

# **Report: Audit and Enforcement of Domain Password and Account Policies**

## **1. Introduction**

In any Active Directory (AD) environment, enforcing strong password and account policies is critical for maintaining the integrity and security of user authentication. Weak or outdated password practices are among the most exploited vulnerabilities in enterprise networks. Therefore, regular audits of password settings, account attributes, and enforcement mechanisms help organizations align with security best practices and compliance requirements.

This task set focused on evaluating and testing key password and account policy controls within a domain environment. By examining the Default Domain Password Policy, querying specific user account attributes such as password expiration and age, and testing policy enforcement through manual password resets, we aimed to ensure that the domain configuration effectively supports secure user authentication and account lifecycle management.

## **2. Objective**

The primary objective of this task was to review and validate the current password and account policy configurations applied at the domain level. This included identifying parameters such as minimum password length, complexity requirements, and expiration duration, which are crucial for protecting against brute-force attacks and credential misuse. By documenting these settings, we ensured that the environment enforces adequate password hygiene across all domain users.

In addition, this task aimed to detect deviations from standard policies, such as accounts with the "Password Never Expires" attribute enabled or those with passwords older than 90 days. Identifying and addressing such accounts reduces the risk of stale credentials being exploited by attackers. The final step—resetting a test account's password—was conducted to verify that the domain enforces its password policy in real-time, ensuring operational effectiveness.

## **Key goals include:**

- **Ensure Strong Password Policy Enforcement**

To reduce the risk of unauthorized access, it's essential that password policies enforce minimum length, complexity, and expiration. This ensures users create secure passwords that are regularly updated.

- **Identify and Eliminate Security Loopholes**

Accounts with attributes like "Password Never Expires" can pose long-term security risks. Identifying and correcting such configurations prevents attackers from exploiting stale or static credentials.

- **Monitor Password Age for Compliance**

Checking for users with passwords older than 90 days ensures adherence to organizational or compliance requirements. Regular password changes help limit the window of opportunity for attackers.

- **Verify Real-Time Policy Enforcement**

Testing policy enforcement by resetting a test account's password helps confirm that password complexity and expiration rules are actively being applied, not just configured.

- **Maintain Domain Security Hygiene**

Regular audits of password and account policies support overall AD hygiene, ensuring that inactive or misconfigured accounts do not become entry points for attacks or privilege escalation.

## **3. Methodology**

The Methodology section outlines the approach used to perform and validate each task related to auditing and enforcing password and account policies in an Active Directory environment. It is structured in multiple phases, with each phase comprising a logical set of steps. These steps were performed sequentially using PowerShell commands and manual validation techniques to assess the current state of policy configuration and its enforcement.

Each phase plays a crucial role in ensuring that password policies are not only set correctly but are also effectively applied across all user accounts. Screenshots and proof of concept (POC) are included where applicable to support findings and demonstrate execution. This structured approach ensures thoroughness and repeatability of the audit process.

