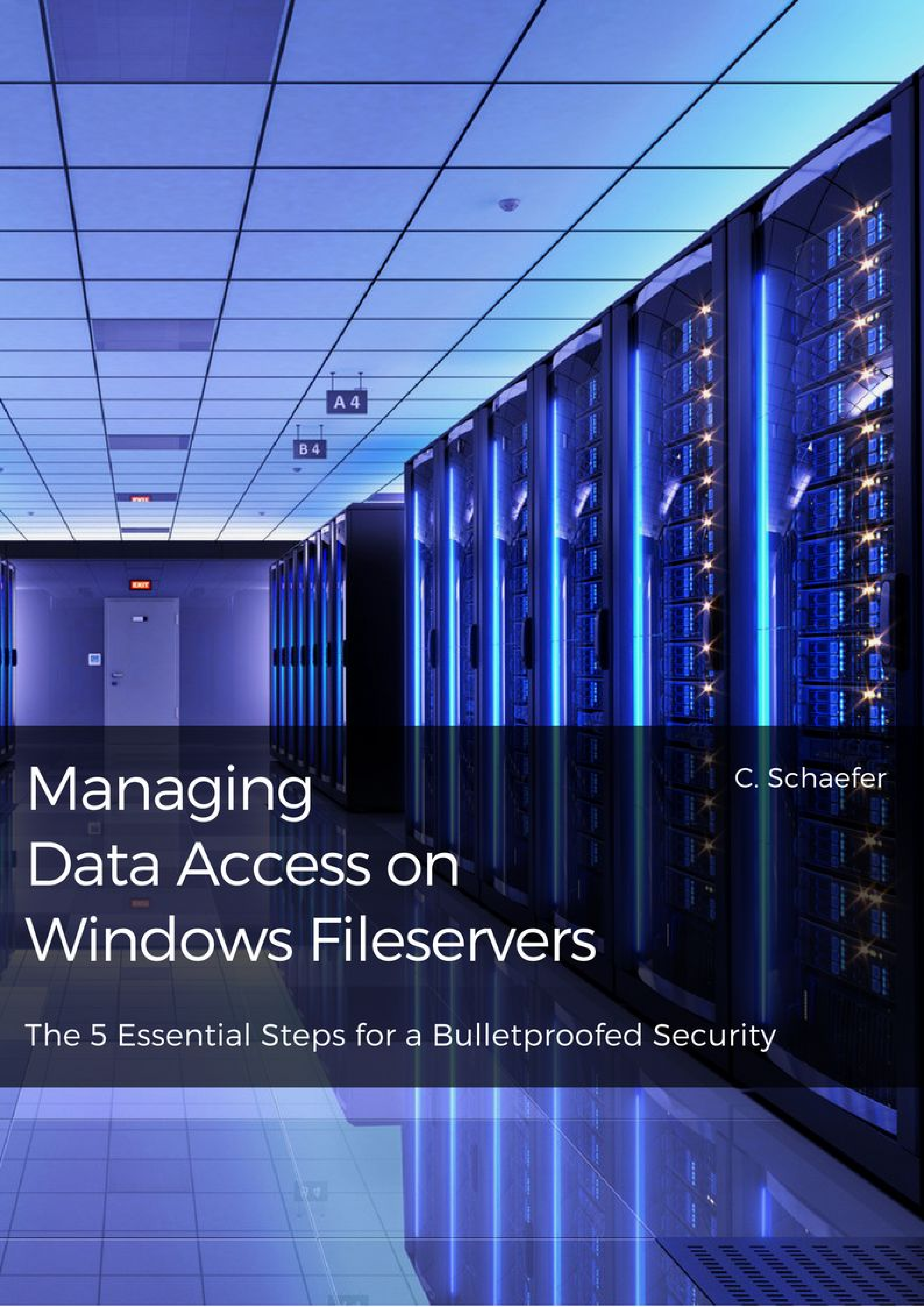


Managing Data Access on Windows Fileservers

C. Schaefer

The 5 Essential Steps for a Bulletproofed Security

Chapter 1

Planning

Step 1

Designing Folder Structure and Policies for Permission Assignment

Why do I Have to Design a Folder Structure?

To successfully and efficiently operate a complex Windows Folder Structure without any hassles or security leaks, you have to take the following points into consideration:

- Plan a folder structure to store the users' data files (documents, slides, graphics, drawings, etc.)
- Plan the shares
- Plan the Active Directory security groups
- Plan the permissions

If there is a lack of definition for any of the above topics or if substantial mistakes are made in the planning phase, the problems that occur during operation will increase with each day. Thus, you will require more time for operations, analysing problems will become more difficult, and the necessary enhancements will require far more effort to archive. Most of the time, the only solution will be to plan and create a completely new Windows filesystem environment, which will include a time-intensive data migration into the new folder structure.

The first step is to setup a folder structure and assign the appropriate permissions to that structure. The next step is the long-term daily management and operation of that environment.

Below are some of the real life situations that a Windows administrator could have a hard time dealing with if a proper folder structure is not designed from the start:

- The project manager urgently needs a new folder added to the project share with permissions set for only the project office.
- The employees in the accounting department change so often that, every day, a new employee needs to have permission assignments while exiting teammates need their permissions removed.
- The boss of the legal department has doubts that his data is secure and requests a list of the data trustees for his folders.

Huge mistakes will lead make the administrator's job far more stressful and will force them to do many routine operations and tedious tasks. Such wasted time can be invested in much more useful technologies.

What's Wrong with Having no Authorization Concept?

A solid, comprehensive plan will help avoid problems! The key to a secure and stable Windows Share and Folder environment is a solid authorization concept. If this is in place, you can trust in the security of your data!

It is important to have an access ***authorization concept*** before your IT administrators create new data structures within your system, no matter if those structures are for file data, web pages (Microsoft SharePoint), databases (MS SQL Server), applications, mailing lists, or folders (Microsoft Exchange).

