What happens with Kerberos Authentication?

Key Components:   
**KDC** (Key Distribution Center) which includes AS (Authentication Server) and TGS (Ticket Granting Server)

**Principles** (users, applications, services)

**Kerberos** **Software** (integrated into most Operating Systems. MS Windows 2000 and up support Kerberos)

**Main** **Goal**: User needs to authenticate himself/herself without sending passwords across the network—needs to prove he/she knows the password without actually sending it across the wire.

If a user wants to access a service, for instance, the print service, on a network printer

1. A user (principle) logs into a local system with username and password.

2. Username is sent to the Authenticating Server.

3. Authenticating Server looks up the username and uses that user’s secret key (password) to encrypt a ticket called the TGT (Ticket Granting Ticket).

4. If user had entered the correct password, the password is used to decrypt the ticket.

IMPORTANT POINT: User’s password never traverses the network. User has proved he knows the password, because he can decrypt the TGT.

5. When user wants to access another principle, the Kerberos client software will send the TGT over to the TGS (Ticket Granting Server). The TGT proves that the user has already been authenticated. The TGS will generate a new ticket that will include two instances of the same key. This is a “session key” which is only good for this one session. The first instance of the session key is encrypted with the user’s secret key. The second instance is encrypted with the secret key of the service the user wishes to access (the print service).

6. The TGS sends the ticket to the user. The user decrypts the session key with his secret key. The user now knows the short term session key. The user then forwards the TGT to the service.   
7. The service decrypts it’s instance of the session key, using it’s secret key.

Now both the user and the service have securely exchanged a key that will be used to encrypt their communications. They have also authenticated to each other, by the fact that they were able to decrypt the session key correctly. Remember each had to use their own secret key (which only they have) to decrypt/use the session key.

\*\*\*Kerberos is extremely time sensitive as well. It uses time stamping to prevent the possibility of replay attacks. All servers must be synchronized within 5 minutes of each other.