TNE10005/TNE60002

Network Administration

Lab 6

Configuring a Windows Server 2016 Domain

SWINBURNE UNIVERSITY OF TECHNOLOGY

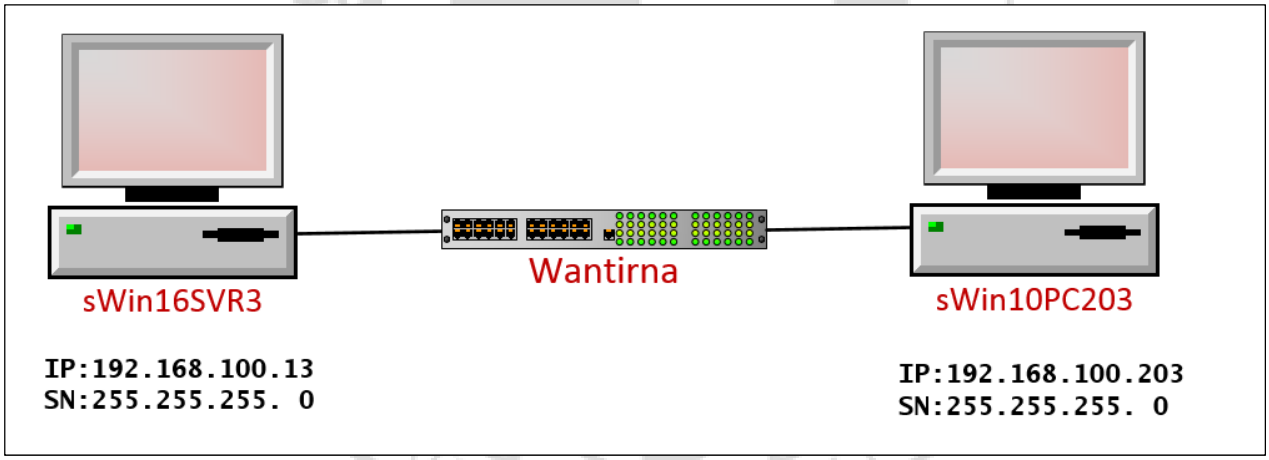
Aims:

- Install AD DS role to Windows Server 2016
- Join Windows 10 computers to a domain
- Create Domain User Accounts
- Create Domain Computer Accounts
- Create Domain Group Accounts
- Secure resources on Domain

Preliminary settings

1. Download and launch **sWin16SVR3** and **sWin10PC203**
2. **Connect** the Network Adapter of sWin16SVR3 and sWin10PC203 to the Virtual switch "**Wantirna**".
3. Ensure that both virtual machines have IP addresses in the subnet **192.168.100.0/24** before proceeding. Test by pinging from **sWin10PC203** to **sWin16SVR3** (remember default firewall settings block ping, you can allow pinging by creating a firewall rule or by sharing a folder – if you don't know how to do that, but you are confident that both Virtual machines are on the same subnet, proceed).

