP address of 207.68.78.1. The users were called. One user picked up the phone and unplugged the network cable at the request of the security team. The other user was not available. A member of the security team went to the floor where the machine was located and unplugged it. A note was affixed to the monitor advising the user of the actions the security team had taken. The user was asked to call the security team before reconnecting the system.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

Because of the possibility that other machines might be infected with a trojan with a number of capabilities, a decision was made turn off Internet access to ensure that no internal company data could be sent out.

The following chart is included in the incident response report to capture the details of the update. The severity of the incident was upgraded because a decision was made to shut down the Internet access for Admin network users. The possibility that information about the infected machines could be sent to a server on the Internet was unacceptable to management.

The reason for the closure was that the list of systems exhibiting the behavior changed from one to three. One of the machines listed belonged to a user in the legal department. Information was contained locally on that machine that should not leave the confines of the company. To contain the spread the three machines were physically unplugged from the network and then the Internet connection was shut down.

Associated tickets capture in report form all of the tickets that are linked to the main ticket. One ticket (47698) was about an incident that took place on one of the affected systems one week prior. The system owner called to report that his domain password changed and he had not changed it. The help desk opened and closed the ticket without contacting security.

Updated Severity Level	2 – High
Updated Incident Category	Virus – Trojan – Worm
System(s) Affected	• LocA-LAPTOP-275 • LocA-DESKTOP-740 • LocA-DESKTOP-026
Associated Tickets	47698, 50817
Fail-over Solution Initiated?	Yes
Systems Backed Up?	No
Initial Incident Cause	Demarc reported multiple attempts to reach 207.68.78.1 from internal hosts.
Immediate Protective Measures Taken	Unplugged machines exhibiting symptoms from the network. Performed anti-virus scans of the entire LocA domain. Performed independent anti-virus scans of allegedly infected machines. Checked if any registry settings had been changed.

Figure 0-6 Updated Notification Communication

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

The severity was upgraded to Severity Two because the decision was made that the connection to the Internet was shut down.

Factors involved in response stance determination:

- To identify quickly the source of the suspected infection.
- To eliminate any possible propagation of the worm.
- To remove the suspected worm or virus from any infected machine.
- To update virus scanners to prevent any future infections.

No fail-over solution plan is in place for Admin workstations. However, a recommended fail-over solution would be to provide new workstations to users with machines that show symptoms of a possible infection and cannot be cleaned.

The possible causes of the incident are:

1) An infected E-mail attachment*.
2) An infected downloaded file*.
3) An exploitation of an installed peer-to-peer or peer-to-many application
4) An infected file introduced with removable media.

*These machines could have had access to the Internet via the Admin network or any Internet Service Provider.

The possible causes listed are the most prevalent methods of infection for a worm. However, there are other infection methods that could have been used. Based on the fact that forensic data from Company X was deleted a definitive choice cannot be made. In the case of Business Y, it is highly probable that the cause of the infection was an infected file downloaded from IRC.

The IM and the technical specialists identified traffic samples that follow standard patterns for worms in a network. These were multiple and consecutive attempted connections to different machines within or outside of the network. At approximately 1300 there were 3234 attempts and at approximately 1400 there were 3703 attempts. These machines also were found to have UDP port 1338 open. This port is generally indicative of the Millennium Worm.

The IM downloaded an evaluation copy of a port-scanning tool from the Internet to perform additional analysis. It was noted at that time that a port-scanning tool was a necessary item that was currently missing from the forensic toolkit.

To prevent the suspected worm from possibly infecting the rest of the network the following actions were taken:

1. Identify any machines that could possibly be infected with the virus through virus scans (with updated virus definitions) and monitor Demarc for additional indications.
2. Identify the possible source(s) of the possible infection.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

3. Quarantine from the network any machines that show any symptoms of being infected.
4. Block any access to the external IP that the worm was apparently attempting to contact in the Internet firewall.
5. Remove Internet access temporarily to prevent the dissemination of any information outside the network and from any propagation of the possible virus.
6. Check the suspected computers to assess if they are infected with the worm and attempt to remove it from the machine.
7. Remove any instances of the worm found and any authorized software from the machines.
8. Monitor systems to make sure infection symptoms do not reappear.

Eradication

The machines had been virus scanned and an abbreviated forensic scan took place. The virus scan showed no signs of infection and nothing was found during the forensics effort. The forensic effort mainly consisted of checking the history in Internet Explorer. However the users had an opportunity to clear their history so much important data was lost. Since no infection was found, no steps had to be taken to eradicate the infection from the network.

The machines were re-attached to the network. Security personnel monitored the machines for approximately one hour. After that period they were determined to be safe. A note was made in the security logbook that a continued vigilance should be in place.

The specific protection mechanisms for this incident are:

- To update the virus definitions on the machines in the Admin network.
- To perform further scans of the domain to identify any possibly infected machines.
- To remove any viruses detected from network workstations or servers.
- To continue to monitor the network for any symptoms of re-infection.

Additional protection mechanisms include reinforcing the policy of not allowing unauthorized software to be loaded on company machines.

Recovery

The goal of the Recovery phase is to bring the environment back to an operational status as soon as possible. None of the Production machines showed any symptoms of being infected.

A virus scans was performed on all of the Admin NT machines by running a centralized scan of the entire Windows domain. The machines that showed symptoms were

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

specifically scanned by running the antivirus locally since they were all disconnected from the network.

No systems were re-imaged. A peer-to-peer application was removed from LocA-LAPTOP-127 by the security team. No changes were made to routers or firewalls or any other infrastructure device. The most current set of virus definitions was manually downloaded and installed. Normally the virus definitions are checked automatically at night and downloaded only if new ones are detected.

It was suggested that an after action report be written to capture deficiencies in processes and procedures. Management tasked security to conduct additional training on incident identification. They stated that an after action report would be a waste of time and resources based on the fact that no machines were actually infected.

The original case was opened on the January 17, 2002 at approximately 1359. The case was closed on the same day at approximately 1426 for an elapsed time of 2 hours and 26 minutes.

The IT Center Manager, Security Duty Manger, Security Team on shift, and Functional Area Specialists were all notified of resolution. The Incident Report for 50819 was reported as part of the Management Committee report for 18 January 2002.

