

Authentifizierung – Client Certificate

Das Ende der Passwörter

Stefan Helmert

entroserv.de

02/2020

Ausgangssituation

Geteiltes Geheimnis

Nicht geteiltes Geheimnis

Problem

Komplexität und Datenschutz

Folgen

Inkonsistenz

Problem

Komplexität und Datenschutz

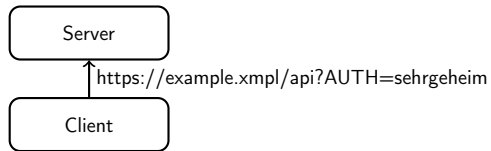
Lösung

Public Key Infrastructure

Ausgangssituation

Geteiltes Geheimnis

Authtoken

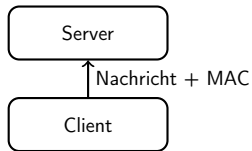


- ▶ jeder Client im Server registriert
- ▶ Server kennt Geheimnis
- ▶ Client kennt Geheimnis
- ▶ Übertragungsweg erfährt Geheimnis

Ausgangssituation

Geteiltes Geheimnis

Message Authentication Code

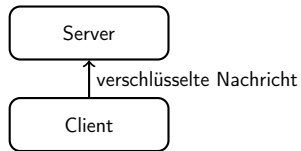


- ▶ jeder Client im Server registriert
- ▶ Server kennt Geheimnis
- ▶ Client kennt Geheimnis
- ▶ Übertrag. erfährt Geheimn. nicht

Ausgangssituation

Geteiltes Geheimnis

Symmetrische Schlüssel

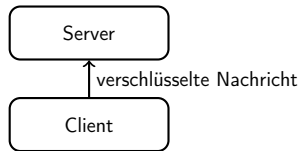


- ▶ jeder Client im Server registriert
- ▶ Server kennt Geheimnis
- ▶ Client kennt Geheimnis
- ▶ Übertrag. erfährt Geheimn. nicht

Ausgangssituation

Nicht geteiltes Geheimnis

Asymmetrische Schlüssel

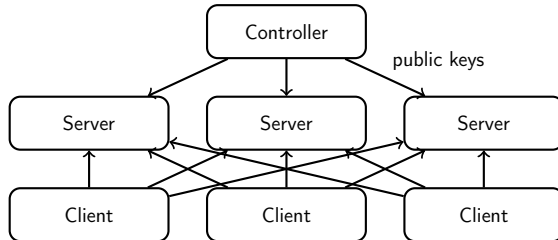


- ▶ jeder Client im Server registriert
- ▶ Server erfährt Geheimnis nicht
- ▶ Client kennt Geheimnis
- ▶ Übertrag. erfährt Geheimn. nicht

Problem

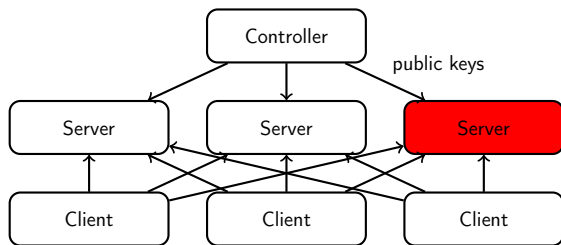
Komplexität und Datenschutz

Asymetrische Schlüssel



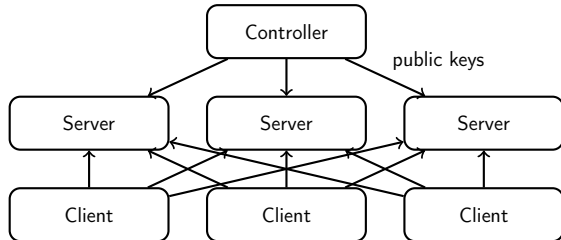
- ▶ jeder Client in jedem Server registriert
- ▶ aufwendige Schlüsselverteilung
- ▶ Konsistenz der Schlüssel
- ▶ Datenschutz – Schlüssel als Identifikationsmerkmal

Asymmetrische Schlüssel



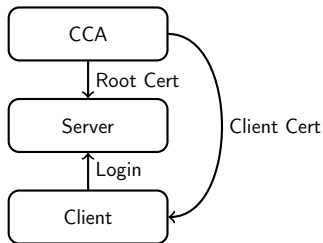
- ▶ Server besitzt ungültigen Schlüssel
- ▶ Server hat neuen Schlüssel nicht
- ▶ abgelaufener, aber verbliebener Schlüssel verrät, welcher Client Zugriff hatte (Löschfrist DSGVO) – Abmahnrisiko

Asymetrische Schlüssel



- ▶ jeder Client in jedem Server registriert
- ▶ aufwendige Schlüsselverteilung
- ▶ Konsistenz der Schlüssel
- ▶ Datenschutz – Schlüssel als Identifikationsmerkmal

Client Certificate



- ▶ Server kennt nur Root Certificate
- ▶ Server erfährt Client Certificate nur beim Login
- ▶ CCA kennt Root Certificate und zugehörigen Private Key
- ▶ Client kennt Client Certificate und zugehörigen Private Key

Ablauf

1. CCA: Schlüsselpaar erzeugen
2. CCA: Root CSR erzeugen
3. CCA: Root Cert erzeugen (CSR selfsigning)
4. Root Cert CCA → Server
5. Client: Schlüsselpaar erzeugen
6. Client: CSR erzeugen (inkl. pub key)
7. CSR Client → CCA
8. CCA: signiert Client CSR (Ergebnis Client Cert)
9. Client Cert CCA → Client
10. Login-Anfrage Client → Server
11. Challenge Server → Client
12. Client signiert Challenge (=Response)
13. Response + Client Cert
Client → Server
14. Server prüft Response, Client Cert
15. Session Server ↔ Client