Real world computer security Információ biztonság, vírusfertőzések

Boldizsár Bencsáth

(and Levente Buttyán)

Aurora experiment

- https://www.youtube.com/watch?v=fJyWngDco3g
- Cyber-phisical attack test
- Code can make physical damage
- https://www.youtube.com/watch?v=7g0pi4J8auQ

Example: Stuxnet

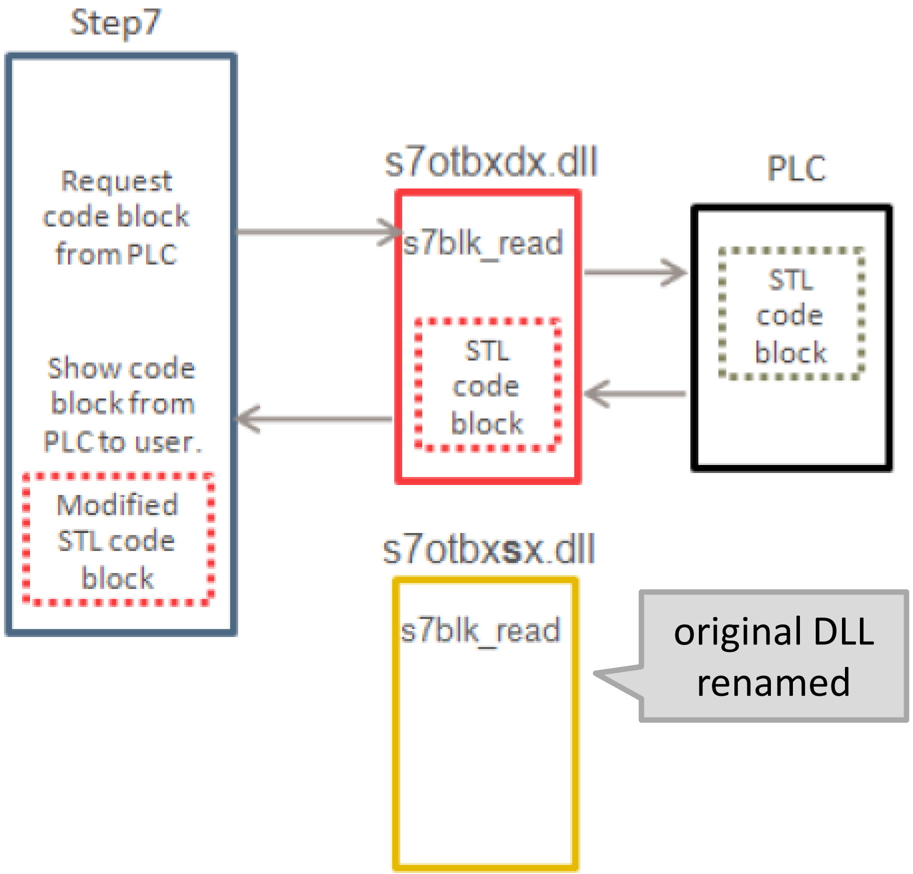
- a computer worm first discovered in July 2010
- designed to physically destroy uranium centrifuges in the Natanz enrichment facility in Iran
- infected computers running Windows and spread by
  - infecting removable drives
  - copying itself over the network using a variety of means
  - copying itself to Step 7 projects (runs automatically when project is opened)
- if not in the target environment, it did nothing
- once inside the target environment, it reprogramed PLCs controlling the rotation speed of the uranium centrifuges
- manipulation of the rotation speed led to physical damage
  - hundreds of centrifuges were destroyed

Stuxnet – Special features

- very specific target (nuclear facility)
- objective was physical destruction by logical means (sabotage)
- worm-like spreading → thousands of infected machines
- yet, remained uncovered for months (years?)
  - time was enough to reach its target
  - careful testing during development to avoid anomalies on infected machines
- used multiple zero-day exploits and a digitally signed driver
  - signature was created with the possibly compromised key of a Taiwanese hardware manufacturer
- used advanced privilege escalation, code injection, and rootkit techniques, as well as a peer-to-peer update mechanism
- first known malware that contained also a PLC rootkit
- required a testbed similar to the target environment
  - who has a testbed with uranium centrifuges?
- state sponsored attackers behind

Stuxnet – PLC attack

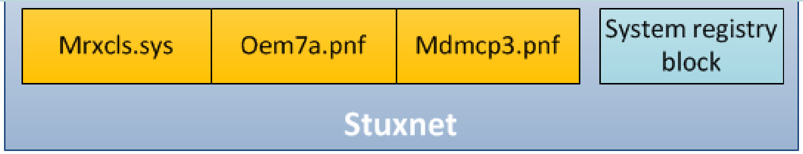
- PLC devices are loaded with blocks of code and data by a programming device (engineering workstation running a PLC management software on Windows)
- PLC block exchange between the programming device and the PLC is handled by a DLL (s7otbxdx.dll)
- Stuxnet replaces this DLL with its own copy
  - can monitor PLC blocks being written to and read from the PLC
  - can infect a PLC by inserting its own blocks and replacing or infecting existing blocks
  - can mask the fact that a PLC is infected

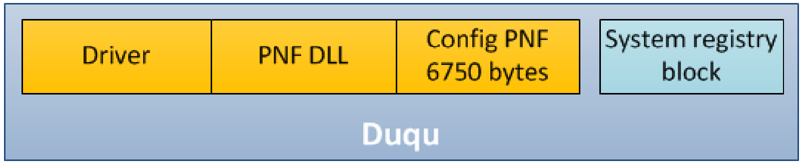


DPRK

- Stuxnet kernel drivers

File name	Size (bytes)	Compilation date	Where and when it was used	Digital signature/signing date
Mrxcls.sys	19840	01.01.2009	Stuxnet (22.06.2009)	No
Mrxcls.sys	26616	01.01.2009	Stuxnet (01.03.2010/14.04.2010)	Realtek, 25.01.2010
Mrxnet.sys	17400	25.01.2010	Stuxnet (01.03.2010/14.04.2010)	Realtek, 25.01.2010
Jmidebs.sys	25552	14.07.2010	Presumably, Stuxnet	Jmicron, unknown