Ticket 50917 was opened to track all follow up analysis of the three suspected systems. This ticket was opened as a Severity Five (Informational).

The follow-up included performing a full forensics scan of the machines involved. The scan involved looking at the history in Internet Explorer and a final virus scan of each machine. The forensics were problematic because proper chain of custody did not occur on the machines involved in this incident. The machine owners were able because they had unattended access to the machines to remove software and clear histories in the time between the incident and the scan the following day.

Following the incident, no specific monitoring of the three machines that were suspected to be infected was done. The only post-incident monitoring to watch for the problem to reoccur was a note was added to the Demarc system to watch for activity on port 1338.

According to the incident response plan, informational tickets do not require a formal report write-up. The plan should be updated to note that low level tickets that are follow-ups to high level tickets should be written up so that a full picture of the incident can be gained.

Lessons Learned

The most important lesson learned during this pseudo-incident was that team members must relax, collect themselves, and remain calm during an incident. If team members

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

are cool headed they are more likely to remember and follow procedures and to use all available tools and resources. All other lessons are listed in no particular order.

Other lessons learned included updating the Information Security Policy with a statement that laptops may not be connected to the Internet without antivirus running. In addition, they should have a personal firewall loaded with settings in accordance with the company security policy.

Currently no standard for personal firewalls exists within the company. Choosing a standard and installing personal firewalls should rectify this gap. There are many products that are available commercially. With approximately 150 laptops out of 2000 users, the probability of users connecting to the Internet for personal use at home is high.

Every incident, whether handled well or poorly, needs to undergo a thorough analysis after it is over. This process allows the company to improve its processes and permits the security team members to improve and share their knowledge and skills. The post-incident analysis should involve a meeting with all personnel involved. The face-to-face meetings could be individual interviews with each person or a group meeting. Each format has its benefits and detriments so every company must determine which format works best for them. Complete notes of the meeting should be taken so that the insights presented are captured for future use.

As part of the investigation and response to this event, a need to ensure that a secure location for seized equipment was defined. All IRT members must be properly trained to implement a well organized chain-of-custody.

Investigation of possible incidents needs to have a causation validation process. In this instance there was very little information available on the Millennium Worm. This lack of information should not translate to making a mistake in categorization of the incident.

Training for security personnel should be documented via a sign-up sheet. It should be validated that all personnel who are required to attend training have actually attended training. It can then be added to their personnel folders.

An additional level of training needs to be developed and conducted. Currently there is training for end users and members of the security team. Based on the incident response structure, employees who are associated with incident response who are not direct members of the security team should also receive training. This could be the same training as the security team or a slightly modified course.

Further integration between groups that open and close tickets was noted. The ticket about the user whose password did not work should have been courtesy copied to the security team. A meeting should be held to discuss with each team about what types of issues should be copied to the security team for review and oversight.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

A fail over plan should be developed or a stockpile of spares should be maintained for all Admin machines. Currently only Production servers have a fail over plan and spares are available for production terminals.

Forensics disks and instructions need to be developed for every operating system in use at the company instead of just the most prevalent. Each operating system, including different versions of the same OS, have important differences that may not be fully known or understood by every analyst that may have to work an incident. Additionally each toolkit needs to be tested in several settings to ensure that the tool works in the expected manner under ideal and not-so-ideal situations.

In fact, Windows experts may need to perform forensics on a UNIX box. If that is the case, the instructions need to be clearly written and easy to use. All personnel should undergo training on the steps to perform when doing forensics. They should all have the chance to perform forensics at least once on every type of machine that they may be called upon to work within the company.

A port-scanning tool needs to be installed on an Admin machine and a Production machine to use for incident analysis.

A screen shot tool should be installed on all security machines so that during an incident screen captures can be generated. Those captures should, at a minimum, be included in the ticketing system history. If the incident is significant and a report is written, the captures can be used to illustrate the text description of the incident.

A screen shot program should be installed on the security machines so that captures can be made of screens that show possibly malicious activity. A secure screen capture tool should be added to the forensic toolkit so that appropriate screen shots of suspected machines can be taken.

A harmonization process between what is written in the security policy and incident response policy needs to occur against the procedures that were developed to support the policies. Based on the short time period that security was in place prior to the incident, many of the procedures were not fully formed based on what was required in policy. The security policy states,

> "All scans performed by the Information Security Team must be documented prior to actually being performed. The documentation process will allow the IRT to compare current or recent scans with possible security breaches within the network. As the IRT responds to possible scans or attacks, having the database with a history of authorized scans will allow the investigation to continue to build the needed information base."

However, during the possible Millenium incident the scan received appropriate approvals from management and the approvals were documented. However no database for incidents had been created. Most of the pertinent information should be

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

available in the incident response report, but decisions about what should be captured in the database should be made. The decisions have important long-term effects, so the process of deciding what should be captured should not be taken lightly.

Following the incident an email was sent out from the CIO enforcing the policies of the company as described in the Security Policy. Having the highest levels of management involved in an incident is a good way to gain their support of the activities of the security team. A strong way to reinforce the authority of the security team is to have them send out messages to employees after an incident as part of the security training program.

Upper management really found the charts contained within the incident response report to be the most valuable part according to anecdotal evidence. Creating a "super chart" that encapsulates all the pertinent information from the case is a worthwhile effort. Having a formal written report is important for archival purposes. The super chart would include the following fields:

Brand Y Ticketing Case ID	
Who Initiated First Contact	
How was the IRT Contacted	
When was the IRT Contacted	
What were the symptoms	
Where did the incident occur	
Who was the IM Assigned	
Technical Specialist(s) Assigned	
IM's initial Severity Level	
IM's initial Categorization	
Updated Severity Level	
Updated Incident Category	
System(s) Affected	
Associated Tickets	
Fail-over Solution Initiated?	
Systems Backed Up?	
Immediate Protective Measures Taken	
Additional Protective Measures Taken	
Final Measures Update	
Final Severity Level	
Final Incident Category	
Follow-up Planned	

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

References

Boxmeyer, Jim (undated). Possible Trojan/BackDoor. (www document). http://www.onctek.com/trojanports.html (reviewed January, 2002).

