PACE UNIVERSITY

CYB633: MALWARE ANALYSIS AND REVERSE ENGINEERING

Malware Analysis Project Report on

Sepsis malware

Mohammed Fouzan Aamiri

Contents

1. **SUMMARY** 3
2. **IDENTIFICATION** 4
3. **Capabilities & Characteristics** 5
4. **Dependencies** 9
5. **Static Analysis** 13
6. **Dynamic Analysis** 21

1. SUMMARY

Sepsis Ransomware emerges as a sophisticated threat capable of encrypting files and coercing victims into paying ransom for decryption. Its multifaceted approach encompasses privilege checking, encryption, and ransom note delivery, ensuring persistent access and maximizing impact. The ransomware utilizes a hard-coded public key for encryption and embeds HTML ransom messages within its binary, streamlining the ransom payment process for victims. Understanding its mechanisms is pivotal for developing effective mitigation strategies against ransomware attacks.

Key Takeaways:

1. **Sophisticated Encryption Mechanism:** Sepsis Ransomware employs AES-128 CBC encryption with a hard-coded public key, ensuring strong encryption of files on infected systems. This mechanism, combined with evasion tactics and persistence techniques, underscores the ransomware's potency and resilience to traditional defenses.

2. **Coercive Ransom Note Delivery:** Ransomware presents victims with HTML ransom messages, coercing them into paying ransom in bitcoins for decryption. By embedding these messages within its binary, Sepsis streamlines the ransom payment process, increasing the likelihood of victim compliance and successful ransom collection.

3. **Risk Mitigation Imperative:** Organizations and individuals must prioritize proactive measures, including regular backups, endpoint protection, and user education, to mitigate the risks posed by Sepsis and similar ransomware variants. Additionally, collaboration with law enforcement and cybersecurity professionals is essential for swift response and recovery efforts in the event of a ransomware attack.

Malware Sample Download:

URL:
https://malshare.com/sample.php?action=detail&hash=3c7d9ecd35b21a2a8fac7cce4fdb3e11c1950d5a02a0c0b369f4082acf00bf9a

OR

https://drive.google.com/file/d/1HIEeNoLN6xjfcVa0TGmGP_FdbBxt6csJ/view?usp=sharing

Downloading and analyzing the malware sample can provide valuable insights into its behavior and facilitate the development of effective detection and mitigation strategies against Sepsis Ransomware and similar threats.

2. IDENTIFICATION

2.1 Filename: *3c7d9ecd35b21a2a8fac7cce4fdb3e11c1950d5a02a0c0b369f4082acf00bf9a.*

File size: *16.50 KB (16896 bytes).*

File type: *Win32 EXE.*

2.2 **Mac time stamps:**

MODIFIED: *2024-04-09 03:40:53 UTC.*

ACCESSED: *2024-04-27*

CREATED: *2018-05-09 13:29:35 UTC.*

2.3 Hashes (md5, sha1, sha256, fuzzy):

MD5: 1221ac9d607af73c65fd6c62bec3d249

SHA-1: 518d5a0a8025147b9e29821bccdaf3b42c0d01db

SHA-256: 3c7d9ecd35b21a2a8fac7cce4fdb3e11c1950d5a02a0c0b369f4082acf00bf9a

Fuzzy Hash (SSDEEP):

384:ET1G/RzvHvTb7r7LsBphNizjHnf3fXAsNNN5BZ/KiTSLgNsa9JGiAd7ei:E4rRKiLsMGiYV

2.4 Signing Information (Certificates):

2.5 Packer information: *DetectItEasy identifies the compiler as "Microsoft Visual C/C++" with version 19.13.26131, and the linker as "Microsoft Linker" with version 6.00.8168.*

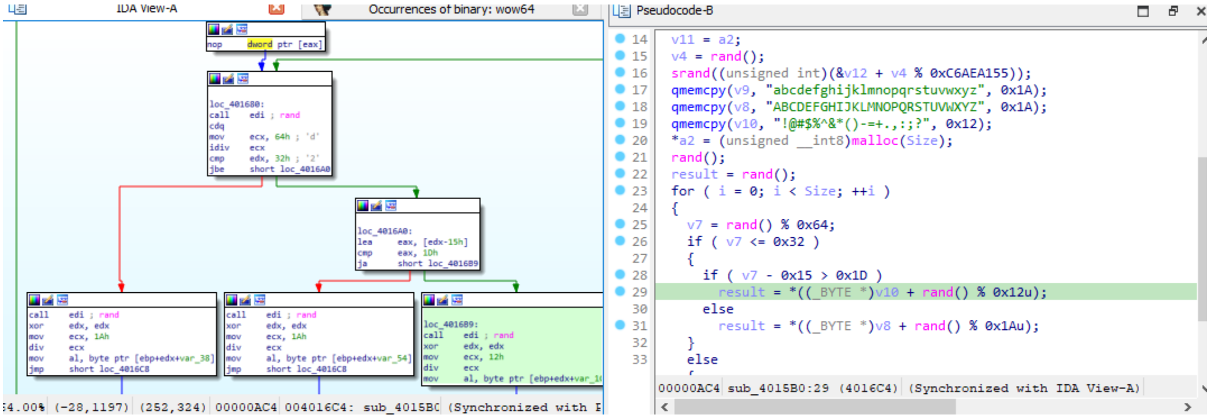
2.6 Aliases: *sepsis.ransom, ,SadeghSample_5afc4a7c9931365644caeb64.exe, Sepsis_Ransomware.exe*

2.7 Dependencies: *CRYPT32.dll, SHLWAPI.dll, MSVCRT.dllKERNEL32.dll, ADVAPI32.dll, SHELL32.dll*

3. Capabilities & Characteristics

Encryption Capability

Malware uses AES encryption to encrypt the files. The key and IV are passed as values.



```
nop     dword ptr [eax]
```

```
loc_401680:
call    edi ; rand
cdq
mov     ecx, 64h ; 'd'
idiv    ecx
cmp     edx, 32h ; '2'
jbe     short loc_4016A0
```

```
loc_4016A0:
lea     eax, [edx-15h]
cmp     eax, 1Dh
ja      short loc_401689
```

```
call    edi ; rand
xor     edx, edx
mov     ecx, 1Ah
div     ecx
mov     al, byte ptr [ebp+edx+var_38]
jmp     short loc_4016C8
```

```
call    edi ; rand
xor     edx, edx
mov     ecx, 1Ah
div     ecx
mov     al, byte ptr [ebp+edx+var_54]
jmp     short loc_4016C8
```

```
loc_401689:
call    edi ; rand
xor     edx, edx
mov     ecx, 12h
div     ecx
mov     al, byte ptr [ebp+edx+var_1(
```

```
54.00% (-28,1197) (252,324) 00000AC4 004016C4: sub_4015BC (Synchronized with I
```

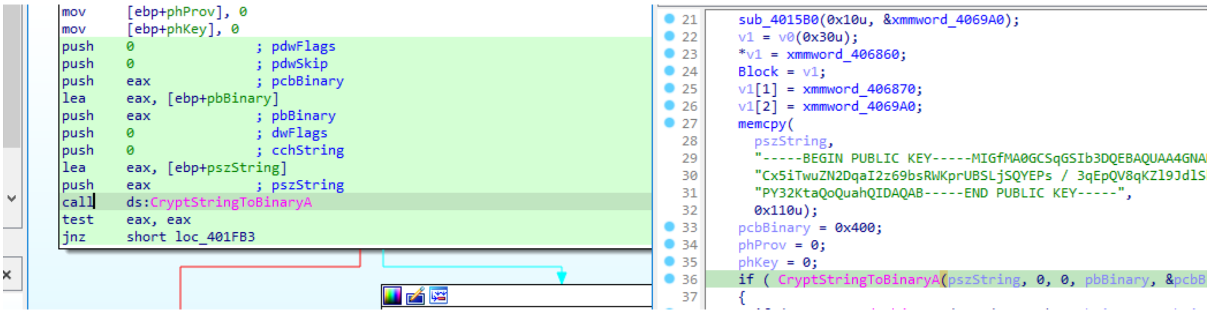
```
v11 = a2;
v4 = rand();
srand((unsigned int)(&v12 + v4 % 0xC6AEA155));
qmemcpy(v9, "abcdefghijklmnopqrstuvwxyz", 0x1A);
qmemcpy(v8, "ABCDEFGHIJKLMNOPQRSTUVWXYZ", 0x1A);
qmemcpy(v10, "!@#$%^&*()-=+.,:;?", 0x12);
*a2 = (unsigned __int8)malloc(Size);
rand();
result = rand();
for ( i = 0; i < Size; ++i )
{
  v7 = rand() % 0x64;
  if ( v7 <= 0x32 )
  {
    if ( v7 - 0x15 > 0x1D )
      result = *((_BYTE *)v10 + rand() % 0x12u);
    else
      result = *((_BYTE *)v8 + rand() % 0x1Au);
  }
  else
```

```
00000AC4 sub_4015B0:29 (4016C4) (Synchronized with IDA View-A)
```

Looking at the key forming function that the malware first forms 2 string buffers of 32bytes and 16 bytes respectively using a function which uses **"rand"** function calls to form string.

The 32byte string is used as the key and 16-byte string as IV by the ransomware for performing encryption. IV is concatenated with the key using **memcpy**.

"CryptStringToBinaryA" function converts the hard-coded public key from PEM format to DER format, Unlike PEM format a DER file should not have any BEGIN/END.



```
mov     [ebp+phProv], 0
mov     [ebp+phKey], 0
push    0               ; pdwFlags
push    0               ; pdwSkip
push    eax             ; pcbBinary
lea     eax, [ebp+pbBinary]
push    eax             ; pbBinary
push    0               ; dwFlags
push    0               ; cchString
lea     eax, [ebp+pszString]
push    eax             ; pszString
call    ds:CryptStringToBinaryA
test    eax, eax
jnz     short loc_401FB3
```

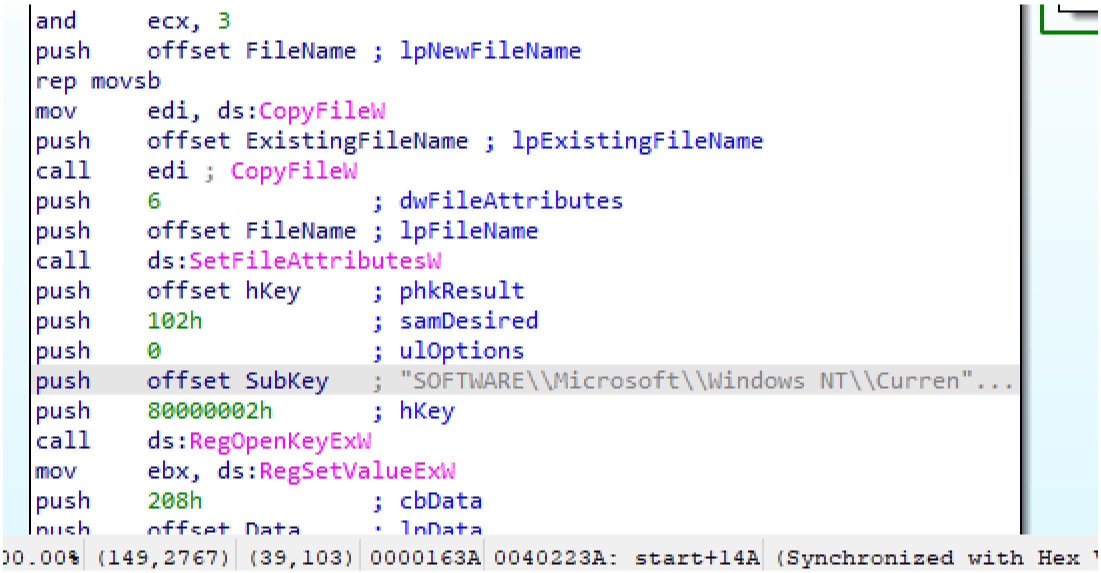
```
sub_4015B0(0x10u, &xmmword_4069A0);
v1 = v0(0x30u);
*v1 = xmmword_406860;
Block = v1;
v1[1] = xmmword_406870;
v1[2] = xmmword_4069A0;
memcpy(
  pszString,
  "-----BEGIN PUBLIC KEY-----MIGfMA0GCSqGSIb3DQEBAQUAA4GNA
  "Cx5iTwuZN2DqaI2z69bsRWKprUBSLjSQYEPs / 3qEpQV8qKZl9JdlS
  "PY32KtaQoQuahQIDAQAB-----END PUBLIC KEY-----",
  0x110u);
pcbBinary = 0x400;
phProv = 0;
phKey = 0;
if ( CryptStringToBinaryA(pszString, 0, 0, pbBinary, &pcbB
{
```

The key that was created had a size of 32bytes, but the malware uses only first 16 bytes of the key.

Capability to Modify Registry Keys:

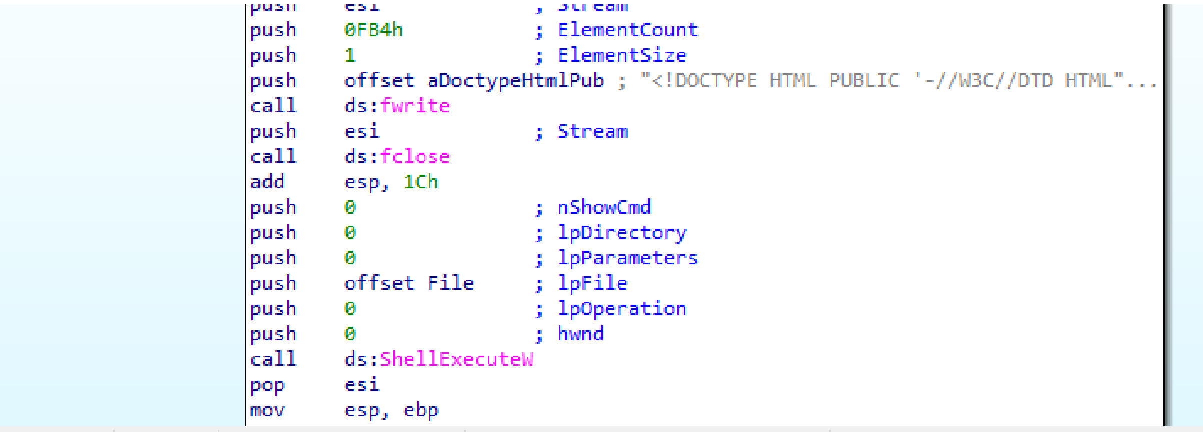
The malware initially accesses the registry key "SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon" with edit privileges. This key, Winlogon, is responsible for automatic login functionality in Windows systems. Enabling auto-login can pose a security risk since it allows anyone with physical access to the computer to gain entry to its contents, including connected networks. Moreover, when auto-login is enabled, the password is stored in plain text within the registry, making it accessible to the Authenticated Users group remotely.

