Akira Ransomware

trellix.com/about/newsroom/stories/research/akira-ransomware/

Blogs

The latest cybersecurity trends, best practices, security vulnerabilities, and more

By Alexandre Mundo, Max Kersten · November 29, 2023

First discovered in early 2023, Akira ransomware seemed to be just another ransomware family that entered the market. Its continued activity and numerous victims are our main motivators to investigate the malware’s inner workings to empower blue teams to create additional defensive rules outside of their already in-place security.

MD-5

f526a8ea744a8c5051deefbf2c6010af

SHA-1

d4f6241abe5f46e6b18f10da95d004924eac4ed3

SHA-256

8bfa4c2c1065b105ec80a86f460e0e0221b39610109cc6cd4b441dd86e6b4aef

Detection names

EX/NX:

- FEC_Trojan_Win64_Generic_4
- Ransomware.Win64.Akira.FEC3
- Ransomware.Win.Akira.MVX

HX AV:

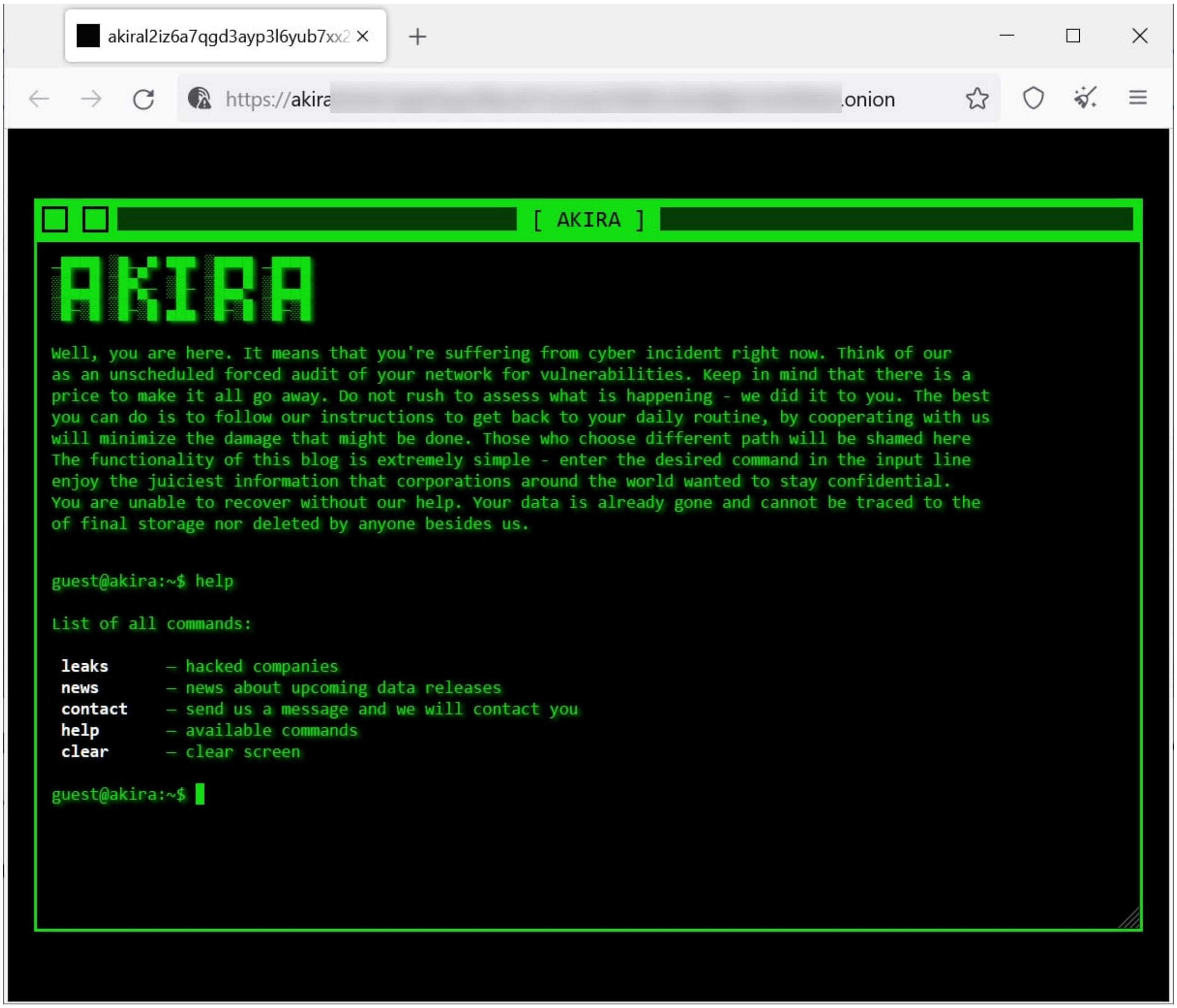
- Generic.Ransom.Akira.A.6926E830
- Generic.mg.f526a8ea744a8c50

ENS:

- AkiraRansom!F526A8EA744A

About Akira

The ransomware’s name likely comes from an 1988 anime movie with the same name (spoilers ahead). The movie’s cyberpunk aesthetic is emulated by the ransom group on their leak site, as can be seen on the image below, courtesy of BleepingComputer.



```
Well, you are here. It means that you're suffering from cyber incident right now. Think of our
as an unscheduled forced audit of your network for vulnerabilities. Keep in mind that there is a
price to make it all go away. Do not rush to assess what is happening - we did it to you. The best
you can do is to follow our instructions to get back to your daily routine, by cooperating with us
will minimize the damage that might be done. Those who choose different path will be shamed here
The functionality of this blog is extremely simple - enter the desired command in the input line
enjoy the juiciest information that corporations around the world wanted to stay confidential.
You are unable to recover without our help. Your data is already gone and cannot be traced to the
of final storage nor deleted by anyone besides us.


guest@akira:~$ help

List of all commands:

 leaks      – hacked companies
 news       – news about upcoming data releases
 contact    – send us a message and we will contact you
 help       – available commands
 clear      – clear screen

guest@akira:~$ █
```

Figure 1: The Akira leak site

The movie is set in Neo-Tokyo, which was built after Akira destroyed the city. In the movie, Akira destroys Neo-Tokyo in order to save the world from an ever growing mass of flesh within the city. The ransomware authors based their name on the powerful entity within the anime movie, or from its related manga, as they might perceive themselves as such.

The ransom group employs a double extortion scheme which includes exfiltrating data prior to the encryption of devices within the targeted network. As such, the ransom needs to be paid for the removal of the stolen files, which are otherwise leaked, and to obtain the decryptor to regain access to the encrypted files.

Victimology

Knowing if a group favors a certain sector, a geographical area, or acts purely based on opportunities is of great benefit for blue teams. It allows threat intelligence teams to understand their potential adversaries and act accordingly. Threat detection engineering teams and security operation centers can improve their detection based on known tactics, techniques, and procedures (TTPs). Noteworthy here is that “known” TTPs do not necessarily mean publicly known, but rather internally known under any of the traffic light protocol’s options.

Akira’s victims, based on their blog posts, are plotted on the pie chart below. The country of origin of each company is based on the headquarters of the company, meaning that any company which has offices in multiple countries will only contribute to one country. A final note as to what counts as a victim in the numbers used in this blog: each unique company which has been published on Akira’s blog counts. Victims who do not show up on the blog, are not included.

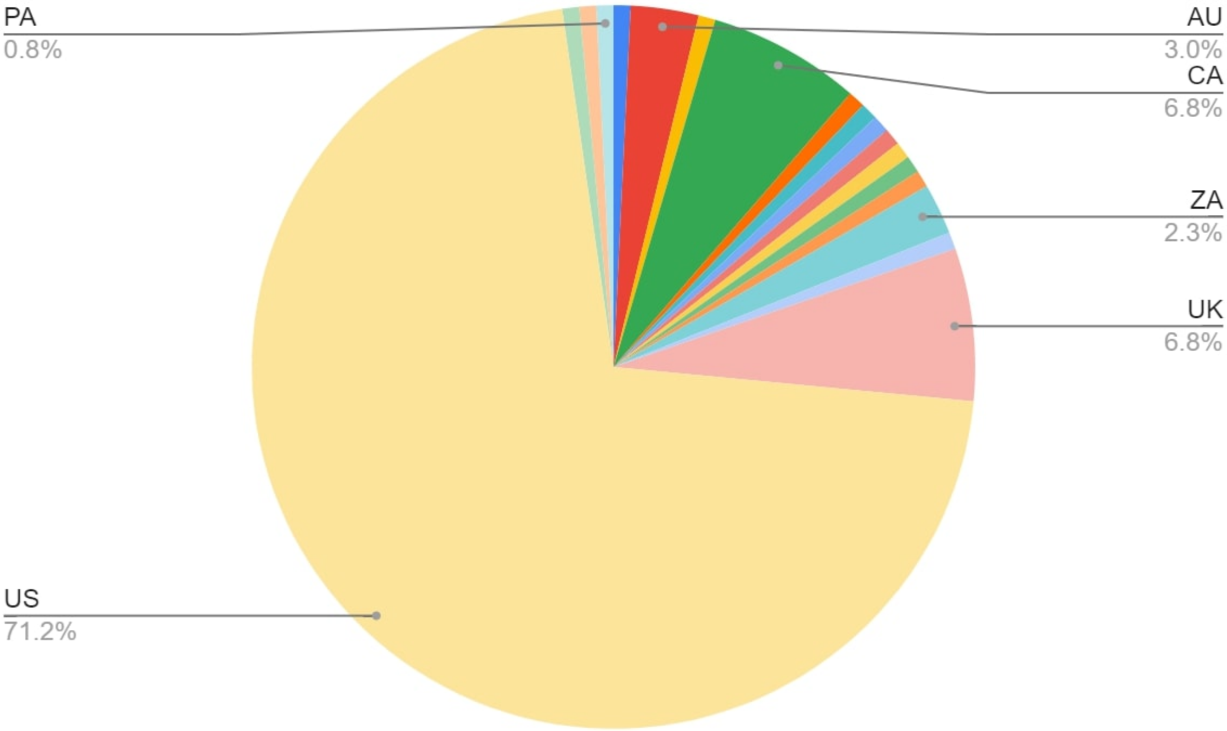


Figure 2: Victim country distribution by company headquarter location

The overwhelming number of victims in the United States ensures that the color of any of the other countries remains low. Removing the United States from the data set provides a clearer picture of the rest of the victims, especially when plotted on a world map.

