Active Directory Basics

Learned:

- Windows Domains is an efficient way to manage and secure multiple computers by assigning permissions and access privileges to specific users, allowing them to perform important tasks such as resetting passwords or granting access to certain applications.

- Security Groups are the center of a domains protection as they control user access and permissions across the domain network, ensuring that the right users have access to certain resources.

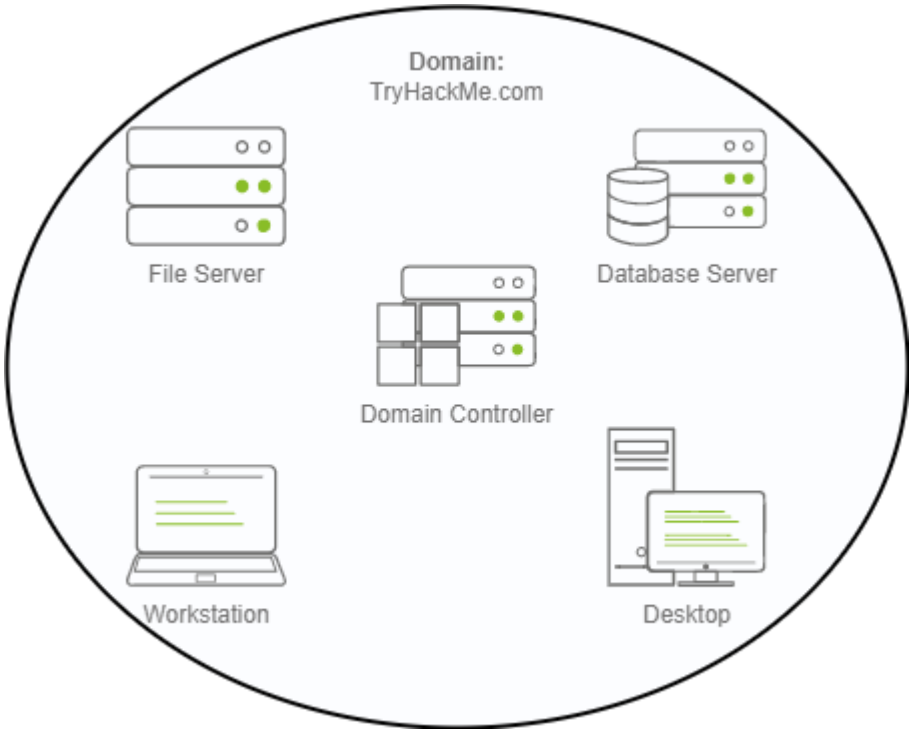
- Group Policies allow administrators to apply rules to an entire Organizational Unit instead of applying it to each user individually, making management easier.

- NetNTLM **and** Kerberos are authentication protocols that help keep credentials secure in a domain. Kerberos uses TGT and TGS to verify a user's identity when accessing resources like websites or databases.

- A user's password hash is stored on the server, allowing authentication systems like NTLM or Kerberos to verify login credentials without exposing the actual password.

Windows Domains

Windows domain is a group of users and computers under the administration of a business. The main idea behind a domain is to centralize the administration of common components of a Windows computer network in a repository called **Active Directory (AD)**. The server that runs the Active Directory services is known as a **Domain Controller (DC)**.



The main advantages of having a configured Windows domain are:

- **Centralised identity management:** All users across the network can be configured from Active Directory with minimum effort.
- **Managing security policies:** You can configure security policies directly from Active Directory and apply them to users and computers across the network as needed.

A Real-World Example

In school/university networks, you will have a username and password that you can use on any of the computers available on campus. Your credentials are valid for all machines because whenever you input them on a machine, it will forward the authentication process back to the Active Directory, where your credentials will be checked. Thanks to Active Directory, your credentials don't need to exist in each machine and are available throughout the network.

Active Directory is also the component that allows your school/university to restrict you from accessing the control panel on your school/university machines.

In a Windows domain, credentials are stored in a centralized repository called Active directory.

The server in charge of running the Active Directory services is called a domain controller.

Active Directory

The core of any Windows Domain is the **Active Directory Domain Service (AD DS)**. This service acts as a catalogue that holds the information of all of the "objects" that exist on your network. Amongst the many objects supported by AD, we have users, groups, machines, printers, shares and many others.

Users

Users are one of the most common object types in Active Directory. Users are one of the objects known as **security principals**, meaning that they can be authenticated by the domain and can be assigned privileges over **resources** like files or printers.

Users can be used to represent two types of entities:

- **People:** users will generally represent persons in your organization that need to access the network, like employees.
- **Services:** you can also define users to be used by services like IIS or MSSQL. Every single service requires a user to run, but service users are different from regular users as they will only have the privileges needed to run their specific service.

Machines

Machines are another type of object within Active Directory; for every computer that joins the Active Directory domain, a machine object will be created. Machines are also considered "security principals" and are assigned an account just as any regular user.

The machine accounts themselves are local administrators on the assigned computer, they are generally not supposed to be accessed by anyone except the computer itself, but as with any other account, if you have the password, you can use it to log in.

The machine account name is the computer's name followed by a dollar sign. For example, a machine named `DC01` will have a machine account called `DC01$`.

Security Groups

If you are familiar with Windows, you probably know that you can define user groups to assign access rights to files or other resources to entire groups instead of single users. This allows for better manageability as you can add users to an existing group, and they will automatically inherit all of the group's privileges. Security groups are also considered security principals and, therefore, can have privileges over resources on the network.

Groups can have both users and machines as members. If needed, groups can include other groups as well.

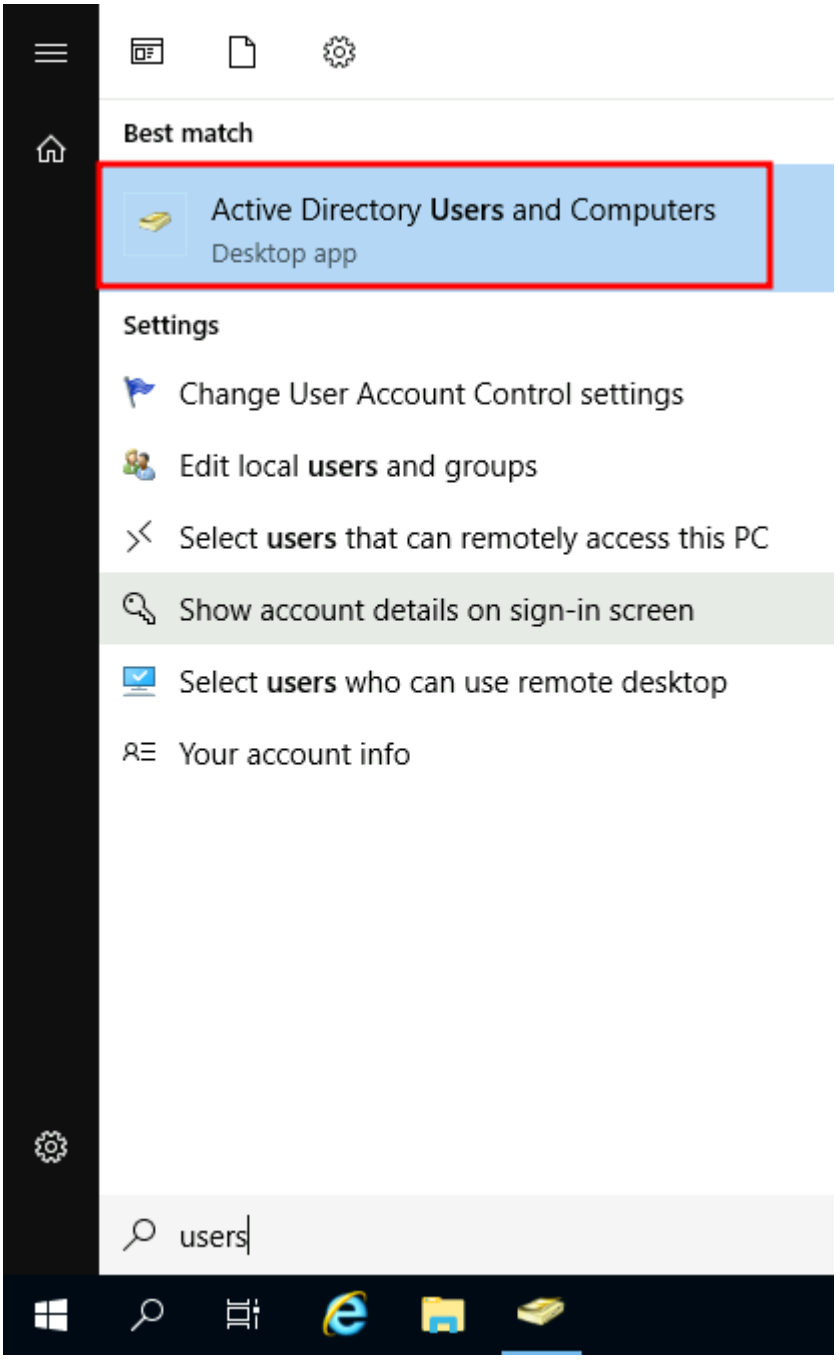
Important Groups in a domain

Security Group	**Description**
Domain Admins	Users of this group have administrative privileges over the entire domain. By default, they can administer any computer on the domain, including the DCs.
Server Operators	Users in this group can administer Domain Controllers. They cannot change any administrative group memberships.
Backup Operators	Users in this group are allowed to access any file, ignoring their permissions. They are used to perform backups of data on computers.
Account Operators	Users in this group can create or modify other accounts in the domain.
Domain Users	Includes all existing user accounts in the domain.

