Active Directory Administrative Center Enhancements

Applies To: Windows Server 2012

[Content in this topic that applies specifically to Windows Server 2012 R2 Preview is preliminary and subject to change in future releases.]

ADAC in Windows Server 2012 includes management features for the following:

- Active Directory Recycle Bin
- Fine-Grained Password Policy
- Windows PowerShell History Viewer

Active Directory Recycle Bin

Accidental deletion of Active Directory objects is a common occurrence for users of Active Directory Domain Services (AD DS) and Active Directory Lightweight Directory Services (AD LDS). In past versions of Windows Server, prior to Windows Server 2008 R2, one could recover accidentally deleted objects in Active Directory, but the solutions had their drawbacks.

In Windows Server 2008, you could use the Windows Server Backup feature and **ntdsutil** authoritative restore command to mark objects as authoritative to ensure that the restored data was replicated throughout the domain. The drawback to the authoritative restore solution was that it had to be performed in Directory Services Restore Mode (DSRM). During DSRM, the domain controller being restored had to remain offline. Therefore, it was not able to service client requests.

In Windows Server 2003 Active Directory and Windows Server 2008 AD DS, you could recover deleted Active Directory objects through tombstone reanimation. However, reanimated objects' link-valued attributes (for example, group memberships of user accounts) that were physically removed and non-link-valued attributes that were cleared were not recovered. Therefore, administrators could not rely on tombstone reanimation as the ultimate solution to accidental deletion of objects. For more information about tombstone reanimation, see Reanimating Active Directory Tombstone Objects.

Active Directory Recycle Bin, starting in Windows Server 2008 R2, builds on the existing tombstone reanimation infrastructure and enhances your ability to preserve and recover accidentally deleted Active Directory objects.

When you enable Active Directory Recycle Bin, all link-valued and non-link-valued attributes of the deleted Active Directory objects are preserved and the objects are restored in their entirety to the same consistent logical state that they were in immediately before deletion. For example, restored user accounts automatically regain all group memberships and corresponding access rights that they had immediately before deletion, within and across domains. Active Directory Recycle Bin works for both AD DS and AD LDS environments. For a detailed description of Active Directory Recycle Bin, see What's New in AD DS: Active Directory Recycle Bin.

What's new? In Windows Server 2012, the Active Directory Recycle Bin feature has been enhanced with a new graphical user interface for users to manage and restore deleted objects. Users can now visually locate a list of deleted objects and restore them to their original or desired locations.

If you plan to enable Active Directory Recycle Bin in Windows Server 2012, consider the following:

- By default, Active Directory Recycle Bin is disabled. To enable it, you must first raise the forest functional level of your AD DS or AD LDS environment to Windows Server 2008 R2 or higher. This in turn requires that all domain controllers in the forest or all servers that host instances of AD LDS configuration sets be running Windows Server 2008 R2 or higher.

- The process of enabling Active Directory Recycle Bin is irreversible. After you enable Active Directory Recycle Bin in your environment, you cannot disable it.

- To manage the Recycle Bin feature through a user interface, you must install the version of Active Directory Administrative Center in Windows Server 2012.

Note

You can use **Server Manager** to install Remote Server Administration Tools (RSAT) on Windows Serv
version of Active Directory Administrative Center to manage Recycle Bin through a user interface.

You can use RSAT on Windows® 8 computers to use the correct version of Active Directory Administr
through a user interface.

Active Directory Recycle Bin step-by-step

In the following steps, you will use ADAC to perform the following Active Directory Recycle Bin tasks in Windows Server 2012:

- Step 1: Raise the forest functional level

- Step 2: Enable Recycle Bin

- Step 3: Create test users, group and organizational unit

- Step 4: Restore deleted objects

Note

Membership in the Enterprise Admins group or equivalent permissions is required to perform the following step

Step 1: Raise the forest functional level

In this step, you will raise the forest functional level. You must first raise the functional level on the target forest to be Windows Server 2008 R2 at a minimum before you enable Active Directory Recycle Bin.