### 3.1 Password Policy Review and Validation

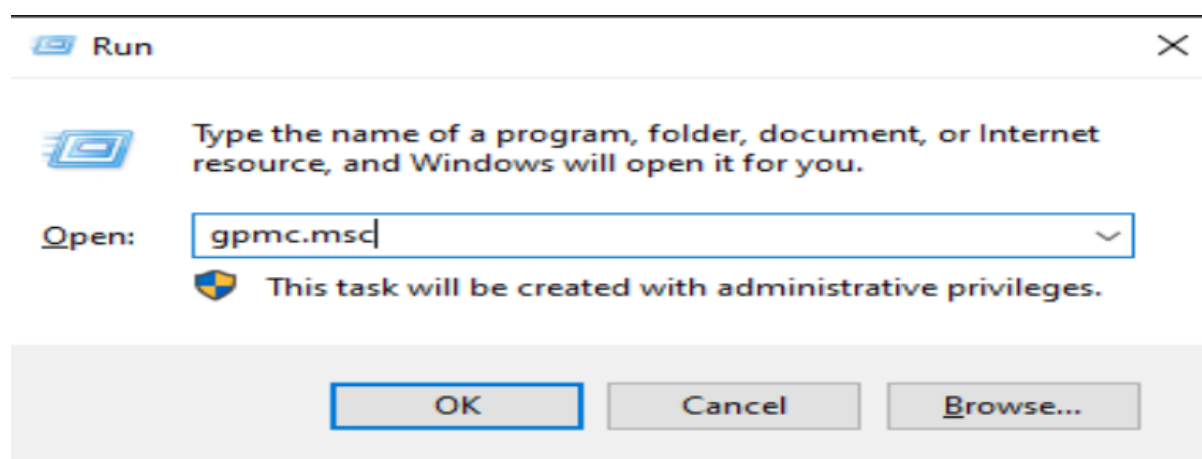
In this phase, the focus was on reviewing and verifying the Default Domain Password Policy. This helps establish a baseline understanding of how secure the existing configuration is. Parameters such as minimum password length, complexity requirements, history, and maximum password age were examined. We also verified the policy using PowerShell to ensure it matches the documented values and confirm that Group Policy settings are being properly applied to the domain.

- **Review Password Policy**

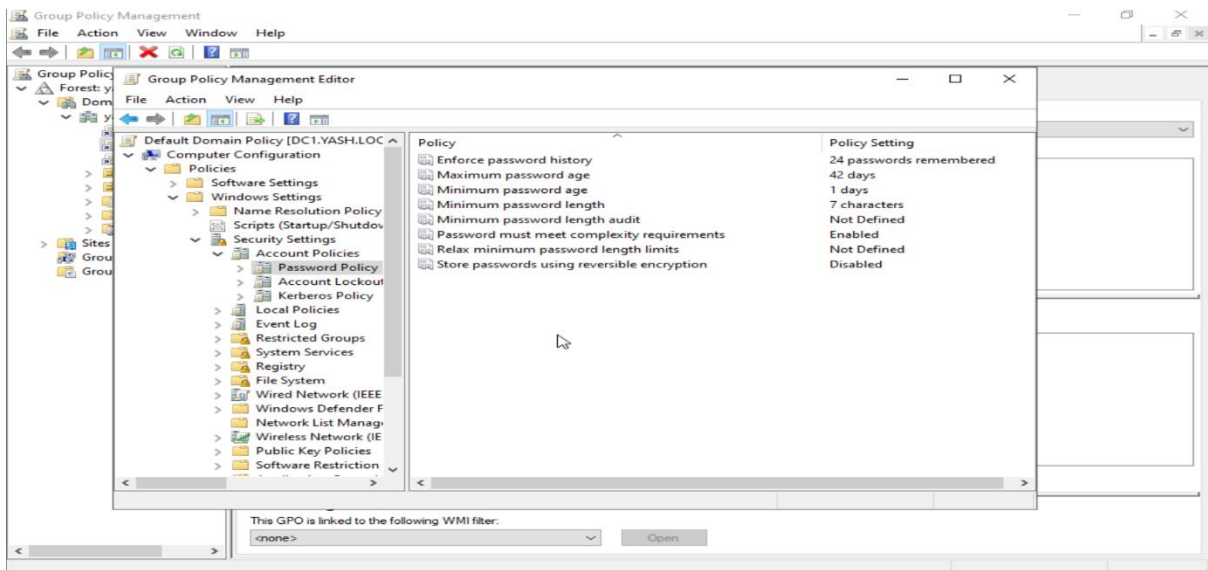
Using the Group Policy Management Console (GPMC), the Default Domain Policy was reviewed. We navigated to: **Computer Configuration > Policies > Windows Settings > Security Settings > Account Policies > Password Policy**.

Here, key settings like password complexity, minimum/maximum age, and history were analyzed.