If this authorization concept is missing on all levels, especially for:

a) Use Cases, such as:

- Permission assignments for users
- Withdrawal of permissions for individual users in individual access areas
- Simple reporting of access rights

b) And Business processes, such as:

- Approval processes for data access
- Approval processes for the creation of new objects in the data structure

then, the tasks of day-to-day management and medium-term reporting will no longer be easily implementable. These tasks will grow increasingly time-intensive as more uncertainties and security risks manifest. This is a nightmare for every IT administrator and security officer.

What kind of plan do I need to have smooth daily operations?

The first step will be to define your structures. This will likely lead to many questions:

- How should data files get organized? Are users allowed to create folders on their own?
- Who is responsible for the data? Whom do I speak to if an employee from one department requests permissions for a folder in a different department?
- How should the shares be designed? For instance, does every department need its own share? Is one share per business domain enough?
- How should the structure for the folders in the Active Directory be built? How should the Active Directory security groups be designed?
- How should I name the shares, folders, and Active Directory security groups?
- Should folder depth be limited? Is it efficient to manage the permissions of folders five levels deep?
- How should users who are assigned to specific folders gain access? Why shouldn’t users be directly assigned to those folders?
- How should the files and folders be backed up? How do I guarantee the software will be able to access all the data within the structure?

To answer these questions, we have to define specifications for these topics:

- Planning of file and folder structure
- Data Owner, who is responsible for this folder
- Planning of shares
- Planning of security groups in the Active Directory
- Definition of a naming convention for shares, folders, and groups
- Definition of a folder nesting depth limit
- Rules for permission assignments of users to gain access
- More necessary permission assignments for backup service accounts, operators, and administrators

Some Suggestions

To get a clear picture, you will need to look at some real-life examples:

- Shares: The amount of shares is not limited.
- Shares: The names may not be longer than 10 characters. Special characters are not allowed.
- Folders: The amount of folders is not limited.
- Folders with Permissions: The name of the folder may not exceed 15 characters. Special characters (_ and ,) are not allowed.
- Permissions: Permissions are assigned to folders, never to shares or files. Only permissions of type “Allow” are allowed. Never assign permissions of type “Deny”.
- Folder Nesting: Only assign security groups permissions to folders in the first or second hierarchy level. Child folders do inherit the permissions of their parent folders.

- Security Groups: For every folder with necessary permissions, an appropriate security group is created in the Active Directory.
- Naming Convention: Name security groups like this: FS_<sharename>_<foldername>[_<foldername>]_<permissions>
- Quota: Every folder with permissions will get a default quota of 100 GB. Enhancements should be requested by the data owner.
- Responsibility: For every folder with permissions, a responsible individual must be defined to manage said permissions. This person will decide who gets which kind of access permissions or quota enhancements.

When the IT team takes all these rules into consideration, you will be able to avoid most of the problems mentioned earlier.

Best Practices

Here are some practices you should follow to ensure quality planning when setting up your structures:

- Define your rules in detail. This step will help you ensure simple administration and smooth daily operations.
- Exceptions must always be documented.
- Never assign permissions to shares. Only assign permissions to the underlying folders!
- Never assign full control to shares or folders. This could lead to administrators accidentally being locked out by users.
- Remove “creator” and “owner” permissions. Having such permissions could lead to lock outs.
- Only assign full control to the folders within the internal system account.
- Set the limit for managed folders to go as far as the third level. Users may create more folders based on their needs, but those folders will not have any permissions assigned to them.
- The IT department may never be the data owner. Data owners must be an employee of an appropriate department.

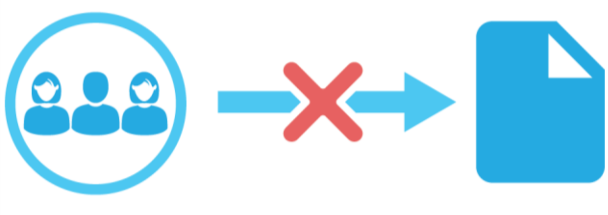
Chapter 2

Processes and Responsibilities

Step 2

Definition of Business Processes and Responsibilities

Since controlling access to business data is the foundation of data security and, in some cases, of data privacy, universally applicable, mandatory processes must be defined together with someone from senior management. If ISO certifications are planned or have already been received, then such processes will have been a basic precondition for those ISO certifications. Management must be willing to give its full support to the implementation and enforcement of these rules.



What are Business Processes?

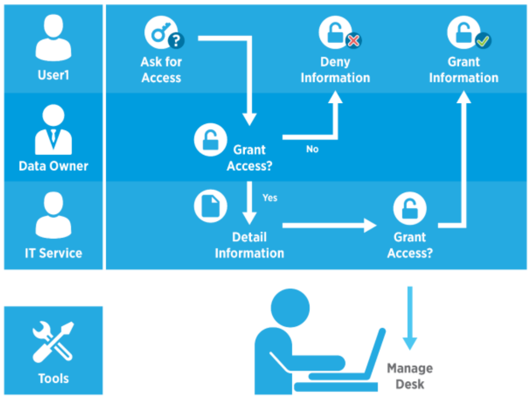
Business processes define work flows and responsibilities. In addition, processes can also outline the tools required or recommend for the execution of said processes.

Some examples of processes:

- Data requests
  - Requesting permissions
  - Changing permissions
  - Withdrawing permissions
- Creating new objects
- Assignment of and changes to responsibilities
- Assignment and modification of owners
- Expansion of storage requirements

Processes are often presented in the form of diagrams. Attached you will find a simple example of an assignment of permissions.

Compliance with these processes must be mandatory.
To ensure mandatory compliance, one must have the support of senior management or IT management, who must then communicate the necessary rules to all employees.



Executive Board or Management Responsibility

Administrators are responsible for the management of IT infrastructure, but they are not responsible for file structures or business processes concerning the assignment of permissions to data or other IT objects.

Often there will be few or no documented IT processes, which indicates that documentation is not being done well enough.

Unfortunately, senior management will often place the responsibility for permissions in the hands of the IT administrator. This is not a good idea. For instance, decisions regarding whether an employee shall have access to sales data will time and again prove to be poor decisions, especially if the “applicant” has better argumentation skills than the IT staff member or is located in another level of the company hierarchy. In other words, if the head of the “service” department wants his employees to receive permission to access data in the “sales” department, the decision should only be made by the head of the “sales” department.

