Client Color Clie	NR	Message Name		Ticketname/KeyName verschlüsselt mit Schlüssel ist beka						t Erklärungen
As REQ (1) Cere + 20C Severe Request 2 As RP Cere Request 2 As RP Cere Cere Request				·						
Particular Region Part							AS	TG	Dier	
Authoritication Service Reph.	1	AS_REQ (1) Authenticati Service Request	: → KDC							Dieser Request enthält den Principal-Namen des Clients und den Prinzipal-Namen des TGS
Authoritication Service Rips	2	AS REP (2)	AS_REP	TGS_S _{A,KDC} (Session Key)	Ka					Mit dem TGS-Session-Key kann der Client seine
Clear C NO. AS_REP TGT-TGS. Space tempTime FTGT-TGS = Prompal-Name des Cleans TGT-TGS =	ı	Authentication	Client-Teil	The state of the s	**	x	×			ldentität dem KDC beweisen, weil nur der
Clem 6: 50C AS_REP TGT-TGEI AS_REP TGT-TGEI				+TGS Service Name						Client und der KDC diesen Schlüssel kennen
AS_REP TOT- TGS_Syste + exp Time Principal-Name dec Clerre Principal-Name dec Service Principal-Name d										
AS_REP_TIGT=TGL_S_Name * explaine		Client ←	- KDC							
Dec Clear Assemble Service Person des TOT richt entereditization, des Clears	ı	-								und kann somit den TGS_Session-Key aus der Meldung
TEST-Test +Principal-Hane des Clients stangestechtizzed des	1		AS_REP	TGT= TGS Saunc + expTime	Keno		х			
Service Ser	Ī		TGT-Teil		=Langzeitschlüssel					Der Client kann das TGT nicht entschlüsseln,
dots TGS-REC (3) Description from special for Client in Server Requered Discover Requered Discove	•									weil er K _{KDC} nicht kennt. Somit kann der Client
Service Tips Service Servic	1									das TGT nicht manipulieren und vorallem nicht
1 1 1 1 1 1 1 1 1 1										
Ticket Granting Server Require Discor RED Jonnat Discor RED Jonnat										
Ticket Granting Server Requered Client → DC										11 GS_Se, Kbc, expiration time, Client (Vame,). Kkbc.
Ticket Granting Server Request Client → DC										Den TGS-Session Keu (TGS Sa voc) und das TGT in seiner
Description										verschlüsselten Form speichert der Client in seinem Credential-
Ticket Granting Server Requer Elementer	_									Cache (klist) ab !
Service FRequency Elementen	3	TGS_REQ (3)		AUTHENTICATOR		X				
Dieser REG kommt										
Discre REQ homms Discre REQ homms Client value Client valu		Server Request	Llementen		Sentitic (MS_NEF)					
Disest RED konent Client-Prinzipal-Name Timestamp Checksumme Service FEST konent SERVICE NAME (Dienst in Net Langrestschlüssel des Service Fest) SERVICE NAME (Dienst in Net Langrestschlüssel des Service Fest) SERVICE NAME (Dienst in Net Langrestschlüssel des Service Fest) SERVICE NAME (Dienst in Net Langrestschlüssel des Service Fest) Service Fest		Clien	t → KDC	enthält:						
Timestamp Checksumme Service S		Dieser REQ kommt								
Settlede, weak der Client widen SERVICE NAME (Dienst im Net Langseitschilüssel SERVICE NAME (Dienst im Net SERVICE NAME (Dien										
Clies we deno Ruthorisisters Dieset SERVICE NAME (Dienst im Not Service (Service Service) Client—I TGS_RP (4) Ticket Granting Server Repty Client—I Client										
Service NAME (Dienst in Net Description		zustande, wenn der								
Client + KDC Client + KDC Service-Session Keg (Service-Kap) Client + KDC Service-Session Keg (Service-Session-Keg)		Client auf einen		TICKET GRANTING TICKET	K _{KDC}			х		
Service Client - Server/Service Client - Server/Service Service - Session Key (Service - Session - Key (Service - Session		kerberisierten Dienst		SERVICE NAME (Dienst im Net	_					
beinkings or in Tricket ven Trick ven Tricket ven Tricket Ven Tricket Granting 1 ToS_REP (4) 1 ToS_REP (4) 1 ToS_REP (5) AP_REQ (5) AP_REQ (5) AP_REVECTICKET Service—Session-Key (Service—Ka_B) Client + Service—Session-Key (Service—Session-Key (Service—S		zugreifen will. Dafür		EXPIRATION TIME des TGT	des					
Ticket Granting Server Repls Client—Teil CLIENT TICKET TIGK_S_R_R_DD										
