SRN Semesterprojekt

Zwei-Faktor-Authentifizierung

> René Schmitt, IB, 1131971
> Christian Titze, IB, 1123377
> Timo Notheisen, IB, 1120722
> Martin Tänzer, IB, 1123643

Quelle: cultofandroid.com

Inhalt

- Was ist Zwei-Faktor-Authentifizierung?
- Grundlagen
- Funktionsweise
- Sicherheitsaspekte
- Demo

Was ist Zwei-Faktor-Authentifizierung?

- Identitätsnachweis eines Nutzers mittels Kombination zweier verschiedener Komponenten
- Komponenten können sein:
  - etwas, was der Nutzer *besitzt* (Bankkarte, …)
  - etwas, was der Nutzer *weiß* (Passwort, …)
  - etwas, was der Nutzer *ist* (Fingerabdruck, …)

Bestandteile des Systems.

GRUNDLAGEN

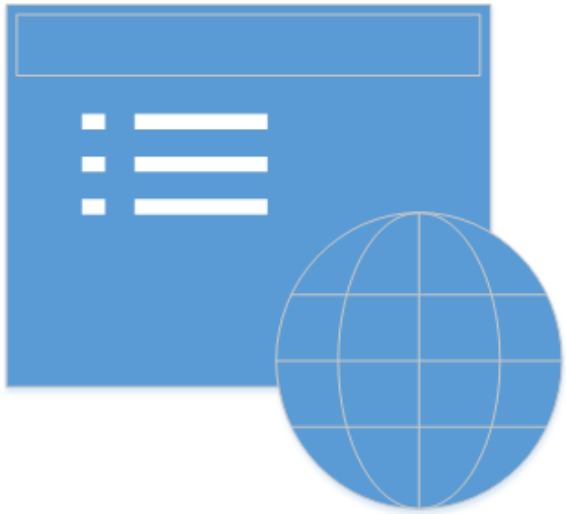
Kontext

- Benutzer will sich über Desktop-App mittels 2FA sicher einloggen:
  - Login in Desktop-App mittels Benutzername und Passwort
  - Desktop-App zeigt sichere Zufallszahl (Token) an
  - Website verifiziert Token + Geheimnis (zweiter Faktor) und benachrichtigt Desktop-App über Erfolg

Akteure

Client mit
Desktop App

Server mit
Datenbankanbindung



Webserver

Datenbankschema

Spalte	Typ	Beschreibung
userId	int(11)	Primärschlüssel, auto-inkrement
username	varchar(50)	Benutzername im Klartext
password	varchar(200)	Passwort mit Salt gehashed mit SHA-2
salt	varchar(200)	Salt für Passwort-Hashing
secret	varchar(20)	Zweites Geheimnis im Klartext
token	varchar(20)	Zufälliger sicherer Token
expirationDate	varchar(50)	Zeitpunkt, ab dem Token ungültig ist
secondAuthentication	bit(1)	Zweite Authentifizierung erfolgreich?
tokenUsed	bit(1)	Wurde der Token bereits benutzt?

Datenübertragung

- Alle Daten werden als JSON versendet
- JSONs sind zur Gewährleistung der Integrität mit SHA-2 **gehashed**
- JSONs sind mit symmetrischem Key AES-**verschlüsselt**
- JSONs vom Server werden **signiert** mit Private Key