Description of Processes

Giving new staff members a handbook that describes the IT environment of the company and all of the company's IT processes has been proved to be extremely beneficial. For example, data access privileges will only be awarded when the mandatory approval process has been successfully carried out - without exception.

To demonstrate the execution of the individual steps in a process, it is necessary to have so-called "Use Cases". Use Cases are step-by-step explanations of how an administrator (for example) creates a new file folder with individual permissions.

Further examples are:

- Awarding a user permission to access a particular data area
- Revocation of a user's permission to access a particular data area

Lack of Compliance with Business Processes and Requirements

Employees will frequently attempt to circumvent business processes. A typical example of this would be a call to the IT department, without filling out the required permission forms, to gain certain permissions. This will typically be justified by arguments like “it’s important”, “it’s urgent”, “I forgot to fill out the forms, but the new staff member is already here”, “I have specific instructions from the boss”, “if I don’t get the permissions immediately, then....”

In such a cases, the IT staff member will normally fail to document the assignment of permissions. The reason that there was “not enough time” will often be used to circumvent clear instructions regarding file folder structures or permission concepts.

The following scenario is a good illustration of this:

There is a requirement that access permissions for a file server may only be given on the file folder level. Despite this, the department head makes a request to receive the necessary permissions to access a specific file. To resolve this conflict, the IT administrator will have to create a new file folder and put the file into it. It would then be necessary to create and assign all permissions for this file folder. Instead of doing that, IT administrators will frequently try to save time and give the department head the requisite permissions for the file as a “one-off” exception.

Chapter 3

Assignment of Groups

Step 3

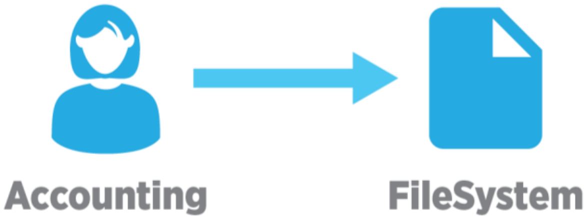
Assignment of Users to IT-Objects (Folders)

To give users access to data (whether the data consists of email distribution lists, file structures on file servers, or SharePoint spaces), they must be given permissions. In our example, they would need permissions for file folders. This authorisation given directly from a specific user's account.

For example,

you can give an employee from the Sales Team direct access to the folder "\\departments\sales" with "Full Control" permission. Doing so will allow the user to read the data and make changes to it. But what else will that user be able to do? With "Full Control" permission, that employee can also assign permissions and revoke them. Potentially, he/she could revoke access permissions for all other users, including administrators. What if this user only needs permission to read data? Should this access be the same for each individual member of the sales team?

Of course, permissions can also be assigned via Active Directory groups.



Assignment of General Security Groups to IT-Objects (Folders)

The usual, though incorrect, approach for creating permission structures is as follows:
A permission group is created for a department (e.g. Sales). At the same time, data areas will be created (e.g. file services, SharePoint spaces, and mail distributions).

The group “Sales” will then be assigned to these data areas. For example, this group gets “Write Access” permission for the file server folder “Sales” and “Read” permission on the web server. The mail distribution group is also taken care of using this authorisation group.

Next, we are faced with the following challenges:

- The managing director would like to have access to "Sales", but does not want to receive email from that group. Should the managing director be automatically put into the "Sales" group?
- A trainee starts in “Sales”. He does not need access to mail distribution, but only requires “Read” access to the data areas. What permissions should he be given?
- An employee from Human Resources needs “Read” access to a subset of the data area, but not to the web server. How will we proceed in this case?

In all the above cases, the simplest approach is no longer viable due to the following dilemma:

The permission groups are constructed on the basis of the organisation’s structure, not on the demands on and requirements of the data objects.

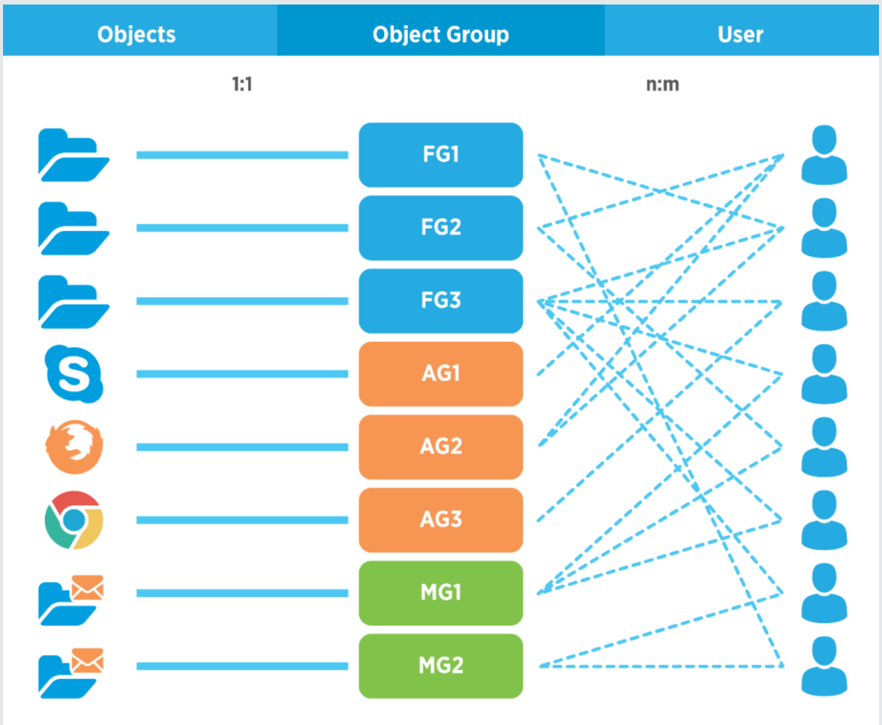
The solution for the problem described above is as follows:

For each IT object, (in our example, each folder in the file system), access requirements must be defined. For all underlying objects, (in this case, more folders and files), this will be done implicitly through inheritance of permissions. This principle means that at least one security group must be created within the Active Directory for each object requiring permission (e.g. each folder).

