Basque CyberSecurity Centre

Akira Ransomware

BCSC-MALWARE-AKIRA

TLP: CLEAR

www.ciberseguridad.eus

EUSKO JAURLARITZA GOBIERNO VASCO

GRUPO spri TALDEA

Índice

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:

FIRST Improving Security Together
TF-CSIRT Trusted Introducer
CSIRT.es

Resumen ejecutivo

Akira es un malware de tipo *ransomware* identificado por primera vez en marzo de 2023. El grupo que opera este *ransomware* ha estado activo desde entonces realizando diferentes campañas donde ha impactado a más de 46 víctimas, la mayoría de ellas localizadas en los Estados Unidos. Diversas industrias, incluyendo la educación, finanzas, bienes raíces, BFSI, construcción o salud entre otras, han sido afectadas por estos ataques.

La estrategia del grupo consiste en la doble extorsión donde no solo cifran los datos, sino que también exfiltran información sensible, amenazando con venderla o filtrarla públicamente si no se cumple con el pago del rescate. Esta estrategia aumenta las posibilidades de pago de las víctimas y, para apoyarla técnicamente, el grupo cuenta con un sitio web de estilo retro en la red Tor, donde hacen públicos los datos robados si las víctimas no pagan el rescate demandado. Además, en dicho sitio web también ofrecen una función de chat para que las víctimas puedan comunicarse con ellos utilizando un ID único que incluyen para cada una en la nota de rescate.

Una peculiaridad destacada de Akira es que parte de su código se basa en la filtración del código fuente de otro *ransomware* conocido como Conti. Esto muestra cómo los grupos de *ransomware* pueden aprovecharse del código de otros malware para mejorar y ampliar sus operaciones cibernéticas.

Al igual que la mayoría de *ransomware*, Akira utiliza criptografía simétrica y asimétrica para cifrar los ficheros en los equipos de sus víctimas. En concreto, cuando se ejecuta Akira calcula una clave de cifrado y vector de inicialización aleatorios para el algoritmo Chacha20. Estos valores son cifrados con RSA a través de una clave pública que se encuentra embebida en el propio código y que los actores cambian por cada víctima. A diferencia de Conti, en el que se basa su código, y de la mayoría de *ransomware*, Akira calcula una única clave de cifrado que es utilizada para cifrar todos los ficheros. Por tanto, si se averigua esta clave, se podrían llegar a descifrar.

En junio de 2023, investigadores de Avast desarrollaron un descifrador para el *ransomware* Akira, ofreciendo a las víctimas una posibilidad de recuperar sus datos sin tener que pagar el rescate exigido por los ciberdelincuentes. Para poder *crackear* la clave de cifrado utilizada, es necesario contar con la versión original de alguno de los ficheros de la máquina víctima y su versión cifrada por Akira. Además, es importante tener en cuenta que este *ransomware* no está relacionado con otra amenaza con el mismo nombre, pero anterior a esta, descubierta en 2017, para la cual el descifrador no es aplicable.

Ante la continua amenaza que representa Akira, y pese a la publicación de este descifrador, es esencial que las organizaciones refuercen sus medidas de seguridad y conciencien a sus empleados sobre las tácticas empleadas por

estos ciberdelincuentes. La colaboración entre expertos en ciberseguridad, investigadores y organizaciones afectadas sigue siendo fundamental para combatir este tipo de amenazas y proteger la información y los datos sensibles de posibles ataques futuros.

Análisis técnico

Flujo de infección

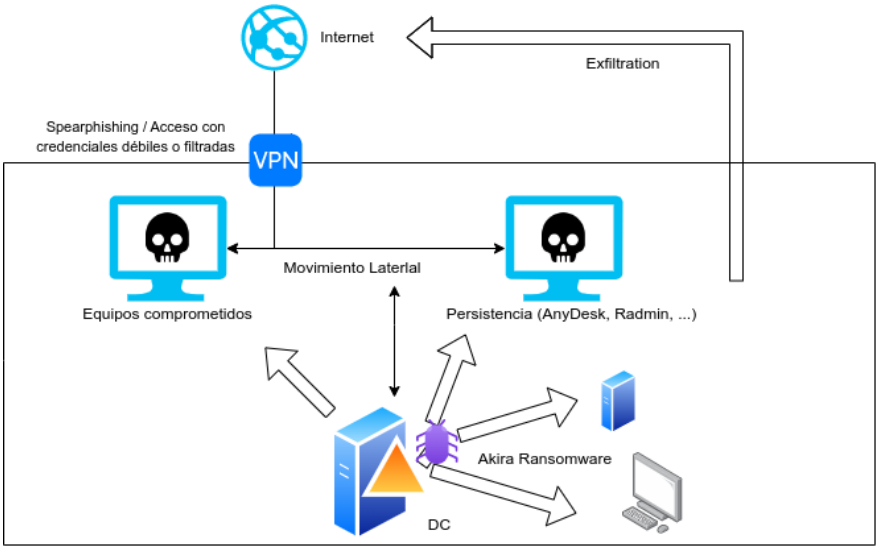


Ilustración 1: Flujo de infección de Akira Ransomware.

El flujo de infección del *ransomware* Akira puede variar para cada organización. A continuación, se describe a grandes rasgos cómo podría ser el proceso de infección, basado en la información de dos incidentes diferentes analizados por la compañía Sophos (https://news.sophos.com/en-us/2023/05/09/akira-ransomware-is-bringin-88-back/).

Para obtener acceso inicial, los atacantes suelen tratar de obtener acceso VPN a la red víctima. Para ello pueden dar con la forma de eludir la Autenticación Multifactor (MFA) o, simplemente obtener acceso VPN con autenticación débil de un solo factor.

Una vez dentro de la red, los atacantes buscan acceder a credenciales de usuario de dominio para moverse lateralmente y adquirir información. Para obtener acceso a las credenciales, utilizan diferentes técnicas como el volcado de la memoria del proceso LSASS o el acceso a archivos de registro de eventos.

El atacante realiza un descubrimiento indirecto para identificar otros sistemas y redes en la organización. Para ello, podría utilizar herramientas como *PCHunter64* o *Advanced IP Scanner*.

Una vez que el atacante ha recopilado información suficiente, puede tratar de mantener acceso remoto persistente en varios sistemas. Para ello, se utilizan programas como *AnyDesk* o *Radmin*.

Finalmente, el atacante ejecuta el *ransomware* Akira para cifrar los archivos y genera una nota de rescate para extorsionar a la víctima. El intervalo de tiempo entre que se produce el compromiso inicial y la ejecución del *ransomware* puede

variar según el caso, siendo el de los casos estudiados por Sophos, de entre 7 días hasta más de 30 días.

Es importante destacar que cada incidente puede presentar diferentes tácticas y técnicas utilizadas por los atacantes, y que las medidas de mitigación para prevenir la infección de *ransomware* Akira incluyen la implementación de autenticación multifactor, la segmentación de cuentas de administrador y la restricción de acceso a herramientas y aplicaciones de escritorio remoto. Además, contar con una solución de seguridad adecuada y configurada correctamente para detectar y prevenir el *ransomware* es esencial para proteger a las organizaciones de estas amenazas.

Portal de Akira en la red TOR

Los actores que operan ***Akira*** cuentan con un sitio en la red **TOR** para enumerar las organizaciones presuntamente afectadas por su *ransomware* y ofrecer enlaces de descarga de los datos recopilados por ellos en caso de no pagar el rescate demandado. El sitio tiene un aspecto retro y para navegar por él es necesario especificar comandos como si de una terminal se tratase. Si se indica el comando "*leaks*" se puede acceder a la descarga de los ficheros de las compañías que aparentemente no han pagado el rescate demandado. Por otra parte, con el comando "*news*" se obtiene un listado de todas las compañías a las que habrían comprometido hasta el momento. Con el comando "*contact*" se le puede enviar un mensaje a los actores.

