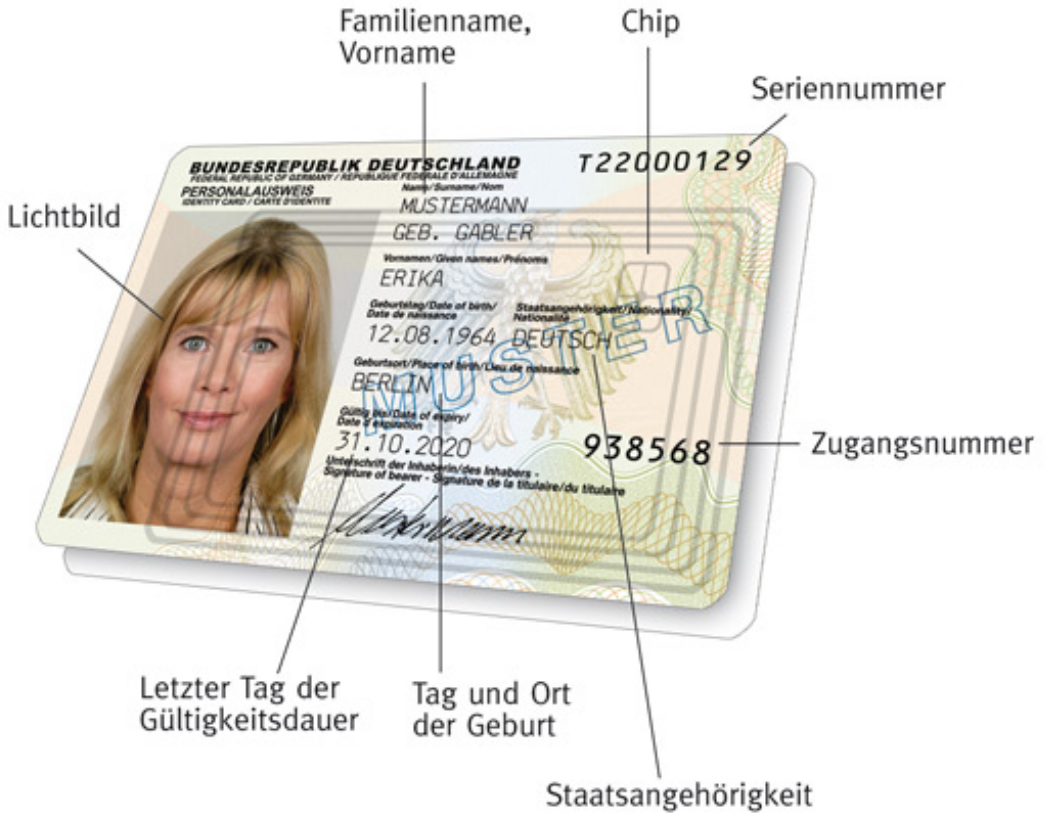
Jörg Schilling

Die Technik des elektronischen Personalausweises

Fokus Fraunhofer

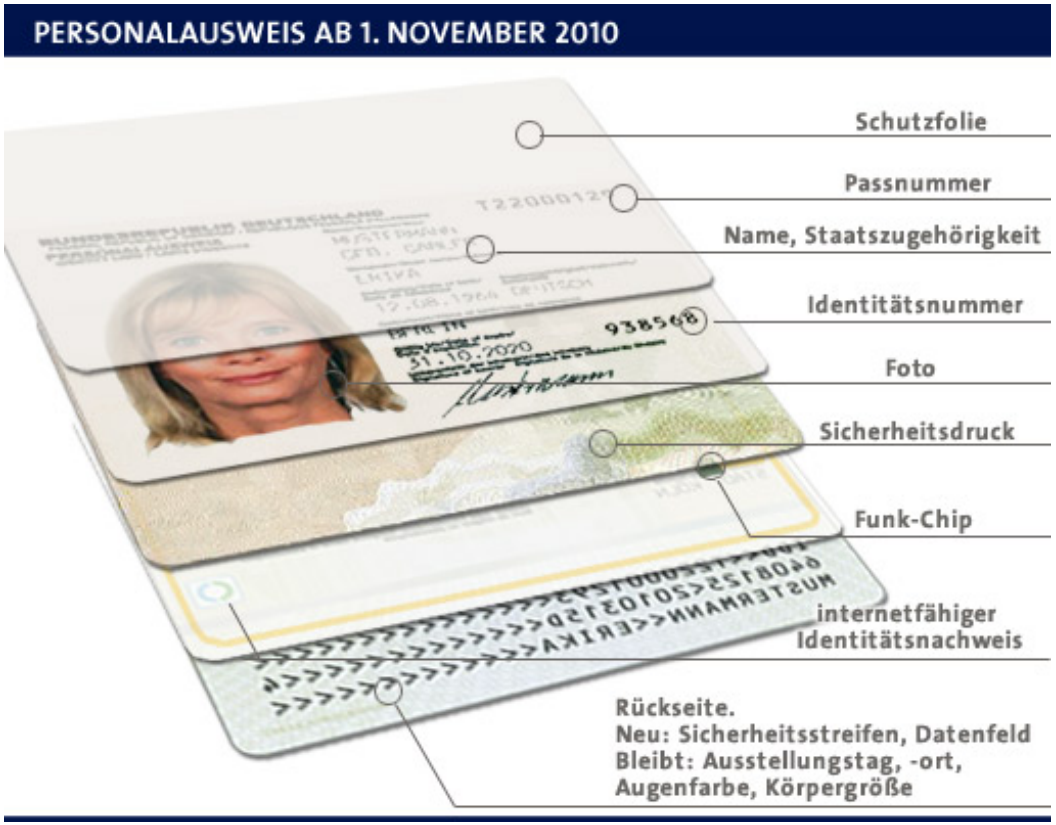
Fraunhofer FOKUS

Vorderseite des neuen Personalausweises



Rückseite des neuen Personalausweises

Schichtaufbau



Daten auf dem Chip des Personalausweises

- **Der Chip enthält alle aufgedruckten Daten außer:**
  - **Ausweisnummer**
  - **CAN**
  - **Unterschrift**
  - **Augenfarbe**
  - **Körpergröße**
  - **Ausstellungsdatum**
  - **Ausstellende Behörde**
- **Der Chip enthält folgende zusätzliche Informationen:**
  - **Ein digitales biometrisches Foto**
  - **Optional: Fingerabdrücke**

Der Chip auf dem Ausweis

- **Kontaktloser Chip nach ISO/IEC-14443**
- **Kommunikation nach ISO/IEC-7816 Teil 4**
  - **Kommando Antwort Paare**
  - **Application Protocol Data Unit (APDU)**
  - **Secure Messaging**
  - **Struktur von Anwendungen und Daten**
  - **Dateiorganisation**
  - **Zugriffsrechte**
- **ISO-24727 Teil 3**
  - **Schnittstellen zwischen Chipkarten und externen Applikationen (Service Access layer)**

Verfügbare Funktionen auf dem Chip