Topology

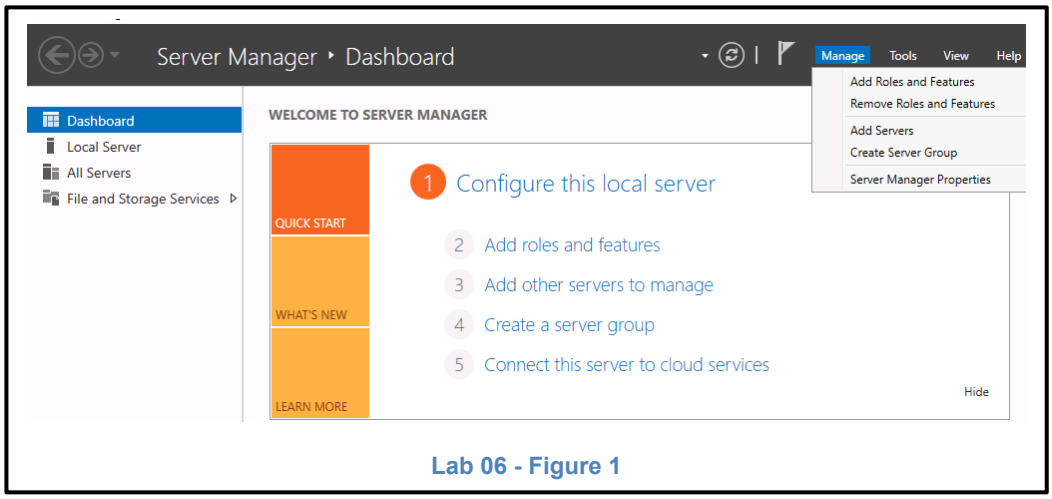


Creating a Domain

Configuring a Domain Controller

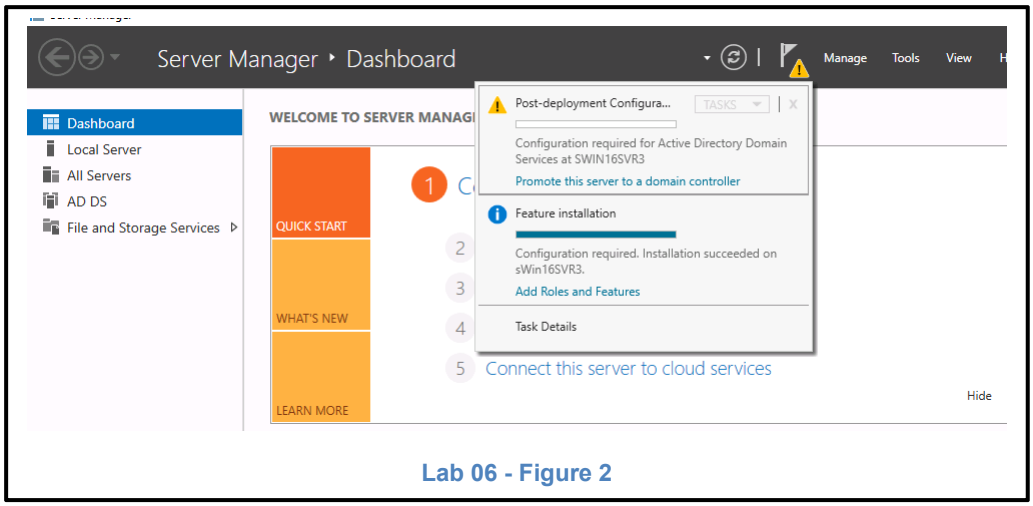
4. Log into **sWin16SVR3** as the Administrator with ***Pa55w.rd***.

5. From **Server Manager,** in the ***Manage*** tools, use ***Add Roles and Features*** to add the role of ***Active Directory Domain Services***, accepting the **default** settings.



Lab 06 - Figure 1

6. From ***Server Manager***, you should notice an alert symbol (a yellow triangle with a '!') to the left of the **Manage** tools. Click on this and select ***Promote this server to be a domain controller***.



Lab 06 - Figure 2

7. **Add a new forest** and call the domain **sWin.Local** (**.Local** is added so we don't interfere with any other domains if our server is connected accidentally to others).

8. For both the **Domain Functional Level** and **Forest Functional Level**, choose the **Windows Server 2016** level (Note: if there were older DCs already in the domain we would need to choose a domain functional level that would equal the oldest DC, as we are creating a new forest with only one DC we can set the domain functional level at the highest).
9. Accept the default options for installing a **DNS server** and a **Global catalog** (a GC is required as it is this will be the only DC in this forest). For the DSRM password type ***Pa55w.rd***. Keep in mind that if you are doing this for real, this password needs to be recorded and stored in a safe place where replacement administrators can find it (e.g. in a document called 'emergency passwords' stored in the company safe).
10. Accept the defaults for the remainder of the wizard. Note that errors may be displayed when the wizard completes the tests. This is mainly due to the virtual machine not being connected to the internet.
11. At the end of the installation, **restart** the computer and log in again as the **sWin\administrator**.

Creating resources

12. Create a folders at **C:** called **sWinData** and **Home**
13. In the **sWinData** folder create 3 text documents named **General.txt**, **Restricted.txt** and **TopSecret.txt**
14. Share the **Home** folder so that the **Everyone** group has **Full Control** (i.e. right click the folder, select **Properties**, select the **Sharing** tab, click **Advanced Sharing**..., click **Share this folder**, click the **Permissions** button, tick **Full Control** in the **Allow** column).

Joining a Domain

15. From **sWin10PC203**, log in as **Administrator** (Administrator is the administrator account of sWin10PC203). Ensure that the DNS address is configured with the IP address **of sWin16SVR3**'s address.

