Active Directory basics

The main idea behind a domain is to centralise the administration of common components of a Windows computer network in a single repository called **Active Directory (AD)**.

The server that runs the Active Directory services is known as a **Domain Controller (DC)**.

- **Managing security policies:** You can configure security policies directly from Active Directory and apply them to users and computers across the network as needed.

Amongst the many objects supported by AD:

-*Users (people,services).*

- *Machines(*computer that joins the Active Directory domain*).*

These objects are organised in **Organizational Units (OUs)** which are container objects that allow you to classify users and machines.

Delegation

allows you to grant users specific privileges to perform advanced tasks on OUs without needing a Domain Administrator to step in. ,, we will try to reset password for another user called ‘sophia’ by the user ‘phillip’ , we can do it through the gui or the Windows PowerShell

```
Set-ADAccountPassword sophie-Reset-NewPassword (Read-Host-AsSecureString-Prompt 'New Password') -Verbose
```

we can also force a password reset at the next logon

```
Set-ADUser-ChangePasswordAtLogon $true-Identity sophie-Verbose
```

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.

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.

it might take up to 2 hours for computers to catch up, so If you want to force any particular computer to sync its GPOs immediately, run the following command

```
gpupdate /force
```

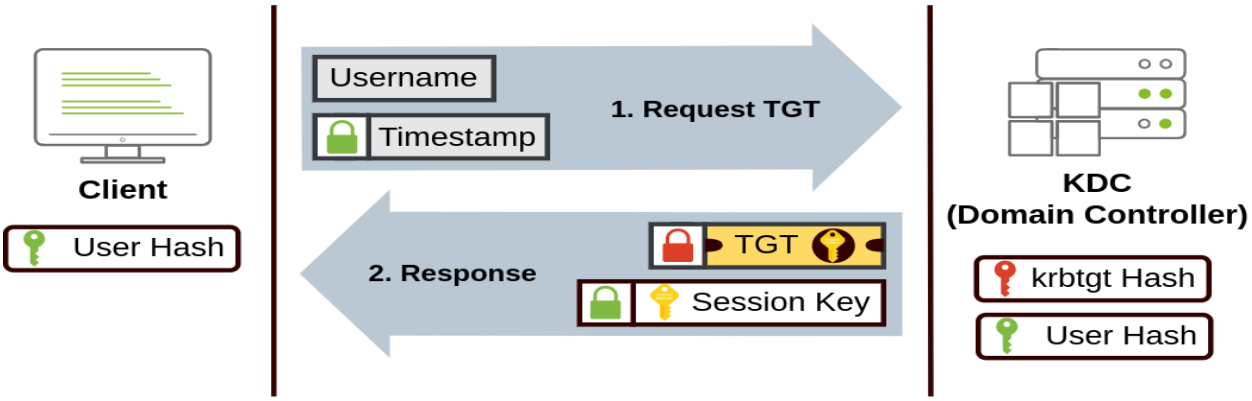
Authentication Methods:

Kerberos : Used by any recent version of Windows. This is the default protocol in any recent domain.

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.

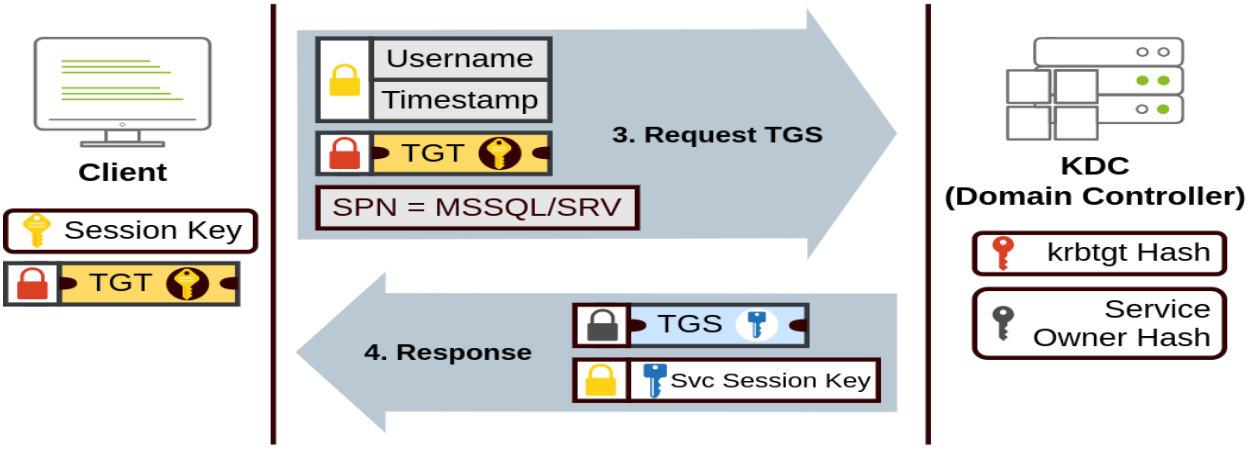
The KDC will create and send back a **Ticket Granting Ticket (TGT)**, which will allow the user to request additional tickets to access specific services. The need for a ticket to get more tickets may sound a bit weird, but it allows users to request service tickets without passing their credentials 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.

