KERBEROS

PRÉSENTER: RUBEN SERRAF

What is Kerberos?

It is a network authentication protocol that uses third-party authorization for validating the profiles of users. It also employs symmetric key cryptography for plaintext encryption and ciphertext decryption. The keys in cryptography consist of a secret key that shares confidential information between two or more objects.

How Does It Work?

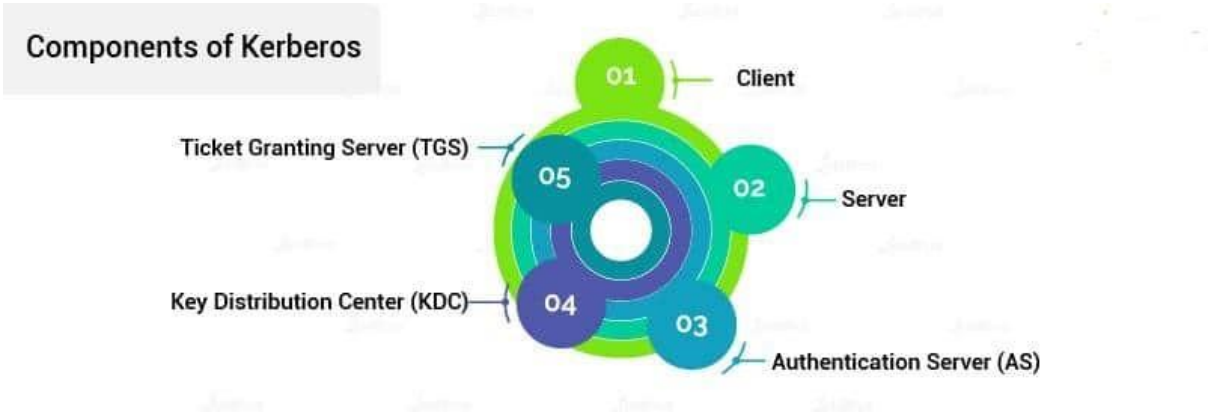
Kerberos being an authentication protocol. It is proved to be one of the essential components of client/server applications and used in various fields for network security, providing mutual authentication. In this section, we will discuss how Kerberos works. For that, first, we need to know the components of it.

Components of Kerberos

It mainly provides two services, and they are:

- Authentication service
- Ticket granting service

For providing these services, it uses its various components. Further, let's discuss these principal components used for authentication:



1. Client: The client helps initiate a service request for communicating with the user.

2. Server: All the services required by the user are hosted by the server.

3. Authentication server (AS): As the name suggests, it is used for the authentication of the client and the server. It assigns a ticket through TGT (ticket granting ticket) to the client. The assigned ticket ensures the authentication of the client to other servers.

4. Key distribution center (KDC): There are three parts to the Kerberos authentication service:

- Database
- Ticket granting service (TGS)
- Authentication server (AS)

These parts reside in a single unit known as the key distribution center.

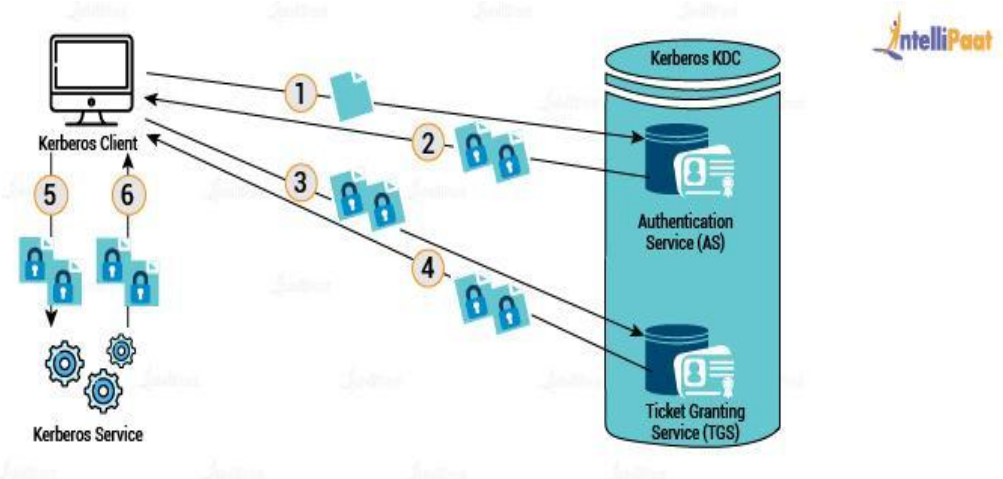
5. Ticket granting server (TGS): This server provides a service to assign tickets to the user as a unique key for authentication.

