Privilege Escalation via Client Management Software

November 21, 2015

BSidesVienna 0x7DF

Who am I?

Dipl.-Inf. Matthias Deeg

Expert IT Security Consultant

CISSP, CISA, OSCP, OSCE

- Interested in information technology – especially IT security – since his early days
- Studied computer science at the University of Ulm, Germany
- IT Security Consultant since 2007

Agenda

1. Client Management Software
2. Common Security Vulnerabilities
3. Use Cases & Attack Scenarios
4. Demo
5. Conclusion & Recommendations
6. Q&A

Client Management Software

- Client management is a very important task in modern enterprise IT environments as all computer systems, whether client or server, should be managed throughout their entire system life cycle.
- There are many client management software solutions from different manufacturers that support IT managers and IT administrators in client management tasks like
  - inventory
  - patch management
  - software deployment
  - license management

Client Management Software

- As a matter of principle, in order to perform these tasks, client management software requires high privileges, usually administrative rights, on the managed client and server systems.
- Therefore, client management software is an interesting target for attackers as vulnerabilities in this kind of software may be leveraged for privilege escalation attacks within corporate networks.

Common Security Vulnerabilities

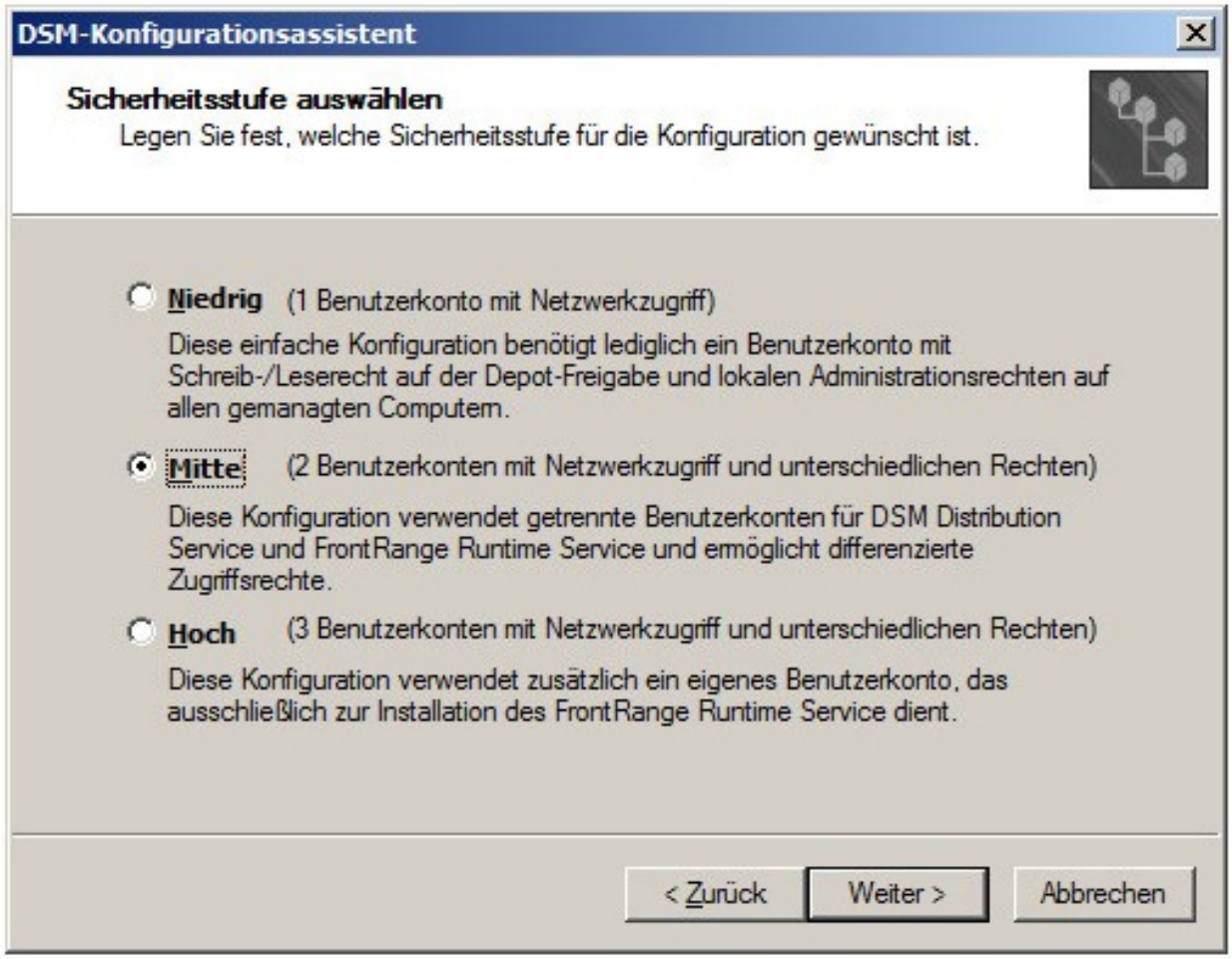
- During security assessments of client systems managed with different client management software solutions, the SySS GmbH could find the following common security vulnerabilities:
  1. Insufficiently Protected Credentials (CWE-522)
  2. Use of Hard-coded Cryptographic Key (CWE-321)
  3. Violation of Secure Design Principles (CWE-657)

