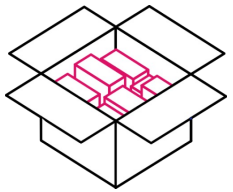
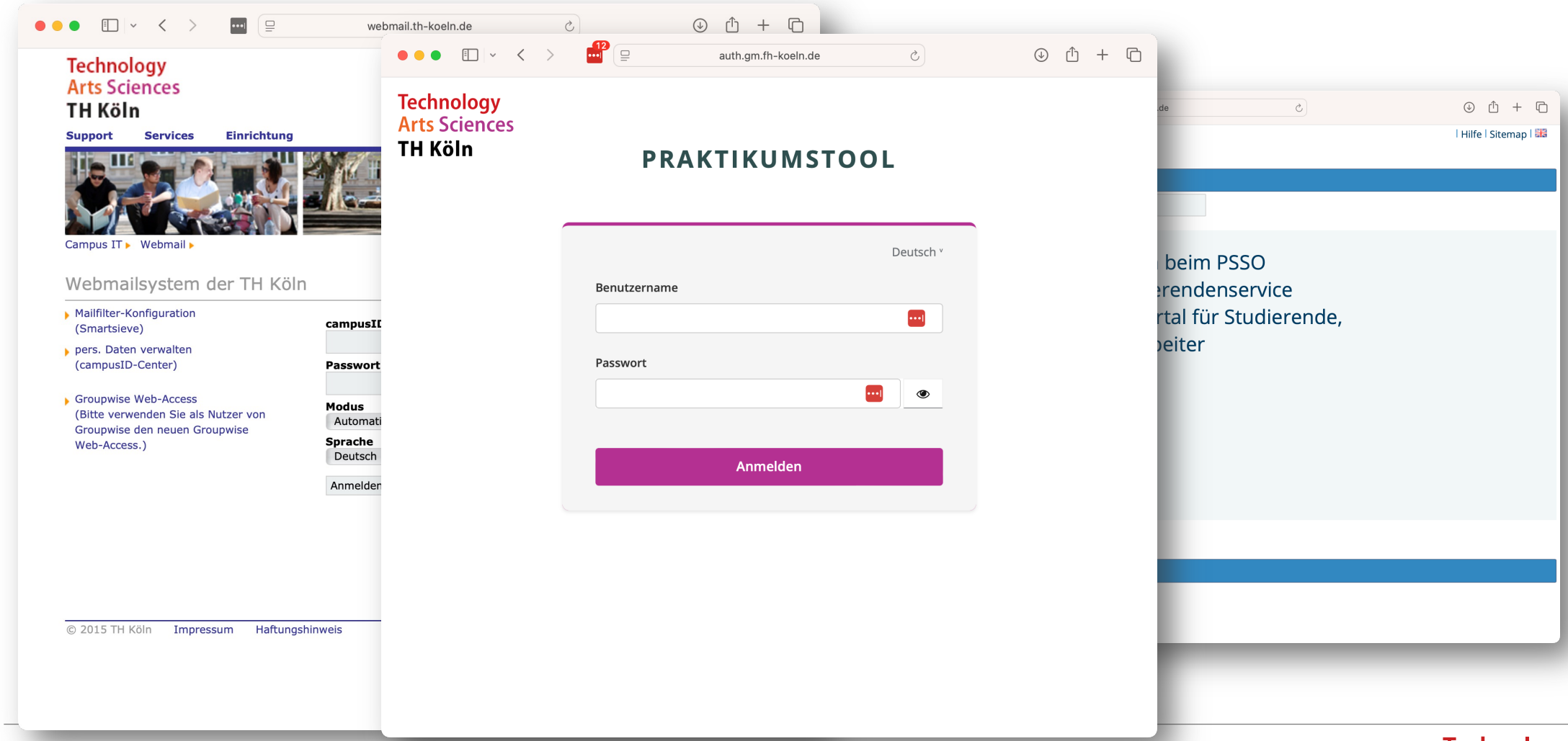
Authentifizierungsverfahren im Web