The malware proceeds by creating a new key named "Shell" and assigning its value as the path to "svchost.exe".



```
and     ecx, 3
push    offset FileName ; lpNewFileName
rep movsb
mov     edi, ds:CopyFileW
push    offset ExistingFileName ; lpExistingFileName
call    edi ; CopyFileW
push    6               ; dwFileAttributes
push    offset FileName ; lpFileName
call    ds:SetFileAttributesW
push    offset hKey     ; phkResult
push    102h            ; samDesired
push    0               ; ulOptions
push    offset SubKey   ; "SOFTWARE\\Microsoft\\Windows NT\\Curren"...
push    80000002h       ; hKey
call    ds:RegOpenKeyExW
mov     ebx, ds:RegSetValueExW
push    208h            ; cbData
push    offset Data     ; lpData
00.00% (149,2767) (39,103) 0000163A 0040223A: start+14A (Synchronized with Hex
```

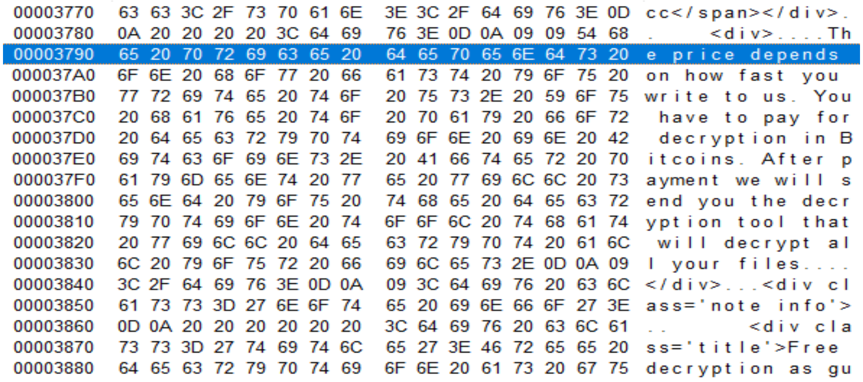
The malware executes Eventvwe.exe using the "ShellExecuteW" function. Event Viewer is a component of Microsoft's NT line of operating systems, allowing administrators and users to view event logs on both local and remote machines.



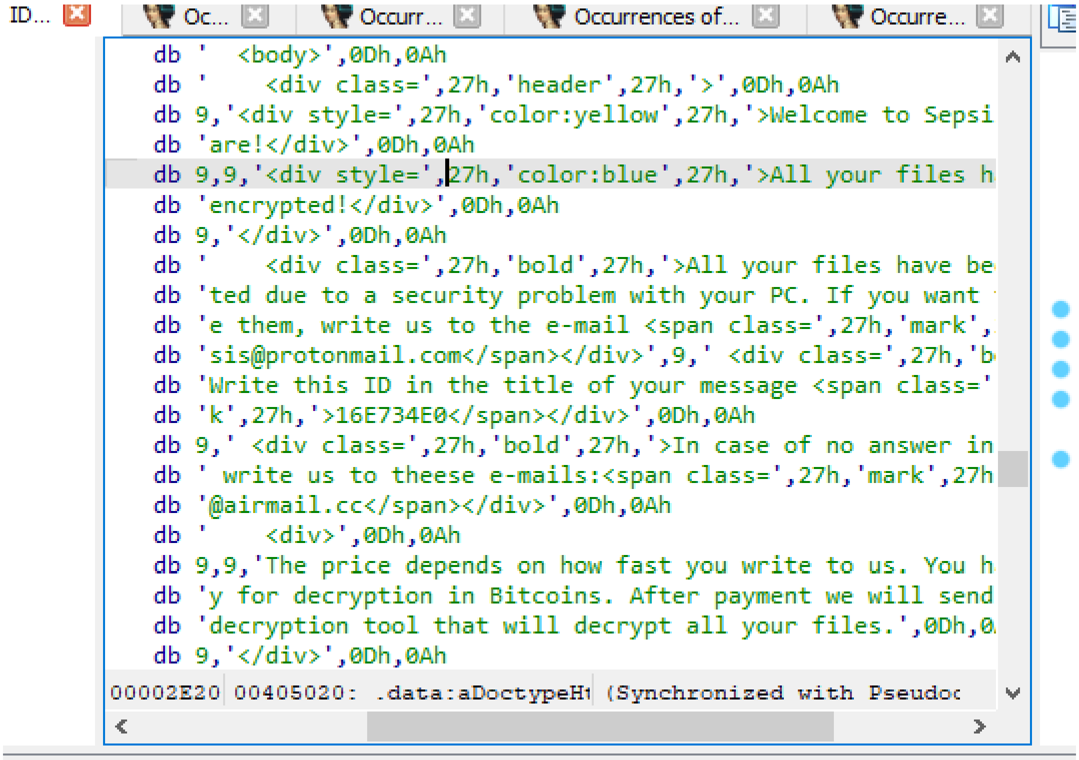
```
push    esi             ; Stream
push    0FB4h           ; ElementCount
push    1               ; ElementSize
push    offset aDoctypeHtmlPub ; "<!DOCTYPE HTML PUBLIC '-//W3C//DTD HTML"...
call    ds:fwrite
push    esi             ; Stream
call    ds:fclose
add     esp, 1Ch
push    0               ; nShowCmd
push    0               ; lpDirectory
push    0               ; lpParameters
push    offset File     ; lpFile
push    0               ; lpOperation
push    0               ; hwnd
call    ds:ShellExecuteW
pop     esi
mov     esp, ebp
```

Characteristics of user Interaction and Ransom Notes:

Now let’s explore how the ransomware interacts with the user, including the display of ransom notes or instructions for paying the ransom. Within the malware, we observe the presence of HTML text intended for display to the victim.

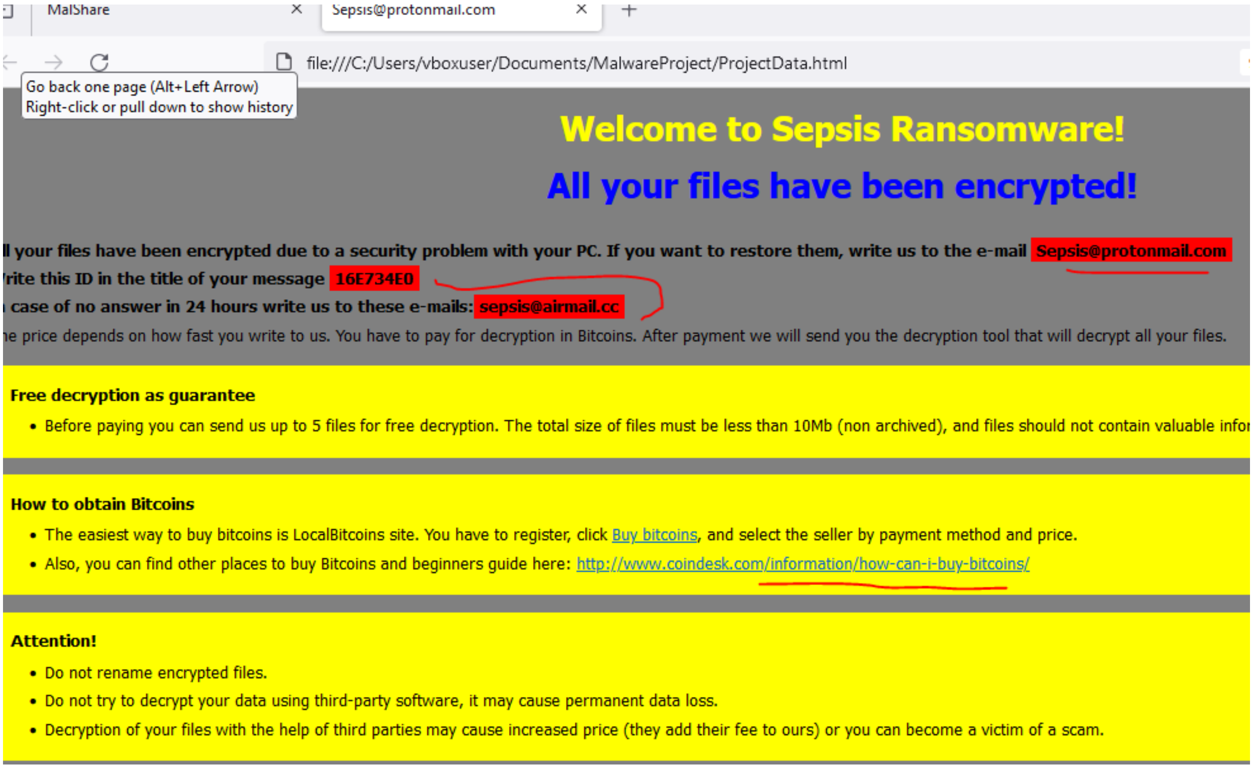
1. 

```
00003770  63 63 3C 2F 73 70 61 6E  3E 3C 2F 64 69 76 3E 0D  cc</span></div>.
00003780  0A 20 20 20 20 3C 64 69  76 3E 0D 0A 09 09 54 68  .    <div>....Th
00003790  65 20 70 72 69 63 65 20  64 65 70 65 6E 64 73 20  e price depends 
000037A0  6F 6E 20 68 6F 77 20 66  61 73 74 20 79 6F 75 20  on how fast you 
000037B0  77 72 69 74 65 20 74 6F  20 75 73 2E 20 59 6F 75  write to us. You
000037C0  20 68 61 76 65 20 74 6F  20 70 61 79 20 66 6F 72   have to pay for
000037D0  20 64 65 63 72 79 70 74  69 6F 6E 20 69 6E 20 42   decryption in B
000037E0  69 74 63 6F 69 6E 73 2E  20 41 66 74 65 72 20 70  itcoins. After p
000037F0  61 79 6D 65 6E 74 20 77  65 20 77 69 6C 6C 20 73  ayment we will s
00003800  65 6E 64 20 79 6F 75 20  74 68 65 20 64 65 63 72  end you the decr
00003810  79 70 74 69 6F 6E 20 74  6F 6F 6C 20 74 68 61 74  yption tool that
00003820  20 77 69 6C 6C 20 64 65  63 72 79 70 74 20 61 6C   will decrypt al
00003830  6C 20 79 6F 75 72 20 66  69 6C 65 73 2E 0D 0A 09  l your files....
00003840  3C 2F 64 69 76 3E 0D 0A  09 3C 64 69 76 20 63 6C  </div>...<div cl
00003850  61 73 73 3D 27 6E 6F 74  65 20 69 6E 66 6F 27 3E  ass='note info'>
00003860  0D 0A 20 20 20 20 20 20  3C 64 69 76 20 63 6C 61  ..      <div cla
00003870  73 73 3D 27 74 69 74 6C  65 27 3E 46 72 65 65 20  ss='title'>Free 
00003880  64 65 63 72 79 70 74 69  6F 6E 20 61 73 20 67 75  decryption as gu
```



```
db '  <body>',0Dh,0Ah
db '    <div class=',27h,'header',27h,'>',0Dh,0Ah
db 9,'<div style=',27h,'color:yellow',27h,'>Welcome to Sepsi
db 'are!</div>',0Dh,0Ah
db 9,9,'<div style=',27h,'color:blue',27h,'>All your files h
db 'encrypted!</div>',0Dh,0Ah
db 9,'</div>',0Dh,0Ah
db '    <div class=',27h,'bold',27h,'>All your files have be
db 'ted due to a security problem with your PC. If you want 
db 'e them, write us to the e-mail <span class=',27h,'mark',
db 'sis@protonmail.com</span></div>',9,' <div class=',27h,'b
db 'Write this ID in the title of your message <span class='
db 'k',27h,'>16E734E0</span></div>',0Dh,0Ah
db 9,' <div class=',27h,'bold',27h,'>In case of no answer in
db ' write us to theese e-mails:<span class=',27h,'mark',27h
db '@airmail.cc</span></div>',0Dh,0Ah
db '    <div>',0Dh,0Ah
db 9,9,'The price depends on how fast you write to us. You h
db 'y for decryption in Bitcoins. After payment we will send
db 'decryption tool that will decrypt all your files.',0Dh,0
db 9,'</div>',0Dh,0Ah
00002E20 00405020: .data:aDoctypeHt (Synchronized with Pseudoc
```

The HTML code is stored as a string buffer embedded within the malware. This code is then written to a file using **fwrite** and subsequently executed using the "**ShellExecuteW**" API.
When you take the HTML code out of it and open in browser, you will get this:


MalShare
Sepsis@protonmail.com
file:///C:/Users/vboxuser/Documents/MalwareProject/ProjectData.html
Go back one page (Alt+Left Arrow)
Right-click or pull down to show history

Welcome to Sepsis Ransomware!

All your files have been encrypted!

ll your files have been encrypted due to a security problem with your PC. If you want to restore them, write us to the e-mail Sepsis@protonmail.com

rite this ID in the title of your message 16E734E0

case of no answer in 24 hours write us to these e-mails: sepsis@airmail.cc

he price depends on how fast you write to us. You have to pay for decryption in Bitcoins. After payment we will send you the decryption tool that will decrypt all your files.

Free decryption as guarantee

- Before paying you can send us up to 5 files for free decryption. The total size of files must be less than 10Mb (non archived), and files should not contain valuable infor

How to obtain Bitcoins

- The easiest way to buy bitcoins is LocalBitcoins site. You have to register, click Buy bitcoins, and select the seller by payment method and price.
- Also, you can find other places to buy Bitcoins and beginners guide here: http://www.coindesk.com/information/how-can-i-buy-bitcoins/

Attention!

- Do not rename encrypted files.
- Do not try to decrypt your data using third-party software, it may cause permanent data loss.
- Decryption of your files with the help of third parties may cause increased price (they add their fee to ours) or you can become a victim of a scam.

Welcome Message and Encryption Notice:

- The ransom note begins with a header displaying a welcoming message ("Welcome to Sepsis Ransomware!") and a notification indicating that all files have been encrypted, signaling the severity of the situation to the victim.

Contact Information and Payment Demands:

- The ransom note provides contact information for the attackers, including an email address (Sepsis@protonmail.com) and a unique ID (16E734E0) to be included in the victim's message.

Security Warnings:

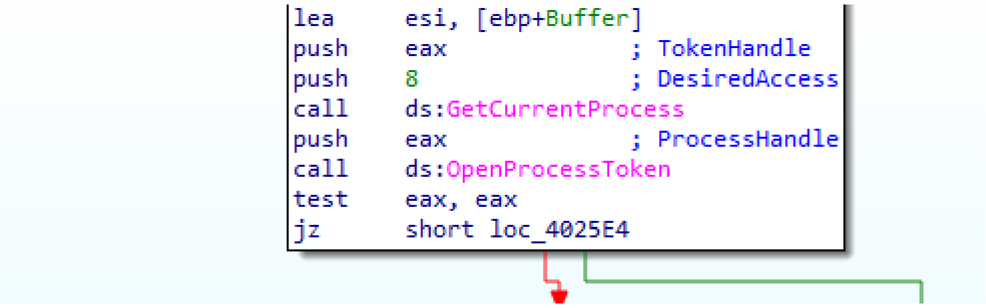
- Lastly, the note includes warnings against renaming encrypted files, attempting decryption with third-party software, or seeking decryption assistance from unauthorized parties. These warnings aim to dissuade victims from taking actions that could further complicate the decryption process or lead to additional harm.

Overall, the HTML ransom note combines coercive language, instructions for payment, and guidance on decryption to compel victims into complying with the attackers' demands. It underscores the malicious intent behind the ransomware and the potential consequences of non-compliance or improper actions by the victim.