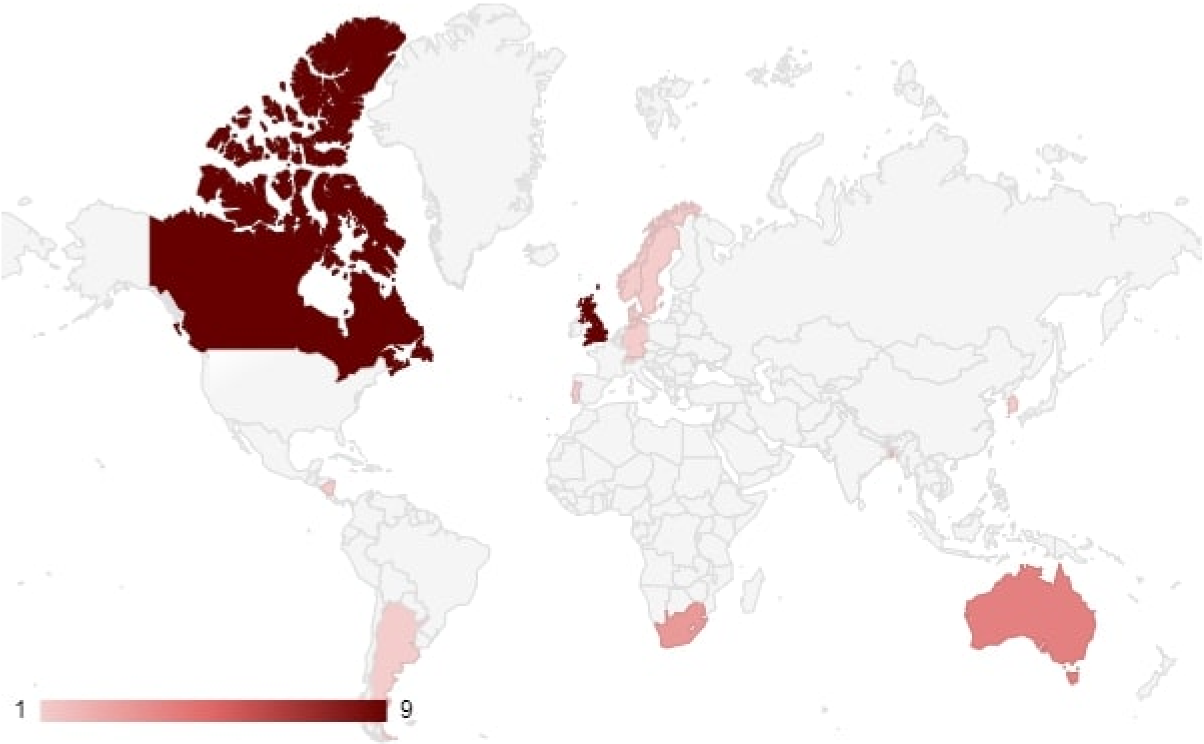


Figure 3: Victim country distribution by headquarter location plotted on a world map, excluding the United States

This shows that countries who are aligned with the United States (i.e. the United Kingdom, Canada, Australia, and South Korea) make up the majority of the victims on the list, aside from the United States themselves.

When looking at the U.S., one can plot the victims within the country per state. Similar to the way the countries are connected to a victim, the location of a company's headquarters defines the listed state.

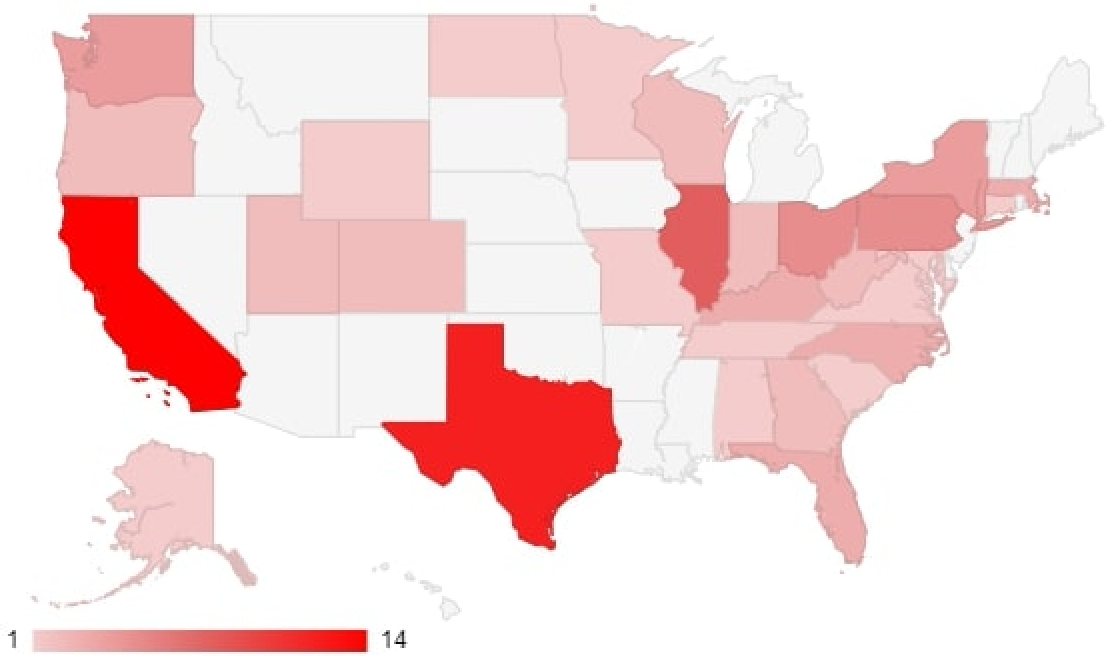


Figure 4: Victim distribution by state for victims within the United States

California and Texas are, respectively, the most populous states, which could be an explanation for the increased number of victims in those regions.

When plotting the frequency of the victims with data from April through October of this year, it shows May, June, and August as the busiest months for the blog. The cutoff date for the data is the 20th of November 2023. Do note that the victim count here is slightly higher, since some of Akira’s blogs were about the same company.

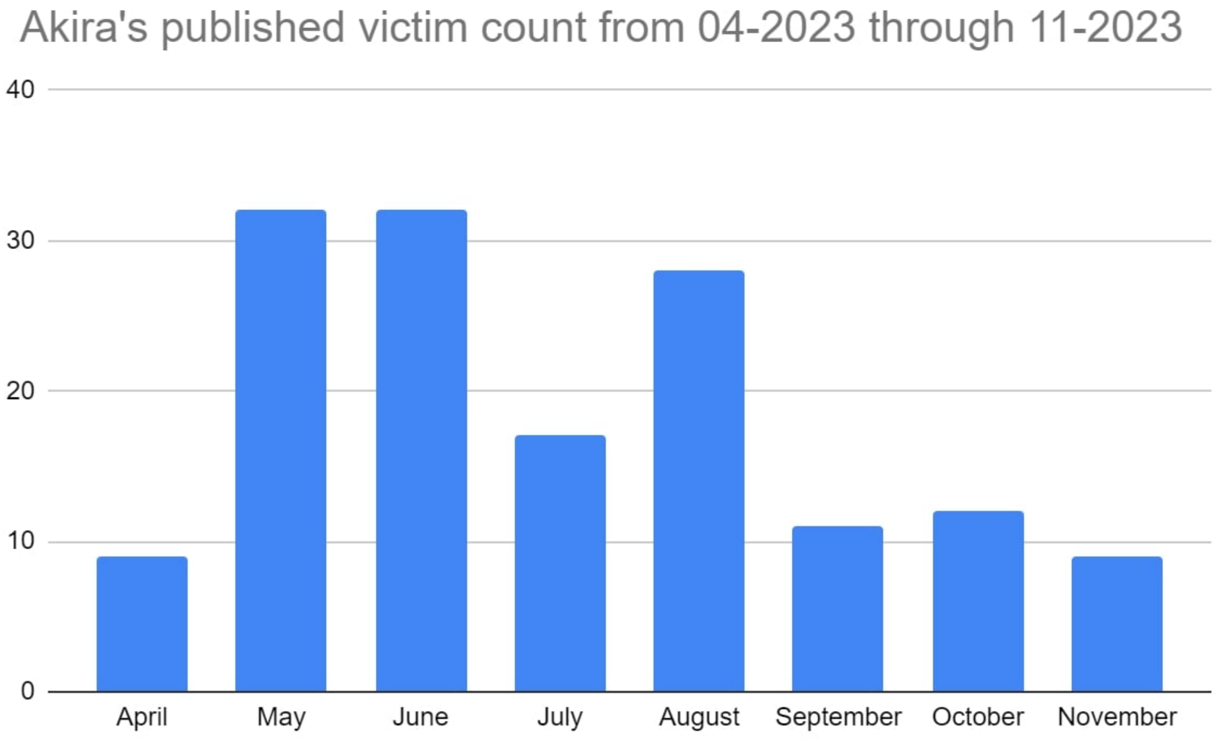


Figure 5: Number of published victims on Akira’s blog, the cutoff date for November is the 20th

Even in ‘slow’ months, the group still averaged roughly 10 published victims. Since it is unknown how many victims there are in total, and how many of those victims pay, the number of published victims is not a definitive indicator as to how many victims were made overall.

