New Version Of Apostle Ransomware Reemerges In Targeted Attack On Higher Education

sentinelone.com/labs/new-version-of-apostle-ransomware-reemerges-in-targeted-attack-on-higher-education/

Amitai Ben Shushan Ehrlich

SentinelLabs has been tracking the activity of Agrius, a suspected Iranian threat actor operating in the Middle East, throughout 2020 and 2021 following a set of destructive attacks starting December 2020. Since we last reported on this threat actor in May 2020, Agrius lowered its profile and was not observed conducting destructive activity. This changed recently as the threat actor likely initiated a ransomware attack on the Israeli university Bar-Ilan utilizing the group's custom Apostle ransomware.

Although the full technical details of the incident were not disclosed publicly, some information was released to the public, most notably the ransom demand text file dropped on victim machines. The `.txt` file matches that from a new version of Apostle compiled on August 15, 2021, the day of the attack.

The new version of Apostle is obfuscated, encrypted and compressed as a resource in a loader we call Jennlog, as it attempts to masquerade payload in resources as log files. Before executing the Apostle payload, Jennlog runs a set of tests to verify that it is not being executed in an analysis environment based on an embedded configuration. Following the analysis of the Jennlog loader, SentinelLabs retrieved an additional variant of Jennlog, used to load and run OrcusRAT.

Jennlog Analysis

Jennlog (`5e5e526a69490399494dcd7195bb6c67`) is a `.NET` loader that deobfuscates, decompresses and decrypts a `.NET` executable from a resource embedded within the file. The resources within the loader appear to look like log files, and it contains both the binary to run as well as a configuration for the malware's execution.

Jennlog attempts to extract two different resources:

- `helloworld.pr.txt` – stores Apostle payload and the configuration.
- `helloworld.Certificate.txt` – contains None. If configured to do so, the malware compares the MD5 value of the system information (used as system fingerprint) to the contents of this resource.

The payload hidden in " `helloworld.pr.txt` " appears to look like a log file at first sight:

```
2019-12-01 09:29:54, Info                       LogSession: Starting a new log session at [C:\$SysReset\Logs]
2019-12-01 09:29:54, Info                       Registry: Loading SOFTWARE hive from online OS
2019-12-01 09:29:54, Info                       ResetCreateSession: Succeeded
2019-12-01 09:29:54, Info OnlineUI: Reset session created successfully
2019-12-01 09:29:54, Info                       OnlineUI: Determining whether Factory scenario is available
2019-12-01 09:29:54, Info ResetValidateScenario: Scenario: [2] Target: [C:]
2019-12-01 09:29:54, Info Session: Checking main OS for a test ID
2019-12-01 09:29:54, Info Telemetry: Checking[C:] for a test ID
2019-12-01 09:29:54, Info Registry: Loading SOFTWARE hive from online OS
2019-12-01 09:29:54, Info                       Telemetry: No test ID present
2019-12-01 09:29:54, Info                       Session: Reading system configuration
2019-12-01 09:29:54, Info SenseContext: Opening system BCD
2019-12-01 09:29:54, Info SenseContext: Reading WinRE config
2019-12-01 09:29:54, Info[SystemSettingsAdminFlows.exe] Enter WinReGetConfig
2019-12-01 09:29:54, Info[SystemSettingsAdminFlows.exe] Parameters: configWinDir: NULL
2019-12-01 09:29:54, Info[SystemSettingsAdminFlows.exe] WinRE config file path: C:\Windows\system32\Recovery\ReAgent.xml
2019-12-01 09:29:54, Info[SystemSettingsAdminFlows.exe] Update enhanced config info is enabled.
2019-12-01 09:29:55, Info[SystemSettingsAdminFlows.exe] WinRE is installedJennifer~~1~~0~~0~~0~~0~~0~~0~~0~~0~~0~~0~~0~~0Jennifer
Log  8v 3 F 2pal Sdrx C X W I Y 4u/ S Yogdxylr F 8du 7xu Ok Wb Cjlyo Z N 3jq Eq Cd
Log  5 Ru 3 V Np U 1e Be Lb 8+ 6g 4 E Mpf Cq 2 P 4 1 D 9 D Yh Z Ra 6 Yw 2xjv W 2v 2 Pu 0 Qm 6b V 0 Qyz C E F Ti 5 0z Dp Od 4q Z U Q Lbluz 6k V Cr E Sd Krvk C 2z 1o 3 Vwzoe 3 U
Q Z L E Edw 4ms Q Dj W Y Q Oh/ P Y N N 0n Xzhw F Lkzuc 0 Yn 6qv W Ha J 8t Kr Jw 1 Hg 0bau F 2 U/ht Kpv 1 Dyd I M P H K D I/ Tu 1g J Muq 4 Lp 4surzeg Ybne 8vxa 5/ Mv 4o N E Zl
Y Dt/ I+g Mp 7u L I Eb Wkl 0jyr 8i Kk Joe 5 Mfh Mq 9u Zreyg X 1 G J 0 0 Pn 3ego Se 6oo P Da G Y+ Y M 2e N Truhw Cvv W L/ R I Si 7q Ci 3 R Y 3 M Nhyb Dgd 8b H Xog Ha S/yeq 8n E
D Cwdvy Y 4nwf Yy Zr+ 3 Zo Poy Jmmq Uz T/+bvykql 2bs K Hjh Iq Vpitvjn Scdh Yg 5p 7 M 8e 2 8d F C Sh U 1t Esv U Cb W 1h L Eps O Wc X S Jw V Tmui X E/p Z Cr 7 P N+ Wj Jryw C
```

