NCCIC NATIONAL CYBERSECURITY AND COMMUNICATIONS INTEGRATION CENTER | US-CERT UNITED STATES COMPUTER EMERGENCY READINESS TEAM

Malware Analysis Report (MAR) - 10132963

2017-08-14

Notification

This report is provided "as is" for informational purposes only. The Department of Homeland Security (DHS) does not provide any warranties of any kind regarding any information contained within. The DHS does not endorse any commercial product or service, referenced in this bulletin or otherwise.

This document is marked TLP:WHITE. Disclosure is not limited. Sources may use TLP:WHITE when information carries minimal or no foreseeable risk of misuse, in accordance with applicable rules and procedures for public release. Subject to standard copyright rules, TLP:WHITE information may be distributed without restriction. For more information on the Traffic Light Protocol, see http://www.us-cert.gov/tlp/.

Summary

Description

US-CERT received three files associated with the DeltaCharlie attack malware. The files are designed to conduct three types of attacks, NTP_Attack, DNS_Attack, and CGN_Attack. The files also establish backdoor command-and-control capability on the victim system.

Files

Processed	3 584ac94142f0b7c0df3d0adde6e661ed (mimefilter.xml_584AC94142F0B7C0DF3D0ADDE6E661ED) 5d29dfe2ea9ca8da3ff7a14fb20c5e86 (5D29DFE2EA9CA8DA3FF7A14FB20C5E86) 8f4fc2e10b6ec15a01e0af24529040dd (8F4FC2E10B6EC15A01E0AF24529040DD)

IPs

Identified	2 202.126.90.89 153.68.198.14

Files

5D29DFE2EA9CA8DA3FF7A14FB20C5E86

Details

Name	5D29DFE2EA9CA8DA3FF7A14FB20C5E86
Size	180224
Type	PE32 executable (GUI) Intel 80386, for MS Windows
MD5	5d29dfe2ea9ca8da3ff7a14fb20c5e86
SHA1	3fdf856b6fbcb23e7c3372a3f53ce26c0fe6de77
ssdeep	3072:9sCh49HhQS2qaWuLYyJHYnGerQJDu70cSrzdZHllbFX:9sCh4TQqaZYyJHYGen70lzdZFSZ
Entropy	6.13711245238

Antivirus

ClamAV	Win.Trojan.Agent-1388767
Kaspersky	HackTool.Win32.Agent.aesh
Microsoft Security Essentials	Backdoor:Win32/Winsec.B!dha
TrendMicro House Call	BKDR_SCADPRV.B
TrendMicro	BKDR_SCADPRV.B

PE Information

Compiled	2014-12-17T14:03:38Z

PE Sections

Name	MD5	Raw Size	Entropy
(header)	6a5356bedf23ccecac180cd887c15de8	4096	0.792314879114
.text	72d9f7da3d7eb917a18954668399ce67	77824	6.14523436219
.rdata	af59deeeff5d5f41ecdd092b80536d25	8192	3.96837828979
.data	b994d715f522732213ea03cb2013a469	12288	4.24722552284
.rsrc	219125d84f95e9ec104a49383da7b991	77824	6.31904971708

Packers

Name	Version	Entry Point
Microsoft Visual C++ v6.0	NA	NA

Relationships

(F) 5D29DFE2EA9CA8DA3FF7A14FB20C5E86 (5d29d)	Connected_To	(I) 202.126.90.89
(F) 5D29DFE2EA9CA8DA3FF7A14FB20C5E86 (5d29d)	Related_To	(S) Screenshot 1: Program Connection Log
(F) 5D29DFE2EA9CA8DA3FF7A14FB20C5E86 (5d29d)	Connected_From	(I) 153.68.198.14

Description

This file contains three embedded resources. Two of the resources are 32bit and 64bit versions of the winpcap packet driver called npf.sys. The third resource is the program's configuration file, netplg.log.

When the program is executed, it will look for any previous existence of itself by looking for the mutex '\Global\NetplugDiscovery0.7'.

The malware will then install the packet driver described above based on the operating system architecture.

If this is the first time the program has started, the program will create and install a new service called 'netplug".

---Begin Service Details---

netplug
Network Card Service
"This service monitors the network interface, turning it off or on depending on signal, used mainly for laptos that may not always be connected."

