ActivePasswords Introduction

ActivePasswords is a compact, lightweight and powerful customizable Windows password filter. ActivePasswords makes it possible to manage multiple password complexity policies for a selection of Active Directory security groups and/or organizational units.

30-day trial period and Licensing

ActivePasswords will be fully functional for 30 days after installation if no license key is entered. After this period ActivePasswords will stop functioning and your servers will continue to work like they did before installing ActivePasswords. Pricing is based on the number of enabled and targeted Active Directory users and is subscription based. A subscription gives you usage rights, updates and e-mail support for 1 year.

Contents of the ActivePasswords.zip archive

ADMX (folder)	This folder contains the custom policy template definitions for ActivePasswords (admx/adml files).
wsap.dll (in MSI)	Password filter DLL that contains functionality called by Windows when an Active Directory user's password is changed.
pcr.exe (in MSI)	Password Change Request client utility.
wsapmailer.exe (in MSI)	Password Reminder mail service.
Readme.pdf	The file you are reading now.
Agreement.pdf	End user license agreement (EULA)

Prepare your Domain

It is a requirement to install ActivePasswords *on every domain controller* (DC) in your Active Directory domain (AD). This is necessary because password changes are sent to one random DC near the user that initiated the password change request. You achieve this by simply running the silent installer ActivePasswords.msi on all DC's. A manual reboot is necessary to complete the setup. The setup will NOT do this automatically.

The installer and restart initialization perform the following actions:

- Copy wsap.dll to %windir%\system32
- Add wsap to the registry value HKLM\SYSTEM\CurrentControlSet\Control\Lsa\Notification Packages
- Copy ActivePasswords.admx and the related .adml file to the central GP template store
- Create %windir%\wsap and netlogon\wsap and adjust folder permissions

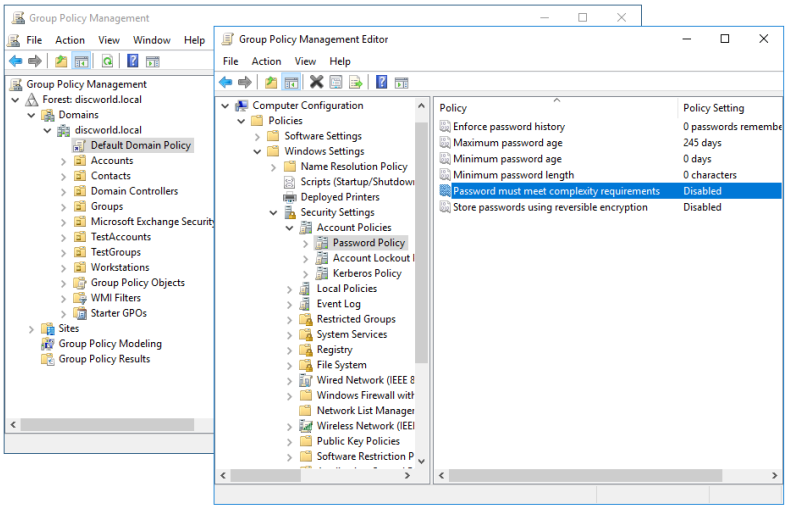
ActivePasswords Folders

%windir%\wsap	
\setup	Additional setup files/templates
public	Symlink to NETLOGON\wsap\public (e.g. C:\Windows\SYSVOL\domain\scripts\wsap\public). Public folder.
system	Symlink to NETLOGON\wsap\system (e.g. C:\Windows\SYSVOL\domain\scripts\wsap\system). Private secured folder.
\idh	Secure one-way salted password hashes. Example validation algorithm in .NET is available on request so you may use it in your own authentication software.
\idp	Contains optional IDP files; highly sensitive data IF used (see notes)
\log	Log files for each domain controller (%computername%.log)
\mail	Contains wsapmailer.exe plus small configuration and data files.
\words	Contains optional 'forbidden word' dictionaries

Due to UAC filtering, even an administrator will get the "access denied" message when trying to access the WSAP system folder with Windows Explorer. Only the system account and elevated administrators have access by default. Windows UAC filters the administrator token by default (except for the build-in Administrator account). You can either run a file browser application like Total Commander elevated as administrator or add specific named administrator accounts to the access permissions list. Easiest is going to the local SYSVOL folder and trying to access the wsap\system folder (e.g. C:\Windows\SYSVOL\domain\scripts\wsap\system).

