Case Study: Deactivating Endpoint Protection Software in an Unauthorized Manner

How to Bypass the Password-Based Authentication for Unloading* Trend Micro OfficeScan *as a Limited User

Dipl.-Inform. Matthias Deeg
Dipl.-Inform. Sebastian Schreiber

SySS GmbH

June 18, 2012

1 Introduction

Endpoint security software like antivirus or firewall software often allows users to disable the offered protection by entering a so-called unload password. Sometimes the protection can only be deactivated temporarily for a few minutes, sometimes it can be deactivated until the protection is manually enabled again or the system is restarted.

In some situations, this feature can be useful for IT support.

But if the password-based authentication is not implemented properly, attackers or malware are able to deactivate the protection in an unauthorized manner without having to know the correct unload password rendering the endpoint protection software useless.

In this case study the SySS GmbH demonstrates the above stated security issue using the example of the antivirus software Trend Micro OfficeScan.

2 Security Analysis

In some configurations the antivirus software Trend Micro OfficeScan offers non-administrative, limited Windows users the possibility to unload the antivirus software with an unload password, as Fig. 1 showing the menu item *Unload OfficeScan* in the corresponding popup menu illustrates.

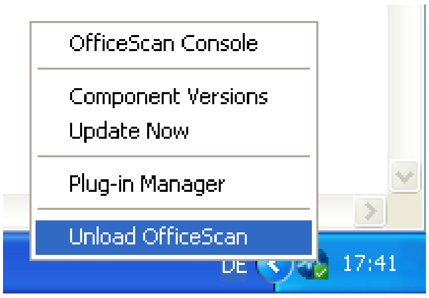
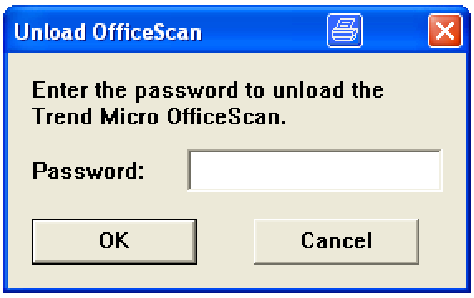


Figure 1: Option to unload Trend Micro OfficeScan

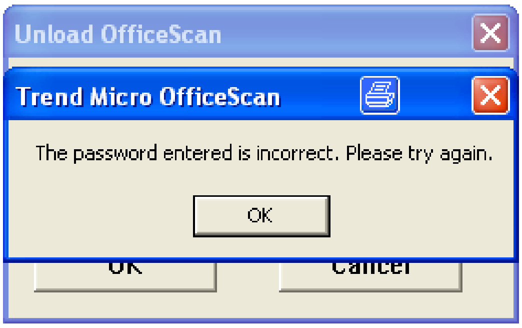
In order to unload Trend Micro OfficeScan, the user has to enter the correct unload password in the dialog window shown in Fig. 2.



Enter the password to unload the Trend Micro OfficeScan.

Figure 2: Password entry in order to deactivate Trend Micro OfficeScan

If an incorrect password is entered, an error message will be shown, as Fig. 3 illustrates.



The password entered is incorrect. Please try again.