---End Service Details---

When the netplug service is executed, it will load 'netplg.log'. This resource will then be stored in %System32% and contains the hard-coded IP address, 153.68.198.14. This IP address is used to calculate the true command and control (C2) IP address by XORing the IP address with the hex string 0x579C3A53 and attempting to connect to the newly created IP address on TCP Port 443. The malware generates a log file in the current directory where activity regarding the installation of the bot and the connection are stored. This file is named the same as the malware with the <malware_name>.log. If the malware is able to connect, it will send the log file to the C2. In this analysis, the C2 was determined to be 202.126.90.89. See Screenshot 1. If no results are returned, the malware will terminate.

The malware contains an attack component that can perform the following commands:

---Begin Bot Commands---

DownExec - Downloads and executes files (calls URLDownloadToFile)
ChngBotconfig - Changes the configuration of the bot
BotUpdate - Updates the attack modules
BotDie - Terminates the bot by calling a self-deleting batch file, msvcrt.bat
[No Name] - Starts a new attack
[No Name] - Stops the attack

---End Bot Commands---

The malware is capable of conducting three different types of attacks:

---Begin Attack List---

NTP_ATTACK - Network Time Protocol attack via UDP flood
CGN_ATTACK - Carrier Grade NAT attack targeting CGN IP addresses
DNS_ATTACK - Domain Name Service attack via UDP flood

---End Attack List---

When the Network Card Service (netplug) is started, the malware will begin logging activity to the file, <malware_name>.log which is stored in the current directory. The log file records all installation and connection activity associated with the bot and is written in plaintext. The following is a sample of log file entries associated with the service startup:

---Log File Entries---

AtkNum:
TotalPackets:

Resovle DnsName Falied: --> Written if unable to resolve DNS name from configuration file
__ResolveDnsName: --> Written if resolution is successful

:Connecting...<target>:<port> --> Written during connection process
:Connected<target><port> --> Written if connection is successful
:HS Success<name><port> --> If unsuccessful the socket will be closed
:Connection Failed<target><port> --> Written if connection fails

MyMain Started --> Service is initiated
CreateService Success --> Service is successfully created
StartService Success --> Service is successfully started
CreateBotMutex: ERROR_ALREADY_EXISTS --> Mutex is successfully created
LoadConfig Failed: ERROR_ALREADY_EXISTS --> The service is already running
SERVICE_CONTROL_SHUTDOWN, error code = --> The service failed to start
SetServiceStatus failed, error code = --> The service is not configured correctly
**********Connection Fins... --> Written when the log is successfully sent to the C2

---End File Entries---

Each time the service is started, it will attempt to open and read data from the configuration file:

---Log File Entries---

ExtractPackage Failed: %d --> Written if the service fails to open the file
ExtractConfig Failed: %d --> Written if the service fails to read the file
ExtractPackage Success --> Written if the service successfully opens the file
ExtractConfig Success --> Written if the service successfully reads the file
Install and Run Success --> Written the new config installation is successful

---End File Entries---

When the service receives an attack command, the program creates a new log file called edbchk.log. This file is stored in C:\Windows \System32\catroot2\ and records all activity associated with the attack bot only. The following entries can be written to the log:

---Begin File Entries---

############### Received Attack Cmd %d#... --> Written when the attack command is received
Waiting For NTP Attack <target> ...Remain <time> --> Written when the NTP attack is staged
Waiting For NTP Fake Attack <target>...Remain <time> --> Written when the NTP fake attack is staged
############### NTP Attack Started <target> --> Written when the NTP Attack starts
############### NTP Fake Attack Started <target> --> Written when the NTP Attack starts
Reamin Time: --> Written at intervals during the attack
############### NTP Attack Ended <target> --> Written when the attack ends
############### NTP AttackTime is up --> Written if the attack fails

############### CGN Attack Started <name> --> Written when the CGN attack starts
Waiting for CGN Attack <target>...Remain <time> --> Written when the CGN attack is staged
############### CGN Attack Ended <target>### --> Written when the CGN attack ends
############### CGN AttackTime is up --> Written if the CGN attack fails

Waiting for DNS Attack <target>...Remain<time> --> Written when the DNS Attack is staged
############### DNS Attack Started <target> --> Written when the DNS Attack starts
############### DNS Attack Ended <target> ### --> Written when the DNS Attack ends
DNS AttackTime is up <time> --> Written if the DNS Attack fails

############### Received Stop Cmd <target> --> Written when the attack bot is stopped

