# Local Administrator Password Management Detailed Technical Specification

**Published:** June 2015

**Authors:** Jiri Formacek, Microsoft Services

**Abstract:** This document summarizes technical details of implementation of Local Administrator Password Solution (LAPS)

# Table of Contents


**1 Executive Summary ..... 1**

**2 Project Vision/Scope Summary ..... 2**

**3 Requirements and design Goals ..... 2**

3.1 Business Requirements Summary ..... 2

3.2 User Requirements Summary ..... 2

3.3 Security Requirements Summary ..... 3

3.4 Installation requirements ..... 4

**4 Solution architecture ..... 4**

4.1 Components of the solution ..... 5

**5 Solution Design ..... 6**

5.1 Client Side Group Policy Extension ..... 6

5.1.1 Implementation ..... 6

5.1.2 Configuration ..... 7

5.1.3 Logging ..... 8

5.1.4 Information security ..... 10

5.1.5 Protection against deletion of computer account ..... 11

5.2 Active Directory infrastructure ..... 11

5.2.1 AD Schema ..... 12

5.3 Group Policy ..... 12

5.4 User interface ..... 13

5.4.1 Fat client UI ..... 13

5.4.2 Powershell module ..... 13

5.5 MSI Installer ..... 13

**6 Installation and configuration procedures ..... 14**

6.1 AD schema extension ..... 14

6.2 Delegation of permissions on computer accounts ..... 14

6.2.1 Remove All Extended rights permission ..... 15

6.2.2 Add Write permission to ms-Mcs-AdmPwdExpirationTime and ms-Mcs-AdmPwd attributes to SELF 15

6.2.3 Add CONTROL_ACCESS permission to ms-Mcs-AdmPwd attribute ..... 15

6.2.4 Add Write permission to ms-Mcs-AdmPwdExpirationTime attribute ..... 16

6.3 Installation of CSE ..... 16

6.4 Setup of auditing of password reads ..... 16

6.5 Creation of custom admin account during CSE setup ..... 16

**7 Dependencies..... 16**

7.1 CSE ..... 16

7.2 Management tools ..... 16

# 1 Executive Summary

Purpose of this document is to provide reader with detailed technical specification of solution for management of password of local (built-in or custom) Administrator password on domain-joined computers (servers and workstations).

Technical specification covers the following areas:

- Summary of requirements for the solution
- Architecture of the solution
- Functional specification of particular components of solution
- Installation and configuration procedures

Solution is built as a component of Group Policy framework, a built-in mechanism for management of configuration of domain-joined Windows-based computers.

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 on managed computer. It periodically checks whether the password of local Administrator account has expired or not, and in case it has expired, it generates completely random password for this accounts, resets the account's password to this new value and stores the new password in the Active Directory.

Passwords stored in the Active Directory are stored in confidential attributes (so as special access right is required to read the password) and protected from reading by standard Access Control List (ACL), so only users who are explicitly given the permission to read the password for certain workstation can actually read it.

Users who are given additional permission, can force the change the password for certain workstation.

Transfer of password from managed computer to Active Directory is protected by Kerberos Encryption, so it is not possible to know the password by sniffing the network traffic.

Managed machine itself only has a permission to write the password to its own computer account. This means that it does not have a permission to read any password back from Active Directory - so in case that machine is compromised, attacker still can't read the password of built-in administrator account from AD.

CSE is configured via GPO, the following parameters are configurable:

- Name of administrator account (when not configured, built-in local administrator account is managed)
- Complexity of password
- Length of password
- Maximum age of password (password is automatically changed when password is older than maximum age)

# 2 Project Vision/Scope Summary

Support scenarios for servers and workstations include scenarios when it is not possible to use domain account to log on to server and perform administrative tasks. Such scenarios include:

- Machine loses connection to corporate network and there is not cached credential with administrative privileges
- Machine loses connection with domain or is accidentally disjoined from domain, so domain credentials cannot be used to log on to the server and repair it

For this type of support scenarios, support staff needs to know the password of local Administrator account to be able to log on to computer and perform necessary administrative tasks.