Insufficiently Protected Credentials

- In order to perform different management tasks, client management software products usually require one or more user/service account and access to the corresponding credentials.
- If a low-privileged user has access to password information that are not required to perform her tasks, it is usually a security issue.
- Furthermore, if the accessible credentials are only protected in an insufficient way, it definitely is a security issue.
- In case of the tested client management software products, password information was in some cases accessible by low-privileged users and insufficiently protected

⇒ Unauthorized access to credentials of a foreign user account allowing impersonation and privilege escalation attacks

Example: FrontRange DSM



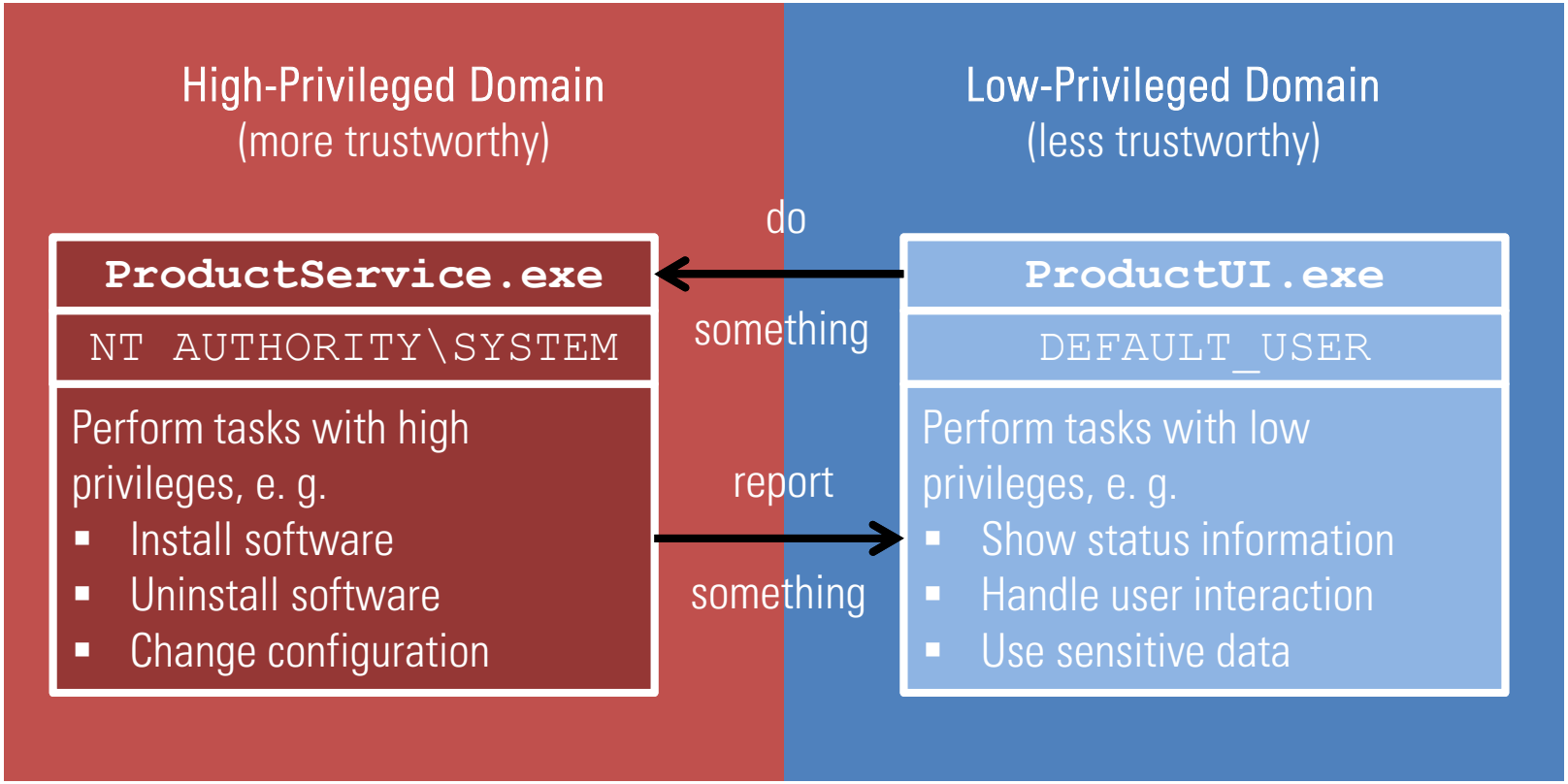
Use of Hard-coded Cryptographic Key

- Different client management software products use hard-coded cryptographic keys in order to protect sensitive data, for example
  - User credentials (usually username and password)
  - Configuration data
- In general, the used hard-coded keys are valid for all software installations (i.e. not system- or customer-dependent)
- If an attacker knows how user credentials are protected (encryption algorithm and cryptographic key) and has access to the password data, she can always recover the clear-text passwords.