---End File Entries---

If the bot is terminated (BotDie) the program will generate a self-deleting script called msvcrt.bat to delete itself. Msvcrt.bat contains the following data:

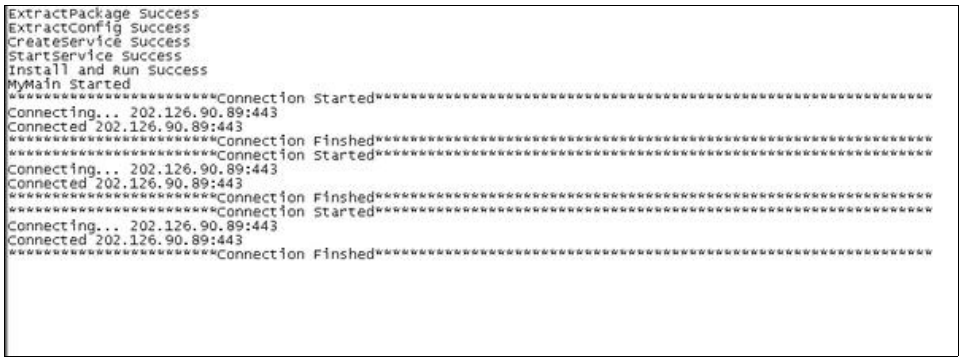
---Begin Msvcrt File---

```
@echo off
del /a %1
if exist %1 goto D1
del /a %0
%s "%s"
```

---End Msvcrt File---

Screenshots

- **Screenshot 1: Program Connection Log**



8F4FC2E10B6EC15A01E0AF24529040DD

Details

Name	8F4FC2E10B6EC15A01E0AF24529040DD
Size	53248
Type	PE32 executable (GUI) Intel 80386, for MS Windows

MD5	8f4fc2e10b6ec15a01e0af24529040dd
SHA1	b164ba5e5734c469839292ede4d5c04e76523bae
ssdeep	768:wH75DjuOD73BTzuqK6C1C+UjuoxxPDzREAY8aTk0kjo:W5pZCTUVjQpk0so
Entropy	5.20908628282

Antivirus

nProtect	Backdoor/W32.Agent.53248.LS
McAfee	RDN/Generic BackDoor
NetGate	Trojan.Win32.Malware
K7	Riskware (0040eff71)
Systweak	trojan.deltacharlie
F-secure	Trojan.GenericKD.5400227
Kaspersky	Backdoor.Win32.Agent.guhi
BitDefender	Trojan.GenericKD.5400227
Microsoft Security Essentials	Trojan:Win32/Dynamer!rfn
Sophos	Troj/DeltaC-A
TrendMicro House Call	BKDR_ESCAD.SMHA
TrendMicro	BKDR_ESCAD.SMHA
Emsisoft	Trojan.GenericKD.5400227 (B)
Avira	TR/Fuery.kevww
Ahnlab	Backdoor/Win32.Escad
ESET	a variant of Generik.DXNZOSG trojan
NANOAV	Trojan.Win32.Agent.eqhpcw
Vir.IT eXplorer	Trojan.Win32.Genus.BWG
Quick Heal	DDoS.HidenCobra.S1166387
Ikarus	Trojan.SuspectCRC
AVG	SCGeneric2.BDVR

PE Information

Compiled	2015-08-25T09:09:28Z

PE Sections

Name	MD5	Raw Size	Entropy
(header)	a4fc300b72266ccce1977f93b1bca3b5	4096	0.640698472599
.text	11eab7228491af5ac109f58055c8f94f	28672	6.07747984156
.rdata	6dd10b0e9a62a4943665e32d36c02b9f	12288	3.84897647617
.data	1bdda8ad01a81904160d4aaff5028678	8192	3.74298941886

Packers

Name	Version	Entry Point
Microsoft Visual C++ v6.0	NA	NA

Relationships

(F) 8F4FC2E10B6EC15A01E0AF24529040DD (8f4fc)	Related_To	(F) mimefilter.xml_584AC94142F0B7C0DF3D0ADDE6E661ED (584ac)

Description

This malicious file is a utility that allows an operator to push secondary payloads to the victim system. When executed from the command line with the -i argument, the program will install itself and launch a service named 'DnsQuerySvc'. During runtime, the program will bind and listen for data on TCP Port 443. An operator can connect to the compromised system and send command and control data to the victim system. The data passed is encoded using a simple XOR cipher to make it difficult to identify as C2 traffic. Of particular note, the malware does not connect to the C2 server, but instead requires the operator to connect. Therefore, no network traffic would be detected until the operator decides to connect to push new payloads or commands to the victim system.