Additionally, there are security aspects of managing local administrative account's password in distributed environment:

- In many environments, password is the same on many machines, which opens the space for Pass-the-hash (PTH) attack
- It is difficult to maintain strong, unique local administrator's passwords and provide access to them on need to know basis.
- It is difficult to regularly change such passwords, force the password change or plan password expiration on certain machine(s)

# 3 Requirements and design Goals

The following paragraphs summarize requirements that solution must fulfil.

## 3.1 Business Requirements Summary

There are the following business requirements for the solution:

- Solution is required to be resistant against tampering with by user of the computer it is implemented on, even if the user of the computer is member of local Administrators group
- Solution must be centrally manageable. This includes:
    - Ability to know the password for certain computer without the need to directly touch it, either locally, or remotely
    - Ability to install, update and uninstall the solution in unattended way and on many computers at the same time
- Solution must support built-in or custom (other than built-in) local administrator account
- Solution must be able to handle the scenario when built-in Administrator account is renamed, without the knowledge of the new name
- Solution must be able to correctly handle the situation when computer is disconnected from corporate network, i.e. not to change the password when it is not possible to report it to the password repository
- Solution must support OS Windows Vista and above and Windows Server 2003 and above
- Solution must support x86 and amd64 hardware platforms

## 3.2 User Requirements Summary

There are the following requirements in the area of end user experience:

- Solution must contain simple to use tool for retrieval of password for administrator account on given computer

- In default configuration, solution must not show any traces of activity on the computer it is installed on – it must be hidden from user as much as possible
- When configured by an administrator, solution must provide with logging of its activity

## 3.3 Security Requirements Summary

There are the following security requirements for the solution:

- Solution must generate unique random password of managed local Administrator account for every managed computer
- Generated passwords must fulfil the following complexity requirements:
    - Password must be 12 characters long by default
        - Password length must be configurable by the administrator of the solution to allow longer password length if required
    - Password complexity must be configurable. Most complex password must contain at least 1 character from each of the following character groups:
        - Capital letters
        - Small letters
        - Numbers
        - Special characters

      *Note*: for characters belonging to each category, see table below
- Maximum age of password must be configurable with default of 30 days. After this time, solution must automatically change the password to new value
- Above mentioned values shall be considered as default and should be configurable
- Solution must allow only authorized personnel to know the password Administrator account for particular computer
- Solution must allow for granular access control for reading the password, on per-workstation basis
- Solution must support changing the password of Administrator account on demand, without the need to directly touch the workstation either locally or remotely, so it is possible to force password change when necessary, before password gets automatically changed because of its age
    - It must be possible to plan the password expiration on per-workstation basis, to support scenarios such as “Password is set to expire today at midnight”
- Solution must allow for auditing of password reads from password repository

Characters for password generation contained in particular categories are specified in table below:

| Category | Characters |
|---|---|
| **Capital letters** | ABCDEFGHIJKLMNOPQRSTUVWXYZ |
| **Small letters** | abcdefghijklmnopqrstuvwxyz |
| **Numbers** | 0123456789 |
| **Special characters** | ,.-+;!#&@{}[]+$/()% |

## 3.4 Installation requirements

Requirements for the installer are:

- Must support unattended installation
- Should be single file performing all tasks related to installation
- Must run on Windows Vista/2003 and above
- Must support x86 and amd64 hardware platforms
- Must support creation of custom admin account during installation

# 4 Solution architecture

Core of the functionality of solution is implemented as Client Side Group Policy Extension (CSE), installed on every managed computer. Password repository is implemented using newly defined attributes in AD schema, added to `may-contain` property set of computer accounts. This implementation model will bring the following benefits:

- **Resistance against tampering with from the side of user of the computer**: security of CSE will be basically the same as security of GPO framework itself
- **Provide privileged security context for local execution**: all local operations will be performed under LOCAL SYSTEM security. This will ensure high enough privileges for local operations (especially password reset of managed admin account).
- **Provide security context for network operations**: Network operations (especially interaction with password repository) will use identity of computer account of managed computer.
- **Automatic timing of operations**: password management (check of password age and change of password if necessary) will be performed every time GPO refresh event occurs on the computer
- **Automatic detection of offline state**: when managed workstation is offline, GPO refresh event will not occur and CSE execution is not triggered
- **Scalability**: locally installed solution is more independent, reliable and scalable than any central solution that touches every managed computer across the network.