Domain Computers	Includes all existing computers in the domain.
Domain Controllers	Includes all existing DCs on the domain.

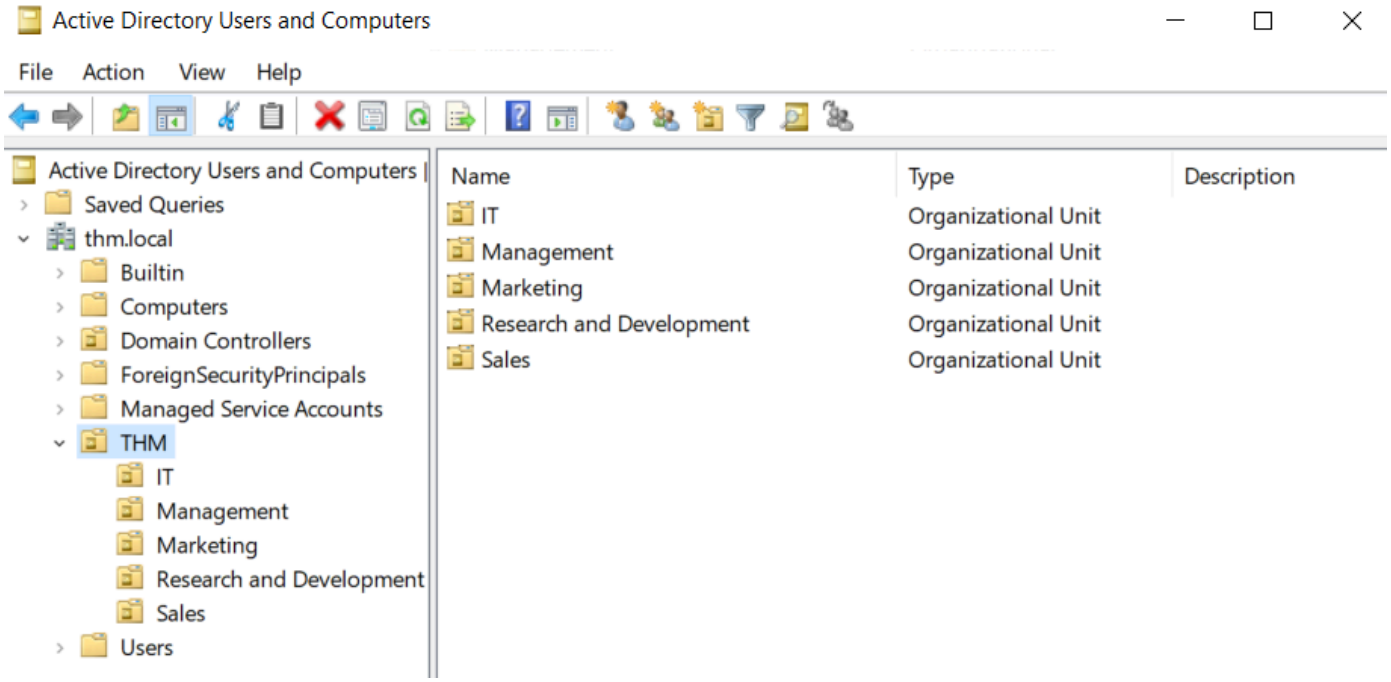
Active Directory Users and Computers

To configure users, groups or machines in Active Directory, you would log in to the Domain Controller and run "Active Directory Users and Computers" from the start menu:



This will open up a window where you can see the hierarchy of users, computers and groups that exist in the domain.

we can see that there is already an OU called `THM` with five child OUs for the IT, Management, Marketing, R&D, and Sales departments. It is very typical to see the OUs mimic the business' structure, as it allows for efficiently deploying baseline policies that apply to entire departments.



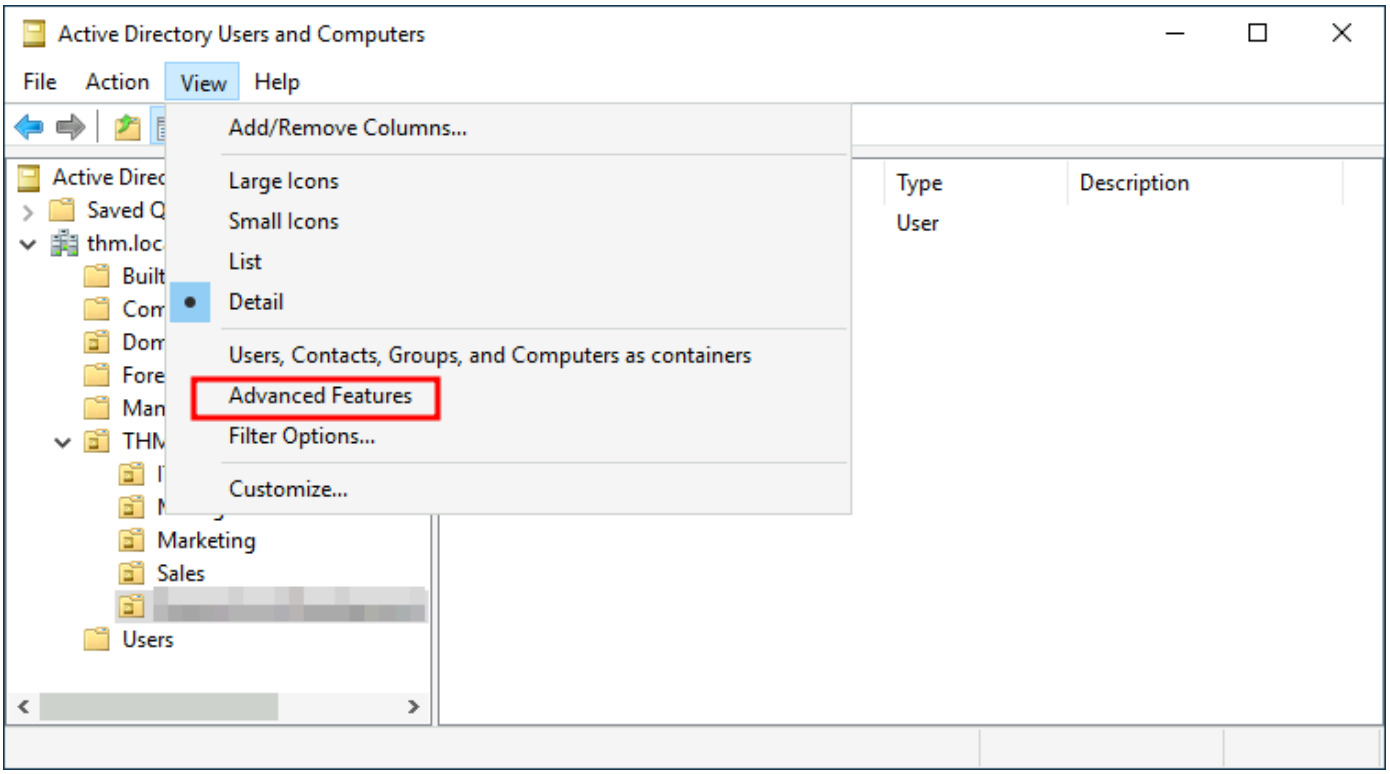
Security Groups vs OUs

OUs are useful for **applying policies** to users and computers, which include specific configurations that are for different sets of users depending on their role.

