GIAC
CERTIFICATIONS

Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

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

Interested in learning more?

Check out the list of upcoming events offering
"Security Essentials (Security 401)"
at http://www.giac.org/registration/gsec

KLEZ.H

From Propagation to Prevention

Michael Bakes
June 13, 2003

GIAC Security Essentials Certification (GSEC)
Assignment: v1.4b
Option: Research on Topics in Information Security

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Summary

This study reviews the properties of the Klez.H worm, key findings from a set of infection experiments, and some of the network security tools needed to detect Klez.H infection. Both reported results and new unreported findings from this study show that Klez.H exploits several known SANS/FBI Top 20 List of vulnerabilities to propagate and infect local and remote computers on a Local Area Network. These include a sleep / wake routine for scanning the network for new files and directories to infect, creation and deletion of stealth processes for file infection, creation of root level shares with Full Control Permissions for Everyone, and the creation of a back door internet-bot on port 1027. The experimental results of this study highlight that virus protection involves not only the downloading and updating of a new virus signature, but also the deployment of secondary security measures beyond antivirus patterns and scanning routines. These secondary security measures include user training / awareness, patching of known software vulnerabilities, and disabling of exploitable controls at the application and operating system level (such as those identified by the SANS/FBI Top 20 List of vulnerabilities). Finally, this study suggests the ongoing need through non-repudiation, authenticity and encryption tools to provide comfort to email recipients that their email is virus-free.

Introduction

In today's computer integrated society, worms, viruses and other forms of malicious software are a daily occurrence. Sophos, an antivirus company specializing in the detection of new variants, reports that new detection routines for email viruses occur at a rate of 25 routines per day.[1] Furthermore, MessageLabs reports that one in every 215 emails contains a virus, up from 380 in 2001 and 790 in 2000[2]. The bottom line is that companies without adequate protection of their information systems infrastructure will eventually suffer a compromise of one or more of the three foundations of computer security, *Confidentiality*, *Integrity* or *Availability*. *Confidentiality* ensures secrecy during data processing and prevents intentional or unintentional unauthorized disclosure of sensitive information.[3] *Integrity* ensures that information is reliable and accurate and prevents unauthorized modification of company data.[3] *Availability* ensures that networks and systems are reliable and provide appropriate capacity to prevent

disruption of service and productivity.[3] In this review the most prevalent email worm in 2002, Klez.H, is the model virus used to not only show the general ways and not so general ways in which viruses attack IT systems, but also the importance of properly defending a company from such attacks and protecting the three cornerstones of information security.

Many people use the word "virus" in association with computers everyday, most without thinking of just how analogous the computer virus is to a virus that occurs in nature. Just as the biological virus invades a host and integrates into the DNA structure of the host to propagate itself, so to does the computer virus infect a host file to replicate itself and spread by a variety of means to a new host file. While the virus of nature is a "living" entity composed of DNA enveloped by a protein coat, the *in silico* virus is a piece of software that modifies other host files to incorporate a copy of itself. This propagation routine usually requires user interaction to initiate the process, which once launched, begins replication. As a final analogy, not all biological viruses kill the host organism. In fact, many viruses use the host to make copies of itself and as a vehicle for propagation of the species. Not all *in silico* viruses erase your computer's hard-drive, however every newly reported *in silico* virus today usually performs one or more of the functions that compromise the three foundations of computer security: *Confidentiality*, *Integrity* or *Availability*. As the *in silico* virus has evolved, there have been new categories of viruses that are now more broadly grouped as malware (malicious software). These include worms, Trojan horses (hereafter referred to as Trojans), octopii, germs, droppers and generators, to name a few.[4] This review will focus on one of these categories of malware, the worm.

The worm, also referred to as a virus, virus variant, and hybrid worm/virus, was first introduced to the information systems security community back in the early 1980's via The Xerox Worm, a program that was designed to utilize distributed computing.[5] This event contributed to the current definition of the worm as a self propagating program that creates and executes copies of itself on a computer or file server, typically exploiting software vulnerabilities, and that is able to spread via a wide diversity of computer-computer connections (i.e. network connections and file sharing) and applications (email, messenger software etc.). Unlike their virus counterparts, worms generally lack the ability to infect other host files, but they are capable of carrying malicious code such as a virus or Trojan. Some examples of worms that have previously posed a threat to our foundations of security are W32.Sircam (2001, *Confidentiality*), Code Red (2001, *Integrity*) and The Morris/Internet Worm (1988, *Availability*).[6,7] In 2002 and early 2003, the Klez family and specifically Klez.H, has demonstrated that it was and still is a significant threat to Windows computing environments around the world. The Klez family utilizes multiple propagation routines to spread via networks and email, and also carries an infectious viral payload that infects system files. Both propagation and infection have a direct impact on data *Availability*.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

The Klez Family

The name "Klez" is derived from text within a message that the worm carries ("Win323 Klez V2.01 …"), although the message is not displayed at any time during infection.[8] The Klez family of worms emerged late in 2001 and began to spread using email as the primary mechanism for infection.[9] Like many of the worms of the past; such as BubbleBoy (ActiveX vulnerability) and The Morris/Internet Worm (Sendmail vulnerability), the Klez family of worms propagate by exploiting a known vulnerability in Internet Explorer-based email clients (e.g. Microsoft Outlook and Outlook Express) that results in automatic execution of attachments. The key differentiating routines of the Klez family are their ability to utilize both network applications (email) and connections (shares) for replication and secondly, the routines leveraged to compile an email (TO:, FROM: and Subject:). Since its emergence, Klez has since diverged into 10 variants denoted sequentially with the letters A through J (Appendix 1). Review of the literature shows that all of the variants act similarly in their propagation (email and network shares) and infection (ElKern family of viral payloads). Depending on the anti-virus organization providing the detection signature for the worm variant, most of the corresponding aliases are listed in Appendix 1, along with the estimated size of the worm code and its date of discovery. Appendix 1 also lists the various viral payloads carried by each of the Klez variants, which up until today, has been a family of four viruses called ElKern. Some literature reporting Klez.B suggests that it does not carry the viral payload, but instead, carries a non-destructive Trojan allowing remote access to the infected computer.[10,11,12,13]

As already established, the Klez family of worms is able to propagate using an Outlook / Outlook Express vulnerability. Klez.H also contains a virus code in the form of ElKern, which first appeared approximately on the 25th or 26th of October, 2001.[14] This ElKern virus solely targets the Microsoft suite of operating systems (Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP and Windows Me) and thus, the Macintosh, UNIX and Linux operating systems all remain unaffected by this worm/virus. Prior to the identification of the Klez worms (Klez.A, 10/25/2001), there were no reported cases of the ElKern virus (ElKern A, 10/25/2001; ElKern B 01/17/2002; ElKern C 04/08/2002 and ElKern D 04/17/2002). Indeed it would accurate to say that Klez is a hybrid worm-virus, although for the purposes of this investigation, we'll refer to the worm and the virus as two separate entities as there does not appear to be dependence by one on the other.

The ElKern family of viruses are polymorphic viruses that are capable of unpredictably changing their code in ways to evade signature-based antivirus scanning technologies. The concept of a polymorphic virus is best explained by thinking of the virus as being composed of two units, an encrypted main body and its accompanying randomly-generated decryption key.[15] Each time that a new copy of the virus is created, a polymorphic engine (e.g. the Trident Polymorphic engine[16]) performs two tasks on the new copy. The first task is an encryption routine on the main body that is unique for every copy of the virus and the second task is the generation of a decryption key, the decryptor. Both are packaged together and allow the virus to have a constantly changing string, thus making detection more difficult.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

The ElKern variants are also file system-infecting viruses, meaning that the virus will write itself into the host file on shared folders and \Windows\System or \System32 folders, usually targeting executable *.com* or *.exe* files, but they can also target *.dll*, *.sys* or *.obj* files.[17,18] File infectors on the Windows platform are usually recognized by symptoms such as, but not limited to, file size changes, missing free memory, new processes, and decreased system / program performance. Similar to other file infecting viruses that are capable of being automatically triggered to launch by date, some of the variants of ElKern are also programmed in this manner (Appendix 1).[10,13,17]

Klez is an interesting family of worms in that they utilize the malicious routines of both worms and viruses. The Klez family of worms has a direct impact of data *Availability* through its network and email replication routines. Additionally, the Klez viral payload impacts *Availability* by deploying the ElKern family of viruses that in turn, infect system files. This study will examine the Klez family in more detail by specifically discussing the Klez.H variant, the most threatening "virus" in 2002 according to major antivirus vendors around the world.

The Klez.H Variant

KLEZ.H is a variant of KLEZ.A that was first detected on 15th April, 2002. It is an 85-90 Kbytes worm that damages computer file systems and has been transmitted to over 206 countries and detected in over 5 million emails by one ISP alone. [19,20,21] Klez.H became notorious when in May 2002 writers at Wired News named Klez.H "... the most pervasive e-mail virus in cyber history …".[22] Sophos recorded Klez.H as the number one "virus" for seven straight months, making it the most prolific virus of 2002, and MessageLabs recorded 4.9 million copies of the virus being stopped during 2002, five times greater than the second most prevalent virus Yaha.E.[1,2] Trend Micro's World Virus Tracking Center recorded the peak of Klez.H between July and September, during which time MessageLabs identified one in every 169 emails as containing the worm.[2,23] In 2003, Klez.H has continued to be one of the more threatening worms distributed via email.

Delivery and Execution

Klez.H arrives in an email message as an executable file attachment that, when viewed or previewed, is automatically launched by exploiting the "Incorrect MIME header vulnerability". The vulnerability allows "malicious" HTML emails to be constructed with a modified MIME header so that Internet Explorer-based email clients Outlook and Outlook Express will automatically execute an attachment when receiving the email message.[24] The vulnerability affects Microsoft Internet Explorer versions 5.01 and 5.5 and patches are available from Microsoft. Manual execution of the attachment will results in the launch of the same routines.

The following events take place during execution of Klez.H, but are not necessarily in the order in which they take place:

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

- Windows Startup: The executed Klez.H copies itself to the windows system folder under the name Wink*.exe and then adds or creates either of the following keys to registry for automatic execution upon system start-up:[25]
  HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run
  HKEY_LOCAL_MACHINE\System\CurentControlSet\Services\Wink*
- Immobilization of Antivirus Engines: Klez.H contains routines to disable and delete anti-virus programs. Upon worm execution, Klez.H locates the Type II files listed in Appendix 1. It immediate disables active processes, deletes the corresponding executable files and searches the following registry location to identify, and disable, any listings of the files shown in Table 2:[25]
  HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run
- Stealth Mode: Although not well documented, TrendMicro describes Klez.H as being able to hide itself by creating a system service process (hidden from taskbar under Windows 95 and 98).[19] On Windows 2000, this process is registered with a service control dispatcher allowing Klez.H to have a communication channel between the service and the start service control dispatcher.[26]

Email and Network Propagation of Klez.H

The primary and very efficient mechanism of propagation of the Klez.H is via the use of the Simple Mail Transfer Protocol (SMTP, port 25), which is used for about one half of all Transmission Control Protocol (TCP) connections.[27] Windows registry provides the source of the SMTP servers, which are listed under the following key:[19]
HKEY_LOCAL_MACHINE\Software\Microsoft\Internet Account Manager\Accounts\SMTP Server

As shown in Appendix 1, Klez.H first targets the infected computer's Windows Address Book (WAB) to find email addresses in order to propagate. The WAB file is specific to each user that has logged on to the host computer and can be identified in the windows registry key:
HKEY_CURRENT_USER\Software\Microsoft\WAB\WAB file name

Locally on the hard drive of the infected computer, this WAB file is typically located at:
C:\Documents and Settings\username\Application Data\Microsoft\Address Book\username.wab

More advanced than some of its counterpart variants, Klez.H also uses a variety of other file types to obtain its list of new victims (Appendix 1). Once the email addresses have been compiled, Klez.H creates a spoofed "From:" field using email addresses that it obtains from the infected users WAB file, and immediately initiates a mass-emailing routine. Because Klez.H uses a spoofed "From:" email address, Klez.H can essentially evade detection because recipients of the infected email will usually innocently respond to the spoofed email address, believing that this was the source of the worm / virus, when in fact it was not the real sender.[8] Wired News summarize this " …Many computer users say that friends, co-workers and business associates are angrily -- or patronizingly -- accusing them of sending out viruses. Some victims say they fear their professional reputations have been harmed…"[28]

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Klez.H generates its own list of subject lines for the infected emails rather than using a source file on the infected computer system. The list of subject line strings Klez H uses is taken from the Trend Micro web site:[19] "how are you", "let's be friends", "darling", "so cool a flash, enjoy it", "Your password", "honey", "some questions", "please try again", "welcome to my hometown", "the Garden of Eden", "introduction on ADSL", "meeting notice", "questionnaire", "congratulations", "sos!", "japanese girl VS playboy", "look, my beautiful girl friend", "eager to see you", "spice girls' vocal concert", "japanese lass' sexy pictures", "Worm Klez.E immunity", Undelivarable mail-"%s", Returned mail-"%s", Special, a %s %s game, a %s %s tool, a %s %s Web site, a %s %s patch, %s removal tools (%s = new, funny, nice, humour, excite, powful, good, special, WinXP, IE 6.0, W32.Elkern, W32.Klez.E, Symantec, Mcafee, F-Secure, Sophos, Trendmicro, Kaspersky).

A secondary method of propagation is via a routine that utilizes network connections. Launch of the worm executable file results in a replication routine creating copies of itself in directory trees on local and enumerated remote shared drives with read/write access.[29,30] For files in shared folders, Klez.H will create a randomly generated file name and add the extensions *.exe*, *.pif*, *.com*, *.bat*, *.scr* and *.rar*; with all attribute flags (Read-only, Hidden, System & Archive) enabled.[19] On some occasions, the files are given a double extension that is any combination of *.exe*, *.scr*, *.pif*, *.bat*, *.txt*, *.htm*, *.html*, *.wab*, *.doc*, *.rtf*, *.xls*, *.jpg*, *.cpp*, *.c*, *.pas*, *.mpg*, *.mpeg*, *.bak*, *.mp3* and *.pdf*.[19] This double extension vulnerability was previously exploited by SirCam.[31]

Infection with ElKern

As already established, Klez.H contains malicious viral code referred to as ElKern.D. A component of the launch of Klez.H is initiation of its infection routine which in turn, generates a copy of ElKern with a random filename. The file is placed in C:\Program Files directory and launched, whereby it immediately begins searching for files to insert its code. It first searches the current directory, and then randomly generated drive letters A through to Z, to infect programs and processes in drives and sub-directories.[29,32] It can also infect files in enumerated network shares identified in the Windows Network Neighborhood.[32]

Like most file infecting viruses, ElKern.D first targets executable files. It infects Portable Executable files (PE; because it can be executed on multiple platforms[33]) by inserting code into the original file without changing the file size. It does this by using garbage data as padding, and if no free space is available, it attaches itself to the end of the file.[32] This infection pattern (padding without file size change) is characteristic of a cavity virus. ElKern does not infect programs with a .DLL extension; however, it does search the addresses of the following API functions using numerical values: KERNEL32.DLL, SFC.DLL, MPR.DLL and USER32.DLL.[32] The search results are used to assemble code in memory and create new threads of execution. The original thread returns control to the infected host, resulting in an increased number of threads for the running process.[32]

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

In addition to the .DLL exclusions, the D variant of ElKern does not infect files protected by the System File Checker (SFC) or WinZip and WinRAR compression/extraction utilities. A more comprehensive list of files exceptions can be found in the literature.[32]

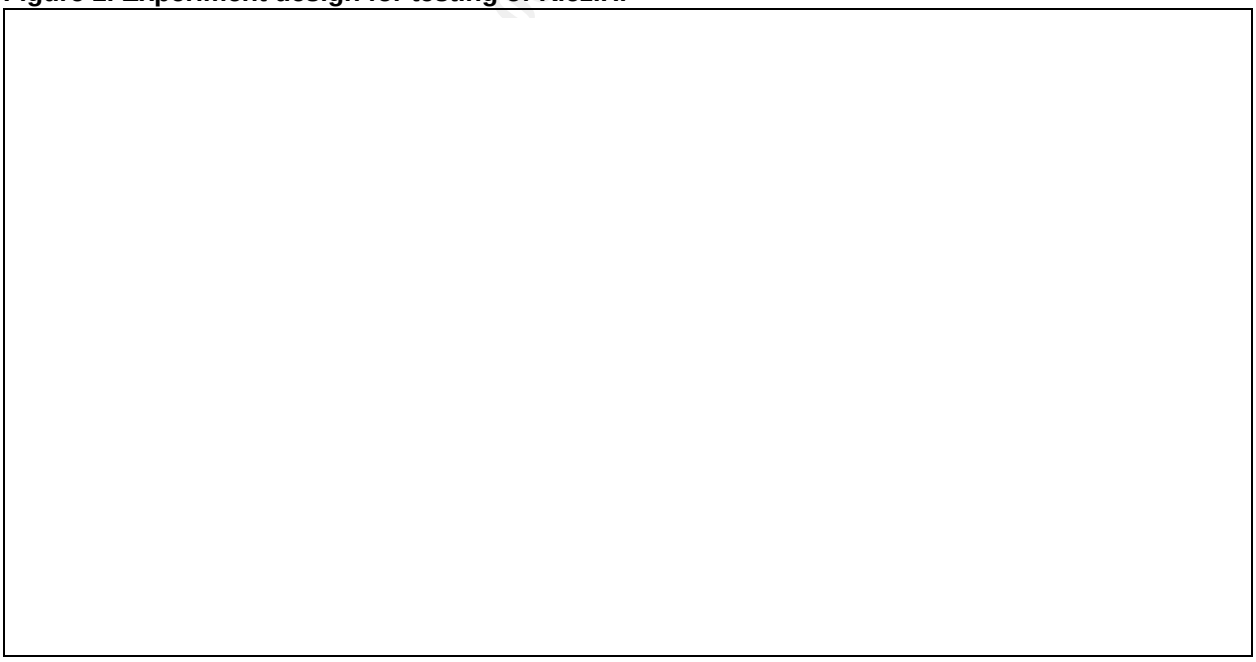
Unlike the other ElKern variants that either copy or create an autostart entry into Windows registry for either Wqk.dll or Wqk.exe, PE_ElKern.D does not deploy these files or modify registry.[29]

Experimental Investigation of Klez.H

An experimental investigation of the reported capabilities of Klez.H was undertaken to study and understand this notorious email worm. These experiments were carried out to both demonstrate a number of the tools studied in the SANS GIAC certification program and the importance of *Defense in Depth* virus protection. The experiments are:

1. List of Services and All Shares using DumpSec
2. Port scanning using Nmap
3. Execution of a virus
4. Review of Performance events using Trace Log
5. Services running using FPORT
6. List of Shares using DumpSec (after Klez.H)
7. Registry changes using Regdmp
8. Configuration and capture of TCP communications using Snort
9. Virus removal attempts

Figure 1. Experiment design for testing of Klez.H.



© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Figure 1 shows a schematic of the experiment design. The experiment consisted of two Dell Inspiron notebooks connected together via a LinkSys DSL (Digital Subscriber Line) router, configured in Gateway mode. Table 1 shows the configuration of the LinkSys router; most of the settings were left as their default for the purposes of showing the full potential of the Klez.H worm and its payload. Furthermore, because the router was not connected to a Wide Area Network (WAN), the WAN settings were left as default (Table 1). Class C Internet Protocol (IP) addresses (192.168.1.0/24) and subnets (255.255.255.0) were defined for the experiment. To demonstrate the importance of password protection, the password was changed from the default to "Kl3zHwtmpwi2oo2" (Kl3zH was the most prominent worm in 2oo2). It is common knowledge that on this router, the default user name field is blank (i.e. no user name) and the password is "admin", and like many other products, this logon information is available online (http://www.linksys.com). Default passwords on network hardware, whether used in a residential or commercial environment, are too commonly left unchanged. As part of a good password policy, passwords on newly installed hardware should be changed from the default immediately, and thereafter, changed regularly.

Before starting the experiments listed above, all of the software to be studied was installed, including the WinPcap 3.0 alpha 4 client, which was downloaded from http://winpcap.polito.it/install/default.htm. In addition, and again an important component of forensics analysis and log file correlation, the system clocks were synchronized. Although there was no SMTP server available to test the propagation of Klez.H using email, Outlook Express was configured as a dummy Post Office Protocol (POP) client with settings as described in Table 2. It is possible that Klez.H might attempt to send email to another SMTP server if no local server is available, and the TCP traffic can be captured using Snort. Finally, Norton Ghost was used to take an image of both computers to allow the experiment to be repeated if necessary, to validate observations.

Table 1. Summary of LinkSys DSL router configuration used for Klez.H experiments

Feature	LinkSys Setting
Host Name	-
Domain Name	-
Media Access Control (MAC) Address	00-04-5A-EC-CA-75
Local Area Network IP Address	192.168.1.1
Subnet mask	255.255.255.0
WAN (DHCP mode)	No configured
Point-to-Point Protocol over Ethernet (PPPoE)	Disabled
Keep Alive	Enabled
Advance settings	Default

To ensure the integrity of the downloaded software binaries, the Message Digest 5 (MD5) checksum was used to validate the signature of downloaded files (when available). It is a fact that some of these tools are quite often the target of malicious software, and without the appropriate security checksums, an innocent user might not realize before its too late that their system has been compromised.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Table 2 shows the detailed configuration of Notebook-A and Notebook-B. Notebook-A did contain some 3rd party software used for other application testing and thus has shares and software that are not pertinent to this experiment. Notebook-B on the other hand was created from a formatted hard drive and a fresh operating system (OS) install.

Table 2. Summary of configuration of Systems used for Klez.H experiment

Feature	Notebook-A	Notebook-B
Operating System	Windows 2000	Windows 2000
Service Pack	3	3
Shares (Appendix 2)	Admin$	Admin$
	\Dumpsec (C:\Documents and Settings\Administrator\Desktop\Klez_tools\dumpsec)	\Snort (C:\snort)
	\TTC1 (C:\Temp)	
	\Accelrys (C:\Programs Files\Accelrys)	
MAC Address	00-04-76-42-FD-55	00-20-E0-67-35-AE
IP Address	192.168.1.100	192.168.1.112
Gateway	192.168.1.1	192.168.1.1
Nmap ports open (Appendix 3)	80/tcp, 135/tcp, 139/tcp, 443/tcp, 445/tcp, 1031/tcp, 1033/tcp	135/tcp, 139/tcp, 445/tcp, 1025/tcp
Email address	Notebook-A@sanspractical.com	Notebook-B@sanspractical.com
Mail server	Mail.sanspractical.com	Mail.sanspractical.com
SMTP server	SMTP.sanspractical.com	SMTP.sanspractical.com
Anti-virus (TrendMicro Officescan V.6.510)	Yes (disabled)	No

Experiment #1: List of Services and All Shares using DumpSec (before Klez.H infection)

The goal of this experiment was to view all services and shares on both of the test notebook computers using DumpSec software from Somarsoft.

The Somarsoft DumpSec software client (Version 2.8.2, May 8, 2002) was downloaded from http://www.systemtools.com/somarsoft and installed on both notebook computers. From within DumpSec, a number of reports are available by first connecting to a computer on the Local Area Network (LAN) and then selecting the appropriate Report. In this case, the following reports were compiled: “Permissions for Shares”, “All Shares” and “Services”. Several new shares with default permissions and rights were created on each test notebook while the default admin$ shares were unchanged from the OS installation.

The detailed results of the available Shares and Services on each test notebook are shown in Appendix 2. The list of Services running on each notebook was compiled as a reference point to allow correlation of data collected post-Klez.H infection. Table 2 shows a summary of the Shares that were identified on each notebook. On Notebook-A, the new shares created for the experiment were \DumpSec, \TTC1 and \Accelrys

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

while on Notebook-B, the share \Snort was created. The experiment clearly demonstrates the ease at which LAN networks can be probed for information on networked computers and provides enough information to compromise a vulnerable network or system. The spread of malicious software via unprotected shares is a common vulnerability that exploits NETBIOS. This will be discussed in more detail later (see Prevention and Eradication)

Experiment #2: Port Scan using Nmap

The goal of Experiment #2 was to use the Nmap tool to perform a simple port scan on a remote system, identify open ports and the operating system, and then gather log files for review.

The NmapWin (v1.3.0) software binaries were downloaded from http://sourceforge.net/projects/nmapwin). Nmap was executed with the syntax:

Nmap -v -O 192.168.1.X -oN Notebook-Y

where X=100 (Notebook-A) or 112 (Notebook-B) and Y=A or B, respectively
-v, option was added for verbose mode
-O, identify remote operating system using TCP/IP fingerprinting
-oN, stored output as a normal log file

The detailed scans from Nmap are presented in Appendix 3 while a summary of the Nmap scans is shown in Table 2. The Nmap results from Notebook-A showed the following ports open: 80/tcp (Hypertext Transfer Protocol; HTTP), 135/tcp (epmap; dce endpoint resolution, location service, ncs local location broker[34]), 139/tcp (Network Basic Input/Output System, netbios-ssn; netbios session service), 443/tcp (https; secure http (ssl), http protocol over tls/ssl), 445/tcp (Microsoft-ds), 1031/tcp (iad2; bbn iad) and 1033/tcp (outbound connection port[35]). Nmap was unsuccessful in guessing the remote operating system on Notebook-A.

On Notebook-B, which was built from a bare drive with a default OS installation, Nmap identified the following ports open: 135/tcp, 139/tcp, 445/tcp and 1025/tcp (network blackjack, listener rfs remote_file_sharing[34]). Nmap was successful in guessing the operating system on this notebook as Windows 2000/XP/ME.

This experiment demonstrates the effectiveness of using Nmap to probe a remote computer and identify open ports that could potentially be exploited. The feature for identification of the remote OS was successful for 1 out of 2 attempts.

Experiment #3: Execution of Klez.H

This experiment shows the capture and isolation of a Klez.H-infected file and then the subsequent chronology of events surrounding the launch of Klez.H.

A Klez.H-infected file was isolated from an email that arrived to the employer of the author. Every week Klez.H worm attachments are quarantined at their arrival to the company email server. Following identification of the Klez.H infected file, it was copied

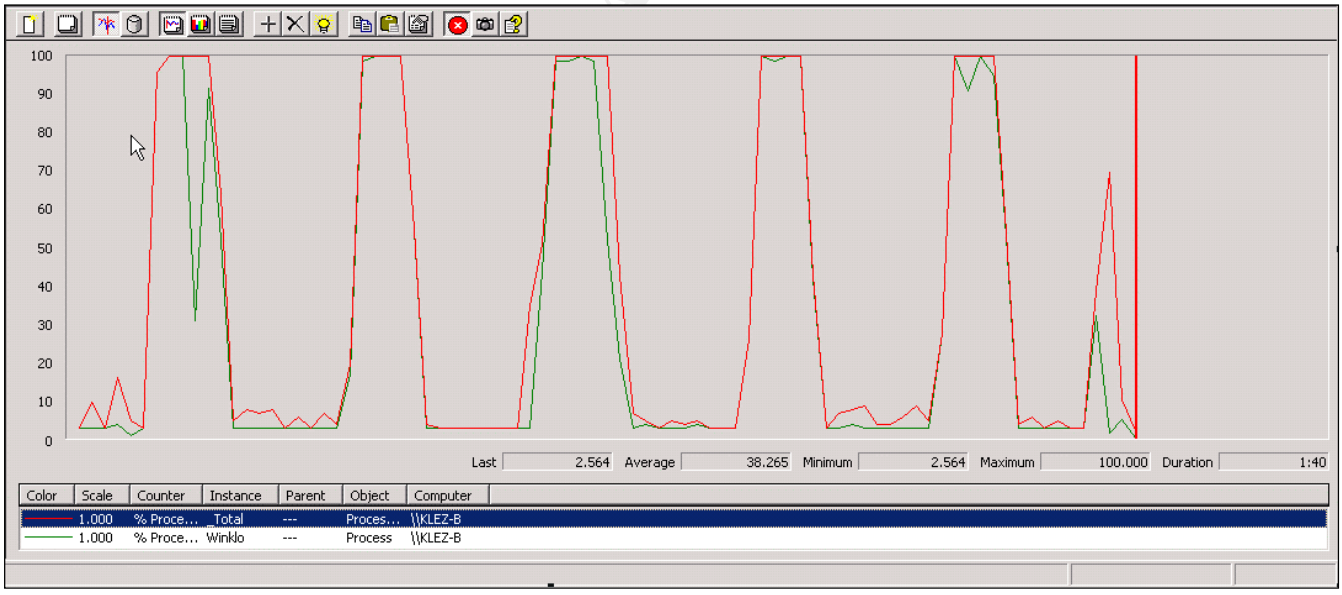
© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

from quarantine to a floppy disk, where as to be expected, the real time corporate virus scanner immediately quarantined the file on the floppy disk. The floppy disk was taken to a stand alone computer without virus protection, where the deleted file was recovered using Badboy Copy Pro 3.63 (http://www.jufsoft.com/badcopy/). The executable file, which was previously identified by TrendMicro ScanMail as being infected with Klez.H, was now recovered and ready for testing. The infected file, I-worm.Klez.H, was copied to C:\ on Notebook-B in preparation for launch.

4:25:00 Start trace logs on both Notebook-A and Notebook-B
4:25:10 Start Snort on both notebooks
4:25:20 Execute I-worm.Klez.H from command prompt on Notebook-B
4:27:00 Stop trace log on Notebook-A and Notebook-B
4:27:10 Stop Snort on both notebooks

Immediately following execution of Klez.H, Notebook-B showed a dramatic increase in CPU utilization for approximately 10-15 seconds. Although this was not captured, a representative graph showing the CPU utilization post-infection is shown in Figure 2. There is a clear cycling of total CPU utilization (red) that correlates directly with the CPU utilized by the Winklo.exe process (green). This sleep and wakeup routine of the CPU is speculated to result from the system performing regular network scanning for new files and shares to infect.

Figure 2. Total CPU Utilization (red) versus Winklo process CPU utilization (green)



Experiment #4: Capture and Review of Performance Events

Experiment #4 was designed to capture the Performance Trace Log during launch of Klez.H as a means of identifying processes and events at the OS and kernel levels.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

The Performance Trace Log feature is a default component of a Windows 2000 installation and is accessed within the Administrative Tools configuration settings. For this experiment, the Trace Log was configured as follows:

1. Events logged by system provider (default)
2. Set end filenames (mmddhh)
3. System was set to manual start and stop log
4. Transfer from buffer (enabled)
5. Buffer size = 4kb
6. Number of buffers: min=2 and max=25

A detailed report of the Trace Log data was compiled for each notebook computer using the "tracerpt –report" command and is shown in Appendix 4. The Trace Log from Notebook-A indicated that it was mostly idle with the exception of the csrss.exe process that consumed approximately 15% of the processor time. CSRSS.EXE is the client/server run-time subsystem and is used by Windows for console windows, creating and/or deleting threads and for some parts of the 16-bit virtual MS-DOS environment[36]. The observation of increased thread activity correlates with observations in the literature regarding the role of ElKern in creating threads on infected hosts.[32]

Review of the Trace Log for Notebook-B clearly shows the detail of a Klez.H-infected system. First, the introduction of the Winklo.exe process (size=88182 bytes), a new process that started shortly after the launch of the Klez.H worm. The process name (Wink*.exe) is consistent with reported observations, and clearly has an active role, since it is responsible for the launch of thirty six threads. The percentage of processor time required to execute these instructions is 34.12%, again indicative of a system performing several tasks that require significant CPU time. Disk utilization was identified as being elevated based on the 10% read count that was recorded by Trace Log. Compared to a stateful system like Notebook-A, winlogon.exe (user logon / logoff) had a high disk read count (787), with one explanation for this being the requirement for Klez.H to locate shares and permissions for network propagation. The System (system kernel threads) was recorded as having a high number of disk write counts (2631), which is believed to be related to the activity of ElKern as it inserts code into program files and system processes.

On Notebook-B, the process AnuF.exe launched seven threads and contributed to a significant portion (982) of total disk read counts. A search of the file system did not locate this file but the filename was suspiciously similar to the random file names generated by Klez.H and ElKern.D. With the process being no longer detectable, it is speculated that Anuf.exe is a malicious process that was running and the source file is suspected to be infected with ElKern.D. The "tracerpt –o" command was used to generate a more detailed trace log report, however this did not provide any obvious information regarding the creation and subsequent deletion of the process and file. The DumpSec Services report (Experiment #1) showed all of the existing process that are both stopped and running prior to infection, and clearly the Anuf.exe process did not exist.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

To validate the speculation of a stealth-like process being created and deleted, Notebook-B was cleaned of Klez.H using the "fixklez" tool available from Symantec (see Experiment #9) and the worm launch repeated. Real-time monitoring of the both the C:\Program Files directory and the current processes showed the creation of a new process, Un2.exe. This process started with memory usage of 866K, low CPU utilization, but a high number of disk I/O reads. The process continued to run for approximately 90 seconds, during which time the memory usage increased to 906K, but the disk I/O and CPU utilization remained constantly low (<3% CPU). Almost as quick as it started, the Un2.exe Image Name disappeared from the active processes and the file disappeared from the C:\Program Files directory. Searching of the C:\ failed to locate the file. These observations highlight two key points:

1. The characteristics of the process Un2.exe are consistent with the characteristics observed for the AnuF.exe file and process that were captured during the first Klez.H infection.
2. In both experiments, the file created in the C:\Program Files directory matches the literature description of the deployment of the ElKern virus.

One final comment on the trace log results collected from both Notebook-A and Notebook-B is the creation of two new processes, atiptaxx.exe and ati2evxx.exe, of which the latter contributes to a small number of disk read counts on the infected host. The processes are likely related to infection with Klez.H and fit the observation provided by Symantec that ElKern.D will search for all programs and processes and attempt to insert its code. When the experiment was repeated, two processes with identical names to the first experiment were created during infection, along with source files located in C:\WINNT\System32 directory. Thinking about the random file naming schema used Klez.H and ElKern.D, one explanation is that Klez.H creates a copy (or ElKern.D modifies) an existing file to create a file with a new name of the format ati***xx.exe, which is subsequently launched as a process.

Experiment #4 demonstrates the utility of the Trace Log function provided with the Windows operating system. The results highlight some of the routines utilized by Klez.H and ElKern.D during infection of a host computer and the stealthy nature of those routines. In this experiment, Trace Log was used to capture critical data prior to and during infection with Klez.H, the results of which both validated literature observations and furthermore, provided the grounds for elucidation of two previously unreported observations.

Experiment #5: Services running using FPORT

The goal of Experiment #5 was to use FPORT following infection with Klez.H to identify the services that were running along with a listing of the ports that were being used by each service.

The FPORT command is useful for correlation of open ports with a running host service, and following download from http://www.foundstone.com, it was installed on both notebooks to look for any new suspicious port activity. The FPORT results for

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Notebook-B are shown in Figure 3 and indicate that following infection with Klez.H, all of the ports identified prior to infection (using Nmap) are still active on the Notebook-B [Table 2; epmap (135), netbios (139), Microsoft-ds (445) and blackjack (1025)].

Figure 3. FPORT scan of Notebook-B

```
E:\>fport
FPort v1.33 - TCP/IP Process to Port Mapper
Copyright 2000 by Foundstone, Inc.
http://www.foundstone.com

Pid   Process          Port  Proto Path
412   svchost       -> 135   TCP   C:\WINNT\system32\svchost.exe
8     System        -> 139   TCP
8     System        -> 445   TCP
580   MSTask        -> 1025  TCP   C:\WINNT\system32\MSTask.exe
8     System        -> 1027  TCP
8     System        -> 1135  TCP

412   svchost       -> 135   UDP   C:\WINNT\system32\svchost.exe
8     System        -> 137   UDP
8     System        -> 138   UDP
8     System        -> 445   UDP
232   lsass         -> 500   UDP   C:\WINNT\system32\lsass.exe
220   services      -> 1026  UDP   C:\WINNT\system32\services.exe
```

In addition, FPORT identified two new open ports with active system services. The first is port 1027 which is used by ICQ messenger[37], an internet application commonly used by black hats to communicate. Port activity involving the use of chat servers is becoming a increasingly common component of malware and is referred to as "internet-bots", which basically allow connection of a compromised system with an IRC chat server.[38] An known example of a bot is the recently identified W32/Cult-A worm, which runs as a background process waiting for a remote system to issue commands.[39] The second newly identified is port 1135, of which the function is not known. Further elucidation of the role is these ports after Klez.H infection requires a more in-depth analysis. The experiment does highlight the need for a proficient understand of the use of multiple computer forensics tools in order to effectively build a picture of how and why a system was compromised.

Experiment #6: Identification of Shares using DumpSec (after Klez.H infection)

The goal of this experiment was to use DumpSec to evaluate file shares on both Notebook-A and Notebook-B, after launch of Klez.H worm. The configuration and use of the DumpSec Reporting features are described in Experiment #1.

The DumpSec Share results from Notebook-A identified all of the shares available prior to infection with Klez.H, and two new shares that were created following infection with Klez.H. The new shares on Notebook-A had the name format of *x*Notebook-B where *x*=drive letter C or D (Figure 4, text in bold) and were shared at the root level (C:\ and D:\). Furthermore, DumpSec showed that both of the shares were established with full control permissions set to "Everyone" (i.e. no discretionary access control list, dacl).

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Although not provided, a DumpSec report on shares on Notebook-B following infection showed that no additional shares were created.

Figure 4. Additional Shares on Notebook-A following Klez.H execution

3/11/2003 9:30 PM - Somarsoft DumpSec (formerly DumpAcl) - \\Notebook-A

Share and path	Account	Own	Permission
CNOTEBOOK-B=C:\ (disktree)		**unprotected (no dacl)**	
IPC$= (special admin share)		admin-only (no dacl)	
TTC1=C:\TEMP (disktree)	Everyone	all	
TTC1=C:\TEMP (disktree)	?unknown	all	
print$=C:\WINNT\System32\spool\drivers (disktree)		Everyone	read
print$=C:\WINNT\System32\spool\drivers (disktree)		Notebook-A\Administrators	all
print$=C:\WINNT\System32\spool\drivers (disktree)		Notebook-A\Power Users	all
D=D:\ (disktree)		unprotected (no dacl)	
Accelrys=C:\Program Files\Accelrys (disktree)		unprotected (no dacl)	
ADMIN$=C:\WINNT (special admin share)		admin-only (no dacl)	
C$=C:\ (special admin share)		admin-only (no dacl)	
DNOTEBOOK-B=D:\ (disktree)		**unprotected (no dacl)**	

Inspection of the existing shares on Notebook-A indicated that the \ACCELRYS share now contained two new files, btui.xls.rar (95 kB) and class.htm.rar (95 kB). A second share \DUMPSEC also contained two new files, name.txt.rar (90kB) and onclick.wab.exe (91kB). Finally, the third share \TTC1, contained new file members Jud.rtf.rar (95 kB) and tcxvw.exe (87 kB). These file names are consistent with naming nomenclature reported in the literature for copies of files created by Klez.H and deposited in remote share directories during the Klez.H infection routine. As a test of the payload contained within these files, the executable file "tcxvw" was launched, at which time it demonstrated its Klez.H payload by performing activities that were identical to those observed during the launch of Klez.H on Notebook-B.