    You can use **ping** to confirm, **ping sWin.Local**. If an IP address is returned in the ping attempt, then you know DNS is working.

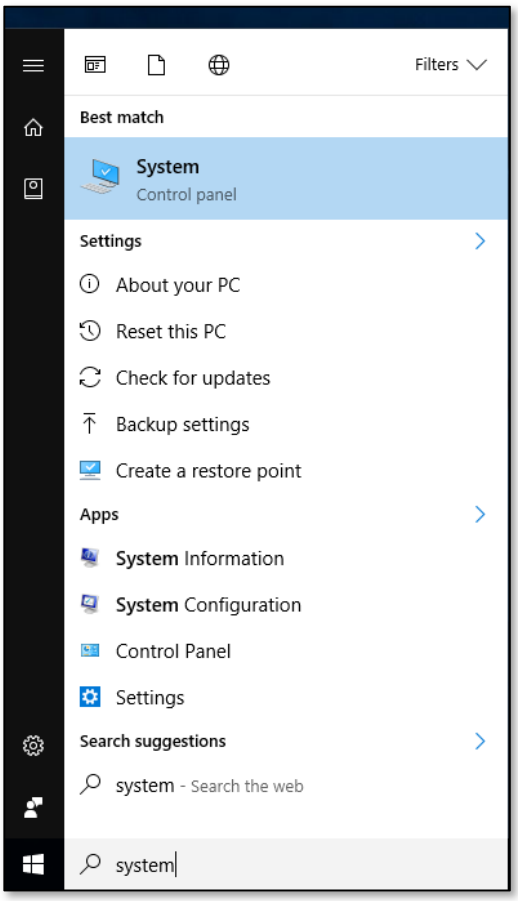
    Optional

    An alternative is to use nslookup is a command line utility that allows us to trouble shoot DNS problems. In this situation we can run a cmd console and type **nslookup** <enter>. You will be presented with the prompt **>**. At this prompt type the name of the domain, URL or machine name you want to resolve and press enter. If DNS is working it should return the IP address of that name.

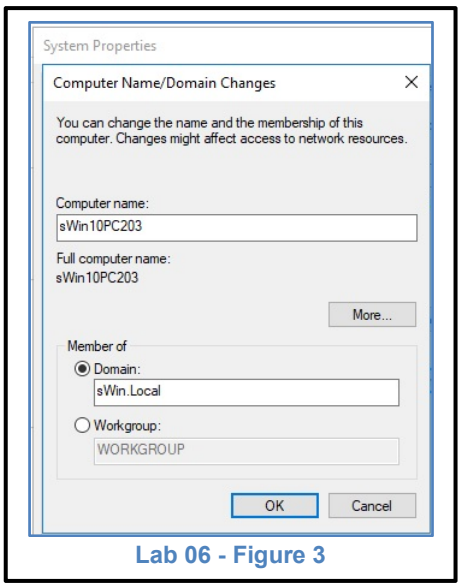
    In this situation we want to resolve our domain name. So type **sWin.Local** <enter>. If it returns the IP address of the domain controller, it is working (hint: type ***exit*** to leave NSLookup)

16. At the Start screen, start typing **System**.

    Note that when you start typing, a search bar appears and lists the applications and files that match you typing. Click **System** (Control Panel) to launch it. In the **System** window, select **Change Settings**.



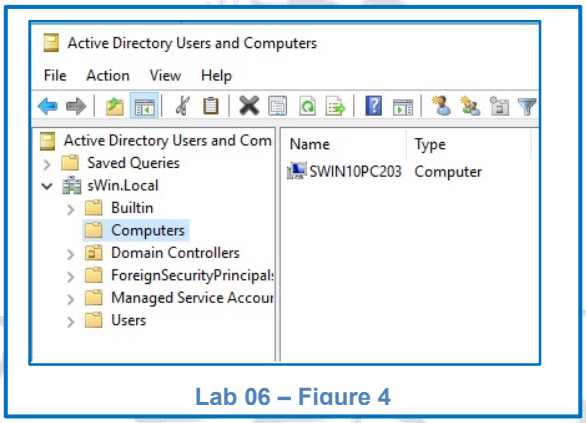
17. In the Member of **Domain** field type **sWin.Local** and press **OK**.