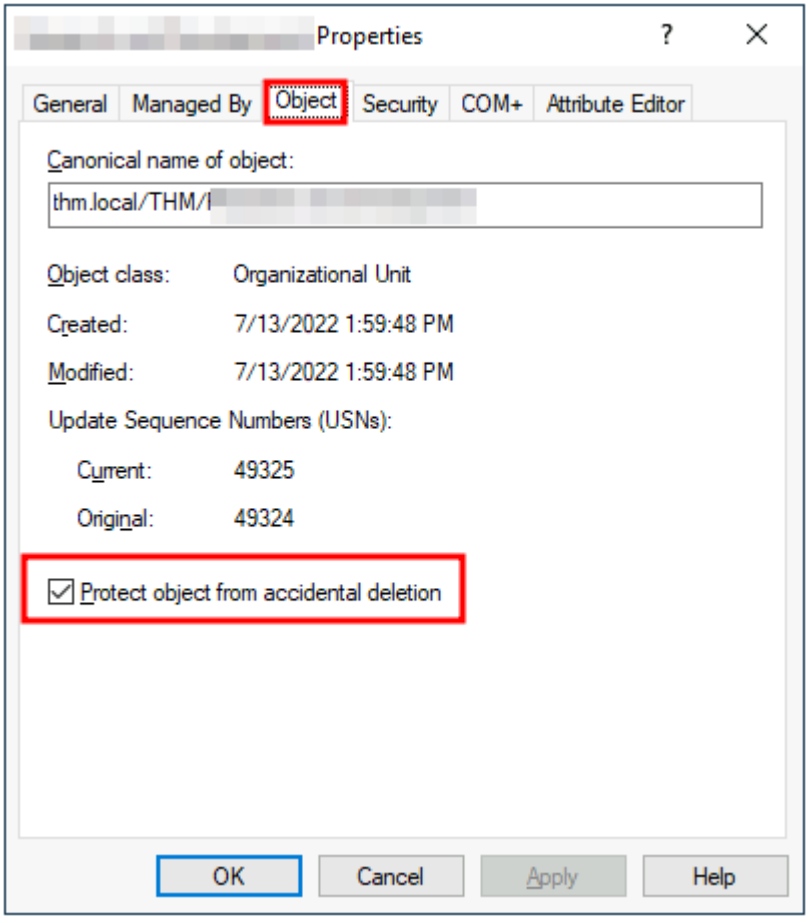
Security Groups, on the other hand, are used to **grant permissions over resources**. For example, you will use groups if you want to allow some users to access a shared folder or network printer.

Managing Users in AD

By default, OUs are protected against accidental deletion. To delete the OU, we need to enable the **Advanced Features** in the View menu:



This will show you some additional containers and enable you to disable the accidental deletion protection. To do so, right-click the OU and go to Properties. You will find a checkbox in the Object tab to disable the protection:



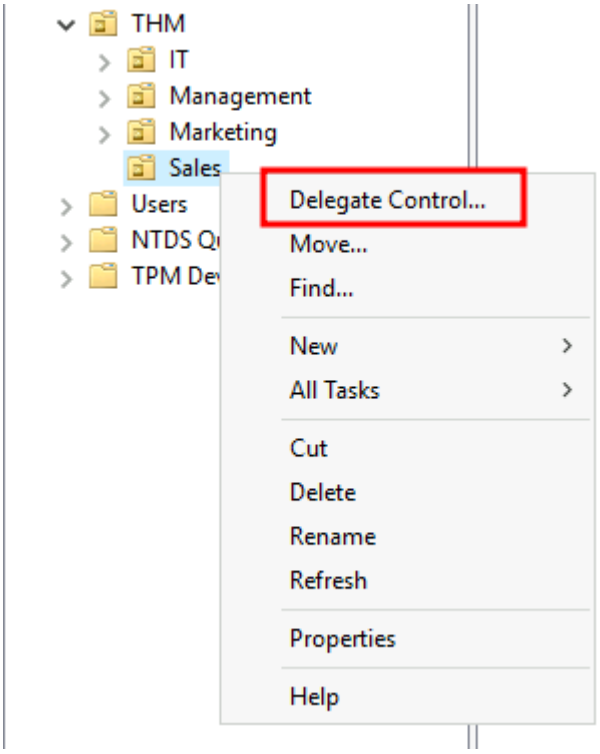
uncheck the box and try deleting the OU again.

Delegation

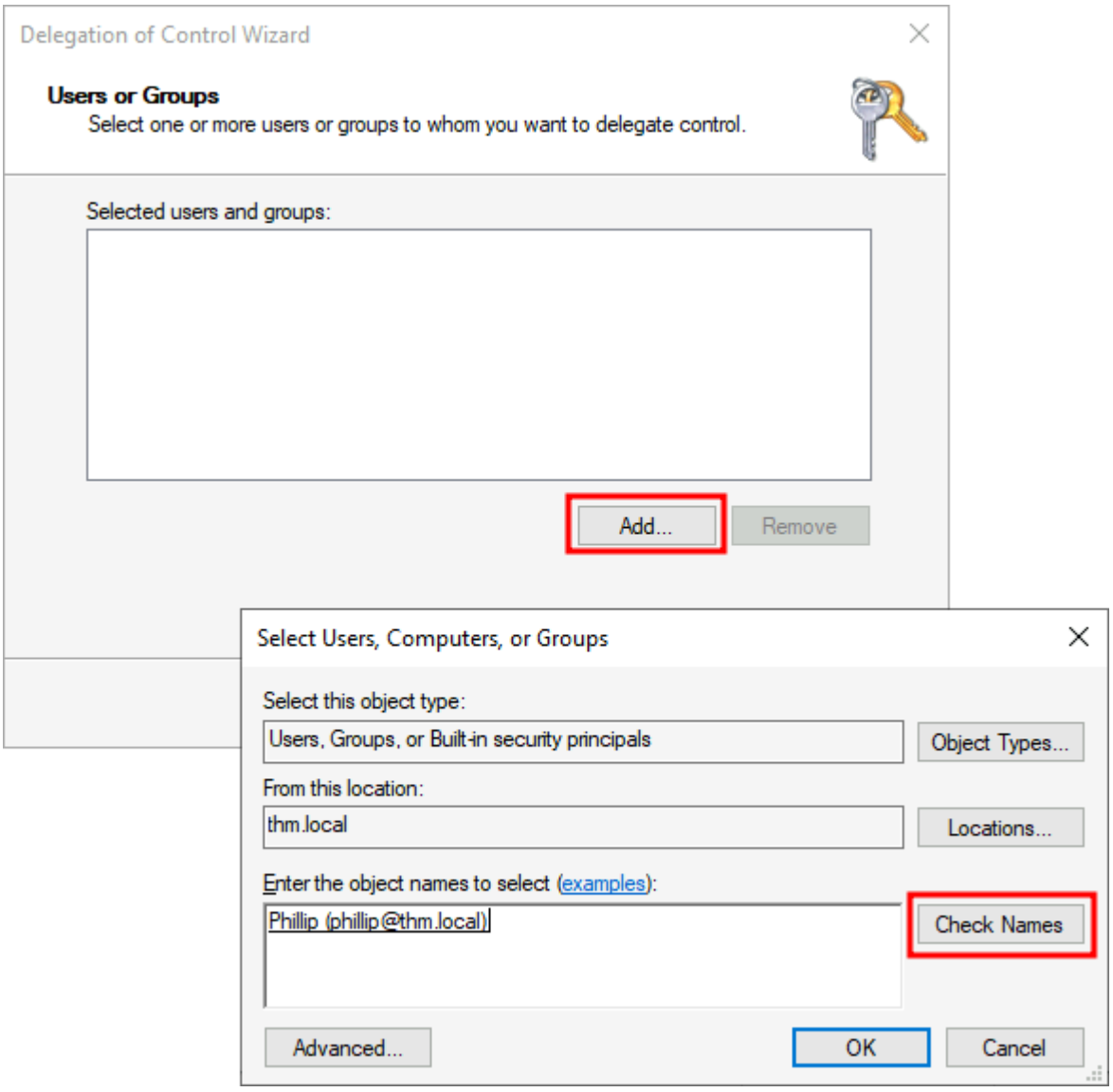
In AD you can give specific users some control over some OUs. This process is known as **delegation** and allows you to grant users specific privileges to perform advanced tasks on OUs without needing a Domain Administrator to step in.

One of the most common use cases for this is granting `IT support` the privileges to reset other low-privilege users' passwords.