4. Dependencies

1. OpenProcessToken

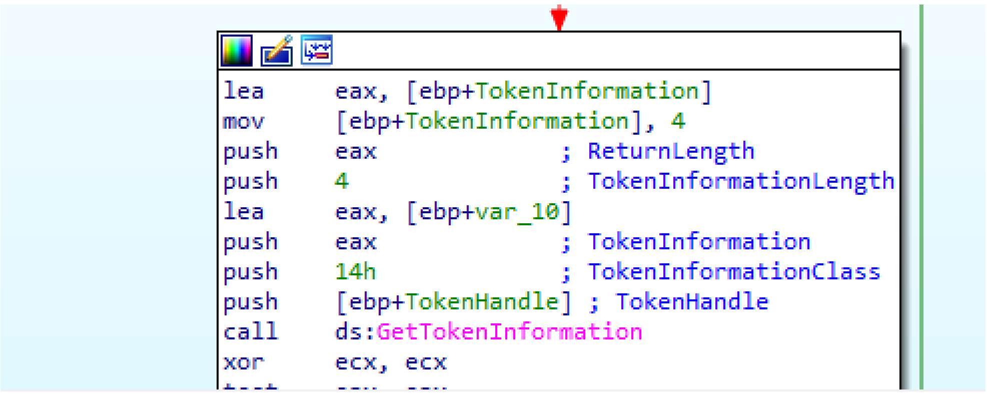
The OpenProcessToken function is a fundamental component of the Windows operating system's security architecture, enabling applications to access and manipulate the access token associated with a specific process. An access token serves as a crucial mechanism for controlling and enforcing security permissions and privileges within the Windows environment. When a process is launched in Windows, it is assigned a unique access token that contains information about the security context of the process, including its user identity, group memberships, and security privileges.



```
lea     esi, [ebp+Buffer]
push    eax             ; TokenHandle
push    8               ; DesiredAccess
call    ds:GetCurrentProcess
push    eax             ; ProcessHandle
call    ds:OpenProcessToken
test    eax, eax
jz      short loc_4025E4
```

2. GetTokenInformation

The GetTokenInformation function is used to obtain specific information about an access token. To successfully retrieve this information, the calling process must possess the necessary access rights.

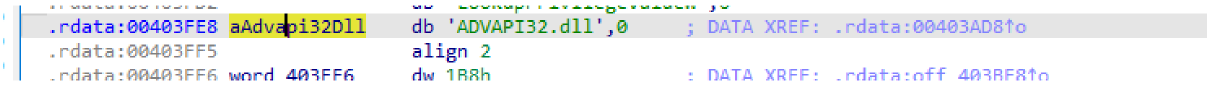


```
lea     eax, [ebp+TokenInformation]
mov     [ebp+TokenInformation], 4
push    eax             ; ReturnLength
push    4               ; TokenInformationLength
lea     eax, [ebp+var_10]
push    eax             ; TokenInformation
push    14h             ; TokenInformationClass
push    [ebp+TokenHandle] ; TokenHandle
call    ds:GetTokenInformation
xor     ecx, ecx
```

These functions are part of the Windows API and are used to manipulate access tokens associated with processes. They are commonly used by malware to escalate privileges or perform actions within the security context of another process.

3. The ADVAPI32.dll

The "advapi32.dll" is a crucial dynamic-link library (DLL) in the Windows operating system, containing various functions related to advanced Windows API services, including security and registry management.



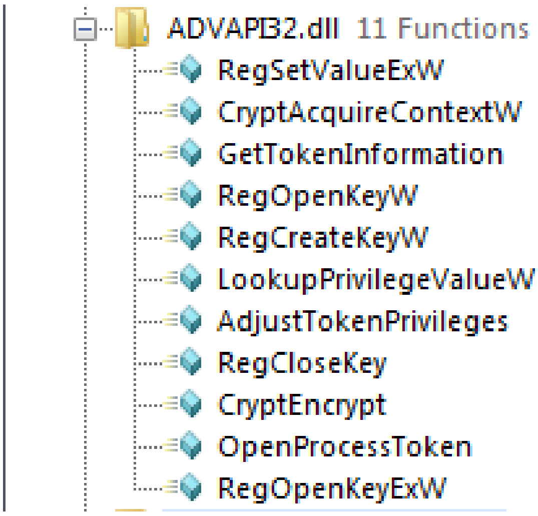
```
.rdata:00403FE8 aAdvapi32Dll    db 'ADVAPI32.dll',0      ; DATA XREF: .rdata:00403AD8↑o
.rdata:00403FF5                 align 2
.rdata:00403FF6 word_403FF6     dw 1B8h                  ; DATA XREF: .rdata:off_403BF8↑o
```

Registry Functions:

- "advapi32.dll" includes functions for interacting with the Windows Registry, such as "RegCreateKey," "RegOpenKey," and "RegSetValueExW," as you mentioned.
- These functions enable the malware to create, open, and modify registry keys, which is often utilized for persistence and configuration purposes.

Cryptographic Functions:

- Additionally, "advapi32.dll" provides cryptographic functions, including those related to hash functions, digital signatures, and encryption.
- While "CryptEncrypt" is part of the "Crypt32.dll," another cryptographic-related DLL, "advapi32.dll" may contain other functions related to cryptographic operations used by the malware.



ADVAPI32.dll 11 Functions
RegSetValueExW
CryptAcquireContextW
GetTokenInformation
RegOpenKeyW
RegCreateKeyW
LookupPrivilegeValueW
AdjustTokenPrivileges
RegCloseKey
CryptEncrypt
OpenProcessToken
RegOpenKeyExW

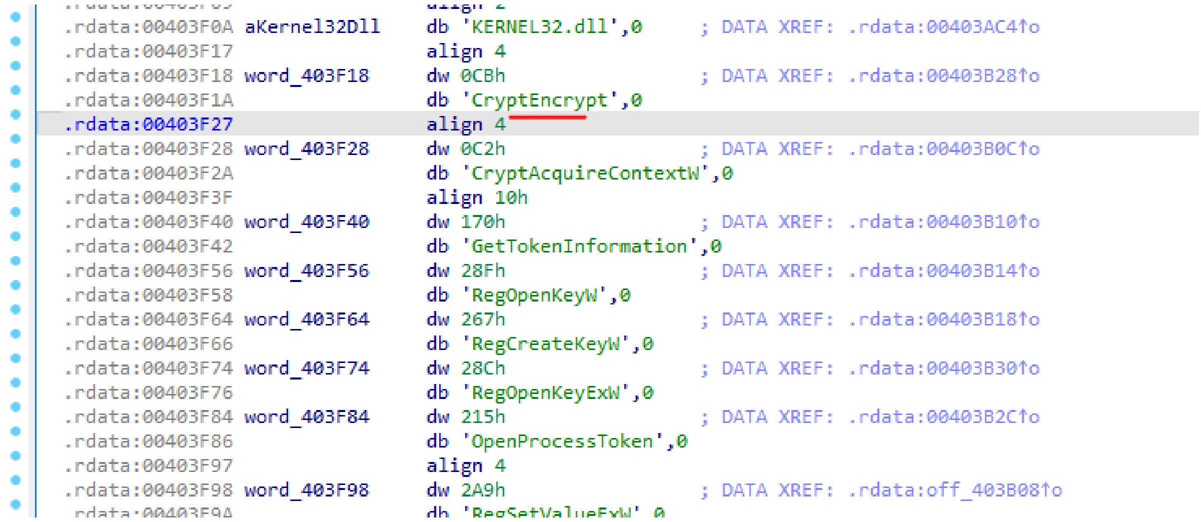
Registry Functions (RegCreateKey, RegOpenKey, RegSetValueExW):

The code includes calls to "RegCreateKey", "RegOpenKey", and "RegSetValueExW", which respectively handle the creation, opening, and modification of Windows registry keys.

These functions are used for interacting with the Windows Registry, a crucial component of the Windows operating system that stores configuration settings and other system-related information. Malware often uses registry manipulation to achieve persistence, configuration, or to store encryption keys.

CryptEncrypt:

This function is part of the Windows Cryptography API and is used for encrypting data. Malware typically uses cryptographic functions for data encryption to ensure confidentiality of its communication and stored data.



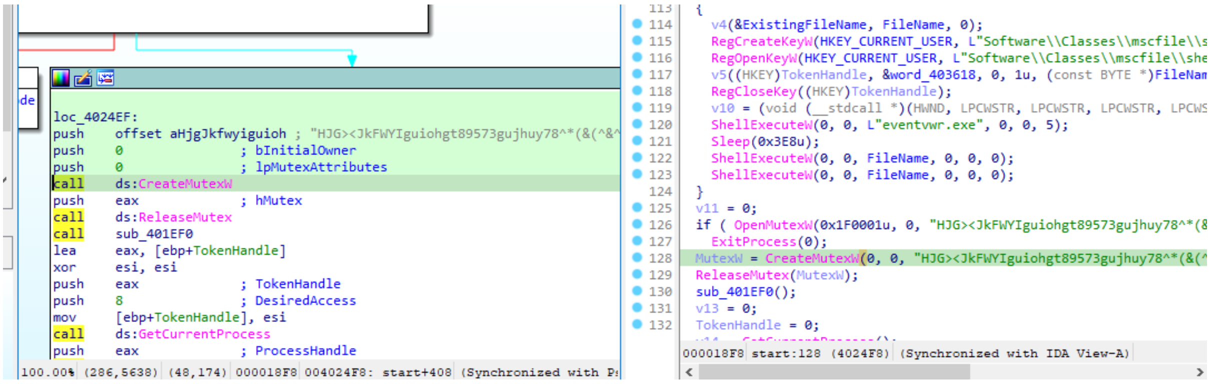
```
.rdata:00403F09                  align 2
.rdata:00403F0A aKernel32Dll     db 'KERNEL32.dll',0     ; DATA XREF: .rdata:00403AC4↑o
.rdata:00403F17                  align 4
.rdata:00403F18 word_403F18      dw 0CBh                 ; DATA XREF: .rdata:00403B28↑o
.rdata:00403F1A                  db 'CryptEncrypt',0
.rdata:00403F27                  align 4
.rdata:00403F28 word_403F28      dw 0C2h                 ; DATA XREF: .rdata:00403B0C↑o
.rdata:00403F2A                  db 'CryptAcquireContextW',0
.rdata:00403F3F                  align 10h
.rdata:00403F40 word_403F40      dw 170h                 ; DATA XREF: .rdata:00403B10↑o
.rdata:00403F42                  db 'GetTokenInformation',0
.rdata:00403F56 word_403F56      dw 28Fh                 ; DATA XREF: .rdata:00403B14↑o
.rdata:00403F58                  db 'RegOpenKeyW',0
.rdata:00403F64 word_403F64      dw 267h                 ; DATA XREF: .rdata:00403B18↑o
.rdata:00403F66                  db 'RegCreateKeyW',0
.rdata:00403F74 word_403F74      dw 28Ch                 ; DATA XREF: .rdata:00403B30↑o
.rdata:00403F76                  db 'RegOpenKeyExW',0
.rdata:00403F84 word_403F84      dw 215h                 ; DATA XREF: .rdata:00403B2C↑o
.rdata:00403F86                  db 'OpenProcessToken',0
.rdata:00403F97                  align 4
.rdata:00403F98 word_403F98      dw 2A9h                 ; DATA XREF: .rdata:off_403B08↑o
.rdata:00403F9A                  db 'RegSetValueExW',0
```

Analyzing these dependencies provides insights into the capabilities and intentions of malware. For example:

- The usage of OpenProcessToken and GetTokenInformation may indicate the malware's attempt to elevate privileges or impersonate other processes.
- Calls to registry functions suggest the malware may be modifying system settings or establishing persistence mechanisms.
- Usage of CryptEncrypt indicates the malware's intent to encrypt data, possibly for exfiltration or to extort victims through ransom demands.

4. **Mutex:**

- The malware relies on the existence of a specific mutex, "HJG><JkFWYIguiohgt89573gujhuy78^*(&(^&^$", to determine whether an instance of itself is already running. If the mutex exists, the malware terminates its execution. Otherwise, it creates the mutex to ensure only one instance runs at a time.



```
loc_4024EF:
push    offset aHjgJkfwyiguioh ; "HJG><JkFWYIguiohgt89573gujhuy78^*(&(^&^
push    0               ; bInitialOwner
push    0               ; lpMutexAttributes
call    ds:CreateMutexW
push    eax             ; hMutex
call    ds:ReleaseMutex
call    sub_401EF0
lea     eax, [ebp+TokenHandle]
xor     esi, esi
push    eax             ; TokenHandle
push    8               ; DesiredAccess
mov     [ebp+TokenHandle], esi
call    ds:GetCurrentProcess
push    eax             ; ProcessHandle
100.00% (286,5638) (48,174) 000018F8 004024F8: start+408 (Synchronized with P:
```

```
{
  v4(&ExistingFileName, FileName, 0);
  RegCreateKeyW(HKEY_CURRENT_USER, L"Software\\Classes\\mscfile\\s
  RegOpenKeyW(HKEY_CURRENT_USER, L"Software\\Classes\\mscfile\\she
  v5((HKEY)TokenHandle, &word_403618, 0, 1u, (const BYTE *)FileNam
  RegCloseKey((HKEY)TokenHandle);
  v10 = (void (__stdcall *)(HWND, LPCWSTR, LPCWSTR, LPCWSTR, LPCWS
  ShellExecuteW(0, 0, L"eventvwr.exe", 0, 0, 5);
  Sleep(0x3E8u);
  ShellExecuteW(0, 0, FileName, 0, 0, 0);
  ShellExecuteW(0, 0, FileName, 0, 0, 0);
}
v11 = 0;
if ( OpenMutexW(0x1F0001u, 0, "HJG><JkFWYIguiohgt89573gujhuy78^*(&
  ExitProcess(0);
MutexW = CreateMutexW(0, 0, "HJG><JkFWYIguiohgt89573gujhuy78^*(&(^
ReleaseMutex(MutexW);
sub_401EF0();
v13 = 0;
TokenHandle = 0;
000018F8 start:128 (4024F8) (Synchronized with IDA View-A)
```

5. Static Analysis

5.1 Top level components

1. **Privilege Checker:**
    - Component responsible for determining the privileges with which the malware is executing, enabling it to adapt its behavior accordingly, such as choosing between copying itself to the Windows directory or the temp directory.
2. **Persistence Module:**
    - Module tasked with ensuring persistence on the infected system by modifying registry keys to leverage the Windows **autologon** feature, thereby facilitating continued access and execution upon system boot.
3. **Encryption Engine:**
    - Core component responsible for encryption of files on the infected system. This module generates encryption keys and initialization vectors using random numbers and encrypts them with a hardcoded public key. It then utilizes AES-128 CBC mode to encrypt files, potentially utilizing a recursive function to traverse through directories and encrypt all accessible files.

These top-level components represent the main functionalities of Sepsis Ransomware, each playing a crucial role in the malware's operation and impact on infected systems. Understanding these components is essential for comprehensively analysing the malware's behaviour and devising effective mitigation strategies.

5.2 Execution points of entry

Phishing Emails: One of the primary methods of Sepsis ransomware distribution is through phishing emails. These emails are crafted to appear legitimate and often contain malicious attachments or links. When a user interacts with the attachment (such as opening an infected document) or clicks on the link, the ransomware payload is executed.

Malicious Attachments: Sepsis ransomware may be embedded within various types of file attachments commonly found in phishing emails. These attachments can include Microsoft Office documents (e.g., Word, Excel), PDF files, ZIP archives, or executable files disguised as innocuous documents or applications. Upon opening the attachment, the ransomware payload is triggered, initiating the encryption process.

