VIRLOCK

BCSC-MALWARE-VIRLOCK

TLP:WHITE

www.basquecybersecurity.eus

Agosto 2021

TABLA DE CONTENIDO

Cláusula de exención de responsabilidad

El presente documento se proporciona con el objeto de divulgar las alertas que el BCSC considera necesarias en favor de la seguridad de las organizaciones y de la ciudadanía interesada. En ningún caso el BCSC puede ser considerado responsable de posibles daños que, de forma directa o indirecta, de manera fortuita o extraordinaria pueda ocasionar el uso de la información revelada, así como de las tecnologías a las que se haga referencia tanto de la web de BCSC como de información externa a la que se acceda mediante enlaces a páginas webs externas, a redes sociales, a productos de software o a cualquier otra información que pueda aparecer en la alerta o en la web de BCSC. En todo caso, los contenidos de la alerta y las contestaciones que pudieran darse a través de los diferentes correos electrónicos son opiniones y recomendaciones acorde a los términos aquí recogidos no pudiendo derivarse efecto jurídico vinculante derivado de la información comunicada.

Cláusula de prohibición de venta

Queda terminantemente prohibida la venta u obtención de cualquier beneficio económico, sin perjuicio de la posibilidad de copia, distribución, difusión o divulgación del presente documento.

SOBRE EL BCSC

El Centro Vasco de Ciberseguridad (Basque Cybersecurity Centre, BCSC) es la entidad designada por el Gobierno Vasco para elevar el nivel de madurez de la ciberseguridad en Euskadi.

Es una iniciativa transversal que se enmarca en la Agencia Vasca de Desarrollo Empresarial (SPRI), sociedad dependiente del Departamento de Desarrollo Económico, Sostenibilidad y Medio Ambiente del Gobierno Vasco. Así mismo, involucra a otros tres Departamentos del Gobierno Vasco: el de Seguridad, el de Gobernanza Pública y Autogobierno, y el de Educación, y a cuatro agentes de la Red Vasca de Ciencia, Tecnología e Innovación: Tecnalia, Vicomtech, Ikerlan y BCAM.

El BCSC es la entidad de referencia para el desarrollo de la ciberseguridad y de la confianza digital de ciudadanos, empresas e instituciones públicas en Euskadi, especialmente para los sectores estratégicos de la economía de la región.

La misión del BCSC es por tanto promover y desarrollar la ciberseguridad en la sociedad vasca, dinamizar la actividad empresarial de Euskadi y posibilitar la creación de un sector profesional que sea referente. En este contexto se impulsa la ejecución de proyectos de colaboración entre actores complementarios en los ámbitos de innovación tecnológica, investigación y transferencia tecnológica a la industria de fabricación avanzada y otros sectores.

Así mismo, ofrece diferentes servicios en su rol como Equipo de Repuesta a Incidentes (en adelante CERT, por sus siglas en inglés “Computer Emergency Response Team”) y trabaja en el ámbito de la Comunidad Autónoma del País Vasco para aumentar la capacidad de detección y alerta temprana de nuevas amenazas, la respuesta y análisis de incidentes de seguridad de la información, y el diseño de medidas preventivas para atender a las necesidades de la sociedad vasca. Con el fin de alcanzar estos objetivos forma parte de diferentes iniciativas orientadas a la gestión de incidentes de ciberseguridad:

1. RESUMEN EJECUTIVO

Virlock es una familia de virus informático con capacidades de *screenlocker* y *ransomware* que fue descubierto por primera vez en 2014. El malware posee una naturaleza polimórfica, lo que quiere decir que cada instancia generada es diferente desde una perspectiva heurística. Una estrategia pensada para evitar detecciones de soluciones antivirus basados en firmas y cuya ejecución genera tres instancias de sí mismo, donde cada una es diferente y que, además, llevan a cabo funcionalidades diferentes, todo con el fin de evadir los sistemas de detección implantados en el equipo.

A diferencia del resto de ransomware, Virlock no solo cifra los ficheros de la víctima, sino que también los convierte en la propia amenaza, incrustando su código en cada uno de dichos ficheros de forma que cualquier usuario que abra un archivo infectado también se infecte, cifrando así todos los archivos de este usuario también.

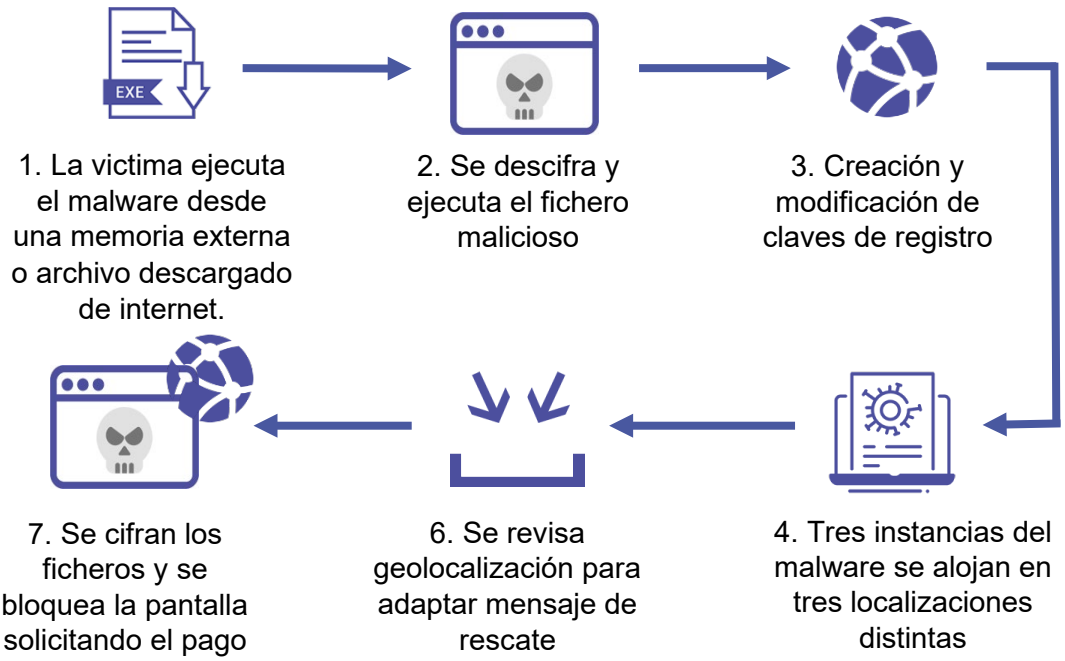
Por tanto, aunque Virlock puede ser propagado por sus autores, al igual que el resto del malware, el verdadero potencial de esta amenaza está en esta funcionalidad de virus y, con el paso de los años se han observado diferentes versiones del malware, lo que demuestra también un desarrollo activo por parte de sus creadores.