(WIN+R - Write gpmc.msc to open Group Policy Management Console)



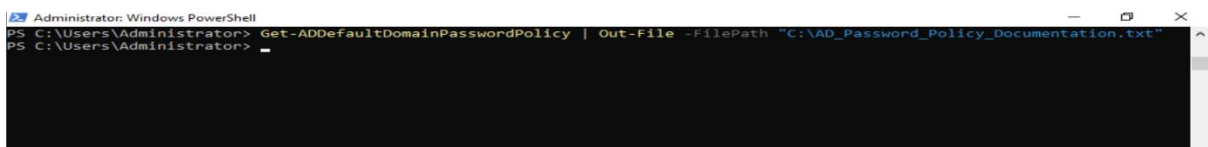
(This step proves that domain-level policies are configured with specific security requirements.)



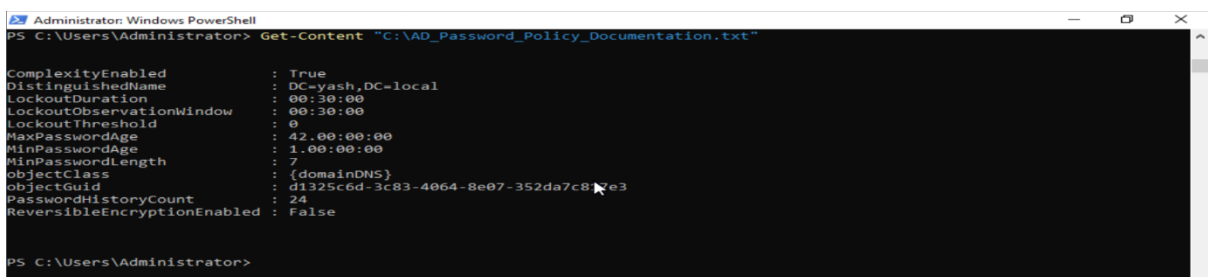
- **Document the Password Policy**

The values observed in GPMC were manually documented. This included noting down the current settings for each policy element such as length, complexity, and expiration. Documentation ensures clarity on what security measures are expected and helps detect deviations during auditing.

(Capturing and listing these settings allows for policy tracking and change comparison.)



(Documented content of Default domain Policy saved in this file.)



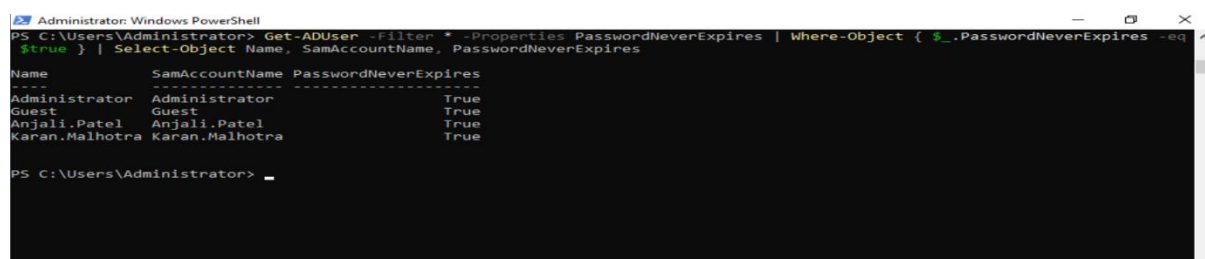
### 3.2 Identify Risky Account Configurations

This phase focuses on detecting potentially insecure user account configurations. The goal is to find accounts with either outdated credentials or bypassed security controls, such as the "Password Never Expires" attribute. Auditing such accounts helps prevent long-term credential exposure and maintain password hygiene.

- **Identify Users with “Password Never Expires”**

A PowerShell query was used to list all accounts where PasswordNeverExpires is set to True. These accounts may not be forced to change passwords regularly, posing a security risk. Reviewing and correcting such accounts is important for maintaining secure credential practices.