Another important component of the solution is password repository. In this solution, Active Directory (AD) infrastructure will be used as a password repository. This will bring the following benefits:

- **Availability**: Design goal is to manage passwords on domain-joined computers, so for every managed computer, AD infrastructure is reachable by design
- **Security**: AD infrastructure offers advanced tools for implementation of security model for the solution by allowing for per-attribute Access Lists (ACLs) and implementing confidential attributes for password storage
- **Auditing**: AD infrastructure implements auditing model on per-attribute level. When there is has security monitoring of AD infrastructure in place, integration of auditing of password reads into security monitoring framework will be straightforward
- **Independence**: Solution is self-contained. It depends only on AD infrastructure and nothing else, which makes it more secure and robust and makes implementation of desired security model easier.
- **Simplicity of implementation of transport encryption**: When transferring passwords from managed workstations to the AD, it is necessary to protect it from eavesdropping on the wire. AD client on managed workstation supports Kerberos-based encryption for LDAP protocol operations. Encryption relies only on Kerberos authentication protocol that is

available to any domain-joined workstation by default. That means that there is no need to implement other encryption means (such as SSL or IPSec) that require additional planning and implementation of prerequisites (such as deployment of server certificates to domain controllers and PKI infrastructure in place)

- **Scalability**: Using AD infrastructure as password repository will allow reporting the password to any writable DC, typically the one that is closest to the workstation; thus password repository is not a single point of failure
- **Protection against attacks**: AD database is one of most important assets for each company, as it contains user identities including their passwords. That means that it is usually accordingly protected, including backup media. This solution just reuses current protection model of AD database for its sensitive data – passwords of managed local Administrator account of managed computers. Additionally, AD infrastructure supports Read-Only Domain Controllers (RODCs) that are designed for environments with insufficient physical security. This solution is not blocker for RODC implementation: passwords of local administrators of managed computers are by default prevented from replication to RODCs.

## 4.1 Components of the solution

Core of the solution is AD infrastructure and custom GPO CSE that were introduced in previous paragraph, however there are more components that make the solution complete. Following list specifies all components of the solution and their responsibilities:

- **Client Side Group Policy Extension** that is installed on each domain-joined computer. CSE will be responsible for the following tasks:
  - Management of password of Administrator password:
    - Checking whether the password of Administrator account has expired or not
    - Generating the new password when old password expires or is required to be changed prior to expiration
    - Validating newly generated password against password policy that is in place
    - Reporting the password to password repository
    - Reporting the next expiration time to password repository
    - Changing the password of Administrator account
  - Logging of activity to the Application Event log
  - Publishing event log viewer templates so as event messages in Application Event Log of managed computer are correctly displayed
  - Publishing of COM-style installation/uninstallation functions (DllRegisterServer, DllUnregisterServer) for case that MSI installation does not fit
- **Active Directory infrastructure**. AD will be responsible for the following tasks:
  - Will be used as a password repository
  - Will enforce security and auditing model upon passwords
- **Group Policy**. GPO will be responsible for the following tasks:
  - Triggering the execution of CSE on managed computer. CSE will be triggered every time GPO refresh event occurs on the computer
  - Configuration of the solution. Solution comes with ADMX templates defining configuration options
- **User console**. Any tool for viewing AD data (such as Active Directory Users and Computers, LDP, or ADSIEDIT) can be used to view the solution data in AD. Additionally, this solution contains additional UI to retrieve passwords:

  - Simple fat client UI
  - PowerShell module
- Both types of UI offer the following functionality:
  - Allow user to enter computer name
  - Contact AD infrastructure in the security context of user who runs the tool
  - Show the computer name and password to the user
  - Provide the user with UI to force expiration of password for computer (immediate or planned for certain time)
- **Windows Installer package for x86 and amd64 platforms.** Installation package by default installs CSE and can install User console components (fat client and PowerShell module)

Detailed description of particular components is subject of the following paragraphs

# 5 Solution Design

## 5.1 Client Side Group Policy Extension

### 5.1.1 Implementation

CSE is implemented as single DLL file, publishing the following entry points:

- DllRegisterServer
  - Can be used for manual registration of CSE with GPO framework and with Event Log service during the CSE installation/upgrade
- DllUnregisterServer
  - Can be used for manual deregistration of CSE from GPO framework and Event Log service during the uninstallation process of CSE
- ProcessGroupPolicy
  - It is main entry point for Group Policy framework. This entry point implements ProcessGroupPolicy callback as described in MSDN[1]

Files:

- %ProgramFiles%\LAPS\CSE\AdmPwd.dll

Logic of the processing is as follows:

- CSE connects to computer object in Active Directory; to the computer object for workstation or server it is running on
- CSE the reads the value of attribute "`ms-Mcs-AdmPwdExpirationTime`". This attribute stores the expiration time of current password
  - When the attribute is empty, password was never changed, so CSE knows it is the time to reset the password
  - When the timestamp is not older that current time, password has not expired yet, and CSE does not perform any other operation and finishes processing
  - When the timestamp is older than current time, CSE knows it is the time to reset the password
- When password needs to be reset, CSE detects the local Administrator account to manage (either via name configured using GPO or via well-known SID) and connects to it

[1] See http://msdn.microsoft.com/en-us/library/aa374377(VS.85).aspx

- Then CSE invents new password according to required criteria (length and complexity)
- Then CSE validates the password against password policy to make sure that password reset attempt later on will not be rejected
- Then CSE reports new password and timestamp to Active Directory, to the following attributes of computer account for workstation it runs on:
    - `ms-Mcs-AdmPwd`: password in clear text
    - `ms-Mcs-AdmPwdExpirationTime`: timestamp of current time plus maximum age of password, in FILETIME format (64-bit integer), in UTC
    - *Note*: This communication is encrypted with Kerberos encryption
- After password and expiration timestamp are successfully reported to AD, the password of managed Administrator account is reset to new value
    - Reason for this sequence of steps is that we cannot report and reset password as a single transaction. So we consider the reporting of password to AD as more "risky" – more things can get wrong as there is network between workstation and domain controller, whereas password reset operation works against local computer. We try to perform the operation considered more risky first to be able to catch any errors prior resetting the password. This order of steps minimizes the risk that reported password will be different than actual password of managed Administrator account
- After successfully resetting the password, CSE finishes execution reporting success to GPO framework that called it
- In case that some error occurs during the execution, CSE logs the error to Application log and finishes execution, reporting the error to GPO framework that called it

### 5.1.2 Configuration

CSE is configurable using registry values specified in the registry key:

`HKLM\Software\Policies\Microsoft Services\AdmPwd`

Currently the following configuration values are supported:

| Value | Type | Meaning |
|---|---|---|
| **AdmPwdEnabled** | REG_DWORD | Setting to non-zero enables the solution.<br>Resulting policy must have this value set to non-zero so as the solution is enabled to work.<br>Managed by policy "Enable local admin password management" |
| **AdminAccountName** | REG_SZ | Name of local account to manage password for.<br>If not configured, CSE manages built-in Administrator password regardless of its name (detects it via well-known SID)<br>Managed by policy "Customize administrator account name" |
| **PasswordLength** | REG_DWORD | Length of password generated<br>Minimum: 8<br>Maximum: 64<br>Default: 14<br>Managed by policy "Password Settings" |

| Value | Type | Meaning |
|---|---|---|
| **PasswordComplexity** | REG_DWORD | Complexity of generated password<br>Minimum: 1<br>Maximum: 4<br>Default: 4<br>(see paragraph 3.3 for details)<br>Meaning of values:<br>1 ... large letters<br>2 ... large_letters+small letters<br>3 ... large_letters + small_letters + numbers<br>4 ... large_letters + small_letters + numbers + spec_chars<br>Managed by policy “Password Settings” |
| **PasswordAgeDays** | REG_DWORG | Maximum age of password<br>Minimum: 1<br>Maximum: 365<br>Default: 30<br>Managed by policy “Password Settings” |
| **PwdExpirationProtectionEnabled** | REG_DWORD | Whether CSE shall enforce password age to be aligned with PasswordAgeDays parameter<br>If set to non-zero, when password expiration time set on computer exceeds PasswordAgeDays policy, password is reset upon next GPO refresh and expiration is set according to policy<br>Managed by policy “Do not allow password expiration time longer than required by policy” |

### 5.1.3 Logging

CSE logs all events in Application Event Log of local computer. Log messages are English only, but can be localized or additional language can be added, if necessary.