There are unique keys used by the authentication server and the TGS for both the clients and the servers. Now, let’s look at the cryptographic secret keys used for authentication:

- Client/User secret key: It is the hash of the password set by the user that acts as the client/user secret key.
- TGS secret key: It is the secret key that helps in deciding TGS.
- Server secret key: The server secret key helps determine the server that is providing the services.

Architecture of Kerberos

Now, we will understand how Kerberos works by checking out its architecture. The below diagram shows the workflow of the Kerberos protocol:



Here are the steps involved in the Kerberos workflow:

Step 1: Initially, there is an authentication request from the client. The user requests TGS from the authentication server.

Step 2: After the client's request, the client data is validated by KDC. The authentication server verifies the client and the TGS from the database.

It then generates a cryptographic key (SK1) after checking both the values, implementing the hash of the password. The authentication server also computes a session key. This session key uses the secret key (SK2) of the client for the encryption.

Step 3: Then, the authentication server creates a ticket that consists of the ID, network address, secret key, and the lifetime of the client.

Step 4: The decryption of the message is then performed by the client using the client's secret key.

Step 5: Now, the client demands entrance into the server using TGS. The ticket generating service creates a ticket that acts as an authenticator here.

Step 6: Another ticket is generated by KDC for the file server. Then, the TGS decrypts the ticket for obtaining the secret key initiated by the client. It checks the network address and the ID by decrypting the authenticator. If the client ID and the network address match successfully, then KDC shares a service key with the client and the server.

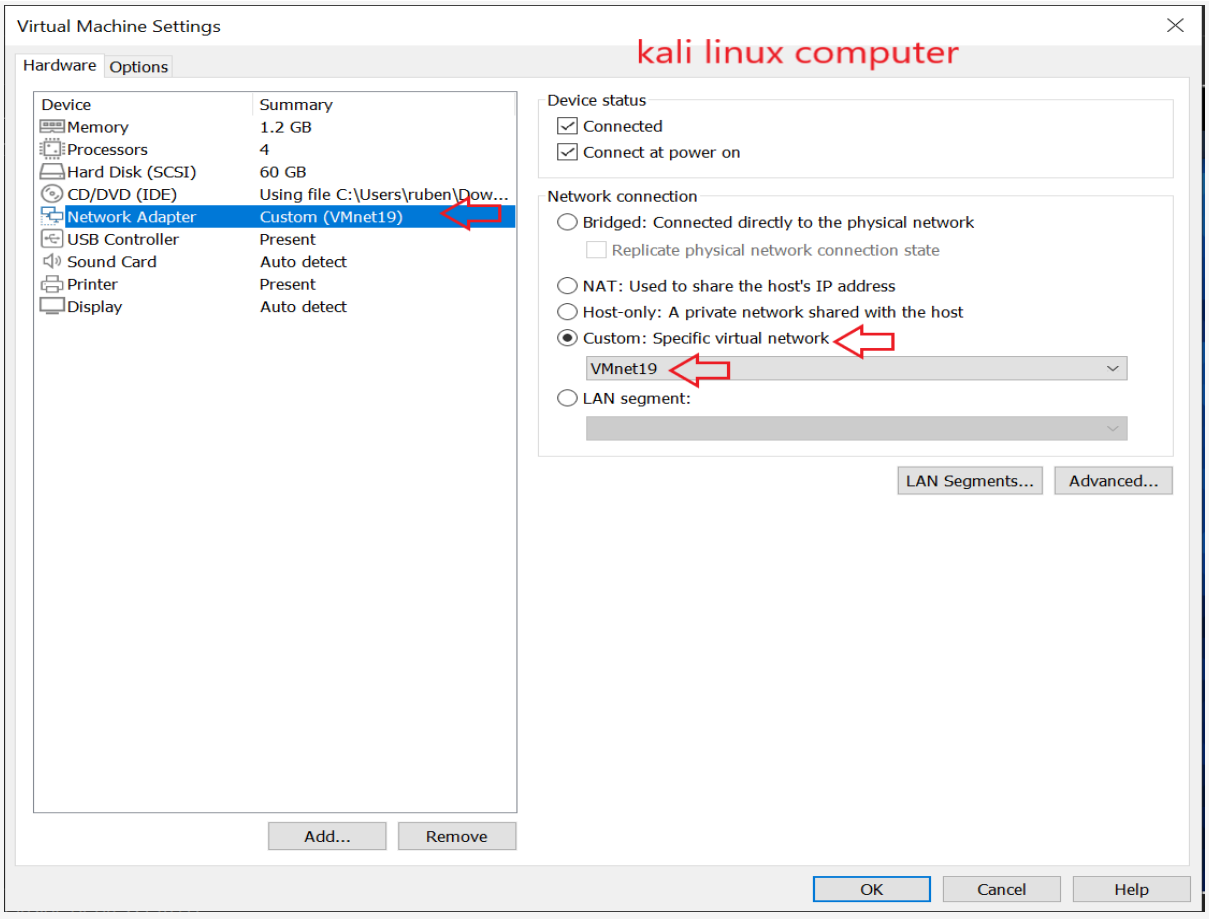
Step 7: The client utilizes the file ticket for authentication. The message is decrypted using SK1 to obtain SK2. Again, the TGS generates a new ticket to send to the target server.