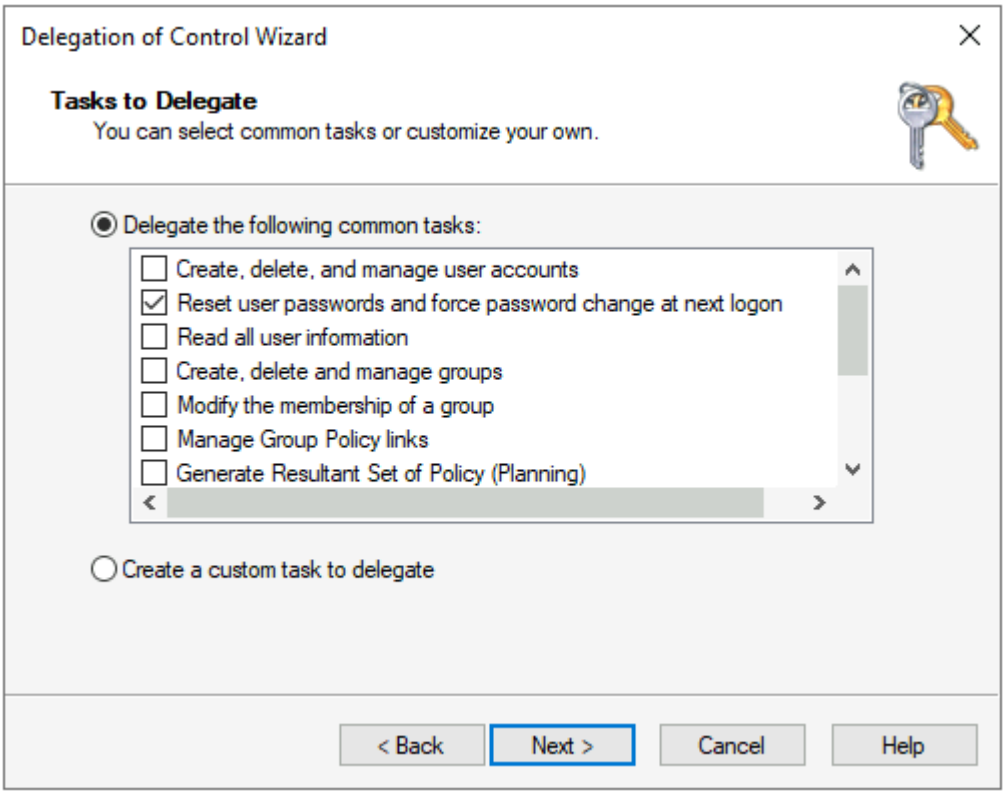
To delegate control over an OU, you can right-click it and select **Delegate Control**:



This should open a new window where you will first be asked for the users to whom you want to delegate control:

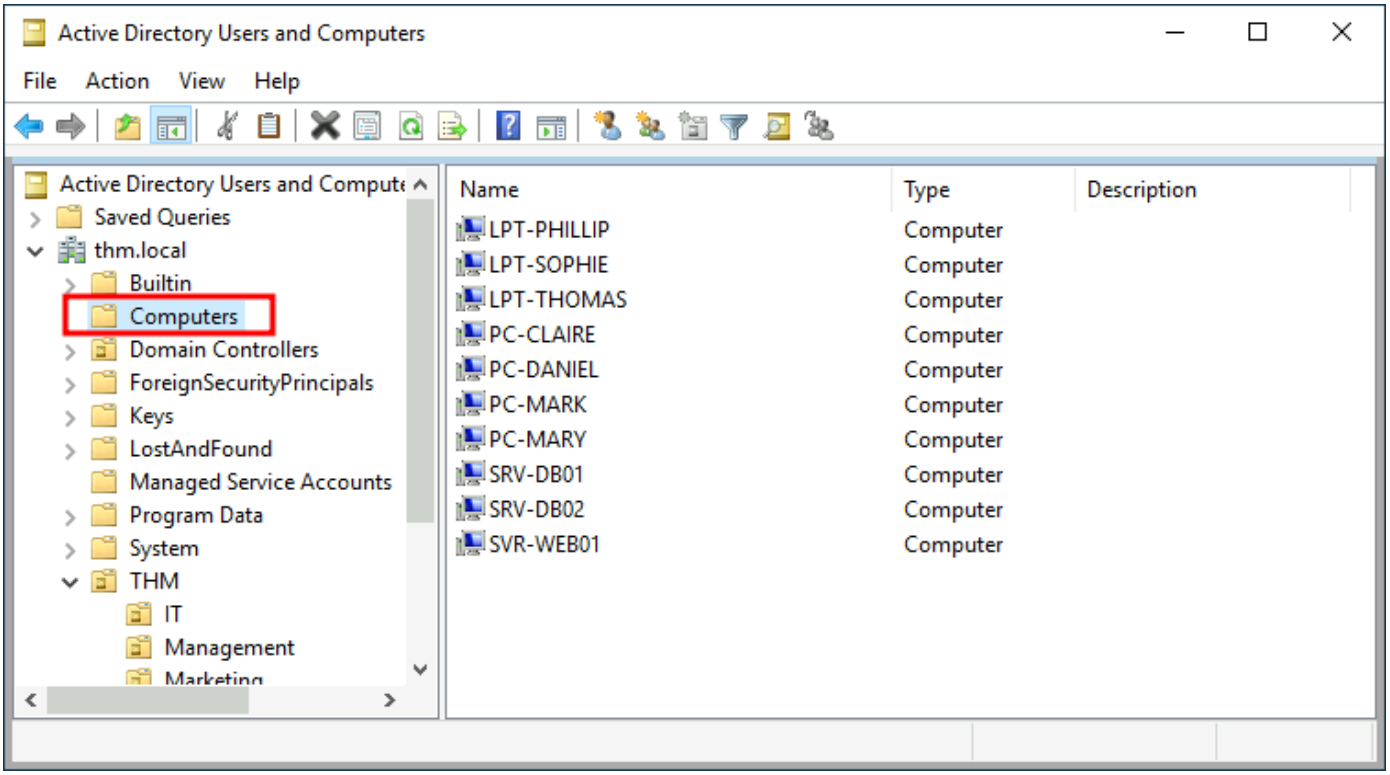


Click OK, and on the next step, select the following option:



Managing Computers in AD

By default, all the machines that join a domain (except for the DCs) will be put in the container called "Computers".



In general, you'd expect to see devices divided into three following categories:

1. Workstations

Workstations are one of the most common devices within an Active Directory domain. Each user in the domain will likely be logging into a workstation. This is the device they will use to do their work or normal browsing activities. These devices should never have a privileged user signed into them.

2. Servers

Servers are the second most common device within an Active Directory domain. Servers are generally used to provide services to users or other servers.