- Personen- und Dokumentenbezogene Daten
- Keine Seriennummer (Zufallszahl bei Power ON)
- Alle Daten sind gegen Auslesen gesichert
  - Auslesen nur mit gültigem Zertifikat möglich
- Zugriff auf biometrische Daten nur hoheitlich
- eID Funktion kann deaktiviert werden
  - Zugriff mit PIN ist dann nicht möglich
  - Zugriff mit CAN oder MRZ (hoheitlich) ist weiter möglich
  - eID Funktion mit PIN kann später wieder aktiviert werden
- Qualifizierte elektronische Signatur nachladbar

Auch „Nicht-Hoheitlich“ auslesbar

- **DG1: Dokumenten Typ (ID – Test / TP – Referenz / TA - Referenz-Aufenthaltserlaubnis / Echt Ausweis / Echt Aufenthaltserlaubnis)**
- **DG2: Ausgebender Staat**
- **DG3: Ablaufdatum**
- **DG4: Vornamen**
- **DG5: Nachnamen**
- **DG6: Ordensname / Künstername / Pseudonym**
- **DG7: Akademischer Titel**
- **DG8: Geburtstag**
- **DG9: Geburtsort**
- **DG10: Staatsangehörigkeit (Nicht auf dem Ausweis)**
- **DG11: Geschlecht (nicht auf dem Ausweis)**
- **DG17: Wohnort**
- **DG18: Gemeindeschlüssel**
- **DG19: Meldebehörde I (nicht auf dem Ausweis)**
- **DG20: Meldebehörde II (nicht auf dem Ausweis)**

