

Verteilte Systeme

17. Schutz

Motivation

Schutz (Protection)

Zugang zu den Ressourcen eines Rechners tatsächlich kontrollieren

- Wer darf welche Ressourcen wie nutzen?
- Dateien
- Seiten im Adreßraum
- Ein- und Ausgabegeräte
- Semaphore
- ...

Authentisierung

Schutz umsetzen

Form der Beschreibung

Sicherheit (Security)

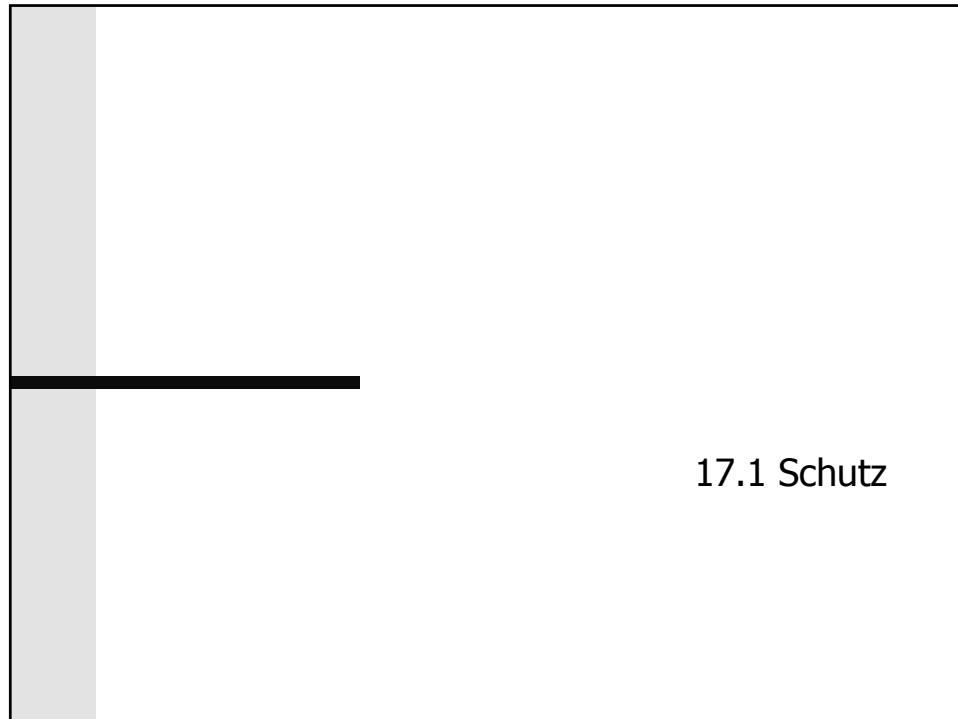
Schutz vor unautorisiertem Zugriff auf Ressourcen

- Systemeigenschaft
- Wie weit kann ein Benutzer dem System vertrauen?

Benutzer authentifizieren

Große Bedeutung durch die Vernetzung von Rechnern

- Eingehende Nachrichten können von jedem kommen



Schutz

Umfang

- Anwendungen stehen bestimmte Ressourcen zur Verfügung
- Jede Ressource besitzt Namen und Typ
- Abhängig vom Typ sind bestimmte Operationen erlaubt

Schutz vor unzulässigem Zugriff

Erhöhte Zuverlässigkeit

- Schutz vor unbewußter falscher Nutzung von Ressourcen
- Aufdecken versteckter Fehler durch Zugriffsverletzung

Need-to-know-Prinzip: Nur die Ressourcen verwenden, die zur Erfüllung der Aufgabe unabdingbar sind

Schutzpolitik (Policy)

- Administration

Schutzmechanismen (Umsetzung)

Verteilte Systeme, Wintersemester 2000/2001 Folie 17.4

Schutzdomänen

Domäne 1

<O1, {Read, Write}>
 <O2, {Execute}>
 <O3, {Print}>

Domäne 2

<O4, {Read}>
 <O5, {Write}>
 <O3, {Execute, Write}>

Grundstruktur

- Zu schützende Objekte
 - Name
 - Typ
 - Definierte Operationen
- Kontrollflüsse (Subjekte), deren Objektzugriffe überwacht werden müssen

