Digital Health

Kap. 3: Fortgeschrittene Identity-Management Konzepte für Digital Health

Prof. Dr. Georgios Raptis

Identity Management im Digital Health Kontext

Repräsentation (Identität) einer physischen Person in der elektronischen Welt einer Digital Health Anwendung

- **Identifizierung** von Leistungserbringern (engl.: Health Care Provider/Professional, HPC), z.B. Ärzte, Apotheker usw.
- Identifizierung von Patienten / Versicherten / Bürger (je nach Skalierung der E-Health Anwendung)

- **Authentisierung** der o.g. Leistungserbringer / Patienten
  - Unterschiedliche Methoden / Sicherheitsniveaus möglich

- **Autorisierung** (Attribute, Rollen, Rechte, Privilegien)

- **Verwaltung** der elektronischen Identitäten (Lifecycle)

Wieso ist Identity Management für Digital Health wichtig?

Angriff: Identitätsdiebstahl

- Diebstahl einer Patientenidentität
  - Bruch der Vertraulichkeit für medizinische Daten des Patienten
  - Falsche Meldungen des Patienten an Ärzte

- Diebstahl einer Arzt-Identität
  - Bruch der Vertraulichkeit für medizinische Daten schlimmstenfalls von **allen** von ihn betreuten Patienten
    → **Impact des Angriffs** ist viel größer!
  - Manipulation von Patientendaten → Lebensgefährlich!
    - z.B. Ausstellung eines falschen eRezepts, Veränderung von Diagnosen usw.
  - Schädigung des (realen) Arztes und des Patienten (Verbindlichkeit)

Identifizierung vs. Authentifizierung, Autorisierung

Identifizierung: **Behauptung** einer Identität

Authentifizierung (Syn.: Authentisierung): **Nachweis** der Identität

- Wir haben ein mehr oder weniger sicheres Identity Management, je nach dem
  - wie sicher dieser Nachweis erbracht wird
  - wie stark die elektronische Identität an die Person gebunden ist

Autorisierung: **Rechte**verwaltung (nachdem Identität nachgewiesen wurde): Was darf ich machen?

- Berufsgruppenbasiert (*attribute based access control*), z.B. alle Ärzte dürfen Notfalldaten lesen
- Individuell, z.B. Dr. Maier darf auf die ePatientenakte von Hr. Huber zugreifen
- Rollenbasiert (*role based access control*): z.B. diensthabender Stationsarzt hat Zugriff auf Daten der Patienten auf Station
- Kombinationen verschiedener Modelle: z.B. Patient Meier hat Dr. Huber autorisiert (→ individuell), auf Dokumente des Typs „Impfpass“ zuzugreifen (Voraussetzung: Dr. Huber ist Arzt → Attribut)

Grundsätzliche Methoden der Authentifizierung

Identität wird über **Wissen**, d.h. ein Geheimnis nachgewiesen

- Username / Passwort

Identität wird über **Besitz** nachgewiesen

- Z.B. Besitz eines Mobiltelefons, eines speziellen USB-Tokens, einer Chipkarte
- Meist als **Zwei-Faktor Authentifizierung** (Besitz UND Wissen oder Besitz UND Biometrie)
- Kryptographische Authentifizierung mit Hilfe einer Chipkarte
  → Besitz des Schlüssels auf der Chipkarte & Wissen der PIN der Karte

Identität wird über ein **körperliches Merkmal** nachgewiesen

- Biometrie
- Starke Bindung an Person
- Schon jetzt in Verbindung mit Mobiltelefonen (Fingerprint, FaceID)
- Mit einigen Problemen behaftet
  - In Zukunft m.E. große Verbreitung!

Authentifizierung mit Username / Passwort

- Geheimnis wird zum Authentifizierer übertragen
  - Wird dort – schlimmstenfalls unsicher (Klartext) – gespeichert
- Viele nutzen das gleiche Passwort für verschiedene Dienste
  - Chance eines erfolgreichen Angriffs ist groß
- Grundsätzliches Problem: Übertragung des Geheimnisses → kann gestohlen werden
  - Wie besser machen?
  - Man weist nach, dass man das Geheimnis kennt, ohne es zu übertragen!
  - Mit Geheimnis rechnen, Ergebnis übertragen, Server prüft Ergebnis (wie?)