- Das Gesichtsfoto
- Die Fingerabdrücke (optional)

Nicht-Hoheitliche Sonderfunktionen

- **Anfrage durch Aux-Data bei PACE:**
  - **Abfrage ob der Ausweis zu einem gegebenen Datum noch gültig ist**
  - **Abfrage ob der Inhaber vor einem gegebenen Datum geboren wurde**
  - **Abfrage ob der Inhaber in der Nähe einer Gemeinde wohnt**
- **Berechnung durch Kryptografie:**
  - **Berechnung einer „Restricted ID“ (Synomym) aus Zertifikat und geheimen Schlüssel im Ausweis**

Passive und aktive Authentisierung

- **Standard für Reisedokumente durch *International Civil Aviation Organisation* (ICAO)**
- **Passive Authentisierung (Pflicht)**
  - **Signierte Daten**
  - **In Personalisierungsphase Signierung d. Hersteller d. Ausweises**
  - **Authentizität d. Daten verhindert kein Klonen**
- **Aktive Authentisierung (Optional)**
  - **Chip besitzt eigenes Schlüsselpaar**
  - **Challenge-Response Protokoll**
  - **Datenschutzprobleme (Challenge kann Infos preisgeben)**

Passive und aktive Authentisierung

- Access Control:
- Basic Access Control – optional
  - Physischer Zugang zum Dokument nötig
  - Symmetrischer Schlüssel aus MRZ abgeleitet
- Extended Access Control – optional
  - Starker Schlüssel durch Chip Authentisierung
  - Zugriffskontrolle durch Terminal Authentisierung

Reisepass und Personalausweis

- Beim Reisepass: *Basic Access Control*
  - Sitzungsschlüssel auf Basis der MRZ
  - Geringe Entropie
  - Knackbar
- Beim Personalausweis: *Extended Access Control*
  - PIN-Überprüfung ohne Übertragung, durch PACE
  - Zugriffskontrolle durch TA und Zertifikate
  - Erkennen von Fälschungen durch CA