Kontrollflüsse arbeiten innerhalb bestimmter Schutzdomäne

Schutzdomäne definiert Zugriffsrechte eines Kontrollflusses auf die Objekte

Schutzdomäne = Menge von Zugriffsrechten <Objekt, Rechte>

Zuordnung Domäne-Kontrollfluß

- Statisch (Inhalte ändern)
- Dynamisch (Domäne wechseln)

Verteilte Systeme, Wintersemester 2000/2001
Folie 17.5

Schutzmechanismen

Supervisor/User-Modus

- Unterscheidung zweier Modi
- Wohldefinierte Übergänge
 - Interrupts
 - Traps
- Problem: Verschiedene Prozesse auch untereinander schützen

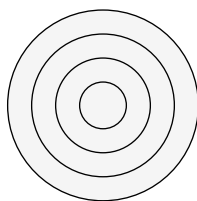
Schutzringe

- Verallgemeinerung
- z.B. Intel-Prozessor

User-Modus

Schutzwall

Supervisor-Modus



Verteilte Systeme, Wintersemester 2000/2001
Folie 17.6

Schutzdomänen sind auch Objekte

Beschreibung dynamischer Aspekte

- Wechsel der Schutzdomäne (Switch)
- Übertragung, Hinzufügen und Änderung von Rechten

	Objekt 1	Objekt 2	Objekt 3	Domäne 1	Domäne 2	Domäne 3	Domäne 4
Domäne 1	read				switch		
Domäne 2		read					
Domäne 3			execute			switch	switch
Domäne 4	read write						

Verteilte Systeme, Wintersemester 2000/2001

Folie 17.7

Administration

Zusätzliche Rechte

- Copy: (Stern hinter dem jeweiligen Recht)
Prozesse einer Domäne dürfen das Recht in der Spalte vergeben
- Owner: Hinzufügen und Löschen von Rechten in der Spalte
- Control (auf Domäne): Hinzufügen und Löschen in der Zeile

	Objekt 1	Objekt 2	Objekt 3	Domäne 1	Domäne 2	Domäne 3	Domäne 4
Domäne 1	read owner				switch		
Domäne 2		read*					
Domäne 3			execute			switch control	switch
Domäne 4	read write						

Copy

Verteilte Systeme, Wintersemester 2000/2001

Folie 17.8

Realisierungsvarianten

Zugriffskontrolllisten

Speicherung bei Objekten

- Welche Subjekte dürfen welche Operationen ausführen

Beispiele

- Windows NT
- UNIX (Eingeschränkt auf 3 Domänen)

Capabilities

Speicherung beim Subjekt

- Welche Operationen dürfen auf welchen Objekten ausgeführt werden?

Subjekte erhalten nur Verweise auf Capabilities

- Änderung nur durch Betriebssystem

Verteilte Systeme, Wintersemester 2000/2001
Folie 17.9

17.2 Sicherheit

"Arsenal" der Angreifer

Wer greift an?

- Primär reguläre Benutzer
- "Einbrecher"
(Paßwort ermittelt)

Bedrohungen

- Unerlaubtes Mithören
- Fälschen von Nachrichten und Programmen
- Unerlaubte Nutzung von Ressourcen
- Vandalismus



Angriffsformen

- Nachrichtenverkehr abhören (Eavesdropping)
- Vorspiegelung falscher Tatsachen (Masquerading)
- Nachrichten im Verlauf der Übertragung fälschen (Message Tampering)
 - Store-and-Forward-Netze
 - Man in the Middle
- Nachrichten speichern und später wieder einspielen (Replay)
 - Verschlüsseln allein hilft nicht

Verteilte Systeme, Wintersemester 2000/2001

Folie 17.11

... und der Verteidiger



Vertrauenswürdige Basis (Trusted Base)

Sicherheitsanforderungen an Client/Server-Systeme

- Kommunikationskanäle sichern auch Speicherbereiche auf einem Rechner verhindert Abhören
- Client und Server müssen sich wechselseitig mißtrauen
 - Authentifizierung
 - Server: Ist Client rechtmäßiger Vertreter seines Benutzers?
 - Client: Ist Server authentisch?
- Nachrichten mit integriertem Verfallsdatum verhindert u.a. Replay