Number of events that are logged is configurable via the following registry REG_DWORD value:

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

Semantic of possible values is as follows:

| Value | Meaning |
|---|---|
| **0** | Silent mode; log errors only<br>When no error occurs, no information is logged about CSE activity<br>This is a default value |
| **1** | Log Errors and warnings |
| **2** | Verbose mode, log everything |

Event source for all events reported by CSE is always “AdmPwd”. The following table summarizes the events that can occur in the Event Log:

| ID | Severity | Description | Comment |
|---|---|---|---|
| **2** | Error | Could not get computer object from AD. Error %1 | This event is logged in case that CSE is not able to connect to computer account for local computer in AD.<br>%1 is a placeholder for error code returned by function that retrieves local computer name, converts it to DN and connects to object, specified by the DN |
| **3** | Error | Could not get local Administrator account. Error %1 | This event is logged in case that CSE is not able to connect to managed local Administrator account.<br>%1 is a placeholder to error code returned by function that detects the name of local administrator’s account and connects to the account |
| **4** | Error | Could not get password expiration timestamp from computer account in AD. Error %1. | This event is logged in case that CSE is not able to read the value of `ms-Mcs-AdmPwdExpirationTime` of computer account in AD<br>%1 is a placeholder for error code returned by function that reads the value of the attribute and converts the value to `unsigned __int64` type |
| **5** | Error | Validation failed for new local admin password against local password policy. Error %1. | This event is logged when password validation against local password policy fails. |
| **5** | Information | Validation passed for new local admin password. | This event is logged when password is successfully validated against local password policy |
| **6** | Error | Could not reset local Administrator's password. Error %1 | This event is logged in case that CSE is not able to reset the password of managed local Administrator account.<br>%1 is a placeholder for error returned by `NetUserSetInfo() API` |
| **7** | Error | Could not write changed password to AD. Error %1. | This event is logged in case that CSE is not able to report new password and timestamp to AD.<br>%1 is a placeholder for error code returned by ldap_mod_s call |
| **10** | Warning | Password expiration too long for computer (%1 days). Resetting password now. | This event is logged in case that CSE detects that password expiration for computer is longer than allowed by policy in place while protection against excessive password age is turned on |
| **11** | Information | It is not necessary to change password yet. Days to change: %1. | This event is logged after CSE detects that it is not yet the time to reset the password<br>%1 is a placeholder for number of 24-hour’s intervals that remain till the password will be reset |
| **12** | Information | Local Administrator's password has been changed. | This event is logged after CSE resets the password of managed local Administrator account |
| **13** | Information | Local Administrator's password has been reported to AD. | This event is logged after CSE reports the password and timestamp to AD |

| ID | Severity | Description | Comment |
|---|---|---|---|
| **14** | Information | Finished successfully | This event is logged after CSE performed all required tasks and is about to finish |
| **15** | Information | Beginning processing | This event is logged when CSE starts processing |
| **16** | Information | Admin account management not enabled, exiting | This event is logged when admin account management is not enabled |

Notes:

- Generally, all events with severity “Error” are blocking, so in case that any error occurs, no other tasks are performed and CSE terminates processing
- Event source for the Event Log is embedded in the same DLL as main GPO executive. Reason for this decision was to make the deployment simple

### 5.1.4 Information security

Solution maintains 2 pieces of information for managed Administrator account in Active Directory:

- Current password
- Timestamp of expiration of current password

Permission model around this information is as follows:

| Information | Who can read | Who can write |
|---|---|---|
| **Password** | IT support staff responsible for workstation support | Computer that owns the computer account (so every computer can write only own password to AD) |
| **Password Expiration Timestamp** | Anyone who can read other attributes of computer account<br><br>Computer that owns the computer account (so every computer can know whether it is the time to change the password) | Computer that owns the computer account (so every computer can write only own password expiration timestamp to AD)<br><br>IT support staff responsible for workstation support (so they are able to force password reset upon next GPO refresh or explicitly set password expiration time ) |

*Note*: Domain administrators and anyone who has full control on computer objects in AD can obviously read and write both pieces of information.

When transferred over the network, both password and timestamp are encrypted by Kerberos encryption

When stored in AD, both password and timestamp are stored in clear text.

