Fortinet’s FortiGuard Labs captured a phishing campaign that delivers three fileless malware onto a victim’s device. Once executed, they are able to control and steal sensitive information from that device to perform other actions according to the control commands from their server.

In Part I of this analysis, I introduced how these three fileless malware are delivered to the victim’s device via a phishing campaign, and what mechanism it uses to load, deploy, and execute these fileless malware in the target process.

In Part II, I will focus on the three malware payloads and elaborate on how they steal sensitive information from the victim’s device, how they submit data to their C2 server, details about the control commands, as well as what they can perform with those control commands.

Affected platforms: Microsoft Windows Impacted parties: Microsoft Windows Users Impact: Controls victim’s device and collects sensitive information Severity level: Critical

Fileless Malware 1 - AveMariaRAT

“Ave Maria” is a RAT (Remote Access Trojan), also known as WARZONE RAT. It offers a wide range of features, such as stealing victim’s sensitive information and remote controlling an infected device, including privilege escalation, remote desktop control, camera capturing, and more.

It is the first of the three malware (refer to Figure 3.3 of the previous analysis) to be injected into a newly-created “aspnet_compiler.exe” process on the victim’s device and then run.

Step One:

Ave Maria has a configuration block that is RC4 encrypted within its PE structure’s “.bss” section. The decryption key and encrypted data are together within the “.bss”. When the malware starts, it first decrypts the configuration block. Figure 1.1 shows the decrypted data in memory.

Figure 1.1 — RC4-decrypted configuration block

It not only contains the C2 server (“mubbibun.duckdns.org”) and port (0x3E7), but also a number of switch flags, such as whether to add itself into the auto-run group, bypass UAC (User Account Control), or bypass Windows Defender.

Stage Two:

Once Ave Maria establishes its connection to the C2 server it starts to control the victim’s device. According to my research, the traffic between its client and the C2 server is RC4 encrypted with a constant encryption key "warzone160".

I’ll explain what the plaintext packet consists of through an instance packet as below.

29 BB 66 E4 70 EA 00 00 1E 00 00 00 00 00 00 00

00 FA 07 00 00 00 00 00 60 EA 00 00 4D 5A 90 00

…

- The first dword 0xE466BB29 is a magic value; each packet must start with this value.
- The 0xEA70 is the size of the command data. It is 0x0 if no command data.
- The 0x1E is the command number of this packet.
- The subsequence data is the command data, which is an executable file in this example packet. It doesn’t appear if there is no command data for the command.

Ave Maria provides these features:

Remote VNC (Virtual Network Computing), Remote Shell, File Explorer, Process Manager, Remote Webcam, Password Manager, Reverse Socks, Download & Execute a file, Remote Keylogger, HRDP Manager as well as Privilege Escalation.

Figure 1.2 — Display of Remote Keylogger feature

Figure 1.2 shows the feature (on the C2 server side) of the online Remote Keylogger (command 24H). You can see here what is recorded when I open a Chrome browser and type in “www.fortinet.com” and press Enter on a victim’s device.

Its Password Manager feature aims to steal credentials from a group of apps, listed below, including internet browsers and email clients.

Google Chrome, Epic Privacy browser, Microsoft Edge, UCBrowser, Tencent QQBrowser, Opera, Blisk, Chromium, Brave-Browser, Vivaldi, Comodo Dragon, Torch, Slim, CentBrowser, Microsoft Internet Explorer, Mozilla Firefox, Microsoft Outlook, Microsoft Messaging, Mozilla Thunderbird, Tencent Foxmail, and more.

Figure 1.3 — The pseudo code of stealing credentials

Figure 1.3 is a screenshot of the pseudo code where Ave Maria steals the credentials (command 20H) from the defined files for several internet browsers. It calls a function to RC4-encrypt the credentials and send them to the C2 server.

Table 1.1 presents most of the control commands that Ave Maria supports.

Cmd Num	Description
00H	Ask for basic information of the victim’s device.
02H	List running processes.
04H	Start File Explorer.
06H	Navigate file.
08H	Retrieve a file from Ave Maria home folder (%LocalAppData%\Microsoft Vision\) on the victim device.
0AH	Delete a file.
0Ch	Kill a process.
0EH	Execute a shell command.
12H	List victim’s camera device information.
14H	Start victim’s camera.
16H	Stop the camera.
18H	Obtain the title of the active program.
1Ah	Uninstall Ave Maria client from the victim’s device.
1Ch	Transfer a file from C2 server to the victim's device.
1EH	Transfer an executable file to the victim’s device and run.
20H	Obtain the credentials of the apps from the victim’s device.
22H	Download a file from a given URL and execute.
24H	Start the online keylogger.