Verteilte Systeme, Wintersemester 2000/2001

Folie 17.12

Konzepte und Techniken



Verschlüsselungstechniken (Kryptographie)

- Verschlüsseln von Nachrichten
- Grundlage für Authentifizierung
 - Wer eine Nachricht mit einem bestimmten Schlüssel erfolgreich entschlüsselt ist authentisch
- Digitale Signaturen (authentische Unterschriften)

Echtheit beglaubigen (Authentifizierung)

- Ein Geheimnis, daß nur Einer kennen kann, identifiziert ihn
- Authentifizierungsdienst

Zugriff kontrollieren (Schutz)

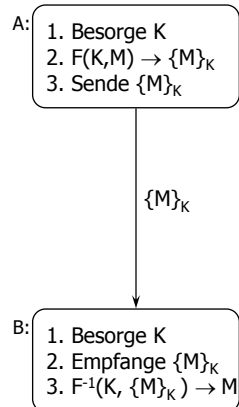
- Objekt-basiert
- Subjekt-basiert

Verteilte Systeme, Wintersemester 2000/2001

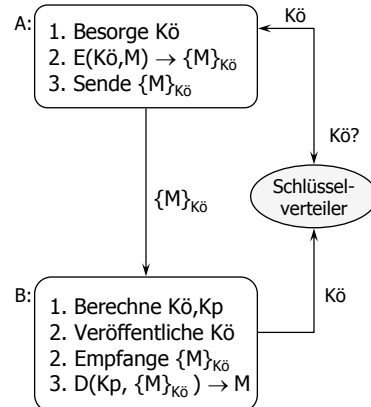
Folie 17.13

Kryptographie

Geheime Schlüssel



Öffentliche Schlüssel



Verteilte Systeme, Wintersemester 2000/2001

Folie 17.14

Geheime Schlüsselverfahren

Data Encryption Standard (DES)

$$F=F^{-1}$$

16 schlüsselabhängige Runden (Rounds)

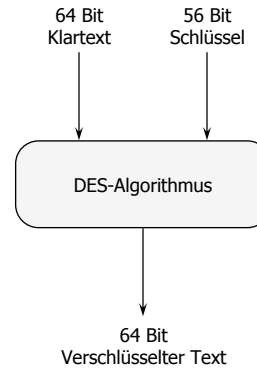
- Bit-Rotationen

3 schlüsselunabhängige Transpositionen

spezielle DES-Chips

- Faktor 1000 schneller als öffentliche Schlüsselverfahren

siehe z.B. Tanenbaum, *Computer Networks*, 2. Auflage, 1996



Öffentliche Schlüsselverfahren

P = 9484758362143462738472734648389448928374783928347920048238479237282717262623627223487283482347238428349823472983883
 Q = 3458374593875240294203723472987439845730958098538752987429874203482-483045739845739529296492764287649549385739833311

RSA von Rivest, Shamir und Adelman (1978)

Ansatz: Faktorisierung sehr großer Zahlen aufwendig

Falltürfunktion $f(x)=y$

- Umkehrfunktion schwierig zu berechnen

Berechnung von K_0 und K_p :

- Wähle Primzahlen P und Q (beide größer als 10^{100})
- $N := P*Q$
- $Z := (P-1)*(Q-1)$
- $d :=$ Zahl relativ prim zu Z
- $e := e*d = 1 \text{ mod } Z$
- $K_0 := \langle e, N \rangle$ und $K_p := \langle d, N \rangle$

Beispiel

- P = 13
- Q = 17
- N := $13*17 = 221$
- Z := $12*16 = 192$
- d := 5
- $e*5 = 1 \text{ mod } 192$
- $1 \text{ mod } 192 = 193, 385, 577, \dots$
- $e := 385/5 = 77$
- Kodiert werden k Bits ($2^k < N$)
- $E(e, N, M) = M^e \text{ mod } N$
- $D(d, N, C) = C^d \text{ mod } N$

Kombinierte Techniken

Geheime Schlüssel

- Austausch des gemeinsamen Schlüssels aufwendig
- Verschlüsselung großer Datenmengen effizient möglich (Hardware-Unterstützung)

Öffentliche Schlüssel

- Bekanntgabe und Weitergabe des öffentlichen Schlüssels unkritisch
- Verschlüsselung zeitaufwendig