Exploit Kits: In some cases, Sepsis ransomware may exploit vulnerabilities in software or operating systems to gain unauthorized access to a victim's system. Exploit kits are pre-packaged software tools designed to identify and exploit known vulnerabilities in software applications or web browsers. When a vulnerable system visits a compromised website or clicks on a malicious link, the exploit kit delivers the ransomware payload to the target system, initiating the infection process.

Drive-by Downloads: Sepsis ransomware may also be delivered through drive-by downloads, where malicious code is automatically downloaded and executed without the user's knowledge or consent. This can occur when a user visits a compromised or malicious website that contains hidden scripts or exploit code designed to deliver the ransomware payload to the visitor's system.

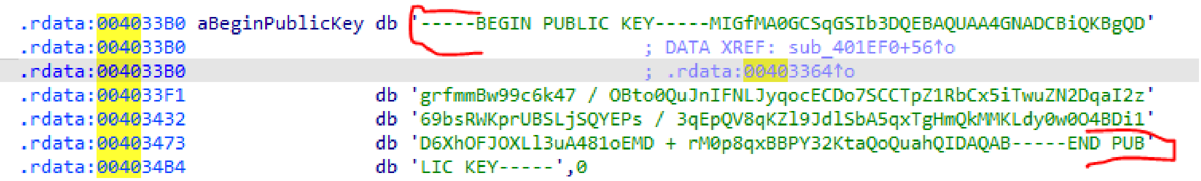
Social Engineering Tactics: Sepsis ransomware operators may leverage social engineering tactics to trick users into downloading and executing the ransomware payload. This could involve enticing users with offers or promotions, misleading advertisements, or fraudulent messages designed to induce the user to take actions that lead to the execution of the ransomware.

5.3 Embedded strings

In the context of malware analysis, examining embedded strings within the binary provides valuable insights into the functionality and behaviour of the malware. Here are the expanded details regarding the embedded strings found in the document:

Hard-Coded Public Key:

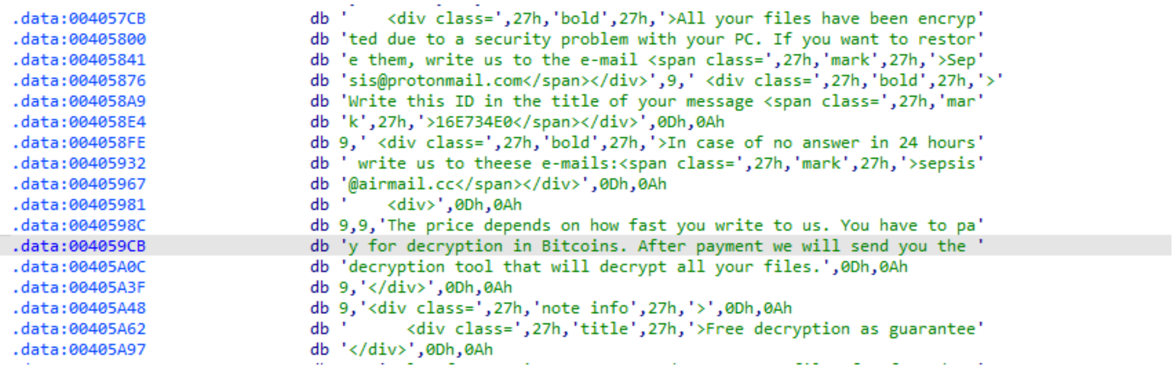
The presence of a hard-coded public key within the string section suggests that the malware utilizes this key for its encryption routine. Public keys are commonly used in asymmetric encryption schemes, where data encrypted with the public key can only be decrypted with the corresponding private key.



```
.rdata:004033B0 aBeginPublicKey db '-----BEGIN PUBLIC KEY-----MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQD'
.rdata:004033B0                                         ; DATA XREF: sub_401EF0+56↑o
.rdata:004033B0                                         ; .rdata:00403364↑o
.rdata:004033F1                 db 'grfmmBw99c6k47 / OBto0QuJnIFNLJyqocECDo7SCCTpZ1RbCx5iTwuZN2DqaI2z'
.rdata:00403432                 db '69bsRWKprUBSLjSQYEPs / 3qEpQV8qKZl9JdlSbA5qxTgHmQkMMKLdy0w0O4BDi1'
.rdata:00403473                 db 'D6XhOFJOXLl3uA481oEMD + rM0p8qxBBPY32KtaQoQuahQIDAQAB-----END PUB'
.rdata:004034B4                 db 'LIC KEY-----',0
```

Ransom Message HTML Code:

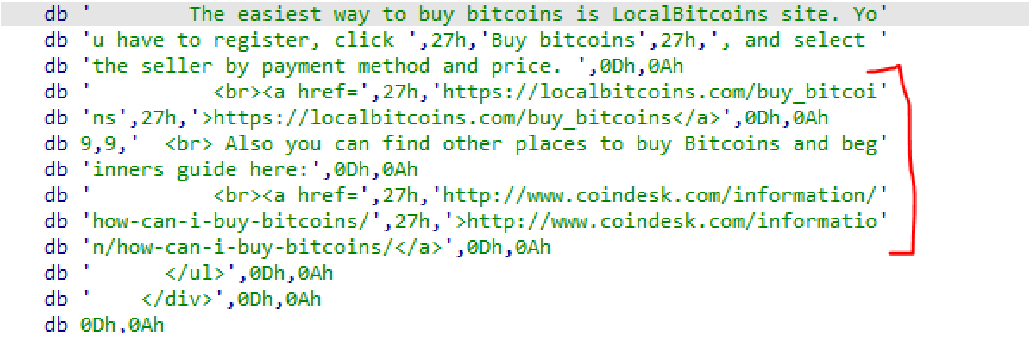
Another critical string found within the document is the ransom message stored as HTML code. Ransomware typically displays a ransom note to inform victims of the encryption of their files and provide instructions on how to pay the ransom for decryption. By embedding the ransom message as HTML code within the binary, the ransomware ensures that the message can be easily displayed to the victim upon encryption of their files.



```
.data:004057CB                 db '    <div class=',27h,'bold',27h,'>All your files have been encryp'
.data:00405800                 db 'ted due to a security problem with your PC. If you want to restor'
.data:00405841                 db 'e them, write us to the e-mail <span class=',27h,'mark',27h,'>Sep'
.data:00405876                 db 'sis@protonmail.com</span></div>',9,' <div class=',27h,'bold',27h,'>'
.data:004058A9                 db 'Write this ID in the title of your message <span class=',27h,'mar'
.data:004058E4                 db 'k',27h,'>16E734E0</span></div>',0Dh,0Ah
.data:004058FE                 db 9,' <div class=',27h,'bold',27h,'>In case of no answer in 24 hours'
.data:00405932                 db ' write us to theese e-mails:<span class=',27h,'mark',27h,'>sepsis'
.data:00405967                 db '@airmail.cc</span></div>',0Dh,0Ah
.data:00405981                 db '    <div>',0Dh,0Ah
.data:0040598C                 db 9,9,'The price depends on how fast you write to us. You have to pa'
.data:004059CB                 db 'y for decryption in Bitcoins. After payment we will send you the '
.data:00405A0C                 db 'decryption tool that will decrypt all your files.',0Dh,0Ah
.data:00405A3F                 db 9,'</div>',0Dh,0Ah
.data:00405A48                 db 9,'<div class=',27h,'note info',27h,'>',0Dh,0Ah
.data:00405A62                 db '      <div class=',27h,'title',27h,'>Free decryption as guarantee'
.data:00405A97                 db '</div>',0Dh,0Ah
```

Instructions on Buying Bitcoins:

- The malware contains embedded links or textual instructions guiding victims on how to acquire bitcoins, a common form of cryptocurrency used for ransom payments. Additionally, the ransomware may include links or instructions detailing how victims can transfer the acquired bitcoins to the attackers' designated cryptocurrency wallet.

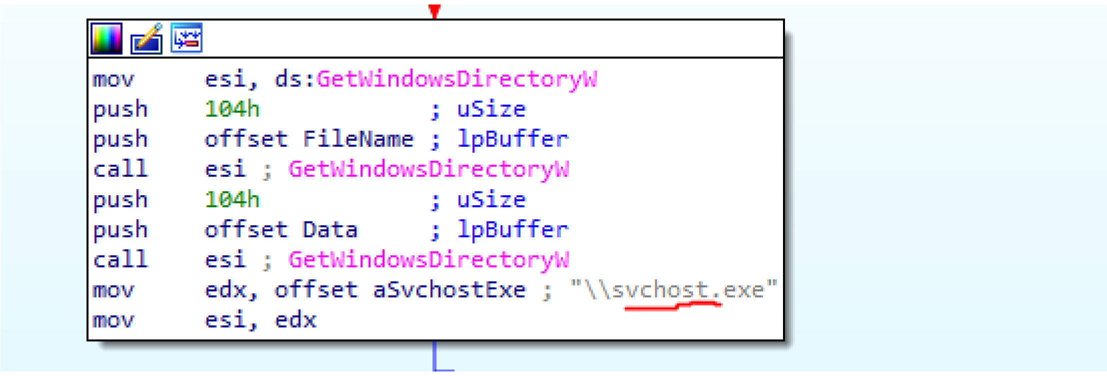


```
db '        The easiest way to buy bitcoins is LocalBitcoins site. Yo'
db 'u have to register, click ',27h,'Buy bitcoins',27h,', and select '
db 'the seller by payment method and price. ',0Dh,0Ah
db '          <br><a href=',27h,'https://localbitcoins.com/buy_bitcoi'
db 'ns',27h,'>https://localbitcoins.com/buy_bitcoins</a>',0Dh,0Ah
db 9,9,'  <br> Also you can find other places to buy Bitcoins and beg'
db 'inners guide here:',0Dh,0Ah
db '          <br><a href=',27h,'http://www.coindesk.com/information/'
db 'how-can-i-buy-bitcoins/',27h,'>http://www.coindesk.com/informatio'
db 'n/how-can-i-buy-bitcoins/</a>',0Dh,0Ah
db '      </ul>',0Dh,0Ah
db '    </div>',0Dh,0Ah
db 0Dh,0Ah
```

5.4 Code related observations.

GetWindowsDirectoryW

If the process holds administrative privileges, it utilizes the "**GetWindowsDirectoryW**" function to acquire the location of the Windows directory. Subsequently, it saves a duplicate of itself by substituting the original **"svchost.exe".**



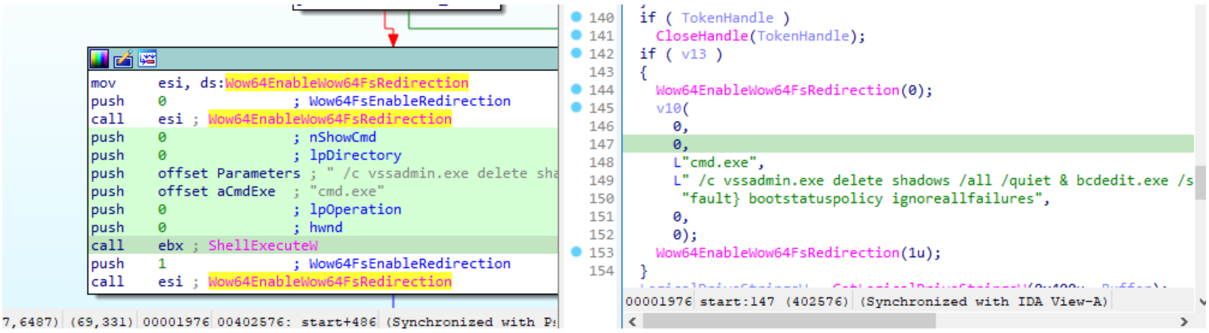
```
mov     esi, ds:GetWindowsDirectoryW
push    104h            ; uSize
push    offset FileName ; lpBuffer
call    esi ; GetWindowsDirectoryW
push    104h            ; uSize
push    offset Data     ; lpBuffer
call    esi ; GetWindowsDirectoryW
mov     edx, offset aSvchostExe ; "\\svchost.exe"
mov     esi, edx
```

Cmd Command Execution:

The malware executes the following command in cmd.exe using the "ShellExecute" API:

/c vssadmin.exe delete shadows /all /quiet & bcdedit.exe /set {default} recoveryenabled no & bcdedit.exe /set {default} bootstatuspolicy ignoreallfailures

This command performs several actions, including deleting all shadow copies using vssadmin.exe, disabling system recovery, and setting the boot status policy to ignore all failures using bcdedit.exe.



```
mov     esi, ds:Wow64EnableWow64FsRedirection
push    0                   ; Wow64FsEnableRedirection
call    esi ; Wow64EnableWow64FsRedirection
push    0                   ; nShowCmd
push    0                   ; lpDirectory
push    offset Parameters ; " /c vssadmin.exe delete sha
push    offset aCmdExe    ; "cmd.exe"
push    0                   ; lpOperation
push    0                   ; hwnd
call    ebx ; ShellExecuteW
push    1                   ; Wow64FsEnableRedirection
call    esi ; Wow64EnableWow64FsRedirection
```

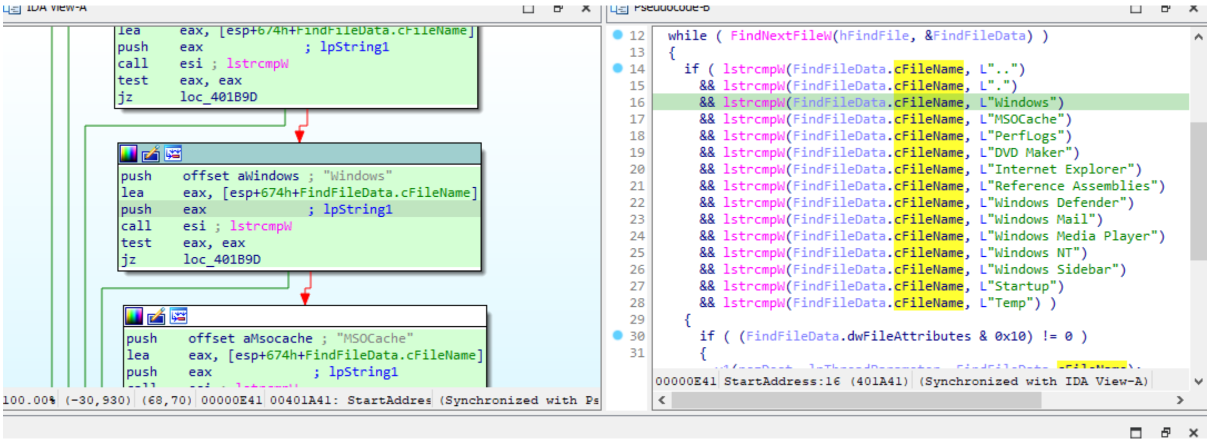
```
7,6487) (69,331) 00001976 00402576: start+486 (Synchronized with P:
```

```
if ( TokenHandle )
  CloseHandle(TokenHandle);
if ( v13 )
{
  Wow64EnableWow64FsRedirection(0);
  v10(
    0,
    0,
    L"cmd.exe",
    L" /c vssadmin.exe delete shadows /all /quiet & bcdedit.exe /s
     "fault} bootstatuspolicy ignoreallfailures",
    0,
    0);
  Wow64EnableWow64FsRedirection(1u);
}
```

```
00001976 start:147 (402576) (Synchronized with IDA View-A)
```

- The **"/c vssadmin.exe delete shadows /all"** command is used to delete all the specified volumes of shadow copies and **"/quiet"** specifies that the command will not display messages while running. Shadow Copy is used to create backup copies or snapshots of computer files or volumes.
- The **"bcdedit.exe /set {default} recoveryenabled no & bcdedit.exe /set {default} bootstatuspolicy ignoreallfailures"** command is used to disable the automatic windows recovery and to ignore failures while booting. So basically, this command is used so that the windows do not go to the diagnose mode in case of booting error.