Hoai Viet Nguyen – TH Köln

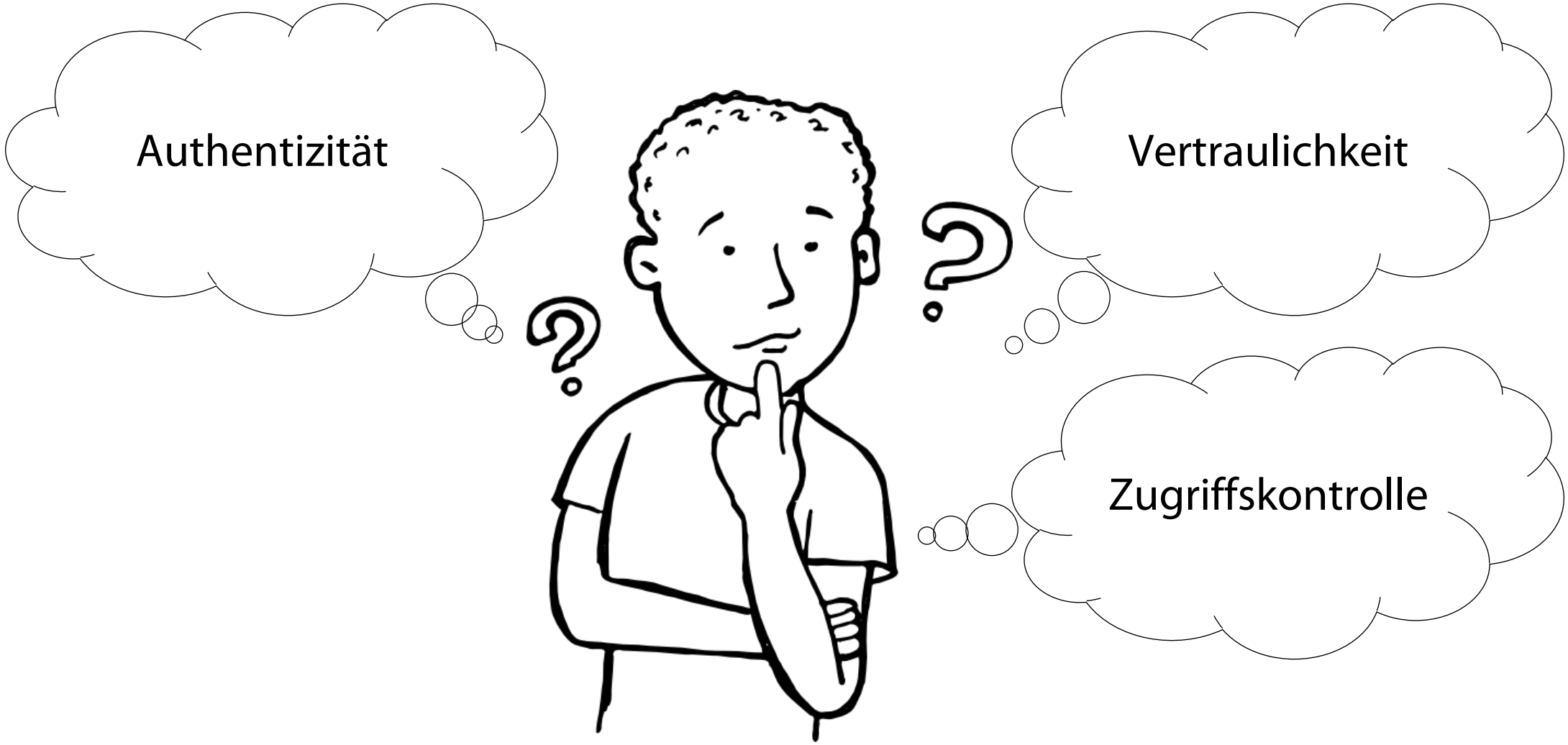
Technology
Arts Sciences
TH Köln



Formular-basierte Authentifizierung



Welche Schutzziele verfolgt ein Password-basiertes Login-Verfahren ?