To raise the functional level on the target forest

1. Right click the Windows PowerShell icon, click **Run as Administrator** and type **dsac.exe** to open ADAC.
2. Click **Manage**, click **Add Navigation Nodes** and select the appropriate target domain in the **Add Navigation Nodes** dialog box and then click **OK**.
3. Click the target domain in the left navigation pane and in the **Tasks** pane, click **Raise the forest functional level**. Select a forest functional level that is at least Windows Server 2008 R2 or higher and then click **OK**.

Windows PowerShell equivalent commands

The following Windows PowerShell cmdlet or cmdlets perform the same function as the preceding procedure. Enter each cmdlet on a single line, even though they may appear word-wrapped across several lines here because of formatting constraints.

```
Set-ADForestMode –Identity contoso.com -ForestMode Windows2008R2Forest –
Confirm:$false
```

For the **–Identity** argument, specify the fully qualified DNS name.

Step 2: Enable Recycle Bin

In this step, you will enable the Recycle Bin to restore deleted objects in AD DS.

To enable Active Directory Recycle Bin in ADAC on the target domain

1. Right click the Windows PowerShell icon, click **Run as Administrator** and type **dsac.exe** to open ADAC.
2. Click **Manage**, click **Add Navigation Nodes** and select the appropriate target domain in the **Add Navigation Nodes** dialog box and then click **OK**.
3. In the **Tasks** pane, click **Enable Recycle Bin ...** in the **Tasks** pane, click **OK** on the warning message box, and then click **OK** to the refresh ADAC message.
4. Press F5 to refresh ADAC.

Windows PowerShell equivalent commands

The following Windows PowerShell cmdlet or cmdlets perform the same function as the preceding procedure. Enter each cmdlet on a single line, even though they may appear word-wrapped across several lines here because of formatting constraints.

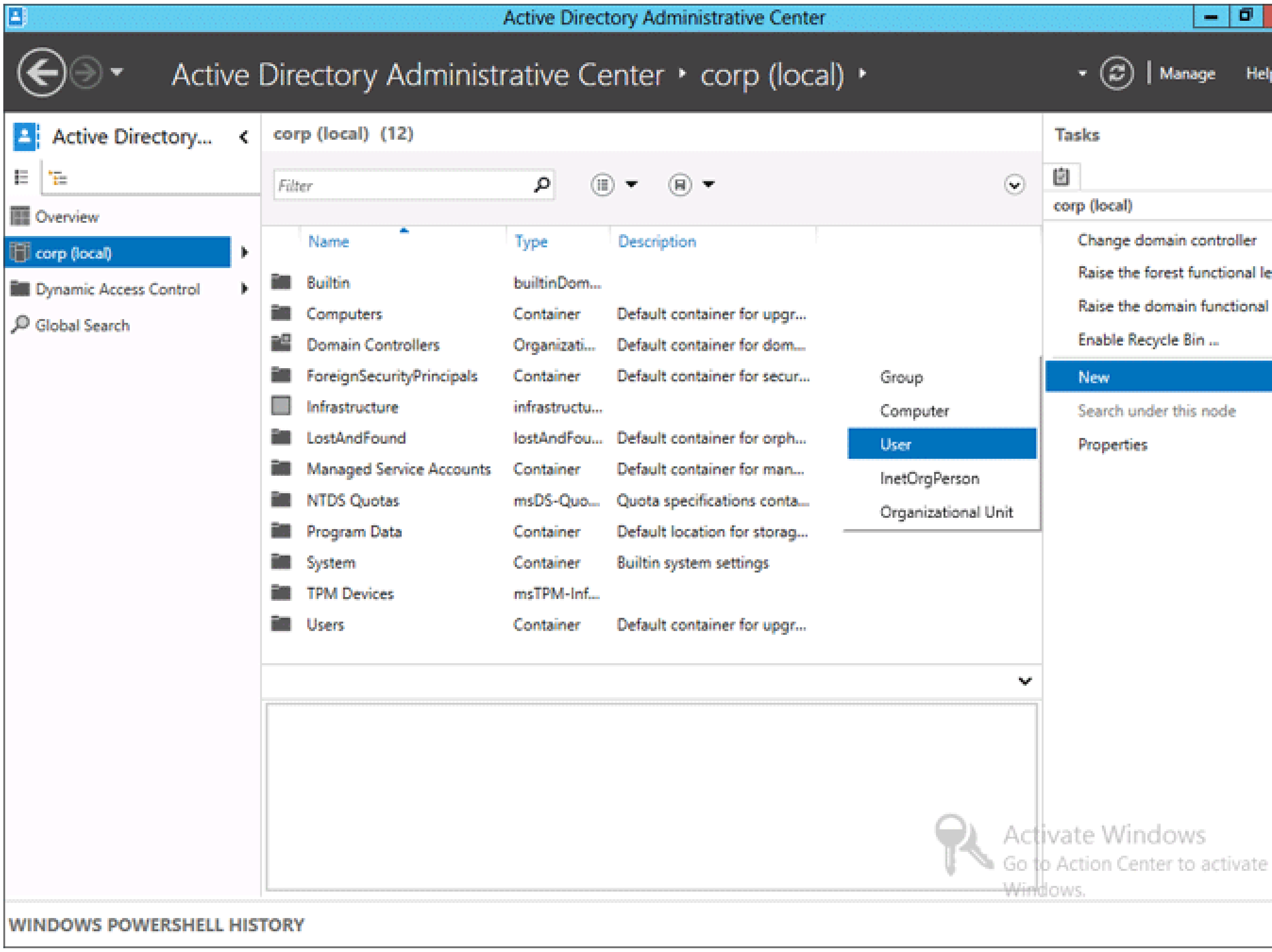
```
Enable-ADOptionalFeature –Identity 'CN=Recycle Bin Feature,CN=Optional
Features,CN=Directory Service,CN=Windows
NT,CN=Services,CN=Configuration,DC=contoso,DC=com' –Scope ForestOrConfigurationSet –
Target 'contoso.com'
```