We decided to store password in AD in clear text because:

- Password is protected by ACL, so it is possible to define who can and who cannot read it
- Password encryption in AD would make the solution much more difficult to implement while the level of security would not increase much:

- When using symmetric encryption, key distribution and protection mechanism would need to be implemented, because:
  - The managed computer would need to encrypt the password
  - IT support staff would need to decrypt the password
  - Both parties would need to use the same key
- When using asymmetric encryption, workstation could encrypt by its private key, and IT support staff would decrypt using public key. Distribution and protection of public key would still need to be implemented so as all users in IT Support staff role (and no one else) could have the public key

Above means that distribution of decryption key would need to be implemented, which leads to complexity in implementation of key distribution and protection mechanism (much bigger complexity than password management solution itself. Proper solution for management of encryption/decryption keys would probably resemble Information Rights Management infrastructure). So we decided not to encrypt the password in AD and rely on protection of AD database that most organizations have already implemented as a protection means for sensitive information it contains.

### 5.1.5 Protection against deletion of computer account

Computer accounts might be subject of accidental deletion. In such case (especially when AD Recycle Bin feature of Windows 2008 R2 is not implemented) password of managed local Administrator account would be lost and there would not be an easy way for support staff to read it: it would require using the SystemState backup to read the password – unless the Forest Functional Level (FFL) is Windows 2008 R2 and AD Recycle Bin feature is turned on.

Approach for protection against accidental deletion of computer account will be implemented as follows:

- `ms-Mcs-AdmPwd` attribute is added to the set of attributes that will not be stripped off the object during the deletion
- This means that password will still be available on tombstone of computer account for the lifetime of tombstone – which is 180 days by default
- So when accidental deletion of computer account occurs, Domain admin role will be able to quickly recover the password from the tombstone object
- Only after tombstone expires, the password is definitely lost. Tombstone lifetime is long enough for the purpose of password recovery

Main benefit of this approach is that it allows not exporting passwords from AD infrastructure to independent location where it would need to be specially protected (which could be difficult, especially in case when owner of the independent storage would not be Domain admin role) – just for covering the special case of accidentally deleted computer account.

## 5.2 Active Directory infrastructure

Active Directory infrastructure supports the solution by:

- Implementing the shared storage of information maintained by the solution
- Implementing GPO framework that is used to trigger CSE activity

The following paragraphs summarize changes that are required on the Active Directory level when implementing the solution.

### 5.2.1 AD Schema

It is required to extend the schema of AD by two new attributes that store password of managed local Administrator account for each workstation and timestamp of password expiration.

- Both attributes are added to `may-contain` attribute set of `computer` class.

Specification of new attributes is in the table below, full AD schema extension LDIF script is attached in file AdmPwd_SchemaExtension that is part of delivery.

| Attribute | Parameter | Value |
|---|---|---|
| **ms-Mcs-AdmPwd** | Syntax | 2.5.5.5<br>(Printable case-sensitive string) |
| | omSyntax | 19 |
| | isSingleValued | True |
| | searchFlags | 904<br>(`fCONFIDENTIAL` \| `fPRESERVEONDELETE` \| `fRODCFilteredAttribute` \| `fNeverAuditValue`) |
| | isMemberOfPartialAttributeSet | False |
| | OID | 1.2.840.113556.1.8000.2554.50051.45980.28112.18903.35903.6685103.1224907.2.1 |
| **ms-Mcs-AdmPwdExpirationTime** | Syntax | 2.5.5.16<br>(Large integer) |
| | omSyntax | 65 |
| | isSingleValued | True |
| | searchFlags | 0 |
| | isMemberOfPartialAttributeSet | False |
| | OID | 1.2.840.113556.1.8000.2554.50051.45980.28112.18903.35903.6685103.1224907.2.2 |

*Note*: In case that RODC is installed in the environment, and you really need to replicate the value of attribute `ms-Mcs-AdmPwd` to RODC, set the bit 10 of `searchFlags` attribute value for `ms-Mcs-AdmPwd` schema objet to 0 (substract 512 from current value of `searchFlags` attribute)

## 5.3 Group Policy

Solution installs GPO templates that implement UI for setting configuration options described in 5.1.2 .