La forma en que esta amenaza está programada denota un alto nivel de conocimiento técnico por parte de sus creadores. Sin embargo, parte de su funcionalidad también parece carecer de lógica e incluso existen algunos errores de programación que podrían llevar a considerar a esta amenaza como un experimento malicioso.

2. ANÁLISIS TÉCNICO

2.1. Flujo de infección

Dada sus características de virus, Virlock infecta archivos para propagarse y conseguir que otros usuarios se infecten en caso de abrir alguno de los ficheros infectados. No obstante, al igual que otras amenazas, este virus también puede tratar de ser ejecutado por parte de sus autores, aunque no se conocen métodos específicos utilizados para ello.

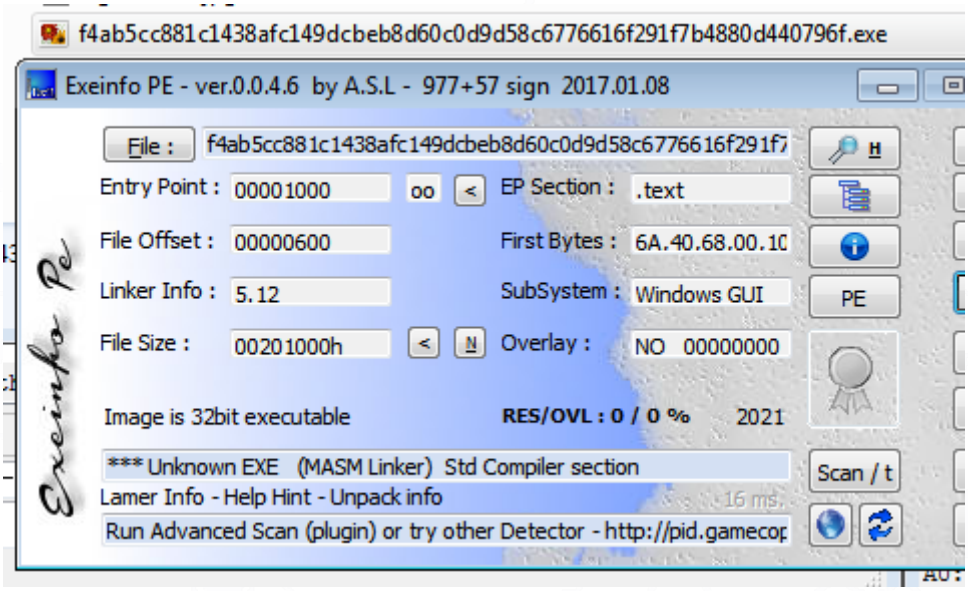


Una vez ejecutado, Virlock descifra y abre el fichero original que hubiera infectado, pero paralelamente, infecta la máquina víctima mediante la ejecución de tres instancias del propio virus que es copiado a tres localizaciones diferentes de forma que, cada una de las copias, se encarga de una tarea diferente en el flujo de infección: una se encarga de buscar ficheros a infectar en el equipo, otra se encarga de bloquear la pantalla y, la última, establece persistencia en el equipo.

2.2. Análisis técnico

El análisis está basado en una muestra cuya firma SHA-256 es:

f4ab5cc881c1438afc149dcbeb8d60c0d9d58c6776616f291f7b4880d440796f



Se trata de un ejecutable para plataformas Windows de *32 bits* con únicamente dos secciones y compilado directamente desde código ensamblador con el compilador MASM.

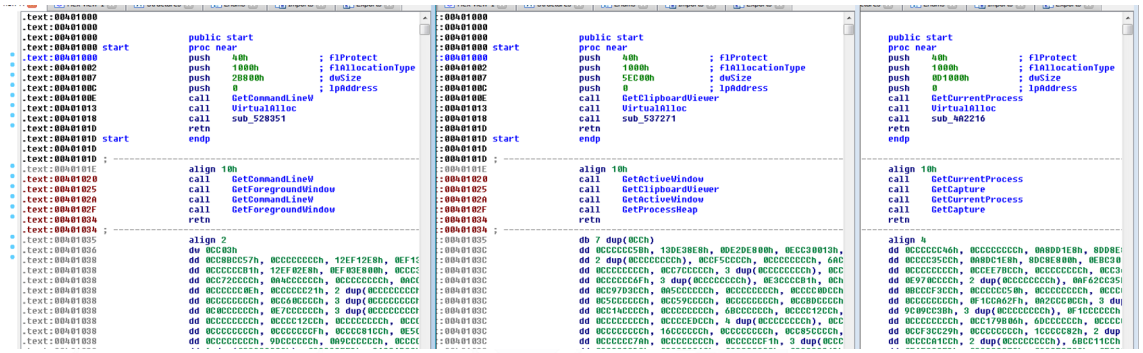
Tras iniciar su análisis se ha comprobado que la muestra se encuentra altamente ofuscada, lo que dificulta la tarea de ingeniería inversa.

2.1.1. Packer y algoritmos metamórfico y polimórfico

El binario analizado se encuentra altamente ofuscado e incluye diversas técnicas de anti-debugging como la comprobación del flag “BeingDebugged” mediante el PEB, técnicas que se pueden saltar mediante el uso de plugins como ScyllaHide.