(Shows which accounts bypass password expiration.)



```
Administrator: Windows PowerShell
PS C:\Users\Administrator> Get-ADUser -Filter * -Properties PasswordNeverExpires | Where-Object { $_.PasswordNeverExpires -eq $true } | Select-Object Name, SamAccountName, PasswordNeverExpires

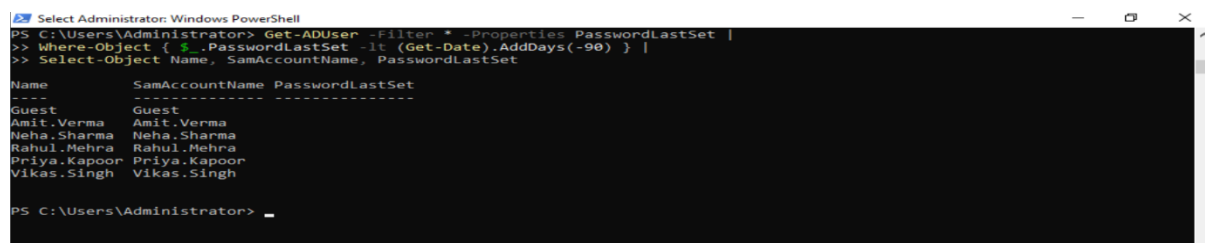
Name                SamAccountName PasswordNeverExpires
----                -
Administrator      Administrator      True
Guest               Guest              True
Anjali.Patel        Anjali.Patel      True
Karan.Malhotra      Karan.Malhotra    True

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

- **Find Users with Passwords Older Than 90 Days**

This step detects accounts that haven't changed passwords in over 90 days using GET-ADUser. These accounts may be vulnerable due to prolonged password reuse or potential compromise. This step helps enforce periodic password updates across the domain.

(Identifies stale credentials and supports enforcing security compliance timelines.)



```
Select Administrator: Windows PowerShell
PS C:\Users\Administrator> Get-ADUser -Filter * -Properties PasswordLastSet |
>> Where-Object { $_.PasswordLastSet -lt (Get-Date).AddDays(-90) } |
>> Select-Object Name, SamAccountName, PasswordLastSet

Name                SamAccountName PasswordLastSet
----                -
Guest               Guest
Amit.Verma          Amit.Verma
Neha.Sharma         Neha.Sharma
Rahul.Mehra         Rahul.Mehra
Priya.Kapoor        Priya.Kapoor
Vikas.Singh         Vikas.Singh

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

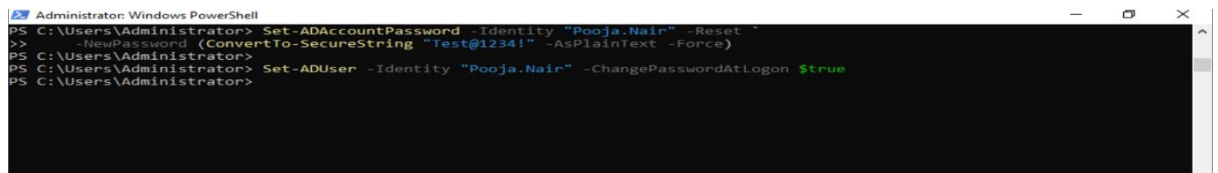
### 3.3 Password Reset Enforcement Test

This final phase verifies whether the password policy is actually being enforced when users change or reset their passwords. A test account was used to simulate a password reset and confirm the enforcement of complexity and history rules. This validates that policy settings aren't just configured but also actively enforced by the domain.

- **Reset Password for a Test Account**

The test account password was reset manually or via PowerShell to a new value. This step is used to trigger policy validation during the reset. Any violations were flagged by the system if enforcement was active.