This assignment of dedicated permission groups to each folder with permissions has all the desired benefits for daily operations and reports. For each folder, it is possible to say exactly who has which permissions and access to the data in said folder, such as the users who are members of these particular permission groups. Furthermore, we know what a user's permissions will be thanks to the uniqueness of the assignment of an object (folder) to a permission group.

It can be said that a 1:1 relationship exists between the objects (our folders) and the groups within the Active Directory, while a many:many relationship exists between the users and permission groups. For the moment, we will ignore the fact that different groups are created for “Read” and “Write” permissions.

The below diagram shows how this works:

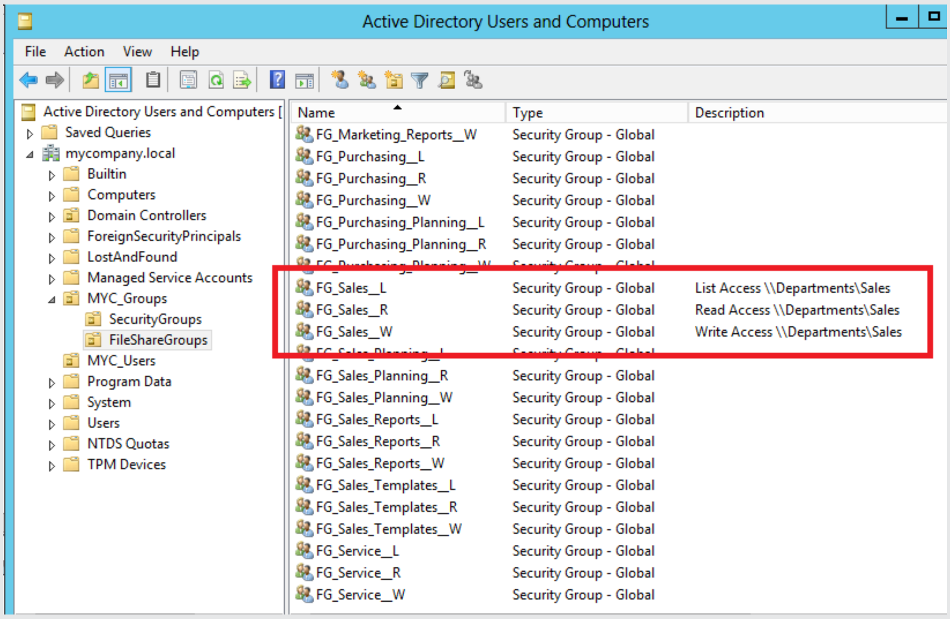


Different Permission Groups for One Folder

For our example, we will create three security groups within the Active Directory for each folder that requires permissions:

- A group for the award of **L**IST permissions
- A group for the award of **R**EAD Permissions
- A group for the award of **W**RITE Permissions

The below screenshot shows the three permission groups for the folder "\\departments\Sales":



Name	Type	Description
FG_Marketing_Reports__W	Security Group - Global	
FG_Purchasing__L	Security Group - Global	
FG_Purchasing__R	Security Group - Global	
FG_Purchasing__W	Security Group - Global	
FG_Purchasing_Planning__L	Security Group - Global	
FG_Purchasing_Planning__R	Security Group - Global	
FG_Purchasing_Planning__W	Security Group - Global	
FG_Sales__L	Security Group - Global	List Access \\Departments\Sales
FG_Sales__R	Security Group - Global	Read Access \\Departments\Sales
FG_Sales__W	Security Group - Global	Write Access \\Departments\Sales
FG_Sales_Planning__L	Security Group - Global	
FG_Sales_Planning__R	Security Group - Global	
FG_Sales_Planning__W	Security Group - Global	
FG_Sales_Reports__L	Security Group - Global	
FG_Sales_Reports__R	Security Group - Global	
FG_Sales_Reports__W	Security Group - Global	
FG_Sales_Templates__L	Security Group - Global	
FG_Sales_Templates__R	Security Group - Global	
FG_Sales_Templates__W	Security Group - Global	
FG_Service__L	Security Group - Global	
FG_Service__R	Security Group - Global	
FG_Service__W	Security Group - Global	

“Read” permissions are assigned when a user only needs to read files within a folder. For example, all public information about a project in the folder "Project Office" or all lists with sales prices in the folder "\\departments\Sales\Items" would be covered under “Read” permissions.

“Write” permissions will be awarded only if a user needs to alter files. It is important to keep in mind that assigning “Write” permissions also gives the user the permission to delete. In the two examples above, “Write” permissions would be assigned to the staff members in project management or the project office who create and maintain information, as well as the staff members from the sales team who specify the sales prices based on internal calculations.

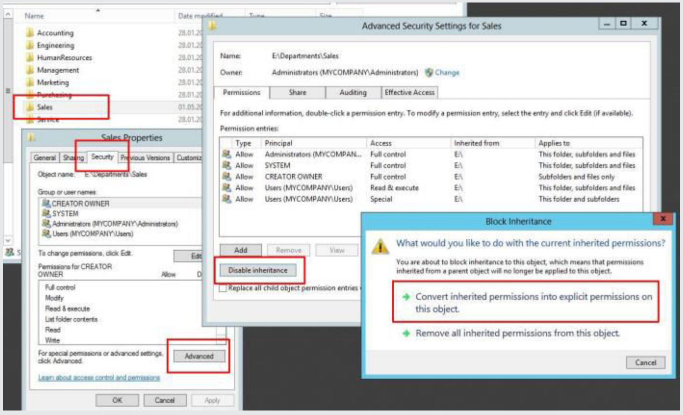
List permissions are required when a user needs rights to the folders deeper down in the file tree, but he does not have “Read” or “Write” permissions for all the folders on the levels above. This will ensure that the user can access the folder to which he/she has received permissions.

With Excel and some knowledge of scripting, it is possible to construct a simple way to create and administrate these security groups with folder permissions.