Además, tras comparar diferentes muestras de la familia se puede observar que, aunque el código puede parecer similar, varía en función de la copia gracias a su capacidad metamórfica con la que se generan códigos de operación diferentes pero que terminan realizando las mismas acciones.

Virlock se reconstruye tras cada ejecución y, como se puede observar decora el núcleo de la funcionalidad inicial con llamadas a API aleatorias de módulos elegidos al azar. El malware elige un número aleatorio de bibliotecas que importará el fichero resultante para la infección futura, y de esas bibliotecas, algunas API aleatorias dentro de cada una de ellas se eligen como importaciones.

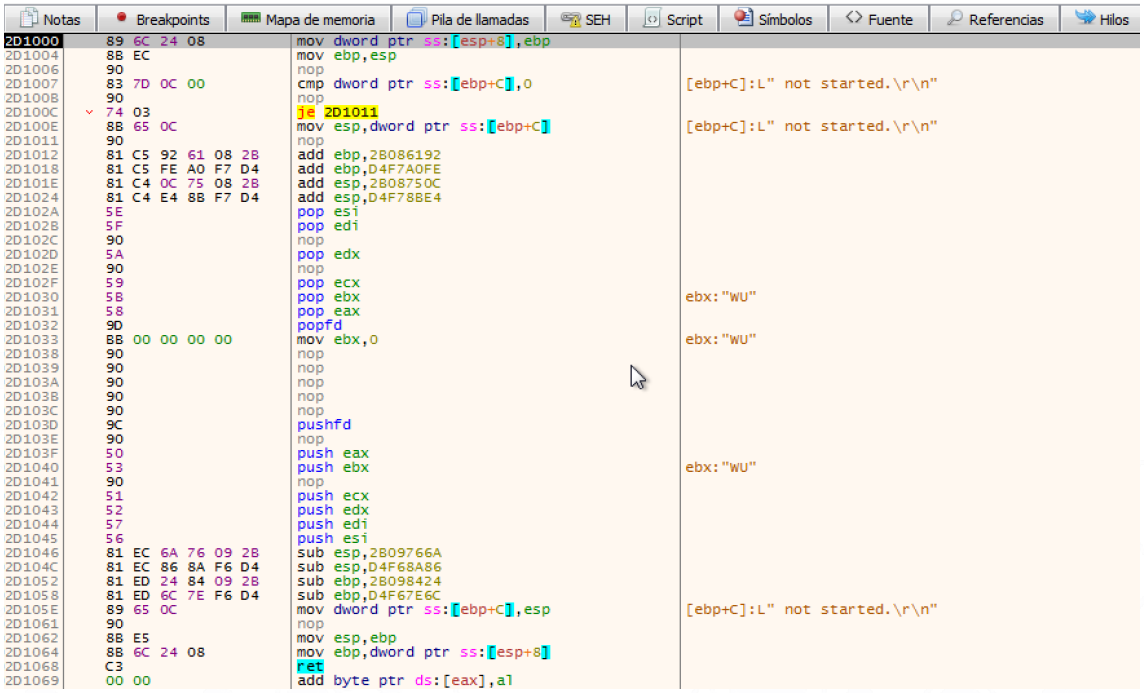


Cuando el binario de Virlock es cargado en memoria, la única sección de código no cifrada es la que puede apreciarse en el desensamblado de IDA. El resto del código, datos y, el archivo original (en caso de que se trate de un fichero infectado) se encuentran cifrados.

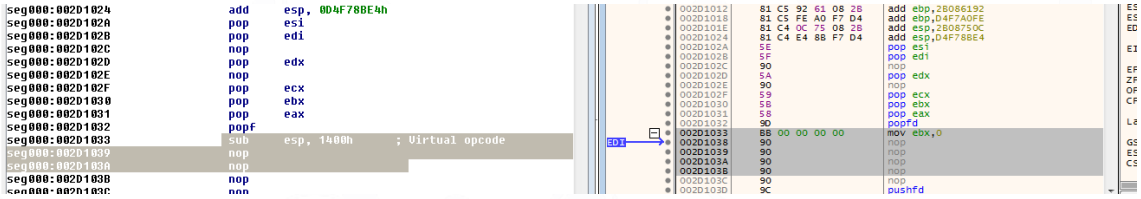
Tras reservar memoria de forma dinámica, el binario transfiere la ejecución a una función encargada de descifrar y cargar una shellcode que es la responsable de descifrar el resto del binario. Esta rutina de descifrado de la shellcode utiliza un bucle XOR tras lo que realiza un salto a dicha sección de memoria para continuar con la ejecución.

```
:00528356 decode_run_second_stage_528356 proc near
:00528356                                        ; CODE XREF: j_decode_run_second_stage_528356↑j
:00528356                 sub     esp, 36Ch
:0052835C                 mov     esi, offset unk_638DF4
:00528361                 jmp     $+5
:00528366 ; ---------------------------------------------------------------------------
:00528366
:00528366 loc_528366:                            ; CODE XREF: decode_run_second_stage_528356+B↑j
:00528366                 mov     edi, offset unk_5321B8
:0052836B                 mov     [edi], eax
:0052836D                 nop
:0052836E                 mov     edi, eax
:00528370                 mov     ebx, edi
:00528372                 nop
:00528373                 jmp     $+5
:00528378 ; ---------------------------------------------------------------------------
:00528378
:00528378 loc_528378:                            ; CODE XREF: decode_run_second_stage_528356+1D↑j
:00528378                 mov     ecx, 36Ch
:0052837D                 mov     edx, 0D5h
:00528382                   |
:00528382 next_byte_528382:                      ; CODE XREF: decode_run_second_stage_528356+50↓j
:00528382                 mov     al, [esi]
:00528384                 jmp     decrypt_loop_528397
:00528389 ; ---------------------------------------------------------------------------
:00528389
:00528389 call_second_stage_528389:              ; CODE XREF: decode_run_second_stage_528356+56↓j
:00528389                 call    ebx
:0052838B                 add     esp, 36Ch
:00528391                 retn
:00528392 ; ---------------------------------------------------------------------------
:00528392                 jmp     $+5
:00528397 ; ---------------------------------------------------------------------------
:00528397
:00528397 decrypt_loop_528397:                   ; CODE XREF: decode_run_second_stage_528356+2E↑j
:00528397                                        ; decode_run_second_stage_528356+3C↑j
:00528397                 xor     al, dl
:00528399                 mov     [edi], al
:0052839B                 inc     edx
:0052839C                 nop
:0052839D                 inc     esi
:0052839E                 nop
:0052839F                 inc     edi
:005283A0                 nop
:005283A1                 dec     ecx
:005283A2                 cmp     ecx, 0
:005283A5                 nop
:005283A6                 jnz     next_byte_528382
:005283AC                 jmp     call_second_stage_528389
:005283AC decode_run_second_stage_528356 endp
```

Esta shellcode se encarga de ir descifrando diferentes códigos de instrucciones (opcodes) que son inyectados de forma dinámica.