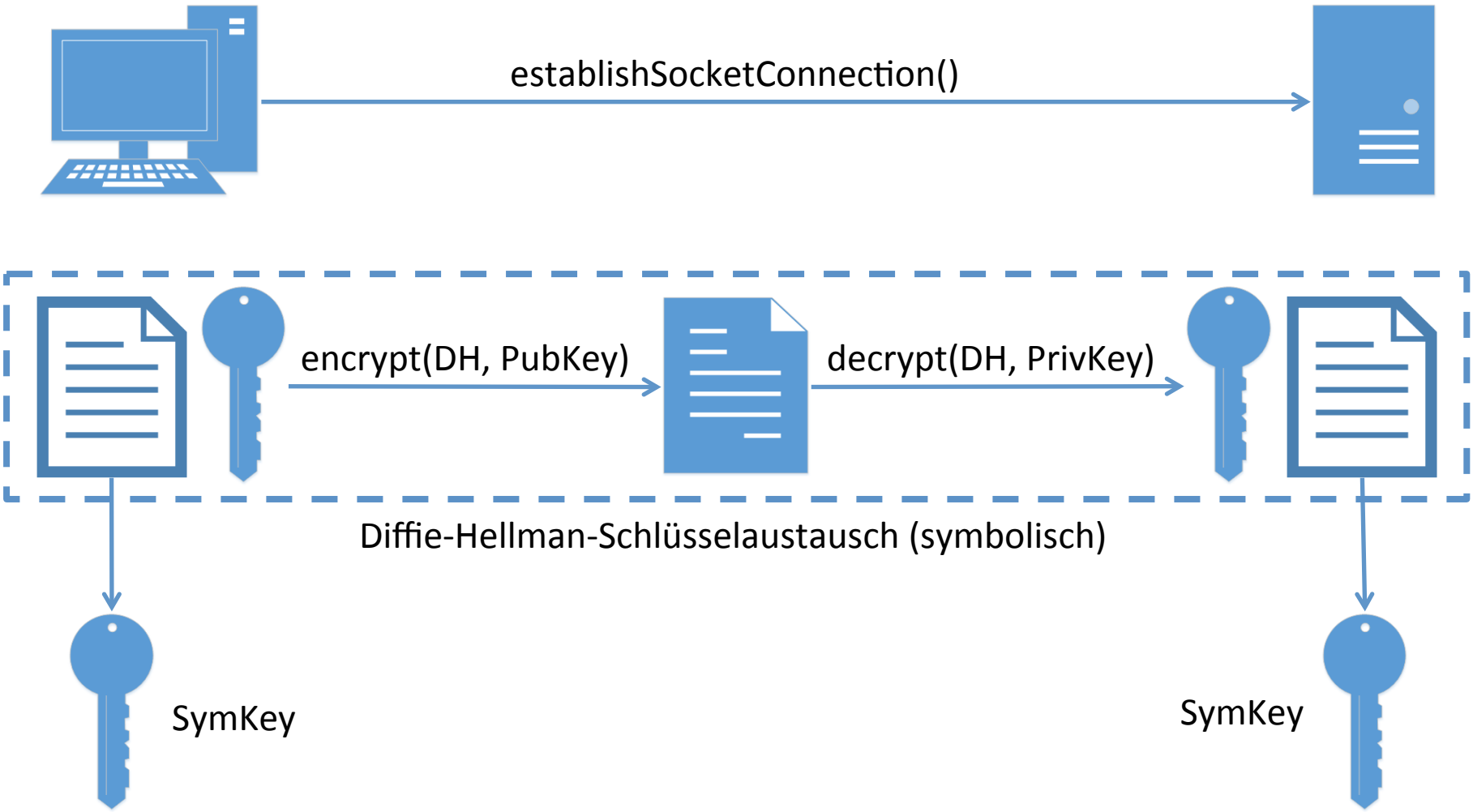
Ablauf der Zwei-Faktor-Authentifizierung.