Ticket Granting Server Repls Principal-Name des Service Service—Session-Key (Service—Key K.) Expiration Time Das TGS-REP hat 2 Anteile Principal-Name des Service Service—Tick Service—Tick Service—Tick Glient ← KDC Client ← KDC Client ← KDC Client ← KDC Application Server Client → Server/Service SERVICE TICKET Service—Session-Key (Service—Ka, B.) Client → Server/Service Client → Server/Service SERVICE TICKET Service—Session-Key Service—Ka, B. Client → Server/Service SERVICE TICKET Service—Session-Key Service—Session-Ke	4		Client-Teil	CLIENT TICKET	TOS S	х	Н	x		erstellt dann einen neuen Session Keu (=Service-Session-Keu)
Principal-Name des Service Service Session-Key Service-Session-Key Service-Sess										für Client und Service. Der TGS entnimmt der KDC-Datenbank
Service—Session-Key (Service_K_AB) Expiration Time Das TGS-REP hat 2 Anteile Service—Tic Service—Tic Service—Tic Service—Session-Key (Client ← KDC Client ← KDC Client ← KDC Client ← KDC AP_REQ (5) Application Server Client → Server/Service Service—Session-Key Client + Service—K_AB Client ← KDC Client → Service—Session-Key Service—K_AB Client → Service—Session-Key Service—K_AB Client → Service—Session-Key Servi										
Service - Session - Key (Service_Kab) Expiration Time										
Expiration Time Das TGS-REP hat 2 Antelle				•						welcher bei jeder Neuanmeldung immer wieder generiert wird.
Das TGS-REP hat 2 Artefule Service—Tick SERVICE TICKET KB Service-Session-Key (Service_Ka, a) Client ← KDC Service-Session-Key (Service_Ka, a) Client Name Client N										
Das TGS-REP hat 2 Anteile Service—Tic Service—Session-Key (Service_Kas) Client Name Expiration Time AP_REQ (5) Application Server Client → Server/Service Service—Session Key Service_Kas Client Name Expiration Time AUTHENTICATOR Service—Session Key Service_Kas Client Name Expiration Time AP_REQ (5) Application Server Client → Server/Service SERVICE TICKET Service Session Key Service_Kas Client Name Expiration Time AP_REP (6) AP_REP (6) AP_REP (6) Application Server Client ← Server/Service OPTIONAL! Service—Sesver/Service Application Server Client ← Server/Service OPTIONAL! Er muss also (1) und (2) nicht mehr durchführen. X X Das in Schritt (2) eingegebene Paswort genügt. SSO wird mi diesem Trick möglich! X X Authenticator zu entschlüsseln, deshabl (½). Den Service-Session Key entnehmen. Mit diesem Key let vien der Lage den Authenticator zu entschlüsseln, deshabl (½). Den Service-Session Key entnehmen. Mit diesem Key let vien der Lage den Authenticator zu entschlüsseln, deshabl (½). Den Service-Session Key entnehmen. Mit diesem Key let vien der Lage den Authenticator zu entschlüsseln, deshabl (½). Den Service-Session Key entnehmen. Mit diesem Key let vien der Lage den Authenticator zu entschlüsseln kann, hat der Client und KDC. Dieser Key let also ein Authenticator zu entschlüsseln kann, hat der Client seine Authenticator entschlüsseln kann, hat der Client, hätte diesen Authenticator generieren können! X X X X X X X X X X X X X X X X X X				Expiration Time						
Service-TickET Client ← KDC Service-Session-Key (Service_Kap) Client Name Expiration Time		Das TGS-REP hat 2								
Client ← KDC Service-Session-Key (Service_Kap) Client Name Expiration Time AP_REQ (5) Application Server Client → Server/Service SERVICE TICKET Service Session Key Service_Kap Client Name Expiration Time Service Session Key Service Name Client → Server/Service SERVICE TICKET Service Session Key Service Name Expiration Time AP_REP (6) Application Server Client ← Server/Service OPTIONAL! Service-Session-Key (Service_Kap) Client ← Server/Service OPTIONAL! AUTHENTICATOR Service-Session-Key Service Name Expiration Time Service Session-Key enthemen. Mit diesem Key ist er in der Lage den Authenticator zu entschlüsseln, deshalb (X). Den Service-Session-Key enthemen. Mit diesem Key ist er in der Lage den Authenticator zu entschlüsseln, deshalb (X). Den Service-Session-Key enthemen. Mit diesem Key den Authenticator in der Lage den Authenticator zu entschlüsseln kann, hat der Client und KDC. Dieser Key ist also ein gemeins amers Geheimnis zwischen Dienst und Client. Venn der Service nun mit diesem Key den Authenticator entschlüsseln kann, hat der Client, hätte diesen Authenticator generieren können! **X** **X** **X** **X** ** **X** **X** **Ditional: Falls der Client, hätte diesen Authenticator generieren können! **X** **X** **X** **X** **X** **X** **X** **Ditional: Falls der Client sicher sein will, dass es der richtige Server ist. Z.B. Telebanking-Server		Anteile								
