Kerberos

1 Allgemeines

Kerberos ist ein Authentifizierungsprotokoll, das auf dem Needham-Schröder-Protokoll mit symmetrischem Verfahren aufbaut. Es ist z.B. (in leicht abgeänderter Form) in Microsoft Windows implementiert, und wird zur authentifizierten Nutzung von Serverdiensten in offenen Netzwerken verwendet. Kerberos ist ein sicherer (d.h. es werden nie Passwörter im Klartext über das Netzwerk übertragen) und beiderseitiger (sensitive Informationen werden geschützt, indem sich beide Parteien authentifizieren müssen) Authentifizierungsservice, bei dem nur eine Anmeldung (zur Nutzung aller Netzwerkressourcen) benötigt wird (Stichwort: Single-Sign-On) und der über einen vertrauenswürdigen Dritten (in Kerberos das Key Distribution Center, dem alle Beteiligten von sich aus vertrauen) abgewickelt wird. Die Ziele von Kerberos sind, dass in offenen Netzwerken eine größtmögliche Sicherheit und ein Schutz vor Angriffen (z.B. Man-in-the-Middle-Attacks) erreicht wird, und gleichzeitig nur eine Anmeldung benötigt wird, um alle Netzwerkressourcen zu nutzen. Zu beachten ist, dass Kerberos nur ein Authentifizierungsservice ist, und nicht autorisiert, d.h. es wird nur sichergestellt, dass man der ist, der man vorgibt, zu sein, und nicht, ob derjenige die Netzwerkressource auch nutzen darf. Dies muss der jeweilige Server nach erfolgreicher Authentifizierung z.B. anhand von Benutzerlisten selbst entscheiden.
Von Kerberos sind die Versionen 4 und 5 im Umlauf. Das Protokoll von Ersterer soll nun anhand eines Beispiels erläutert werden.

2 Protokollablauf von Kerberos 4

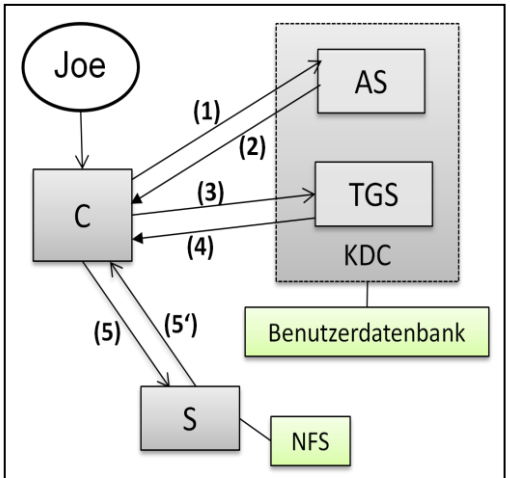

Abbildung 1: Beteiligte

An dem Prozess beteiligt ist der Clientcomputer C, an dem sich der fiktive Benutzer Joe einloggt. Dieser will in unserem Beispiel den NFS-Dienst auf dem Server S verwenden. In Kerberos benötigt Joe ein sog. Ticket, welches als „Eintrittskarte“ fungiert, um sich gegenüber dem Server authentifizieren zu können. Dieses bekommt er vom Ticket Granting Server (TGS) gegen Vorlage des Ticket Granting Tickets (TGT). Das TGT bekommt Joe vom Authentification Server (AS), sobald er sich am Clientcomputer einloggt. Das TGT ist i.d.R. etwa zehn Stunden gültig. D.h., wenn sich Joe (am Morgen) auf seinem Clientcomputer einloggt, bekommt er vom AS das TGT (Schritte (1) und (2)), mit dem er während seines ganzen Arbeitstages vom TGS Tickets bekommt (Schritte (3) und (4)), mit denen wiederum er sich gegenüber den einzelnen Servern authentifizieren kann (Schritte (5) und (5‘)). Normalerweise sind der AS und der TGS als Dienste auf einem Server installiert, der als Key Distribution Center (KDC) bezeichnet wird. Dieser verfügt außerdem über eine Benutzerdatenbank, in der alle Benutzer und Server, sowie deren jeweiligen Master-Schlüssel ($K_{Benutzer}$) gespeichert sind. Dieser Benutzerschlüssel lässt sich aus dem Benutzerpasswort berechnen. Daneben gibt es Sitzungsschlüssel zur Kommunikation zwischen zwei Parteien (z.B. $K_{Benutzer,Server}$), den auch nur diese beiden Parteien kennen. Im Folgenden sollen nun die Prozessschritte von Kerberos 4, die in Tabelle 1 dargestellt sind, genauer erläutert werden.