Step 3: Create test users, group and organizational unit

In the following procedures, you will create two test users. You will then create a test group and add the test users to the group. In addition, you will create an OU.

To create test users

1. Right click the Windows PowerShell icon, click **Run as Administrator** and type **dsac.exe** to open ADAC.
2. Click **Manage**, click **Add Navigation Nodes** and select the appropriate target domain in the **Add Navigation Nodes** dialog box and then click **OK**.
3. In the **Tasks** pane, click **New** and then click **User**.



Name	Type	Description
Builtin	builtinDom...	
Computers	Container	Default container for upgr...
Domain Controllers	Organizati...	Default container for dom...
ForeignSecurityPrincipals	Container	Default container for secur...
Infrastructure	infrastructu...	
LostAndFound	lostAndFou...	Default container for orph...
Managed Service Accounts	Container	Default container for man...
NTDS Quotas	msDS-Quo...	Quota specifications conta...
Program Data	Container	Default location for storag...
System	Container	Builtin system settings
TPM Devices	msTPM-Inf...	
Users	Container	Default container for upgr...

4. Enter the following information under **Account** and then click OK:
   - Full name: test1
   - User SamAccountName logon: test1
   - Password: p@ssword1
   - Confirm password: p@ssword1
5. Repeat the previous steps to create a second user, test2.

To create a test group and add users to the group

1. Right click the Windows PowerShell icon, click **Run as Administrator** and type **dsac.exe** to open ADAC.
2. Click **Manage**, click **Add Navigation Nodes** and select the appropriate target domain in the **Add Navigation Nodes** dialog box and then click **OK**.
3. In the **Tasks** pane, click **New** and then click **Group**.
4. Enter the following information under **Group** and then click **OK**:
    - **Group name:group1**

5. Click **group1**, and then under the **Tasks** pane, click **Properties**.
6. Click **Members**, click **Add**, type **test1;test2**, and then click **OK**.