La dirección actual para acceder a este sitio es la siguiente:

hxxps://akiral2iz6a7qgd3ayp3l6yub7xx2uep76idk3u2kollpj5z3z636bad[.]onion

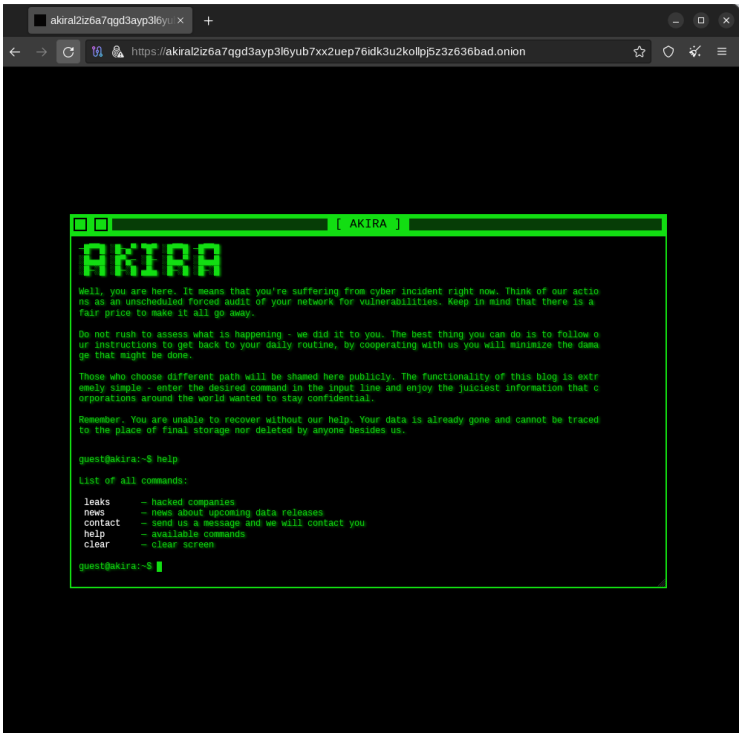


Ilustración 2: Sitio oficial de publicación de leaks de los actores de Akira en la red TOR

Además, los actores cuentan con otro sitio específico destinado a la negociación de los pagos. La dirección de este sitio y el identificador para acceder son especificados en la nota de rescate que deja Akira en los equipos cifrados. En el momento del análisis la dirección del sitio en la red Tor es:

hxxps://akiralkzxzq2dsrzsrvbr2xgbbu2wgsmxryd4csgfameg52n7efvr2id[.]onion

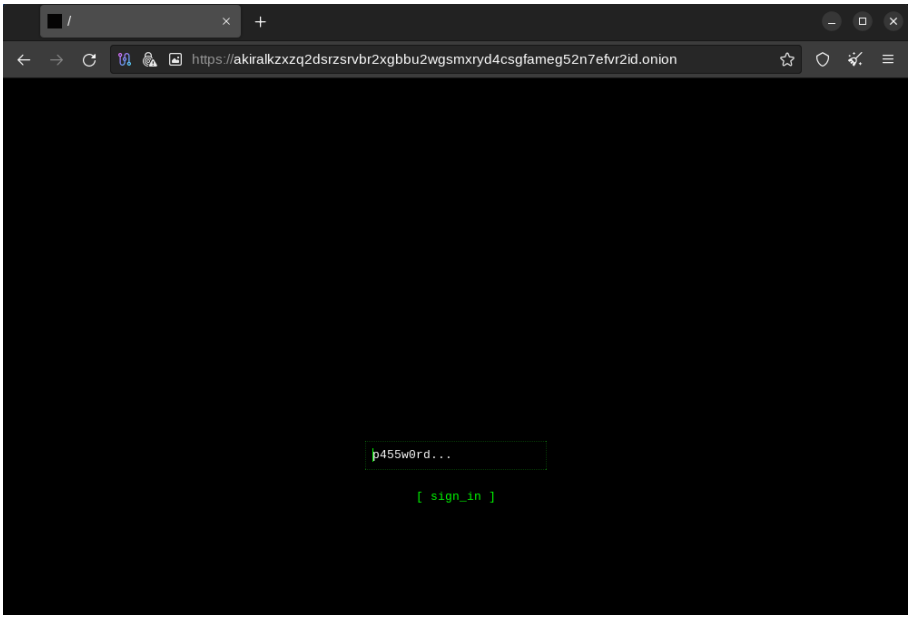


Ilustración 3: Sitio oficial de chat de los actores de Akira en la red TOR

Muestra analizada (Windows)

La muestra analizada corresponde con la versión para Windows de la familia de *ransomware* **Akira**. Se trata de un binario Portable Ejecutable (PE) de Windows de 64 bits cuya firma **SHA256** es la siguiente:

8631ac37f605daacf47095955837ec5abbd5e98c540ffd58bb9bf873b1685a50

El binario está desarrollado con C++ y no parece encontrarse empaquetado mediante ningún software de protección.

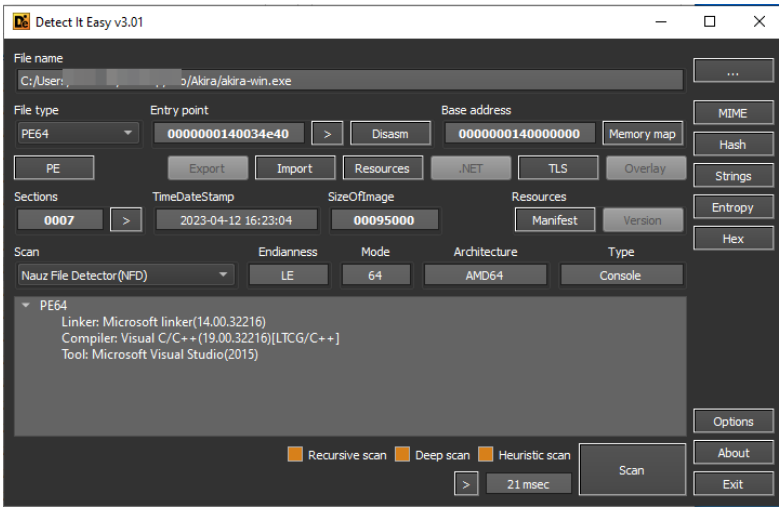


Ilustración 4: Posibles parámetros para su ejecución.

Argumentos de línea de comandos

El *ransomware* está diseñado para ser ejecutado de forma manual y, una vez en funcionamiento, se comporta como una aplicación de consola de comandos. Incluso muestra información durante su ejecución como cuando se producen ciertos errores como los de acceso a ficheros por falta de permisos. El binario puede funcionar simplemente ejecutándolo, pero también acepta ciertos parámetros que modifican su comportamiento:

- -p / --encryption_path: ruta a cifrar. Utilizado para únicamente cifrar la ruta indicada.
- -s / --share_file: ruta a un fichero que debe contener un listado de unidades de red a incluir en el proceso de cifrado.
- -n / --encryption_percent: para indicar el porcentaje de cuanto contenido del fichero debe ser cifrado.
- -l: para indicar si se deben mostrar logs o no (por defecto se muestran).

```
169  LOBYTE(v112) = 0;
170  sub_14001E4E0(&v105, argc, argv);
171  v117[0] = "-p";
172  v117[1] = "--encryption_path";
173  v123[0] = v117;
174  v123[1] = v118;
175  v9 = sub_14001EF70(&v105, v127, v123, v8);
176  sub_140020430(v9, v85);
177  *(v127 + *(v127[0] + 4)) = &std::istringstream::`vftable';
178  *(&v126 + *(v127[0] + 4) + 4) = *(v127[0] + 4) - 144;
179  std::stringbuf::~stringbuf(v128);
180  *(v127 + *(v127[0] + 4)) = &std::istream::`vftable';
181  *(&v126 + *(v127[0] + 4) + 4) = *(v127[0] + 4) - 24;
182  *v129 = &std::ios_base::`vftable';
183  std::ios_base::_Ios_base_dtor(v129);
184  v120 = 0i64;
185  v121 = 2i64;
186  v122 = 15i64;
187  strcpy(&v120, "-l");
188  arg_1 = sub_14001EE00(&v105, &v120);
189  if ( v122 >= 16 )
190  {
191    v12 = v120;
192    if ( v122 + 1 >= 0x1000 )
193    {
194      v12 = *(v120 - 8);
195      if ( (v120 - v12 - 8) > 0x1F )
196        invalid_parameter_noinfo_noreturn();
197    }
198    j_j_free(v12);
199  }
200  v118[0] = "-s";
201  v118[1] = "--share_file";
202  v124[0] = v118;
203  v124[1] = v119;
204  v13 = sub_14001EF70(&v105, v130, v124, v10);
205  sub_140020430(v13, lpMultiByteStr);
```

Ilustración 5: Posibles parámetros para su ejecución.

Cifrado de cadenas de caracteres

Akira hace uso de una función para cifrar algunas de sus cadenas de caracteres. Esta función se aplica en tiempo de compilación por lo que su código se embebe directamente en el código en lugar de ser una llamada a función.