Files:

- Admpwd.admx
- En-us\AdmPwd.adml

ADMX templates are installed into %SystemRoot%\PolicyDefinitions folder. In case that organization uses centralized policy store (ADMX templates stored in SYSVOL share), administrator is required to cope the ADMX templates into central policy store in SYSVOL.

## 5.4 User interface

Solution supports 2 types of management UI:

- Fat client AdmPwd.UI.exe that provides the functionality of password retrieval for given computer and planned/immediate password reset for a computer
- Powershell module AdmPwd.PS that provides the same functionality as fat client plus the following:
  - Cmdlet for AD schema extension
  - Cmdlets for delegation of permissions for computer accounts themselves (to be able to write passwords to AD) and for IT staff (to read passwords and request password resets)
  - Cmdlet to find who has permission to read password on computers in given container
  - Cmdlet for setting up auditing of password reads from AD

### 5.4.1 Fat client UI

Fat client installs into folder %ProgramFiles%\LAPS

Files:

- AdmPwd.UI.exe
- Admwd.Utils.dll
- AdmPwd.Utils.config

### 5.4.2 Powershell module

Powershell module name is AdmPwd.PS and installs into $pshome\Modules\AdmPwd.PS

Files:

- AdmPwd.PS.dll
- AdmPwd.PS.psd1
- Admwd.Utils.dll
- AdmPwd.Utils.config
- AdmPwd.PS.format.ps1xml
- En-us\ AdmPwd.PS.dll-Help.xml

## 5.5 MSI Installer

All components are contained in single MSI package.

MSI package supports unattended install of any component.

Installing MSI without specific parameters installs just CSE.

Use the following command line for non-default installs:

```
msiexec /q /i <path>\LAPS.<platform>.msi ADDLOCAL=<FeatureID>
```

Supported feature IDs:

| Feature | ID |
|---|---|
| **CSE** | CSE |
| **Fat client** | Management.UI |
| **Powershell** | Management.PS |
| **ADM templates** | Management.ADMX |

# 6 Installation and configuration procedures

Installation of binaries and related files in handled by MSI package. Package installs the following:

- GPO CSE: must be present on each managed machine
- Management tools:
  - Fat client
  - Powershell module AdmPwd.PS
  - Group Policy Editor admin templates

Default is to install CSE only; management tools are installed on demand

Configuration procedures include procedures that will be performed manually. Those procedures include:

- Mandatory: Schema extension
- Mandatory: Delegation of permissions on computer accounts
- Mandatory: Installation of CSE on managed computer – via MSI
  - Or copy the AdmPwd.dll to target computer and call DllRegisterServer on it (i.e. via regsvr32.exe)
- Optional: Installation of fat client and Powershell module – when using this types of management UI
- Optional: Setup of auditing of password reads

Paragraphs below provide more details on some of mentioned installation procedures

## 6.1 AD schema extension

AD schema extension will be performed using the following PowerShell script:

```
Import-module AdmPwd.PS

Update-AdmPwdADSchema
```

This task needs to be performed by user in Schema Admin role

## 6.2 Delegation of permissions on computer accounts

Delegation of permissions on computers accounts is performed on OU (OUs) that contain computer accounts in scope of the solution.

This task covers the following operations:

- Remove `All Extended Permissions` permission from users and groups that are not allowed to read the value of attribute `ms-Mcs-AdmPwd`. This is required because All Extended permissions permission gives also permission to read confidential attributes.
- Add `Write` permission on `ms-Mcs-AdmPwdExpirationTime` and `ms-Mcs-AdmPwd` attributes of computer accounts to `SELF` built-in account. This is required so as the machine could update password and expiration timestamp of own managed local Administrator password
- Add `CONTROL_ACCESS` permission on `ms-Mcs-AdmPwd` attribute of computer accounts to group or user that shall be allowed to read password of managed local Administrator account on managed computers
- Add `Write` permission on `ms-Mcs-AdmPwdExpirationTime` attribute of computer accounts to a group or user that shall be allowed to force password reset for managed local Administrator account on managed computers

### 6.2.1 Remove All Extended rights permission

This task will be performed using Powershell module AdmPwd.PS and cmdlet Find-AdmPwdExtendedRights. Run the following commands in Powershell window:

```
Import-module AdmPwd.PS
```

```
Find-AdmPwdExtendedRights -Identity <name of OU on which you want to delegate the permissions>
```

This command lists all containers that have CONTROL_ACCESS permission in their ACL, along with holders of the permission.