Authentifizierungsverfahren (Auszug)

- Formular-basierte Authentifizierung
- HTTP-Basic/Digest
- API-Keys
- Lernzielkontrolle und Zusammenfassung

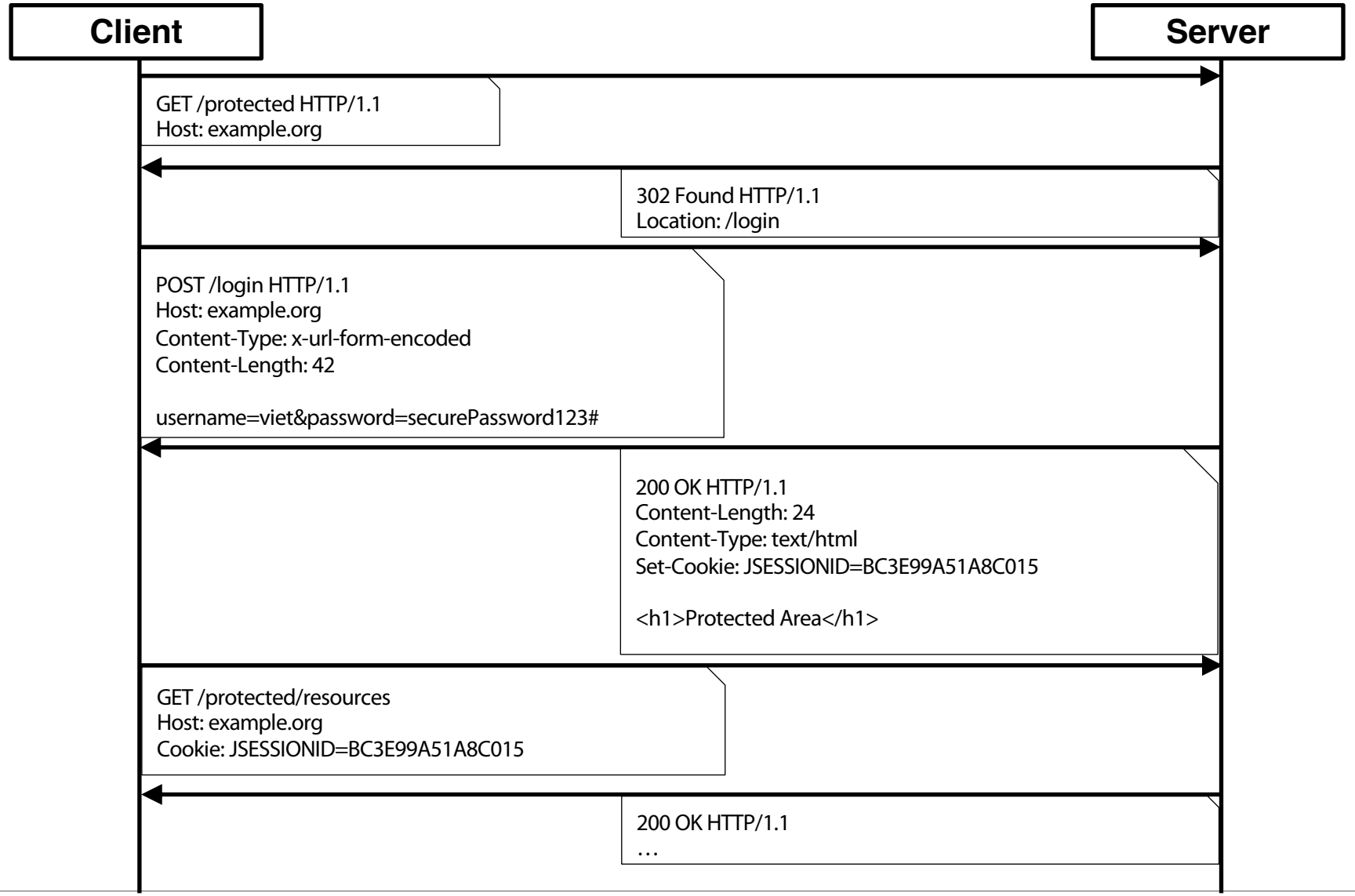
Authentifizierungsverfahren (Auszug)