⇒ Unauthorized access to credentials of a foreign user account allowing impersonation and privilege escalation attacks

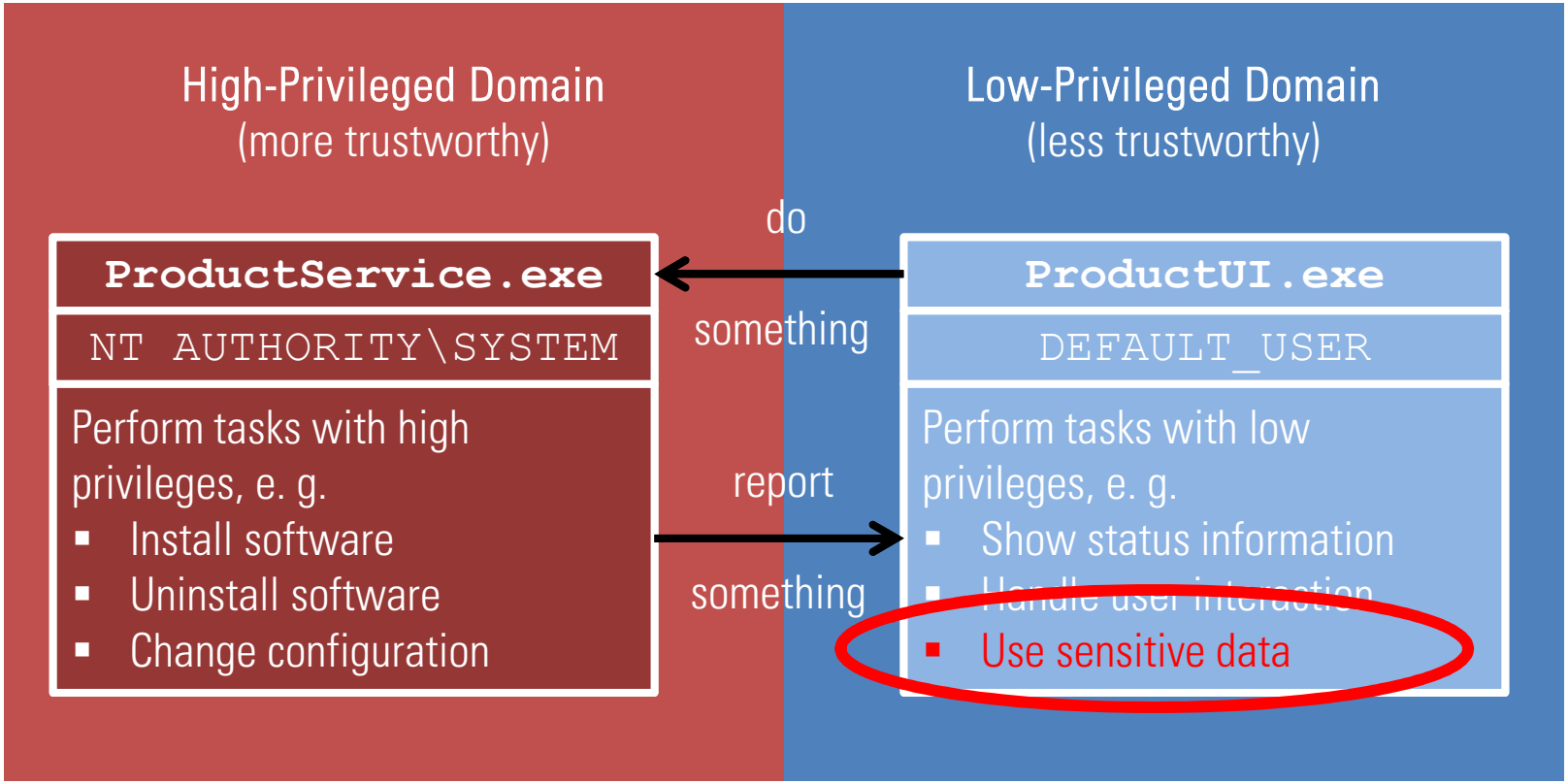
Violation of Secure Design Principles

What is the problem?



Violation of Secure Design Principles

What is the problem?