Contents of " `helloworld.pr.txt` " resource embedded within Jennlog

The payload is extracted from the resource by searching for a separator word – " `Jennifer` ". Splitting the contents of the resource results in an array of three strings:

1. Decoy string – Most likely there to make the log file look more authentic.
2. Configuration string – Used to determine the configuration of the malware execution.
3. Payload – An obfuscated, compressed and encrypted file.

Configuration

The configuration of Jennlog consists of 13 values, 12 of which are actually used in this version of the malware. In the variants we were able to retrieve, all of these flags are set to 0.

#	Title (given by SIGNAL)	Description
0	Ignored Flag	This flag is simply ignored.
1	Window Flag	Show the console window.
2	Sandbox Flag	Checks for indications of running in a sandbox environment using indicative process names as well as handles for specific DLLs.
3	Debugger Flag	Checks if a debugger is attached to the program.
4	VM Flag	Checks if the malware is run inside a VM based on process names and manufacturer names.
5	TaskManager Flag	Disables the task manager for the user by setting the registry value Software\\Microsoft\\Windows\\CurrentVersion\\Policies\\System\\DisableTaskMgr to 1
6	Sleep Flag	The malware performs sleep before executing.
7	Sleep Value	If the sleep flag is set to 1, sleep for the number of seconds stored in this value.
8	Message Flag	Presents a message upon execution.
9	Message Body	If the message flag is set to 1, display the content here as the message's body.
10	Message Title	If the message flag is set to 1, display the content here as the message's title.
11	Certificate Flag	Compare the system MD5 fingerprint to the one stored in the "helloworld.Certificate.txt" resource. If it does not match, either exit the program or delete the file, based on values stored in the stop flag.
12	Stop Flag	1 - Simply exit the program in case the fingerprint was matched. 0 - Run self delete function utilizing a BAT script.

Jennlog configuration values

One of the most interesting flags found here is the certificate flag. If this flag is set, it will cause the malware to run only on a specific system. If this system does not match the configured MD5 fingerprint, the malware either stops operation or deletes itself utilizing the function `ExecuteInstalledNodeAndDelete()`, which creates and runs a BAT file as observed in other Agrius malware.

```
public static void ExecuteInstalledNodeAndDelete()
{
  string path = "build.bat";
  string fileName = Process.GetCurrentProcess().MainModule.FileName;
  string contents = "@echo off\nPING 127.0.0.1 -n 5\n:loop\ndel \"" + fileName + "\"\nif Exist \"" + fileName
      + "\" GOTO loop\n%windir%\\system32\\rundll32.exe advapi32.dll,ProcessIdleTasks\ndel %0";
  File.WriteAllText(path, contents);
  Process.Start(new ProcessStartInfo()
  {
    WindowStyle = ProcessWindowStyle.Hidden,
    CreateNoWindow = true,
    FileName = path
  });
}
```

Jennlog `ExecuteInstalledNodeAndDelete()` function

Following all the configuration based-checks, Jennlog continues to unpack the main binary from within the resource " `helloworld.pr.txt` " by performing the following string manipulations in the function `EditString()` on the obfuscated payload:

- Replace all " `\nLog` " with " `A` ".
- Reverse the string.
- Remove all whitespaces.