Configuration

Step 1: Set the Windows default password settings

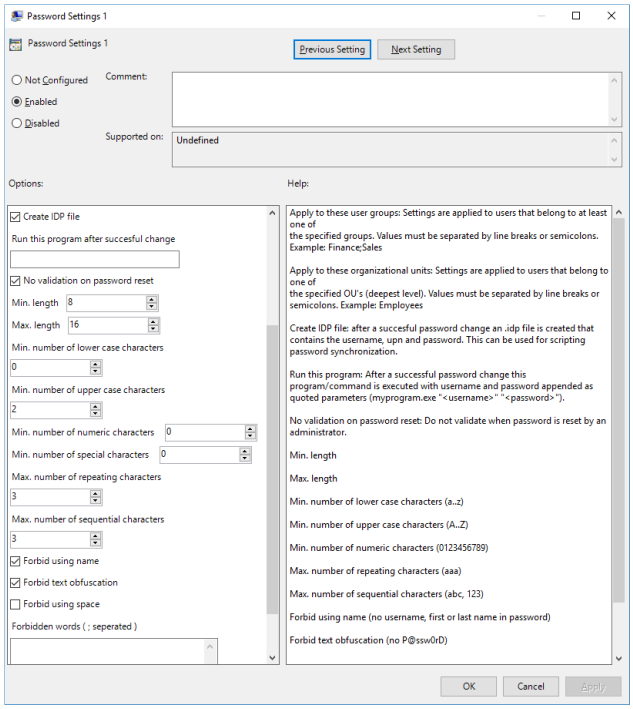


Open the Group Policy Management editor (gpmc.msc) and navigate to Forest/Domains/[DOMAIN]/Group Policy Objects and edit the "Default Domain Policy" object (also linked at the root of your domain). Go to "Computer Configuration/Policies/Windows Settings/Security Settings/Account Policies/Password Policy". Set the history and age options to your liking and *make sure to set "Password must meet complexity requirements" to **Disabled** and "Minimum password length" to **8** (or any other minimum requirement you wish to enforce)*. You will not use the default Windows complexity requirements because ActivePasswords is more flexible and powerful and replaces this functionality. The setting *"Minimum password length"* is only used in situations where no ActivePasswords policy is found or when the license has expired.

If you have configured any 'fine-grained' password settings, make sure to remove any targeted users and/or groups. These settings can be found in the Active Directory Administrative Center (dsac.exe). ActivePasswords will create its own fine-grained password settings which are all administrated through one group policy. It also allows you to target organizational units.

Test these changes by resetting a user password from Active Directory Users and Computers (dsa.msc). A very simple password should be allowed at this moment: the default Windows settings cannot interfere anymore with ActivePasswords settings. Of course, we will now make an ActivePasswords policy to force the use of strong passwords.

Step 2: Create ActivePasswords policy



Navigate to the organizational unit (OU) that contains *your **domain controllers** (not your user accounts!)*. Right click on the OU and choose “Create a GPO in this domain, and Link it here...”. Name the policy ActivePasswords. Right click on the newly created GPO and choose “Edit”. Navigate to “Computer Configuration/Administrative Templates/ActivePasswords”.

First, you need to setup some Common Settings: (next page)

Common Settings

Title	Registry Value	Description
License	License	Copy and paste the license you received (the entire line). Leave empty for trial mode. A license has this format: [Company;Domain;Amount;EndDateYyyymmdd;Hash].
Disable inactive user accounts (days)	DisableInactiveAccounts	Accounts with a last-logon-date of at least the specified number of days ago will be automatically disabled. A value of 0 means the feature is disabled. Suggested value: 45
Skip these users	SkipUsers	No log entry is created for and no password policy is applied to the specified users.
Skip these groups	SkipGroups	No log entry is created for and no password policy is applied to the specified groups.