Authentifizierung mit Username / Passwort

Für einen effektiven Schutz von Patientendaten nicht angemessen

- Sollte für die Authentifizierung von Patienten in einer E-Health Anwendung vermieden werden
- D.h. nur in Ausnahmefällen, gut begründet
- Sollte für die Authentifizierung von Ärzten nicht eingesetzt werden

Authentifizierung mit Username / Passwort

→ Üblich in der US-Amerikanischen eHealth Infrastruktur für den Zugriff der Patient*innen

Grund: Voraussetzungen für staatliche Förderung der Ärzte / Krankenhäuser umfassen nicht die **Stärke** der Authentifizierung. Sehr wohl aber die häufige Nutzung der Dienste

- Technische Hürden so weit wie möglich reduzieren, damit möglichst viele Patient*innen teilnehmen
- Offenbar wird keine Pflichtverletzung den Ärzten / Krankenhäusern beim Kompromittieren von Passwörtern vorgeworfen. Hacker werden dagegen mit drakonischen Strafen belegt
    - Eher rechtlicher als technischer Schutz

2-Faktor Authentifizierung

Üblich heutzutage: Username/Passwort + ein weiteres Geheimnis

- SMS mit Einmalpasswort oder Push-Nachricht auf App mit Einmalpasswort
- Security Token auf Smartphone, z.B. Google Authenticator, AppleID
- Security Token als spezialisiertes Gerät, generiert Einmalpasswort

Warum ist das sicherer?

- Zusätzlich zum Passwort (Wissen) muss man irgendetwas besitzen (Handy, Smartphone, Security Token)

Ist das praktisch, praktikabel?

- Ja, kann leicht in einer E-Health Anwendung integriert werden
- Fast jeder (Arzt oder Patient) hat ein Handy / Smartphone

Relevante Standards

- TOTP (ältestes Protokoll, zeit-/geheimnisbasiert, z.B. Google Authenticator)
- U2F (Universal 2nd Factor, wird von gängigen Browsern unterstützt, wurde abgelöst durch:)
- **FIDO2** (WebAuthn & CTAP Protokolle, aktuell der modernste Standard in diesem Bereich)

ELGA: Zugriff über 2-Faktor Authentifizierung

Zugriff des Patienten (allein): Bürger-Karte oder Handy-Signatur am ELGA-Portal

- Bürger-Karte: Authentifizierung mit Chipkarte & PIN im Portal

- Handy-Signatur:
  - Vorhandene eGovernment ID-Management Infrastruktur in Österreich
  - Handy-Nr. & „Signatur-Passwort“ auf Portal eingeben
  - „Vergleichswert“ (Zeichenkette, Zuordnung der Authentifizierung) wird angezeigt
    - Entweder es wird eine TAN als SMS geschickt → in Portal eingeben
    - Oder man kann mit Hilfe einer App („speed-sign“) die Authentifizierung auslösen

ELGA: Zugriff über 2-Faktor Authentifizierung

Zugriff des Arztes:

- Stecken der e-card des Patienten in der Praxis (ohne PIN) → Authentifizierung in ELGA

UND

- Authentifizierung des Arztes per „Ordinationskarte“ in die ELGA

→ Zugriff auf Patientenakte wird für 28 Tage für den Arzt freigeschaltet

→ Der/Die Patient*in kann aber Arzt sperren oder nachträglich den Zeitraum verändern

Grundsätzlich in Österreich: Opt-out für Patient*innen (ca. 3%)

- Jede/r hat automatisch eine ePatientenakte, es sei denn er/sie widerspricht.

EPD: Zugriff über 2-Faktor Authentifizierung

Keine festgelegte „Identifikationsmittel“, müssen jedoch ein Sicherheitsniveau mit „Hohem Vertrauen“ (Assurance Level 3 nach ISO/IEC29115:2013) erfüllen

- z.B. USB-Stick, Mobiltelefon, Chipkarte. eID wird nach persönlicher Identifizierung mit amtlichem Dokument erstellt und ausgegeben (mit Person verknüpft)

EPD: Rechtemanagement und Speicherung

Rechtemanagement nach einem (modifizierten) *Multilevel-Security* Zugriffsmodell

- Patient*in vergibt Vertraulichkeitsstufe an Dokumente (Classification)
- Patient*in vergibt Zugriffsstufe an „Gesundheitsfachpersonen“ (Clearance)
- Arzt kann auf Dokumente zugreifen, die eine niedrigere oder gleiche Vertraulichkeitsstufe haben, als seine Zugriffsstufe (Classification <= Clearance)
- Notfallzugriff ohne explizite Rechtevergabe, kann aber allg. abgeschaltet werden

EPD: Opt-in für Bürger. Dokumente werden dezentral gespeichert, dort wo sie entstehen. Suche über zentrale Registry mit Metadaten der Dokumente

Wie baut man Authentifizierung in einer E-Health Anwendung ein?

- Selber machen in der Anwendung → isolierte Identität. Scope ist nur ein Dienst

- Integration eines Identity Providers
  - Ein spezialisierter Dienst, der Authentifizierungen durchführt und diese gegenüber „Service-Provider“ bestätigt
  - Mehrere Methoden und Sicherheitsniveaus möglich
  - Nach erfolgter Authentifizierung bestätigt der Identity Provider die Identität mit Hilfe einer Assertion (z.B. SAML)
  - Single Sign On Konzepte realisierbar
  - **Empfehlung**, falls ID-Provider verfügbar, sicher, kostengünstig

  - Heutzutage empfehlenswert: **openID Connect**, basiert auf OAuth2.0

Challenge-Response Verfahren zur Authentifizierung

Authentifizierung mit Passwort, TANs, Codes usw.