This manipulation will result in a long base64-encoded deflated content, which is inflated using the function `stringCompressor.Unzip()`. The inflated content highly resembles the contents of the original obfuscated payload, and it is deobfuscated again using the `EditString()` function.

The deobfuscation of the inflated content is carried out in a rather peculiar way, being run as a "catch" statement after attempting to turn a string containing a URL to int, which will always result in an error. The domain presented in the URL was never bought, and highly resembles other Agrius malware unpurchased domains, often used as "Super Relays". Here, however, the domain is not actually contacted.

```
try
{
  Convert.ToInt32("https://[redacted].com");
}
catch
{
  Program.EditString(ref str4);
}
```

Execution of `EditString()` function as a catch statement

Following a second run of the `EditString()` function, Jennlog decodes the extracted content and decrypts it using an implementation of RC4 with a predefined key. The extracted content found in this sample is a new version of the Apostle ransomware, which is loaded into memory and ran using the parameters given to Jennlog at execution.

Apostle Ransomware Analysis

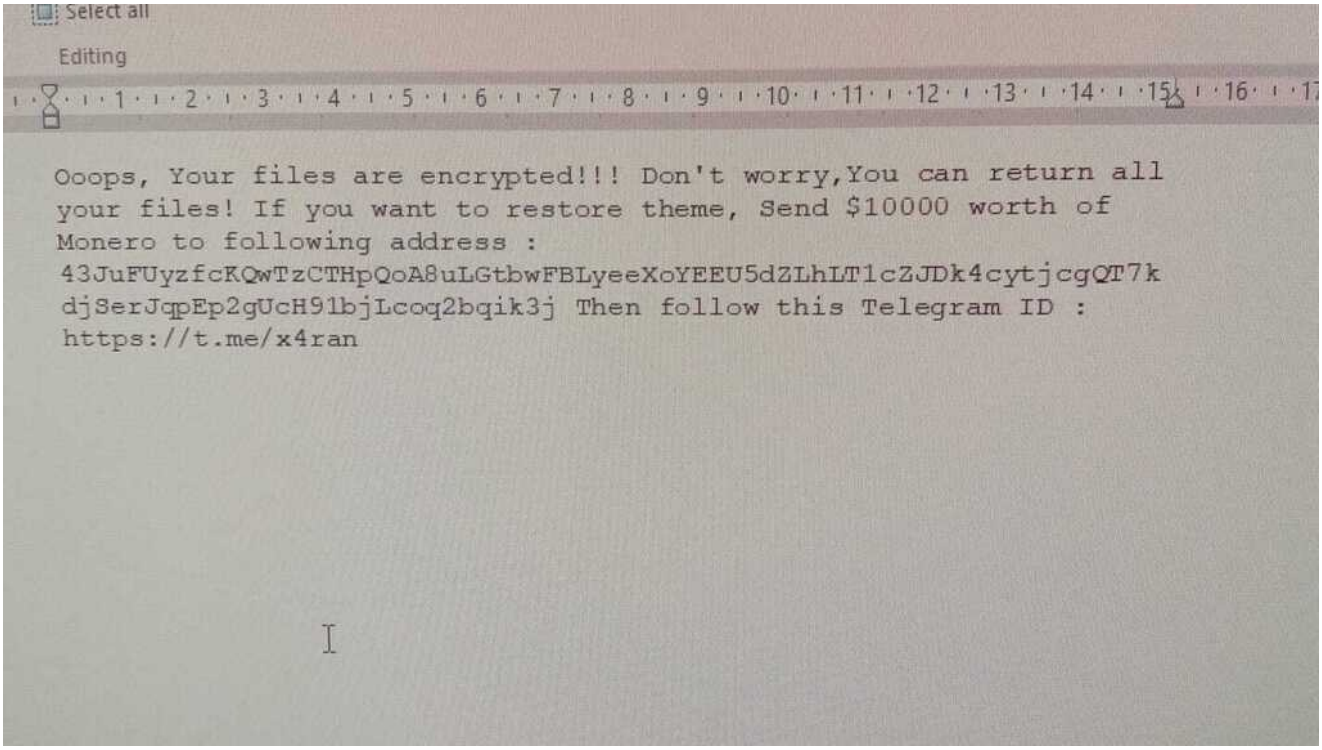
The new variant of Apostle (`cbdbda089f7c7840d4daed22c34969fd876315b6`) embedded within the Jennlog loader was compiled on August 15, 2021, the day the attack on Bar-Ilan university was carried out. Its execution flow is highly similar to the variant described in previous reports, and it even checks for the same Mutex as the previous ransomware variant.