Step 8: Here, the target server decrypts the file ticket by using the secret key. After that, the server performs checks on the client details by decrypting SK2. The target server also checks the validity of the ticket. Finally, when all the client's encrypted data is decrypted, and the data is verified, the server authenticates the client to use the services.

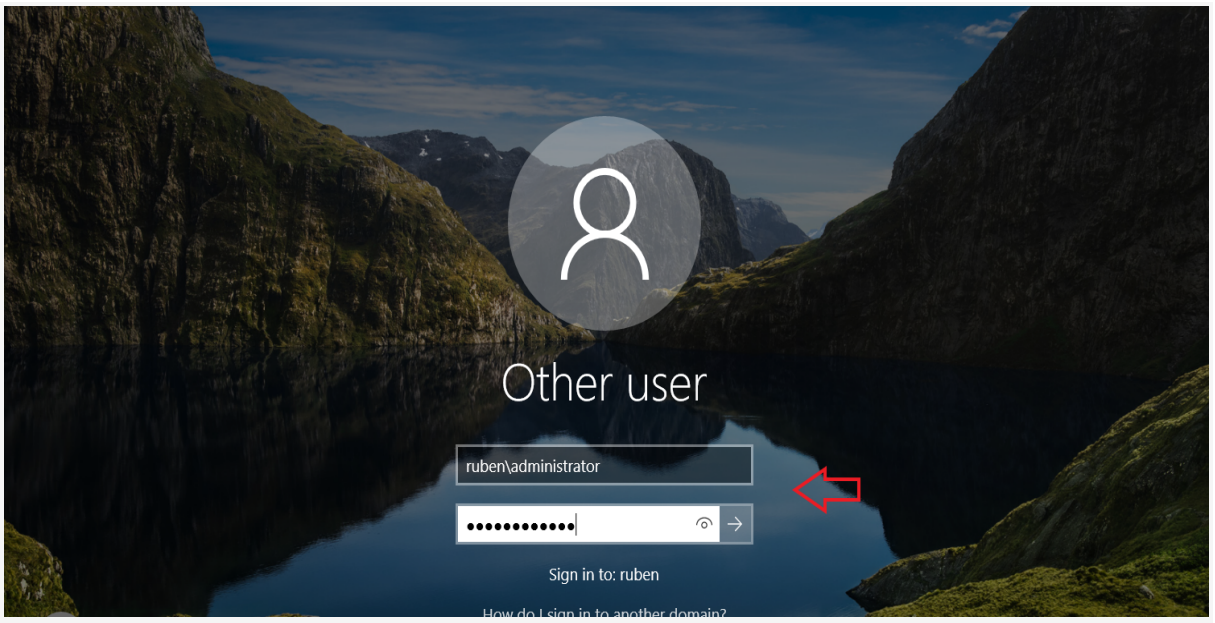
This is how we use and implement the Kerberos protocol for securing a system and the client-server interactions.

1) Démonstration :

First of all I connected all my computers to vm19 for received data flow in the application wireshake