Analysis of this application reveals it provides operator decision based command and control capabilities over a victim system. It accepts blocks of data from an operator, decodes them, and then uses eight bytes of this decoded data to determine which activity to perform on

compromised systems. These commands are displayed below:

---Begin Commands---

0x26B9A0BA - Starts a C2 session.
0x1AB0918C - This command allows the operator to replace the configuration file mimefilter.xml. The command also replaces the file extensions of the files edbres00001.jrs, edbres00002.jrs, edbres 00003.jrs and edbres00004.jrs with the following four respective file names -- .jrdb1, .jrbd2, .jrbd3 and .jrbd4. The purpose of replacing these files names and the purpose of the edbres * files are unknown as these files were not included within this submission.
0x1AB0918F - This command allows an operator to replace the file mimefilter.xml that the malware expects to be installed as C:\Windows\System32\mimefilter.xml. This file is an RC4 configuration file that contains the working update directories the malware uses.
0x1AB0918D - This command allows an operator to simultaneously exfiltrate four files at once from the victim system. The nature of these four files is not known. Importantly, exfiltrated data will also be protected via the same XOR cipher as data received by the implant.
0x1AB09190 - This command allows an operator to write a payload to the victim system's temp folder. The file name for this uploaded payload will start with 'oem'. The malware then reads this payload and processes it through an algorithm that appears to be a loader function. None of these payloads were included within this submission, but analysis indicates they will be Win32 DLLs.
0x1AB09191 - This command removes the extensions from any files which have an extension named .jrbt1, .jrbt2, .jrbt3, or .jrbt4. The command then uploads four files to the victim system that may have these extensions. This technique is likely used to ensure there are no file name conflicts between files uploaded to the victim system. It appears this command is designed to allow an operator to push out multiple payloads to their collection of compromised systems simultaneously.
0x1AB0918E - This command allows an operator to write 4 files to the victim system simultaneously. This command is similar to the command 0x1AB09191 except that it does not remove the .jrbt extensions from existing files.
0x1AB09192 - This command allows the operator to upload a file to the victim system using the Win32 API WriteFile.
0x1AB09193 - This command provides the operator with information about the victim system. It gets this information using the Win32 APIs GetComputerNameW and GetVersionExA.

---End Commands---

The program is designed to mimic the Windows Update process, in that it uses the same folders in C:\Windows\System32\catroot2\ as its primary working folders:

---Begin Catroot Folders---

C:\WINDOWS\system32\catroot2\{A750E6C3-38EE-17D5-85E5- 10D03DA378DE}
C:\WINDOWS\system32\catroot2\{12CD0A1D-4EA2-11D1-8608- 00C04FC295EF}

---End Catroot Folders---

Additional payloads are uploaded to these folders. The Windows OS also uses these folders to store updates.

It appears the malware uses this location to mask its payloads as legitimate Windows updates. The C2 structure of the malware enables an operator to easily replace this RC4 encrypted file to dynamically adjust the working directory of their implant. A listing of the folders is found in the program's configuration file, mimefilter.xml. After loading the configuration file, the malware will attempt to search the folders for all files that begin with the name 'oem'. The malware attempts to read each of these files it finds, and processes their data through a function, which appears to be a loader method. None of these oem* files were included within this submission, however analysis indicates they may be Windows DLLs.

The program can also modify settings to the firewall by invoking the netsh command.

---Begin Firewall Settings---

cmd.exe /c netsh firewall add portopening protocol=tcp port=%d name="Windows Media Player Network Sharing"
cmd.exe /c netsh advfirewall add rule name="Windows Media Player Network Sharing" dir=in action=allow Protocol=TCP localport=%d
cmd.exe /c netsh firewall delete portopening protocol=tcp port=%d
cmd.exe /c netsh advfirewall firewall delete rule name="Windows Media Player Network Sharing" Protocol=TCP localport=%d

---End Firewall Settings---

The following YARA rule can be used to detect the presence of this updater program:

---Begin YARA Rule---

```
rule Malware_Updater
{
meta:
  Author="US-CERT Code Analysis Team"
  Date="2017/08/02"
  Incident="10132963"
  MD5_1="8F4FC2E10B6EC15A01E0AF24529040DD"
```

```
  MD5_2="584AC94142F0B7C0DF3D0ADDE6E661ED"
  Info="Malware may be used to update multiple systems with secondary payloads"
  super_rule=1
strings:
  $s0 = { 8A4C040480F15D80C171884C04044083F8107CEC }
  $s1 = { 8A4D0080F19580E97C884D00454B75F0 }

 condition: any of them

}
---End YARA Rule---
```

mimefilter.xml_584AC94142F0B7C0DF3D0ADDE6E661ED

Details

Name	mimefilter.xml_584AC94142F0B7C0DF3D0ADDE6E661ED
Size	528
Type	data
MD5	584ac94142f0b7c0df3d0adde6e661ed
SHA1	1f21185303b7992d6ef54b23e816d48911496b9d
ssdeep	12:N80aKgpdlWhMwlpIh1XdPDFVxzsSCe2nI8xm062UdYoPP4jySeNTi:N80ngJKrILd1vEm062UdNPor
Entropy	7.59623010182

Antivirus

No matches found.

Relationships

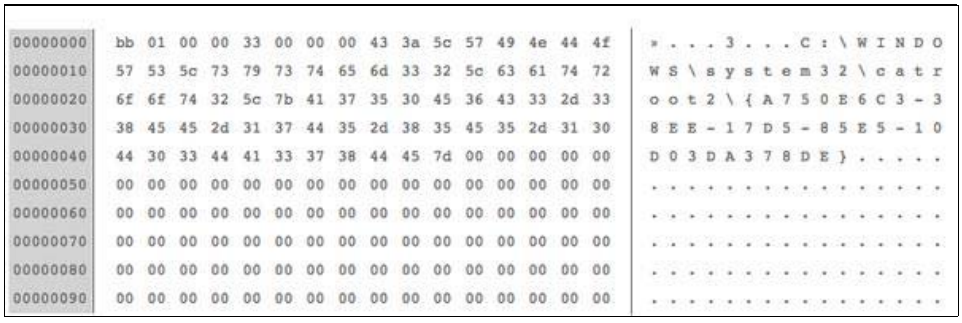
(F) mimefilter.xml_584AC94142F0B7C0DF3D0ADDE6E661ED (584ac)	Related_To	(F) 8F4FC2E10B6EC15A01E0AF24529040DD (8f4fc)
(F) mimefilter.xml_584AC94142F0B7C0DF3D0ADDE6E661ED (584ac)	Related_To	(S) Screenshot 2: Decrypted Config File

Description

mimefilter.xml is an RC4 encrypted file that contains configuration data that is read by 8F4FC2E10B6EC15A01E0AF24529040DD. The data is decrypted using the RC4 cipher and the key 'InitializeSecurityContextA'. See Screenshot 2.

Screenshots

- **Screenshot 2: Decrypted Config File**



```
00000000  bb 01 00 00 33 00 00 00 43 3a 5c 57 49 4e 44 4f  » . . . 3 . . . C : \ W I N D O
00000010  57 53 5c 73 79 73 74 65 6d 33 32 5c 63 61 74 72  W S \ s y s t e m 3 2 \ c a t r
00000020  6f 6f 74 32 5c 7b 41 37 35 30 45 36 43 33 2d 33  o o t 2 \ { A 7 5 0 E 6 C 3 - 3
00000030  38 45 45 2d 31 37 44 35 2d 38 35 45 35 2d 31 30  8 E E - 1 7 D 5 - 8 5 E 5 - 1 0
00000040  44 30 33 44 41 33 37 38 44 45 7d 00 00 00 00 00  D 0 3 D A 3 7 8 D E } . . . . .
00000050  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  . . . . . . . . . . . . . . . .
00000060  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  . . . . . . . . . . . . . . . .
00000070  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  . . . . . . . . . . . . . . . .
00000080  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  . . . . . . . . . . . . . . . .
00000090  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  . . . . . . . . . . . . . . . .
```