Kombination

- Mit Hilfe öffentlicher Schlüssel wird ein geheimer Sitzungsschlüssel (Session Key) ausgetauscht
- Verschlüsselung der Daten mit Session Key

```

sequenceDiagram
    participant Client
    participant Schlüsseldienst
    participant Server
    Client->>Schlüsseldienst: Kö,s?
    Schlüsseldienst->>Client: {S}_{Kö,c}
    Client->>Server: {S}_{Kö,sr}{Daten}_s
    Schlüsseldienst->>Server: Kö
    Client->>Server: {Daten}_s
    
```

Verteilte Systeme, Wintersemester 2000/2001
Folie 17.17

Authentifizierung

```

graph TD
    Client[Client] -- "1. Authentifizierung" --> AS[Authentication Server]
    AS -- "2. Kommunikation mit Server (Ticket)" --> Server[Server]
    Server -.-> AS: "3. eventuell Ticket validieren"
    
```

Authentifizierungsdienst besonders schützen

- Keine Benutzerprozesse
- Firewall
- Besonderer physischer Schutz

Needham und Schröder, 1978

- Authentifizierung mit geheimen Schlüsseln
- Authentifizierung mit öffentlichen Schlüsseln

Verteilte Systeme, Wintersemester 2000/2001
Folie 17.18

Authentifizierung mit geheimen Schlüsseln

- $C \rightarrow AS: C, S, N_C$
- Client C möchte mit Server S kommunizieren
 - $N_C =$ Frischedatum
- $AS \rightarrow C: \{N_C, S, K_{Session}, \{K_{Session}, C\}_{K_S}\}_{K_C}$
- Authentifizierungsdienst
 - Sitzungsschlüssel $K_{Session}$
 - Ticket $\{K_{Session}, C\}_{K_S}$
- $C \rightarrow S: \{K_{Session}, C\}_{K_S}$
- $S \rightarrow C: \{N_S\}_{K_{Session}}$
- Ist Client authentisch oder wiederholt er nur eine Nachricht
- $C \rightarrow S: \{N_S + 1\}_{K_{Session}}$

Verteilte Systeme, Wintersemester 2000/2001

Folie 17.19

Schwachstelle

- $C \rightarrow S: \{K_{Session}, C\}_{K_S}$
- Woher weiß der Server, daß empfangenes Ticket noch frisch ist?

Man will vermeiden, daß Tickets eine beliebig lange Lebensdauer haben

- Server steht Client nur zeitlich begrenzt zur Verfügung
- Client wird das Nutzungsrecht zu einem späteren Zeitpunkt entzogen

Lösung

- Zeitstempel oder Verfallsdatum in Ticket integrieren:
 $\{K_{Session}, C, \text{Verfallsdatum}\}_{K_S}$



Verteilte Systeme, Wintersemester 2000/2001

Folie 17.20

Kerberos (Version 4)

Authentifizierungsprotokoll (Steiner et al., 1988)

basiert auf dem Verfahren von Needham und Schröder

- Geheime Schlüssel
- Erweiterung um Zeitstempel

Integriert u.a. in

- UNIX (MIT)
- AFS
- OSF/DCE

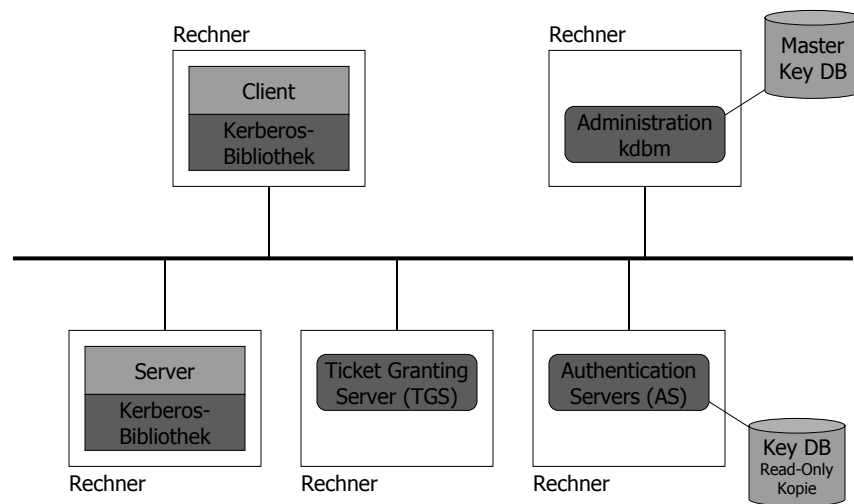
Kerberos-Ticket: $\{\text{Ticket}(C,S)\}_{PK(S)} := \{C,S,t1,t2,K\text{session}\}_{PK(S)}$