Nr	Von→An	Bez.*	Inhalt
(1)	C → AS	AS_REQ	Joe, TGS, $Nonce_1$
(2)	AS → C	AS_REP	$\{K_{Joe,TGS}, Nonce_1\}^{K_{Joe}}$, $\{T_{Joe,TGS}\}^{K_{TGS}}$ $\{T_{Joe,TGS}\}^{K_{TGS}} = \{TGS, Joe, IP\text{-}Adresse_{Joe}, Zeitstempel, Laufzeit, K_{Joe,TGS}\}^{K_{TGS}}$
(3)	C → TGS	TGS_REQ	$\{A_{Joe}\}^{K_{Joe,TGS}}$, $\{T_{Joe,TGS}\}^{K_{TGS}}$, NFS, $Nonce_2$ $\{A_{Joe}\}^{K_{Joe,TGS}} = \{Joe, IP\text{-}Adresse_{Joe}, Zeitstempel\}^{K_{Joe,TGS}}$
(4)	TGS → C	TGS_REP	$\{K_{Joe,NFS}, Nonce_2\}^{K_{Joe,TGS}}$, $\{T_{Joe,NFS}\}^{K_{NFS}}$
(5)	C → S	NFS_REQ	$\{A_{Joe}\}^{K_{Joe,NFS}}$, $\{T_{Joe,NFS}\}^{K_{NFS}}$
(5')	S → C	NFS_REP	$\{A_{Joe}^{*}\}^{K_{Joe,NFS}}$

Tabelle 1: Prozessschritte *Bez. bedeutet die Bezeichnung der Nachricht

Sobald sich Joe am Clientcomputer einloggt, schickt dieser in Schritt (1) eine Nachricht mit der Identität des Clients (Joe), der Identität des TGS und einer Nonce (dazu später mehr) im Klartext an den AS. Ziel dabei ist es, von diesem ein Ticket Granting Ticket (TGT) zu erhalten.

Der AS verifiziert nun, dass der Client existiert und generiert den Sitzungsschlüssel $K_{Joe,TGS}$, der später zur Kommunikation zwischen Client und TGS verwendet wird. Dieser wird nun in Schritt (2) zusammen mit der Nonce mit dem Master-Key von Joe verschlüsselt an den Client zurückgeschickt. Der Client kann also den Sitzungsschlüssel nur entschlüsseln, wenn Joe sein Passwort korrekt eingibt. Nonce bedeutet „Number generated once" und ist eine Zufallszahl, die nur einmal verwendet wird. Sie dient zur Erkennung von Replays: Wäre sie nicht vorhanden, könnte ein Angreifer die AS_REP abfangen, und bei einer erneuten AS_REQ von C wiedereinspielen, so dass erneut der gleiche Sitzungsschlüssel verwendet würde, was ein Sicherheitsrisiko darstellen würde. Daneben schickt der AS dem Client das TGT ($\{T_{Joe,TGS}\}^{K_{TGS}}$) mit, welches Identität des TGS und des Clients, die Gültigkeitsdauer des Tickets und den Sitzungsschlüssel $K_{Joe,TGS}$ beinhaltet. Es ist mit dem Master-Key des TGS verschlüsselt. Somit kann es nicht von Joe gelesen werden, was jedoch auch nicht nötig ist.

Sobald nun Joe einen Netzwerkdienst nutzen will (in unserem Beispiel NFS), benötigt er ein entsprechendes Ticket. Dazu schickt er das TGT in Schritt (3) zusammen mit dem Namen des Dienstes, den er nutzen will und einer zweiten Nonce an den TGS. Zusätzlich wird der Authentikator $\{A_{Joe}\}^{K_{Joe,TGS}}$ mitgeschickt, der die Identität von Joe sowie einen aktuellen Zeitstempel, verschlüsselt mit dem Sitzungsschlüssel, enthält. Dadurch weist der Client gegenüber dem TGS nach, dass er den Sitzungsschlüssel kennt. Der Client ist somit authentifiziert, da nur er und das KDC den Sitzungsschlüssel kennen.

Der TGS generiert, nach einer Verifikation des Authentikators, den Sitzungsschlüssel $K_{Joe,NFS}$, welcher später zur Kommunikation zwischen Client und dem Server verwendet wird, und schickt diesen, zusammen mit der Nonce und mit dem Sitzungsschlüssel $K_{Joe,TGS}$ verschlüsselt, zum Client. Somit ist auch der TGS authentifiziert, da nur der TGS den Schlüssel $K_{Joe,TGS}$ aus dem TGT entschlüsseln kann. Eine Replay-Attacke wird durch die Nonce (vgl. oben) verhindert. Zusätzlich wird das Ticket für den NFS-Server ($\{T_{Joe,NFS}\}^{K_{NFS}}$) mitgeschickt. Dieses hat entsprechenden Inhalt wie das TGT in Schritt (2).