26H	Stop the online keylogger.
28H	Install HRDP Manager on the victim's device.
2AH	Reverse connect to C2 server for HRDP.
30H	Start the Remote VNC.
32H	Stop the Remote VNC
38H	Start reverse sock.
3AH	Execute a specified file on the victim's device.
3CH	Start the offline keylogger.
3EH,40H	Privilege Escalation.
48H	Transfer a file to the victim.
4AH	Retrieve a folder from the victim device.

Table 1.1 — Ave Maria’s Control Commands

Fileless Malware 2 — PandorahVNC RAT

The second fileless malware injected into “RegAsm.exe” is “PandorahVNC Rat,” which is a commercial software. It was developed using C#, a Microsoft .Net framework. It supports features to steal credentials from some popular applications, like Chrome, Microsoft Edge, Firefox, Outlook, Foxmail, and so on. It also supports control commands to control the victim’s device, such as starting a process, capturing the screenshot, manipulating the victim’s mouse and keyboard, and more.

Stage One:

When it starts, it defines the following variables. They are the C2 server address, port, and the group id that will be used when sending data to the C2 server.

```
string str = "vncgoga.duckdns.org"; //C2 server string str2 = "1338"; // TCP port string identifier = "3H4RHL"; // Group id
```

Next, it proceeds to extract a core module from a base64-encoded string, which performs all features of PandoraHVNC RAT. It then deploys the core module into a newly-created process, “cvtres.exe” (a file from Microsoft .Net framework), using process hollowing. It tries to find the file from one of the following:

“C:\Windows\Microsoft.NET\Framework\v4.0.30319\" "C:\Windows\Microsoft.NET\Framework\v2.0.50727\”

If it fails to locate the file, it quits without running PandoraHVNC RAT’s core module. The C2 server address, port, and group id will be passed to the new process during the process hollowing. Figure 2.1 shows the code segment to perform the process hollowing.

Figure 2.1 — Code to perform the process hollowing

Stage Two:

It collects basic information from the victim’s device and sends it to the C2 server to register the client in the server. Below is the data of such packet.

Figure 2.2 — The registration packet to the C2 server

The first 64-bit integer 0x62 is the entire packet size. The subsequent data is sealed in a serialized binary object. The 16H data ("00 01 … 0000") is kind of header. The next 0x4a is the size of the following strings, which is a variable length integer. The last 0x0b is a close flag.

Let's take a look the sealed string, which consists of a packet number ("654321"), client group id ("3H4RHL"), the victim's username and pc name ("Bobs@BOBS-PC"), the victim's location code ("US"), the system information ("Windows 7 Pro"), date ("05/09/2022"), client version ("3.1"), and whether any antivirus product is being used ("False").

Once the C2 server receives this packet, it shows the victim's information in a list, as shown in Figure 2.3, "Connected client".

Figure 2.3 - PandoraHVNC RAT C2 server interface and features

The hacker is then able to control the victim's machine by right-clicking the client and clicking the items on the menu. Figure 2.3 also demonstrates the right-click menu and features.

Figure 2.4 — The credentials packet to C2 server

Figure 2.4 is a screenshot of two packets, the first line is the received command control packet with command number "3308". The rest is the packet sent to the C2 server with packet number "3308". This is followed by the stolen credentials from the victim when "Pandora Recovery" is clicked by the hacker. As mentioned before, the two packets are sealed in the serialized binary object.

Table 2.1 demonstrates the details of all the control commands and features that PandoraHVNC RAT provides.

Cmd Num	Description
0	Start to capture the screenshot.
1	Abort the screenshot.
2	Simulate mouse left button DOWN.
3	Simulate mouse right button DOWN.
4	Simulate mouse left button UP.
5	Simulate mouse right button UP.
6	Perform mouse double click.
7	Simulate to press a Key.
8	Move the mouse to a given point.
9	Send the data of the system clipboard to its C2 server.
10	Set given data to system clipboard.
11	Start a Chrome browser with specified parameters.
12	Start Mozilla Firefox with specified parameters.

13	Show the StartMenu.
14	Minimize Pandora HVNC Rat.
15	Show Pandora HVNC Rat to the victim.
16	Show a pop-up message to the victim.
17	Set screenshot interval.
18	Set screenshot quality.
19	Set screenshot size.
21	Start Explorer program.
24	Kill the current process.
30	Start Microsoft Edge browser with specified parameters
32	Start Brave browser with specified parameters
50	Call KillMiner() to kill a process.
55	Download a file into %temp% folder as a Miner.
56	Download a file and execute.
444	Start an Opera browser with specified parameters.
555	Restart Outlook.
556	Restart FoxMail.
557	Restart Thunderbird.
666	Kill current Pandora HVNC Rat.
1337	Send Pong packet.
3306	Push data to override the system clipboard.
3307	Obtain the data from the system clipboard.
3308	Obtain credentials and cookies from the victim’s browsers.