The existing share on Notebook-B (\Snort) was the recipient of four new files, namely cphl.asp.exe (92 kB), pvklx.jpg.rar (89 kB), Zjufy.exe (89 kB) and dcvmw.bak.rar (86 kB). Again, these files are named in a manner consistent with the random-name nomenclature reported by Klez.H in the literature.[19,25]

Experiment #6 showed that two new root shares on the local drives of the remote notebook (Notebook-A) were created. The Klez.H host, Notebook-B, did not contain these shares. Upon review of the available literature regarding infection with Klez.H, the creation of root level shares appears to be a new and important observation. A share with root level access and full permissions for Everyone is a potential compromise of system *Integrity* and *Confidentiality*. The naming of the shares can be generalized as *x*hostname, where *x* = local drive letter on the remote computer (e.g. A, B, C etc.) and hostname = name of the computer hosting the Klez.H worm, in this case, Notebook-B. It is speculated the these drives were created via the exploitation of a null session vulnerability (see Prevention and Eradication).

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

This experiment also highlights the variation in the size of the files that Klez.H creates, which in this study, varied from 86kB to 95kB in size. The literature reports that the variable size of infected files is attributable to the polymorphic behavior of ElKern; during the infection routine when a file is inserted with viral code, the size of the new file is dictated by the level of encryption and compression performed, and consequently the size is not always constant.[14]

Experiment #7: Review of Registry Changes on Notebook-B using "regdmp"
This experiment reviewed the changes in the Windows registry by using the Regdmp utility provided in the Windows 2000 Resource Kit.

The Windows 2000 Resource Kit was installed on Notebook-B. Using the Windows registry Find function, all instances of Wink* were located. At each instance, the regdmp utility was used from the command prompt to export the registry key. The full details of the keys are shown in Appendix 5, but are summarized and discussed below:

HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Applets\Regedit
The key above shows the last key that was accessed using the regedit applet. The registry key shows the entry:

"LastKey=My\Computer\HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\Winklo"

HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Enum\Root\LEGACY_WINKLO
HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\Winklo

HKEY_LOCAL_MACHINE\SYSTEM\ControlSet002\Enum\Root\LEGACY_WINKLO
HKEY_LOCAL_MACHINE\SYSTEM\ControlSet002\Services\Winklo

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_WINKLO

The HKEY_LOCAL_MACHINE\SYSTEM describes the device drivers and services that can be used by Windows 2000 during loading. ControlSet001 and ControlSet002 are copies of the required boot information while the CurrentControlSet is a pointer to one of these clones. An interested user can use the registry key HKEY_LOCAL_MACHINE\SYSTEM\Select to determine which of these ControlSets is loaded at startup[40]. The two ControlSet entries above, ControlSet001 and ControlSet002, demonstrate that Klez.H has made the appropriate changes to registry to ensure that each time the infected system is booted, regardless whether booted in "Normal" or "Last Good Known Configuration" modes, the following events will take place:

\Enum\Root\LEGACY_WINKLO – this key contains settings and resources for legacy (non-Plug and Play) bus connections using the WINKLO control. It is likely that winklo uses these to communicate with non-plug and play devices.

\Services\Winklo contains the entry that causes C:\WINNT\System32\Winklo.exe to be launched as a service during start up.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

HKEY_USERS\S-1-5-21-1614895754-1993962763-1060284298-500\Software\Microsoft\Windows\CurrentVersion\Applets\Regedit

The key represents the currently logged on user that in this case, has Security Identified (SID; S-1-5-21-1614895754-1993962763-1060284298-500). Because the HKEY_CURRENT_USER\ and HKEY_USERS\SID are both mapped from NTUSER.DAT, we know that the entry for HKEY_USERS\SID and HKEY_CURRENT_USER\ is the same. That is, the last modification made in registry was: "LastKey=My\Computer\HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\Winklo"

Experiment #8 demonstrates the utility of the regdmp tool in the Windows 2000 Resource Kit. It has been shown here that following launch of the Klez.H worm, the executable Wink* is deposited into the C:\WINNT\System32 folder whereby entries are made to the registry to ensure that every boot of the system starts this service. This is in agreement with previously reported observations of Klez.H. Klez.H also makes an entry into the \enum\root key of registry and this is speculated to provide legacy device compatibility for non-plug and play communication devices, primarily network interface cards. Although not observed in this experiment, some literature reported that Klez.H created an entry in the key:[19]

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run

Experiment #8: Configuration of Snort Intrusion Detection System (IDS)

The goal of Experiment #8 was to configure the Snort intrusion detection client to log TCP communication between the two notebooks during infection of the host (Notebook-B) with Klez.H.

The Snort 1.9 IDS software was downloaded from http://www.snort.org/dl/binaries/win32/ and installed. Snort was configured with the following syntax:

"**snort –dev –l logdir**"

where "–dev" is the interface for acquiring data (default is 1; to collect from interface 2, add the prefix -i2)
-l, option to log the information to directory "logdir".

During infection of Notebook-B, there was a burst of TCP traffic between the two notebook computers. Because of the amount of log data collected, sections of the detailed communication activity is captured in Appendix 6 and discussed below. Studying of the data set shows that TCP communication was limited to two ports, 1070 on Notebook-B (infected host) and 139 on Notebook-A. There was no attempted SMTP activity on port 25 which is believed to result from failure to identify an available SMTP server in registry.[19] As a consequence, Klez.H did initiate or utilize any of its propagation routines using email.

Snort also recorded several Address Resolution Protocols (ARP) from Notebook-B. ARP provides a mechanism of mapping Media Access Control (MAC) and IP addresses by using an Ethernet broadcast request.[41] An example of this mapping is provided below where Snort successfully logged an ARP request immediately prior to the TCP handshake between Notebook-B and Notebook-A:

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

03/07-16:25:21.938091 ARP who-has 192.168.1.100 tell 192.168.1.112
03/07-16:25:21.938376 ARP reply 192.168.1.100 is-at 0:4:76:42:FD:55

There was no ICMP traffic (ping etc.) identified prior to the ARP request and thus is still remains unclear as to how the infected host, 192.168.1.112, was able to determine that 192.168.1.100 existed. However, the ARP mapping did allow the two computers to begin communication as reflected by the first packet sent between our virulent host 192.168.1.112 on port 1070 (outbound connection port) and the recipient, 192.168.1.100 on port 139 (netbios). The recipient, 192.168.1.100 immediately responds with a syn-ack as the second step of the three-way handshake. Finally, 192.168.1.112 responds with an ack to complete the handshake. The conversation continues with the push (psh) of packets and the ack of the sequence, and based upon the packet annotations, also includes communications with the Windows LAN Manager to identify local and remote shares. One of these transactions (t=16:25:21.788044) clearly results in the communication of all printer and file shares from 192.168.1.100 to 192.168.1.112, key pieces of information that will allow the propagation of Klez.H via network shares. This information is made available through exploitation of the netbios open share vulnerability (see Prevention and Eradication).

At t=16:25:21.721363, 192.168.1.112 sends a packet of data to 192.168.1.100 for the use of the IPC$ resource. IPC$ is an essential component for program communication and remote administration and in this case, might be indicative of an attempt by 192.168.1.112 to execute commands, such as defining share rights, by remote administration. This information is made available through exploitation of the null session vulnerability (see Prevention and Eradication).

At t=16:25:21.805406, 192.168.1.112 sends the request to 192.168.1.100 requesting the creation of the first share, CNOTEBOOK-B, a root level share on the local C drive of 192.168.1.100. At t=16:25:21.833450, an additional request is sent from 192.168.1.112 to 192.168.1.100 to create a second root level share, DNOTEBOOK-B, the local D drive of 192.168.1.100. A third request to establish a root share of the local E drive is incomplete when 192.168.1.100 fails to acknowledge that it has an E drive available. The creation of these shares is consistent with the observations made in Experiment #6, where DumpSec identified the creation of root level shares of all the local drives on Notebook-A.

During the remainder of the Snort TCP data capture, we see a cycling routine involving the following communications from 192.168.1.112 to 192.168.1.100:

1. Attempted access to root level share on C drive
2. Attempted access to root level share on D drive
3. Attempted access to root level share on E drive
4. Communication and activity involving a one of the three local shares on Notebook-A

The Snort data show that at t=16:25:21.883573, t=16:25:23.153093 and t=16:25:24.304417, 192.168.1.112 attempted to access the respective shares of \TTC1, \ACCELRYS and \DUMPSEC. All of these shares were identified during Experiment #6

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

as containing files with a random name that is consistent with an infected file created by Klez.H. Hence, it would be accurate to say that the TCP communication between 192.168.1.112 and 192.168.1.100 captured here is the deployment of those infected files to Notebook-A.

The TCP communication logged by Snort is conclusive in that it clearly identifies the infected host, 192.168.1.112, requesting the MAC of the remote computer, and is followed by a three-way handshake and establishment of communication between the two notebook computers. With communication established, 192.168.1.112 sends commands to probe the remote machine for available shares and it then uses the IPC$ resource to remotely administer 192.168.1.100. With this capability, 192.168.1.112 creates new root level shares with permissions set to "Everyone", compromising data *Confidentiality* and *Integrity*. Notebook 192.168.1.112 then begins a routine of scanning for remote local drives and deploying copies of itself into each of the shares; the routine then starts over.

Experiment #9: Virus Removal Attempts

Notebook-A

As indicated in Table 2, the Trend Micro Officescan AV scanner was previously installed on Notebook-A, but disabled for the experiment. With the real time monitor re-enabled, the suspicious tcxvw.exe file identified in Experiment #7 was launched. Upon launch, it immediately disabled the Officescan product, a characteristic routine of Klez.H. Thus our first attempt to remove Klez.H failed (as to be expected).

A copy of TrendMicro PCcillin 2003 was purchased and installed on Notebook-A. During installation, PCcillin performs a scan for Trojans and viruses, and in this case, detected the presence of KLEZ_WORM. Before continuing, PCcillin required the system to be rebooted to effectively remove the worm. When windows restarted, the PCcillin installation was launched again, thus requiring PCcillin to perform another virus scan. During this second scan, a number of PE_ELKERN.D-infected files were found in the C:\WINNT\System32 directory. According to PCcillin, the installation did complete. Review of the running processes revealed that the PCcillin 2003 process was not running, or failed to start. Inspection of C:\Program Files\TrendMicro\pccillin2003\ showed that all of the program files had been deleted. Our second attempt to remove Klez.H had also failed.

Upon review of the file system, inspection of C:\OfficeScan NT (the directory in which the Officescan product had been installed) showed that all of the Officescan files were still present. This observation correlates with the literature which reports that only certain antivirus software file names are corrupted or deleted during antivirus process termination. These files listed in Appendix 1. The names of the Officescan programs files are not in the Appendix 1 list, and thus, were not targeted by Klez.H. There were however, several kernel errors preventing these files from being launched.

Finally, the Klez.H removal tool was downloaded from Symantec at http://securityresponse.symantec.com/avcenter/FixKlez.com. The file was copied to

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

floppy disk and launched via the instructions provided. After scanning, the following report was generated:

- Total number of the scanned files = 52731
- Number of deleted files = 44
- Number of repaired files = 658
- Number of terminated viral processes = 5
- Number of deleted viral services = 0
- Number of fixed registry entries = 0

Removal of Klez.H from an infected system was successful using the free tool downloaded from Symantec.

Notebook-B

The Klez.H worm was removed from Notebook-B using the tool downloaded from Symantec. The results are shown below:

- Total number of the scanned files = 8366
- Number of deleted files = 64
- Number of repaired files = 138
- Number of terminated viral processes = 1
- Number of deleted viral services = 0
- Number of fixed registry entries = 0

Despite the fact that the downloaded tool indicated the number of registry entries fixed was 0, review of the registry following reboot showed that all indications of wink* (Experiment #7) had been removed.

Hoax Tools for Klez Removal

After Klez.H was first discovered and its inner workings detailed, black hats used new strategies for infecting computers. One such attempt is to trick infected or potentially infected users into downloading or executing an attachment that advocated a patch for Klez.H.[42] Additionally, one recent hands-on example of these tools was also identified. On Tuesday 25th of March, 2003, the Author detected a virus-infected inbound email using corporate email scanners. The characteristics of the alert are interesting:

Subject = Patch for Klez.H
Scanning Time = 03/25/2003 07:07:03
Engine/Pattern = 6.510-1002/498
Action on virus found:
The attachment FixKlez.com contains WORM_YAHA.P virus. ScanMail has Moved it.

The virus programmer(s) that created WORM_YAHA.P are taking advantage of the widespread dissemination and damage done by Klez.H as the basis for a preventative tool that might entice an uneducated user into executing an attachment as defense against Klez.H. Ironically, the name of the tool is similar to the downloadable file available at Symantec that removes Klez.H from an infected host. In this case, however, the tool carries the WORM_YAHA.P payload. This highlights the need to (1)

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Provide user awareness as to the role of corporate virus scanners, (2) Ensure that the tools are downloaded from trusted sources and, (3) Always use a checksum such as MD5 to validate the signature of the tool prior to its execution.

Prevention and Eradication

The immediate answer to prevent infection by viruses is to stay current with the available virus definitions and patterns. The patches and pattern releases from the three major antivirus providers that allow detection of Klez.H are listed below:

- Computer Associates, eTrust Antivirus 5.*x* (5.4 / 1987), eTrust InoculateIT 6.0 and eTrust Antivirus 6.0 (23.53.05), InoculateIT 4.x (35.05) and Vet 10.4*x* (10.4/1987)
- Symantec, Virus definitions April 17, 2002
- Trendmicro, Scan Engine 5.200, Pattern File: 265

If your computer is already infected by Klez.H, all of the antivirus vendors offer a free downloadable tool for automatic Klez.H removal. If an internet connection does not exist, a manual procedure for Klez.H removal is also provided. These free tools from the three major antivirus vendors are listed below :

- Computer Associates: http://www3.ca.com/Files/VirusInformationAndPrevention/ClnKlez.zip
- Symantec: http://securityresponse.symantec.com/avcenter/venc/data/w32.klez.removal.tool.html
- Trendmicro: http://www.trendmicro.com/download/tsc.asp

The experimental findings are the focal point of this study. These findings not only provide a keen understanding of the Klez.H worm, but also the blueprint for a good *Defense in Depth* strategy. The lessons learned start with proper perimeter antivirus protection and proper controls on the internal network. *Defense in Depth* adds another dimension by examining methods to prevent malicious software damage by using secondary security measures. These secondary security measures are initially aimed at preventing propagation; to stop the worm from exploiting a known vulnerability in Internet Explorer-based email clients Outlook and Outlook Express and spreading via SMTP. This vulnerability highlights the need to perform vigilant patching of vendor software vulnerabilities, as well as the need to apply added measures like disabling of automatic controls (*e.g.*, ActiveX and Active Scripting). The SANS/FBI Top 20 list these controls as Number 8 on their top ten most exploitable Windows vulnerabilities.[43] With regards to network propagation, Klez.H spreads and infects via network shares by exploiting several weaknesses. The first weakness allows unauthenticated access to key information and / or connection to network shares through exploitation of a null session, or anonymous logon. The SANS/FBI Top 20 identifies this null session vulnerability as numbers 5 on its top 10 Windows list.[44] The second weakness is the

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Network Basic Input/Output System (netbios) communications service used by client-server applications that leaves the Windows network shares open to virus and malicious software integration and further deployment as identified in Experiment 8 via the use of port 139. Again, this weakness is number 4 in the SANS/FBI Top 10 Windows list.[45] *Defense in Depth* provides protection by patching against these two security vulnerabilities, thus alleviating *Confidentiality, Integrity* and *Availability* concerns. In reviewing the SANS/FBI top Windows vulnerabilities, it is conceivable that viruses like Klez.H could also exploit the Remote Registry Access vulnerability as another vehicle to infect remote systems. Indeed, this would be a good reason to look to this list for system hardening.

Finally, since Klez.H has routinely been disseminated throughout the world, all email should be treated with caution. There is an ongoing need for non-repudiation, or the ability to know that the email you receive is truly from the identified sender. Digital signatures and encryption are a means of providing protection against non-repudiation, authenticity and confidentiality.

This study highlights that three of the SANS/FBI Top 10 Windows vulnerabilities are exploited by Klez.H to allow its propagation and infection. All three of these vulnerabilities can be patched to provide a secondary level of protection against malicious code, should malicious software penetrate your network perimeter or be launched internally on the network.

Conclusions

This study reviews the most prevalent email worm in 2002, Klez.H, as a model virus to demonstrate the importance of properly protecting a company from such virus attacks. Computer viruses have become a part of daily life, and just like the biological virus that occurs in nature to assault our health, so to does the computer virus invade host files to cause havoc. The past twelve months has seen a significant increase is the prevalence of malware and viruses that use email, fast becoming one of the most efficient and common forms of communication, to propagate. Just as biological viruses use the host as a vehicle for propagation of the species, so to do *in silico* viruses use computers as vehicles to compromise one or more of the three foundations of computer security: *Confidentiality*, *Integrity* or *Availability*.

This study utilized a series of experiments to demonstrate the use of software tools to investigate computer security in relation to malware, specifically worms and viruses. Not only did the experiments validate the findings reported in the breadth of computer reports, but also they uncovered subtle unreported findings on the Klez family of worms. With regards to *Confidentiality and Integrity*, two of the three cornerstones of the computer security foundation, the new findings showed that Klez.H (1) opening of ICQ messenger port 1027, notoriously used by black hats for communication, and potentially creating a back door for an “internet-bot” to receive remote commands for execution, and (2) creation of root level shares on remote systems with Full Control Permissions

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

available to Everyone. With regards to the last cornerstone, *Availability*, this cornerstone was compromised by (1) the use of a mass-emailing engine (not demonstrated) that sends emails to a recipient list disguised as being “from” an entry in the infected systems Windows Address Book, (2) infection of system and program files that directly effect system performance, and (3) creation of processes that undergoes sleep and wakeup routines for scanning the network for new sites of infection.

The findings of this study reinforce the issue that sound information security strategies can provide adequate protection against attacks against *Confidentiality*, *Integrity* and *Availability*. This study demonstrates that a *Defense in Depth* strategy is composed of both the proper end-user awareness training and the enforcement of tactical approaches. These approaches might include, but are not limited to protection via both adequate virus protection / scanning at the perimeter and inside the network, and patching against known exploitable vulnerabilities such as the SANS/FBI Top 20.

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

List of References

[1] Sophos Press Release. “Klez worm is most prolific virus of the year: Windows 32 viruses take clean sweep of 2002 virus chart”. December 4, 2002.
URL: http://www.sophos.com/pressoffice/pressrel/uk/20021204yeartopten.html (March 31, 2003)

[2] MessageLabs Press Release. “Klez heads “rogues gallery” as MessageLabs stops a virus every three seconds in 2002”. December 16 2002.
URL: http://www.messagelabs.com/viewNewsPR.asp?id=113&cmd=PR (March 31, 2003)

[3] Klutz, R.L. and Vines, R.D. The CISSP Prep Guide. John Wiley & Sons. New York. 2001. 2-4.