- Geheimnisse werden vom Client des Nutzers an einem Server übertragen und dort verglichen. Können beim Client, unterwegs oder beim Server gestohlen werden.
- Dadurch sicherheitstechnisch schwach

Wie kann man es besser machen?

- → Nachweis, dass man das Geheimnis besitzt, ohne es zu übertragen

Authentifizierung über Challenge-Response Verfahren, z.B. mit Schlüssel auf Chipkarte

- Challenge-Response Verfahren: Server schickt „Challenge“ (Zufallszahl) an Client
- Erstellung einer elektronischen Signatur auf die Challenge mit Hilfe eines auf der Chipkarte gespeicherten kryptographischen privaten Schlüssels → Ergebnis ist die „Response“
- Der Schlüssel selbst wird NICHT übertragen. Damit wird nur gerechnet → das Ergebnis (Response) produziert und zum Server übertragen. Ein Angreifer kann aus der Response den Schlüssel nicht berechnen.
- Das Ergebnis wird serverseitig mit Hilfe eines elektronischen Zertifikats (X.509-Standard) überprüft.

Zertifikat (X.509): bestätigt **Zuordnung eines öffentlichen Schlüssels zu einer Person**

- Kann auch Attribute der Person enthalten (z.B. „Ärztin/Arzt“)
- Online Prüfung des Zertifikats auf Sperrung
  - Verzeichnisdienst → Sperrliste (CRL) oder
  - **OCSP-Responder** (OCSP=Online Certificate Status Protocol)
- Zertifikate können auch ohne Chipkarte eingesetzt werden
  - Gute Sicherheit: priv. Schlüssel werden in security chips eines Smartphones oder z.B. in einem FIDO2 USB-Stick gespeichert
  - Schwache Sicherheit: private Schlüssel sind auf die Festplatte des Rechner oder Flash-Speicher des Smartphones gespeichert, können gestohlen werden

Methoden zur Aktivierung der Chipkarte (→ Setzen eines *Security Condition* in der Karte)

Authentifizierung des Benutzers oder einer anderen Instanz gegenüber der Chipkarte, z.B. damit Daten ausgelesen oder ein Kryptoschlüssel verwendet werden kann:

- üblicherweise **mit PIN**
  → Ergibt dann: Wissen (PIN) und Besitz (Chipkarte)

- **kryptographische (C2C) Authentifizierung** einer anderen Instanz, z.B. Aktivierung der eGK durch Authentifizierung eines HBA / SMC, allein oder ggf. zusätzlich zur eGK-PIN

Hintergrund: Jedes Informationsobjekt (z.B. Schlüssel, Datei) hat eine *Access Control List* mit „Security Conditions“ für jeden Kartenbefehl. Sie müssen erfüllt werden, um den Befehl auszuführen

- z.B. READ BINARY auf Informationsobjekt EF.NFD, oder
- z.B. INTERNAL AUTHENTICATE auf privaten Schlüssel für Authentifizierung

Biometrische Authentifizierung

- Sehr starke Bindung zur Person
- Authentifizierung nicht deterministisch → probabilistisch
  - False Accept Rate, False Reject Rate → Balance finden
- Einschränkungen in der Widerstandsfähigkeit gegen Angriffe
- Zertifizierung der Sicherheit schwierig
- Technische und auch organisatorische Umgebungsbedingungen sind entscheidend
- Bei Smartphones ist Biometrie inzwischen etabliert
  - Jedoch nie allein, sondern in Verbindung mit weiteren Maßnahmen (regelmäßige PIN-Abfragen, PIN-Abfrage nach x Fehlversuchen)

Zertifikatsbasiertes Identity Management

- Zuverlässige physische Identifizierung der Person
  - Für eGK: Meist bei Ausgabe des PIN-Briefs
- Registrierung der Person
  - HBA: Antragstellung, Bestätigung des Berufsgruppen-Attributs
  - eGK: Daten der Krankenkasse
- Sichere Schlüsselerzeugung
  - Auf der Chipkarte selbst (HBA)
  - In einem Hardware Security Modul (HSM) → eGK
- Zertifikatserstellung durch vertrauenswürdige Instanz
  - Das Zertifikat wird durch den Aussteller elektronisch unterschrieben
  - Eine E-Health Anwendung / Infrastruktur muss den Aussteller vertrauen
  - Z.B. Speicherung der CA-Zertifikate in einer Trust-Service Status List (TSL)

Zertifikatsbasiertes Identity Management

- Bedruckung der Chipkarte
- Versand von Chipkarte und PIN-Brief, ggf. mit besonderen Sicherheitsmaßnahmen
- Bestätigung des Empfangs
  → Freischaltung der Zertifikate im Verzeichnisdienst

- Ggf. Sperrung des Zertifikats bei Bedarf
- Ablauf der Gültigkeit des Zertifikats

- Abgelaufene/gesperrte Signaturzertifikate bleiben dann bis zu 30 Jahre prüfbar beim OCSP-Responder
  - Warum 30 Jahre? Warum nur Signaturzertifikate?