Cisco (undated). White Paper – Cisco Express Forwarding (CEF). (www document). http://www.cisco.com/warp/public/cc/pd/iosw/iore/tech/cef_wp.htm (reviewed March, 2002).

Gilbert, Howard (2 February 1995). Introduction to TCP/IP. (www document). http://www.yale.edu/pclt/COMM/TCPIP.HTM (reviewed March, 2002).

G-Lock Software (undated). ... \ Port Scanner \ Trojans Port List \ Millenium Worm. (www document). http://www.glocksoft.com/trojan_list/Millenium_Worm.htm (reviewed January, 2002).

Howard, Aaron. (5 December 2000). Millennium Trojan. (www email document). http://cert.uni-stuttgart.de/archive/incidents/2000/12/msg00017.html

Howard, Aaron. (9 December 2000). RE: Millennium Trojan. (www email document). http://cert.uni-stuttgart.de/archive/incidents/2000/12/msg00024.html

ISS X-Force Database (undated). backdoor-millenium (3111). (www document). http://www.iss.net/security_center/static/3111.php (reviewed January, 2002).

Kerby, Fred (August, 2001). Incident Handling Foundations: Malicious Software (SANS Courseware online).

Lemos, Robert (April, 2002). Klez Worm's on the Loose Again. (www document). ZDNet (reviewed April 2002).

Lo, Joseph (21 January 2002). Trojan Horse Attacks. (www document). http://www.irchelp.org/irchelp/security/trojan.html.

Road Runner Security - Time Warner Cable Raleigh (undated) Computer TCP/UDP Ports. (www document). http://www.aroundtownnc.com/security/computer_ports.html (reviewed January, 2002).

SANS Institute (September, 2001). Computer Security Incident Handling Step by Step. SANS Institute.

Symantec (undated). Backdoor.Millenium (www document). http://securityresponse.symantec.com/avcenter/venc/dyn/7670.html (reviewed January, 2002).

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

Vision, Max (1999). Origin and Brief Analysis of the Millennium Worm (www document). http://openbsd.org.br/ouah/mworm.htm (reviewed January, 2002).

von Braun, Joakim (9 February 2001). Intrusion Detection FAQ – What port numbers do well-known trojan horses use? (www document). http://www.sans.org/newlook/resources/IDFAQ/oddports.htm (reviewed January, 2002)

© SANS Institute 2000 - 2002, Author retains full rights.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

Appendix A

Below is an email from Business Y about a possible infection by Millenium worm or a variant.

I just caught a consultant we had hired using mIRC on our dime and later discovered his computer was infected with a program purporting to be the Millennium Trojan.

I think, however, that this may be a new variant as the latest virus-defs from Norton (11/27/00) don't recognize it as a virus or trojan.

I have analyzed it quite fully and would be willing to share my travails with interested parties. It was originally written with Delphi and I have recreated most of the source code. Also, if anyone else has come across this, I'd be interested in knowing what you have found.

For the rest of you, beware of machines trying to connect to Internet IP addresses on port 6667 for no obvious reason and lookout for any file named kernel32.vxc hidden away in the c:\windows\system directory. This program is a key logger and then some...like NetBus and Back Orifice. But it appears as though it connects to IRC servers and accepts commands as an IRC bot.

I believe it will only run properly on Win9x boxen, not NT/2000.

© SANS Institute 2000 - 2002, Author retains full rights.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

Appendix B

Below is the text from the second email from Business Y about a machine possibly infected by Millennium worm or some sort of variant.

Well, since I have received several requests, I'll include a more full analysis of this trojan.

Please note that this is all from reviewing the executable, not actually running it. I'm not confident enough in my abilities to keep it from doing damage if I run it.

So, a little background...

We hired an outside consultant to help us set up an accounting/distribution software package. He came in and was seated at an open PC.

We block a lot of things, but I have heretofore been lenient on Outbound traffic (allowing all machines inside with valid source addresses to establish connections on any local port > 1023 to any other machine outside our network on any port > 1023.

The idea was to allow our users to run IRC, MSN Messenger, AOL Instant Messenger, Yahoo Instant Messenger, ICQ, RealAudio, etc. (Modification of this policy is already underway...)

But, even though we allow(ed) these outbound connections, they are all logged to a central logging machine and that log is constantly scrolling in the background of my screen.

So one day I'm working away and notice a bunch of connections on Destination port 6667...this peaks my curiosity because I KNOW nobody at our company uses IRC but me...and this wasn't me.

Here are the actions I took ...

1. nbtstat -A internal.source.ip.address

this returned me the following:

```
NetBIOS Remote Machine Name Table
Name            Type            Status
---------------------------------------------
PCNAME          <00> UNIQUE         Registered
OURDOMAIN           <00> GROUP          Registered
PCNAME          <03> UNIQUE         Registered
```


© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

CONSULTANT <03> UNIQUE Registered

MAC Address = xx-xx-xx-xx-xx-xx

The key here was that is showed the source IP address in question was in use by our consultant.

Now usually I'd just call the user up and say, "What are you doing?" but this being a consultant, I decided it was best to be more discreet.

We use VNC on all our internal machines for support-related issues. So, I checked him out.

2. I used VNC to view his screen and take screen shots of him chatting via mIRC for nearly an hour.

3. After this, we let him go and immediately blocked outbound connections on all ports > 1023. However, we left the machine on. Then I started noticing blocked connections on port 6667 from that machine. Blocked attempts looked like this:

denied tcp x.x.x.2(1068) -> 130.243.43.71(6667)
denied tcp x.x.x.2(1040) -> 151.189.12.20(6667)
denied tcp x.x.x.2(1376) -> 194.75.152.237(6667)
denied tcp x.x.x.2(1029) -> 198.139.244.22(6667)
denied tcp x.x.x.2(1500) -> 198.63.2.192(6667)
denied tcp x.x.x.2(1336) -> 198.88.88.99(6667)
denied tcp x.x.x.2(1348) -> 199.232.159.166(6667)
denied tcp x.x.x.2(1046) -> 209.25.152.162(6667)
denied tcp x.x.x.2(1072) -> 209.254.98.88(6667)
denied tcp x.x.x.2(1049) -> 212.43.196.5(6667)

(Note: destination ip addresses were not attempted in this order, this is a sorted list of unique destination IPs...)

