LooCipher: The New Infernal Ransomware

blog.yoroi.company/research/loocipher-the-new-infernal-ransomware/

July 2, 2019

07/02/2019

Introduction

A new Ransomware began to threats the digital world. This time using a nice but scary name: **LooCipher**. The name is at the same time an allusion to its capabilities (thank to the term “Cipher”) and to the popular religious figure, Lucifer. Despite its evocative nickname, the functionalities of this malware are pretty straight forward, not very different from those belonging to many other ransomware families, but digging into its internals we also found elements suggesting its operators could be able to run large scale campaigns.

Technical Analysis

Unlike most ransomware, LooCipher uses a macro-weaponized document as dropper of the real threat. We identified two different document files involved to deploy the ransomware, they are called: *“Info_BSV_2019.docm”* and *“Info_Project_BSV_2019.docm”*. Both files are very poor in design and contain a single text line inviting the user to enable macro execution.

Figure 1. Document content

Exploring the content in-depth, we retrieved its minimal macro code payload: its only purpose is to download the ransomware from the *"hxxp://hcwyo5rfapkytajg.onion[.]pet/2hq68vxr3f.exe"* dropurl and launch it.

The author did not care to obfuscate in any sophisticated way his malicious code, even some comment strings like *"//binary"* and *"//overwrite" are still visible*.

Figure 2. Macro code

Once run, it starts the encryption of all the victim's files, except for the system and programs folders: *"Program Files"*, *"Program Files (x86)"*, *"Windows"*. Obviously, this trick allows to avoid the corruption of the files needed to start the operating system, letting the user login to its PC to see the ransom request.

Figure 3. Ransomware excluded folders

After a long files enumeration phase, the ransomware encrypts all files ending with the following extensions:

```
.jpg, .jpeg, .xml, .xsl, .wps, .cmf, .vbs, .accdb, .cdr, .svg, .conf, .config, .wb2, .msg, .azw,
.azw1, .azw3, .azw4, .lit, .apnx, .mobi, .p12, .p7b, .p7c, .pfx, .pem, .cer, .key, .der, .mdb,
.htm, .html, .class, .java, .asp, .aspx, .cgi, .php, .py, .jsp, .bak, .dat, .pst, .eml, .xps, .sqllite,
.sql, .jar, .wpd, .crt, .csv, .prf, .cnf, .indd, .number, .pages, .x3f, .srw, .pef, .raf, .rf, .nrw, .nef,
.mrw, .mef, .kdc, .dcr, .crw, .eip, .fff, .iiq, .k25, .crwl, .bay, .sr2, .ari, .srf, .arw, .cr2, .raw, .rwl,
.rw2, .r3d, .3fr, .eps, .pdd, .dng, .dxf, .dwg, .psd, .png, .jpe, .bmp, .gif, .tiff, .gfx, .jge, .tga,
.jfif, .emf, .3dm, .3ds, .max, .obj, .a2c, .dds, .pspimage, .yuv, .3g2, .3gp, .asf, .asx, .mpg,
.mpeg, .avi, .mov, .flv, .wma, .wmv, .ogg, .swf, .ptx, .ape, .aif, .av, .ram, .m3u, .movie,
.mp1, .mp2, .mp3, .mp4, .mp4v, .mpa, .mpe, .mpv2, .rpf, .vlc, .m4a, .aac, .aa3, .amr, .mkv,
.dvd, .mts, .vob, .3ga, .m4v, .srt, .aepx, .camproj, .dash, .zip, .rar, .gzip, .mdk, .mdf, .iso,
.bin, .cue, .dbf, .erf, .dmg, .toast, .vcd, .ccd, .disc, .nrg, .nri, .cdi, .ai, .doc, .docm, .docx,
.dxg, .odb, .odm, .odp, .ods, .odt, .orf, .ppt, .pptm, .pptx, .rtf, .xlk, .xls, .xlsb, .xlsm, .xlsx,
.pdf, .mobi, .epub, .sage
```

During the encryption phase, for each file to be encrypted, the malware creates the encrypted copy of the files but it does not delete the original ones, rather it empties them and forces a 0-byte size.

Figure 4. Example of ciphered file with empty original file

It is not clear if this mechanism derives from buggy code or it is a specific peculiarity of this malware, intentionally introduced by the author.

Figure 5. Actions during encryption phase

When the encryption phase ends, it creates a FAQ folder within victim's desktop reporting the instructions to proceed with ransom payment in a "friendly" Q&A form.

Figure 6. File containing the payment instructions

As stated in the payment instruction file, the victim has only five days to proceed with the payment. After this period, the key will be automatically destroyed, preventing any way to recover the user content. Similar information is also displayed in the image set as background and into the interactive pop-up window.