Client ← KDC Service-Session-Key (Service_Kap) Clicant Name Expiration Time AP_REQ (5) Application Server Client → Server/Service SERVICE TICKET Service Session Key Service_Kap Client Name Expiration Time Service Session Key Service_Kap Client Name Expiration Time 6 AP_REP (6) Application Server Client ← Server/Service OPTIONAL! OPTIONAL! Client ← Server/Service Client ← Server/Service OPTIONAL! AUTHENTICATOR Service-Session-Key (Service_Kap) Client Name Expiration Time Service Session Key Service_Kap Client Name Expiration Time Service Session Key Service_Kap Client Name Expiration Time Service Session Key Service Name Expiration Time OPTIONAL! OPTIONAL! Client ← Server/Service OPTIONAL! Aus dem Service Ticket kann der Server den Service Session-Key enthempt. Mit diesem Key ist et also ein Authenticator Service Authenticator of generic num mit diesem Key den Authenticator entschlüsseln kann, hat der Client, hätte diesen Authenticator genericen können! X X Optional: Falls der Client, hätte diesen Authenticator genericen können! X X Optional: Falls der Client sicher sein will, dass es der richtige Server ist. Z.B. Telebanking-Server > wenn der Client den Zeit- stempel mit Service_Kap entschlüsseln kann, weiss er, dass dieser vom "richtigen" Server gesendet wurde,			Service-Tic	SERVICE TICKET	K _B			х	х	Das in Schritt (2) eingegebene Passwort genügt. SSO wird mit
AP_REQ (5) Application Server Client → Server/Service SERVICE TICKET Service Session Key Service — Key Expiration Time AP_REQ (5) AP_REQ (5) APPLICATION Checksumme Checksumme SERVICE TICKET Service Session Key Service — Key Expiration Time Service Session Key Service — Key Expiration Time AP_REQ (6) AP_REQ (6) AP_REQ (6) AP_REQ (6) APPLICATIONAL! Client → Server/Service Client ← Server/Service AV X X Client Name Expiration Time Service — Ka,B X Client ← Server/Service Authenticator Xey is the Lage of Authenticator of the Service — Servi		Client 4	- VDC							diesem Trick möglich!
AP_REQ (5) APplication Server Client → Service Session Key Service Ka, B Client Name Eupiration Time Eupiration Time AP_REP (6) APPLICATIONAL! Client Service Service C		Client 4	NDC							
AP_REQ (5) Application Server Client → Server/Service Service Session Key Service Kap Client Name Expiration Time APPREP (6) APPREP (6) Application Server Client ← Server/Service OPTIONAL! OPTIONAL! AUTHENTICATOR Service Session- Application Service SERVICE TICKET KB Service Session Key Service KB Service Session Key Service Client Name Expiration Time Service Session Key Service Client Name Expiration Time Service Session Key Service Client Name Expiration Time Service Session Key Service Client Server/Service OPTIONAL! Service Session Key Service Client Server/Service OPTIONAL! Application Server Client Server/Service OPTIONAL! Service Session Key Service Client Server Service Application Server Client Server/Service OPTIONAL! Service Session Key Service Client Server Service Application Server Client Server/Service OPTIONAL! Service Server Service OPTIONAL! Service Server Service OPTIONAL! Service Server Service OPTIONAL! Service Server Service Service Service Service Service Service Service Authenticator zu entschlüsseln key den Authenticator entschlüsseln kennen nur Dienst, Client und KDC. Dieser Key ist also ein gemeinsames Geheimnis zwischen Dienst und Client. Wenn der Client diesen Key den Authenticator entschlüsseln kennen nur Dienst, Client und KDC. Dieser Key ist also ein gemeinsames Geheimnis zwischen Dienst und Client. Wenn der Client sicher Service in vill diesen Key den Authenticator entschlüsseln kennen nur Dienst, Client und KDC. Dieser Key ist also ein gemeinsames Geheimnis zwischen Dienst und Client. Wenn der Client den Celient dienst entschlüsseln kann, weiss en, dass dieser vom "richtigen" Server gesendet wurde,										
