Ethical Hacker In a Big4 Firm

What society thinks I do

What my colleagues think I do

What my mom thinks I do

What I think I do

What I actually do

www.pwc.lu

Hack.lu 2017

Malicious use of Microsoft LAPS: "Local Administrator Password Solution"

October 2017

Maxime Clementz
Antoine Goichot

pwc

Maxime Clementz

🐦@maxime_tz

Antoine Goichot

🐦@AntoineGoichot

5 years at PwC

We are **Ethical Hackers** at **PwC Luxembourg**

Current activities (& hobbies? \o/)

Ethical hacking & Penetration tests
Vulnerability assessment and research
IT & Information Security
Computer Forensics

Education

TELECOM Nancy (France)

Previous publications

2012 & 2015 Hack.lu ☺
2015 IEEE Symposium on Integrated Network and Service Management
2015 LORIA-INRIA Security Seminar
2011 Network Management Research Group

2 years at PwC

PwC Luxembourg: Cyber Security Advisory

Penetration tests & vulnerability assessments

FTS investigation & eDiscovery

Threat & Vulnerability management

Cybersecurity governance & Architecture assessments

Risks management & Third party assessments

Cybersecurity Awareness

Agenda

Introducing LAPS		**6**
1	The Local Admin Problem	7
2	State of the art	9
3	How does LAPS work?	12
Analysis		**15**
4	Preliminary result	16
5	Additional research	18
Exploitation		**19**
6	Objectives of our PoC	20
7	Demos	22
Recommendations & Conclusion		**26**
8	Recommendations	27
9	Conclusion	31

Disclaimer

- **No** new CVE, `0day` exploit, hardcore RE, but ways to abuse Microsoft LAPS
- Local admin privileges are needed to use this **stealthy persistence** tactic
- Escalation of Privileges possible **under favourable conditions** ☺
- **Simple** but effective approach

Introducing LAPS

The Local Admin Problem

And how does LAPS solve it

Identical/guessable local admin password
⇒ trivial lateral movement

“[LAPS] mitigates the risk of lateral escalation that results when customers use the same administrative local account and password combination on their computers.”

https://technet.microsoft.com/en-us/library/security/3062591

(Local Administrator Password Solution (LAPS) Now Available, May 2015)

The Local Admin Problem

And how does LAPS solve it

Identical/guessable local admin password
⇒ trivial lateral movement

"[LAPS] mitigates the risk of lateral escalation that results when customers use the same administrative local account and password combination on their computers."

https://technet.microsoft.com/en-us/library/security/3062591

(Local Administrator Password Solution (LAPS) Now Available, May 2015)

➔ LAPS sets a ***different, random*** password for the built-in ***local administrator*** account on ***every managed computer*** in the domain, and automatically change them in compliance with the ***policy for characters*** and ***validity period***.

State of the art

And motivation

Several great blog posts (and tools) that describe and exploit **server side issues/operations**:

- Sean Metcalf (@PyroTek3) – adsecurity.org/?p=1790 & adsecurity.org/?p=3164
- Will Schroeder (@harmj0y) – www.harmj0y.net/blog/powershell/running-laps-with-powerview/
- Karl Fosaaen (@kfosaaen) – blog.netspi.com/running-laps-around-cleartext-passwords/
- Andy Robbins (@_wald0) and Will Schroeder (@harmj0y) – www.blackhat.com/us-17/briefings/schedule/index.html#an-ace-up-the-sleeve-designing-active-directory-dacl-backdoors-6223

State of the art

And motivation

Several great blog posts (and tools) that describe and exploit **server side issues/operations**:

- Sean Metcalf (@PyroTek3) – adsecurity.org/?p=1790 & adsecurity.org/?p=3164
- Will Schroeder (@harmj0y) – www.harmj0y.net/blog/powershell/running-laps-with-powerview/
- Karl Fosaaen (@kfosaaen) – blog.netspi.com/running-laps-around-cleartext-passwords/
- Andy Robbins (@_wald0) and Will Schroeder (@harmj0y) – www.blackhat.com/us-17/briefings/schedule/index.html#an-ace-up-the-sleeve-designing-active-directory-dacl-backdoors-6223

LAPS is well documented *(RTFM!)*:

