PowerShell Security: PowerShell Attack Tools, Mitigation, & Detection

DerbyCon 6 (2016) Talk – Attacking EvilCorp: Anatomy of a Corporate Hack

AUG 15

# Microsoft LAPS Security & Active Directory LAPS Configuration Recon

By Sean Metcalf in Microsoft Security, Technical Reference

Over the years, there have been several methods attempted for managing local Administrator accounts:

1. Scripted password change – Don't do this. The password is exposed in SYSVOL.
2. Group Policy Preferences. The credentials are exposed in SYSVOL.
3. Password vault/safe product (Thycotic, CyberArk, Lieberman, Quest, Exceedium, etc).

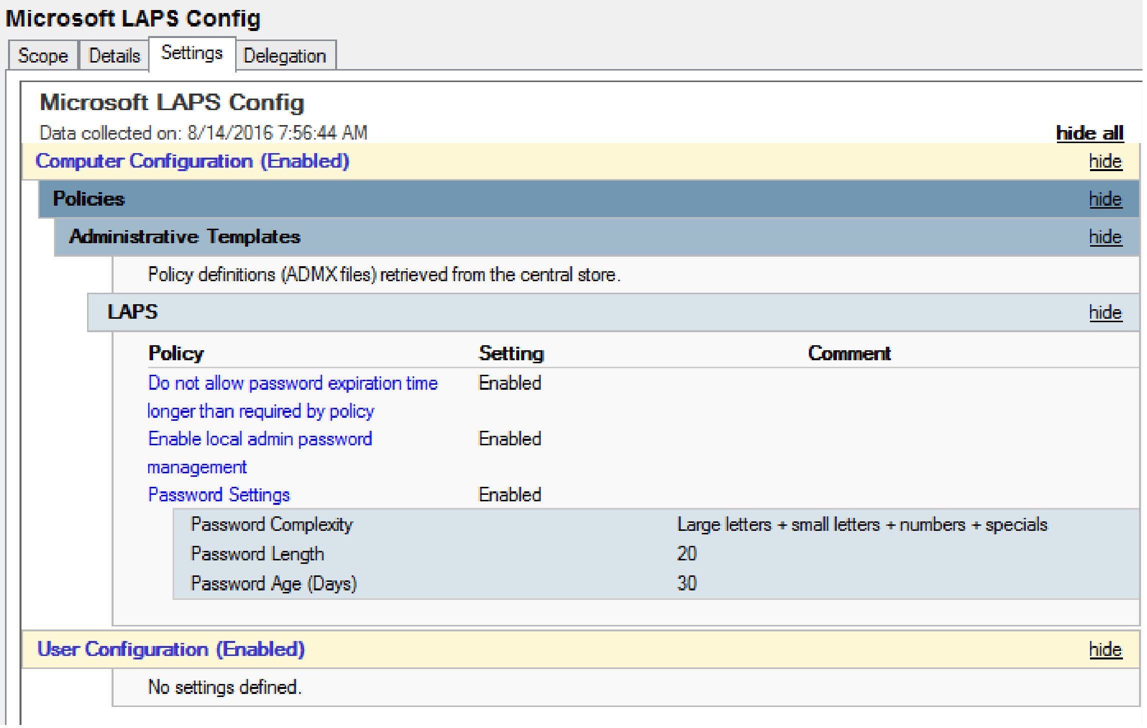
4. Microsoft Local Administrator Password Solution (LAPS).

## LAPS Overview

Microsoft's LAPS is a useful tool for automatically managing Windows computer local Administrator passwords. It's important to ensure every computer changes their local Administrator password regularly, that it's unique for every computer, there's a way to track when it gets changed, and there's a way to force password changes. I cover LAPS in an earlier post, including deployment, pros & cons, among other information.

Here's a quick overview of LAPS:

- Initial install which includes schema extensions – adds ms-mcs-AdmPwd (clear-text password) & ms-mcs-AdmPwdExpirationTime (date/time when password expires which forces the LAPS client to reset the password) attributes to computer objects.
- Deploy the LAPS client to all computers to manage their local Administrator account password.
- Delegate all computers access to update the ms-mcs-AdmPwd & ms-mcs-AdmPwdExpirationTime LAPS attributes on their own computer account (SELF write access).
- Delegate the LAPS computer attributes so the appropriate users have access to view the LAPS password and/or force a reset of the LAPS password (clearing the value of ms-mcs-AdmPwdExpirationTime forces the LAPS client to change the local Administrator password).
- Configure a new Group Policy Object (GPO) to enable & configure LAPS management of local Administrator account password management.



Microsoft LAPS Config

Microsoft LAPS Config

Data collected on: 8/14/2016 7:56:44 AM

Computer Configuration (Enabled)

Policies

Administrative Templates

Policy definitions (ADMX files) retrieved from the central store.

LAPS

| Policy | Setting | Comment |
|---|---|---|
| Do not allow password expiration time longer than required by policy | Enabled | |
| Enable local admin password management | Enabled | |
| Password Settings | Enabled | |

| | |
|---|---|
| Password Complexity | Large letters + small letters + numbers + specials |
| Password Length | 20 |
| Password Age (Days) | 30 |

User Configuration (Enabled)

No settings defined.