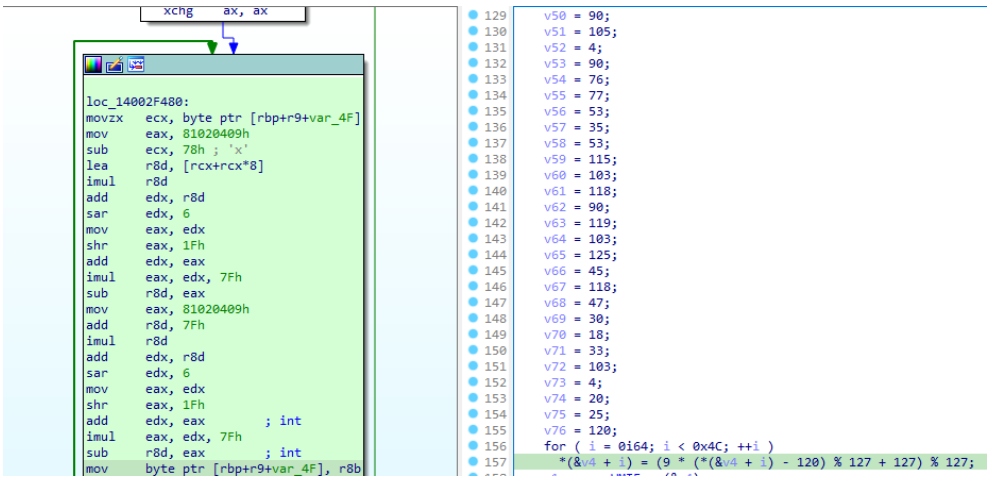


Ilustración 6: Función de descifrado de caracteres embebida en el código.

Akira utiliza la misma función de ofuscación que Conti la cual, a su vez, es tomada del proyecto ADVobfuscator (https://github.com/gharty03/Conti-Ransomware/blob/main/locker/MetaString.h).

Listado de unidades de disco

La primera acción que realiza Akira tras comprobar los parámetros pasados al programa es obtener el listado de unidades de disco a cifrar.

```
}
mw::GetDriveStrings(&v88);
if ( arg_1 )
{
  v22 = mw::PrintMessageDriveList();
  sub_1400156D0(v22);
  v24 = *(&v88 + 1);
  for ( i = v88; i != v24; i += 4 )
  {
    v26 = i;
    if ( i[3] >= 8ui64 )
      v26 = *i;
    v27 = sub_140017210(v23, v26, i[2]);
    sub_1400156D0(v27);
  }
  goto LABEL_99;
}
```

Ilustración 7: Llamada a la función de obtención del listado de discos a cifrar.

Para ello, el malware hace uso de la API *GetLogicalDriveStringsW* para obtener y procesar el listado completo de unidades de disco disponibles en el sistema y almacenarlo en una estructura propia que procesar posteriormente.

```
IDA View-A | Pseudocode-B | Stack of main | Structures
1  DRIVE_LIST *__fastcall mw::GetDriveStrings(DRIVE_LIST *Output)
2  {
3    // [COLLAPSED LOCAL DECLARATIONS. PRESS KEYPAD CTRL-"+" TO EXPAND]
4
5    Output->tqh_first = 0i64;
6    Output->tqh_last = 0i64;
7    Output[1].tqh_first = 0i64;
8    v2 = operator new(8231ui64);
9    if ( !v2 )
10     goto LABEL_16;
11   v3 = ((v2 + 39) & 0xFFFFFFFFFFFFFFE0ui64);
12   *(v3 - 1) = v2;
13   memset(v3, 0, 0x2000ui64);
14   if ( GetLogicalDriveStringsW(260u, v3) )
15   {
16     v4 = 0i64;
17     v5 = v3 + 4;
```

Ilustración 8: Llamada a la API GetLogicalDriveStringsW

Cierre de procesos

El *ransomware* también es capaz de matar procesos en ejecución para poder cifrar también los archivos utilizados por ellos sin que los bloqueen. Para ello, hace uso de la API *WTSEnumerateProcessesW* para encontrar todos los procesos en ejecución y recuperar los PID que luego se compararán con una lista de nombres de procesos en lista blanca y almacena los PID de proceso de los que estén en dicha lista para evitar cerrarlos.

```
v20 = 0;
if ( WTSEnumerateProcessesW(0i64, 0, 1u, &pMemory, &v20) )
{
  v0 = 0;
  if ( v20 )
  {
    while ( 1 )
    {
      *Block = 0i64;
      v18 = 0i64;
      v1 = -1i64;
      v19 = 0i64;
      v2 = 24i64 * v0;
      v3 = *(pMemory + v2 + 8);
      do
        ++v1;
      while ( v3[v1] );
      mw::strcpy(Block, v3, v1);
      v4 = mw::List::CheckMatch(&ignore_process_list_14008CEC8, v16, Block);
      v7 = Block[0];
      v8 = *(v4 + 16);
      if ( *(v8 + 25) )
```

Ilustración 9: Comprobación de procesos en ejecución

Una vez generada esa lista, más tarde, durante el proceso de cifrado de ficheros, por cada fichero que se encuentre bloqueado, pasará dicho fichero a una función que iniciará una sesión de administrador de reinicio de Windows y registrará el fichero mediante *RmRegisterResources*. A continuación, obtiene el listado de aplicaciones y servicios que están haciendo uso del recurso registrado, mediante la API *RmGetList*. Luego compara el PID del proceso actual, el de explorer.exe y los procesos incluidos en la lista blanca con los PID de los procesos obtenidos por *RmGetList*. Si coinciden no se hace nada pero, de lo contrario, se cerrarán por la fuerza todos los procesos registrados que están usando el archivo llamando a *RmShutdown* con el indicador *RmForceShutdown*.

```
55  {
56    sub_14002E860(rgAffectedApps, (668 * pnProcInfoNeeded));
57    v5 = pnProcInfoNeeded;
58    v6 = v18;
59  }
60  pnProcInfo = v5;
61  v7 = rgAffectedApps[0];
62  if ( !RmGetList(pSessionHandle, &pnProcInfoNeeded, &pnProcInfo, rgAffectedApps[0], &dwRebootReasons)
63    && pnProcInfoNeeded )
64  {
65    CurrentProcess = GetCurrentProcess();
66    ProcessId = GetProcessId(CurrentProcess);
67    if ( pnProcInfo )
68    {
69      v10 = v7;
70      while ( v10->Process.dwProcessId != ProcessId )
71      {
72        v11 = Src;
73        if ( Src != pid_to_ignore_14008CF08 )
74        {
75          while ( v10->Process.dwProcessId != *v11 )
76          {
77            if ( ++v11 == pid_to_ignore_14008CF08 )
78              goto LABEL_17;
79          }
80          goto LABEL_25;
81        }
82  LABEL_17:
83        ++v3;
84        ++v10;
85        if ( v3 >= pnProcInfo )
```

xrefs to pid_to_ignore_14008CF08

Directio	Type	Address	Text
Up	r	mw__GetIgnoreProcessPIDs+173	mov rdx, qword ptr cs:pid_to_ignore_14008CF08
Up	r	mw__GetIgnoreProcessPIDs+181	cmp rdx, qword ptr cs:pid_to_ignore_14008CF08+8
Up	w	mw__GetIgnoreProcessPIDs+18F	add qword ptr cs:pid_to_ignore_14008CF08, 4
Up	r	mw__RestartManagerProcessTermination+1B4	mov rdx, qword ptr cs:pid_to_ignore_14008CF08

Ilustración 10: Cierre de procesos durante el proceso de cifrado

Borrado de Shadow Copies

A continuación, Akira llama a una función que se encarga de borrar las *Shadow Copies* del equipo víctima con el fin de evitar la restauración de cualquier fichero tras el proceso de cifrado. Para ello, desofusca un comando de powershell. Este comando elimina las *Shadow Copies* con la ayuda de WMIC mediante la creación de un objeto *WebmLocator* en ROOT\\CIMV2 con la API *CoCreateInstance*.

```
powershell.exe -Command “Get-WmiObject Win32_Shadowcopy | Remove-WmiObject”
```