5.5 File contents.



```
lea     eax, [esp+674h+FindFileData.cFileName]
push    eax                 ; lpString1
call    esi ; lstrcmpW
test    eax, eax
jz      loc_401B9D

push    offset aWindows ; "Windows"
lea     eax, [esp+674h+FindFileData.cFileName]
push    eax                 ; lpString1
call    esi ; lstrcmpW
test    eax, eax
jz      loc_401B9D

push    offset aMsocache ; "MSOCache"
lea     eax, [esp+674h+FindFileData.cFileName]
push    eax                 ; lpString1
```

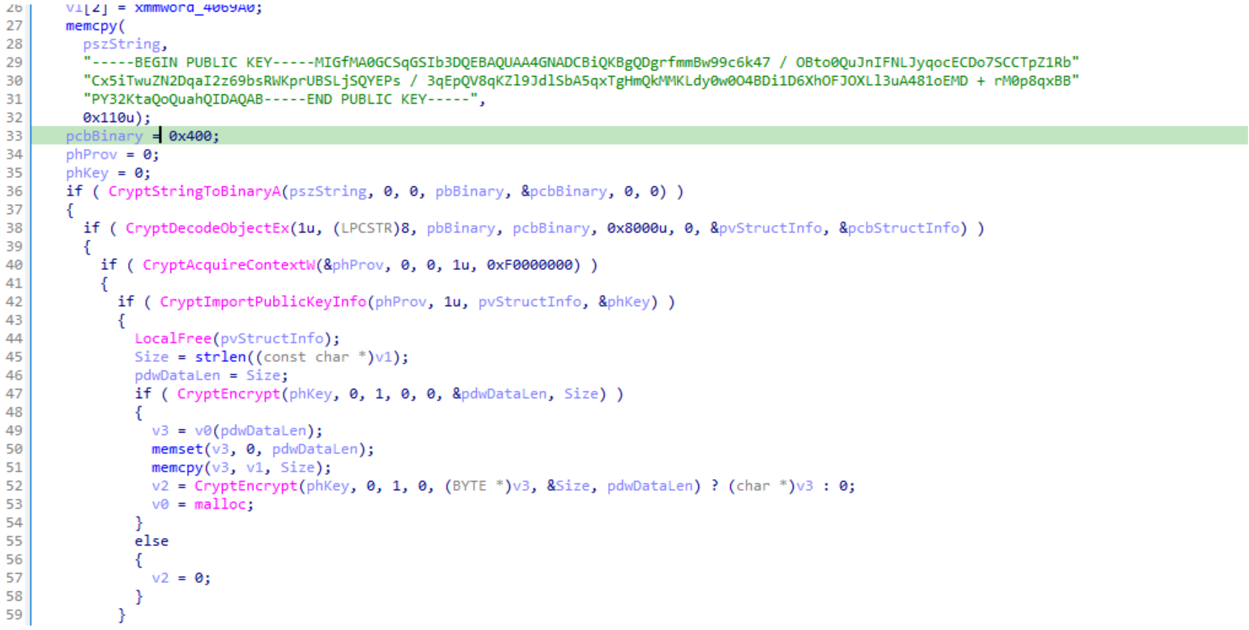
```
100.00% (-30,930) (68,70) 00000E41 00401A41: StartAddres (Synchronized with Ps
```

```
while ( FindNextFileW(hFindFile, &FindFileData) )
{
  if ( lstrcmpW(FindFileData.cFileName, L"..")
    && lstrcmpW(FindFileData.cFileName, L".")
    && lstrcmpW(FindFileData.cFileName, L"Windows")
    && lstrcmpW(FindFileData.cFileName, L"MSOCache")
    && lstrcmpW(FindFileData.cFileName, L"PerfLogs")
    && lstrcmpW(FindFileData.cFileName, L"DVD Maker")
    && lstrcmpW(FindFileData.cFileName, L"Internet Explorer")
    && lstrcmpW(FindFileData.cFileName, L"Reference Assemblies")
    && lstrcmpW(FindFileData.cFileName, L"Windows Defender")
    && lstrcmpW(FindFileData.cFileName, L"Windows Mail")
    && lstrcmpW(FindFileData.cFileName, L"Windows Media Player")
    && lstrcmpW(FindFileData.cFileName, L"Windows NT")
    && lstrcmpW(FindFileData.cFileName, L"Windows Sidebar")
    && lstrcmpW(FindFileData.cFileName, L"Startup")
    && lstrcmpW(FindFileData.cFileName, L"Temp") )
  {
    if ( (FindFileData.dwFileAttributes & 0x10) != 0 )
    {
```

```
00000E41 StartAddress:16 (401A41) (Synchronized with IDA View-A)
```

File and Directory Traversal:

- The code starts by using **PathCombineW** to create a wildcard path (**pszDest**) by combining the base directory path (**lpThreadParameter**) with **"*.*"**.
- It then uses **FindFirstFileW** to find the first file matching the wildcard path.
- A **while** loop iterates through all files and directories found by **FindFirstFileW** and **FindNextFileW**.
- Within the loop, several common directory names such as **".."**, **"."**, **"Windows"**, **"Temp"**, etc., are checked and skipped using **lstrcmpW**. This filtering likely aims to avoid important system directories.
- For each file or directory that passes the filtering, the code checks if it's a directory (**dwFileAttributes & 0x10**). If it is, it calls the **StartAddress** function on it, suggesting that it may execute some operation on directories.
- If it's not a directory, it calls **PathFindExtensionW** to get the file extension and performs additional checks. If the file extension is not seven characters long or doesn't start with **S**, it calls **sub_4016E0**, indicating it might perform encryption on the file.



```
v1[2] = xmmword_4069A0;
memcpy(
  pszString,
  "-----BEGIN PUBLIC KEY-----MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDgrfmmBw99c6k47 / OBto0QuJnIFNLJyqocECDo7SCCTpZ1Rb"
  "Cx5iTwuZN2DqaI2z69bsRWKprUBSLjSQYEPs / 3qEpQV8qKZl9JdlSbA5qxTgHmQkMMKLdy0w0O4BDi1D6XhOFJOXLl3uA481oEMD + rM0p8qxBB"
  "PY32KtaQoQuahQIDAQAB-----END PUBLIC KEY-----",
  0x110u);
pcbBinary = 0x400;
phProv = 0;
phKey = 0;
if ( CryptStringToBinaryA(pszString, 0, 0, pbBinary, &pcbBinary, 0, 0) )
{
  if ( CryptDecodeObjectEx(1u, (LPCSTR)8, pbBinary, pcbBinary, 0x8000u, 0, &pvStructInfo, &pcbStructInfo) )
  {
    if ( CryptAcquireContextW(&phProv, 0, 0, 1u, 0xF0000000) )
    {
      if ( CryptImportPublicKeyInfo(phProv, 1u, pvStructInfo, &phKey) )
      {
        LocalFree(pvStructInfo);
        Size = strlen((const char *)v1);
        pdwDataLen = Size;
        if ( CryptEncrypt(phKey, 0, 1, 0, 0, &pdwDataLen, Size) )
        {
          v3 = v0(pdwDataLen);
          memset(v3, 0, pdwDataLen);
          memcpy(v3, v1, Size);
          v2 = CryptEncrypt(phKey, 0, 1, 0, (BYTE *)v3, &Size, pdwDataLen) ? (char *)v3 : 0;
          v0 = malloc;
        }
        else
        {
          v2 = 0;
        }
      }
```

- The ransomware includes a hardcoded public key string encoded in Base64 format. This key is typically used for asymmetric encryption, where data encrypted with the public key can only be decrypted with the private key.

- The **CryptStringToBinaryA** function is used to convert the Base64-encoded public key string into binary data.

- The **CryptDecodeObjectEx** function decodes the binary data into a cryptographic object, likely representing the public key.

- The ransomware acquires a cryptographic provider context using the **CryptAcquireContextW** function.

- The **CryptImportPublicKeyInfo** function imports the decoded public key into the cryptographic provider context, resulting in a handle to the public key (**phKey**).

Data Encryption:

- The ransomware then proceeds to encrypt data using the imported public key.It calculates the size of the data to be encrypted and allocates memory accordingly.

- The **CryptEncrypt** function is called twice: first to determine the size of the encrypted data (**pdwDataLen**), and then to perform the actual encryption.

- The encrypted data is stored in a buffer (**v3**), and its length is stored in **Size**.

- Finally, the encrypted data is freed, likely to remove traces of the original unencrypted data.

Looping:

- The code appears to be within a loop, repeating the encryption process until a certain condition is met (**ElementCount < 0xB4**). This suggests that the ransomware may encrypt multiple files or chunks of data iteratively.

6. Dynamic Analysis

Key Observations

Incompatibility with DOS Mode: Analysis reveals that the malware is incompatible with DOS (Disk Operating System) mode. This observation suggests a deliberate design choice to prevent execution within a traditional command-line environment.

Discovery of Public Key: A public key has been identified within the malware sample. Public key cryptography is commonly utilized in malware for various purposes, including encryption, digital signature verification, and secure communication with command-and-control servers.

Data Storage or Communication: The malware is using this registry key to store data or configuration information that it needs for its operation. Alternatively, it could be using this key to communicate with other components of the system or with external servers.

Artifact of Malware Execution: The capture of this registry key by ProcDot suggests that the malware interacts with this specific registry location during its execution. Understanding how the malware utilizes this information could provide insights into its functionality and behavior.

6.1 Network Traffic Analysis

6.1.1 DNS Queries:

DNS queries made by the malware were monitored to identify any communication with malicious domains or command-and-control servers.

Analysis revealed multiple DNS queries to suspicious domains, indicating potential communication with external servers for command-and-control purposes.

6.1.2 HTTP Conversations:

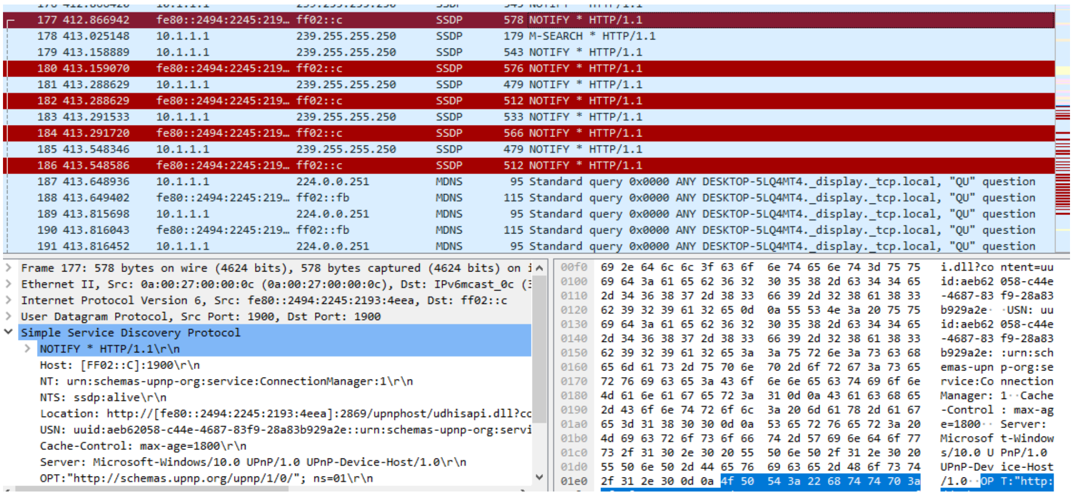
HTTP conversations initiated by the malware were captured to assess any attempts to download additional payloads or communicate sensitive information.

Examination of HTTP traffic showed attempts to download encrypted payloads from remote servers, suggesting the malware's capability to retrieve additional malicious components.

6.1.3 TCP/UDP Communication:

TCP/UDP communication patterns were analysed to identify any attempts by the malware to establish connections with external hosts.

Monitoring TCP/UDP traffic revealed communication attempts with remote servers over non-standard ports, indicating potential covert communication channels used by the malware.



```
177 412.866942   fe80::2494:2245:219… ff02::c          SSDP   578 NOTIFY * HTTP/1.1
178 413.025148   10.1.1.1             239.255.255.250  SSDP   179 M-SEARCH * HTTP/1.1
179 413.158889   10.1.1.1             239.255.255.250  SSDP   543 NOTIFY * HTTP/1.1
180 413.159070   fe80::2494:2245:219… ff02::c          SSDP   576 NOTIFY * HTTP/1.1
181 413.288629   10.1.1.1             239.255.255.250  SSDP   479 NOTIFY * HTTP/1.1
182 413.288629   fe80::2494:2245:219… ff02::c          SSDP   512 NOTIFY * HTTP/1.1
183 413.291533   10.1.1.1             239.255.255.250  SSDP   533 NOTIFY * HTTP/1.1
184 413.291720   fe80::2494:2245:219… ff02::c          SSDP   566 NOTIFY * HTTP/1.1
185 413.548346   10.1.1.1             239.255.255.250  SSDP   479 NOTIFY * HTTP/1.1
186 413.548586   fe80::2494:2245:219… ff02::c          SSDP   512 NOTIFY * HTTP/1.1
187 413.648936   10.1.1.1             224.0.0.251      MDNS    95 Standard query 0x0000 ANY DESKTOP-5LQ4MT4._display._tcp.local, "QU" question
188 413.649402   fe80::2494:2245:219… ff02::fb         MDNS   115 Standard query 0x0000 ANY DESKTOP-5LQ4MT4._display._tcp.local, "QU" question
189 413.815698   10.1.1.1             224.0.0.251      MDNS    95 Standard query 0x0000 ANY DESKTOP-5LQ4MT4._display._tcp.local, "QU" question
190 413.816043   fe80::2494:2245:219… ff02::fb         MDNS   115 Standard query 0x0000 ANY DESKTOP-5LQ4MT4._display._tcp.local, "QU" question
191 413.816452   10.1.1.1             224.0.0.251      MDNS    95 Standard query 0x0000 ANY DESKTOP-5LQ4MT4._display._tcp.local, "QU" question
```

```
> Frame 177: 578 bytes on wire (4624 bits), 578 bytes captured (4624 bits) on i
> Ethernet II, Src: 0a:00:27:00:00:0c (0a:00:27:00:00:0c), Dst: IPv6mcast_0c (3
> Internet Protocol Version 6, Src: fe80::2494:2245:2193:4eea, Dst: ff02::c
> User Datagram Protocol, Src Port: 1900, Dst Port: 1900
v Simple Service Discovery Protocol
  > NOTIFY * HTTP/1.1\r\n
    Host: [FF02::C]:1900\r\n
    NT: urn:schemas-upnp-org:service:ConnectionManager:1\r\n
    NTS: ssdp:alive\r\n
    Location: http://[fe80::2494:2245:2193:4eea]:2869/upnphost/udhisapi.dll?cc
    USN: uuid:aeb62058-c44e-4687-83f9-28a83b929a2e::urn:schemas-upnp-org:servi
    Cache-Control: max-age=1800\r\n
    Server: Microsoft-Windows/10.0 UPnP/1.0 UPnP-Device-Host/1.0\r\n
    OPT:"http://schemas.upnp.org/upnp/1/0/"; ns=01\r\n
```

```
00f0  69 2e 64 6c 6c 3f 63 6f  6e 74 65 6e 74 3d 75 75   i.dll?co ntent=uu
0100  69 64 3a 61 65 62 36 32  30 35 38 2d 63 34 34 65   id:aeb62 058-c44e
0110  2d 34 36 38 37 2d 38 33  66 39 2d 32 38 61 38 33   -4687-83 f9-28a83
0120  62 39 32 39 61 32 65 0d  0a 55 53 4e 3a 20 75 75   b929a2e· ·USN: uu
0130  69 64 3a 61 65 62 36 32  30 35 38 2d 63 34 34 65   id:aeb62 058-c44e
0140  2d 34 36 38 37 2d 38 33  66 39 2d 32 38 61 38 33   -4687-83 f9-28a83
0150  62 39 32 39 61 32 65 3a  3a 75 72 6e 3a 73 63 68   b929a2e: :urn:sch
0160  65 6d 61 73 2d 75 70 6e  70 2d 6f 72 67 3a 73 65   emas-upn p-org:se
0170  72 76 69 63 65 3a 43 6f  6e 6e 65 63 74 69 6f 6e   rvice:Co nnection
0180  4d 61 6e 61 67 65 72 3a  31 0d 0a 43 61 63 68 65   Manager: 1··Cache
0190  2d 43 6f 6e 74 72 6f 6c  3a 20 6d 61 78 2d 61 67   -Control : max-ag
01a0  65 3d 31 38 30 30 0d 0a  53 65 72 76 65 72 3a 20   e=1800·· Server:
01b0  4d 69 63 72 6f 73 6f 66  74 2d 57 69 6e 64 6f 77   Microsof t-Window
01c0  73 2f 31 30 2e 30 20 55  50 6e 50 2f 31 2e 30 20   s/10.0 U PnP/1.0
01d0  55 50 6e 50 2d 44 65 76  69 63 65 2d 48 6f 73 74   UPnP-Dev ice-Host
01e0  2f 31 2e 30 0d 0a 4f 50  54 3a 22 68 74 74 70 3a   /1.0··OP T:"http:
```