*Note that the LAPS GPO setting "Do not allow password expiration time longer than required by policy" is set to Enabled. This is important as you'll see at the end of this post.*

## LAPS Key Points

There are a few interesting key points regarding LAPS:

- When the schema extension is performed, there are two new attributes created for computer objects in Active Directory:
  - ms-mcs-AdmPwd – a "confidential" computer attribute that stores the clear-text LAPS password. Confidential attributes can only be viewed by Domain Admins by default, and unlike other attributes, is not accessible by Authenticated Users. This value is blank until the LAPS password is

    changed. No one but Domain Admins can view this attribute by default. For this reason, delegation of the ms-mcs-AdmPwd attribute has to be carefully planned and performed.
  - ms-mcs-AdmPwdExpirationTime – a regular attribute computer attribute that stores the LAPS password reset date/time value in integer8 format. This value is blank until the LAPS password is changed. When the LAPS password is changed, the value in this attribute is updated based on the LAPS password change threshold (Password Age in days) configured in the LAPS GPO.
- The interesting thing is that while only Domain Admins and delegated groups/accounts can view the LAPS password value stored in the ms-mcs-AdmPwd attribute, any authenticated user can view the value of the ms-mcs-AdmPwdExpirationTime attribute. This means that any user in the Active Directory forest (and any user in a trusted forest/domain) can enumerate the value of this attribute for all computers providing interesting LAPS information:
  - If a computer is managed by LAPS (no value vs value present)
  - When the computer's local Administrator password was last changed (read value in LAPS GPO and subtract this value from the date/time value in the attribute).
  - If a computer's local Administrator password is no longer managed by LAPS (value is equal to a date/time in the past).
- The LAPS GPO Client Side Extension (CSE) runs at the client Group Policy refresh time (90 minutes by default plus a random offset of 0 – 30 minutes to ensure that all clients don't refresh at the same time). At this time, the LAPS CSE checks the ms-mcs-AdmPwdExpirationTime value and if the value is less than the current date/time, LAPS will generate a new password (following settings in the LAPS config GPO), update the value on the computer object's ms-mcs-AdmPwd attribute with this new password, and if successful, change the local account password to the newly generated password, and update the ms-mcs-AdmPwdExpirationTime attribute value to be today + the Password Age value in the LAPS GPO.

## Identifying if LAPS is Installed on a Computer

When the LAPS client is installed, the Group Policy Client Side Extension (CSE) is configured on the computer. This is a DLL (admpwd.dll) located in c:\program files\LAPS\CSE and is configured in the registry: HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\GPExtensions.

Using PowerShell, we can check for the presence of the DLL:

```
Get-ChildItem 'c:\program files\LAPS\CSE\Admpwd.dll'
```

Using PowerShell v5, we can check the file hash & hashing algorithm on the DLL:

```
Get-FileHash 'c:\program files\LAPS\CSE\Admpwd.dll'
```

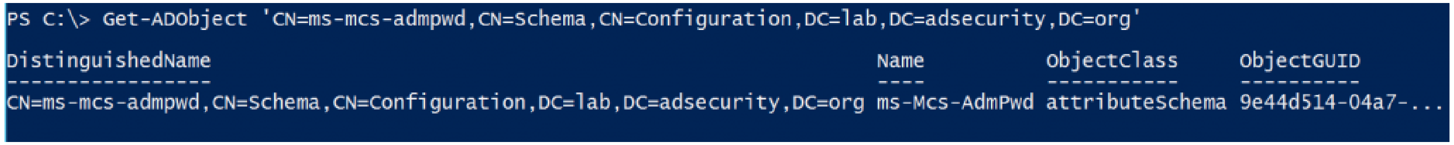
Using PowerShell v5, we can check the digital signature on the DLL:

```
Get-AuthenticodeSignature 'c:\program files\LAPS\CSE\Admpwd.dll'
```

## Discovering LAPS in Active Directory

Since LAPS requires the computer attributes to be present, we can check to see if LAPS is “installed” in Active Directory by checking for the presence of the LAPS attributes in AD.

Here we use the Active Directory PowerShell module cmdlet Get-ADObject to check for the LAPS password attribute ms-mcs-admpwd.

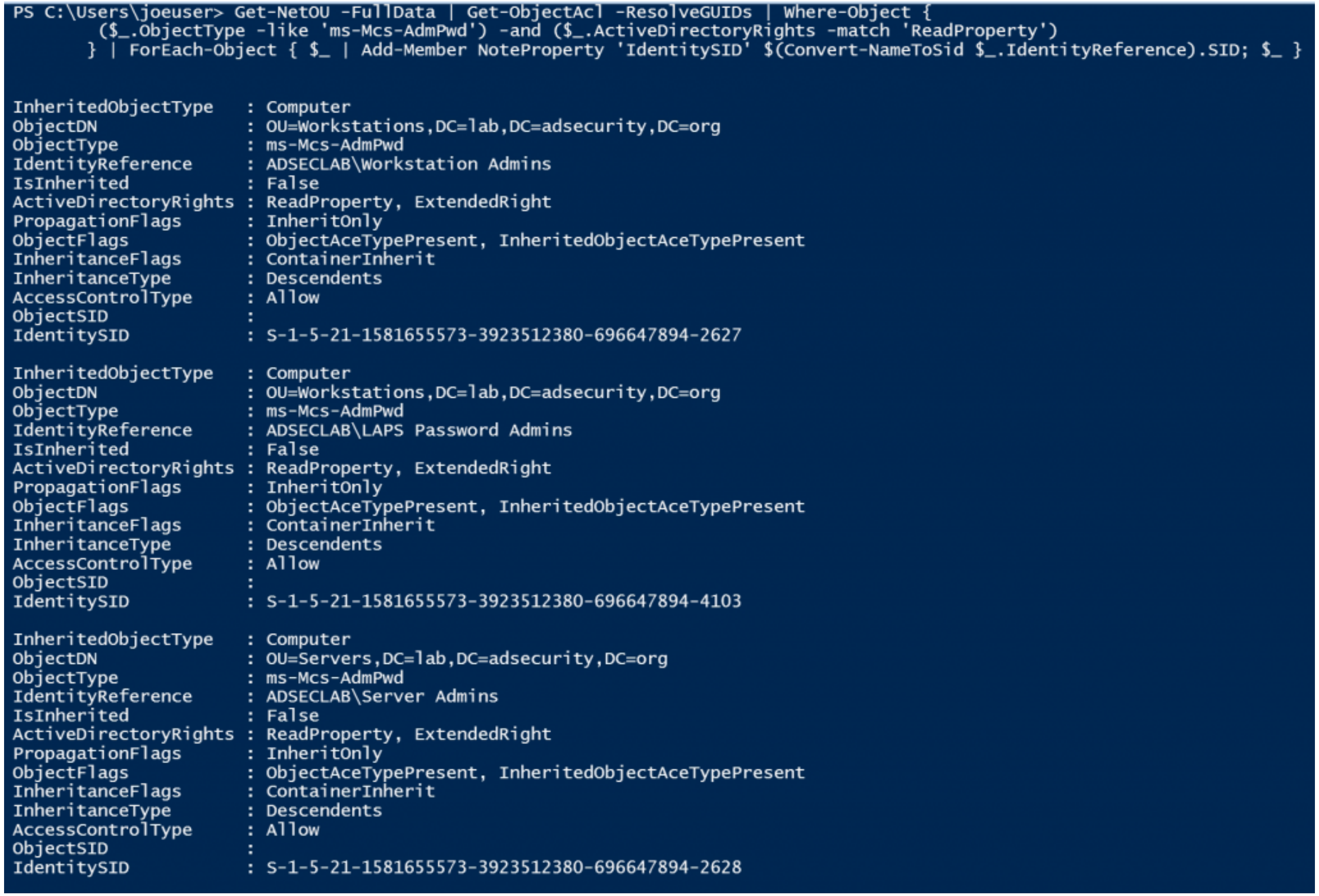


```
PS C:\> Get-ADObject 'CN=ms-mcs-admpwd,CN=Schema,CN=Configuration,DC=lab,DC=adsecurity,DC=org'

DistinguishedName                                               Name         ObjectClass     ObjectGUID
-----------------                                               ----         -----------     ----------
CN=ms-mcs-admpwd,CN=Schema,CN=Configuration,DC=lab,DC=adsecurity,DC=org ms-Mcs-AdmPwd attributeSchema 9e44d514-04a7-...
```