Windows PowerShell equivalent commands

The following Windows PowerShell cmdlet or cmdlets perform the same function as the preceding procedure. Enter each cmdlet on a single line, even though they may appear word-wrapped across several lines here because of formatting constraints.

```
Add-ADGroupMember -Identity group1 -Member test1
```

To create an organizational unit

1. Right click the Windows PowerShell icon, click **Run as Administrator** and type **dsac.exe** to open ADAC.
2. Click **Manage**, click **Add Navigation Nodes** and select the appropriate target domain in the **Add Navigation Nodes** dialog box and then click **OK**.
3. In the **Tasks** pane, click **New** and then click **Organizational Unit**.
4. Enter the following information under **Organizational Unit** and then click **OK**:
    - **NameOU1**

Windows PowerShell equivalent commands

The following Windows PowerShell cmdlet or cmdlets perform the same function as the preceding procedure. Enter each cmdlet on a single line, even though they may appear word-wrapped across several lines here because of formatting constraints.

```
1..2 | ForEach-Object {New-ADUser -SamAccountName test$_ -Name "test$_" –Path
"DC=fabrikam,DC=com" -AccountPassword (ConvertTo-SecureString -AsPlainText
"p@ssword1" -Force) -Enabled $true}
New-ADGroup -Name "group1" -SamAccountName group1 -GroupCategory Security -GroupScope
Global -DisplayName "group1"
New-ADOrganizationalUnit -Name OU1 -Path "DC=fabrikam,DC=com"
```

Step 4: Restore deleted objects

In the following procedures, you will restore deleted objects from the **Deleted Objects** container to their original location and to a different location.

To restore deleted objects to their original location

1. Right click the Windows PowerShell icon, click **Run as Administrator** and type **dsac.exe** to open ADAC.
2. Click **Manage**, click **Add Navigation Nodes** and select the appropriate target domain in the **Add Navigation Nodes** dialog box and then click **OK**.
3. Select users **test1** and **test2**, click **Delete** in the **Tasks** pane and then click **Yes** to confirm the deletion.

   Windows PowerShell equivalent commands

   The following Windows PowerShell cmdlet or cmdlets perform the same function as the preceding procedure. Enter each cmdlet on a single line, even though they may appear word-wrapped across several lines here because of formatting constraints.

   ```
   Get-ADUser –Filter 'Name –Like "*test*"'|Remove-ADUser -Confirm:$false
   ```

4. Navigate to the **Deleted Objects** container, select **test2** and **test1** and then click **Restore** in the **Tasks** pane.
5. To confirm the objects were restored to their original location, navigate to the target domain and verify the user accounts are listed.

> **Note**
>
> If you navigate to the **Properties** of the user accounts **test1** and **test2** and then click **Member Of**, you w
> was also restored.

The following Windows PowerShell cmdlet or cmdlets perform the same function as the preceding procedure. Enter each cmdlet on a single line, even though they may appear word-wrapped across several lines here because of formatting constraints.

Windows PowerShell equivalent commands

```
Get-ADObject –Filter 'Name –Like "*test*"' –IncludeDeletedObjects | Restore-ADObject
```

To restore deleted objects to a different location

1. Right click the Windows PowerShell icon, click **Run as Administrator** and type **dsac.exe** to open ADAC.
2. Click **Manage**, click **Add Navigation Nodes** and select the appropriate target domain in the **Add Navigation Nodes** dialog box and then click **OK**.
3. Select users **test1** and **test2**, click **Delete** in the **Tasks** pane and then click **Yes** to confirm the deletion.
4. Navigate to the **Deleted Objects** container, select **test2** and **test1** and then click **Restore To** in the **Tasks** pane.
5. Select **OU1** and then click **OK**.
6. To confirm the objects were restored to **OU1**, navigate to the target domain, double click **OU1** and verify the user accounts are listed.