Identity Management, Fortsetzung

- Registrierung der eID in der E-Health Anwendung
- Verknüpfung der eID mit Rechten / Privilegien
  - z.B. ein Patient berechtigt seinen Hausarzt, auch in Abwesenheit der eGK auf seine Patientenakte zuzugreifen
- Verwaltung der eID in der Anwendung
  - Was ist, wenn die Karte (=Schlüssel/Zertifikat) ausgetauscht wird?
    - → Rechte-Erhalt erforderlich

  - In der Telematik-Infrastruktur, für den HBA: Telematik-ID, wird von der Kammer ausgestellt und bestätigt, die die Ausstellung des HBA freigegeben hat (→ den Arzt kennt). Für die eGK: Krankenversichertennummer (KVNR)

In der Regel wird eID für eine Zugangskontrolle in einer Anwendung verwendet

Klassisches Konzept für Zugriffskontrolle

- Access Control List (ACL), assoziiert mit Informationsobjekt
- Einträge mit ID des Berechtigten & Rechte (Privilegien)

- Alternative Zugriffskontrollsysteme, z.B. Mandatory Access Control (MAC), Role based AC (RBAC), Attribute based access control (ABAC), Multilevel Security usw.

- Nach Authentifizierung entscheidet das Rechtemanagement-System anhand bestimmter Regeln (z.B. Eintrag in der ACL und Rolle), ob man Zugriff zum Informationsobjekt erhält
  - Täuschen / Umgehen des Systems („hacken“)
    → unberechtigter Zugriff auf Information

- Krypto-Schlüssel in Chipkarte ist nur Authentifizierungsmittel!

Kryptographisch unterstützte Zugriffskontrolle

Zugriff auf die Daten nur möglich, wenn man einen zugehörigen Kryptoschlüssel hat

- Zugriff wird nicht (nur) durch die Entscheidung eines Servers nach Auswertung der ACL gewährt
- Sondern → Verschlüsselung der Daten, Schlüssel im Besitz bzw. unter Kontrolle der berechtigten Personen
  - Übliches Konzept: Hybridverschlüsselung, symmetrischer Schlüssel wird verschlüsselt mit asymmetrischen Public Keys aller berechtigten Personen
  - → Private Key eines Berechtigten muss symm. Schlüssel → Dokument entschlüsseln
- Vorteil: Zugriffskontrolle in Software kann grundsätzlich gehackt werden (Softwarefehler → Schwachstelle). Kryptoverfahren (Mathematik) zu hacken ist ein ganz anderes Kaliber
  - Es ist also ein sehr wirksamer zusätzlicher Schutz gegen Angriffe mit Umgehen der Zugriffskontrolle, kann natürlich aber auch keine 100% Sicherheit garantieren. Mögliche Angriffe sind i.d.R. aufwändiger
- Weitere Konzepte, hier nicht relevant: ***Secret Sharing*** *m* von *n* Kryptoalgorithmen (z.B. nach Shamir): Schlüssel wird in *n* Teilen aufgeteilt. Rekonstruktion, wenn mindestens *m* Teile zusammenkommen

Kryptographisch unterstützte Zugriffskontrolle

Beispiel für eine kryptographisch unterstützte Zugriffskontrolle:

Altes Zugriffskontroll-Konzept der Telematik-Infrastruktur (wird nicht implementiert)

- Alle online gespeicherten Patientendaten sollten mit dem öffentlichen Schlüssel der eGK des Patienten und der HBAs / SMCs der von ihm berechtigten Ärzten verschlüsselt werden
- D.h. sollte die Zugangskontrolle der TI gehackt werden, hätte man nur verschlüsselte Daten gefunden
- Man hätte WIRKLICH den Schlüssel haben müssen, um an Daten heranzukommen
  - Dieser ist in der Chipkarte gespeichert und nicht auslesbar sondern nur nutzbar (nach PIN-Eingabe)

Kryptographisch unterstützte Zugriffskontrolle nicht zwingend notwendig, aber besonders sicher

Anmerkung: Man muss nicht unbedingt eine kryptographisch unterstützte Zugriffskontrolle implementieren

- eHealth-Infrastrukturen anderer Länder haben sie i.d.R. auch nicht, inzwischen auch die TI nicht
  - Nachteile einer Krypto-Zugriffskontrolle: erhöhte Komplexität, Schlüsselmanagement
- Man läuft dann jedoch Gefahr, bei Cyberangriffen (leichter) gehackt zu werden
  - s. z.B. Norwegen 2018: ePatientenakten von 3 Millionen Norweger gehackt, vermutlich ein östlicher Geheimdienst mit Hilfe von Insidern
  - Bruch der Vertraulichkeit → Schwerwiegende Konsequenzen für Patienten und für das System (kein Vertrauen mehr)
  - Bruch der Integrität / Authentizität / Verbindlichkeit → Kein Vertrauen mehr an die Daten

Beispiel 1: Ticket-Konzept

- Rechtemanagement auf Dokumentenebene
  - **Policy** auf Ebene von Dokumentenklassen (alle Dokumente einer Anwendung)
  - Patient kann aber Rechte für einzelne Dokumente individuell festlegen

Ursprünglich 2005 konzipiert von Fraunhofer FOKUS/SIT für die TI.