Violation of Secure Design Principles

- Password information is used (encoded and/or encrypted) in the context of a low-privileged user process.
- Thus, an attacker or malware running in the same low-privileged user context can analyze and control the process and in this way gain access to decrypted clear-text passwords.

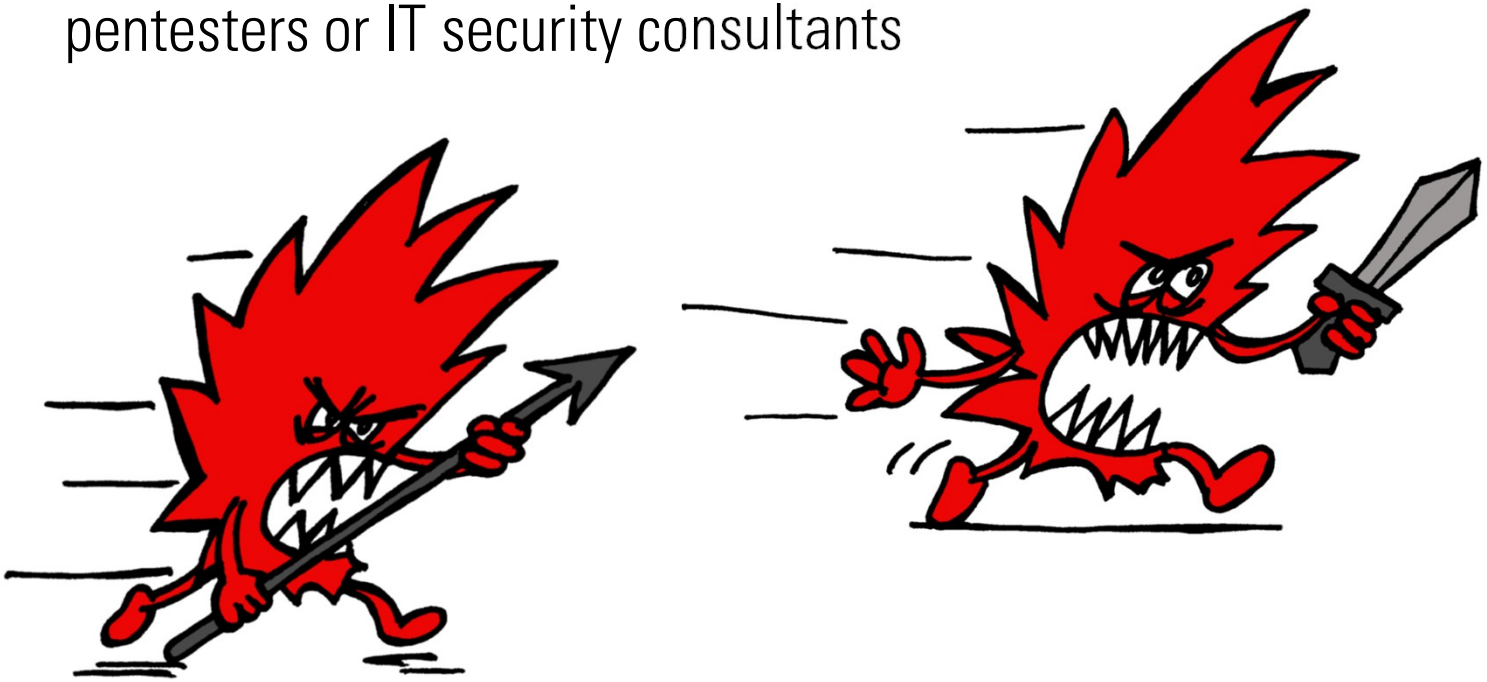
⇒ Unauthorized access to credentials of a foreign user account allowing impersonation and privilege escalation attacks

Use Cases & Attack Scenarios

SySS THE PENTEST EXPERTS.

Use Cases:

1. Bad guys doing bad things for fun and profit
2. Good guys doing bad things with permission for fun and profit, e. g. pentesters or IT security consultants



Use Cases & Attack Scenarios

THE PENTEST EXPERTS.

Attack Scenario: Owning a Windows Domain Network in 5 (Easy) Steps

1. Gain access to a managed system (as a low-privileged user).
2. Choose your attack:
   a. **Offline:** Read the encrypted user credentials of the client management software stored on the system and decrypt them using a suitable software tool.
   b. **Online:** Extract the clear-text user credentials from a process of the client management software running in the low-privileged user context.
3. Use the recovered credentials to gain unauthorized administrative access to other managed systems within the corporate network (e. g. client systems, client management software server, file server, print server, application server).
4. Search for authentication data (e. g. passwords, NTLM hashes, Windows access tokens) of high-privileged Windows domain users on the accessible systems.
5. Own the Windows domain.