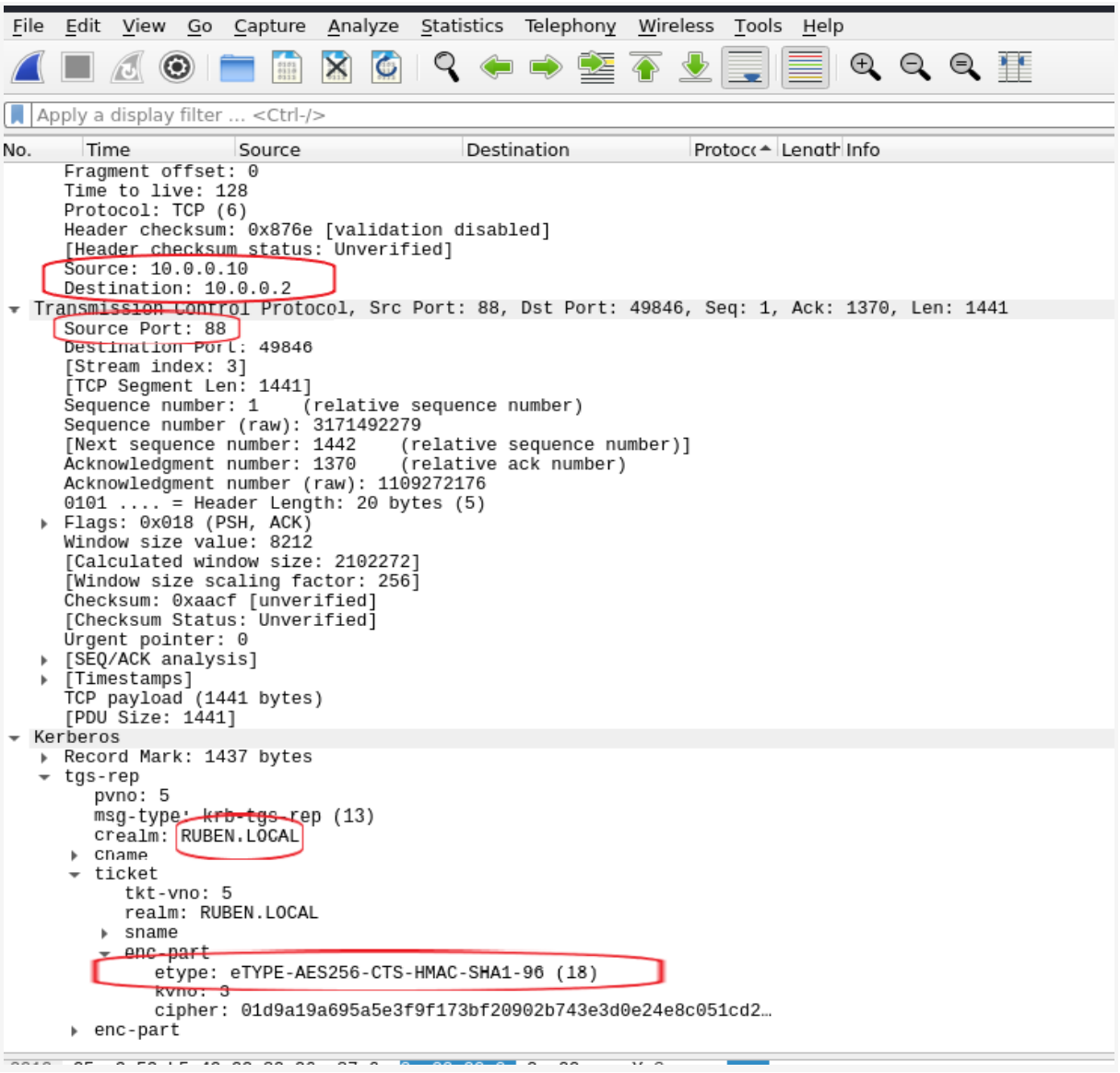
After to see data flow (the protocol kerberos) i need to access as an admin or user and request serveur dc so that there are exchanges



When trying to log into my computer (windows 10) it sends a verification request to the main server

No.	Time	Source	Destination	Protocc	Length	Info
96	16.208831354	10.0.0.10	10.0.0.2	KRB5	1681	TGS-REP
104	16.211593079	10.0.0.2	10.0.0.10	KRB5	1479	TGS-REQ
105	16.212347921	10.0.0.10	10.0.0.2	KRB5	1508	TGS-REP
205	36.047993869	10.0.0.2	10.0.0.10	KRB5	277	AS-REQ
206	36.049299525	10.0.0.10	10.0.0.2	KRB5	239	KRB Error: KRB5KDC_ERR_PREAUTH_REQUIRED
213	36.074099609	10.0.0.2	10.0.0.10	KRB5	357	AS-REQ
214	36.076536944	10.0.0.10	10.0.0.2	KRB5	1621	AS-REP
222	36.078318767	10.0.0.2	10.0.0.10	KRB5	1529	TGS-REQ
224	36.083315009	10.0.0.10	10.0.0.2	KRB5	1601	TGS-REP
244	36.402493889	10.0.0.2	10.0.0.10	KRB5	1686	TGS-REQ
246	36.404863813	10.0.0.10	10.0.0.2	KRB5	1691	TGS-REP

In my computer (kali linux) I run a network scan and I looked for the protocol that interests me (KRB5)



- Source : 10.0.0.10 where the request comes
- Destination :10.0.0.2 where does the request go

So i know 10.0.0.10 is my server Because the requests are made to the server and 10.0.0.2 is my user or admin who tries to connect

- I see the port is 88
- Ruben.local is my serveur

As it is mentioned at the start kerberos is protocol also includes an encryption system in my scan we see that the encryption is

AES256 and hash SHA1