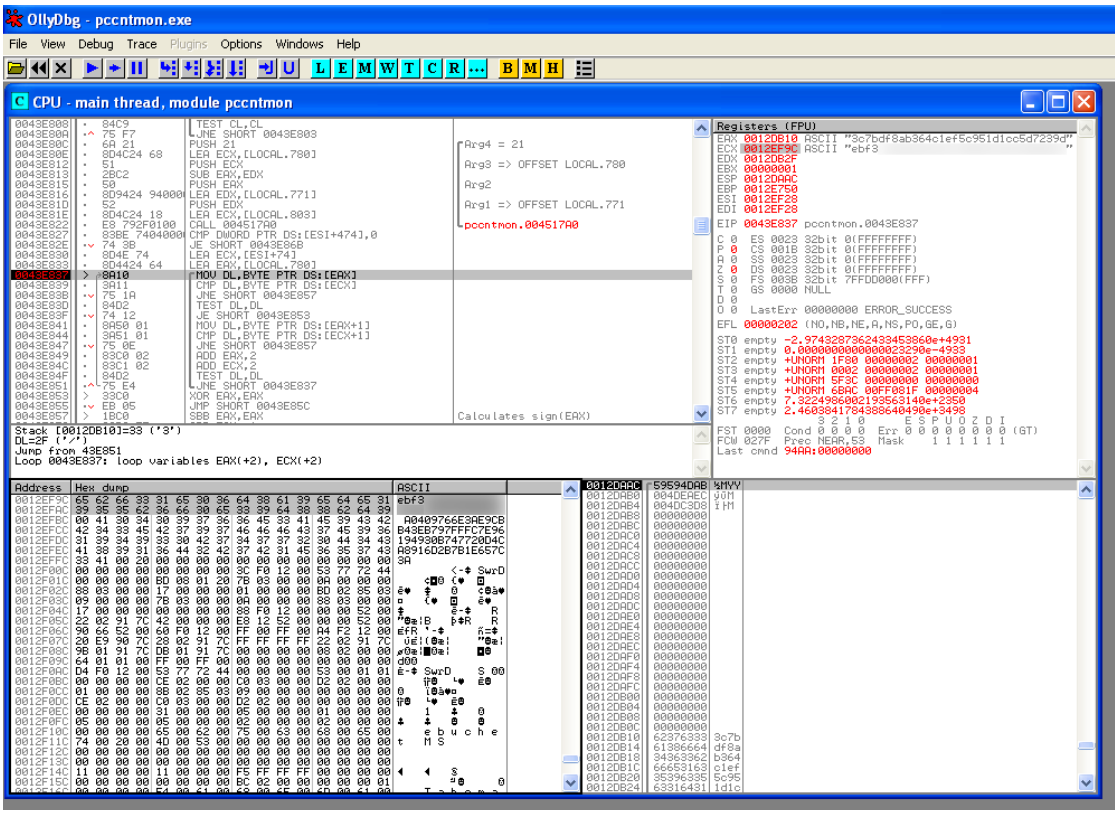
Figure 3: Error message concerning the entry of an incorrect password

By analyzing the password-based authentication for unloading Trend Micro OfficeScan, the SySS GmbH found out, that the password comparison is done within the process `pccntmon.exe`, which runs in the context of the current Windows user, who can also be a standard, limited user.

This fact allows a further analysis and – what is even more interesting – the manipulation of the password comparison during runtime without administrative privileges, as every user is able to debug and manipulate the processes running with his user privileges.

Fig. 4 shows the corresponding code for comparing the MD5 hash of the password entered by the user with the MD5 hash of the correct unload password for a 32 bit software version of TrendMicro OfficeScan within the Windows debugger OllyDbg[1].