## Identifying LAPS Password View Access (Delegation)

Active Directory objects and their attributes are typically accessible by Authenticated Users. This also includes the security permissions (ACLs) on the objects. Since we have the ability to view these permissions, including delegation, we can

Using PowerView, we can enumerate permissions in the Domain and OUs to discover LAPS attribute delegation.



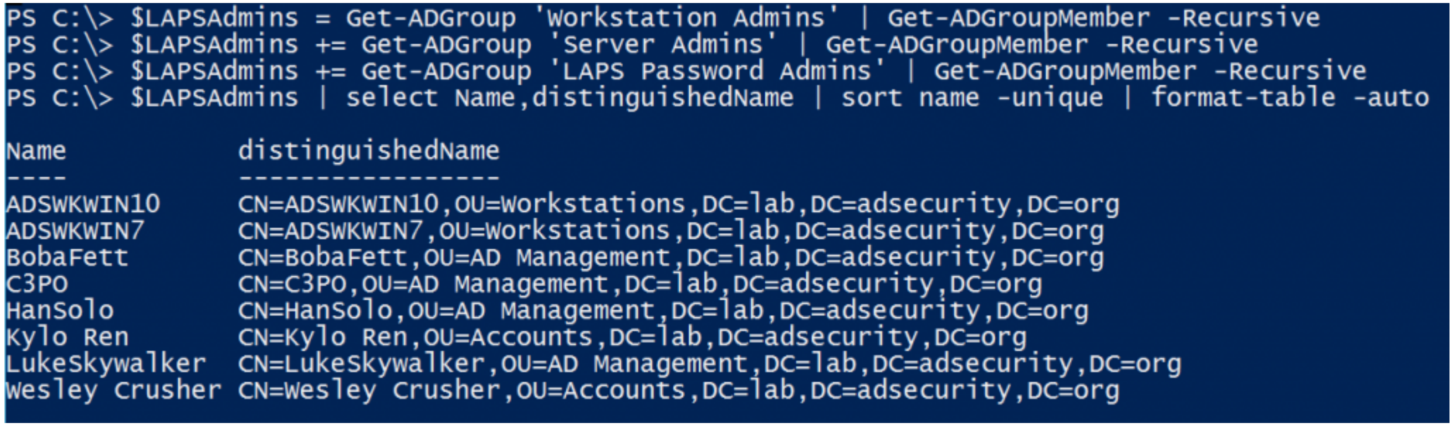
```
PS C:\Users\joeuser> Get-NetOU -FullData | Get-ObjectAcl -ResolveGUIDs | Where-Object {
        ($_.ObjectType -like 'ms-Mcs-AdmPwd') -and ($_.ActiveDirectoryRights -match 'ReadProperty')
       } | ForEach-Object { $_ | Add-Member NoteProperty 'IdentitySID' $(Convert-NameToSid $_.IdentityReference).SID; $_ }


InheritedObjectType   : Computer
ObjectDN              : OU=Workstations,DC=lab,DC=adsecurity,DC=org
ObjectType            : ms-Mcs-AdmPwd
IdentityReference     : ADSECLAB\Workstation Admins
IsInherited           : False
ActiveDirectoryRights : ReadProperty, ExtendedRight
PropagationFlags      : InheritOnly
ObjectFlags           : ObjectAceTypePresent, InheritedObjectAceTypePresent
InheritanceFlags      : ContainerInherit
InheritanceType       : Descendents
AccessControlType     : Allow
ObjectSID             :
IdentitySID           : S-1-5-21-1581655573-3923512380-696647894-2627

InheritedObjectType   : Computer
ObjectDN              : OU=Workstations,DC=lab,DC=adsecurity,DC=org
ObjectType            : ms-Mcs-AdmPwd
IdentityReference     : ADSECLAB\LAPS Password Admins
IsInherited           : False
ActiveDirectoryRights : ReadProperty, ExtendedRight
PropagationFlags      : InheritOnly
ObjectFlags           : ObjectAceTypePresent, InheritedObjectAceTypePresent
InheritanceFlags      : ContainerInherit
InheritanceType       : Descendents
AccessControlType     : Allow
ObjectSID             :
IdentitySID           : S-1-5-21-1581655573-3923512380-696647894-4103

InheritedObjectType   : Computer
ObjectDN              : OU=Servers,DC=lab,DC=adsecurity,DC=org
ObjectType            : ms-Mcs-AdmPwd
IdentityReference     : ADSECLAB\Server Admins
IsInherited           : False
ActiveDirectoryRights : ReadProperty, ExtendedRight
PropagationFlags      : InheritOnly
ObjectFlags           : ObjectAceTypePresent, InheritedObjectAceTypePresent
InheritanceFlags      : ContainerInherit
InheritanceType       : Descendents
AccessControlType     : Allow
ObjectSID             :
IdentitySID           : S-1-5-21-1581655573-3923512380-696647894-2628
```