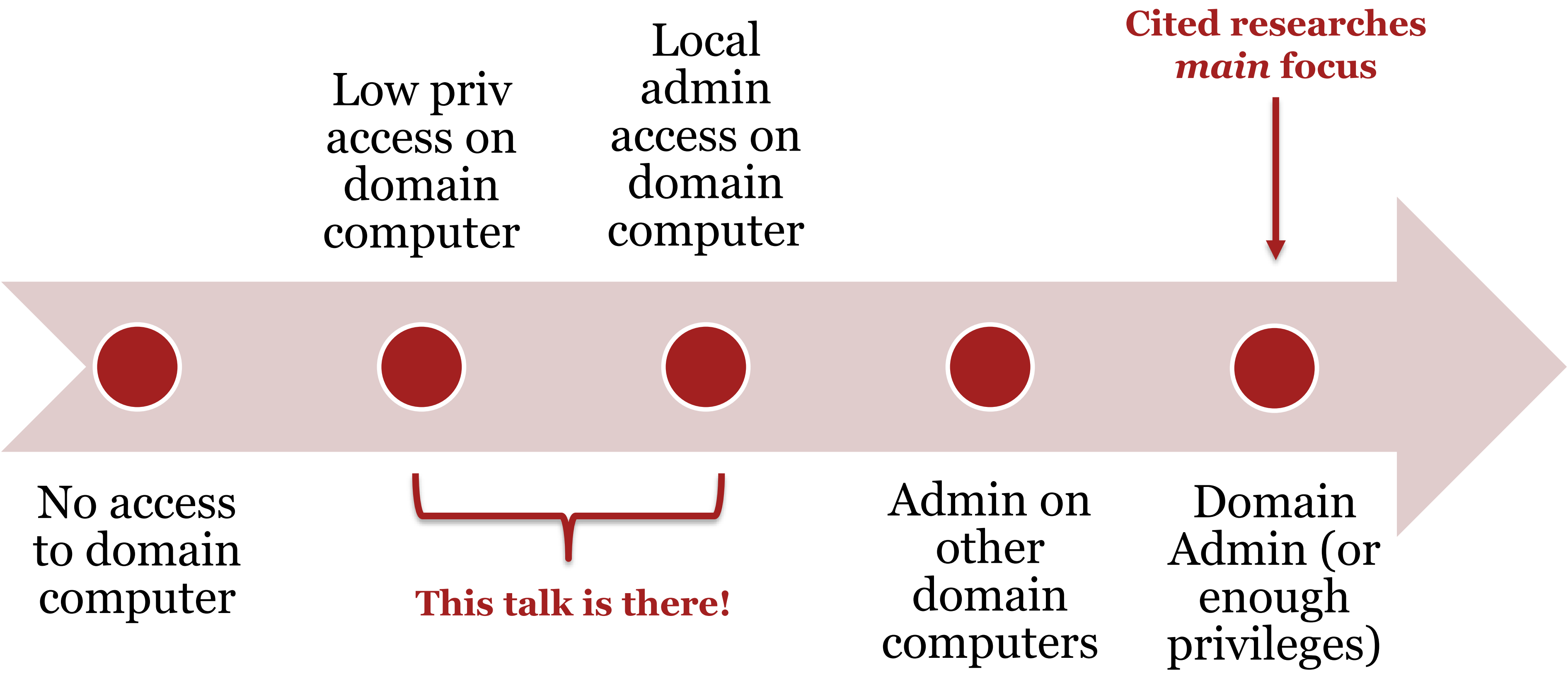
> "Solution has a ***client side component*** – Group Policy Client Side Extension (CSE) - that automatically performs ***all*** tasks related to maintenance of the password of local Administrator account."

➔ All critical tasks are done client side

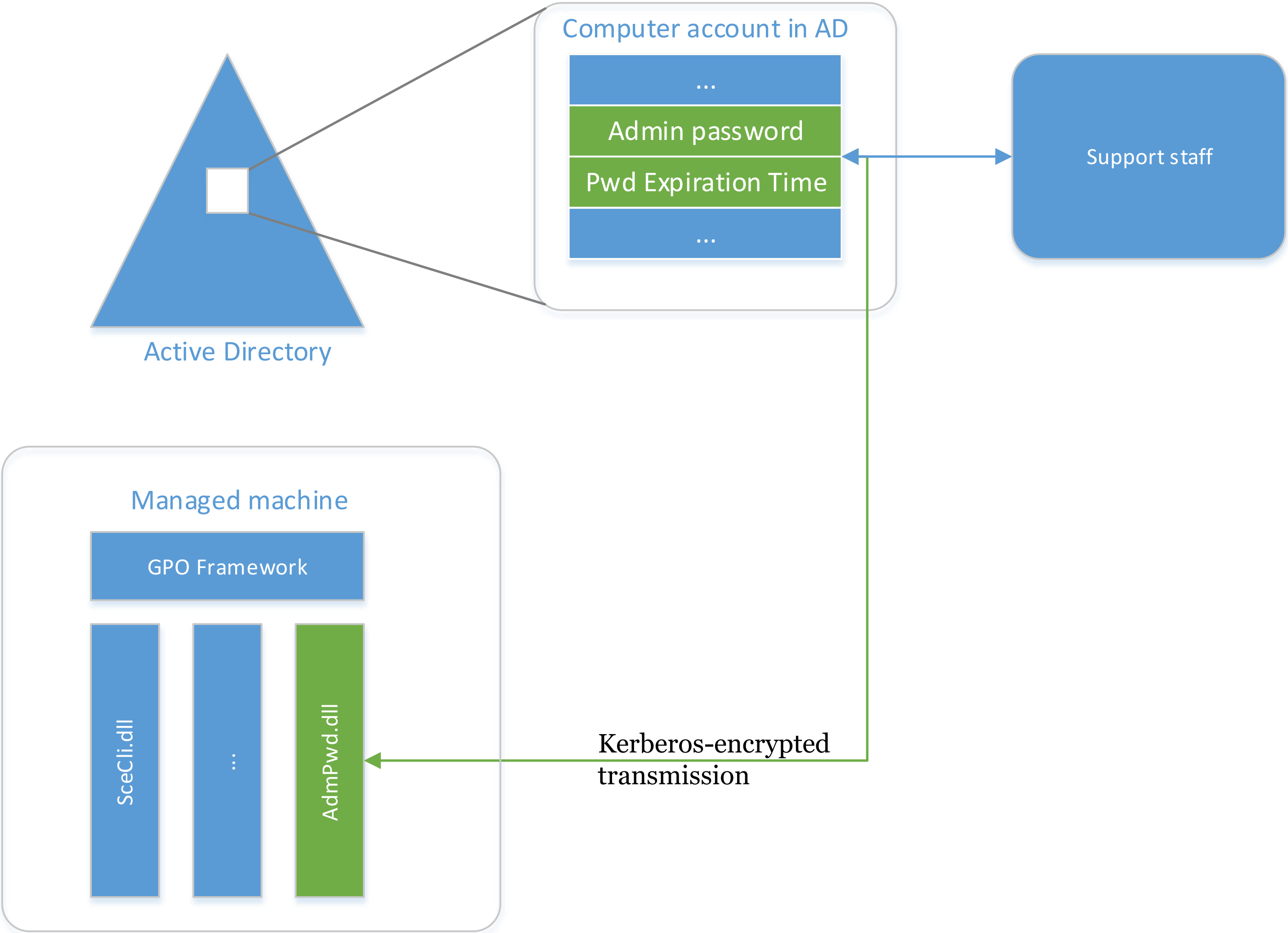
➔ Above cited researches/attacks focused ***on the server side*** (i.e. looking for accounts who can read the passwords) and ***not on the client side*** (systems in the domain)