IPs

202.126.90.89

Ports

- 443

Whois

inetnum: 202.126.90.0 - 202.126.90.255
netname: ULUSNET

descr: ULUSNET mobile WiMax subs pool#1
country: MN
admin-c: UNT1-AP
tech-c: UNT1-AP
status: ASSIGNED NON-PORTABLE
mnt-by: MAINT-MN-ULUSNET
mnt-irt: IRT-ULUSNET-MN
changed: tuvshinbayar[@]mobicom.mn 20170727
source: APNIC

irt: IRT-ULUSNET-MN
address: MPRP building, 313, Ulaanbaatar, Mongolia
e-mail: manlai[@]ulusnet.mn
abuse-mailbox: manlai[@]ulusnet.mn
admin-c: NT331-AP
tech-c: NT331-AP
auth: # Filtered
mnt-by: MAINT-MN-ULUSNET
changed: manlai[@]ulusnet.mn 20110329
source: APNIC

role: Ulusnet Network Team
address: Sambuu street - 47, Post office-38, Chingeltei district, Ulaanbaatar - 15171, Mongolia
country: MN
phone: +976-75759944
e-mail: peering[@]mobicom.mn
admin-c: NT331-AP
tech-c: NT331-AP
nic-hdl: UNT1-AP
mnt-by: MAINT-MN-ULUSNET
changed: tuvshinbayar[@]mobicom.mn 20170727
source: APNIC

% Information related to '202.126.90.0/24AS38218'

route: 202.126.90.0/24
descr: MN-MONGOLIA-ULUSNET
origin: AS38218
mnt-by: MAINT-MN-ULUSNET
changed: manlai[@]ulusnet.mn 20090418
source: APNIC

Relationships

(I) 202.126.90.89	Related_To	(P) 443
(I) 202.126.90.89	Characterized_By	(W) inetnum: 202.
(I) 202.126.90.89	Connected_From	(F) 5D29DFE2EA9CA8DA3FF7A14FB20C5E86 (5d29d)

Description

202.126.90.89 is the command-and-control IP that is decoded by 5D29DFE2EA9CA8DA3FF7A14FB20C5E86 using the XOR string 0x579C3A53.

153.68.198.14

Whois

Queried whois.arin.net with "n 153.68.198.14"...

NetRange: 153.66.0.0 - 153.87.255.255
CIDR: 153.80.0.0/13, 153.72.0.0/13, 153.68.0.0/14, 153.66.0.0/15
NetName: NCRWIN17
NetHandle: NET-153-66-0-0-1
Parent: APNIC-ERX-153 (NET-153-0-0-0-0)
NetType: Direct Allocation
OriginAS:
Organization: NCR Corporation (NCR)
RegDate: 1991-09-23
Updated: 2014-01-08

Ref: https[:]//whois.arin.net/rest/net/NET-153-66-0-0-1

OrgName: NCR Corporation
OrgId: NCR
Address: GNCS - WHQ
Address: 3097 Satellite Blvd.
City: Duluth
StateProv: GA
PostalCode: 30096
Country: US
RegDate: 1989-03-29
Updated: 2012-09-11
Ref: https[:]//whois.arin.net/rest/org/NCR

OrgTechHandle: CGH3-ARIN
OrgTechName: Haug, Chris Gordon
OrgTechPhone: +1-905-819-4168
OrgTechEmail: ch134537[@]ncr.com
OrgTechRef: https[:]//whois.arin.net/rest/poc/CGH3-ARIN

OrgAbuseHandle: CGH3-ARIN
OrgAbuseName: Haug, Chris Gordon
OrgAbusePhone: +1-905-819-4168
OrgAbuseEmail: ch134537[@]ncr.com
OrgAbuseRef: https[:]//whois.arin.net/rest/poc/CGH3-ARIN

OrgTechHandle: SPEAR14-ARIN
OrgTechName: spear, Bryan
OrgTechPhone: +1-770-689-2237
OrgTechEmail: BS185095[@]corp.ncr.com
OrgTechRef: https[:]//whois.arin.net/rest/poc/SPEAR14-ARIN

RTechHandle: CGH3-ARIN
RTechName: Haug, Chris Gordon
RTechPhone: +1-905-819-4168
RTechEmail: ch134537[@]ncr.com
RTechRef: https[:]//whois.arin.net/rest/poc/CGH3-ARIN

Relationships

(I) 153.68.198.14	Characterized_By	(W) Queried whois.arin.n
(I) 153.68.198.14	Connected_To	(F) 5D29DFE2EA9CA8DA3FF7A14FB20C5E86 (5d29d)