4875	Start a CMD program.
4876	Start a PowerShell program.
8585	Start a Chrome browser with a default URL.
8586	Kill all Chrome browsers.
8587	Reset Scale.
8589	Same as 56. Download a file and execute.

Table 2.1 - List of control commands of PandoraHVNC RAT

Fileless Malware 3 — BitRAT

The third fileless malware injected into “aspnet_compiler.exe” is “BitRat”, which is said to be a high quality and efficient RAT. It provides information collection like clipboard logger, keylogger, application credentials, Webcam logging, and Voice Recording. It has wide control commands for controlling the victim’s device, including downloading and executing a file, performing remote desktop control, controlling processes and services, reverse socks, and more.

Stage One:

BitRat has a configuration block encrypted similar to the Ave Maria Rat. Figure 3.1 shows the just decrypted configuration block in memory, where it contains C2 server (“maraipasoo[.]duckdns[.]org”) & port (890), client ID (“f2b8b66873ca913a”), and more.

Figure 3.1 — Decrypted configuration block

It proceeds to connect to the C2 server. It then uses TLS 1.2 protocol plus a cipher suite of RAS+AES 256 to transfer and encrypt its packet. Figure 3.2 shows the model that it uses to encode the plain text data with Base64 and encrypt with AES-256. And finally, it sends the encrypted data to the C2 server over TLS 1.2 protocol.

Figure 3.2 — The outline of how a packet is transformed and sent to the C2 server

I’ll take a moment here to explain what the plaintext packet looks like. Figure 3.3, below, is a screenshot of the debugger when BitRat was about to Base64-encode the plaintext packet with the basic information of the victim’s device.

Figure 3.3 — Basic information packet before Base64 encoding

The packet consists of many parts separated by “|”, including a client ID (“f2b8b66873ca913a”), user name, computer name, CPU information, GPU card, system name (“Win 7”), system’s uptime, system idle time, RAM amount, IP address, whether or not the login user is an administrator, BitRat client version (“1.38”), and so on. Next, the packet goes through Base64-encoding and AES-256 encryption, which is eventually sent to the C2 server.

Once the C2 server receives this packet, the victim’s device shows up in its control interface, where the hacker is able to control the infected device.

Stage Two:

BitRat is more powerful than AveMariaRAT and PandoraHVNC because it provides a great number of control commands (172 commands) to control the victim’s device.

Figure 3.4 shows the dashboard to a connected victim on the C2 server side. On the left is the basic information of the victim’s device, while there are some features listed on the right.

Figure 3.4 — Dashboard and features that BitRat supports

Other than the features from the dashboard, it also supports the following features from the main context menu:

- Chat
- Clear browsers
- Clipboard management
- DLL injection
- Change desktop background
- Open website
- Notes
- UAC bypass
- Kill Windows Defender
- Show preview of screen or webcam
- Keylog download & search
- Reverse socks
- System management (reboot, shutdown, sleep, etc.)
- BitRat client's update and uninstall
- DDoS attack (plugin)
- Mining (plugin)
- Telegram bot (plugin)
- Passwords Logins (plugin)
- And more

Stage Three:

While BitRat receives the control command packet, it only needs a AES-256 decryption to restore the plaintext packet. I will explain the structure of a plaintext command packet using the following example:

"ddos_start|MTkyLjE2OC4yMi4xNQ==|3333|1|tcp|tcp|1|0|L3NpdGUucGhwP3g9dmFsMSZ5PXZhbDI="

Every command packet starts with a command name string and subsequent parameters, which are separated by “|”.

The above example command asks the infected device to start a DDoS attack, where “ddos_start” is the command name, “MTkyLjE2OC4yMi4xNQ==” is the Base64-encoded target IP, “3333” is the target port, flood method is “tcp”, protocol is “tcp”, thread number is “1”, size is “0”, and the last field is the data.