Here we can see that on the Workstations OU (OU=Workstations, DC=lab,DC=adsecurity,DC=org), the “Workstation Admins” group is delegated read access (ReadProperty) for the ms-Mcs-AdmPwd attribute on computer objects (InheritedObjectType). The “Server Admins” group is also delegated read access to the ms-Mcs-AdmPwd attribute on computer objects in the Servers OU (OU=Servers, DC=lab,DC=adsecurity,DC=org). Furthermore, there is another group called “LAPS Password Admins” which is delegated read access to the ms-Mcs-AdmPwd attribute on computer objects in the Workstations OU (OU=Workstations, DC=lab,DC=adsecurity,DC=org).

From this information we know there are three AD groups that have view access to LAPS passwords on one or more OU:

- Workstation Admins
- Server Admins
- LAPS Password Admins

With this data, we can get a list of accounts that have LAPS password view access in the domain.
The easy way to do this is to get membership of all the identified delegation groups (and their sub-groups) and enumerate all unique members.



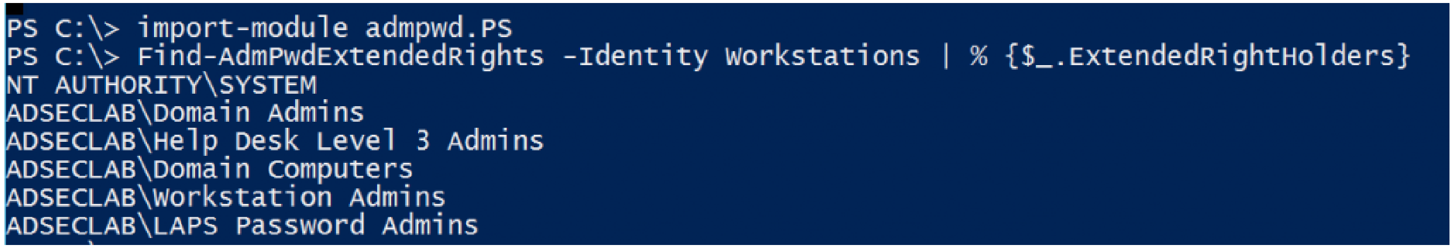
```
PS C:\> $LAPSAdmins = Get-ADGroup 'Workstation Admins' | Get-ADGroupMember -Recursive
PS C:\> $LAPSAdmins += Get-ADGroup 'Server Admins' | Get-ADGroupMember -Recursive
PS C:\> $LAPSAdmins += Get-ADGroup 'LAPS Password Admins' | Get-ADGroupMember -Recursive
PS C:\> $LAPSAdmins | select Name,distinguishedName | sort name -unique | format-table -auto

Name           distinguishedName
----           -----------------
ADSWKWIN10     CN=ADSWKWIN10,OU=Workstations,DC=lab,DC=adsecurity,DC=org
ADSWKWIN7      CN=ADSWKWIN7,OU=Workstations,DC=lab,DC=adsecurity,DC=org
BobaFett       CN=BobaFett,OU=AD Management,DC=lab,DC=adsecurity,DC=org
C3PO           CN=C3PO,OU=AD Management,DC=lab,DC=adsecurity,DC=org
HanSolo        CN=HanSolo,OU=AD Management,DC=lab,DC=adsecurity,DC=org
Kylo Ren       CN=Kylo Ren,OU=Accounts,DC=lab,DC=adsecurity,DC=org
LukeSkywalker  CN=LukeSkywalker,OU=AD Management,DC=lab,DC=adsecurity,DC=org
Wesley Crusher CN=Wesley Crusher,OU=Accounts,DC=lab,DC=adsecurity,DC=org
```

With this information, we know what accounts in Active Directory have the ability to view LAPS password data for at least one OU.
Since Active Delegation is challenging to get right, though the LAPS PowerShell scripts make it pretty straightforward, and even more difficult to track and report on.

## Backdoor to View LAPS Password Data

If a group is delegated "All Extended Rights" to an OU that contains computers managed by LAPS, this group has the ability to view confidential attributes, including the LAPS attribute ms-mcs-admpwd which contains the clear text password. Any group with these rights should be evaluated and removed if necessary. Domain Admins has this right by default, as does SYSTEM. Leave SYSTEM alone.

The issue is this may include groups that shouldn't have LAPS password view access.
We can enumerate these rights using the LAPS PowerShell module cmdlet "Find-AdmPwdExtendedRights".