Los opcodes son inyectados entre los offsets 0x1032 y 0x103B de la shellcode que cambian con cada llamada a la función



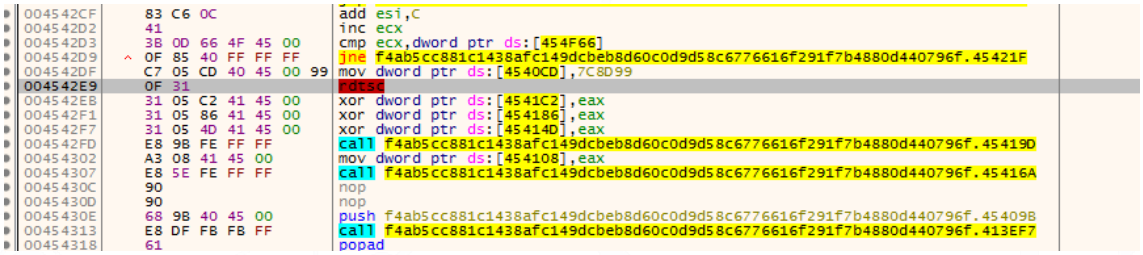
Tras realizar este proceso, se transfiere la ejecución de vuelta al binario. La mayoría de malware utiliza un algoritmo de descifrado simple para descifrar el código de malware en memoria, lo que da a los investigadores la oportunidad de leer y analizar el código de malware completo una vez es desempaquetado y, aunque se utilicen claves de cifrado diferentes se seguirá generando el mismo código descifrado al completo en memoria.

Sin embargo, en este caso, tanto el código del malware, como el fichero original continúan cifrados. El cuerpo de cada una de las funciones que ejecuta Virlock contiene un pequeño código de cifrado y de descifrado al principio y final de la función. De esta forma, Virlock solo descifra el código que necesita en un momento dado y luego lo vuelve a cifrar y, para complicar aún más las cosas, utiliza una clave diferente cuando lo vuelve a cifrar en la memoria.

Esto significa que los investigadores no pueden extraer todo el malware de una vez para poder analizarlo en un desensamblador. No es posible volcar una copia completamente descifrada del malware a disco porque solo se descifra un bloque de código en un momento dado. Por tanto, se complica enormemente la tarea de analizar la funcionalidad de este malware.

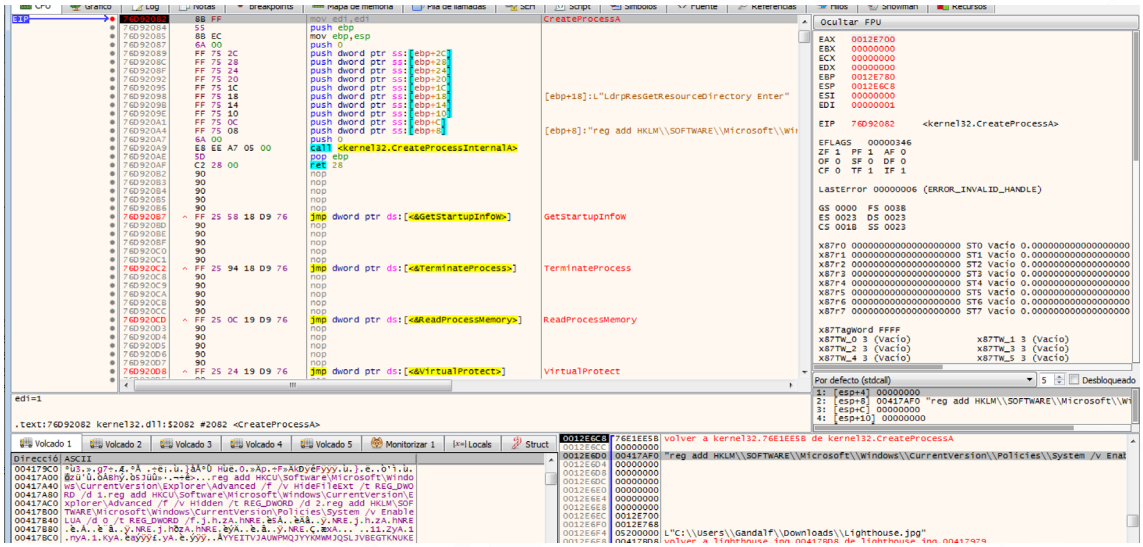
Cuantas más secciones del código de Virlock se ejecuten con su serie continua de patrones de descifrado y cifrado, más diferente se verá el malware en la memoria con respecto a la copia cifrada original antes de que se comenzara a

ejecutar. De esta forma, generar una copia diferente de sí mismo consiste en copiar el contenido de la memoria una vez ejecutado el malware en un archivo. Para conseguir generar claves de forma aleatoria, el malware utiliza la instrucción “RDTSC”.



2.1.2. Modificación de claves de registro

Virlock utiliza la herramienta interna de Windows “reg.exe” para modificar diferentes claves de registro que le permitan pasar desapercibido al usuario en el sistema y continuar su ejecución.



