Analysing the HotKeyCamo

-A GUI shell for compiling AutoHotKey scripts -

Related Forum/download:
http://www.autohotkey.com/forum/topic49952.html

Okay I launched a Diff on the same script.
Once compiled with the 'normal' AHK-Compiler and the other time with HotKeyCamo.
(Make sure the ahkExe is uncompressed. May be decompress it with 'Upx –d *.exe')

Okay here are the changes…
…and how to deal with them to be able to decompile the script with myAutToExe

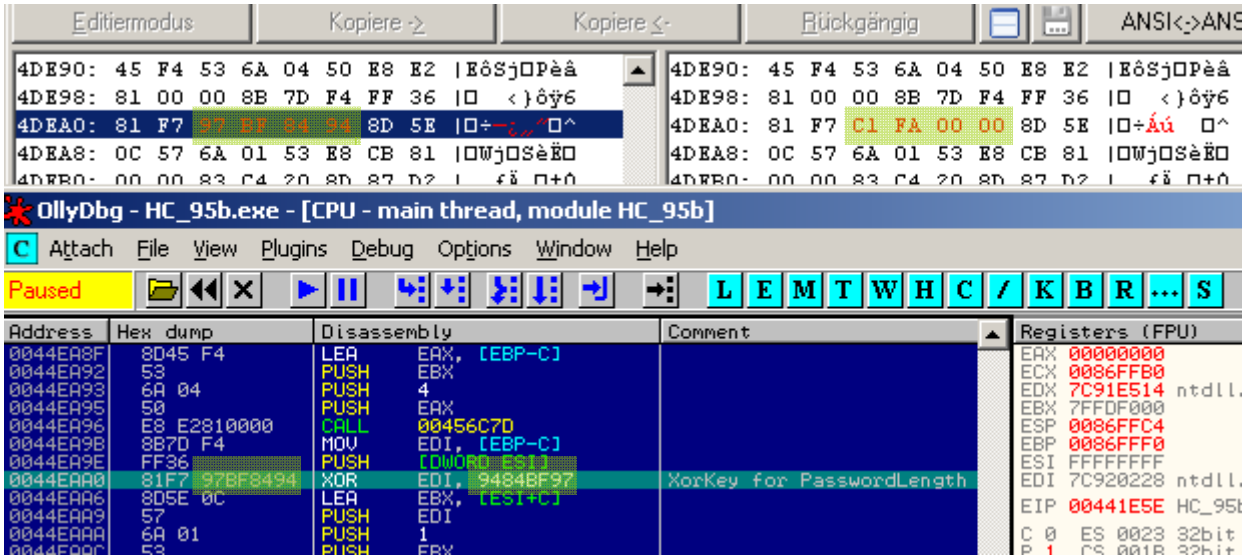
Hmm well for better understand I put that part on the top(even when it comes last in the File)

The Script (Camo vs Normal)


```
The Compiled Script AutoIT File format:
---------------------------------------

AutoIt_Signature        size 0x10 Bytes  String "£HK..."
SubType                 size 0x1 Byte   Should be 0x03 (0x01 AutoIT2; )
PassphraseLen           size 0x4 Bytes   [XorKey=0x000FAC1]
Passphrase              size (depends on PassphraseLen)[StrKey=C3D2]
ResType                 size 0x4 Byte   eString: "FILE"     [
StrKey=16FA]
ScriptType              eString ">AUTOIT SCRIPT<"           [LenKey=29BC,
StrKey=A25E]
CompiledPathName        eString "C:\...\Temp\aut26A.tmp"    [LenKey=29AC,
StrKey=F25E]
IsCompressed            size 0x1 Byte
ScriptSize   Compressed size 0x4 Byte                       [XorKey=45AA]
```

Now the changes in the AHK-Interpreter stub that's at the beginning of each AHKExe

1. LenKeyXorValue Change



Critical Mod:
This Value is used the get the len of the Passphrase. (see below for FileFormat details)
MyAutToExe need to know that length to correct read in the Password and the data that follows.

So here you'll need to go into the source code
Search for ' FAC1' in the whole Project until you get here:

```
    ' ===> Get Script Password
      Dim MD5PassphraseHash As New StringReader
      If bIsOldScript Then
       ' Old AutoIT Script if branch...
       ' Move three bytes back since SubType is only 1 Byte but befroe we read 4
byte
         .Move -3
         MD5PassphraseHash = GetEncryptStr(64193, 50130, File) '&HFAC1, &HC3D2
```

Change the last line to

```
         MD5PassphraseHash = GetEncryptStr(&H9484BF97, 50130, File) '&HFAC1,
&HC3D2
```

Now it should work.

Decompiled script should look like that:

```
; HotkeyCamo ~0.9.5.0>
DetectHiddenText, On
SetTi…
```

Hint on finding the '9484BF97' value.