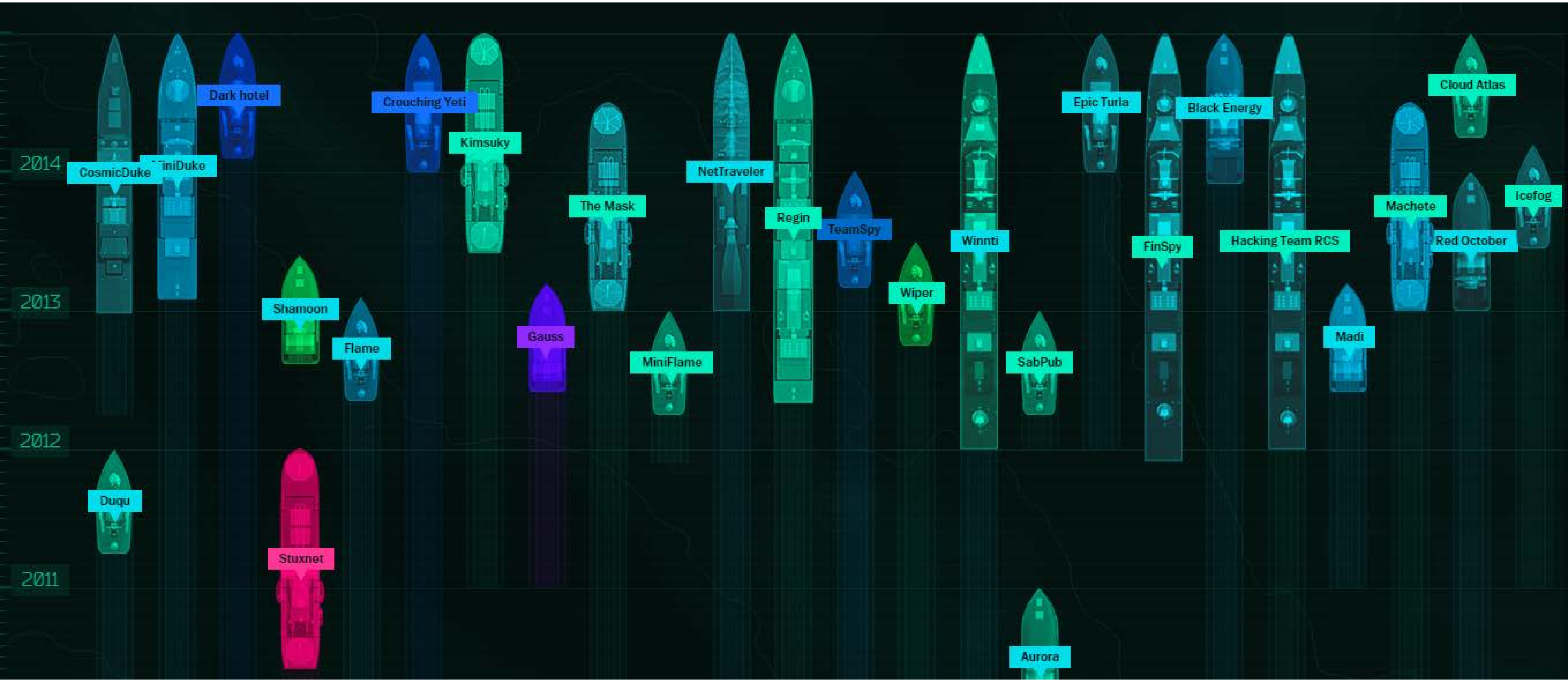




DPRK – North Korea

- Dark Hotel: A DPRK related APT
- Nuclear program: many unsuccessful rocket experiments … strange
- Stuxnet kernel driver: maybe it is not related to Stuxnet, but to DPRK somehow?