You can configure up to 8 Password Settings for different selections of Active Directory groups. To create one, double click “Password Settings 1” and click “Enabled”. A security groups is specified by entering the *group name* (sAMAccountName attribute). Settings:

Password Settings 1-8

Title	Registry Value	Description
Apply to these user groups	UserGroups	Settings are applied to users that belong to at least one of the specified groups (**sAMAccountName**). Values must be separated by line breaks or semicolons. Example: Finance;Sales
Apply to these organizational units	UserOus	Settings are applied to users that are located in or under one of the specified OU’s (**use ‘path notation’**) to one of the specified OU's (deepest level name). Values must be separated by line breaks or semicolons. Example: /Accounts/Employees
Create IDP file	CreateIDP	After a successful password change an .idp file is created that contains the username, UPN and password. This can be used for scripting password synchronization.
Run this program after successful change	NotifyProgram	After a successful password change this program/command is executed with username and password appended as quoted parameters (myprogram.exe "<username>" "<password>").
Base64-encode program parameters	NotifyProgramBase64	The username and password parameters of the specified program will be base64-encoded
No validation on password reset	NoValidateOnReset	Do not validate when password is reset by an administrator.
HaveIBeenPwned check enabled	PwnedEnabled	Use the HaveIBeenPwned.com web service to check if the password is known to be compromised. *The password will NOT be checked during password reset events.*
Min. length	MinLength	
Max. length	MaxLength	
Min. number of words	MinWords	Password must contain at least the specified number of words (1 or more spaces act as a word separator).
Min. number of lower case characters	LowerCase	a..z

Min. number of upper case characters	UpperCase	A..Z
Min. number of numeric characters	Number	0123456789
Min. number of special characters	Special	~!@#$%^&*()_+{}|:?`-=[]\;',./<> "
Min. number of character categories	CharCategories	Password must contain at least the specified number of character categories (lower case, upper case, numeric, special). This setting has no effect if any of the 'min. number of x characters' policies are configured because they conflict with each other.
Max. number of repeating characters	MaxRepeating	aaa
Max. number of sequential characters	MaxSequential	abc, 123
Forbid similar passwords when changing	NoSimilar	Forbid similar passwords when changing: new password must be different enough from previous 8 passwords.
Forbid using name	NoName	No username, first or last name in password
Forbid text obfuscation	NoObfuscation	No P@ssw0rD
Forbid using space	NoSpace	No ‘ ‘
Forbid using vowel	NoVowel	No vowels are allowed (AEIOU)
Only allow these characters	AllowedChars	Password must contain only a selection of these specific characters (e.g. ABCDEFGHIJKLMNOPQRSTUVWXYZ: only upper-case letters are allowed)
Forbid these characters	ForbiddenChars	Password must not contain any of these characters (e.g. _#)
Forbid these characters at start	ForbiddenStartChars	Password must not start with any of these characters (e.g. 0123456789)
Forbidden words*1	Forbidden	Custom prohibited words like company;department;brand (the password myCompany will not be accepted). Words must be separated by line breaks or semicolons.
Forbidden words file	ForbiddenFile	Enter the name of a file that contains a list of forbidden words (must be located at %windir%\wsap\system\words). Words must be separated by line breaks.
Forbidden words threshold %	ForbiddenThreshold	Specify how much of the entire password must consist of blacklisted words before the password is rejected. 0 means any found forbidden word will block the password change request. Example: 60
RegEx	RegEx	'abC' will pass '[a-z]b[A-Z]'; 'abc' will not
RulesExplanation	RulesText	This text will be displayed to the user when the password almost expires (only displayed when the optional pwcr.exe utility runs on the users workstation). You can use \n to add extra new lines. Use variables $givenName, $sn and $displayName for personalization purposes. The GPO editor does not allow empty lines; the workaround is to add a space character. Allowed variables: *$sAMAccountName,*

		$userPrincipalName, $cn, $displayName, $sn, $givenName, $passwordExpiryDate
User must change password after (days)	MaxPasswordAgeDays	Suggested value: 365
User cannot change password within (days)	MinPasswordAgeDays	Suggested value: 0
Number of exact passwords remembered	PasswordHistory	Suggested value: 5
Number of failed logon attempts allowed	LockoutThreshold	Suggested value: 10
Reset failed logon attempts count after (mins)	LockoutObservationWindowMin	Suggested value: 1
Account will be locked out for a duration of (mins)	LockoutDurationMin	Suggested value: 1

Suggestion: First specify a global password policy in 'Password Settings 1' which targets all users, then specify password settings for specific groups/organizational units in Password Settings 2..8. Processing order is 8..1 and stops at first match, which means only one setting will be applied to each user.

After closing the Group Policy editor, you can run the command *gpupdate* to immediately update the configuration of the domain controller. The default Group Policy refresh rate on Domain Controllers is 5 minutes.