[4] Harley D., Slade R. and Gattiker U.E. Viruses Revealed. Osborne/McGraw Hill, New York. 2001. 51-79.

[5] Harley D., Slade R. and Gattiker U.E. Viruses Revealed. Osborne/McGraw Hill, New York. 2001. 17-49.

[6] Bosworth S. & Kabay M.E. Computer Security Handbook. John Wiley & Sons, Inc. New York. 2002. 9-1 – 9-20.

[7] Harley D., Slade R. and Gattiker U.E. Viruses Revealed. Osborne/McGraw Hill, New York. 2001. 335-435.

[8] Walter, Russ. The Secret Guide to Computers: Virus Secrets. 28th Ed., Russ Walter. Manchester. 2003.
URL: http://www.secretguide.net/read/index.php?filename=viruses (March 31, 2003)

[9] Internet Security Systems Security Alert. “Outbreak of Klez Family Hybrid Threats”. April 26, 2002.
URL: http://bvlive01.iss.net/issEn/delivery/xforce/alertdetail.jsp?oid=20289 (March 31, 2003)

[10] Hindocha, Neal and Gudmundsson, Atli. “W32.Klez.D@mm”. Symantec Security Response. January 20, 2003.
URL: http://securityresponse.symantec.com/avcenter/venc/data/w32.klez.d@mm.html (March 31, 2003).

[11] Norman Virus Control. “W32/Klez.B@mm”. November 8, 2001
URL: http://www.norman.com/virus_info/w32_klez_b_mm.shtml.(March 31, 2003)

[12] Trend Micro. “WORM_KLEZ.B”. May 21, 2002.
URL: http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_KLEZ.B&VSect=T (March 31, 2003)

[13] Trend Micro Virus Encyclopedia. “WORM_KLEZ”.
URL: http://www.trendmicro.com/en/security/advisories/klez.htm (April 1, 2003)

[14] Podrezov, Alexey. “F-Secure Virus Descriptions: Elkern”. F-Secure Corp. January 17th, 2002.
URL: http://www.f-secure.com/v-descs/elkern.shtml (March 31, 2003)

[15] Ludwig M. The Giant Black Book of Computer Viruses. American Eagle Publications. Show Low. 1998. 317-320.

[16] Kaspersky, Eugene. “TPE - Trident Polymorphic Engine”. Metropolitan Network BBS Inc. 2000.
URL: http://www.avp.ch/avpve/poly-gen/tpe.stm (March 31, 2003)

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

[17] Grimes, Roger A. Malicious Mobile Code: Virus Protection for Windows. O'Reilly & Associates, Inc. Sebastopol, CA. USA. 2001. 22-58, 93-129.

[18] Gudmundsson, Atli. "W32.ElKern.3326". Symantec Security Response. January 16, 2003.
URL: http://securityresponse.symantec.com/avcenter/venc/data/w32.elkern.3326.html (March 31, 2003)

[19] Trend Micro. "WORM_KLEZ.H". April 25, 2002.
URL: http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_KLEZ.H&VSect=T (March 31, 2003)

[20] Ionescu, Costin. " Win32.Klez.H@mm". BitDefender Virus Info. 2003.
URL: http://www.bitdefender.com/virusi/virusi_descrieri.php?virus_id=73 (March 31, 2003)

[21] MessageLabs. "High Risk Alert: W32/klez.H-mm".
URL: http://www.messagelabs.com/viruseye/toptrump.asp?wi=W32/klez.H-mm (March 31, 2003)

[22] Delio, Michelle. "Klez: Hi Mom, We're No. 1". Wired News. May. 24, 2002
URL: http://www.wired.com/news/technology/0,1282,52765,00.html (March 31, 2003)

[23] Trend Micro World Virus Tracking Center. "WORM_KLEZ.H". 2003.
URL:http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_KLEZ.H&VSect=S&Period=All (March 31, 2003)

[24] Microsoft Security Bulletin (MS01-020). "Incorrect MIME Header Can Cause IE to Execute E-mail Attachment". Microsoft Corporation. March 29, 2001.
URL: http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/MS01-020.asp (March 31, 2003)

[25] Hindocha, Neal. "W32.Klez.H@mm". Symantec Security Response. January 22, 2003.
URL: http://securityresponse.symantec.com/avcenter/venc/data/w32.klez.h@mm.html (March 31, 2003)

[26] Microsoft Knowledge Base Article – 156138. "INFO: Limitations of DAO, DAO SDK in NT Service or with Threads". Microsoft Corporation. February 15, 2002.
URL: http://support.microsoft.com/default.aspx?scid=kb%3Ben-us%3B156138 (March 31, 2003)

[27] Stevens W.R. TCP/IP Illustrated, Volume 1: The Protocols. Addison-Wesley. Boston. 1994. 441-459.

[28] Delio, Michelle. "Klez Worm, Not Sender, Hates You". Wired News. April 24, 2002.
URL: http://www.wired.com/news/technology/0,1282,52055,00.html (April 1, 2003)

[29] Symantec Security Response. "W32.ElKern.4926".Last updated January 16, 2003.
URL: http://securityresponse.symantec.com/avcenter/venc/data/pf/w32.elkern.4926.html (April 1, 2003)

[30] Kaspersky, Eugene. "I-Worm.Klez". Metropolitan Network BBS Inc. 2000.
URL: http://www.avp.ch/avpve/worms/email/klez.stm

[31] Danyliw, Roman, Dougherty, Chad and Householder, Allen. "CERT® Advisory CA-2001-22 W32/Sircam Malicious Code". CERT Coordination Center, Carnegie Mellon. Pittsburg. August 23, 2001.
URL: http://www.cert.org/advisories/CA-2001-22.html (April 1, 2003)

[32] Trend Micro Security Info. "PE_ELKERN.D". April 17, 2002.
URL: http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=PE_ELKERN.D&VSect=T (April 1, 2003)

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

[33] Microsoft Knowledge Base Article – 121460. “Common Object File Format (COFF)”. Microsoft Corporation. August 8, 2001.
URL: http://support.microsoft.com/default.aspx?scid=kb%3Ben-us%3B121460 (April 1, 2003)

[34] URL: http://www.graffiti.com/services (April 1, 2003)

[35] Seifried, Kurt.“Port 1033 TCP, UDP“. March 27, 2002.
URL: http://www.seifried.org/security/ports/1000/1033.html (April 1, 2003)

[36] Microsoft Knowledge Base Article – 263201.“Default Processes in Windows 2000”. Microsoft Corporation. October 10, 2002.
URL:http://support.microsoft.com/default.aspx?scid=kb%3Ben-us%3B263201 (April 1, 2003)

[37] Seifried, Kurt.“ Port 1027 TCP, UDP“. March 27, 2002.
URL: http://www.seifried.org/security/ports/1000/1027.html (April 1, 2003)

[38] Fisher, Dennis. “More Net Attacks Loom, CERT Says”. eWEEK. March 13, 2003.
URL: http://www.eweek.com/article2/0,3959,935790,00.asp (April 1, 2003)

[39] Sophos Virus Info. “W32/Cult-A”.
URL: http://www.sophos.com/virusinfo/analyses/w32culta.html (April 1, 2003)

[40] Sheresh, Beth, Sheresh, Doug and Cowart, Robert. “Understanding and Using the NT Registry”, taken from: Microsoft Windows NT Server Administrator's Bible: Option Pack Edition. IDG Books, Boston, MA. 1999.
URL: http://www.windowsitlibrary.com/Content/405/11/3.html (April 1, 2003)

[41] Stevens W.R. TCP/IP Illustrated, Volume 1: The Protocols. Addison-Wesley, Boston. 1994. 53-60.

[42] TrendMicro. “WORM_YAHA.K”. Updated February 5, 2003.
URL: http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_YAHA.K (April 1, 2003)

[43] SANS/FBI Top 20 List. “Top Vulnerabilities to Windows Systems”. March 3, 2003
URL: http://www.sans.org/top20/#W8 (April 1, 2003)

[44] SANS/FBI Top 20 List. “Top Vulnerabilities to Windows Systems”. March 3, 2003
URL: http://www.sans.org/top20/#W5 (April 1, 2003)

[45] SANS/FBI Top 20 List. “Top Vulnerabilities to Windows Systems”. March 3, 2003
URL: http://www.sans.org/top20/#W4 (April 1, 2003)

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Appendix 1: Comparison of Klez family of worms

Summary of Klez variants and their payloads

Variant	Alias	Size (bytes)	Date of Discovery	Payload
Klez.A	KLEZA.A W32.Klez W32.Klez.A W32.Klez.gen@mm W32.Poverty.A@mm	57,345	October 25, 2001	W32.ElKern.3326 Aliases: PE_ELKERN.A W95/Elkern.A
Klez.B	TROJ_KLEZ.B WORM_KLEZ.B	61,441	October 30, 2001	Non-destructive payload[9,10]
Klez.C	KLEZ.C I-Worm.Klez.C W32/Klez.C@mm W32.Klez.gen@mm WORM_KLEZ.C	135,168	November 15, 2001	W32.ElKern.3326 Aliases: PE_ELKERN.A W95/Elkern.A
Klez.D	W32.Klez.D W32.Klez.D@mm W32/Klez.d@MM	65,536	November 8, 2001	W32.ElKern.3326 Aliases: PE_ELKERN.A W95/Elkern.A
Klez.E	Klez.E I-Worm.Klez.E W32.Klez.E W32/Klez.e@MM W32/Klez-E Win32.Klez.E WORM_KLEZ.E	Approx. 85,000	January 17, 2002	W32.Elkern.3587 Aliases: W32.ElKern.B PE_ELKERN.B
Klez.F	Klez KLEZ.F W32/Klez.F@mm W32.Klez.gen@mm WORM_KLEZ.I	78,624 – 87,431	January 22, 2002	W32.Elkern.3587 Aliases: W32.ElKern.B PE_ELKERN.B
Klez.G	I-Worm.W32/Klez.gen@MM W32/Klez-G	94,932	April 17, 2002	W32.Elkern.3587 Aliases: W32.ElKern.B PE_ELKERN.B
Klez.H	Klez.H I-Worm.Klez.h I-Worm.W32/Klez.gen@MM Win32.Klez.H W32.Klez.H W32/Klez.h@MM W32.Klez.H@mm W32/Klez-H WORM_KLEZ.H Related to: Klez.E	Approx. 90,000	April 17, 2002	W32.ElKern.4926 Aliases: Win32.Elkern.C PE_ELKERN.D
Klez.I	I-Worm.Klez.i WORM_KLEZ.I Win32.Klez.I W32/Klez.gen@mm	unknown	April 20, 2002	W32.ElKern.4926 Aliases: Win32.Elkern.C PE_ELKERN.D
Klez.J	KLEZ.J	88,447	September 26, 2002	W32.Elkern.3587 Aliases: W32.ElKern.B PE_ELKERN.B

© SANS Institute 2003, Author retains full rights.

Comparison of the Klez family propagation properties.

Variant	SMTP Server	Source of email addresses	Address in "From:" field	Mail Subject	Message Body[A]	Network infection	Antivirus Process Termination[B]
Klez.A	smtp.yahoo.com smtp.hotmail.com smtp.sina.com	MP8, EXE, SCR, PIF, BAT, TXT, HTM, HTML, WAB, DOC, XLS, CPP, C, PAS, MPQ, MPEG, BAK, MP3	Worm body	Worm body Same for A, C & D	Pre-defined	Yes	Yes Type I
Klez.B	-	-	-	-	-	No	
Klez.C	smtp.yahoo.com smtp.hotmail.com smtp.sina.com	WAB	Worm body	Worm body Same for A, C & D	Pre-defined	Yes	Yes Type I
Klez.D	smtp.yahoo.com smtp.hotmail.com smtp.sina.com	WAB	Worm body	Worm body Same for A, C & D	Pre-defined	No	Yes Type I
Klez.E	Domain identified using "From:" field i.e. smtp.domain.com smtp.domain.net smtp.domain.org	MP8, EXE, SCR, PIF, BAT, TXT, HTM, HTML, WAB, DOC, XLS, CPP, C, PAS, MPQ, MPEG, BAK, MP3	Worm body	Worm body Same for E & F	Random	Yes	Yes Type II
Klez.F	Domain identified using "From:" field i.e. smtp.domain.com smtp.domain.net smtp.domain.org	MP8, EXE, SCR, PIF, BAT, TXT, HTM, HTML, WAB, DOC, XLS, CPP, C, PAS, MPQ, MPEG, BAK, MP3	Worm body	Worm body Same for E & F	Random	Yes	Yes Type II
Klez.G	SMTP server from Registry	MP8, EXE, SCR, PIF, BAT, TXT, HTM, HTML, WAB, DOC, XLS, CPP, C, PAS, MPQ, MPEG, BAK, MP3	infected user's address book	Worm body	Random	Yes	Yes
Klez.H	SMTP server from Registry	MP8, EXE, SCR, PIF, BAT, TXT, HTM, HTML, WAB, DOC, XLS, CPP, C, PAS, MPQ, MPEG, BAK, MP3	infected user's address book	Worm body	Random	Yes	Yes Type II
Klez.I	SMTP server from Registry	MP8, EXE, SCR, PIF, BAT, TXT, HTM, HTML, WAB, DOC, XLS, CPP, C, PAS, MPQ, MPEG, BAK, MP3	infected user's address book	Worm body	Random	Yes	Yes
Klez.J	Failed SMTP tests[a]					?	

Table Notes:

A. "Predefined message body": I'm sorry to do so, but it's helpless to say sorry; I want a good job, I must support my parents; Now you have seen my technical capabilities; How much my year-salary now? NO more than $5,500; What do you think of this fact?; Don't call my names, I have no hostility; Can you help me?

B. Each of these variants will terminate memory processes and in some cases, delete the executable files of some antivirus software:

Type I files containing[b]; SMSS, SCAN, NSPLUGIN, NSCH, EDNT, NSCHED32, NRESQ32, NPSSVC, NOD32, NAVWNT, NAVW32, NAVRUNR, NAVLU32, NAVAPW32, NAVAPSVC, N32SCANW, AVPM, AVPCC, AVP32, AMON, ALERTSVC, _AVPM, _AVPCC, _AVP32

[a] Trend Micro. "WORM_KLEZ.J". September 26, 2002. URL: http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_KLEZ.J&VSect=T (April 1, 2003)

[b] Trend Micro. "WORM_KLEZ.A". October 26, 2002. URL: http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_KLEZ.A&VSect=T (April 1, 2003)

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Type II files containing[c]; _AVP32, _AVPCC, NOD32, NPSSVC, NRESQ32, NSCHED32, NSCHEDNT,NSPLUGIN, NAV, NAVAPSVC, NAVAPW32, NAVLU32, NAVRUNR, NAVW32, _AVPM, ALERTSVC, AMON, AVP32, AVPCC, AVPM, N32SCANW, NAVWNT, ANTIVIR, AVPUPD,AVGCTRL, AVWIN95, SCAN32, VSHWIN32, F-STOPW, F-PROT95, ACKWIN32, VETTRAY, VET95, SWEEP95, PCCWIN98, IOMON98, AVPTC, AVE32, AVCONSOL, FP-WIN, DVP95, F-AGNT95, CLAW95, NVC95, SCAN, VIRUS, LOCKDOWN2000, Norton, Mcafee, Antivir, TASKMGR2

© SANS Institute 2003, Author retains full rights.

[c] Trend Micro. "WORM_KLEZ.E". Last updated March 5, 2002.
URL: http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_KLEZ.E&VSect=T (April 1, 2003)

Appendix 2: Detail of DumpSec

DumpSec on Notebook-B (from Notebook-A)

"All Shares"

3/7/2003 8:11 AM - Somarsoft DumpSec (formerly DumpAcl) - \\Notebook-B

Path (exception dirs and files)	Account	Own	Dir	File
all non-administrative shared directories				
\\Notebook-B\Snort=C:\Snort (disktree)		unprotected (no dacl)		
\\Notebook-B\Snort\	Everyone	all	all	
\\Notebook-B\Snort\	Notebook-B\Administrators		o	

"Shares"

3/7/2003 8:11 AM - Somarsoft DumpSec (formerly DumpAcl) - \\Notebook-B

Share and path	Account	Own	Permission
IPC$= (special admin share)		admin-only (no dacl)	
Snort=C:\Snort (disktree)		unprotected (no dacl)	
ADMIN$=C:\WINNT (special admin share)		admin-only (no dacl)	
C$=C:\ (special admin share)		admin-only (no dacl)	

"Services"

3/7/2003 8:46 PM - Somarsoft DumpSec (formerly DumpAcl) - \\Notebook-B

FriendlyName	Name	Status	Type	Account
Abiosdsk	Abiosdsk	Stopped	Kernel	
abp480n5	abp480n5	Stopped	Kernel	
ACPIEC	ACPIEC	Stopped	Kernel	
adpu160m	adpu160m	Stopped	Kernel	
AFD Networking Support Environment	AFD	Running	Kernel	
Aha154x	Aha154x	Stopped	Kernel	
aic116x	aic116x	Stopped	Kernel	
aic78u2	aic78u2	Stopped	Kernel	
aic78xx	aic78xx	Stopped	Kernel	
Alerter	Alerter	Stopped	Win32	LocalSystem
ami0nt	ami0nt	Stopped	Kernel	
amsint	amsint	Stopped	Kernel	
Application Management	AppMgmt	Stopped	Win32	LocalSystem
asc	asc	Stopped	Kernel	
asc3350p	asc3350p	Stopped	Kernel	
asc3550	asc3550	Stopped	Kernel	
Atdisk	Atdisk	Stopped	Kernel	
Ati HotKey Poller	Ati HotKey Poller	Running	Win32	LocalSystem
ati2mtai	ati2mtai	Running	Kernel	
ATM ARP Client Protocol	Atmarpc	Stopped	Kernel	
Audio Stub Driver	audstub	Running	Kernel	
Automatic Updates	wuauserv	Running	Win32	LocalSystem
Background Intelligent Transfer Service	BITS	Stopped	Win32	LocalSystem
Beep	Beep	Running	Kernel	
BusLogic	BusLogic	Stopped	Kernel	
cd20xrnt	cd20xrnt	Stopped	Kernel	
Cdaudio	Cdaudio	Stopped	Kernel	
Cdfs	Cdfs	Running	Kernel	
CD-ROM Driver	Cdrom	Running	Kernel	
Changer	Changer	Stopped	Kernel	
ClipBook	ClipSrv	Stopped	Win32	LocalSystem
COM+ Event System	EventSystem	Running	Win32	LocalSystem
Computer Browser	Browser	Running	Win32	LocalSystem
Cpqarray	Cpqarray	Stopped	Kernel	
cpqarry2	cpqarry2	Stopped	Kernel	
cpqfcalm	cpqfcalm	Stopped	Kernel	
cpqfws2e	cpqfws2e	Stopped	Kernel	
dac960nt	dac960nt	Stopped	Kernel	
deckzpsx	deckzpsx	Stopped	Kernel	
FriendlyName	**Name**	**Status**	**Type**	**Account**