Other examples for targeted attacks



https://apt.securelist.com/

The Duqu font vulnerability

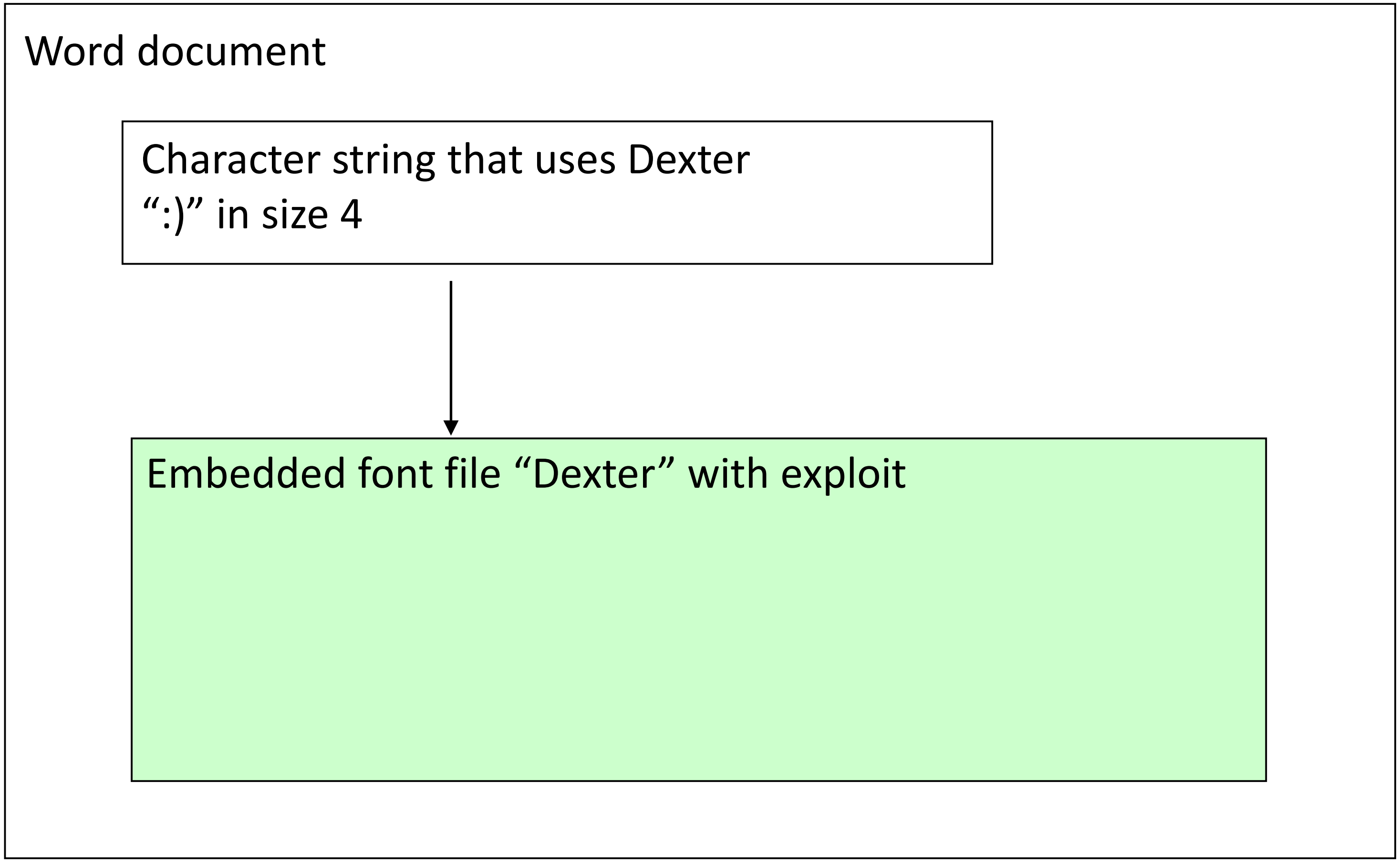
- Font parsing problem
- Kernel space
- All windows versions (nearly)
- Bitmap fonts – composite bitmap offset
- Glyph routines

- Font dump
- Write-up
- Repro font
- After fix, Microsoft ran a project to find cloned code with same problem

Duqu dropper – the idea

- Duqu dropper was a .doc file
- With embedded font
- Font exploited Windows kernel vulnerability (CVE-2011-3402)
- Creating such exploit needs lots of effort, even understanding it needs much work
- Shell code runs then at kernel level – designing it needs precise work, much effort
- (It took a long time for exploit pack creators to incorporate Duqu exploit)
- Idea: Let’s change only user space components from the dropper
- Duqu exploit and kernel level parts will do the hard work for us

Dropper structure

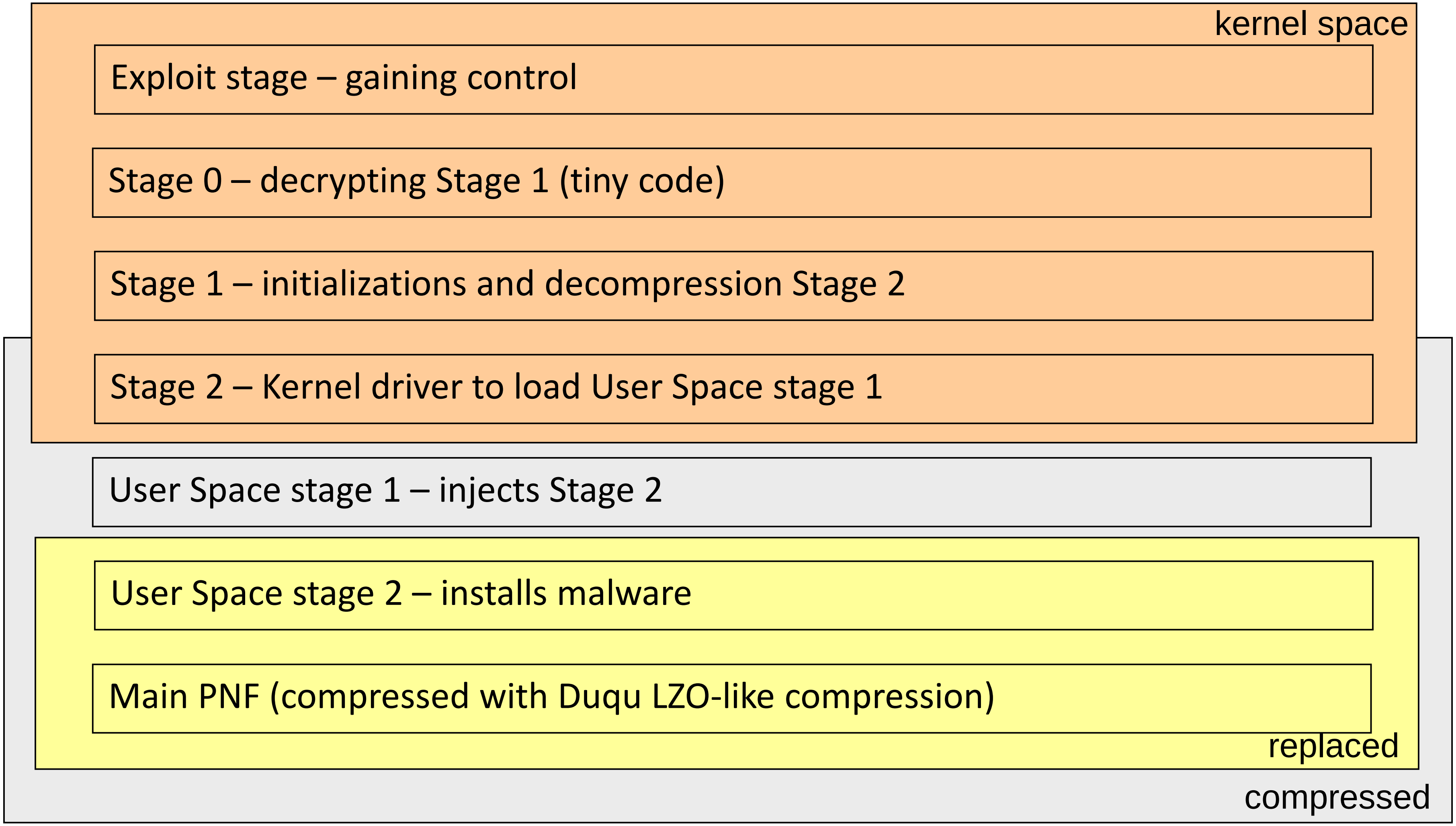


Word document

Character string that uses Dexter
“:)” in size 4

Embedded font file “Dexter” with exploit

Dropper font file logical structure



kernel space

Exploit stage – gaining control

Stage 0 – decrypting Stage 1 (tiny code)