Restriction of Folder Permissions and Assignment of Permissions to Permission Groups

After all the necessary security groups have been created within the Active Directory, it is necessary to give these groups permissions for all appropriate folders. One should start with the highest folder in the hierarchy. In our example, that would be the "Sales" folder. Allocation of folder permissions are done in three steps:

I) In the first step, any existing inherited permissions must be deactivated or revoked.



Sales Properties

Object name: E:\Departments\Sales

Group or user names:
- CREATOR OWNER
- SYSTEM
- Administrators (MYCOMPANY\Administrators)
- Users (MYCOMPANY\Users)

To change permissions, click Edit.

Permissions for CREATOR OWNER

For special permissions or advanced settings, click Advanced.

Advanced Security Settings for Sales

Name: E:\Departments\Sales

Owner: Administrators (MYCOMPANY\Administrators) Change

For additional information, double-click a permission entry. To modify a permission entry, select the entry and click Edit (if available).

Permission entries:

Type	Principal	Access	Inherited from	Applies to
Allow	Administrators (MYCOMPAN...	Full control	E:\	This folder, subfolders and files
Allow	SYSTEM	Full control	E:\	This folder, subfolders and files
Allow	CREATOR OWNER	Full control	E:\	Subfolders and files only
Allow	Users (MYCOMPANY\Users)	Read & execute	E:\	This folder, subfolders and files
Allow	Users (MYCOMPANY\Users)	Special	E:\	This folder and subfolders

□ Replace all child object permission entries

Block Inheritance

What would you like to do with the current inherited permissions?

You are about to block inheritance to this object, which means that permissions inherited from a parent object will no longer be applied to this object.

- Convert inherited permissions into explicit permissions on this object.
- Remove all inherited permissions from this object.

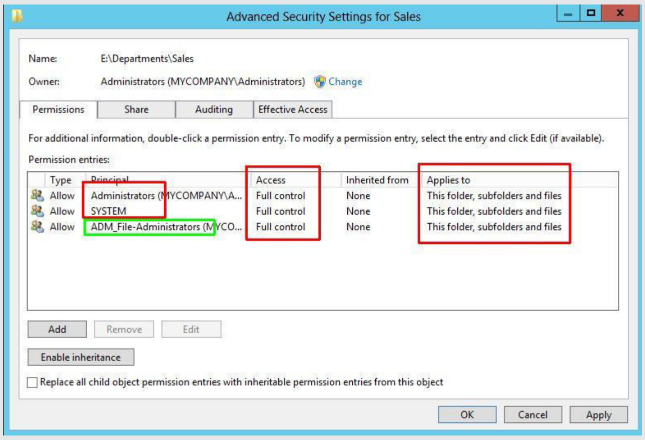
This will ensure that the folder will only have explicitly assigned permissions.

II) In the second step, the permissions for the administrator group must be created. The following best practices are worth taking note of:

a. The built-in account "system" receives "Full Control" permission. This is important since the operating system uses this account for certain services and processes. Thus, you should ensure that this permission is always granted.

b. The local group "Administrators" will likewise receive "Full Control". This ensures that the server administrators always have access to the necessary data and permissions. In addition, some backup programs also need these permission to function correctly.

c. Furthermore, you should create a security group for operators and administrators with "Full Control". This guarantees that the IT administrators have the necessary permissions for the daily operation of the file server.



Advanced Security Settings for Sales

Name: E:\Departments\Sales

Owner: Administrators (MYCOMPANY\Administrators) Change

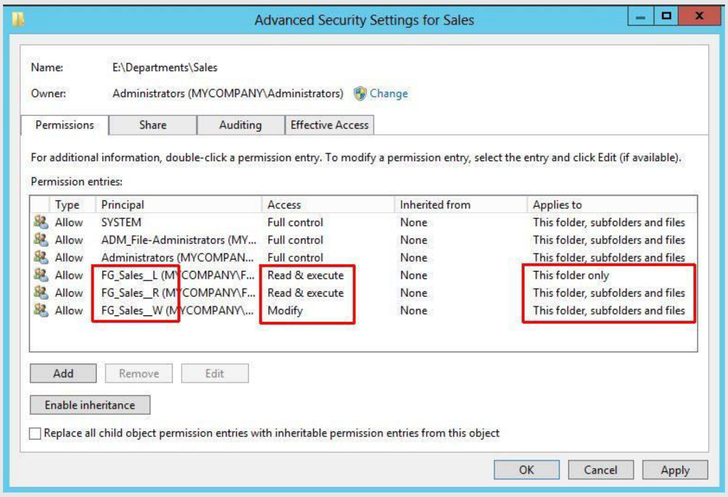
For additional information, double-click a permission entry. To modify a permission entry, select the entry and click Edit (if available).

Permission entries:

Type	Principal	Access	Inherited from	Applies to
Allow	Administrators (MYCOMPANY\A...	Full control	None	This folder, subfolders and files
Allow	SYSTEM	Full control	None	This folder, subfolders and files
Allow	ADM_File-Administrators (MYCO...	Full control	None	This folder, subfolders and files

□ Replace all child object permission entries with inheritable permission entries from this object

III) In the third step, the security groups created for each folder must be assigned to the folder. The awarded permissions will be assigned as follows:



Advanced Security Settings for Sales

Name: E:\Departments\Sales

Owner: Administrators (MYCOMPANY\Administrators) Change

For additional information, double-click a permission entry. To modify a permission entry, select the entry and click Edit (if available).