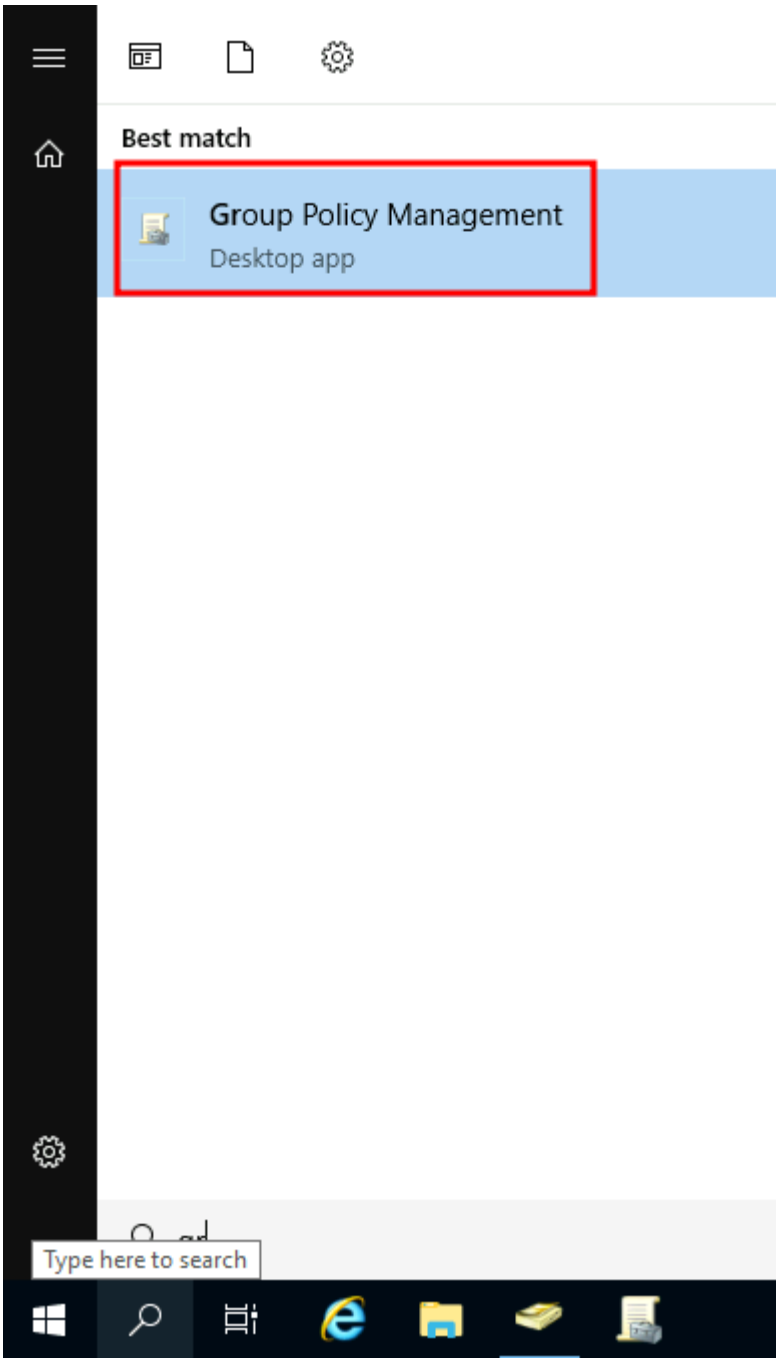
3. Domain Controllers

Domain Controllers are the third most common device within an Active Directory domain. Domain Controllers allow you to manage the Active Directory Domain. These devices are often deemed the most sensitive devices within the network as they contain hashed passwords for all user accounts within the environment.

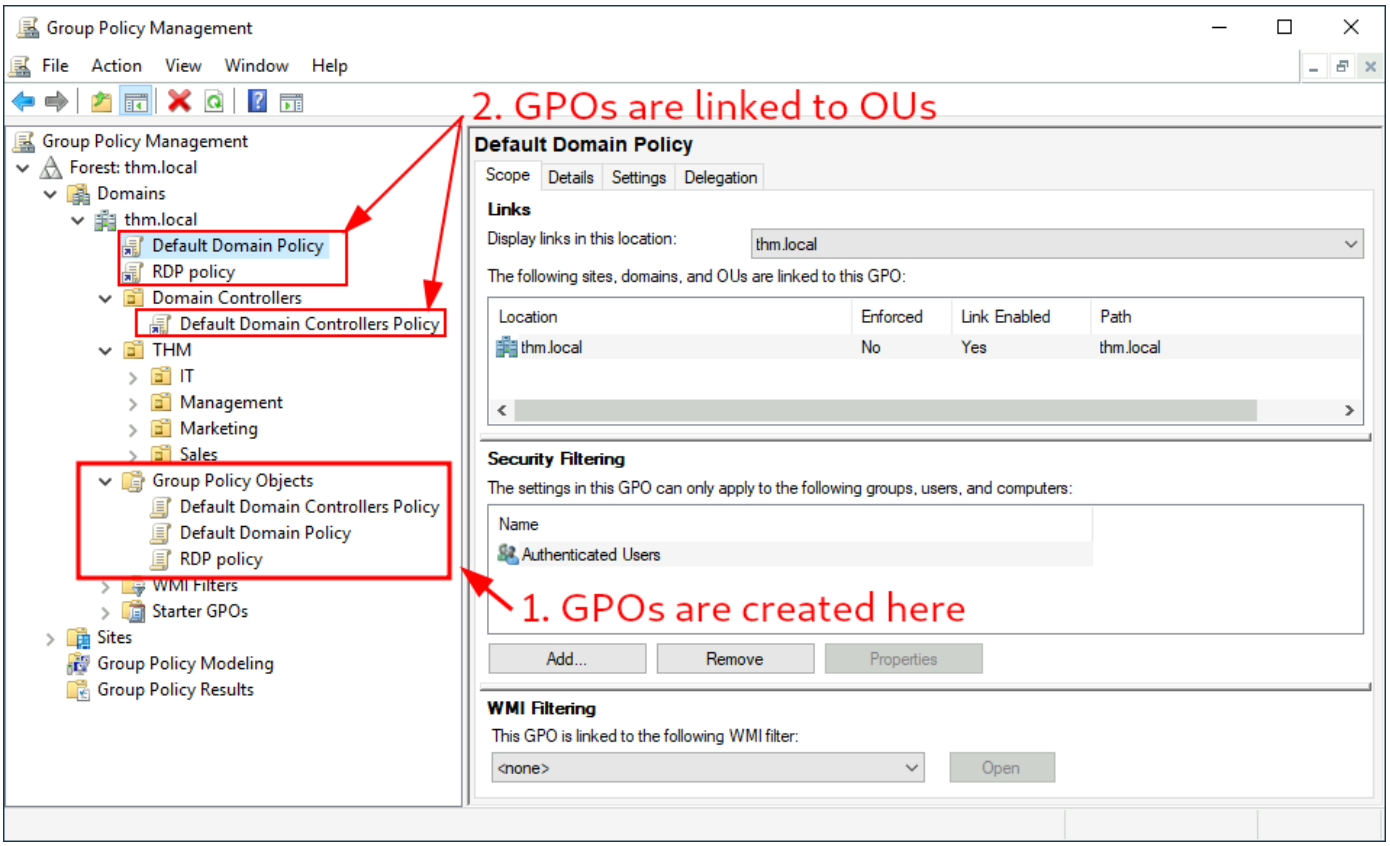
Group Policies

Windows manages such policies through **Group Policy Objects (GPO)**. GPOs are simply a collection of settings that can be applied to OUs. GPOs can contain policies aimed at either users or computers, allowing you to set a baseline on specific machines and identities.

To configure GPOs, you can use the **Group Policy Management** tool, available from the start menu:



The first thing you will see when opening it is your complete OU hierarchy, as defined before. To configure Group Policies, you first create a GPO under **Group Policy Objects** and then link it to the OU where you want the policies to apply.



We can see in the image above that 3 GPOs have been created. From those, the `Default Domain Policy` and `RDP Policy` are linked to the `thm.local` domain as a whole, and the `Default Domain Controllers Policy` is linked to the `Domain Controllers` OU only. Something important to have in mind is that any GPO will apply to the linked OU and any sub-OUs under it. For example, the `Sales` OU will still be affected by the `Default Domain Policy`.

GPO distribution

GPOs are distributed to the network via a network share called `SYSVOL`, which is stored in the DC. All users in a domain should typically have access to this share over the network to sync their GPOs periodically. The SYSVOL share points by default to the `C:\Windows\SYSVOL\sysvol\` directory on each of the DCs in our network.

Once a change has been made to any GPOs, it might take up to 2 hours for computers to catch up. If you want to force any particular computer to sync its GPOs immediately, you can always run the following command on the desired computer:

WindowsPowerShell

```
PS C:\> gpupdate /force
```

Authentication Method

When using Windows domains, all credentials are stored in the Domain Controllers. Whenever a user tries to authenticate to a service using domain credentials, the service will need to ask the Domain

Controller to verify if they are correct. Two protocols can be used for network authentication in windows domains:

- **Kerberos:** Used by any recent version of Windows. This is the default protocol in any recent domain.
- **NetNTLM:** Legacy authentication protocol kept for compatibility purposes.