State of the art

And motivation


Low priv access on domain computer
Local admin access on domain computer
Cited researches *main focus*
No access to domain computer
This talk is there!
Admin on other domain computers
Domain Admin (or enough privileges)

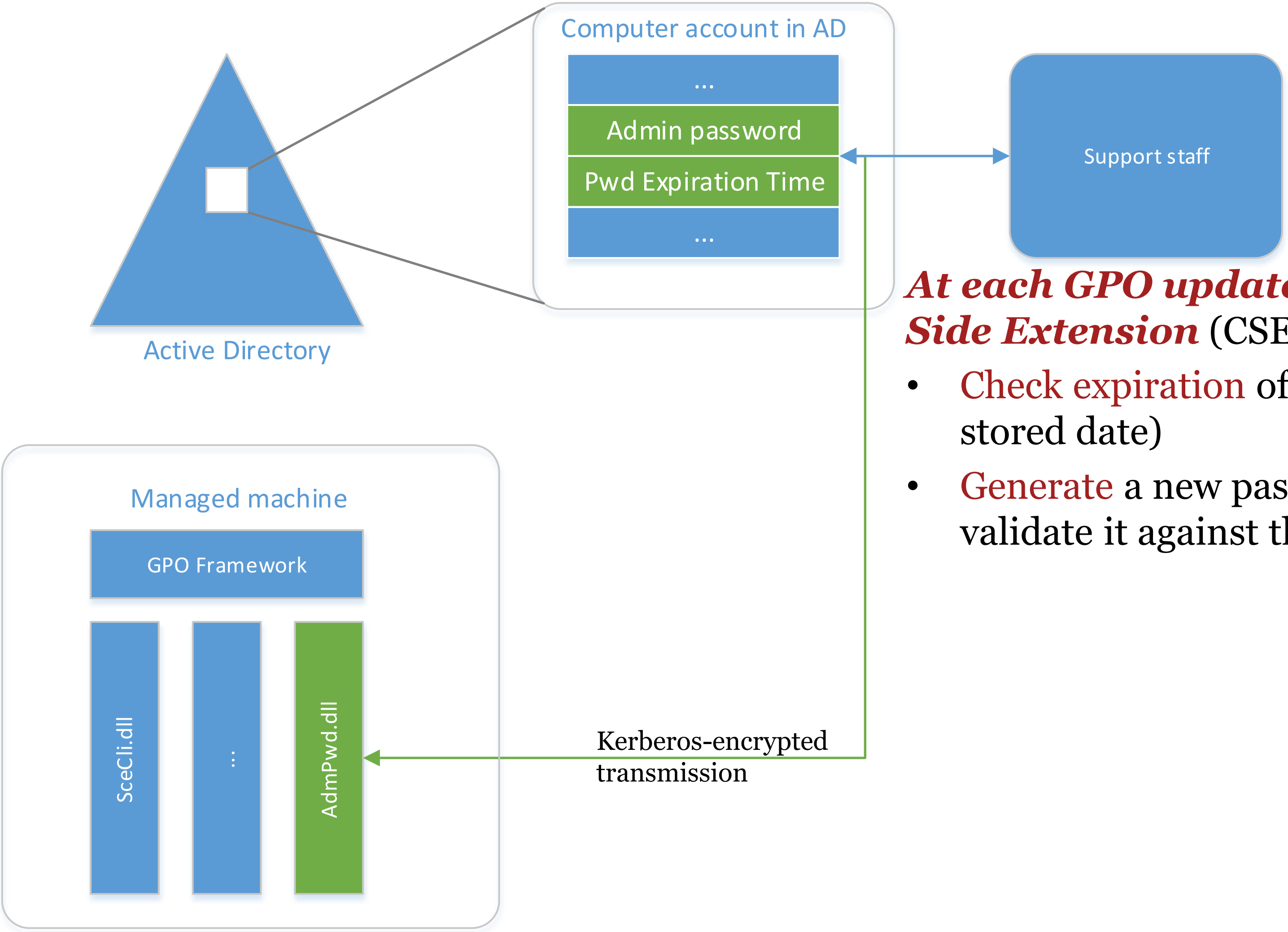
How does LAPS work? (or RTFM!)



Password can be read from AD by users who are allowed to do so. They can request a password change for a computer.

Drawing from LAPS Datasheet (June 2015)
https://www.microsoft.com/en-us/download/details.aspx?id=46899&751be11f-ede8-5a0c-058c-2ee190a24fa6=True

How does LAPS work? (or RTFM!)