Application Server Checksumme Checksumme Checksumme Checksumme SERVICE TICKET Service Session Key Service_Ka,B Client Name Expiration Time Application Server Application Server Service Session Key Service_Ka,B Client Name Expiration Time Application Server Client ← Server/Service OPTIONAL! OPTIONAL! Application Server Client Server Service OPTIONAL! Application Server Client Server Service OPTIONAL! Checksumme Authenticator and entschibisseln, deshalb (X). Den Service-Session Key service Session Key Expiration Time X X X Authenticator zu entschibisseln, deshalb (X). Den Service-Session Key not call the service service in multidiesem Key den Authenticator entschibisseln kann, hat der Client seine Authenticator generieren können! X X X X Optional: Falls der Client sicher sein will, dass es der richtige Server ist. Z.B. Telebanking-Server		AD REO (E)		· · · · · · · · · · · · · · · · · · ·	Service K	х		x	(X)	Aus dem Service Ticket kann der Server den Service-Session-
Client → Server/Service SERVICE TICKET Service Session Key Service_K _{B,B} Client Name Expiration Time Authenticator zu entschlüsseln, deshalb (X). Den Service-Sessio kennen nur Dienst, Client und KDC. Dieser Key ist also ein gemeinsames Geheimnis zwischen Dienst und Client. Wenn der Service nun mit diesem Key den Authenticator entschlüsseln kann, hat der Client seine Authentizität bewiesen. Kein anderer als der Client, hätte diesen Authenticator generieren können! Application Serve Client ← Server/Service OPTIONAL! OPTIONAL! Authenticator zu entschlüsseln, deshalb (X). Den Service-Sessio kennen nur Dienst, Client und KDC. Dieser Key ist also ein gemeinsames Geheimnis zwischen Dienst und Client. X X gemeinsames Geheimnis zwischen Dienst und Client. Wenn der Service nun mit diesem Key den Authenticator entschlüsseln kann, hat der Client seine Authenticator generieren können! X X Optional: Falls der Client sicher sein will, dass es der richtige Server ist. Z.B. Telebanking-Server > wenn der Client den Zeit- stempel mit Service_K _{B,B} entschlüsseln kann, weiss er, dass dieser vom "richtigen" Server gesendet wurde,										
Client → Server/Service SERVICE TICKET Service Session Key Service_Ka,p Client Name Expiration Time AP_REP (6) Application Service Client ← Server/Service OPTIONAL! Client Service Authenticator Application Service Client ← Server/Service Client ← Server/Service Application Service Client ← Server/Service Client ← S				· ·						y
SERVICE TICKET Service Session Key Service_Ka,p Client Name Expiration Time 6 AP_REP (6) Application Service Client ← Server/Service OPTIONAL! DESCRIPTIONAL! Service Session Key Service_Ka,p Service Session Key Service_Ka,p X Service Manual Authenticator entschilüsseln kann, hat der Client seine Authenticator entschilüsseln kann, hat der Client seine Authenticator generieren können! X Wenn der Service nun mit diesem Key den Authenticator entschilüsseln kann, hat der Client, hätte diesen Authenticator generieren können! X Wenn der Service nun mit diesem Key den Authenticator entschilüsseln kann, hat der Client, hätte diesen Authenticator generieren können! X Wenn der Service nun mit diesem Key den Authenticator entschilüsseln kann, hat der Client, hätte diesen Authenticator generieren können! X Wenn der Service nun mit diesem Key den Authenticator entschilüsseln kann, hat der Client seine Authenticat				Checksumme						Authenticator zu entschlüsseln, deshalb (X). Den Service-Sessio