- Formular-basierte Authentifizierung
- HTTP-Basic/Digest
- API-Keys
- Lernzielkontrolle und Zusammenfassung

Formular-basierte Authentifzierung

- Clientseitige Authentifizierung gegenüber dem Server
  - Nach der Authentifizierung wird ein Cookie mit Session ID gesetzt
  - Session ID wird bei jedem weiteren Request mitgeschickt
  - Server prüft bei jeder Anfrage die Session ID
  - Nach dem Logout wird die Session ID gelöscht
- Kann i.d.R nur mit Hilfe eines Webframeworks implementiert werden
- Geringes Sicherheitsniveau, da Passwörter und Session ID im Klartext übertragen werden
- Sollte nur in Verbindung mit TLS verwendet werden
- Anmeldeformular individualisierbar
- Weite Verbreitung

Formular-basierte Authentifzierung Login



Demo – Formular-basierte Authentifizierung

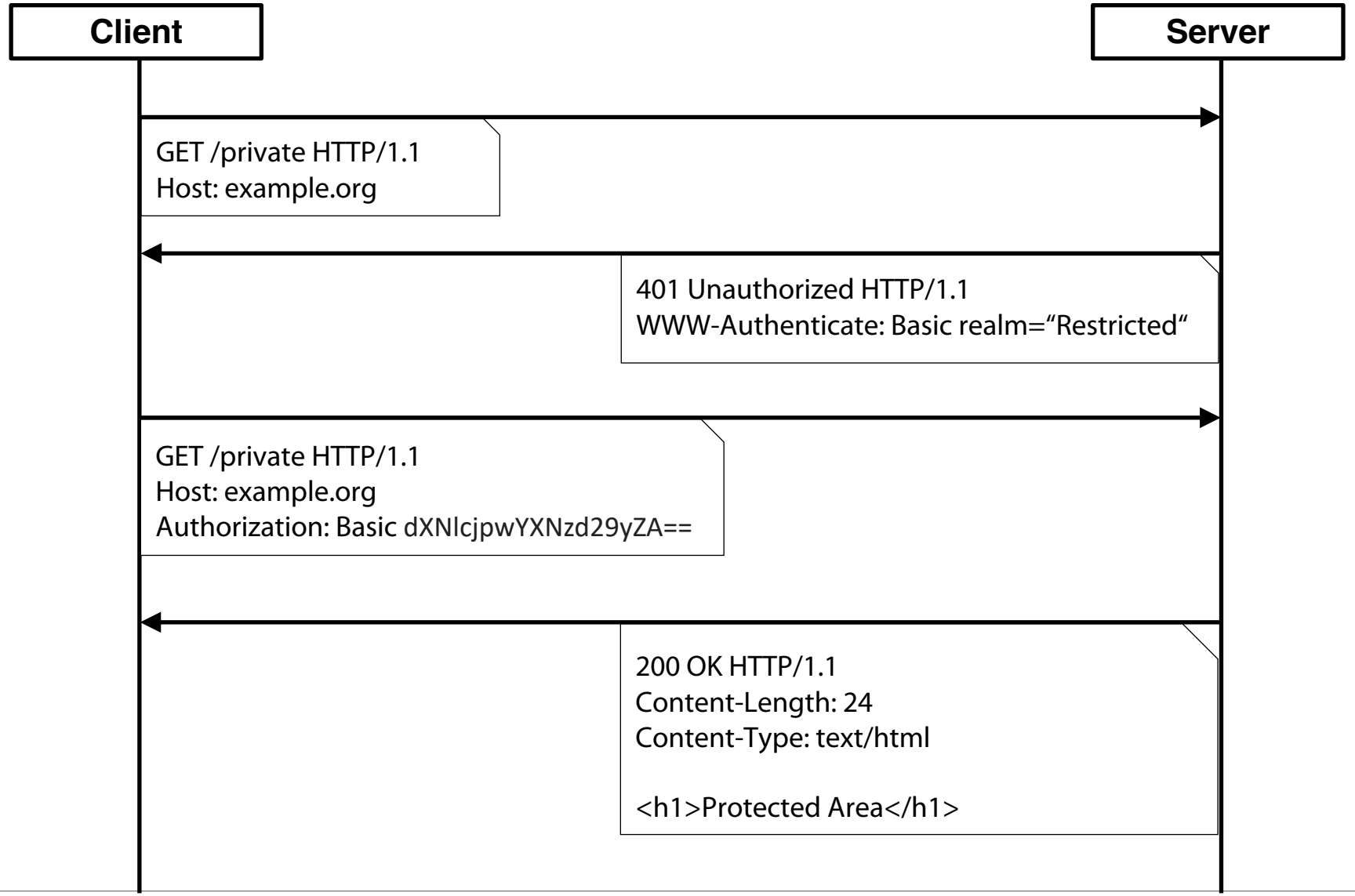
Authentifizierungsverfahren (Auszug)