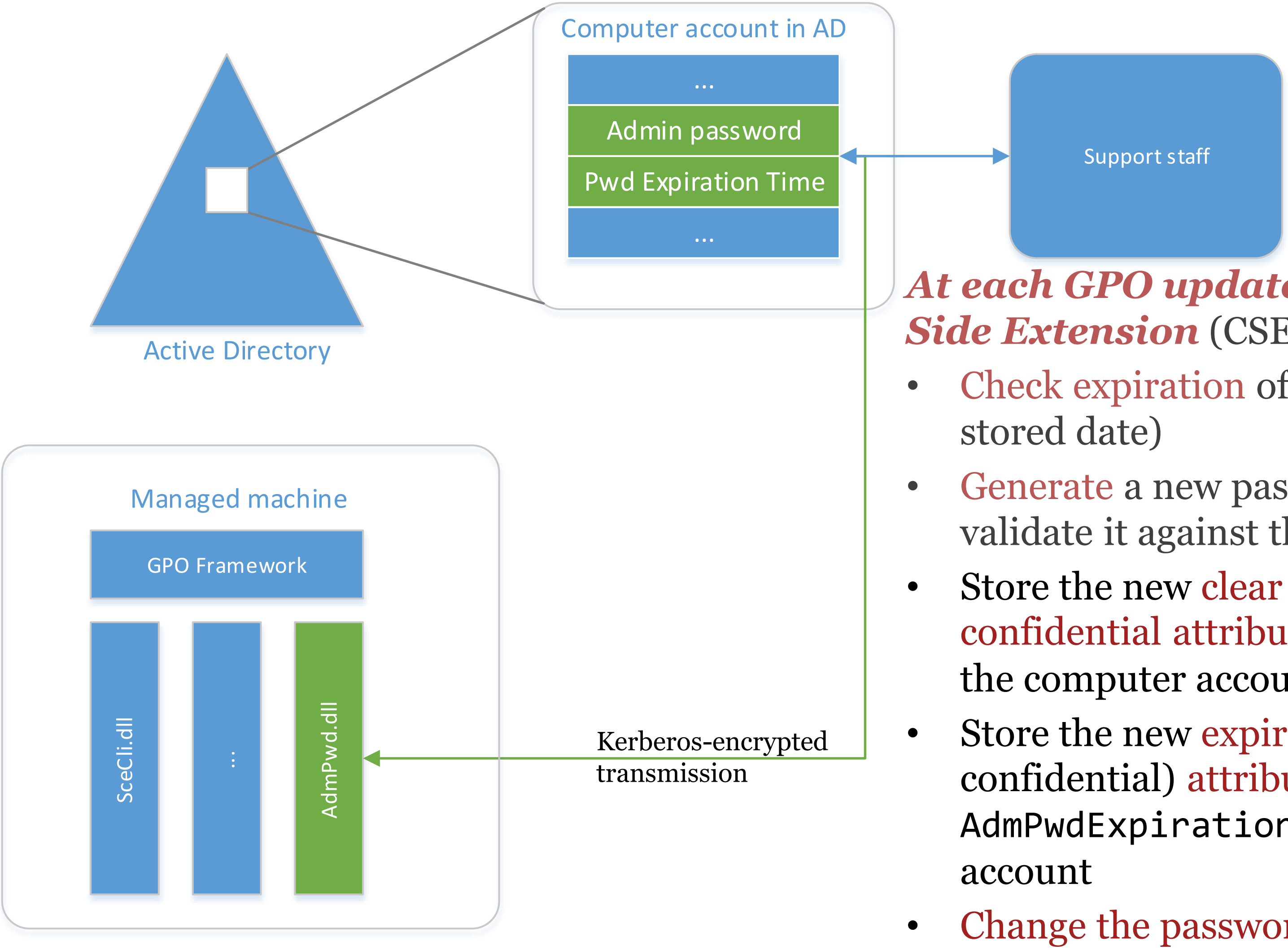
Password can be read from AD by users who are allowed to do so. They can request a password change for a computer.

At each GPO update, LAPS uses a ***Client Side Extension*** (CSE) to:

- Check expiration of password (with AD stored date)
- Generate a new password when needed (and validate it against the policy)

Drawing from LAPS Datasheet (June 2015)

How does LAPS work? (or RTFM!)



Password can be read from AD by users who are allowed to do so. They can request a password change for a computer.

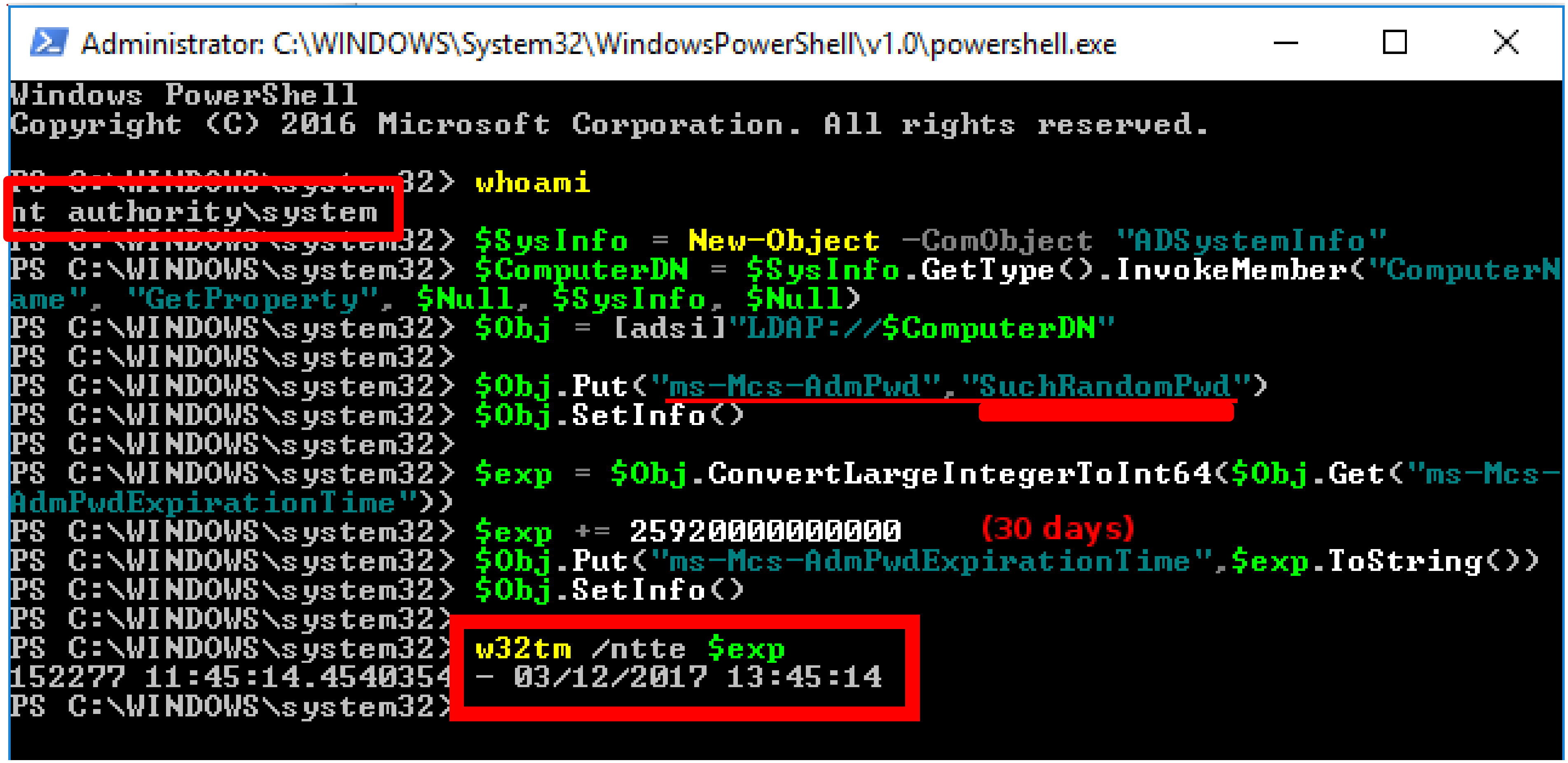
Drawing from LAPS Datasheet (June 2015)