What is known about the published victims, is the primary sector of each company. Based on the names and manual verification, all sectors were mapped. For all victims in our data set, the following sectors were observed.

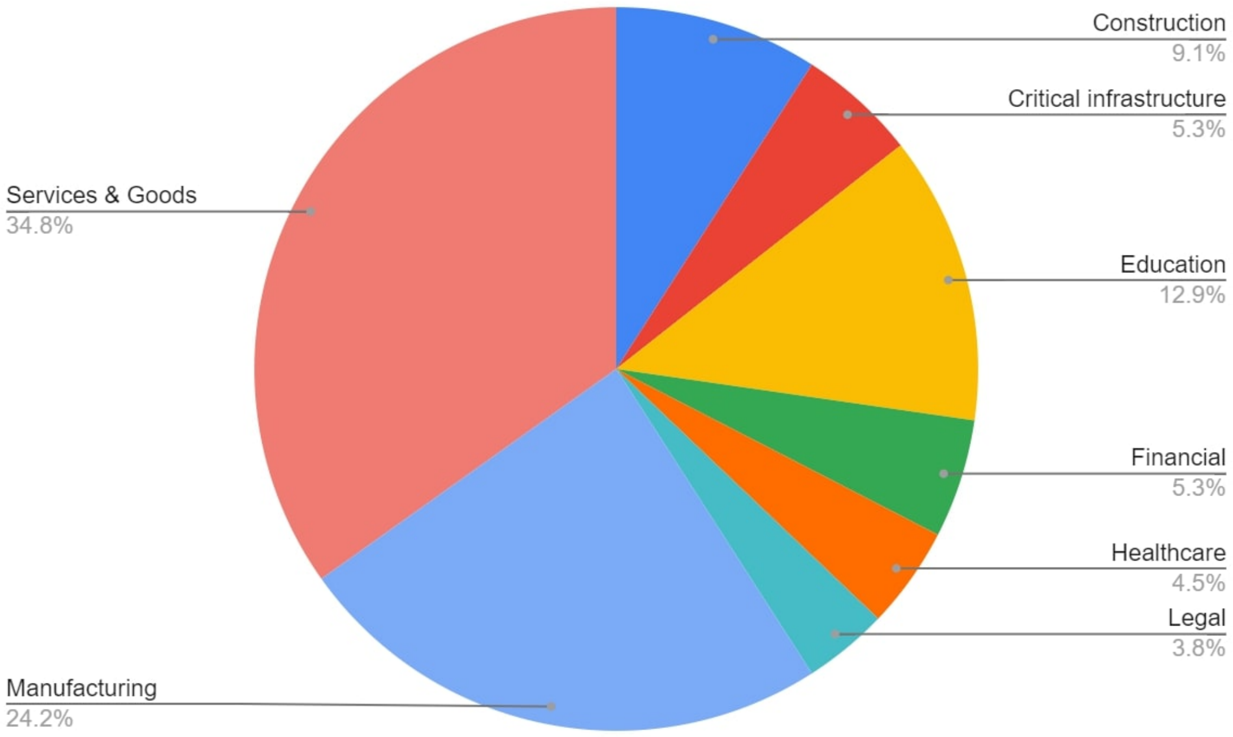


Figure 6 : The sectors of the victims
Notably here, are the major segments for services & goods, as well as manufacturing. Victims within the educational sector are often impacting thousands, since students are affected, as well as faculty staff. Critical infrastructure and legal are two sectors which might not make up a large portion of the victim base, but each victim contains a trove of information for attackers, and can impact the lives of many.

Known Tactics, Techniques, and Procedures

Note that ransomware groups often work with affiliates. These affiliates can work for multiple ransomware gangs at the same time. As such, there is no single set of TTPs which can define how the Akira ransomware can end up in one's network. In this section, multiple sources will be used to provide a clear overview. The used sources are TrendMicro, SentinelOne, Sophos, DarkTrace, and LogPoint, along with Trellix' comments and observations. Note that not all sources are used in each subsection.

For more information with regards to ransomware attacks, refer to our overview of common TTPs related to ransomware attacks.

Initial Access

The initial foothold on the system is obtained via several methods. Multi-factor authentication (MFA) exploitation (i.e. CVE-2023-20269) is mostly used in observed campaigns, along with known vulnerabilities in public facing services, such as RDP. Spear phishing is also used to gain a foothold, which is generally more effective than plain phishing, as it’s addressed to a specific user (group) and/or a relevant theme for the recipient(s).

Escalation and Lateral Movement

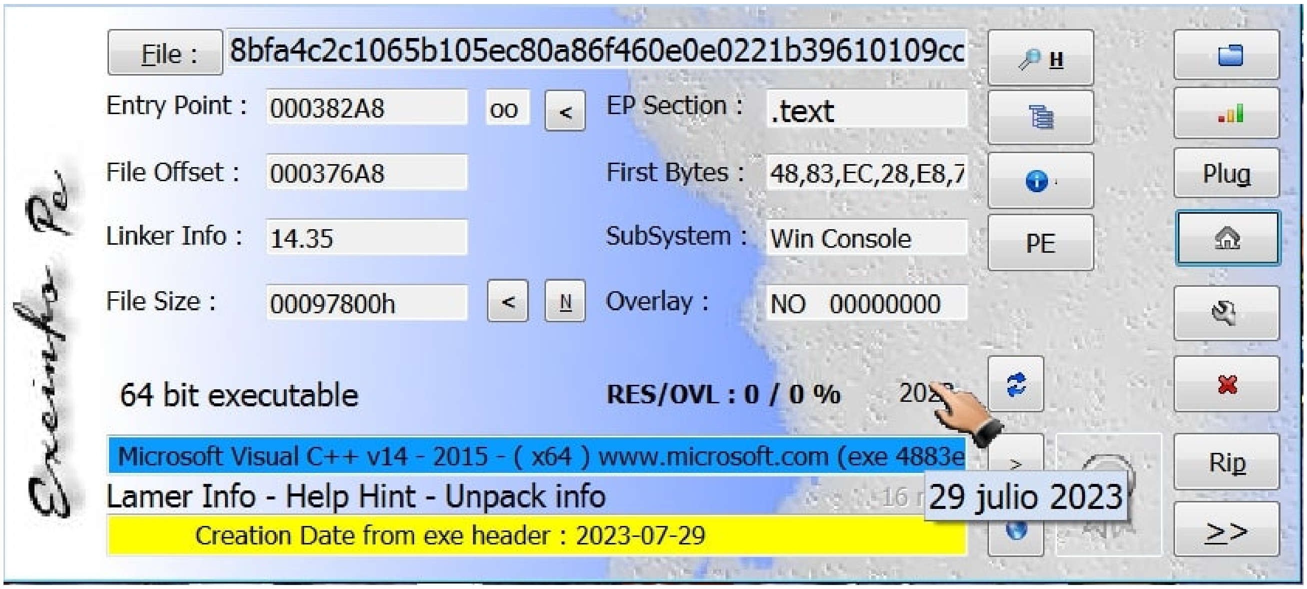
To escalate privileges and/or move laterally, LSASS dumps are used. Additionally, or alternatively, RDP is used to connect to other machines within the network while moving laterally. Other tools used are PCHunter64, LaZagne, and Mimikatz.

Data Collection and Exfiltration

Once the actors are in the system, data is exfiltrated by the actor. This way, the victim can be extorted twice: once to recover encrypted files, and once to ensure the stolen data is not made available publicly on the Akira extortion blog. To upload the gathered files, RClone, WinSCP, and FileZilla have been observed in use.

Technical analysis

The malware is written in C++ and uses benign libraries. It is compiled for 64-bit Windows.



File : 8bfa4c2c1065b105ec80a86f460e0e0221b39610109cc

Entry Point : 000382A8 oo <

EP Section : .text

File Offset : 000376A8

First Bytes : 48,83,EC,28,E8,7

Linker Info : 14.35

SubSystem : Win Console

File Size : 00097800h < N

Overlay : NO 00000000

64 bit executable

RES/OVL : 0 / 0 % 202

Microsoft Visual C++ v14 - 2015 - (x64) www.microsoft.com (exe 4883e

Lamer Info - Help Hint - Unpack info 16 r 29 julio 2023

Creation Date from exe header : 2023-07-29

Figure 7: Information about the malware

The compilation date of the analyzed sample is the 29th of July 2023, and it is a console application. Arguments to such an application are usually shared via the command-line and do not require a graphical interface of sorts.

Akira supports a number of arguments, which instruct the malware to execute certain functions. Below, the options are given.