- ✓ Formular-basierte Authentifizierung
- ➢ HTTP-Basic/Digest
- API-Keys
- Lernzielkontrolle und Zusammenfassung

HTTP Basic Authentication

- Wird von vielen Webservern und Framewoks unterstützt u.a. Apache HTTPD, Nginx und Spring
- Bietet einseitige Authentifizierung des Clients gegenüber Server
- Password im Klartext bzw. Base64-kodiert übertragen
- Um Vertraulichkeit des Passworts bei der Übertragung zu garantieren muss HTTP Basic mit TLS verwendet werden
- Unterstütze Password-Hashfunktionen für Speicherung:
  - crypt (default)
  - MD5
  - bcrypt

HTTP Basic Authentication



Demo – HTTP Basic Authentication

HTTP Digest Authentication

- Wird von vielen Webservern und Framewoks unterstützt
- Keine native Unterstützung von Nginx
- Höheres Sicherheitsniveau bei der Übertragung als HTTP Basic:
  - Passwort wird nicht in Klartext, sondern als Hash übertragen
  - Passwort-Token ist ein Hash aus Benutzername, Passwort, HTTP-Methode, Realm, Nonce (Zufallszahl von Server) und Optional cnonce (Zufallszahl vom Client)
- RFC spezifiziert MD5 und SHA2-256 bzw. SHA2-512 als Hashfunktion
- Allerdings unterstützen viele HTTP-Engines nur MD5
- MD5 wird bei vielen HTTP-Engines auch für Passwort-Speicherung verwendet