Lab 06 - Figure 3

18. When prompted, enter the username and password of an account that has the permissions to create computer accounts in the domain. In this situation we will use the **administrator**'s account, but normal we would use another account. Reboot the virtual machine when prompted.

19. You will notice that a computer account has been created in the **Computers** container in **Active Directory Users and Computers** (which can be accessed from Server Manager, Tools).



Lab 06 – Figure 4

Creating Accounts

Logging on With an Existing User Account

Logging on with existing accounts is a bit tricky with Windows 10.

Now that we have joined the domain, we can now log on with a local user account (e.g. **sWin10PC203\Jill**) or we can log on with a domain user account (e.g **sWin\Jill**).

If we log on with a local user account, we will be authenticated by the SAM, but consequently we can only have authorisation to resources on the local PC.

If we log on with a domain user account we will be authenticated by Active Directory and can have authorisation to any resource on the domain.

The question is how to determine whether we are logging in to the domain or the local machine.

20. Notice that when the **sWin10PC203** machine has rebooted, it now asks us to press **"CTRL + ALT + DELETE"** to log on. But when we do press CTRL+ALT+DELETE it is still defaulting to the local Jill's login. Click on **Other User**. Notice how it now states under the password field "Log on to: **sWin**". This means that the log on expected is from a sWin domain account. The problem is, we have not created any domain user accounts.

Creating User Accounts

We will now create a domain user account for the user Jill St John.

21. On **sWin16SVR3** run **Active Directory Users and Computers** from the administrative tools (or run **dsa.msc**).

22. Right click on the **Users** container and select **New…**, **User**.

23. Type **Jill** for the first name, **St John** for the last name and **Jill** for the user logon name, then click **Next**.

24. Enter the standard lab password for the password and confirm password.

    Normally we would keep the default setting of **User must change password at next logon** so that the user's password is known only by them. If creating a batch of user accounts we would also disable the account so it could not be used until we could confirm that the new user had started. For service accounts we would want the password to never expire.

    Clear the checkbox next to **User must change password at next logon**, so that we don't have to change the password when Jill logs on.

25. Go back to **sWin10PC203** and log on as **sWin\Jill**.

26. On **sWin16SVR3**, repeat the steps above to create a new user account called **Jack.**

Creating Computer Accounts with DSA.msc

27. In **Active Directory Users and Computers** right click on the **Computers** container and select **New…, Computer**

28. The computer name you type must exactly match the computer name of the PC you are creating an account for. In this case we are going to create an account for our second Windows 10 PC.

    In the **Computer name:** field, type **sWin10PC201.**

    Click on the **Change…** button and type in **Jill** then click the **Check Names** button. This gives Jill permission to join her computer to the domain, otherwise we would need to enter in the administrator details as we did in step 18. Click **OK.**

    Jill can now follow steps similar to steps 15 to 19 to join her computer to the domain.

    Don't try this now, as our host lab machines don't have in sufficient RAM, but at the end of the lab, after you have shut down sWin10PC203, start up sWin10PC201 and join it to the sWin.Local domain with Jill's username and password.

Configuring Account Properties

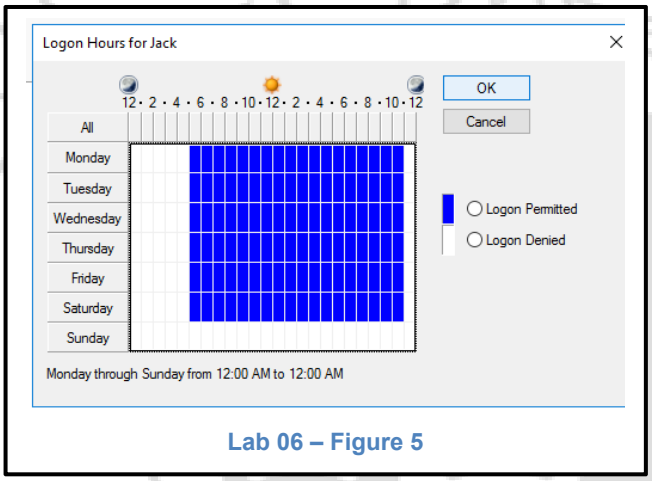
Setting User Account Properties

29. In **Active Directory Users and Computers**, in the **Users** container, right click on **Jack**'s user account and select **Properties**.
30. Configure the following settings for the following

tabs: General: Display name = Jack Nguyen
Description = Company Accountant
Office = 477B-101A
E-mail = Jack@sWin.Local

Address: Street = 477B Burwood Rd
City = Hawthorn
State = Vic.
Post Code = 3122
Country = Australia

Account: Logon Hours = 5AM-11PM, Monday to Saturday
Logon To = sWin10PC203, sWin10PC201



Lab 06 – Figure 5

Note: There are other important properties in the Account tab, such as **Unlock account, User must change password at next logon**, etc. We saw many of these in the new user wizard, but here is where we change these properties once the account has been created.