Windows PowerShell equivalent commands

The following Windows PowerShell cmdlet or cmdlets perform the same function as the preceding procedure. Enter each cmdlet on a single line, even though they may appear word-wrapped across several lines here because of formatting constraints.

```
Get-ADObject –Filter 'Name –Like "*test*"' –IncludeDeletedObjects | Restore-ADObject
–TargetPath "OU=OU1,DC=contoso,DC=com"
```

Fine-Grained Password Policy

The Windows Server 2008 operating system provides organizations with a way to define different password and account lockout policies for different sets of users in a domain. In Active Directory domains prior to Windows Server 2008, only one password policy and account lockout policy could be applied to all users in the domain. These policies were specified in the Default Domain Policy for the domain. As a result, organizations that wanted different password and account lockout settings for different sets of users had to either create a password filter or deploy multiple domains. Both are costly options.

You can use fine-grained password policies to specify multiple password policies within a single domain and apply different restrictions for password and account lockout policies to different sets of users in a domain. For example, you can apply stricter settings to privileged accounts and less strict settings to the accounts of other users. In other cases, you might want to apply a special password policy for accounts whose passwords are synchronized with other data sources. For a detailed description of Fine-Grained Password Policy, see AD DS: Fine-Grained Password Policies

What's new? In Windows Server 2012, fine-grained password policy management is made easier and more visual by providing a user interface for AD DS administrators to manage them in ADAC. Administrators can now view a given user's resultant policy, view and sort all password policies within a given domain, and manage individual password policies visually.

If you plan to use fine-grained password policies in Windows Server 2012, consider the following:

- Fine-grained password policies apply only global security groups and user objects (or inetOrgPerson objects if they are used instead of user objects). By default, only members of the Domain Admins group can set fine-grained password policies. However, you can also delegate the ability to set these policies to other users. The domain functional level must be Windows Server 2008 or higher.

- You must use the Windows Server 2012 version of Active Directory Administrative Center to administer fine-grained password policies through a graphical user interface.

Note

You can use **Server Manager** to install Remote Server Administration Tools (RSAT) on Windows Serv
version of Active Directory Administrative Center to manage Recycle Bin through a user interface.

You can use RSAT on Windows® 8 computers to use the correct version of Active Directory Administr
through a user interface.

Fine-Grained Password Policy step-by-step

In the following steps, you will use ADAC to perform the following fine-grained password policy tasks:

- Step 1: Raise the domain functional level
- Step 2: Create test users, group, and organizational unit
- Step 3: Create a new fine-grained password policy
- Step 4: View a resultant set of policies for a user
- Step 5: Edit a fine-grained password policy
- Step 6: Delete a fine-grained password policy

> **Note**
>
> Membership in the Domain Admins group or equivalent permissions is required to perform the following steps.

Step 1: Raise the domain functional level

In the following procedure, you will raise the domain functional level of the target domain to Windows Server 2008 or higher. A domain functional level of Windows Server 2008 or higher is required to enable fine-grained password policies.

To raise the domain functional level

1. Right click the Windows PowerShell icon, click **Run as Administrator** and type **dsac.exe** to open ADAC.
2. Click **Manage**, click **Add Navigation Nodes** and select the appropriate target domain in the **Add Navigation Nodes** dialog box and then click **OK**.
3. Click the target domain in the left navigation pane and in the **Tasks** pane, click **Raise the domain functional level**. Select a forest functional level that is at least Windows Server 2008 or higher and then click **OK**.

Windows PowerShell equivalent commands

The following Windows PowerShell cmdlet or cmdlets perform the same function as the preceding procedure. Enter each cmdlet on a single line, even though they may appear word-wrapped across several lines here because of formatting constraints.

```
Set-ADDomainMode -Identity contoso.com -DomainMode 3
```