Service Session Key Service_K _{a,b} Client Name Expiration Time 6 AP_REP (6) Application Service Client ← Server/Service □PTIONAL! Client Session Key Service_K _{a,b} Wenn der Service nun mit diesem Key den Authenticator entschlüsseln kann, hat der Client seine Authenticator entschlüsseln kann, hat der Client seine Authenticator generieren können! X X Optional: Falls der Client sicher sein will, dass es der richtige Server ist. Z.B. Telebanking-Server -> wenn der Client den Zeit- stempel mit Service_K _{a,b} entschlüsseln kann, weiss er, dass dieser vom "richtigen" Server gesendet wurde,		Client → Ser	ver/Service							
Client Name Expiration Time 6 AP_REP (6) Application Server Client ← Server/Service OPTIONAL! TIMESTAMP Application Server Client ← Server/Service OPTIONAL! Application Server Client ← Server/Service Timestampel mit Service ← Application Server Tichtige Server ist. Z.B. Telebanking-Server Tichtige Server ist. Z.B. Telebanking-Server Tichtige Server ist. Age entschlüsseln Kann, weiss er, dass dieser vom Trichtigen" Server gesendet wurde,				SERVICE TICKET	K _B			Х	X	gemeinsames Geheimnis zwischen Dienst und Client.
Client Name Expiration Time 6 AP_REP (6) Application Server Client ← Server/Service OPTIONAL! OPTIONAL! Client Name Expiration Time Dewiesen. Kein anderer als der Client, hätte diesen Authenticator generieren können! X X Optional: Falls der Client sicher sein will, dass es der richtige Server ist. Z.B. Telebanking-Server -> wenn der Client den Zeit- stempel mit Service_Ka,e entschlüsseln kann, weiss er, dass dieser vom "richtigen" Server gesendet wurde,				Service Session Key Service_K _{e,e}						
Expiration Time AP_REP (6) Application Service Client ← Server/Service OPTIONAL! Coptional: Falls der Client, hätte diesen Authenticator generieren können! Kein anderer als der Client, hätte diesen Authenticator generieren können! X Optional: Falls der Client sicher sein will, dass es der richtige Server ist. Z.B. Telebanking-Server > wenn der Client den Zeit- stempel mit Service_Ka,a entschlüsseln kann, weiss er, dass dieser vom "richtigen" Server gesendet wurde,				Client Name						
generieren können! 6 AP_REP (6) TIMESTAMP Service_K _{A,B} X Application Servic Client ← Server/Service OPTIONAL! V APPLICATION Server Service Serv										
6 AP_REP (6) TIMESTAMP Service_K _{A,B} X Doptional: Falls der Client sicher sein will, dass es der richtige Server ist. Z.B. Telebanking-Server Application Serve Client ← Server/Service OPTIONAL! OPTIONAL! X Optional: Falls der Client sicher sein will, dass es der richtige Server ist. Z.B. Telebanking-Server > wenn der Client den Zeit- stempel mit Service_K _{A,B} entschlüsseln kann, weiss er, dass dieser vom "richtigen" Server gesendet wurde,										
Application Serve Client Server/Service OPTIONAL! irichtige Server ist. Z.B. Telebanking-Server > wenn der Client den Zeit- stempel mit Service_Ka,e entschlüsseln kann, weiss er, dass dieser vom "richtigen" Server gesendet wurde,										
Application Serve Client Server/Service OPTIONAL! irichtige Server ist. Z.B. Telebanking-Server > wenn der Client den Zeit- stempel mit Service_Ka,e entschlüsseln kann, weiss er, dass dieser vom "richtigen" Server gesendet wurde,	8	AD DED (C)		TIMESTAMP	Saudaa V	x		x	X	Ontional: Falls der Client sicher sein will dass es der
Serve Client ← Server/Service □PTIONAL! > wenn der Client den Zeit- stempel mit Service_K _{A,b} entschlüsseln kann, weiss er, dass dieser vom "richtigen" Server gesendet wurde,	0			TIMESTAM	Dervice_K _{A,B}	^		^	 ^	<u>'</u>
OPTIONAL! OPTIONAL! in the server service when der Client den Zeitstempel mit Service_Ka,e entschlüsseln kann, weiss er, dass dieser vom "richtigen" Server gesendet wurde,		Corne	/c ·							norrage perver ist. 2.0. Telepanking-perver
kann, weiss er, dass dieser vorn "richtigen" Server gesendet wurde,		Client ← Ser	ver/Service							> wenn der Client den Zeit-
"richtigen" Server gesendet wurde,		OPTIONAL!								- ·
				1						kann weige er dage dieger vom
weil nur dieser den Schlitissel Seruine K mittels K. kennt										
weimul deser den owniesser derivide_ N _A & Hilliters N _A Aerillit.										"richtigen" Server gesendet wurde,