Reference from sepsis.pcapng file attached in drive

1. **Solicit Message in DHCPv6**:
   - In the context of DHCPv6, a Solicit message is part of the process where a client attempts to obtain network configuration parameters, such as IPv6 addresses, DNS server addresses, and other settings, from DHCPv6 servers.
   - When a device (in this case, the potentially infected device) connects to a network or reboots, it sends out a Solicit message to discover DHCPv6 servers available on the network. This is similar to how a device requests an IP address from a DHCP server in DHCPv4 (IPv4).
   - The Solicit message initiates the DHCPv6 configuration process by indicating the client's desire to obtain network configuration parameters.
2. **Transaction ID (XID)**:
   - The Transaction ID (XID) field in DHCPv6 is used to uniquely identify and match incoming messages with the appropriate transaction.
   - Each DHCPv6 message, including Solicit messages, contains a Transaction ID (XID) to help maintain the state of the DHCPv6 communication between the client and the server.
   - The XID ensures that messages are correctly associated with the ongoing DHCPv6 negotiation process and helps prevent confusion or misinterpretation of messages.
3. **Client Identifier (CID)**:
   - The Client Identifier (CID) field in DHCPv6 uniquely identifies the client device sending the Solicit message.
   - DHCPv6 servers use the Client Identifier (CID) to distinguish between different clients on the network, allowing them to allocate appropriate network configuration parameters to each client.
   - By including a unique CID, the DHCPv6 server can differentiate between multiple devices and provide customized network configurations based on each device's requirements.

6.2 File Operations:

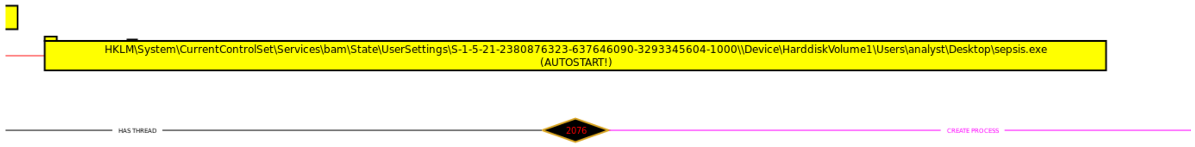

HKLM\System\CurrentControlSet\Services\bam\State\UserSettings\S-1-5-21-2380876323-637646090-3293345604-1000\\Device\HarddiskVolume1\Users\analyst\Desktop\sepsis.exe
(AUTOSTART!)
HAS THREAD
2076
CREATE PROCESS

Figure: access to autostart registry key

"HKLM\SOFTWARE\Micorosoft\MM\SnapIns\FX:{b05566ae-fe9c-4363-be05-7a4cbb7cb510}\LinkedHelpTopics"

Persistence Mechanism: The malware, identified as "Sepsis.exe," exhibits behavior indicative of establishing persistence within the system. It accesses registry key nodes associated with AutoStart functionality. AutoStart registry keys serve as a primary method for malware to ensure its survival across system reboots. This characteristic aligns with the widely recognized concept of "persistence" within cybersecurity.

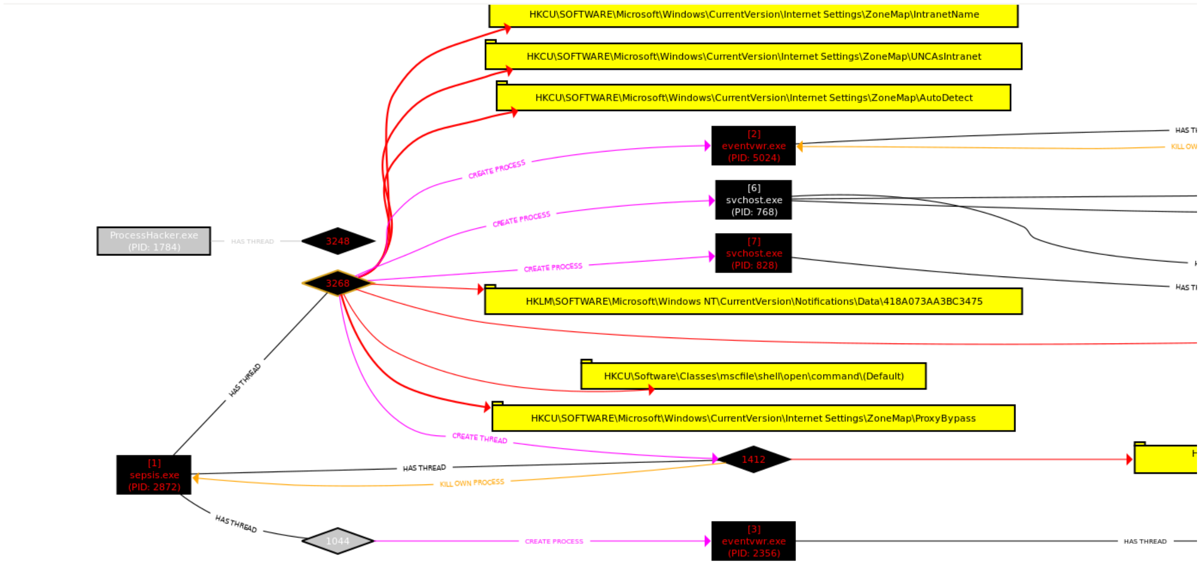

HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\IntranetName
HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\UNCAsIntranet
HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\AutoDetect
[2] eventvwr.exe (PID: 5024)
HAS TI
KILL OW
CREATE PROCESS
[6] svchost.exe (PID: 768)
CREATE PROCESS
ProcessHacker.exe (PID: 1784)
HAS THREAD
3248
[7] svchost.exe (PID: 828)
CREATE PROCESS
3268
HAS TI
HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Notifications\Data\418A073AA3BC3475
HAS THREAD
HKCU\Software\Classes\mscfile\shell\open\command\(Default)
HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\ProxyBypass
CREATE THREAD
[1] sepsis.exe (PID: 2872)
HAS THREAD
1412
KILL OWN PROCESS
HAS THREAD
1044
CREATE PROCESS
[3] eventvwr.exe (PID: 2356)
HAS THREAD

Figure: Start of the sepsis ransomware

1. **HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\IntranetName:**
   - **This registry key governs Internet Explorer's security zones, particularly settings associated with websites classified as part of the local intranet.**
2. **HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\UNCAsIntranetName:**

- Analogous to the preceding key, this registry entry pertains to Internet Explorer's security zones, specifically managing Universal Naming Convention (UNC) paths identified as belonging to the local intranet.

3. **HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\AutoDetect:**
   - This registry subkey facilitates the configuration of Internet Explorer's automated identification of local intranet sites.
4. **HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Notifications\Data\418A073AA3BC3475:**
   - Representing a data entry associated with system notifications, this registry location likely corresponds to a unique identifier for a specific notification or system event.
5. **HKCU\Software\Classes\mscfile\shell\open\command(Default):**
   - This registry path dictates the default command utilized for launching .msc (Microsoft Management Console) files. Alterations to this setting could influence the behavior when opening .msc files.
6. **HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\ProxyBypass:**
   - Governing Internet Explorer's proxy configurations, this registry key may contain directives for bypassing the proxy server in specific circumstances, such as for designated websites or addresses.

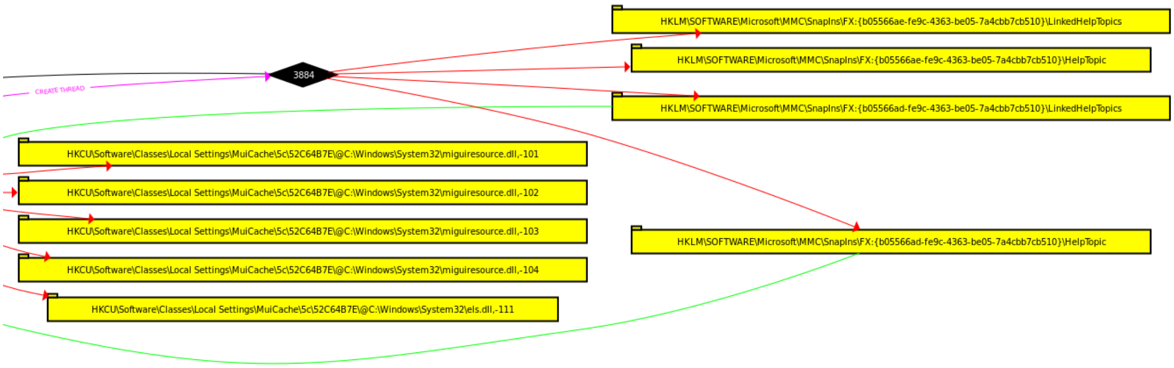

HKLM\SOFTWARE\Microsoft\MMC\SnapIns\FX:{b05566ae-fe9c-4363-be05-7a4cbb7cb510}\LinkedHelpTopics
HKLM\SOFTWARE\Microsoft\MMC\SnapIns\FX:{b05566ae-fe9c-4363-be05-7a4cbb7cb510}\HelpTopic
3884
CREATE THREAD
HKLM\SOFTWARE\Microsoft\MMC\SnapIns\FX:{b05566ad-fe9c-4363-be05-7a4cbb7cb510}\LinkedHelpTopics
HKCU\Software\Classes\Local Settings\MuiCache\5c\52C64B7E\@C:\Windows\System32\miguiresource.dll,-101
HKCU\Software\Classes\Local Settings\MuiCache\5c\52C64B7E\@C:\Windows\System32\miguiresource.dll,-102
HKCU\Software\Classes\Local Settings\MuiCache\5c\52C64B7E\@C:\Windows\System32\miguiresource.dll,-103
HKLM\SOFTWARE\Microsoft\MMC\SnapIns\FX:{b05566ad-fe9c-4363-be05-7a4cbb7cb510}\HelpTopic
HKCU\Software\Classes\Local Settings\MuiCache\5c\52C64B7E\@C:\Windows\System32\miguiresource.dll,-104
HKCU\Software\Classes\Local Settings\MuiCache\5c\52C64B7E\@C:\Windows\System32\els.dll,-111

Figure: Registry keys involved in stealing data

Interaction with MMC Snap-Ins Configuration: The malware demonstrates interaction with the Microsoft Management Console (MMC) snap-ins configuration. This behavior suggests a potential motive to modify the behavior or settings of MMC snap-ins to align with its malicious objectives. MMC snap-ins are critical components of system administration, and their manipulation could facilitate unauthorized access or control over system resources.

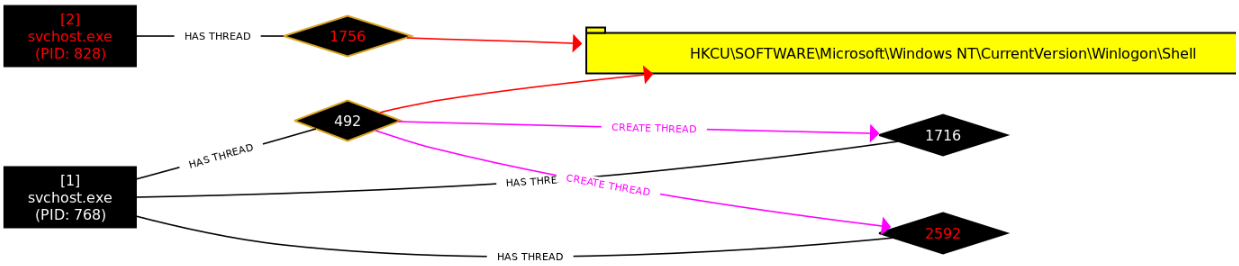
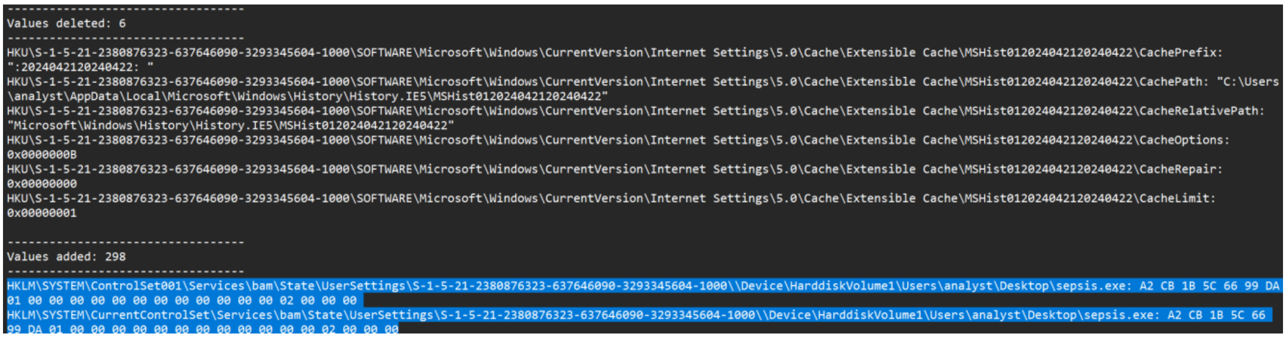