At each GPO update, LAPS uses a ***Client Side Extension*** (CSE) to:

- Check expiration of password (with AD stored date)
- Generate a new password when needed (and validate it against the policy)
- Store the new clear text password in the AD confidential attribute (`ms-Mcs-AdmPwd`) of the computer account
- Store the new expiration date in the AD (not confidential) attribute (`ms-Mcs-AdmPwdExpirationTime`) of the computer account
- Change the password of the built-in local administrator account

Analysis

Preliminary result (or messing with the new attributes)



Administrator: C:\WINDOWS\System32\WindowsPowerShell\v1.0\powershell.exe

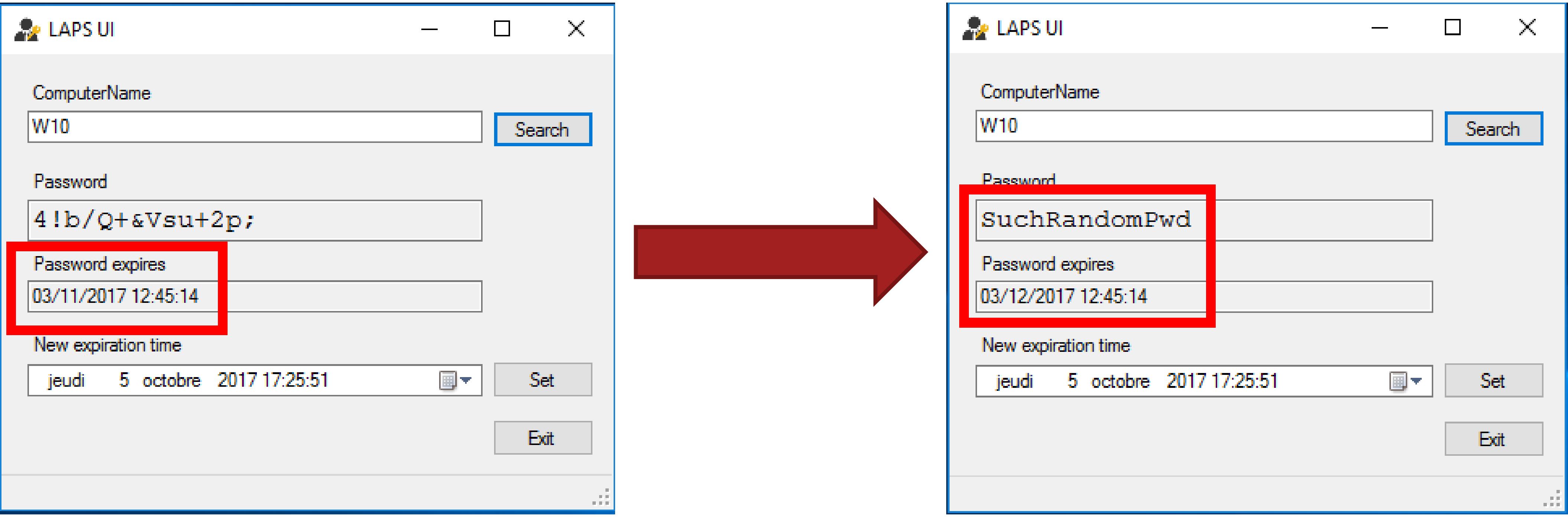
```
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\WINDOWS\system32> whoami
nt authority\system
PS C:\WINDOWS\system32> $SysInfo = New-Object -ComObject "ADSystemInfo"
PS C:\WINDOWS\system32> $ComputerDN = $SysInfo.GetType().InvokeMember("ComputerN
ame", "GetProperty", $Null, $SysInfo, $Null)
PS C:\WINDOWS\system32> $Obj = [adsi]"LDAP://$ComputerDN"
PS C:\WINDOWS\system32>
PS C:\WINDOWS\system32> $Obj.Put("ms-Mcs-AdmPwd","SuchRandomPwd")
PS C:\WINDOWS\system32> $Obj.SetInfo()
PS C:\WINDOWS\system32>
PS C:\WINDOWS\system32> $exp = $Obj.ConvertLargeIntegerToInt64($Obj.Get("ms-Mcs-
AdmPwdExpirationTime"))
PS C:\WINDOWS\system32> $exp += 25920000000000
PS C:\WINDOWS\system32> $Obj.Put("ms-Mcs-AdmPwdExpirationTime",$exp.ToString())
PS C:\WINDOWS\system32> $Obj.SetInfo()
PS C:\WINDOWS\system32>
PS C:\WINDOWS\system32> w32tm /ntte $exp
152277 11:45:14.4540354 - 03/12/2017 13:45:14
PS C:\WINDOWS\system32>
```