(Demonstrates that reset operation is subject to domain-level password policy.)



```
Administrator: Windows PowerShell
PS C:\Users\Administrator> Set-ADAccountPassword -Identity "Pooja.Nair" -Reset "
>> -NewPassword (ConvertTo-SecureString "Test@1234!" -AsPlainText -Force)
PS C:\Users\Administrator>
PS C:\Users\Administrator> Set-ADUser -Identity "Pooja.Nair" -ChangePasswordAtLogon $true
PS C:\Users\Administrator>
```

- **Verify Password Reset and Policy Enforcement**

After resetting the password, login was attempted using the new password to confirm success. If the password did not meet the policy, an error would appear, showing enforcement is working. This also confirms that the account can function post-reset and that there are no hidden misconfigurations.

(Verifies that password reset obeys the domain's security policies and the account functions correctly.)

**Pooja.Nair Properties** ? X

Published Certificates	Member Of	Password Replication	Dial-in	Object
Security	Environment	Sessions	Remote control	
Remote Desktop	Services Profile	COM+	Attribute Editor	
General	Address	Account	Profile	Telephones
				Organization

User logon name:

User logon name (pre-Windows 2000):

☐ Unlock account

Account options:

☒ User must change password at next logon  
☐ User cannot change password  
☐ Password never expires  
☐ Store password using reversible encryption

Account expires

☒ Never  
☐ End of:

Security Group... Designated administrato...  
 User Disabled due to inactivit...  
 User

## 4. Results and Findings

This section summarizes the outcomes of the audit and testing process. It highlights the key observations discovered during the review of password and account policies.

- **Password Policy is Enforced Correctly**

The Default Domain Password Policy includes strong settings such as minimum length, complexity, and expiration. PowerShell confirmed these settings are actively enforced on the domain.

- **Policy Matches GUI and CLI Outputs**

Group Policy settings seen via GPMC matched the results from Get-ADDefaultDomainPasswordPolicy. This confirms consistency between administrative tools and PowerShell output.

- **Accounts with “Password Never Expires” Found**

Several user accounts were found to have the “Password Never Expires” flag enabled. These may bypass security checks and should be reviewed for necessity.

- **Aged Passwords Detected**

Some user accounts had passwords older than 90 days. This indicates a lapse in regular password updates, which can weaken account security over time.

- **Password Reset Policy Was Enforced**

The test account reset process required a complex password, confirming enforcement of domain password rules. An invalid password triggered a policy violation error, as expected.

## 5. Recommendations

These actionable steps aim to strengthen domain password policies and account management practices based on the findings.

- **Disable “Password Never Expires” Where Not Needed**

Review and remove this setting from all accounts except critical service accounts that require it. Use alternative protections like managed service accounts.

- **Enforce Regular Password Changes**

Ensure all users comply with the 90-day password change requirement. Automate reminders or enforce this via policy enforcement.

- **Perform Periodic Account Audits**

Schedule monthly or quarterly reviews to identify non-compliant accounts, password age issues, or misconfigurations.



- **Monitor Policy Enforcement with Scripts**

Use scheduled PowerShell scripts to regularly check for accounts violating password policies. This adds automation and reduces manual effort.

- **Educate Users on Password Best Practices**

Conduct awareness training to encourage users to follow secure password habits, avoid reuse, and recognize enforcement messages.

## **6. Conclusion**

The audit of the domain's password and account policies revealed a generally secure configuration aligned with industry best practices. The Default Domain Password Policy enforces key parameters such as minimum password length, complexity, and expiration, which are crucial for protecting user accounts from unauthorized access. Through PowerShell validation and practical testing, it was confirmed that these policies are not only configured correctly but also actively enforced during user interactions, such as password resets.

However, the discovery of accounts with the "Password Never Expires" attribute and passwords older than 90 days indicates areas that require administrative attention. These configurations, if left unchecked, could become security vulnerabilities over time. Regular audits, automated compliance checks, and user education are essential to maintaining a strong security posture. By addressing the identified issues and implementing the recommended actions, the organization can significantly reduce the risk of credential-based attacks and enhance the overall resilience of its Active Directory environment.