Affected Client Management Software

Product Name	Tested Software Version
Altiris Inventory Solution	7.1.7580.0
Empirum	14.2.1, 15.0.1, 16.0
FrontRange DSM	7.2.1.2020, 7.2.2.2331

PoC Software Tools

- The SySS GmbH developed proof-of-concept software tools for different client management software products in order to recover cleartext-passwords:
  - `Altiris Password Decryptor`
  - `Empirum Password Decryptor`
  - `FrontRange DSM Password Decryptor`

Demo

Demo: FrontRange DSM

- FrontRange DSM stores passwords for different user accounts encrypted in two configuration files named `NiCfgLcl.ncp` and `NiCfgSrv.ncp`.
- These configuration files contain encrypted password information for different required FrontRange DSM user accounts, e. g.
  - DSM Runtime Service
  - DSM Distribution Service
  - Business Logic Server (BLS) Authentication
  - Database account

Demo: FrontRange DSM

- A limited Windows domain user has read access to these configuration files that are usually stored in the following locations:
  - `%PROGRAMFILES(X86)\NetInst\NiCfgLcl.ncp`
    (local on a managed client)
  - `%PROGRAMFILES(X86)\NetInst\NiCfgSrv.ncp`
    (local on a managed client)
  - `\\<FRONTRANGE SERVER>\DSM$\NiCfgLcl.ncp`
    (remote on a DSM network share)
  - `\\<FRONTRANGE SERVER>\DSM$\NiCfgSrv.ncp`
    (remote on a DSM network share)

Demo: FrontRange DSM

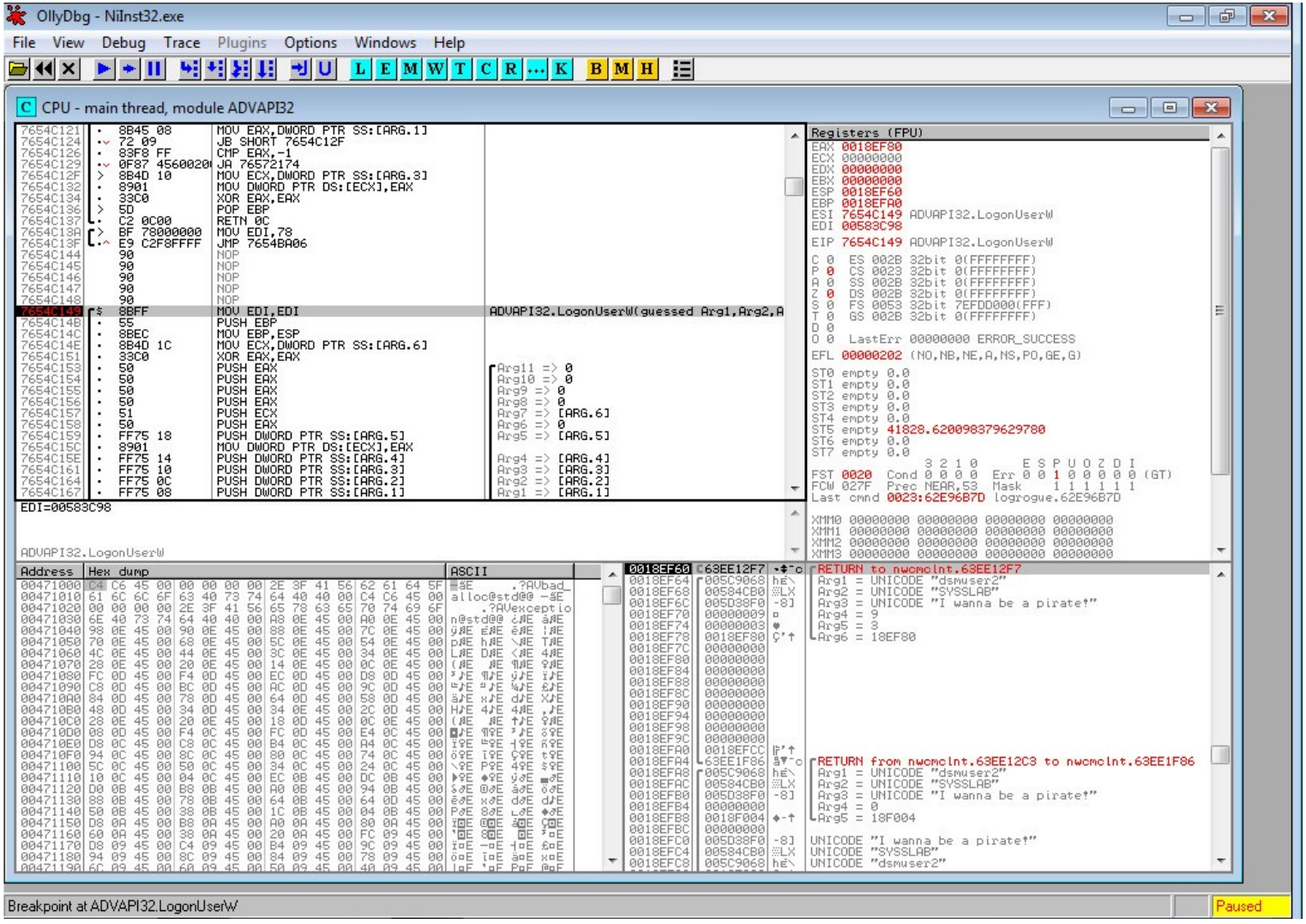
- The SySS GmbH developed a proof-of-concept software tool named `FrontRange DSM Password Decryptor` which is able to decrypt all password information stored within the FrontRange configuration files `NiCfgLcl.ncp` and `NiCfgSrv.ncp`.
- This software tool can be used for offline password recovery.

```
>fpd.exe k22D01816EADA56F850G09218CCD5GC1C4537FC70768629C14FF5B
FrontRange DSM Password Decryptor v1.0 by Matthias Deeg
<matthias.deeg@syss.de> - SySS GmbH (c) 2014
[+] Decrypted password: I wanna be a pirate!
```