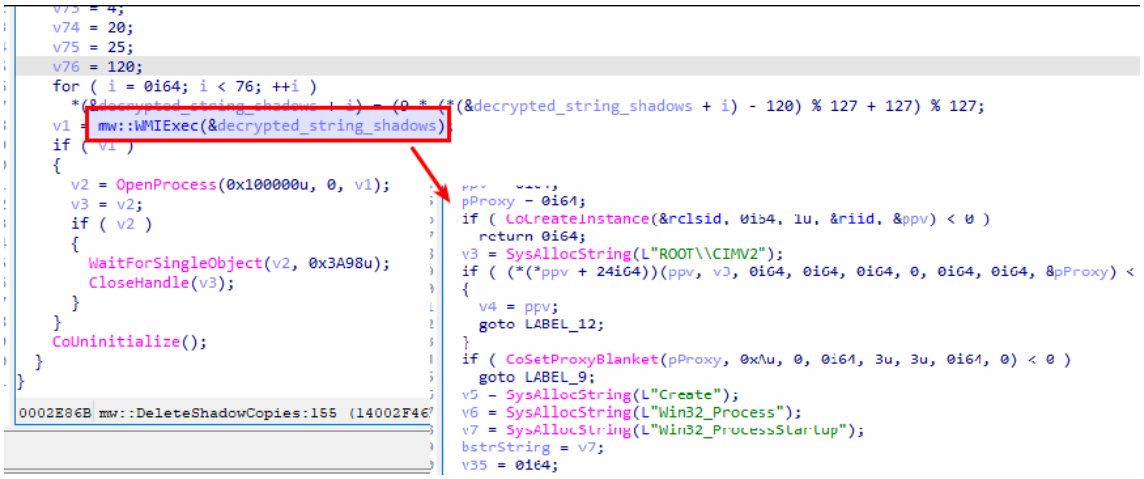


Ilustración 11: Función de borrado de Shadow Copies

Ejecución multihilo

Con la ayuda de la biblioteca Boost.Asio C++ se crean varios subprocesos (4 por cada procesador existente en el sistema) para acelerar la ejecución de Akira de forma asíncrona donde, mediante las API *QueueUserAPC* y *WaitForMultipleObjects*, se controlará la ejecución de las funciones del *ransomware* en una cola de subprocesos.

```
316     goto LABEL_100;
317   }
318   mw_calc_thread_pool_14001E1B0(&thread_pool, (4 * dwNumberOfProcessors));
319   *Block = 0i64;
```

Ilustración 12: Generación de múltiples hilos

Inicialización criptográfica

Antes de comenzar el proceso de cifrado, Akira realiza una serie de ejecuciones para inicializar los servicios criptográficos. Por un lado, se importa la clave RSA embebida en el código. Por otro, se genera de forma aleatoria, mediante la API *CryptGenRandom*, una clave de cifrado de longitud 32 bytes y un vector de inicialización (IV) de 8 bytes para el algoritmo Chacha20 con el que se cifraran los ficheros posteriormente. Estos dos valores y el resultado de cifrarlos

mediante RSA con la clave pública importada previamente son guardados en una estructura para pasárselos al proceso de cifrado de ficheros.

```
  pcbStructInfo = 0;
  pcbBinary = 2048;
  if ( !CryptAcquireContextW(&phProv, 0i64, L"Microsoft Enhanced RSA and AES Cryptographic Provider", 0x18u, 0xF0000000) )
    goto LABEL_99;
  if ( !CryptStringToBinaryA(rsa_pub_key, 0, 0, pbBinary, &pcbBinary, 0i64, 0i64) )// Import RSA Key
    goto LABEL_99;
  v34 = 1;
  if ( !CryptDecodeObjectEx(1u, 8, pbBinary, pcbBinary, 0x8000u, 0i64, &pInfo, &pcbStructInfo)
     || !CryptImportPublicKeyInfo(phProv, 1u, pInfo, &phKey)
     || (v35 = phProv, v36 = phKey, !phProv)
     || !phKey
     || (memset(&akira_encryption_config, 0, sizeof(akira_encryption_config)),
         !CryptGenRandom(phProv, 32u, akira_encryption_config.ChachaKey))// Chacha20 key: 32bytes
     || !CryptGenRandom(v35, 8u, akira_encryption_config.ChachaIV)// Chacha20 IV
     || (*akira_encryption_config.EncryptedKey = *akira_encryption_config.ChachaKey,
         *&akira_encryption_config.EncryptedKey[16] = *&akira_encryption_config.ChachaKey[16],
         *&akira_encryption_config.EncryptedKey[32] = *akira_encryption_config.ChachaIV,
         pdwDataLen = 40,
         !CryptEncrypt(v36, 0i64, 1, 0, akira_encryption_config.EncryptedKey, &pdwDataLen, 0x20Cu)) )
  {
LABEL_99:
```

Ilustración 13: Cálculo de clave y vector de inicialización Chacha20 y cifrado RSA

A diferencia de otros *ransomware*, Akira solo utiliza una única clave para cifrar todos los ficheros por lo que, si se consigue averiguar esta clave, se podrían descifrar todos los ficheros del equipo.

Para disponer de estos datos en el proceso de cifrado, Akira almacena la clave, el vector y el resultado de cifrar ambos con RSA en una estructura.

```
typedef struct akira_encyption_config {
    BYTE ChachaIV[8];
    BYTE ChachaKey[32];
    BYTE EncryptedKey[524];
} AKIRA_ENCRYPTION_CONFIG, * LPAKIRA_ENCRYPTION_CONFIG;
```

Iteración de directorios y nota de rescate

Akira llama a la función de iteración de directorios para ser ejecutada de forma asíncrona pasándole la estructura anterior.

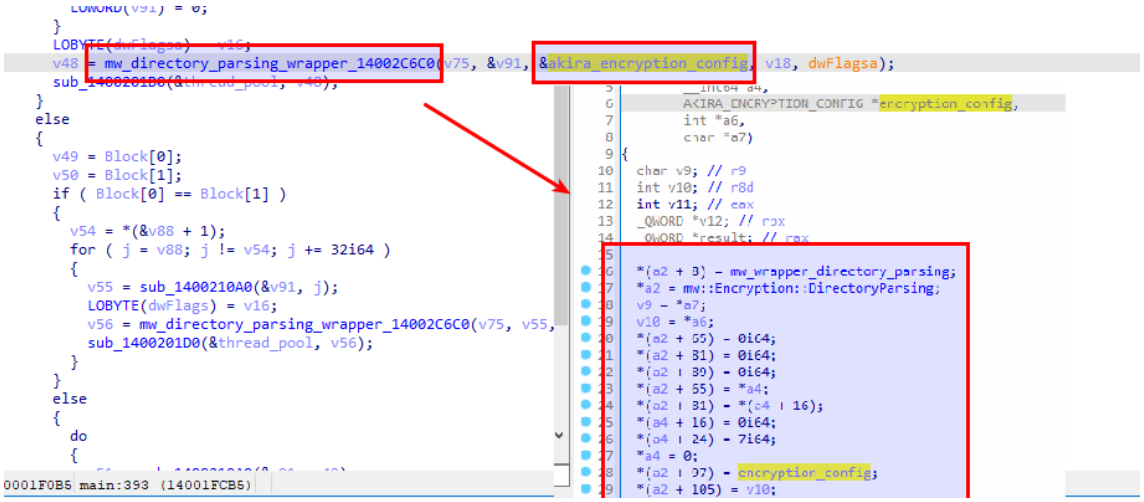


Ilustración 14: Llamada a la función de iteración de directorios

Por cada directorio encontrado por esta función, el malware escribe la nota de rescate en él. Para ello, se recupera el texto de la nota desde una variable, así como el nombre de fichero de la misma y se llama a la API WriteFile.

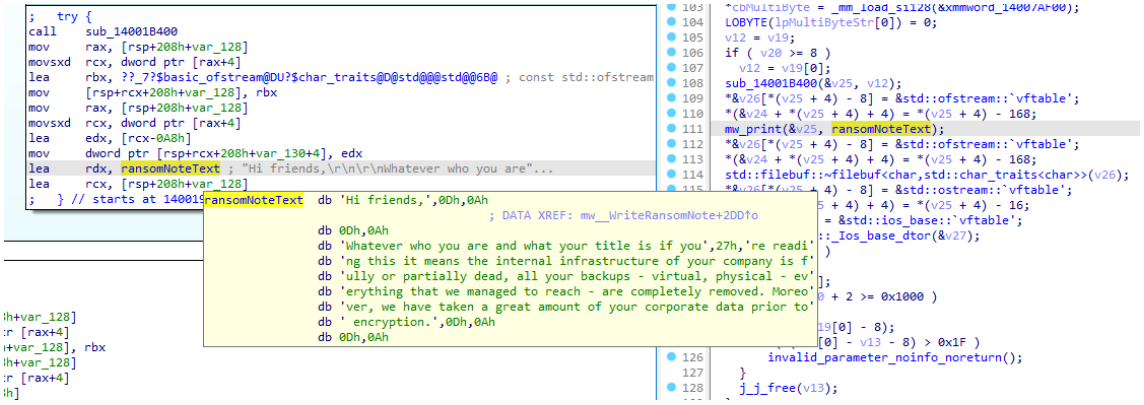


Ilustración 15: Escritura de la nota de rescate de Akira

El nombre de archivo utilizado en la muestra analizada es “akira_readme.txt”. El texto de la nota contiene un identificador único utilizado para autenticarse en el sitio web de la red Tor que los cibercriminales utilizan para negociar el pago del rescate.