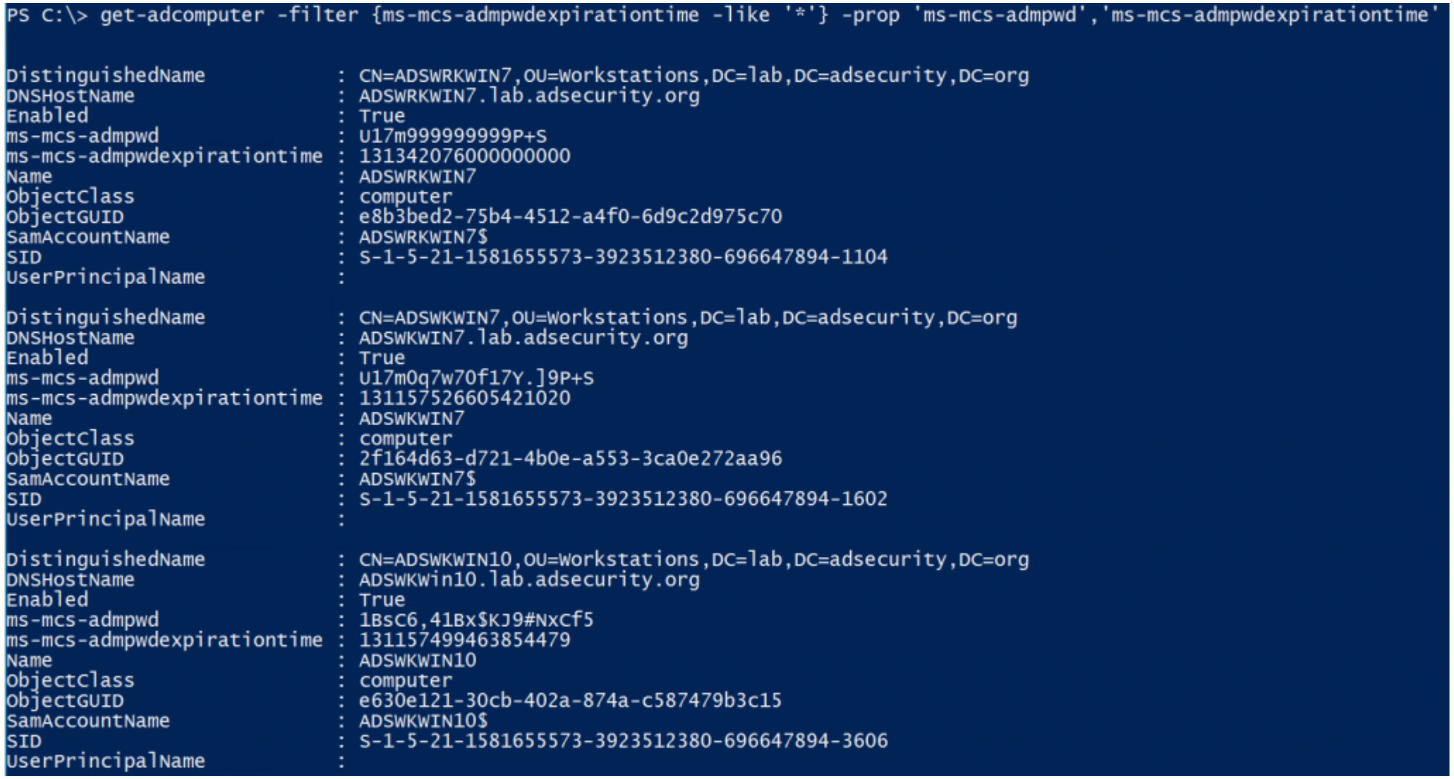
```
PS C:\> import-module admpwd.PS
PS C:\> Find-AdmPwdExtendedRights -Identity Workstations | % {$_.ExtendedRightHolders}
NT AUTHORITY\SYSTEM
ADSECLAB\Domain Admins
ADSECLAB\Help Desk Level 3 Admins
ADSECLAB\Domain Computers
ADSECLAB\Workstation Admins
ADSECLAB\LAPS Password Admins
```

In this scenario, Help Desk Level 3 Admins should *not* have access to view passwords on the Workstations OU.

Additionally, if a group/account has "Owner Rights" on the computer objects, then the group/account has the ability to modify permissions on the objects and attributes. I'll cover AD rights more in a later post.