On the very first time, BitRat initializes a linked list with nodes containing a command name and a command number (like “ddos_start” for 85H) as well as some flags. BitRat has a method to go through these nodes looking for a node that matches the command name from the packet by string comparison. The command name starts at offset +10H of the node and the corresponding command number is saved in a dword at offset +28H. Below is a dumped node of “ddos_start”.

```
Offset+00 A0 2E 30 00 00 71 3E 00 A0 2E 30 00 00 00 00 00 .0..q>. .0.....

Offset+10 64 64 6F 73 5F 73 74 61 72 74 00 00 00 00 00 00 ddos_start......

Offset+20 0A 00 00 00 0F 00 00 00 85 00 00 00 .......…..
```

BitRat performs the action according to the command number. Table 3.1 lists the most control commands, with brief command descriptions that BitRat supports.

Cmd Name	Num	Description
"cli_up"	00H	Update BitRat client.
"rc"	01H	Reconnect to the C2 server.

"cli_dc"	02H	Disconnect to the C2 server.
"cli_un"	03H	Uninstall BitRat client from the victim's device.
"cli_sleep"	04H	Put the victim's system into sleep.
"cli_hib"	05H	Put the victim's system into hibernation.
"cli_log"	06H	Have the victim's system to log out the current user.
"cli_rs"	07H	Restart the victim's device.
"cli_off"	08H	Shutdown the victim's device.
"cli_bsod"	09H	Make the victim's system crash with a blue screen.
"info"	0AH	Request for the basic information of the victim's device.
"drives_get"	0BH	List drivers, like "C:\", "D:\" and etc.
"files_exec"	0CH	Execute a file on the victim's disk with given parameters.
"files_delete_normal"	0FH	Delete a specified file.
"files_delete_secure"	10H	Delete a specified file with a security way.
"files_rename"	11H	Rename a file.
"files_new_dir"	12H	Create a folder.
"files_zip"	13H	Make a zip archive of a file.
"files_zip_dir"	14H	Make a zip archive of a folder.
"files_get"	15H	List files under a specified path.
"files_search"	16H	Search files by filter string.
"files_search_stop"	17H	Stop file searching.
"files_download"	18H	Transfer a file from the victim's device to the C2 server.
"files_upload"	1AH	Transfer a file from the C2 server onto the victim's device.
"prc_list"	1DH	List running processes.

"prc_suspend"	1EH	Suspend a process with its PID.
"prc_resume"	1FH	Resume a suspended process with its PID.
"prc_priority"	20H	Set a process's priority with a given PID.
"prc_kill"	21H	Kill a process with its PID.
"prc_restart"	22H	Restart a process.
"srv_list"	23H	List system services on the victim's device.
"srv_start"	24H	Start a service.
"srv_control"	25H	Pause, stop, continue a service.
"wnd_list"	27H	List all windows being opened on the victim's device.
"wnd_cmd"	28H	Control a window. such as hide, show, maximize, minimize, etc.
"dlexec"	2AH	Download and execute an executable file.
"screenlive"	2CH	Start screen capture.
"screenlive_stop"	2DH	Stop screen capture.
"screenlive_monitor"	2EH	Start screenlive monitor.
"screenlive_size"	2FH	Set screenlive size.
"screenlive_quality"	30H	Set screenlive quality"
"screenlive_cursor"	31H	Set screenlive cursor to show or hide.
"screenlive_color"	32H	Set screenlive color to gray or color.
"screenlive_click"	35H	Simulate to perform mouse click on screenlive windows.
"screenlive_move"	36h	Move screenlive to a given position.
"screen_preview_start"	38H	Start screen preview.
"screen_preview_stop"	39H	Stop screen preview.
"monitors_refresh"	3BH	Refresh monitors.

"webcam_devices"	3CH	List webcam interfaces.
"webcam_quality"	3DH	Set webcam quality.
"webcam_start"	3EH	Start webcam capture.
"webcam_stop"	3FH	Stop webcam.
"klgoff_list"	43H	List offline keylogger files.
"klgoff_get"	44H	Transfer an offline keylogger file.
"klgoff_dl_all"	45H	Transfer all keylogger files.
"klgoff_del"	46H	Delete an offline keylogger file.
"klgonlinestart"	48H	Start the online keylogger.
"klgonlinestop"	49H	Stop the online keylogger.
"klg_search"	4AH	Search keywords in keylogger data.
"aud_rec_list"	4DH	List audio devices.
"shell_start"	4EH	Start a remote shell on the victim's device.
"shell_stop"	4FH	Stop the remote shell.
"shell_exec"	50H	Exeucte a command trhough the remote shell.
"con_list"	51H	List all processes with network connections.
"crd_logins_data"	64H	Collect the credentials from apps on the victim's device.
"crd_logins_req"	65H	Transfer the collected credentials.
"remotebrowser"	6DH	Remotely start the victim's default browser invisible.
"remotebrowser_stop"	6EH	Stop the remote browsers.
"remotebrowser_key"	6FH	Press a keyboard key on the remote browser.
"remotebrowser_click"	70H	Click on the remote browser.
"remotebrowser_quality"	72H	Set the remote browser quality.