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

DHCP Client	Dhcp	Running	Win32	LocalSystem
Digital CD Audio Playback Filter Driver	redbook	Stopped	Kernel	
Direct Parallel	Raspti	Running	Kernel	
Direct Parallel Link Driver	Ptilink	Running	Kernel	
Disk Driver	Disk	Running	Kernel	
Diskperf	Diskperf	Running	Kernel	
Distributed Link Tracking Client	TrkWks	Running	Win32	LocalSystem
Distributed Transaction Coordinator	MSDTC	Stopped	Win32	LocalSystem
dmboot	dmboot	Stopped	Kernel	
dmload	dmload	Stopped	Kernel	
DNS Client	Dnscache	Running	Win32	LocalSystem
EFS	EFS	Running	Kernel	
ESS Maestro Audio Driver (WDM)	maestro	Running	Kernel	
Event Log	Eventlog	Running	Win32	LocalSystem
Fastfat	Fastfat	Running	Kernel	
Fax Service	Fax	Stopped	Win32	LocalSystem
Fd16_700	Fd16_700	Stopped	Kernel	
Fips	Fips	Running	Kernel	
fireport	fireport	Stopped	Kernel	
flashpnt	flashpnt	Stopped	Kernel	
Floppy Disk Controller Driver	Fdc	Running	Kernel	
Floppy Disk Driver	Flpydisk	Running	Kernel	
Generic Packet Classifier	Gpc	Running	Kernel	
i8042 Keyboard and PS/2 Mouse Port Driver	i8042prt	Running	Kernel	
Indexing Service	cisvc	Stopped	Win32	LocalSystem
ini910u	ini910u	Stopped	Kernel	
Intel AGP Bus Filter	agp440	Running	Kernel	
Intel PRO Adapter Driver	E100B	Running	Kernel	
IntelIde	IntelIde	Running	Kernel	
Internet Connection Sharing	SharedAccess	Stopped	Win32	LocalSystem
IP in IP Tunnel Driver	IpInIp	Stopped	Kernel	
IP Network Address Translator	IpNat	Stopped	Kernel	
IP Traffic Filter Driver	IpFilterDriver	Stopped	Kernel	
IPSEC driver	IPSEC	Running	Kernel	
IPSEC Policy Agent	PolicyAgent	Running	Win32	LocalSystem
ipsraidn	ipsraidn	Stopped	Kernel	
IPX Traffic Filter Driver	NwlnkFlt	Stopped	Kernel	
IPX Traffic Forwarder Driver	NwlnkFwd	Stopped	Kernel	
IR Enumerator Service	IRENUM	Stopped	Kernel	
Keyboard Class Driver	Kbdclass	Running	Kernel	
KSecDD	KSecDD	Running	Kernel	
lbrtfdc	lbrtfdc	Stopped	Kernel	
Logical Disk Manager	dmserver	Running	Win32	LocalSystem
Logical Disk Manager Administrative Service	dmadmin	Stopped	Win32	LocalSystem
Logical Disk Manager Driver	dmio	Running	Kernel	
lp6nds35	lp6nds35	Stopped	Kernel	
LT Modem Driver	ltmodem5	Running	Kernel	
Messenger	Messenger	Running	Win32	LocalSystem
Microcode Update Driver	Update	Running	Kernel	
Microsoft ACPI Control Method Battery Driver	CmBatt	Running	Kernel	
Microsoft ACPI Driver	ACPI	Running	Kernel	
Microsoft Composite Battery Driver	Compbatt	Running	Kernel	
Microsoft DirectMusic SW Synth (WDM)	DMusic	Stopped	Kernel	
Microsoft HID Class Driver	HidUsb	Running	Kernel	
Microsoft Kernel GS Wavetable Synthesizer	swmidi	Stopped	Kernel	
Microsoft Kernel Wave Audio Mixer	kmixer	Running	Kernel	
Microsoft Streaming Clock Proxy	MSPCLOCK	Stopped	Kernel	
Microsoft Streaming Network Raw Channel Access	RCA	Stopped	Kernel	
Microsoft Streaming Quality Manager Proxy	MSPQM	Stopped	Kernel	
Microsoft Streaming Service Proxy	MSKSSRV	Stopped	Kernel	
Microsoft System Audio Device	sysaudio	Running	Kernel	
Microsoft USB Standard Hub Driver	usbhub	Running	Kernel	
Microsoft USB Universal Host Controller Driver	uhcd	Running	Kernel	
Microsoft WINMM WDM Audio Compatibility Driver	wdmaud	Running	Kernel	
mnmdd	mnmdd	Running	Kernel	
Modem	Modem	Running	Kernel	
MountMgr	MountMgr	Running	Kernel	
Mouse Class Driver	Mouclass	Running	Kernel	
Mouse HID Driver	mouhid	Running	Kernel	
FriendlyName	**Name**	**Status**	**Type**	**Account**

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

mraid35x	mraid35x	Stopped	Kernel	
MRxSmb	MRxSmb	Running	Kernel	
Msfs	Msfs	Running	Kernel	
Mup	Mup	Running	Kernel	
Ncrc710	Ncrc710	Stopped	Kernel	
NDIS Proxy	NDProxy	Running	Kernel	
NDIS System Driver	NDIS	Running	Kernel	
Net Logon	Netlogon	Stopped	Win32	LocalSystem
NetBIOS Interface	NetBIOS	Running	Kernel	
NetBios over Tcpip	NetBT	Running	Kernel	
NetDetect	NetDetect	Stopped	Kernel	
Netgroup Packet Filter	NPF	Running	Kernel	
NetMeeting Remote Desktop Sharing	mnmsrvc	Stopped	Win32	LocalSystem
Network Connections	Netman	Running	Win32	LocalSystem
Network DDE	NetDDE	Stopped	Win32	LocalSystem
Network DDE DSDM	NetDDEdsdm	Stopped	Win32	LocalSystem
NMap	NMap	Running	Win32	LocalSystem
Npfs	Npfs	Running	Kernel	
NT LM Security Support Provider	NtLmSsp	Stopped	Win32	LocalSystem
Ntfs	Ntfs	Running	Kernel	
Null	Null	Running	Kernel	
Parallel class driver	Parallel	Running	Kernel	
Parallel port driver	Parport	Running	Kernel	
PartMgr	PartMgr	Running	Kernel	
ParVdm	ParVdm	Running	Kernel	
PCI Bus Driver	PCI	Running	Kernel	
PCIDump	PCIDump	Stopped	Kernel	
PCIIde	PCIIde	Running	Kernel	
Pcmcia	Pcmcia	Running	Kernel	
Performance Logs and Alerts	SysmonLog	Stopped	Win32	LocalSystem
Plug and Play	PlugPlay	Running	Win32	LocalSystem
PnP ISA/EISA Bus Driver	isapnp	Running	Kernel	
Print Spooler	Spooler	Running	Win32	LocalSystem
Protected Storage	ProtectedStorage		Running	Win32 LocalSystem
ql1080	ql1080	Stopped	Kernel	
Ql10wnt	Ql10wnt	Stopped	Kernel	
ql1240	ql1240	Stopped	Kernel	
ql2100	ql2100	Stopped	Kernel	
QoS RSVP	RSVP	Stopped	Win32	LocalSystem
RAS Asynchronous Media Driver	AsyncMac	Stopped	Kernel	
Rdbss	Rdbss	Running	Kernel	
Remote Access Auto Connection Driver	RasAcd	Running	Kernel	
Remote Access Auto Connection Manager	RasAuto	Stopped	Win32	LocalSystem
Remote Access Connection Manager	RasMan	Stopped	Win32	LocalSystem
Remote Access IP ARP Driver	Wanarp	Running	Kernel	
Remote Access NDIS TAPI Driver	NdisTapi	Running	Kernel	
Remote Access NDIS WAN Driver	NdisWan	Running	Kernel	
Remote Procedure Call (RPC)	RpcSs	Running	Win32	LocalSystem
Remote Procedure Call (RPC) Locator	RpcLocator	Stopped	Win32	LocalSystem
Remote Registry Service	RemoteRegistry		Running	Win32 LocalSystem
Removable Storage	NtmsSvc	Running	Win32	LocalSystem
Routing and Remote Access	RemoteAccess	Stopped	Win32	LocalSystem
RunAs Service	seclogon	Running	Win32	LocalSystem
Security Accounts Manager	SamSs	Running	Win32	LocalSystem
Serenum Filter Driver	serenum	Running	Kernel	
Serial port driver	Serial	Running	Kernel	
Server	lanmanserver	Running	Win32	LocalSystem
Sfloppy	Sfloppy	Stopped	Kernel	
sglfb	sglfb	Stopped	Kernel	
Simbad	Simbad	Stopped	Kernel	
Smart Card	SCardSvr	Stopped	Win32	LocalSystem
Smart Card Helper	SCardDrv	Stopped	Win32	LocalSystem
Software Bus Driver	swenum	Running	Kernel	
Sparrow	Sparrow	Stopped	Kernel	
Srv	Srv	Running	Kernel	
Standard IDE/ESDI Hard Disk Controller	atapi	Running	Kernel	
sym_hi	sym_hi	Stopped	Kernel	
symc810	symc810	Stopped	Kernel	
FriendlyName	**Name**	**Status**	**Type**	**Account**
symc8xx	symc8xx	Stopped	Kernel	

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

System Event Notification	SENS	Running	Win32	LocalSystem
Task Scheduler	Schedule	Running	Win32	LocalSystem
TCP/IP NetBIOS Helper Service	LmHosts	Running	Win32	LocalSystem
TCP/IP Protocol Driver	Tcpip	Running	Kernel	
Telephony	TapiSrv	Running	Win32	LocalSystem
Telnet	TlntSvr	Stopped	Win32	LocalSystem
Texas Instruments OHCI Compliant IEEE 1394 Host Controller	ohci1394	Running	Kernel	
tga	tga	Stopped	Kernel	
Udfs	Udfs	Stopped	Kernel	
ultra66	ultra66	Stopped	Kernel	
Uninterruptible Power Supply	UPS	Stopped	Win32	LocalSystem
Utility Manager	UtilMan	Stopped	Win32	LocalSystem
VgaSave	VgaSave	Running	Kernel	
Volume Manager Driver	Ftdisk	Running	Kernel	
WAN Miniport (L2TP)	Rasl2tp	Running	Kernel	
WAN Miniport (PPTP)	PptpMiniport	Running	Kernel	
Windows Installer	MSIServer	Stopped	Win32	LocalSystem
Windows Management Instrumentation	WinMgmt	Running	Win32	LocalSystem
Windows Management Instrumentation Driver Extensions	Wmi	Running	Win32	LocalSystem
Windows Time	W32Time	Stopped	Win32	LocalSystem
Workstation	lanmanworkstation	Running	Win32	LocalSystem
Xircom Cardbus Ethernet 10/100 Adapter Driver	CBEN5	Stopped	Kernel	

DumpSec on Notebook-A (from Notebook-B)

"All Shares"

3/7/2003 8:18 AM - Somarsoft DumpSec (formerly DumpAcl) - \\Notebook-A

Path (exception dirs and files)	Account	Own	Dir	File
all non-administrative shared directories				
\\Notebook-A\dumpsec=C:\Documents and Settings\Administrator\Desktop\Klez_tools\dumpsec (disktree)		unprotected (no dacl)		
\\Notebook-A\dumpsec\	Everyone	o		
\\Notebook-A\dumpsec\		unprotected (null dacl)		
\\Notebook-A\Accelrys=C:\Program Files\Accelrys (disktree)		unprotected (no dacl)		
\\Notebook-A\Accelrys\	Everyone	o		
\\Notebook-A\Accelrys\		unprotected (null dacl)		
\\Notebook-A\TTC1=C:\TEMP (disktree)	Everyone	all		
\\Notebook-A\TTC1=C:\TEMP (disktree)	?unknown	all		
\\Notebook-A\TTC1\	Everyone	o		
\\Notebook-A\TTC1\		unprotected (null dacl)		
\\Notebook-A\print$=C:\WINNT\System32\spool\drivers (disktree)			Everyone	read
\\Notebook-A\print$=C:\WINNT\System32\spool\drivers (disktree)			Notebook-A\Administrators	all
\\Notebook-A\print$=C:\WINNT\System32\spool\drivers (disktree)			Notebook-A\Power Users	all
\\Notebook-A\print$\	Everyone	o		
\\Notebook-A\print$\		unprotected (null dacl)		
\\Notebook-A\D=D:\ (disktree)		unprotected (no dacl)		
\\Notebook-A\D\		==>access denied		

"Shares"

3/7/2003 9:29 PM - Somarsoft DumpSec (formerly DumpAcl) - \\Notebook-A

Share and path	Account	Own	Permission

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

IPC$= (special admin share) admin-only (no dacl)

TTC1=C:\TEMP (disktree) Everyone all
TTC1=C:\TEMP (disktree) ?unknown all

print$=C:\WINNT\System32\spool\drivers (disktree) Everyone read
print$=C:\WINNT\System32\spool\drivers (disktree) Notebook-A\Administrators all
print$=C:\WINNT\System32\spool\drivers (disktree) Notebook-A\Power Users all

D=D:\ (disktree) unprotected (no dacl)

Accelrys=C:\Program Files\Accelrys (disktree) unprotected (no dacl)

ADMIN$=C:\WINNT (special admin share) admin-only (no dacl)

C$=C:\ (special admin share) admin-only (no dacl)

dumpsec=C:\Documents and Settings\Administrator\Desktop\Klez_tools\dumpsec (disktree) unprotected (no dacl)

"Services"

3/7/2003 9:04 PM - Somarsoft DumpSec (formerly DumpAcl) - \\Notebook-A

FriendlyName	Name	Status	Type	Account
%CW10.Service.DispName%	CW10	Stopped	Kernel	
3Com 10/100 Mini PCI Ethernet Adapter NDIS5 Driver	EL556	Running	Kernel	
Abiosdsk	Abiosdsk	Stopped	Kernel	
abp480n5	abp480n5	Stopped	Kernel	
adpu160m	adpu160m	Stopped	Kernel	
AFD Networking Support Environment	AFD	Running	Kernel	
Aha154x	Aha154x	Stopped	Kernel	
aic116x	aic116x	Stopped	Kernel	
aic78u2	aic78u2	Stopped	Kernel	
aic78xx	aic78xx	Stopped	Kernel	
Alerter	Alerter	Stopped	Win32	LocalSystem
ami0nt	ami0nt	Stopped	Kernel	
amsint	amsint	Stopped	Kernel	
Application Management	AppMgmt	Stopped	Win32	LocalSystem
asc	asc	Stopped	Kernel	
asc3350p	asc3350p	Stopped	Kernel	
asc3550	asc3550	Stopped	Kernel	
Atdisk	Atdisk	Stopped	Kernel	
Ati HotKey Poller	Ati HotKey Poller	Running	Win32	LocalSystem
ati2mpab	ati2mpab	Stopped	Kernel	
ati2mtai	ati2mtai	Running	Kernel	
atirage3	atirage3	Stopped	Kernel	
ATM ARP Client Protocol	Atmarpc	Stopped	Kernel	
Audio Stub Driver	audstub	Running	Kernel	
Automatic Updates	wuauserv	Running	Win32	LocalSystem
Background Intelligent Transfer Service	BITS	Stopped	Win32	LocalSystem
Beep	Beep	Running	Kernel	
BusLogic	BusLogic	Stopped	Kernel	
cd20xrnt	cd20xrnt	Stopped	Kernel	
Cdaudio	Cdaudio	Stopped	Kernel	
Cdfs	Cdfs	Running	Kernel	
Cdr4_2K	Cdr4_2K	Running	Kernel	
Cdralw2k	Cdralw2k	Running	Kernel	
CD-ROM Driver	Cdrom	Running	Kernel	
cdudf	cdudf	Running	Kernel	
Changer	Changer	Stopped	Kernel	
FriendlyName	**Name**	**Status**	**Type**	**Account**
Cisco Systems IPsec Driver	CVPNDRV	Running	Kernel	
Cisco Systems, Inc. VPN Service	CVPND	Running	Win32	LocalSystem
ClipBook	ClipSrv	Stopped	Win32	LocalSystem

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

COM+ Event System	EventSystem	Running	Win32	LocalSystem
Computer Browser	Browser	Running	Win32	LocalSystem
Cpqarray	Cpqarray	Stopped	Kernel	
cpqarry2	cpqarry2	Stopped	Kernel	
cpqfcalm	cpqfcalm	Stopped	Kernel	
cpqfws2e	cpqfws2e	Stopped	Kernel	
dac960nt	dac960nt	Stopped	Kernel	
deckzpsx	deckzpsx	Stopped	Kernel	
Deterministic Network Enhancer Miniport	DNE	Running	Kernel	
DHCP Client	Dhcp	Running	Win32	LocalSystem
Digital CD Audio Playback Filter Driver	redbook	Stopped	Kernel	
Direct Parallel	Raspti	Running	Kernel	
Direct Parallel Link Driver	Ptilink	Running	Kernel	
Disk Driver	Disk	Running	Kernel	
Diskperf	Diskperf	Running	Kernel	
Distributed Link Tracking Client	TrkWks	Running	Win32	LocalSystem
Distributed Transaction Coordinator	MSDTC	Stopped	Win32	LocalSystem
dmboot	dmboot	Stopped	Kernel	
dmload	dmload	Stopped	Kernel	
DNS Client	Dnscache	Running	Win32	LocalSystem
dvd_2K	dvd_2K	Stopped	Kernel	
EFS	EFS	Stopped	Kernel	
ESS Maestro Audio Driver (WDM)	maestro	Running	Kernel	
Event Log	Eventlog	Running	Win32	LocalSystem
Fastfat	Fastfat	Running	Kernel	
Fax Service	Fax	Stopped	Win32	LocalSystem
Fd16_700	Fd16_700	Stopped	Kernel	
Fips	Fips	Running	Kernel	
fireport	fireport	Stopped	Kernel	
flashpnt	flashpnt	Stopped	Kernel	
Floppy Disk Controller Driver	Fdc	Running	Kernel	
Floppy Disk Driver	Flpydisk	Stopped	Kernel	
Game Port Enumerator	gameenum	Stopped	Kernel	
Generic Packet Classifier	Gpc	Running	Kernel	
i8042 Keyboard and PS/2 Mouse Port Driver	i8042prt	Running	Kernel	
Indexing Service	cisvc	Stopped	Win32	LocalSystem
ini910u	ini910u	Stopped	Kernel	
Intel AGP Bus Filter	agp440	Running	Kernel	
IntelIde	IntelIde	Running	Kernel	
Internet Connection Sharing	SharedAccess	Stopped	Win32	LocalSystem
IP in IP Tunnel Driver	IpInIp	Stopped	Kernel	
IP Network Address Translator	IpNat	Stopped	Kernel	
IP Traffic Filter Driver	IpFilterDriver	Stopped	Kernel	
IPSEC driver	IPSEC	Stopped	Kernel	
IPSEC Policy Agent	PolicyAgent	Stopped	Win32	LocalSystem
ipsraidn	ipsraidn	Stopped	Kernel	
IPX Traffic Filter Driver	NwlnkFlt	Stopped	Kernel	
IPX Traffic Forwarder Driver	NwlnkFwd	Stopped	Kernel	
IR Enumerator Service	IRENUM	Stopped	Kernel	
Keyboard Class Driver	Kbdclass	Running	Kernel	
KSecDD	KSecDD	Running	Kernel	
lbrtfdc	lbrtfdc	Stopped	Kernel	
Logical Disk Manager	dmserver	Running	Win32	LocalSystem
Logical Disk Manager Administrative Service	dmadmin	Stopped	Win32	LocalSystem
Logical Disk Manager Driver	dmio	Running	Kernel	
lp6nds35	lp6nds35	Stopped	Kernel	
Messenger	Messenger	Running	Win32	LocalSystem
Microcode Update Driver	Update	Running	Kernel	
Microsoft ACPI Control Method Battery Driver	CmBatt	Running	Kernel	
Microsoft ACPI Driver	ACPI	Running	Kernel	
Microsoft Composite Battery Driver	Compbatt	Running	Kernel	
Microsoft DirectMusic SW Synth (WDM)	DMusic	Stopped	Kernel	
Microsoft Embedded Controller Driver	ACPIEC	Running	Kernel	
Microsoft HID Class Driver	HidUsb	Stopped	Kernel	
Microsoft Kernel GS Wavetable Synthesizer	swmidi	Stopped	Kernel	
FriendlyName	**Name**	**Status**	**Type**	**Account**
Microsoft Kernel Wave Audio Mixer	kmixer	Stopped	Kernel	
Microsoft MPU-401 MIDI UART Driver	ms_mpu401	Stopped	Kernel	
Microsoft Streaming Clock Proxy	MSPCLOCK	Stopped	Kernel	
Microsoft Streaming Network Raw Channel Access	RCA	Stopped	Kernel	