Wird in der TI nun so nicht realisiert, ist aber ein gutes Beispiel für eine mögliche kryptographisch unterstützte Zugriffskontrolle.

Beispiel 1: Ticket-Konzept

- Informationsobjekt wird hybrid verschlüsselt
  - Symmetrische Verschlüsselung, z.B. mit AES256, mit einem zufälligen Dokumentenschlüssel
  - Asymmetrische Verschlüsselung mit dem öffentlichen Schlüssel eines Berechtigten

- Access Control List (ACL, „Objekt-Ticket“), assoziiert mit Informationsobjekt
  - Einträge mit ID des Berechtigten & Rechte (Privilegien)
  - symmetrischer Dokumentenschlüssel, asymmetrisch verschlüsselt mit dem öffentlichen Schlüssel von jeden Berechtigten

Beispiel 1: Ticket-Konzept

- Objekt-Ticket: mit Informationsobjekt assoziierte Access Control List (ACL), enthält alle Zugriffsberechtigten für das Info-Objekt, die erlaubten Zugriffe und die Kryptoschlüssel

- Service-Ticket: „Template“ einer ACL für eine Informationsobjekt-Klasse eines Patienten mit allen Zugriffsberechtigten, die erlaubten Zugriffe und die Kryptoschlüssel
  - Bei jedem neuen Informationsobjekt dieser Klasse werden Objekt-Tickets nach dem Muster des Service-Tickets erstellt

Beispiel 1: Ticket-Konzept, Zugriff auf Information

- Authentifizierung eines Berechtigten (z.B. eGK des Patienten)
- Rechtemanagement-System entscheidet anhand der ACL und gibt Zugriff auf Ticket und verschlüsseltem Informationsobjekt frei
- Datenübertragung z.B. zum Konnektor der Arztpraxis

- Entschlüsselung des verschlüsselten Dokumentenschlüssels aus Ticket mit dem privaten Schlüssel der eGK
- Entschlüsselung des Dokuments mit dem gerade gewonnenen symmetrischen Dokumentenschlüssel

Beispiel 1: Ticket-Konzept, Rechtemanagement

Aufgabe: Dauer-Berechtigung eines Arztes z.B. zum Zugriff auf Pflegeakte des Patienten

- Erstellung eines **Service-Tickets**: enthält ID und öffentlichen Schlüssel des Arztes
- Signatur des Service-Tickets durch eGK des Patienten

- Bei jedem neuen Pflegeakte-Dokument wird anhand des Service-Tickets im zugehörigen Objekt-Ticket
  - die ID des berechtigten Arztes aufgenommen
  - der Dokumentenschlüssel mit dem öffentlichen Schlüssel des Arztes verschlüsselt und im Ticket aufgenommen

Beispiel 1: Ticket-Konzept, Rechtemanagement

Aufgabe: Dauer-Berechtigung eines Arztes zum Zugriff auf bereits vorhandene Pflegeakte-Dokumente des Patienten

- Alle Tickets vom Typ „Pflegeakte-Dokument“ des Patienten werden (nach Authentifizierung der eGK) lokal geholt
- Die verschlüsselten Dokumentenschlüssel werden mit dem privaten eGK-Schlüssel entschlüsselt
- wieder zusätzlich mit dem öffentlichen Schlüssel des Arztes verschlüsselt
- die Tickets werden mit neuen Einträgen für den berechtigten Arzt ergänzt

Wie schützt man dabei die entschlüsselten Schlüssel am besten?

- Indem o.g. Operationen innerhalb eines *Hardware Security Moduls* (HSM) durchgeführt werden

Beispiel 2: Anwendungsschlüssel

- Rechtemanagement auf Anwendungsebene
  - Rechte für alle Dokumente einer Anwendung (z.B. elektronische Patientenakte)
  - Patient kann Arzt/Krankenhaus für die gesamte Akte autorisieren
    - Alles oder nichts
    - Schön einfach, aber ziemlich grobgranular

Beispiel 2: Anwendungsschlüssel

- Dokument wird mit einem individuellen symmetrischen Schlüssel verschlüsselt
- Alle symmetrischen Dokumentenschlüssel der Anwendung werden mit einem Anwendungsschlüssel (pro Patient) symmetrisch verschlüsselt
- Der Anwendungsschlüssel wird mit den jeweiligen asymmetrischen öffentlichen Schlüssel der Berechtigten verschlüsselt
  - Zugriffsmuster: priv. Schlüssel z.B. auf HBA kann Anwendungsschlüssel entschlüsseln. Mit Anwendungsschlüssel können Dokumentenschlüssel entschlüsselt werden. Damit können die Dokumente entschlüsselt werden
  - Berechtigter erhält dann Zugriff auf alle Dokumente der Anwendung