Demo: FrontRange DSM

- It is also possible to perform an online attack targeting the running process `NiInst32.exe` using an application-level debugger like OllyDbg from the perspective of a low-privileged Windows user.
- In order to gain access to the decrypted password, it is sufficient to set a breakpoint on the Windows API function `LogonUserW` of the module `ADVAPI32.DLL`.

Demo: FrontRange DSM



Demo: FrontRange DSM

- FrontRange DSM user credentials are used when the Windows API function `LogonUserW` is called within the process `NiInst32.exe`.

```
7654C145      90          NOP
7654C146      90          NOP
7654C147      90          NOP
7654C148      90          NOP
7654C149  $   8BFF        MOV EDI,EDI                    ADVAPI32.LogonUserW(guessed Arg1,Arg2,A
7654C14B  .   55          PUSH EBP
7654C14C  .   8BEC        MOV EBP,ESP
7654C14E  .   8B4D 1C     MOV ECX,DWORD PTR SS:[ARG.6]
7654C151  .   33C0        XOR EAX,EAX
```

```
0018EF60  63EE12F7  ▪↕¯c  RETURN to nwcmclnt.63EE12F7
0018EF64  005C9068  hÉ\   Arg1 = UNICODE "dsmuser2"
0018EF68  00584CB0  ▒LX   Arg2 = UNICODE "SYSSLAB"
0018EF6C  005D38F0  -8]   Arg3 = UNICODE "I wanna be a pirate!"
0018EF70  00000009  ▫      Arg4 = 9
0018EF74  00000003  ♥      Arg5 = 3
0018EF78  0018EF80  Ç'↑   Arg6 = 18EF80
```

Demo: Empirum

- Empirum supports the following four password formats for storing password information in an encrypted way in different configuration files or in the Windows registry:

1. SETUP

   example: *SKZjk`&gp2

2. SYNC

   example:
   12B65B9A30D4237D0A5F8D50341581B64207CE74CDE2ED7632D8D55EDE775EF4
   A71631812F2E4E39BD951E26991F307F

3. EIS

   example: A"z!' |-%-*),$ "!&(xiYJ|+./'(=&)+#$,#%./*X

4. MD5

   example: 8a24367a1f46c141048752f2d5bbd14b

Demo: Empirum

- The Empirum SETUP, SYNC, and EIS password formats use reversible encryption methods and can be created by a software tool called `EmpCrypt.exe`.
- Usually, only Empirum administrators have access to this software tool and it is not installed on managed systems.
- But Empirum software components like Empirum Inventory and its modules (for example `EmpInventory.exe`, `ShowInventory.exe`) that are installed on managed systems contain the functionality for decrypting these Empirum password formats.
- The used MD5 passwords are simply unsalted raw MD5 hashes.

Demo: Empirum

- An analysis of the used encryption methods showed, that three different encryption algorithms are used, each with its own hard-coded secret (for example a cryptographic key or permutation table).
- Configuration files containing encrypted password information are either located on the managed system itself, for example in the configuration file `AgentConfig.xml`, or in INI files stored on network shares of Empirum servers.
- A limited Windows domain user has read access to the locally stored XML configuration file and to the INI configuration files that are typically stored in the following locations:
  - `\\<EMPIRUM SERVER>\Configurator$`
  - `\\<EMPIRUM SERVER>\Values$`