"settings"	78H	Configure the BitRat client.
"soft_list"	79H	List the installed software on the victim's device.
"soft_uninstall"	7AH	Uninstall software from the victim's device.
"reg_hkeys_get"	7EH	Obtain a list of HKEYs (Handles to the Keys) of the victim's system registry.
"reg_keys_root_get"	7FH	List the root keys under a HKEY of the system registry.
"reg_keys_get"	80H	Navigate a sub-key of the system registry.
"reg_val_edit"	81H	Add a value into the system registry.
"reg_val_del"	82H	Delete a value from the system registry.
"reg_key_add"	83H	Add a sub-key into the system registry.
"reg_key_del"	84H	Delete a sub-key from the system registry.
"ddos_start"	85H	Start a DDOS attack from the victim's device.
"ddos_stop"	86H	Stop the DDOS attack.
"bypass"	87H	Attempt the UAC bypass using exploit.
"prc_protect"	88H	Protect a process.
"wd_kill"	89H	Kill the Windows defender service.
"autoruns_req"	92H	Collect a list of auto run progress from the system registry.
"autoruns_data"	93H	Request the auto run data.
"autoruns_del"	94H	Delete an auto-run item.
"s_list"	95H	List the tasks from the system Task Scheduler of the infected system.
"task_del"	96H	Delete a task from the system Task Scheduler.
"spread"	97H	Spread usb.
"bg_change"	98H	Change the desktop background of the victim's desktop.
"scr_off"	99H	Turn off screen.

"browsers_clear"	9BH	Close the browsers such as Chrome, Firefox, Edge, Opera, IE, Vivaldi, Brave, Chromium, Torch, UCBrowser and clean its data.
"notes_get"	9CH	Obtain notes that were set to the victim’s device.
"notes_set"	9DH	Set notes to the victim device.
"website_open"	9FH	Open a website with the default web browser on the victim's device.
"vol_edit"	A0H	Change master volume.
"msgbox"	A3H	Display the victim a message box with a message.
"clipboard_get"	A2H	Obtain the system clipboard data from the victim's system.
"injdll"	A4H	Inject a dll into a specified process or all processes.
"chat_start"	A9H	Pop up a chatting box to the victim.
"chat_msg"	AAH	Chat with the victim using the chatting box.
"chat_stop"	ABH	Stop chatting.

Table 3.1 — The most control commands of BitRat

Conclusion

In this second part the series, I examined the three fileless malware payloads included in the phishing campaign. I also explained what processes they inject into and execute.

Next, I introduced how these three malware connect to their C2 server and described the structure of the packets sending to the C2 server. I also presented the values in the control command packets sent to the malware clients to control the victim’s device to perform further malicious tasks.

I elaborated the features that the three malware provide and used several examples to prove how the attacker uses them. From my research, you also learned the differences between their features.

I also made three tables to list the control commands with brief descriptions.

Fortinet Protections

Fortinet customers are already protected from this malware by FortiGuard’s Web Filtering, AntiVirus, FortiMail, FortiClient, FortiEDR services and CDR (content disarm and reconstruction) feature, as follows:

All relevant URLs have been rated as "Malicious Websites" by the FortiGuard Web Filtering service.

The phishing email attached Excel document can be disarmed by the FortiGuard CDR (content disarm and reconstruction) feature.

The captured Excel sample at the beginning and the downloaded html file as well as the Powershell file with three fileless malware payload files are detected as "VBA/Agent.DDON!tr", "JS/Agent.DDON!tr.dldr" and "PowerShell/Agent.e535!tr" and are blocked by the FortiGuard Antivirus service.

FortiEDR detects both the Excel file and the huge Powershell file as malicious based on its behavior.

In addition to these protections, we suggest that organizations have their end users also go through the FREE NSE training: NSE 1 – Information Security Awareness. It includes a module on Internet threats that is designed to help end users learn how to identify and protect themselves from phishing attacks.

IOCs

URLs:

vncgoga[.]duckdns[.]org:1338 mubbibun[.]duckdns[.]org:999 danseeeee[.]duckdns[.]org:2022 maraipasoo[.]duckdns[.]org:890

Visit part one of this series here.

Learn more about Fortinet’s FortiGuard Labs threat research and intelligence organization and the FortiGuard Security Subscriptions and Services portfolio.