[1] `http://www.ollydbg.de/`


OllyDbg - pccntmon.exe
File View Debug Trace Plugins Options Windows Help
L E M W T C R ... B M H
CPU - main thread, module pccntmon

```
0043E808  .   84C9           TEST CL,CL
0043E80A  .^  75 F7          JNE SHORT 0043E803
0043E80C  .   6A 21          PUSH 21                          Arg4 = 21
0043E80E  .   8D4C24 68      LEA ECX,[LOCAL.780]
0043E812  .   51             PUSH ECX                         Arg3 => OFFSET LOCAL.780
0043E813  .   2BC2           SUB EAX,EDX
0043E815  .   50             PUSH EAX                         Arg2
0043E816  .   8D9424 94000(  LEA EDX,[LOCAL.771]
0043E81D  .   52             PUSH EDX                         Arg1 => OFFSET LOCAL.771
0043E81E  .   8D4C24 18      LEA ECX,[LOCAL.803]
0043E822  .   E8 792F0100    CALL 004517A0                    pccntmon.004517A0
0043E827  .   83BE 7404000(  CMP DWORD PTR DS:[ESI+474],0
0043E82E  .v  74 3B          JE SHORT 0043E86B
0043E830  .   8D4E 74        LEA ECX,[ESI+74]
0043E833  .   8D4424 64      LEA EAX,[LOCAL.780]
0043E837  >   8A10           MOV DL,BYTE PTR DS:[EAX]
0043E839  .   3A11           CMP DL,BYTE PTR DS:[ECX]
0043E83B  .v  75 1A          JNE SHORT 0043E857
0043E83D  .   84D2           TEST DL,DL
0043E83F  .v  74 12          JE SHORT 0043E853
0043E841  .   8A50 01        MOV DL,BYTE PTR DS:[EAX+1]
0043E844  .   3A51 01        CMP DL,BYTE PTR DS:[ECX+1]
0043E847  .v  75 0E          JNE SHORT 0043E857
0043E849  .   83C0 02        ADD EAX,2
0043E84C  .   83C1 02        ADD ECX,2
0043E84F  .   84D2           TEST DL,DL
0043E851  .^  75 E4          JNE SHORT 0043E837
0043E853  >   33C0           XOR EAX,EAX
0043E855  .v  EB 05          JMP SHORT 0043E85C
0043E857  >   1BC0           SBB EAX,EAX                      Calculates sign(EAX)
```

```
Stack [0012DB10]=33 ('3')
DL=2F ('/')
Jump from 43E851
Loop 0043E837: loop variables EAX(+2), ECX(+2)
```

```
Registers (FPU)
EAX 0012DB10 ASCII "3c7bdf8ab364c1ef5c951d1cc5d7239d"
ECX 0012EF9C ASCII "ebf3[redacted]"
EDX 0012DB2F
EBX 00000001
ESP 0012DAAC
EBP 0012E750
ESI 0012EF28
EDI 0012EF28

EIP 0043E837 pccntmon.0043E837

C 0  ES 0023 32bit 0(FFFFFFFF)
P 0  CS 001B 32bit 0(FFFFFFFF)
A 0  SS 0023 32bit 0(FFFFFFFF)
Z 0  DS 0023 32bit 0(FFFFFFFF)
S 0  FS 003B 32bit 7FFDD000(FFF)
T 0  GS 0000 NULL
D 0
O 0  LastErr 00000000 ERROR_SUCCESS

EFL 00000202 (NO,NB,NE,A,NS,PO,GE,G)

ST0 empty -2.9743287362433453860e+4931
ST1 empty 0.0000000000000023290e-4933
ST2 empty +UNORM 1F80 00000002 00000001
ST3 empty +UNORM 0002 00000002 00000001
ST4 empty +UNORM 5F3C 00000000 00000000
ST5 empty +UNORM 6BAC 00FF081F 00000004
ST6 empty 7.3224986002193563140e+2350
ST7 empty 2.4603841784388640490e+3498
               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  (GT)
FCW 027F  Prec NEAR,53  Mask    1 1 1 1 1 1
Last cmnd 94AA:00000000
```

```
Address   Hex dump                                         ASCII
0012EF9C  65 62 66 33 31 65 30 36 64 38 61 39 65 64 65 31  ebf3[redacted]
0012EFAC  39 35 35 62 36 66 30 65 33 39 64 38 38 62 64 39
0012EFBC  00 41 30 34 30 39 37 36 36 45 33 41 45 39 43 42   A0409766E3AE9CB
0012EFCC  42 34 33 45 42 37 39 37 46 46 46 43 37 45 39 36  B43EB797FFFC7E96
0012EFDC  31 39 34 39 33 30 42 37 34 37 37 32 30 44 34 43  194930B747720D4C
0012EFEC  41 38 39 31 36 44 32 42 37 42 31 45 36 35 37 43  A8916D2B7B1E657C
0012EFFC  33 41 00 20 00 00 00 00 00 00 00 00 00 00 00 00  3A
0012F00C  00 00 00 00 00 00 00 00 3C F0 12 00 53 77 72 44          <-‡ SwrD
0012F01C  00 00 00 00 BD 08 01 20 7B 03 00 00 0A 00 00 00      ¢◘☺ {♥  ◙