4. I VNC'ed over to it and saw NO applications running. Nothing in the task list at all.

5. So, I created bootable Norton Antivirus 5.0 disks with the latest virus defs (11/27/00) and went to that machine and scanned it. Nothing. So, I started MSINFO32.EXE to check loaded modules and found something called kernel32.vxc was in memory but it had no version info. And it was in \windows\system... I scanned it specifically again, NAV said not a virus. It was attrib-ed as HIDDEN/SYSTEM.

6. I copied it to a floppy to take to another machine for testing and renamed it BADGUY.EXE.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

7. QuickView of the EXE showed very little other than that the EXE was mangled (packed) to prevent viewing like this.

8. So I checked backlogs of my bugtraq e-mails and found a few sites with reverse engineering tools. www.suddendischarge.com was most helpful.

9. I downloaded several tools from sudden discharge: 1) Universal File Scanner (fs11-27-00.zip), 2) Anti-Aspack 0.2 (unaspack02.zip), 3) DeDe 2.431 (dede2431full.zip), 4) PE Explorer 1.0 Beta (pex_b090.zip)

a. I used fs to determine the following:

```
file scanner by SMT
+--------------------------e:\ahoward\badguy1.exe----------------------
------+
¦extension: executable file
¦
+---------------------------MZ-EXE DOS
executable---------------------------+¦
¦¦sizes: header 28, relocs 0, empty 644, image 192, overlay 291488 bytes ¦¦
¦¦dos/exe DOS stub from Borland tlink32 ¦¦
¦+---------------------------Portable
executable---------------------------¦¦
¦¦subsystem: Win32 GUI, cpu: i386 ¦¦
¦¦linktime: Fri, 19.Jun.1992 at 17:22.17 (UTC 22:22.17) ¦¦
¦¦checksum: correct ¦¦
¦¦linker: Borland TLINK/TLINK32 ¦¦
¦¦sizes: stub 64, header 960, image 291328, overlay 0 ¦¦
¦¦pe/exe.packer ASPack 1.061b,1.07b (type 2)------------------------unpacker¦¦
¦+------------------------------------------------------------------ -----+¦
+---------------------------------------------------------------------- ------+
```

I thought one thing was odd about this...the linktime says 19.Jun.1992...so was this program REALLY compiled in 1992? Not likely. I imagine a little hex-editing of the file and you can make the link date whatever you want.

Or change the date on your system before linking.

b. Since I couldn't tell much more about the file without unASPack-ing it, I used unaspack to remove the packing and created badguy2.exe, then fs showed the following...

```
file scanner by SMT
+--------------------------e:\ahoward\badguy2.exe---------------------------+
¦extension: executable file ¦
+---------------------------MZ-EXE DOS executable---------------------------+¦
```


© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

```
¦¦sizes: header 28, relocs 0, empty 644, image 192, overlay 622240 bytes ¦¦
¦¦dos/exe DOS stub from Borland tlink32 ¦¦
¦+---------------------------Portable executable---------------------------¦¦
¦¦subsystem: Win32 GUI, cpu: i386 ¦¦
¦¦linktime: Fri, 19.Jun.1992 at 17:22.17 (UTC 22:22.17)¦¦
¦¦checksum: correct¦¦
¦¦linker: Borland TLINK/TLINK32¦¦
¦¦sizes: stub 64, header 960, image 622080, overlay 0¦¦
¦+-----------------------------------------------------------------------+¦
+-------------------------------------------------------------------------+
```

(No packing now)

c. I used DeDe to disassemble it and generate the attached form1.pas file. (see below)
(Note: I removed the password as I see no need for it to be included...)

d. I generated a strings reference from the "source" DeDe creates in strings.txt. From this we can tell the trojan will accept a number of commands and take certain actions based on those commands.

e. I used PE Explorer to grab out a little more (saved as nmshow.pas) Which shows they are using a component from NetMasters in this trojan...

Source files created by DeDe and PE Explorer are not all attached as I keep getting my message rejected for being over 3000 lines.

...but I think what's here explains well enough.

```
* Possible String Reference to: 'éyšøÿëë^[‹å]Ã'
* Possible String Reference to: ' :'
* Possible String Reference to: ' :'
* Possible String Reference to: 'PC_END'
* Possible String Reference to: '^[‹å]Ã'
* Possible String Reference to: 'TROJAN_CLOSED :void'
* Possible String Reference to: 'windows.dll'
* Possible String Reference to: 'KeyHook_Start'
* Possible String Reference to: 'é¬£øÿëè_^[‹å]Ã'
* Possible String Reference to: '\Kernel32.vxc /nomsg'
* Possible String Reference to: 'Kernel32'
* Possible String Reference to: '#SquashCentre'
* Possible String Reference to: 'v.1.6.'
* Possible String Reference to: 'handle.ini'
* Possible String Reference to: 'Handle'
* Possible String Reference to: 'MainHandle'
* Possible String Reference to: 'Ã•@'
* Possible String Reference to: 'software\microsoft\windows\currentversion\setup'
```


© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

* Possible String Reference to: 'sysdir'
* Possible String Reference to: '\Windows.dll'
* Possible String Reference to: 'click'
* Possible String Reference to: '\Windows.dll'
* Possible String Reference to: '49 33 x'
* Possible String Reference to: '50 34 x'
* Possible String Reference to: '51 163 x'
* Possible String Reference to: '52 36 x'
* Possible String Reference to: '53 37 x'
* Possible String Reference to: '54 94 x'
* Possible String Reference to: '55 38 x'
* Possible String Reference to: '56 42 x'
* Possible String Reference to: '57 40 x'
* Possible String Reference to: '48 41 x'
* Possible String Reference to: '188 44 60'
* Possible String Reference to: '190 46 62'
* Possible String Reference to: '191 47 63'
* Possible String Reference to: '186 59 58'
* Possible String Reference to: '192 39 64'
* Possible String Reference to: '222 35 126'
* Possible String Reference to: '219 91 123'
* Possible String Reference to: '221 93 125'
* Possible String Reference to: '189 45 95'
* Possible String Reference to: '187 61 43'
* Possible String Reference to: '223 96 172'
* Possible String Reference to: 'Windows.dll'
* Possible String Reference to: 'KeyHook_Start'
* Possible String Reference to: '_^[‹å]Ã'
* Possible String Reference to: 'é ½øÿë‡_^[‹å]Ã'
* Possible String Reference to: 'REQUESTLOGIN'
* Possible String Reference to: 'LOGON_GRANTED :Welcome to the millenium trojan. '
* Possible String Reference to: ' Awaiting commands.'
* Possible String Reference to: 'LOGON_GRANTED :Welcome to the millenium trojan. '
* Possible String Reference to: ' Awaiting commands.'
* Possible String Reference to: 'ICONS_HIDE'
* Possible String Reference to: 'progman'
* Possible String Reference to: 'SYSTEM_MESSAGE :Desktop icons hidden'
* Possible String Reference to: 'ICONS_SHOW'
* Possible String Reference to: 'progman'
* Possible String Reference to: 'SYSTEM_MESSAGE :Desktop icons shown'
* Possible String Reference to: 'SYSKEYS_OFF'
* Possible String Reference to: 'SYSTEM_MESSAGE :System keys disabled'
* Possible String Reference to: 'SYSKEYS_ON'
* Possible String Reference to: 'SYSTEM_MESSAGE :System keys enabled'
* Possible String Reference to: 'DESKTOP_LOCK'
* Possible String Reference to: 'SYSTEM_MESSAGE :Desktop Locked'

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

* Possible String Reference to: 'DESKTOP_UNLOCK'
* Possible String Reference to: 'SYSTEM_MESSAGE :Desktop Unocked'
* Possible String Reference to: 'DESKTOP_WALLPAPER'
* Possible String Reference to: 'SYSTEM_MESSAGE :Wallpaper changed to '"
* Possible String Reference to: 'PLUGINS_LIST'
* Possible String Reference to: 'PLUGIN_NAME'
* Possible String Reference to: 'PLUGIN_ADD'
* Possible String Reference to: 'PLUGIN_ADDED :Plugin '"
* Possible String Reference to: '" from '"
* Possible String Reference to: '" has been added'
* Possible String Reference to: 'PLUGIN_REMOVE'
* Possible String Reference to: 'PLUGIN_REMOVED :Plugin '"
* Possible String Reference to: '" has been removed'
* Possible String Reference to: 'RELAY_ADDRESS'
* Possible String Reference to: 'RELAY_PORT'
* Possible String Reference to: 'RELAY_CONPORT'
* Possible String Reference to: 'RELAY_START'
* Possible String Reference to: 'RELAY_STOP'
* Possible String Reference to: 'KEYS_DISABLE_ALL'
* Possible String Reference to: 'KEY_MESSAGE :keyboard disabled'
* Possible String Reference to: 'KEYS_ENABLE_ALL'
* Possible String Reference to: 'KEY_MESSAGE :keyboard enabled'
* Possible String Reference to: 'KEYS_DISABLE'
* Possible String Reference to: 'KEY_MESSAGE :keys '"
* Possible String Reference to: '" disabled'
* Possible String Reference to: 'KEYS_ENABLE'
* Possible String Reference to: 'KEY_MESSAGE :keys '"
* Possible String Reference to: '" enabled'
* Possible String Reference to: 'KEY_LISTEN_START'
* Possible String Reference to: 'Windows.dll'
* Possible String Reference to: 'KeyHook_Start'
* Possible String Reference to: 'KEY_MESSAGE :Sending keystrokes'
* Possible String Reference to: 'KEY_LISTEN_STOP'
* Possible String Reference to: 'KEY_MESSAGE :Keystroke sending is now off'
* Possible String Reference to: 'SYSTEM_SCREENSHOT'
* Possible String Reference to: 'SCREENSHOT_INSIZE :'
* Possible String Reference to: 'SCREENSHOT_INITIALIZE :764371'
* Possible String Reference to: 'FILE_FILENAME'
* Possible String Reference to: 'FILE_DSET'
* Possible String Reference to: 'FILE_GET_ATTRIBUTES'
* Possible String Reference to: 'FILE_ATTRIBUTE_ARCHIVE :1'
* Possible String Reference to: 'FILE_ATTRIBUTE_ARCHIVE :0'
* Possible String Reference to: 'FILE_ATTRIBUTE_COMPRESSED :1'
* Possible String Reference to: 'FILE_ATTRIBUTE_COMPRESSED :0'
* Possible String Reference to: 'FILE_ATTRIBUTE_DIRECTORY :1'
* Possible String Reference to: 'FILE_ATTRIBUTE_DIRECTORY :0'