--encryption_path or -p

Specifies the path where files will be recursively encrypted

-localonly or ly

Only encrypts the victim’s device, excluding any remote devices

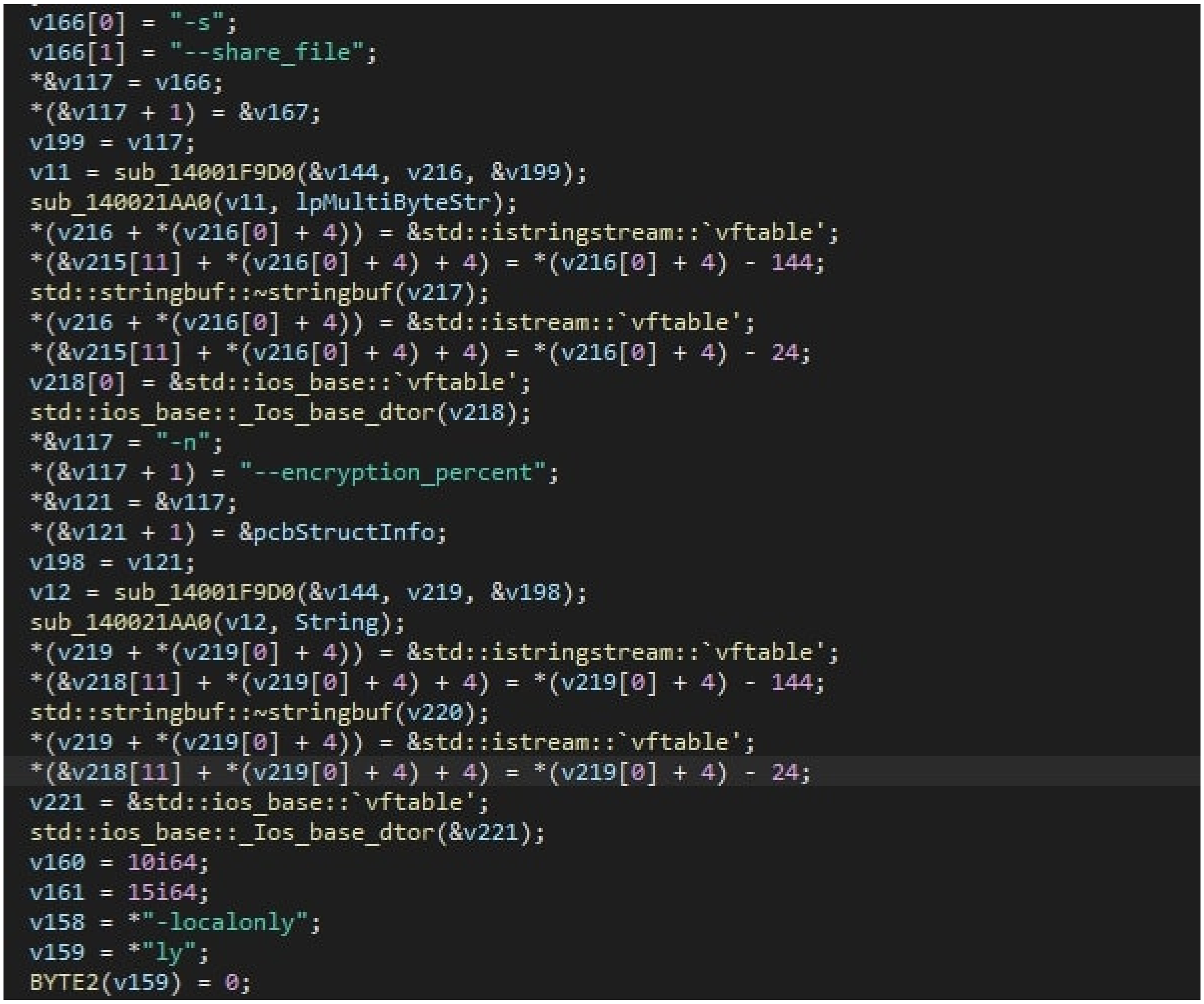
--encryption_percent or -n

The type of encryption to apply. Files until 2 megabytes in size will be encrypted for 50%. Larger files will be encrypted in multiple blocks.

--share_file or -s

A file which contains paths and devices to encrypt

The code below shows how the command-line interface arguments are handled.

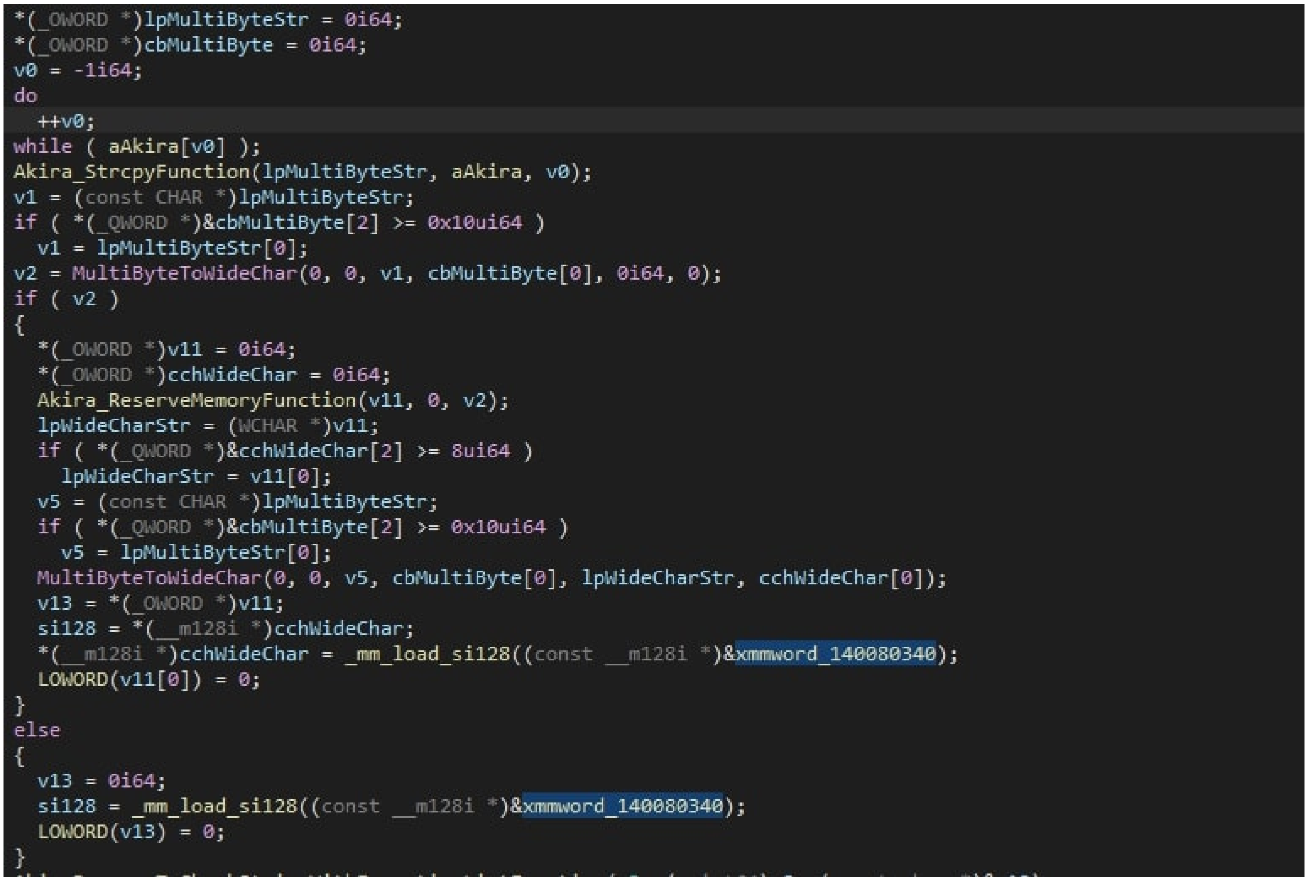


```
v166[0] = "-s";
v166[1] = "--share_file";
*&v117 = v166;
*(&v117 + 1) = &v167;
v199 = v117;
v11 = sub_14001F9D0(&v144, v216, &v199);
sub_140021AA0(v11, lpMultiByteStr);
*(v216 + *(v216[0] + 4)) = &std::istringstream::`vftable';
*(&v215[11] + *(v216[0] + 4) + 4) = *(v216[0] + 4) - 144;
std::stringbuf::~stringbuf(v217);
*(v216 + *(v216[0] + 4)) = &std::istream::`vftable';
*(&v215[11] + *(v216[0] + 4) + 4) = *(v216[0] + 4) - 24;
v218[0] = &std::ios_base::`vftable';
std::ios_base::_Ios_base_dtor(v218);
*&v117 = "-n";
*(&v117 + 1) = "--encryption_percent";
*&v121 = &v117;
*(&v121 + 1) = &pcbStructInfo;
v198 = v121;
v12 = sub_14001F9D0(&v144, v219, &v198);
sub_140021AA0(v12, String);
*(v219 + *(v219[0] + 4)) = &std::istringstream::`vftable';
*(&v218[11] + *(v219[0] + 4) + 4) = *(v219[0] + 4) - 144;
std::stringbuf::~stringbuf(v220);
*(v219 + *(v219[0] + 4)) = &std::istream::`vftable';
*(&v218[11] + *(v219[0] + 4) + 4) = *(v219[0] + 4) - 24;
v221 = &std::ios_base::`vftable';
std::ios_base::_Ios_base_dtor(&v221);
v160 = 10i64;
v161 = 15i64;
v158 = *"-localonly";
v159 = *"ly";
BYTE2(v159) = 0;
```

Figure 8: The command-line interface argument handling related code

To encrypt files on the device, the ransomware requires a command-line interface argument for either a file path to start at, or a file which contains the paths to start at. Without either of these, the execution will only result in the creation of threads. If the file reference is provided but the path does not exist, an error will be shown and the malware will terminate itself.