Notice the TGT is encrypted using the **krbtgt** account's password hash, and therefore the user can't access its contents. It is essential to know that the encrypted TGT includes a copy of the Session Key as part of its contents, and the KDC has no need to store the Session Key as it can recover a copy by decrypting the TGT if needed.

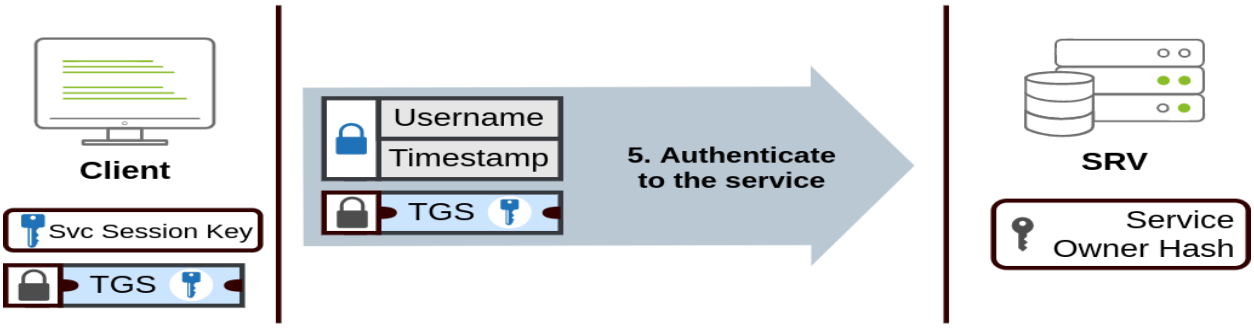


When a user wants to connect to a service on the network like a share, website or database, they will use their TGT to ask the KDC for a **Ticket Granting Service (TGS)**. TGS are tickets that allow connection only to the specific service they were created for. To request a TGS, the user will send their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

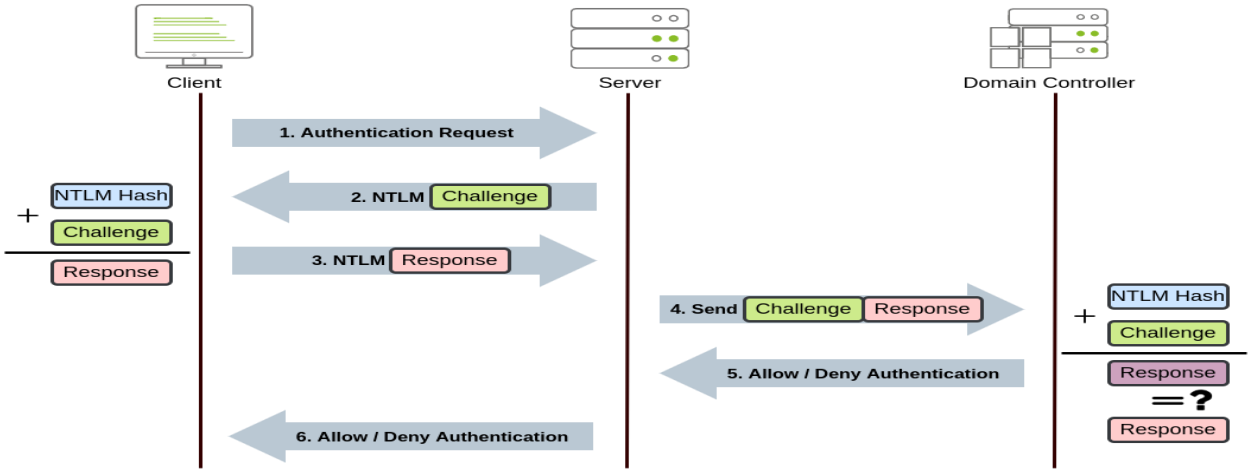


The TGS can then be sent to the desired service to authenticate and establish a connection. The service will use its configured account's password hash to decrypt the TGS and validate the Service Session Key.



NetNTLM : Legacy authentication protocol kept for compatibility purposes.

NetNTLM works using a challenge-response mechanism



Trees: 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: Having multiple domains organised in trees and forest allows you to have a nice compartmentalised network in terms of management and resources. But 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**.