HTTP Digest Authentication

Demo – HTTP Digest Authenticaton

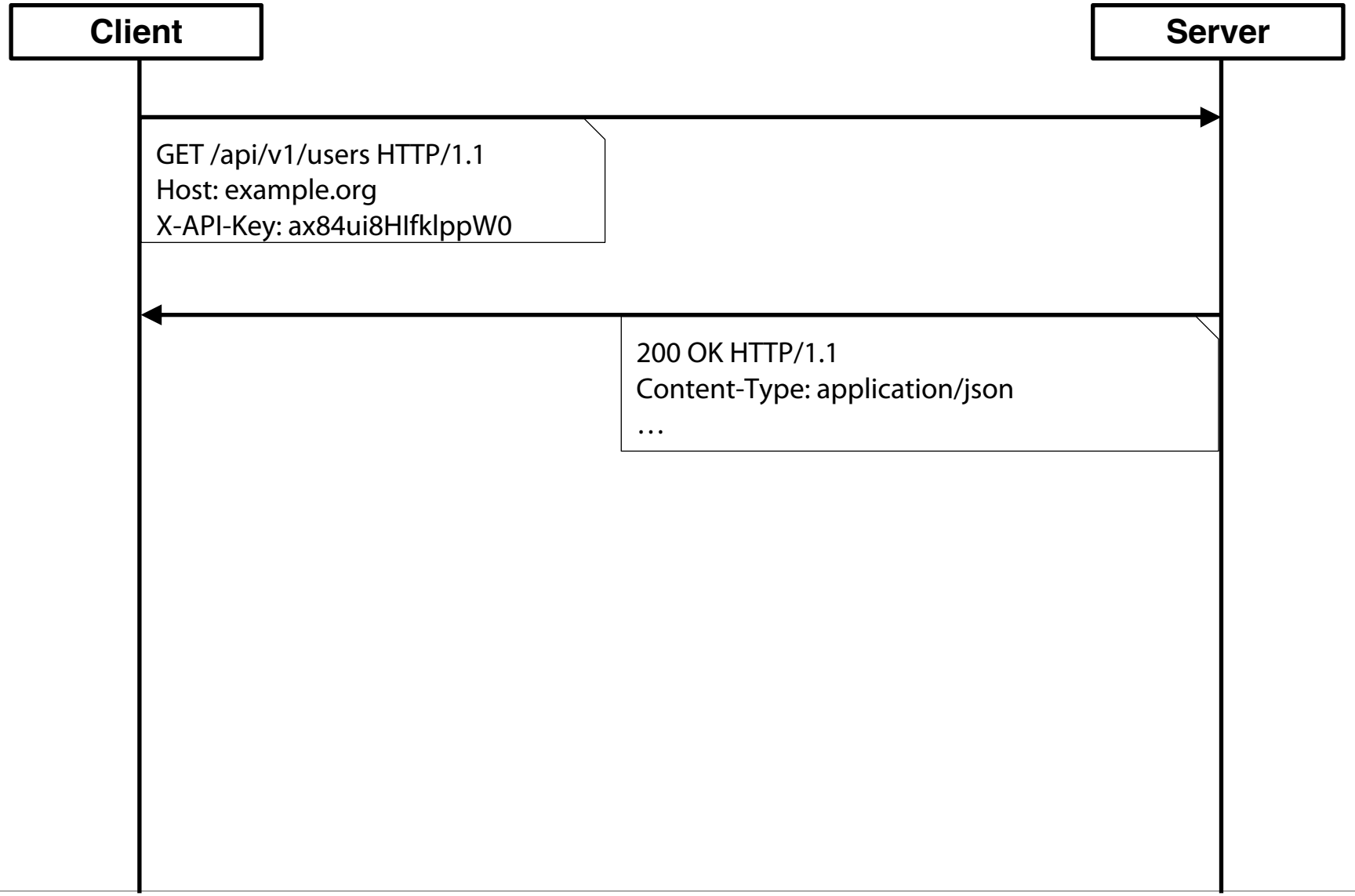
Authentifizierungsverfahren (Auszug)

- ✓ Formular-basierte Authentifizierung
- ✓ HTTP-Basic/Digest
- ➢ API-Keys
- ▪ Lernzielkontrolle und Zusammenfassung

API-Keys

- Wird i.d.R. für Clientseitige Authentifizierung von REST-APIs verwendet
- API-Key wird einem Headerfeld oder in der URL übertragen
- Geringes Sicherheitsniveau, da Passwort in Klartext übertragen wird
- In allen HTTP-Engines unterstützt bzw. trivial zu implementieren
- Kein standardisiertes Verfahren, dennoch weite Verbreitung

API-Keys


Client
Server
GET /api/v1/users HTTP/1.1
Host: example.org
X-API-Key: ax84ui8HlfklppW0
200 OK HTTP/1.1
Content-Type: application/json
…

Authentifizierungsverfahren (Auszug)

- ✓ Formular-basierte Authentifizierung
- ✓ HTTP-Basic/Digest
- ✓ API-Keys
- ➢ Lernzielkontrolle und Zusammenfassung

Lernzielkontrolle

- Was sind die Schutzziele von Password-basierte Authentifizierungsverfahren?
- Was ist HTTP Basic/Digest Authentication?
- Was ist Formular-basierte Authentifizierung?
- Was sind API-Keys?