(30 days)

Domain workstation: W10

Preliminary result (or messing with the new attributes)



ComputerName: W10

Password: 4!b/Q+&Vsu+2p;

Password expires: 03/11/2017 12:45:14

New expiration time: jeudi 5 octobre 2017 17:25:51

ComputerName: W10

Password: SuchRandomPwd

Password expires: 03/12/2017 12:45:14

New expiration time: jeudi 5 octobre 2017 17:25:51

Domain Controller

Additional research (or RTFM cont'd & LMGTFY)

- The "Client Side Extension" is a single DLL that manages the password

```
C:\Program Files\LAPS\CSE\AdmPwd.dll
```

- LAPS was based on an open source solution named "AdmPwd"
  - Developed by Jiri Formacek since 2011/2012
  - Is part of MS product portfolio since May 2015

code.msdn.microsoft.com/Solution-for-management-of-ae44e789
github.com/jformacek/admpwd

First observations when playing with GPO and gpupdate:

- No integrity check or signature verification of the DLL file
- "AdmPwd" solution is retro-compatible with LAPS

➔ Let's have fun with the source code ☺

Exploitation

Objectives of our PoC

➔ We elaborated ***3 scenarios*** (1 post-exploitation persistence + 2 EoP) depending on the method used to deploy LAPS and the installed KBs.

Both password change and retrieval could be triggered remotely with many covert channels… depending on

- the attacker’s situation (physical access, LAN, Internet, etc.)
- the complementary controls on the targeted system (logs, firewall, etc.).

(➔ Out of scope but interesting Red Team considerations: *what would be the stealthiest channel(s) to make use of that kind of backdoor on a properly monitored remote system?*)

Objectives of our PoC

We want everything to appear as ***normal*** as possible to the Blue Team:

- Password must be synced with the AD and compliant with LAPS policy
- Once backdoor-ed, ***no privilege*** is required to get new passwords

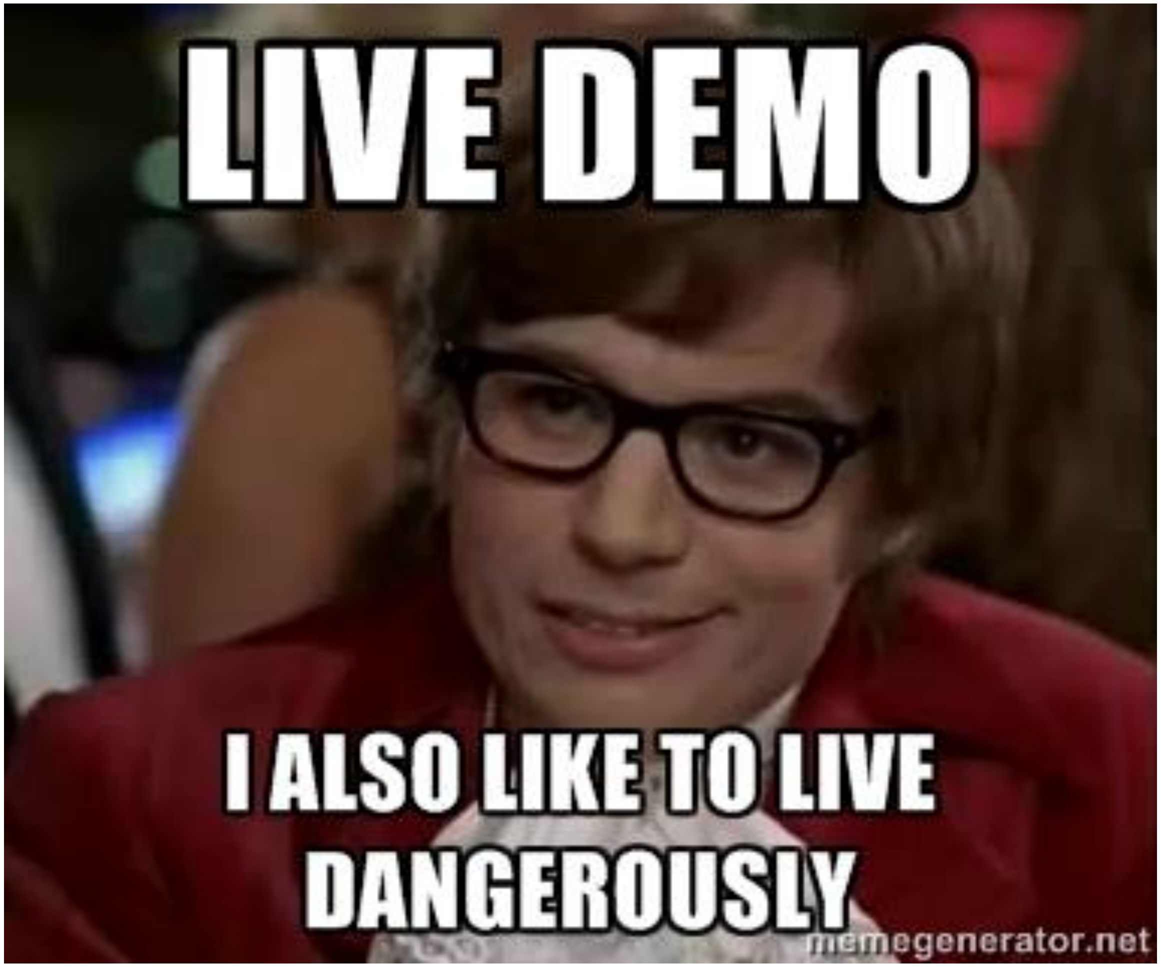
For our PoC, we modified AdmPwd source code to compile a DLL that

