**Logic document for finding blank password**

Methods Available:-

1. We can manually login with blank password for the target user and if the login is successful we consider the user has blank password, this method creates the failure audit event log and may lock the user ID.
2. We can try resetting the user’s password by defining current password as blank and if the password is changed then we can conclude the password of the said user is blank. This would also create the password change success/failure event which is not acceptable.
3. We can read the USERAccountControl attribute and based on certain values we can conclude the password is blank. This method is not reliable and could not be able to capture users with blank password in certain scenarios.
4. Compare the NT and LM hashes of AD users in AD. This method uses third party PowerShell Cmdlet and then provide the users with blank password however it has below disadvantages.

* Pre-requisite
* .Net 4.5.1.
* PowerShell 3.0 and above.
* VC++ 2017 Redistributable Run time package.
* Additional PowerShell Module
* Risks

1. We may need to upgrade .net framework in some system which may cause unexpected behaviour in partner's production applications.
2. We will need to download and install DSInternals module on DCs.
3. DSInternals disclaimer says that the methods used to access password hashes are not supported by Microsoft and may cause functional issues with AD database and DC’s functionality.
4. DSInternals disclaimer also says that loading this modules lowers the AD security.

* **Manual Steps to Use the Module**
* Load the DSinternals Module in PowerShell.
* Make sure the Pre-requisites are fulfilled.
* Run below script and output would be similar as below.

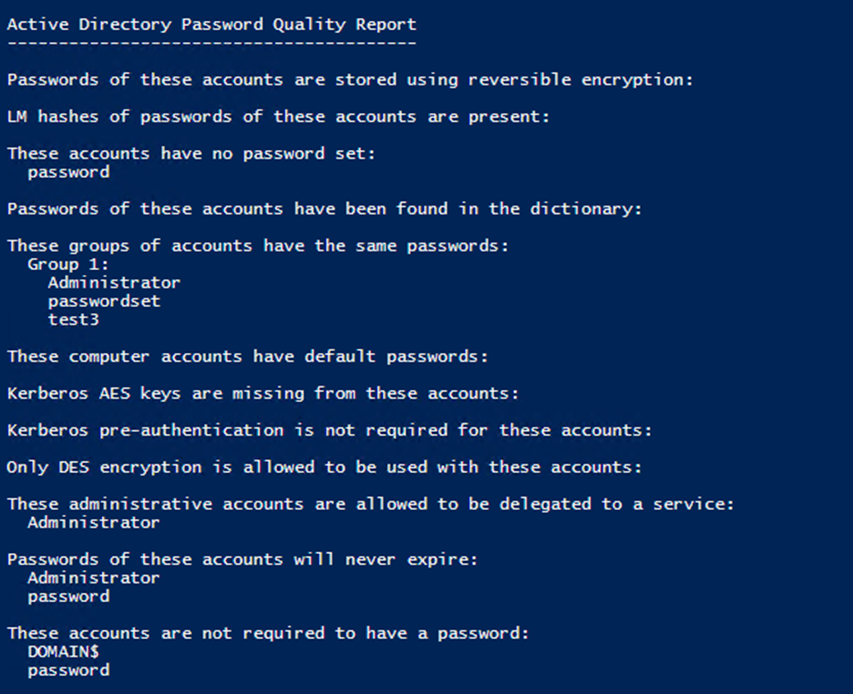
Import-Module C:\Users\administrator.CHILD\Desktop\DSInternals\DSInternals\DSInternals.psd1

$DC = "DC2"

$Domain = "dc=child,DC=domain,dc=local"

$results = Get-ADReplAccount -All -NamingContext $Domain -Server $DC |Test-PasswordQuality

$results



* **Initial Testing:-**
* Script has been tested in single and multiple DC scenario
* Script has been tested in 2012 OS as per the minimum PowerShell V3 dependency.
* Automation Expectations.
* This module review/compares the hashes of Active Directory user’s passwords and based on the same it gives the output currently.
* We need to review and understand the attached module.
  + 
* First we need to find the feasibity and time may require to complete below objectives.

1. Need to eliminate all the other functionalities of the module and get the output of only blank passwords i.e. where scripts outputs “These accounts have no passwords.”
2. Need to eliminate the dependencies of VC++ 2017 Redistributable Run time package and dot net 4.5.1
3. Need the module to be re-created as PowerShell script to deployed using portal

Below links have additional details on the module.

<http://woshub.com/auditing-users-password-strength-in-ad/>

<https://github.com/MichaelGrafnetter/DSInternals/releases>

<https://www.powershellgallery.com/packages/DSInternals/3.6>