Beispiel 3: Anwendungsschlüssel kombiniert mit Multilevel-Security Elementen

- Rechtemanagement auf Anwendungsebene plus klassische Zugriffskontrolle mit Multilevel Security Elementen
  - Dokumente werden vom Patienten klassifiziert (z.B. normal, geheim, streng geheim)
  - Berechtigte erhalten vom Patienten Freigabe (engl. „clearance“ z.B. bis normal, bis geheim, bis streng geheim)
    - Freigabe und Klassifikation sind Labels auf die jeweiligen Informationsobjekte (Dokumente und Personen/Institutionen)

Beispiel 3: Anwendungsschlüssel kombiniert mit Multilevel-Security Elementen

- Kryptographische Unterstützung wie beim Anwendungsschlüssel-Konzept
- Rechtemanagement-System prüft jedoch zusätzlich (klassische Zugriffskontrolle in Software), ob Freigabe des Berechtigten gleich oder höher als Klassifizierung des Dokumentes ist.
  - Nur dann wird das Dokument herausgerückt
  - Konzept einfach, relativ feingranular
  - Wird in der Schweiz für das Elektronische Patientendossier (Akte) implementiert

Nachteile der kryptographisch unterstützten Zugriffskontrolle

Kompliziertes Rechtemanagement

- Chipkarte o.ä. ist nicht nur Authentisierungsmittel, enthält notwendigen Schlüssel zum Zugriff auf die Daten
- Zusätzlich zur Software-Entscheidung der Zugriffskontrolle: viele Krypto-Operationen
- Rechteverwaltung braucht viel mehr, als nur Anpassung des Eintrags in der ACL
  - Schlüssel gewinnen (entschlüsseln), für neuen Berechtigten bereitstellen

Kompliziertes Schlüsselmanagement

- Wenn der Schlüssel weg ist, sind die Daten unzugänglich
- Aufwändige Konzepte, um dies zu verhindern, ohne die Sicherheit zu kompromittieren (Datenerhalt-Konzepte, datenschutzfreundliche (!) „key escrow“ Konzepte)
  - Ohne dass jemand einen Generalschlüssel erhält
  - Ohne Speicherung des privaten Schlüssels
  - Am Besten mit aktiver Beteiligung des Versicherten
  - Konzepte müssen auch nach Kartenausgabe implementierbar sein

Ein mögliches Datenerhalt-Konzept

- Erforderlich ist eine „Umschlüsselung“ aller zentral gespeicherten Gesundheitsdaten eines Versicherten, wenn die eGK ausgetauscht wird
- Wie macht man dies, wenn die eGK schon weg ist? (z.B. verloren)
- Datenerhalt-Konzept passend zu Beispiel 1 (Ticketkonzept)
  - Konzept **wird so nicht in der TI implementiert**,
    dient nur als **Beispiel** für Angewandte Kryptographie in E-Health

Ein mögliches Datenerhalt-Konzept

- Patient schließt Vertrag mit Datenerhalt-Anbieter ab

- Anbieter erzeugt in HSM einen für den Patienten individuellen Schlüsselpaar
  - Privater Schlüssel wird noch im HSM mit öffentlichem HSM-Schlüssel verschlüsselt
    - Bindung an HSM, Umschlüsselung kann später nur im HSM erfolgen

- Privater Schlüssel wird dann noch im HSM mit einem symmetrischen „Aktivierungsschlüssel“ verschlüsselt