- Nur gültig im Zeitintervall $[t1,t2]$

Authentication Server (AS)

Ticket-Granting Server (TGS)

Komponenten von Kerberos



Kerberos-Namen

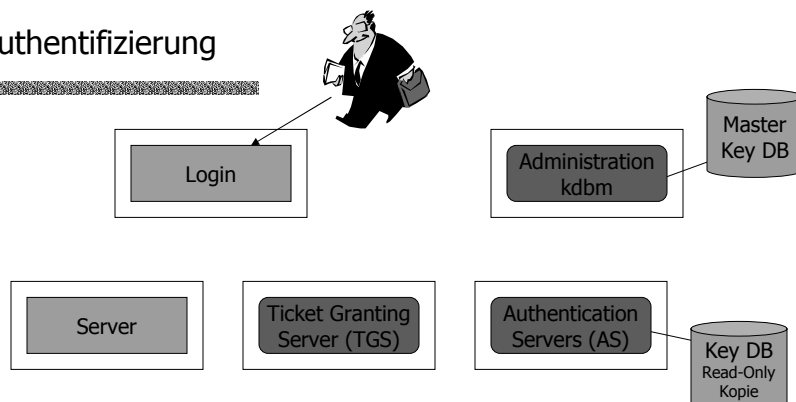
3 Komponenten

- Benutzername (name)
 - Verschiedene Sicherheitsstufen für einen Benutzer
 - Administratorrechte: root oder admin
 - Default: NULL
- Instanz (instance)
 - Verschiedene Sicherheitsstufen für einen Benutzer
 - Administratorrechte: root oder admin
 - Default: NULL
- Realm
 - Schutz- und Authentifizierungsdomäne

Beispiel

- sturm@informatik.uni-trier.de
- grutter.admin@informatik.uni-trier.de

1. Authentifizierung



Anmelden

- Eingabe des Benutzernamens

1. Authentifizierung

Anfordern eines Tickets für den TGS

- Client, TGS

Antwort von AS

- $\{ \text{Session}_{\text{Client,TGS}}, \{ \text{T}_{\text{TGS}} \}_{\text{K}_{\text{TGS}}} \}_{\text{K}_{\text{client}}}$
- $\text{T}_{\text{TGS}} = (\text{TGS}, \text{Client}, \text{Client-Adresse}, \text{Timestamp}, \text{TTL}, \text{Session}_{\text{Client,TGS}})$

Verteilte Systeme, Wintersemester 2000/2001 Folie 17.25

1. Authentifizierung

Client empfängt

- $\{ \text{Session}_{\text{Client,TGS}}, \{ \text{T}_{\text{TGS}} \}_{\text{K}_{\text{TGS}}} \}_{\text{K}_{\text{client}}}$

Eingabe des Passwords

Nur der wirkliche „User“ kann entschlüsseln

- Session-Key zwischen Client und TGS
- Ticket für TGS

Beachte: Password wurde nicht ausgetauscht

Ticket längere Zeit nutzbar (Zeitstempel)

Verteilte Systeme, Wintersemester 2000/2001 Folie 17.26

2. Ticket für Server

Anfordern eines Tickets für den Server S

- $S, \{T_{TGS}\}K_{TGS}, \{A_{Client}\}Session_{Client,TGS}$

Authentizitätsnachweis A_{Client}

- $A_{Client} = (Client, Client-Adresse, Zeitstempel)$

Verteilte Systeme, Wintersemester 2000/2001 Folie 17.27

2. Ticket für Server

TGS empfängt

- $S, \{T_{TGS}\}K_{TGS}, \{A_{Client}\}Session_{Client,TGS}$

Ticket öffnen

- $T_{TGS} = (TGS, Client, Client-Adresse, Timestamp, TTL, Session_{Client,TGS})$