FUNKTIONSWEISE

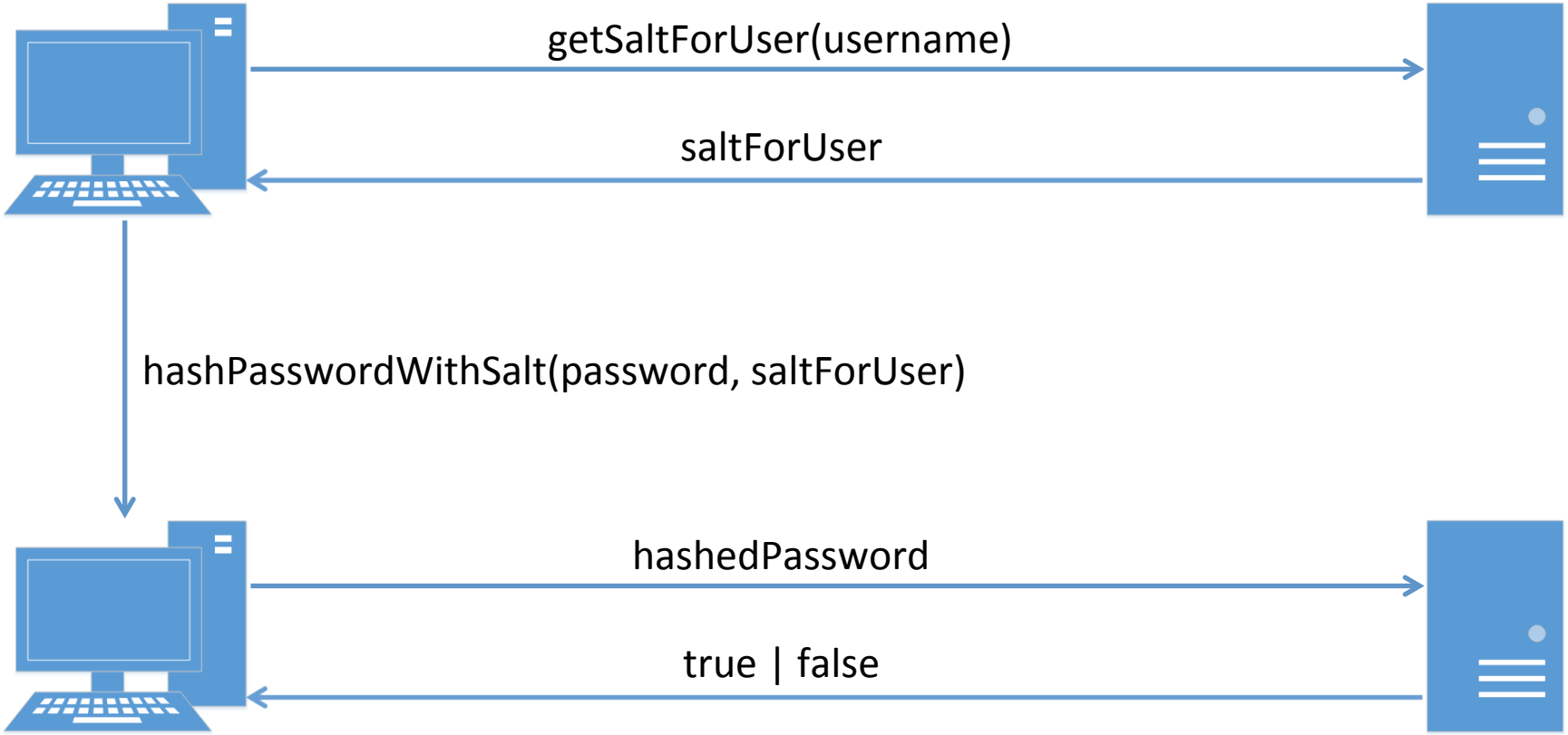
Information

- Die im Folgenden vorgestellte Implementierung demonstriert keine *echte* Zwei-Faktor-Authentifizierung. Vielmehr ist das vom Benutzer gewählte Secret als der Faktor anzusehen, den er besitzt. In einer realen Umgebung wäre dies z.B. eine Kreditkarte oder ein USB-Stick.