Note that this is the len of the Passphrase. Usually that value will be in a range of 0 to 255 (0x000000 00 to 0x000000 FF).
So the this last three bytes will be the same nearly all the time xx 00 00 00 .
Goto to ScriptStart +0x11 there is 'B7 BF 84 94' or '9484BFB7'. So a search for the hexstring 'BF 84 94' in the uncompressed *.ahkExe will reveal that there a 97 before them and so the Full XorKeyValue is 97 BF 84 94 -> '9484BF97'.

And well there's as well an alternative in case you somehow can't find this '9484BF97' value (or I explained it to messy)
Change the code as the following:

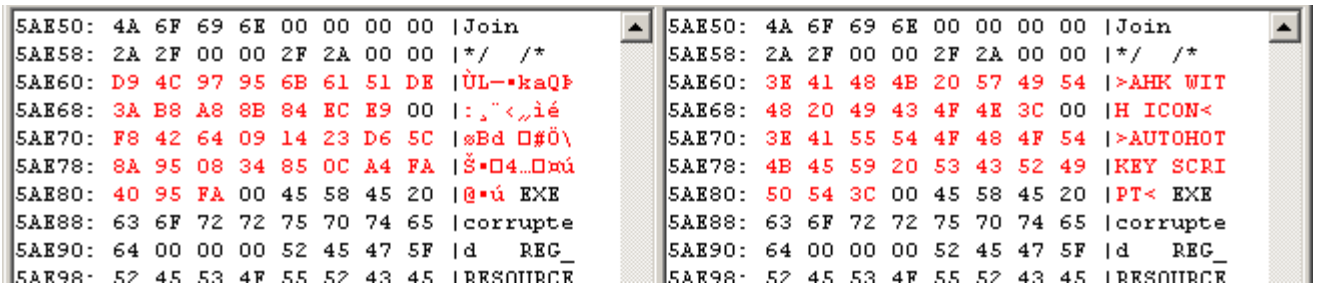
```
' ===> Get Script Password
      Dim MD5PassphraseHash As New StringReader
```

```
      If bIsOldScript Then
       ' Old AutoIT Script if branch...
       ' Move three bytes back since SubType is only 1 Byte but before we read 4
byte
         .Move -3
         'MD5PassphraseHash = GetEncryptStr(64193, 50130, File) '&HFAC1, &HC3D2
         'MD5PassphraseHash = GetEncryptStr(&H9484BF97, 50130, File) '&HFAC1,
&HC3D2

      .Move 4

      Dim StrLen&
      StrLen = 32

      MD5PassphraseHash = DeCrypt(.FixedString(StrLen), 50130 + StrLen)
```

-> guess/changing the StrLen = 32 that long till it fits.

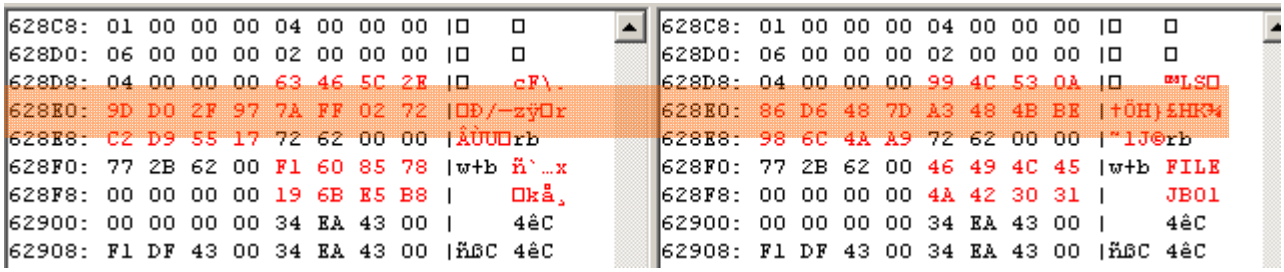
Okay now the rest of the changes:

2. MainScript FileName messed up



Uncritical:
Hmm well it will mess updetection what script 'flavor' this script is and so the decompiled file will get the extension *.au3 (since AutoIT is the standard). But I think you can handle that ;)
Hint: Rename *.au3 -> *.ahk

3. AHK-Script Start Pattern; FILE Start Pattern and Start Pattern for compressed data are messed



Uncritical:
Check if myAutToExe correctly found the start of the script. (It's using the heuristic EndOf_PE-ExeFile => Start of Script)
So in the log there should be:
`---> ScriptStartOffset: 00064E00`
… else enter '64E00' in the Textbox (for the start offset of the script) manually

Click on yes if myAutToExe complains about invalid File marker.
(This JB01 thing don't comes into games since the script don't gets compressed)