Repeat this procedure for any additional containers that contain computer accounts that are in scope of the solution and are not subcontainers of already processed containers

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

This task will be performed using Powershell module AdmPwd.PS and cmdlet `Set-AdmPwdComputerSelfPermission`. Run the following commands in Powershell window:

```
Import-module AdmPwd.PS
```

```
Set-AdmPwdComputerSelfPermission -Identity <name of OU on which you want to delegate the permissions>
```

Repeat this procedure for any additional OUs that contain computer accounts that are in scope of the solution and are not subcontainers of already processed containers

### 6.2.3 Add CONTROL_ACCESS permission to ms-Mcs-AdmPwd attribute

This task will be performed using Powershell module AdmPwd.PS and cmdlet `Set-AdmPwdReadPasswordPermission`. Run the following commands in Powershell window:

```
Import-module AdmPwd.PS
```

```
Set-AdmPwdReadPasswordPermission -Identity <name of OU on which you want to delegate the permissions> -AllowedPrincipals <identification of users/groups that should be allowed to read password>
```

Repeat this procedure for any additional OUs that contain computer accounts that are in scope of the solution and are not subcontainers of already processed containers

### 6.2.4 Add Write permission to ms-Mcs-AdmPwdExpirationTime attribute

This task will be performed using Powershell module AdmPwd.PS and cmdlet `Set-AdmPwdResetPasswordPermission`. Run the following commands in Powershell window:

```
Import-module AdmPwd.PS
```

```
Set-AdmPwdResetPasswordPermission -Identity <name of OU on which you want to delegate the permissions> -AllowedPrincipals <identification of users/groups that should be allowed to reset password>
```

Repeat this procedure for any additional OUs that contain computer accounts that are in scope of the solution and are not subcontainers of already processed containers

## 6.3 Installation of CSE

Solution supports unattended installation. Supported command lines below:

`msiexec /q /i <path>\LAPS.<platform>.msi` – installs just CSE

`msiexec /q /i <path>\LAPS.<platform>.msi ADDLOCAL=<comma separated list of feature IDs>` - installs just specified features

For feature ID's, see 5.4

## 6.4 Setup of auditing of password reads

This task can be accomplished via Set-AdmPwdAuditing cmdlet:

```
Import-module AdmPwd.PS
```

```
Set-AdmPwdAuditing -Identity:<identification of OU where are located computers you need to set audit for> -AuditedPrincipals:<list of security principals to audit>
```

## 6.5 Creation of custom admin account during CSE setup

MSI based setup is capable of creation of custom admin account during installation of CSE. When this feature is enabled, custom admin account is made member of local Administrators group and receives complex random password; this password is not reported anywhere. This makes the newly created admin password ready to be managed by the solution – during next GPO refresh, solution creates new password according to configured criteria and reports password to AD.

Feature is enabled via property CUSTOMADMINNAME from command line as follows:

```
msiexec /q /i <path>\LAPS.<platform>.msi CUSTOMADMINNAME=<name of custom local admin account>
```

Alternatively, this property can be set via MST file.

# 7 Dependencies

## 7.1 CSE

CSE is native C++ code compiled with Visual C++ 2013. Compiled statically with C+ runtime library, so installation of Visual C++ Redistributable is not necessary

## 7.2 Management tools

Management tools rely on .NET Framework 4 runtime, so you need to have .NET Framework 4 installed on machines where you want to use management UI (fat client and/or PowerShell module)

Note: When importing the PowerShell module in PowerShell 2.0, you may need to create/edit powershell.exe.config[i] file to allow loading of assemblies compiled for .NET Framework 4 runtime.

Sample content of file below:

```
<?xml version="1.0"?>
<configuration>
    <startup useLegacyV2RuntimeActivationPolicy="true">
        <supportedRuntime version="v4.0.30319"/>
        <supportedRuntime version="v2.0.50727"/>
    </startup>
</configuration>
```

[i] For details, see https://msdn.microsoft.com/en-us/library/w4atty68(v=vs.110).aspx