At first, the ".akira" string, used as the file extension for encrypted files where it appended to the original file name and extension.



```
*(_OWORD *)lpMultiByteStr = 0i64;
*(_OWORD *)cbMultiByte = 0i64;
v0 = -1i64;
do
  ++v0;
while ( aAkira[v0] );
Akira_StrcpyFunction(lpMultiByteStr, aAkira, v0);
v1 = (const CHAR *)lpMultiByteStr;
if ( *(_QWORD *)&cbMultiByte[2] >= 0x10ui64 )
  v1 = lpMultiByteStr[0];
v2 = MultiByteToWideChar(0, 0, v1, cbMultiByte[0], 0i64, 0);
if ( v2 )
{
  *(_OWORD *)v11 = 0i64;
  *(_OWORD *)cchWideChar = 0i64;
  Akira_ReserveMemoryFunction(v11, 0, v2);
  lpWideCharStr = (WCHAR *)v11;
  if ( *(_QWORD *)&cchWideChar[2] >= 8ui64 )
    lpWideCharStr = v11[0];
  v5 = (const CHAR *)lpMultiByteStr;
  if ( *(_QWORD *)&cbMultiByte[2] >= 0x10ui64 )
    v5 = lpMultiByteStr[0];
  MultiByteToWideChar(0, 0, v5, cbMultiByte[0], lpWideCharStr, cchWideChar[0]);
  v13 = *(_OWORD *)v11;
  si128 = *(__m128i *)cchWideChar;
  *(__m128i *)cchWideChar = _mm_load_si128((const __m128i *)&xmmword_140080340);
  LOWORD(v11[0]) = 0;
}
else
{
  v13 = 0i64;
  si128 = _mm_load_si128((const __m128i *)&xmmword_140080340);
  LOWORD(v13) = 0;
}
```

Figure 9: The creation of the Akira string

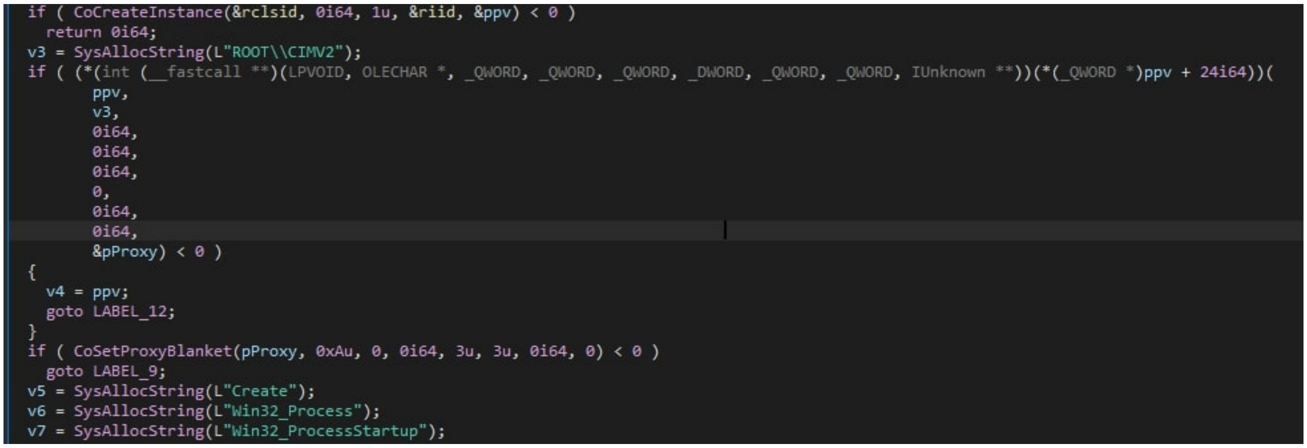
The malware excludes some file extensions, listed below, along with the “akira_readme.txt” file name to avoid encrypting the ransom note it drops.

- .exe
- .dll
- .sys
- .msi
- .lnk
- akira_readme.txt

Files with any other extension will be encrypted. Next, a PowerShell command is decrypted, and subsequently executed. The command is given below and is used to delete the shadow copies on the device. Shadow copies are used to restore files and could be used to restore encrypted files.

```
powershell.exe -Command "Get-WmiObject Win32_Shadowcopy | Remove-WmiObject".
```

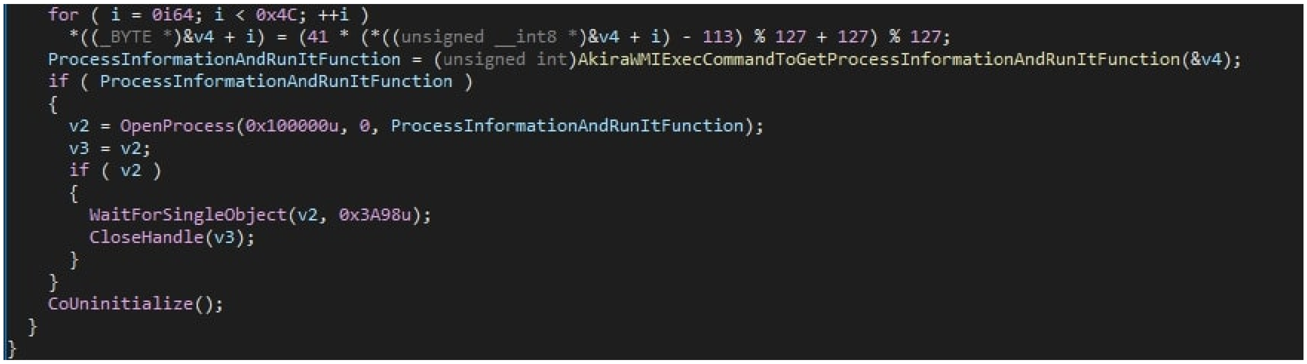
The command is executed with the help of COM objects to avoid being detected. The process ID (PID) of the newly created process is obtained and used to verify if the execution of the command was successful.



```
  if ( CoCreateInstance(&rclsid, 0i64, 1u, &riid, &ppv) < 0 )
    return 0i64;
  v3 = SysAllocString(L"ROOT\\CIMV2");
  if ( (*(int (__fastcall **)(LPVOID, OLECHAR *, _QWORD, _QWORD, _QWORD, _DWORD, _QWORD, _QWORD, IUnknown **))(*(_QWORD *)ppv + 24i64))(
         ppv,
         v3,
         0i64,
         0i64,
         0i64,
         0,
         0i64,
         0i64,
         &pProxy) < 0 )
  {
    v4 = ppv;
    goto LABEL_12;
  }
  if ( CoSetProxyBlanket(pProxy, 0xAu, 0, 0i64, 3u, 3u, 0i64, 0) < 0 )
    goto LABEL_9;
  v5 = SysAllocString(L"Create");
  v6 = SysAllocString(L"Win32_Process");
  v7 = SysAllocString(L"Win32_ProcessStartup");
```

Figure 10: The process creation

To ensure the shadow copies are deleted prior to moving on, the ransomware will use the previously obtained process ID, and wait 15 seconds. If no process ID can be obtained, it assumes the deletion has already finished, and the ransomware's execution will proceed without waiting.

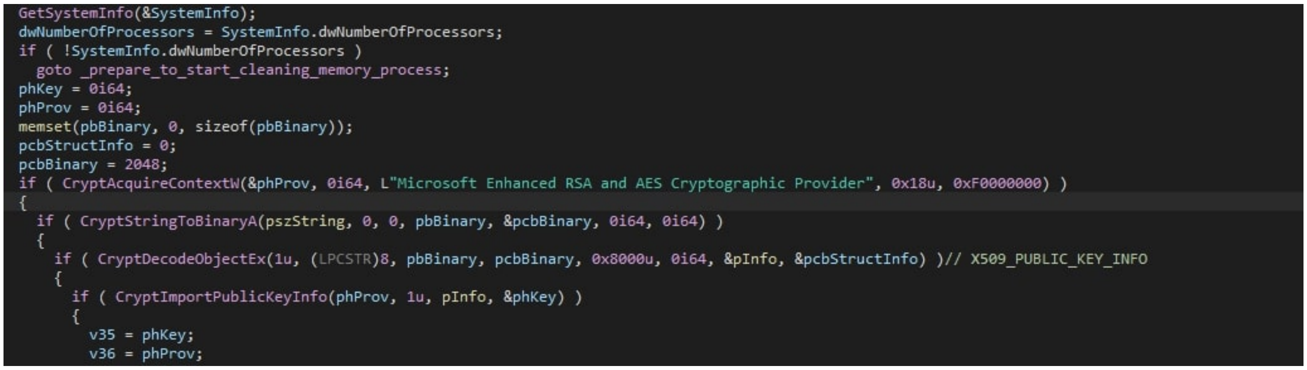


```
    for ( i = 0i64; i < 0x4C; ++i )
      *((_BYTE *)&v4 + i) = (41 * (*((unsigned __int8 *)&v4 + i) - 113) % 127 + 127) % 127;
    ProcessInformationAndRunItFunction = (unsigned int)AkiraWMIExecCommandToGetProcessInformationAndRunItFunction(&v4);
    if ( ProcessInformationAndRunItFunction )
    {
      v2 = OpenProcess(0x100000u, 0, ProcessInformationAndRunItFunction);
      v3 = v2;
      if ( v2 )
      {
        WaitForSingleObject(v2, 0x3A98u);
        CloseHandle(v3);
      }
    }
    CoUninitialize();
  }
}
```

Figure 11: Wait until the process finishes the execution.

Using GetSystemInfo, the number of processors is obtained. This number is used to determine how many threads will be created. Way more threads than the number of processors will cause inefficient thread scheduling, and too few will not utilise the available number of processors. If the obtained number of processors is zero, the malware terminates itself.