The message embedded within it, however, is quite different:

```
Ooops, Your files are encrypted!!! Don't worry,You can return all your files!
If you want to restore theme, Send $10000 worth of Monero to following address :
43JuFUyzfcKQwTzCTHpQoA8uLGtbwFBLyeeXoYEEU5dZLhLT1cZJDk4cytjcgQT7kdjSerJqpEp2gUcH91bjLc

Then follow this Telegram ID :  hxxps://t[.]me/x4ran
```

This is the exact same message that was released to the media in the context of the Bar-Ilan ransomware incident, as reported on ynet:



Ransom demand text file as seen in Bar-Ilan university

Other than the ransom demand note, the wallpaper picture used on affected machines was also changed, this time presenting an image of a clown:

New Apostle variant wallpaper image

OrcusRAT Jennlog Loader

An additional variant of Jennlog (`43b810f918e357669be42030a1feb727`) was uploaded to VirusTotal on July 14, 2021 from Iran. This variant is highly similar to the one used to load Apostle, and contains a similar configuration scheme (all set to 0). It is used to load a variant of OrcusRAT, which is extracted from the files resources in a similar manner.

The OrcusRAT variant (`add7b6b60e746c36a66f5ec233873372`) extracted from within it was submitted to VT on June 20, 2021 using the same submitter ID from Iran. It seems to connect to an internal IP address – `192.168.178.114` , indicating it might have been used for testing. It also contained the following PDB path:

```
C:\Users\dou\Desktop\repo\arcu-win\src\Orcus\obj\Debug\Orcus.pdb
```

Conclusion

Agrius has shown a willingness to strategically wipe systems and has continued to evolve its toolkit to enable ransomware operations. At this time, we don't know if the actor is committed to financially-motivated operations, but we do know the original intent was sabotage. We expect the sort of subterfuge seen here to be deployed in future Agrius operations. SentinelLabs continues to track the development of this nascent threat actor.

Technical Indicators

Jennlog Loader (Apostle Loader)

- 5e5e526a69490399494dcd7195bb6c67
- c9428afa269bbf8c48a08a7109c553163d2051e7
- 0ba324337b1d76a5afc26956d4dc9f57786483230112eaead5b5c92022c089c7

Apostle – Bar-Ilan variant

- fc8221382521a40ec0042431a947a3ca
- cbdbda089f7c7840d4daed22c34969fd876315b6
- 44c13c46d4f597ea0625f1c87eecffe3cd5dcd257c5fac18a6fa931ba9b5f97a

Jennlog Loader (OrcusRAT Loader)

- 43b810f918e357669be42030a1feb727
- 3de36410a99cf3bd8e0c56fdeafa32bbf7625af1
- 14659857df1753f720ac797a43a9c3f3e241c3df762de7f50bbbae00feb818c9

OrcusRAT

- add7b6b60e746c36a66f5ec233873372
- a35bffc49871bb3a48bdd35b4a4d04d208f23487
- 069686119adc13e1785cb7a425611d1ec13f33ae75962a7e50e00414209d1809