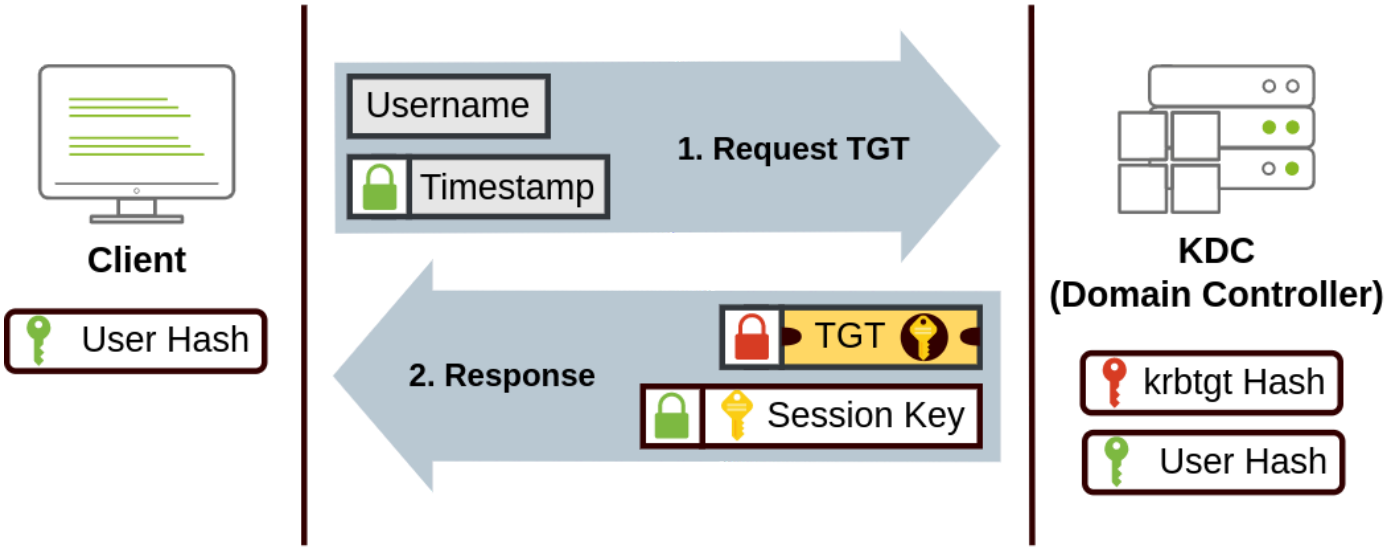
Kerberos Authentication

Kerberos authentication is the default authentication protocol for any recent version of Windows. Users who log into a service using Kerberos will be assigned tickets. Think of tickets as proof of a previous authentication. Users with tickets can present them to a service to demonstrate they have already authenticated into the network before and are therefore enabled to use it.

When Kerberos is used for authentication, the following process happens:

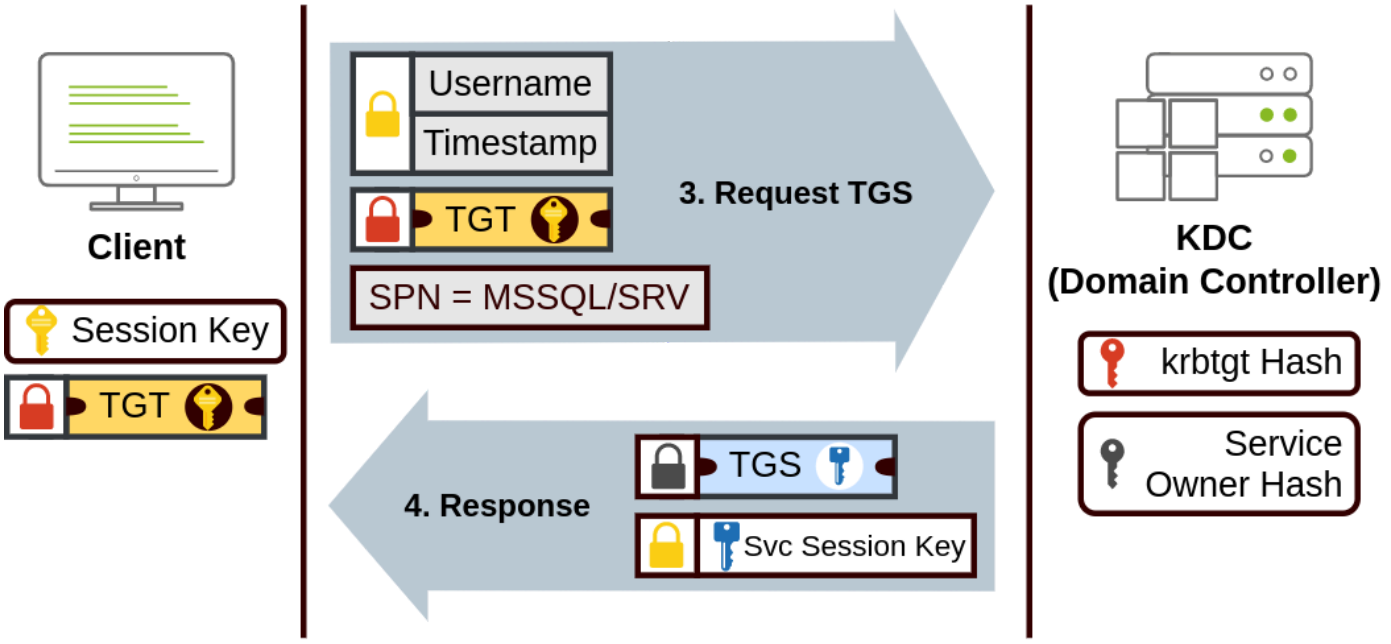
1. The user sends their username and a timestamp encrypted using a key derived from their password to the **Key Distribution Center (KDC)**, a service usually installed on the Domain Controller in charge of creating Kerberos tickets on the network.

   The KDC will create and send back a **Ticket Granting Ticket (TGT)**, which will allow the user to request additional tickets to access specific services. The need for a ticket to get more tickets may sound a bit weird, but it allows users to request service tickets without passing their credentials every time they want to connect to a service. Along with the TGT, a **Session Key** is given to the user, which they will need to generate the following requests.

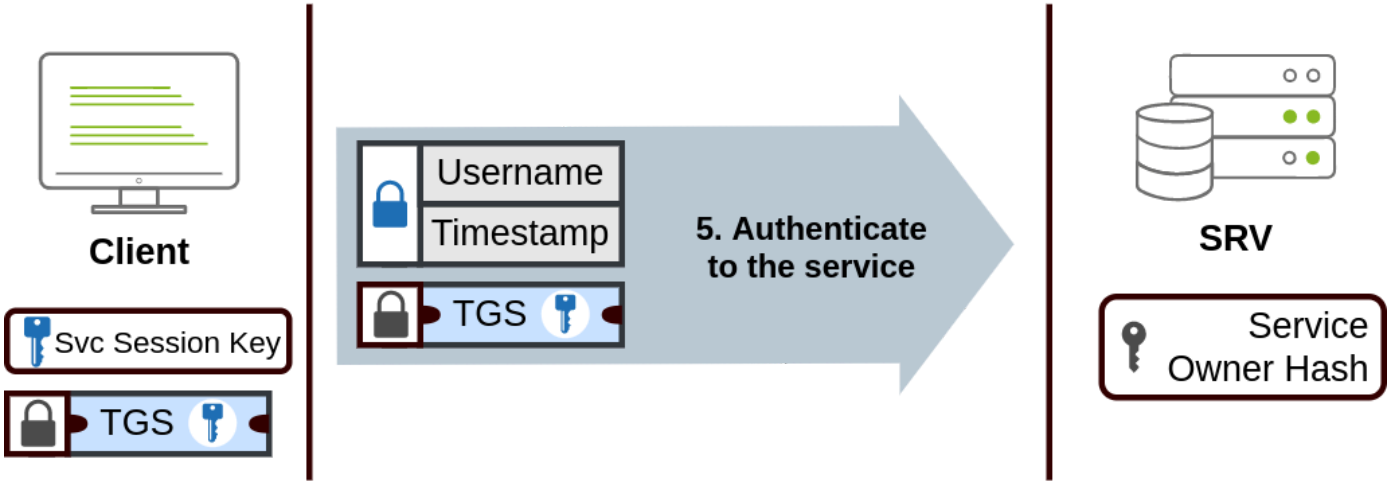


1. When a user wants to connect to a service on the network like a share, website or database, they will use their TGT to ask the KDC for a **Ticket Granting Service (TGS)**. TGS are tickets that allow connection only to the specific service they were created for. To request a TGS, the user will send