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Microsoft Streaming Quality Manager Proxy	MSPQM	Stopped	Kernel	
Microsoft Streaming Service Proxy	MSKSSRV	Stopped	Kernel	
Microsoft System Audio Device	sysaudio	Running	Kernel	
Microsoft USB Standard Hub Driver	usbhub	Running	Kernel	
Microsoft USB Universal Host Controller Driver	uhcd	Running	Kernel	
Microsoft WINMM WDM Audio Compatibility Driver	wdmaud	Running	Kernel	
mmc_2K	mmc_2K	Running	Kernel	
mnmdd	mnmdd	Running	Kernel	
Modem	Modem	Running	Kernel	
MountMgr	MountMgr	Running	Kernel	
Mouse Class Driver	Mouclass	Running	Kernel	
Mouse HID Driver	mouhid	Stopped	Kernel	
mraid35x	mraid35x	Stopped	Kernel	
MRxSmb	MRxSmb	Running	Kernel	
Msfs	Msfs	Running	Kernel	
Mup	Mup	Running	Kernel	
Ncrc710	Ncrc710	Stopped	Kernel	
NDC Network Agent	ndcprtns	Running	Kernel	
NDIS Proxy	NDProxy	Running	Kernel	
NDIS System Driver	NDIS	Running	Kernel	
Net Logon	Netlogon	Stopped	Win32	LocalSystem
NetBIOS Interface	NetBIOS	Running	Kernel	
NetBios over Tcpip	NetBT	Running	Kernel	
NetDetect	NetDetect	Stopped	Kernel	
Netgroup Packet Filter	NPF	Running	Kernel	
NetMeeting Remote Desktop Sharing	mnmsrvc	Stopped	Win32	LocalSystem
Network Connections	Netman	Running	Win32	LocalSystem
Network DDE	NetDDE	Stopped	Win32	LocalSystem
Network DDE DSDM	NetDDEdsdm	Stopped	Win32	LocalSystem
NMap	NMap	Running	Win32	LocalSystem
Npfs	Npfs	Running	Kernel	
NT LM Security Support Provider	NtLmSsp	Stopped	Win32	LocalSystem
Ntfs	Ntfs	Stopped	Kernel	
Null	Null	Running	Kernel	
OfficeScanNT Listener	tmlisten	Running	Win32	LocalSystem
OfficeScanNT RealTime Scan	ntrtscan	Running	Win32	LocalSystem
OracleOraHome81Agent	OracleOraHome81Agent	Stopped	Win32	LocalSystem
OracleOraHome81ClientCache	OracleOraHome81ClientCache	Stopped	Win32	LocalSystem
OracleOraHome81DataGatherer	OracleOraHome81DataGatherer	Running	Win32	LocalSystem
OracleOraHome81HTTPServer	OracleOraHome81HTTPServer	Running	Win32	LocalSystem
OracleOraHome81PagingServer	OracleOraHome81PagingServer	Stopped	Win32	LocalSystem
OracleOraHome81TNSListener	OracleOraHome81TNSListener	Stopped	Win32	LocalSystem
OracleServiceTTC	OracleServiceTTC	Running	Win32	LocalSystem
Parallel class driver	Parallel	Running	Kernel	
Parallel port driver	Parport	Running	Kernel	
PartMgr	PartMgr	Running	Kernel	
ParVdm	ParVdm	Running	Kernel	
PCI Bus Driver	PCI	Running	Kernel	
PCIDump	PCIDump	Stopped	Kernel	
PCIIde	PCIIde	Stopped	Kernel	
Pcmcia	Pcmcia	Running	Kernel	
Performance Logs and Alerts	SysmonLog	Stopped	Win32	LocalSystem
Plug and Play	PlugPlay	Running	Win32	LocalSystem
PnP ISA/EISA Bus Driver	isapnp	Running	Kernel	
Print Spooler	Spooler	Running	Win32	LocalSystem
Protected Storage	ProtectedStorage	Running	Win32	LocalSystem
pwd_2k	pwd_2k	Running	Kernel	
ql1080	ql1080	Stopped	Kernel	
Ql10wnt	Ql10wnt	Stopped	Kernel	
ql1240	ql1240	Stopped	Kernel	
ql2100	ql2100	Stopped	Kernel	
QoS RSVP	RSVP	Stopped	Win32	LocalSystem
RAS Asynchronous Media Driver	AsyncMac	Stopped	Kernel	
Rdbss	Rdbss	Running	Kernel	
FriendlyName	**Name**	**Status**	**Type**	**Account**
Remote Access Auto Connection Driver	RasAcd	Running	Kernel	
Remote Access Auto Connection Manager	RasAuto	Stopped	Win32	LocalSystem
Remote Access Connection Manager	RasMan	Running	Win32	LocalSystem
Remote Access IP ARP Driver	Wanarp	Running	Kernel	
Remote Access NDIS TAPI Driver	NdisTapi	Running	Kernel	

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Remote Access NDIS WAN Driver	NdisWan	Running	Kernel	
Remote Procedure Call (RPC)	RpcSs	Running	Win32	LocalSystem
Remote Procedure Call (RPC) Locator	RpcLocator	Stopped	Win32	LocalSystem
Remote Registry Service	RemoteRegistry	Running	Win32	LocalSystem
Removable Storage	NtmsSvc	Running	Win32	LocalSystem
Routing and Remote Access	RemoteAccess	Stopped	Win32	LocalSystem
RunAs Service	seclogon	Running	Win32	LocalSystem
Security Accounts Manager	SamSs	Running	Win32	LocalSystem
Serenum Filter Driver	serenum	Running	Kernel	
Serial port driver	Serial	Running	Kernel	
Server	lanmanserver	Running	Win32	LocalSystem
Sfloppy	Sfloppy	Stopped	Kernel	
sglfb	sglfb	Stopped	Kernel	
Simbad	Simbad	Stopped	Kernel	
Smart Card	SCardSvr	Stopped	Win32	LocalSystem
Smart Card Helper	SCardDrv	Stopped	Win32	LocalSystem
Software Bus Driver	swenum	Running	Kernel	
Sparrow	Sparrow	Stopped	Kernel	
Srv	Srv	Running	Kernel	
Standard IDE/ESDI Hard Disk Controller	atapi	Running	Kernel	
Still Image Service	StiSvc	Running	Win32	LocalSystem
sym_hi	sym_hi	Stopped	Kernel	
symc810	symc810	Stopped	Kernel	
symc8xx	symc8xx	Stopped	Kernel	
SymEvent	SymEvent	Stopped	Kernel	
Synaptics TouchPad Driver	SynTP	Running	Kernel	
System Event Notification	SENS	Running	Win32	LocalSystem
Task Scheduler	Schedule	Running	Win32	LocalSystem
TCP/IP NetBIOS Helper Service	LmHosts	Running	Win32	LocalSystem
TCP/IP Protocol Driver	Tcpip	Running	Kernel	
Telephony	TapiSrv	Running	Win32	LocalSystem
Telnet	TlntSvr	Stopped	Win32	LocalSystem
tga	tga	Stopped	Kernel	
Trend Micro Filter	TmFilter	Running	Kernel	
Trend Micro VSAPI NT	VSApiNt	Running	Kernel	
UdfReadr	UdfReadr	Running	Kernel	
Udfs	Udfs	Stopped	Kernel	
ultra66	ultra66	Stopped	Kernel	
Uninterruptible Power Supply	UPS	Stopped	Win32	LocalSystem
USB Mass Storage Driver	USBSTOR	Stopped	Kernel	
USB Scanner Driver	usbscan	Stopped	Kernel	
Utility Manager	UtilMan	Stopped	Win32	LocalSystem
VgaSave	VgaSave	Running	Kernel	
Volume Manager Driver	Ftdisk	Running	Kernel	
WAN Miniport (L2TP)	Rasl2tp	Running	Kernel	
WAN Miniport (PPTP)	PptpMiniport	Running	Kernel	
WDHABBGMiniPCI Winmodem	WDHABBG	Running	Kernel	
Windows Installer	MSIServer	Stopped	Win32	LocalSystem
Windows Management Instrumentation	WinMgmt	Running	Win32	LocalSystem
Windows Management Instrumentation Driver Extensions	Wmi	Running	Win32	LocalSystem
Windows Time	W32Time	Stopped	Win32	LocalSystem
WMDM PMSP Service	WMDM PMSP Service	Running	Win32	LocalSystem
Workstation	lanmanworkstation	Running	Win32	LocalSystem
Xircom CardBus Ethernet 10/100 Adapter family	CBEN5	Stopped	Kernel	

© SANS Institute 2003, Author retains full rights.

Appendix 3: Detail of Nmap scans

Nmap on Notebook-B (from Notebook-A)

```
# nmap (V. 3.00) scan initiated Fri Mar 07 16:08:14 2003 as: nmap -v -O -oN Notebook-B 192.168.1.112
Interesting ports on NOTEBOOK-B (192.168.1.112):
(The 1597 ports scanned but not shown below are in state: closed)
Port       State       Service
135/tcp    open        loc-srv
139/tcp    open        netbios-ssn
445/tcp    open        microsoft-ds
1025/tcp   open        NFS-or-IIS
Remote operating system guess: Windows 2000/XP/ME
TCP Sequence Prediction: Class=random positive increments
                         Difficulty=16658 (Worthy challenge)
IPID Sequence Generation: Incremental

# Nmap run completed at Fri Mar 07 16:08:35 2003 -- 1 IP address (1 host up) scanned in 21 seconds
```

Nmap on Notebook-B (from Notebook-A)

```
# nmap (V. 3.00) scan initiated Fri Mar 07 16:09:01 2003 as: nmap -v -O -oN Notebook-A 192.168.1.100
Interesting ports on NOTEBOOK-A (192.168.1.100):
(The 1593 ports scanned but not shown below are in state: closed)
Port       State       Service
80/tcp     open        http
135/tcp    open        loc-srv
139/tcp    open        netbios-ssn
443/tcp    open        https
445/tcp    open        microsoft-ds
1031/tcp   open        iad2
1033/tcp   open        netinfo
12345/tcp  open        NetBus
No exact OS matches for host (If you know what OS is running on it, see http://www.insecure.org/cgi-bin/nmap-
submit.cgi).
TCP/IP fingerprint:
SInfo(V=3.00%P=i686-pc-windows-windows%D=3/7%Time=3E691898%O=80%C=1)
TSeq(Class=RI%gcd=1%SI=2AF3%IPID=I%TS=0)
TSeq(Class=RI%gcd=1%SI=3EC4%IPID=I%TS=0)
TSeq(Class=RI%gcd=1%SI=28A4%IPID=I%TS=0)
T1(Resp=Y%DF=Y%W=FF00%ACK=S++%Flags=AS%Ops=MNWNNT)
T2(Resp=Y%DF=N%W=0%ACK=S%Flags=AR%Ops=)
T3(Resp=Y%DF=Y%W=FF00%ACK=S++%Flags=AS%Ops=MNWNNT)
T4(Resp=Y%DF=N%W=0%ACK=O%Flags=R%Ops=)
T5(Resp=Y%DF=N%W=0%ACK=S++%Flags=AR%Ops=)
T6(Resp=Y%DF=N%W=0%ACK=O%Flags=R%Ops=)
T7(Resp=Y%DF=N%W=0%ACK=S++%Flags=AR%Ops=)
PU(Resp=Y%DF=N%TOS=0%IPLEN=38%RIPTL=148%RIPCK=E%UCK=E%ULEN=134%DAT=E)

TCP Sequence Prediction: Class=random positive increments
Difficulty=10404 (Worthy challenge)
IPID Sequence Generation: Incremental

# Nmap run completed at Fri Mar 07 16:09:28 2003 -- 1 IP address (1 host up) scanned in 27 seconds
```

© SANS Institute 2003, Author retains full rights.

Appendix 4: Detail of Windows Trace Log Reports

Windows Event Trace Session Report: Notebook-A

Version:	2600	
Type:	Default	
Build:	2195	
Processors:	1	
Start Time:	7-Mar-03	16:25:00
End Time:	7-Mar-03	16:27:00
Duration:	120	sec
Trace Name:		
File Name:		
Start Time:	7-Mar-03	16:25:00
End	7-Mar-03	16:27:00
Duration	120	sec

Image Statistics

Image Name	PID	Threads Launched	Threads Used	Process KCPU(ms)	Process UCPU(ms)	Transaction KCPU(ms)	Transaction UCPU(ms)	CPU%
snort.exe	0x00000788	1	1	180	120	0	0	0.25
smlogsvc.exe	0x0000077C	4	0	0	0	0	0	0
mmc.exe	0x00000728	5	1	20	10	0	0	0.02
rundll32.exe	0x00000798	1	0	0	0	0	0	0
wuauclt.exe	0x00000734	4	0	0	0	0	0	0
cmd.exe	0x00000714	1	1	0	0	0	0	0
config.exe	0x000006F8	1	0	0	0	0	0	0
DirectCD.exe	0x000006EC	3	0	0	0	0	0	0
CreateCD50.exe	0x000006E0	1	0	0	0	0	0	0
INSTAN~1.EXE	0x000006B8	1	0	0	0	0	0	0
atiptaxx.exe	0x00000578	2	0	0	0	0	0	0
SynTPEnh.exe	0x000006AC	3	1	30	0	0	0	0.02
SynTPLpr.exe	0x00000698	3	0	0	0	0	0	0
Explorer.EXE	0x000005F0	13	1	10	0	0	0	0.01
svchost.exe	0x000004D4	6	0	0	0	0	0	0
mspmspsv.exe	0x000004C8	2	0	0	0	0	0	0
WinMgmt.exe	0x000004A0	8	0	0	0	0	0	0
tmlisten.exe	0x000003C4	6	0	0	0	0	0	0
stisvc.exe	0x00000388	4	0	0	0	0	0	0
MSTask.exe	0x00000370	6	0	0	0	0	0	0
Apache.exe	0x00000368	53	0	0	0	0	0	0
java.exe	0x0000035C	10	0	0	0	0	0	0
regsvc.exe	0x00000350	2	0	0	0	0	0	0
ORACLE.EXE	0x00000320	16	3	50	20	0	0	0.06
Apache.exe	0x00000314	4	0	0	0	0	0	0
vppdc.exe	0x000002F0	4	0	0	0	0	0	0
ntrtscan.exe	0x00000258	9	0	0	0	0	0	0
nmapserv.exe	0x0000023C	2	0	0	0	0	0	0
svchost.exe	0x00000224	26	0	0	0	0	0	0
Ati2evxx.exe	0x0000020C	3	1	10	10	0	0	0.02
spoolsv.exe	0x000001D4	11	0	0	0	0	0	0
svchost.exe	0x000001B4	10	0	0	0	0	0	0
cvpnd.exe	0x0000016C	4	0	0	0	0	0	0
lsass.exe	0x000000F8	11	3	10	0	0	0	0.01
services.exe	0x000000EC	40	2	20	20	0	0	0.03
winlogon.exe	0x000000B8	17	0	0	0	0	0	0
csrss.exe	0x000000BC	11	5	17950	140	0	0	15.04
smss.exe	0x000000A4	6	0	0	0	0	0	0
System	0x00000008	68	13	80	0	0	0	0.07
Idle	0x00000000	2	1	101200	0	0	0	84.16
Total:		384	33	119560	320	0	0	99.69%

Disk Totals

Disk Name	Reads	Kb	Writes	Kb
0	96	18	256	5

Disk 0

Authority	PID	Image Name	Read Count	Kb	Write Count	Kb
system	0x0788	snort.exe	51	13	0	0
system	0x0728	mmc.exe	7	19	0	0
system	0x0714	cmd.exe	1	4	0	0
\\NT AUTHORITY\SYSTEM	0x0320	ORACLE.EXE	0	0	117	8
\\NT AUTHORITY\SYSTEM	0x00F8	lsass.exe	11	22	0	0
\\NT AUTHORITY\SYSTEM	0x00BC	csrss.exe	1	4	0	0
\\NT AUTHORITY\SYSTEM	0x0008	System	25	29	132	3

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Windows Event Trace Session Report: Kl Notebook-B

Version:	2600	
Type:	Default	
Build:	2195	
Processors:	1	
Start Time:	7-Mar-03	16:25:00
End Time:	7-Mar-03	16:27:00
Duration:	120	sec
Trace Name:		
File Name:		
Start Time:	7-Mar-03	16:25:00
End	7-Mar-03	16:27:00
Duration	120	sec

Image Statistics

Image Name	PID	Threads Launched	Threads Used	Process KCPU(ms)	Process UCPU(ms)	Transaction KCPU(ms)	Transaction UCPU(ms)	CPU%
AnuF.exe	0x00000450	7	2	380	390	0	0	0.64
Winklo.exe	**0x0000047C**	**36**	**8**	**390**	**40750**	**0**	**0**	**34.12**
I-Worm.Klez.h	**0x00000120**	**1**	**1**	**0**	**30**	**0**	**0**	**0.02**
snort.exe	0x0000011C	6	1	100	80	0	0	0.15
smlogsvc.exe	0x0000020C	9	0	0	0	0	0	0
mmc.exe	0x000003EC	10	1	30	100	0	0	0.11
cmd.exe	0x000002EC	6	1	0	0	0	0	0
cmd.exe	0x00000458	6	0	0	0	0	0	0
wuauclt.exe	0x000003A0	9	0	0	0	0	0	0
atiptaxx.exe	0x00000354	8	0	0	0	0	0	0
explorer.exe	0x00000340	20	1	0	10	0	0	0.01
svchost.exe	0x00000274	12	1	0	10	0	0	0.01
winmgmt.exe	0x0000026C	14	0	0	0	0	0	0
mstask.exe	0x00000244	12	0	0	0	0	0	0
regsvc.exe	0x00000230	8	0	0	0	0	0	0
nmapserv.exe	0x00000208	8	0	0	0	0	0	0
svchost.exe	0x000001E8	24	0	0	0	0	0	0
Ati2evxx.exe	0x000001D8	9	1	40	20	0	0	0.05
SPOOLSV.EXE	0x000001BC	16	1	0	10	0	0	0.01
svchost.exe	0x0000019C	16	0	0	0	0	0	0
lsass.exe	0x000000E8	22	2	70	30	0	0	0.08
services.exe	0x000000DC	43	2	20	10	0	0	0.02
winlogon.exe	0x000000A8	26	4	1490	2460	0	0	3.28
csrss.exe	0x000000AC	16	5	1440	30	0	0	1.22
smss.exe	0x00000094	12	0	0	0	0	0	0
System	0x00000008	51	11	1280	0	0	0	1.06
Idle	0x00000000	2	1	70910	0	0	0	58.82
Total:		409	43	76150	43930	0	0	99.60%