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

* Possible String Reference to: 'FILE_ATTRIBUTE_HIDDEN :1'
* Possible String Reference to: 'FILE_ATTRIBUTE_HIDDEN :0'
* Possible String Reference to: 'FILE_ATTRIBUTE_NORMAL :1'
* Possible String Reference to: 'FILE_ATTRIBUTE_NORMAL :0'
* Possible String Reference to: 'FILE_ATTRIBUTE_OFFLINE :1'
* Possible String Reference to: 'FILE_ATTRIBUTE_OFFLINE :0'
* Possible String Reference to: 'FILE_ATTRIBUTE_READONLY :1'
* Possible String Reference to: 'FILE_ATTRIBUTE_READONLY :0'
* Possible String Reference to: 'FILE_ATTRIBUTE_SYSTEM :1'
* Possible String Reference to: 'FILE_ATTRIBUTE_SYSTEM :0'
* Possible String Reference to: 'FILE_ATTRIBUTE_TEMPORARY :1'
* Possible String Reference to: 'FILE_ATTRIBUTE_TEMPORARY :0'
* Possible String Reference to: 'SYSTEM_MONITOR_OFF'
* Possible String Reference to: 'SHUTDOWN_MESSAGE :Monitor turned off'
* Possible String Reference to: 'SYSTEM_MONITOR_ON'
* Possible String Reference to: 'SHUTDOWN_MESSAGE :Monitor turned on'
* Possible String Reference to: 'SYSTEM_RESTART'
* Possible String Reference to: 'SHUTDOWN_MESSAGE :Restarting system'
* Possible String Reference to: 'SYSTEM_SHUTDOWN'
* Possible String Reference to: 'SHUTDOWN_MESSAGE :Shutting down system'
* Possible String Reference to: 'SYSTEM_FORCE'
* Possible String Reference to: 'SHUTDOWN_MESSAGE :Forcing down system'
* Possible String Reference to: 'SYSTEM_POWEROFF'
* Possible String Reference to: 'SHUTDOWN_MESSAGE :Powering down system'
* Possible String Reference to: 'SYSTEM_LOGOFF'
* Possible String Reference to: 'SHUTDOWN_MESSAGE :Logging off current user'
* Possible String Reference to: 'DRIVE_SERIAL'
* Possible String Reference to: 'DRIVE_NAME :Drive name of drive "'
* Possible String Reference to: '" is "'
* Possible String Reference to: 'DRIVE_SERIAL :Serial number of drive "'
* Possible String Reference to: '" is "'
* Possible String Reference to: 'DRIVE_OPEN'
* Possible String Reference to: 'SYSTEM_MESSAGE :Drive "'
* Possible String Reference to: '" has been opened'
* Possible String Reference to: 'DRIVE_CLOSE'
* Possible String Reference to: 'SYSTEM_MESSAGE :Drive "'
* Possible String Reference to: '" has been closed'
* Possible String Reference to: 'FILE_EXECUTE'
* Possible String Reference to: 'FILE_MESSAGE :"'
* Possible String Reference to: '" was executed normally'
* Possible String Reference to: 'FILE_EXECUTE_INVIS'
* Possible String Reference to: 'FILE_MESSAGE :"'
* Possible String Reference to: '" was executed invisibly'
* Possible String Reference to: 'FILE_EXECUTE_NONEXE'
* Possible String Reference to: 'FILE_MESSAGE :"'
* Possible String Reference to: '" was opened'

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

```
* Possible String Reference to: 'FILE_EXECUTE_NONEXE_INVIS'
* Possible String Reference to: 'FILE_MESSAGE :"'
* Possible String Reference to: '" was opened invisibly'
* Possible String Reference to: 'FILE_DELETE'
* Possible String Reference to: 'FILE_MESSAGE :"'
* Possible String Reference to: '" was deleted'
* Possible String Reference to: 'FILE_MESSAGE :"'
* Possible String Reference to: '" was not deleted'
* Possible String Reference to: 'FILE_COPY_LOC1'
* Possible String Reference to: 'FILE_COPY_LOC2'
* Possible String Reference to: 'FILE_MESSAGE :"'
* Possible String Reference to: '" was copied to '
* Possible String Reference to: 'FILE_MESSAGE :"'
* Possible String Reference to: '" failed to copy to "'
* Possible String Reference to: 'FILE_RENAME_NAME1'
* Possible String Reference to: 'FILE_RENAME_NAME2'
* Possible String Reference to: 'FILE_MESSAGE :"'
* Possible String Reference to: '" was renamed to "'
* Possible String Reference to: 'FILE_MESSAGE :"'
* Possible String Reference to: '" failed to rename to "'
* Possible String Reference to: 'FTP_PORT'
* Possible String Reference to: 'FTP_MAX'
* Possible String Reference to: 'FTP_START'
* Possible String Reference to: 'FTP_MESSAGE :FTP server started on port '
* Possible String Reference to: ' for '
* Possible String Reference to: ' connections'
* Possible String Reference to: 'FTP_STOP'
* Possible String Reference to: 'FTP_MESSAGE :FTP server stopped'
* Possible String Reference to: 'ADMIN_SET_PASSWORD'
* Possible String Reference to: 'SYSTEM_MESSAGE :Password set to "'
* Possible String Reference to: 'ADMIN_GETOLDPASSWORD :void'
* Possible String Reference to: 'ADMIN_CLEAR_PASSWORD'
* Possible String Reference to: 'SYSTEM_MESSAGE :Password cleared'
* Possible String Reference to: 'SYSTEM_MESSAGE :Password not cleared'
* Possible String Reference to: 'ADMIN_OLDPASSWORD'
* Possible String Reference to: 'SYSTEM_MESSAGE :Password changed to "'
* Possible String Reference to: 'SYSTEM_MESSAGE :Password not changed'
* Possible String Reference to: 'PROCESS_LIST_ALL'
* Possible String Reference to: 'PROCESS_BEGINLIST :All Processes'
* Possible String Reference to: 'PROCESS_LIST_VISIBLE'
* Possible String Reference to: 'PROCESS_BEGINLIST :Visible Processes'
* Possible String Reference to: 'PROCESS_LIST_INVISIBLE'
* Possible String Reference to: 'PROCESS_BEGINLIST :Inisible Processes'
* Possible String Reference to: 'PROCESS_MINIMIZE'
* Possible String Reference to: 'PROCESS_MESSAGE :"'
* Possible String Reference to: '" was minimized'
```


© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

```
* Possible String Reference to: 'PROCESS_MAXIMIZE'
* Possible String Reference to: 'PROCESS_MESSAGE :"'
* Possible String Reference to: '" was maximized'
* Possible String Reference to: 'PROCESS_RESTORE'
* Possible String Reference to: 'PROCESS_MESSAGE :"'
* Possible String Reference to: '" was restored'
* Possible String Reference to: 'PROCESS_HIDE'
* Possible String Reference to: 'PROCESS_MESSAGE :"'
* Possible String Reference to: '" was made invisible'
* Possible String Reference to: 'PROCESS_SHOW'
* Possible String Reference to: 'PROCESS_MESSAGE :"'
* Possible String Reference to: '" was made visible'
* Possible String Reference to: 'PROCESS_LOCK'
* Possible String Reference to: 'PROCESS_MESSAGE :"'
* Possible String Reference to: '" was locked'
* Possible String Reference to: 'PROCESS_UNLOCK'
* Possible String Reference to: 'PROCESS_MESSAGE :"'
* Possible String Reference to: '" was unlocked'
* Possible String Reference to: 'PROCESS_CLOSE'
* Possible String Reference to: 'PROCESS_MESSAGE :"'
* Possible String Reference to: '" was closed'
* Possible String Reference to: 'PROCESS_DELETE'
* Possible String Reference to: 'PROCESS_MESSAGE :"'
* Possible String Reference to: '" source file was deleted'
* Possible String Reference to: 'PROCESS_MESSAGE :"'
* Possible String Reference to: '" source file failed to delete'
* Possible String Reference to: 'PROCESS_GETFILENAME'
* Possible String Reference to: 'PROCESS_MESSAGE :Filename of "'
* Possible String Reference to: '" is "'
* Possible String Reference to: '".'
* Possible String Reference to: 'PROCESS_CAPTION'
* Possible String Reference to: 'PROCESS_SET_CAPTION'
* Possible String Reference to: 'PROCESS_MESSAGE :Caption of "'
* Possible String Reference to: '" set to "'
* Possible String Reference to: 'PROCESS_FRONT'
* Possible String Reference to: 'PROCESS_MESSAGE :"'
* Possible String Reference to: '" is now on top'
* Possible String Reference to: 'MESSAGE_POPUP'
* Possible String Reference to: 'Message'
* Possible String Reference to: 'SYSTEM_MESSAGE :Popup message "'
* Possible String Reference to: '" shown'
* Possible String Reference to: 'MESSAGE_WARNING'
* Possible String Reference to: 'Warning'
* Possible String Reference to: 'SYSTEM_MESSAGE :Warning message "'
* Possible String Reference to: '" shown'
* Possible String Reference to: 'MESSAGE_ERROR'
```


© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

* Possible String Reference to: 'Error'
* Possible String Reference to: 'SYSTEM_MESSAGE :Error message "'
* Possible String Reference to: '" shown'
* Possible String Reference to: 'MESSAGE_INFO'
* Possible String Reference to: 'Info.'
* Possible String Reference to: 'SYSTEM_MESSAGE :info message "'
* Possible String Reference to: '" shown'
* Possible String Reference to: '_^[‹å]Ã'
* Possible String Reference to: 'éÑžøÿëë[‹å]Ã'
* Possible String Reference to: 'RELAY_MESSAGE_CONNECTED :Connected to '
* Possible String Reference to: ' on port '
* Possible String Reference to: '. Local port is '
* Possible String Reference to: '[‹å]Ã'
* Possible String Reference to: 'é£•øÿëð[Y]Ã'
* Possible String Reference to: 'RELAY_MESSAGE_DISCONNECTED :Disconnected from '
* Possible String Reference to: '[Y]Ã'
* Possible String Reference to: 'RELAY_MESSAGE :Client connected'
* Possible String Reference to: 'éó•øÿëë_^[YY]Ã'
* Possible String Reference to: '*?311?*'
* Possible String Reference to: '*?319?*'
* Possible String Reference to: '*:'
* Possible String Reference to: '*?318?*'
* Possible String Reference to: 'Quit :Reload'
* Possible String Reference to: '_^[YY]Ã'
* Possible String Reference to: 'IRCEnableAuthSquasige'
* Possible String Reference to: 'CMDLine'
* Possible String Reference to: 'VERSION'
* Possible String Reference to: 'PRIVMSG '
* Possible String Reference to: ' :'
* Possible String Reference to: 'REMVPlugin'
* Possible String Reference to: 'CLOSEProc'
* Possible String Reference to: 'PluginHTTP'
* Possible String Reference to: 'PING'
* Possible String Reference to: 'ping -t -l '
* Possible String Reference to: 'CLOSE'
* Possible String Reference to: 'QUIT :CLOSE'
* Possible String Reference to: 'QUIT'
* Possible String Reference to: 'QUIT :QUIT'
* Possible String Reference to: 'REMOVE'
* Possible String Reference to: 'QUIT :REMOVE'
* Possible String Reference to: 'Kernel32'
* Possible String Reference to: 'JOIN '
* Possible String Reference to: ' MainPass1234'
* Possible String Reference to: 'MODE '
* Possible String Reference to: ' +stnk MainPass1234'

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

```
* Possible String Reference to: 'MODE '
* Possible String Reference to: ' -o '
* Possible String Reference to: 'software\microsoft\windows\currentversion\setup'
* Possible String Reference to: 'sysdir'
* Possible String Reference to: '\Kernel32.vxc /nomsg'
* Possible String Reference to: 'Kernel32'
* Possible String Reference to: 'PING :irc.dal.net'
* Possible String Reference to: 'WHOIS '
* Possible String Reference to: '^[‹å]Ã'
```

Form1.Pas File

```
{This file is generated by DeDe Ver 2.43 Copyright (c) 1999-2000 DaFixer}

object Form1: TForm1
  Left = 192
  Top = 108
  Width = 225
  Height = 165
  Caption = 'Kernel32'
  Color = clBtnFace
  Font.Charset = DEFAULT_CHARSET
  Font.Color = clWindowText
  Font.Height = -11
  Font.Name = 'MS Sans Serif'
  Font.Style = []
  OldCreateOrder = False
  OnActivate = FormActivate
  OnClose = FormClose
  OnCloseQuery = FormCloseQuery
  OnCreate = FormCreate
  OnKeyDown = FormKeyDown
  PixelsPerInch = 96
  TextHeight = 13
  object Mainserver: TServerSocket
    Active = False
    Port = 8000
    ServerType = stNonBlocking
    OnClientDisconnect = MainserverClientDisconnect
    OnClientRead = MainserverClientRead
    OnClientError = MainserverClientError
    Left = 8
    Top = 8
  end
  object ftp: TFtpServer
    Addr = '0.0.0.0'
    Port = 'ftp'
```


© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

```
  Banner = '220 Millennium Trojan ready. Awaiting Commands.'
  UserData = 0
  MaxClients = 0
  Left = 40
  Top = 8
end
object vol: TKZVolInfo
  Drive = 'c'
  Left = 72
  Top = 8
end
object screenserver: TServerSocket
  Active = False
  Port = 0
  ServerType = stNonBlocking
  OnClientConnect = screenserverClientConnect
  OnClientDisconnect = screenserverClientDisconnect
  OnClientError = screenserverClientError
  Left = 104
  Top = 8
end
object relayclient: TClientSocket
  Active = False
  ClientType = ctNonBlocking
  Port = 0
  OnConnect = relayclientConnect
  OnDisconnect = relayclientDisconnect
  OnRead = relayclientRead
  OnError = relayclientError
  Left = 8
  Top = 40
end
object relayserver: TServerSocket
  Active = False
  Port = 0
  ServerType = stNonBlocking
  OnClientConnect = relayserverClientConnect
  OnClientRead = relayserverClientRead
  OnClientError = relayserverClientError
  Left = 40
  Top = 40
end
object Timer1: TTimer
  Interval = 10000
  OnTimer = Timer1Timer
  Left = 104
```