Demo: Empirum

```
$ ./epd '*SKZjk`&gp2'

____ ___  ___
|___ |__] |  \
|___ |    |__/
Empirum Password Decryptor v2.0 by Matthias Deeg - SySS GmbH (c) 2009-2015
[*] Read Empirum SETUP password
[+] The decrypted password is: P@ssw0rd!

$ ./epd 12B65B9A30D4237D0A5F8D50341581B64207CE74CDE2ED7632D8D55EDE775EF4A71631812F2E4E39BD951E26991F307F

____ ___  ___
|___ |__] |  \
|___ |    |__/
Empirum Password Decryptor v2.0 by Matthias Deeg - SySS GmbH (c) 2009-2015
[*] Read Empirum SYNC password
[+] The decrypted password is: P@ssw0rd!

E:\>epd.exe "A\"z!' ^|-%-*),$ \"!&(xiYJ|+./'(=&)+#$,#%./*X"

____ ___  ___
|___ |__] |  \
|___ |    |__/
Empirum Password Decryptor v2.0 by Matthias Deeg - SySS GmbH (c) 2009-2015
[*] Read Empirum EIS password
[+] The decrypted password is: P@ssw0rd!
```

Demo: Empirum

- The process `EmpInventory.exe`, that is executed in the context of a low-privileged user, decrypts and uses user credentials contained in the Empirum configuration files.
- Thus, an attacker or malware running in the same low-privileged user context can analyze and control the process `EmpInventory.exe` and in this way gain access to decrypted clear-text passwords.
- Such an online attack targeting the running process `EmpInventory.exe` can be performed using an application-level debugger like OllyDbg from the perspective of a limited Windows user.

Demo: Altiris Inventory Solution

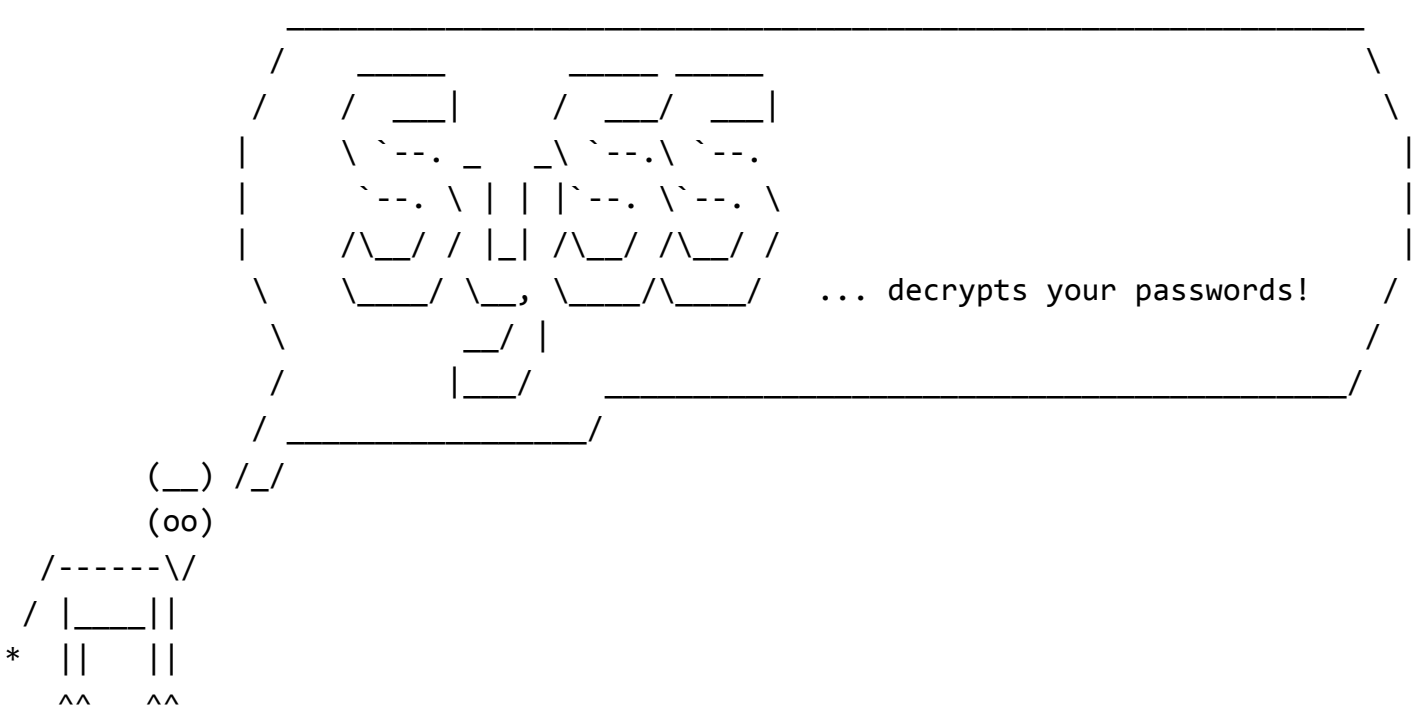
- In some configurations, Altiris Inventory Solution stores user credentials (package access user and password) in the Windows registry using the following registry keys:
  - `HKEY_LOCAL_MACHINE\SOFTWARE\Altiris\Communications\Package Access Password`
  - `HKEY_LOCAL_MACHINE\SOFTWARE\Altiris\Communications\Package Access User`
- In the default configuration, administrative privileges are required in order to read these registry keys (⇒ not useful for local privilege escalation).
- The user name (package access user) is stored as plaintext and the password (package access password) is stored as ciphertext.