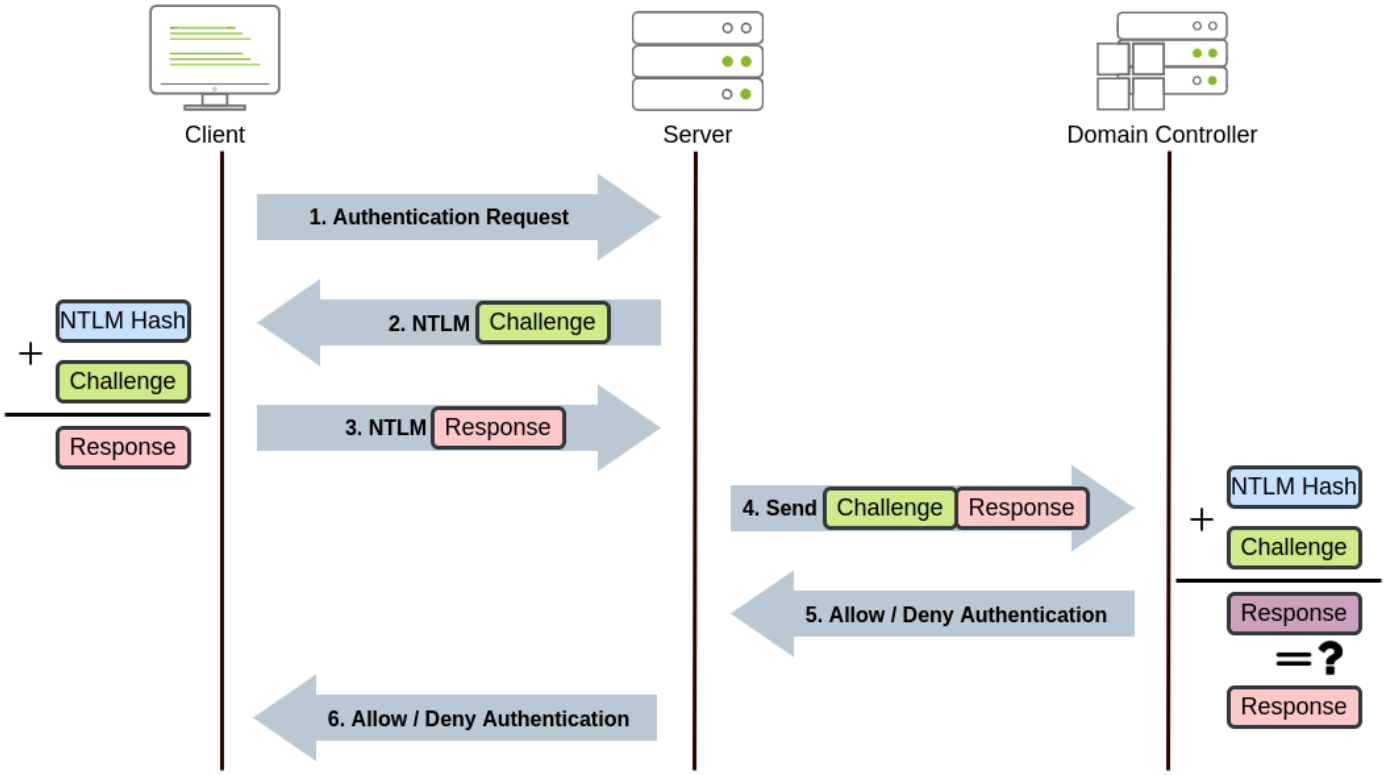
their username and a timestamp encrypted using the Session Key, along with the TGT and a **Service Principal Name (SPN),** which indicates the service and server name we intend to access. As a result, the KDC will send us a TGS along with a **Service Session Key**, which we will need to authenticate to the service we want to access. The TGS is encrypted using a key derived from the **Service Owner Hash**. The Service Owner is the user or machine account that the service runs under. The TGS contains a copy of the Service Session Key on its encrypted contents so that the Service Owner can access it by decrypting the TGS.



1. The TGS can then be sent to the desired service to authenticate and establish a connection. The service will use its configured account's password hash to decrypt the TGS and validate the Service Session Key.

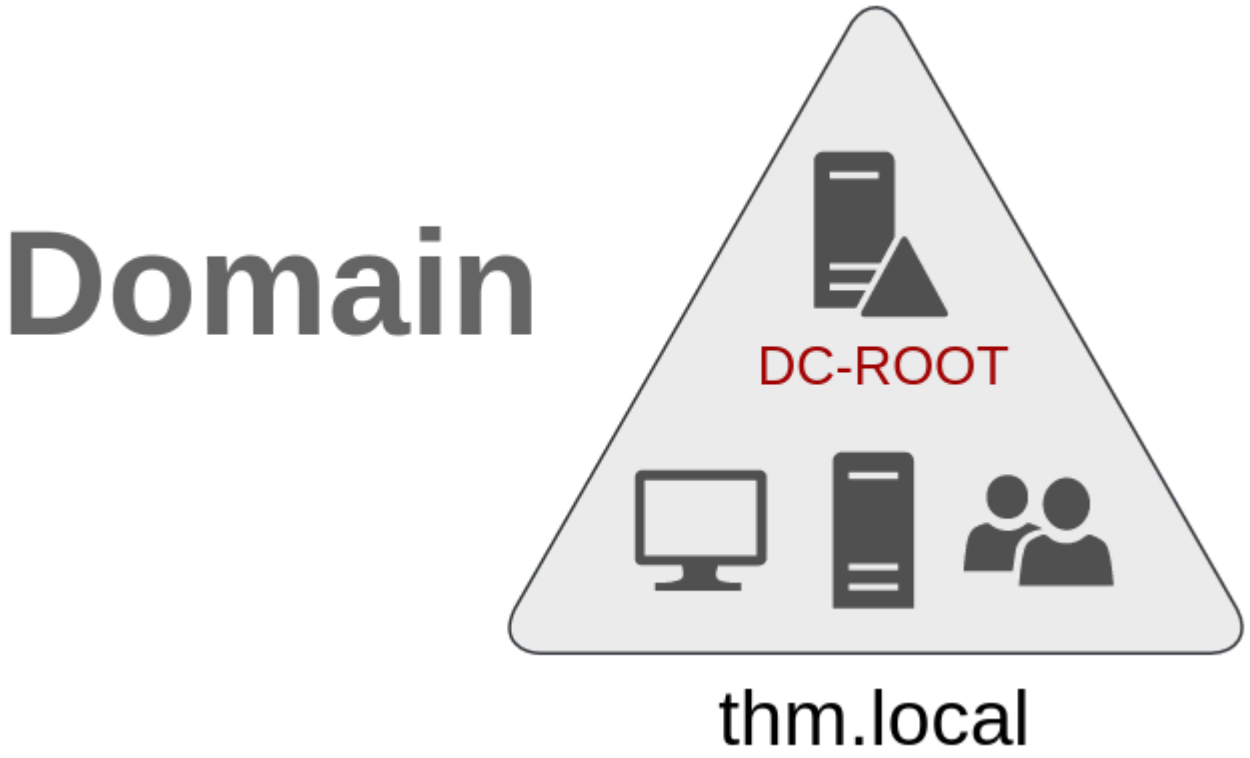


NetNTLM Authentication



1. The client sends an authentication request to the server they want to access.
2. The server generates a random number and sends it as a challenge to the client.
3. The client combines their NTLM password hash with the challenge (and other known data) to generate a response to the challenge and sends it back to the server for verification.
4. The server forwards the challenge and the response to the Domain Controller for verification.
5. The domain controller uses the challenge to recalculate the response and compares it to the original response sent by the client. If they both match, the client is authenticated; otherwise, access is denied. The authentication result is sent back to the server.
6. The server forwards the authentication result to the client.