Figure 7. Background image and pop-up window reporting info about the payment
As soon as the encryption phase is ended, the malicious process contacts its C2 sending information about the infected machine and retrieving the BTC address to display in the pop-up window.

Figure 8. Example of HTTP request to retrieve the BTC address
The C2 is hosted in the TOR Network, at the *"hxxp://hcwyo5rfapkytajg[.]onion"* address, so the malware uses some services which act as proxies between the Darknet and clearnet to easily perform its malicious actions, avoiding the installation of TOR libraries on the victim machine. The abused services are:

The request sent by the malware includes information like the User-ID assigned to the victim machine during the encryption phase *"u=rEui7jhIJk6SaRTyhL08N7h1Sft"* and its public IP address *"i=xxx.xxx.xxx.xxx"*. The C2 server replies specifying the BTC Address the user will pay the requested amount to, for instance *"BTC_ADDR: 16HDCwCuy2R5b7YFCmsidXzHQrvHmT7VHGG*".

We noticed that every time the ransomware contacts its C2 at the *"k.php"* resource, the server generates a new BTC Address. Probably, the backend embeds a BTC wallet factory able to register a new wallet on the blockchain for each ransomware infection. This trick surely allows to make more stealthy BTC transactions, avoiding a huge number of transactions towards the same wallet and hardening the cash flow reconstruction. In the following table we inserted some of the BTC addresses generated by the C2:

1LhT45NdcrRBeFfxp67gcKteKp7K5BR374
1QGq13GGdDtfUiBKLS4Re8fdYlVkK8Zbe
1M1ZS5QfZ3Z3ufFagJ455QDqkMvHJhNkwT
15XWd5ixtznsinWFZ9YEk8HUCaMqcm4SiZ
1AUfa421Huj5Hmh5JDFmg36X8VmJPHy7LS
17BvolK1P1kFQq7BPB4iNocisdqE6sEKkv
1UwSDTuTkbPxQt7zglQVsigQunpxhL9Qk
13YNF7U7VTt9DGw7QNWpTEGCrYEmV2qjcx
1MPKAcpe8pnZubBQgUuw3k8wfkTB6sFYAT
16HDCwCuy2RSb7YFCwwdXzHQrvilmT7VHGG
...

Table. Example of Generated BTC Addresses

However, if the victim machine is offline the ransomware is not able to download the BTC address to display in the window. For this reason, the malware also embeds a fallback addresses list to use when it fails to reach the C2.

Figure 9. Other BTC addresses embedded in ransomware binary

1Ps5Vd9dKWuy9FuMDkec9qquCyTLjc2Bxe
19YmdTjw7ZWHEDac8wWzCNdZT8oXsDedtV
1CrdZvvtzrZTJ78k92XuPizhhgtDxQ8c4B
1JHEqi4QsTWz4gB9qZTACP7JggJzAmf6eA
1Azfk7fWwCRynRk8p7qupLqqaADsjwFm4N

Table. Hardcoded BTC Addresses on sample

An interesting peculiarity of this ransomware is its capability to work both as encryptor and as decryptor. The last answer of the instruction file, in fact, reports that the decryptor software is embedded into the ransomware binary in order to make the decryption process as simple as possible.

In fact, after the payment the victim can click on *“Check Payment”* button included in the pop-up window, and so, if the transaction has been confirmed, the *“DECRYPT”* button will be enabled. Moreover, if the user accidentally closes the pop-up window needed to trigger the decryption, he can download a new copy of the ransomware and use it as decryptor. That copy is hosted on the MEGA repository “hxxps://mega
[.nz/#!KclRVIRY!YrUgGjvldsoTuNZbCOjebAz5La7hbB41nJHk1mlgqZo”.

Clicking the *“Check Payment”* button, the process sends a new HTTP request to its C2 to *“/d.php”* in order to check if the payment related to the specific User-ID has been received.

Figure 10. Example of HTTP request to check if the payment has been executed
In the specific case, the server replies with the “0” value, indicating the payment has not been approved, so the “DECRYPT” button will not be enabled. Moreover, if the contacted server is down, the malware tries to reach its TOR C2 using one of the other above-mentioned proxies, avoiding proxy service failures.

Figure 11. HTTP-TOR proxy services used by the malware

Conclusion

In the nowadays, Ransomware is one of the quickest ways to monetize cyber-criminal activities, and for this reason a wide-range of threat actors, including micro cyber-criminals, leverage these "tools" to threaten organizations and companies. LooCipher is a new entry in this sector: it's a Ransomware family spreading through malicious emails embedding infected Office documents, differently from the recent Sodinokibi campaign that used redirectors to land the victims on Exploit Kits infected pages (eg RIG EK).

LooCipher encrypts all files on victim computer, it abuses Clearnet-to-Tor proxy services to connect to its Command and Control hidden behind onion sites. Cybaze-Yoroi ZLAB advises to always keep a recent, tested and offline backup of all the business critical data.