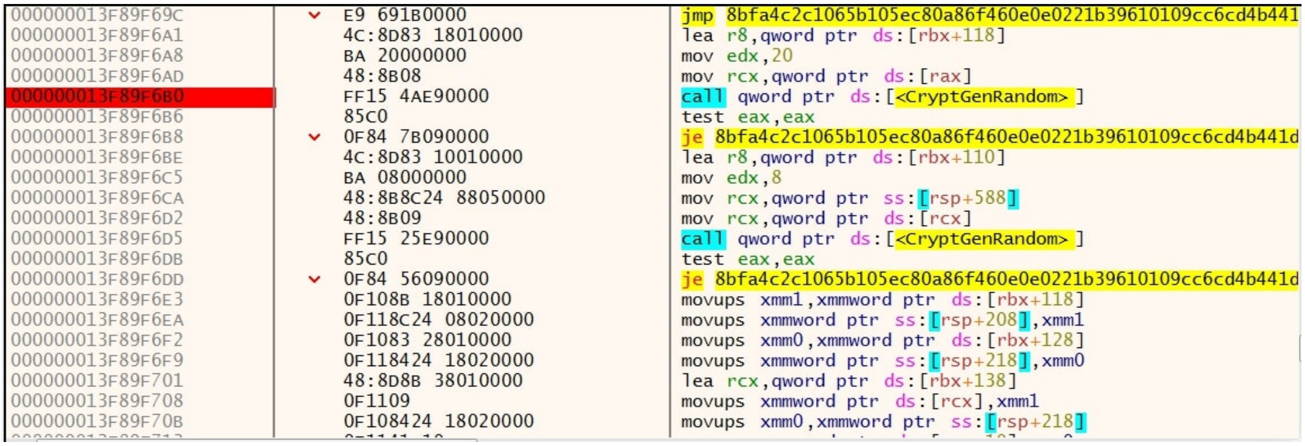
The encrypted embedded public RSA key is then decrypted using several WinAPI calls, starting with CryptAcquireContextW. This call returns a handler to the Windows cryptographic context. Using CryptStringToBinaryA, a given input string of a given format is converted into a byte string. The provided text in this case is "CRYPT_STRING_BASE64HEADER". With CryptDecodeObjectEx, the final block is obtained, which is the decrypted public key. Said key is then imported using CryptImportPublicKeyInfo, ready to be used in the subsequent encryption process.



```
GetSystemInfo(&SystemInfo);
dwNumberOfProcessors = SystemInfo.dwNumberOfProcessors;
if ( !SystemInfo.dwNumberOfProcessors )
  goto _prepare_to_start_cleaning_memory_process;
phKey = 0i64;
phProv = 0i64;
memset(pbBinary, 0, sizeof(pbBinary));
pcbStructInfo = 0;
pcbBinary = 2048;
if ( CryptAcquireContextW(&phProv, 0i64, L"Microsoft Enhanced RSA and AES Cryptographic Provider", 0x18u, 0xF0000000) )
{
  if ( CryptStringToBinaryA(pszString, 0, 0, pbBinary, &pcbBinary, 0i64, 0i64) )
  {
    if ( CryptDecodeObjectEx(1u, (LPCSTR)8, pbBinary, pcbBinary, 0x8000u, 0i64, &pInfo, &pcbStructInfo) )// X509_PUBLIC_KEY_INFO
    {
      if ( CryptImportPublicKeyInfo(phProv, 1u, pInfo, &phKey) )
      {
        v35 = phKey;
        v36 = phProv;
```

Figure 12: The import of the public key

If the previously obtained number of processors is less than 4, the stored value will be set to 4 instead. As such, a minimum of four threads are created. Next, the key and IV are generated using CryptGenRandom, and a subsequent call to CryptEncrypt. This last sequence is also observed in Conti ransomware samples. To ensure the targeted file can be encrypted, the file's attributes are read and checked using GetFileAttributesW. The file is accessed using CreateFileW, the size is obtained using GetFileSizeEx, and the used encryption algorithm is ChaCha. Again similar to Conti, the key and IV are encrypted with the ChaCha algorithm using the earlier decrypted RSA key.



```
000000013F89F69C   E9 691B0000          jmp 8bfa4c2c1065b105ec80a86f460e0e0221b39610109cc6cd4b441
000000013F89F6A1   4C:8D83 18010000     lea r8,qword ptr ds:[rbx+118]
000000013F89F6A8   BA 20000000          mov edx,20
000000013F89F6AD   48:8B08              mov rcx,qword ptr ds:[rax]
000000013F89F6B0   FF15 4AE90000        call qword ptr ds:[<CryptGenRandom>]
000000013F89F6B6   85C0                 test eax,eax
000000013F89F6B8   0F84 7B090000        je 8bfa4c2c1065b105ec80a86f460e0e0221b39610109cc6cd4b441d
000000013F89F6BE   4C:8D83 10010000     lea r8,qword ptr ds:[rbx+110]
000000013F89F6C5   BA 08000000          mov edx,8
000000013F89F6CA   48:8B8C24 88050000   mov rcx,qword ptr ss:[rsp+588]
000000013F89F6D2   48:8B09              mov rcx,qword ptr ds:[rcx]
000000013F89F6D5   FF15 25E90000        call qword ptr ds:[<CryptGenRandom>]
000000013F89F6DB   85C0                 test eax,eax
000000013F89F6DD   0F84 56090000        je 8bfa4c2c1065b105ec80a86f460e0e0221b39610109cc6cd4b441d
000000013F89F6E3   0F108B 18010000      movups xmm1,xmmword ptr ds:[rbx+118]
000000013F89F6EA   0F118C24 08020000    movups xmmword ptr ss:[rsp+208],xmm1
000000013F89F6F2   0F1083 28010000      movups xmm0,xmmword ptr ds:[rbx+128]
000000013F89F6F9   0F118424 18020000    movups xmmword ptr ss:[rsp+218],xmm0
000000013F89F701   48:8D8B 38010000     lea rcx,qword ptr ds:[rbx+138]
000000013F89F708   0F1109               movups xmmword ptr ds:[rcx],xmm1
000000013F89F70B   0F108424 18020000    movups xmm0,xmmword ptr ss:[rsp+218]
```

Figure 13: The key and IV are generated using CryptGenRandom

This information is required to decrypt the file, which is done with a given or recreated decryptor. Additionally, the ransomware will leave ransom notes on the victim's device, stating how to recover their files by paying the ransom.

Anatomy of an encrypted file

To illustrate the encryption mechanism of the ransomware, this section contains a sample file which has been encrypted. The sample file is plain text and has the ".ini" extension. Its size is 843 (0x34B) bytes in size. The encryption shows:

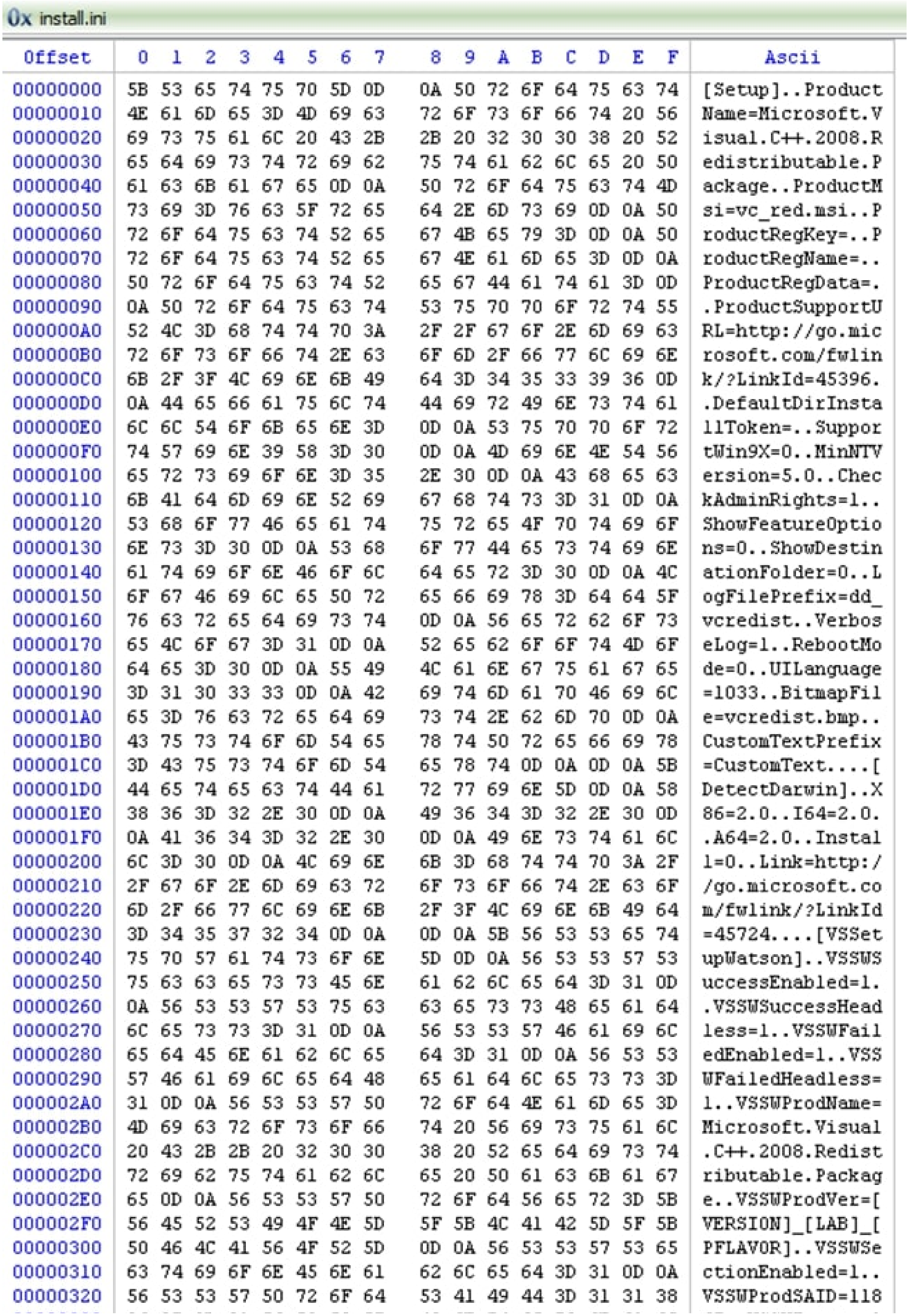
- Half of the file got encrypted
- The other half of the file remains untouched

- A block got added at the end of the file, containing the information required to decrypt said file

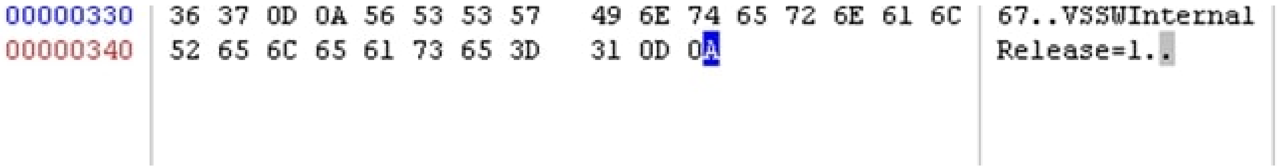
The file’s layout is as follows:

0x200 bytes block	Holds the key and IV used to encrypt the RSA-encrypted file.
Block with zeros	12 zeros
Type of encryption	One byte containing the mode used to encrypt the file. In this case it contains a 1, indicating only half of the file has been encrypted.
Version	The version of the malware, usually a value of 0x32, which equals 2 in the given sample.
Original size	8 bytes containing the original size of the encrypted file

The following screenshot shows the original file:



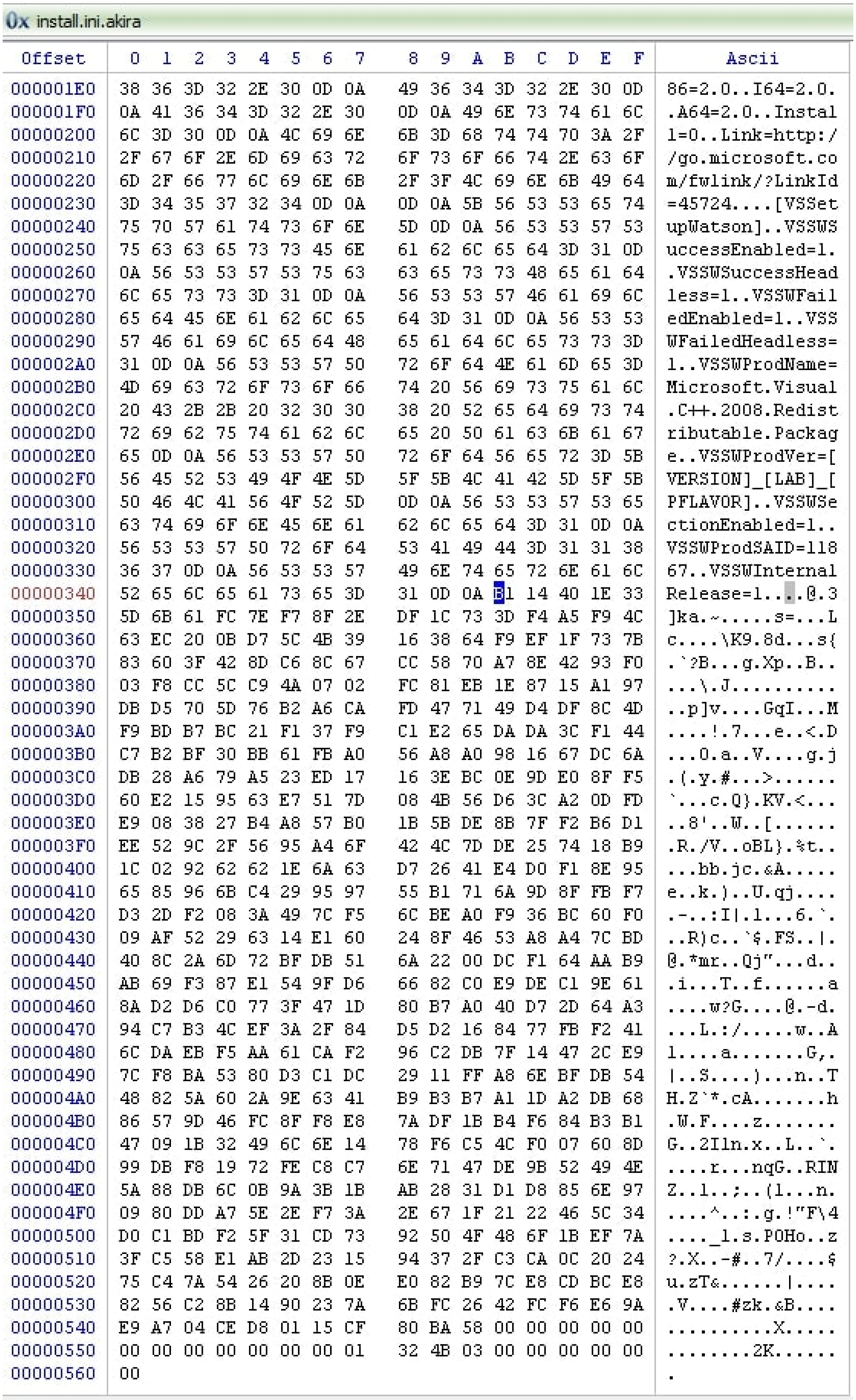
```
0x install.ini

Offset    0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F   Ascii
00000000  5B 53 65 74 75 70 5D 0D  0A 50 72 6F 64 75 63 74  [Setup]..Product
00000010  4E 61 6D 65 3D 4D 69 63  72 6F 73 6F 66 74 20 56  Name=Microsoft.V
00000020  69 73 75 61 6C 20 43 2B  2B 20 32 30 30 38 20 52  isual.C++.2008.R
00000030  65 64 69 73 74 72 69 62  75 74 61 62 6C 65 20 50  edistributable.P
00000040  61 63 6B 61 67 65 0D 0A  50 72 6F 64 75 63 74 4D  ackage..ProductM
00000050  73 69 3D 76 63 5F 72 65  64 2E 6D 73 69 0D 0A 50  si=vc_red.msi..P
00000060  72 6F 64 75 63 74 52 65  67 4B 65 79 3D 0D 0A 50  roductRegKey=..P
00000070  72 6F 64 75 63 74 52 65  67 4E 61 6D 65 3D 0D 0A  roductRegName=..
00000080  50 72 6F 64 75 63 74 52  65 67 44 61 74 61 3D 0D  ProductRegData=.
00000090  0A 50 72 6F 64 75 63 74  53 75 70 70 6F 72 74 55  .ProductSupportU
000000A0  52 4C 3D 68 74 74 70 3A  2F 2F 67 6F 2E 6D 69 63  RL=http://go.mic
000000B0  72 6F 73 6F 66 74 2E 63  6F 6D 2F 66 77 6C 69 6E  rosoft.com/fwlin
000000C0  6B 2F 3F 4C 69 6E 6B 49  64 3D 34 35 33 39 36 0D  k/?LinkId=45396.
000000D0  0A 44 65 66 61 75 6C 74  44 69 72 49 6E 73 74 61  .DefaultDirInsta
000000E0  6C 6C 54 6F 6B 65 6E 3D  0D 0A 53 75 70 70 6F 72  llToken=..Suppor
000000F0  74 57 69 6E 39 58 3D 30  0D 0A 4D 69 6E 4E 54 56  tWin9X=0..MinNTV
00000100  65 72 73 69 6F 6E 3D 35  2E 30 0D 0A 43 68 65 63  ersion=5.0..Chec
00000110  6B 41 64 6D 69 6E 52 69  67 68 74 73 3D 31 0D 0A  kAdminRights=1..
00000120  53 68 6F 77 46 65 61 74  75 72 65 4F 70 74 69 6F  ShowFeatureOptio
00000130  6E 73 3D 30 0D 0A 53 68  6F 77 44 65 73 74 69 6E  ns=0..ShowDestin
00000140  61 74 69 6F 6E 46 6F 6C  64 65 72 3D 30 0D 0A 4C  ationFolder=0..L
00000150  6F 67 46 69 6C 65 50 72  65 66 69 78 3D 64 64 5F  ogFilePrefix=dd_
00000160  76 63 72 65 64 69 73 74  0D 0A 56 65 72 62 6F 73  vcredist..Verbos
00000170  65 4C 6F 67 3D 31 0D 0A  52 65 62 6F 6F 74 4D 6F  eLog=1..RebootMo
00000180  64 65 3D 30 0D 0A 55 49  4C 61 6E 67 75 61 67 65  de=0..UILanguage
00000190  3D 31 30 33 33 0D 0A 42  69 74 6D 61 70 46 69 6C  =1033..BitmapFil
000001A0  65 3D 76 63 72 65 64 69  73 74 2E 62 6D 70 0D 0A  e=vcredist.bmp..
000001B0  43 75 73 74 6F 6D 54 65  78 74 50 72 65 66 69 78  CustomTextPrefix
000001C0  3D 43 75 73 74 6F 6D 54  65 78 74 0D 0A 0D 0A 5B  =CustomText....[
000001D0  44 65 74 65 63 74 44 61  72 77 69 6E 5D 0D 0A 58  DetectDarwin]..X
000001E0  38 36 3D 32 2E 30 0D 0A  49 36 34 3D 32 2E 30 0D  86=2.0..I64=2.0.
000001F0  0A 41 36 34 3D 32 2E 30  0D 0A 49 6E 73 74 61 6C  .A64=2.0..Instal
00000200  6C 3D 30 0D 0A 4C 69 6E  6B 3D 68 74 74 70 3A 2F  l=0..Link=http:/
00000210  2F 67 6F 2E 6D 69 63 72  6F 73 6F 66 74 2E 63 6F  /go.microsoft.co
00000220  6D 2F 66 77 6C 69 6E 6B  2F 3F 4C 69 6E 6B 49 64  m/fwlink/?LinkId
00000230  3D 34 35 37 32 34 0D 0A  0D 0A 5B 56 53 53 65 74  =45724....[VSSet
00000240  75 70 57 61 74 73 6F 6E  5D 0D 0A 56 53 53 57 53  upWatson]..VSSWS
00000250  75 63 63 65 73 73 45 6E  61 62 6C 65 64 3D 31 0D  uccessEnabled=1.
00000260  0A 56 53 53 57 53 75 63  63 65 73 73 48 65 61 64  .VSSWSuccessHead
00000270  6C 65 73 73 3D 31 0D 0A  56 53 53 57 46 61 69 6C  less=1..VSSWFail
00000280  65 64 45 6E 61 62 6C 65  64 3D 31 0D 0A 56 53 53  edEnabled=1..VSS
00000290  57 46 61 69 6C 65 64 48  65 61 64 6C 65 73 73 3D  WFailedHeadless=
000002A0  31 0D 0A 56 53 53 57 50  72 6F 64 4E 61 6D 65 3D  1..VSSWProdName=
000002B0  4D 69 63 72 6F 73 6F 66  74 20 56 69 73 75 61 6C  Microsoft.Visual
000002C0  20 43 2B 2B 20 32 30 30  38 20 52 65 64 69 73 74  .C++.2008.Redist
000002D0  72 69 62 75 74 61 62 6C  65 20 50 61 63 6B 61 67  ributable.Packag
000002E0  65 0D 0A 56 53 53 57 50  72 6F 64 56 65 72 3D 5B  e..VSSWProdVer=[
000002F0  56 45 52 53 49 4F 4E 5D  5F 5B 4C 41 42 5D 5F 5B  VERSION]_[LAB]_[
00000300  50 46 4C 41 56 4F 52 5D  0D 0A 56 53 53 57 53 65  PFLAVOR]..VSSWSe
00000310  63 74 69 6F 6E 45 6E 61  62 6C 65 64 3D 31 0D 0A  ctionEnabled=1..
00000320  56 53 53 57 50 72 6F 64  53 41 49 44 3D 31 31 38  VSSWProdSAID=118
```



```
00000330  36 37 0D 0A 56 53 53 57   49 6E 74 65 72 6E 61 6C  67..VSSWInternal
00000340  52 65 6C 65 61 73 65 3D   31 0D 0A                 Release=1..
```