En concreto, las acciones que realiza y claves del registro que accede son las siguientes:

- No mostrar archivos ocultos:

  HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Advanced\Hidden

- Ocultar extensions de fichero:

  HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Advanced\HideFileExt

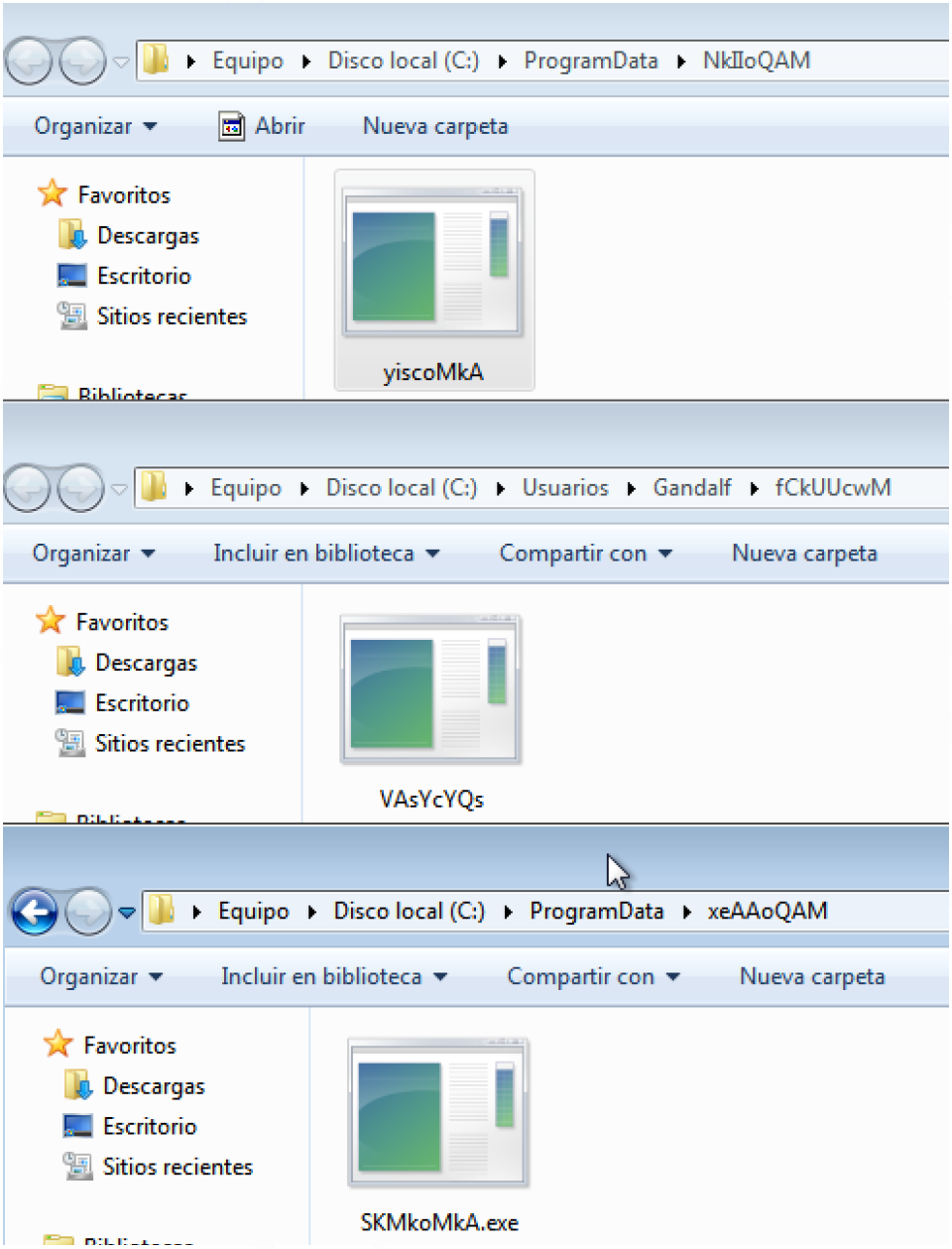
- Desactivar UAC:

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System\EnableLUA

2.1.3. Ejecución por instancias

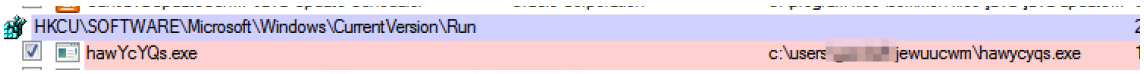
Como se ha comentado, el malware escribe al menos, tres instancias diferentes del virus para llevar a cabo su ejecución. Cada una de estas instancias, pese a contener la misma funcionalidad en su totalidad, reconoce de forma interna que debe realizar una u otra. La primera y segunda copia se realizan en rutas aleatorias dentro de la carpeta %AllUserProfile%. La tercera copia, al igual que las dos anteriores, genera una ruta aleatoria, pero dentro de la carpeta %UserProfile% esta vez.

La primera instancia es la encargada de escanear el equipo e infectar los ficheros objetivo, la segunda se encarga de establecer persistencia en el equipo y la tercera lanza la pantalla de bloqueo con la nota de rescate.