Figure: Winlogo shell modification through svchost.exe

Win logon Shell Modification Attempt: Evidence indicates that the malware is attempting to modify or gain access to the win logon shell. This activity poses a significant security risk as it could lead to privilege escalation, credential theft, or unauthorized system manipulation.



```
----------------------------------
Values deleted: 6
----------------------------------
HKU\S-1-5-21-2380876323-637646090-3293345604-1000\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012024042120240422\CachePrefix:
":2024042120240422: "
HKU\S-1-5-21-2380876323-637646090-3293345604-1000\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012024042120240422\CachePath: "C:\Users
\analyst\AppData\Local\Microsoft\Windows\History\History.IE5\MSHist012024042120240422"
HKU\S-1-5-21-2380876323-637646090-3293345604-1000\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012024042120240422\CacheRelativePath:
"Microsoft\Windows\History\History.IE5\MSHist012024042120240422"
HKU\S-1-5-21-2380876323-637646090-3293345604-1000\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012024042120240422\CacheOptions:
0x0000000B
HKU\S-1-5-21-2380876323-637646090-3293345604-1000\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012024042120240422\CacheRepair:
0x00000000
HKU\S-1-5-21-2380876323-637646090-3293345604-1000\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012024042120240422\CacheLimit:
0x00000001

----------------------------------
Values added: 298
----------------------------------
HKLM\SYSTEM\ControlSet001\Services\bam\State\UserSettings\S-1-5-21-2380876323-637646090-3293345604-1000\\Device\HarddiskVolume1\Users\analyst\Desktop\sepsis.exe: A2 CB 1B 5C 66 99 DA
01 00 00 00 00 00 00 00 00 00 00 00 00 02 00 00 00
HKLM\SYSTEM\CurrentControlSet\Services\bam\State\UserSettings\S-1-5-21-2380876323-637646090-3293345604-1000\\Device\HarddiskVolume1\Users\analyst\Desktop\sepsis.exe: A2 CB 1B 5C 66
99 DA 01 00 00 00 00 00 00 00 00 00 00 00 00 02 00 00 00
```

Figure: Regshot analysis

6.3 Services/Processes started:

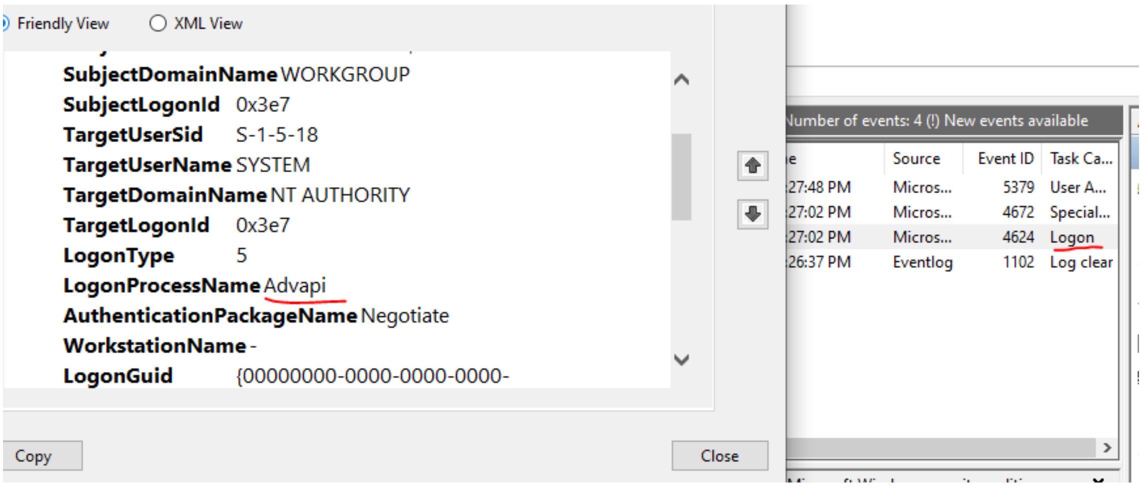
The ransomware initiates DNS queries, HTTP conversations, and TCP/UDP communication to establish connections with external servers for command-and-control purposes. Additionally, it attempts to modify the Windows registry, including interactions with MMC snap-ins configuration and modification attempts related to win logon shell, potentially indicating efforts to maintain persistence and manipulate system settings to align with its malicious objectives.

The ransomware's behavior suggests a sophisticated approach to file encryption, ransom note delivery, and potential data exfiltration, leveraging network traffic analysis and file operations to execute its malicious activities.

Sepsis Ransomware primarily focuses on file encryption and ransom delivery, rather than directly starting new services or processes. However, it starts existing Windows services or processes to execute its malicious activities and maintain persistence on the infected system.:

1. **Windows Autologon Feature:**

    - Sepsis Ransomware modifies registry keys to leverage the Windows autologon feature, ensuring that it executes automatically upon system boot. By manipulating these registry keys, the ransomware ensures persistence and maintains its foothold on the infected system.



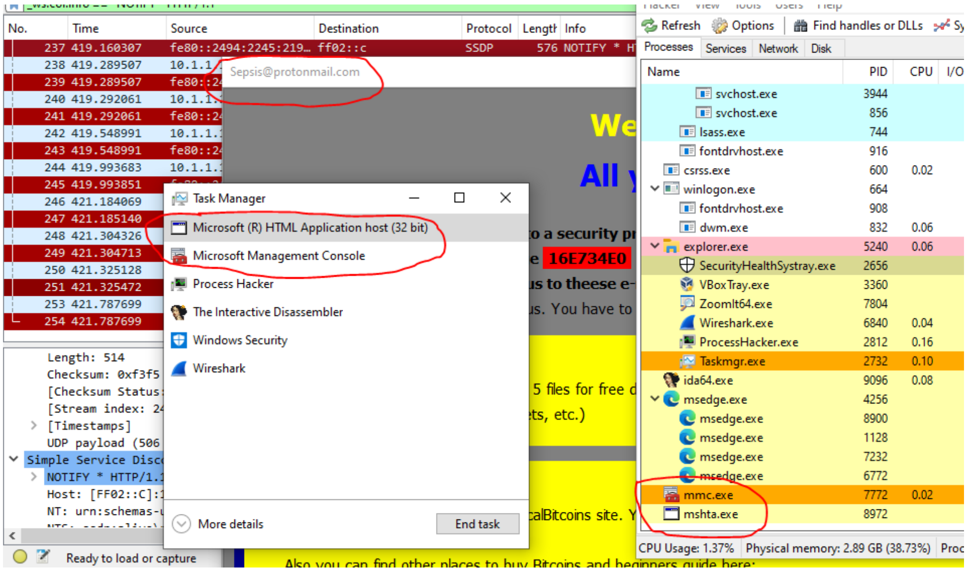
```
SubjectDomainName WORKGROUP
SubjectLogonId    0x3e7
TargetUserSid     S-1-5-18
TargetUserName    SYSTEM
TargetDomainName  NT AUTHORITY
TargetLogonId     0x3e7
LogonType         5
LogonProcessName  Advapi
AuthenticationPackageName Negotiate
WorkstationName   -
LogonGuid         {00000000-0000-0000-0000-
```

Number of events: 4 (!) New events available

e	Source	Event ID	Task Ca...
27:48 PM	Micros...	5379	User A...
27:02 PM	Micros...	4672	Special...
27:02 PM	Micros...	4624	Logon
26:37 PM	Eventlog	1102	Log clear

2. **Cryptographic Services:**

    - As part of its encryption routine, the ransomware interact with cryptographic services provided by the Windows operating system to generate encryption keys, encrypt files, and communicate with command and control servers securely. These services facilitate the encryption process and enable the ransomware to securely transmit encryption keys to the attackers' servers.



```
No.  Time        Source                  Destination  Protocol  Length Info
237  419.160307  fe80::2494:2245:219…    ff02::c      SSDP      576 NOTIFY * H
238  419.289507  10.1.1.
239  419.289507  fe80::2
240  419.292061  10.1.1.
241  419.292061  fe80::2
242  419.548991  10.1.1.
243  419.548991  fe80::2
244  419.993683  10.1.1.
245  419.993851
246  421.184069
247  421.185140
248  421.304326
249  421.304713
250  421.325128
251  421.325472
253  421.787699
254  421.787699
```

```
    Length: 514
    Checksum: 0xf3f5
    [Checksum Status:
    [Stream index: 24
  > [Timestamps]
    UDP payload (506
v Simple Service Disc
  > NOTIFY * HTTP/1.1
    Host: [FF02::C]:1
    NT: urn:schemas-u
```

Ready to load or capture

Sepsis@protonmail.com

We

All y

o a security pr

e 16E734E0

us to theese e-

us. You have to

5 files for free d

ts, etc.)

alBitcoins site. Y

Also you can find other places to buy Bitcoins and beginners guide here:

Task Manager

- Microsoft (R) HTML Application host (32 bit)
- Microsoft Management Console
- Process Hacker
- The Interactive Disassembler
- Windows Security
- Wireshark

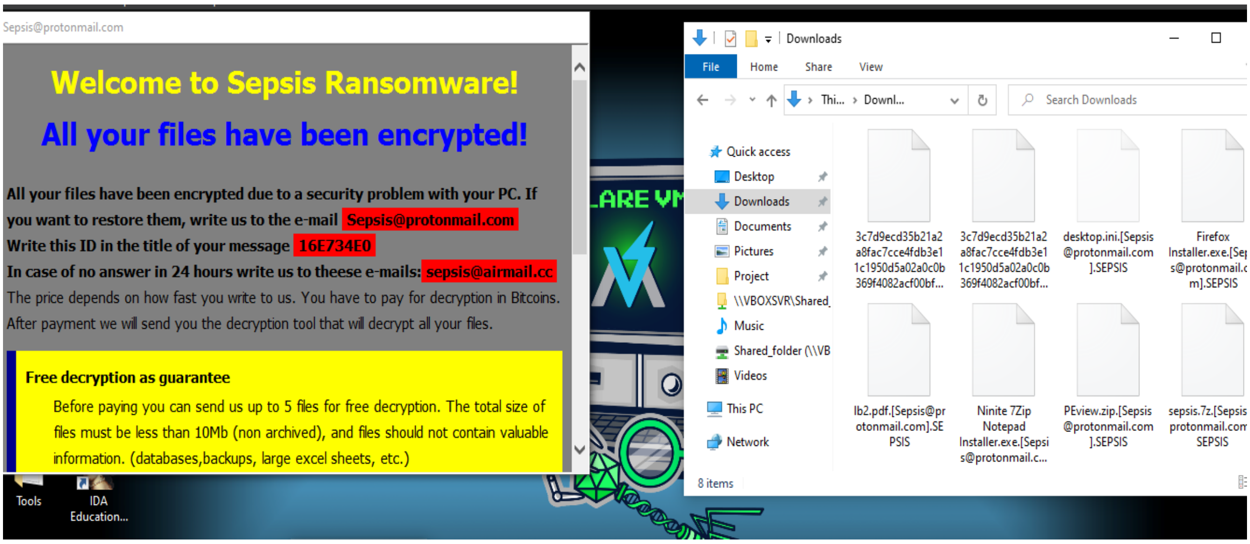
Name	PID	CPU	I/O
svchost.exe	3944		
svchost.exe	856		
lsass.exe	744		
fontdrvhost.exe	916		
csrss.exe	600	0.02	
winlogon.exe	664		
fontdrvhost.exe	908		
dwm.exe	832	0.06	
explorer.exe	5240	0.06	
SecurityHealthSystray.exe	2656		
VBoxTray.exe	3360		
ZoomIt64.exe	7804		
Wireshark.exe	6840	0.04	
ProcessHacker.exe	2812	0.16	
Taskmgr.exe	2732	0.10	
ida64.exe	9096	0.08	
msedge.exe	4256		
msedge.exe	8900		
msedge.exe	1128		
msedge.exe	7232		
msedge.exe	6772		
mmc.exe	7772	0.02	
mshta.exe	8972		

CPU Usage: 1.37% | Physical memory: 2.89 GB (38.73%)

3. **Network Communication Processes:**

- Sepsis Ransomware communicates with command and control servers over HTTP or HTTPS protocols to receive instructions, transmit encrypted data, and facilitate ransom negotiations. It interact with network communication processes such as the Windows networking service (svchost.exe) to establish and maintain network connections for communication with external servers.

6.4 Data Encrypted



Sepsis@protonmail.com

Welcome to Sepsis Ransomware!

All your files have been encrypted!

All your files have been encrypted due to a security problem with your PC. If you want to restore them, write us to the e-mail Sepsis@protonmail.com

Write this ID in the title of your message 16E734E0

In case of no answer in 24 hours write us to theese e-mails: sepsis@airmail.cc

The price depends on how fast you write to us. You have to pay for decryption in Bitcoins. After payment we will send you the decryption tool that will decrypt all your files.

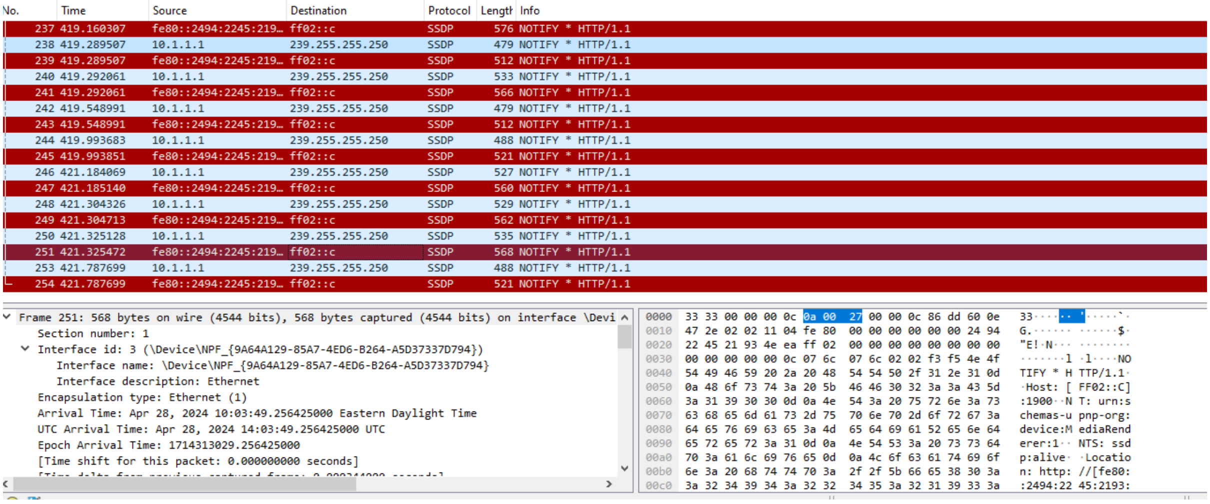
Free decryption as guarantee

Before paying you can send us up to 5 files for free decryption. The total size of files must be less than 10Mb (non archived), and files should not contain valuable information. (databases,backups, large excel sheets, etc.)