Step 2: Create test users, group, and organizational unit

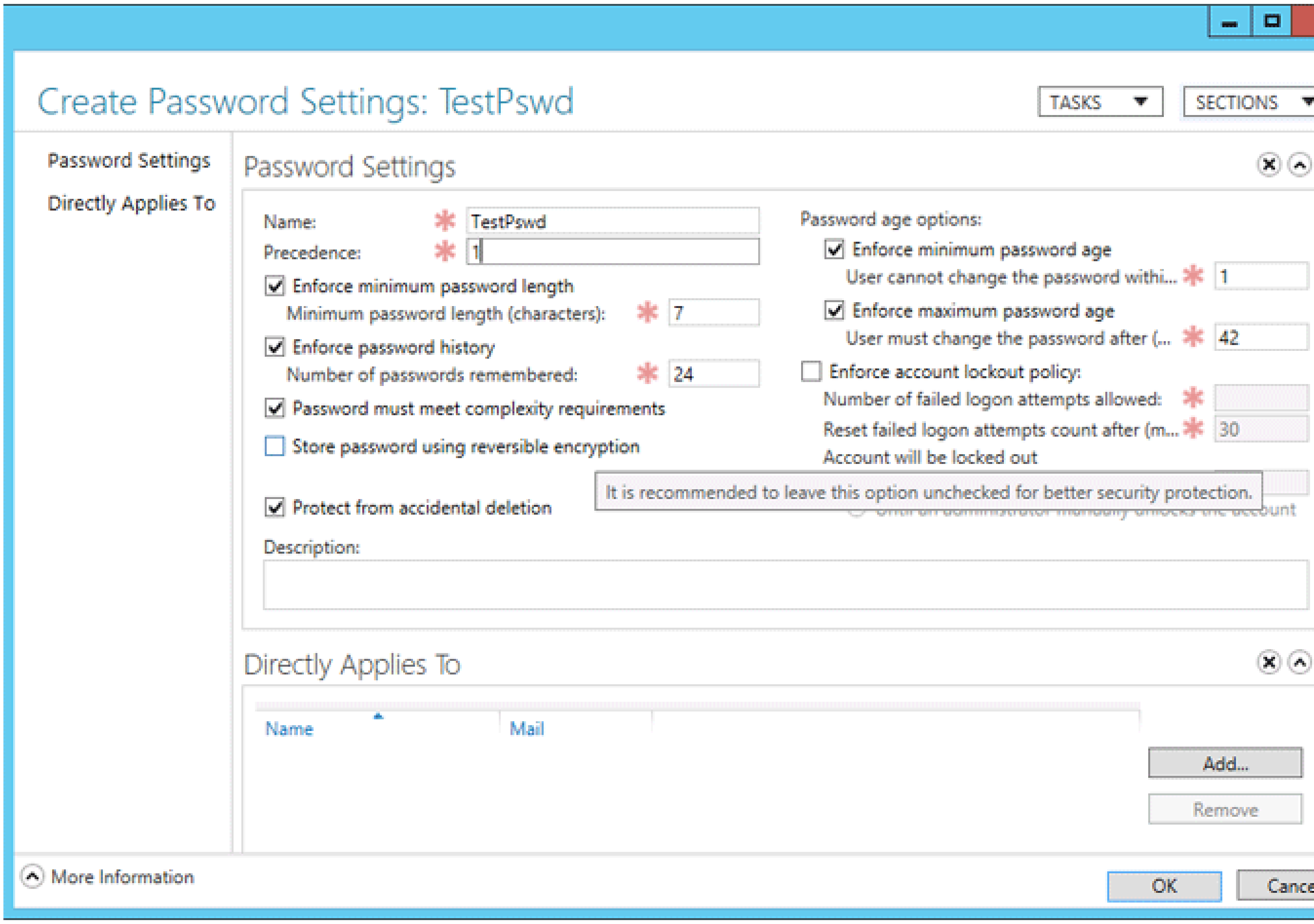
To create the test users and group need for this step, follow the procedures located here: Step 3: Create test users, group and organizational unit (you do not need to create the OU to demonstrate fine-grained password policy).

Step 3: Create a new fine-grained password policy

In the following procedure you will create a new fine-grained password policy using the UI in ADAC.