Nun kann der Client in Schritt (5) Kontakt mit dem NFS-Server aufnehmen. Dabei schickt er ihm das Ticket aus Schritt (4), sowie einen Authentikator, der mit $K_{Joe,NFS}$ verschlüsselt ist, um sich zu authentifizieren (vgl. Schritt (3)).

In dem optionalem Schritt (5') schickt der Server einen leicht veränderten Authentikator mit einem um 1 erhöhten Zeitstempel zurück. Somit hat auch der Server nachgewiesen, dass er $K_{Joe,NFS}$ kennt, es sind also Server und Client authentifiziert und ein Sitzungsschlüssel ausgetauscht, d.h. die Übertragung kann beginnen.

3 Probleme in Kerberos 4 und deren Lösung in Kerberos 5

Kerberos 4 weist einige Sicherheitslücken auf, die in Kerberos 5 eliminiert wurden. Da ansonsten beide Versionen sehr ähnlich sind, werden hier nur die Änderungen besprochen.
In Kerberos 4 kann ein Angreifer eine Offline-Wörterbuchattacke auf das TGT bzw. den Master-Key von Joe in Schritt (2) ausführen, indem ein Angreifer in Schritt (1) eine AS_REQ mit falscher Identität des Clients verschickt und somit ein TGT eines anderen Benutzers erhält. In Kerberos 5 wird dies verhindert, indem der Client in Schritt (1) einen (leicht abgewandelten) Authentikator mitschickt, und somit die Kenntnis über K_{Joe} beweist.
Ein weiteres Problem ist, dass in Firmen oft mehrere KDCs verwendet werden, die jeweils einen eigenen Zuständigkeitsbereich besitzen, damit es keinen „Single Point of Failure" gibt. In Kerberos 4 kann daher ein Client nicht auf einen Server zugreifen, der im Zuständigkeitsbereich eines anderen KDCs liegt. In Kerberos 5 wird dies hingegen durch eine Hierarchie von KDCs, die untereinander dieses sog. „Forwarding" zulassen, ermöglicht.
Des Weiteren ist in Kerberos 4 DES als Verschlüsselungsmethode vorgeschrieben, das jedoch einige Sicherheitslücken aufweist. In Kerberos 5 hingegen können verschiedene (sicherere) Verschlüsselungsmethoden verwendet werden.
Außerdem gibt es in Kerberos 4 keine Möglichkeit, im Auftrag Dritter tätig zu sein. In der Nachfolgeversion hingegen wird dieses als „Delegationsprinzip" bezeichnete Vorgehen erlaubt.

4 Schwachstellen und Vorteile von Kerberos

Beide Kerberos-Versionen haben einige Schwachstellen. So werden standardmäßig die Tickets im /tmp-Verzeichnis abgelegt, auf das alle Benutzer Zugriff haben. Dies sollte also unbedingt geändert werden, wenn man Kerberos benutzt. Des Weiteren basiert Kerberos auf Passwörtern. Sinnvoll wäre hier z.B. eine Smartcard-Unterstützung. Eine weitere Schwachstelle ist, dass das KDC alle Benutzerpasswörter mit dem gleichen Serverpasswort verschlüsselt. Ist dieses einmal kompromittiert, müssen alle Benutzerpasswörter ausgetauscht werden. Daneben müssen alle Server-Zeiten synchronisiert sein. Gelingt es nämlich, einem Server eine falsche Zeit unterzuschieben, ist ein Maskierungsangriff mit einem bereits verwendeten Authentikator möglich. Auch die Laufzeit der Tickets, die i.d.R. zehn Stunden beträgt, kann eine Gefahr darstellen: Schafft man es, den Sitzungsschlüssel schnell genug zu kompromittieren, kann man während der gesamten Laufzeit des Tickets einen Maskierungsangriff durchführen. In Kerberos 5 kann man sogar die Laufzeit der Tickets bis zum Jahr 9999 einstellen, wodurch o.g. Angriff noch erheblichere Auswirkungen hätte. Daher sollte die Laufzeit der Tickets unbedingt auf maximal einen Tag beschränkt werden.

Trotzdem bietet Kerberos in offenen Netzwerken einen erheblichen Sicherheitsgewinn, da authentifiziert zugegriffen werden kann und keine Passwörter im Klartext übertragen werden müssen. Außerdem bietet es ein Single-Sign-On, was die Akzeptanz bei den Benutzern stärkt, und, statistisch nachgewiesen, zu sichereren Passwörtern führt. Kerberos sollte also in offenen Netzwerken unbedingt verwendet werden.

Quellennachweis:

[Q1] „IT-Sicherheit" von Claudia Eckert, Oldenburg-Verlag, Seiten 498 bis 508
[Q2] „Kerberos – The Definitive Guide" von Jason Garman, Verlag O'Reilly, Seiten 1 bis 36
[Q3] http://de.wikipedia.org/wiki/Kerberos_(Informatik), aufgerufen am 14.04.2009