Permission entries:

Type	Principal	Access	Inherited from	Applies to
Allow	SYSTEM	Full control	None	This folder, subfolders and files
Allow	ADM_File-Administrators (MY...	Full control	None	This folder, subfolders and files
Allow	Administrators (MYCOMPAN...	Full control	None	This folder, subfolders and files
Allow	FG_Sales__L (MYCOMPANY\F...	Read & execute	None	This folder only
Allow	FG_Sales__R (MYCOMPANY\F...	Read & execute	None	This folder, subfolders and files
Allow	FG_Sales__W (MYCOMPANY\...	Modify	None	This folder, subfolders and files

☐ Replace all child object permission entries with inheritable permission entries from this object

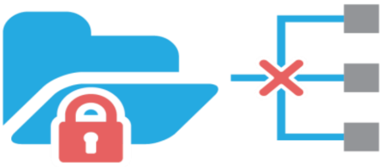
Awarding Additional Permissions for Deep Data Sub-Structures

If there are no restrictions on the number of levels in the file structure for the assignment of permissions, the complexity of the administration tasks increases exponentially. Suppose that the average number of subfolders in a file system is 10. The complexity of the administration and documentation of the highest-level folder will be 10. If a second level is included, the complexity will increase to 10x10 or 100. If we further assume that the average folder depth is 10 and that there are no restrictions on folder authorisations, the management complexity will be 10 billion. That means an IT administrator may theoretically be required to manage 10 billion permissions. This is true for documentation, reports, and changes.

Avoiding “Deny” Permissions

The need to deny a user access to a specific folder does not mean that you should use the “Deny” permission, as doing so increases the complexity of administration, documentation, and reporting by an unnecessarily large magnitude. For example, during each assignment of permissions, all “deny” groups within the parent data areas must be checked. When planning the folder structure, one must always keep this consideration in mind and structure the files with their permission groups in such a way that the “Deny” permission is not used at all. In practice, this is easily possible if you present the users with a folder structure and do not capitulate to the requests of every staff member.

Do Not Use the “Share” Permission



When creating shares within a file system, it is possible to restrict access to the shares to which “Share” permissions have been given. This unnecessarily doubles the complexity of administration. Instead, one should generally assign a “Write” permission.

Furthermore, to avoid unwanted attempts to gain access, one can hide the shares (by putting a $ sign at the end of the share name). In Windows Server 2012, the "access-based enumeration” setting has the effect that a user will only be able to see a folder if he/she has a permission to see it.

Chapter 4

Assignment of Users

Step 4

Assignment of Users to Active Directory Security Groups

The basis for the assignment of users to folders is a rather complex question and answer game. For each folder that needs to be protected with permissions, ask the person responsible for the data which users should receive which access rights. Please follow the processes described in the previous chapter.

A permissions matrix helps gather necessary data and provides documentation. These matrixes can easily be made using an Excel table.

For each folder that needs its own permissions, make a row. In the columns, the users that have access to the folder will be recorded. The necessary permissions can be specified with a “W” for “Write” permissions and an “R” for “Read” permissions.

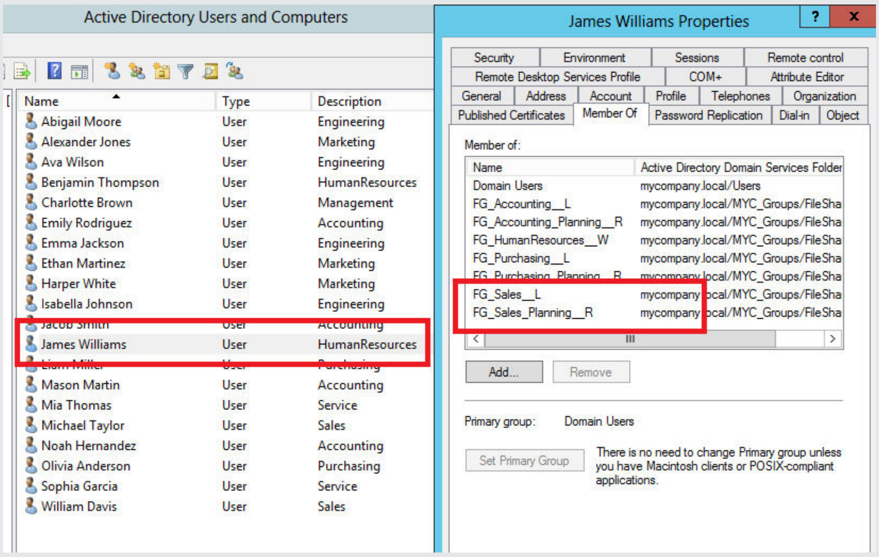
With this matrix as a starting point, you can plan and create security groups within the Active Directory and assign users to the appropriate groups.

	User	Emily Rodriguez	Jacob Smith	Mason Martin	Noah Hernandez	Abigail Moore	Ava Wilson	Emma Jackson	Isabella Johnson	Benjamin Thompson	James Williams	Charlotte Brown	Alexander Jones	Ethan Martinez	Harper White	Liam Miller	Olivia Anderson	Michael Taylor	William Davis	Mia Thomas	Sophia Garcia
	Department	Ac	Ac	Ac	Ac	En	En	En	En	Hu	Hu	Ma	Mar	Mar	Mar	Pu	Pu	Sa	Sa	Se	Se
1st folder level	2nd folder level																				
Accounting		W	W	W	W							R									
Accounting	Planning	R	W	R							R	R									
Accounting	Templates	R	W	R	R							R									
Engineering						W	W	W	W			R									
Engineering	Reports					W	W	R				R									
Engineering	Research						W	W				R									
HumanResources										W	W	R									
Management												W									
Marketing												R	W	W	W						
Marketing	Projects											R		R	W						
Marketing	Reports											R		R	W						
Purchasing												R				W	W				
Purchasing	Planning										R	R				W	R				
Sales												R						W	W		
Sales	Planning										R	R						R	W		
Sales	Reports											R							W		
Sales	Templates											R						W	W		
Service												R								W	W

These tables should ideally be administrated directly by the person responsible for the data in question (the data owner). A matrix should be created and maintained for each department. Otherwise, a very large matrix should be used to administrate the permissions for all departments, in which case other persons should not be allowed to change the content of cells.

