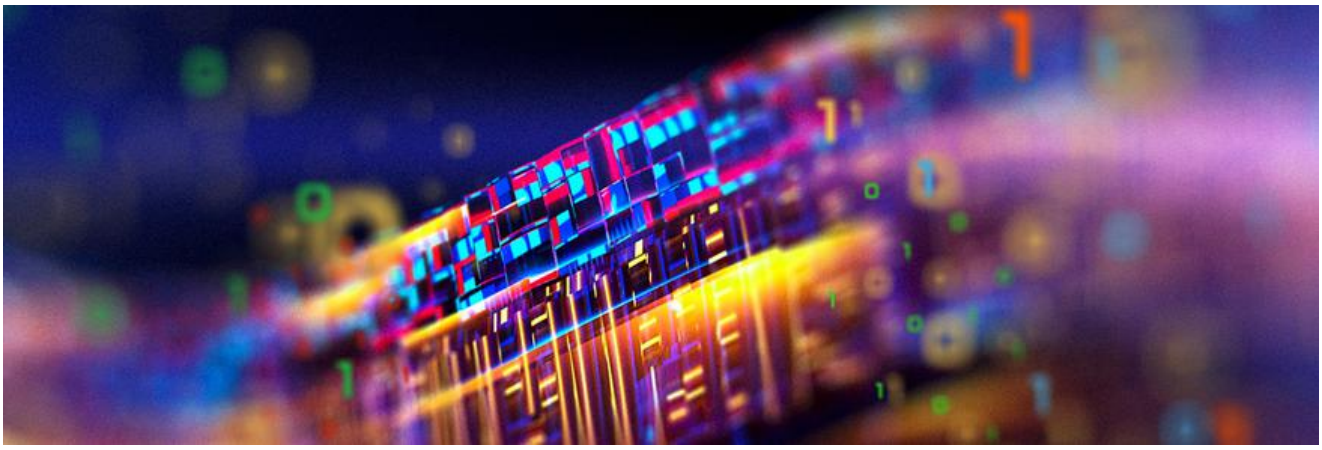
# Threat Spotlight: Analyzing AZORult Infostealer Malware

threatvector.cylance.com/en_us/home/threat-spotlight-analyzing-azorult-infostealer-malware.html

The BlackBerry Cylance Threat Research Team



RESEARCH & INTELLIGENCE / 06.04.19 / The BlackBerry Cylance Threat Research Team

AZORult is an information stealer first analyzed in 2016[1]. It steals browsing history, cookies, ID/passwords, cryptocurrency information, and more. Based on its configuration settings, it can also behave as a downloader. In this blog, we investigate AZORult v3.2 and v3.3[2],[3]. According to ANY.RUN, AZORult ranked as a Top 10 threat from October 2018 to December 2018 [4].

Exploit kits and phishing emails are major infection vectors for this threat. Other malware families such as Ramnit and Emotet also download AZORult. This report details our threat research team's recent technical observations of AZORult.

## Technical Analysis

AZORult is an infostealer malware. Its general behavior is summarized in Figure 1. Once a victim's computer is infected, the malware exfiltrates sensitive data. First, AZORult generates a unique ID of the victim's computer and applies XOR encryption using the generated ID. A masked ID is used for the initial request to command and control (C2) servers.

C2 servers respond with configuration data which contains target web browser names, web browser path information, API names, sqlite3 queries, and legitimate DLLs. AZORult then harvests sensitive information from the victim's computer according to rules set by its

configuration data. The collected information is packed and XORed. AZORult can also download and run additional programs. We did not observe any anti-analysis code in AZORult's malware files:

*Figure 1: AZORult attack cycle*

## ID Generation

Once AZORult runs in the victim's environment, it generates a unique ID from the following data (see Figure 2):

- Machine GUID
- Product name
- Username
- Computer name
- A packed archive file containing the above information

*Figure 2: ID generation*

Each value is acquired with Win32 APIs and registry calls, and converted to 4-byte values with the following function:

```
def IDgeneration(plaintext, key):
    bytetext = bytearray(plaintext.encode('ascii'))
    length = len(bytetext)
    tmp = 0
    for i in range(0, length):
        xoredValue = bytetext[i] ^ key
        xoredValue += tmp
        xoredValue &= 0xFFFFFFFF
        tmp1 = xoredValue >> 19
        tmp2 = (xoredValue << 13) & 0xFFFFFFFF
        tmp = xoredValue - (tmp1 | tmp2) & 0xFFFFFFFF
    return tmp
```

A 4-byte XOR key, 0x6521458A, is passed to the function as well. The same XOR mask is used in AZORult v3.2 and v3.3. Then, all 4-bytes data are concatenated with a hyphen. The generated ID is used for two purposes:

- To create a mutex
- To provide a unique id for the C2 communication

## Command and Control Communication

AZORult attempts to convert the generated ID as follows:

- Append a prefix: "***G***" for AZORult v3.2 and ***3-byte XOR key*** for AZORult v3.3
- URL encoded
- XOR mask

The converted ID is used for the initial C2 communication. AZORult sends an initial request to its C2 server to receive configuration data. The configuration data format is described in Figure 3:

*Figure 3: Configuration data format (v3.3)*

The configuration data has three types of content:

1. **Web browser path information, email client software, hardcoded sqlite3 queries, etc.**
   See Figure 4 for excerpted configuration data.
   AZORult utilizes this information to scan victim's machines and steal sensitive data:

*Figure 4: Excerpted configuration data of this part*

2. **Commands from the C2 servers**
   Commands are also in the configuration data. See Figure 5 as an example:

   ***-F***
   AZORult to upload additional files
   ***-I***
   AZORult to send geolocation of IP addresses of victim machines
   ***-L***
   AZORult to download and run files

*Figure 5: Command list example*

3. **Legitimate DLLs**
   These are used for stealing sensitive information on victim computers:

- api-ms-win-core-console-l1-1-0.dll
- api-ms-win-core-datetime-l1-1-0.dll
- api-ms-win-core-debug-l1-1-0.dll

- api-ms-win-core-errorhandling-l1-1-0.dll
- api-ms-win-core-file-l1-1-0.dll
- api-ms-win-core-file-l1-2-0.dll
- api-ms-win-core-file-l2-1-0.dll
- api-ms-win-core-handle-l1-1-0.dll
- api-ms-win-core-heap-l1-1-0.dll
- api-ms-win-core-interlocked-l1-1-0.dll
- api-ms-win-core-libraryloader-l1-1-0.dll
- api-ms-win-core-localization-l1-2-0.dll
- api-ms-win-core-memory-l1-1-0.dll
- api-ms-win-core-namedpipe-l1-1-0.dll
- api-ms-win-core-processenvironment-l1-1-0.dll
- api-ms-win-core-processthreads-l1-1-0.dll
- api-ms-win-core-processthreads-l1-1-1.dll
- api-ms-win-core-profile-l1-1-0.dll
- api-ms-win-core-rtlsupport-l1-1-0.dll
- api-ms-win-core-string-l1-1-0.dll
- api-ms-win-core-synch-l1-1-0.dll
- api-ms-win-core-synch-l1-2-0.dll
- api-ms-win-core-sysinfo-l1-1-0.dll
- api-ms-win-core-timezone-l1-1-0.dll
- api-ms-win-core-util-l1-1-0.dll
- api-ms-win-crt-conio-l1-1-0.dll
- api-ms-win-crt-convert-l1-1-0.dll
- api-ms-win-crt-environment-l1-1-0.dll
- api-ms-win-crt-filesystem-l1-1-0.dll
- api-ms-win-crt-heap-l1-1-0.dll
- api-ms-win-crt-locale-l1-1-0.dll
- api-ms-win-crt-math-l1-1-0.dll
- api-ms-win-crt-multibyte-l1-1-0.dll
- api-ms-win-crt-private-l1-1-0.dll
- api-ms-win-crt-process-l1-1-0.dll
- api-ms-win-crt-runtime-l1-1-0.dll
- api-ms-win-crt-stdio-l1-1-0.dll
- api-ms-win-crt-string-l1-1-0.dll
- api-ms-win-crt-time-l1-1-0.dll
- api-ms-win-crt-utility-l1-1-0.dll
- freebl3.dll
- mozglue.dll
- msvcp140.dll
- nss3.dll
- nssdbm3.dll

- softokn3.dll
- ucrtbase.dll
- vcruntime140.dll

## How AZORult Steals Victim's Sensitive Information

AZORult scans the victim's environment looking to steal personal information such as passwords, cookies, browsing history, and more. This section explains how the malware grabs password information saved in the victim's Google Chrome browser:

1. **Search target file:**
   In the case of Google Chrome, password information is saved in *%LOCALAPPDATA%\Google\Chrome\User Data\Default\Login Data.* (The path is constructed by the information in the configuration data)

2. **Copy the file "Login Data" to %TEMP%:**
   The destination filename in %TEMP% contains digits constructed with the result data of *GetTickCount()* and *QueryPerformanceCounter()* as shown in Figure 6:

*Figure 6: A temporary file of copied "Login Data"*

3. **Run sqlite3 query:**
   The file "Login Data" is SQLite 3.x database code. To query sqllite3 data, AZORult reads the configuration data to load a dll (in this case, "nss3.dll"). It also loads the sqllite3 query data, then runs the query to extract sensitive information. Once the data is extracted, AZORult saves the information as shown in Figure 7:

*Figure 7: AZORult saves stolen password information with this format*

4. **Delete copied "Login Data" from %TEMP% directory.**
   Based on our analysis, AZORult tries to get sensitive information from these programs:

**Web Browser:**

- Google Chrome
- Comodo Dragon
- Amigo
- Orbitum
- Bromium
- Chromium
- Nichrome
- RockMelt
- Vivaldi
- Go Browser
- Sputnik
- Kometa
- Uran
- QIP Surf
- Epic Privacy Browser
- Brave
- Cent Browser
- CocCoc
- 7 Star
- Elements Browser
- Safer Technologies
- Mustang
- Superbird
- Chedot
- Torch
- Firefox
- Waterfox
- IceDragon
- Cyberfox
- PaleMoon
- InternetExplorer
- Microsoft Edge
- Opera
- Xpom
- YandexBrowser
- 360Browser
- TorBro
- Suhba

- **E-mail client:**
  - Outlook
  - Thunderbird

- **Cryptocurrency:**
  - Electrum
  - Electrum-LTC
  - ElectrumG
  - Electrum-btcp
  - Jaxx
  - MultiBitHD
  - Monero
  - Bitcoin
  - BitcoinGold
  - BitCore
  - Litecoin
  - BitcoinABC
  - Exodus
  - Exodus Eden
  - Ethereum

  **Others:**
  - Filezilla
  - PSI+
  - WinScp
  - Skype
  - Telegram
  - Steam

## Data Format to C2 Server

AZORult collects sensitive data from victim machines and sends it to C2 servers. We confirmed several types of data sent to C2 servers including:

- Basic information of victim's computers such as OS version
- Password information saved by web browsers
- Domain name lists accessed by web browser
- Auto-complete, cookies, and browsing history of web browser
- C2 command result
- Infected host IP address information
- Screenshots of victim host

- Detailed system information
  - Display resolution
  - Running process tree
  - Installed program list
  - Others

The data is packed and a hardcoded separator is added, as shown in Figure 8. Hardcoded separators differ between v3.2 and v3.3. Before the packed data is sent to C2 servers it is XOR masked:

*Figure 8: Packed Information (separator is grayed-out)*

Zip data contains some files and folders as shown in Figure 9:

*Figure 9: Files and folders in zip data*

## Loader Function

AZORult may be configured to download and run additional payloads. It saves the malicious files under %TEMP% or %ProgramFiles%. AZORult checks the extension of downloaded files and if it finds ".exe", it runs them via *CreateProcessW()* as shown in Figure 10. Otherwise, it calls *ShellExecuteW()*. During our investigation, we found hXXp://cindysonam[.]org/putty[.]exe. The file was digitally signed and legitimate putty.exe v0.68. However, the URL also distributed a banking Trojan, Panda Banker: [1],[2]

*Figure 10: AZORult checks extension of downloaded file,*
*then it is run via CreateProcessW or ShellExecuteW*

## Administrator Panel

Blackberry Cylance discovered a builder website and the administrator panel web application. The description, which highlighted several AZORult improvements, was written in Russian:

*Figure 11: A screenshot of builder website of AZORult*

The administrator panel is designed for showing information captured from victim computers. On the top page it displays statistics of stolen information and the countries of victims:

*Figure 12: Top page of administrator panel*

Administrators can edit stealer config to specify information they want to steal and include additional malware samples. AZORult will receive these administrator preferences as configuration data:

*Figure 13: AZORult's stealer config menu*

## Blackberry Cylance Blocks AZORult

Threat actors use AZORult to steal system information, browsing history, cookies, IDs/passwords saved in browsers, cryptocurrency information and more. Though AZORult does not currently implement anti-analysis measures, we suspect it will in the future.

Blackberry Cylance uses artificial intelligence-based agents trained for threat detection on millions of both safe and unsafe files. This allows Blackberry Cylance to spot a threat based on countless file attributes instead of a specific file signature to block AZORult. Blackberry Cylance, which offers a predictive advantage over zero-day threats, is trained on and effective against both new and legacy cyberattacks.

## Indicators of Compromise (IOCs)

### AZORult payload

494EDDDC91292A5B25681C985F52850518AC9F9F5634232866F8D821B1B645C0: AZORult v3.2

E022B5AFC18C2E5E9E74307CD27B0ADD7A5A0CE7BE41678223CEEA76DBED6F26: AZORult v3.2

12B6A633B470216952DB405356C9B565EE58C6DCB27D57ED6492DFAF51D22E61: AZORult v3.3

748C94BFDB94B322C876114FCF55A6043F1CD612766E8AF1635218A747F45FB9: AZORult v3.3

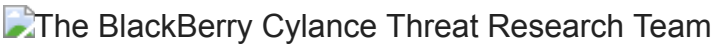
### C2 servers

- hXXp://nagoyashi[.]chimkent[.]su/index[.]php
- hXXp://ivanoffol3[.]temp[.]swtest[.]ru/index[.]php
- hXXp://mockerton[.]top/index[.]php

- hXXp://www[.]jma-go[.]jp/java/java9356/index[.]php
- hXXp://cindysonam[.]org/putty[.]exe

**Mutexes**

- A{Generated ID by AZORult}




## 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