Disk Totals

Disk Name	Reads	Kb	Writes	Kb
0	1920	28	2775	7

Disk 0

Authority	PID	Image Name	Read Count	Kb	Write Count	Kb
\\NT AUTHORITY\SYSTEM	0x0450	AnuF.exe	982	26	0	0
\\NT AUTHORITY\SYSTEM	**0x047C**	**Winklo.exe**	**116**	**15**	**3**	**8**
system	**0x0120**	**I-Worm.Klez.h**	**5**	**15**	**0**	**0**
system	0x011C	snort.exe	3	11	0	0
system	0x02EC	cmd.exe	1	4	0	0
\\NT AUTHORITY\SYSTEM	0x01D8	Ati2evxx.exe	25	22	0	0
\\NT AUTHORITY\SYSTEM	0x00DC	services.exe	1	4	64	4
\\NT AUTHORITY\SYSTEM	0x00A8	winlogon.exe	787	32	0	0
\\NT AUTHORITY\SYSTEM	0x0008	System	0	0	2631	7
system	0x0000	Idle	0	0	77	4

© SANS Institute 2003, Author retains full rights.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Appendix 5: Detail of Regdmp from Notebook-B

```
regdmp HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Applets\Regedit
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Applets\Regedit
   View = REG_BINARY 0x00000040 0x0000002c 0x00000000 0x00000001 0xffff8300 \
        0xffff8300 0xffffffff 0xffffffff 0x000000e3 0x0000001d 0x000004fd \
        0x0000030a 0x00000164 0x00000078 0x00000078 0x00000120 0x00000001
   FindFlags = REG_DWORD 0x0000000e
   LastKey = My \
          Computer\HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\Winklo
   Favorites
```

```
regdmp HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Enum\Root\LEGACY_WINKLO
HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Enum\Root\LEGACY_WINKLO [17 8]
   NextInstance = REG_DWORD 0x00000001
   0000 [17 8]
      Service = Winklo
      Legacy = REG_DWORD 0x00000001
      ConfigFlags = REG_DWORD 0x00000000
      Class = LegacyDriver
      ClassGUID = {8ECC055D-047F-11D1-A537-0000F8753ED1}
      DeviceDesc = Winklo
      Control [17 8]
         ActiveService = Winklo
```

```
regdmp HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\Winklo
HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\Winklo
   Type = REG_DWORD 0x00000110
   Start = REG_DWORD 0x00000002
   ErrorControl = REG_DWORD 0x00000000
   ImagePath = REG_EXPAND_SZ C:\WINNT\System32\Winklo.exe
   DisplayName = Winklo
   ObjectName = LocalSystem
   Security [17 1]
      Security = REG_BINARY 0x000000b8 0x80140001 0x000000a0 0x000000ac \
               0x00000014 0x00000030 0x001c0002 0x00000001 0x00148002 \
               0x000f01ff 0x00000101 0x01000000 0x00000000 0x00700002 \
               0x00000004 0x00180000 0x000201fd 0x00000101 0x05000000 \
               0x00000012 0x000df910 0x001c0000 0x000f01ff 0x00000201 \
               0x05000000 0x00000020 0x00000220 0x000df970 0x00180000 \
               0x0002018d 0x00000101 0x05000000 0x0000000b 0x00000220 \
               0x001c0000 0x000201fd 0x00000201 0x05000000 0x00000020 \
               0x00000223 0x000df970 0x00000101 0x05000000 0x00000012 \
               0x00000101 0x05000000 0x00000012
   Enum
      0 = Root\LEGACY_WINKLO\0000
      Count = REG_DWORD 0x00000001
      NextInstance = REG_DWORD 0x00000001
```

```
regdmp HKEY_LOCAL_MACHINE\SYSTEM\ControlSet002\Enum\Root\LEGACY_WINKLO
HKEY_LOCAL_MACHINE\SYSTEM\ControlSet002\Enum\Root\LEGACY_WINKLO [17 8]
   NextInstance = REG_DWORD 0x00000001
   0000 [17 8]
      Service = Winklo
      Legacy = REG_DWORD 0x00000001
      ConfigFlags = REG_DWORD 0x00000000
      Class = LegacyDriver
      ClassGUID = {8ECC055D-047F-11D1-A537-0000F8753ED1}
      DeviceDesc = Winklo
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

regdmp HKEY_LOCAL_MACHINE\SYSTEM\ControlSet002\Services\Winklo

```
HKEY_LOCAL_MACHINE\SYSTEM\ControlSet002\Services\Winklo
   Type = REG_DWORD 0x00000110
   Start = REG_DWORD 0x00000002
   ErrorControl = REG_DWORD 0x00000000
   ImagePath = REG_EXPAND_SZ C:\WINNT\System32\Winklo.exe
   DisplayName = Winklo
   ObjectName = LocalSystem
   Security [17 1]
      Security = REG_BINARY 0x000000b8 0x80140001 0x000000a0 0x000000ac \
            0x00000014 0x00000030 0x001c0002 0x00000001 0x00148002 \
            0x000f01ff 0x00000101 0x01000000 0x00000000 0x00700002 \
            0x00000004 0x00180000 0x000201fd 0x00000101 0x05000000 \
            0x00000012 0x000df910 0x001c0000 0x000f01ff 0x00000201 \
            0x05000000 0x00000020 0x00000220 0x000df970 0x00180000 \
            0x0002018d 0x00000101 0x05000000 0x0000000b 0x00000220 \
            0x001c0000 0x000201fd 0x00000201 0x05000000 0x00000020 \
            0x00000223 0x000df970 0x00000101 0x05000000 0x00000012 \
            0x00000101 0x05000000 0x00000012
```

regdmp HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_WINKLO

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\Root\LEGACY_WINKLO [17 8]
   NextInstance = REG_DWORD 0x00000001
   0000 [17 8]
      Service = Winklo
      Legacy = REG_DWORD 0x00000001
      ConfigFlags = REG_DWORD 0x00000000
      Class = LegacyDriver
      ClassGUID = {8ECC055D-047F-11D1-A537-0000F8753ED1}
      DeviceDesc = Winklo
      Control [17 8]
         ActiveService = Winklo
```

regdmp HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Winklo

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Winklo
   Type = REG_DWORD 0x00000110
   Start = REG_DWORD 0x00000002
   ErrorControl = REG_DWORD 0x00000000
   ImagePath = REG_EXPAND_SZ C:\WINNT\System32\Winklo.exe
   DisplayName = Winklo
   ObjectName = LocalSystem
   Security [17 1]
      Security = REG_BINARY 0x000000b8 0x80140001 0x000000a0 0x000000ac \
            0x00000014 0x00000030 0x001c0002 0x00000001 0x00148002 \
            0x000f01ff 0x00000101 0x01000000 0x00000000 0x00700002 \
            0x00000004 0x00180000 0x000201fd 0x00000101 0x05000000 \
            0x00000012 0x000df910 0x001c0000 0x000f01ff 0x00000201 \
            0x05000000 0x00000020 0x00000220 0x000df970 0x00180000 \
            0x0002018d 0x00000101 0x05000000 0x0000000b 0x00000220 \
            0x001c0000 0x000201fd 0x00000201 0x05000000 0x00000020 \
            0x00000223 0x000df970 0x00000101 0x05000000 0x00000012 \
            0x00000101 0x05000000 0x00000012
   Enum
      0 = Root\LEGACY_WINKLO\0000
      Count = REG_DWORD 0x00000001
      NextInstance = REG_DWORD 0x00000001
```

Regdmp HKEY_USERS\S-1-5-21-1614895754-1993962763-1060284298-500\Software\Microsoft\Windows\CurrentVersion\Applets\Regedit

```
HKEY_USERS\S-1-5-21-1614895754-1993962763-1060284298-500\Software\Microsoft\Wind
ows\CurrentVersion\Applets\Regedit
   View = REG_BINARY 0x00000040 0x0000002c 0x00000000 0x00000001 0xffff8300 \
         0xffff8300 0xffffffff 0xffffffff 0x000000e3 0x0000001d 0x000004fd \
         0x0000030a 0x00000164 0x00000078 0x00000078 0x00000120 0x00000001
   FindFlags = REG_DWORD 0x0000000e
   LastKey = My \
            Computer\HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\Winklo
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Appendix 6: Detail of TCP communication captured by Snort

THE LOG FILES DEPICT KLEZ-A AS NOTEBOOK-A AND KLEZ-B AS NOTEBOOK-B

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
FIRST STEP OF TCP HANDSHAKE (SYN)
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:21.515305 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x3E
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62036 IpLen:20 DgmLen:48 DF
******S* Seq: 0x3F476BEF  Ack: 0x0  Win: 0xFAF0  TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
SECOND STEP OF TCP HANDSHAKE (SYN-ACK)
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:21.515357 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x3E
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59745 IpLen:20 DgmLen:48 DF
***A**S* Seq: 0x7F762A0C  Ack: 0x3F476BF0  Win: 0xFF00  TcpLen: 28
TCP Options (4) => MSS: 1360 NOP NOP SackOK

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
THIRD STEP OF TCP HANDSHAKE (ACK) AND PSH
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:21.515588 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x7E
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62038 IpLen:20 DgmLen:112 DF
***AP*** Seq: 0x3F476BF0  Ack: 0x7F762A0D  Win: 0xFF00  TcpLen: 20
81 00 00 44 20 45 4C 45 4D 45 46 46 4B 43 4E 45  ...D ELEMEFFKCNE
42 43 41 43 41 43 41 43 41 43 41 43 41 43 41 43  BCACACACACACACAC
41 43 41 43 41 00 20 45 4C 45 4D 45 46 46 4B 43  ACACA. ELEMEFFKC
4E 45 43 43 41 43 41 43 41 43 41 43 41 43 41 43  NECCACACACACACAC
41 43 41 43 41 41 41 00                          ACACAAA.

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
WINDOWS LAN MANAGER COMMUNICATION
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:21.518722 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0xBF
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62040 IpLen:20 DgmLen:177 DF
***AP*** Seq: 0x3F476C38  Ack: 0x7F762A11  Win: 0xFEFC  TcpLen: 20
00 00 00 85 FF 53 4D 42 72 00 00 00 00 18 53 C8  .....SMBr.....S.
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
00 00 00 00 00 62 00 02 50 43 20 4E 45 54 57 4F  .....b..PC NETWO
52 4B 20 50 52 4F 47 52 41 4D 20 31 2E 30 00 02  RK PROGRAM 1.0..
4C 41 4E 4D 41 4E 31 2E 30 00 02 57 69 6E 64 6F  LANMAN1.0..Windo
77 73 20 66 6F 72 20 57 6F 72 6B 67 72 6F 75 70  ws for Workgroup
73 20 33 2E 31 61 00 02 4C 4D 31 2E 32 58 30 30  s 3.1a..LM1.2X00
32 00 02 4C 41 4E 4D 41 4E 32 2E 31 00 02 4E 54  2..LANMAN2.1..NT
20 4C 4D 20 30 2E 31 32 00                        LM 0.12.

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
IPC$ REMOTE ADMINISTRATION ATTEMPT
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:21.721363 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x88
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62043 IpLen:20 DgmLen:122 DF
***AP*** Seq: 0x3F476E5B  Ack: 0x7F762BDC  Win: 0xFD31  TcpLen: 20
00 00 00 4E FF 53 4D 42 75 00 00 00 00 18 07 C8  ...N.SMBu.......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
00 08 C0 F9 04 FF 00 4E 00 08 00 01 00 23 00 00  .......N.....#..
5C 00 5C 00 4B 00 4C 00 45 00 5A 00 2D 00 41 00  \.\.K.L.E.Z.-.A.
5C 00 49 00 50 00 43 00 24 00 00 00 3F 3F 3F 3F  \.I.P.C.$...????
3F 00                                            ?.
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
RELEASE OF ALL FILE AND PRINTER SHARES
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:21.788044 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x2B6
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59759 IpLen:20 DgmLen:680 DF
***AP*** Seq: 0x7F762F9F  Ack: 0x3F47734B  Win: 0xFCF6  TcpLen: 20
00 00 02 7C FF 53 4D 42 25 00 00 00 00 98 07 C8  ...|.SMB%.......
00 00 00 00 00 00 00 00 00 00 00 00 01 08 7C 04  .............|.
01 08 40 FA 0A 00 00 44 02 00 00 00 00 38 00 00  ..@....D.....8..
00 44 02 38 00 00 00 00 00 45 02 00 05 00 02 03  .D.8.....E......
10 00 00 00 44 02 00 00 01 00 00 00 2C 02 00 00  ....D.......,...
00 00 00 00 01 00 00 00 01 00 00 00 38 C3 0E 00  ............8...
08 00 00 00 48 D1 11 00 08 00 00 00 28 F1 11 00  ....H.......(...
03 00 00 80 32 F1 11 00 1C F1 11 00 00 00 00 00  ....2...........
26 F1 11 00 EE F0 11 00 00 00 00 00 FC F0 11 00  &...............
E8 F0 11 00 00 00 00 00 EC F0 11 00 D4 F0 11 00  ................
00 00 00 00 E6 F0 11 00 AC F0 11 00 00 00 00 80  ................
BA F0 11 00 8A F0 11 00 00 00 00 80 90 F0 11 00  ................
78 F0 11 00 00 00 00 00 88 F0 11 00 05 00 00 00  x...............
00 00 00 00 05 00 00 00 49 00 50 00 43 00 24 00  ........I.P.C.$.
00 00 00 00 0B 00 00 00 00 00 00 00 0B 00 00 00  ................
52 00 65 00 6D 00 6F 00 74 00 65 00 20 00 49 00  R.e.m.o.t.e. .I.
50 00 43 00 00 00 00 00 05 00 00 00 00 00 00 00  P.C.............
05 00 00 00 54 00 54 00 43 00 31 00 00 00 00 00  ....T.T.C.1.....
01 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00  ................
07 00 00 00 00 00 00 00 07 00 00 00 70 00 72 00  ............p.r.
69 00 6E 00 74 00 24 00 00 00 00 00 10 00 00 00  i.n.t.$.........
00 00 00 00 10 00 00 00 50 00 72 00 69 00 6E 00  ........P.r.i.n.
74 00 65 00 72 00 20 00 44 00 72 00 69 00 76 00  t.e.r. .D.r.i.v.
65 00 72 00 73 00 00 00 02 00 00 00 00 00 00 00  e.r.s...........
02 00 00 00 44 00 00 00 01 00 00 00 00 00 00 00  ....D...........
01 00 00 00 00 00 00 00 09 00 00 00 00 00 00 00  ................
09 00 00 00 41 00 63 00 63 00 65 00 6C 00 72 00  ....A.c.c.e.l.r.
79 00 73 00 00 00 00 00 01 00 00 00 00 00 00 00  y.s.............
01 00 00 00 00 00 00 00 07 00 00 00 00 00 00 00  ................
07 00 00 00 41 00 44 00 4D 00 49 00 4E 00 24 00  ....A.D.M.I.N.$.
00 00 00 00 0D 00 00 00 00 00 00 00 0D 00 00 00  ................
52 00 65 00 6D 00 6F 00 74 00 65 00 20 00 41 00  R.e.m.o.t.e. .A.
64 00 6D 00 69 00 6E 00 00 00 00 00 03 00 00 00  d.m.i.n.........
00 00 00 00 03 00 00 00 43 00 24 00 00 00 00 00  ........C.$.....
0E 00 00 00 00 00 00 00 0E 00 00 00 44 00 65 00  ............D.e.
66 00 61 00 75 00 6C 00 74 00 20 00 73 00 68 00  f.a.u.l.t. .s.h.
61 00 72 00 65 00 00 00 08 00 00 00 00 00 00 00  a.r.e...........
08 00 00 00 64 00 75 00 6D 00 70 00 73 00 65 00  ....d.u.m.p.s.e.
63 00 00 00 01 00 00 00 00 00 00 00 01 00 00 00  c...............
00 00 00 00 08 00 00 00 00 00 00 00 00 00 00 00  ................

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
REQUEST TO ESTABLISH ROOT SHARE OF NOTEBOOK-A “C-DRIVE"
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:21.805406 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x14A
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62055 IpLen:20 DgmLen:316 DF
***AP*** Seq: 0x3F477480  Ack: 0x7F763351  Win: 0xFB4E  TcpLen: 20
00 00 01 10 FF 53 4D 42 25 00 00 00 00 18 07 C8  .....SMB%.......
00 00 00 00 00 00 00 00 00 00 00 00 01 08 7C 04  .............|.
01 08 80 FA 10 00 00 BC 00 00 00 00 04 00 00 00  ................
00 00 00 00 00 00 00 00 00 54 00 BC 00 54 00 02  .........T...T..
00 26 00 01 40 CD 00 05 5C 00 50 00 49 00 50 00  .&..@...\.P.I.P.
45 00 5C 00 00 00 00 00 05 00 00 03 10 00 00 00  E.\.............
BC 00 00 00 01 00 00 00 A4 00 00 00 00 00 0E 00  ................
D0 EF E2 00 09 00 00 00 00 00 00 00 09 00 00 00  ................
5C 00 5C 00 4B 00 4C 00 45 00 5A 00 2D 00 41 00  \.\.K.L.E.Z.-.A.
00 00 C9 11 02 00 00 00 02 00 00 00 78 F0 E2 00  ............x...
68 EF E2 00 00 00 00 00 68 EF E2 00 7F 00 00 00  h.......h.......
FF FF FF FF 00 00 00 00 10 F0 E2 00 00 00 00 00  ................
08 00 00 00 00 00 00 00 08 00 00 00 43 00 4B 00  ............C.K.
4C 00 45 00 5A 00 2D 00 42 00 00 00 08 00 00 00  L.E.Z.-.B.......
00 00 00 00 08 00 00 00 43 00 4B 00 4C 00 45 00  ........C.K.L.E.
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
5A 00 2D 00 42 00 00 00 04 00 00 00 00 00 00 00  Z.-.B...........
04 00 00 00 43 00 3A 00 5C 00 00 00 AC F0 E2 00  ....C.:.\.......
00 00 00 00                                      ....
```

=+

REQUEST TO ESTABLISH ROOT SHARE OF NOTEBOOK-A "D-DRIVE"

=+

```
03/07-16:25:21.833450 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x14A
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62059 IpLen:20 DgmLen:316 DF
***AP*** Seq: 0x3F4776C9  Ack: 0x7F7634E3  Win: 0xF9BC  TcpLen: 20
00 00 01 10 FF 53 4D 42 25 00 00 00 00 18 07 C8  .....SMB%.......
00 00 00 00 00 00 00 00 00 00 00 00 01 08 7C 04  ..............|.
01 08 C0 FA 10 00 00 BC 00 00 00 00 04 00 00 00  ................
00 00 00 00 00 00 00 00 00 54 00 BC 00 54 00 02  .........T...T..
00 26 00 02 40 CD 00 00 5C 00 50 00 49 00 50 00  .&..@...\.P.I.P.
45 00 5C 00 00 00 00 00 05 00 00 03 10 00 00 00  E.\.............
BC 00 00 00 01 00 00 00 A4 00 00 00 00 00 0E 00  ................
D0 EF E2 00 09 00 00 00 00 00 00 00 09 00 00 00  ................
5C 00 5C 00 4B 00 4C 00 45 00 5A 00 2D 00 41 00  \.\.K.L.E.Z.-.A.
00 00 C9 11 02 00 00 00 02 00 00 00 78 F0 E2 00  ............x...
68 EF E2 00 00 00 00 00 68 EF E2 00 7F 00 00 00  h.......h.......
FF FF FF FF 00 00 00 00 10 F0 E2 00 00 00 00 00  ................
08 00 00 00 00 00 00 00 08 00 00 00 44 00 4B 00  ............D.K.
4C 00 45 00 5A 00 2D 00 42 00 00 00 08 00 00 00  L.E.Z.-.B.......
00 00 00 00 08 00 00 00 44 00 4B 00 4C 00 45 00  ........D.K.L.E.
5A 00 2D 00 42 00 00 00 04 00 00 00 00 00 00 00  Z.-.B...........
04 00 00 00 44 00 3A 00 5C 00 00 00 AC F0 E2 00  ....D.:.\.......
00 00 00 00                                      ....
```

=+

REQUEST TO ESTABLISH ROOT SHARE OF NOTEBOOK-A "E-DRIVE"

=+