Stage 1 – initializations and decompression Stage 2

Stage 2 – Kernel driver to load User Space stage 1

User Space stage 1 – injects Stage 2

User Space stage 2 – installs malware

Main PNF (compressed with Duqu LZO-like compression)

replaced

compressed

Major problems, work to be done

- Kernel level parts are not yet documented in detail publicly
- Decrypting parts and analysis of kernel level code was needed
- Compression used in kernel level is not documented
- User space stages were also not documented in detail

How to perform

- Let all kernel level stuff as it is (from exploit to stage 2)
- Let user space stage 1 to inject our malware
- Replace User space stage 2 and PNF payload

- First we had to decipher encrypted parts and analyze code
- Kernel level parts are not detailed much in public reports
- Problem: Some parts are compressed by stage 1 kernel code
- Compression is not documented by public reports either
- The code contains the decompression routine. We cannot compress our own payload as we need the proper compression routine (or a workaround to turn off decompression at all)

Decompressor in Duqu dropper

Duqu dropper decompressor	LZMA at read.pudn.com/downloads94/sourcecode/zip/372835/Source/lzma_depack.inc__.htm
seg000:000011C0 000 lea eax, [ebx+eax*4] seg000:000011C3 000 mov ecx, eax seg000:000011C5 000 mov eax, [ecx] seg000:000011C7 000 mov edx, [ebp-0Ch] seg000:000011CA 000 **shr edx, 0Bh** ; seg000:000011CD 000 **mul edx** seg000:000011CF 000 cmp eax, [ebp-10h] seg000:000011D2 000 jbe short loc_11FC seg000:000011D4 000 mov [ebp-0Ch], eax seg000:000011D7 000 **mov edx, 800h** seg000:000011DC 000 sub edx, [ecx] ; seg000:000011DE 000 **shr edx, 5** ; seg000:000011E1 000 add [ecx], edx	@loc_401320: mov ecx,[edi] mov edx,eax **shr edx,0Bh** **imul edx,ecx** cmp [ebp+0Ch],edx jnb @loc_40136C mov esi,[ebp-10h] mov eax,edx **mov edx,800h** sub edx,ecx **shr edx,5** add edx,ecx xor ecx,ecx