To create a new fine grained password policy

1. Right click the Windows PowerShell icon, click **Run as Administrator** and type **dsac.exe** to open ADAC.
2. Click **Manage**, click **Add Navigation Nodes** and select the appropriate target domain in the **Add Navigation Nodes** dialog box and then click **OK**.
3. In the ADAC navigation pane, open the **System** container and then click **Password Settings Container**.
4. In the **Tasks** pane, click **New**, and then click **Password Settings**.
   Fill in or edit fields inside the property page to create a new **Password Settings** object.
   The **Name** and **Precedence** fields are required.



Create Password Settings: TestPswd

Password Settings

Name: * TestPswd

Precedence: * 1

☑ Enforce minimum password length

Minimum password length (characters): * 7

☑ Enforce password history

Number of passwords remembered: * 24

☑ Password must meet complexity requirements

☐ Store password using reversible encryption

☑ Protect from accidental deletion

Description:

Password age options:

☑ Enforce minimum password age

User cannot change the password withi... * 1

☑ Enforce maximum password age

User must change the password after (... * 42

☐ Enforce account lockout policy:

Number of failed logon attempts allowed: *

Reset failed logon attempts count after (m... * 30

Account will be locked out

It is recommended to leave this option unchecked for better security protection.

Until an administrator manually unlocks the account

Directly Applies To

5. Under **Directly Applies To**, click **Add**, type **group1**, and then click **OK**.
   This associates the Password Policy object with the members of the global group you created for the test environment.
6. Click **OK** to submit the creation.

Windows PowerShell equivalent commands

The following Windows PowerShell cmdlet or cmdlets perform the same function as the preceding procedure. Enter each cmdlet on a single line, even though they may appear word-wrapped across several lines here because of formatting constraints.

```
New-ADFineGrainedPasswordPolicy TestPswd -ComplexityEnabled:$true -
LockoutDuration:"00:30:00" -LockoutObservationWindow:"00:30:00" -LockoutThreshold:"0"
-MaxPasswordAge:"42.00:00:00" -MinPasswordAge:"1.00:00:00" -MinPasswordLength:"7" -
PasswordHistoryCount:"24" -Precedence:"1" -ReversibleEncryptionEnabled:$false -
ProtectedFromAccidentalDeletion:$true
Add-ADFineGrainedPasswordPolicySubject TestPswd -Subjects group1
```

Step 4: View a resultant set of policies for a user

In the following procedure, you will view the resultant password settings for a user that is a member of the group to which you assigned a fine grained password policy in Step 3: Create a new fine-grained password policy.

To view a resultant set of policies for a user

1. Right click the Windows PowerShell icon, click **Run as Administrator** and type **dsac.exe** to open ADAC.
2. Click **Manage**, click **Add Navigation Nodes** and select the appropriate target domain in the **Add Navigation Nodes** dialog box and then click **OK**.
3. Select a user, **test1** that belongs to the group, **group1** that you associated a fine-grained password policy with in Step 3: Create a new fine-grained password policy.
4. Click **View Resultant Password Settings** in the **Tasks** pane.
5. Examine the password setting policy and then click **Cancel**.

Windows PowerShell equivalent commands

The following Windows PowerShell cmdlet or cmdlets perform the same function as the preceding procedure. Enter each cmdlet on a single line, even though they may appear word-wrapped across several lines here because of formatting constraints.

```
Get-ADUserResultantPasswordPolicy test1
```

Step 5: Edit a fine-grained password policy

In the following procedure, you will edit the fine grained password policy you created in Step 3: Create a new fine-grained password policy

To edit a fine-grained password policy

1. Right click the Windows PowerShell icon, click **Run as Administrator** and type **dsac.exe** to open ADAC.
2. Click **Manage**, click **Add Navigation Nodes** and select the appropriate target domain in the **Add Navigation Nodes** dialog box and then click **OK**.
3. In the ADAC **Navigation Pane**, expand **System** and then click **Password Settings Container**.
4. Select the fine grained password policy you created in Step 3: Create a new fine-grained password policy and click **Properties** in the **Tasks** pane.
5. Under **Enforce password history**, change the value of **Number of passwords remembered** to **30**.
6. Click **OK**.