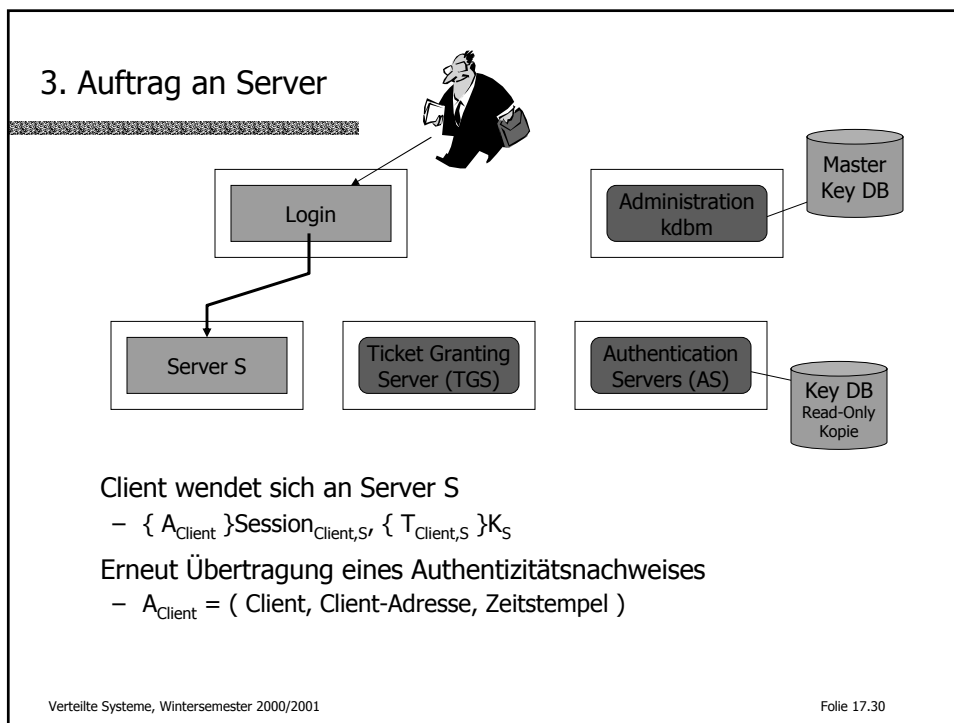
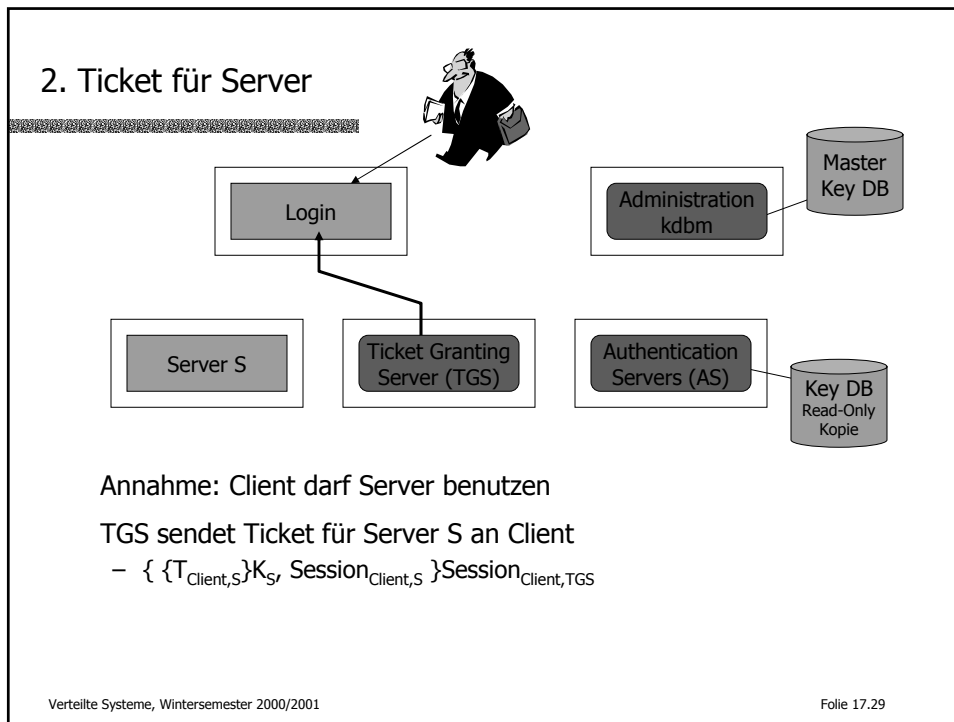
Authentizität überprüfen