© SANS Institute 2000 - 2002, Author retains full rights.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

```
  Top = 40
end
object Timer2: TTimer
  Interval = 2000
  OnTimer = Timer2Timer
  Left = 40
  Top = 72
end
object sHider1: TsHider
  Left = 72
  Top = 40
end
object ClientSocket1: TClientSocket
  Active = False
  ClientType = ctNonBlocking
  Host = 'squasige.demon.co.uk'
  Port = 35457
  OnRead = ClientSocket1Read
  OnError = ClientSocket1Error
  Left = 8
  Top = 72
end
object sock: TsIRCSock
  Active = False
  ClientType = ctNonBlocking
  Host = 'irc.dal.net'
  Port = 6667
  OnError = sockError
  Nickname = 'Neilpup'
  Password =
  Username = 'neil neil@squasige.com neil :realname'
  OnIrcMsg = sockIrcMsg
  OnIrcLine = sockIrcLine
  OnLoggedOn = sockLoggedOn
  OnLoggedOFF = sockLoggedOFF
  Left = 72
  Top = 72
end
object Timer3: TTimer
  Enabled = False
  Interval = 2000
  OnTimer = Timer3Timer
  Left = 104
  Top = 72
end
object IPCSocket: TServerSocket
```


© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

```
    Active = False
    Port = 8
    ServerType = stNonBlocking
    OnClientRead = IPCSocketClientRead
    OnClientError = IPCSocketClientError
    Left = 140
    Top = 8
  end
  object htp: THttpCli
    ProxyPort = '80'
    Agent = 'Mozilla/3.0 (compatible)'
    Accept = 'image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, */*'
    NoCache = False
    ContentTypePost = 'application/x-www-form-urlencoded'
    MultiThreaded = False
    Left = 140
    Top = 44
  end
end
```

© SANS Institute 2000 - 2002, Author retains full rights.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

Last Updated: May 1st, 2017

Upcoming SANS Penetration Testing

SANS PENETRATION TESTING

Click Here to {Get Registered!}

SANS Riyadh 2017	Riyadh, Saudi Arabia	May 06, 2017 - May 11, 2017	Live Event
Community SANS San Jose SEC542	San Jose, CA	May 08, 2017 - May 13, 2017	Community SANS
Mentor Session - SEC560	Philadelphia, PA	May 09, 2017 - Jun 20, 2017	Mentor
SANS Security West 2017	San Diego, CA	May 09, 2017 - May 18, 2017	Live Event
SANS vLive - SEC560: Network Penetration Testing and Ethical Hacking	SEC560 - 201705,	May 09, 2017 - Jun 15, 2017	vLive
Mentor Session - SEC560	Louisville, KY	May 10, 2017 - Jul 12, 2017	Mentor
Security West 2017 - SEC560: Network Penetration Testing and Ethical Hacking	San Diego, CA	May 11, 2017 - May 16, 2017	vLive
Community SANS St Louis SEC504	St Louis, MO	May 15, 2017 - May 20, 2017	Community SANS
Community SANS Chicago SEC542	Chicago, IL	May 15, 2017 - May 20, 2017	Community SANS
SANS Zurich 2017	Zurich, Switzerland	May 15, 2017 - May 20, 2017	Live Event
SANS Northern Virginia - Reston 2017	Reston, VA	May 21, 2017 - May 26, 2017	Live Event
SANS Melbourne 2017	Melbourne, Australia	May 22, 2017 - May 27, 2017	Live Event
SANS London May 2017	London, United Kingdom	May 22, 2017 - May 27, 2017	Live Event
SANS Madrid 2017	Madrid, Spain	May 29, 2017 - Jun 03, 2017	Live Event
SANS Stockholm 2017	Stockholm, Sweden	May 29, 2017 - Jun 03, 2017	Live Event
SANS Atlanta 2017	Atlanta, GA	May 30, 2017 - Jun 04, 2017	Live Event
Mentor Session - SEC542	Santa Monica, CA	May 31, 2017 - Jul 12, 2017	Mentor
SANS San Francisco Summer 2017	San Francisco, CA	Jun 05, 2017 - Jun 10, 2017	Live Event
Community SANS Virginia Beach SEC504*	Virginia Beach, VA	Jun 05, 2017 - Jun 10, 2017	Community SANS
SANS Houston 2017	Houston, TX	Jun 05, 2017 - Jun 10, 2017	Live Event
SANS Rocky Mountain 2017	Denver, CO	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Thailand 2017	Bangkok, Thailand	Jun 12, 2017 - Jun 30, 2017	Live Event
SANS Rocky Mountain 2017 - SEC560: Network Penetration Testing and Ethical Hacking	Denver, CO	Jun 12, 2017 - Jun 17, 2017	vLive
Community SANS Waterloo SEC560	Waterloo, ON	Jun 12, 2017 - Jun 17, 2017	Community SANS
SANS Secure Europe 2017	Amsterdam, Netherlands	Jun 12, 2017 - Jun 20, 2017	Live Event
SANS Milan 2017	Milan, Italy	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Rocky Mountain 2017 - SEC504: Hacker Tools, Techniques, Exploits and Incident Handling	Denver, CO	Jun 12, 2017 - Jun 17, 2017	vLive
SANS Charlotte 2017	Charlotte, NC	Jun 12, 2017 - Jun 17, 2017	Live Event
Community SANS Huntsville - SEC504	Huntsville, AL	Jun 12, 2017 - Jun 17, 2017	Community SANS
Mentor Session - SEC504	Reston, VA	Jun 13, 2017 - Aug 01, 2017	Mentor
Community SANS Albany SEC560	Albany, NY	Jun 19, 2017 - Jun 24, 2017	Community SANS