- ***ignores the expiration*** date (if a file exists: "backdoor flag")
  ➔ admin password is changed at will (and then synced with the AD)
- ***writes the clear-text admin password*** at a given location
  ➔ admin password is under control at each renewal
- has the same file ***properties*** (desc., version, etc.) that the latest LAPS DLL
  ➔ DLL seems legit at first sight (and "signed")

github.com/secretsquirrel/SigThief

Demo: post-exploitation persistence scenario

Prerequisite: illegitimate temporary privileges \o/



Thoughts on LAPS deployment (or yes, again RTFM!)

LAPS can be deployed on clients with Software Installation feature of Group Policy, SCCM, login script, ***manual install***, etc.

Example: `msiexec /i \\server\share\LAPS.x64.msi /quiet`

Another *documented* method is to ***copy the AdmPwd.dll*** to the target computer and use this command (as admin):

```
regsvr32.exe AdmPwd.dll
```

➔ Combined with bad practices, both allow Escalation of Privilege

LAPS Operations Guide:
https://www.microsoft.com/en-us/download/details.aspx?id=46899&751be11f-ede8-5a0c-058c-2ee190a24fa6=True

Privilege escalation #1 – CVE-2014-1814

Prerequisites:

- Client vulnerable to CVE-2014-1814
- LAPS ***msi*** installed from a *user-writable location* (C:\temp\, share, etc.)

MS14-049: "**Vulnerability in Windows Installer Service** could allow **elevation of privilege** if an attacker runs a specially crafted application that attempts to **repair** a previously-installed application."

Spot the MS14-049: check `C:\Windows\System32\msi.dll` version

e.g. **Win 7 SP1**

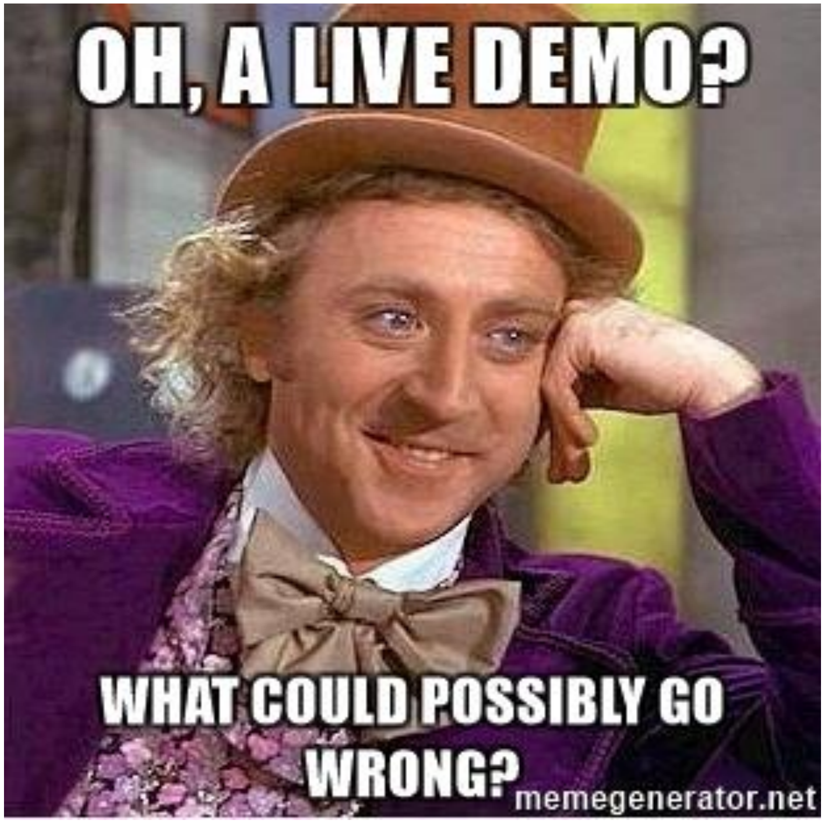
`msi.dll < 5.0.7601.18493` ➔ Vulnerable ☺

The Windows Installer Version Matrix – http://www.installsite.org/pages/en/msifaq/a/1001-matrix.htm

Demo: Privilege escalation #2 – regsvr32.exe

Prerequisite: LAPS installed with "regsvr32" from an *user-writable location*

(works on an up-to-date system)



Recommendations & Conclusion

Recommendations

Validate the integrity/signature of the LAPS DLL.

Example with Powershell v5:

```
Get-FileHash 'C:\Program Files\LAPS\CSE\AdmPwd.dll'
Get-AuthenticodeSignature 'C:\Program Files\LAPS\CSE\AdmPwd.dll'
```

⚠ Admin privileges can give the attacker the possibility to corrupt the signature verification routines locally!