Para evitar corromper el funcionamiento del sistema, Akira contiene un listado de directorios a los que no accede. Este listado se encuentra cifrado mediante el mismo algoritmo de ofuscación de cadenas descrito previamente.

tmp
winnt
temp
thumb
$Recycle.Bin
$RECYCLE.BIN
System Volume
Information Boot
Windows Trend Micro
ProgramData

Cifrado de ficheros

Por cada fichero encontrado en el proceso anterior, se creará una tarea en la cola de subprocesos para proceder a cifrarlo. Akira contiene un listado de nombres de fichero y de extensiones que evitará cifrar para evitar corromper el funcionamiento normal del sistema o cifrar archivos previamente afectados por una ejecución previa del ransomware.

Nombres de fichero	Extensiones
akira_readme.txt Bootmgr	.exe .dll

BOOTNXT DumpStack.log.tmp pagefile.sys swapfile.sys ntuser.dat	.sys .msi .lnk .akira

Como se ha indicado previamente, durante la ejecución el *ransomware* genera una única clave de cifrado simétrica y vector IV utilizando la API *CryptGenRandom*, que es el generador de números aleatorios implementado por Windows CryptoAPI. Los archivos son ahora cifrados utilizando esos valores mediante el algoritmo Chacha20. La clave simétrica y el vector IV, a su vez, se cifran con RSA-4096 y son agregados al final del archivo cifrado para poder ser accedidos y descifrados en caso de pagar el rescate y ejecutar el descifrador en el equipo.

A cada fichero cifrado, Akira le añade un pie de fichero que contiene una estructura de datos con cierta información que necesita conservar para realizar el descifrado de los ficheros. Esta estructura tiene la siguiente definición:

```
typedef struct _FILE_TAIL_AKIRA
{
   BYTE RsaEncrypted[512];
   BYTE Zeroed[12];
   BYTE EncryptionType;
   BYTE Version;
   BYTE OriginalSize[8];
} FILE_TAIL_AKIRA, *PFILE_TAIL_AKIRA;
```

Akira tiene en cuenta el tamaño de fichero a la hora de cifrarlo. De esta forma, en función del tamaño, lo cifrará de forma diferente. Para archivos de 2.000.000 bytes (2MB) o menos, el *ransomware* cifra la primera mitad del archivo y añade la estructura de pie de fichero.

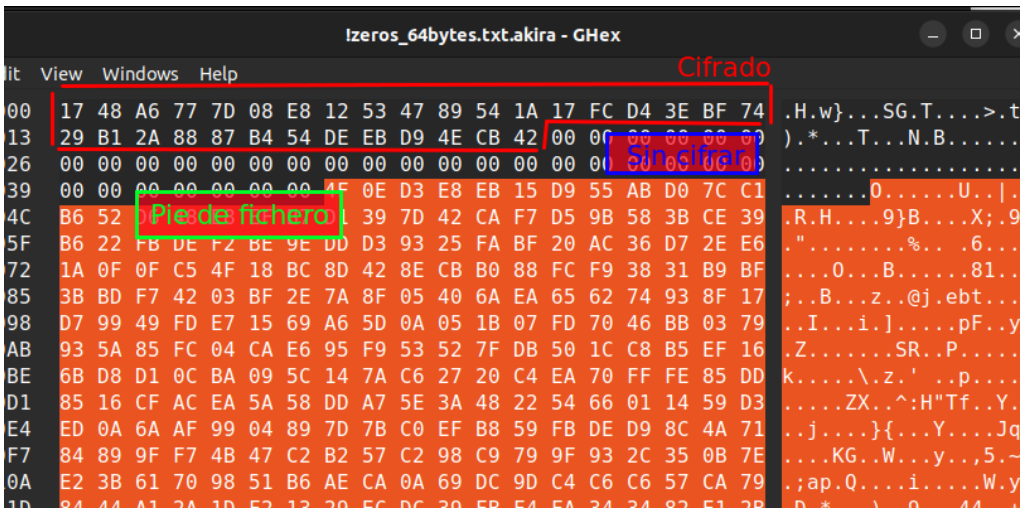


Ilustración 16: Fichero pequeño cifrado por Akira

Para ficheros mayores de 2MB, Akira cifra el fichero en bloques. Para calcular el tamaño de bloque a dejar intacto y el tamaño de bloque a cifrar realiza cálculos en función del tamaño de fichero.

```
len_mitad_fichero = filesize / 2;
len_block = (filesize - len_mitad_fichero / 5 * 4) / 5;
len_encrypt = filesize * 50 / 500
```

De esta forma, el fichero se cifraría en cuatro bloques:

- Bloque cifrado 1 (len_encrypt)
- Bloque sin cifrar (len_block – len_encrypt)
- Bloque cifrado 2 (len_encrypt)
- Bloque sin cifrar (len_block – len_encrypt)
- Bloque cifrado 3 (len_encrypt)
- Bloque sin cifrar (len_block – len_encrypt)
- Bloque cifrado 4 (len_encrypt)
- Resto del fichero sin cifrar
- Pie de fichero

Volviendo a la estructura de pie de fichero, la variable *EncryptionType* indica la forma de cifrado utilizada: 0x01 para ficheros pequeños (mitad de fichero cifrado) y 0x02 para ficheros grandes (cuatro bloques cifrados).

Descifrador de Avast

Debido al hecho de que solo se utiliza una única clave de cifrado para cifrar todos los ficheros del sistema, el equipo de la firma de seguridad Avast ha logrado desarrollar un programa para crackear esa clave de los ficheros afectados por este *ransomware*. Para poder utilizarlo, es necesario disponer de la versión original y cifrada de un fichero y, cuanto mayor sea éste, mejor realizará el proceso.

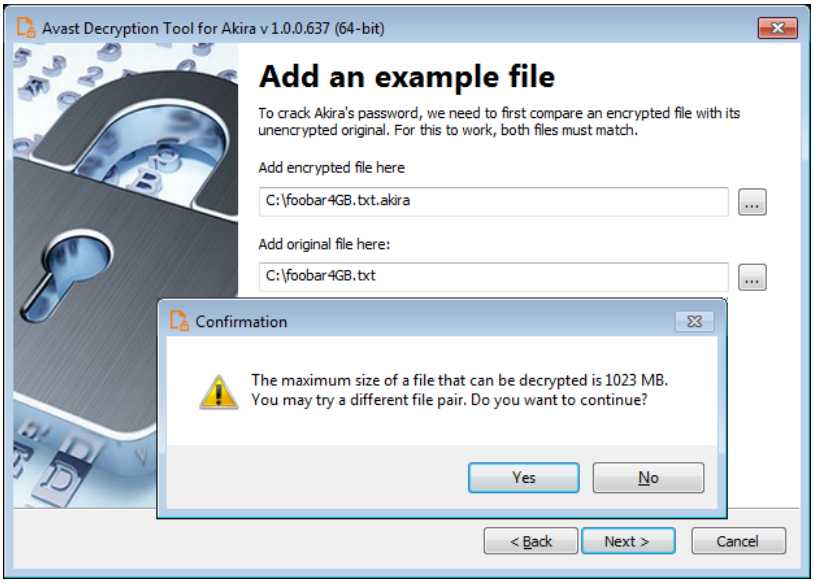


Ilustración 17: Descifrador de Akira publicado por Avast

Tras esto, comenzará el proceso de fuerza bruta para encontrar la clave de cifrado.