2.1.3. Persistencia

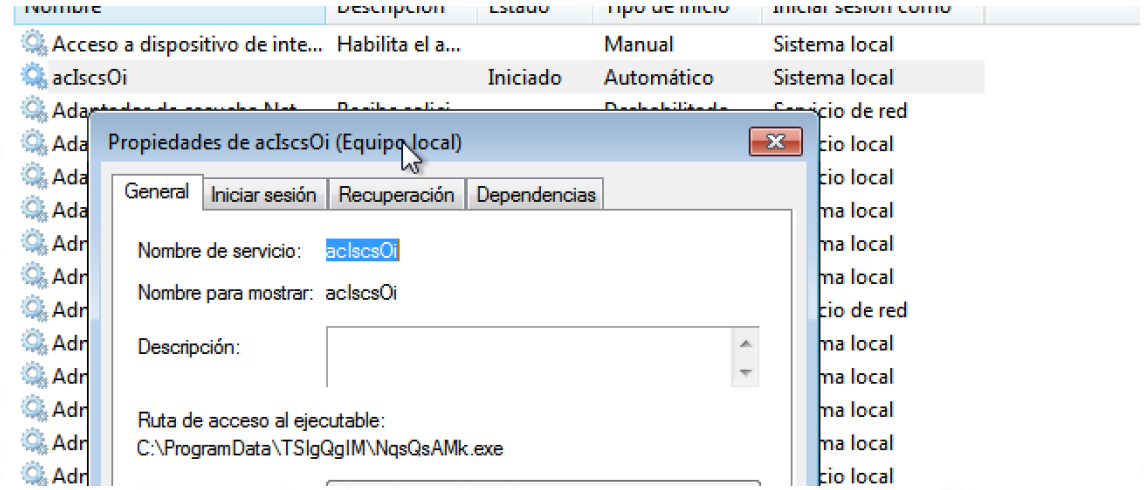
Virlock modifica el registro del equipo y establece un valor para una de las instancias de sí mismo con nombres aleatorios en la clave de registro "CurrentVersion\Run", lo cual hace que se ejecute cada vez que un usuario inicia sesión.



Para otra de las instancias, Virlock hace uso de la clave userinit de HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon para establecer persistencia.

```
C:\Windows\system32\userinit.exe,C:\ProgramData\soUMkEAU\qwscEAUQ.exe,
userinit.exe,C:\ProgramData\soUMkEAU\qwscEAUQ.exe,
```

Por último, para la otra instancia, crea un servicio con un nombre aleatorio.



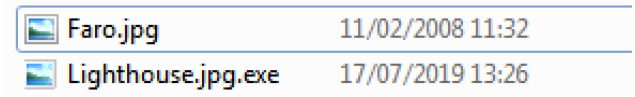
2.1.4. Infección de ficheros

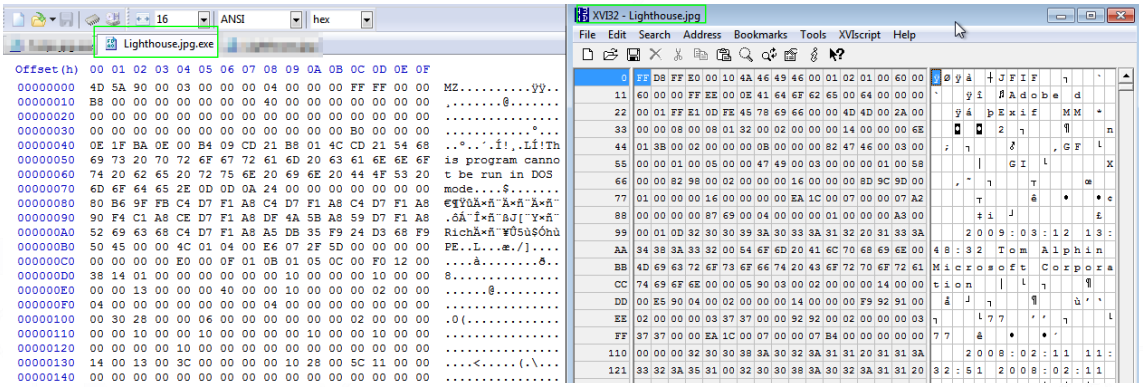
Después de copiar las tres instancias de sí mismo, la primera de ellas es iniciada mediante CreateProcess y se encarga de buscar tipos de archivos específicos para infectar. Una vez que se ha identificado el archivo de destino, el malware cifra el archivo y luego lo reemplaza con una copia del código del propio virus con el contenido del archivo original cifrado adjunto.

Virlock contiene una serie de extensiones que el virus busca para infectar, entre las que se incluyen las siguientes:

.exe, .doc, .xls, .pdf, .ppt, .mdb, .zip, .rar, .mp3, .mpg, .wma, .png, .gif, .bmp, .jpg, .jpeg, .psd, .p12, .cer, .crt, .p7b, .pfx, .pem.

Tras infectar un fichero, el malware le añade la extensión .exe al final, pero gracias a la modificación del registro donde oculta la extensión del fichero y la sustitución del logo del binario por el correspondiente al tipo de fichero original, en el caso de ficheros que no sean del tipo ".exe" (para los que se mantiene el icono legítimo), permite pasar oculto ante los ojos del usuario.



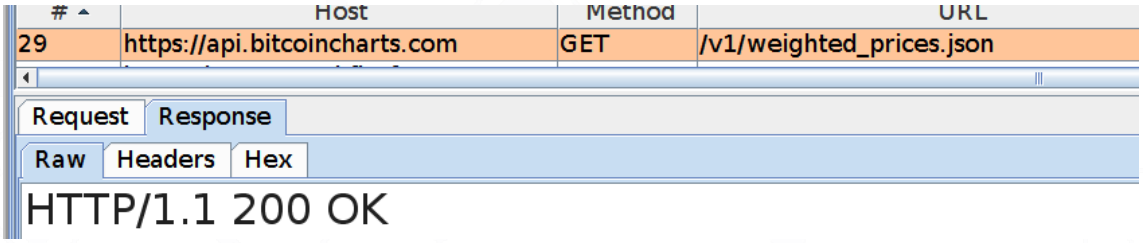


A diferencia de otros ransomware que se basan en la combinación de cifrado simétrico y asimétrico para proteger los ficheros de ser descifrados sin la intervención de los creadores del malware, Virlock únicamente realiza un cifrado simétrico de los ficheros basado en operaciones XOR y ROL, por lo que es posible restaurar los ficheros revirtiendo el algoritmo. La compañía de seguridad ESET ha creado para ello un descifrador que puede descargarse desde https://descargas.eset.es/virlock-cleaner.