0012F02C  88 03 00 00 17 00 00 00 01 00 00 00 BD 02 85 03  ê♥  ↨   ☺   ¢☻à♥
0012F03C  09 00 00 00 7B 03 00 00 0A 00 00 00 88 03 00 00  ○   {♥  ◙   ê♥
0012F04C  17 00 00 00 00 00 00 00 88 F0 12 00 00 00 52 00  ↨       ê-‡   R
0012F05C  22 02 91 7C 42 00 00 00 E8 12 52 00 00 00 52 00  "☻æ|B   Þ‡R   R
0012F06C  90 66 52 00 60 F0 12 00 FF 00 FF 00 A4 F2 12 00  ÉfR `-‡     ñ=‡
0012F07C  20 E9 90 7C 28 02 91 7C FF FF FF FF 22 02 91 7C   ÚÉ|(☻æ|    "☻æ|
0012F08C  9B 01 91 7C DB 01 91 7C 00 00 00 00 08 02 00 00  ø☺æ|█☺æ|    ◘☻
0012F09C  64 01 01 00 FF 00 FF 00 00 00 00 00 00 00 00 00  d☺☺
0012F0AC  D4 F0 12 00 53 77 72 44 00 00 00 00 53 00 01 01  È-‡ SwrD    S ☺☺
0012F0BC  00 00 00 00 CE 02 00 00 C0 03 00 00 D2 02 00 00      ╬☻  └♥  Ê☻
0012F0CC  01 00 00 00 8B 02 85 03 09 00 00 00 00 00 00 00  ☺   ï☻à♥○
0012F0DC  CE 02 00 00 C0 03 00 00 D2 02 00 00 00 00 00 00  ╬☻  └♥  Ê☻
0012F0EC  00 00 00 00 31 00 00 00 05 00 00 00 01 00 00 00      1   ♣   ☺
0012F0FC  05 00 00 00 05 00 00 00 02 00 00 00 02 00 00 00  ♣   ♣   ☻   ☻
0012F10C  00 00 00 00 65 00 62 00 75 00 63 00 68 00 65 00      e b u c h e
0012F11C  74 00 20 00 4D 00 53 00 00 00 00 00 00 00 00 00  t   M S
0012F12C  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0012F13C  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0012F14C  11 00 00 00 11 00 00 00 F5 FF FF FF 00 00 00 00  ◄   ◄   §
0012F15C  00 00 00 00 00 00 00 00 BC 02 00 00 00 00 00 01          ╝☻     ☺
```

```
0012DAAC  59594DAB  ½MYY
0012DAB0  004DEAEC  ÿÛM
0012DAB4  004DC3D8  Ï├M
0012DAB8  00000000
0012DABC  00000000
0012DAC0  00000000
0012DAC4  00000000
0012DAC8  00000000
0012DACC  00000000
0012DAD0  00000000
0012DAD4  00000000
0012DAD8  00000000
0012DADC  00000000
0012DAE0  00000000
0012DAE4  00000000
0012DAE8  00000000
0012DAEC  00000000
0012DAF0  00000000
0012DAF4  00000000
0012DAF8  00000000
0012DAFC  00000000
0012DB00  00000000
0012DB04  00000000
0012DB08  00000000
0012DB0C  00000000
0012DB10  62376333  3c7b
0012DB14  61386664  df8a
0012DB18  34363362  b364
0012DB1C  66653163  c1ef
0012DB20  35396335  5c95
0012DB24  63316431  1d1c
```

Figure 4: Comparison of MD5 hashes

In order to bypass this password-based authentication, an attacker only has to patch this password comparison, so that it always returns true, for example by comparing the correct unload password with itself.

It is also possible to extract the MD5 hash of the correct unload password and try to recover it using a password guessing attack, for example by using a password recovery tool like JOHN THE RIPPER[2].

The process of unloading TREND MICRO OFFICESCAN can also be automated, so that this security issue can also be exploited by malware.

[2] `http://www.openwall.com/john/`

3 Recommendation

The SySS GmbH recommends to perform the password comparison not within the process `pccntmon.exe`, which is running with the privileges of the current user, but within a service process of Trend Micro OfficeScan running with higher privileges, for example `SYSTEM`.

In this case, a limited Windows user or malware running in the context of such a user is not able to manipulate the password comparison of Trend Micro OfficeScan during runtime in order to unload the antivirus software in an unauthorized manner.

The SySS GmbH informed Trend Micro about the stated security vulnerability on April 26, 2012.

According to information of Trend Micro, the security issue allowing to unload Trend Micro OfficeScan in an unauthorized manner has been fixed with the release of the latest service pack for Trend Micro OfficeScan.