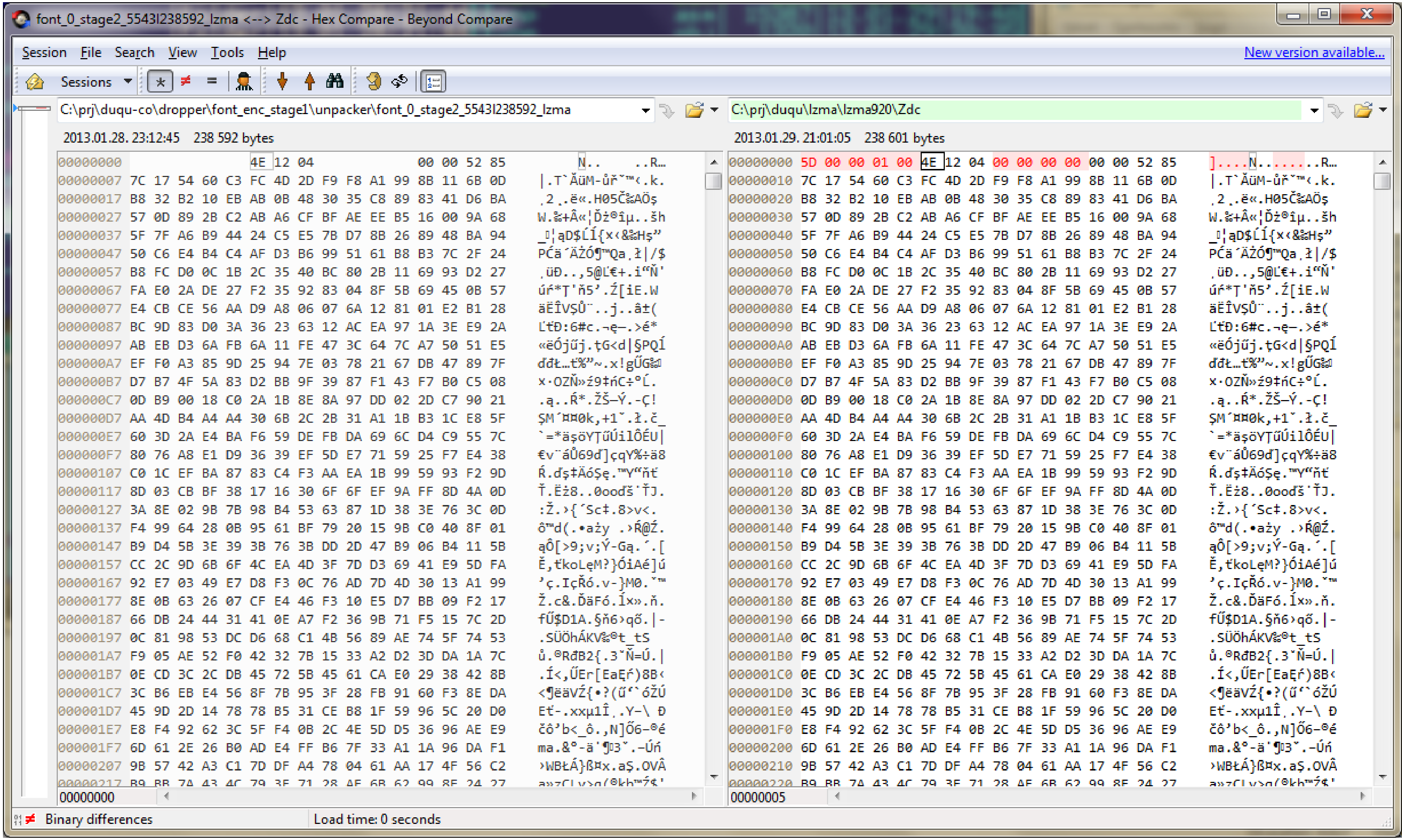
Duqu dropper compression

- We found very similar code chunks in LZMA
- However, we could not find an exactly same implementation
- We ran Duqu decompressor to decompress payload
- Re-compressed with LZMA to prove that it is LZMA
- We got back the original bye stream with command line:

lzma.exe e Zd Zdc -a1 -d16

- Dictionary size is in Duqu between d15-17, default of lzma.exe is d22

Duqu dropper LZMA verified



font_0_stage2_5543l238592_lzma <--> Zdc - Hex Compare - Beyond Compare

C:\prj\duqu-co\dropper\font_enc_stage1\unpacker\font_0_stage2_5543l238592_lzma

2013.01.28. 23:12:45   238 592 bytes

```
00000000                4E 12 04             00 00 52 85     N..   ..R…
00000007 7C 17 54 60 C3 FC 4D 2D F9 F8 A1 99 8B 11 6B 0D   |.T`ĂüM-ůř˘™‹.k.
00000017 B8 32 B2 10 EB AB 0B 48 30 35 C8 89 83 41 D6 BA   ¸2˛.ë«.H05Č‰ƒAÖş
00000027 57 0D 89 2B C2 AB A6 CF BF AE EE B5 16 00 9A 68   W.‰+Â«¦Ďż®îµ..šh
00000037 5F 7F A6 B9 44 24 C5 E5 7B D7 8B 26 89 48 BA 94   _.¦ąD$ĹĺÍ{×‹&‰Hş”
00000047 50 C6 E4 B4 C4 AF D3 B6 99 51 61 B8 B3 7C 2F 24   PĆä´ÄŻÓ¶™Qa¸ł|/$
00000057 B8 FC D0 0C 1B 2C 35 40 BC 80 2B 11 69 93 D2 27   ¸üĐ..,5@Ľ€+.i“Ň'
00000067 FA E0 2A DE 27 F2 35 92 83 04 8F 5B 69 45 0B 57   úŕ*Ţ'ň5’ƒ.Ź[iE.W
00000077 E4 CB CE 56 AA D9 A8 06 07 6A 12 81 01 E2 B1 28   äËÎVŞŮ¨..j..âą(
00000087 BC 9D 83 D0 3A 36 23 63 12 AC EA 97 1A 3E E9 2A   ĽťƒĐ:6#c.¬ę—.>é*
00000097 AB EB D3 6A FB 6A 11 FE 47 3C 64 7C A7 50 51 E5   «ëÓjűj.ţG<d|§PQĺ
000000A7 EF F0 A3 85 9D 25 94 7E 03 78 21 67 DB 47 89 7F   ďđŁ…ť%”~.x!gŰG‰.
000000B7 D7 B7 4F 5A 83 D2 BB 9F 39 87 F1 43 F7 B0 C5 08   ×·OZƒŇ»ź9‡ńC÷°Ĺ.
000000C7 0D B9 00 18 C0 2A 1B 8E 8A 97 DD 02 2D C7 90 21   .ą..Ŕ*.ŽŠ—Ý.-Ç.!
000000D7 AA 4D B4 A4 A4 30 6B 2C 2B 31 A1 1B B3 1C E8 5F   ŞM´¤¤0k,+1ˇ.ł.č_
000000E7 60 3D 2A E4 BA F6 59 DE FB DA 69 6C D4 C9 55 7C   `=*äşöYŢűÚilÔÉU|
000000F7 80 76 A8 E1 D9 36 39 EF 5D E7 71 59 25 F7 E4 38   €v¨áŮ69ď]çqY%÷ä8
00000107 C0 1C EF BA 87 83 C4 F3 AA EA 1B 99 59 93 F2 9D   Ŕ.ďş‡ƒÄóŞę.™Y“ňť
00000117 8D 03 CB BF 38 17 16 30 6F 6F EF 9A FF 8D 4A 0D   Ť.Ëż8..0ooďš˙ŤJ.
00000127 3A 8E 02 9B 7B 98 B4 53 63 87 1D 38 3E 76 3C 0D   :Ž.›{.´Sc‡.8>v<.
00000137 F4 99 64 28 0B 95 61 BF 79 20 15 9B C0 40 8F 01   ô™d(.•aży .›Ŕ@Ź.
00000147 B9 D4 5B 3E 39 3B 76 3B DD 2D 47 B9 06 B4 11 5B   ąÔ[>9;v;Ý-Gą.´.[
00000157 CC 2C 9D 6B 6F 4C EA 4D 3F 7D D3 69 41 E9 5D FA   Ě,ťkoLęM?}ÓiAé]ú
00000167 92 E7 03 49 E7 D8 F3 0C 76 AD 7D 4D 30 13 A1 99   ’ç.IçŘó.v}M0.ˇ™
00000177 8E 0B 63 26 07 CF E4 46 F3 10 E5 D7 BB 09 F2 17   Ž.c&.ĎäFó.ĺ×».ň.
00000187 66 DB 24 44 31 41 0E A7 F2 36 9B 71 F5 15 7C 2D   fŰ$D1A.§ň6›qő.|-
00000197 0C 81 98 53 DC D6 68 C1 4B 56 89 AE 74 5F 74 53   ...SÜÖhÁKV‰®t_tS
000001A7 F9 05 AE 52 F0 42 32 7B 15 33 A2 D2 3D DA 1A 7C   ů.®RđB2{.3˘Ň=Ú.|
000001B7 0E CD 3C 2C DB 45 72 5B 45 61 CA E0 29 38 42 8B   .Í<,ŰEr[EaĘŕ)8B‹
000001C7 3C B6 EB E4 56 8F 7B 95 3F 28 FB 91 60 F3 8E DA   <¶ëäVŹ{•?(ű‘`óŽÚ
000001D7 45 9D 2D 14 78 78 B5 31 CE B8 1F 59 96 5C 20 D0   Eť-.xxµ1Î¸.Y–\ Đ
000001E7 E8 F4 92 62 3C 5F F4 0B 2C 4E 5D D5 36 96 AE E9   čô’b<_ô.,N]Ő6–®é
000001F7 6D 61 2E 26 B0 AD E4 FF B6 7F 33 A1 1A 96 DA F1   ma.&°ä˙¶.3ˇ.–Úń
00000207 9B 57 42 A3 C1 7D DF A4 78 04 61 AA 17 4F 56 C2   ›WBŁÁ}ß¤x.aŞ.OVÂ
00000217 B9 BB 7A 43 4C 79 3F 71 28 AE 6B 62 99 8E 24 27   ą»zCLy?q(®kb™Ž$'
```