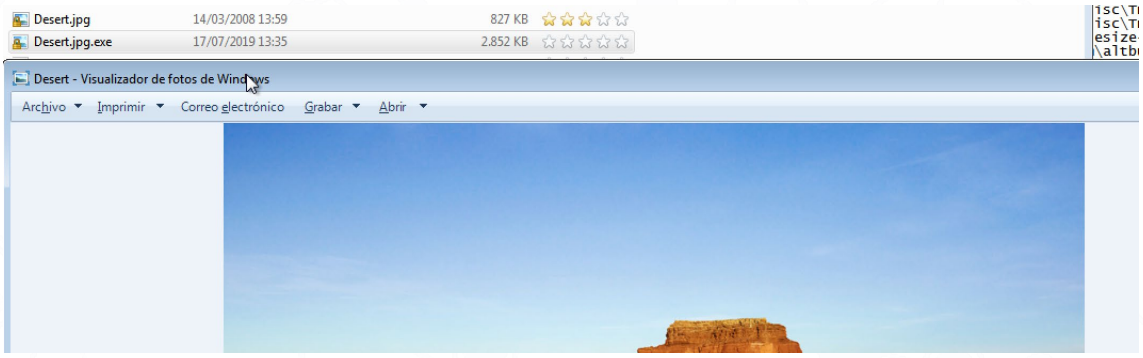
2.1.5. Bloqueo de pantalla y nota de rescate

Mientras se ejecuta la rutina de cifrado de ficheros, otra instancia aguarda para, una vez finalice el proceso, bloquear la pantalla de la máquina víctima. Antes de hacerlo, Virlock termina el proceso explorer.exe y el administrador de tareas, si se estuviera ejecutando. Además, mediante la clave de registro HKEY_CURRENT_USER\Control Panel\International\Geo comprueba la geolocalización del usuario para adaptar el mensaje a mostrar, tras lo cual se lanza la ventana de bloqueo.

El código de transferencia que el programa espera que se introduzca en el campo “Transfer ID” es una cadena de tamaño 64 caracteres. Se ha comprobado que, cualquier cadena de este tamaño introducida en el cuadro de texto será aceptada como válida por el malware, cerrando así la ventana y permitiendo volver a utilizar el sistema. El único requisito, es que el ordenador disponga de conexión a internet, la cual comprueba mediante el estado de las interfaces de red y una petición posterior al dominio “api.bitcoincharts.com”.



Una vez hecho esto, si se presiona el botón "PAY FINE" desaparecerá la pantalla de bloqueo y VirLock creerá ahora que se ha pagado el rescate. Debido a esto, cualquiera de los archivos infectados, al hacer doble clic en ellos para abrirlos, ya no iniciará la ventana de bloqueo, sino que extraerá el archivo original contenido dentro de él, restaurándolo así en la ruta donde se encuentre.



2.2. Técnicas MITRE ATT&CK

Initial Access	T1091	Replication Through Removable Media
Execution	T1106	Native API
	T1569	System Services
	T1204	User Execution
Persistence	T1547	Boot or Logon Autostart Execution
	T1554	Compromise Client Software Binary
	T1543	Create or Modify System Process
Privilege Escalation	T1547	Boot or Logon Autostart Execution
	T1543	Create or Modify System Process
Defense Evasion	T1222	File and Directory Permissions Modification
	T1564	Hide Artifacts
Discovery	T1083	File and Directory Discovery
	T1012	Query Registry
	T1614	System Location Discovery
	T1016	System Network Configuration Discovery
Lateral Movement	T1091	Replication Through Removable Media
Command and Control	T1001	Data Obfuscation
Impact	T1486	Data Encrypted for Impact

En el Apéndice A se puede consultar el mapa de tácticas y técnicas utilizadas por Virlock.

3. MITIGACIÓN

3.1. Medidas a nivel de endpoint

La medida principal consiste en tener el sistema actualizado y utilizar un programa antimalware.

El código de *Virlock* no está firmado, por lo que implementar una política que no permita la ejecución de binarios que no estén firmados podría prevenir la ejecución de este *ransomware* y de otro tipo de malware. No obstante, gran cantidad de desarrolladores y paquetes de software no distribuyen sus productos firmados, por lo que esta estrategia podría no resultar práctica en algunos casos.

En concordancia con lo anterior, pero empleando mecanismos más generales, se recomienda que las organizaciones prohíban o, al menos, monitoricen la ejecución de binarios no conocidos previamente dentro de ella o aquellos no provenientes de fuentes confiables. Aunque imperfecto, por la forma en la que se crea y distribuye el software legítimo, esta medida puede servir como una alarma inicial para impulsar una mayor investigación y, posiblemente, limitar su propagación.

3.2. Medidas a nivel de red

Si se dispone de mecanismos para inspeccionar el tráfico que ocurre dentro de la red, se debería identificar la transferencia de binarios desconocidos.

Por otro lado, es altamente recomendable mantener una segmentación adecuada de la red para evitar desplazamientos laterales y que finalmente se alcancen los sistemas críticos de la organización.

En adición y conociendo el comportamiento de esta familia de *ransomware*, se podría analizar el tráfico interno y generar reglas que comprueben las conexiones a recursos de un servidor y las conexiones a recursos de red, de forma que, si provienen de algún proceso no reconocido, se pueda parar lo antes posible o detectar al menos la intrusión.

3.3. Medidas y consideraciones adicionales

En caso de incidente con este malware, se debe de reportar a las autoridades competentes lo más rápido posible.

A diferencia de otros ransomware donde la ejecución de la amenaza viene de forma posterior a un proceso de descubrimiento e intrusión en la red de la empresa, en el caso de Virlock una organización puede verse simplemente afectada mediante la ejecución de un fichero infectado por parte de alguno de sus empleados desde una memoria externa o archivo descargado de internet.

4. INDICADORES DE COMPROMISO

Los indicadores de compromiso y reglas de detección también están disponibles para su consulta y descarga en el repositorio público del Basque Cybersecurity Centre:

https://github.com/basquecscentre/technical-reports

4.1. Red

Web consultada por el *screenlocker*:
hxxp://api.bitcoincharts.com/v1/weighted_prices.json

4.2. Hashes

4.2.1. SHA256:

f4ab5cc881c1438afc149dcbeb8d60c0d9d58c6776616f291f7b4880d440796f

4.3. YARA rules

Esta regla sirve para identificar algunas de las muestras generadas durante el análisis. Sin embargo, dada la naturaleza polimórfica del virus, es muy complicado generar una regla genérica que pueda detectar cualquier muestra de este malware.