Indicators of Compromise

Hashes:

- ff24d9575694ae2a1e6a6101a2dbaa95dd1ab31b44a3931f6d6a62bbf5be2cbd
- e824650b66c5cdd8c71983f4c4fc0e1ac55cd04809d562f3b6b4790a28521486
- 43cfb0a439705ab2bd7c46b39a7265ff0a14f7bd710b3e1432a9bdc4c1736c49
- 924cc338d5d03f8914fe54f184596415563c4172679a950245ac94c80c023c7d

DropURL:

- hxxp://hcwyo5rfapkytajg[.]onion/2hq68vxr3f.exe
- hxxp://hcwyo5rfapkytajg[.]onion/3agpke31mk.exe
- hxxp://hcwyo5rfapkytajg[.]onion/Info_BSV_2019.docm
- hxxp://hcwyo5rfapkytajg[.]onion/Info_Project_BSV_2019.docm
- hxxp://hcwyo5rfapkytajg.onion[.]pet/2hq68vxr3f.exe
- hxxp://hcwyo5rfapkytajg.onion[.]pet/3agpke31mk.exe
- hxxp://hcwyo5rfapkytajg.onion[.]pet/Info_BSV_2019.docm
- hxxp://hcwyo5rfapkytajg.onion[.]pet/Info_Project_BSV_2019.docm
- hxxps://hcwyo5rfapkytajg.darknet[.]to/2hq68vxr3f.exe
- hxxps://hcwyo5rfapkytajg.darknet[.]to/3agpke31mk.exe
- hxxps://hcwyo5rfapkytajg.darknet[.]to/Info_BSV_2019.docm
- hxxps://hcwyo5rfapkytajg.onion[.]sh/2hq68vxr3f.exe
- hxxps://hcwyo5rfapkytajg.onion[.]sh/3agpke31mk.exe
- hxxps://hcwyo5rfapkytajg.onion[.]sh/Info_BSV_2019.docm
- hxxps://hcwyo5rfapkytajg.onion[.]ws/2hq68vxr3f.exe
- hxxps://hcwyo5rfapkytajg.onion[.]ws/3agpke31mk.exe
- hxxps://hcwyo5rfapkytajg.onion[.]ws/Info_BSV_2019.docm
- hxxps://hcwyo5rfapkytajg.tor2web[.]xyz/2hq68vxr3f.exe
- hxxps://hcwyo5rfapkytajg.tor2web[.]xyz/3agpke31mk.exe
- hxxps://hcwyo5rfapkytajg.tor2web[.]xyz/Info_BSV_2019.docm

C2s:

- hxxp://hcwyo5rfapkytajg.onion[.]pet/k.php
- hxxp://hcwyo5rfapkytajg.onion[.]pet/d.php
- hxxps://hcwyo5rfapkytajg.darknet[.]to/k.php
- hxxps://hcwyo5rfapkytajg.darknet[.]to/d.php
- hxxps://hcwyo5rfapkytajg.onion[.]sh/k.php
- hxxps://hcwyo5rfapkytajg.onion[.]sh/d.php
- hxxps://hcwyo5rfapkytajg.onion[.]ws/k.php
- hxxps://hcwyo5rfapkytajg.onion[.]ws/d.php
- hxxps://hcwyo5rfapkytajg.tor2web[.]xyz/k.php
- hxxps://hcwyo5rfapkytajg.tor2web[.]xyz/d.php

Yara Rules

```
import "pe"
rule LooCipher_dropper_1906 {
meta:
        description = "Yara Rule for LooCipher ransomware .docm dropper"
        author = "Cybaze - Yoroi ZLab"
        last_updated = "2019-06-21"
        tlp = "white"
        category = "informational"
strings:
        $s1 = {FF FD 72 77 6D 3A 3F 96 45 70 00 63 85 92 19 8A}
        $s2 = {35 58 34 CB AF AF 52 A6 13 A6 0C BC 18 A5 C1 38}

        $a1 = { 50 4B 03 04 }

condition:
        $a1 and 1 of ($s*)
}

rule LooCipher_1906 {
meta:
        description = "Yara Rule for LooCipher ransomware"
        author = "Cybaze - Yoroi ZLab"
        last_updated = "2019-06-21"
        tlp = "white"
        category = "informational"
strings:
        $s1 = ".lcphr"
        $s2 = "hcwyo5rfapkytajg"
        $s3 = "LooCipher_wallpaper.bmp"

        $a1 = { 4D 5A }

condition:
        $a1 and 1 of ($s*) and pe.sections[6].name == ".00cfg"
}
```

This blog post was authored by Antonio Farina, Antonio Pirozzi and Luca Mella of Cybaze-Yoroi Z-LAB