00000000

C:\prj\duqu\lzma\lzma920\Zdc

2013.01.29. 21:01:05   238 601 bytes

```
00000000 5D 00 00 01 00 4E 12 04 00 00 00 00 00 00 52 85   ]....N.......R…
00000010 7C 17 54 60 C3 FC 4D 2D F9 F8 A1 99 8B 11 6B 0D   |.T`ĂüM-ůř˘™‹.k.
00000020 B8 32 B2 10 EB AB 0B 48 30 35 C8 89 83 41 D6 BA   ¸2˛.ë«.H05Č‰ƒAÖş
00000030 57 0D 89 2B C2 AB A6 CF BF AE EE B5 16 00 9A 68   W.‰+Â«¦Ďż®îµ..šh
00000040 5F 7F A6 B9 44 24 C5 E5 7B D7 8B 26 89 48 BA 94   _.¦ąD$ĹĺÍ{×‹&‰Hş”
00000050 50 C6 E4 B4 C4 AF D3 B6 99 51 61 B8 B3 7C 2F 24   PĆä´ÄŻÓ¶™Qa¸ł|/$
00000060 B8 FC D0 0C 1B 2C 35 40 BC 80 2B 11 69 93 D2 27   ¸üĐ..,5@Ľ€+.i“Ň'
00000070 FA E0 2A DE 27 F2 35 92 83 04 8F 5B 69 45 0B 57   úŕ*Ţ'ň5’ƒ.Ź[iE.W
00000080 E4 CB CE 56 AA D9 A8 06 07 6A 12 81 01 E2 B1 28   äËÎVŞŮ¨..j..âą(
00000090 BC 9D 83 D0 3A 36 23 63 12 AC EA 97 1A 3E E9 2A   ĽťƒĐ:6#c.¬ę—.>é*
000000A0 AB EB D3 6A FB 6A 11 FE 47 3C 64 7C A7 50 51 E5   «ëÓjűj.ţG<d|§PQĺ
000000B0 EF F0 A3 85 9D 25 94 7E 03 78 21 67 DB 47 89 7F   ďđŁ…ť%”~.x!gŰG‰.
000000C0 D7 B7 4F 5A 83 D2 BB 9F 39 87 F1 43 F7 B0 C5 08   ×·OZƒŇ»ź9‡ńC÷°Ĺ.
000000D0 0D B9 00 18 C0 2A 1B 8E 8A 97 DD 02 2D C7 90 21   .ą..Ŕ*.ŽŠ—Ý.-Ç.!
000000E0 AA 4D B4 A4 A4 30 6B 2C 2B 31 A1 1B B3 1C E8 5F   ŞM´¤¤0k,+1ˇ.ł.č_
000000F0 60 3D 2A E4 BA F6 59 DE FB DA 69 6C D4 C9 55 7C   `=*äşöYŢűÚilÔÉU|
00000100 80 76 A8 E1 D9 36 39 EF 5D E7 71 59 25 F7 E4 38   €v¨áŮ69ď]çqY%÷ä8
00000110 C0 1C EF BA 87 83 C4 F3 AA EA 1B 99 59 93 F2 9D   Ŕ.ďş‡ƒÄóŞę.™Y“ňť
00000120 8D 03 CB BF 38 17 16 30 6F 6F EF 9A FF 8D 4A 0D   Ť.Ëż8..0ooďš˙ŤJ.
00000130 3A 8E 02 9B 7B 98 B4 53 63 87 1D 38 3E 76 3C 0D   :Ž.›{.´Sc‡.8>v<.
00000140 F4 99 64 28 0B 95 61 BF 79 20 15 9B C0 40 8F 01   ô™d(.•aży .›Ŕ@Ź.
00000150 B9 D4 5B 3E 39 3B 76 3B DD 2D 47 B9 06 B4 11 5B   ąÔ[>9;v;Ý-Gą.´.[
00000160 CC 2C 9D 6B 6F 4C EA 4D 3F 7D D3 69 41 E9 5D FA   Ě,ťkoLęM?}ÓiAé]ú
00000170 92 E7 03 49 E7 D8 F3 0C 76 AD 7D 4D 30 13 A1 99   ’ç.IçŘó.v}M0.ˇ™
00000180 8E 0B 63 26 07 CF E4 46 F3 10 E5 D7 BB 09 F2 17   Ž.c&.ĎäFó.ĺ×».ň.
00000190 66 DB 24 44 31 41 0E A7 F2 36 9B 71 F5 15 7C 2D   fŰ$D1A.§ň6›qő.|-
000001A0 0C 81 98 53 DC D6 68 C1 4B 56 89 AE 74 5F 74 53   ...SÜÖhÁKV‰®t_tS
000001B0 F9 05 AE 52 F0 42 32 7B 15 33 A2 D2 3D DA 1A 7C   ů.®RđB2{.3˘Ň=Ú.|
000001C0 0E CD 3C 2C DB 45 72 5B 45 61 CA E0 29 38 42 8B   .Í<,ŰEr[EaĘŕ)8B‹
000001D0 3C B6 EB E4 56 8F 7B 95 3F 28 FB 91 60 F3 8E DA   <¶ëäVŹ{•?(ű‘`óŽÚ
000001E0 45 9D 2D 14 78 78 B5 31 CE B8 1F 59 96 5C 20 D0   Eť-.xxµ1Î¸.Y–\ Đ
000001F0 E8 F4 92 62 3C 5F F4 0B 2C 4E 5D D5 36 96 AE E9   čô’b<_ô.,N]Ő6–®é
00000200 6D 61 2E 26 B0 AD E4 FF B6 7F 33 A1 1A 96 DA F1   ma.&°ä˙¶.3ˇ.–Úń
00000210 9B 57 42 A3 C1 7D DF A4 78 04 61 AA 17 4F 56 C2   ›WBŁÁ}ß¤x.aŞ.OVÂ
00000220 B9 BB 7A 43 4C 79 3F 71 28 AE 6B 62 99 8E 24 27   ą»zCLy?q(®kb™Ž$'
```