Figure 14: The plaintext file

The same file post encryption:



0x install.ini.akira

```
Offset     0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F    Ascii
000001E0  38 36 3D 32 2E 30 0D 0A  49 36 34 3D 32 2E 30 0D   86=2.0..I64=2.0.
000001F0  0A 41 36 34 3D 32 2E 30  0D 0A 49 6E 73 74 61 6C   .A64=2.0..Instal
00000200  6C 3D 30 0D 0A 4C 69 6E  6B 3D 68 74 74 70 3A 2F   l=0..Link=http:/
00000210  2F 67 6F 2E 6D 69 63 72  6F 73 6F 66 74 2E 63 6F   /go.microsoft.co
00000220  6D 2F 66 77 6C 69 6E 6B  2F 3F 4C 69 6E 6B 49 64   m/fwlink/?LinkId
00000230  3D 34 35 37 32 34 0D 0A  0D 0A 5B 56 53 53 65 74   =45724....[VSSet
00000240  75 70 57 61 74 73 6F 6E  5D 0D 0A 56 53 53 57 53   upWatson]..VSSWS
00000250  75 63 63 65 73 73 45 6E  61 62 6C 65 64 3D 31 0D   uccessEnabled=1.
00000260  0A 56 53 53 57 53 75 63  63 65 73 73 48 65 61 64   .VSSWSuccessHead
00000270  6C 65 73 73 3D 31 0D 0A  56 53 53 57 46 61 69 6C   less=1..VSSWFail
00000280  65 64 45 6E 61 62 6C 65  64 3D 31 0D 0A 56 53 53   edEnabled=1..VSS
00000290  57 46 61 69 6C 65 64 48  65 61 64 6C 65 73 73 3D   WFailedHeadless=
000002A0  31 0D 0A 56 53 53 57 50  72 6F 64 4E 61 6D 65 3D   1..VSSWProdName=
000002B0  4D 69 63 72 6F 73 6F 66  74 20 56 69 73 75 61 6C   Microsoft.Visual
000002C0  20 43 2B 2B 20 32 30 30  38 20 52 65 64 69 73 74   .C++.2008.Redist
000002D0  72 69 62 75 74 61 62 6C  65 20 50 61 63 6B 61 67   ributable.Packag
000002E0  65 0D 0A 56 53 53 57 50  72 6F 64 56 65 72 3D 5B   e..VSSWProdVer=[
000002F0  56 45 52 53 49 4F 4E 5D  5F 5B 4C 41 42 5D 5F 5B   VERSION]_[LAB]_[
00000300  50 46 4C 41 56 4F 52 5D  0D 0A 56 53 53 57 53 65   PFLAVOR]..VSSWSe
00000310  63 74 69 6F 6E 45 6E 61  62 6C 65 64 3D 31 0D 0A   ctionEnabled=1..
00000320  56 53 53 57 50 72 6F 64  53 41 49 44 3D 31 31 38   VSSWProdSAID=118
00000330  36 37 0D 0A 56 53 53 57  49 6E 74 65 72 6E 61 6C   67..VSSWInternal
00000340  52 65 6C 65 61 73 65 3D  31 0D 0A B1 14 40 1E 33   Release=1....@.3
00000350  5D 6B 61 FC 7E F7 8F 2E  DF 1C 73 3D F4 A5 F9 4C   ]ka.~.....s=...L
00000360  63 EC 20 0B D7 5C 4B 39  16 38 64 F9 EF 1F 73 7B   c....\K9.8d...s{
00000370  83 60 3F 42 8D C6 8C 67  CC 58 70 A7 8E 42 93 F0   .`?B...g.Xp..B..
00000380  03 F8 CC 5C C9 4A 07 02  FC 81 EB 1E 87 15 A1 97   ...\.J..........
00000390  DB D5 70 5D 76 B2 A6 CA  FD 47 71 49 D4 DF 8C 4D   ..p]v....GqI...M
000003A0  F9 BD B7 BC 21 F1 37 F9  C1 E2 65 DA DA 3C F1 44   ....!.7...e..<.D
000003B0  C7 B2 BF 30 BB 61 FB A0  56 A8 A0 98 16 67 DC 6A   ...0.a..V....g.j
000003C0  DB 28 A6 79 A5 23 ED 17  16 3E BC 0E 9D E0 8F F5   .(.y.#...>......
000003D0  60 E2 15 95 63 E7 51 7D  08 4B 56 D6 3C A2 0D FD   `...c.Q}.KV.<...
000003E0  E9 08 38 27 B4 A8 57 B0  1B 5B DE 8B 7F F2 B6 D1   ..8'..W..[......
000003F0  EE 52 9C 2F 56 95 A4 6F  42 4C 7D DE 25 74 18 B9   .R./V..oBL}.%t..
00000400  1C 02 92 62 62 1E 6A 63  D7 26 41 E4 D0 F1 8E 95   ...bb.jc.&A.....
00000410  65 85 96 6B C4 29 95 97  55 B1 71 6A 9D 8F FB F7   e..k.)..U.qj....
00000420  D3 2D F2 08 3A 49 7C F5  6C BE A0 F9 36 BC 60 F0   .-..:I|.l...6.`.
00000430  09 AF 52 29 63 14 E1 60  24 8F 46 53 A8 A4 7C BD   ..R)c..`$.FS..|.
00000440  40 8C 2A 6D 72 BF DB 51  6A 22 00 DC F1 64 AA B9   @.*mr..Qj"...d..
00000450  AB 69 F3 87 E1 54 9F D6  66 82 C0 E9 DE C1 9E 61   .i...T..f......a
00000460  8A D2 D6 C0 77 3F 47 1D  80 B7 A0 40 D7 2D 64 A3   ....w?G....@.-d.
00000470  94 C7 B3 4C EF 3A 2F 84  D5 D2 16 84 77 FB F2 41   ...L.:/.....w..A
00000480  6C DA EB F5 AA 61 CA F2  96 C2 DB 7F 14 47 2C E9   l....a.......G,.
00000490  7C F8 BA 53 80 D3 C1 DC  29 11 FF A8 6E BF DB 54   |..S....)...n..T
000004A0  48 82 5A 60 2A 9E 63 41  B9 B3 B7 A1 1D A2 DB 68   H.Z`*.cA.......h
000004B0  86 57 9D 46 FC 8F F8 E8  7A DF 1B B4 F6 84 B3 B1   .W.F....z.......
000004C0  47 09 1B 32 49 6C 6E 14  78 F6 C5 4C F0 07 60 8D   G..2Iln.x..L..`.
000004D0  99 DB F8 19 72 FE C8 C7  6E 71 47 DE 9B 52 49 4E   ....r...nqG..RIN
000004E0  5A 88 DB 6C 0B 9A 3B 1B  AB 28 31 D1 D8 85 6E 97   Z..l..;..(1...n.
000004F0  09 80 DD A7 5E 2E F7 3A  2E 67 1F 21 22 46 5C 34   ....^..:.g.!"F\4
00000500  D0 C1 BD F2 5F 31 CD 73  92 50 4F 48 6F 1B EF 7A   ...._1.s.POHo..z
00000510  3F C5 58 E1 AB 2D 23 15  94 37 2F C3 CA 0C 20 24   ?.X..-#..7/....$
00000520  75 C4 7A 54 26 20 8B 0E  E0 82 B9 7C E8 CD BC E8   u.zT&......|....
00000530  82 56 C2 8B 14 90 23 7A  6B FC 26 42 FC F6 E6 9A   .V....#zk.&B....
00000540  E9 A7 04 CE D8 01 15 CF  80 BA 58 00 00 00 00 00   ..........X.....
00000550  00 00 00 00 00 00 00 01  32 4B 03 00 00 00 00 00   ........2K......
00000560  00                                                 .
```

Figure 15: The encrypted file

If the file is larger than 2 megabytes (based on 1000 bytes per kilobyte, and so forth, the total number of bytes is 2000000 in total), the malware will split the file in four blocks, where each block is partially encrypted. The goal here is to ensure that the file is unusable by the victim, while being able to encrypt more files per time unit, since files are only partially encrypted.

MITRE ATT&CK Techniques

Below are the relevant MITRE ATT&CK Techniques for the Akira ransomware.

This document and the information contained herein describes computer security research for educational purposes only and the convenience of Trellix customers.