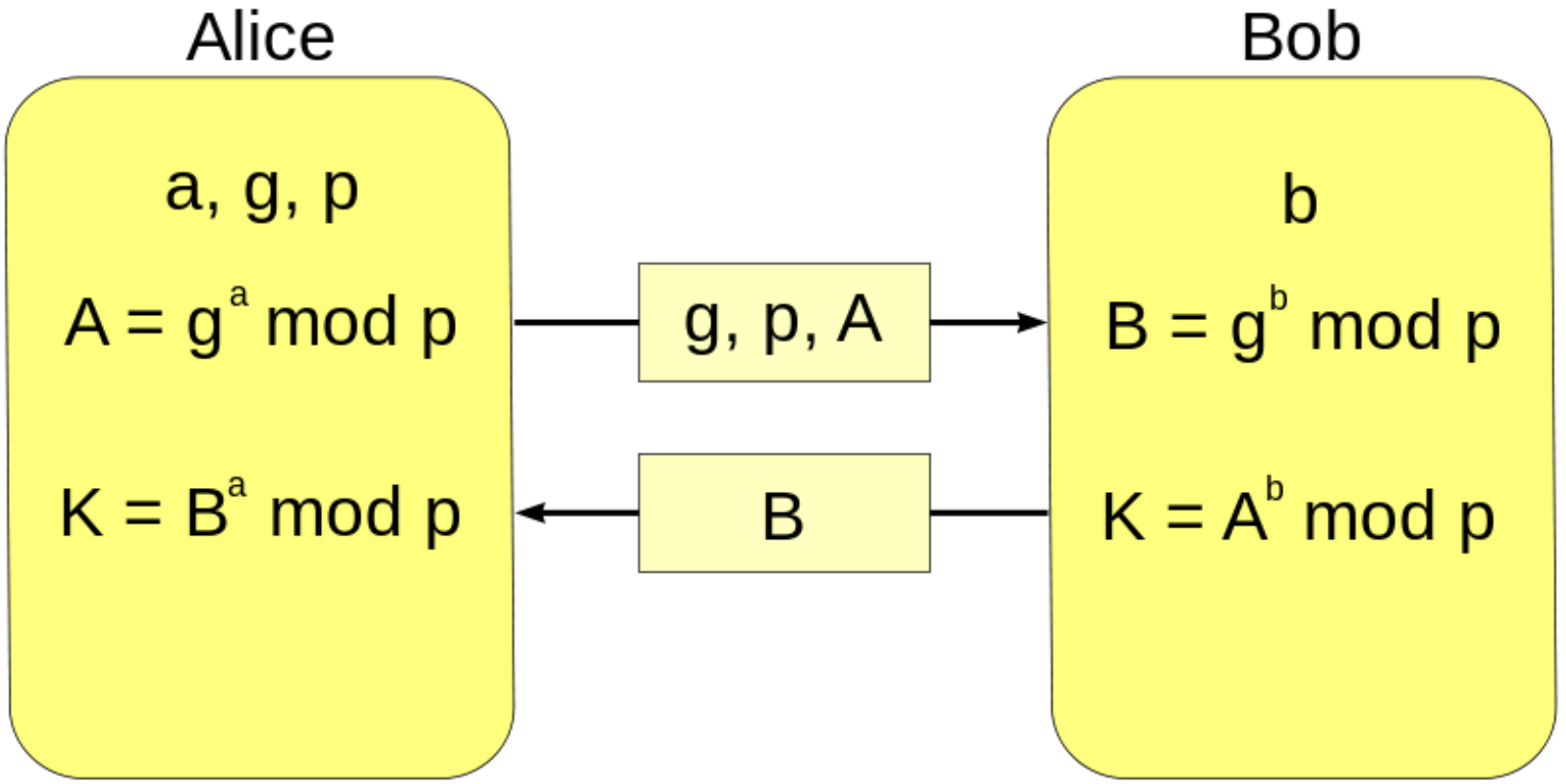
Die Kryptographie des nPA

- Der Ausweis verwendet RFID wegen Lebensdauer
  - Kryptographie ist daher notwendig um Abhören zu verhindern
- PACE (Password Authenticated Connection Establishment)
  - Sicherheit ist mathematisch nachgewiesen
  - Sicherheit v. Sitzungsschlüssel unabhängig von PIN
- Elliptische Kryptographie mit 256 Bits für die Schlüsselerzeugung

Terminologie

- Proximity Integrated Circuit Card (PICC) → kontaktloser Chip
- Proximity Coupling Device (PCD) → Kartenleser
- Public Key Infrastructure (PKI)
- Message Authentication Code (MAC) → Signatur
- Certificate Authority Reference (CAR) → Ausstellende CHR
- Certificate Holder Reference (CHR) → Zertifikatsname
- Country Verifying Certification Authority (CVCA)
- Document Verifying Cerfification Authority (DVCA)
- Certificate Holder Authority Template (CHAT)

Diffie Hellman Schlüsselaustausch



$K = A^b \bmod p = (g^a \bmod p)^b \bmod p = g^{ab} \bmod p = (g^b \bmod p)^a \bmod p = B^a \bmod p$

Diffie Hellman Schlüsselaustausch

- Partner einigen sich auf eine Primzahl p
- Und auf eine Primitivwurzel g modulo p mit 2≤ g ≤ p-2
  - Diese Parameter brauchen nicht geheim zu sein
- Beide Partner erzeugen jeweils eine Zufallszahl a bzw. b
  - A und b sind 1..p-2 und werden nicht übertragen
- Die Partner berechnen A und B und übertragen sie
- Die Partner berechnen nun jeweils K und verwenden K als Schlüssel