## Discovering LAPS Password Data

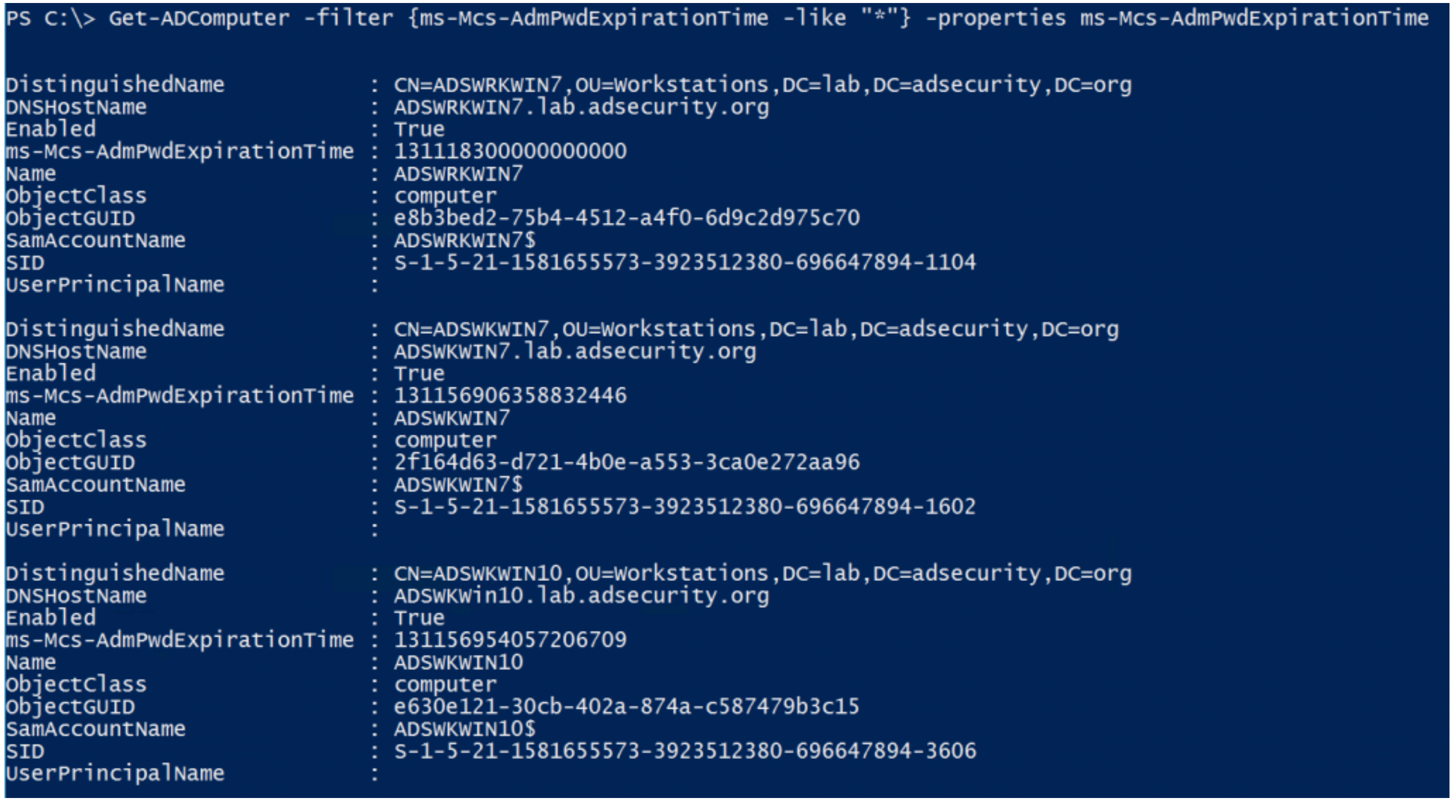
Once we have the appropriate rights (in this example, an account that's a member of LAPS Password Admins group) we can pull the list of computers and their LAPS passwords.



```
PS C:\> get-adcomputer -filter {ms-mcs-admpwdexpirationtime -like '*'} -prop 'ms-mcs-admpwd','ms-mcs-admpwdexpirationtime'


DistinguishedName           : CN=ADSWRKWIN7,OU=Workstations,DC=lab,DC=adsecurity,DC=org
DNSHostName                 : ADSWRKWIN7.lab.adsecurity.org
Enabled                     : True
ms-mcs-admpwd               : U17m999999999P+S
ms-mcs-admpwdexpirationtime : 131342076000000000
Name                        : ADSWRKWIN7
ObjectClass                 : computer
ObjectGUID                  : e8b3bed2-75b4-4512-a4f0-6d9c2d975c70
SamAccountName              : ADSWRKWIN7$
SID                         : S-1-5-21-1581655573-3923512380-696647894-1104
UserPrincipalName           :

DistinguishedName           : CN=ADSWKWIN7,OU=Workstations,DC=lab,DC=adsecurity,DC=org
DNSHostName                 : ADSWKWIN7.lab.adsecurity.org
Enabled                     : True
ms-mcs-admpwd               : U17m0q7w70f17Y.]9P+S
ms-mcs-admpwdexpirationtime : 131157526605421020
Name                        : ADSWKWIN7
ObjectClass                 : computer
ObjectGUID                  : 2f164d63-d721-4b0e-a553-3ca0e272aa96
SamAccountName              : ADSWKWIN7$
SID                         : S-1-5-21-1581655573-3923512380-696647894-1602
UserPrincipalName           :

DistinguishedName           : CN=ADSWKWIN10,OU=Workstations,DC=lab,DC=adsecurity,DC=org
DNSHostName                 : ADSWKWin10.lab.adsecurity.org
Enabled                     : True
ms-mcs-admpwd               : 1BsC6,41Bx$KJ9#NxCf5
ms-mcs-admpwdexpirationtime : 131157499463854479
Name                        : ADSWKWIN10
ObjectClass                 : computer
ObjectGUID                  : e630e121-30cb-402a-874a-c587479b3c15
SamAccountName              : ADSWKWIN10$
SID                         : S-1-5-21-1581655573-3923512380-696647894-3606
UserPrincipalName           :
```