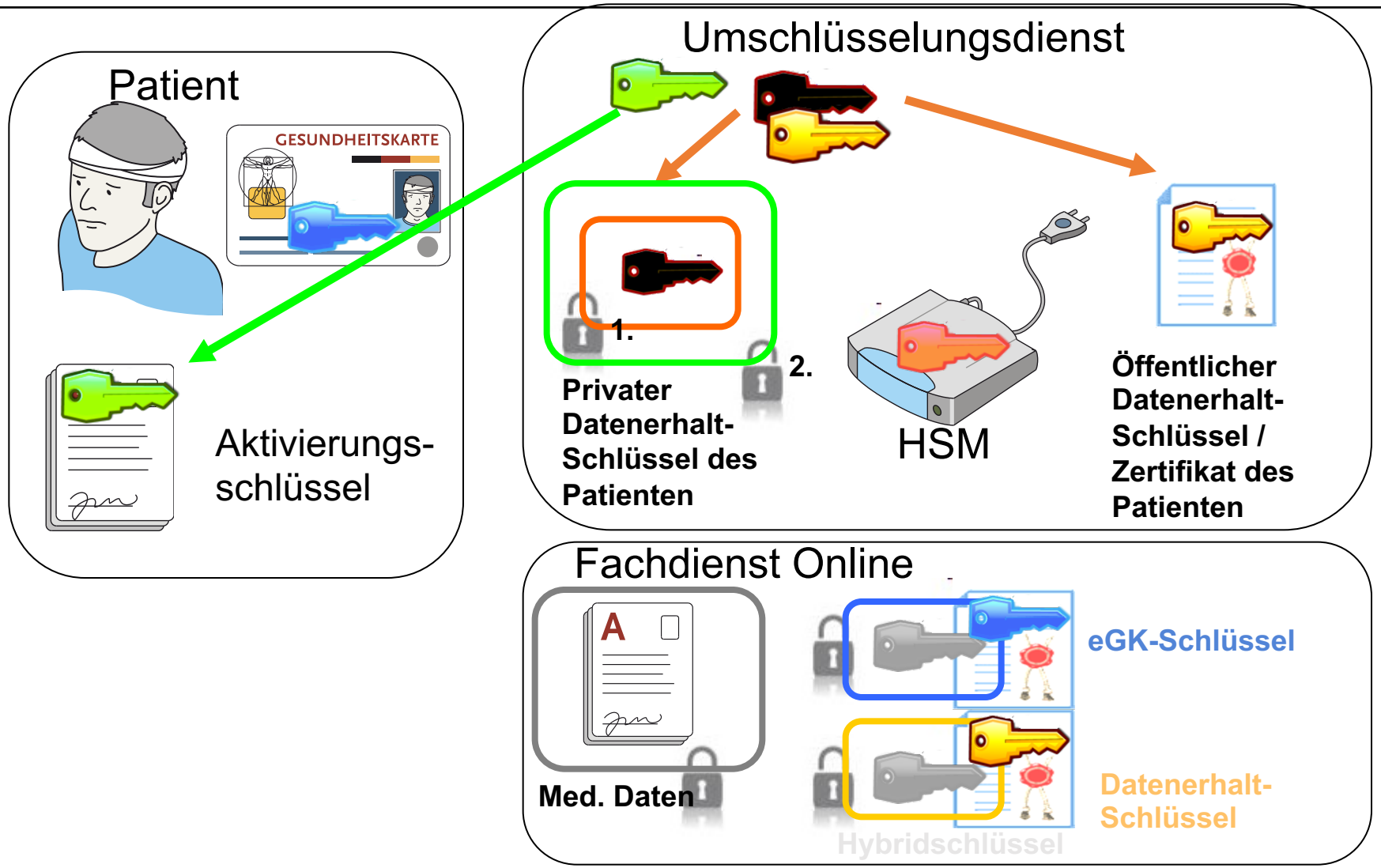
Ein mögliches Datenerhalt-Konzept

- Der nun doppelt verschlüsselte private Datenerhalt-Schlüssel wird vom HSM exportiert und vom Anbieter archiviert
- Der Aktivierungsschlüssel wird ausgedruckt und dem Patienten als PIN-Brief zugestellt
- Für den öffentlichen Schlüssel wird ein Zertifikat erzeugt
  - mit den persönlichen Daten des Patienten
  - und bei allen Fachdiensten für eGK-Anwendungen als Datenerhalt-Zertifikat autorisiert