00000005

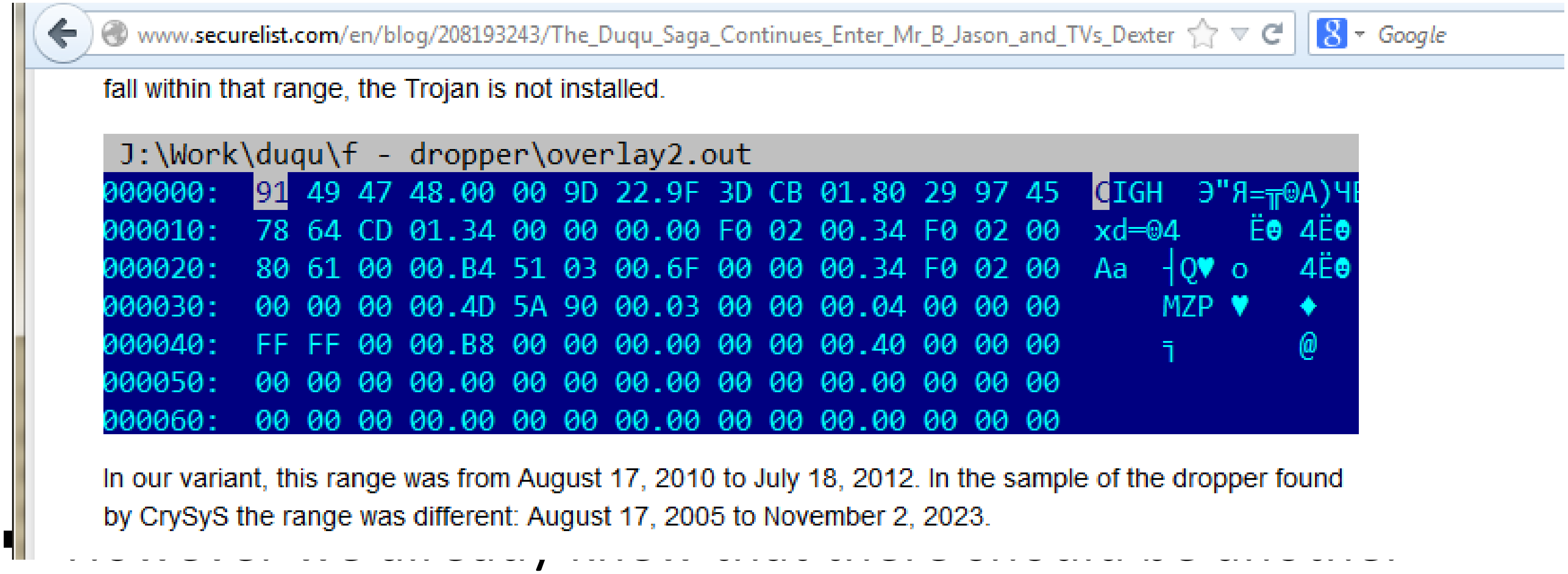
≠ Binary differences   Load time: 0 seconds

Further steps

- We made our own malware DLL with four exports, Duqu will call them
- Replaced User Space Stage 2 code with that
- Recompressed the parts “Kernel space stage 2” – end of file and inserted raw compressed block into dropper
- Re-wrote compressed part header (size of compressed and uncompressed part in 32-bit integers)
- Modified **activation date limits** (not documented)
- All done, ready to test

Dropper time limit

- It was known that User Space stage 2 has some date limit



www.securelist.com/en/blog/208193243/The_Duqu_Saga_Continues_Enter_Mr_B_Jason_and_TVs_Dexter
Google

fall within that range, the Trojan is not installed.

```
 J:\Work\duqu\f - dropper\overlay2.out
000000:  91 49 47 48.00 00 9D 22.9F 3D CB 01.80 29 97 45  CIGH  Э"Я=╥☺A)Ч
000010:  78 64 CD 01.34 00 00 00.00 F0 02 00.34 F0 02 00  xd═☺4     Ё☻ 4Ё☻
000020:  80 61 00 00.B4 51 03 00.6F 00 00 00.34 F0 02 00  Aa  ┤Q♥ o   4Ё☻
000030:  00 00 00 00.4D 5A 90 00.03 00 00 00.04 00 00 00      MZP ♥   ♦
000040:  FF FF 00 00.B8 00 00 00.00 00 00 00.40 00 00 00      ╕       @
000050:  00 00 00 00.00 00 00 00.00 00 00 00.00 00 00 00
000060:  00 00 00 00.00 00 00 00.00 00 00 00.00 00 00 00
```

