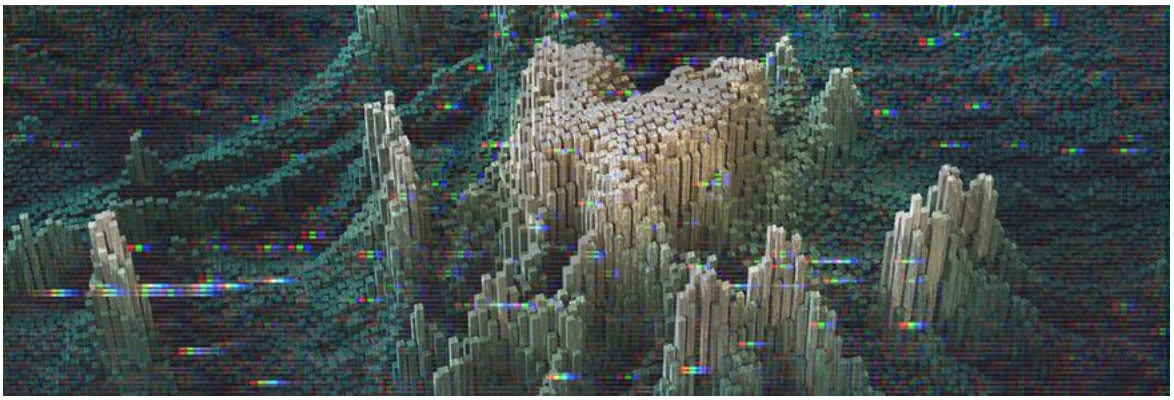
Threat Spotlight: URSNIF Infostealer Malware

cylance.com/en_us/blog/threat-spotlight-ursnif-infostealer-malware.html

The BlackBerry Cylance Threat Research Team



RESEARCH & INTELLIGENCE / 02.07.18 / The BlackBerry Cylance Threat Research Team
Research by Yasmine Ison of the Cylance Threat Guidance Team

URSNIF (Gozi) is a multifaceted malware family with an emphasis on information stealing that has been leveraged to exfiltrate sensitive data from targets, and has been particularly pervasive throughout 2016 and 2017. Since 2007, variants of the malware have been detected in Europe, Japan, and Australia, with more recent outbreaks in the US and UK.

The malware is most often used to target banks, but has also been used to attack email, cloud, commercial, and cryptocurrency trading websites, and is typically propagated by way of phishing campaigns utilizing tainted email attachments. Based on memory analysis, the malware also appears to have the capability to spread to external USB devices and hard drives.

When an URSNIF infection is successful, it will fingerprint the system, monitor web browser traffic, and then send all the data out to the command and control (C&C) server. The server then will drop second stage malware based on the information that tailored the targeted system.

File Information

SHA256: 3E840F21F0EAE2F688BA9E8204AEC22985CC69757B928202A8ADEF0885404EA2
Type: Win32 EXE
Size: 233472 bytes
Timestamp: Tue Nov 28, 10:28:50 2017
ITW Names: avicbrkr.exe, adprtext.exe, catsobby.exe, cmdlnsta.exe

Technical Analysis

The payload is packed with a Delphi overlay, making it a little harder to disassemble and analyze on a binary level. With that being said, memory analysis was also used to complete this analysis along with some disassembly. The file analyzed in the paper needs to run in Windows 7, 32-bit or later, unlike previous versions of URSNIF that were able to run on systems as early as Windows XP.

There, malware authors employ a check in the binary for the file *C:\%filename%.txt* – if it exists, checks for a virtualized environment are ignored. When the file is executed in a Virtual Machine (VM), it fingerprints the machine and later sends the information in a POST.

The file then creates two registry keys for persistence before calling out (Ref IoC Registry Keys). This version of URSNIF also has API hooks, which have been seen in other versions, that give the file the capability to collect email credentials, webcam footage, image files, audio files, and screen captures *(Figure 1)*.

Figure 1: *Known hooked API calls*

URSNIF is known for hooking various executable files in order to monitor browsers. This version hooks *WS2_32.DLL and KERNEL32.DLL* and *CHROME.DLL* to monitor Google Chrome, *NSS3.DLL* and *NSPR4.DLL* to monitor Mozilla Firefox, and *WININET.DLL* to monitor Internet Explorer. Even though it was run on a Windows machine, it can also monitor Opera, as references in the code suggests it hooks to Opera.exe.

The file will test to see if it is connected to the Internet using command prompt and nslookup, and it will also use command to spawn a sub process of itself *(%AppData%\Microsoft\%randomfile%\%randomfile%)*. The sub process creates a bat file that will later check for secondary malware *(%LocalAppData%\Temp\%randomfolder%\%random%.bat)* (Figure 2).

