

PKI-Outsourcing: Vertrauen ist gut, Kryptografie ist besser

Tobias Straub



*Theoretische Informatik – Prof. Johannes Buchmann
Technische Universität Darmstadt*

*Graduiertenkolleg „Enabling Technologies
for Electronic Commerce“*



11. DFN-CERT / PCA Workshop 2004: „Sicherheit in vernetzten Systemen“, Hamburg, 3.–4.2.2004

Überblick

1. Szenario

- Komponenten einer PKI
- Aufgabenverteilung beim Outsourcing
- Risiken

2. Kryptografisches 4-Augen-Prinzip

- Grundidee
- Schwellwert-Kryptografie
- Verteilte RSA-Signaturen

3. Absicherung des Enrollments

- Herkömmliches Enrollment
- Neuer Ansatz
- Prozess-Integration

4. Zusammenfassung

- Erweiterungen
- Offene Fragen



Szenario

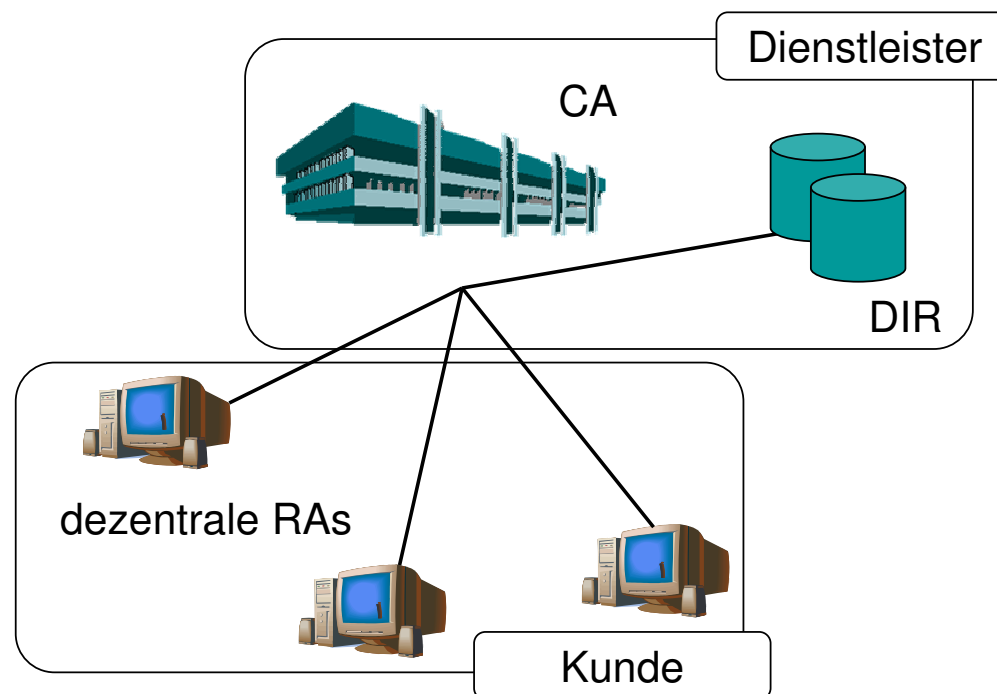
- Public-Key-Infrastruktur mit **hierarchischem Vertrauensmodell**
Vertrauensanker: **Root Certificate** mit öff. Schlüssel pub_{Root}

Komponente	Aufgabe	Anforderungen
CA <i>certification authority</i>	zertifiziert mit Schlüsselpaar $(\text{pub}_{\text{Root}}, \text{priv}_{\text{Root}})$	hohe Sicherheit wg. $\text{priv}_{\text{Root}}$ \Rightarrow Betrieb offline
RA <i>registration authority</i>	registriert Benutzer(-daten) gibt geprüfte Anträge an CA	persönliche Identifizierung \Rightarrow oft mehrere dezentrale RAs
DIR <i>directory</i>	Verzeichnis aller Zertifikate & Sperrlisten	hohe Verfügbarkeit nötig zur Gültigkeitsprüfung



Szenario (Forts.)