Directory Service = Verzeichnisdienst für Netzwerkobjekte (User, Drucker, Mail)

Authentisierung = Nachweis der eigenen Identität (Hauptaufgabe von Kerberos)

Autorisierung = wird festgelegt, mit welchen Berechtigungen Benutzer auf Ressourcen im Netzwerk zugreifen dürfen SSO = Zentrale Anmeldung, netzwerkweit möglich, bis Ablauf von ticket

Daten Speicherung:

- Langzeitschüssel = Keytab datei

- TGT und TGS = Credential cache (bei Client)

Abkürzungen:

- DIT = Directory Information Tree
- DN = Distinguished Name
- OU = Organizational Unit
- CN = Common Name
- KDC = Key Distribution Center, wird auch noch Trusted Third Party genannt, Ein Kreditkartenherausgeber ist die Trusted Third Party zwischen Käufer und Verkäufer.
- Der KDC kennt alle Schlüssel
- DC = Domain Component
- TGS = Ticket Granting Service / Server, im TGS Session Key wird die Authentizität des Clients gespeichert.
- TGT = Ticket Granting Ticket, das TGT besteht aus dem TGS Session Key + expTime + Principal Name des Client
- RDN = Relative Distinguished Names
- SessionKey = temporär
- ServiceKey = bleibt gleich

Befehle:

- Kdestroy = Löschen von Tickets
- Kinit = TGT abrufen und im cache speichern / cache erzeugen
- Klist = aktuelle tickets anzeigen, Credential cache anzeigen
- Kpasswd = kerberos passwort ändern
- Ktutil = Schlüsseltabellendateien verwalten
- Kadmin = Datenbank verwalten

Ldap:

Ldap steht für Lightweight Directory Access Protocol. Ldap verwendet Port 389 (Standardport) und 636 für LDAPS (verschlüsseltes LDAP). Ldap wurde 1993 an der Universität vom Michigan entwickelt.

Principal:

Ein Principal ist eine eindeutige Identität, der ein Kerberos Ticket zugewiesen werden kann. Principals können eine beliebige Anzahl von Komponenten haben. Jede Komponente wird durch ein Trennzeichen getrennt, meistens «/». Die letzte Komponente ist der Realm, der vom rest des Principals durch das Realm-Trennzeichen, in der Regel @ getrennt wird.

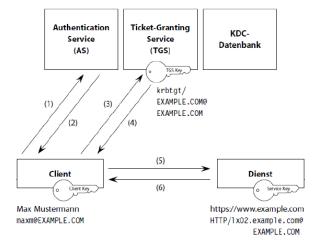
Bsp: jennifer@ATHENA.MIT.EDU -> Endung in Grossbuchstaben

Computer und User Principal

Man in the Middle Angriff

Kerberos verhindert Man in the Middle Attacken, in dem es die 3 Parteien Authentifizierung verwendet und somit immer weiss, wer den Service verwendet.

LDIF: Bei einer LDIF-Datei handelt es sich um ein standardmäßiges Klartext-Datenaustauschformat zur Darstellung von LDAP-Verzeichnisinhalten und Aktualisierungsanforderungen (LDAP = Lightweight Directory Access Protocol). LDIF übermittelt Verzeichnisinhalte als Datensatzgruppe mit einem Datensatz für jedes Objekt (oder jeden Eintrag). Auch Aktualisierungsanforderungen wie z. B. Hinzufügen, Ändern, Löschen und Umbenennen werden als Datensatzgruppe mit einem Datensatz für jede Aktualisierungsanforderung dargestellt



Die gesamte Kommunikation mit dem KDC erledigt der Client. Eine Kommunikation zwischen dem Dienst und dem KDC findet nicht statt. Das KDC muss für den Dienst also während der Client-Authentifizierung nicht erreichbar sein. Es gibt zwei Varianten, wie ein Client Tickets beziehen kann. Eine davon ist die Verwendung des Authentication Service (AS), bei der anderen bezieht der Client die Tickets vom Ticket-Granting Service (TGS). Beide Dienste sind sich relativ ähnlich, der TGS ist aber ein kerberisierter Dienst. Der Client benötigt seinen Langzeitschlüssel (also in der Regel das User-Passwort), um den Inhalt der Antworten des AS zu entschlüsseln.