Figure 2: *Temp file created.*

If the test works, the file will inject code into a running process *(explorer.exe)* to send a POST to the C&C server, and hide while continuing to run. The URL will contain *"/data.php?version=% &user=%&server=% &id=% &type=% &name=%"*, replacing this temp data with the victim machine information (Ref IoC Network).

Then the URL data is XOR'ed and base64 encoded (with padding removed "=") and passed as part of the URL which is meant to look like an ordinary image file (jpeg) (Figure 3). After that it will add "/" at random offsets of the string and change every unique letter which doesn't match *[a-zA-Z0-9]* to its hexadecimal format starting with "_".

The last step, adding the "/" slash character at random offsets. The result will look something like this:

LUk2l6EnlB57pGW/_2BdI_2Bw_2FMaug0Z/ZXD_2Fip0/epva0VXtqAj_2FEqP_2B/k2U5Gx_2BJhQW_2B_2B/II3nn6TLQFnbVVfZUpPCFz/FM_2F

Figure 3: Injected data into Explorer.exe

The file has the capability to use DGA (domain name generator algorithm) to make a small list of malicious domains. It employs the open-top-domains algorithm and gathers a few DNS domains from OpenDNS, which is also sometimes seen in in the DreamBot variant of URSNIF.

The open-top-domains algorithm is possibly a backup feature because a known flaw in the code doesn't allocate enough space to correctly construct URLs from a word list. However, this specific sample uses hardcoded network indicators, which can be seen in the network traffic (Ref IoC Network).

When the file calls out to them, it will attempt to download secondary/tertiary stage malware and store in the %temp% directory. A temporary folder and file, or sometimes just a file, will be created.

The file will check to see if the secondary/tertiary stage malware file was downloaded. If it is not, it will delete the temp file. If it is downloaded, the bat file will delete the original malware file.

Conclusion:

Based on recent activity, URSNIF (Gozi) attacks are likely to remain a prevalent threat to high value targets, and organizations who fit the victim profile should exercise diligence in not only basic security hygiene but also in phishing awareness education for staff and others who have access to critical operations systems.

If you are a Cylance customer using CylancePROTECT®, you were already protected from this attack.

IOCs

- **Files/Folders:**

◦ %AppData%\Microsoft\Cmipprop\catsobby.exe
◦ %LocalAppData%\Temp\4B78\2F91.bat

- **Registry Keys:**

◦ HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnce
◦ HKCU\Software\Microsoft\Windows NT\CurrentVersion\Winlogon

- **SHA-256:**

◦ 3E840F21F0EAE2F688BA9E8204AEC22985CC69757B928202A8ADEF0885404EA2

- **Network:**

◦ http|://|constitution|.|org/usdeclar.txt
◦ myip|.|opendns|.|com
◦ resolver1|.|opendns|.|com
◦ curlmyip|.|net

◦ **GET REQUEST:**
advisevisa|.|com/wp-content/themes/porto/css/a|.|zip
alizarineparis|.|com/wp-content/themes/kancing/fonts/b|.|bin

anandbora|.|in/wp-content/themes/apoa/images/a|.|zip
baharguzellik|.|com/assets/lib/gui/a|.|bin
farimon|.|at/jvassets/lg/fsdmkuir_xx|.|rar
farimon|.|at/jvassets/lg/xclek4f|.|zip

- **POST:**
  deepmoler|.|cn/infolot
  farimon|.|at/infolot
  arimon|.|at/infolot
  umbrapo|.|su/infolot
  wlanoer|.|su/infolot
  enzenco|.|at/infolot
  fjroom|.|su/infolot
  fvnoop|.|at/infolot
  glencon|.|at/infolot
  golangland|.|cn/infolot
  hheepet|.|at/infolot

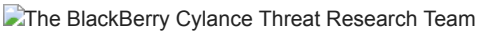
- **TOR Traffic**
  jesteoq7glp3cpkf|.|onion/infolot
  pzgxy5elkuywloqc|.|onion/infolot
  q7nxkpgras35dwxk|.|onion/infolot
  rbhqdxwdwrlp67g6|.|onion/infolot

References:

https://securityintelligence.com/ursnif-campaign-waves-breaking-on-japanese-shores/

About the Cylance Threat Guidance Team

The Cylance Threat Guidance team examines malware and suspected malware to better identify its abilities, function and attack vectors. Threat Guidance is on the frontline of information security and often deeply examines malicious software, which puts them in a unique position to discuss never-seen-before threats.



About The BlackBerry Cylance Threat Research Team

The BlackBerry Cylance Threat Research team examines malware and suspected malware to better identify its abilities, function and attack vectors. Threat Research is on the frontline of information security and often deeply examines malicious software, which puts us in a unique position to discuss never-seen-before threats.

Back