See “Subverting Trust in Windows” from Matt Graeber:

specterops.io/assets/resources/SpecterOps_Subverting_Trust_in_Windows.pdf

➔ **Strict application whitelisting can also be countered this way!**

Recommendations

Monitoring: Server side (Domain Controller)

1. Enable Global Audit Policy & Enable Change Auditing Policy
   - "Audit directory services access" → Audit these attempts [Success/Fail]
   - `auditpol /set /subcategory:"directory service changes" /success:enable`

2. Set up auditing in object SACLs (for each OU or any other object for "Write all properties")

   A. Monitor the changes of the password attribute (`ms-Mcs-AdmPwd`)

   By default the password attribute is not "auditable" it means that changes will not appear in the event logs

   ➔ Switch the "Never Auditing" bit in the `searchFlags` of "`ms-Mcs-AdmPwd`"

Recommendations

Monitoring: Server side (Domain Controller)

1. Enable Global Audit Policy & Enable Change Auditing Policy
   - "Audit directory services access" → Audit these attempts [Success/Fail]
   - `auditpol /set /subcategory:"directory service changes" /success:enable`

2. Set up auditing in object SACLs (for each OU or any other object for "Write all properties")

A. Monitor the changes of the password attribute (`ms-Mcs-AdmPwd`)

By default the password attribute is not "auditable" it means that changes will not appear in the event logs

➔ Switch the "Never Auditing" bit in the `searchFlags` of "`ms-Mcs-AdmPwd`"

➔ ***Bad idea because the cleartext passwords are now accessible to everyone with access to the logs!***

B. Monitor the changes of the expiration time (`ms-Mcs-AdmPwdExperiationTime`)

And correlate these changes with real password expiration and reset by authorised staff!

⚠ Malicious DLL could update the password without changing the expiration time!

Recommendations

Monitoring: Client side (on EACH managed workstation!)

Increase LAPS log level ***and collect/analyse clients' logs!***

Registry key:

```
HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\GPExtensions\{D76B9641-3288-4f75-942D-087DE603E3EA}\ExtensionDebugLevel
```

Value	Meaning
0 (Default value)	Silent mode (errors only) when no error occurs, no information is logged
1	Log Errors and warnings
2	Verbose mode, log everything

⚠ Event logs can be killed (e.g. https://github.com/hlldz/Invoke-Phant0m), logs can be cleared (e.g. Metasploit), or a malicious DLL with disabled logging can be used!

Conclusion

LAPS seems to be a ***convenient way*** to solve the “local admin problem” that many companies face when choosing the Microsoft ecosystem.

Being designed with simplicity in mind, LAPS is not bulletproof, ***its limitations combined with deployment mistakes can be critical***.

Our contribution is an alternative to `@gentilkiwi`’s `mimilib.dll` being used as a malicious Security Support Provider (SSP) as explained there: `adsecurity.org/?p=1760` (this works in a local admin scenario too).

Detecting our tactic is not easy on a large network, it also ***ultimately relies on client-side checks integrity***.
➔ It is questionable when privileges are within reach of attackers.

Future work?
AdmPwd.E

“From the creator of open source AdmPwd solution, that was later adopted by Microsoft as LAPS product, comes Admin Password Manager for Enterprise (AdmPwd.E), built on the same concept as original design […].”

Interesting features (non exhaustive list):

- Maintenance of password ***history***
- Password is ***encrypted*** in Active Directory
- Encryption keys maintained by a dedicated service
- Password management of domain users (in addition to built-in local admin)

http://www.admpwd.com/

Bonus: Microsoft Security Response Center

Their answer for our post-exploitation persistence scenario:

“Our analysis of this issue is that the scenario described requires administrative access to the victim computer. This type of scenario is **not one we consider a security vulnerability**. Elevation from an administrative user to system level access is trivial once a process has administrative access to that same computer. We do not defend against this types of things because the access required for the scenario is greater or equal to that possible post exploitation.”

➔ This is what we expected.

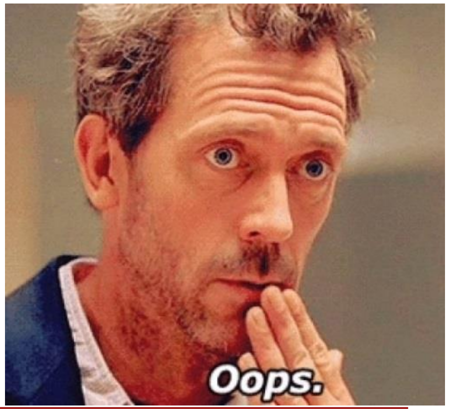
Their answer for our EoP scenario using MSI “repair” feature:

“What is the version of msi.dll?

Please send the Windows Update history.

Please confirm the client has all Windows Update packages installed.”

➔ Oops! Shame on us! Wrong VM… *More haste, less speed.*



Bonus: Microsoft Security Response Center

Their last email after we admit we messed up and that we will present this talk:

“Thanks for getting back to me. Regardless of the mix up, we greatly appreciate your report. Sometimes it's a situation like this, but often it unfortunately isn't and we appreciate a chance to protect customers before findings become public. **We would definitely like to see your slides** if you're willing to share them. I can also solicit **feedback for you from our engineering teams**, which can help clarify details sometimes. I'd also like to state that **we have no issues with you presenting your findings.**

I'll look forward to your slides, and should your research uncover any other issues we would appreciate hearing about them.”

➔ Thank you very much Jason from MSRC!

Thank you!
Any questions?

pwc

This publication has been prepared for general guidance on matters of interest only, and does not constitute professional advice. You should not act upon the information contained in this publication without obtaining specific professional advice. No representation or warranty (express or implied) is given as to the accuracy or completeness of the information contained in this publication, and, to the extent permitted by law, PricewaterhouseCoopers, Société coopérative, its members, employees and agents do not accept or assume any liability, responsibility or duty of care for any consequences of you or anyone else acting, or refraining to act, in reliance on the information contained in this publication or for any decision based on it.