Das PACE Protokoll

- Der Chip wählt eine Zufallszahl „nonce“
  - Die Zufallszahl wird mit dem PIN verschlüsselt
  - Die verschlüsselte Zufallszahl wird übermittelt
  - Die Zufallszahl wird mit Hilfe der PIN entschlüsselt
- Chip und Terminal bilden *nonce* auf einen *Erzeuger* ab
  - Das erfolgt durch Mapping auf eine *Gruppe*
- Diffie Hellman mit *nonce* als Basis
- Chip und Terminal leiten Schlüssel ab
  - Ab hier ist die Verschlüsselung aktiv

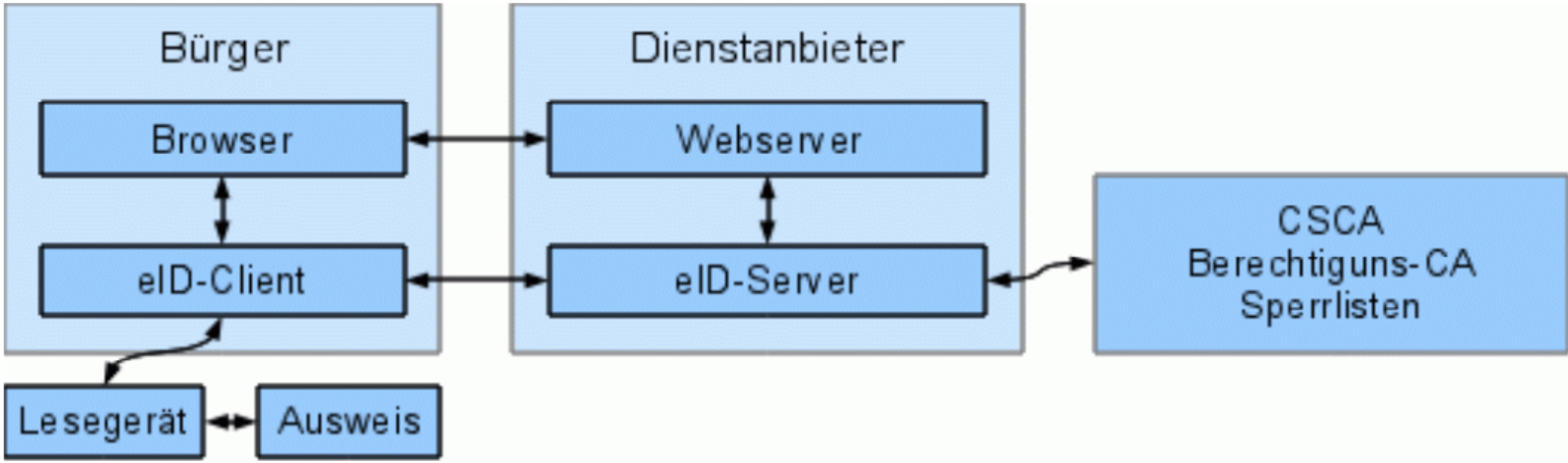
Terminal Authentifikation (TA)

- Das „Terminal“ muß sich dem Ausweis authentifizieren
- Ausweis überprüft Berechtigungen des Terminals
- Dazu wird dem Ausweis eine Zertifikatskette präsentiert
  - Die Kette beginnt nach dem root-Zertifikat
  - Die Kette endet mit dem Terminalzertifikat
  - Das Terminalzertifikat enthält die maximalen Rechte
  - Das Terminal muß die Kenntnis des geheimen Schlüssels des Terminalzertifikats beweisen

Chip Authentifikation (CA)

- Der Ausweis (Chip) muß Echtheit beweisen
- Terminal überprüft Echtheit des Dokuments
  - Challenge response
  - Chip muß geheimen Schlüssel kennen

eID Klient und eID Server



Das eID Protokoll

- Nutzer geht auf die Portal Webseite des Anbieters
- Nutzer klickt auf den Aktivierungslink zum Protokoll
  - http://127.0.0.1:24747/eID-Client?tcTokenURL=...
  - Damit wird der eID-Klient kontaktiert
- Der eID-Klient sendet einen HTTPS Auftrag mit TokenURL
- Die Antwort darauf sieht etwa so aus: ...