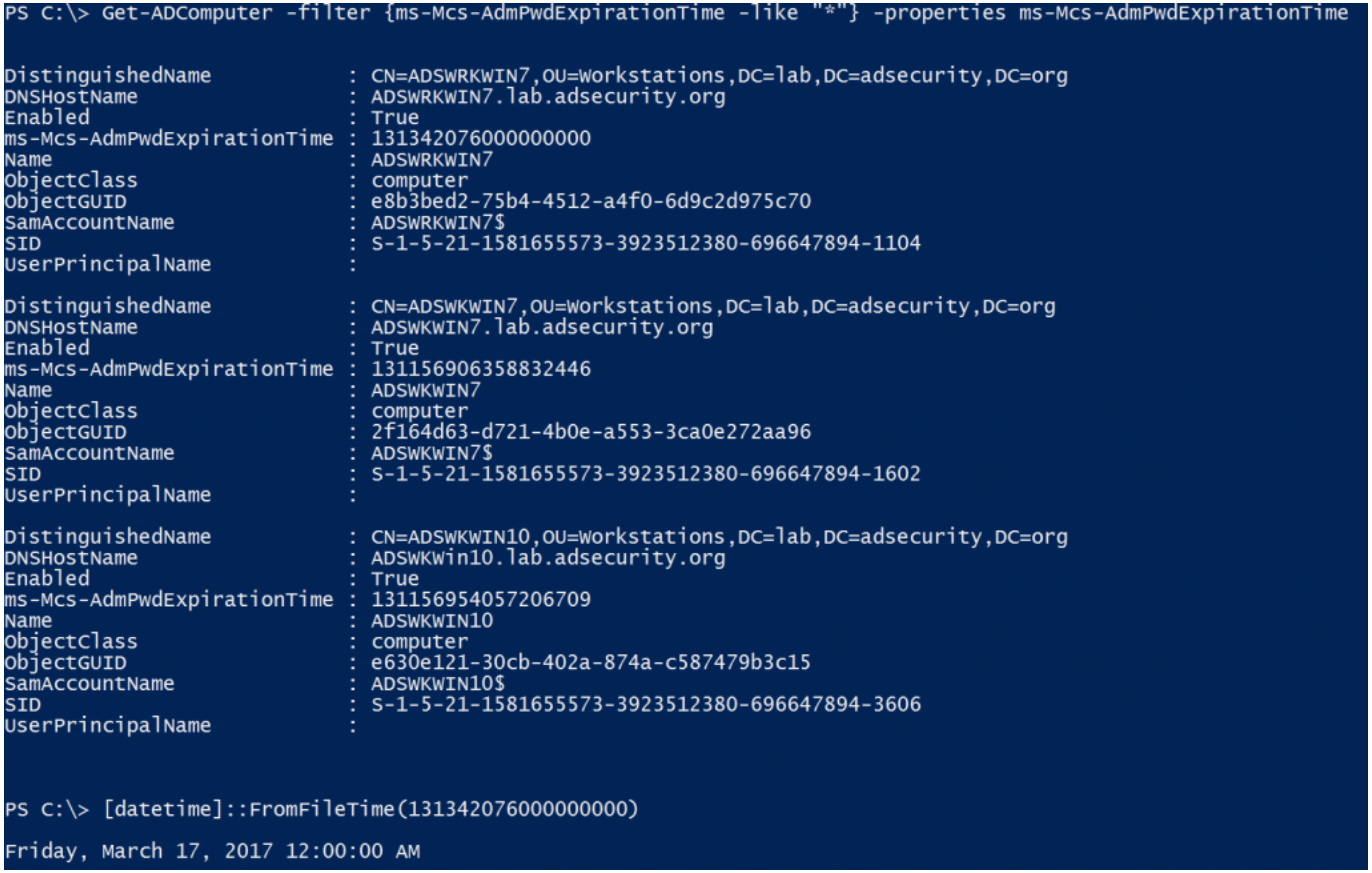
## Identifying LAPS Computer Management

Since the LAPS computer attribute ms-mcs-AdmPwdExpirationTime is a regular attribute, authenticated users have read access, we can track LAPS usage in an environment, including the discovery of computers being managed by LAPS as well as those that aren't actively LAPS managed.



```
PS C:\> Get-ADComputer -filter {ms-Mcs-AdmPwdExpirationTime -like "*"} -properties ms-Mcs-AdmPwdExpirationTime


DistinguishedName           : CN=ADSWRKWIN7,OU=Workstations,DC=lab,DC=adsecurity,DC=org
DNSHostName                 : ADSWRKWIN7.lab.adsecurity.org
Enabled                     : True
ms-Mcs-AdmPwdExpirationTime : 131118300000000000
Name                        : ADSWRKWIN7
ObjectClass                 : computer
ObjectGUID                  : e8b3bed2-75b4-4512-a4f0-6d9c2d975c70
SamAccountName              : ADSWRKWIN7$
SID                         : S-1-5-21-1581655573-3923512380-696647894-1104
UserPrincipalName           :

DistinguishedName           : CN=ADSWKWIN7,OU=Workstations,DC=lab,DC=adsecurity,DC=org
DNSHostName                 : ADSWKWIN7.lab.adsecurity.org
Enabled                     : True
ms-Mcs-AdmPwdExpirationTime : 131156906358832446
Name                        : ADSWKWIN7
ObjectClass                 : computer
ObjectGUID                  : 2f164d63-d721-4b0e-a553-3ca0e272aa96
SamAccountName              : ADSWKWIN7$
SID                         : S-1-5-21-1581655573-3923512380-696647894-1602
UserPrincipalName           :

DistinguishedName           : CN=ADSWKWIN10,OU=Workstations,DC=lab,DC=adsecurity,DC=org
DNSHostName                 : ADSWKWin10.lab.adsecurity.org
Enabled                     : True
ms-Mcs-AdmPwdExpirationTime : 131156954057206709
Name                        : ADSWKWIN10
ObjectClass                 : computer
ObjectGUID                  : e630e121-30cb-402a-874a-c587479b3c15
SamAccountName              : ADSWKWIN10$
SID                         : S-1-5-21-1581655573-3923512380-696647894-3606
UserPrincipalName           :
```

The fun thing about this is if delegation of the ms-Mcs-AdmPwdExpirationTime attribute is too broad, compromise of one of these accounts can change the value of this attribute to be a date/time far in the future, just forcing the local administrator password to not change until then. This is handy for an attacker who has identified the LAPS password(s) and can then extend the use of these passwords.



```
PS C:\> Get-ADComputer -filter {ms-Mcs-AdmPwdExpirationTime -like "*"} -properties ms-Mcs-AdmPwdExpirationTime


DistinguishedName           : CN=ADSWRKWIN7,OU=Workstations,DC=lab,DC=adsecurity,DC=org
DNSHostName                 : ADSWRKWIN7.lab.adsecurity.org
Enabled                     : True
ms-Mcs-AdmPwdExpirationTime : 131342076000000000
Name                        : ADSWRKWIN7
ObjectClass                 : computer
ObjectGUID                  : e8b3bed2-75b4-4512-a4f0-6d9c2d975c70
SamAccountName              : ADSWRKWIN7$
SID                         : S-1-5-21-1581655573-3923512380-696647894-1104
UserPrincipalName           :

DistinguishedName           : CN=ADSWKWIN7,OU=Workstations,DC=lab,DC=adsecurity,DC=org
DNSHostName                 : ADSWKWIN7.lab.adsecurity.org
Enabled                     : True
ms-Mcs-AdmPwdExpirationTime : 131156906358832446
Name                        : ADSWKWIN7
ObjectClass                 : computer
ObjectGUID                  : 2f164d63-d721-4b0e-a553-3ca0e272aa96
SamAccountName              : ADSWKWIN7$
SID                         : S-1-5-21-1581655573-3923512380-696647894-1602
UserPrincipalName           :

DistinguishedName           : CN=ADSWKWIN10,OU=Workstations,DC=lab,DC=adsecurity,DC=org
DNSHostName                 : ADSWKWin10.lab.adsecurity.org
Enabled                     : True
ms-Mcs-AdmPwdExpirationTime : 131156954057206709
Name                        : ADSWKWIN10
ObjectClass                 : computer
ObjectGUID                  : e630e121-30cb-402a-874a-c587479b3c15
SamAccountName              : ADSWKWIN10$
SID                         : S-1-5-21-1581655573-3923512380-696647894-3606
UserPrincipalName           :



PS C:\> [datetime]::FromFileTime(131342076000000000)

Friday, March 17, 2017 12:00:00 AM
```