Profile: Home folder, Connect = Z: **\\sWin16SVR3\Home\%username%**

Note: The **%username%** is a system variable that will be replaced by the user's name (in this case Jack). The benefit of using this variable is that if you ever want to copy this user account, this property will then point to the new user's home folder. See Creating Account Templates, later in the lab.

Memberof: Remote Desktop Users

Note: We will add more memberof groups when we have created more groups later in the lab.

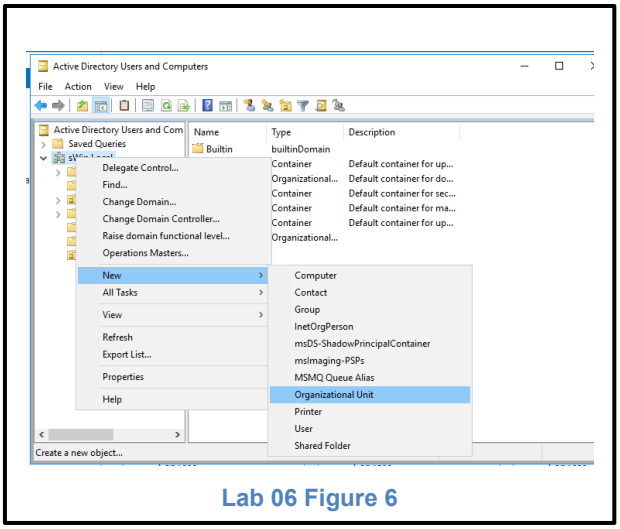
Setting Computer Account Properties

31. In **Active Directory Users and Computers**, in the **Computers** container right click on click on **sWin10PC203** and select **Properties**.
32. On the **General** tab, in the **Description** field enter a description of **sWin10PC203** e.g. **Jill's Virtual machine**. On the **Location** tab click and in the field enter **477B-101**.

Creating an Organisational Unit

An Organisational Unit is an administrator created container for accounts and groups. The administration of OU's can be delegated to other users so that the main administrator can organise the work load amongst the administration team.

33. In **Active Directory Users and Computers** right click on the **sWin.Local** domain root and choose **New…**, **Organizational Unit**. Name the OU **AccountDept**. If the OU is a temporary OU, remove the tick from the **Protect container from accidental deletion**. This OU is not going to be used in a live organisation so we'll remove the tick.



Lab 06 Figure 6

Creating Group Accounts

Creating Resource Groups

Resource groups are also called ACL groups. Their purpose is to streamline the way administrators control access to resources and to minimise the size of DACLs in order to keep servers functioning efficiently. The group scope Domain Local groups are best suited as ACL groups. They can only be given permissions to local resources yet they can have accounts from any trusted domain as members. Thus we can use domain local groups to control access from anywhere in the forest.

34. Right click on the new OU we created in step 33 and select **New…**, **Group**.

35. Ensure that you click the **Domain local** control button, so that the new group is of the right scope.

    The names of ACL groups should adhere to the following conventions. They should begin with ACL to reflect their purpose or DL to reflect their scope. The next part of the name should reflect the resource or resources they are controlling access to. The name should not reflect the users that are accessing the resource.

    ACL groups can only give one set of permissions. In other words you cannot use and ACL group to give Read permissions to one user and Full Control to another user. ACL groups can only give the same set of permissions, all read, or all read write or all full control. Consequently we reflect the level of permissions being given in the name.

    We will be creating DL groups to grant access to the sWinData folder created in step 12.

    Examples of group names that comply with the convention are DL_sWinData_RW, ACL sWinData RO.

36. Devise a naming scheme that meets the conventions above and create **two** DL groups for the sWinData folder. One DL group should give members **Read Only** access, the other group should give members **Read Write** access.