Example:

An employee from Human Resources needs to read the vacation lists from Sales. This is stored in: "\\Department\sales\planning"



Name	Type	Description
Abigail Moore	User	Engineering
Alexander Jones	User	Marketing
Ava Wilson	User	Engineering
Benjamin Thompson	User	HumanResources
Charlotte Brown	User	Management
Emily Rodriguez	User	Accounting
Emma Jackson	User	Engineering
Ethan Martinez	User	Marketing
Harper White	User	Marketing
Isabella Johnson	User	Engineering
Jacob Smith	User	Accounting
James Williams	User	HumanResources
Liam Miller	User	Purchasing
Mason Martin	User	Accounting
Mia Thomas	User	Service
Michael Taylor	User	Sales
Noah Hernandez	User	Accounting
Olivia Anderson	User	Purchasing
Sophia Garcia	User	Service
William Davis	User	Sales

Member of:

Name	Active Directory Domain Services Folder
Domain Users	mycompany.local/Users
FG_Accounting__L	mycompany.local/MYC_Groups/FileSha
FG_Accounting_Planning__R	mycompany.local/MYC_Groups/FileSha
FG_HumanResources__W	mycompany.local/MYC_Groups/FileSha
FG_Purchasing__L	mycompany.local/MYC_Groups/FileSha
FG_Purchasing_Planning__R	mycompany.local/MYC_Groups/FileSha
FG_Sales__L	mycompany.local/MYC_Groups/FileSha
FG_Sales_Planning__R	mycompany.local/MYC_Groups/FileSha

Primary group: Domain Users

There is no need to change Primary group unless you have Macintosh clients or POSIX-compliant applications.

So that the HR employee does not gain access to the entire “Sales” folder and subfolders, he/she must first be put into the LIST Group for that folder. By using this step, the HR employee can open the “Sales” folder, but cannot read or change data. At that point, the HR employee must be assigned to the group “FG Sales Planning R”, which granted them “Read” permission for the subfolder. That employee will then be able to access the subfolder planning and read the data within.

In short, this “LIST” permission allows someone to “take a walk” through a closed area.

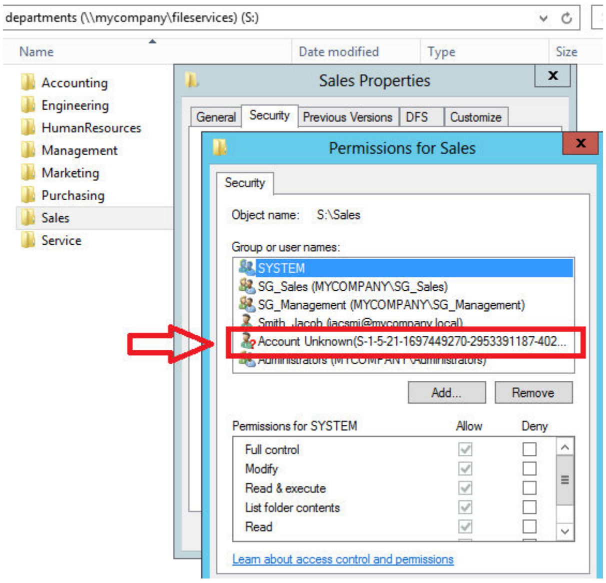
No Assignment of Individual Permissions

The permission structure is not always as simple as the company structure. Often, it will be necessary to create permissions that originate outside of the data area. For instance, it might be necessary for an HR employee to have access to the company's personnel planning table, which is located in the data area of "Sales".

The IT administrator should not assign the HR employee to the Sales group. Instead, the administrator should provide that employee with permissions for this folder as an individual or even provide only the permission for a single file.

Failing to do so will have fatal consequences:

- A search to find out where the user has access permissions would have to be conducted on all servers.
- If an employee changes their area of work or department, it is no longer easy to know which permissions must be changed. If there is no documentation, no one will what permissions that employee had.
- If an employee leaves the company and their account is deleted, then an "SSID corpse" (an unreadable identification code no longer be associated with a person) will remain in the ACL list of the folder.



Accounting
Engineering
HumanResources
Management
Marketing
Purchasing
Sales
Service

Object name: S:\Sales

Group or user names:
SYSTEM
SG_Sales (MYCOMPANY\SG_Sales)
SG_Management (MYCOMPANY\SG_Management)
Smith, Jacob (jacsmi@mycompany.local)
Account Unknown(S-1-5-21-1697449270-2953391187-402...
Administrators (MYCOMPANY\Administrators)

Permissions for SYSTEM	Allow	Deny
Full control	☑	□
Modify	☑	□
Read & execute	☑	□
List folder contents	☑	□
Read	☑	□

Chapter 5

Tooling & Reporting

Step 5

Tools, Quality Checks, and Reporting

By creating multiple security groups with permissions for each individual data object, there would, at first glance, seem to be a very large number of groups in the AD. Some administrators argue that this method is confusing and requires extra resources, but, in reality, the AD resource requirements will be minimal and the advantages will compensate for them in the long run.

Support Using Scripts

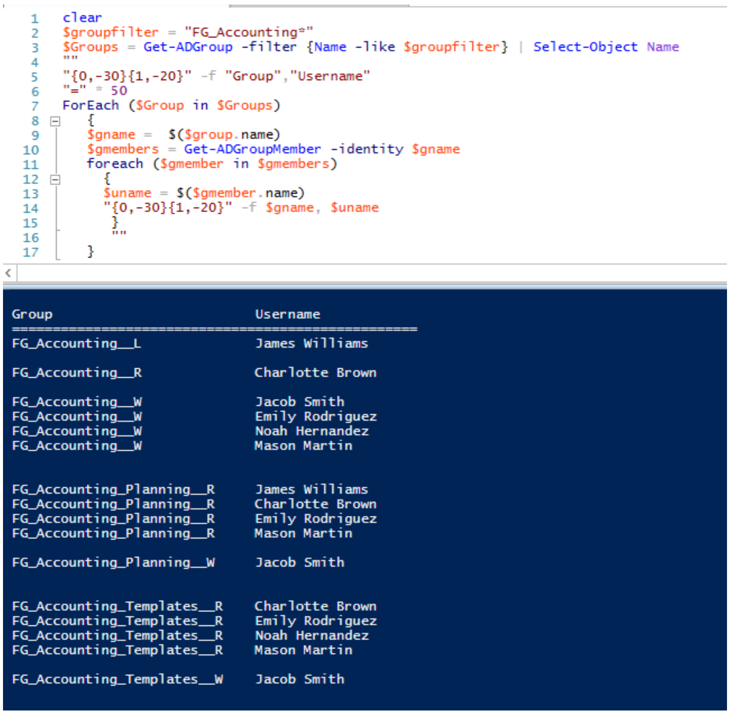
The administration may seem confusing at first glance, but the construction can be mapped in an Excel worksheet, as explained in the previous chapter. That is, the administration of permissions can be monitored using a simple Excel table. One can, for instance, create a matrix for every object type (file access, mail distribution, SharePoint access). By using a simple VBA or PowerShell script, the permissions can be transferred to the AD automatically.