- **Aufgabenverteilung** beim PKI-Outsourcing:
 - CA hoch sicher, DIR hoch verfügbar \Rightarrow ausgelagert zum Dienstleister
 - RA für Teilnehmer leicht erreichbar \Rightarrow verbleibt beim Kunden



Risiken

- **priv_{Root}** in **alleiniger** Verantwortung/Kontrolle des **Dienstleisters**
- **Garantie** für korrekte Verwendung d. Zertifizierungs-Schl. **priv_{Root}**?
 - **Dokumentation** interner Abläufe durch *Certificate Policy* / *Certificate Practice Statement*
 - **aber: kein technischer**, höchstens juristischer **Schutz**
 - Dienstleister kann Zertifikate **ohne Kunden-Antrag** ausstellen!
- **bekannter Schadensfall**: gefälschte *Code-Signing*-Zertifikate für „Microsoft“-Mitarbeiter, ausgestellt von VeriSign



Kryptografisches 4-Augen-Prinzip

- bisher: Kunde muss Dienstleister **in hohem Maße vertrauen**
- **neue Idee: Zertifizierungs-Schlüssel $\text{priv}_{\text{Root}}$ soll von**
 - CA (Dienstleister) **und**
 - RA (Kunde)

nur „gemeinsam“ verwendet werden können
⇒ Kunde weiß, wozu $\text{priv}_{\text{Root}}$ benutzt wird
- Technik: **Schwelldwert-Kryptografie** (*threshold cryptography*)



Schwellwert-Kryptografie

- Shamirs (t, k) -Secret Sharing

Idee: teile Geheimnis G unter k Parteien derart, dass

- Gruppen aus t Parteien G wiederherstellen können
- kleinere Gruppen nichts über G erfahren

Problem: G wird Beteiligten explizit bekannt

- Erweiterung: Threshold Function Sharing einer Funktion f_G

t der k Parteien können $f_G(x)$ berechnen

- ohne G wiederherzustellen
- ohne etwas über $f_G(x')$ für $x' \neq x$ zu erfahren