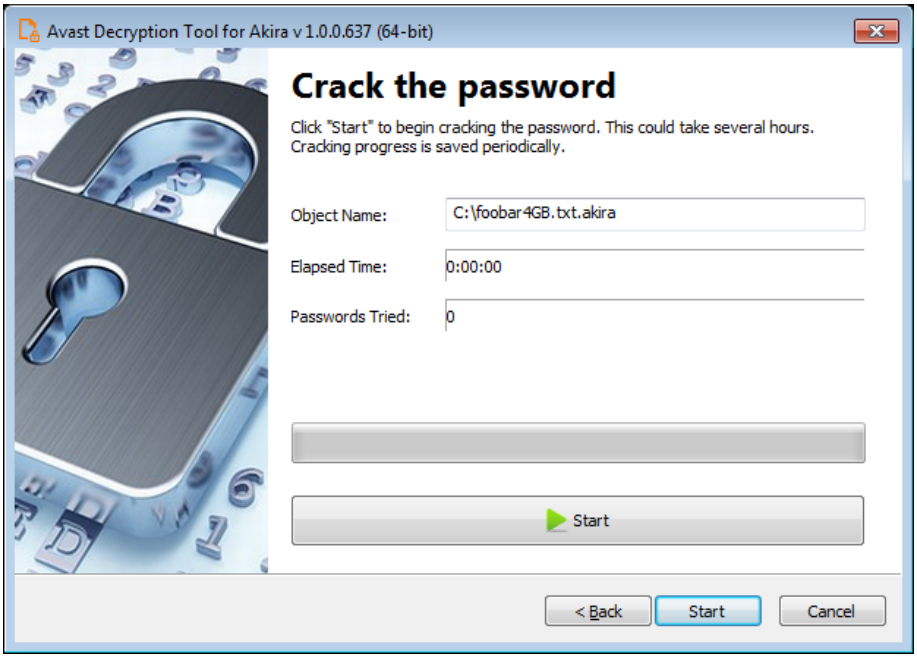


Ilustración 18: Proceso de crackeo de clave de Akira

Muestra analizada (Linux)

Los actores detrás de Akira han desarrollado una versión de su ransomware para cifrar sistemas Linux. La muestra obtenida es un ejecutable de Linux (ELF) para 64 bits cuyo hash SHA256 es:

1d3b5c650533d13c81e325972a912e3ff8776e36e18bca966dae50735f8ab296.

```
$ file 1d3b5c650533d13c81e325972a912e3ff8776e36e18bca966dae50735f8ab296
1d3b5c650533d13c81e325972a912e3ff8776e36e18bca966dae50735f8ab296: ELF 64-bit LSB executable, x86-64, version 1 (GNU/Linux), BuildID[sha1]=ea7a79da771763411b0d0ed75a30acf20b5debbe, for GNU/Linux 3.2.0, statically linked, no section header
```

Ilustración 19: Binario para Linux de Akira

La versión de Linux de Akira funciona de manera idéntica a su contraparte de Windows y los archivos cifrados tienen la misma extensión y el mismo esquema de cifrado.

Obviamente, la CryptoAPI de Windows no está disponible en Linux, por lo que los autores del *ransomware* han usado la biblioteca Crypto++ para cubrir las partes que maneja la CryptoAPI en Windows.

Los parámetros aceptados por el binario de Linux son los siguientes muy parecidos a los de Windows:

- -p / --encryption_path: ruta de los archivos/carpetas que se cifrarán.
- -s / --share_file: ruta de la unidad de red compartida que se cifrará.
- -n / --encryption_percent: porcentaje de los archivos que se cifrarán.
- -fork: creación de un proceso secundario para el cifrado.

```
v39 = __readfsqword(0x28u);
sub_41EB98(v37);
sub_41DD8E(v37, a1, a2, 1LL);
v35 = "-p";
v36 = "--encryption_path";
sub_41E996(v38, v37, &v35, 2LL);
sub_5B6260(v32, v38);
ZNSt7__cxx1119basic_istringstreamIcSt11char_traitsIcESaIcEED1Ev(v38);
v35 = "-s";
v36 = "--share_file";
sub_41E996(v38, v37, &v35, 2LL);
sub_5B6260(v33, v38);
ZNSt7__cxx1119basic_istringstreamIcSt11char_traitsIcESaIcEED1Ev(v38);
v35 = "-n";
v36 = "--encryption_percent";
sub_41E996(v38, v37, &v35, 2LL);
sub_5B6260(v34, v38);
ZNSt7__cxx1119basic_istringstreamIcSt11char_traitsIcESaIcEED1Ev(v38);
sub_541DA0(v31);
ZNSt7__cxx1112basic_stringIcSt11char_traitsIcESaIcEEC2EPKcRKS3_(v38, "-fork", v31);
v2 = (__int64 (__fastcall *)(std::ostream *))v38;
v18 = mprPushItem((MprList_0 *)v37, (cvoid *)v38);
ZNSt13__facet_shims12_GLOBAL__N_116__destroy_stringIcEEvPv_0_0(v38);
sub_541DC0(v31);
if ( v18 )
{
  v21 = sub_682F00(v31, v38);
  if ( v21 < 0 )
  {
    v3 = ZStlsISt11char_traitsIcEERSt13basic_ostreamIcT_ES5_PKc((std::ostream *)&unk_7D27A0, "Failed to fork");
    v2 = ZSt4endlIcSt11char_traitsIcEERSt13basic_ostreamIT_T0_ES6_;
    sub_5AD1E0(v3, ZSt4endlIcSt11char_traitsIcEERSt13basic_ostreamIT_T0_ES6_);
  }
```

Ilustración 20: Procesado de argumentos pasados a la versión Linux de Akira

Al igual que en Windows, el binario trae una clave RSA pública embebida.

Ilustración 21: Clave pública RSA en la versión Linux de Akira

La rutina de cifrado de la versión de Linux de Akira funciona exactamente igual, cifrando los ficheros mediante Chacha20.

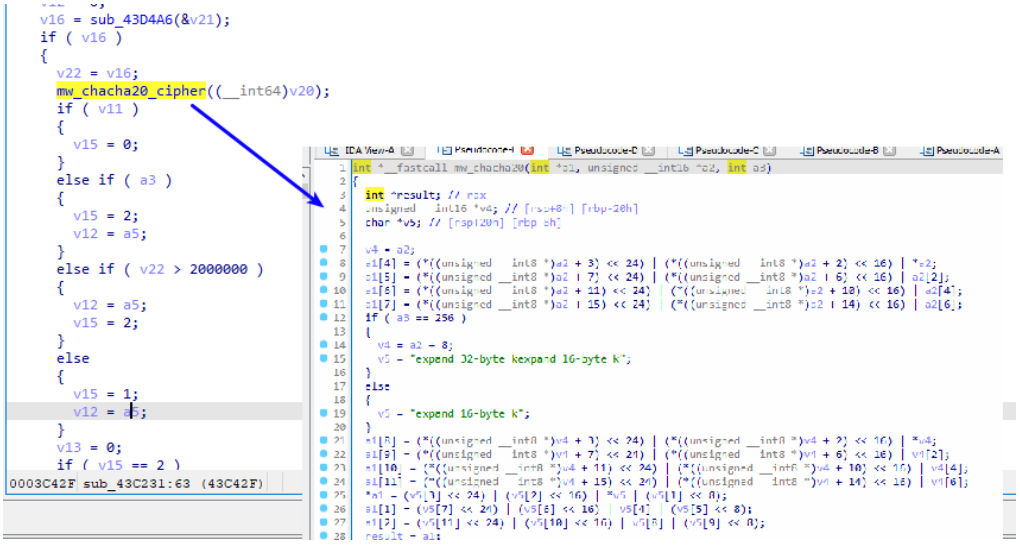


Ilustración 22: Algoritmo Chacha20 en la versión Linux de Akira

Con respecto al descifrador de Avast, su equipo está trabajando actualmente en desarrollar una versión Linux del descifrador. Mientras tanto, ya que el esquema de cifrado es el mismo, podría utilizarse la versión de Windows para descifrar los ficheros cifrados en Linux mediante el software *Wine*.

Vulnerabilidades explotadas

No se identifican, a priori, ningunas vulnerabilidades específicas que estén siendo explotadas por los actores involucrados con Akira de forma general o por el propio binario del *ransomware*.