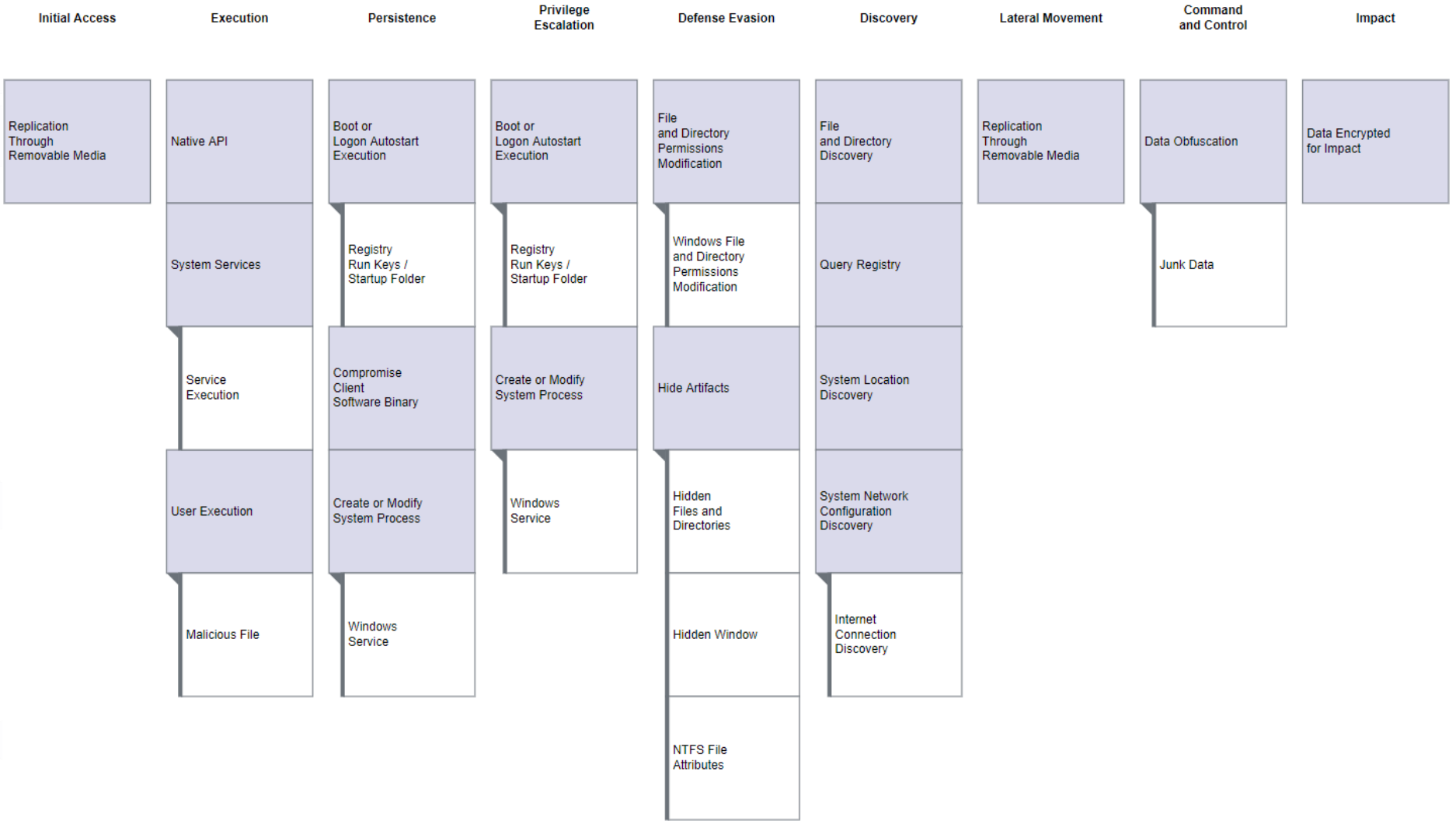
```
rule Virlock {
   strings:
      $op0 = { 6a 40 68 00 10 00 00 68 00 }
      $op1 = { e9 00 00 00 00 81 ec }
      $op2 = { 03 00 00 be }

      $s1 = "kernel32.dll" ascii
      $s2 = "user32.dll" ascii
      $s3 = "!This program cannot be run in DOS mode."
   condition:
      uint16(0) == 0x5a4d and filesize < 6000KB and all of ($s*) and
      all of ($op*)
}
```

5. REFERENCIAS ADICIONALES

- https://blogs.blackberry.com/en/2019/07/threat-spotlight-virlock-polymorphic-ransomware
- https://www.welivesecurity.com/2014/12/22/win32virlock-first-self-reproducing-ransomware-also-shape-shifter/
- https://www.knowbe4.com/virlock-ransomware
- https://www.virusradar.com/en/Win32_Virlock/detail
- https://blog.trendmicro.com/trendlabs-security-intelligence/virlock-combines-file-infection-and-ransomware/
- https://www.virusbulletin.com/virusbulletin/2016/12/vb2015-paper-its-file-infector-its-ransomware-its-virlock/
- https://www.linkedin.com/pulse/weird-ransomware-strain-spreads-like-virus-cloud-stu-sjouwerman/
- https://blog.malwarebytes.com/threat-analysis/2017/01/virlockers-comeback-including-recovery-instructions/
- https://www.fortinet.com/blog/threat-research/real-time-polymorphic-code-in-ransomware
- https://insomnihack.ch/wp-content/uploads/2017/04/RA_metamorphic_malware.pdf

APÉNDICE A: MAPA DE TÉCNICAS MITRE ATT&CK



Reportar incidente

Si has detectado algún incidente de ciberseguridad, avísanos para que tomemos las medidas oportunas para evitar su propagación.

900 104 891

incidencias@bcsc.eus

Catálogo de ciberseguridad

¿Necesitas ayuda con tu ciberseguridad o la de tu empresa?