Das eID Protokoll …

<TCTokenType>

<ServerAddress>https://testpaos.governikus-eid.de:443/ecardpaos/paosreceiver</ServerAddress>

<SessionIdentifier>3b7b7479b91bf42796188b4217e0c5342eb255b8</SessionIdentifier>

<RefreshAddress>https://test.governikus-eid.de/gov_autent/async?refID=3b7b7479b91bf42796188b4217e0c5342eb255b8</RefreshAddress>

<Binding>urn:liberty:paos:2006-08</Binding>

<PathSecurity-Protocol>urn:ietf:rfc:4279</PathSecurity-Protocol>

<PathSecurity-Parameter>

<PSK>9996226EB4E334A16C783A2919F52798AEF3E6DB68A15AA697A9EEFC71BA2D716AAEF647CC58B0533F5287EEDBE8322318E825F0C88A15376E8DF0F55E-B827DF</PSK>

</PathSecurity-Parameter>

Das eID Protokoll …

- Der eID-Klient extrahiert daraus:
  - Die eID-Server Adresse
  - Den Preshared Key für die SSL Verbindung
  - Den Session Identifier
  - Die Refresh-Adresse
- Der Klient beginnt die Kommunikation mit dem Server
  - Verwendet wird dazu das PAOS Protokoll
- Der Server antwortet mit *Initialize Framework*
- Der Klient antwortet mit *GetNext Command*

Das eID-Protokoll …

- Der eID-Server schickt Zertifikate, das CHAT, Aux Data
  - Aux-Data: Altersverifikation + Ausweisgültigkeit
- Der eID-Klient präsentiert Zertifikatsbeschreibung, CHAT
  - Der Nutzer kann nun das CHAT reduzieren
  - Bei „dummem“ Leser erfolgt hier auch PIN-Eingabe
- Danach initiiert der eID-Klient PACE zum Ausweis
  - Die PACE Ergebnisse werden dem Server gemeldet
  - Ab hier ist die Kommunikation zum Ausweis ver-schlüsselt

Das eID-Protokoll …

- Der eID-Server verlangt nun Terminal Authentifikation
- Der eID-Klient initiiert die TA
- Der eID-Server verlangt Chip Authentifikation
- Der eID-Klient initiiert die CA
  - Ab hier End-zu-End Verschlüsselung Ausweis/eID-Server
- Der eID-Server sendet verschlüsselte APDUs
- Der eID-Klient leitet die APDUs an den Ausweis
- Der eID-Klient leitet die Antwort an den eID-Server

Das eID-Protokoll …

- Der eID-Server antwortet mit einer Bestätigung
- Nun antwortet der eID-Klient an den Browser
  - Die Daten enthalten die Refresh-URL
- Der Browser leitet an die Refresh-URL weiter
  - Damit wird das Portal informiert die Ergebnisse vom eID-Server abzuholen
  - Das Portal dechiffriert die verschlüsselten Daten vom eID-Server
- Unser Testportal zeigt uns nun die Daten

Sicherheitsprobleme beim nPA

- Der Ausweis und seine Verfahren sind sicher, aber:
- Unsicherer eID-Klient: Lücke im Updatemechanismus
- Unsichere PIN Eingabe
  - Entwenden der PIN durch Key-Logger
  - Mißbrauch der PIN, wenn Ausweis im Lesegerät
  - Standard- und Komfort-Lesegerät haben sichere PIN Eingabe

Fragen?

Danke!

Bildnachweis

- https://commons.wikimedia.org/wiki/File:Personalausweis_logo.svg
- http://britz-chorin-oderberg.de/img/nPA-front-t.jpg
- http://britz-chorin-oderberg.de/img/nPA-back-t.jpg
- http://www.tagesschau.de/multimedia/bilder/personalausweis120_v-gross16x9.jpg
- https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03127/BSI-TR-03127_en_pdf.html
- https://commons.wikimedia.org/wiki/File:Diffie-Hellman-Schlüsselaustausch.svg