MITRE ATT&CK			
Initial Access	**T1078**	Valid Accounts	**M1036: Account Use Policies** Use conditional access policies to block logins from non-compliant devices or from outside defined organization IP ranges.
			M1015: Active Directory Configuration Disable legacy authentication, which does not support MFA, and require the use of modern authentication protocols instead.
			M1013: Application Developer Guidance Ensure that applications do not store sensitive data or credentials insecurely. (e.g. plaintext credentials in code, published credentials in repositories, or credentials in public cloud storage).
			M1027: Password Policies Applications and appliances that utilize default username and password should be changed immediately after the installation, and before deployment to a production environment.[66] When possible, applications that use SSH keys should be updated periodically and properly secured. Policies should minimize (if not eliminate) reuse of passwords between different user accounts, especially employees using the same credentials for personal accounts that may not be defended by enterprise security resources.
			M1026: Privileged Account Management Audit domain and local accounts as well as their permission levels routinely to look for situations that could allow an adversary to gain wide access by obtaining credentials of a privileged account. These audits should also include if default accounts have been enabled, or if new local accounts are created that have not be authorized. Follow best practices for design and administration of an enterprise network to limit privileged account use across administrative tiers.
			M1018: User Account Management Regularly audit user accounts for activity and deactivate or remove any that are no longer needed.
			M1017: User Training

			Applications may send push notifications to verify a login as a form of multi-factor authentication (MFA). Train users to only accept valid push notifications and to report suspicious push notifications.
	T1566.001	Spearphishing Attachment	**M1049: Antivirus/Antimalware** Anti-virus can also automatically quarantine suspicious files.
			M1031: Network Intrusion Prevention Network intrusion prevention systems and systems designed to scan and remove malicious email attachments can be used to block activity.
			M1021: Restrict Web-Based Content Block unknown or unused attachments by default that should not be transmitted over email as a best practice to prevent some vectors, such as .scr, .exe, .pif, .cpl, etc. Some email scanning devices can open and analyze compressed and encrypted formats, such as zip and rar that may be used to conceal malicious attachments.
			M1054: Software Configuration Use anti-spoofing and email authentication mechanisms to filter messages based on validity checks of the sender domain (using SPF) and integrity of messages (using DKIM). Enabling these mechanisms within an organization (through policies such as DMARC) may enable recipients (intra-org and cross domain) to perform similar message filtering and validation.
			M1017: User Training Users can be trained to identify social engineering techniques and spearphishing emails.
Execution	**T1204.002**	Malicious File	**M1040: Behavior Prevention on Endpoint** On Windows 10, various Attack Surface Reduction (ASR) rules can be enabled to prevent the execution of potentially malicious executable files (such as those that have been downloaded and executed by Office applications/scripting interpreters/email clients or that do not meet specific prevalence, age, or trusted list criteria). Note: cloud-delivered protection must be enabled for certain rules. (Citation: win10_asr)
			M1017: User Training Use user training as a way to bring awareness to common phishing and spearphishing techniques and how to raise suspicion for potentially malicious events.
			M1038: Execution Prevention Application control may be able to prevent the running of executables masquerading as other files.

	T1106	Native API	**M1038: Execution Prevention** Identify and block potentially malicious software executed that may be executed through this technique by using application control (Citation: Beechey 2010) tools, like Windows Defender Application Control(Citation: Microsoft Windows Defender Application Control), AppLocker, (Citation: Windows Commands JPCERT) (Citation: NSA MS AppLocker) or Software Restriction Policies (Citation: Corio 2008) where appropriate. (Citation: TechNet Applocker vs SRP)
			M1040: Behavior Prevention on Endpoint On Windows 10, enable Attack Surface Reduction (ASR) rules to prevent Office VBA macros from calling Win32 APIs. (Citation: win10_asr)
	T1059	Command and Scripting Interpreter	**M1049: Antivirus/Antimalware** Anti-virus can be used to automatically quarantine suspicious files.
			M1021: Restrict Web-Based Content Script blocking extensions can help prevent the execution of scripts and HTA files that may commonly be used during the exploitation process. For malicious code served up through ads, adblockers can help prevent that code from executing in the first place.
			M1026: Privileged Account Management When PowerShell is necessary, restrict PowerShell execution policy to administrators. Be aware that there are methods of bypassing the PowerShell execution policy, depending on environment configuration.(Citation: Netspi PowerShell Execution Policy Bypass)
			M1045: Code Signing Where possible, only permit execution of signed scripts.
			M1040: Behavior Prevention on Endpoint On Windows 10, enable Attack Surface Reduction (ASR) rules to prevent [Visual Basic](https://attack.mitre.org/techniques/T1059/005) and [JavaScript](https://attack.mitre.org/techniques/T1059/007) scripts from executing potentially malicious downloaded content (Citation: win10_asr).
			M1038: Execution Prevention Use application control where appropriate.
			M1042: Disable or Remove Feature or Program Disable or remove any unnecessary or unused shells or interpreters.

	T1204	User Execution	**M1040: Behavior Prevention on Endpoint** On Windows 10, enable Attack Surface Reduction (ASR) rules to prevent executable files from running unless they meet a prevalence, age, or trusted list criteria and to prevent Office applications from creating potentially malicious executable content by blocking malicious code from being written to disk. Note: cloud-delivered protection must be enabled to use certain rules. (Citation: win10_asr)
			M1021: Restrict Web-Based Content If a link is being visited by a user, block unknown or unused files in transit by default that should not be downloaded or by policy from suspicious sites as a best practice to prevent some vectors, such as .scr, .exe, .pif, .cpl, etc. Some download scanning devices can open and analyze compressed and encrypted formats, such as zip and rar that may be used to conceal malicious files.
			M1031: Network Intrusion Prevention If a link is being visited by a user, network intrusion prevention systems and systems designed to scan and remove malicious downloads can be used to block activity.
			M1038: Execution Prevention Application control may be able to prevent the running of executables masquerading as other files.
			M1017: User Training Use user training as a way to bring awareness to common phishing and spearphishing techniques and how to raise suspicion for potentially malicious events.
	T1059.003	Windows Command Shell	**M1038: Execution Prevention** Use application control where appropriate.
	T1059.001	PowerShell	**M1038: Execution Prevention** Use application control where appropriate.
			M1049: Antivirus/Antimalware Anti-virus can be used to automatically quarantine suspicious files.
			M1026: Privileged Account Management When PowerShell is necessary, restrict PowerShell execution policy to administrators. Be

			aware that there are methods of bypassing the PowerShell execution policy, depending on environment configuration.(Citation: Netspi PowerShell Execution Policy Bypass)
Defense Evasion	**T1140**	Deobfuscate/Decode Files or Information	**This type of attack technique cannot be easily mitigated with preventive controls since it is based on the abuse of system features.**
Discovery	**T1135**	Network Share Discovery	**M1028: Operating System Configuration** Enable Windows Group Policy "Do Not Allow Anonymous Enumeration of SAM Accounts and Shares" security setting to limit users who can enumerate network shares.(Citation: Windows Anonymous Enumeration of SAM Accounts)
	T1083	File and Directory Discovery	**This type of attack technique cannot be easily mitigated with preventive controls since it is based on the abuse of system features.**
Impact	**T1486**	Data Encrypted for Impact	**M1053: Data Backup** Consider implementing IT disaster recovery plans that contain procedures for regularly taking and testing data backups that can be used to restore organizational data.(Citation: Ready.gov IT DRP) Ensure backups are stored off system and is protected from common methods adversaries may use to gain access and destroy the backups to prevent recovery. Consider enabling versioning in cloud environments to maintain backup copies of storage objects.(Citation: Rhino S3 Ransomware Part 2)
			M1040: Behavior Prevention on Endpoint On Windows 10, enable cloud-delivered protection and Attack Surface Reduction (ASR) rules to block the execution of files that resemble ransomware. (Citation: win10_asr)
	T1490	Inhibit System Recovery	**M1053: Data Backup** Consider implementing IT disaster recovery plans that contain procedures for taking regular data backups that can be used to restore organizational data.(Citation: Ready.gov IT DRP) Ensure backups are stored off system and is protected from common methods adversaries may use to gain access and destroy the backups to prevent recovery.
			M1028: Operating System Configuration Consider technical controls to prevent the disabling of services or deletion of files involved in system recovery.

Mitigación

Medidas a nivel de endpoint

Implementar una política que no permita la ejecución de binarios no firmados puede prevenir la ejecución del malware Akira. Sin embargo, esta estrategia puede no ser práctica debido a que muchos desarrolladores y paquetes de software no distribuyen productos firmados.

Prohibir o al menos monitorizar la ejecución de binarios desconocidos o de fuentes no confiables puede servir como una alarma inicial para detectar la presencia del malware y limitar su propagación. Esta medida es más general y se ajusta a la forma en que se crea y distribuye el software legítimo.

Mantener endpoints vigilados con soluciones de monitorización, antivirus y EDR, y establecer una política de actualizaciones para mantener los sistemas al día con las últimas correcciones de vulnerabilidades.

Realizar programas de capacitación para concienciar a los usuarios sobre las prácticas de ciberseguridad. Esto incluye enseñarles a identificar correos electrónicos o sitios web sospechosos, no abrir archivos adjuntos o enlaces desconocidos, y evitar descargar software de fuentes no confiables. Los usuarios capacitados son menos propensos a caer en trampas y ejecutar malware.

Medidas a nivel de red

Utilizar herramientas de análisis de tráfico de red para monitorear y examinar el tráfico en busca de patrones o comportamientos sospechosos. Esto puede ayudar a identificar posibles comunicaciones de comando y control utilizadas por el *ransomware* para comunicarse con los servidores de los atacantes.

Implementar una solución de filtrado de contenido web que bloquee el acceso a sitios web maliciosos o de alto riesgo. Esto puede evitar que los usuarios accedan accidentalmente a páginas que contienen descargas de *ransomware* o enlaces a sitios comprometidos.