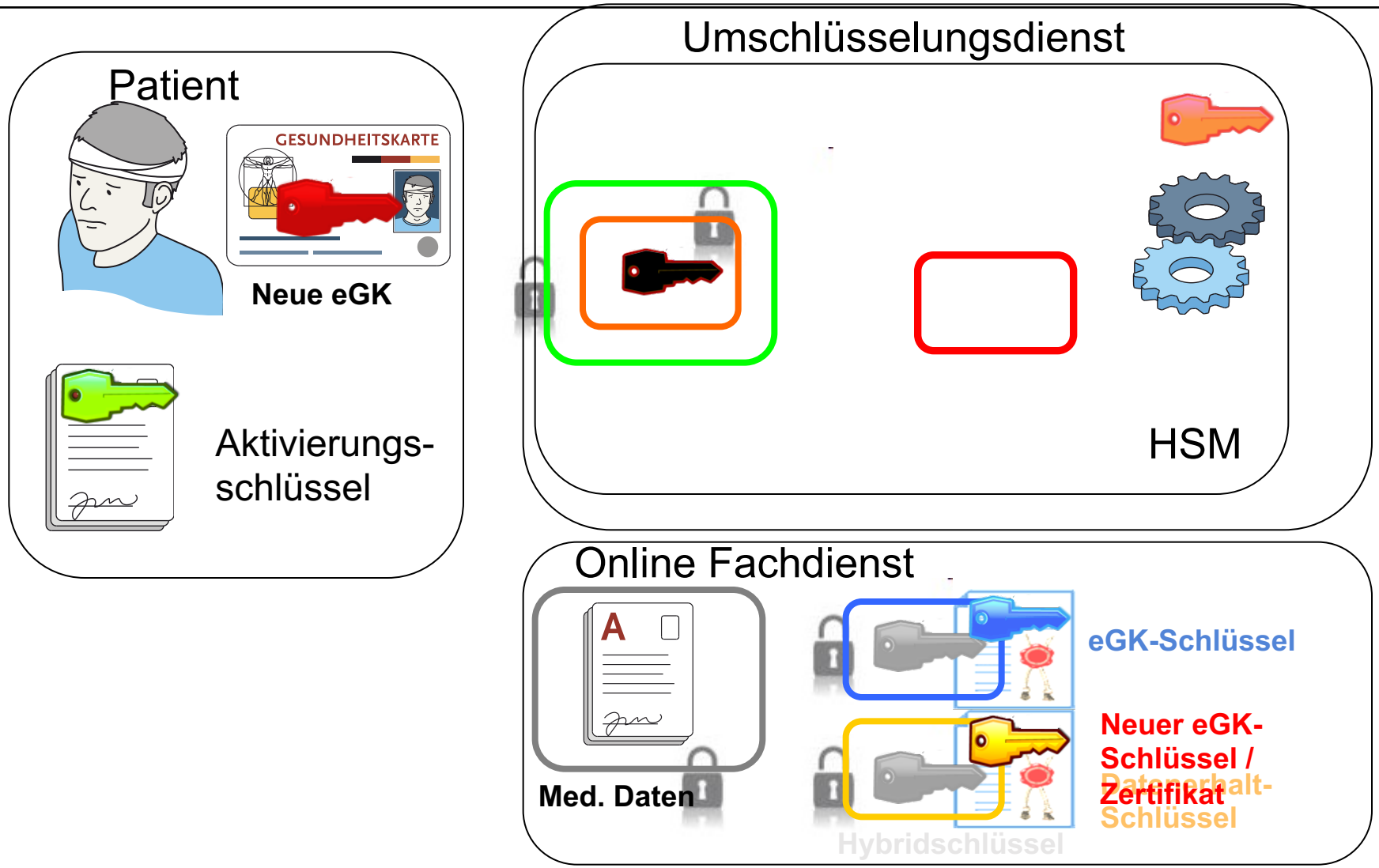
Ein mögliches Datenerhalt-Konzept

- Datenerhalt-Zertifikat wird nun als ein weiterer „Berechtigter“ in den Tickets der verschlüsselten Datensätze geführt
  - So wie z.B. der Hausarzt des Patienten dauerhaft für einen Zugriff autorisiert wird, kann auch der Datenerhalt-Dienst autorisiert werden
    - jedoch mit einem individuellen Schlüssel für jeden Patienten (Kritikalität des Schlüssels reduziert)
    - mit weitaus weniger Rechten
      → Kein Zugriff auf Daten, nur auf Schlüssel, zwecks Umschlüsselung!
- Gesundheitsdaten werden also hybrid mit dem eGK-Schlüssel und zusätzlich mit dem Datenerhalt-Schlüssel verschlüsselt

Ein mögliches Datenerhalt-Konzept
Initialisierung



Ein mögliches Datenerhalt-Konzept, Umschlüsselung



eID und mHealth

- Eine Chipkarte wird gerne benutzt, weil sie einen sicheren Schlüsselspeicher bereitstellt
- Smartphones haben ebenfalls Sicherheitskomponenten
  - Apple: Secure Enclave
  - Android: Titan M/M2 Chip / StrongBox Keymaster
    - Sogar mit *key attestation*: über ein Google-Zertifikat kann nachgewiesen werden, dass der Schlüssel innerhalb des Chips sicher generiert wurde und es somit nie verlassen kann
    - Bisher nicht so hoch zertifiziert, wie Chipkarten

- Entweder kann eine kontaktlose Chipkarte mit einem Smartphone als Kartenterminal verwendet werden
  - in Deutschland: Heilberufsausweis ab Generation 2, eGK ab Dezember 2019

- Oder das Smartphone selbst als Sicherheitskomponente (Schlüsselspeicher) verwendet werden

Im Digitale Versorgung und Pflege Modernisierungs-Gesetz (DVPMG): Digitale Identitäten ab 2023

Mögliche Implementierung

- Starke Authentifizierung gegenüber einem Identity Provider
  - Mögliche Authentifizierungsmittel: Smartphone, USB FIDO2 Token, interne eIDs von Krankenhäusern bei entsprechend hohem Sicherheitsniveau
- Identity Provider stellt (langfristiges) refreshToken aus
- Mit refreshToken: Identity Provider stellt kurzlebiges accessToken für einen bestimmten Dienst aus

- Damit kann ein Client (z.B. eine App im Smartphone oder Rechner) einen Dienst nutzen

- Vorteile: Verwaltung der Identitäten wird im Identity Provider gemacht, flexibler als (statische) X.509 Zertifikate (falls es so technisch realisiert wird)