In this way, an administrator does not have to have direct access to the file system. In addition, administrators will no longer have to fight through the AD and the file server ADLs. By running a script, requests requiring changes in the permission structure can be easily taken care of.

Support Using Tools

There are, of course, comfortable systems for taking care of administration tasks that provide the administrator or person in charge of permissions with a nice UI with many possibilities, especially if none of them have scripting experience. These tools make it possible for a less-skilled administrator, who does not have a deep knowledge of permissions assignments in the NTFS file system, to carry out administration tasks quickly and easily. In other words, these tools make it possible for the administrator to have a life without worries, since he/she will no longer need to spend a lot of time administrating groups in the AD and does not have to work on the permissions in the file system again and again. Instead, the administrator will be able to allot additional time to more meaningful tasks.

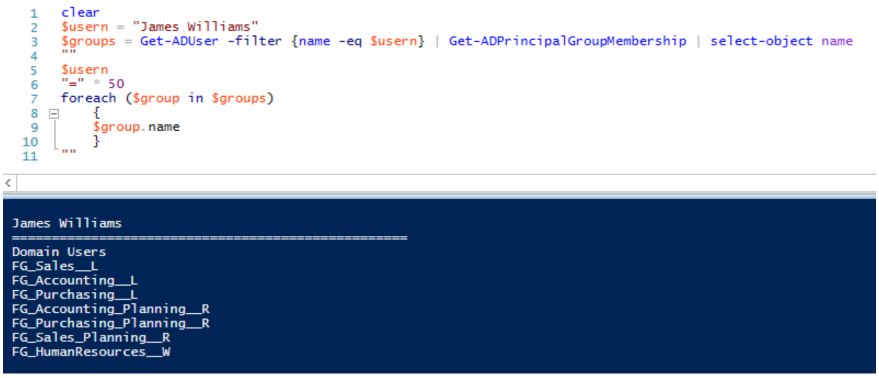
When the assignment of groups to folders has been done, it is possible, with some effort, to do manual analyses of the effective permission holders. Everyone who is a member of a certain group will have a specific access permission to a specific object and, if necessary, its child objects. That means that the access possibilities in each area can be shown by a simple analysis of group memberships. Here is an example of a simple PowerShell script that lists who can access "Accounting" and which permissions each of them have.



```
clear
$groupfilter = "FG_Accounting*"
$Groups = Get-ADGroup -filter {Name -like $groupfilter} | Select-Object Name
""
"{0,-30}{1,-20}" -f "Group","Username"
"=" * 50
ForEach ($Group in $Groups)
   {
   $gname =  $($group.name)
   $gmembers = Get-ADGroupMember -identity $gname
   foreach ($gmember in $gmembers)
     {
     $uname = $($gmember.name)
     "{0,-30}{1,-20}" -f $gname, $uname
      }
      ""
   }
```

```
Group                         Username
==================================================
FG_Accounting__L              James Williams

FG_Accounting__R              Charlotte Brown

FG_Accounting__W              Jacob Smith
FG_Accounting__W              Emily Rodriguez
FG_Accounting__W              Noah Hernandez
FG_Accounting__W              Mason Martin

FG_Accounting_Planning__R     James Williams
FG_Accounting_Planning__R     Charlotte Brown
FG_Accounting_Planning__R     Emily Rodriguez
FG_Accounting_Planning__R     Mason Martin

FG_Accounting_Planning__W     Jacob Smith

FG_Accounting_Templates__R    Charlotte Brown
FG_Accounting_Templates__R    Emily Rodriguez
FG_Accounting_Templates__R    Noah Hernandez
FG_Accounting_Templates__R    Mason Martin

FG_Accounting_Templates__W    Jacob Smith
```

The same is true for persons (Individual Active Directory accounts). An employee is a member of certain access groups. Because of the uniqueness of the membership, it is possible to show immediately which objects the employee can access and what permissions he/she has.



```
clear
$usern = "James Williams"
$groups = Get-ADUser -filter {name -eq $usern} | Get-ADPrincipalGroupMembership | select-object name
""
$usern
"=" * 50
foreach ($group in $groups)
    {
    $group.name
    }
""
```

```

James Williams
==================================================
Domain Users
FG_Sales__L
FG_Accounting__L
FG_Purchasing__L
FG_Accounting_Planning__R
FG_Purchasing_Planning__R
FG_Sales_Planning__R
FG_HumanResources__W
```

But what if the security groups are nested? For instance, what if a group is itself a member of another group, whose members have access to a specific folder. In such a case, the analysis will be more time-consuming and prone to errors, as it is easy to lose track of things in more complex contexts.

As already mentioned, professional tools offer far more possibilities and a more comfortable user interface. In addition, such tools do not require coding expertise and can be provided to the data owner or even directly to users.

Documentation

Surveys show that only about half of all IT administrators document their work. Thus, many are not documenting how the administration of permissions is taken care of or who, why, when, and where permission was gained or revoked and by whom. If there is a data security breach or audit, this lack of planning can have serious consequences.

No Tools for Managing Security Rights

The complexity of IT is constantly increasing. This applies not only to applications, networks, data quantities, and possibilities, but also to globalisation and the use of resources that are not on-site or that have been leased.

Administrators are being confronted with increasing demands and are frequently overburdened and unable to cover the breadth of all of their activities. Because of that, it is imperative to deploy tools to make the work easier, reduce the effort required, generate automatic documentation, and maintain the integrity of all processes.