Tools
IDA Education...
Downloads
File
Home
Share
View
Thi... › Downl...
Search Downloads
Quick access
Desktop
Downloads
Documents
Pictures
Project
\\VBOXSVR\Shared_
Music
Shared_folder (\\VB
Videos
This PC
Network
3c7d9ecd35b21a2a8fac7cce4fdb3e11c1950d5a02a0c0b369f4082acf00bf...
3c7d9ecd35b21a2a8fac7cce4fdb3e11c1950d5a02a0c0b369f4082acf00bf...
desktop.ini.[Sepsis@protonmail.com].SEPSIS
Firefox Installer.exe.[Sepsis@protonmail.com].SEPSIS
lb2.pdf.[Sepsis@protonmail.com].SEPSIS
Ninite 7Zip Notepad Installer.exe.[Sepsis@protonmail.c...
PEview.zip.[Sepsis@protonmail.com].SEPSIS
sepsis.7z.[Sepsis protonmail.com SEPSIS
8 items

7. Supporting DATA

7.1Log files

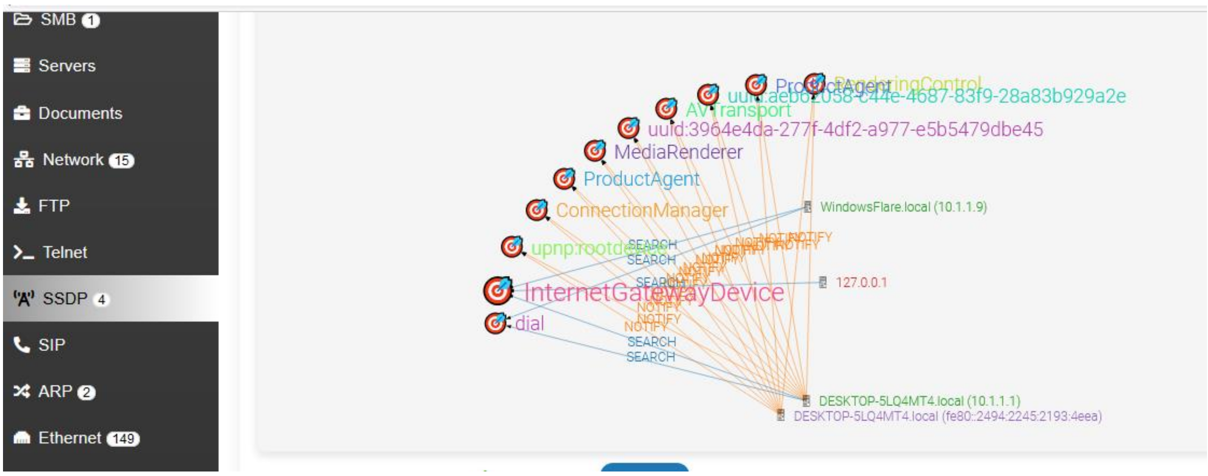


```
No.  Time        Source                  Destination      Protocol Length Info
 237 419.160307  fe80::2494:2245:219…  ff02::c          SSDP      576 NOTIFY * HTTP/1.1
 238 419.289507  10.1.1.1                239.255.255.250  SSDP      479 NOTIFY * HTTP/1.1
 239 419.289507  fe80::2494:2245:219…  ff02::c          SSDP      512 NOTIFY * HTTP/1.1
 240 419.292061  10.1.1.1                239.255.255.250  SSDP      533 NOTIFY * HTTP/1.1
 241 419.292061  fe80::2494:2245:219…  ff02::c          SSDP      566 NOTIFY * HTTP/1.1
 242 419.548991  10.1.1.1                239.255.255.250  SSDP      479 NOTIFY * HTTP/1.1
 243 419.548991  fe80::2494:2245:219…  ff02::c          SSDP      512 NOTIFY * HTTP/1.1
 244 419.993683  10.1.1.1                239.255.255.250  SSDP      488 NOTIFY * HTTP/1.1
 245 419.993851  fe80::2494:2245:219…  ff02::c          SSDP      521 NOTIFY * HTTP/1.1
 246 421.184069  10.1.1.1                239.255.255.250  SSDP      527 NOTIFY * HTTP/1.1
 247 421.185140  fe80::2494:2245:219…  ff02::c          SSDP      560 NOTIFY * HTTP/1.1
 248 421.304326  10.1.1.1                239.255.255.250  SSDP      529 NOTIFY * HTTP/1.1
 249 421.304713  fe80::2494:2245:219…  ff02::c          SSDP      562 NOTIFY * HTTP/1.1
 250 421.325128  10.1.1.1                239.255.255.250  SSDP      535 NOTIFY * HTTP/1.1
 251 421.325472  fe80::2494:2245:219…  ff02::c          SSDP      568 NOTIFY * HTTP/1.1
 253 421.787699  10.1.1.1                239.255.255.250  SSDP      488 NOTIFY * HTTP/1.1
 254 421.787699  fe80::2494:2245:219…  ff02::c          SSDP      521 NOTIFY * HTTP/1.1
```

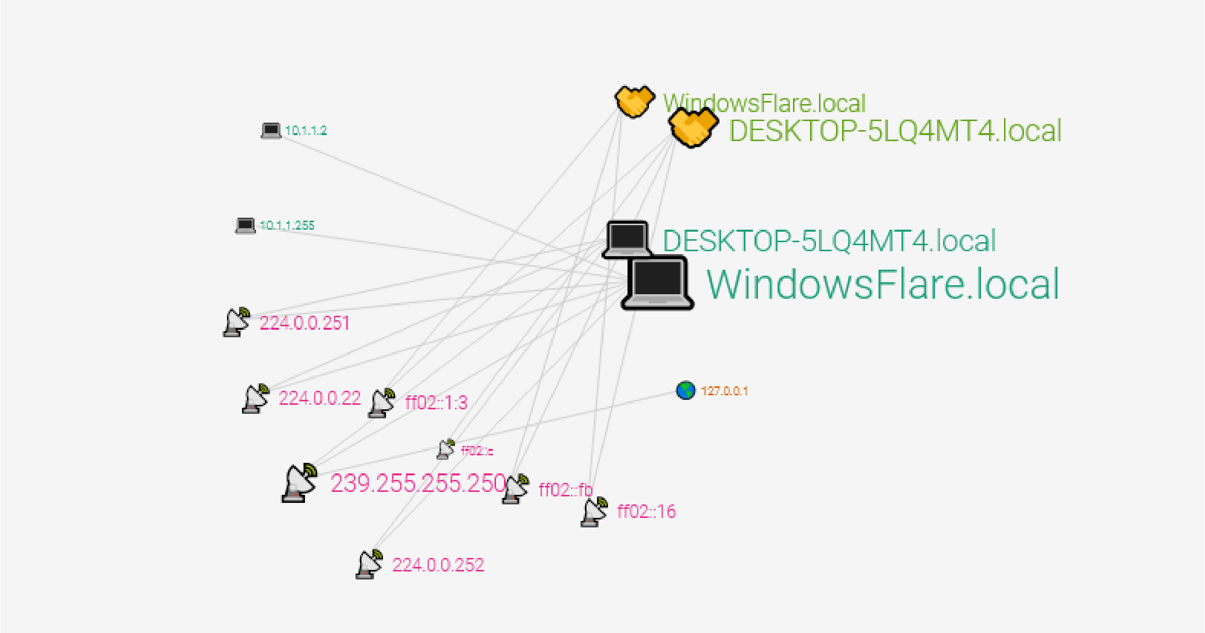
```
Frame 251: 568 bytes on wire (4544 bits), 568 bytes captured (4544 bits) on interface \Devi
   Section number: 1
   Interface id: 3 (\Device\NPF_{9A64A129-85A7-4ED6-B264-A5D37337D794})
      Interface name: \Device\NPF_{9A64A129-85A7-4ED6-B264-A5D37337D794}
      Interface description: Ethernet
   Encapsulation type: Ethernet (1)
   Arrival Time: Apr 28, 2024 10:03:49.256425000 Eastern Daylight Time
   UTC Arrival Time: Apr 28, 2024 14:03:49.256425000 UTC
   Epoch Arrival Time: 1714313029.256425000
   [Time shift for this packet: 0.000000000 seconds]
```

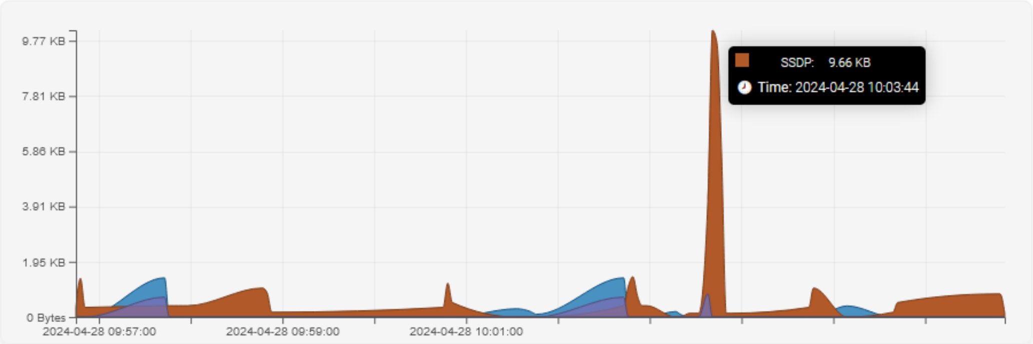
```
0000  33 33 00 00 00 0c 0a 00  27 00 00 0c 86 dd 60 0e   33······ ·······`·
0010  47 2e 02 02 11 04 fe 80  00 00 00 00 00 00 24 94   G.······ ······$·
0020  22 45 21 93 4e ea ff 02  00 00 00 00 00 00 00 00   "E!·N··· ········
0030  00 00 00 00 00 0c 07 6c  07 6c 02 02 f3 f5 4e 4f   ·······l ·l····NO
0040  54 49 46 59 20 2a 20 48  54 54 50 2f 31 2e 31 0d   TIFY * H TTP/1.1·
0050  0a 48 6f 73 74 3a 20 5b  46 46 30 32 3a 3a 43 5d   ·Host: [ FF02::C]
0060  3a 31 39 30 30 0d 0a 4e  54 3a 20 75 72 6e 3a 73   :1900··N T: urn:s
0070  63 68 65 6d 61 73 2d 75  70 6e 70 2d 6f 72 67 3a   chemas-u pnp-org:
0080  64 65 76 69 63 65 3a 4d  65 64 69 61 52 65 6e 64   device:M ediaRend
0090  65 72 65 72 3a 31 0d 0a  4e 54 53 3a 20 73 73 64   erer:1·· NTS: ssd
00a0  70 3a 61 6c 69 76 65 0d  0a 4c 6f 63 61 74 69 6f   p:alive· ·Locatio
00b0  6e 3a 20 68 74 74 70 3a  2f 2f 5b 66 65 38 30 3a   n: http: //[fe80:
00c0  3a 32 34 39 34 3a 32 32  34 35 3a 32 31 39 33 3a   :2494:22 45:2193:
```

7.2 Network traces



7.3 Screenshots





7.4 Other data (database dumps, config files, etc)

GDRIVE Link:

https://drive.google.com/drive/folders/1J3ULVEkHueaGC0o8E6gDY6f8qz5rL2fI?usp=sharing

Identified Malicious Indicators:

1. **Creation of Unfamiliar Registry Keys:**
   - The addition of registry keys under HKU\S-1-5-21-2380876323-637646090-3293345604-1000\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012024042120240422 raises concerns. While some modifications may be legitimate, the creation of unfamiliar keys demands scrutiny, as they could serve as hiding places for malware or facilitate unauthorized access.
2. **Altered Values Pertaining to Internet Cache:**
   - Deletion and addition of values within the Internet Cache settings, particularly under HKU\S-1-5-21-2380876323-637646090-3293345604-1000\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\Cache\Extensible Cache\MSHist012024042120240422, indicate potential tampering with browser cache configurations. Malicious actors often manipulate cache settings to conceal their activities or facilitate the persistence of malware.
3. **Addition of Numerous Registry Values:**
   - The insertion of a substantial number of registry values, notably those associated with system services and components, suggests a significant alteration to the system's configuration. While some additions may be benign, the sheer volume raises suspicion and warrants thorough examination to identify any malicious payloads or unauthorized modifications.

8 RECOMMENDATIONS

8.1 Response Plan for Malware Infection

In the event of suspected malware infection, prompt action is imperative to mitigate potential damage and restore system integrity. The following steps outline a comprehensive response plan:

1. Recovery:

- Isolation: Immediately disconnect the affected system from the network to prevent the spread of malware to other devices.
- Backup: If feasible, create backups of critical files and data to prevent loss during the recovery process.
- System Restore: Utilize system restore or recovery tools to revert the system to a previously known clean state before the malware infection occurs.
- Reinstallation: In severe cases, consider reinstalling the operating system and applications from a clean backup to ensure complete removal of the malware.

2. Mitigation:

- Antivirus Scan: Conduct a thorough antivirus scan using updated security software to detect and remove any remnants of malware.
- Patch and Update: Ensure all operating system, applications, and security software are up to date with the latest patches and updates to address known vulnerabilities exploited by the malware.
- Password Change: Change passwords for user accounts, especially if there is a risk of credential theft or compromise.
- Enhance Security Measures: Strengthen security measures such as firewalls, intrusion detection systems, and endpoint protection to prevent future infections.

3. Additional Steps:

- Forensic Analysis: Perform a forensic analysis of the affected system to determine the extent of damage, identify the attack vector, and gather evidence for further investigation.
- Incident Response: Follow established incident response procedures to contain and mitigate the impact of the malware infection on the organization's network and infrastructure.
- User Awareness Training: Educate users about malware and phishing risks, emphasizing the importance of vigilance and safe computing practices to prevent future infections.
- Incident Reporting: If the malware infection is part of a larger-scale attack or sensitive data has been compromised, report the incident to relevant authorities or cybersecurity organizations for further assistance and investigation.

It is essential to execute the recovery and mitigation process systematically to ensure thorough removal of the malware and restoration of the system's security posture. Implementing proactive security measures can help prevent similar incidents in the future.

8.2 Defending Against Sepsis Ransomware:

1. Empowering Employees:

Encourage and support employees by providing regular cybersecurity training sessions in a friendly and accessible manner. Help them understand the importance of staying vigilant against suspicious emails and attachments, just like being cautious about unexpected packages arriving at their doorstep.

2. Protecting Every Workstation:

Equip each employee's workstation, whether it's a desktop, laptop, or server, with robust antivirus software, treating it like a personal guardian against online threats, including the sneaky Sepsis ransomware. Regularly update this software to keep it sharp and ready to fend off any new dangers.

3. Keeping Systems Healthy:

Maintain the health of your digital ecosystem by regularly applying patches and updates to your computers and software, just like giving them their daily vitamins. This helps strengthen their defenses against the Sepsis ransomware and other cyber nasties.

4. Guarding the Digital Gates:

Set up digital gatekeepers, like firewalls and web filters, to watch over your network and keep out unwanted guests, such as the Sepsis ransomware. They're like the vigilant guards at your business's entrance, making sure only authorized visitors get through.

5. Granting Access Wisely:

Treat user permissions like keys to your business's most sensitive areas. Only give employees access to what they need to do their jobs, just like handing out keys to specific rooms in your office building. This minimizes the risk of the Sepsis ransomware accessing critical data and systems.

6. Backing Up Your Business:

Keep your business's valuable data safe by regularly backing it up, just like making copies of important documents. Store these backups securely offline, like putting them in a locked safe, to ensure they're out of reach of the Sepsis ransomware.

7. Preparing for the Worst:

Have a plan in place for when things go wrong, just like having a fire drill. Make sure everyone knows what to do if the Sepsis ransomware strikes and practice your response regularly to ensure everyone can spring into action if needed.