```
03/07-16:25:21.850703 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x14A
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62063 IpLen:20 DgmLen:316 DF
***AP*** Seq: 0x3F477912  Ack: 0x7F763675  Win: 0xFDCE  TcpLen: 20
00 00 01 10 FF 53 4D 42 25 00 00 00 00 18 07 C8  .....SMB%.......
00 00 00 00 00 00 00 00 00 00 00 00 01 08 7C 04  ..............|.
01 08 00 FB 10 00 00 BC 00 00 00 00 04 00 00 00  ................
00 00 00 00 00 00 00 00 00 54 00 BC 00 54 00 02  .........T...T..
00 26 00 03 40 CD 00 00 5C 00 50 00 49 00 50 00  .&..@...\.P.I.P.
45 00 5C 00 00 00 00 00 05 00 00 03 10 00 00 00  E.\.............
BC 00 00 00 01 00 00 00 A4 00 00 00 00 00 0E 00  ................
D0 EF E2 00 09 00 00 00 00 00 00 00 09 00 00 00  ................
5C 00 5C 00 4B 00 4C 00 45 00 5A 00 2D 00 41 00  \.\.K.L.E.Z.-.A.
00 00 C9 11 02 00 00 00 02 00 00 00 78 F0 E2 00  ............x...
68 EF E2 00 00 00 00 00 68 EF E2 00 7F 00 00 00  h.......h.......
FF FF FF FF 00 00 00 00 10 F0 E2 00 00 00 00 00  ................
08 00 00 00 00 00 00 00 08 00 00 00 45 00 4B 00  ............E.K.
4C 00 45 00 5A 00 2D 00 42 00 00 00 08 00 00 00  L.E.Z.-.B.......
00 00 00 00 08 00 00 00 45 00 4B 00 4C 00 45 00  ........E.K.L.E.
5A 00 2D 00 42 00 00 00 04 00 00 00 00 00 00 00  Z.-.B...........
04 00 00 00 45 00 3A 00 5C 00 00 00 AC F0 E2 00  ....E.:.\.......
00 00 00 00                                      ....
```

=+

START ROUTINE OF LOCATING AND INFECTING SHARES (TTC1)

=+

```
03/07-16:25:21.883573 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x88
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62066 IpLen:20 DgmLen:122 DF
***AP*** Seq: 0x3F477AA5  Ack: 0x7F763738  Win: 0xFD0B  TcpLen: 20
00 00 00 4E FF 53 4D 42 75 00 00 00 00 18 07 C8  ...N.SMBu.......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 30 FB 04 FF 00 4E 00 08 00 01 00 23 00 00  ..0....N.....#..
5C 00 5C 00 4B 00 4C 00 45 00 5A 00 2D 00 41 00  \.\.K.L.E.Z.-.A.
5C 00 54 00 54 00 43 00 31 00 00 00 3F 3F 3F 3F  \.T.T.C.1...????
3F 00                                            ?.
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:21.883601 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59774 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F763738  Ack: 0x3F477AF7  Win: 0xFAD2  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 30 FB 00 00 00                             ..0....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:21.883622 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59774 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F763738  Ack: 0x3F477AF7  Win: 0xFAD2  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 30 FB 00 00 00                             ..0....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:21.884011 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x88
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62067 IpLen:20 DgmLen:122 DF
***AP*** Seq: 0x3F477AF7  Ack: 0x7F76375F  Win: 0xFCE4  TcpLen: 20
00 00 00 4E FF 53 4D 42 75 00 00 00 00 18 07 C8  ...N.SMBu.......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 40 FB 04 FF 00 4E 00 08 00 01 00 23 00 00  ..@....N.....#..
5C 00 5C 00 4B 00 4C 00 45 00 5A 00 2D 00 41 00  \.\.K.L.E.Z.-.A.
5C 00 54 00 54 00 43 00 31 00 00 00 3F 3F 3F 3F  \.T.T.C.1...????
3F 00                                            ?.

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:21.884037 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59775 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F76375F  Ack: 0x3F477B49  Win: 0xFA80  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 40 FB 00 00 00                             ..@....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:21.884058 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59775 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F76375F  Ack: 0x3F477B49  Win: 0xFA80  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 40 FB 00 00 00                             ..@....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:21.884419 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x88
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62068 IpLen:20 DgmLen:122 DF
***AP*** Seq: 0x3F477B49  Ack: 0x7F763786  Win: 0xFCBD  TcpLen: 20
00 00 00 4E FF 53 4D 42 75 00 00 00 00 18 07 C8  ...N.SMBu.......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 50 FB 04 FF 00 4E 00 08 00 01 00 23 00 00  ..P....N.....#..
5C 00 5C 00 4B 00 4C 00 45 00 5A 00 2D 00 41 00  \.\.K.L.E.Z.-.A.
5C 00 54 00 54 00 43 00 31 00 00 00 3F 3F 3F 3F  \.T.T.C.1...????
3F 00                                            ?.

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:21.884503 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59776 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F763786  Ack: 0x3F477B9B  Win: 0xFA2E  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 50 FB 00 00 00                             ..P....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
03/07-16:25:21.884527 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59776 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F763786  Ack: 0x3F477B9B  Win: 0xFA2E  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 50 FB 00 00 00                             ..P....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:21.906010 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x88
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62069 IpLen:20 DgmLen:122 DF
***AP*** Seq: 0x3F477B9B  Ack: 0x7F7637AD  Win: 0xFC96  TcpLen: 20
00 00 00 4E FF 53 4D 42 75 00 00 00 00 18 07 C8  ...N.SMBu.......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 60 FB 04 FF 00 4E 00 08 00 01 00 23 00 00  ..`....N.....#..
5C 00 5C 00 4B 00 4C 00 45 00 5A 00 2D 00 41 00  \.\.K.L.E.Z.-.A.
5C 00 54 00 54 00 43 00 31 00 00 00 3F 3F 3F 3F  \.T.T.C.1...????
3F 00                                            ?.

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:21.911740 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59777 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F7637AD  Ack: 0x3F477BED  Win: 0xF9DC  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 60 FB 00 00 00                             ..`....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:21.911814 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59777 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F7637AD  Ack: 0x3F477BED  Win: 0xF9DC  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 60 FB 00 00 00                             ..`....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:21.955935 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x88
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62070 IpLen:20 DgmLen:122 DF
***AP*** Seq: 0x3F477BED  Ack: 0x7F7637D4  Win: 0xFC6F  TcpLen: 20
00 00 00 4E FF 53 4D 42 75 00 00 00 00 18 07 C8  ...N.SMBu.......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 70 FB 04 FF 00 4E 00 08 00 01 00 23 00 00  ..p....N.....#..
5C 00 5C 00 4B 00 4C 00 45 00 5A 00 2D 00 41 00  \.\.K.L.E.Z.-.A.
5C 00 54 00 54 00 43 00 31 00 00 00 3F 3F 3F 3F  \.T.T.C.1...????
3F 00                                            ?.

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:21.956079 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59778 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F7637D4  Ack: 0x3F477C3F  Win: 0xFF00  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 70 FB 00 00 00                             ..p....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:21.956124 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59778 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F7637D4  Ack: 0x3F477C3F  Win: 0xFF00  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 70 FB 00 00 00                             ..p....
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

REPEAT ROUTINE TO LOCATE AND INFECT SHARES (ACCELRYS)

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:23.153093 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x90
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62101 IpLen:20 DgmLen:130 DF
***AP*** Seq: 0x3F478B71  Ack: 0x7F76422A  Win: 0xFCE7  TcpLen: 20
00 00 00 56 FF 53 4D 42 75 00 00 00 00 18 07 C8  ...V.SMBu.......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 50 FD 04 FF 00 56 00 08 00 01 00 2B 00 00  ..P....V.....+..
5C 00 5C 00 4B 00 4C 00 45 00 5A 00 2D 00 41 00  \.\.K.L.E.Z.-.A.
5C 00 41 00 43 00 43 00 45 00 4C 00 52 00 59 00  \.A.C.C.E.L.R.Y.
53 00 00 00 3F 3F 3F 3F 3F 00                    S...?????.

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:23.153765 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59808 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F76422A  Ack: 0x3F478BCB  Win: 0xFA7C  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 50 FD 00 00 00                             ..P....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:23.153818 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59808 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F76422A  Ack: 0x3F478BCB  Win: 0xFA7C  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 50 FD 00 00 00                             ..P....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:23.261999 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x90
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62102 IpLen:20 DgmLen:130 DF
***AP*** Seq: 0x3F478BCB  Ack: 0x7F764251  Win: 0xFCC0  TcpLen: 20
00 00 00 56 FF 53 4D 42 75 00 00 00 00 18 07 C8  ...V.SMBu.......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 60 FD 04 FF 00 56 00 08 00 01 00 2B 00 00  ..`....V.....+..
5C 00 5C 00 4B 00 4C 00 45 00 5A 00 2D 00 41 00  \.\.K.L.E.Z.-.A.
5C 00 41 00 43 00 43 00 45 00 4C 00 52 00 59 00  \.A.C.C.E.L.R.Y.
53 00 00 00 3F 3F 3F 3F 3F 00                    S...?????.

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:23.272098 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59809 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F764251  Ack: 0x3F478C25  Win: 0xFA22  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 60 FD 00 00 00                             ..`....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:23.272152 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59809 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F764251  Ack: 0x3F478C25  Win: 0xFA22  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 60 FD 00 00 00                             ..`....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:23.359312 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x90
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62103 IpLen:20 DgmLen:130 DF
***AP*** Seq: 0x3F478C25  Ack: 0x7F764278  Win: 0xFC99  TcpLen: 20
00 00 00 56 FF 53 4D 42 75 00 00 00 00 18 07 C8  ...V.SMBu.......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 70 FD 04 FF 00 56 00 08 00 01 00 2B 00 00  ..p....V.....+..
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
5C 00 5C 00 4B 00 4C 00 45 00 5A 00 2D 00 41 00  \.\.K.L.E.Z.-.A.
5C 00 41 00 43 00 43 00 45 00 4C 00 52 00 59 00  \.A.C.C.E.L.R.Y.
53 00 00 00 3F 3F 3F 3F 3F 00                    S...?????.

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:23.360566 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59810 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F764278  Ack: 0x3F478C7F  Win: 0xF9C8  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 70 FD 00 00 00                             ..p....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:23.360614 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59810 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F764278  Ack: 0x3F478C7F  Win: 0xF9C8  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 70 FD 00 00 00                             ..p....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:23.451966 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x90
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62104 IpLen:20 DgmLen:130 DF
***AP*** Seq: 0x3F478C7F  Ack: 0x7F76429F  Win: 0xFC72  TcpLen: 20
00 00 00 56 FF 53 4D 42 75 00 00 00 00 18 07 C8  ...V.SMBu.......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 80 FD 04 FF 00 56 00 08 00 01 00 2B 00 00  .......V.....+..
5C 00 5C 00 4B 00 4C 00 45 00 5A 00 2D 00 41 00  \.\.K.L.E.Z.-.A.
5C 00 41 00 43 00 43 00 45 00 4C 00 52 00 59 00  \.A.C.C.E.L.R.Y.
53 00 00 00 3F 3F 3F 3F 3F 00                    S...?????.

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:23.454212 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59811 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F76429F  Ack: 0x3F478CD9  Win: 0xFF00  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 80 FD 00 00 00                             .......

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:23.454258 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59811 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F76429F  Ack: 0x3F478CD9  Win: 0xFF00  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 80 FD 00 00 00                             .......

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:23.557959 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x90
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62105 IpLen:20 DgmLen:130 DF
***AP*** Seq: 0x3F478CD9  Ack: 0x7F7642C6  Win: 0xFC4B  TcpLen: 20
00 00 00 56 FF 53 4D 42 75 00 00 00 00 18 07 C8  ...V.SMBu.......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 90 FD 04 FF 00 56 00 08 00 01 00 2B 00 00  .......V.....+..
5C 00 5C 00 4B 00 4C 00 45 00 5A 00 2D 00 41 00  \.\.K.L.E.Z.-.A.
5C 00 41 00 43 00 43 00 45 00 4C 00 52 00 59 00  \.A.C.C.E.L.R.Y.
53 00 00 00 3F 3F 3F 3F 3F 00                    S...?????.

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:23.592631 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59812 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F7642C6  Ack: 0x3F478D33  Win: 0xFEA6  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 90 FD 00 00 00                             .......

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:23.592688 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59812 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F7642C6  Ack: 0x3F478D33  Win: 0xFEA6  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 90 FD 00 00 00                             .......

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

REPEAT ROUTINE TO LOCATE AND INFECT SHARES (DUMPSEC)

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:24.304417 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x8E
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62119 IpLen:20 DgmLen:128 DF
***AP*** Seq: 0x3F47940E  Ack: 0x7F7647A3  Win: 0xFCE7  TcpLen: 20
00 00 00 54 FF 53 4D 42 75 00 00 00 00 18 07 C8  ...T.SMBu.......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 60 FE 04 FF 00 54 00 08 00 01 00 29 00 00  ..`....T.....)..
5C 00 5C 00 4B 00 4C 00 45 00 5A 00 2D 00 41 00  \.\.K.L.E.Z.-.A.
5C 00 44 00 55 00 4D 00 50 00 53 00 45 00 43 00  \.D.U.M.P.S.E.C.
00 00 3F 3F 3F 3F 3F 00                          ..?????.

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:24.316778 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59825 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F7647A3  Ack: 0x3F479466  Win: 0xFCC7  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 60 FE 00 00 00                             ..`....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:24.316832 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59825 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F7647A3  Ack: 0x3F479466  Win: 0xFCC7  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 60 FE 00 00 00                             ..`....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:24.474784 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x8E
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62120 IpLen:20 DgmLen:128 DF
***AP*** Seq: 0x3F479466  Ack: 0x7F7647CA  Win: 0xFCC0  TcpLen: 20
00 00 00 54 FF 53 4D 42 75 00 00 00 00 18 07 C8  ...T.SMBu.......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 70 FE 04 FF 00 54 00 08 00 01 00 29 00 00  ..p....T.....)..
5C 00 5C 00 4B 00 4C 00 45 00 5A 00 2D 00 41 00  \.\.K.L.E.Z.-.A.
5C 00 44 00 55 00 4D 00 50 00 53 00 45 00 43 00  \.D.U.M.P.S.E.C.
00 00 3F 3F 3F 3F 3F 00                          ..?????.

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:24.484213 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59826 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F7647CA  Ack: 0x3F4794BE  Win: 0xFC6F  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 70 FE 00 00 00                             ..p....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:24.484261 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59826 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F7647CA  Ack: 0x3F4794BE  Win: 0xFC6F  TcpLen: 20
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 70 FE 00 00 00                             ..p....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:24.603693 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x3C
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62121 IpLen:20 DgmLen:40 DF
***A**** Seq: 0x3F4794BE  Ack: 0x7F7647F1  Win: 0xFC99  TcpLen: 20

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:24.614159 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x8E
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62122 IpLen:20 DgmLen:128 DF
***AP*** Seq: 0x3F4794BE  Ack: 0x7F7647F1  Win: 0xFC99  TcpLen: 20
00 00 00 54 FF 53 4D 42 75 00 00 00 00 18 07 C8  ...T.SMBu.......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 80 FE 04 FF 00 54 00 08 00 01 00 29 00 00  .......T.....)..
5C 00 5C 00 4B 00 4C 00 45 00 5A 00 2D 00 41 00  \.\.K.L.E.Z.-.A.
5C 00 44 00 55 00 4D 00 50 00 53 00 45 00 43 00  \.D.U.M.P.S.E.C.
00 00 3F 3F 3F 3F 3F 00                          ..?????.

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:24.632005 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59827 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F7647F1  Ack: 0x3F479516  Win: 0xFC17  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 80 FE 00 00 00                             .......

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:24.632060 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59827 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F7647F1  Ack: 0x3F479516  Win: 0xFC17  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 80 FE 00 00 00                             .......

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:24.651223 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x8E
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62123 IpLen:20 DgmLen:128 DF
***AP*** Seq: 0x3F479516  Ack: 0x7F764818  Win: 0xFC72  TcpLen: 20
00 00 00 54 FF 53 4D 42 75 00 00 00 00 18 07 C8  ...T.SMBu.......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 90 FE 04 FF 00 54 00 08 00 01 00 29 00 00  .......T.....)..
5C 00 5C 00 4B 00 4C 00 45 00 5A 00 2D 00 41 00  \.\.K.L.E.Z.-.A.
5C 00 44 00 55 00 4D 00 50 00 53 00 45 00 43 00  \.D.U.M.P.S.E.C.
00 00 3F 3F 3F 3F 3F 00                          ..?????.

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:24.651720 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59828 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F764818  Ack: 0x3F47956E  Win: 0xFBBF  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 90 FE 00 00 00                             .......

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:24.651768 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59828 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F764818  Ack: 0x3F47956E  Win: 0xFBBF  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 90 FE 00 00 00                             .......
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:24.656557 0:20:E0:67:35:AE -> 0:4:76:42:FD:55 type:0x800 len:0x8E
192.168.1.112:1070 -> 192.168.1.100:139 TCP TTL:128 TOS:0x0 ID:62124 IpLen:20 DgmLen:128 DF
***AP*** Seq: 0x3F47956E  Ack: 0x7F76483F  Win: 0xFC4B  TcpLen: 20
00 00 00 54 FF 53 4D 42 75 00 00 00 00 18 07 C8  ...T.SMBu.......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 A0 FE 04 FF 00 54 00 08 00 01 00 29 00 00  .......T.....)..
5C 00 5C 00 4B 00 4C 00 45 00 5A 00 2D 00 41 00  \.\.K.L.E.Z.-.A.
5C 00 44 00 55 00 4D 00 50 00 53 00 45 00 43 00  \.D.U.M.P.S.E.C.
00 00 3F 3F 3F 3F 3F 00                          ..?????.

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:24.656616 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59829 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F76483F  Ack: 0x3F4795C6  Win: 0xFB67  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 A0 FE 00 00 00                             .......

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

03/07-16:25:24.656646 0:4:76:42:FD:55 -> 0:20:E0:67:35:AE type:0x800 len:0x5D
192.168.1.100:139 -> 192.168.1.112:1070 TCP TTL:128 TOS:0x0 ID:59829 IpLen:20 DgmLen:79 DF
***AP*** Seq: 0x7F76483F  Ack: 0x3F4795C6  Win: 0xFB67  TcpLen: 20
00 00 00 23 FF 53 4D 42 75 22 00 00 C0 98 07 C8  ...#.SMBu"......
00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FE  ................
01 08 A0 FE 00 00 00                             .......
```

© SANS Institute 2003, Author retains full rights.