Assigning Permissions for Resources

Assigning NTFS Permissions

37. Right click on the folder **C:\sWinData** and select **Properties** and click on the **Security** tab.

38. This properties tab allows us to view the NTFS permissions currently allocated to the sWinData folder, but unlike Windows Server 2003, we cannot edit the NTFS permissions. Click the **Edit** button.

39. Here we can edit the currently assigned permissions. We want to add the resource groups we create in the Creating Resource Groups section of the lab.

    Click the **Add** button, and in the **Enter the object names…** field type the first two or three letters of the resource group names, e.g. **DL**. Then click **Check Names** to list the matching groups.

40. Click on the **Read Write** version of your resource group and click **OK**.

41. Make sure there are ticks in the allow column for the permissions:

    i. Read & Execute
    ii. List Folder Contents
    iii. Read
    iv. Write

    The first three permissions are default, but the write permission will need to be set. You may need to scroll the permissions down in order to see the write permission.

42. Now add the **Read Only** version of your resource group.

43. Make sure there are ticks in the allow column for the permissions:

    i. Read & Execute

    ii. List Folder Contents

    iii. Read

Then click **OK** and verify that the correct permissions have been set.

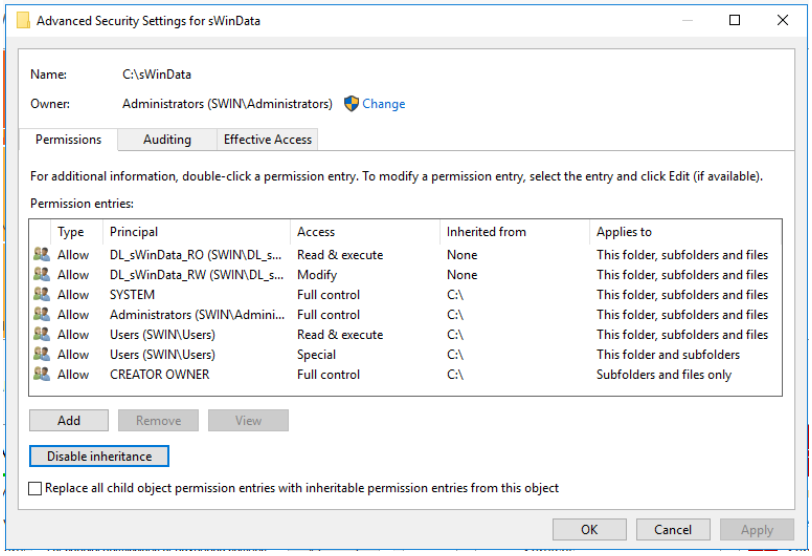
Notice that at the bottom of the list of groups with permissions to sWinData is the **Users** group.

What sort of permissions do they have?_____.

Removing Inherited Permissions

We will need to remove the permissions assigned to the **Users** group if we don't want everyone to have access to the files in this folder.

44. In the **Security** tab of the SWinData folder **Properties**, click on the **Advanced** button.



Lab 06 Figure 7

45. Click the **Disable inheritance** button, and select **Convert inherited permissions into explicit permissions on this object** option.

    If we had selected the **Remove all inherited permissions for this object** option we would have deleted all of the inherited permissions, even the Administrator permissions, thus potentially cutting our own access to the folder.

    For documents that need to be very secure we would remove the inherited permissions and then add the accounts that we want to provide access to.

    In this situation, there is only **one group** we want to remove from the list, so we choose **Convert...**.

46. In the list of **Permission entries:** click on every entry for the **User** group (and no other group!, then click **Remove**. Then click **OK** for the next two dialogs until you are back at the **sWinData Properties** dialog.

Assigning Share Permissions

NTFS permissions should always form the foundation of our access control, as they are applied in all circumstances. But if we want accounts to be able to access the data in a folder via the network, we must create a share. Thus we must also allocate share permissions.

47. Click on the **Sharing** tab and then the **Advanced Sharing…** button

48. By placing a tick in the **Share this folder** checkbox we activate the other controls on this dialog.

We will accept the default share name. By default it is the same as the folder name, but they can be different. This allows us to create a number of shares for any particular folder. We can create one share that will only allow read only access to a folder. We can then create another share for that folder that will allow full control access.