Dividir la red en segmentos o subredes más pequeñas y restringir el tráfico entre ellas. Esto limita la propagación del ransomware en caso de una infección, ya que el malware tendría dificultades para moverse de un segmento a otro. Además, se pueden aplicar políticas de seguridad más estrictas en los segmentos críticos que contienen datos sensibles.

Medidas y consideraciones adicionales

Enviar todos los eventos del sistema, especialmente los más importantes, a un sistema externo que centralice los registros de todos los equipos de la red. Esto garantiza la trazabilidad y ayuda a detectar intrusiones en el sistema.

Mantener una política de actualizaciones para asegurarse de que todos los sistemas estén al día y no tengan vulnerabilidades que los atacantes puedan explotar.

Eliminar las contraseñas por defecto en todos los sistemas y aplicar una política de contraseñas que exija contraseñas seguras y cambios periódicos. Además, utilizar autenticación de dos factores en todos los sistemas que lo permitan.

Mantener al equipo de seguridad actualizado sobre las nuevas vulnerabilidades conocidas y asegurarse de que tienen conocimiento de todos los sistemas utilizados en la infraestructura tecnológica. De ser necesario, aplicar medidas de mitigación adicionales en situaciones específicas.

En caso de incidente con este malware, debe ser reportado a las autoridades pertinentes lo más rápido posible.

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

Hashes

- 1b6af2fbbc636180dd7bae825486ccc45e42aefbb304d5f83fafca4d637c13cc
- 3c92bfc71004340ebc00146ced294bc94f49f6a5e212016ac05e7d10fcb3312c
- 678ec8734367c7547794a604cc65e74a0f42320d85a6dce20c214e3b4536bb33
- 6cadab96185dbe6f3a7b95cf2f97d6ac395785607baa6ed7bf363deeb59cc360
- 7b295a10d54c870d59fab3a83a8b983282f6250a0be9df581334eb93d53f3488
- 8631ac37f605daacf47095955837ec5abbd5e98c540ffd58bb9bf873b1685a50
- 9ca333b2e88ab35f608e447b0e3b821a6e04c4b0c76545177890fb16adcab163
- d0510e1d89640c9650782e882fe3b9afba00303b126ec38fdc5f1c1484341959
- 1d3b5c650533d13c81e325972a912e3ff8776e36e18bca966dae50735f8ab296

Yara:

- Estas reglas sirven para identificar las muestras de la familia *Akira* en sistemas Windows.

YARA

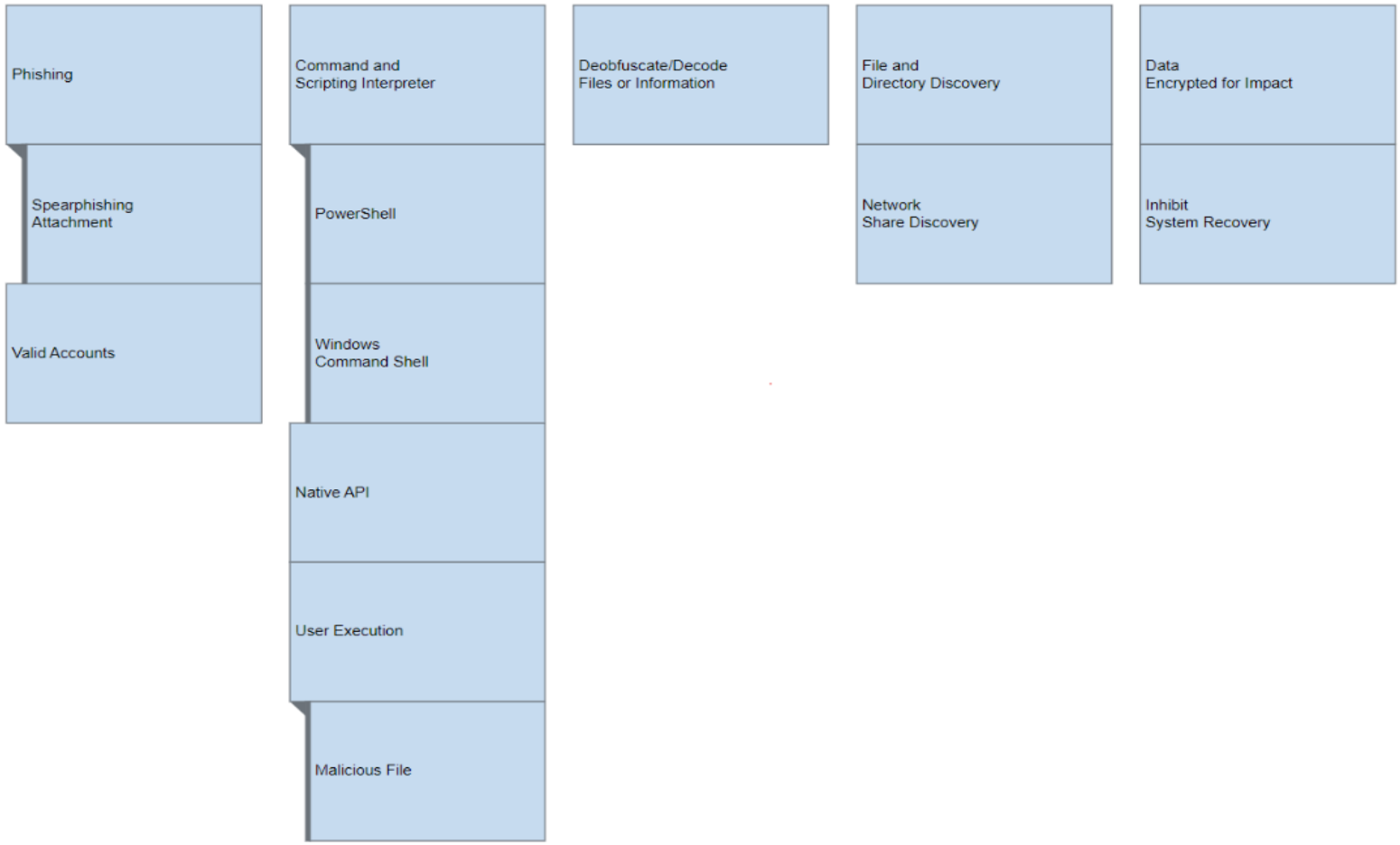
```
rule MALWARE_Win_Akira {
    meta:
        author = "ditekSHen"
        description = "Detects Akira Ransomware Windows"
    strings:
        $x1 = "https://akira" ascii
        $x2 = ":\\akira\\" ascii
        $x3 = ".akira" ascii
        $x4= "akira_readme.txt" ascii
        $x5 = "\\akira\\asio\\include\\asio\\impl\\co_spawn.hpp" ascii
        $s1 = "Get-WmiObject Win32_Shadowcopy | Remove-WmiObject" ascii
        $s2 = "Win32_ProcessStartup" fullword wide
        $s3 = /Failed\sto\smake\s(part|full|spot)\sencrypt/ ascii wide
        $s4 = "--encryption_" ascii
        $s5 = "--share_file" ascii
        $s6 = { 24 00 52 00 45 00 43 00 59 00 43 00 4C 00 45 00 2E 00 42 00
49 00 4E 00 00 00 00 00 6? 6? 6? 00 (24|57) 00 (52|69) 00 }
        $s7 = " PUBLIC KEY-----" ascii
        $s8 = ".onion" ascii
        $s9 = "/Esxi_Build_Esxi6/./" ascii nocase
```

```
        $s10 = "No path to encrypt" ascii
        $s11 = "-fork" fullword ascii
    condition:
        uint16(0) == 0x5a4d and (3 of ($x*) or (1 of ($x*) and 4 of ($s*))
or 6 of ($s*))
}
```

Referencias adicionales

- https://labs.k7computing.com/index.php/akira-ransomware-unleashing-chaos-using-conti-leaks/
- https://decoded.avast.io/threatresearch/decrypted-akira-ransomware/
- https://blog.cyble.com/2023/05/10/unraveling-akira-ransomware/
- https://www.blackhatethicalhacking.com/news/new-akira-ransomware-operation-hits-corporate-networks/
- https://news.sophos.com/en-us/2023/05/09/akira-ransomware-is-bringin-88-back/
- https://blog.cyble.com/2023/06/28/akira-ransomware-extends-reach-to-linux-platform/
- https://ieeexplore.ieee.org/document/9895237

Apéndice A: Mapa de técnicas de ATT&CK



EUSKO JAURLARITZA GOBIERNO VASCO
GRUPO spri TALDEA