Demo: Altiris Inventory Solution

- The password is encrypted using a hard-coded cryptographic key and initialization vector (IV) via the symmetric-key block cipher Triple DES (3DES) in CBC mode.
- The hard-coded cryptographic key and IV are contained within the two dynamic link libraries `InvAgent.dll` and `AeXNetComms.dll` of the Altiris Inventory Solution software.
- The use of a hard-coded cryptographic key and IV enables an attacker with access to the encrypted password information to recover the clear-text password for all affected installations of Altiris Inventory Solution software.

Demo: Altiris Inventory Solution

```
$ ./altirispd.py vZW7Vflp5qwh2k4dfVQmFcaHEIcwkvuO
```

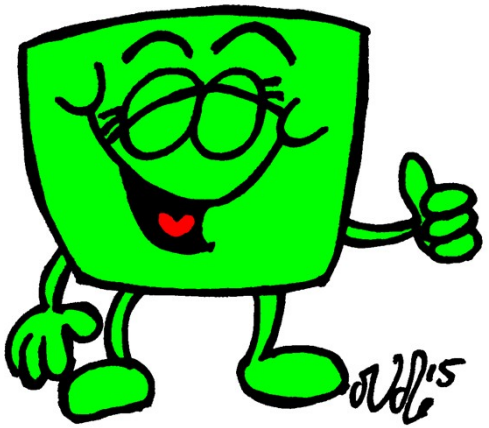



```
Altiris Password Decryptor v1.0 by Matthias Deeg <matthias.deeg@syss.de> - SySS GmbH (c) 2013
[*] The plaintext password is: P4ssw0rd!
```

Conclusion

- Security vulnerabilities in different client management software products can be leveraged in attacks against corporate networks.
- Generally, the access to password information, no matter whether encrypted or not, should be restricted as much as possible.
- Configuration files that are readable by low-privileged users are not the proper place to store sensitive password information, and low-privileged user processes are not the proper place to use them.
- Security-related tasks should be performed in a (more) trustworthy environment.

Recommendations



- Always consider trust in IT security:
  - Trust domains
  - Trust boundaries
  - Trust relationships
- Do not assume *too much*™
- Properly protect password information by restricting access to password information to required users and processes only
- Follow the principle of least privilege
- Update affected software products
- Configure used client management software products according to provided best practices and security guidelines by the manufacturer.

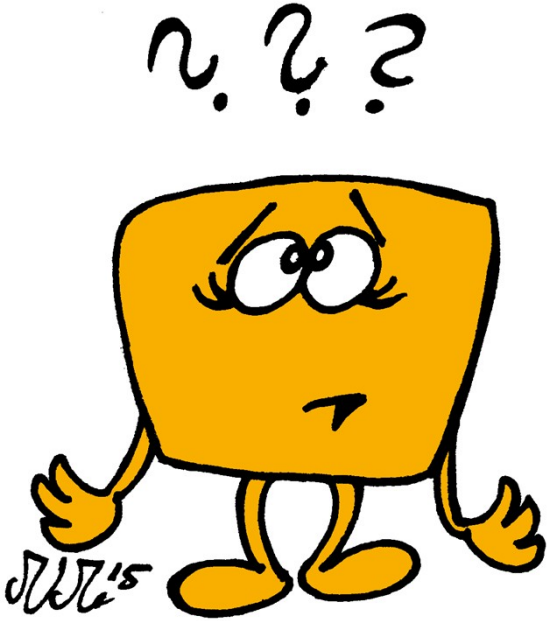
References

- *SySS Security Advisory SYSS-2014-007*, Matthias Deeg, *https://www.syss.de/fileadmin/dokumente/Publikationen/Advisories/SYSS-2014-007.txt*, 2015
- *Privilege Escalation via Client Management Software – Part I*, Matthias Deeg, https://www.syss.de/fileadmin/dokumente/Publikationen/2015/Privilege_Escalation_via_Client_Management_Software.pdf, 2015
- *Privilege Escalation via Client Management Software – Part II*, Matthias Deeg, https://www.syss.de/fileadmin/dokumente/Publikationen/2015/Privilege_Escalation_via_Client_Management_Software__Part_II.pdf, 2015

Thank you very much ...

… for your attention.

Do you have any questions?



E-mail: matthias.deeg@syss.de

PGP Fingerprint: D1F0 A035 F06C E675 CDB9 0514 D9A4 BF6A 34AD 4DAB

THE PENTEST EXPERTS

WWW.SYSS.DE