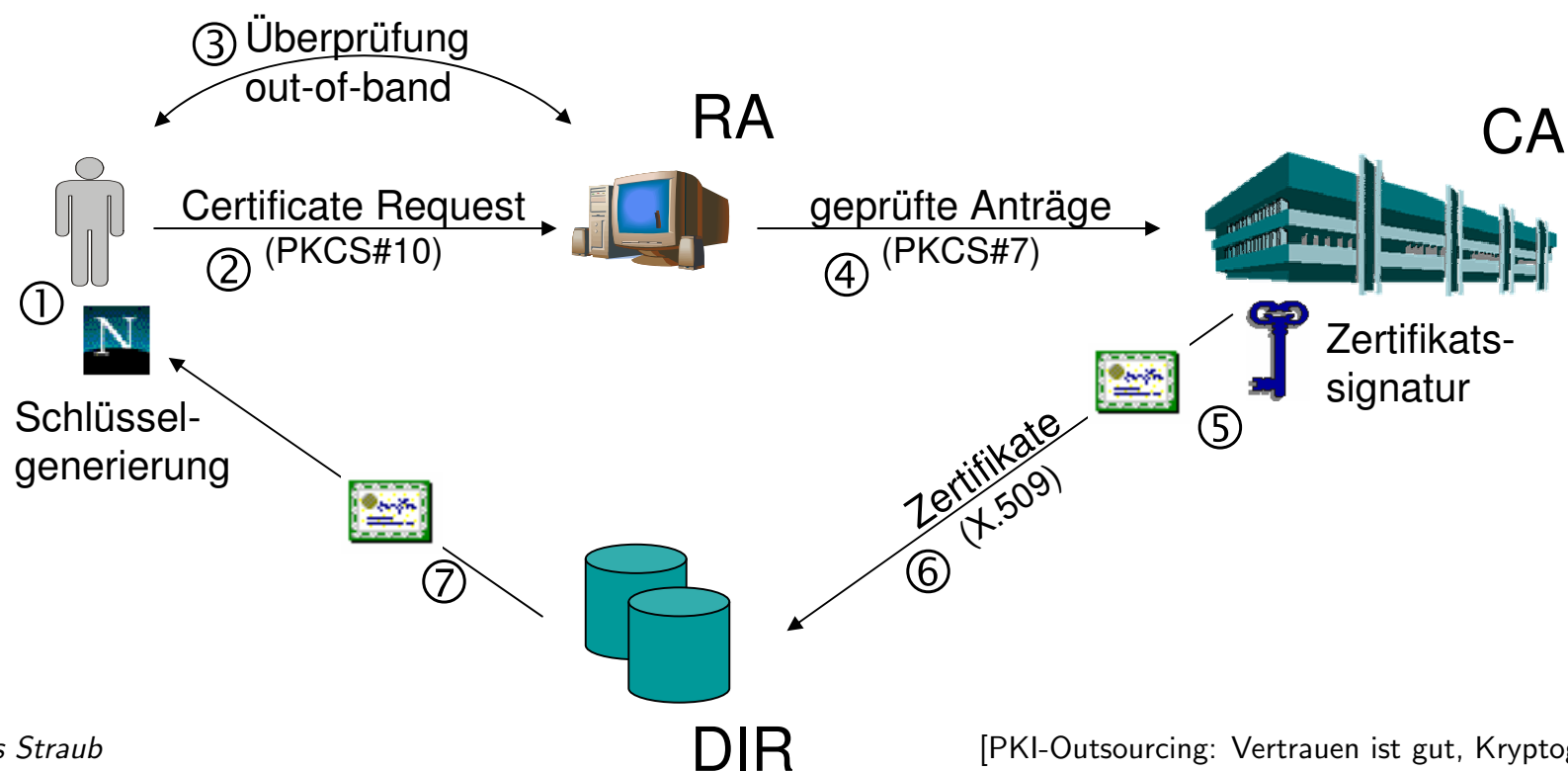
Verteilte RSA-Signatur

- **4-Augen-Prinzip:** $k = 2$ Parteien, Schwellwert $t = 2$
- RSA-Signatur in a nutshell:
 - **man nehme:** N Produkt zweier Primzahlen, $e, d \in \mathbb{N}$
so dass $(x^d)^e \bmod N = x$ für alle $x \in \mathbb{N}$, **Geheimnis:** d
 \Rightarrow **Schlüsselpaar:** $(\text{pub} = (N, e), \text{priv} = (N, d))$
 - **Signieren:** $f_d(x) := h(x)^d \bmod N$ mit Hashfunktion h
Verifizieren: $f_d(x)^e \bmod N \stackrel{?}{=} h(x)$
- Verteilte Signatur: teile Geheimnis d in **Teilschlüssel** $d_1 + d_2 = d$



Absicherung des Enrollments

- Prozess, in dem Benutzer Schlüsselpaar & Zertifikat erhält
- typischer Ablauf eines **dezentralen Enrollments**:



Enrollment in outgesourcter PKI

- **Ziel:** Verhindern, dass CA Zertifikate ohne Antrag ausstellt
- Schwache Lösung: CA archiviert signierte Anträge
 - Beweis, dass Enrollment ordnungsgemäß
 - **aber:** kein effektiver Schutz für Kunde, **Langzeit-Archivierung** aufwändig
- Starke Lösung: **Verteilte Signatur des Zertifikats(-rumpfs) durch Kunde und Dienstleister**
 - ⇒ gefälschte Zertifikate **kryptografisch** ausgeschlossen!
 - ⇒ keine Beweispflicht des Kunden bei Fälschungsverdacht
 - ⇒ keine Archivierung der Anträge erforderlich



Prozess-Integration der Verteilten Signatur

Möglichkeiten abhängig von Verteilung der Rollen „Alice“ & „Bob“ :

1. RA=Alice, CA=Bob:

- RA sendet **teilsigniertes Zertifikat** (x, s_1) an CA (statt Antrag)
- CA berechnet Signatur $s = s_1 \cdot s_2$
- Zertifikat (x, s) mit gültiger Signatur im DIR

2. RA=Bob, CA=Alice: \Rightarrow **keine Änderung der CA-Software**

- RA sendet Anträge wie bisher
- CA **signiert mit Teilschlüssel** d_1 (statt mit d)
- Zertifikat (x, s_1) mit **ungültiger Signatur** im DIR
- RA muss **gegenzeichnen**, Signatur ersetzen $\rightsquigarrow (x, s)$



Zusammenfassung

- Dienstleister kontrolliert Zertifizierungsschlüssel in outgesourcter PKI
- **Gefahr:** er kann Zertifikate ohne Kunden-Antrag ausstellen, Kunde kann nur vertrauen, dass er's nicht tut
- **neues Verfahren:**
 - beweisbarer kryptografischer Schutz durch Verteilte Signatur
 - völlig transparent für externe Nutzer der Zertifikate
 - ohne Änderung der CA-Software nachrüstbar
- Prototyp in JAVA existiert



Erweiterungen, Offene Fragen

- F: Wie erhalten Parteien ihren Teilschlüssel?
A: **Verteilte Schlüsselgenerierung statt vertrauenswürdigen Dritten**
- Signaturen mit **DSA** sind möglich
- **mehrere RAs** sind möglich
- Einsatz in **nicht-outgesourceten PKIs** \Rightarrow **höherer Schutz** von $\text{priv}_{\text{Root}}$
- Policy für Handhabung von Sperrlisten?
- Konformität zu Signaturgesetz?



Vielen Dank!



FRAGEN?

Tobias Straub, TU Darmstadt
tstraub@gkec.tu-darmstadt.de



Tobias Straub

[PKI-Outsourcing: Vertrauen ist gut, Kryptografie ist besser]