When we create a share, by default the share is advertised to all computers that have Network Discovery enabled. However if we have a share that provides full control access to a folder, we may not want it to be advertised. Consequently we are able to hide a share by adding a **$** to the end of the share name. For example, in this situation we would name the share **sWinData$** and it would no longer be advertised and hence users could not browse to the share. A user would have to know the full UNC path (i.e. \\server\sharename, e.g. **\\sWin16SVR3\sWinData$**) in order to access the share.

Note that we have not changed the name of the folder, it is still called sWinData, but by adding a **$** to the end of the share name, it becomes a hidden share.

49. Click on the **Permissions** button, and assign the **Everyone** group **Full Control**.

    Some may ask “isn’t Full Control a security risk?” The answer is yes, if you don’t have a good NTFS permission scheme in place.

    In this circumstance we intend for at least one group to have write access to the files in the folder. If we left the share permissions as the default read only, then no user would be able to write to the files in the folder while accessing them through the share. Not even the administrators! Not very functional! So knowing that we have a sound permission scheme in place (e.g. we removed the users group inheritance), I can safely set the share permission to full control. This way I have flexibility with future resource groups I create to give them whatever permission I feel the resource group needs.

    Click **OK** for the next three dialogs, until you are back to the **Local Disk (C:)** window.

Creating Account Groups

Account groups enable an administrator to streamline the allocation of permissions to a number of user or computer accounts. Account groups are created to group user and computer accounts that have similar requirements.

The Global group scope is typically used for creating account groups.

Requirements may be based on function within the organisation, or geography in an organisation that is spread over a wide area. For example we may be an administrator for a large company based in a tall building. We may want to prevent users from accidentally printing to printers on different floors. So we create account groups that will group the user accounts of users who are based in each floor. More typically, we create account groups based on function.

For example, in most medium to large organisations, people are employed in departments e.g. Sales, Production, Research, Support, Accounts, Maintenance, etc. Typically the users in each department perform a similar function, thus they tend to need access to the same resources. So we create an account group based on function in the organisation and give the group a name that reflects both the scope of the group and the functional area of the organisation. For example G_Accounts.

In distributed organisations (i.e. over a number of sites) our naming convention may need to be a hybrid of function and geography. This is certainly the case in a multi-domain forest.

For example, using the SWin campuses as a framework, for the Accounts department we may use group names such as: G_Accounts_Hwn, G_Accounts_Swk, G_Accounts_Lil, etc.

50. In **Active Directory Users and Computers** right click on the **AccountsDept** OU, and select **New…, Group**.

    Ensure that the group scope is **Global** and name the group **G_Acc_Mgrs**. Then click **OK**.

    Right click on our new group and choose **Properties,** then the **Members** tab. Add **Jill** as a member of the **G_Acc_Mgrs** group.

51. Repeat step 50, but this time name the group **G_Acc_Pay** and make **Jack** a member of this group.

Nesting Groups

We now need to link all of the groups together, so that our user accounts have the intended access to the SWinData folder. When we make one group a member of another group we call it group **nesting**.

52. For the **DL_SWinData_RW** group, using the same method in step 50, add the **G_Acc_Mgrs** group as a member.

53. For the next nesting we will use a slightly different approach. Right click on the **G_Acc_Pay** group, select **Properties**, but this time click the **Memberof** tab. Add the **DL_SWinData_RO** group as a member of group.

54. Ensure that you click **OK** to apply the configuration changes.

Testing the configuration

55. On **sWin10PC203**, log on as **Jill** and check to ensure that she has read and write access to the folder (remember the UNC path, step 48, will take us to the share directly if network browsing is too slow).

    Remember access tokens are created when a user account logs in. If Jill is already logged on to sWin10PC203 then none of the SIDs of the new group memberships will be present in her access token.

    Whenever group membership changes, *the user needs to log off and log back on* in order to create a new access token with the SIDs of the new groups.

56. Log on as **Jack** and ensure that he only has read access to the folder.

Extension

- Create some appropriate resource groups for the files TopSecret and Restricted. Also create some more user accounts making them members of the account groups we have created. Play around with different permission to the different groups and different group membership for the user accounts. See if you can predict the effective permissions for each group. Test your predictions by logging in as a user and try to access the different objects.
- Close down **sWin10PC203** and start **sWin10PC201.** See if you can add **sWin10PC201** to the domain using **Jill**'s domain credentials.

Pack up

1. Shut down and revert all virtual machines used in this lab.
2. Log off the Host server **ATC626-XY**
3. Push your chair in as you leave.

End of Lab