About Have I Been Pwned?

Please visit the website www.haveibeenpwned.com for more information. If the "HaveIBeenPwned" check is enabled, a new password will be checked for exposure with this web service. A partial password hash will be sent in a HTTPS request. Because of web service performance issues and rate limits the password will NOT be checked during reset events (script or administrator initiated).

Password expiration warning

Optionally you can run the small tool PCR (short for PasswordChangeRequest) on client computers that periodically warns the end user starting 8 days before the password expires to change the password. The tool can be found in \\domain.local\netlogon\wsap\public\pcr.exe . One way to start it is from a login script by adding the command [start \\domain.local\netlogon\wsap\public\pcr.exe]. For testing purposes, use the /test <username> parameter which will cause pcr.exe to display the configured message for the current or specified user.

Password expiration e-mail notification

Included is a small utility that can query AD for users whose password is about to expire and send out reminder e-mails based on templates. To configure:

1. Choose, and logon to the Windows domain controller you want to use for this feature.
2. Go to folder C:\Windows\wsap\system\mail, open the wsapmailer.cfg file in a text editor and adjust the values to your used e-mail environment.
3. The contents of the reminder e-mail are determined by the ActivePassword's 'Rules' policy. This can be overruled by creating one or more text files in the folder C:\Windows\wsap\system\mail named "Password Settings X.txt" where X is replaced with the number 1..8. The file contents can optionally be HTML formatted (<html>…</html>). You can use these specific place holders/variables in the text: *$sAMAccountName, $userPrincipalName, $cn, $displayName, $sn, $givenName, $passwordExpiryDate*
4. Test connection settings by opening a console (cmd) and run C:\Windows\wsap\system\mail \wsapmailer.exe /test <username> . It will try to send an e-mail to the specified AD account (using the mail or userPrincipalName user attribute).
5. Go to Scheduled Tasks and enable the preconfigured WSAPMailer task, which will run every 4 hours. You could also start it manually by running C:\Windows\wsap\system\mail\wsapmailer.exe /mail . To see who *would* be mailed, run wsapmailer.exe /whatif .

Troubleshooting

If things do not work the way they should, these suggestions may help you:

- The ActivePasswords computer group policy must be applied to the OU that contains *your domain controllers*
- The ActivePasswords policy settings must include the targeted active directory user group(s) (eg mygroup1;mygroup2) or organizational unit(s)
- Use the **wsaptest.exe** command line utility to test if a given password is accepted or not for the specified user and what policies are applied. The password is not changed, so safe to use with real usernames. *wsaptest.exe /p <username> <password>*
- You can disable the setting ' No validation on password reset ' to make testing as administrator easier (reset user passwords as admin from dsa.msc and see if the password is accepted or not)
- DC's must be rebooted at least once, and it does not hurt to run *gpupdate* on each DC after changing policies when testing
- Open the text file c:\windows\wsap\system\logs\%computername%.log . Does it contain an invalid license message or any other indications of what goes on?
- Check the registry key HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa . It's value should contain the text 'wsap'.
- On password change an event is logged that is viewable in eventvwr and a log entry is added to the file c:\windows\wsap\system\logs\%computername%.log .
- Did you adjust the default Windows password filter as mentioned in Configuration? If any of the installed password filters do not accept the new password, the change will not be accepted and functionality in WSAP will not be called.

Logging

ActivePasswords keeps a log file in *%windir%\wsap\system\logs\%computername%.log* . This log file contains information about password change events in the format *datetime<tab>Password Changed<tab>Password of user $Username has been changed ($PrincipalName; $DisplayName)* . Denied password change events are logged as *Requested password change for user $Username ($PrincipalName; $DisplayName) was not accepted because it did not meet these requirements: $Details*

ActivePasswords also creates events in the Windows Application event log. The following events are logged with source "ActivePasswords":

Event ID	Level	Text
1	Informational	Password Changed<tab>Password of user $Username has been changed ($PrincipalName; $DisplayName)
2	Informational	Requested password change for user $Username ($PrincipalName; $DisplayName) was not accepted because it did not meet these requirements: $Details
11	Error	Invalid or expired ActivePasswords license on server $Computer

Uninstalling ActivePasswords

Simply run *appwiz.cpl* ("Uninstall or change a program"), select ActivePasswords and click on Uninstall. A reboot is necessary.