- $A_{Client} = (Client, Client-Adresse, Zeitstempel)$

Zeitstempel darf nur wenige Minuten alt sein

- Synchronisierte Uhren

Verteilte Systeme, Wintersemester 2000/2001 Folie 17.28



3. Auftrag an Server

S empfängt

- $\{ A_{Client} \} Session_{Client,S} \{ T_{Client,S} \} K_S$



Ticket öffnen

- $T_{client,S} = (S, Client, Client-Adresse, Timestamp, TTL, Session_{Client,S})$

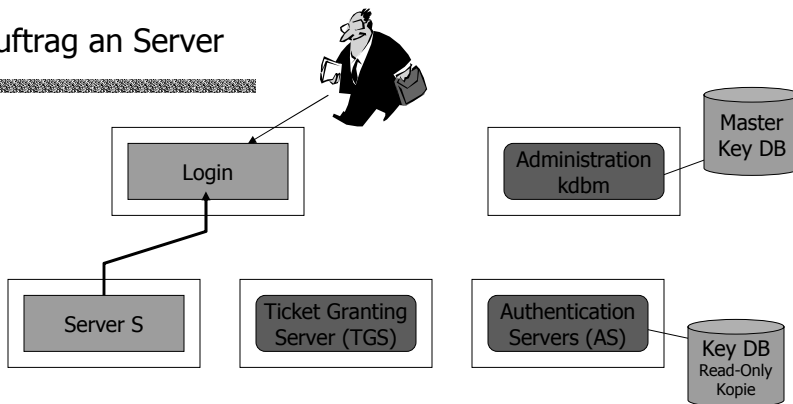
Authentizität überprüfen

- $A_{Client} = (Client, Client-Adresse, Zeitstempel)$

Zeitstempel überprüfen

Services ggf. ausführen

3. Auftrag an Server



Client will Authentizität des Servers

- Challenge/Response

Server sendet an Client

- $\{ \text{Zeitstempel}+1 \} Session_{Client,S}$

Ändern eines Passwords

Client wendet sich AS

- Eingabe des alten Passwords
- AS sendet ein Ticket für kdbm an Client
- Client sendet neues Password an kdbm
Verschlüsselt?

Kdbm

- Instanz = NULL
Nur eigene Passwords änderbar
- Instanz != NULL
Durchsuchen einer Berechtigungsliste
Ggf. Password ändern

Logging aller Vorgänge

Analog Eintrag neuer Clients und Löschungen

Verteilte Systeme, Wintersemester 2000/2001

Folie 17.33

Digitale Signaturen

Wie kann man sicher sein, daß eine Nachricht von X tatsächlich von X kommt?

Eine Lösung

- Öffentliche Schlüssel
- Einschalten eines vertrauenswürdigen Notars N:

1. A → B: M, A, $\{M\}_{SK(A)}$
2. B → N: A
3. N → B: A, PK(A)

- Braucht nicht gesamte Nachricht nochmal verschlüsseln (MD5, ...)

Anwendungen

- Email, z.B. Privacy Enhanced Mail (PEM)
- WWW
- Allgemeine digitale Signaturen, z.B. Pretty Good Privacy (PGP)
- Sichere Capabilities

Verteilte Systeme, Wintersemester 2000/2001

Folie 17.34

Literatur

- R.M. Needham, M.D. Schröder (1978)
Using encryption for authentication in large networks of computers
CACM, Vol. 21, No. 12, pp. 993-999
- R.L. Rivest, A. Shamir, L. Adelman (1978)
A method of obtaining digital signatures and public key cryptosystems
CACM, Vol. 21, No. 2, pp. 120-126
- B. Schneier (1995)
Applied Cryptography: Protocols, Algorithms, and Source Code
Wiley, (2. Auflage)
- J. Steiner, C. Neuman, J. Israel (1988)
Kerberos: an authentication service for open network systems
Proc. Usenix Winter Conference, Berkeley