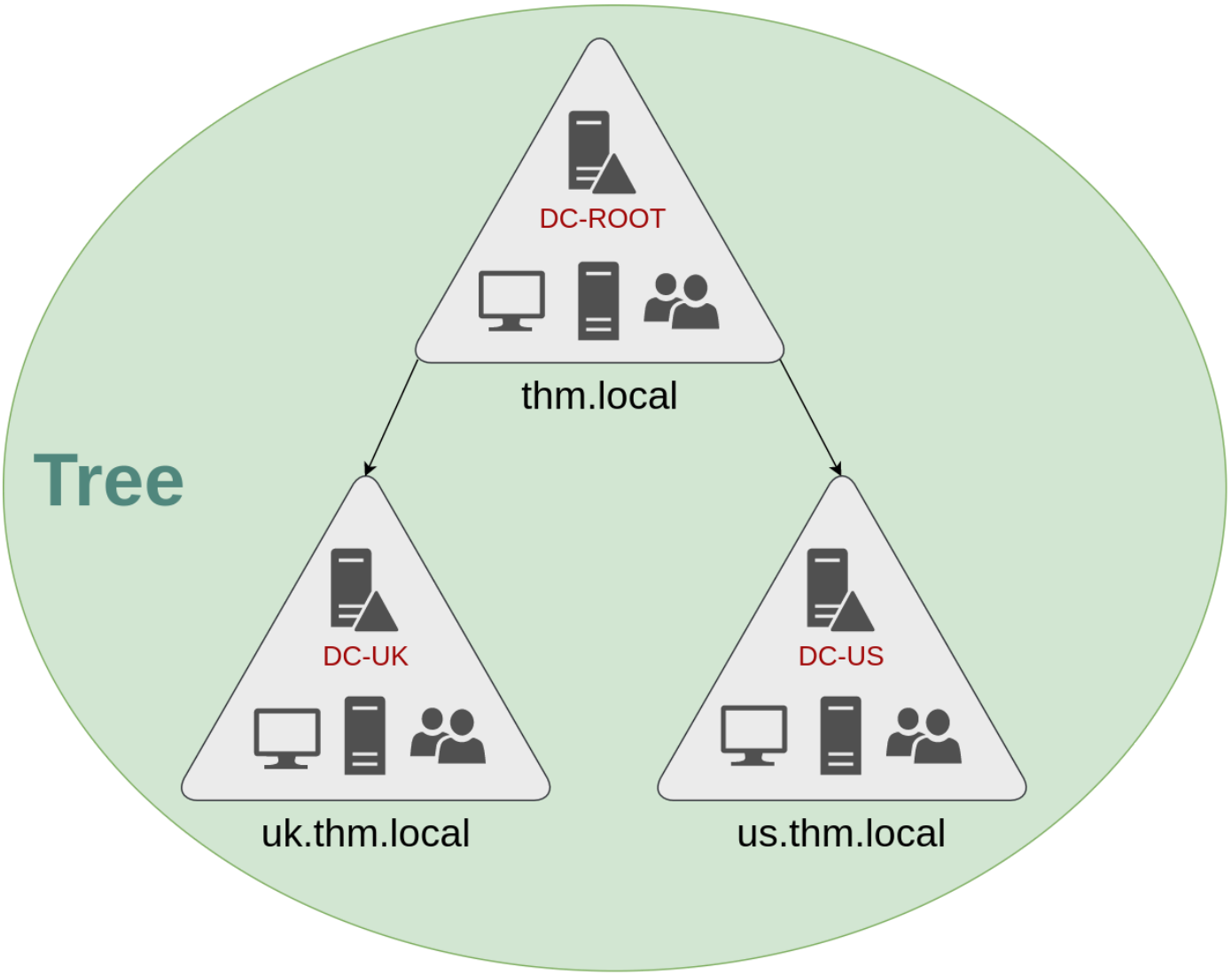
Trees, Forests and Trusts



Trees

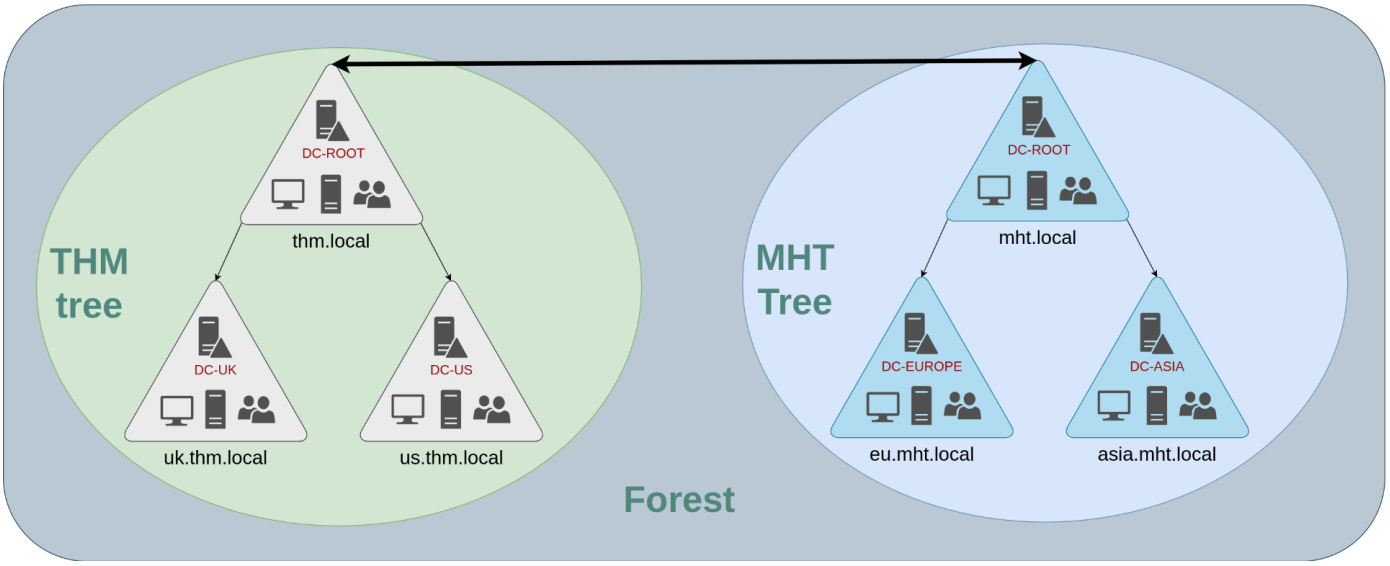
If you have two domains that share the same namespace (`thm.local` in our example), those domains can be joined into a **Tree**.

If our `thm.local` domain was split into two subdomains for UK and US branches, you could build a tree with a root domain of `thm.local` and two subdomains called `uk.thm.local` and `us.thm.local`, each with its AD, computers and users:



Forests

The domains you manage can also be configured in different namespaces. Suppose your company continues growing and eventually acquires another company called `MHT Inc.` When both companies merge, you will probably have different domain trees for each company, each managed by its own IT department. The union of several trees with different namespaces into the same network is known as a **forest**.

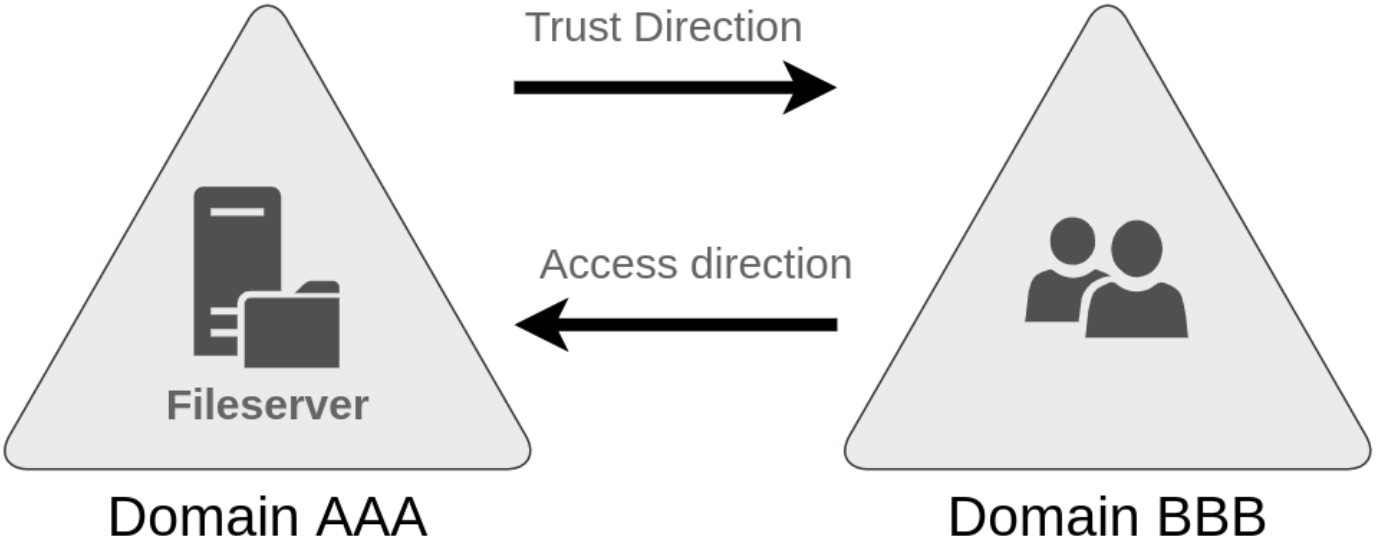


Trust Relationships

At a certain point, a user at THM UK might need to access a shared file in one of MHT ASIA servers. For this to happen, domains arranged in trees and forests are joined together by **trust relationships**.

In simple terms, having a trust relationship between domains allows you to authorize a user from domain `THM UK` to access resources from domain `MHT EU`.

The simplest trust relationship that can be established is a **one-way trust relationship**. In a one-way trust, if `Domain AAA` trusts `Domain BBB`, this means that a user on BBB can be authorised to access resources on AAA:



Two-way trust relationships can also be made to allow both domains to mutually authorise users from the other. By default, joining several domains under a tree or a forest will form a two-way trust relationship.