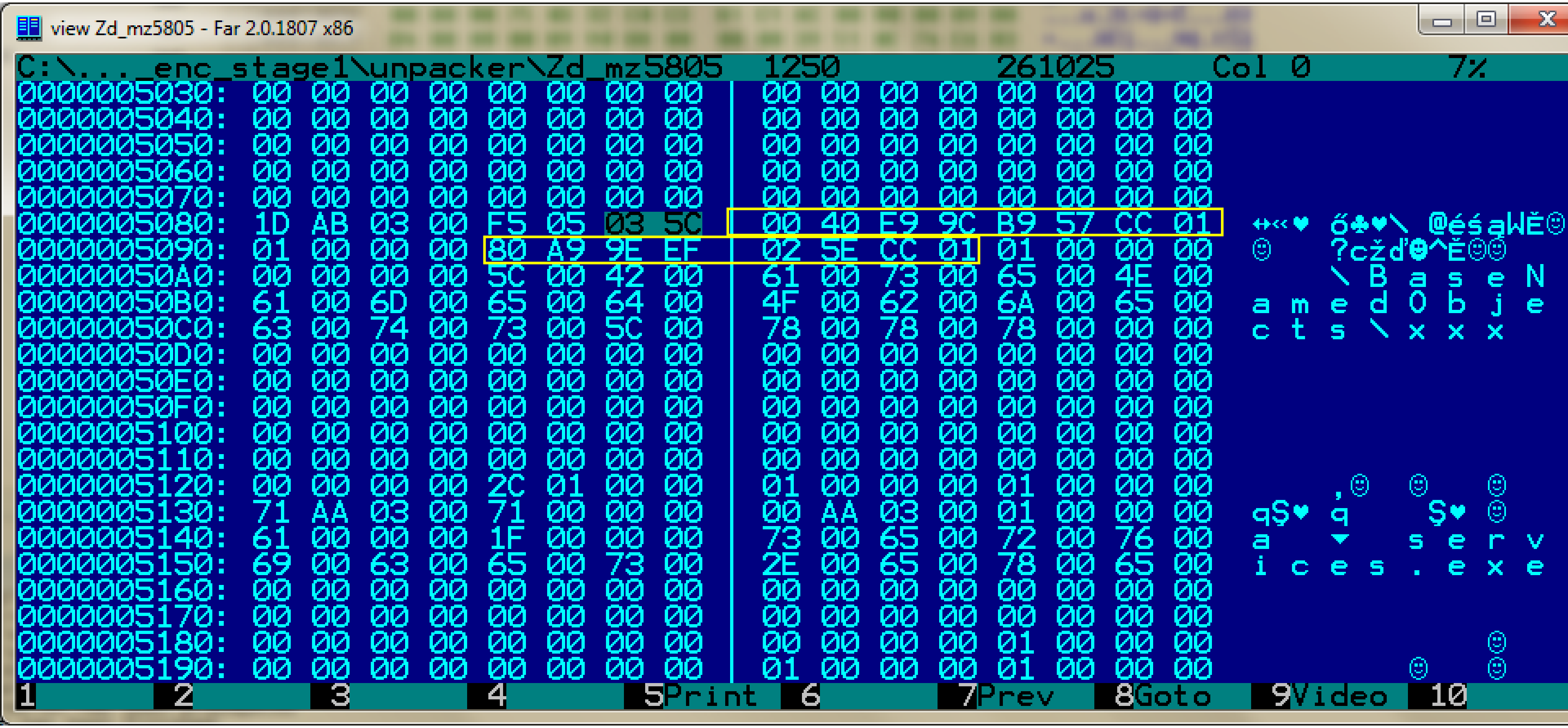
In our variant, this range was from August 17, 2010 to July 18, 2012. In the sample of the dropper found by CrySyS the range was different: August 17, 2005 to November 2, 2023.

- However we already know that there should be another checking as dropper reproduction only worked withing much tight date points

Demo

- Video

User Space Stage 1 time checking



```
C:\..._enc_stage1\unpacker\Zd_mz5805   1250           261025        Col 0        7%
0000005030: 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00
0000005040: 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00
0000005050: 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00
0000005060: 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00
0000005070: 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00
0000005080: 1D AB 03 00 F5 05 03 5C | 00 40 E9 9C B9 57 CC 01  ↔«♥ ő♣♥\ @éśąWĚ☺
0000005090: 01 00 00 00 80 A9 9E EF | 02 5E CC 01 01 00 00 00  ☺   ?čžď☻^Ě☺☺
00000050A0: 00 00 00 00 5C 00 42 00 | 61 00 73 00 65 00 4E 00      \ B a s e N
00000050B0: 61 00 6D 00 65 00 64 00 | 4F 00 62 00 6A 00 65 00  a m e d O b j e
00000050C0: 63 00 74 00 73 00 5C 00 | 78 00 78 00 78 00 00 00  c t s \ x x x
00000050D0: 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00
00000050E0: 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00
00000050F0: 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00
0000005100: 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00
0000005110: 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00
0000005120: 00 00 00 00 2C 01 00 00 | 01 00 00 00 01 00 00 00      ,☺  ☺   ☺
0000005130: 71 AA 03 00 71 00 00 00 | 00 AA 03 00 01 00 00 00  qŞ♥ q    Ş♥ ☺
0000005140: 61 00 00 00 1F 00 00 00 | 73 00 65 00 72 00 76 00  a   ▼   s e r v
0000005150: 69 00 63 00 65 00 73 00 | 2E 00 65 00 78 00 65 00  i c e s . e x e
0000005160: 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00
0000005170: 00 00 00 00 00 00 00 00 | 00 00 00 00 00 00 00 00
0000005180: 00 00 00 00 00 00 00 00 | 00 00 00 00 01 00 00 00              ☺
0000005190: 00 00 00 00 00 00 00 00 | 01 00 00 00 01 00 00 00          ☺   ☺
1       2       3       4       5Print  6       7Prev   8Goto   9Video  10
```

- Time limits: 2011-08-11 Thu Aug 11 02:00:00 to 2011-08-19 Fri Aug 19 01:59:59