Name:erfan ahmad beigi

Master :mr.milad nourian nasab

What is Kerberos and how does it work?



Kerberos is **a computer network security protocol that authenticates service requests between two or more trusted hosts across an untrusted network, like the internet**. It uses secret-key cryptography and a trusted third party for authenticating client-server applications and verifying users' identities

What is Kerberos?

In mythology, Kerberos (also known as Cerberus) is **a large, three-headed dog that guards the gates to the underworld to keep souls from escaping**. In our world, Kerberos is the computer network authentication protocol initially developed in the 1980s by Massachusetts Institute of Technology (MIT) computer scientists.

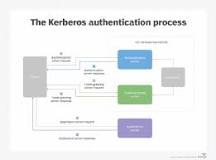
What are the 3 main parts of Kerberos?

Kerberos has three parts: a client, server, and trusted third party (KDC) to mediate between them. Clients obtain tickets from the Kerberos Key Distribution Center (KDC), and they present these tickets to servers when connections are established

Why is Kerberos important?

Kerberos is designed **to completely avoid storing any passwords locally or having to send any passwords through the internet and provides mutual authentication**, meaning both the user and the server's authenticity are verified.

Where is Kerberos used?



Kerberos is used **to authenticate entities requesting access to network resources, especially in large networks to support SSO**. The protocol is used by default in many widely used networking systems. Some systems in which Kerberos support is incorporated or available include the following: Amazon Web Services.

Is Kerberos Active Directory?

However, **Microsoft Active Directory is the most widely consumed Kerberos implementation**. It is based on Kerberos Network Authentication Service (V5).

What port is Kerberos?

port 88

Kerberos clients need to send UDP and TCP packets on **port 88** and receive replies from the Kerberos servers.

What kind of protocol is Kerberos?

network authentication protocol

Kerberos is a **network authentication protocol**. It is designed to provide strong authentication for client/server applications by using secret-key cryptography. A free implementation of this protocol is available from the Massachusetts Institute of Technology. Kerberos is available in many commercial products as well.

How Kerberos works step by step?

**Below are the steps required to authenticate through Kerberos:**

1. Step 1: The User Sends a Request to the AS. ...
2. Step 2: The AS Issues a TGT. ...
3. Step 3: The User Sends a Request to the TGS. ...
4. Step 4: TGS Issues a Service Ticket. ...
5. Step 5: The User Contacts the File Server with the Service Ticket. ...
6. Step 6: The User Opens the Document.
7. What is the main feature of Kerberos?
8. Kerberos **provides a centralized authentication server whose function is to authenticate users to servers and servers to users**. In Kerberos Authentication server and database is used for client authentication. Kerberos runs as a third-party trusted server known as the Key Distribution Center (KDC)

Does Kerberos encrypt data?

**A Kerberos ticket is encrypted data that's issued for authentication**. Tickets are issued by a Key Distribution Center (KDC), which is a service that runs on every DC. When a user logs on, the user authenticates to Active Directory using a password or smart card.

**What does the Kerberos authentication protocol do?**

The original objective of Kerberos was to provide a way for users of the MIT network to securely authenticate themselves to the systems they needed to use. It also enabled those users to be [authorized](https://www.techtarget.com/searchsoftwarequality/definition/authorization) to access those systems.

At that time, networked systems typically authenticated users with a user ID and password combination. Systems routinely transmitted passwords "in the clear," meaning [unencrypted](https://www.techtarget.com/searchsecurity/definition/encryption). Attackers with access to the network could easily eavesdrop on network transmissions, intercept user IDs and passwords, and then attempt to access systems for which they were not authorized.

Kerberos developers set out to provide a network authentication protocol that could be used to authenticate trusted hosts communicating over untrusted networks. In particular, they intended to provide system administrators a mechanism for authenticating access to systems over an open network -- the internet.

Kerberos was initially designed as the "Kerberos Authentication and Authorization System" in a [paper](https://web.mit.edu/Saltzer/www/publications/athenaplan/e.2.1.pdf) with the same name written by S.P. Miller, B.C. Neuman, J.I. Schiller and J.H. Saltzer. The designers aimed to provide a foundation for ensuring that only authorized users can get access to specific networked resources. They intended Kerberos' authentication as a means for supporting authorization. While a user can be authenticated as having some access rights to some network resources, authorization tools enable more finely grained access to specific resources, like storage and databases.

Kerberos was also designed to interface with secure accounting systems. This provided the third "A" of the authentication, authorization and accounting ([AAA](https://www.techtarget.com/searchsecurity/definition/authentication-authorization-and-accounting)) triad.

### Kerberos objectives, concepts and terms

Goals for the Kerberos system are spelled out in a [tutorial](https://www.kerberos.org/software/tutorial.html) written by Fulvio Ricciardi of the National Institute of Nuclear Physics in Lecce, Italy. They include the following:

* Passwords must never be transmitted over the network.
* Passwords must never be stored on client systems and must always be discarded immediately after they are used.
* Passwords are never stored in [plaintext](https://www.techtarget.com/searchsecurity/definition/plaintext), even on the [authentication servers](https://www.techtarget.com/searchsecurity/definition/authentication-server).
* A password is entered only once each session. This is an early form of single sign-on ([SSO](https://www.techtarget.com/searchsecurity/definition/single-sign-on)) authentication, and it means that users can authenticate themselves just once but still access any systems for which they are authorized.
* All authentication information is maintained in a centralized authentication server. The application servers themselves do not store any authentication information. This enables the following features:
  + An administrator can disable authorization for a user to use any application server from the centralized authentication server. Access to individual servers is not necessary to revoke authorization.
  + A single user password is enough to access all Kerberos-authenticated services. A user can reset their password just once, no matter how many services they are authenticated to use.
  + Protecting user information is simplified since all user authentication information is stored on one centralized authentication server rather than on all the individual servers the user is authorized to use.
* All parties -- users, as well as application servers -- must authenticate themselves when prompted. Users authenticate when they sign in. Application services may be required to authenticate themselves to the client.
* Kerberos provides a mechanism for clients and servers to set up an encrypted circuit so that networked communications are private.

**How does the Kerberos authentication protocol work?**

A simplified description of how Kerberos works follows; the actual process is more complicated and may vary from one implementation to another:

1. **Authentication server request.** To start the Kerberos client authentication process, the initiating client sends an authentication request to the Kerberos KDC authentication server. The initial authentication request is sent as plaintext because no [sensitive information](https://www.techtarget.com/whatis/definition/sensitive-information) is included in the request. The authentication server verifies that the client is in the KDC database and retrieves the initiating client's [private key](https://www.techtarget.com/searchsecurity/definition/private-key).
2. **Authentication server response.** If the initiating client's username isn't found in the KDC database, the client cannot be authenticated, and the authentication process stops. Otherwise, the authentication server sends the client a TGT and a [session key](https://www.techtarget.com/searchsecurity/definition/session-key).
3. **Service ticket request.**Once authenticated by the authentication server, the client asks for a service ticket from the TGS. This request must be accompanied by the TGT sent by the KDC authentication server.
4. **Service ticket response.** If the TGS can authenticate the client, it sends credentials and a ticket to access the requested service. This transmission is encrypted with a session key specific to the user and service being accessed. This proof of identity is used to access the requested "Kerberized" service. That service validates the original request and then confirms its identity to the requesting system.
5. **Application server request.** The client sends a request to access the application server. This request includes the service ticket received in step 4. If the application server can authenticate this request, the client can access the server.
6. **Application server response.** In cases where the client requests the application server to authenticate itself, this response is required. The client has already authenticated itself, and the application server response includes Kerberos authentication of the server.