Windows PowerShell equivalent commands

The following Windows PowerShell cmdlet or cmdlets perform the same function as the preceding procedure. Enter each cmdlet on a single line, even though they may appear word-wrapped across several lines here because of formatting constraints.

```
Set-ADFineGrainedPasswordPolicy TestPswd -PasswordHistoryCount:"30"
```

Step 6: Delete a fine-grained password policy

To delete a fine-grained password policy

1. Right click the Windows PowerShell icon, click **Run as Administrator** and type **dsac.exe** to open ADAC.
2. Click **Manage**, click **Add Navigation Nodes** and select the appropriate target domain in the **Add Navigation Nodes** dialog box and then click **OK**.
3. In the ADAC Navigation Pane, expand **System** and then click **Password Settings Container**.
4. Select the fine grained password policy you created in Step 3: Create a new fine-grained password policy and in the **Tasks** pane click **Properties**.
5. Clear the **Protect from accidental deletion** checkbox and click **OK**.
6. Select the fine grained password policy, and in the **Tasks** pane click **Delete**.
7. Click **OK** in the confirmation dialog.

Windows PowerShell equivalent commands

The following Windows PowerShell cmdlet or cmdlets perform the same function as the preceding procedure. Enter each cmdlet on a single line, even though they may appear word-wrapped across several lines here because of formatting constraints.

```
Set-ADFineGrainedPasswordPolicy –Identity TestPswd –ProtectedFromAccidentalDeletion
$False
Remove-ADFineGrainedPasswordPolicy TestPswd -Confirm
```

Windows PowerShell History Viewer

ADAC is a user interface tool built on top of Windows PowerShell. In Windows Server 2012, IT administrators can leverage ADAC to learn Windows PowerShell for Active Directory cmdlets by using the Windows PowerShell History Viewer. As actions are executed in the user interface, the equivalent Windows PowerShell command is shown to the user in Windows PowerShell History Viewer. This allows administrators to create automated scripts and reduce repetitive tasks, thus increasing IT productivity. Also, this feature reduces the time to learn Windows PowerShell for Active Directory and increases the users’ confidence in the correctness of their automation scripts.

When using the Windows PowerShell History Viewer in Windows Server 2012 consider the following:

- To use Windows PowerShell Script Viewer, you must use the Windows Server 2012 version of ADAC

Note
You can use **Server Manager** to install Remote Server Administration Tools (RSAT) on Windows Serv version of Active Directory Administrative Center to manage Recycle Bin through a user interface. You can use RSAT on Windows® 8 computers to use the correct version of Active Directory Administr through a user interface.

- Have some basic Windows PowerShell knowledge. For example, you need to know how piping in Windows PowerShell works. For more information about piping in Windows PowerShell, see Piping and the Pipeline in Windows PowerShell.

Windows PowerShell History Viewer step-by-step

In the following procedure, you will use the Windows PowerShell History Viewer in ADAC to construct a Windows PowerShell script. Before you begin this procedure, remove user, **test1** from the group, **group1**.

To construct a script using PowerShell History Viewer

1. Right click the Windows PowerShell icon, click **Run as Administrator** and type **dsac.exe** to open ADAC.
2. Click **Manage**, click **Add Navigation Nodes** and select the appropriate target domain in the **Add Navigation Nodes** dialog box and then click **OK**.
3. Expand the **Windows PowerShell History** pane at the bottom of the ADAC screen.
4. Select user, **test1**.
5. Click **Add to group...** in the **Tasks** pane.
6. Navigate to **group1** and click **OK** in the dialog box.
7. Navigate to the **Windows PowerShell History** pane and locate the command just generated.
8. Copy the command and paste it into your desired editor to construct your script.
   For example, you can modify the command to add a different user to **group1**, or add **test1** to a different group.