Relationship Summary

(F) 5D29DFE2EA9CA8DA3FF7A14FB20C5E86 (5d29d)	Connected_To	(I) 202.126.90.89
(F) 5D29DFE2EA9CA8DA3FF7A14FB20C5E86 (5d29d)	Related_To	(S) Screenshot 1: Program Connection Log
(F) 5D29DFE2EA9CA8DA3FF7A14FB20C5E86 (5d29d)	Connected_From	(I) 153.68.198.14
(S) Screenshot 1: Program Connection Log	Related_To	(F) 5D29DFE2EA9CA8DA3FF7A14FB20C5E86 (5d29d)
(I) 202.126.90.89	Related_To	(P) 443
(I) 202.126.90.89	Characterized_By	(W) inetnum: 202.
(I) 202.126.90.89	Connected_From	(F) 5D29DFE2EA9CA8DA3FF7A14FB20C5E86 (5d29d)
(I) 153.68.198.14	Characterized_By	(W) Queried whois.arin.n
(I) 153.68.198.14	Connected_To	(F) 5D29DFE2EA9CA8DA3FF7A14FB20C5E86 (5d29d)

(F) 8F4FC2E10B6EC15A01E0AF24529040DD (8f4fc)	Related_To	(F) mimefilter.xml_584AC94142F0B7C0DF3D0ADDE6E661ED (584ac)
(F) mimefilter.xml_584AC94142F0B7C0DF3D0ADDE6E661ED (584ac)	Related_To	(F) 8F4FC2E10B6EC15A01E0AF24529040DD (8f4fc)
(F) mimefilter.xml_584AC94142F0B7C0DF3D0ADDE6E661ED (584ac)	Related_To	(S) Screenshot 2: Decrypted Config File
(S) Screenshot 2: Decrypted Config File	Related_To	(F) mimefilter.xml_584AC94142F0B7C0DF3D0ADDE6E661ED (584ac)
(P) 443	Related_To	(I) 202.126.90.89
(W) inetnum: 202.	Characterizes	(I) 202.126.90.89
(W) Queried whois.arin.n	Characterizes	(I) 153.68.198.14

Mitigation Recommendations

US-CERT recommends monitoring activity to the following domain(s) and/or IP(s) as a potential indicator of infection:

- 202.126.90.89
- 153.68.198.14

US-CERT would like to remind users and administrators of the following best practices to strengthen the security posture of their organization's systems:

- Maintain up-to-date antivirus signatures and engines.
- Restrict users' ability (permissions) to install and run unwanted software applications.
- Enforce a strong password policy and implement regular password changes.
- Exercise caution when opening e-mail attachments even if the attachment is expected and the sender appears to be known.
- Keep operating system patches up-to-date.
- Enable a personal firewall on agency workstations.
- Disable unnecessary services on agency workstations and servers.
- Scan for and remove suspicious e-mail attachments; ensure the scanned attachment is its "true file type" (i.e., the extension matches the file header).
- Monitor users' web browsing habits; restrict access to sites with unfavorable content.
- Exercise caution when using removable media (e.g., USB thumbdrives, external drives, CDs, etc.).
- Scan all software downloaded from the Internet prior to executing.
- Maintain situational awareness of the latest threats; implement appropriate ACLs.

Contact Information

- 1-888-282-0870
- soc@us-cert.gov (UNCLASS)
- us-cert@dhs.sgov.gov (SIPRNET)
- us-cert@dhs.ic.gov (JWICS)

US-CERT continuously strives to improve its products and services. You can help by answering a very short series of questions about this product at the following URL: https://forms.us-cert.gov/ncsd-feedback/

Document FAQ

What is a MAR? A Malware Analysis Report (MAR) is intended to provide detailed code analysis and insight into specific tactics, techniques, and procedures (TTPs) observed in the malware.

Can I edit this document? This document is not to be edited in any way by recipients. All comments or questions related to this document should be directed to the US-CERT Security Operations Center at 1-888-282-0870 or soc@us-cert.gov.

Can I submit malware to US-CERT? Malware samples can be submitted via three methods. Contact us with any questions.

- Web: https://malware.us-cert.gov
- E-Mail: submit@malware.us-cert.gov
- FTP: ftp.malware.us-cert.gov/malware (anonymous)

US-CERT encourages you to report any suspicious activity, including cybersecurity incidents, possible malicious code, software vulnerabilities, and phishing-related scams. Reporting forms can be found on US-CERT's homepage at www.us-cert.gov.