That’s interesting… One of the computers has an expiration date well past the 30 days configured in the GPO. This means that the LAPS CSE won’t change the local admin password until after that date. Nice little LAPS hack. 🙂

**Note:**
The scenario in the last graphic – preventing LAPS from updating the local password until some point in the future only works if the LAPS GPO setting “*Do not allow password expiration time longer than required by policy*” is set to “Not Configured” or “Disabled”.

The best way to mitigate this potential issue is to ensure the LAPS GPO setting "*Do not allow password expiration time longer than required by policy*" is set to "Enabeld" as shown in the graphic at the beginning of this post.

(Visited 21,158 times, 6 visits today)

Active Directory LAPS, AD, admpwd.dll, Change Local Administrator Password, Find-AdmPwdExtendedRights, HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\GPExtensions, LAPS, LAPS Recon, Local Admin password management, Local Administrator Password Solution, Microsoft LAPS, ms-Mcs-AdmPwd, ms-Mcs-AdmPwdExpirationTime, PowerView

## Sean Metcalf

I improve security for enterprises around the world working for TrimarcSecurity.com
Read the About page (top left) for information about me. :)
https://adsecurity.org/?page_id=8

✉

## 2 comments

**Ryan** on *August 16, 2016 at 5:13 am*

I would be interested to know your thoughts on the password stored in clear text? Our security team are not happy to approve LAPS as they see that as a security risk and want the password encrypted.

my initial thoughts are that the password is stored on a confidential attribute so you would need to compromise AD and the ACLs to get access which at that point does it really matter where your passwords are stored..don't you have bigger issues?

> **Sean Metcalf** on *August 16, 2016 at 9:47 pm*
>
> Author
>
> I agree that it would be better if the password was encrypted and note as much in my earlier post on LAPS (https://adsecurity.org/?p=1790).
> Since the password is stored in a confidential attribute which only Domain Admins have access to by default (not including custom delegation), the password data is appropriately secured. IF the issue is the difference between changing local admin passwords versus not, the answer is simple, use LAPS (or another password management solution).

**Comments have been disabled.**

### RECENT POSTS

Attacking Read-Only Domain Controllers (RODCs) to Own Active Directory

Securing Microsoft Active Directory Federation Server (ADFS)

Gathering AD Data with the Active Directory PowerShell Module

Beyond Domain Admins – Domain Controller & AD Administration

Scanning for Active Directory Privileges & Privileged Accounts

### TRIMARC ACTIVE DIRECTORY SECURITY SERVICES

Have concerns about your Active Directory environment? Trimarc helps enterprises improve their security posture.

Find out how... TrimarcSecurity.com

### POPULAR POSTS

Attack Methods for Gaining Domain Admin Rights in...

PowerShell Encoding & Decoding (Base64)

Securing Windows Workstations: Developing a Secure Baseline

The Most Common Active Directory Security Issues and...

Building an Effective Active Directory Lab…

Detecting Offensive PowerShell Attack Tools

Securing Domain Controllers to Improve Active…

Microsoft Local Administrator Password Solution (LAPS)

Finding Passwords in SYSVOL & Exploiting Group…

PowerShell Version 5 is Available for Download (again)

## CATEGORIES

ActiveDirectorySecurity

Apple Security

Cloud Security

Continuing Education

Entertainment

Exploit

Hacking

Hardware Security

Hypervisor Security

Linux/Unix Security

Malware

Microsoft Security

Mitigation

Network/System Security

PowerShell

RealWorld

Security

Security Conference Presentation/Video

Security Recommendation

Technical Article

Technical Reading

Technical Reference

TheCloud

Vulnerability

## TAGS

Active Directory ActiveDirectory ActiveDirectoryAttack ActiveDirectorySecurity Active Directory Security ADReading ADSecurity DCSync DEFCON DomainController EMET5 GoldenTicket HyperV Invoke-Mimikatz KB3011780 KDC Kerberos KerberosHacking KRBTGT LAPS LSASS MCM MicrosoftEMET MicrosoftWindows mimikatz MS14068 PassTheHash PowerShell PowerShellCode PowerShellHacking PowerShellv5 PowerSploit Presentation Security SIDHistory SilverTicket SneakyADPersistence SYSVOL TGS TGT Windows7 Windows10 WindowsServer2008R2 WindowsServer2012 WindowsServer2012R2

## COPYRIGHT

Content Disclaimer: This blog and its contents are provided "AS IS" with no warranties, and they confer no rights. Script samples are provided for informational purposes only and no guarantee is provided as to functionality or suitability. The views shared on this blog reflect those of the authors and do not represent the views of any companies mentioned. Content Ownership: All content posted here is intellectual work and under the

current law, the poster owns
the copyright of the article.

Content Disclaimer: This blog and its contents are provided "AS IS" with no warranties, and they confer no rights. Script samples are provided for informational purposes only and no guarantee is provided as to functionality or suitability. The views shared on this blog reflect those of the authors and do not represent the views of any companies mentioned.

Made with ♥ by Graphene Themes.