

# Kerberos Protocol

## Intro

Kerberos protocol is an **Authentication Protocol**, specifically the main protocol used in Active Directory used for users and host authentication. The protocol is relatively old, and was created in 1983, developed by MIT.

In Windows Server 2000, Microsoft replaced their protocol, **NTLM** as the authentication default protocol, for Kerberos protocol. Specifically for **Authentication Against Domain**.

NTLM keeps being used as local host authentication, that is, to users that do not belong to a Domain.

Kerberos It is the oldest commonly used protocol today.

Is very important to know how this protocol works, because *many of the most prominent Active Directory hacking techniques* will target Kerberos protocol.

## ? How Kerberos Protocol Works?

In the past people used to work using one terminal with poor capacity that sent commands and orders to a central computer that was used by all users. That central computer, or workstation, had to share its resources.

So seeing the results of this, The need arises for each user to have their own workstation.

But, how does Kerberos work in Active Directory? Step by step there is a explication of that:

## Key Concepts

- **KDC (Key Distribution Center)** : This is a key service in Kerberos and is split into two parts in AD: the **TGS (Ticket Granting Server)** and the **AS (Authentication Server)**.
- **Tickets**: These are temporary credentials that prove the identity of a user or service.
- **TGT (Ticket Granting Ticket)**: This is an authentication ticket issued once the user logs in and can be used to obtain other tickets to access different services.

## Kerberos Authentication Process:

When a user attempts to access a service in an Active Directory environment, Kerberos follows this process:

### User Login

- The user enters their username and password on their machine.

- The system sends a request to the **AS (Authentication Server)** of the **KDC**. This request contains the username, but the password is never transmitted directly. Instead, the AS uses an encrypted version of the password to authenticate the user.

#### **Authentication and TGT Issuance:**

- If the AS verifies the user's identity (by querying the Active Directory database), it responds with a **TGT (Ticket Granting Ticket)** encrypted with the user's secret key.
- The **TGT** contains information about the user's identity and a validity period. This ticket is valid for a limited time, and only the **TGS** can issue it.

#### **Requesting Access to a Service (TGS):**

- The user wants to access a service (e.g., a database or file server). To do so, the user sends a request to the **TGS** of the **KDC** along with their **TGT**.
- The **TGS** checks the **TGT**, and if it's valid, it issues a **service ticket** specifically for the service being requested. This ticket is encrypted with the service's secret key (e.g., the file server).




#### **Accessing the Service:**

- The user presents the **service ticket** to the server providing the service (e.g., a file server).
- The server verifies the ticket with the **KDC**, and if everything is correct, it grants access to the service.

#### **Ticket Expiration:**

- Both the **TGT** and **service tickets** have a limited lifetime. After they expire, the user needs to request a new ticket.

### **Advantages of Using Kerberos in Active Directory:**

-  **Security:** Kerberos uses encryption to protect credentials, preventing passwords from being transmitted over the network. Additionally, each communication between the client and servers is secured.
-  **Decentralization:** Kerberos allows services to authenticate with each other without sharing user passwords.
-  **Scalability:** Kerberos is suitable for large, distributed environments like Active Directory domains, as it efficiently handles user and service authentication across multiple servers.

## Summary of the Kerberos Flow in AD:

1. The user enters their credentials.
2. The **AS** issues a **TGT** after verifying the identity.
3. The user requests a **service ticket** from the **TGS** using the **TGT**.
4. The **TGS** issues a **service ticket** for access to the requested resource.
5. The user presents the **service ticket** to the target server.
6. The server validates the ticket and grants access.