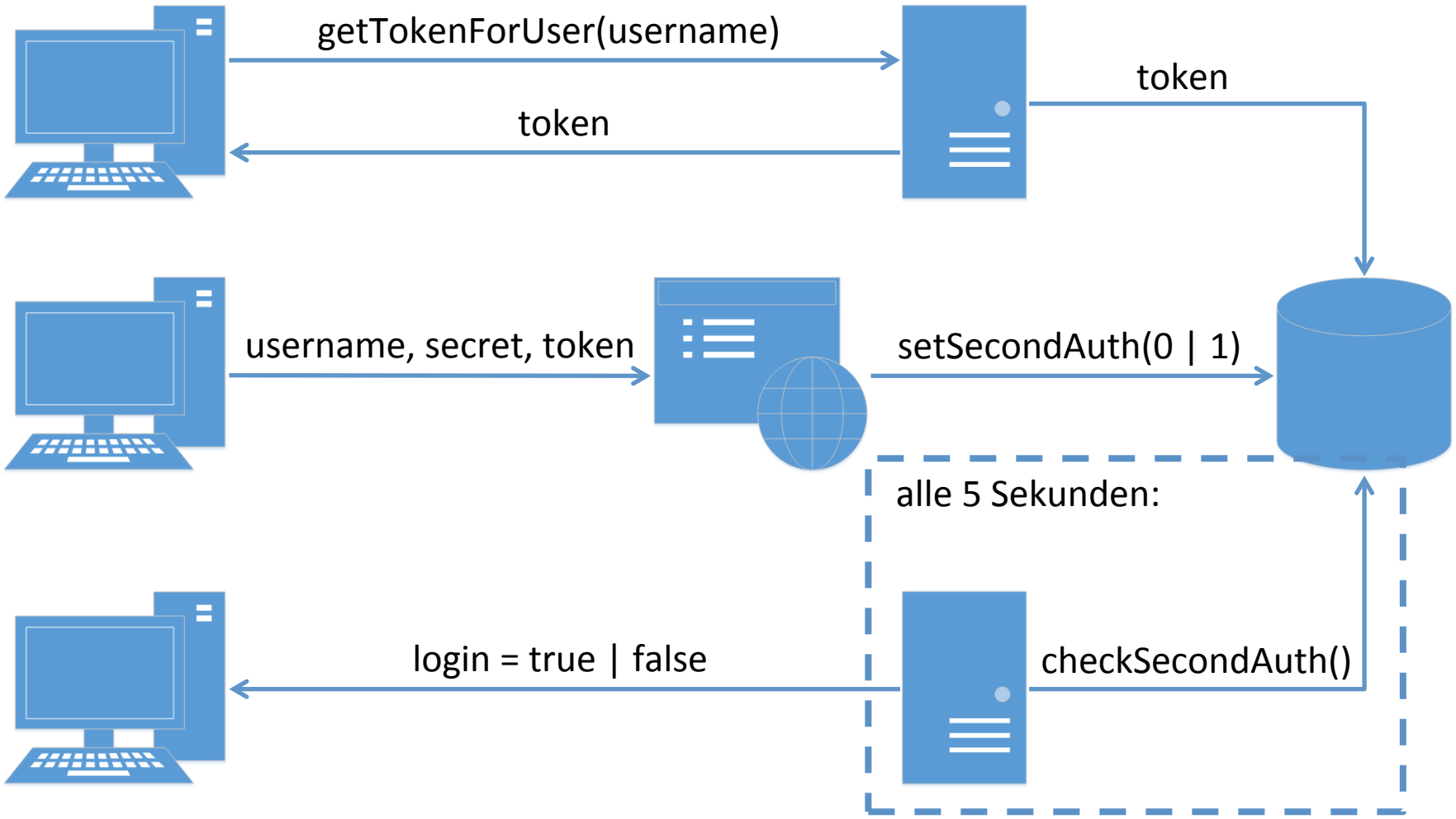
1. Verbindungsaufbau



2. Login (Erster Faktor)



3. Token + Zweiter Faktor



Verhinderung möglicher Angriffe.

SICHERHEITSASPEKTE

Angriffsmethoden

- **Man in the Middle:** Möglich, aufgrund von Verschlüsselung und Private-Key-Signatur auf Serverseite jedoch sinnlos
- **Brute Force:** Möglich, aufgrund von mehrmaligem Hashing jedoch sehr langsam
- **SQL Injection:** Durch Prepared Statements und Input-Regex ausgeschlossen
- **Replay-Angriffe:** Durch einmaliges Token mit Zeitfenster ausgeschlossen
- **“Token abschreiben”:** Möglich, durch wenige Sekunden andauerndes Zeitfenster und ohne Kenntnis des zweiten Faktors jedoch sinnlos

Erfüllte Sicherheitskonzepte

- **Authentifizierung:** Durch signierte Nachrichten garantiert
- **Integrität:** Alle Nachrichten werden gehashed

Vorstellung des Prototyps.

DEMO

Vielen Dank für Ihre Aufmerksamkeit.

- Fragen?
- Source Code verfügbar auf: https://github.com/timnot90/SIT_2FA
- Entwickler:
  - René Schmitt, 6IB, 1131971
  - Christian Titze, 6IB, 1123377
  - Timo Notheisen, 6IB, 1120722
  - Martin Tänzer, 6IB, 1123643
  - Kontakt: {1131971, 1123377, 1120722, 1123643}@stud.hs-mannheim.de